Kidney360

Introduction to Supervised Machine Learning

Aditya Biswas, Ishan Saran, and F. Perry Wilson

KIDNEY360 2: 878–880, 2021. doi: https://doi.org/10.34067/KID.0000182021

The common conception and criticism of machine learning (ML) in medicine is that it centers around a "black box," an inscrutable series of mathematical calculations that take in data and spit out predictions, lacking the pathobiological explanatory rigor to which medical researchers are accustomed. Although this is oversimplified, it is not altogether untrue. It is also not really a problem. The true place for ML in medicine is not in explanation, but in prediction.

The broadest definitions of ML describe processes that take in historical data, learn salient relationships, and use that knowledge to make predictions on new data. To that end, ML algorithms have been present in medicine for decades—hiding in plain sight in the form of, for example, logistic regression and Cox proportional hazard models. Although these algorithms fall within the category of supervised ML (there is a single target outcome, such as death, to predict), they are somewhat special in that they are readily interpretable and assign a "weight" to the various parameters in the model. Indeed, researchers often *over* interpret these weights, suggesting they imply a causal mechanism as opposed to a mere association (1). Modern ML algorithms reorient the central goal to flexibly predicting outcomes for *new* data as closely as possible. This allows us to ease many of the strong assumptions behind classic models, permitting the connection between covariates and an outcome to be mediated by any black-box algorithm, saving considerations of interpretability and plausibility for *post-hoc* discussions (2) In this communication, we hope to orient readers to the techniques, mechanisms, and purpose of supervised ML, which has a goal of predicting outcomes.

Data Preparation

For ML strategies to work, the entire data lifecycle must be carefully considered. The pipeline naturally begins with data selection, which includes cohort selection, but further aims to clarify which aspects of the data will be prospectively available for implementation. For example, biomarker data often carry strong predictive power, but their utility becomes severely limited if the data are too expensive to gather at scale in the future.

Next, the data need to be organized and preprocessed into a format the ML algorithm is expecting. This includes procedures commonly used in medical research, such as deleting data-entry errors, but can also include steps such as transforming text data into more structured data through the use of natural language processing (3). In medical data, missingness is often informative (a patient who has had a lactate measured, even if it is normal, is most likely sicker than one who has not had a lactate measured). Some implementations of tree-based algorithms can handle missing values natively, but most algorithms require some form of imputation.

The final step in preprocessing is feature engineering, which uses domain knowledge to make the learning problem easier for the algorithm. For example, we may believe the variability of a measured value over time is an important outcome predictor. We may then engineer a variable such as "systolic BP variability," on the basis of measured systolic pressures, and allow the model access to this constructed feature. We may also combine features into a new entity on the basis of prior data, as is frequently done when age, race, sex, and creatinine are combined to generate an eGFR (4). The general strategy is to engineer many features that might be relevant and reduce the list during a later feature selection step.

Data Splitting

Because ML algorithms are typically very flexible, they perform suspiciously well on the data they were *trained* on—a phenomenon known as overfitting. To prevent this bias caused by "double dipping" with data, the dataset must be split into training and test sets. All modeling decisions, imputations, and parameter training must only be informed by the training set.

Data splitting is so crucial in ML that it performs yet another task, *model selection.* There are dozens of ML algorithms to choose from, many of which have various "hyperparameters," tuning settings that can alter their overall performance. An ML model is defined by its choices of algorithm, hyperparameters, and predictor variables included. To choose a model among several candidates, the original training set is split into training and validation subsets. In this manner, we can experiment with different models (and variables within models) to determine what generalizes best in the validation subset (Figure 1).

It is at this stage of the data pipeline where feature selection may occur. In general, models with too many features have a higher risk of overfitting the training data, whereas models with too few features may not perform well. Feature selection is beneficial from both statistical and implementational perspectives: having

Clinical and Translational Research Accelerator, Department of Medicine, Yale School of Medicine, New Haven, Connecticut

Correspondence: F. Perry Wilson, 60 Temple St., Suite 6C, New Haven, CT 06510. Email: francis.p.wilson@yale.edu

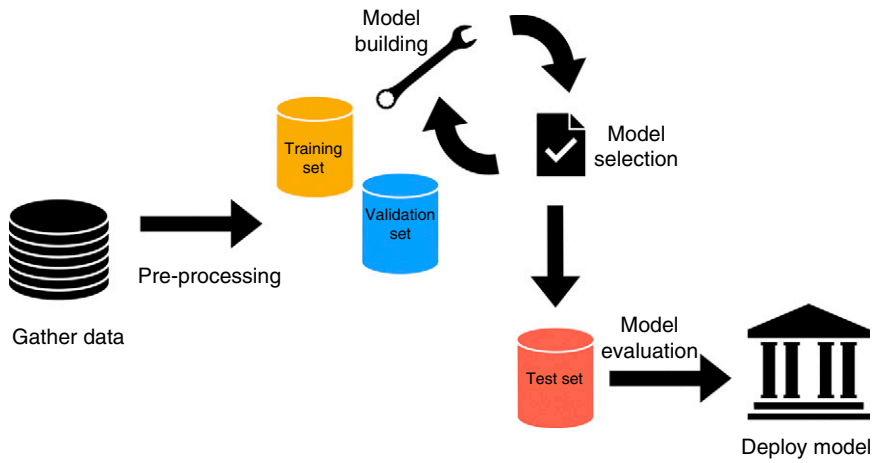

Figure 1. | Schematic of machine learning model training, validation, and implementation.

fewer features controls overfitting and reduces the number of points of potential failure in a production pipeline.

When a set of features, an algorithm, and its hyperparameters are selected, that model is retrained on the full training set (including the validation set) and evaluated on the test set, which has (until this point) been entirely held out from the analysis. This methodology allows the discovery of which model and level of flexibility is best suited for a particular dataset and learning task.

Choice of Algorithm

Although the flexibility of ML algorithms means there are many options to choose from, in practice different data types work best with specific algorithm types. Relevant medical data come in a myriad of different forms: randomized trials, the electronic health record, -omics data, clinical notes, and diagnostic and pathologic imaging. We typically choose the ML algorithm that most naturally handles the underlying structure in our data. Tabular data describe the simplest structure possible: each row indicates a unique datapoint, and each column designates a predictor variable or *feature*. Many easy-to-use algorithms (including logistic regression) are designed to work with tabular data. Unconventional data can be made tabular, of course. For example, images can be made tabular by assigning each pixel location to a column. The Swiss Army knives of tabular data are the Random Forest (5) and Boosted Trees algorithms (6), both of which internally use a collection of decision trees. However, generalized linear models often have acceptable performance and are very easy to implement.

If we allow ourselves to sacrifice ease of use, algorithms better matched to the structure of the data can have significantly better predictive performance. Auto-Regressive Integrated Moving Average models are well designed for forecasting with single feature time series, such as weight changes during heart failure (7). More broadly, Recurrent Neural Networks naturally handle variable-length sequences of data with many dimensions, as may be seen in the electronic health record where some patients are having frequent measurement of a predictor (*e.g.*, blood glucose) and some are not (8). Lastly, Transformer Neural Networks and Convolutional Neural Networks are state of the art for natural language processing and computer vision, respectively (9,10). Unfortunately, neural network algorithms typically require an ML expert to train and tune, and bring with them a suite of implementational hurdles.

Model Evaluation

The critical measure of a ML approach is in how well it makes predictions in the test set. Various evaluation metrics are available. Selecting the right evaluation metric, just as with any other module of the data analysis pipeline, depends on the problem we are interested in. In medicine, we are frequently interested in classification problems, where a confusion matrix provides a good summarization and visualization of key elements of a model's performance. It pays attention not only to what the model gets right but also what it gets wrong. Because it is often the case that models will deliver probabilities instead of binary outputs, metrics that evaluate a continuous score (risk of death) against a binary true outcome (death) are often used—the prototypical being the well-known area under the receiver-operator characteristic curve. For regression problems, mean squared error is a classic evaluation metric with great utility; however, on the basis of the dataset and learning problem, robust alternatives may be preferable.

Separate from evaluating the raw performance of a model, in clinical medicine it is imperative the models we build are usefully adopted by physicians. Adoption by physicians will require faith in the ML model being deployed, and the black-box nature of the models as we have described them thus far is not conducive to that end. Luckily, methods exist to probe the inner workings of the model—Local Surrogate Interpretable ML and Shapley Additive explanations values are among the more popular techniques to probe specific outputs the ML model delivers. By picking representative examples, one can reconstruct the space to get a better idea of how the model is making its decisions. However, the techniques *do not* identify causal mechanisms. Causality carries with it a much larger burden of proof and a good deal more effort is required to tease out causal mechanisms. One important *post-hoc* exploration of an ML model is to

assess the model for fairness; that is to say, its bias against a particular group of people (whether defined by race, sex, or other factors). As models are trained on real-world data, they can learn the implicit (and explicit) biases that real-world data reveal, learning, for example, that minority patients may have worse outcomes in a given disease state, and thus predicting those worse outcomes. As such, model performance should be considered in the population as a whole and within protected classes. Implementation scientists must also consider these issues before broad dissemination of models that may reinforce implicit biases.

Conclusions

Although integrating the ML toolset into the researcher's arsenal may not always be straightforward, the magnitude of improvement in prediction can be well worth it. ML approaches have a fundamentally different goal than traditional multivariable modeling and should be interpreted in that light. Finally, although the performance of an ML model is a critical metric, the true test of any medical technology is whether the application of the technology benefits the patients. This is an area that demands high-quality research.

Disclosures

F.P. Wilson reports consultancy agreements with Translational Catalyst, LLC; reports having an ownership interest in Efference, LLC; reports being a scientific advisor or member of the Editorial Board *American Journal of Kidney Disease* and *CJASN*; and reports other interests/relationships through the Board of Directors of Gaylord Health Care and Medical Commentator of Medscape. All remaining authors have nothing to disclose.

Funding

This work was sponsored by National Institute of Diabetes and Digestive and Kidney Diseases grants R01DK11391 and P30DK079310 (to F.P. Wilson).

Author Contributions

A. Biswas and F.P. Wilson provided supervision; F.P. Wilson was responsible for the resources; and all authors conceptualized the study, wrote the original draft, and reviewed and edited the manuscript.

References

1. Westreich D, Greenland S: The table 2 fallacy: Presenting and interpreting confounder and modifier coefficients. *Am J Epidemiol* 177: 292–298, 2013 https://doi.org/10.1093/aje/kws412
2. Breiman L: Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statist Sci* 16: 199–231, 2001 https://doi.org/10.1214/ss/1009213726
3. Cambria E, White B: Jumping NLP curves: A review of natural language processing research. *IEEE Comput Intell Mag* 9: 48–57, 2014 https://doi.org/10.1109/MCI.2014.2307227
4. Levey AS, Stevens LA, Schmid CH, Zhang YL, Castro AF 3rd, Feldman HI, Kusek JW, Eggers P, Van Lente F, Greene T, Coresh J; CKD-EPI (Chronic Kidney Disease Epidemiology Collaboration): A new equation to estimate glomerular filtration rate. *Ann Intern Med* 150: 604–612, 2009 https://doi.org/10.7326/0003-4819-150-9-200905050-00006
5. Biau G, Scornet E: A random forest guided tour. *Test* 25: 197–227, 2016 https://doi.org/10.1007/s11749-016-0481-7
6. Chen T, Guestrin C: Xgboost: A scalable tree boosting system. Presented at the *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, San Francisco, CA, 2016 https://doi.org/10.1145/2939672.2939785
7. Fisher R, Smailagic A, Simmons R, Mizobe K: Using latent variable autoregression to monitor the health of individuals with congestive heart failure. Presented at the 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, December 18–20, 2016 https://doi.org/10.1109/ICMLA.2016.0183
8. Allam F, Nossai Z, Gomma H, Ibrahim I, Abdelsalam M: A recurrent neural network approach for predicting glucose concentration in type-1 diabetic patients. In: *Engineering Applications of Neural Networks*, edited by Iliadis L, Jayne C, Berlin, Springer, 2011, pp 254–259 https://doi.org/10.1007/978-3-642-23957-1_29
9. Shao T, Guo Y, Chen H, Hao Z: Transformer-based neural network for answer selection in question answering. *IEEE Access* 7: 26146–26156, 2019 https://doi.org/10.1109/ACCESS.2019.2900753
10. Wang H, Cruz-Roa A, Basavanhally A, Gilmore H, Shih N, Feldman M, Tomaszewski J, Gonzalez F, Madabhushi A: Mitosis detection in breast cancer pathology images by combining handcrafted and convolutional neural network features. *J Med Imaging (Bellingham)* 1: 034003, 2014 https://doi.org/10.1117/1.JMI.1.3.034003

Received: January 8, 2021 **Accepted:** March 2, 2021