Early Detection of Sepsis from Real-Time Patient Vitals

Adam Hedges, Hazel John
Georgia Institute of Technology, Atlanta, GA, United States

Video presentation available at https://youtu.be/Sxuk_y1dEXI

Abstract

Our work is based almost entirely on that of Dr. Thomas Desautels et al., who developed the InSight predictive model for the advanced detection of sepsis based on readily available patient vital signs.[1] Through this retrospective study, the Desautels team was able to achieve predictive results superior to the commonly used bedside measurements performed at routine intervals. We intend to demonstrate the predictive capabilities of a slightly simplified classification model based purely on patient vital signs. In other words, we are attempting to discover if patient vital signs themselves are indicative enough for the prediction of a septic infection, without requiring any advanced clinical indicators which may themselves change over time.

Ultimately we were able to produce a predictive model that is comparable to InSight, achieving an AUROC score of 0.83 at 1-hour prior to the suspicion of infection, 0.82 at 4-hours prior, and 0.79 at 8-hours prior. While we did not test a 0-hour prediction window, we were able to exceed InSight's AUROC score of 0.74 at the 4-hour window. An exact AUROC score for the 1-hour window was not reported for InSight.

Introduction

The ability to accurately detect the onset of sepsis will provide an easily verifiable signal, much like an adverse reaction to medication, that medical personnel can quickly respond to. The goal of this signal should be to consolidate the warning signs of a sepsis infection into a simple real-time alert. Aside from the *InSight* classifier, there are several other works on the subject of sepsis prediction which provided further inspiration for our project.

First, Shimabukuro et al.[2] conducted a randomized controlled clinical trial at two med-surg ICUs at the University of California, San Francisco Medical Center. Using a machine-learning severe sepsis alerting algorithm, they found a statistically significant decrease in hospital length-of-stay and in-hospital mortality. This approach was similarly based on real-time patient vital signs, and as such provided additional guidance towards the use of vital signs in this project.

Second, researchers at the Hospital of the University of Pennsylvania[3] performed a "silent test" of a machine learning algorithm to predict early onset of sepsis, using a large combination of EHR data points. This real-world test correctly identified 98% of patients that were later diagnosed with sepsis, further providing justification that patient vital signs and passive EHR data can be used successfully in the prediction of a Septic infection.

Method

The source for our experiment is the MIMIC-III database.[4] This openly available database of historic patient EHR data is actively developed by the MIT Lab for Computational Physiology. MIMIC-III contains several large tables with related patient information, but we require only PATIENTS, ICUSTAYS, CHARTEVENTS, PRESCRIPTIONS, and MICROBIOLOGYEVENTS. In an attempt to replicate the ease-of-implementation of the *InSight* classifier, we have limited our relevant patient feature set to the following vital signs:

1) Systolic blood pressure (non-invasive)
2) Diastolic blood pressure (non-invasive)
3) Heart rate
4) Respiration rate
5) Body temperature
6) Capillary oxygen saturation (SpO_2)
7) Glascow Coma Score (motor, verbal, eye opening)

In general this feature set aligns with that employed by *InSight*, but there are several considerations worth mentioning in detail. First, within the MIMIC-III database there are different measurements available for many of these routine vital signs. For example, blood pressure can be captured via a doppler device, a non-invasive cuff, or an invasive intra-arterial needle. For our analysis we chose to limit our blood pressure measurements to the non-invasive captures, as that seemed to be the least restrictive measurement (ie, almost all patients will have non-invasive blood pressure readings, while only those under extreme circumstances may have invasive BP readings).

Next, the *InSight* system utilizes systolic blood pressure and pulse pressure as two distinct patient features. Since pulse pressure is simply the difference between systolic and diastolic blood pressure, we elected to capture the complete blood pressure reading as the underlying information is the same. For example, with a typical BP reading of 120/80, *InSight* will capture 120 and 40 as two distinct features, where we will capture 120 and 80.

Data is loaded and processed directly from the selected MIMIC-III CSV files using Apache Spark 2.3[5] and the Scala programming language. Spark SQL and DataFrames were heavily utilized to efficiently instantiate Scala case classes from MIMIC-III CSV records and filter out unnecessary information. Additionally, various transformations and aggregations were facilitated by Spark SQL, making the process of data set manipulation significantly more manageable.

While most MIMIC-III tables are relatively compact and easily manipulated, the CHARTEVENTS table in particular is of considerable size and requires a significant amount of time to load and transform. We mitigate this issue with two different techniques. First, the initial load of CHARTEVENTS is filtered such that the following criteria are satisfied:

1) Load events for ICU patients only
2) Load events for applicable measurements
3) Load events for patients that are between 15 and 89 years of age (inclusive)[1]

[1] All patients older than 89 years are obfuscated to be 300 years old, and as such may present noise in our training and testing data sets.

Second, we reduce time required for subsequent executions by utilizing the Apache Parquet storage mechanism to store the filtered data required for analysis. This allows us to persist Spark DataFrames to disk, preventing the need to reprocess our primary building blocks for subsequent analysis. Doing so reduces overall processing time from roughly 1 hour to under 2 minutes, for subsequent runs. These framework features allow to us let the Spark/Hadoop infrastructure parallelize the task of input file processing one time only, while using persisted artifacts after-the-fact.

Feature Construction

We adopted the "Gold Standard" definition of sepsis as utilized by Dr. Desautels as it pertains to the "suspicion of infection". Essentially, we are identifying the onset of sepsis by relating an order for antibiotics with an order for a blood culture according to the following sequence of events:

1) If antibiotics are administered first, a blood culture must be drawn within the following 72 hours
2) If a blood culture is drawn first, antibiotics must be prescribed within the following 24 hours

We intentionally ignore any sepsis-related diagnoses, as these are generally retrospective attributes that are not known during the prediction window. With this information, a patient "index date" is constructed such that for all septic patients, the index is the suspected onset of infection, and for non-septic patients the index is the ICU discharge time (OUTTIME).

With the case/control populations identified, we can then capture the patient vitals feature set on which to base a machine learning model for classification. Though Dr. Desautels and team gather the data into hourly bins and use a carry forward mechanism to impute missing data, we use a modified mechanism. We specify prediction window and observation window durations. Initially we tried using averaged values of the vitals within the observation window, but saw better results using the latest vitals within the observation window.

Additionally, we limit PRESCRIPTIONS to only include antibiotic drug orders and MICROBIOLOGYEVENTS to only include blood culture events. One important note to consider is that we also filtered all MIMIC-III data to include only events that were sourced from the Metavision system. We followed this recommendation of the Desautels team due to the limitations around the quality of the Carevue data set. Specifically, negative blood cultures are underreported in the Carevue system, affecting our ability to accurately identify the onset of sepsis per our criteria.

Following several testing iterations, our feature set consists of only 12 dimensions: those identified in Table 1 above, patient age, gender, and qSOFA[6] score. This reduces the size of our training and testing datasets to a mere ~1.25MB for each prediction window, which is a file size that can easily be shared, manipulated, and processed.

Experimental Results

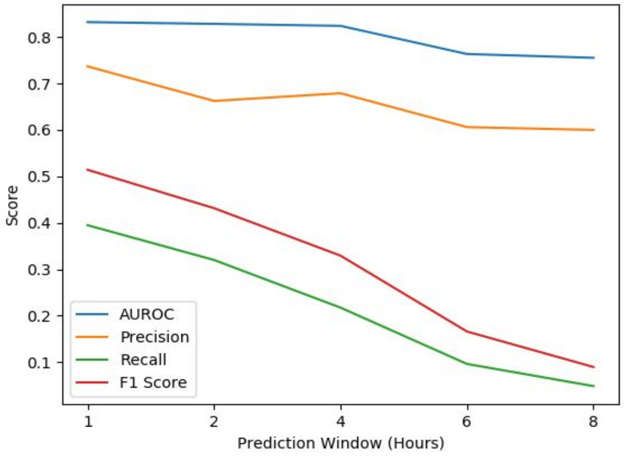
Our experiment is designed primarily to evaluate the efficacy of our feature sets at 1-, 2-, 4-, 6-, and 8-hours prior to the suspicion of infection. For each prediction window, the feature set for each applicable patient is captured from Spark DataFrames by retrieving the latest metrics within the observation window, then persisted to disk in LIBSVM[7] format. Initially, the proof-of-concept for this experiment was conducted using machine learning algorithms provided by Apache Spark. Ultimately, however, we decided to transition our experimentation to the Python framework in order to take advantage of easier metric reporting and charting.

The scikit-learn[8] Python library was utilized for model cross-validation and binary classification for each prediction window. Using our 4-hour feature set, and a 24-hour observation window, we evaluated the performance of several different binary classifiers using K-Fold and Stratified K-Fold cross-validation, using 5 folds for each method. The classification methods evaluated were logistic regression, linear and nonlinear support vector machines, nearest neighbors, basic decision trees, AdaBoost, gradient boosting, and random forests. Ultimately, the random forest classifier, using the default parameters of 100 estimators and GINI split criteria, was chosen for its marginally superior results. K-Fold cross validation metrics are outlined in Table 1 below:

Table 1: K-Fold Cross Validation Metrics

Classification Metrics	**K-Fold**	**Stratified K-Fold**
Accuracy	0.942299071406	0.941786577115
AUC	0.811599408059	0.803129298127
Precision	0.669593105307	0.653566577735
Recall	0.210359509074	0.205763262893
F1-Score	0.319083600593	0.312379384233

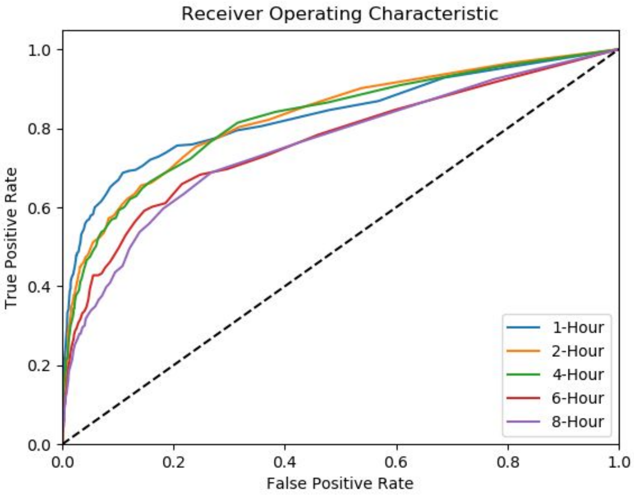
For each of the 1-, 2-, 4-, 6-, and 8-hour feature sets, 75% of the feature file is selected at random[2] for training, and classification metrics are reported on the remaining 25%. Of particular interest is the following chart:



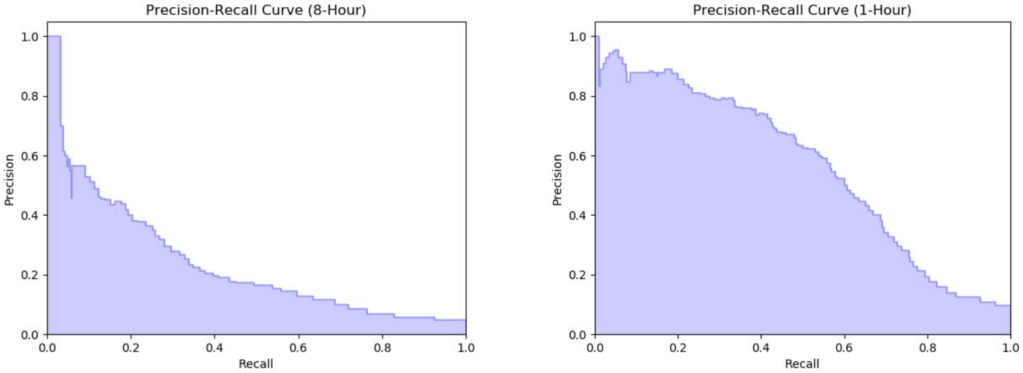
We can see the effect an increasing prediction window has on the precision of our predictions, and this concept aligns with clinical expectations in that a septic infection is very difficult to diagnose proactively.

[2] The random state was seeded with the value 545510477 in order to produce deterministic results for comparison.

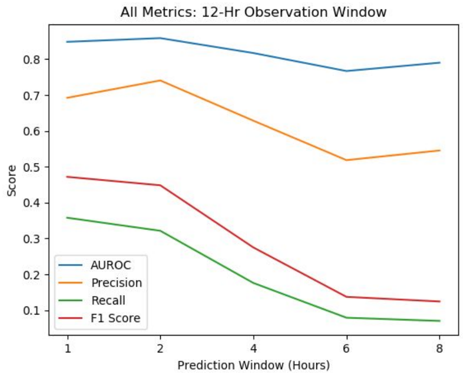
Our random forest classifier achieved AUROC scores ranging from 0.79 to 0.83, from the 8-hour to the 1-hour feature sets respectively. When evaluating the classification results, we decided to utilize soft-labeling for the computation of AUROC scores, which better encapsulates the confidence level of each prediction. Composite receiver operating characteristic curves can be seen in the following chart:

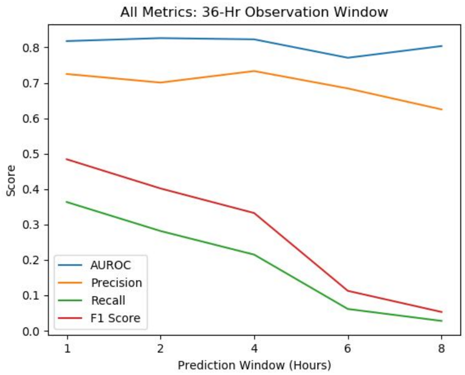


Similarly, we achieved precision/recall scores ranging from 0.86/0.03 to 0.70/0.40, from the 8-hour to the 1-hour feature sets respectively. Probability-based precision/recall curves can be found below for these feature sets:



Finally, we ran each of the 5 feature sets for observation window sizes of 12-, 24- and 36-hours. We observed only minor differences and decided to chose 24 hour observation window as a good median.





Discussion

While we are satisfied with the results obtained, we recognize that there are several factors which contribute to the success (or failure) of any predictive classifier when deployed in non-trivial environments. There are a few key takeaways resulting from this project that we would like to make specific mention of.

First, as can be seen in many subject matter domains, the choice of a machine learning algorithm is often one of little significance. In our own experimentation, we found that many of the classification methods tested produced very similar results despite the different inherent biases of each method. While some methods performed better than others, the differences were not significant enough to suggest that one particular classification method was invalid for approaching this particular problem.

Second, we recognize that feature set construction is of utmost importance in solving classification problems such as this one. Particularly in hospital settings, this feature set can be extremely difficult to obtain in a structured fashion. Based on the personal experience of both authors, we know that monitoring devices often fail to capture adequate data, bedside evaluations often occur outside of scheduled times, and emergency situations often cause interruptions in the collection of basic vitals. All of these factors lead to a potentially noisy data set which can difficult to properly sanitize.

Finally, we are unsure how “concrete” a model such as this one can be. Obviously, more data generally leads for stronger predictive models, but it is possible (even likely) that a geographic bias may exist. For example, the MIMIC-III database represents a patient population in Boston, MA from 2001 to 2012. Is it valid to consider the model constructed from this patient population as applicable to all patient populations? Our intuition is that the model constructed will reflect the overall environment of the population on which it is based. In other words, the model generated from the MIMIC-III database in Boston may not be as successful in prediction the onset of sepsis for a patient population in Seattle, WA or Tampa, FL. As such, the deployment of a predictive model such as this would be subject to continual re-training and re-evaluation.

Conclusion

While a predictive system such as this can never really be considered “complete”, we are confident that our classifier would at least provide some reasonable level of advanced warning for a possible septic infection. Knowing that the human body’s response to such an infection can be recorded through various routine vital signs (increased respiration, decreased blood pressure, etc), it should follow that a model classifier will be able to perform evaluations in real-time to alert medical staff of the suspicion of infection. While this may not be a revolutionary concept, it may provide an avenue for continued medical research on the predictive possibilities of other disease processes that are equally difficult to detect.

References

1) T. Desautels, J. Calvert, J. Hoffman, M. Jay, Y. Kerem, L. Shieh, D. Shimabukuro, U. Chettipally, M. D. Feldman, C. Barton, D. J. Wales, and R. Das. Prediction of sepsis in the intensive care unit with minimal electronic health record data: A machine learning approach. JMIR Med Inform, 4(3):e28, 30 Sept. 2016.
2) Shimabukuro DW, Barton CW, Feldman MD, et al Effect of a machine learning-based severe sepsis prediction algorithm on patient survival and hospital length of stay: a randomised clinical trial BMJ Open Respiratory Research 2017;4:e000234. doi: 10.1136/bmjresp-2017-000234
3) Kim M. (2017, June 19) *Algorithm Predicts Onset of Sepsis.* Retrieved from https://bioengineeringtoday.org/safety/algorithm-predicts-onset-sepsis
4) MIMIC-III, a freely accessible critical care database. Johnson AEW, Pollard TJ, Shen L, Lehman L, Feng M, Ghassemi M, Moody B, Szolovits P, Celi LA, and Mark RG. Scientific Data (2016). DOI: 10.1038/sdata.2016.35. Available from: http://www.nature.com/articles/sdata201635
5) Matei Zaharia, Reynold S. Xin, Patrick Wendell, Tathagata Das, Michael Armbrust, Ankur Dave, Xiangrui Meng, Josh Rosen, Shivaram Venkataraman, Michael J. Franklin, Ali Ghodsi, Joseph Gonzalez, Scott Shenker, and Ion Stoica. 2016. Apache Spark: a unified engine for big data processing. Commun. ACM 59, 11 (October 2016), 56-65. DOI: https://doi.org/10.1145/2934664
6) Seymour, Christopher. “Quick Sepsis Related Organ Failure Assessment.” QSOFA :: Quick Sepsis Related Organ Failure Assessment, www.qsofa.org/.
7) Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm
8) Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011. http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html