An Anytime Querying Algorithm for Predicting Cardiac Arrest in Children: Work-in-Progress

Michael A. Skinner[1,2](✉), Priscilla Yu[2], Lakshmi Raman[2], and Sriraam Natarajan[1]

[1] University of Texas at Dallas, Dallas, TX 75080, USA
[2] University of Texas Southwestern Medical Center, Dallas, TX 75390, USA
mas140130@utdallas.edu

Abstract. Cardiac arrest (CA) is a devastating complication for children in the cardiac intensive care unit (CICU). We developed an "anytime" algorithm to predict CA, using the first few hours of EHR data for initial approximation, and then using information from subsequent time periods to augment the predictive model, improving performance at each iteration. Our initial empirical evaluation on EHR CICU data shows that the model achieves significantly higher performance than learning with all the available data at each iteration when predicting CA inside CICU.

Keywords: Gradient boosting · Anytime algorithms · Cardiac arrest

1 Introduction

Congenital heart disease is a significant cause of death and morbidity in neonates and children. A devastating complication in children with heart disease is cardiac arrest (CA). If the condition is not quickly reversed, there will be significant damage to other organ systems and possibly death [4]. An important challenge is to develop machine learning algorithms using the electronic health record (EHR) to predict which critically ill children in the cardiac intensive care unit (CICU) are at increased risk of cardiac arrest [7]. A desirable property for these algorithms is the ability to generate a reasonable approximation of the desired result with few data, and then to refine the prediction with time as more data accumulate. Such *anytime algorithms* [8] exhibit improved accuracy with time, allowing for reasoning continuously as more data arrive.

We devised an anytime algorithm to compute the probability of CA in children with congenital heart disease in the CICU. Using the Functional Gradient Boosting paradigm [2], we create a set of regression trees whose cardinality grows as more clinical data accumulate in the EHR. The data are collected in increments of several hours, and at each increment, new trees are concatenated to the previous model in a stacking fashion, improving predictive ability with time.

S. Natarajan—Supported in part by NICHD grant 1R01HD101246 and The Precision Health Initiative of Indiana University.

M. Michalowski et al. (Eds.): AIME 2022, LNAI 13263, pp. 353–357, 2022.
https://doi.org/10.1007/978-3-031-09342-5_34

2 Methods

2.1 Patients and EHR Data

We obtained an exemption from the UTSW IRB as the data are deidentified. Hemodynamic, laboratory, demographic, medication data were collected from EHR in patients managed in the pediatric CICU at Children's Medical Center of Dallas. Records were obtained for 160 patients (age ≤ 21) who experienced CA over a 10 year period; EHR data were collected from the 48 h prior to CA. We also collected the first 48 h of data from 711 control (non-CA) patients selected at random from CICU patients managed during the same time period.

The ages in each group ranged from 1 day to 20 years (average age in arrest is 2.88 while in control is 3.45). *Our goal is to predict the probability of CA progressively from 13 h before the arrest to the hour of arrest.*

We extracted 11 EHR features, listed in Table 1. Each of the features was discretized into three "bins", using scikit-learn [6] "kmeans" strategy. To address the challenge of working with pediatric patients whose normal vital signs vary with age, and to account for the CICU patients where "normal" values may be quite abnormal compared to healthy patients, many of the features were normalized to reference values obtained by computing average parameter values over the first four hours of the 48 h trajectory. These features were selected by a pediatric ICU physician, and are marked with "*" in Table 1.

Table 1. Clinical features and measurement units used in predictive models. Those marked with * are standardized for each patient (explained in text).

Feature	units
Pulse rate	*
Diastolic blood pressure	*
Oxygen saturation	%
Urine output	*
Base excess	*
Anion gap	mEq/L
Fraction inspired O2 (fiO2)	*
Vasoactive inotropic score (VIS)	(None)
End tidal pCO2	mmHG
Near infrared spectroscopy rso 1	*
Near infrared spectroscopy rso 2	*

We discretized the time into one-hour increments; when multiple feature values were present during the hour, the mean of the values was used. Finally, to devise models that operate using symbols rather than simple features, and to avoid the imputation of missing results, we converted the data into predicate logic format. For example, the predicate "pulse(subj1001, LE 0.9, 16)" indicates that subject 1001 at 16 h prior to cardiac arrest exhibited a pulse rate less than or equal to 0.9 times his/her reference pulse rate.

2.2 Boosted Predictive Regression Trees

After transforming EHR data into a relational predicate format, we exploit the tools of Statistical Relational Learning (SRL) [3] to create models predicting cardiac arrest. In particular, we employed the SRLBoost framework described by Natarajan *et al.* [5] to create boosted sets of weakly-effective regression trees trained to generate the probability of our target concept, cardiac arrest.

Algorithm 1. Pseudocode to construct anytime predictive model

```
Input: positive and negative example trajectories with time-indexed predicate
       facts, $k$ hours per model stage, $T$ hours in trajectories, $l$ number of trees
       per stage
Output: Boosted predictive model
Initialize: model M = {}
$hourLast = 0$            // Last hour for current model stage.
while $T - hourLast < k$ do
    (We add stages until trajectory ends.)
    $hourLast = hourLast + k$
    $currentFacts = \{facts | fact.time \leq hourLast\}$
    $m = SRLboost(M, l, currentFacts)$
    $M = M + m$
end while
return $M$
```

Briefly, regression trees are constructed in a top-down manner [1] so that each decision node represents an EHR finding at a particular time prior to arrest (in the positive example learning set). At each iteration, the goal is to identify the predicate that maximizes the weighted variance. Leaf nodes contain regression values which can be converted a probability value. The algorithm employs single path semantics – i.e., each instance only satisfies one path in each tree – and thus returns one regression value from each tree. They are then added across the trees and converted to a probability by applying the sigmoid function. The depth of the trees was limited to 4.

In Algorithm 1, the model M is initially empty. Then, l trees are constructed using the observations annotated with times from the first k hours of the trajectory. Then, the next stage of the model is created by training an additional l trees using the initial data augmented by data from observations from the next k hours, and so on until there are fewer than k hours remaining in the trajectories, and the predictive model M is returned.

3 Results

We trained the models using 75% of the patient examples, and tested them with 25% of the examples. Example down sampling was employed to deal with the significant class imbalance.

For training and prediction, we used EHR data from 16 h prior to cardiac arrest (in those who arrested) and the final 16 h of data (of the 48 h of data collected) in the control patients. The model was initialized using the first 4 h of data (16-13 h before arrest), creating a set of 5 boosted regression trees. For each subsequent 4 h period, another 5 trees were learned using the data from all of the hours seen so far; the new trees are concatenated to the model obtained thus far. So, after the data over the entire 16 h trajectory are evaluated, the final model consists of 20 trees.

The results are seen in Fig. 1. The models include our concatenated model (solid red line), a model derived without concatenating previous models but with increasing number of model trees learned from scratch (dashed blue line), and a baseline model where 20 trees are learned for each time point (dotted green line). Presented are the mean, standard deviation of 5 models. The model created as a concatenation of models created over previous time periods exhibits the best predictive performance.

We conclude that a boosted model using EHR data converted to a time-indexed predicate format exhibits improved predictive performance when the model is constructed by iteratively adding new stages to the existing model as new, more recent results become available.

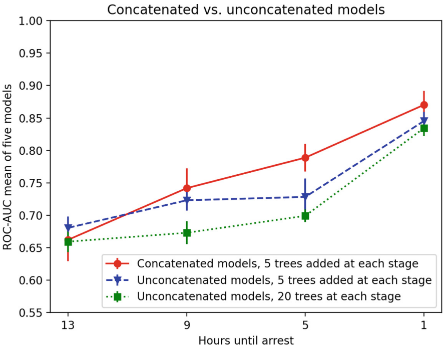

Fig. 1. ROC-AUC for the models predicting CA.

4 Discussion

It is interesting to consider why the concatenation of models by adding regression trees to a previously constructed model provides better predictive performance than simply starting from scratch at each time point, using all of the earlier data to construct a new set of regression trees. We speculate that physiologic data from later time points, which are closer to the time of cardiac arrest in the training set, are more useful in prediction. Thus, whenever a new model is created, the greedy nature of the construction selects later data to create the trees, ignoring earlier data that may be less helpful. However, when we create a model using only clinical facts from a limited period of time, as in the concatenation scheme, the model is forced to do what it can with those data to improve prediction, using data that might be ignored in a full-on greedy algorithm. This finding may have implications for other greedy predictive algorithms.

One advantage of our approach is the fact that we produce fairly robust predictive models from a relatively small number of subjects. In particular, we create the models using 75% of positive examples, or about 120 subjects. This contrasts with deep neural network models, which often require thousands of examples for model training. Moreover, in this non-parametric model, there are only a very few hyper-parameters to select, avoiding the necessity for extensive model tuning. Finally, the conversion of clinical results into a predicate format discharges the need to impute missing data elements; the models depend only on the findings actually present in the medical record by applying a closed world assumption.

It is the hope that any medical predictive model will extract causal factors responsible for the outcome of interest; then, those managing the patient might be able to intervene on the identified cause to improve outcome. Such usefulness in turn is dependant on model interpretability so that model findings can be understood, which is a challenge in many machine learning algorithms. For example, deep neural network models (DNNs) are notoriously difficult to understand. Although we are not yet able to meaningfully understand how the boosted trees in our model can be used to guide medical treatment, the recognizable predicates represent meaningful medical concepts. There exist methods that reweigh samples based on the learned boosted model to learn a single, more interpretable, tree. These techniques are similar to knowledge distillation in DNNs, but do not generally create a tree that is logically equivalent to the boosted model, and are therefore unsatisfactory for this predictive task. Explainability is a topic for future research. Moreover, we aim to more formally evaluate whether there are advantages to the predicate representation of data as against the more commonly used vector representation.

Clinical significance: Even as the medical and surgical management of children with cardiac disease has improved outcomes, CA in the CICU remains a significant challenge. In this preliminary work, we have devised an anytime algorithm to predict this devastating complication; the model holds promise that children at risk of CA can be identified early, allowing intervention and possibly CA prevention.

References

1. Dietterich, T.G., Ashenfelter, A., Bulatov, Y.: Training conditional random fields via gradient tree boosting. In: Proceedings of the Twenty-First International Conference on Machine Learning, p. 28 (2004)
2. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. **29**, 1189–1232 (2001)
3. Getoor, L., Taskar, B.: Statistical Relational Learning (2007)
4. Meyer, L., et al.: Incidence, causes, and survival trends from cardiovascular-related sudden cardiac arrest in children and young adults 0 to 35 years of age: a 30-year review. Circulation **126**(11), 1363–1372 (2012)
5. Natarajan, S., Khot, T., Kersting, K., Gutmann, B., Shavlik, J.: Gradient-based boosting for statistical relational learning: the relational dependency network case. Mach. Learn. **86**(1), 25–56 (2012). https://doi.org/10.1007/s10994-011-5244-9
6. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
7. Ruiz, V.M., et al.: Early prediction of clinical deterioration using data-driven machine learning modeling of electronic health records. J. Thorac. Cardiovasc. Surg. **164**(1), 211–222 (2021)
8. Zilberstein, S.: Operational rationality through compilation of anytime algorithms. AI Mag. **16**(2), 79–79 (1995)