Nomogram for a machine learning model with categorical predictors to predict binary outcome

Herdiantri Sufriyana
Taipei Medical University  https://orcid.org/0000-0001-9178-0222

Emily Chia-Yu Su
emilysu@tmu.edu.tw

Taipei Medical University  https://orcid.org/0000-0003-4801-5159

Method Article

**Keywords:** nomogram, machine learning, multivariable prediction model, clinical decision rules, health information system

**Posted Date:** March 22nd, 2024

**DOI:** https://doi.org/10.21203/rs.3.pex-2587/v1

**License:** ☕️ 🌐 This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License
Abstract

We proposed a protocol to create a nomogram for a machine learning (ML) prediction model. The applicability is to improve availability of an ML prediction model in addition to a computer application, particularly in a situation where a computer, a mobile phone, an internet connection, or the application accessibility are unreliable. This protocol enables a nomogram creation for any ML prediction models, which is conventionally limited to only a linear/logistic regression model. However, this protocol only allows a nomogram creation for a model using categorical predictors to predict a binary outcome. The key stages consisted of providing the input for nomogram creation, and creating and reading the nomogram. This protocol takes 6 to 35 minutes to be completed.

Introduction

Application of artificial intelligence is emerging in medicine.1-3 Particularly, it recently applies more computational machine learning (ML), e.g. random forest, artificial neural network, in addition to statistical ML such as linear or logistic regression for predicting patient outcomes.4,5 This phenomenon is driven by larger and more diverse data due to vast digitalization worldwide.6 Big data take cost leading more expectation to increase data utilization, including to diagnose and prognosticate patient outcomes, particularly using high-dimensional predictors in electronic health records (EHRs).7 Using a sufficient sample size, a prediction model is likely more accurate if it uses high- over low-dimensional predictors, in which both are respectively what computational and statistical ML expected to be good at, as implied by the best practices in their applications.8 Regardless the number of predictors, there is a tendency to develop a prediction model by selecting the best from both MLs instead of statistical ML alone, as mostly practiced in medicine earlier decades before.14,9,10 The selection may end up with a prediction model applying computational ML. For examples, prediction model studies in medicine were more often selecting the best model by either regression or non-regression algorithms, in which none of them was consistently better than another.11-13 Yet, a computational ML results in a complex model, except a classification or regression tree (CART) algorithm, which requires integration to EHR system.14

Complexity probably causes many computational prediction models not yet being deployed in clinical settings,15 or lacking of model availability for public access.16 In the other hand, a statistical prediction model is mostly, if not all, simpler than a computational one, because statistical ML results in a risk prediction rule, formula, or nomogram. It allows a clinician or another researcher to immediately use or test a prediction model published in a scientific paper, as well as a simple computational ML model such as CART.17 One may argue a web or mobile application also allows immediate use or test for a computational prediction. Still, its accessibility is less than a tool that only requires a paper, instead of a computer, a mobile phone, or an internet connection.18 It is not seldom that an informatics tool is inaccessible when it is needed, not to mention a costly maintenance to keep a web application running.19

In this situation, applying a complex prediction model needs a nomogram. However, it is currently only designable for a linear/logistic regression model but not for other ML models.
To solve this problem, we proposed a protocol to create a nomogram for any ML prediction models. Particularly, our protocol currently enables nomogram creation for a model using categorical predictors to predict a binary outcome. While such model does not cover all prediction models, this invention may open a new frontier in ML prediction model deployment studies, i.e., future development of nomogram for other kinds of prediction models. A human user may be also limited in a capability in reading a nomogram with a large number of predictors within a reasonable time, but it is not seldom for a complex ML models using a fairly high dimension but still manually usable. This protocol aimed to delineate procedures for creating and reading nomogram of any ML models that use categorical predictors to predict a binary outcome.

**Reagents**

We used R 4.2.3 programming language (R Foundation, Vienna, Austria) to conduct all procedures, except to train the machine learning (ML) models and to obtain their variable importance. These procedures were conducted using Python 3.11.8. For R and Python, the integrated development environment software was respectively RStudio 2023.03.0 (RStudio PBC, Boston, MA, USA) and jupyterlab 4.0.11. To ensure reproducibility, we used renv 0.17.3 and Bioconductor 3.16 for R; and conda 4.12.0 for Python. For machine learning, we used a Python library of scikit-learn 1.2.2 and xgboost 1.7.3. Details on other R package and Python library versions and all of the source codes (vignette) for the data analysis are available in [https://github.com/herdiantrisufriyana/ml_nomogram](https://github.com/herdiantrisufriyana/ml_nomogram).

**Equipment**

To reproduce our work, a set of hardware requirements may be needed. We used a single machine. It was equipped by 8 logical processors for the 3.40 GHz central processing unit (CPU) (Core™ i7-4770, Intel®, Santa Clara, CA, USA), and 16 GB RAM. However, one can use a machine with only 4 logical processors and 4 GB RAM.

**Procedure**

**Providing the input for nomogram creation**

1. **Develop a machine learning model.** Current procedures for creating the nomogram requires the model to use only categorical predictors and to predict only binary outcome. Predictors with >2 categories should be binarized before model training. Categorizing a numerical candidate predictor should not be determined using the same data with those for developing the model, based on the PROBAST guidelines. The categorization was recommended to be determined by an expert panel consensus or a reference from a large-scale, independent dataset. We recommend the PROBAST guidelines either to develop or to validate the model. In our example, a benchmark dataset was used to develop the model, i.e., the Wisconsin Breast Cancer Dataset. Briefly, the modeling pipeline consisted of data partition, data preprocessing, and predictive modeling (predictor selection, model training, and model selection).
best model was random forest based on calibration (Figure 1A), clinical utility, and discrimination ability. We chose a threshold of 0.35.

2. **Obtain predictor importance of the model using the training set.** The Shapley additive explanation (SHAP) values are computed for the measure of predictor importance (Figure 2). We would use SHAP values later to determine the maximum impact per predictor to order the predictors in the nomogram.

---

### Creating the nomogram

1. **Create data to have all possible combinations of predictor values.** We create the data in order to obtain all possible values for the predicted probabilities. Note, it is important to reduce the number of predictors before developing the model, because the computation will be exponentially more expensive along with the number of predictors. Even if the computation is possible, a user may not be able to use the nomogram due to large number of predictors.

2. **Feed the data into an ML model to obtain the predictions.** If a specific threshold is determined, the predicted probabilities are obtained first, then the predicted positives are defined if the predicted probabilities are greater or equal to the chosen threshold.

3. **Design the nomogram for either positive or negative prediction.** For positive prediction, the procedures below are started from predictor with the highest to lowest maximum impact by SHAP values. For negative prediction, the procedures below are started from predictor with the lowest to highest maximum impact of SHAP values. Here are the procedures to obtain the design:

   a. Select predictors accumulatively from the first to current predictor;

   b. For the first iteration, pass to step c. For next iterations, filter out samples with the same predictor values with those of previous iteration;

   c. For each combination of available predictor values, compute minimum and maximum probabilities for each group of predictor; and

   d. Filter out minimum $\geq$ threshold and maximum $<$ threshold respectively for positive and negative prediction.

4. **Draw the nomogram design.** The filtration results are merged for positive and negative predictions. A plot is created in two dimension (Figure 3). The x-axis is nomogram-reading iterations (see “Reading the nomogram”) from the first to the last iteration (left-to-right positioning). Meanwhile, the y-axis is predictors from the highest to lowest maximum impact (top-to-bottom positioning). In our example, we plotted predictor values with colors of red and cyan for predictors respectively with negative and positive values. The plot panels are divided into positive (left) and negative (right) predictions.
Reading the nomogram

1. Find a predictor with a positive value from the higher to lower maximum impacts; if none, find an iteration at which all available rectangles are negative and jump to step 2a. In our example, we randomly picked two samples, in which 2 of 5 predictors with positive values and were predicted positive (Figure 4A) and negative (Figure 4B). Respectively, the predictors with positive values were epithcsize_6/barenuclei_10 and barenuclei_1/epithcsize_2. We started with a positive predictor, i.e., either epithcsize_6 or barenuclei_1, which has a maximum impact higher than that of another predictor in each sample (Figure 4).

2. Match the colors of available rectangles with the predictor values from the earlier to later iterations. In our example, we matched the colors at iterations 1 and 6 respectively for the predicted positive and negative samples (Figure 4).

   a. If they are matched, then draw a vertical line down to identify which panel the iteration belongs to. The predicted positive sample was already matched for epithcsize_6 at iteration 1 which belonged to positive prediction panel (Figure 4A). In this iteration, the rectangle was only available for epithcsize_6; thus, we did not need to check the values of other predictors.

   b. If they are not matched, then draw a horizontal line to the next right positive rectangle and repeat step 2 until a rectangle of the next positive predictor is available downward. For the predicted negative sample, the predictor value of barenuclei_1 were not matched at iteration 6. Thus, we checked the next right positive rectangle until a rectangle of the next positive predictor is available downward, i.e., epithcsize_2 at iteration 9 (Figure 4B).

3. Repeat step 1. In our example, we repeated step 1 with another positive predictor, i.e. epithcsize_2, for the predicted negative sample. Eventually, the predicted negative sample was matched at iteration 11 which belonged to negative prediction panel (Figure 4B).

Troubleshooting

Please kindly see Table S1 in Supplementary Materials.

Time Taken

Please kindly see Table S2 in Supplementary Materials.

Anticipated Results
The number of predictors are expected to manageable by a human user. The nomogram design is supposed to be cascaded in earlier iterations for either predicted positive or negative samples. The former likely requires less available rectangles in relative the number of maximum iterations. A human user needs a training to read the nomogram.

References


**Acknowledgements**

This protocol was funded by: (1) the Postdoctoral Accompanies Research Project from the National Science and Technology Council in Taiwan [grant number NSTC111-2811-E-038-003-MY2] to HS; (2) the Ministry of Science and Technology in Taiwan [grant numbers MOST110-2628-E-038-001, MOST111-2628-E-038-001-MY2] to ECYS; and (3) the Higher Education Sprout Project from the Ministry of Education in Taiwan [grant numbers DP2-111-21121-01-A-05, DP2-TMU-112-A-13] to ECYS.

**Figures**
Figure 1

Best model predictive performance based on: (A) calibration; (B) clinical utility; and (C) discrimination. A calibration plot (A) is shown with descriptive regression intercept and slope, Brier score, and frequency distributions for predicted negatives (=0) and positives (=1). A decision curve (B) is shown to depict clinical utility by threshold-wise net benefits comparison to treat-all and -none scenarios (respectively all were predicted positives and negatives). An ROC curve (C) is shown to depict discrimination ability by threshold-wise sensitivities and specificities. The chosen threshold (=0.35) is shown as a dashed, vertical line for each panel. CI, confidence interval; ROC, receiver operating characteristics.

Figure 2

Best model predictor importance based on SHAP values. Each predictor in y-axis is follow by a rank by the maximum impact of the predictor on model output. SHAP, Shapley additive explanation.
Figure 3

Nomogram example for the best model. Assessment order is from top to bottom and from left to right.

A

Figure 4

Examples of reading nomogram for the best model in predicting: (A) positive; and (B) negative. The numbered labels correspond to the step for reading the nomogram.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- TableS1andS2.pdf