Paper AR10

Quantifying Dynamic Risk with Predictive Analytics

Connor English, EdjAnalytics LLC, Louisville, Kentucky, USA
Susan Olson, EdjAnalytics LLC, Louisville, Kentucky, USA

ABSTRACT
Predictive analytics enables innovative approaches for leveraging real-time clinical data to assess risk. Predictive methodologies and applications that use dynamic patient information to update health risk assessments over time are especially valuable. For this purpose, two healthcare applications have been developed that assess either risk of hospital readmission or probability of elective, surgical procedures. Both techniques use electronic health record information as a source of features for predictive modeling. These applications are rooted in health system operations and may provide innovative value for monitoring and managing clinical operation risk.

INTRODUCTION
This paper provides readers with a walkthrough for using predictive analytics to assess risk by describing the general workflow and providing two real-world examples. The predictive analytics methods discussed have broad applicability to numerous applications and industries. The authors intend to present a stepping off point for readers looking to use predictive analytics to assess and manage clinical operation risk in a forward-looking way. While these examples are not specific to clinical trial execution, they provide process templates for risk identification in a dynamic data environment.

To describe the scope of the paper, the first portion of the paper called “Predictive Modeling Workflow” provides a primer from data selection to predictive modeling. The second portion of the paper, which is broken into two sections called “Healthcare Application 1” and “Healthcare Application 2,” provides two detailed, practical examples of predictive modeling related to risk management that the authors encountered in a real-world, healthcare setting.

PREDICTIVE MODELING WORKFLOW
The workflow of predictive modeling can be characterized by a series of processes. These processes are described within the sub-sections that follow.

LOCATING AND SELECTING DATA
The key to navigating this step of the workflow successfully is flexibility. First the modeler needs to identify and locate the data sources that will be used for the analysis. Typically, the modeler will identify historical data and outcomes to use as the basis of the predictive model. To identify the data sources appropriately, the modeler needs to build a working relationship with the end user(s) of the model. These users provide critical subject matter expertise, including establishing the goals and objectives of the model and identifying prospective data sources for modeling.

To benefit the most from predictive modeling, the modeler needs to maintain objectivity in selecting the data sources for inclusion in the predictive model. The end user may, intentionally or unintentionally, introduce their own biases or other limitations on the data sources the modeler should consider. For example, the end user seeking to manage risk in a clinical trial might only present health record metrics and clinical history, whereas the modeler may see benefit in expanding the scope to include social determinants of health for a more advantageous model. The modeler should be wary, though, that the availability and connectivity of the data in the implementation setting will ultimately dictate what may be used.

The next step is to locate the data sources and the stewards of those data sources. End users can often be helpful in making those connections. When working with the data of another data steward, the modeler is stepping into the steward’s domain. The infrastructure for how data is stored within each business unit can be vastly different even within the same organization. The modeler needs to work with each data steward to determine the best approach for both practical access and data protection.

Some data may be sensitive as is the case with the healthcare field, which typically contains data that can be classified as personally identifiable information (PII) or protected health information (PHI). PII is any information that can be used to identify, locate, or contact a specific individual. PHI is any health information belonging to a specific individual that was created by or delivered to a health provider. The modeler needs to maintain awareness and protocols for handling protected and sensitive information detailed in business associate agreements or federal acts such as the Health Insurance Portability and Accountability Act of 1996 (HIPAA).

Finally, the modeler needs to consider what database, computing environment and analytics tools to use for predictive analytics. Comprehensive coverage of these topics is beyond the scope of the document. To exemplify, one setup could have data stored in a PostgreSQL database that is accessed through a remote virtual private network and utilizes the Python programming language and associated packages like Pandas, NumPy and Scikit-learn.

DATA ASSESSMENT

The next step in the process is to assess the selected datasets to discover what data is usable for predictive modeling. In many cases, at least some of the input variables are collected or stored in such a way that will not support modeling. For instance, the data collected could be too sparse or have inconsistent formatting. In another example, it may not be possible to explicitly link two datasets through a common key. A modeler should also consider any limitations in the ongoing collection of data for future predictions. The data assessment process helps resolve these and many other scenarios encountered in real-world data.

The data assessment process also lays the foundation for feature engineering, which is discussed in the next sub-section. Some basic guidelines for a data assessment are included in Figure 1.

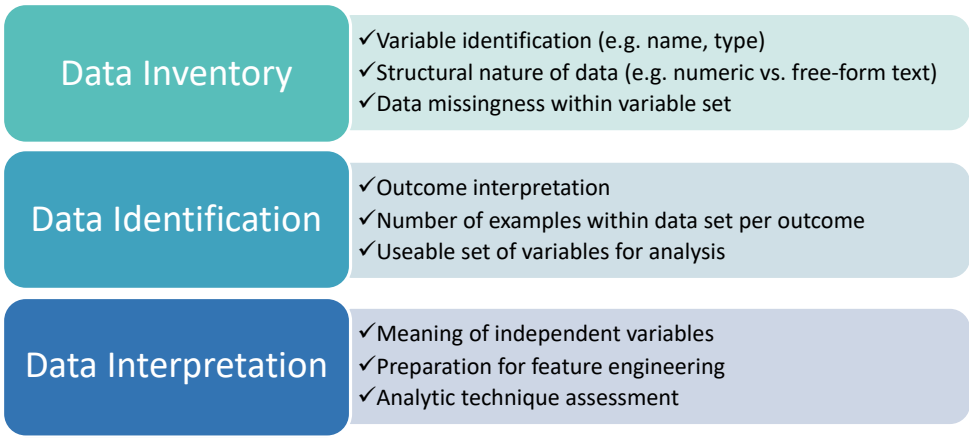

Figure 1: Assessing data readiness and applicability for modeling

The final sub-bullet of Fig. 1, “Analytical technique assessment” deserves some additional explanation. After the data audit is complete, the modeler should have a tangible assessment of the number of observations and number of variables suitable for building a model. The assessment will help pare down to the types of predictive modeling techniques that are most suitable for the problem at hand. For instance, to meet the end user’s desired accuracy requirements, a sufficient number of examples must be available to train, test and validate a machine learning model. Within the set of machine learning models, further constraints exist depending on the type and complexity of the modeling scheme. Datasets that would not achieve requirements for accuracy using complex machine learning may be more suitable to linear regression or decision tree techniques. Large datasets may present no restrictions. Finally, if the end user(s) desires to have transparency into the variables which influence the outcomes or predictions, the modeler might eliminate the use of “black box” type models despite the data suitability for such techniques.

FEATURE ENGINEERING

Engineering features is the process of taking units of information in the data provided and transforming them into actionable information. It can be thought of as taking one or a combination of columns in a data table and creating a new column with more useful information. Data can be stored inside different tables that can be joined by like features (a technical term to describe independent variables used in modeling). Features can be defined and/or constructed in several different ways.

Data cleaning often takes place along with feature engineering; it is the process of identifying and correcting inaccurate or corrupted information when constructing finalized features. A simplified example of a record that needs cleaning could be the identification of a record that indicates a 66-year-old individual weighs 45 pounds. If this individual was last seen in a pediatric setting, then we could deduce that perhaps when age was being entered into the system the number 6 was accidentally entered twice. For larger datasets, the process of cleaning data often requires the creation of code that systematically identifies then corrects or deletes records with invalid data.

As a simple example of feature engineering, consider data for smoking status that is collected in one column as “Non-smoker”, “Occasional Smoker”, “Smoker”, “Frequent Smoker”, and “Former Smoker”. There are a number of problems with the way this data is collected. We are interested in learning from this data whether smoking has altered

an individual's lungs. Therefore, what we want to know is if they are a "Smoker" or a "Non-smoker". For this instance, we might say if someone is labeled as anything other than "Non-smoker" then they should be labeled as "Smoker". This new actionable feature would then be created from the previous information that would have been more granular than necessary for the given purpose and potentially problematic if self-reported. Engineered features may be generated using practices such as one-hot encoding, binning, linear combinations, etc. Any time information is transformed into a new and more useful format for input into the model, features are engineered.

DATA AGGREGATION/IMPUTATION

Aggregation might be necessary depending on how the data was collected. If a new line of data is collected for a given individual every hour for the duration of a hospital stay, the data could contain redundancies. To utilize all of it would be misleading and could jeopardize the validity of a model as it would be weighted toward the patients with longer stays. The data could therefore be aggregated or grouped by each individual patient with the collective information reduced to means or medians.

The modeler may also address certain degrees of missingness through imputation. For example, if height is recorded for all but a few patients out of millions of individuals, it would not be unreasonable to replace a missing value with a measure of centrality from the rest of the data. The modeler could possibly forward-fill information with the patient's height from a previous visit or even use another model to predict an individual's height. Another approach would be to consider dropping the patient from the dataset altogether. Each case is different and, before making a final decision, the pros and cons of available options should be evaluated.

SPLIT DATA INTO TRAINING, VALIDATION AND TESTING SETS

The goal of most projects is to have a predictive model that can take in new information and produce an assessment of the most likely future outcome. When creating the model, the data needs to be split in a way that allows for objective measuring the performance of the model on unseen data. The training set has the most data in it (70-80% of the full set) and is used to train the model. The validation set (10-15% of the full set) is used to evaluate preliminary models as features are engineered and different model types are evaluated. The testing set (10-15% of the full set) is not touched until the model is finalized to determine the diagnostic stats of the model on unseen data.

EXECUTE PREDICTIVE MODELING

This step is iterative as it might send the entire process back to feature engineering multiple times. The features are tweaked, the model type is determined, and hyperparameters (static properties of the model set before training such as how fast it should learn or how many rounds of training should occur) are tuned using the validation set predictions as indications of how the model is performing. Once the model has satisfactory measures with the validation set, the modeler moves forward to the testing set.

EVALUATE PERFORMANCE

Using the testing set, the finalized model diagnostics are determined. These are the measures that best reflect how the model will perform outside of the development and training environment. The diagnostic that is typically used is the area under the curve (AUC) of the receiver operating characteristic (ROC) curve, which is demonstrated in Figure 2. This curve is created using different probability decision thresholds that report the true positive and false positive rates of the model on the testing set. These rates are charted with true positives on the y-axis and false positives on the x-axis. A perfect model would have all true positives and zero false positives at any given probability and would result in an AUC of 1 and an imperfect model would have an AUC of 0.5 which would be no better than a coin toss. The closer the AUC is to 1, the better the model performance from the metric perspective.

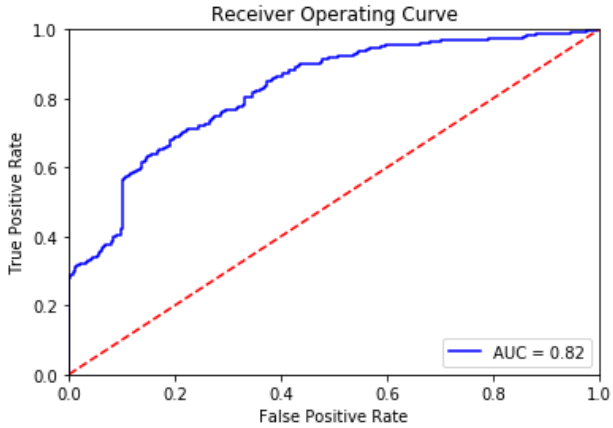

Figure 2: Example of an ROC curve

HEALTHCARE APPLICATION 1 – HOSPITAL READMISSION

After being penalized for a high rate of readmission among congestive heart failure (CHF) patients through the Hospital Readmissions Reduction Program (HRRP) established under the Affordable Care Act, a healthcare system sought to use predictive analytics to analyze and assess the risk of readmission among patients covered by Medicare. Readmission to the hospital can be detrimental as it might exacerbate a patient's ailment since their treatment was delayed, resulting in a more expensive and lengthier stay.[1,2] In 2017, HRRP penalized 79% of hospitals estimating $528 million in penalty costs.[3] Additionally, this longer stay due to hospital readmission can affect a patient's quality of life as they spend more time hospitalized and a large number of readmissions may contribute to a bad reputation for a hospital and result in lost clientele. Therefore, it was imperative for this healthcare system to better understand who would and wouldn't be at risk for readmission within their walls.

Three years of de-identified electronic medical record (EMR) data for Medicare patients with CHF that had stayed in the hospital system were delivered by the client for analysis. This EMR data included past diagnoses, procedures, consults, labs, vitals, medications, demographics, and the Hospital Specific Report compiled by Medicare indicating unplanned readmissions. Model implementation and interpretability within the hospital system led to choosing a logistic regression for the final model for client delivery. A logistic regression is a statistical model that utilizes a logistic function for classification problem as shown in Figure 3.

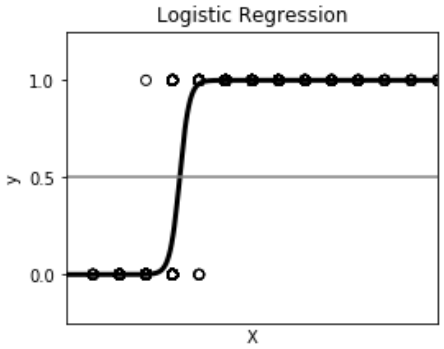

Figure 3: Example of a logistic regression where the outcome is classified as 1 or 0

In this case, the outcome is that a patient either readmits within 30 days or does not. Through the predictive modeling workflow, features were then engineered with the logistic regression in mind. Some of the features engineered were summary vital measures, defined comorbidities, previous visit indicators, and discharge disposition. The features that had the greatest predictive power in the model for readmission risk were found to be:

- The type of facility where a patient is discharged;
- The number of previous encounters where CHF was listed as a co-morbidity;
- The number of times the patient visited the hospital in the previous 90 days
- Potassium, Hemoglobin and Creatinine levels.

The resulting dataset with desired features was then split into training (60%), validation (20%), and testing (20%) sets and predictive modeling was then executed and evaluated. Findings from the model were reinforced by conventional wisdom within the healthcare setting and shed additional light on factors that were previously not given consideration. The importance of vitals when choosing to discharge a patient is traditionally understood. However, equal consideration is not always given to discharge plans and visit history, which are revealed to be more predictive of readmission risk. The final model achieved an AUC of 0.7 on the unseen testing data indicating high accuracy. Comparing this model with a more complex machine learning model did not show a change in accuracy that would invalidate using the logistic regression.

In addition to its predictive capabilities for evaluating risk, the model helped empower the hospital system to create discharge plans effectively through dynamic assessment of risk for Medicare patients with CHF. Figure 4 shows an example of a readmission risk dashboard for a patient. Each day, as information is collected, the model can interpret and report the risk of readmission for each individual patient to avoid discharging higher risk individuals. This type of dynamic information provides aid to hospitals to keep from improperly discharging patients and incurring costly penalties.

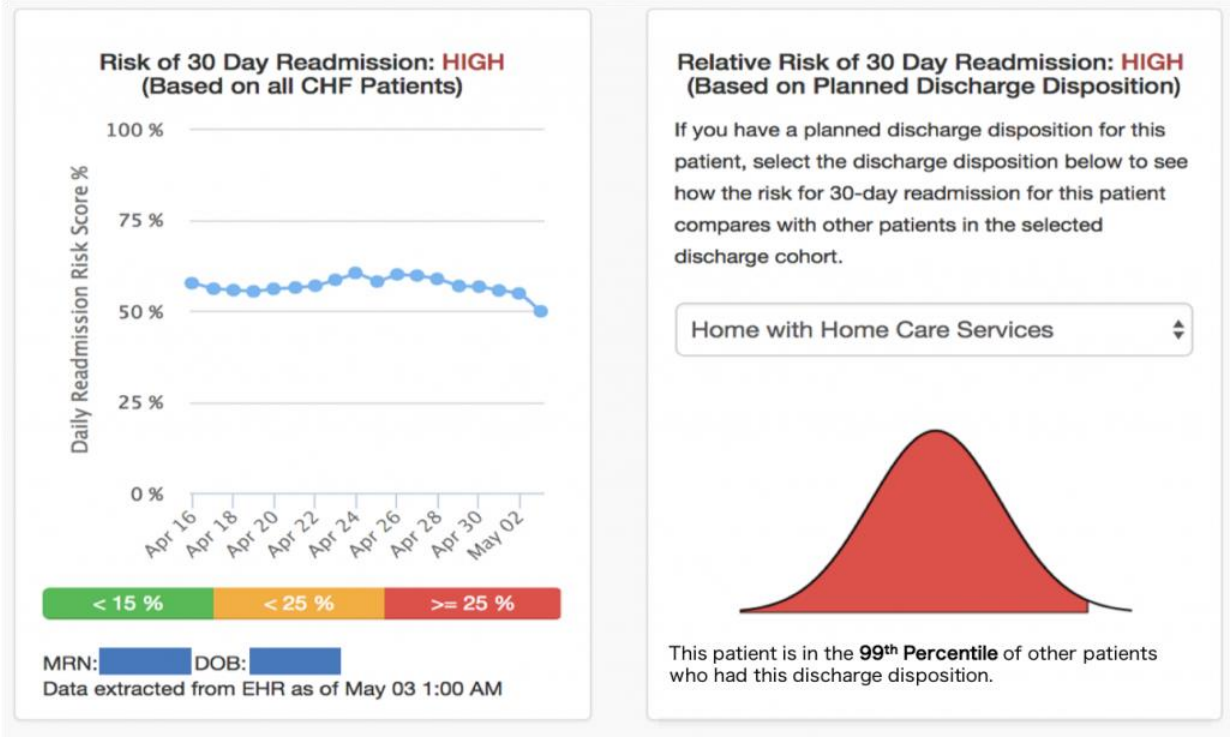

Figure 4: Example of a readmission risk dashboard for an individual patient

HEALTHCARE APPLICATION 2 – ELECTIVE PROCEDURES

Aiming to improve patient outreach related to elective procedures, a regional hospital system sought a predictive analytics solution to assess risk/need at a patient level. The elective procedures of interest were hip replacements, knee replacements, and bariatric surgeries. The targeting logic in use at the time was an age threshold for hip and knee procedures and a body mass index threshold for bariatric procedures. The goal of this project was to utilize the complete medical and hospital interaction history along with demographic information present in the hospital's Electronic Medical Records (EMR) system to identify specific individuals who are most likely to require each procedure to optimize patient selection for education and outreach.

One unique aspect of this predictive challenge deals with the temporality of the data. In trying to predict whether an individual will have a procedure at one point in time, we must take into account data before that point while excluding future information. Otherwise, we would be training the model to key in on data that would only be present as the procedure is happening (e.g. anesthesia being administered). To overcome this obstacle, data needed to be aggregated for a specific timeframe relative to a patients first encounter. To determine which timeframe was most appropriate, timeframes of one month, one quarter, and one year were tested. Evaluating logistic regressions on a balanced subset of the population revealed the best performer for all three procedures to be one quarter of aggregated patient data predicting one quarter into the future.

After testing a variety of machine and deep learning methodologies with the one-quarter time lag, the best performing approach utilized an Extreme Gradient Boosting model (XGBoost). This type of model follows the same principal of gradient boosting as a standard gradient boosted model (GBM) while utilizing a more regularized model formalization to control over-fitting and leverages the structure of computing resources to speed up computing time and optimize memory usage. This type of machine learning model is often used in classification problems in which outcomes are defined categories. In this case, a patient either elects to have a specified procedure or does not. A GBM produces an ensemble for short decision trees in a successive manner so that each new tree generated accounts for loss created by the previously generated tree. Figure 5 illustrates the ensemble of decision trees. These decision trees, when used individually, are weak predictors. Though, when used in a stage-wise manner, the trees together create one strong predictor.

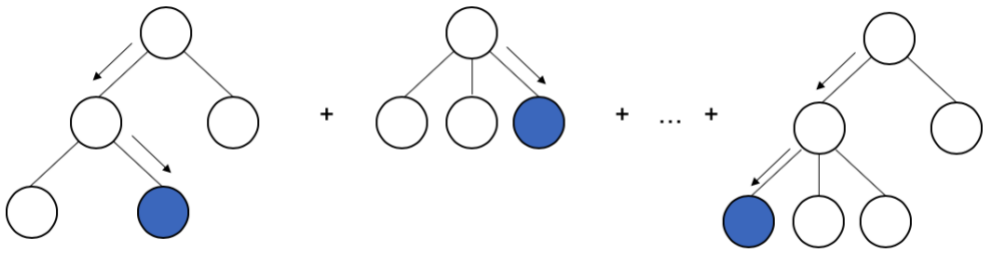

Figure 5: Example of an ensemble of decision trees for a GBM

The dataset with desired features was then split into training (75%), and testing (25%) sets and predictive modeling was then executed and evaluated. Features for vitals alongside age were consistently found to be the most important features when predicting the different elective procedures. Some features such as previous history of the same procedure, specific diagnoses such as osteoarthritis, and previous encounter history were unique to the separate procedure models. The final models included approximately 80 features. While age and BMI were consistently found to be some of the most important features in the models, models using only these variables had about 0.25 lower AUCs for each outcome. As shown in Figure 6, the final models achieved AUCs between 0.88 and 0.93 on unseen holdout data, indicating very high performance. These robust models grant hospital systems the ability to effectively establish targeting logic behind patient outreach, through dynamic assessment of individualized risk for future elective procedures.

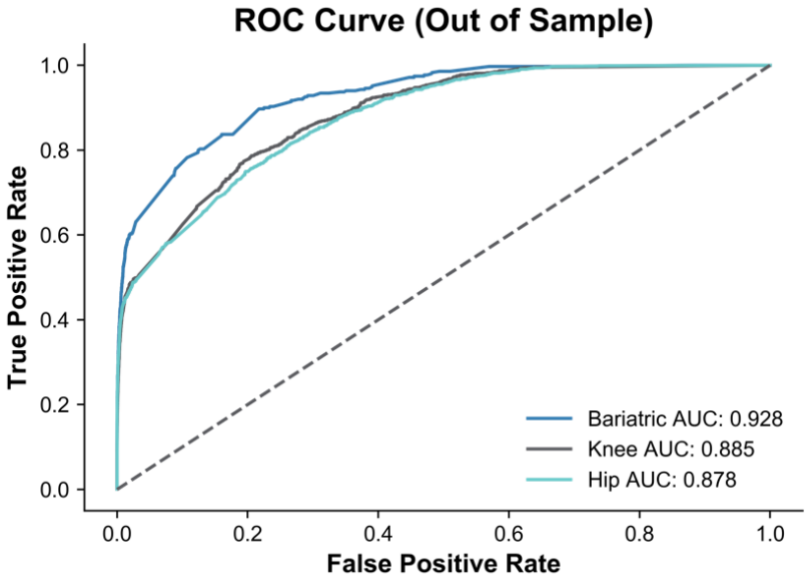

Figure 6: ROC curves for elective procedure models on out-of-sample data

SUMMARY

The paper presented principles for designing predictive analytics for risk assessment that may be generalized to many fields. The paper also presented two real-world healthcare examples of predictive modeling to assess patient risk. Both examples had considerations of temporal or dynamic data that impacts future risk and offered insights on how to tackle time-varying facets.

REFERENCES

1. Jencks SF, Williams MV, Coleman EA. Rehospitalizations among patients in the Medicare fee-for-service program. N Engl J Med. 2009;360:1418–1428. [PubMed]
2. Bueno H, Ross JS, Wang Y, Chen J, Vidan MT, Normand SL, Curtis JP, Drye EE, Lichtman JH, Keenan PS, Kosiborod M, Krumholz HM. Trends in length of stay and short-term outcomes among Medicare patients hospitalized for heart failure, 1993-2006. JAMA. 2010;303:2141–2147. [PMC free article][PubMed]
3. Boccuti, C. and Casillas, G. (2017). *Aiming for Fewer Hospital U-turns: The Medicare Hospital Readmission Reduction Program*. [online] The Henry J. Kaiser Family Foundation. Available at: https://www.kff.org/medicare/issue-brief/aiming-for-fewer-hospital-u-turns-the-medicare-hospital-readmission-reduction-program/ [Accessed 2 Jan. 2019].

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Connor English
EdjAnalytics
732 E. Market St, Suite 203
Louisville, KY, USA 40217
Work Phone: (502) 208-4088
Email: cenglish@edjanalytics.com
Web: www.edjanalytics.com