Automated Machine Learning: An Agile Approach to AI Deployment

An Overview of AutoML and Available Technology

Part 1: Overview – Machine Learning & AutoML

Part 2: AutoML Tool Evaluation Summary

Gene Ferruzza
Sr. Manager Data Science, Valassis Digital

Manini Madireddy, Ph.D.
Director Data Science, Targetbase

Arnav Sharma
Senior Data Scientist, Targetbase

Special thanks to Dimitris Tsioutsias Ph.D.

July 24, 2018

Disclaimer: This paper is funded by Targetbase and is the result of the analysis of a small number of vendors in the marketplace; it is not intended as an analysis of all possible vendors. Additionally, the research was done while the authors worked for Targetbase and the opinions presented in this paper reflect only Targetbase's evaluation of the product during the study period.

Contents

Introduction

The rising need for applied data science and the shortage of data scientists in nearly every business is rapidly growing the automated machine learning (AutoML) space. The number of AutoML software tools has grown by 300% in just two years. During this time there have been a wide array of definitions, expectations and skepticism about automated data science tools, and how they might change time proven approaches to model development and deployment. Understandably there is concern about changes to data science processes, particularly those that have been directly or indirectly driving business decisions across industries for years.

AutoML to date has been focused on replicating the human "know how" required to build models, a primary deliverable in most data science projects. Based on this study, it turns out each step in the model building process can be replicated. Does it do as good of a job as a data scientist? The answer is for what it does, yes. Will AutoML replace a data scientist? Likely never, but AutoML is packaging artificial intelligence (AI) into data science tools that will become invaluable to the data scientist in the future.

Application development in the early days of AI began with enthusiasm for new data-driven business applications. The excitement was dampened as comparatively low compute power, scarce data or access to it drove down feasibility and confidence in AI. Today, AI technology is challenging our status quo in nearly every aspect of how we do things, still often met with skepticism. But it is just at that point, when we draw the line at putting AI technology at the wheel of what we feel humans do best, we witness a successful driverless car.

The historical evolution of machine learning, its place in AI, and its evolution into the big data era has escalated interest in new machine learning applications. Machine learning is a component of AI, but has also stood on its own for decades as the key method for deriving basic intelligence from vast amounts of data. Interestingly, predicting things like weather, loan risk, human behavior, economic trends or molecular reactions have not been classified as AI, but all use machine learning.

Machine learning has been evolving since the advent of computing and has landed in a profound place in today's business and scientific operations. Converting data into intelligence is not only readily feasible, it also has wide scale applications in all the nooks and crannies of our technology driven lives. By 2020, 1.7 megabytes of new information will be created every second for every human on the planet with an accumulated digital universe of 44 trillion gigabytes[1]. Data science is focused on how much of that data can be converted to intelligence.

In this enormous data landscape of opportunity, data science processes are challenged to be better connected, more efficient, and more effective at deriving knowledge from data. Among the many data science tools and platforms, AutoML promises to drive efficiency and effectiveness by directing sophisticated algorithms, replicating expert knowledge, and utilizing increased compute power.

This paper focuses on the "state" of AutoML. Part 1 includes a high level overview of AutoML, its history, its place in data science, and why data scientists will ultimately embrace AutoML as a key tool in their arsenal of data mining capabilities. Part 2 reviews the level of sophistication currently available in AutoML by evaluating available technology in this space.

The escalating growth in data science presents the unavoidable issue of this evaluation quickly becoming out of date. Great effort was made to collect and present the most current and accurate information. All comments will be considered and corrections will be made. Also, in many cases there is little awareness of existing AutoML technologies and many may still reside under my radar. Any new or existing AutoML applications not mentioned will be added as information becomes available.

(1) – Bernard Marr - Forbes – Big Data: 20 Mind-Boggling Facts Everyone Must Read, Sept. 2015

Part 1: Overview – Machine Learning & AutoML

Machine Learning – Pre-Automation

Machine learning rose in popularity in the 1980s as desktop computing became available to scientists, engineers and statisticians. The gift of personal computing provided innovators with compute power to be used at their discretion along with the ability to program and test theoretical mathematical algorithms. This was the golden era of machine learning (ML). The painstakingly slow experimental process of time sharing mainframe environments gave way to accelerated research and development of new statistical and biological inspired learning algorithms.

Spiking the excitement during this period was that a machine (the computer) could systematically learn to attribute patterns of input data with associated outcomes without the assistance of a computer programmer. At the time this seemed to address one aspirational goal in artificial intelligence (AI) - the ability for computers to learn on their own. The introduction of ML capabilities transformed the focus of AI from the knowledge-driven approach of expert systems which duplicated human intelligence, to the data-driven approach of ML which could augment human intelligence. The ability for a computer to learn simply by being presented data spiked imaginations, as well as expectations.

Interestingly, various types of regression models (linear ML algorithms) had already existed for years and were successfully being used in business applications. What was relatively new was the ability to freely store and manipulate data, write programs and algorithms to operate on that data, and to run these programs entirely on the desktop computer. A researcher no longer had to coordinate between people, departments and different computing environments to get things done. Research and development in ML has flourished since.

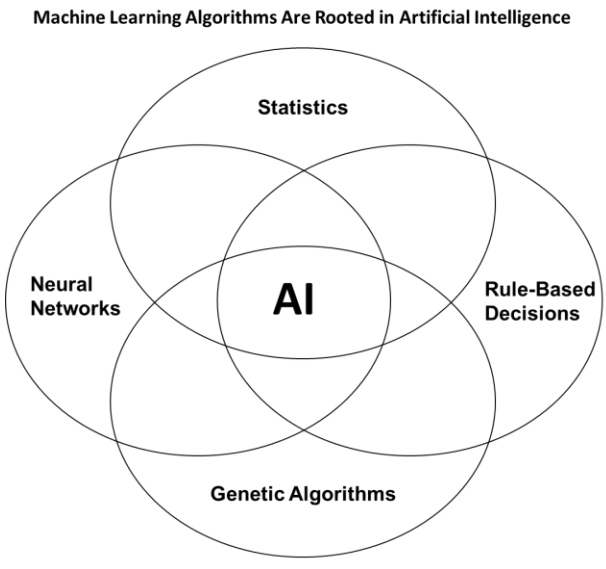
By the early 1990's ML was a fixture in AI, fueling enthusiasm aimed at developing human-like intelligence in computers. Although ML was one of many different AI oriented tools (as it is today), its unique characteristic to learn, whether supervised or unsupervised, from complex data relationships filled a void in AI left by expert systems which dominated the field at that time.

At the heart of each ML method was one of a variety of learning algorithms. From simple statistically driven rule sequences to complex, high dimensional techniques. Most algorithms fell within the four different categories shown in Figure 1. Each category evolved rapidly in the 1990's with more available data and compute power.

1. Sound statistical operations were able to run more efficiently and frequently
2. Multiple neural network paradigms were refined including multi-layer deep learning models
3. Experiential genetic algorithms patterned after evolution theory became a viable learning method
4. Rule-based logic included established expert systems but ML added ability to deduce rules (ultimately decision trees) from datasets using early ID3/C4.5 algorithms

The addition of ML rounded out the AI space for AI applications in business, science and consumer products.

Still, there were barriers. While many ML applications existed, none had the AI "wow factor" that we see today. The theoretical software components of AI were



Source: 1990 International Joint Conference on Neural Networks (IJCNN)
Figure 1

well defined but there were many hurdles left to successfully apply them in the real world.

Key barriers were access to data, available compute power and the quickly maturing (but not quite there) digital insurgence into our business and personal lives.

> **Data Access**: AI methods needed data, and at the time the correct data was hard to pull together, was incomplete, or just did not exist. When data could be supplied manually to develop models, refreshing it was laborious.
>
> **Compute Power**: Two key issues in computer technology slowed the development of AI applications. First, ML algorithms were mathematically intensive and CPU speeds were a fraction of today's high-performance chips. Second, the limitation of RAM memory constrained the size of a dataset or required disk access to operate on larger datasets. These issues combined made ML a slow, grueling process.
>
> **Digital Insurgence**: Even if the issues of data access and compute power were resolved, it wasn't clear how AI decision applications could be incorporated into business or our personal lives. The infiltration of digital technology was picking up momentum, but was at the bottom of an exponential curve leading to what we see today – endless innovation of devices that can both generate data and use data to make decisions.

These technical barriers persisted for over a decade. AI projects often overshot stakeholder expectations and under-delivered on capability leading to less prominent projects and cautious AI announcements. Fortunately, a persistent undercurrent of research and development in corporate labs and universities provided steady advancements and refinements in ML algorithms, albeit without a lot of fanfare.

As technology evolved so did AI, particularly the recognition and usefulness of ML. Today, awareness of the number of applications where ML can now be effective is the primary factor for its popularity, and attributable to the established practice and growth of data science coupled with supporting technology and data

Methods for ML evolved steadily over the years, but the recent resurgence of ML is mostly due to highly efficient creation, collection, storage and access to data; powerful compute power to run ML algorithms faster; and a digital/mechanical environment that is receptive to intelligent, data-driven decision processes.

Figure-2 highlights just a portion of the ML methods currently used in AI applications. These capabilities and more are packaged together in data science platforms.

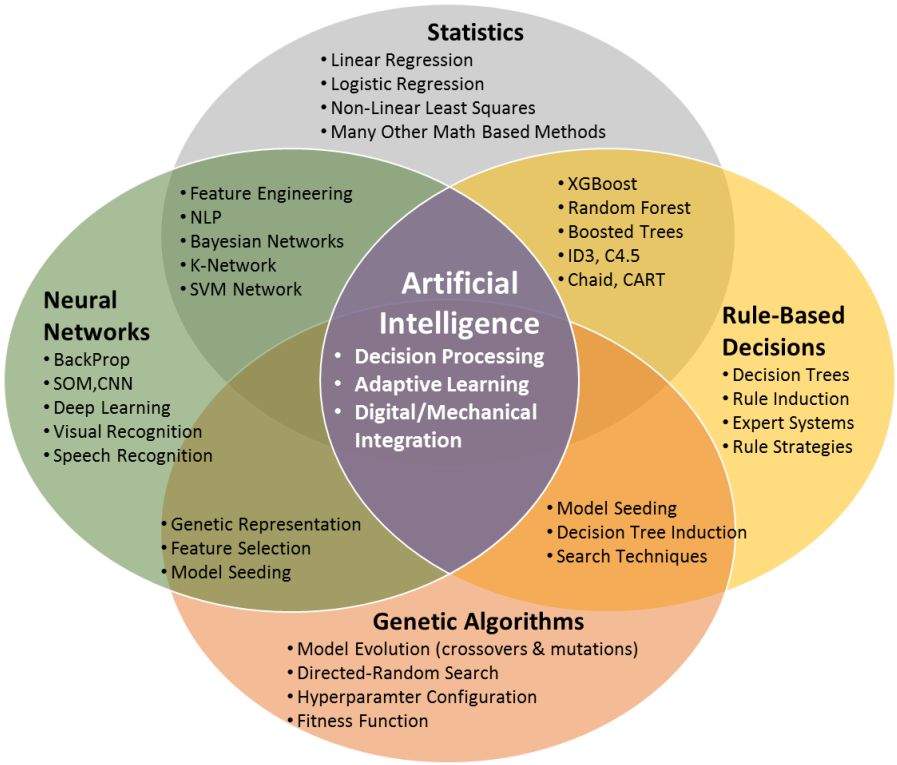


Figure - 2

Data Science Platforms

Data science platforms enable a full spectrum of routines for analytical operations and ML in a single environment and often an efficient method for deploying models. See a list of leading data science platforms in the Table-1 to the right.

Leading Data Science Platforms	
• KNME	• TIBCO Software
• Rapidminer	• MathWorks
• SAS	• SAP
• Alteryx	• Anaconda
• H2O.ai	• Angoss
• Domino	• Teradata
• IBM	• Databricks
• Microsoft	• Dataku

Table-1

Along with numerous commercially available data science platforms, there is also a fairly mature set of open-source libraries in leading programming languages such as R and Python providing analysis and ML methods. Since data scientists often work with open-source methods, many platforms support open source code as part of their API.

Currently, data science platforms always require knowledgeable users who understand how to use them. Some of the leading platforms provide modeling tournaments (the ability to run multiple ML methods concurrently). Tournaments help by testing different methods concurrently, but they do little to add efficiency in data manipulation or optimizing hyper-parameters.

Due to the high demand of ML methods in business operations and the growing shortage of data scientists, there is a need to make the ML process as efficient as possible.

The Automation of Machine Learning

In 2015 the International Conference on Machine Learning (ICML) had the following statement on their landing page:

> *"Machine learning has achieved considerable successes in recent years, and an ever-growing number of disciplines rely on it. However, this success crucially relies on human machine learning experts, who select appropriate features, workflows, machine learning paradigms, algorithms, and their hyper-parameters. As the complexity of these tasks is often beyond non-experts, the rapid growth of machine learning applications has created a demand for off-the-shelf machine learning methods that can be used easily and without expert knowledge. We call the resulting research area that targets progressive automation of machine learning AutoML."*

Automated Machine Learning (AutoML) is not a totally new concept, but the ICML may have been one of the first to publicize. In fact, methods for automating ML processes have evolved for decades and just like ML itself, AutoML can now leverage today's technology, making it more efficient and effective. In its 2016 study The Age of Analytics: Competing in a Data-Driven World, the McKinsey Global Institute (MGI) predicted the shortfall in data scientists could be eased through the automation of data preparation, which accounts for 50 percent of data science work. Today's AutoML technology is exhibiting capabilities well beyond data preparation.

The goal of AutoML is to automate the most repetitive aspects of the data mining process. Automation helps free up data scientists' time to prescribe data, improve production performance and identify other valuable business applications. Many ask how, because AutoML could never replace the domain expertise of a data scientist. This is true. The data scientist understands what data is pertinent to a business solution and how it should be assembled. He also understands how a model should be deployed and how it should be used in production. But, what about the "art" of modeling? The art of modeling in reality is the process and procedures that a data scientist performs for exploratory data analysis; ML algorithm choice and hyperparameter setup;

iterative model fitting procedures; and model performance and ensemble testing. It is this process that AutoML aims to perform automatically and faster, while testing more possibilities and producing models with equal or better performance. It does this by packaging the modeling process in an AI driven modeling tool.

One unique characteristic about AutoML, just as ML plays an important part in AI applications, AI plays an important part in AutoML by automating the art of modeling.

AutoML has created a bit of a stir and raised a lot of eyebrows in the data science community, and like many new concepts, its reputation has been socialized first before any real standard definition has been established. Regardless, there is a very clear benefit to data scientists for advancing AutoML technology, it will make them more effective.

The Scope of AutoML Today

It is difficult to establish a definition for something that is rapidly evolving. The best we can do is describe the state of AutoML, what it does now, and where the technology is heading. The term "AutoML" is widely used by many vendors with varying capabilities. While researching the AutoML space it was necessary to establish a definition to help understand and segment the players.

The initial step was isolating the process which AutoML most often addressed, building the model. A data scientist identifies the problem to solve and the required data. The next step is building outlined in Figure-4 below, starting when data is presented to the AutoML tool.

The Machine Learning Process

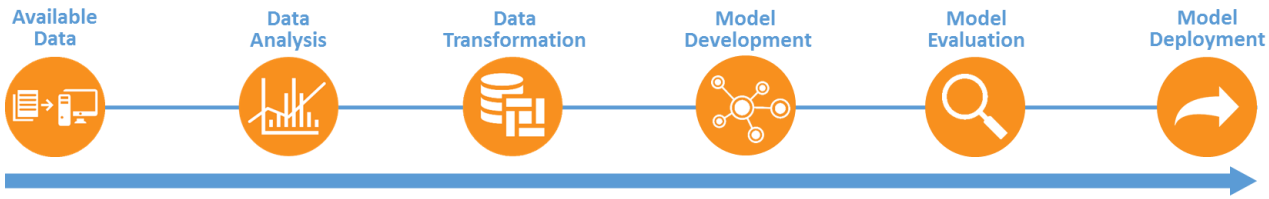


Figure - 4

To varying degrees, AutoML tools apply intelligent automation to data cleansing, feature analyses, transformations and model development. Model development often involves an iterative process with one or more learning algorithms, model testing, hyperparameter adjustments, model evaluation followed by one or more options for model deployment.

Automating the Data Analysis process through to Model Deployment is the scope of AutoML today

What AutoML is Not

There are many misconceptions about AutoML. As the value of ML is more widely recognized, there is confusion on what exactly it is capable of performing – AutoML is no exception. The list below may clarify capabilities that are not typically addressed by AutoML.

- **Data Collection** – AutoML tools all have some level of data connectivity, whether by ingesting a file or directly connecting to external data sources. They are rarely a good application to continuously collect, store and maintain data outside of their immediate needs.

- **Comprehensive Data Science Platforms** – There are numerous widely used data science platforms currently in the market, most are listed in Figure-3. They offer nearly all the data manipulation, statistical operations, visualization, machine learning and model deployment capabilities a data scientist would need. These platforms are a good place to incorporate AutoML and some are moving in that direction, but a data science platform that offered the capabilities of AutoML (as defined above) was not identified.
- **Decision Processing** – To leverage data and data science operations, some form of structured decision logic must take place. This is typically not performed *directly* by machine learning output, or even in a data science platform. Decision engines exist with high connectivity and the ability to execute a decision, usually in real-time. While AutoML tools provide input into decision processing, they do not typically execute decisions.
- **Adaptive Learning** – AutoML is not a form of ML that is adaptive. In this case adaptive refers to the ability for a model, once developed and put into production, to continue to learn. Adaptive learning is dependent on available data that identifies decision outcomes (attribution) which must be available for models to be refit automatically. That said, some modeling tools have the ability to schedule or automate retraining using AutoML capabilities.
- **Modeling Tournaments** – As mentioned above, modeling tournaments are offered by many data science platforms to test multiple methods concurrently. While modeling tournaments are a simple form of AutoML, the minimum requirement for a modeling product to be part of this evaluation was its ability to perform automated exploratory data analysis, transformations and hyperparameter tuning.

AutoML Helps Data Scientists Meet Growing Demands

In an environment where "C" level executives are asking themselves, "how should I be using AI in my business?" the importance of the data scientist has been escalated. He or she is typically focused on four different competencies – data research, model creation, model deployment and contributing to business innovation. If a company is going to leverage ML in their business, having data scientists who understand the whole process is critical for success.

Today's data scientist needs to be competent in data access and manipulation, statistical analysis and modeling, and the strategic business applications of their work. A strong data scientist first determines the analytical approach to solve a problem or create a new application; then works through the data wrangling, analytical procedures and modeling; and finally provides a smooth transition for model deployment into production. AutoML will need to make this process more efficient because even with high numbers of students and professionals steering toward data science, a growing shortage in this area is forecasted.

In its 2011 report, "Big data: The next frontier for innovation, competition, and productivity" MGI predicted that in 2018 "the United States alone could face a shortage of 140,000 to 190,000 people with deep analytical skills as well as 1.5 million managers and analysts with the know-how to use the analysis of big data to make effective decisions."

In a 2016 follow up report, "The Age of Analytics: Competing In A Data-Driven World" MGI confirmed this trend stating "approximately half of executives across geographies and industries reported greater difficulty recruiting analytical talent than filling any other kind of role." They go on to project a 12% annual growth in demand for data scientists, leading to a 250,000 shortfall in the workforce.

Our ability to create, collect and store data has outpaced our capacity to derive its hidden intelligence building a compelling case to advance AutoML technologies.

Why Data Scientists Should Embrace AutoML

AutoML discussions spur varied reactions by data scientists, many skeptical of its capabilities. The truth is current AutoML technology has a lot of shortcomings, but researchers are solving these issues and its capabilities are growing rapidly. Will AutoML be putting data scientists out of a job? Not likely. The scope of data science goes well beyond the scope of AutoML, the latter being highly focused on exploratory data analysis and model creation.

The data scientist is critical for understanding the problem to be solved and the raw data to support the solution. At that point AutoML technology is likely to save the data scientist a lot of time by automatically exploring the data, modeling options and model development. William Vorhies, who blogs regularly for Data Science Central, describes the abilities of AutoML as "One-Click Data-In-Model-Out."

AutoML often produces multiple models and model ensembles, with various performance characteristics from the same dataset. The data scientist is still responsible for choosing the right model and working to properly deploy the model.

In the future, business requirements will specify hundreds of models to be deployed and data scientists will have more responsibility than just building models. Their role is expanding to be the purveyor of innovative ideas and approaches that improve business.

A data scientist will likely be pleased to have all of the feature analyses, selections, transformations, model configurations, fittings and iterations all performed for him automatically with the reliance that the AutoML process will produce a model that is equal or better to what he could do in a fraction of the time.

The AutoML Market Landscape

Currently there are multiple tools and platforms that have AutoML in their marketing literature or on their websites. There are a number of offerings that only set the hyperparameters for model development but do not provide any cleaning, EDA or transformations of data. Interestingly, many of the tools focused on AutoML were offered by relatively younger, early stage companies rather than from more mature data science platform providers. Some focused purely on the underlying functionality of AutoML, and others paid more attention to the UX and provided more data visualization. All stated a similar mission, to continually build out an intelligent automated process for model building.

Another major difference between companies was the use of non-traditional proprietary techniques to achieve automation while others were more reliant on open-source technologies and leaning toward transparency. Open-source plays well with data science platforms where data scientists have the convenience of working in a comprehensive tool environment with the ability to import their own Python or R code to side step any limitations they find with the platform.

AutoML is evolving rapidly in open-source as well as commercially available software. For this evaluation we looked at both, but spent more time evaluating the commercially available software which was more suited for setting up an independent model building environment without coding. That said, one open-source technology was included (H2O) as we found that its GUI made the setup and test more efficient. The companies and open-source technologies that were identified in the AutoML space are listed in Table-2 below[1,2].

Tool	Open-Source	Price Comparison	Evaluated
TIMi (The Intelligent Mining machine)	No	?	No (unable to contact)
Compellon	No	Low	Yes
NeuralStudio.ai	No	Low	Yes
DataRobot	No	Med-High	Yes
DMWay	No	Med-High	Yes
MLJAR	No	Low	Yes
PurePredictive	No	?	No (unable to complete evaluation)
Xpanse Analytics	No	Low	Yes
PredicSis.ai	No	?	No (unable to complete evaluation)
Ople.ai	No	High	Yes
Tazi.ai	No	High	Yes
NumberTheory.ai	No	Low-Med	Yes
Auto-WEKA	Yes	Free	No (required code integration)
Machine-JS/Python auto-ml	Yes	Free	No (required code integration)
H20.automl	Yes	Free	Yes
Auto-SkLearn	Yes	Free	No (required code integration)
TPOT (Tree-based Pipeline Optimization Tool)	Yes	Free	No (required code integration)

1. Data science platforms as mentioned above were not included unless they had AutoML capabilities.
2. Considerable effort was made to identify tools in the AutoML space, but it is likely offerings were missed. If so, please contact the author and this document and the evaluation in Part 2 will be updated.

Table-2

Part 2 of this paper includes a description of the evaluation process and the detailed results of each tool listed as "Evaluated" above.

Part 2: AutoML Tool Evaluation Summary

Machine learning has been making great strides in many application areas. The successful adoption of machine learning (ML) into various disciplines has fueled a growing demand for human machine learning experts (aka Data Scientists) who are short in supply. This motivated the development of off-the-shelf tools that can be easily used in any field, without expert machine learning knowledge. Most of the AutoML tools today are one-click-data-in-model-out platforms designed to automate the most repetitive elements in the model building process. The minimum requirement for a tool/product to be part of the AutoML club is its ability to automate modeling i.e. automatically run multiple algorithms in parallel, auto tune hyperparameters, and select a champion model based on the evaluation metric for implementation.

AutoML Minimum Requirement: For this evaluation, the minimum requirement for an AutoML tool is its ability to automatically perform some level of data preparation and machine learning without human intervention.

AutoML Product Evaluation Framework

AutoML vendors were engaged to understand their product's capabilities, how they compared, and in what direction their technology is evolving. We started by setting up a consistent evaluation framework that would capture the key features of each tool. The following eight areas were considered as base functionality for an AutoML tool:

1. Data Connectivity
2. Capabilities and Automation in Data Processing including Summarization, Exploration & Cleansing
3. Capabilities and Automation in Feature Engineering including Data Transformation and Feature Selection
4. Capabilities and Automation in Learning Algorithms including Hyperparameter Tuning, Type of Problems and Ensembles
5. Data and Model Performance Visualization
6. Model Competency Based on Validation Scoring
7. Deployment Options including in Product GUI, Code Deployment and Embedding
8. Pricing

These criteria were used to create the evaluation sheet outlined in Figure - 5.

AutoML Tool Evaluation Criteria	
Installation - Cloud/On-premise/Both	
Data Connection	*Databases*
	Unstructured data
	Flat file
Data Exploration	*Summary Statistics*
	Visualization
Data Cleaning	*Outlier Detection*
	Missing data imputation
	Dimensionality Reduction
Data Transformation	*Types of transformations*
	Feature Engineering
Feature Selection	
Machine Learning Problem categories	*Classification*
	Regression
	Clustering
	Anomaly Detection
	Time Series
Automated Hyperparameter Tuning	
Modeling Tournament	
Model Ensemble Testing	
Model Evaluation/ Performance Metrics	
Model Deployment Options	GUI
	Web Service/restAPI
	Code Deployment
Visualization/Reporting Analysis	
Additional Comments	
White Labelling/Commercial	
Pricing	

Figure - 5

Once identified, each vendor was contacted and asked to participate in the evaluation which included three steps.

Know the product - The objective of this initial meeting with the vendor was to get a high-level understanding of the product and service offerings. These meetings were usually led by the vendor's marketing/sales team.

In-depth product understanding. During this meeting we gathered technical understanding of the product by having a detailed discussion with the technical/analytic members of the vendor's team who would typically demonstrate the product in operation.

Modeling competency. To evaluate each product competency in building models, three open-source datasets were provided to each vendor to build models with their product. Validation datasets were then scored for performance. The datasets provided are described in Table-3 below.

Class of Problem	Dataset	Evaluation	Brief description of the datasets
Binary Classification	Census-Income dataset (KDD)	Accuracy, AUC	Dataset has 40 variables with "Target_Income" as the binary target. The objective is to classify data points as Income>$50K or Income<$50k. The two classes in target column are marked as "50000-" and "50000+.". Total 224464 rows in training data and 74822 rows in validation data.
Multi-class Classification	Lending Club Loan (Kaggle)	Accuracy, AUC	Dataset has around 75 variables with "loan_status" as the target. The target column has ten different classes. The objective is to classify each row into one of the ten classes of loan status. Number of rows in training data = 665535 and number of rows in validation data = 221847.
Regression	Cycle Share (Kaggle)	RMSE, MAE	Dataset has around 33 variables with 'tripduration' as the target. 'tripduration' is represented in 'seconds'. The objective is to predict duration of a cycle trip based on independent variables which include bike station information, weather information and user information. Number of rows in training data = 38095 and number of rows in validation data = 12700.

Table-3

Vendor Highlights

We evaluated ten AutoML vendors who met the minimal requirements. Below is a brief description of each.

NeuralStudio.ai is a one-click data-in model-out platform. They have two on-premises offerings and a cloud based service. Their products leverage the flexibility of Cascade Learning for building neural networks. This approach starts with a simple model and increases the complexity (depth and width) of the model as it measures improvements in performance. The service includes modeled missing data imputation.

DataRobot provides an automation for non-data scientists with customizable options for data scientists. It is GUI based with both cloud and on-premises offerings using open-source libraries like R, Python, H2O, Spark MLLib, and Tensor Flow in their product. It performs an exhaustive search for possible combinations of hyperparameters, algorithms, data cleaning and feature transformations to get to the final model.

Compellon is built for non-data scientists and focuses more on interpretability. Compellon is GUI based product available through a cloud service. It uses a proprietary AI platform that does not rely on any of the known statistical modeling algorithms. The prescriptive part of the tool consists of 'Advisor' and 'What-If' functions. This functionality helps the user understand hypothetical scenarios and possible actions related to them.

H20.AutoML is an open-source offering. H2O can be installed on a local PC, server or cloud platform. H2O has libraries and packages for popular data science languages like R and Python. Users can use these libraries to include H2O capabilities within their existing environment through code or GUI. The 'AutoML' portion of H2O focuses mainly on hyperparameter tuning and algorithm selection.

MLJAR is a platform available as a cloud service offering and as an on-premises installation. The platform focuses on automated modelling through both a GUI for business users and R/Python code for the more technical users.

Xpanse is offered only as a cloud based service. Users interact with the tool through an easy to use GUI. The interface enables users to handle data and perform modeling without having technical details of algorithms. Xpanse.ai differentiates itself from its competitors in the AutoML space through advanced data import and feature engineering capabilities. At present, the tool is capable of handling only binary classification problems.

Ople.ai is offered on the AWS and Azure clouds and on-premises installation. Users interact through a GUI. This tool uses a two stage approach for picking the best model. The first stage runs a regular modeling tournament and the second is a deep learning behavioral assimilation system (BASS) which captures the best aspects of each of the models from first stage. This information is used to develop the final BASS model.

DMWay is a simple, easy-to-use on-premises AutoML platform built to target non-data scientists. DMWay focuses on building a framework around GLM class of algorithms. They are currently creating a seamless GUI based platform to provide interpretability and volume of models.

Tazi.ai is available as an on-premises installation and also as a cloud service on Amazon. Tazi.ai has an automated easy to use GUI based framework with customization options for more advanced users. The tool has an interactive visualization layer which helps with model comparison. The tool has a unique feature where users can interact to actually manipulate the current model structure based on their domain expertise.

NumberTheory.ai is available both as a cloud service and an on-premises installation. The platform includes ETL focused modules and reinforcement learning. Although the platform includes several modules that perform automated functions, we found it was not a one-click- data-in-model-out platform. NumberTheory is more like Azure ML Studio, SAS EM and IBM SPSS. The difference is the user needs to build the whole workflow to produce final models. This gives the user more control and customization options but requires the user to be well versed in modeling and data analytics.

Vendor Comparison - Pros and Cons

Vendor	Pros	Cons
NeuralStudio	• Comprehensive Data Transformations • Highly Optimized Neural Computing • Use of GAs for Feature Selection • Generates Java/C/.NET code for scoring	• Only works with flat files • Little visualization capability
DataRobot	• Good data source connection options • Transparent algorithms and process flow • Good visualization • Automates most of modeling pipeline • Modeling cycle can be customized	• Users cannot edit the final models. They can import R/Python and train them in DataRobot to compare with other models but users cannot change the models built by DataRobot • Scoring code in Java/Python requires added cost for 'Prime" subscription
H2O	• Good data connectivity options. • Option to use the libraries within Python/R or to using the GUI • Builds ensemble models	• Limited data visualization • Missing data imputation and data preparation not part of AutoML function
Compellon	• Good visualizations • "What-if" and "Advisor" functions • Outputs a model structure graph explaining relationships between variables and target • Explains each prediction with causation • Easy and interactive GUI • Can process text data	• The whole modeling process is a black box • Does not perform data processing, cleaning, transformations, tuning in the traditional sense • Uses proprietary black-box technique to infer model structure
Xpanse	• Extensive feature engineering. Creates a large number of features which can be exported • Can take multiple data files as input and join them based on ID • Good visualization of model performance • Exports SQL code for scoring on database	• No regression • Only csv as input • Limited algorithms available (3) • No missing value imputation
MLJAR	• Good integration with R/Python through API'. • Good summary statistics and visualization • Web based UI	• Only supports binary classification and regression. • Only takes CSV files as input data

DMWay	• The tool exports scoring code in variety of languages including Java, R, Oracle SQL code, MS SQL Server code • User can host Java scoring code and score new data using an API call to their server • Exports final reports to R and Excel	• Missing value imputation does not have actual value imputation options
Ople.ai	• Uses a proprietary behavioral modeling system on top of various open source algorithms • Weights for the final model can be exported	• Only accepts csv file as input • No missing value imputation • No data transformations
NumberTheory	• Wide variety of data connectivity options • Data exploration/summary statistics • Has an option for Reinforcement Learning • Has options to set automated/scheduled model retraining	• Not a one-click model out tool. Users have to build the whole modeling pipeline
Tazi	• Good data connectivity • Online model retraining. Models keep learning as they operate • GDPR compliant models. Explains every prediction	• Does not support regression • Cannot apply user-defined data transformations.

AutoML Vendor Landscape - Data Science Tasks and % Automation

Information from the Detailed Evaluations was consolidated in into four primary areas: Data Processing, Feature Engineering, Modeling and Visualization. The results are in Figure 6 below. Considering all of the options for each primary area (swim lane), each vendor was rated for less/more options on the X-axis, and the percentage of automation (Y-axis).

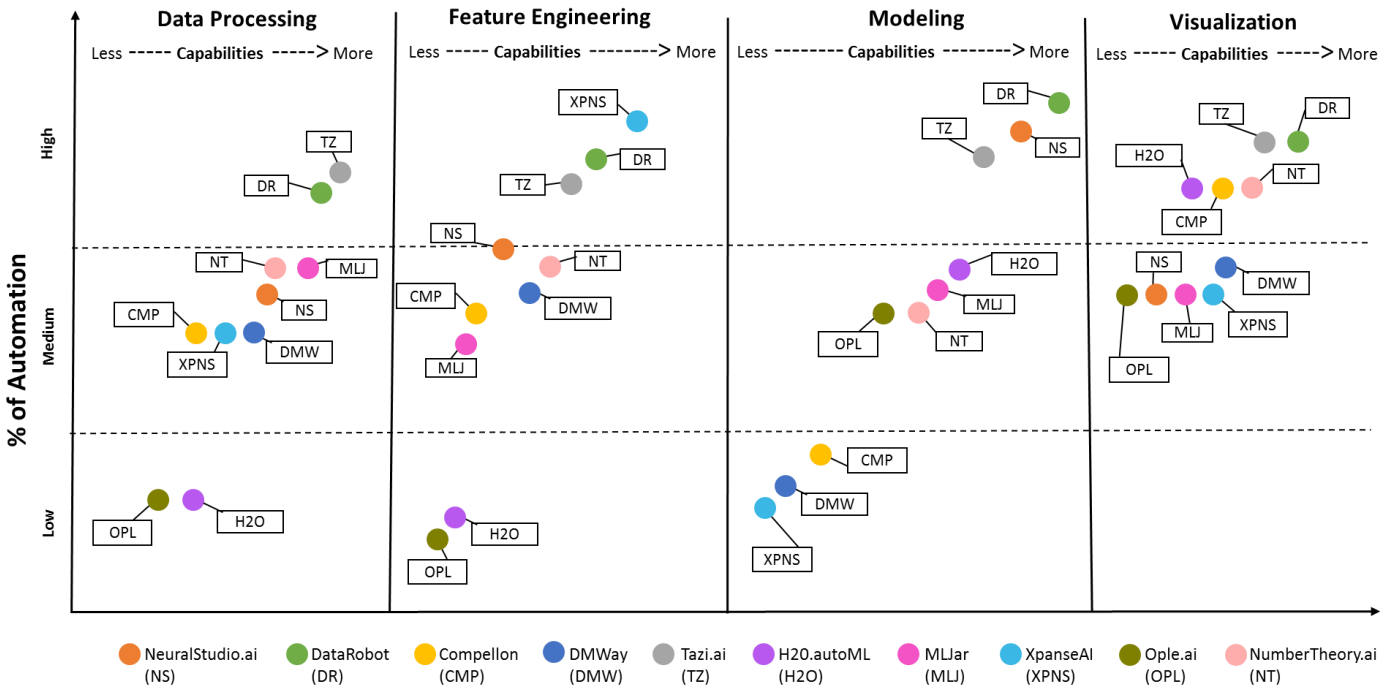


Figure - 6

Figures 7, 8 and 9 below provide top line information for each vendor.

Data Input Supported by Vendor

Vendor	Flat File	Database	Unstructured Data
NeuralStudio.ai	✓		
DataRobot	✓	✓	✓
H2O	✓	✓	✓
Compellon	✓	✓	✓
Xpanse	✓		
MLJar	✓		
DMWay	✓	✓	
Ople.ai	✓		
NumberTheory	✓	✓	✓
Tazi	✓	✓	✓

Figure - 7

Class of Problems Solved by Vendor

Vendor	Single Class	Multi-Class	Regression
NeuralStudio.ai	✓	✓	✓
DataRobot	✓	✓	✓
H2O	✓	✓	✓
Compellon	✓		
Xpanse	✓		
MLJar	✓		✓
DMWay	✓	✓	✓
Ople.ai	✓	✓	
NumberTheory	✓	✓	✓
Tazi	✓	✓	

Figure - 8

Deployment Methods by Vendor

Vendor	GUI	Web-Service/ API	Code Deployed	Run-Time Embedded
NeuralStudio.ai	✓	✓	✓	✓
DataRobot	✓	✓	✓	
H2O	✓		✓	
Compellon	✓	✓	✓	
Xpanse	✓		✓	
MLJar	✓	✓	✓	
DMWay	✓	✓	✓	
Ople.ai	✓	✓		
NumberTheory	✓	✓		
Tazi	✓	✓		

Figure - 9

Modeling Competency by Vendor

Modeling competency was measured with the open-source datasets mentioned above. The training dataset had the targets for building models. A validation dataset without targets was used to score and evaluate. Each vendor used their AutoML technology to build models for binary classification, multi-classification and regression. The results are below in Figures 10, 11 and 12 respectively.

Performance Read Out - Binary Classification

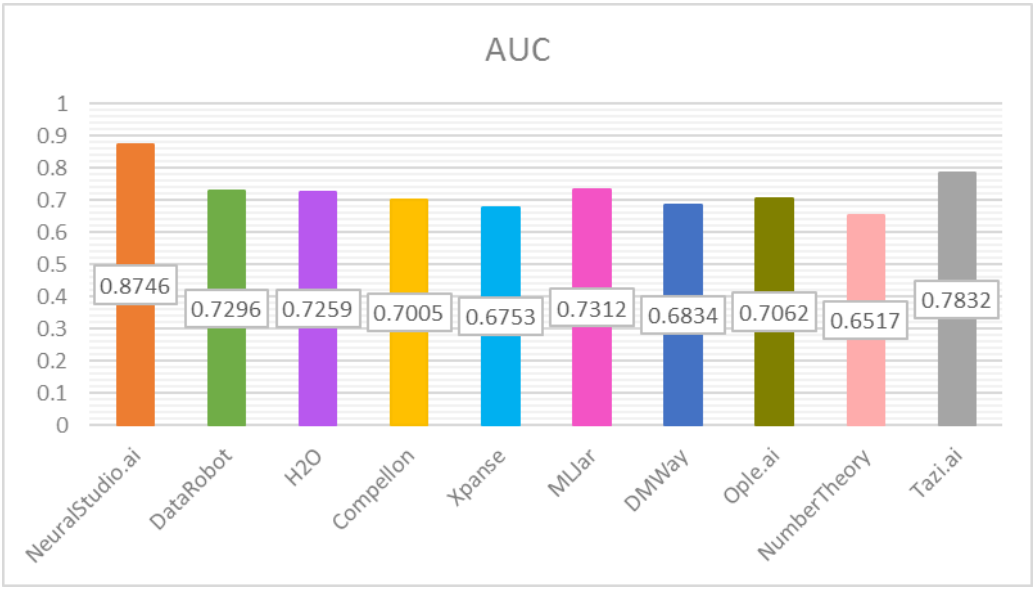


Figure 10

Performance Read Out - Multiclass Classification

(Mean of AUC as defined by Hand and Till, 2001)

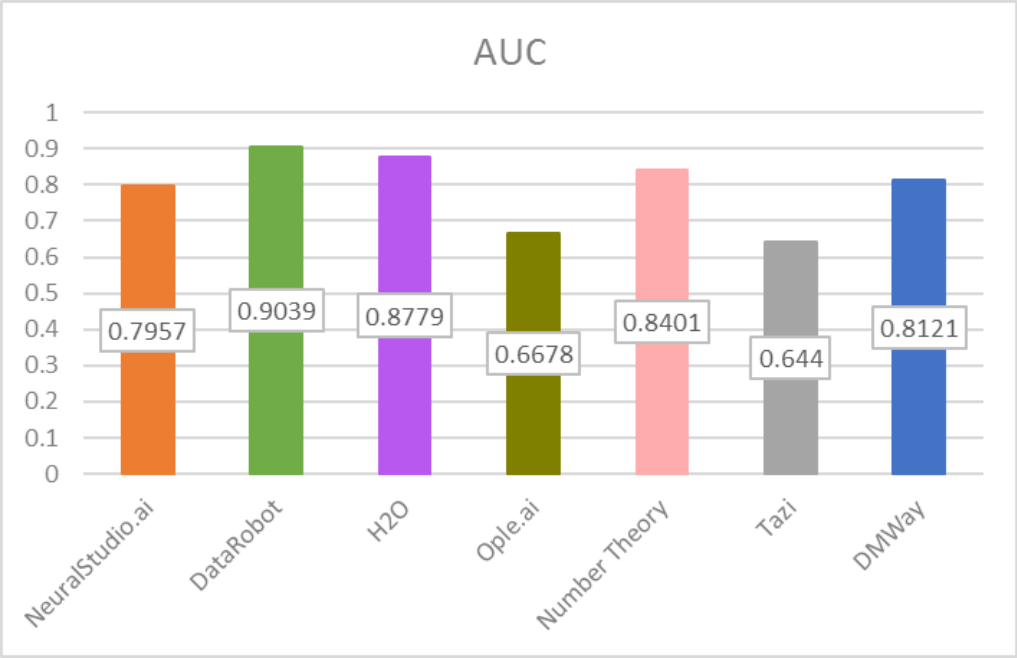


Figure 11

Performance Read Out – Regression

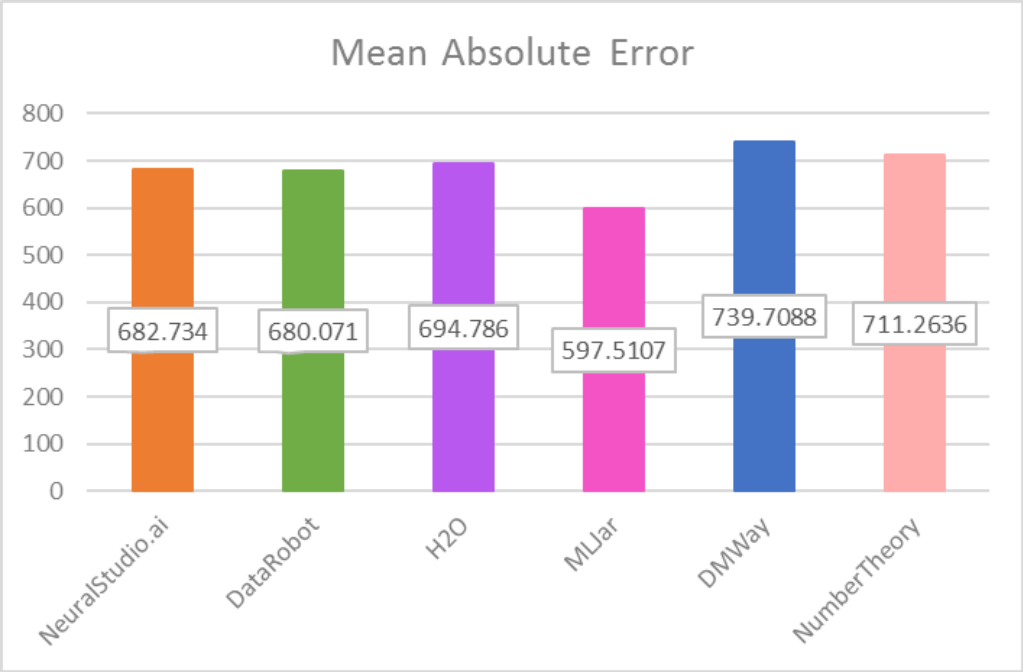


Figure 12

AutoML Tool Evaluation Details

NeuralStudio

Company Overview

NeuralStudio is a Cayman Islands based company with a U.S. office in Pittsburgh, PA. It is one of the early forerunners providing neural technology in the 1990's and AutoML in the 2000's. They offer a cloud service and desktop products for on-premises applications. Their Predict AutoML tool is the core for all products. All products are designed for both data scientists and business users.

Product Overview

NeuralSight – An AI driven framework which drives the Predict engine exploring hyperparameter settings for model configuration, feature selection, model performance, ensemble construction and sensitivity analysis. The base AutoML engine (Predict) is also available as Excel Add-in.

NeuralStudio.ai – Includes proprietary data cleansing technology which provides model automation for missing data imputation using "field models" developed from the primary dataset and a Model Factory that employs genetic algorithm optimization driving NeuralWorks Predict hyperparameters. Models can be run with new data uploaded to the site or a RESTful API for model execution.

Data In

NeuralStudio products accept delimited files as data sources. The on-premises product NeuralSight is not tolerant of missing data and expects all data to be present. For data preparation in the cloud service, missing data and other data attributes are flagged and reported. The cloud service handles missing data by automatically developing a model for each field to impute the missing value. For records that have more than one missing value, the service employs an unsupervised Self-Organizing Map (SOM) in a neural configuration. The SOM maps likely values for missing fields to the records.

Once the data is cleansed, there is a feature engineering process that automatically creates a series of transformations for each field. Then a feature selection process is conducted which develops preliminary models using feature sets chosen by a GA. The best feature sets are used during final model construction.

Learning Algorithms

All NeuralStudio products and services use a proprietary version of Cascade Learning, also known as Cascade Correlation. This approach starts with a simple model, like a logistic regression configuration, and increases the complexity (depth and width) of the model as it measures improvements in performance. The learning process enables linear models and will expand to single layer and multi-layer deep learning neural networks. The performance of the tool can be changed by adjusting the level of parallel processing.

All hyper-parameters are initially set at values such that an inexperienced user can successfully develop a model. The software iteratively explores variations involving feature selection, increased model complexity, learning coefficients, and other parameters for experimentation. A list is maintained of the top performing models (default is 50). All combinations of ensembles are created and measured and the best performing models and ensembles are then identified and retained for deployment by the User.

Model Deployment Options

1. Deployment in the product itself, whether on-premises or the cloud service.
2. Models can also be downloaded as "FlashCode" (C, Python, Fortran) providing the model and all data transformations as a function.
3. Run-Time Kit (RTK). The RTK comes in Java or .NET and includes the code and associated libraries to accept raw data to reconstruct all of the data transformations and execution of the model. Once the RTK is deployed in an application, all new models and model updates can be incorporated without any additional programming or recompiling of the application.

NeuralStudio Evaluation Details

AutoML Tool Evaluation Criteria		NeuralStudio	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Both	The cloud version currently has more features including modeled missing data imputation and more enhanced GA driven hyperparameters. Cloud version may be available on-premises in the future.
Data Connection	*Databases*	No	
	Unstructured data	No	
	Flat file	Yes	Supports data input from flat delimited files.
Data Exploration	*Summary Statistics*	Yes	Automated
	Visualization	No	Field statistics available but not graphical.
Data Cleaning	*Outlier Detection*	Yes	Automated
	Missing data imputation	Yes	Field models developed automatically in cloud version. This technology could be licensed on premises with vendor.
	Dimensionality Reduction	Yes	Automated
Data Transformation	*Types of transformations*	Yes	Performs multiple transformations for both categorical and numeric fields. Creates and evaluates all transformations automatically as part of feature selection.
	Feature Engineering	Yes	Uses a GA to test most valuable sets of features. Does not combine features. Persists most useful groups of features for testing during model development.
Feature Selection		Yes	Uses a modeling process and a genetic algorithm to find feature sets.
Machine Learning Problem categories	*Classification*	Yes	Uses Cascade Correlation Neural Network.
	Regression	Yes	Uses Cascade Correlation Neural Network.
	Clustering	Yes	Uses Kohonen Self Organizing Map.
	Anomaly Detection	No	The product does Self-Organizing Maps which can be used as a basis for anomaly detection.
	Time Series	No	Requires manual data preprocessing.
Automated Hyperparameter Tuning		Yes	Uses random, round robin and genetic algorithm expert system for automated hyper-parmater tuning.
Modeling Tournament		Yes	Uses multiple configurations concurrently of Cascade Correlation with various configurations, parameters and feature sets.
Model Ensemble Testing		Yes	Test all combinations of top XX models. Cloud version uses a genetic algorithm for this task.
Model Evaluation/ Performance Metrics		Yes	Multiple ways to evaluate. Also has sensitivity analysis for key feature effects to output.
Model Deployment Options	*GUI*	Yes	Will deploy in same development environment.
	Web Service/restAPI	Yes	Cloud version only.
	Code Deployment	Yes	Produces flashcode in C, Fortran and Python. Also has JAVA and .NET RTK modules for deployment through the standard model definition file (NPR).
	Run-time Embedded	Yes	Java or .Net embedded function provided that can execute (realt-time or batch) any model created using the cloud or on-premises products.
Visualization/Reporting Analysis		Yes	Lift chart, sensitivity analysis.
Additional Comments			The product uses highly optimized neural network technology to solve all modeling problems. All hyperparameters are accessible. AutoML operations can be configured by user, but not necessary.
White Labelling/Commercial		Yes	Structure is TBD by customer.
Pricing		Yes	Low - Subscription or pay-as-you-go cloud service starting < $500. On-premises product is one-time perpetual license/seat. Multiple seat pricing is negotiable. Offers free guest usage on cloud service.

DataRobot

Company Overview

DataRobot is a company headquartered in Boston, MA. The focus of the company is providing better accuracy, ease-of-use, speed and a comprehensive AutoML ecosystem. The product automates most stages of the model development lifecycle by using a combination of open source methods.

Product Overview

DataRobot – The DataRobot platform is accessible in the cloud or installed on-premises. It uses parallel processing to train and evaluate thousands of ML models in R, Python, H2O, Spark MLLib, Tensor Flow and other open source platforms. It can search through combinations of hyperparameters, algorithms, data cleaning and feature transformations to reach its final outcome.

The tool provides automation while providing customization options for advanced users through a GUI. For more advanced users the tool provides customization options like setting optimization parameters, data splitting, data visualization and feature exclusion. The tool combines models to develop ensembles. The final output includes the predictions, performance metric and their visualizations. The tool also provides a graphical representation of the process and reasoning for each prediction.

The performance of the tool can be changed by adjusting the number of available cores for parallel processing. The users are not allowed to edit the models built by the tool but users have an option to import their open source models and introduce them into the overall leaderboard for comparison.

Data In

The tool accepts data from Postgres SQL, Oracle, MySQL, MongoDB, ODBC, URL endpoints, HDFS and flat files like comma-delimited files. During data preparation, the tool detects outliers but does not automatically remove them. It handles missing values for numeric features by replacing them with median values. For categorical features it replaces the missing value by adding a “missing value” category. Dimensionality reduction is performed using Principal Component Analysis.

Once the data cleaning is done the tool performs data transformations by using mathematical functions. New features are generated by exhaustive calculation of differences and ratios of all numeric variables in pairs. The best features are kept for further analysis. Feature selection is performed using Elastic Net. Visualizations for variable importance are also available. The tool is also capable of processing text data, extracting features from it and converting it into rectangular format to be used in models as predictors.

Learning Algorithms

The algorithms used in the tool are from various open source platforms. Hyperparameters are tuned using smart or brute force grid search. The default optimization metric or the one selected by user is used to rank the models on the leaderboard. The tool also handles cross-validation and testing of these algorithms. Currently binary classification, multi-class classification and regression problems are supported.

The tool builds two kinds of ensemble models. Default is an ensemble of top three individual models and second option is a more advanced ensemble of top eight individual models. At the end of modeling cycle various graphs and common metrics like ROC, Lift Chart, F-1 Score and Confusion Matrix are available for evaluation.

Model Deployment Options

1. First is to upload the new dataset into the tool and score using the GUI.
2. Deploys the final model as web service in the DataRobot cloud. The web service will return the final results of the model to the program that initiated the call.

The tool has option to download model logic code as Java or Python programs. This option is only available to customers of DataRobot Prime.

DataRobot Evaluation Details

AutoML Tool Evaluation Criteria		DataRobot	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Both	Option of private data robot cloud is available. On-premise installation option is also available. Supported cloud vendores are Azure and Cloudera.
Data Connection	*Databases*	Yes	Multiple database connection options like are available including postgres sql, oracle, mysql, mongodb and any ODBC supported database.
	Unstructured data	Yes	Supports HDFS connectivity. Data can also be read through a URL.
	Flat file	Yes	Supports csv format.
Data Exploration	*Summary Statistics*	Yes	Automatically generates basic summary statistics like data types, number of unique values, number of missing values, mean, std dev, etc.
	Visualization	Yes	Automatically generates visualizations of variable distributions.
Data Cleaning	*Outlier Detection*	Yes	Automatically detects outliers but does not remove them. We can do that manually. Good visualization of outliers present.
	Missing data imputation	Yes	Automatically handles missing values for numeric features. Replaces them using median.
	Dimensionality Reduction	Yes	Has option of PCA for dimensionality reduction. This depends on the blueprint (workflow) selected.
Data Transformation	*Types of transformations*	Yes	Automatically applies basic data transformations. Some simple customized transformations (based on a mathematical equation) can be applied by the user.
	Feature Engineering	Yes	Produces new features and keeps relevant ones. All combinations of differences and ratio of numeric variables are calculated to create new features.
Feature Selection		Yes	Feature Selection done using elastic net. Variable importance visualization is generated. Manual feature selection is also allowed.
Machine Learning Problem categories	*Classification*	Yes	
	Regression	Yes	
	Clustering	No	
	Anomaly Detection	No	In Beta for future release.
	Time Series	No	
Automated Hyperparameter Tuning		Yes	Automated hyper-parameter tuning is performed using brute force search or smart grid search.
Modeling Tournament		Yes	Uses multiple standard open-source algorithms.
Model Ensemble Testing		Yes	Two types of ensembles are created. Default: ensemble of top 3 individual models. Advanced: ensemble of top 8 individual models.
Model Evaluation/ Performance Metrics		Yes	Default selection present but can be changed. Metric like ROC, lift, confusion matrix, F1score optimized threshold are generated. Visualization of model performance is also available.
Model Deployment Options	*GUI*	Yes	Batch Scoring- Drag n drop interface for scoring new data in the tool.
	Web Service/restAPI	Yes	API based deployment to score new data using a web service call.
	Code Deployment	Yes	DataRobot Prime (additional service) – provides Java/Python scoring code as output.
	Run-time Embedded	No	
Visualization/Reporting Analysis		Yes	Provides workflow infographic of whole process, explains features used and reason behind each prediction why it was made. Provides a leader board (like Kaggle) for comparing models.
Additional Comments		Yes	Users cannot edit model code for DataRobot models. They can build their own model in R/Python and train it in DataRobot and include that to compete with DataRobot models. Parameters can be customized at every stage (model algorithm, data split, text processing, feature selection).
White Labelling/Commercial		No	
Pricing		Yes	Med-High (priced on a seat/year basis) .

H2O

Company Overview

H2O.ai is a Silicon Valley based company that offers an open-source machine learning platform called H2O. This platform combines the advanced algorithms and scalable in-memory processing to provide one of the more widely used open source platforms in machine learning. After this evaluation started, the company launched a new AutoML product called "Driverless AI" focusing on AutoML. This evaluation focused only on the H2O.AutoML function of their open-source offering H2O.

Product Overview

H2O– H2O.AutoML can be installed on a local PC, server or cloud platform. It includes libraries and packages for popular data science languages like R and Python. Users can use these libraries to include H2O capabilities within their existing environment and code. Another option is to use the GUI of H2O known as "Flow". The GUI works through a web browser by accessing the localhost port.

The H2O.AutoML is a subset of H2O. Most common data science tasks exist outside of this function. Once the user imports data into H2O, missing value imputation, correlations and other tasks need to be performed before starting. It provides basic summary statistics, hyperparameter tuning and algorithm selection. To begin model training the user need to specify the training data; the target column; columns to be excluded; parameters for validation split (optional); and then change default optimization parameters as well asselect a time limit for training. Then the user can start the training process and monitor a model leaderboard in real-time. The output of the function is a list of trained models ranked on the optimization metric. The function also provides variable importance and performance graphs as part of the training output.

Data In

H2O supports data import from JDBC databases including PostgreSQL, Netezza and MySQL, HDFS and Spark, and reads Amazon S3 containers. For flat files it supports csv, xlsx, svmlight, parquet and gzip formats. For databases not directly supported, a workaround is to use in R/Python and a conversion function provided by H2O to import data. There is no automated data cleansing, data transformations or feature engineering.

Learning Algorithms

H2O.AutoML supports the following algorithms: Gradient Boosting Machines, Random Forest, Extremely Random Forest, Deep Neural Networks, Logistic Regression, Linear Regression and Stacked Ensembles. A grid search is used to select the best hyperparameters for each algorithm. A default optimization metric, or one selected by the user, is used to rank the models.

H2O AutoML builds two types of ensemble models. The first ensemble contains all individual models. The second ensemble contains only the best performing models from each algorithm class. At the end of modeling cycle various performance metrics like accuracy, confusion matrix, root mean square error, mean absolute error are available to user for evaluation.

Model Deployment Options

1. Upload the scoring data in the environment through R/Python or Flow GUI, then use the trained model for scoring.
2. For productionizing models there are two options based on a Java Object: POJO (Plain Old Java Object) and MOJO (Model Object Optimized). H2O-generated MOJO and POJO models are intended to be easily embeddable in any Java environment. The Java app that embeds these model objects can be further deployed as a REST API. The only difference between POJO and MOJO is that POJO does not support source files more than 1Gb. MOJO is optimized version which supports larger source files.

H2O Evaluation Details

AutoML Tool Evaluation Criteria		H2O	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Yes	Both cloud and on-premises options are available.
Data Connection	*Databases*	Yes	Multiple database connection options are available including JDBC databases (MySQL, PostgreSQL, Netezza) and Amazon S3. You can import data using R/python and convert to H2O data.
	Unstructured data	Yes	Supports a data connection to HDFS/Spark.
	Flat file	Yes	Supports csv, xlsx, parquet, svmlight and gzip formats.
Data Exploration	*Summary Statistics*	Yes	Very basic. Automatically generates basic summary statisitcs like total number of rows with missing values, mean, max and min skewness, total columns with NA's.
	Visualization	No	Not automated
Data Cleaning	*Outlier Detection*	No	
	Missing data imputation	No	Not automated
	Dimensionality Reduction	No	
Data Transformation	*Types of transformations*	No	Not automated
	Feature Engineering	No	
Feature Selection		No	Not automated. We can select columns that should not be considerd in modeling. Variable importance is shown as part of model summary.
Machine Learning Problem categories	*Classification*	Yes	Algorithms available are GBM, Random Forest, Extremely-Randomized Forests, Deep Neural Nets, Stacked Ensembles.
	Regression	Yes	Algorithms available are GBM, Random Forest, Extremely-Randomized Forests, Deep Neural Nets, Stacked Ensembles
	Clustering	No	
	Anomaly Detection	No	
	Time Series	No	
Automated Hyperparameter Tuning		Yes	Automated hyper-parameter tuning is performed using grid search.
Modeling Tournament		No	
Model Ensemble Testing		Yes	Two types. First contains all models. Second contains best performing models from each algorithm class.
Model Evaluation/ Performance Metrics		Yes	Defaults is 'Auto' but various metrics like rmse, mae, auc, logloss etc can be selected. Users can also select stopping tolerance for training. Mode performance metrics for classification are AUC, for multi-classification is mean per-class and fpr regression is deviance.
Model Deployment Options	*GUI*	Yes	
	Web Service/restAPI	No	
	Code Deployment	Yes	
	Run-time Embedded	No	
Visualization/Reporting Analysis		Yes	Limited visualization is provided as part of model summary.
Additional Comments			
White Labelling/Commercial		Yes	Open-Source
Pricing		No	Open-Source

Compellon

Company Overview

Compellon is a California based company focused on AutoML. The company takes a different approach as compared to other methods under review. The tool goes beyond predictive analytics and includes some prescriptive analytics capabilities.

Product Overview

Compellon - The tool has no on-premises installation option. It is only available through a cloud service. The major portion of logic used in the tool's functionality is hidden, making it friendly to business focused users and less accommodating technically focused data scientists. The tool has no fixed model structure in the traditional sense. The model structure is inferred from data. It focuses on finding causation in data with respect to target, resulting in a model graph structure. The GUI uniquely shows direct actionable impacts of data changes in the model and why each prediction was made. The "Advisor" and "What-If" functions help the user understand hypothetical scenarios and possible related actions. For example, user inputs they want to improve propensity by X% and tool suggests which variables need to be changed and by how much.

Data In

The tool supports ODBC connectivity as an input source. It also supports data import from HDFS and web URL's. For flat files it supports csv, tsv and zip formats. After the data is imported the user receives details on the data including types, summary statistics and automatically excludes columns with zero or hundred percent variability. Visualizations for distributions of each column are provided.

Data exploration tasks like outlier detection and dimensionality reduction are not required. The user has the option to impute missing values of numeric variables with mean or median but there is no option to handle missing values for categorical variables. The tool does not perform any kind of data transformation, scaling or feature engineering because these functions are not required for the tools AutoML functionality. If the user wishes to include data transformations they will need to be performed outside the tool before data is imported. The user does get an option to force data exclusions and as a part of the modeling process the tool identifies a relevant set of features providing a feature relationship graph. The tool can also analyze text data to get word polarities.

Learning Algorithms

The tool does not use any traditional machine learning algorithms. The training algorithm is proprietary, and there is no clear indication of what approach is being used. As a result, there is no concept of hyperparameter tuning or ensemble models. One disadvantage of the tool is that it only supports binary classification problems. There is no direct functionality for regression and multi-class classification problems.

Model Deployment Options

1. Scores directly inside the tool.
2. Deploys the model as a web service and scores through an API endpoint.
3. A jar file output for the final model can be used in a Java app for scoring.

Compellon Evaluation Details

AutoML Tool Evaluation Criteria		Compellon	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Cloud	No on-premise option available.
Data Connection	*Databases*	Yes	Includes ODBC connector for database connectivity.
	Unstructured data	Yes	Supports Hive. Users can read in data from URL.
	Flat file	Yes	Supports csv,tsv and zip formats.
Data Exploration	*Summary Statistics*	Yes	Automatically generates basic summary statisitcs like data types, number of missing values, number of unique values, max, min, mean, std dev, automatically excludes variables with 0% variance or 100% variance.
	Visualization	Yes	Automatically generates visualizations of variable distributions.
Data Cleaning	*Outlier Detection*	No	The tool does not require any kind of exploratory anaysis or data preparation.
	Missing data imputation	Yes	Users can select to impute missing numeric values with mean/median but no imputation for categorical values.
	Dimensionality Reduction	No	
Data Transformation	*Types of transformations*	No	The tool does not require data transformations and scaling. But if we explicitly need to transform a variable we need to do it before ingesting data in the tool.
	Feature Engineering	No	The tool does not need feature engineering. The whole concept is based on infering model structure from existing data.
Feature Selection		Yes	Users can exclude features from being considered in modeling. The tool automatically identifies relevant set of variables and their relationships using a relationship graph.
Machine Learning Problem categories	*Classification*	Yes	
	Regression	Yes	
	Clustering	No	Functionality exists. Will be part of tool in Q3-Q4 2018.
	Anomaly Detection	No	
	Time Series	No	
Automated Hyperparameter Tuning		No	No concept of hyper-parameter tuning in the traditional sense as it does not use traditional ML algorithms.
Modeling Tournament		Yes	
Model Ensemble Testing		Yes	
Model Evaluation/ Performance Metrics		Yes	Output is a model structure graph. The tool has UI to integrate model results with actions (Advisor and What-If functions). For example: we input improve propensity by X% and tool suggests which variables need to be changed and by how much. This also explains the reason why each prediction was made.
Model Deployment Options	*GUI*	Yes	
	Web Service/restAPI	Yes	Models can be used through an API call.
	Code Deployment	Yes	Users can get a jar file output from the tool.
	Run-time Embedded	No	
Visualization/Reporting Analysis		Yes	Visualization is generated but not exportable.
Additional Comments			Compellon is built for non-data scientists and focuses more on interpretability. Compellon is GUI based product available through a cloud service. It uses a proprietary AI platform that does not rely on any of the known statistical modeling algorithms. The prescriptive part of the tool consists of 'Advisor' and 'What-If' functions. This functionality helps the user understand hypothetical scenarios and possible actions related to them.
White Labelling/Commercial		No	
Pricing		Yes	Low - Priced as one-time fee per user.

MLJAR

Company Overview

MLJAR is a company based in Poland. The company offers an AutoML platform that focuses on enabling rapid prototyping, development and deployment of machine learning models.

Product Overview

MLJAR – The platform is available through a cloud service and as an on-premises installation. It automates selecting the best algorithms and hyperparameter tuning. All computations run in parallel. There are two different interfaces for users to interact with the platform. One is through a web-browser using the platforms GUI. The second is through R/Python code by using the MLJAR API. The two options focus on two distinctive classes of users respectively, business users/analysts, and data scientists with a more technical background.

The platform provides selections for model optimization metrics. It uses a heuristic tuning method which at first starts with models with random parameters then, based on their performance, tries to improve the next iteration. There are four differentiated modes of tuning based on number of models built for each algorithm.

Data In

The platform accepts data in .csv file format only. It requires the input data file to have column headers. Once the data is imported the user is provided with basic information like data types, number of unique values, number of missing values, column mean and variable distribution visualization.

Missing values for numeric features are imputed using mean, median or minimum value minus 1. For categorical features the missing values are replaced with most frequently occurring value. The platform provides one-hot encoding to convert categorical features into continuous features. It does not perform any automated data transformations. The user gets an option to forcefully include or exclude any feature from the modeling process. As part of modeling results for some algorithms feature importance is provided.

Learning Algorithms

The platform supports binary classification and regression problems. For binary classification it uses Xgboost, LightGBM, Random Forest, Extra Trees, Regularized Greedy Forest, k-Nearest Neighbors, Logistic Regression and Neural Networks. For regression it uses Xgboost, LightGBM, Random Forest, Extra Trees and Regularized Greedy Forest. For ensembles it uses the average method, which does a greedy search over all modeling results trying to add models to the ensemble that improve performance. The ensemble performance is computed based on out-of-folds predictions. The final output includes model performance statistics and visualization.

Model Deployment Options

1. Scores new data using the web browser base GUI.
2. Deploys the model as web service and scores new data using the API endpoint.

MLJAR Product Details

AutoML Tool Evaluation Criteria		MLJAR	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Both	Both Cloud based and on-premises installation options are available.
Data Connection	*Databases*	No	
	Unstructured data	No	
	Flat file	Yes	Only CSV files are accepted as input.
Data Exploration	*Summary Statistics*	Yes	Summary statistics like data types, number of unique and missing values, mean, max, min etc are generated automatically.
	Visualization	Yes	Generates visualization for variable distributions. Can add a function to export the summary visuals and stats.
Data Cleaning	*Outlier Detection*	No	
	Missing data imputation	Yes	Multiple options are avaliable for missing value imputation. Values can be replaced with median, mean, min value-1 for numeric variables and for categorical variables replaced with the most frequently occuring value.
	Dimensionality Reduction	No	
Data Transformation	*Types of transformations*	No	
	Feature Engineering	Yes	The tool has an option for feature engineering.
Feature Selection		Yes	Users can select which features to use and which to avoid. Shows feature importance for some algorithms.
Machine Learning Problem categories	*Classification*	Yes	Does not support multi-classification problems. Algorithms used are XGBoost, LightGBM, RF, Regularized Greedy Forest, k-nearest neighbors, Logistic regression, Neural Networks are available.
	Regression	Yes	Algortihms used for regression are XGBoost, LightGBM, RF, and Regularized Greedy Forest.
	Clustering	No	
	Anomaly Detection	No	
	Time Series	No	
Automated Hyperparameter Tuning		Yes	The tool includes automated hyperparamter tuning using a heuristic tuning method. It starts with random parameters and based on performance, improves the parameters each iteration.
Modeling Tournament		Yes	
Model Ensemble Testing		Yes	The tool builds ensembles using ensemble average method. It perfoms greedy search over all results and trys to add a model to the ensemble to improve performance.
Model Evaluation/ Performance Metrics		Yes	The user has an option to select which parameter to optimize. It provides statistics and visualization for model performance.
Model Deployment Options	*GUI*	Yes	Web browser based UI for training and scoring.
	Web Service/restAPI	Yes	Users have an option for API based model deployment. The tool also has an R/Python interface through its API.
	Code Deployment	Yes	Model code can be downloaded to be used in R/Python for scoring.
	Run-time Embedded	No	
Visualization/Reporting Analysis		No	
Additional Comments			
White Labelling/Commercial			
Pricing		Yes	Low - Monthly fee per user.

Xpanse Analytics

Company Overview

Xpanse Analytics is a Dublin, Ireland based company offering an automated predictive analytics platform, which delivers models from raw, unprepared data in a small amount of time. The main focus of their offering is detailed and automated feature engineering capability.

Product Overview

Xpanse.ai – The product is a cloud based only offering. Users interact with the tool through an easy to use GUI. The interface enables users to handle data and perform modeling without having to know any deep technical details of algorithms.

Data In

The tool accepts .csv and delimited flat files as input. There is no option to directly connect to any relational database or HDFS. One differentiating point related to data import is that this tool can ingest multiple files and provides the user an option to join these files forming a whole new data table with features from multiple files.

A key unique capability of this tool is feature engineering. It performs automated feature creation using pre-existing features. Various transformations and aggregated functions are used to develop new features. The user can select how exhaustive feature engineering should be. As an example, if the input data contains transactional data in each row and there is a timestamp column present, the tool will automatically generate features like number of transactions in last 7 days, or in the last month. This saves the user time to look through the raw data and calculate new aggregate columns. The tool generates thousands of columns and uses feature selection to include only the best and limited features for modeling. The user has an option to use Xapanse.ai for the modeling process, or export the newly developed feature set and use it in some other modeling environment of their choice.

Learning Algorithms

At present tool is capable of handling only binary classification problems. It provides options of Random Forest, Decision Tree and Logistic Regression algorithms. The hyperparameters for these algorithms are automatically tuned using grid search. At the end of the modeling cycle the user gets model performance metrics and visualizations (e.g. ROC graph).

Model Deployment Options

1. Uploads the new data and scores it using the tool's GUI.
2. Exports SQL code from the tool. The SQL code is available for all three algorithms Random Forest, Decision Tree and Logistic Regression. This is particularly helpful for users who want to perform fast in-database scoring.

Xpanse Analytics Evaluation Details

AutoML Tool Evaluation Criteria		Xpanse Analytics	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Cloud	Currently supports cloud only. On-premises option is in development stage.
Data Connection	*Databases*	No	No database connectivity.
	Unstructured data	No	
	Flat file	Yes	Supports .csv and other delimited files.
Data Exploration	*Summary Statistics*	No	No summary statistics.
	Visualization	Yes	Generates visualization for variable distributions.
Data Cleaning	*Outlier Detection*	No	
	Missing data imputation	No	No imputation with new values, just marks values as missing.
	Dimensionality Reduction	No	
Data Transformation	*Types of transformations*	Yes	A variety of aggregate transformations are applied as part of feature engineering.
	Feature Engineering	Yes	The tool performs extensive automated feature engineering, creating new features from existing features and transformations. The existing features can be from different data files. Level of feature engineering can be selected. A structured table with all the features can be exported in a .CSV file.
Feature Selection		Yes	
Machine Learning Problem categories	*Classification*	Yes	The tool supports only binary classfication. Three algorithm options are available. Random Forest, Decision Tree and Logistic Regression.
	Regression	No	
	Clustering	No	
	Anomaly Detection	No	
	Time Series	No	
Automated Hyperparameter Tuning		Yes	The tool performs automated hyper-parameter tuning using grid search method.
Modeling Tournament			
Model Ensemble Testing		No	The tool only builds ensembles as part of Random Forest. No ensembles made from different algorithms.
Model Evaluation/ Performance Metrics		Yes	The tool generates model performance visualizations for ROC, Lift and Error.
Model Deployment Options	*GUI*	Yes	Can score using the tool's GUI.
	Web Service/restAPI	No	
	Code Deployment	Yes	Exports SQL code for the model. This is availble for all three algorithm options RF, DT and Logistic Regression. This is useful if the end goal is to use the model for in-database scoring.
	Run-time Embedded	No	
Visualization/Reporting Analysis		Yes	Model performance and variable distribution visualization are present but cannot be exported.
Additional Comments			One good point about the tool is it can ingest multiple data files. It joins them and creates a whole new data table with target variable, features from original data files and also newly created features.
White Labelling/Commercial			
Pricing		Yes	Low - Priced as user seats/month

Ople

Company Overview

Ople is a company located in Silicon Valley, CA whose focus is to make artificial intelligence easy, cheap and ubiquitous. The product aims at delivering high performance models in production with limited time, effort and technical expertise. The central point of the technology is a proprietary behavioral assimilation technology.

Product Overview

Ople.ai – Ople.ai is offered as a cloud service in AWS or Azure, or as an on-premises installation. The tool uses advanced machine learning to automate and optimize various tasks involved in the data science process. Users interact through an easy to use GUI.

Data In

The tool only accepts .csv files as input. No database connectivity is available. Once the data is imported users get very summary statistics about the data with limited visualization. There are no automated data transformations, outlier detection or dimensionality reduction. The tool is not capable of handling missing values in the datasets. No explicit feature engineering options are available. Users do get an option to select performance metric for model training.

Learning Algorithms

The tool takes a slightly different approach to model training. The first stage uses various open source algorithms like Decision Tree, K-Nearest Neighbor, Random Forest, Gradient Boosting, Vowpal Wabbit, logistic and linear regression etc. These algorithms are automatically trained and auto-tuning of hyperparameters is performed. Based on the performance metric selected all the models are ranked. This is followed by the second stage in which the deep learning Behavioral Assimilation system (BASS) captures the best aspects of each of the models from first stage. This information is used as metadata and combined to develop the final BASS model.

Model Deployment Options

It should be noted that all the deployment options only work for the final BASS models. None of the individual trained algorithms can be used as actual model output from the tool. There are 3 deployment options.

1. Uploading the new data and score using the GUI.
2. Using a web service for the final BASS model using an API endpoint for scoring.
3. Downloading the weights from the final model and using them to program a scoring function in any programming language.

Ople Evaluation Details

AutoML Tool Evaluation Criteria		Ople.ai	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Both	Currently available as SaaS on AWS. Also works on Azure. They can aslo install this on premises.
Data Connection	*Databases*	No	
	Unstructured data	No	
	Flat file	csv	Only accepts .csv as input.
Data Exploration	*Summary Statistics*	Yes	Generates very basic statistics.
	Visualization	Yes	Very basic.
Data Cleaning	*Outlier Detection*	No	
	Missing data imputation	No	
	Dimensionality Reduction	No	
Data Transformation	*Types of transformations*	No	
	Feature Engineering	No	
Feature Selection		Yes	
Machine Learning Problem categories	*Classification*	Yes	DT, KNN, RF, GBM, VW, logistic regession and other open source algoritms are available.
	Regression	Yes	DT, KNN, RF, GBM, VW, linear regression and other open source algorithms are available.
	Clustering	No	
	Anomaly Detection	No	
	Time Series	No	
Automated Hyperparameter Tuning		Yes	Automated hyperparameter tuning is performed for the open source algorithms.
Modeling Tournament		No	
Model Ensemble Testing		No	No traditional ensembles. Output is always one single model.
Model Evaluation/ Performance Metrics		Yes	All the commonly used performance metrics can be selected for optimization.
Model Deployment Options	*GUI*	Yes	
	Web Service/restAPI	Yes	The tool provides an option to deploy through an API endpoint for the BASS model.
	Code Deployment	No	No code output option but can download a weights file for the final model.
	Run-time Embedded	No	
Visualization/Reporting Analysis		No	
Additional Comments			The whole modeling approach is a bit different. First it uses open source algorithms to build individual models. Based on the performance metric selected it ranks the models and gathers the best part from each model. This information is used as metadata for the BASS model. BASS captures the best parts about each individual model and combines them to form one single final model. This is the actual output of the whole process. BASS uses simplified neural networks in the back end. The model has very fast response time for scoring.
White Labelling/Commercial		No	
Pricing		Yes	High - Priced as user/month with unlimited usage.

DMWay

Company Overview

DMWay is an Israel based company that offers an end-to-end solution that automates the analytical process making predictive analytics accessible to non-technical users. The company's offerings include an analytic engine, scoring engine and model maintenance solutions. The main focus of DMWay is to create precise predictive models at low cost and reduce the time to implementation.

Product Overview

DMWay - DMWay is offered as an on-premises software installation. There is no cloud offering available. There are options for single and multi-core processing to adjust scale and speed. The tool takes a simple approach to the whole modeling cycle and has a rich set of options for easy deployment in various environments. The users interact through a GUI which facilitates a seamless data-in model-out scenario.

Data In

The tool accepts data in various formats. It supports ODBC to connect with a database. It takes .csv and Rdata files as input. A data import function from HDFS is currently in development. After data import the tool provides a visualization for data distributions.

There are not enough options to deal with missing values. If the percentage of missing values is lower than some preset than those rows are simply ignored otherwise the missing values are replaced with a random notation like '999'. The tool has functionality that performs automated stepwise feature selection. At the end of modeling cycle a feature importance visual is also provided. Basic data transformations, binning and correlations are automatically calculated as part of data preparation and exploration.

Learning Algorithms

Although it is a good point about the tool that it is very simplistic in its approach, it limits the variety of algorithm options that users get to train their models. The tool only supports linear models for both binary classification and regression. It does not have any multi-classification algorithms but a work around is to build multiple models using the linear binary method. The tool does have an extra option to build models that predict the number of times an event occurs in any given time period. Automated hyperparameter tuning is performed but is not very relevant since training is limited to linear models. After the training users get basic performance metrics and ROC graph as part of the output. Users can also export reports to Excel and R.

Model Deployment Options

1. Importing the new data and score using the GUI.
2. Downloading Java code as output and hosings it on DMWay's scoring server. The model is then accessible through an API endpoint.
3. Downloading the model code in Java, R, Oracle SQL or MS SQL Server format. The SQL model codes are specifically useful if users want to do in-database scoring.

DMWay Evaluation Details

AutoML Tool Evaluation Criteria		DMWay	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Both	Only on-premises local installation is available.
Data Connection	*Databases*	Yes	There is ODBC database support.
	Unstructured data	No	HDFS connectivity is in development.
	Flat file	Yes	Supports input from .csv and Rdata formats.
Data Exploration	*Summary Statistics*	Yes	Basic variable distribution visualization is generated.
	Visualization	Yes	
Data Cleaning	*Outlier Detection*	Yes	
	Missing data imputation	Yes	If % of missing values is below a threshold then the tool discards them else replaces them with a random number like 999 which forms a new level.
	Dimensionality Reduction	No	
Data Transformation	*Types of transformations*	Yes	Basic data transformations, binning, correlations are generated.
	Feature Engineering	No	
Feature Selection		Yes	Stepwise feature selection is perfomred. Also post modeling a variable importance graph is generated.
Machine Learning Problem categories	*Classification*	Yes	Only binary classification. For multi-class you need to build multiple models. They only use linear models.
	Regression	Yes	Only linear models.
	Clustering	No	
	Anomaly Detection	No	
	Time Series	No	They do have an option which builds models to predict the number of times an event occurs in a given time period.
Automated Hyperparameter Tuning		Yes	As the tool only builds linear models, this is not very relevant.
Modeling Tournament		No	
Model Ensemble Testing		No	
Model Evaluation/ Performance Metrics		Yes	Basic performance metrics and ROC curve are generated after model training.
Model Deployment Options	*GUI*	Yes	
	Web Service/restAPI	Yes	The tool provides Java code as output which can be hosted on DMWay's Scoring Server. This is then accessible through an API endpoint.
	Code Deployment	Yes	Java code, R code, Oracle SQL code, MS SQL Server code. This is useful if the end goal is to incorporate predictive models in some software application or for in-database scoring.
	Run-time Embedded	No	
Visualization/Reporting Analysis		Yes	Can export reports to Excel and R.
Additional Comments			
White Labelling/Commercial		No	
Pricing		Yes	Med-High - Priced as an annual subscription service per user, with a separate charge for a scoring server.

Tazi

Company Overview

Tazi is an Istanbul, Turkey based company offering an AutoML tool with a unique capability for continuous learning. Tazi has been in the European market for a couple of years and recently established a presence in San Francisco.

Product Overview

Tazi.ai – The Tazi.ai tool is available as an on-premises installation and also as a cloud service in AWS. This is one of the few tools that focuses on automating most parts of the machine learning cycle. The tool can run multiple algorithms in parallel with fully automated hyperparameter tuning. The user interacts with the tool through a the tool's GUI.

The default operation of Tazi.ai is focused on automation, but it also provides enough customization options for more advanced users. The tool has an interactive visualization layer which explains new models in comparison to older ones. This is useful for users more comfortable on the business side of operations. They can interact with visualization and actually manipulate the current model structure based on their domain expertise.

Data In

Tazi.ai can read data from multiple sources. It connects to databases using JDBC support. It imports data from unstructured sources including web URL/http calls. The tool supports streaming data input through Kafka and collect data from Google Analytics. It also supports data import from flat files in .csv format. The user can combine multiple internal and external data sources within the tool. Once the data is imported the tool provides basic summary statistics for feature columns and visualizations for feature distributions. Outliers are handled through a human-in-loop feedback mechanism.

Missing data imputation can be performed with default values. For continuous features, missing values can be replaced by the continuous average. For categorical features missing values can be imputed by mode. Data transformation functions are applied to input columns. Transformations cannot be user defined but there is an built-in list of mathematical functions that can be used for automated data transformation. Tazi.ai performs automated feature engineering using combinations of already existing features. Automated feature selection is performed but can be over ruled by the user. The tool automatically selects an optimization metric but users also have an option to override the default selection.

Learning Algorithms

Tazi.ai can handle binary classification, multi-classification, regression, anomaly detections, time-series and segmentation problems. The tool runs multiple algorithms in parallel and presents the best model to the user. For anomaly detection it uses a combination of supervised, unsupervised and semi-supervised approaches. It uses ARIMA and LSTM algorithms for time series data. There was no information available for the types of algorithms used for classification and regression modeling.

A unique point about Tazi.ai is that it supports automated model retraining and online model retraining. This means the models keep on learning as they operate. The tool produces well explainable GDPR compliant models.

Model Deployment Options

1. Using as web service and API endpoint to score new data.
2. Outputting Scala code for use in an application for scoring.

Tazi.ai Evaluation Details

AutoML Tool Evaluation Criteria		Tazi.ai	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Both	On-premise and Amazon Cloud installation options available.
Data Connection	*Databases*	Yes	Database connectivity through JDBC. Multiple data sources can be joined in the tool. Also supports third party external data sources.
	Unstructured data	Yes	Can read in data through URL/http calls. Can also connect to Google Analytics. Also supports streaming data through Kafka.
	Flat file	Yes	Supports .csv format.
Data Exploration	*Summary Statistics*	Yes	Basic summary statistics for the imported data are generated.
	Visualization	Yes	Visualization of variable distributions are generated.
Data Cleaning	*Outlier Detection*	Yes	Handled through a feedback mechanism.
	Missing data imputation	Yes	Can be imputed with default values. For continous variables missing values are replaced with continuous average and for categorical variables missing values are imputed with mode.
	Dimensionality Reduction	Yes	
Data Transformation	*Types of transformations*	Yes	Data transformations are applied to input columns. At this point transformations cannot be user defined but there is a built-in list of mathematical transformations that can be used.
	Feature Engineering	Yes	Automated feature engineering performed to develop new features from previously exiting ones. These new features can be explained based on the transformations and features used.
Feature Selection		Yes	
Machine Learning Problem categories	*Classification*	Yes	Both binary and multi-classification are supported. At this time not much information available about the type and variety of algorithms used.
	Regression	Yes	At this time not much information available about the type and variety of algorithms used.
	Clustering	Yes	Segmentation based on specific business KPI's.
	Anomaly Detection	Yes	Using a combination of supervised, un-supervised and semi-supervised learning.
	Time Series	Yes	ARIMA type models, plus deep learning models like LSTM.
Automated Hyperparameter Tuning		Yes	
Modeling Tournament		Yes	
Model Ensemble Testing		Yes	
Model Evaluation/ Performance Metrics		Yes	Tool automatically selects the optimization metric but user has a chance to override that with their choice of metric for optimization. Most common metrics like accuracy, rmse, ROC, AUC are available.
Model Deployment Options	*GUI*	Yes	
	Web Service/restAPI	Yes	Models can be deployed and accessed through API endpoints.
	Code Deployment	No	
	Run-time Embedded	No	
Visualization/Reporting Analysis		No	
Additional Comments		Yes	This tool supports online retraining of models, meaning models keep learning as they operate. GDPR compliant models with explanations provided. Has option to include domain expert's knowledge through a feedback process.
White Labelling/Commercial		No	
Pricing		Yes	High - Priced on a user/year basis

NumberTheory

Company Overview

NumberTheory is a data science and machine learning focused company based out of India, with a presence in California. The company's mission is delivering a platform including all parts of a data science workflow from ETL to model deployment.

Product Overview

NumberTheory.ai - The tool is available as both a cloud service and an on-premises installation. The platform includes ETL focused modules and reinforcement learning. While there are several modules that perform automated functions this is not an AutoML tool at this point. NumberTheory is closer to tools like Azure ML Studio, SAS EM and IBM SPSS. The user needs to build the whole modeling workflow. This gives more control and customization options but requires the user to be well versed in modeling and data analytics.

The user interacts with the tool through a GUI which has a drag and drop interface to build modeling pipelines. The platform supports parallel processing with performance depending on hardware allocated. The platform also supports automated and scheduled model retraining. This means that trained models can keep on learning (self-healing) based on new data.

Data In

The platform supports a wide range of data connectivity options. It can read in data from Oracle, MySQL and MS SQL Server databases. It supports data connectivity to HDFS, Hive, Elastic Search, Graph DB Neo4j, Kafka for real time streaming data, MongoDB and Casandra. The user can also import data from flat files like .csv or other delimited formats.

Once the data is imported summary statistics are provided. There are modules that provide exploratory visualization capabilities like histograms and boxplots. Data exploration supports outlier detection. The platform allows the user to impute missing fields using standard statistics and has methods for variable reduction.

There is a separate module for data transformation and feature generation with an option to apply predefined or custom mathematical functions. It should be noted that this is not automated requiring user direction and input for execution. At this time the platform does not support automated feature selection.

Learning Algorithms

The platform provides algorithms for classification; regression; clustering; anomaly detection; time series modeling; collaborative filtering for recommendations; market basket analysis; and survival analysis. The tool performs automated hyperparameter tuning using grid search. The platform also includes algorithms like Multi-Layer Perceptron, Convolution Neural Networks, Page Rank and Triangle Counting for graph analysis, and Latent Dirichlet Allocation for NLP. All these options provide users with a high level of control. Users have to include these modules in the pipeline themselves as the tool does not run a set of algorithms automatically and give the best model as output.

Once the model training is done the platform provides performance metrics including AUC, Accuracy, Precision, Recall, RMSE and MAE. The user can build ensemble models. Another good capability of platform is that it supports explore-exploit based Reinforcement Learning.

Model Deployment Options

1. Uploads the new data into the tool and use the trained model for scoring using the GUI.
2. Deploys the model as web service and use the API endpoint for scoring.

NumberTheory.ai Evaluation Details

AutoML Tool Evaluation Criteria		NumberTheory.ai	
		Included	Evaluation Comments
Installation - Cloud/On-premise/Both		Both	Both cloud and on-premises offering available.
Data Connection	*Databases*	Yes	Supports Oracle, MySQL, MS SQL Server.
	Unstructured data	Yes	Supports HDFS, Image data, Hive, Elastic Search, Graph Data Neo4j, Kafka for real time streaming
	Flat file	Yes	Supports .csv and other delimited formats.
Data Exploration	*Summary Statistics*	Yes	Coefficient of Variation, Describe, Median, Quantiles, Distinct Value,Variance, Kurtosis, Skewness, Frequent items, Mean absolute deviation, Range, Detect Outliers, Histogram, BoxPlot, Correlation, Covariance
	Visualization	Yes	The user has options to look at the distribution of features.
Data Cleaning	*Outlier Detection*		Outlier detection is part of data exploration.
	Missing data imputation	Yes	For Continuous uses the Mean, Median, Mode, GroupMean and GroupMedian. For Categorical uses the Mode. For Complete data set supports user input value or deletes null rows.
	Dimensionality Reduction	Yes	There are 3 methods for dimensionality reduction present in platform - PCA, SVD and RBM.
Data Transformation	*Types of transformations*	Yes	Variable derivation - Where one can create new variables using various transformation and aggregation. Multi Expression - Standard operations and custom expression.
	Feature Engineering	Yes	Automated variable derivation is a complete sub module dedicated for new feature derivation.
Feature Selection		Yes	No, No Automated feature selection in current release. However, its part of product roadmap.
Machine Learning Problem categories	*Classification*	Yes	Random Forest, Gradient Boosting, Logistic Regression, Naïve Bayes, KNN, SVM, Decision Tree, FFM - Field-aware Factorization Machines.
	Regression	Yes	Linear Regression, GLM - generalized linear model, Random Forest, Gradient Boosting, Decision Tree.
	Clustering	Yes	KMeans, Bisecting- Kmeans, GMM - gaussian mixture model
	Anomaly Detection	Yes	SVM, KNN,
	Time Series	Yes	ARIMA, ARIMAX. They also have algorithms like ALS for recommendations, market basket analysis and survival analysis. Deep Learning options like CNN, RBM, NLP analysis, graph analysis options are also available.
Automated Hyperparameter Tuning		Yes	Automated Grid Search. Bayesian Optimization will come in next release.
Modeling Tournament		Yes	Reinforement Learning is part of Number theory Platform.
Model Ensemble Testing		Yes	
Model Evaluation/ Performance Metrics		Yes	AreaUnderRoc, Accuracy, Precision, Recall, AreaUnderPR, WSSE, fmeasureMicroAvg, fmeasureMacroAvg, PrecisionMicroAvg, PrecisionMacroAvg, RecallMicroAvg, RecallMacroAvg, RMSE, R2, MAE, Variance
Model Deployment Options	*GUI*	Yes	Options for bacth and real-time scoring are available.
	Web Service/restAPI	Yes	
	Code Deployment	No	
	Run-time Embedded	No	
Visualization/Reporting Analysis		Yes	Can also connect to visualization tools like Tableau and Power BI.
Additional Comments		Yes	The platform has a couple of unique capabilities. One is automated and scheduled model retraining. Second is reinforcement learning.
White Labelling/Commercial		No	
Pricing		Yes	Low-Med - Priced annually as user/year.