Better Predictions. Faster.

A Practical Guide to Automated Machine Learning

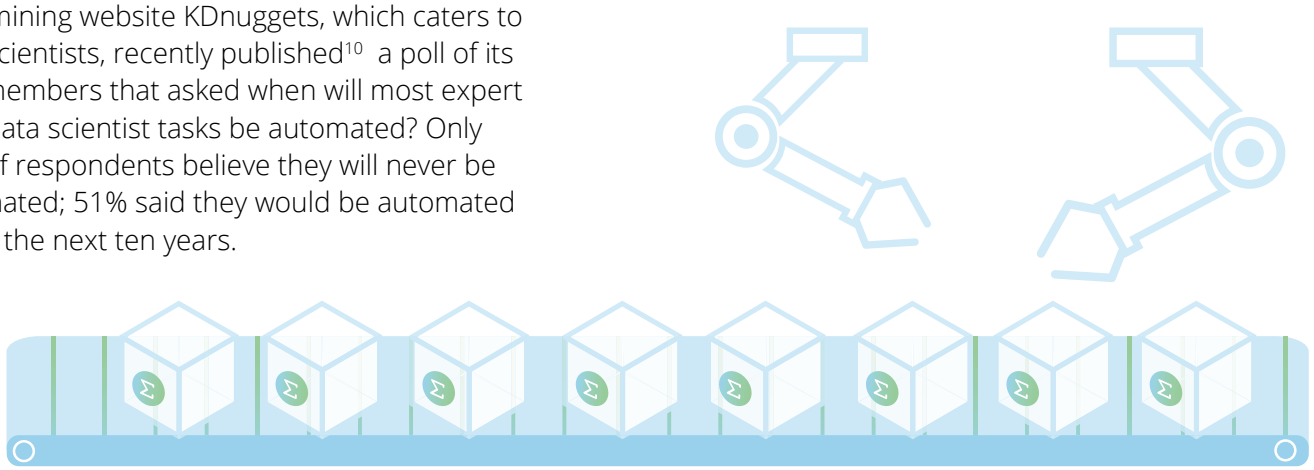
Data Science: A Scarce Resource.

Machine learning drives business advantage everywhere. With better predictions, banks reduce losses from credit defaults and fraud; insurers develop more competitive pricing; retailers personalize offers to consumers; hospitals improve patient outcomes; and telecommunications providers optimize bandwidth.

Businesses collect massive amounts of data, build data science departments, buy data science technology, and hire data scientists all to realize the power of machine learning. Hadoop and other technologies make the collection of data commonplace, but data scientists who make sense of it are still a scarce resource. VentureBeat[1], The Wall Street Journal[2], The Chicago Tribune[3] and many others all note the scarcity; a McKinsey report[4] projects a shortage of people with analytical skills through 2018. The scarcity is so pressing that Harvard Business Review suggests[5] that you stop looking, or lower your standards.

One way to alleviate the pressure, may be to automate some of the work data scientists currently perform. In IT Business Edge, Loraine Lawson wonders[6] if artificial intelligence will replace the data scientist. In a Sloan Management Review article headlined Data Scientist in a Can, Michael Fitzgerald argues[7] that companies are already trying to automate the function; however, he fails to distinguish between outsourcing analytics – which companies have done for years – and automating analytics, which is quite different. In Forbes, technology thought leader Gil Press confidently asserts[8] that the data scientist will be replaced by tools; Scott Hendrickson, Chief Data Scientist at social media integrator Gnip, agrees[9].

Data mining website KDnuggets, which caters to data scientists, recently published[10] a poll of its own members that asked when will most expert level data scientist tasks be automated? Only 19% of respondents believe they will never be automated; 51% said they would be automated within the next ten years.



1 http://venturebeat.com/2015/03/17/why-data-scientists-and-marketing-technologists-are-the-hottest-jobs-of-2015/
2 http://blogs.wsj.com/cio/2014/11/10/for-cios-universities-cant-train-data-scientists-fast-enough/
3 http://www.chicagotribune.com/business/ct-indeed-survey-0514-biz-20150514-story.html
4 http://www.mckinsey.com/features/big_data
5 https://hbr.org/2014/09/stop-searching-for-that-elusive-data-scientist/
6http://www.itbusinessedge.com/blogs/integration/will-artificial-intelligence-replace-the-data-scientist.html
7 http://sloanreview.mit.edu/article/data-scientist-in-a-can/
8 http://www.forbes.com/sites/gilpress/2012/08/31/the-data-scientist-will-be-replaced-by-tools/
9 https://blog.gnip.com/data-scientist-vs-data-tools/#
10 http://www.kdnuggets.com/polls/2015/analytics-data-science-automation-future.html

The March of Automation:

Evidence from other fields is encouraging.

- The Washington Post summarizes[11] successful efforts to automate anesthesia during surgery.
- The MIT Technology Review reports[12] on a machine learning algorithm that classifies paintings more accurately than trained art historians.
- A report from consulting firm A.T.Kearney projects[13] that automated "Robo Advisors" will run $2 trillion in investment portfolios by 2020.
- An article in The Atlantic notes[14] that nearly half of American jobs could be automated.

Automation is neither an all-or-nothing phenomenon, nor does it happen overnight. Consider the automobile, a highly evolved technology. In the era of the Model T Ford, driving was hard; only a fraction of the population could operate the vehicle, change a tire or troubleshoot frequent breakdowns. Over time, auto manufacturers simplified the driving process and made cars more reliable, expanding the pool of potential drivers.

Today, most adults can drive; mass-market cars are loaded with automated features: adaptive cruise control, lane-keeping systems, driver alert systems and blind spot indicators. Full automation is just over the horizon; earlier this year, the state of Nevada cleared[15] Freightliner's self-driving[16] truck for use on public highways.

Reflecting the incremental and progressive nature of automation, the National Highway Transportation Safety Administration's policy[17] on automated vehicles defines five levels of autonomy:

- Level 0: the driver is in complete command at all times.
- Level 1: one or more specific control functions automated.
- Level 2: at least two primary control functions automated to work together.
- Level 3: the driver can relinquish full control under certain conditions.
- Level 4: the vehicle performs all functions for the entire trip.

Suppose we apply a similar model to machine learning; our "levels of autonomy" look something like this:

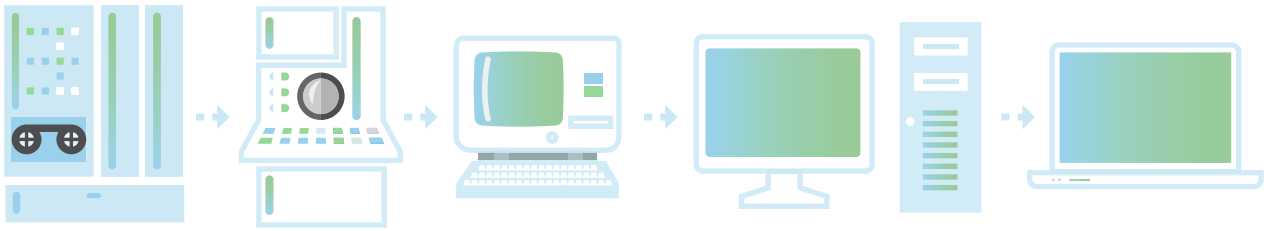
- Level 0: the analyst manually codes all steps in the modeling process.
- Level 1: the software automates one or more individual steps, such as sample splitting.
- Level 2: the software automates multiple steps, such as training and pruning a tree.
- Level 3: the analyst defines a test plan and the software executes the plan.
- Level 4: the analyst specifies a target and the software builds a production-ready model4.

This perspective reveals two key insights. First, before we can automate machine learning, we have to define the steps in the process and the desired outcomes – otherwise, we don't know what it is that we're automating. Second, it's clear that machine learning is already automated to a considerable degree. Even the most hardcore

11 https://www.washingtonpost.com/news/the-switch/wp/2015/05/15/one-anesthesiology-robot-dips-its-toes-into-whats-possible-this-one-jumps-all-in/
12 http://www.technologyreview.com/view/537366/the-machine-vision-algorithm-beating-art-historians-at-their-own-game/
13 http://www.bloomberg.com/news/articles/2015-06-18/robo-advisers-to-run-2-trillion-by-2020-if-this-model-is-right
14 http://www.theatlantic.com/business/archive/2014/01/what-jobs-will-the-robots-take/283239/
15 https://www.newscientist.com/article/dn27485-autonomous-truck-cleared-to-drive-on-us-roads-for-the-first-time/
16 http://www.ccjdigital.com/behind-the-wheel-of-freightliners-inspiration-autonomous-truck/
17 http://www.nhtsa.gov/About+NHTSA/Press+Releases/U.S.+Department+of+Transportation+Releases+Policy+on+Automated+Vehicle+Development

data scientists who work in programming languages use Integrated Development Environments (IDEs) that automate routine programming tasks. Tools like this are at least at the second level of automation.

Automated Machine Learning: A Brief History



Automated machine learning is not new. Before Unica launched its successful suite of marketing automation software, the company's primary business was machine learning, with a particular focus on neural networks. In 1995, Unica introduced Pattern Recognition Workbench (PRW), a software package that used automated trial and error to optimize model tuning for neural networks. Three years later, Unica partnered with Group 1 Software (now owned by Pitney Bowes) to market Model 1, a tool that automated model selection over four different types of predictive models. Rebranded several times, the original PRW product remains as IBM PredictiveInsight, a set of wizards sold as part of IBM's Enterprise Marketing Management suite.

Two other commercial attempts at automated machine learning date from the late 1990s. The first, MarketSwitch[18], consisted of a solution for marketing offer optimization, which included an embedded "automated" machine learning capability. In sales presentations, MarketSwitch bragged that it had hired former Soviet rocket scientists to develop its software, and promised customers they would be able to "fire their SAS programmers". Experian acquired MarketSwitch in 2004, repositioned the product as a decision engine and replaced the "automated machine learning" capability with its own outsourced analytic services.

KXEN, a company founded in France in 1998, built its machine learning engine around an automated model selection technique called structural risk minimization[19]. The original product had a rudimentary user interface, depending instead on API calls from partner applications; more recently, KXEN repositioned itself as an easy-to-use solution for Marketing analytics, which it attempted to sell directly to C-level executives. This effort was modestly successful, leading to sale of the company in 2013 to SAP for an estimated[20] $40 million.

Early efforts at automation from Unica, MarketSwitch and KXEN failed to make an impact for two reasons. First, they "solved" the problem by defining it narrowly; limiting the scope of the solution search to a few algorithms, they minimized the engineering effort at the expense of model quality and robustness. Second, by positioning their tools as a means to eliminate the need for expert analysts, they alienated the few people in customer organizations who understood the product well enough to serve as champions.

In the last several years, the leading analytic software vendors (SAS and IBM SPSS) have added automated modeling features to their high-end products. In 2010, SAS introduced SAS Rapid Modeler[21], an add-in to SAS

18 http://www.experian.com/decision-analytics/marketswitch-optimization.html
19 http://www.svms.org/srm/
20 https://451research.com/report-short?entityId=79713
21 https://support.sas.com/resources/papers/proceedings10/113-2010.pdf

Enterprise Miner. Rapid Modeler is a set of macros implementing heuristics that handle tasks such as outlier identification, missing value treatment, variable selection and model selection.

The user specifies a data set and response measure; Rapid Modeler determines whether the response is continuous or categorical, and uses this information together with other diagnostics to test a range of modeling techniques. The user can control the scope of techniques to test by selecting basic, intermediate or advanced methods. In 2015, SAS introduced a new generation of this product branded as SAS Factory Miner.

IBM SPSS Modeler includes a set of automated data preparation features as well as Auto Classifier, Auto Cluster and Auto Numeric nodes. The automated data preparation features perform such tasks as missing value imputation, outlier handling, date and time preparation, basic value screening, binning and variable recasting. The three modeling nodes enable the user to specify techniques to be included in the test plan, specify model selection rules and set limits on model training.

All of the software discussed so far is commercially licensed. The caret[22] package in open source R includes a suite of productivity tools designed to accelerate model specification and tuning for a wide range of techniques. The package includes pre-processing tools to support tasks such as dummy coding, detecting zero variance predictors, identifying correlated predictors as well as tools to support model training and tuning.

The training function in caret currently supports 192 different modeling techniques; it supports parameter optimization within a selected technique, but does not optimize across techniques. To implement a test plan with multiple modeling techniques, the user must write an R script to run the required training tasks and capture the results.

Auto-WEKA[23] is another open source project for automated machine learning. First released in 2013, Auto-WEKA is a collaborative project driven by four researchers at the University of British Columbia and Freiburg University. In its current release, Auto-WEKA supports classification problems only. The software selects a learning algorithm from 39 available algorithms, including 2 ensemble methods, 10 meta-methods and 27 base classifiers. Since each classifier has many possible parameter settings, the search space is very large; the developers use Bayesian optimization to solve this problem[24].

Challenges in Machine Learning (CHALEARN)[25] is a tax-exempt organization supported by the National Science Foundation and commercial sponsors. CHALEARN organizes the annual AutoML[26] challenge, which seeks to build software that automates machine learning for regression and classification. The most recent conference[27], held in Lille, France in July, 2015, included presentations[28] featuring recent developments in automated machine learning, plus a hackathon.

22 http://topepo.github.io/caret/index.html
23 http://www.cs.ubc.ca/labs/beta/Projects/autoweka/#papers
24 http://www.cs.ubc.ca/labs/beta/Projects/autoweka/papers/autoweka.pdf
25 http://www.chalearn.org
26 http://automl.chalearn.org
27 https://sites.google.com/site/automlwsicml15/
28 https://indico.lal.in2p3.fr/event/2914/

Designing an Automated Machine Learning Platform.

Requirements for a modern automated machine learning platform fall into two categories: support for the machine learning process, and support for enterprise computing.

Automated machine learning software should support the machine learning process from beginning to end:

- Rapid Data Ingestion. Data scientists need to leverage data from across the organization and from external sources. For speedy data ingestion, automated machine learning software should support open-standards based interfaces with relational databases and Hadoop, as well as text files and common desktop formats.

- Automated Data Preparation. Data scientists use expert judgment to examine raw data and make decisions about how to work with it in a predictive model. Automated machine learning software should perform this diagnosis and present the results in clear and concise visuals.

  Data scientists routinely split raw data into learning, validation and holdout samples so they can validate models and protect against overfitting. Automated machine learning software should perform this step for the user and protect the holdout sample.

- Automated Test Planning. There are hundreds of potential algorithms; a recent benchmark study tested[29] 179 for classification alone. The best way to determine the right algorithm for a given problem and data set is a test and learn approach, where the data scientist tests a large number of techniques and chooses the one that works best on fresh data. (The No Free Lunch No Free Lunch[30] theorem formalizes this concept).

  Using information about the target and predictor variables, automated machine learning software should select the most appropriate techniques from the available universe, generate a test plan and execute the plan.

- Automated Feature Engineering. Each machine learning technique has a distinct way that it handles data, which may require pre-processing before model training can begin. Also, data scientists use best practices in feature engineering to prepare data for better results. Automated machine learning software should incorporate and use this expertise

- High-Performance Model Training. Model training is computationally intensive. Moreover, even with heuristics and bootstrapping, a comprehensive experimental design may require thousands of

29 http://jmlr.org/papers/v15/delgado14a.html
30 http://www.no-free-lunch.org

model train-and-test cycles. Automated machine learning software should leverage state-of-the art computing for high performance and rapid learning.

- Transparent Model Evaluation. For "hard money" predictive models that have strategic implications for a company, no executive will approve deployment without first understanding the model's behavior and validity.

  Automated machine learning software should provide tools so that expert and business users can evaluate the results of a modeling experiment, check for bias, compare models and, if appropriate, override the automatic model selection.

- Real-Time and Batch Deployment. The greatest predictive model in the world is worthless if it is not used. That is what happened with the Netflix Prize winner[31], a sophisticated application of a technique called Pragmatic Chaos that now resides in the dustbin of history; Netflix paid out the prize and buried[32] the solution because it was too expensive to deploy. Automated machine learning software should include scalable engines for both batch scoring and for real-time predictions.

To serve the needs of the modern enterprise, there are three additional requirements:

- Open Source Analytics. Automated machine learning software should build on a foundation of open source software. The cadence of innovation in open source analytic languages, such as Python and R, is much faster than in commercial software. Moreover, an open source foundation simplifies integration with Big Data stacks and reduces cost of ownership.

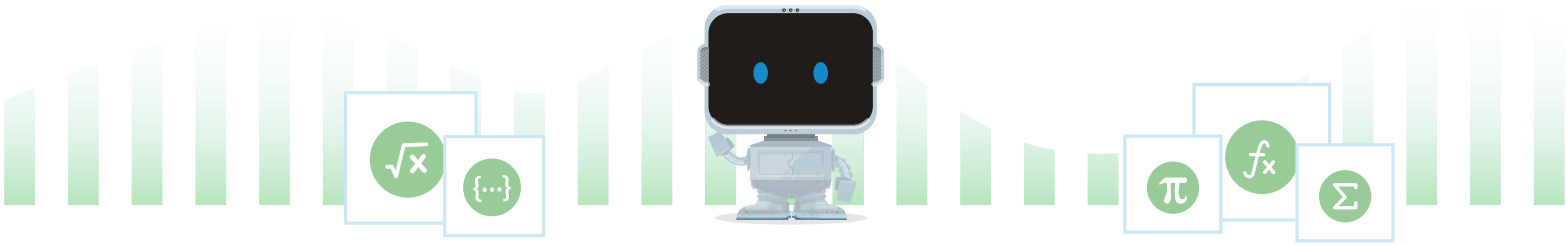
- Business and Expert User Interfaces. Automated machine learning software should support diverse user personas, including:

  - Expert users who want to write custom code.
  - Analysts with deep statistical training but limited programming skills.
  - Sophisticated business users who want to engage with the model development process.
  - Business users who want to visualize the characteristics of a model and how it behaves.

- Enterprise Scalability. Automated machine learning software must scale to the enterprise level, on many dimensions, measured by users, projects, models and data volume. In practical terms, this means that the software should support deployment in Hadoop, standards-based integration with databases and support for low-maintenance provisioning: on premises or in the cloud.

  Any automation is better than no automation. But for a truly agile and automated workflow, machine learning software should meet all of these requirements.

31 http://www.wired.com/2009/09/how-the-netflix-prize-was-won/
32 http://thenextweb.com/media/2012/04/13/remember-netflixs-1m-algorithm-contest-well-heres-why-it-didnt-use-the-winning-entry/



DataRobot, a data science and machine learning company located in Boston, Massachusetts, offers a platform for users of all skill levels to build and deploy accurate predictive models in a fraction of the time needed using conventional tools and methods.

DataRobot uses massively parallel processing to train and evaluate thousands of models in R, Python, Spark MLlib, H2O and other open source libraries. It searches through millions of possible combinations of algorithms, pre-processing steps, features, transformations and tuning parameters to deliver the best models for your dataset and prediction target.

How does DataRobot stack up against the requirements for automated machine learning software outlined earlier in this chapter? Let's review:

- Rapid Data Ingestion. DataRobot loads data from text files, from URLs, through ODBC connections and from HDFS. You can deploy DataRobot in Hadoop, so your data never leaves HDFS.

- Automated Data Preparation. When DataRobot reads data, it automatically detects duplicate variables and variables with no information value – such as blanks and constants. DataRobot profiles target and predictor variables, and uses this information later in the process to build optimal test plans. DataRobot automatically partitions your input dataset into learning, validation and holdout dataset. It locks the holdout dataset, so that users do not accidentally use it prior to final model selection.

- Automated Test Planning. With built-in expertise, DataRobot uses information about your target variable and predictors to define a list of models to test.

- Automated Feature Engineering. DataRobot builds feature engineering into the modeling test plan, so it automatically preprocesses data for best results with the technique to be tested. At the most basic level, DataRobot actually tests pipelines, or "blueprints" consisting of multiple processing steps defined by a team of world-class data scientists.

- High-Performance Model Training. DataRobot's modern software architecture builds on the most current tools and methods to enable rapid parallel execution of large scale modeling experiments.

- Transparent Model Evaluation. Once DataRobot has trained and tested a battery of models, users can work with a number of tools to assess accuracy, understand how the model behaves and identify possible

bias. Users can evaluate model accuracy with many different metrics including AUC, Log-Loss and RMSE. DataRobot uniquely provides partial dependency plots, a technique that helps users visualize how individual variables affect the prediction, a vital tool in areas such as credit risk, where bias can seriously affect the value of a predictive model.

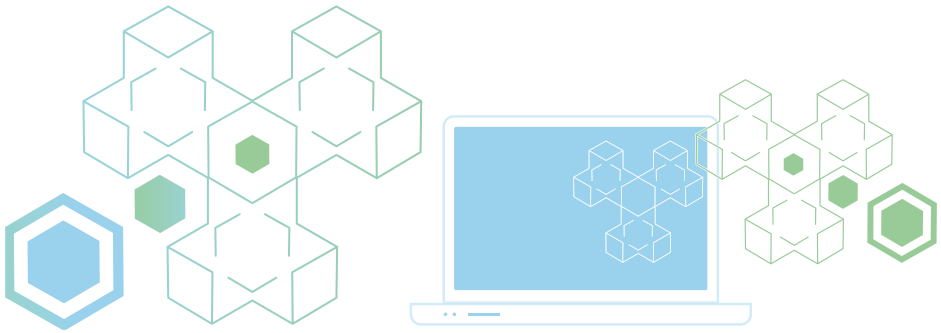
- Real-Time and Batch Deployment. DataRobot offers a real-time prediction engine and a batch prediction engine. The real-time prediction engine is designed so that organizations can implement multiple instances with load balancing to achieve a desired level of speed and throughput. The batch prediction engine runs in Spark, and it can run either in a freestanding Spark instance or it can be co-located with Hadoop.

- Open Source Analytics. DataRobot builds on R, Python, H2O, Spark and XGBoost for machine learning techniques, and open source technologies such as Docker, Hadoop, Spark, GlusterFS, Redis and MongoDB for implementation.

- Business and Expert User Interfaces. For the business user, DataRobot offers an easy-to-use web interface complete with visualization tools to evaluate predictive models. For the expert user, DataRobot offers R and Python APIs for modeling and separate APIs for prediction. Developers can build production-ready processes that call the prediction API to incorporate the power of predictive modeling into applications.

- Enterprise Scalability. DataRobot scales out, not up. Organizations deploy DataRobot on premises in free-standing clusters or in Hadoop, or in the cloud. DataRobot scales easily as usage grows; all components support distributed deployment across multiple virtual or physical machines. DataRobot distributes and manages workloads across the datacenter resources allocated to it.

Implications for Data Science.



The power and impacts of automation on business, and the profession of data science is undeniable. Organizations that embrace this new technology will have a sustainable competitive advantage, and now that the automated machine learning 'genie is out of the bottle' time is of the essence. Those that move fastest, will gain the best advantage.

Automating many of the processes associated with machine learning and predictive analytics empowers the data scientist to shift focus from routine coding and data wrangling to tasks that add real value: understanding the business problem, the deployment context and explaining results. Automation also changes the mix of people engaged in the predictive analytics process, with more focus on business skills and domain knowledge than pure coding skills.

Applying automation to data science is a transformative process, and like any form of digital transformation, should be undertaken in stages. Start by identifying manageable yet impactful initial projects, develop focused small teams, move and learn quickly, celebrate success, and scale as organizational skill and business needs dictate.

We can't eliminate the need for human expertise from predictive analytics, for the same reason that robotic surgery does not eliminate the need for surgeons. Someone has to understand the business problem and explain the results of analysis to executives; models don't explain themselves. We can build tools that improve the productivity of data scientists and help them build better models.