Predicting In-Hospital Patient Mortality Using SciKit-Learn and H2O.ai

Manny Argueta

862258112

University of California Riverside

Data Science

12/4/2021

# Abstract

Electronic medical records (EMRs) make it easier to track nuances in diseases that have plagued the aging population for decades. With healthcare becoming more impacted by an aging population and the effects of COVID-19, it is important to find ways to save patients with more calculated measures that may aid doctors with their diagnoses and treatments. EMR data keeps an incredible amount of detail available to be studied to help stay ahead of acute events that increase mortality rates. In 2012, PhysioNet created a challenge to help predict in-hospital mortality in ICU departments based on time-stamped measurements, including different cardiovascular readings. In this study, the goal was to use modern frameworks to tackle the same challenge and build upon previous findings. To date, there is no published approach to this challenge that combines the machine learning capabilities of Python, Scikit-Learn and H2O.ai. With several healthcare companies using H2O.ai to extract knowledge from their data, it seemed fitting that the package be used to approach an important challenge that many hospitals have faced throughout the pandemic. Although this data comes prior to the pandemic, it may help predict acute problems in patients in order to relieve the pressure on medical staff during trying times.

This study combines two common machine learning algorithms in an ensemble with more predictive power than its individual parts. It tackles a very prominent class imbalance problem in the dataset, as well as a large portion of missing values for many features. Finally, it deals with time series data in a manner that allows data to be processed using common machine learning algorithms without excessive complication. Our results were based upon metrics determined by the PhysioNet challenge for straightforward comparison to other models. Sensitivity and positive productivity resulted in \_ and \_ , respectively using the stated ensemble.

# Introduction

With emergency departments entirely occupied and with hospital staff overloaded with COVID-19 cases, the importance of a highly functional intensive care unit (ICU) department became apparent. As hospitals continue to be tested by the pandemic and data is collected during these difficult times, it is possible to look at pre-pandemic factors that were common among patient visits and had predictive power to their outcomes. These early predictions would allow for faster intervention and potentially save many patient lives. Thanks to the plethora of hospital data, and to the highly detail-oriented and organized methods for healthcare data collection, past visits can be quantified and used to identify important patient metrics. In addition to descriptive statistics studies, the data can be used to make predictive models about the health outcomes of patients in intensive care units. If healthcare professionals can act upon patient data by predicting outcomes, they can save lives be better prepared for difficult times.

Information can be extracted from surveys and databases that have been around for decades. For example, the NHAMCS dataset is a survey-based dataset that is useful for extracting knowledge from emergency department visits. Studies have utilized this dataset to follow the use of blood cultures, for example, to expose the potential overuse of certain emergency department protocols (McCaig et al., 2006). Although this dataset can be very impactful, it does not have data that directly represents a patient’s state. Capturing vitals and other physiological measurements in real-time has been a huge undertaking that has arguably been best accomplished by the Medical Information Mart for Intensive Care (MIMIC) datasets. They have de-identified EMRs from thousands of patients that were admitted to the Beth Israel Deaconess Medical Center in the early 2000s (Johnson et al., 2019). This dataset has already led to impactful information about patient stays in the ICU. A study using neural networks was shown to be able to predict length of stay in the intensive care unit using MIMIC-III (Gentimis et al., 2017). This is extremely useful information that has the potential to help hospitals predict patient volume at any given time in order to properly staff their ICU and house their patients. The MIMIC-III is a publicly available dataset that contains a plethora of features from intensive care units for study (Johnson et al., 2016). Its predecessor, the MIMIC-II dataset, was chosen for this study in order to align results as best as possible to previously published studies. The goal of this study is to prove that it is possible to accomplish meaningful knowledge discovery using frameworks like H2O.ai and Scikit-Learn that is comparable to that of leading data science researchers.

The approach for this project was to gather the EMR data and perform extensive data preprocessing to have a manageable dataset that does not compromise the features included in the time-series data. Then, variable extraction took place to have working scalar variables that would be handled well by the chosen machine learning algorithms. Three algorithms were compared and combined in ensemble methods and the results were reported. Those results were compared to the findings of the winners of the competition and a conclusion was made about the capabilities of the chosen machine learning algorithms in comparison to the published works.

# Literature Survey

In the early 2000s, healthcare data was used to study phenomenon across the country, such as the previously mentioned increased use of blood culture tests in emergency department visits (McCaig, 2006). These descriptive statistics approaches were useful for hospitals to make adaptations to their approaches using new information from healthcare data. Since then, studies have trended towards the use of data science techniques to get new insights. EMR data has been used to help hospitals identify their exposure to disasters, such as those from terrorist attacks, in order to help them better prepare for other disasters, such as a pandemic (Niska, 2007). Machine learning is favored to predict certain healthcare options as well. An emergency medical dataset also has reported success in machine learning prediction modelling studies, including having been shown to predict clinical outcomes in children during emergency department triage data (Goto et al., 2019). As methodology has improved, AI has also made its way to healthcare data, predicting seizure occurrences in patients (Rong et al., 2019). This study works towards the same goals as these studies and bases itself upon the work of many researchers in the past decade, however it uses tools that make it possible to conduct this high-level research within a smaller sized team outside of academia. The uniqueness of the study is that no one has reported the use of H2O.ai in combination with Scikit-Learn in hopes of achieving acceptable results.

# Methodology

**Approach:**

The approach for this project is aligned as much as possible with that of the PhysioNet specifications for their original challenge, however some adjustments were made for better compatibility with the H2O.ai framework. The majority of MIMIC-II data was made freely available for public use, and data collection involved gathering only the two-thirds of the data open to the public. The remaining third required credentialed use and is therefore not included at all in this project. Figure 1 (below) displays the handling of data from start to finish. The general processes were in line with the accepted data processing pipeline: data collection, data preprocessing, analytical processing and output (Aggarwal, 2015). The logic assumed for the project was that to be comparable to previous works, significant effort would have to be placed in data preprocessing and analytics, primarily to have working datasets and models, because data collection is identical. Python and many of its libraries were used for every part of this project.

Diagram

Description automatically generated

**Figure 1: Data Mining Pipeline** (Aggarwal, 2015).

**Data Collection:**

Collection of data was performed at Beth Israel Deaconess Medical Center, as previously stated. Patient records are liable to HIPAA constraints and therefore must be handled with care in order to protect patient privacy. The data included over 40 unique features, including age, weight and height. The majority of the features included in the study were time sensitive features. These included heart rates, glucose levels, etc., and each feature was coded in the dataset in identifiable codes. The MIMIC dataset deidentifies the patients that were included in the study, and each patient is instead given a “Record ID” to be identified by. This project was performed without further knowledge about any particular patient. The challenges that arose from this is that with such a large number of variables, it was nearly impossible to identify complete outliers when looking at any patient in the entire set. Even with significant missing data, only 3 of the 4000 patients used in the data set were removed during training. They were removed because they had almost all features missing. Patients with a large majority of missing data were included in the training because in a healthcare setting, there is no standard for the types of tests that are needed, and ICU patients will have many gaps in certain variables. A leading team in the PhysioNet challenge also developed a classification algorithm that was able to handle the large number of missing values, with great results (Johnson et al., 2012). For example, as shown in figure 3, there are cases where a patient has significant datapoints for one variable and very sparse data for another variable, depending entirely on their unique condition. The researcher would not know the difference without looking individually at their diagnosis. A functional machine learning algorithm, in practice, should be able to predict outcomes with missing data present in order to be useful to medical professionals.

Chart

Description automatically generated

**Figure 3: A single patient from MIMIC-II** (Johnson et al., 2012)

Because most data points were time-series data, they would lead to significantly more missing data if aligned by time. Many machine learning algorithms function at a significantly lower capacity with missing data, and many need missing data to be manually inputted in order to function altogether. It was, therefore, decided that we would impute missing data with mean values for each missing feature in the dataset, after feature extraction.

Feature extraction is desired when it allows machine learning algorithms to function properly and when it improves the results of the machine learning algorithm. There has been evidence to support the fact that different kinds of feature extraction methods can improve results in classification problems (Kuo & Landgrebe, 2004). In this case, time-series data was favorably handled via feature extraction for patients in the dataset. In fact, part of the comparison between this project and earlier work for the cardiology challenge revolves around feature extraction. Previous efforts had taken time series data and extracted median, minimum, maximum, first, last and count values for each parameter measured for each patient. This project extracted not only median, minimum, maximum, first, last and count values for each parameter, but also included mean, first quartile, and third quartile values. It was an intended place for comparison between methods whether extracting more feature from the time-series data would yield favorable results overall.

Python, numpy, pandas and matplotlib were used for feature cleaning and integration, feature extraction and handling during data collection and analytical processing steps. Scikit-Learn has several features that were useful for late-stage data preparation including creation of train and test sets from the cleaned and prepared data. It was also used for early model development decisions including which models were more favorable, although that data was not saved and will not be presented in this paper. H2O.ai has several features for data preprocessing that mirror those of Scikit-Learn. The final algorithms in this project were performed using H2O.ai models, and so much of the cleaned and prepared data had to be uploaded into what are called “H2OFrames,” H2O.ai’s functional dataframes. Although H2O.ai has several options for dimensionality reduction for large datasets, these models were not used in hopes of preserving as much of the prepared data as possible. Because the number of dimensions in the data was relatively high, it is a point of future study to further pursue some dimensionality reduction prior to using analytical methods.

**Analytic Methods:**

After data preprocessing, the work of incorporating the data into a set of machine learning algorithms was a main focus of this study. Favorable results had been reported for using MIMIC data to predict mortality rates in the recent past using ensemble methods that included decision trees, perceptron, K-nearest neigbors and logistic regression (El-Rashidy et al., 2020). In line with this approach, the idea for prediction in this project was to create an ensemble using some of H2O.ai’s high performing methods.

H2O.ai (H2O) features an extensive library of machine learning algorithms to choose from. It works well with the Python programming language and integrates well with many of Scikit-Learn’s features. For this reason, the three frameworks are perfectly suited to tackle a dataset together. When used in combination, Python, Scikit-Learn and H2O expand each other’s capabilities substantially. It was an intension of this study to include some deep learning in an ensemble method with other classic prediction models. There has already been evidence that H2O can be used for deep learning classification problems as difficult as cancer classification from genetic data (Sharma & Rani, 2017. It’s Deep Learning Estimator has also been used for predicting diabetes in patients (Dulhare et al., 2020). Therefore, the chosen algorithm for this study is H2O’s Deep Learning Estimator.

Several more of H2O’s algorithms were also considered for use in an ensemble, eventually landing on the two most favorable choices based on grid searches and trial classifications. H2O’s Gradient Boost Estimator is a powerful interpretation of the boosting method that has been around for decades and can be used for regression and classification tasks (Friedman, 2001). The algorithm has many applications and depends largely on a Huber-M loss function and specific learning rate. H2O’s Random Forest Estimator is the last algorithm chosen for this method. Random forests use many decision trees on varying combinations of features in the dataset to make a classification based on averages (Géron, 2017).

Each separate algorithm that was chosen was used on the training data individually first to ensure compatibility with its capabilities. The strength of machine learning models often increase when used in conjunction, so an ensemble method was also chosen to combine the three algorithms and see if their classification improved. H2O’s Stacked Ensemble Estimator was able to incorporate all three of the separate models for comparison of results.

The results of PhysioNet’s original Computing in Cardiology challenge were measured based on two events. This project focuses only on the first event: predicting in-hospital mortality in ICU patients. The metrics are sensitivity and positive predictivity, which are defined in figure 2 below. True positive (TP), false positive (FP), false negative (FN), and true negative (TN) rates are used for the calculation of each metric. Their meanings are clarified in detaAlthough AUC scores were also included in the results for this project, the primary focus is obtaining high sensitivity and high positive predictivity. Prior to the final results, class imbalance was measured to be aware of the differences between the availability of data for each of the possible prediction outcomes. This approach is useful when developing the machine learning algorithms, as many H2O estimators have functionalities to combat class imbalance. A complete analysis of other metrics is not included; however, mean-squared-error and root-mean-squared-error were also provided from H2O’s models.

Graphical user interface, text, application

Description automatically generated

**Figure 2: Sensitivity and Positive Predictivity** (Johnson et al., 2016)

**Plans for interrupting results:**

The possible outcomes from this project include A) performing better than the proposed algorithms from the PhysioNet competition, B) performing comparably with the proposed algorithms from the competition, and C) performing well below the results of those algorithms. Because the dataset is smaller than the one provided to the competitors, it is possible that an underperforming model may be due to more limited data. This is especially possible considering the high dimensionality of the feature space of the dataset. The original question for this project was, is it possible to created high performing algorithms using open-source machine learning libraries? As long as the results are comparable, they can be improved in the future to possibly outperform other algorithms. This is significant because it means that companies can hire from a bigger pool of data scientists and achieve the same results. This is particularly important in healthcare because the more capable data scientists are, and the more tools are available to them, the sooner they can tackle difficult topics such as mortality in ICUs, cancer detection and prediction of chronic conditions. There is significant benefit in using these machine learning algorithms to protect patient lives.

# Results

Evaluation of the distribution of survival and mortality outcomes show that patients generally survive their ICU visits. While good for the patient, this imposes class imbalance to the study that can affect results. In total, approximately 14% of patients passed away in the ICU and 86% of patient survived. Their distribution is simply shown below in figure 3.

A picture containing shape

Description automatically generated

**Figure 4: Class imbalance between positive and negative outcomes**

Class imbalance raises certain issues in the dataset. It doesn’t allow the dataset to get equal representation for each class of possible outcomes. This is very clear in a binary classification example such as this one. The machine learning algorithms are only seeing a small percentage of in-hospital deaths, which is the variable that we want to be the most accurate in predicting. To compensate for the imbalance, luckily, this dataset has a high number of instances. This means that because there are 4000 patients in the dataset, there may be enough to instances of the negative outcome to get accurate predictions.

The results from variable extraction included new values for the time parameter. Because the MIMIC dataset does not track time-series data in a date-time format, but simply in a time-stamp format, it is not possible to conclude anything based on information from the time of year, day of the week, or hours in the day from this dataset. In other words, we can only see the number of minutes that have elapsed from the patient intake into the ICU. The appropriate time frame for each patient is only from zero hours to forty-eight hours, as this was a requirement for the patients included in the dataset. These numbers were converted to hours elapsed as integers. Sample values for a single patient, as well as sample parameter values and their measurements can be seen in figure 5. Another requirement for the dataset included that patients be at least 15 years of age, and the range from data preprocessing reflected this.

Table

Description automatically generated

**Figure 5: Sample Patient Values with Time converted to hours elapsed**

Feature extraction was done entirely using Python, pandas and numpy. A function was created to successfully populate new features for each parameter, divided into each of the metrics discussed earlier: minimum value, maximum value, first value (based on earliest appearance for the parameter for each patient, last value, median value, mean value, first quartile value, third quartile value and count. All data was organized back into a data frame so that each patient “Record ID” had only one instance and captured all approximately 330 total features. If a patient had no readings for a particular parameter, the data frame would fill with missing values that were handled afterwards. The function described is included in the supplemental figures attached at the end of this paper.

Missing values resulted in issues with the machine learning algorithms. Of the algorithms included in the ensemble, only the H2O Deep Learning Estimator had a built-in feature to handle missing values. To remedy this issue, Scikit-Learn’s Simple Imputer was used to fill the data frame with mean values for each of the features. An example of the resulting data frame for the white blood cell count (WBC) is included below in figure 6. This same approach was used to fill values in the static variables that were either missing or clearly erroneous.

Table

Description automatically generated

**Figure 6: White Blood Cell count (WBC) metrics after feature extraction**

A confusion matrix is commonly used to tabulate and calculate the performance of a binary classification model. The confusion matrices for the H2O Random Forest Estimator, H2O Gradient Boost Estimator, H2O Deep Learning Estimator, and the H2O Stacked Ensemble Estimator of the three are shown in figure 7 below. The top left of the confusion matrices represents the true negatives for each of the machine learning models, where the model predicted that the patient would pass away in the hospital and the patient was reported to have passed away in the dataset. The bottom left represents the false negatives, or the number that were predicted to have died but had survived. The top right of the matrices represents the false positives, or the patients that were predicted to have survived but had passed away in the hospital. And the bottom right of the matrices represents the true positives, or the patients that were predicted to have survived and had survived in the hospital. From these matrices, we can calculate the sensitivity and the positive predictivity of the models.

Table

Description automatically generated

**Figure 7: Confusion Matrices for A) H2O Gradient Boost Estimator, B) H2O Random Forest Estimator, C) H2O Deep Learning Estimator, D) H2O Stacked Ensemble Estimator**

|  |  |  |
| --- | --- | --- |
| **ML Model** | **Sensitivity** | **Positive Predictivity** |
| Random Forest | 0.990 | 0.880 |
| Gradient Boost | 0.993 | 0.889 |
| Deep Learning | 0.995 | 0.882 |
| Stacked Ensemble | 0.972 | 0.903 |

**Table 1: Sensitivity and Positive Predictivity for each model**

Table 1 above shows the sensitivity and positive predictivity for each of the machine learning models. According to the scoring from the Computing in Cardiology challenge, the score for Event 1 in the competition is the lower of the two between sensitivity and positive predictivity. The highest scoring in this case would then be the H2O Stacked Ensemble of the three models, although the other models alone could be considered equally likely to score similarly given other test data.

There are other metrics for binary classification problems that are also great indicators for model performance. The precision-recall curve (PR) is the curve that fittingly results when measuring precision of the model against its recall. The receiver operating characteristic (ROC) curve is a curve that plots the true positive rate against the false positive rate. Both of these curves are used very commonly in binary classifiers. AUC refers to the area under the curve for each of these plots. The AUCPR metric is useful when the positive classification is the less frequent of the two, while the AUROC is preferred when it is the most frequent. It is helpful to calculate both to get an accurate picture of how each algorithm is performing, particularly because this problem involves a heavy class imbalance.

|  |  |  |
| --- | --- | --- |
| **ML Model** | **AUROC** | **AUCPR** |
| Random Forest | 0.786 | 0.931 |
| Gradient Boost | 0.856 | 0.975 |
| Deep Learning | 0.781 | 0.948 |
| Stacked Ensemble | 0.844 | 0.968 |

**Table 2: Sensitivity and Positive Predictivity for each model**

Table 2 above shows each of the scores for each of the models. More likely, the lower represents the model best. Because the AUROC curve is the preferred score for this type of class imbalance, it is practical to use that as the measure for the models. With a score of 0.856, the H2O Gradient Boost Estimator outperformed the other models.

# Discussion and Results

**Discussion:**

The results of this project display the potential that open-source frameworks like Scikit-Learn and H2O have in making machine learning obtainable for many companies without needing intensive academics to back their learning models. However, there are several points of discussion that come to mind when viewing the results.

First, the resulting data frames from feature extraction still have a significant portion of missing values after cleaning and preprocessing. The teams that contributed to the competition in 2012 were able to make strong algorithms that handled missing data well, although their exact methods were undisclosed, as far as was possible to conclude during the literature review from this project. It is possible that their algorithms handled missing data in a way that left the rest of the data unaffected, however, it is equally likely that their algorithms imputed values for the missing data the same as the way that was done for this project. The models for this application need to be able to function without jeopardizing the data set because when data is collected from patients, it will likely always include a large number of missing values. The only way for these machine learning models to be useful is by being robust to the possibly large amounts of missing data points.

Second, the sensitivity and predictivity of the models from this project performed well. They may have performed a bit too well. In comparison to the studies performed in 2012, our score 1 would have been enough to win the competition. But why is this not likely what would have occurred? One possibility is the significant amount of data that is withheld from the public from the MIMIC dataset. The MIMIC dataset was only provided at one-third of the size of the original for this competition, and the teams that competed had their model tested against an equally large set of new instances. That means that the model was tested against 4000 new patients, compared to our 1000 that was withheld as a test set for this project. Another issue that this implies for this project is that this model is overfitting to our data.

It is likely that our model is overfitting this dataset because of the high sensitivity and predictivity scores, but also because of the high AUCPR scores. The lower AUROC scores for this model signify some robustness in this model that is not seen from the other metrics. As an example of why robustness is important, if we look at the H2O Deep Learning Estimator’s sensitivity score, the model can almost perfectly predict true positives. However, the AUROC score for the model is the lowest of the four models. This means that the model does not generalize well and would likely underperform with a new dataset. We need a model that can generalize well so that it can predict the outcome of a single patient with high confidence. Overfitting is a common issue with machine learning models and the results of this project imply that, although it performed well, some future measures need to be taken to improve robustness.

**Conclusion:**

The proposed method involved a large dataset that has been considered one of the great amongst healthcare data. The potential for EMRs to improve healthcare is great, and datasets like the MIMIC datasets are therefore monumental contributions to healthcare. The present study was all inclusive in the data mining process, handling every step of the way after database generation and collection. It provides an big picture view of the data science process as a data scientist would see it, and it has been able to show positive performance from the work of an individual. The hope hereafter is that using these tools with small or even large teams in healthcare companies will yield big results for patient care.

Future directions include working with other combinations of ensembles in the H2O framework to achieve even better results. With more testing, the issues of overfitting will be minimized, and the model would eventually be able to be deployed in healthcare settings. In addition, the data preprocessing that was part of this project would be useful for anyone that wants to use the MIMIC dataset and save time and effort preparing the data for use. The MIMIC dataset is large and encompasses many other approaches that can improve healthcare in the world. The work from this project can be applied in parallel to many other projects and help achieve similar outcomes. Almost ten years after the beginning of this competition, the approach to data science has been able to be extended far enough that even graduate students can take use their knowledge individually to achieve good results. This project shows that H2O, Scikit-Learn and Python are powerful tools that can have an impact in healthcare.

References

Johnson, A. E. W., Pollard, T. J., Shen, L., Lehman, L. H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Anthony Celi, L., & Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. *Scientific Data*, *1*. <https://doi.org/10.1038/sdata.2016.35>

Johnson, A., Pollard, T., & Mark, R. (2019). MIMIC-III Clinical Database Demo (version

1.4). *PhysioNet*. <https://doi.org/10.13026/C2HM2Q>.

McCaig, L. F., McDonald, L. C., Cohen, A. L., & Kuehnert, M. J. (2007). Increasing Blood Culture Use at US Hospital Emergency Department Visits, 2001 to 2004. *Annals of Emergency Medicine*, *1*, 42-48.e2. https://doi.org/10.1016/j.annemergmed.2006.12.002

Goto, T., Camargo, C. A., Faridi, M. K., Freishtat, R. J., & Hasegawa, K. (2019). Machine Learning–Based Prediction of Clinical Outcomes for Children During Emergency Department Triage. *JAMA Network Open*, *1*, e186937. <https://doi.org/10.1001/jamanetworkopen.2018.6937>

Niska, R. W. (2008). Hospital Collaboration with Public Safety Organizations on Bioterrorism Response. *Prehospital Emergency Care*, *1*, 12–17. https://doi.org/10.1080/10903120701709514

Rong, G., Mendez, A., Bou Assi, E., Zhao, B., & Sawan, M. (2020). Artificial Intelligence in Healthcare: Review and Prediction Case Studies. *Engineering*, *3*, 291–301. https://doi.org/10.1016/j.eng.2019.08.015

Aggarwal, C. C. (2015). *Data Mining*. Springer.

Dulhare, U. N., Mubeen, A., & Ahmad, K. (2020). Hands‐On H2O Machine Learning Tool. *Machine Learning and Big Data*, 423–453. https://doi.org/10.1002/9781119654834.ch15

El-Rashidy, N., El-Sappagh, S., Abuhmed, T., Abdelrazek, S., & El-Bakry, H. M. (2020). Intensive Care Unit Mortality Prediction: An Improved Patient-Specific Stacking Ensemble Model. *IEEE Access*, 133541–133564. https://doi.org/10.1109/access.2020.3010556

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, *5*. https://doi.org/10.1214/aos/1013203451

Géron, A. (2017). *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. O’Reilly Media.

Johnson, A., Dunkley, N., Mayaud, L., Kramer, A., & Clifford, G. (2012). Patient Specific Predictions in the Intensive Care Unit Using a Bayesian Ensemble. *Computing in Cardiology*.

Kuo, B.-C., & Landgrebe, D. A. (2004). Nonparametric weighted feature extraction for classification. *IEEE Transactions on Geoscience and Remote Sensing*, *5*, 1096–1105. https://doi.org/10.1109/tgrs.2004.825578

Sharma, A., & Rani, R. (2017). An Optimized Framework for Cancer Classification Using Deep Learning and Genetic Algorithm. *Journal of Medical Imaging and Health Informatics*, *8*, 1851–1856. <https://doi.org/10.1166/jmihi.2017.2266>