Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/jval

COMMENTARY

Potential Application of Machine Learning in Health Outcomes Research and Some Statistical Cautions

William H. Crown, PhD[*]

Optum Labs, Cambridge, MA, USA

ABSTRACT

Traditional analytic methods are often ill-suited to the evolving world of health care big data characterized by massive volume, complexity, and velocity. In particular, methods are needed that can estimate models efficiently using very large datasets containing healthcare utilization data, clinical data, data from personal devices, and many other sources. Although very large, such datasets can also be quite sparse (e.g., device data may only be available for a small subset of individuals), which creates problems for traditional regression models. Many machine learning methods address such limitations effectively but are still subject to the usual sources of bias that commonly arise in observational studies. Researchers using machine learning methods such as lasso or ridge regression should assess these models using conventional specification tests.

Keywords: machine learning, outcomes research, treatment effects.

There is a worldwide explosion in the availability of data to support outcomes research, health economics, and epidemiology. Data availability is expanding along various dimensions simultaneously [1]. One is volume; for example, numerous initiatives are amassing huge repositories of claims and electronic medical record (EMR) data: Food and Drug Administration Mini-Sentinel, the Patient-Centered Outcomes Research Institute Clinical Data Research Networks, Patient-Centered Clinical Research Network, Optum Labs, and many international examples [2–4]. There is also the dimension of velocity—the speed with which users can interact with the data. EMR data are often available almost in real time. Moreover, the variety of data is expanding. Claims and EMR data are increasingly being linked with health risk assessments, sociodemographic data, and vital signs on a broad basis. And, most recently, there is emerging data on genetic characteristics of individuals, as well as data flowing from devices such as FitBits and biometric sensors. Such data are very rich, but they are sparse—you have them only for certain people. This creates challenges for traditional multivariate methods such as ordinary least squares regression analysis because many observations are lost due to missing data.

We have many good statistical methods for analyzing observational data. The sheer volume of data, along with their characteristics, such as the unevenness of data completeness, however, raises questions about the potential for using new methods to analyze questions of treatment effectiveness, health care value, strengths and weaknesses of alternative care organization models, policy interventions, and so on. In particular, machine-learning methods, which have been extensively used in the consumer retail sector (e.g., Amazon.com), may offer some interesting alternatives to traditional statistical methods that could potentially overcome many of the challenges posed by "Big Data."

The term "machine learning" refers to large family of mathematical and statistical methods that have historically been focused on prediction [5]. We are often interested in prediction in health care. What strain of flu is likely to be prevalent in the coming flu season? How many vials of flu vaccination must be prepared to meet treatment demand? But prediction is not quite the same thing as estimating treatment effects. For a physician, the challenge is to isolate the effect of a treatment on patient outcomes so that the correct treatment can be selected. Policy evaluations face the same statistical challenges. Some machine-learning methods have the ability to estimate treatment effects and some do not. But the distinction between prediction and treatment-effect estimation is almost completely absent in the machine-learning literature.

In brief, the basic approach with all machine learning is to segment the data into learning and validation data sets to develop highly accurate classification algorithms. Once the

Note. This article is based on a presentation given during the First Plenary Session, "The Use of 'Big Data'—Where Are We and What Does the Future Hold?" at the ISPOR 19th Annual International Meeting, May 31–June 4, 2014, Montreal, QC, Canada.

[*] *Address correspondence to:* William H. Crown, Optum Labs, One Main Street, 10th Floor, Suite 1040, Cambridge, MA 02142, USA. E-mail: william.crown@optum.com.

1098-3015$36.00 – see front matter Copyright © 2015, International Society for Pharmacoeconomics and Outcomes Research (ISPOR). Published by Elsevier Inc.

http://dx.doi.org/10.1016/j.jval.2014.12.005

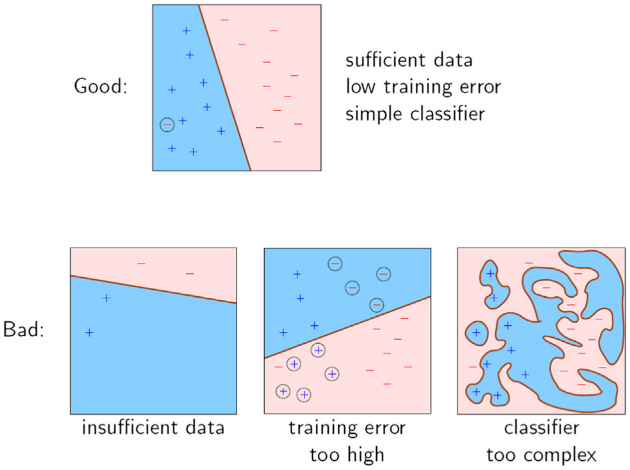

Fig. 1 – **Good classifiers and bad classifiers. Reprinted with permission of Robert Schapire [9].**

algorithms have been developed, they are applied to the full data set to do the prediction. The idea is that one should be able to perform these classifications without human intervention, and the methods should also be able to operate on very large data sets and be very fast. In the machine-learning literature, this process of using learning and training data sets to develop prediction algorithms is known as K-fold cross-validation. The approach is fairly straightforward. The idea is to take the initial data set and randomly split it into several (typically 5 or 10) subsamples. For each subsample that is held aside, the classification algorithms are built on each of the other remaining subsamples. Once the algorithms have been built, each is used to predict the membership prediction error that is associated with each one of the subsamples. Finally, a sum of prediction errors is calculated over all subsamples. Using this approach, one can evaluate different machine-learning methods simultaneously and then compare the average errors associated with each model to determine which method performs the best. The process is completely automated. The best algorithm is applied to the entire data set—typically to do a prediction.

Machine-learning methods consist of a large number of alternative methods including classification trees, random forests, neural networks, support vector machines, and lasso and ridge regression to name a few. Classification trees are a good place to start because they illustrate the machine-learning approach very intuitively and also extend directly to powerful related methods such as random forests that are widely used for predictive model development.

We begin with the notion of classifiers to predict group membership. Figure 1 shows some examples of good and bad classifiers. The box at the top of the figure is a good classifier. Assume that there are two types of observations—the pluses and the minuses. It splits them almost perfectly except that there is one mistake in the good box in which we have a negative. By a very simple rule, just one line through the scatter plot, the data have been classified. Down on the bottom row, we have a variety of different cases. The first one to the left has split the data, but there are so few observations that we would not have much confidence in this particular algorithm and its ability to perform equally well on another data set. In the middle box, there are many errors. This algorithm is classifying only about half of the cases properly, and we have a mix of positives and negatives in each one of the groups. The final box is a classification algorithm that is perfect in the sense that it classifies the positives and the negatives but it is extraordinarily complex. It is possible to create increasingly precise classifiers by adding additional terms and nonlinearities, powers, polynomials, and so forth. But there is no guarantee that the rule is going to work on another data set. Even if it does, complex rules are more difficult to understand and implement, so they are not as useful as simpler rules.

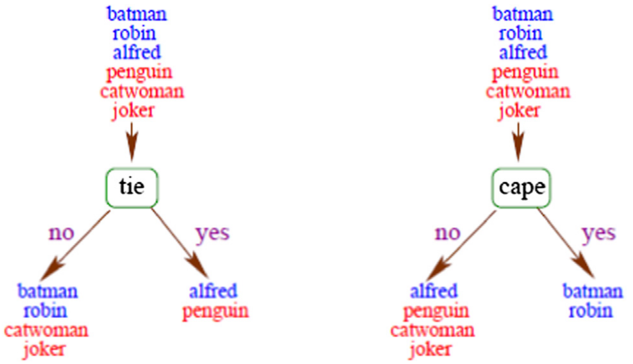

Fig. 2 – **Choosing the classification rule. Reprinted with permission of Robert Schapire [9].**

Figure 2 illustrates how classification algorithms can be used to form predictions. In this simple example, we have a collection of characters from Batman. Some of these characters are good guys and some are bad guys. Assume that we can classify the good guys and the bad guys into groups. Batman, Robin, and Alfred are all good guys. The Penguin, the Catwoman, and the Joker are all bad guys. We have some measured characteristics for all of them including their sex, whether they wear a mask, whether they wear a cape or a tie, whether they have ears, and whether they smoke. These observations constitute our training data. Now, suppose that we have the same measured characteristics for Bat Girl and the Riddler and we want to try to figure out whether they are good or bad. Let us compare two different classification algorithms that could be used for categorizing them as good or bad.

First, let us look at whether they wear a tie. Figure 2 shows that we have the same inputs going into each one of two classification algorithms—whether the character wears a tie and whether the character wears a cape. On the left, it is apparent that the tie does not do a very good job of classifying. We end up with Alfred and the Penguin both wearing ties, so we have got a good guy and a bad guy in the Yes category. And Batman and Robin do not wear ties, nor does Catwoman or the Joker. So, we end up with two good guys and two bad guys in the No category. Using the tie as a classifier did not help at all. Now, let us look at whether the characters wear a cape. Batman and Robin both wear capes, so classifying them as good guys works perfectly. In contrast, the Penguin, the Catwoman, and the Joker do not wear capes, so that is correct as well. But, unfortunately, Alfred is a good guy who does not wear a cape, so he is incorrectly classified. Still, this is a pretty good classification algorithm because it correctly classified all but one of the characters in the sample. This is what we are looking for—the ability to classify as simply as possible with minimum error possible. On this basis, the cape does a pretty good job. Normally, machine-learning methods would build a very large tree and then prune it back.

One of the most powerful and popular machine-learning methods is known as random forests. As the name implies, random forest methods involve estimating a whole forest of classification trees. The process works like this: Randomly select a subset *m* of predictor variables from an initial pool of, say, 1000 variables. The variable that provides the best split is used to do a binary classification on the first node. At the next node, choose