ScienceDirect

Contents lists available at **sciencedirect.com**
Journal homepage: **www.elsevier.com/locate/jval**

Methodology

Machine Learning for Health Services Researchers

Patrick Doupe, PhD,[1,*] James Faghmous, PhD,[2] Sanjay Basu, MD, PhD[3,4]

[1]Zalando SE, Berlin, Germany; [2]Center for Population Health Sciences and Center for Primary Care and Outcomes Research, Departments of Medicine and of Health Research and Policy, Stanford University, Stanford, CA, USA; [3]Research and Analytics, Collective Health, San Francisco, CA, USA; [4]School of Public Health, Imperial College, London, England, United Kingdom

ABSTRACT

Background: Machine learning is increasingly used to predict healthcare outcomes, including cost, utilization, and quality.

Objective: We provide a high-level overview of machine learning for healthcare outcomes researchers and decision makers.

Methods: We introduce key concepts for understanding the application of machine learning methods to healthcare outcomes research. We first describe current standards to rigorously learn an estimator, which is an algorithm developed through machine learning to predict a particular outcome. We include steps for data preparation, estimator family selection, parameter learning, regularization, and evaluation. We then compare 3 of the most common machine learning methods: (1) decision tree methods that can be useful for identifying how different subpopulations experience different risks for an outcome; (2) deep learning methods that can identify complex nonlinear patterns or interactions between variables predictive of an outcome; and (3) ensemble methods that can improve predictive performance by combining multiple machine learning methods.

Results: We demonstrate the application of common machine methods to a simulated insurance claims dataset. We specifically include statistical code in R and Python for the development and evaluation of estimators for predicting which patients are at heightened risk for hospitalization from ambulatory care-sensitive conditions.

Conclusions: Outcomes researchers should be aware of key standards for rigorously evaluating an estimator developed through machine learning approaches. Although multiple methods use machine learning concepts, different approaches are best suited for different research problems.

Keywords: claims data, deep learning, elastic net, gradient boosting machine, gradient forest, health services research, machine learning, neural networks, random forest.

VALUE HEALTH. 2019; 22(7):808–815

Introduction

Machine learning is a rapidly growing field that attempts to extract general concepts from large datasets, commonly in the form of an algorithm that predicts an outcome (commonly referred to as a predictive model or estimator)—a task that has become increasingly difficult to accomplish by humans because data volume and complexity has increased beyond what was capable with traditional statistics and desktop computers. Recently, machine learning has been used to predict healthcare outcomes including cost, utilization, and quality; for example, machine learning methods have been used to predict "cost bloomers," or patients who move from a lower to the highest decile of per capita healthcare expenditures.[1] Machine learning has also been used to predict which patients are most likely to experience a hospital re-admission for congestive heart failure and related conditions.[2] Although causal research identifies what factors cause healthcare outcomes, machine learning will *inter alia* use these factors to identify which patients will have these outcomes.

Because machine learning remains an emerging field and its application to healthcare outcomes research is also nascent, we provide a high-level overview of key concepts and best practices in machine learning for practitioners and readers of healthcare outcomes research. We describe the steps of data preparation, estimator family selection, parameter learning, regularization, and evaluation. We then compare 3 of the most common machine learning methods: (1) decision tree methods that can be useful for identifying how different subpopulations experience different risks for an outcome; (2) deep learning methods that can identify complex non-linear patterns or interactions between variables predictive of an outcome; and (3) ensemble methods that can

* Address correspondence to: Patrick Doupe, PhD, Zalando SE, Veleska-Gert-Strasse 5, Berlin 10243, Germany. Email: patrick.doupe@zalando.de

https://doi.org/10.1016/j.jval.2019.02.012

improve predictive performance by combining multiple machine learning methods.

Methods

To illustrate the principles in this primer, we demonstrate the application of common machine learning methods to a simulated insurance claims dataset (see Appendix in Supplemental Materials found at https://doi.org/10.1016/j.jval.2019.02.012). We specifically include statistical code in R and Python—two of the most common software tools for machine learning—to develop and evaluate estimators that predict which patients in the dataset are at a heightened risk for hospitalization from ambulatory care-sensitive conditions (ACSCs). ACSCs are conditions defined by the Agency of Healthcare Research and Quality as those for which a hospitalization should be preventable through adequate primary care, for example, community acquired pneumonia.[3]

Overview of Machine Learning

In machine learning, the goal is to "learn" an estimator that maps inputs to an output.[4] One definition of learning is:

> "A computer program is said to **learn** from experience … with respect to some class of tasks … and performance measure …, if its performance at tasks …, as measured by [the performance measure], improves with experience."[5(p.2)]

An example task is predicting future hospitalization; experience is the data the researcher collects, and the performance measure is some function of our predictions and the ground truth. The term *learning* refers to the data-driven nature of the process, where unlike in statistical regression, one maintains few to no assumptions about the functional form of the estimator that maps the inputs to outputs.[6] In machine learning an estimator is given experience (say, a set of input–output pairs) and learns to perform a task (say, predicting hospital readmission). We can measure performance by calculating how often the learned estimator correctly labels new patients' readmissions. There is significant work in how to sample and learn estimator parameters, which we discuss in the following sections.

At a high level, we believe that applying a machine learning algorithm involves at least five steps: (1) data preparation; (2) estimator family selection; (3) estimator parameter learning; (4) estimator regularization; and (5) estimator evaluation. We separate steps 3 and 4 because although parameter learning and regularization often occur at the same time, we wish to emphasize the distinction between the two concepts.

Data Preparation

The design of a machine learning estimator begins with a dataset. As an illustrative example, assume that we are given various characteristics, such as age, sex, diagnostic codes, and other variables in insurance claims data, for 100 000 patients. Each input–output pair will consist of the patient covariates (inputs) and the binary outcome variable of whether or not a person had an ACSC hospitalization (output). We are interested in learning an estimator that maps a new patient's information to a predicted ACSC hospitalization value or probability (which has not been observed) to help potentially administer preventive measures. To learn a general estimator that correctly predicts unseen data, we must organize the initial dataset in a manner that simulates the real-world setting of obtaining a new observation. To do so, practitioners generally divide the original dataset into three subsets: a training dataset, a test dataset, and a validation dataset (say, 70%, 15%, and 15%, respectively, of the original dataset[7]).

It is important to ensure that each subset is representative of the overall dataset population, such that we have a representative distribution of inputs and outputs. The training dataset (or subset) is used to initially learn the estimator parameters. The validation dataset is used to iteratively fine-tune the parameters. The test dataset is used only once, after the estimator has been finalized, to assess the generalizability of the estimator. The purpose of hiding the test dataset is to minimize the likelihood of memorizing input–output mappings by repeatedly fine-tuning the estimator. When the estimator memorizes data points in the training set, the estimator is not likely to fit new data to the same level of accuracy. The pitfall of learning fine grain details of the data as opposed to general properties is known as "overfitting" (Fig. 1).

A more common method than the training-validation-test split is K-fold cross-validation. This method divides the nontest dataset into K-folds, with training occurring K times, and each fold being used for validation once. We average the measures across K validation folds, rather than the single fold. Typical values for K are 5 or 10. Importantly, researchers do not need a larger than usual data set size to implement K-fold cross-validation.

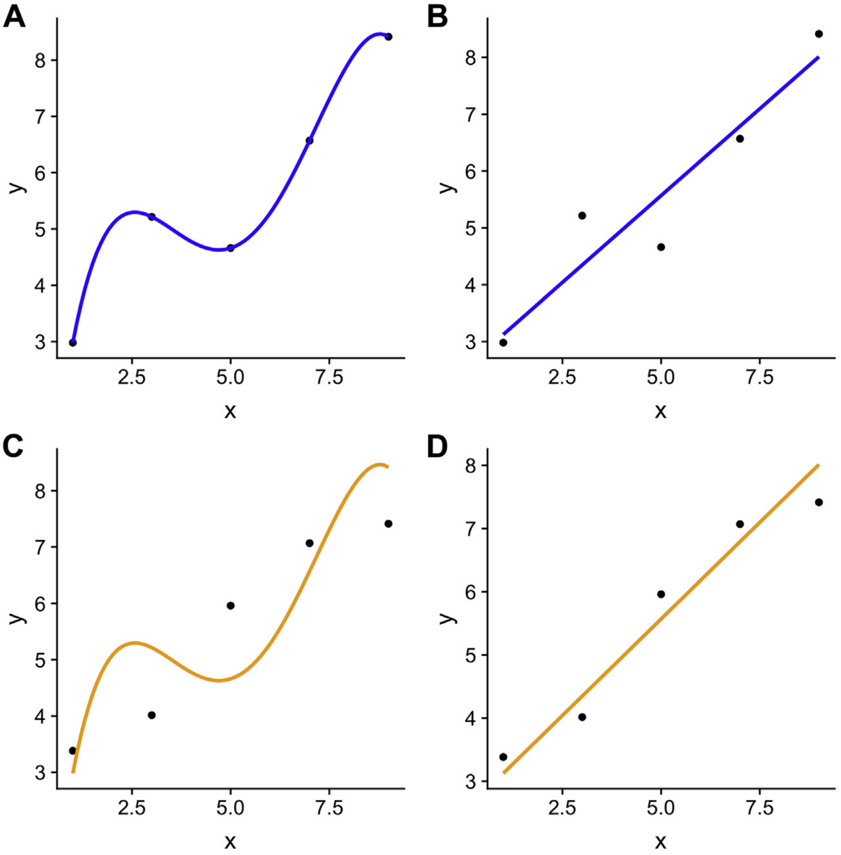
Choosing a Class of Estimator

Second, we select the family of estimators that map inputs to outputs, for example, linear estimators and nonlinear estimators, as detailed further later, which include traditional statistical models, such as a logistic regression model. In most healthcare quality settings, the focus is on using "supervised learning" families, where we know in advance the input–output examples and we wish to learn an estimator that maps inputs to outputs. This is in contrast to exploratory estimators for "unsupervised learning," where we are only given inputs and the goal is to find natural grouping in the data or other high-level characteristics of the data, such as via factor analysis.[7] Estimator family selection is focused on choosing the type of estimator one considers when trying to map inputs to outputs. In theory, there is an infinite number of estimators one can choose from, and given the finite nature of the observed samples, there are numerous estimators that could correctly map inputs to an output in the observed dataset. Nevertheless, exhaustively searching all possible estimators and their permutations would be computationally involved and can lead to estimators that correctly map inputs to an output by random chance. Instead, a researcher tends to constrain their attention to a family of estimators first before learning the estimators' parameters from the data. Function families include linear regression estimators, trees, graphs, or even neural networks, among others (see Section Potentially Useful Estimator Families for Public Health Research for details). Oftentimes, the researcher uses their intuition as to what type of function would best fit the input–output relationship. For the purposes of this article, we describe the 4 most common machine learning estimator families that we believe could be used to predict healthcare outcomes (Table 1).

Parameter Learning

Third, we learn the estimator's parameters by iteratively analyzing the data. For example, a Gaussian's parameters are its mean and standard deviation, whereas a linear regression's parameters are the weights on its covariates. An estimator's parameters are "learned" by repeatedly analyzing the training dataset and applying the inputs to the function and minimizing the difference between the predicted and actual output. Based on the difference between the predicted and actual output, the

Figure 1. Overfitting. Suppose that we have sampled some variable *x* and outcome *y* in a field experiment. (A) The estimator (fitted curve) may appear to be "better" by a performance metric such as the R^2 (which equals 1 because the estimator has perfectly fit the data), but we would not expect the curved estimator to reliably predict outcome *y* given some values of variable *x* or even fit the data very well if we were to repeat the experiment (C) because the curve has fitted random error in the dataset. By contrast, (B) the estimator (line) may have poorer performance on a metric such as the R^2, but does a better job of capturing the general relationship between outcome *y* and variable *x* (D).



estimator adjusts its defining parameters until the difference between predicted and actual output is minimized. There are a host of a optimization methods that attempt to estimate function parameters in an efficient and accurate manner.[8]

Regularization

Fourth, we simplify the estimator by penalizing complexity. The process, known as regularization, was inspired by Occam's Razor and states: given two estimators that perform similarly, preference is given to the simplest of the two.[7] Two common regularization approaches are (1) LASSO (least absolute shrinkage and selection operator), or L1 regularization, which penalizes the absolute sum of regression coefficients, and (2) ridge, or L2 regularization, which penalizes the squared sum of regression coefficients. LASSO tends to select one correlated covariate and drop the others by shrinking their coefficients to zero,[9] whereas ridge shrinks the magnitude of the coefficients for correlated covariates towards zero. The idea behind ridge regression is that we do not want one of the coefficients among many correlated variables to be too extreme based on an outlier data point and thereby influencing predictions more than the other highly correlated variables.[10] Because LASSO and ridge regression constrain the estimator in different ways, many machine learning practitioners will use a combination of both L1 and L2 regularization, known as elastic net regularization,[11] which is demonstrated in our example code linked in the Appendix (see Appendix in Supplemental Materials found at https://doi.org/10.1016/j.jval.2019.02.012).

Regularization methods all require the researcher to select the degree of regularization with a parameter (or two parameters—one for L1 and one for L2—in the case of elastic net). The parameter is chosen to minimize an error metric—often the mean-squared error between the estimator predictions and observed data. The data used to select the regularization parameter are the validation dataset. From this, we use the term "cross-validation" to name the process by which the algorithm selects the regularization parameter value.

Estimator Evaluation

Finally, to test the generalizability of a learned estimator, we evaluate the function's performance by calculating the difference between the values predicted by our estimator and the actual observed measurements in the test dataset. The test dataset has never been analyzed by the estimator and mimics the process of obtaining new unobserved inputs without a corresponding output. The difference between the predicted and actual measurements is known as "the loss." It is important to note that the test process assumes that new incoming data will be generated from the same identical distribution; that is, the data are similar to

Table 1. Common machine learning methods: features, advantages, and disadvantages of common machine learning methods currently applied in healthcare outcomes research.

Method	Intuition	Advantages	Disadvantages
Regularization	To reduce overfitting, penalize estimators that include more covariates, especially correlated covariates (multicollinearity)	Produces more parsimonious (simple) estimators; improves generalizability; helps to produce stable results less sensitive to small changes in estimator choices	Can select the "wrong" predictor when there are many highly correlated predictors. Adds to computational complexity.
Unsupervised learning: factor analysis, principal components analysis, K-means, hierarchical clustering, neural networks	Cluster data into underlying "dimensions," by seeing how key features of the people or institutions in the data correlate together	Can simplify complex/noisy data by finding underlying commonalities; can help researchers categorize people or institutions into groups	Accuracy cannot be determined; researchers' intuition and experience determines usefulness of the result
Decision trees	Sequentially separate data based on values of specific features.	Has good interpretability.	Prone to overfitting.
Ensemble of decision trees: gradient boosting machines	Fit multiple decision trees to weighted resampled subsets of the data, where the errors in prediction from the first tree inform how to improve the next tree	Often achieves the highest performance (lowest error between predicted and ob served outcomes) among modern machine learning methods for tabular data	Requires more researcher effort to "tune" the estimators to ensure optimal performance; does not explain mechanism for result, hence better for prediction than inference
Ensemble of decision trees: random forest	Fit multiple decision trees to bootstrap-resampled versions of the data, then either a) average the resulting trees (for regres sion) or b) take majority vote (for classification).	Requires little researcher effort to "tune" the estimators to en sure optimal performance; fast to implement	Does not explain mechanism for result, hence better for prediction than inference
Deep learning: neural networks	A series of data transformations, where outputs from one series of transformations inform the inputs to the next series of transformations, repeatedly through multiple layers of transformations, ultimately producing abstractions/ generalizations from the data	Can help predict outcomes with highly complex, nonlinear relationships and interactions; can be leveraged to better iden tify the risk of an outcome from extremely large and noisy, and nontabular datasets.	Requires high computing power; requires more researcher effort to "tune" the estimators to ensure optimal performance; does not explain mechanism for result, hence better for prediction than inference
Machine learning meta-learners	Combines multiple machine learning tools to arrive at a summary prediction of an outcome among them	Even if the underlying machine learning estimators do not contain the "true" prediction function, ensembles can produce an excellent approximation of that function	Time- and computing-power-intensive; does not explain mechanism for result, hence better for prediction than inference; may encourage "fishing" across different methods to get high performance without a priori justification

the data used in training and validation. This can be a strong assumption if either the data collected for training are biased, for example, no minority patient data was included, or if the process that generated the data is nonstationary or changes over time. An example of nonstationarity would be data collected when a pathogen becomes drug resistant, and the training and validation data are taken from earlier periods and the test set from a later period.

Overall, the above five steps can be succinctly summarized as follows: machine learning produces algorithms that "minimize loss over a function class, subject to regularization, i.e. penalizing for complexity."[12]

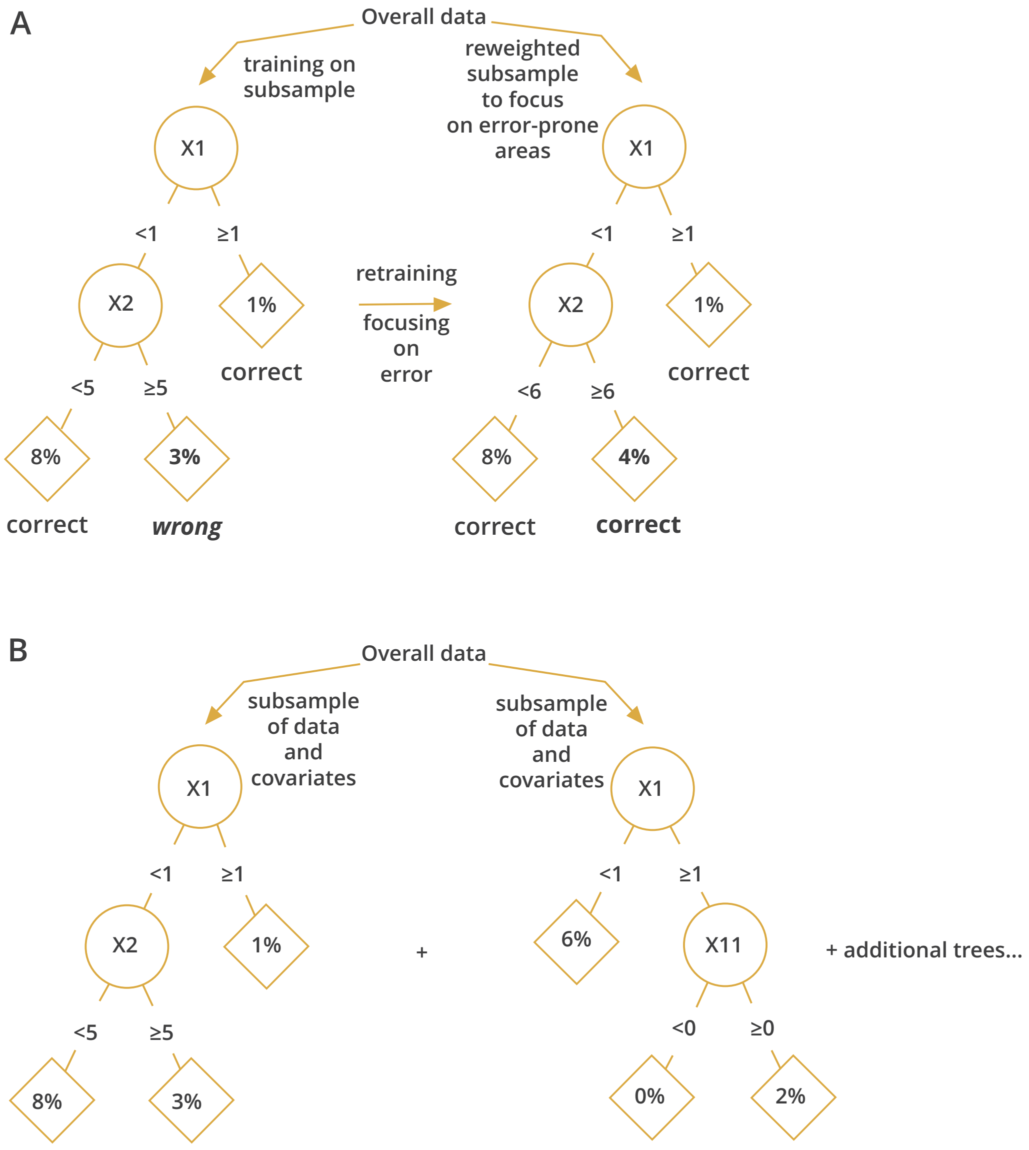
Potentially Useful Estimator Families for Health Services Research

Decision trees

Decision trees are familiar tools used for medical decision making and resemble a flowchart that guides a reader toward classifying a person as either higher risk or lower risk for an outcome (Fig. 2). In a decision tree, each branch of the tree divides the sampled study population into increasingly smaller subgroups that differ in their probability of an outcome of interest.[13] A good decision tree will separate the sampled population into groups that have low within-group variability but high between-group variability in the probability of the outcome. The advantage of a decision tree is the ability to consider nonlinear relationships among multiple covariates that define subgroups in a data-driven way.

Decision tree–based machine learning methods may be most helpful to health services researchers when a research question involves predicting how the risk of an outcome differs among subpopulations or when considering multiple (and potentially multilevel) complex influences on the risk of a health outcome that may be hard to predict through standard logistic regression. For example, people who have the highest risk for an ACSC hospitalization for diabetes complications may have a combination of

Figure 2. Decision trees. (A) Gradient boosting machines (GBM) and (B) random forests (RF). The circles display the covariates (*X* variables) whose values determine each branch point, whereas the diamonds provide the tree-predicted probability of the outcome under study.



high diabetes medication copays, low income, and a long distance to a pharmacy, and also may be in an age group that is older but not old enough to qualify for Medicare. Trees can help decipher such complex dependencies. Trees can also be more useful than standard logistic regression when researchers are trying to predict a rare outcome (such as a very high-cost hospitalization) caused by a constellation of complex interacting factors.[14] A limitation of decision trees is that they are prone to overfitting, such that a subgroup may be identified because the decision tree has over-interpreted noise in the data, and even cross-validation may not detect the overfitting.[15]

Tree ensembles

The two most common methods for overcoming the capacity for trees to overfit are gradient boosting machines (GBM) and random forests (RF). GBMs average many trees that are each grown to re-weighted subsets of the data, where errors made by the first tree contribute to learning of a more optimal tree in the next iteration (called a boosting strategy).[16,17] RF also builds numerous decision trees but averages a forest composed of many trees, where each tree was independently fitted to a random bootstrap-resampled version of the data (called a bagging strategy) with a random subset of covariates selected to be eligible to define the branches.[18]

The GBM approach often requires researchers to experiment with (or "tune") how many trees to average, how deep the trees should be (how many subpopulations to divide the population into), and how quickly the trees should adapt to initial error (the learning rate) to maximize predictive discrimination (measured by a C-statistic). On the other hand, the RF approach tends to produce a reproducible result with maximum discrimination across a wide range of specifications, thereby not requiring

Table 2. Comparison of maximum cross-validated C-statistics produced by alternative methods of predicting hospitalizations for ambulatory care-sensitive conditions (a binary outcome) in a synthetic claims dataset.

Estimator	C-statistic
Logistic regression including all available covariates in dataset	0.67
Logistic regression with backward variable selection using Akaike's information criterion	0.67
Logistic regression with elastic net regularization	0.67
Ensemble of decision trees using gradient boosting machines	0.74
Ensemble of decision trees using random forest	0.72
Deep learning neural network	0.71
Ensemble of gradient boosting machines, random forest, and deep learners	0.72

See links in Appendix to obtain data and statistical code for replication. As shown here, the machine learning algorithms do not necessarily outperform standard regression; their application should be justified for the investigative problem being addressed. In our statistical code linked to the Appendix, we show how different implementations of each of these methods vary widely in performance and how adjustment of key parameters for each method can maximize performance. In the machine learning literature, the C-statistic is also known as the area the under receiver operating characteristic curve (AUC).

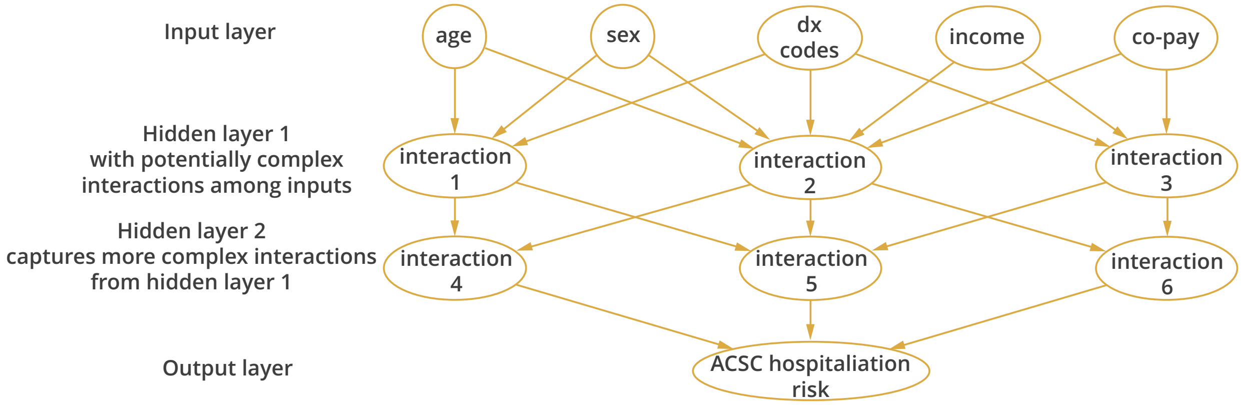
extensive tuning. Prior research suggests that GBMs can produce higher C-statistics to RFs when the task is to predict an outcome with few classes, for example, just a 0 or 1 for absence or presence of the outcome, rather than a multiclass or continuous outcome variables, and vice versa.[19,20] Our code provides examples of both GBMs and RFs applied to the prediction of an ACSC hospitalization and compares their performance to a standard logistic regression (Table 2).

Deep learning

Although the term *deep learning* has been increasingly used in Internet and business literature, the concept refers to a traditional method of machine learning that is not fundamentally new: the development of neural networks.[21] A neural network is a series of data transformations where the outputs from one series of transformations inform the inputs to the next series of transformations (Fig. 3). Each transformer (or neuron) in the network takes a weighted combination of inputs, reflecting a weighted sum of the observed data (for the first layer of neurons) or from a previous layer of neurons (for the second and subsequent layers of neurons), and produces an output based on a nonlinear transformation function known as an activation function. The weights for each sum and activation function values determine the output from the layer. The "deep" in deep learning comes from stacking multiple layers.

Cross-validation techniques are applied for model selection, and researchers can prevent overfitting with regularization. A common regularization method for neural networks is "dropout."

Figure 3. Deep learning conceptualization. Neural networks are based on a loose caricature of the brain as a series of neurons in which inputs (like features of patients in a claims dataset) are processed by a layer of neurons, which then inform another layer of neurons, and so forth, until an output is achieved, for example, risk of hospitalization for an ambulatory care sensitive condition (ACSC). Each neuron takes a weighted combination of input signals reflecting a weighted sum of the input data from the observed data (for the first layer of neurons) or from a previous layer of neurons (for the second and subsequent layers of neurons) and produces an output based on a nonlinear activation function. We can iterate on this procedure to create multiple layers, or "deep" networks. In the overall neural network, an input layer matches the data and is followed by multiple layers of neurons to produce abstractions from the input data, typically ending with a "classification layer" to match a discrete set of outcomes, for example, whether or not a person was hospitalized for an ACSC. The weights and activation function values determine the output from the network. For example, an older woman with a history of recurrent community acquired pneumonia (in diagnostic codes, labeled "dx codes" in the figure) may be particularly predisposed to preventable hospitalizations from community-acquired pneumonia, but only if she is low income and has a high copayment for ambulatory care services. Hidden layer 1 could detect interactions among age, sex, prior diagnostic codes, and among income and copayments, and hidden layer 2 could help identify how the interaction of age, sex, and diagnostic codes, combined with interaction between income and copayments, could identify the subset of women predisposed to an ACSC hospitalization for community-acquired pneumonia.



This method randomly sets parameters to zero with each iteration during training. The network learns not to rely on any small set of parameters; rather, the information required to fulfill the predictive task is spread across the model.

Neural network design has progressed with the ability to train these models "end to end" with backpropogation or training all connected modules at once. The aforementioned network class is often termed a *multilayer perceptron*. Modifications of the multilayer perceptron estimator exploit different data types. For images, the research can use convolutional neural networks that exploit spatial dependencies among image pixels. In healthcare research, convolutional neural networks are used for ophthalmology or radiology applications.[22,23] For language, health records, or time series data, the researcher can use recurrent neural networks to exploit dependencies over time. Researchers have used recurrent neural networks with electronic medical record texts to predict future health outcomes.[24] Sometimes, we want to generate data from a distribution and the researcher can use Generative Adversarial Networks or Variational Autoencoders. A new application in healthcare research is generating health records for analysis (http://proceedings.mlr.press/v68/choi17a.html). Last, already trained neural networks can be used in unsupervised learning. By passing data through an appropriate network, we can extract information that is useful in prediction. The extracted outputs can be used in other models. The amount of activity in these research fields speaks to the flexibility of neural networks.

In our example code, we demonstrate how a neural network can help identify complex interactions that predict ASCS hospitalization in our simulated claims dataset (see Fig. 3 for an example). We show that if multiple input covariate transformations and complicated interactions among covariates are truly influencing the probability of the outcome, a deep learning approach may outperform a tree-based method.

One challenge for researchers implementing deep learning is that a neural network can require complex choices for the activation functions, different network depths (number of layers) and degree of regularization to be applied. A second challenge is that communicating how the neural network is transforming the data is challenging compared with communicating the structure of a regression or decision tree. In the Appendix, we detail a typical strategy to tune a neural network, detailing multiple common options for activation functions, network depths, and regularization settings and comparing the network to tree-based estimators and logistic regression equation estimators (Table 2).

Meta-learners

Machine learning research has consistently suggested that although any one of the previously mentioned methods may improve prediction compared with standard logistic regression, using a combination of the methods may improve prediction more than any single method. One set of research reveals that even if a set of component estimators, or base learners, does not contain the true estimator, an ensemble of them can give a surprisingly good approximation to the truth.[25] A meta-learner can be particularly helpful if there is little a priori reason to believe that one machine learning estimator would be inherently superior to others for prediction, based on the characteristics listed in Table 2. After developing the estimators on a training dataset, a meta-learner estimator (called a super learner or stacking method) can be used to combine the predictions of the underlying base learners. The meta-learner defines the weight given to each of the component base learners, often using an approach such as elastic net regularization to find a combination of estimators that minimizes error between the weighted predictions of the base learners and the observed data. Both GBM and RF are actually examples of meta-learners because they combine single-decision trees to develop a composite classification or prediction. In the code included in the Appendix, we combine all of the machine learning methods described here (logistic regression with regularization, GBMs, RF, and deep learners) into an ensemble for predicting risk of an ACSC hospitalization using simulated claims data (Table 2; see Appendix in Supplemental Materials found at https://doi.org/10.1016/j.jval.2019.02.012).

We note that meta-learners will not provide the best predictions across all problems. No model can do this as there is "no free lunch."[26]

Discussion

Machine learning methods may be useful to health service researchers seeking to improve prediction of a healthcare outcome with large datasets available to train and refine an estimator algorithm. Machine learning methods can help generalizable data-driven estimators when many covariates are being selected among and when the outcome of interest may be produced by complex nonlinear relationships and interaction terms.

Yet machine learning methods offer considerable challenges for healthcare outcomes researchers that are worth considering before engaging in a machine learning activity. They may be difficult to interpret (particularly for deep learning), difficult to glean mechanistic understandings from (a challenge for all methods discussed here), and may require substantial investment of time and resources for computation (particularly for gradient boosting machines, deep learning, and ensembles). Nevertheless, computational improvements in hardware and cloud computing technologies have made machine learning methods increasingly accessible to healthcare outcomes researchers and healthcare organizations. The code that accompanied this article, for example, required only 100 lines and 30 minutes to run all of the estimators described here on a simulated claims database of 100 000 people, using a standard laptop computer.

Because machine learning methods are increasingly adopted for healthcare outcomes research, we offer three points of advice, following guidelines described in the machine learning literature.[27-29]

First, an estimator should provide a solution to a prespecified problem rather than simply detecting associations in a large dataset. Prespecifying the problem being addressed, including notions of success, may help reduce the risk of false-positive findings.[27] In addition, prespecifying the metrics for comparison of estimators can help prevent false claims of estimator improvement. The C-statistic, for example, helps to identify whether an estimator can distinguish a higher-risk from a lower-risk person; for many medical and healthcare outcomes research tasks, it may be equally important to test for calibration, for example, the Hosmer-Lemeshow test, which determines whether predicted event rates and observed event rates for an outcome are concordant with one another or very different, because the absolute magnitude of estimated risk may be used for decision making.[30]

Second, the audience intended to use the estimator should be considered. Whether or not the audience needs to understand what features generate the estimator's predictions, or simply be able to apply it to future datasets through an automated application, for example the backend of an electronic medical record, should be determined. The answer to this decision will affect choices of estimators because it is more difficult to understand why some estimators, for example, deep learners, make predictions from data than others. Often, a better performing model in terms of some accuracy metric will be in conflict with other goals, such as human understanding of predictions. There is no rule of thumb for how much performance improvement is sufficient to justify using less interpretable estimators. We suggest that researchers have clear justification for choosing an estimator.

Third, it is important to have "data empathy," which refers to the idea that no matter how complex the method, a dataset that is poor in quality or poorly informative for a given question will not be useful, even if large in size.[31] For example, analyzing claims data may be appropriate to predict an outcome of hospitalization, which is well captured and carefully adjudicated in the data. But using such data to identify covariates predictive of a diagnosis may be fraught because claims data are known to be subject to significant diagnostic misclassification or underdiagnosis errors. Machine learning methods may have deceptively high accuracy but predict the wrong outcome, such as predicting the probability of being diagnosed with a condition, not the probability of actually having the condition. Hence, measurement and selection biases apply to machine learning methods as much as to any other forms of secondary data analysis.[32]

Ultimately, emerging machine learning methods are potentially useful for the healthcare outcomes researcher if prediction is an important and meaningful endeavor. Prediction can be combined with causal research to improve our understanding (we thank an anonymous reviewer for pointing this out).[33] With this article, we aim to lower the barriers to implementing machine learning methods. As next steps, we recommend the following textbook and Massive Open Online Course: http://ciml.info/ and https://www.coursera.org/learn/machine-learning. These resources will help researchers learn about additional models like naive Bayes and kernel methods in addition to deeper principles. In addition, there are many tutorials online for the researcher to keep up to date. As machine learning methods evolve, we argue that the principles for good practice reviewed in this primer will likely serve health services researchers well into the future.

Source of Financial Support

Funded by NIH DP2MD010478, U54MD010724, and K08HL121056. The funders had no role in the collection, analysis, and interpretation of data; in the writing of the report; or in the decision to submit the article for publication. All authors are fully independent from the funders.

Supplemental Material

Supplementary data associated with this article can be found in the online version at https://doi.org/10.1016/j.jval.2019.02.012

REFERENCES

1. Tamang S, Milstein A, Sørensen HT, et al. Predicting patient "cost blooms" in Denmark: a longitudinal population-based study. *BMJ Open*. 2017;7(1):e011580.
2. Mortazavi BJ, Downing NS, Bucholz EM, et al. Analysis of machine learning techniques for heart failure readmissions. *Circ Cardiovasc Qual Outcomes*. 2016;9(6):629–640.
3. Ansari Z, Laditka JN, Laditka SB. Access to health care and hospitalization for ambulatory care sensitive conditions. *Med Care Res Rev MCRR*. 2006;63(6):719–741.
4. Jordan MI, Mitchell TM. Machine learning: trends, perspectives, and prospects. *Science*. 2015;349(80):255–260.
5. Mitchell TM. *Machine learning*. New York: McGraw Hill; 1997.
6. Breiman L. Statistical modeling: the two cultures (with comments and a rejoinder by the author). *Stat Sci*. 2001;16(3):199–231.
7. Hastie T, Tibshirani R, Friedman J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd ed. New York: Springer; 2011.
8. Bottou L, Curtis FE, Nocedal J. Optimization methods for large-scale machine learning. *SIAM Rev*. 2016;60(2):223–311.
9. Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*. 2008;9(3):432–441.
10. Hoerl AE, Kennard RW. Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*. 1970;12(1):55–67.
11. Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. *J Stat Softw*. 2010;33(1):1–22.
12. Mullainathan S, Spiess J. Machine learning: an applied econometric approach. *J Econ Perspect*. 2017;31(2):87–108.
13. Quinlan JR. Induction of decision trees. *Mach Learn*. 1986;1(1):81–106.
14. Poole S, Grannis S, Shah NH. Predicting emergency department visits. *AMIA Summits Transl Sci Proc*. 2016;2016:438–445.
15. Athey S, Imbens G. Recursive partitioning for heterogeneous causal effects. *Proc Natl Acad Sci U S A*. 2016;113(27):7353–7360.
16. Friedman J, Hastie T, Tibshirani R. Additive logistic regression: a statistical view of boosting. *Ann Stat*. 2000;28(2):337–407.
17. Friedman JH. Greedy function approximation: a gradient boosting machine. *Ann Stat*. 2001;29(5):1189–1232.
18. Breiman L. Random forests. *Mach Learn*. 2001;45(1):5–32.
19. Caruana R, Karampatziakis N, Yessenalina A. An empirical evaluation of supervised learning in high dimensions. Proc. 25th Int. Conf. Mach. Learn.—ICML '08, 2008, pp. 96-103.
20. Caruana R, Niculescu-Mizil A. An empirical comparison of supervised learning algorithms. Proc 23rd Int Conf Mach Learn 2006;C: pp. 161-168.
21. LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*. 2015;521:436–444.
22. Gulshan V, Peng L, Coram M, et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA*. 2016;316(22):2402.
23. Ehteshami Bejnordi B, Veta M, Johannes van Diest P, et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *JAMA*. 2017;318(22):2199.
24. Miotto R, Li L, Kidd BA, Dudley JT, Agarwal P. Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. *Sci Rep*. 2016;6:26094.
25. Van der Laan MJ, Polley EC, Hubbard AE. Super learner. *Stat Appl Genet Mol Biol*. 2007;6. Article 25.
26. Wolpert DH. The lack of a priori distinctions between learning algorithms. *Neural Computat*. 1996;8(7):1341–1390.
27. Luo W, Phung D, Tran T, et al. Guidelines for developing and reporting machine learning predictive models in biomedical research: a multidisciplinary view. *J Med Internet Res*. 2016;18(12):e323.
28. Boulesteix AL. Ten simple rules for reducing overoptimistic reporting in methodological computational research. *PLoS Comput Biol*. 2015;11(4):e1004191.
29. Tanwani AK, Afridi J, Shafiq MZ, Farooq M. *Guidelines to select machine learning scheme for classification of biomedical datasets*. In: *Lecture Notes in Computer Science*. vol. 5483. Berlin: Springer; 2009:128–139.
30. Hosmer DW, Lemeshow S. *Applied Logistic Regression*. 2nd ed. New York: John Wiley & Sons; 2000.
31. Faghmous J. Systems science and population health. In: El-Sayed AM, Galea S, eds. *Systems Science and Population Health*. New York: Oxford University Press; 2017:129–138.
32. Crown WH. Potential application of machine learning in health outcomes research and some statistical cautions. *Value Health*. 2015;18(2):137–140.
33. Luque-Fernandez MA. *Targeted Maximum Likelihood Estimation for a Binary Outcome: Tutorial and Guided Implementation*. GitHub Repository; 2017. http://migariane.github.io/TMLE.nb.html.