Journal of Machine Learning Research 1 (2021) Submitted 11/12; Published 11/12

Mortality Prediction in Heart Failure Patients Admitted to the Intensive Care Unit (ICU)

Rebecca Hadi RHADI1@JH.EDU
Seattle, WA, USA

Editor: Not applicable

Abstract

This study explores the binary classification task for mortality prediction in heart failure patients in the ICU. Five types of models were studied in the 5x5 experimental design. All models were able to show an ROC-AUC of > 0.75 on the validation and test data sets. The final model chosen was XGBoost with SMOTE, with ROC-AUC of > 0.75. The most important feature was blood sodium, which is consistent with external research into survival analysis of heart failure patients.

Keywords: Supervised learning, Logistic Regression, Multi-Layer Perceptron, Support Vector Machine, XGBoost, Random Forest, Heart Failure, Mortality

1. Introduction

This study will contain the comparison of five different binary classifiers on the prediction of mortality among heart failure patients. This application is important because treatment could be adjusted patient on risk factors if a certain lab value or co-morbidity indicated a higher likelihood of mortality to optimize physician resources. The goal of this project is model accuracy, which will inform the final selection of model because we will accept a loss in interpretability for a gain in prediction.

2. Problem Statement

How accurately can mortality in heart failure patients be predicted? Accuracy will be measured using the ROC-AUC (receiver operating characteristic area under the curve) and we require a score above .75 on the validation and test data sets in order to consider the model to be sufficient.

2.1 Hypotheses

- The null hypothesis for the set of experiments is that no ROC-AUCs for any model run will be > 0.75.

- The alternative hypothesis is that at least one of the ROC-AUCs will be > 0.75.

3. Data Background and Pre-Processing

The data set used in this project is from the "MIMIC-III Clinical Database" that contains de-identified health-related data associated with over 40,000 patients who stayed in critical care units of the Beth Israel Deaconess Medical Center between 2001 and 2012[1]. This data set initially had 51 columns, one for the 'outcome' variable (the response), and 48 features made up of discrete and continuous.

3.1 Pre-processing steps

3.1.1 Inspection of nulls

using the heatmap() function from the Seaborn library, we can see that there are nulls in different fields in the data set. Some fields looked like they were missing at random, while some seemed to be more concentrated around certain rows (e.g. PH).

There was 1 null for the 'outcome', which was discarded from the data set because this is a supervised learning problem that requires a class label, and also it was only 1 record out of 1,177 records. For the rest of the fields, the nulls were imputed with the mean value. There were no discrete features that had null values, so no imputation was needed. The final number of observations included was 1,176 records.

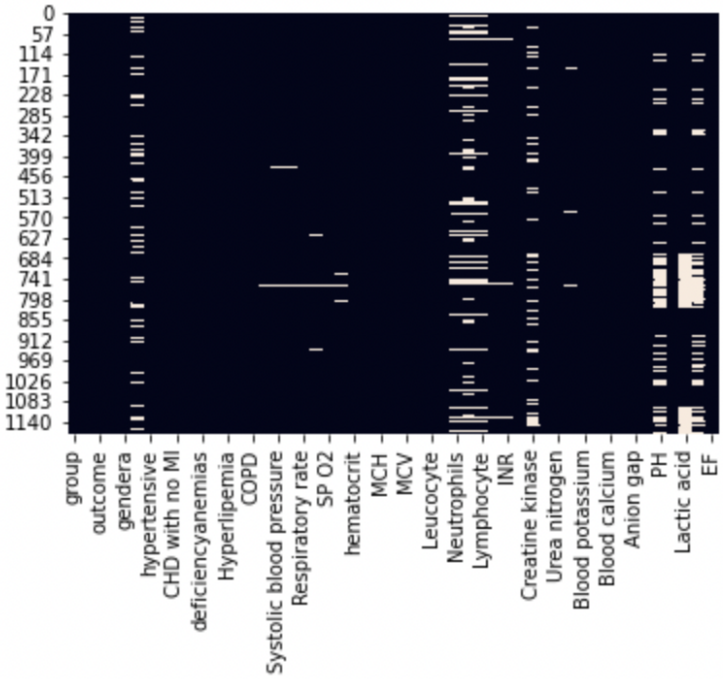
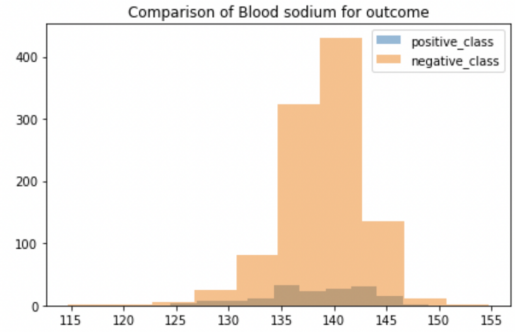


Figure 1: Null Heatmap

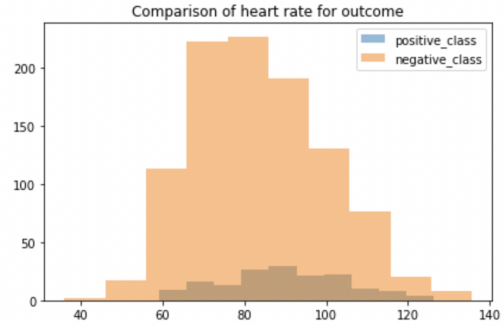
3.2 Feature Extraction

The curse of dimensionality will be addressed by attempting to lower the number of relevant features used for modeling. Toward this aim, t-test and two-proportion z-tests were conducted on the continuous and discrete features, respectively. The two populations were defined as the array of observations where the outcome class was positive (1 = mortality), compared against the array where the outcome class was negative (0=no mortality). Each feature was then tested for statistical significance at the $\alpha = 0.05$ level of significance. The features that had a statistically significant different were included as candidate features in the model, the final set was 27 features compared to the original 48.




```
Blood sodium
 Ttest_indResult(statistic=-2.278967234192661, pvalue=0.02284791147635487)
Difference is statistically significant at alpha = 0.05
```

Figure 2: Distribution of blood sodium levels across positive and negative classes - Statistically significant




```
heart rate
 Ttest_indResult(statistic=4.460635913313653, pvalue=8.964104077003178e-06)
Difference is statistically significant at alpha = 0.05
```

Figure 3: Distribution of heart rate) levels across positive and negative classes - Statistically significant

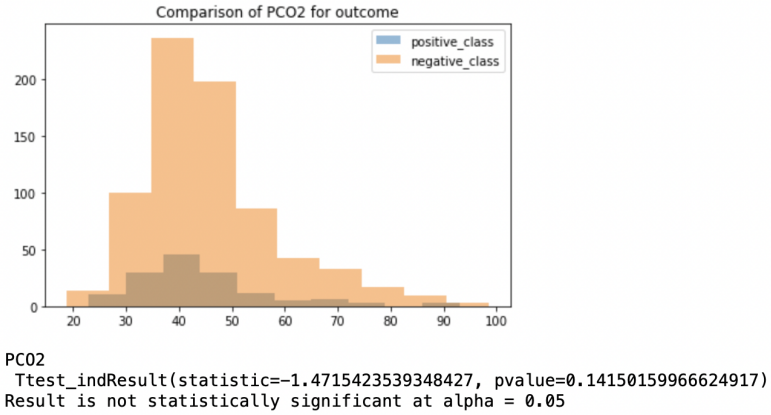


Figure 4: Distribution of partial pressure carbon dioxide (PCO2) levels across positive and negative classes - Not Statistically significant

These plots for all significance tests (both continuous and discrete) are omitted from this paper for the sake of brevity. Full results can be found in the accompanying Jupyter Notebook PDF.

3.3 Principal Component Analysis (PCA) and Variance Inflation Factor

Further dimensionality reduction was conducted using principal component analysis (PCA) in one the experimental runs. The optimal number of components was n=20, so the dimensionality of the data set was further reduced. The author also tested the variance inflation factor of the features and removed two that had high col-linearity. Outside of PCA, the final data set was comprised of 25 features and 1 response variable.

3.4 Standardization of Continuous Features

To put the continuous variables on the same scale, each continuous feature was standardized to create a z-score where $z = \frac{x-\mu}{\sigma}$. The mean and standard deviation are based on the training data set and applied to the test and validation data sets.

3.5 Class Imbalance

The mortality rate is 13%, so there is an imbalance between the positive class and negative class. Ideally, the split between positive and negative classes is 50%, which would represent balanced classes. In one experiment, this is addressed using Synthetic Minority Oversampling Technique (SMOTE) to balance the classes so they are equivalent size. The oversampling gives more positive class records to train on and learn. The validation and test sets are not touched in the SMOTE process.

4. Experimental Approach

The full data set after removal of the null outcome class was 1,176. This data set was split into 20% test and 80% training. The training set then split again into 20% test and 80%, to generate the validation and training sets. This results in a total of 20% test, 16% validation, 64% training.

Each experiment involves uses the training data set to train the model, then running that model on the validation and test sets to understand the ROC-AUC. We then test this value against our hypothesis.

5. Tuning

Each of the 5 models was tuned on the training set using randomizedsearchCV() from sklearn. A dictionary of parameters for each model was output based on the optimization toward ROC-AUC. The best parameters for each model were selected and a trial was run to understand the performance of the tuned models compared to the default models.

6. Results and Discussion

Based on the results of the experimental trials, the study can reject the null hypothesis that no models will have an ROC-AUC value of at least 0.75 in favor of the alternative hypothesis that at least one of the models will have an ROC-AUC value of 0.75.

	Default				SMOTE			
Model Type	Training	Validation	Test	Avg	Training	Validation	Test	Avg
Logistic Regression	86.2%	70.8%	77.9%	74.4%	84.0%	74.2%	78.8%	76.5%
Random Forest	100.0%	74.7%	78.6%	76.7%	100.0%	77.3%	78.0%	77.6%
XgBoost	100.0%	74.3%	73.7%	74.0%	100.0%	79.4%	77.4%	78.4%
Support Vector Machine (SVM)	99.2%	76.2%	74.5%	75.4%	99.2%	76.6%	75.7%	76.1%
Multi-layer Perceptron	100.0%	73.5%	75.7%	74.6%	100.0%	73.7%	78.0%	75.9%

	Tuned				SMOTE + Tuned			
Model Type	Training	Validation	Test	Avg	Training	Validation	Test	Avg
Logistic Regression	86.2%	70.8%	77.9%	74.4%	91.3%	74.2%	78.8%	76.5%
Random Forest	90.6%	70.0%	76.5%	73.3%	90.4%	76.1%	75.0%	75.6%
XgBoost	100.0%	75.6%	74.8%	75.2%	100.0%	73.5%	72.6%	73.1%
Support Vector Machine (SVM)	86.0%	70.2%	75.3%	72.7%	85.5%	73.0%	78.5%	75.7%
Multi-layer Perceptron	100.0%	73.8%	69.6%	71.7%	100.0%	75.8%	77.2%	76.5%

	PCA			
Model Type	Training	Validation	Test	Avg
Logistic Regression	86.0%	70.1%	77.2%	73.7%
Random Forest	n/a	n/a	n/a	n/a
XgBoost	n/a	n/a	n/a	n/a
Support Vector Machine (SVM)	n/a	n/a	n/a	n/a
Multi-layer Perceptron	n/a	n/a	n/a	n/a

Figure 5: ROC-AUC across all experiments and average validation/test ROC-AUC

The comparison of the training error to the test and validation errors indicates overfitting. The ROC-AUC in the training data of 1.0 means that the data were memorized - so there is opportunity to further refine the models (e.g. early stopping), to lower the training error with the aim of better generalization.

Based on ROC-AUC average across the validation and test sets, the best performing model was XGBoost with SMOTE.

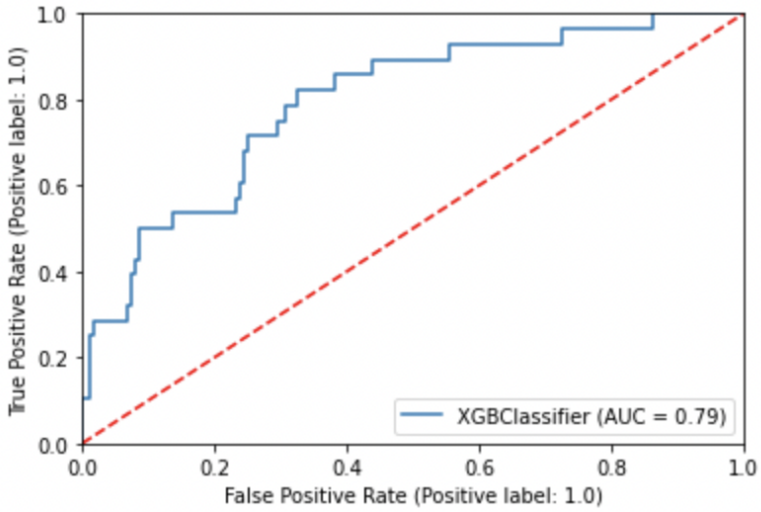


Figure 6: ROC Curve for XGBoost with SMOTE on validation set

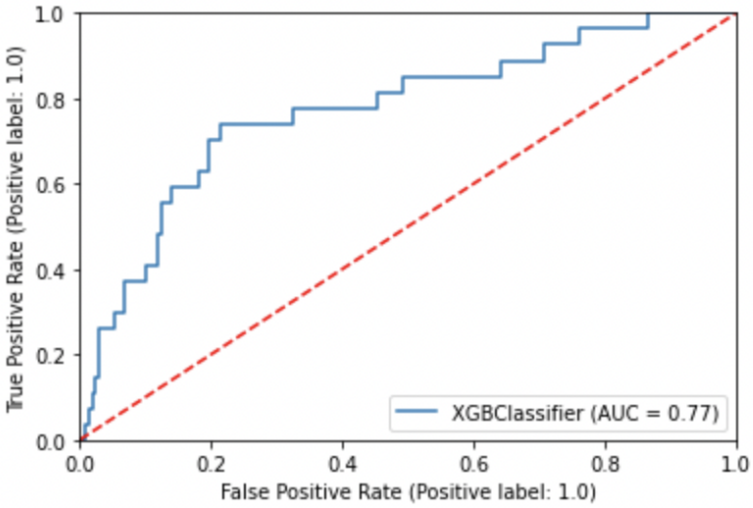


Figure 7: ROC Curve for XGBoost with SMOTE on test set

On the validation and test sets (data which the model has not seen before), the final model had ROC-AUC value of 79% and 77%, respectively.

However to fully make a recommendation we need to go a level deeper and understand how the model performs in precision and recall. For the final model, the precision for the positive class was 46% and the recall was 43%. This means that the model was evenly balanced on precision and recall. It was able to recall 43% of the positive class, and of the positive classes labeled in the data set, 46% were actually positive. Comparing against the classification report for the XGBoost without SMOTE (precision: 42%; recall: 29%), we see that the synthetic samples improve recall of the positive class without a significant decrease in precision.

From a use case perspective, if the application was such that it was more beneficial to have higher recall than precision (i.e. false positives preferred over false negatives) then we may have selected a different model that had a slightly lower ROC-AUC but more suitable precision-recall values for the positive class. Further research can shed light on the trade-offs.

	precision	recall	f1-score	support
0.0	0.90	0.91	0.91	160
1.0	0.46	0.43	0.44	28
accuracy			0.84	188
macro avg	0.68	0.67	0.68	188
weighted avg	0.84	0.84	0.84	188

Figure 8: XGBoost with SMOTE Classification Report

6.1 Inference

From an inference perspective, we can plot the feature importances from the XGBoost Model with SMOTE. The most important feature was blood sodium (standardized) followed by red cell distribution width (RDW) standardized. Based on this model, blood sodium may play a role in the mortality of a heart failure patient. The author checked the result against a study on mortality in heart failure and blood sodium level (specifically hyponatremia if sodium < 135 mmol/L)[2]. The result showed a statistically significant difference in mortality rate for patients with hyponatremia (low blood sodium).

7. Conclusion

In this study we have shown that we are able to predict mortality in heart failure patients with 75% to 79% accuracy. We have drawn inference that blood sodium level was the most important feature in the final model, XGBoost with SMOTE.

7.1 Ethical Considerations

Representation is important in the data. This data set is built from a Boston hospital. The project can be improved by collecting data from different hospitals in different regions to ensure representation across different groups of people.

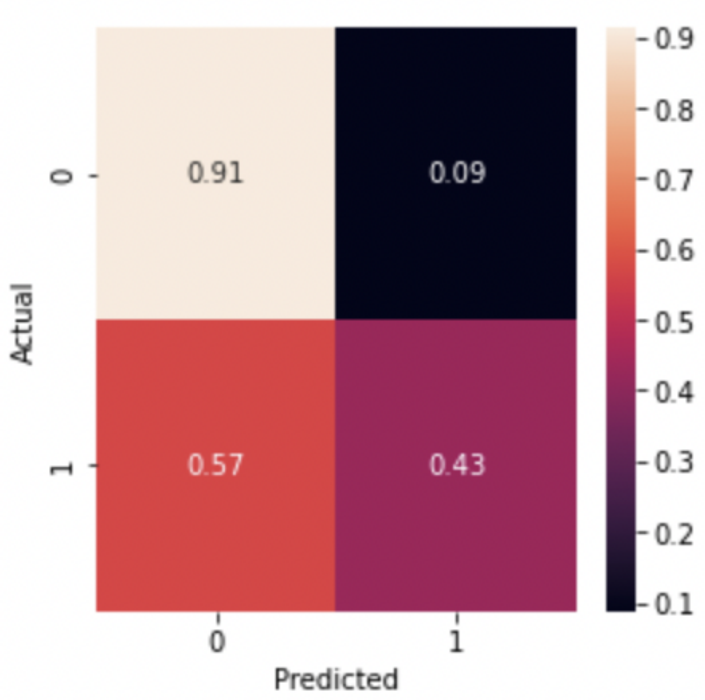


Figure 9: XGBoost with SMOTE Confusion Matrix

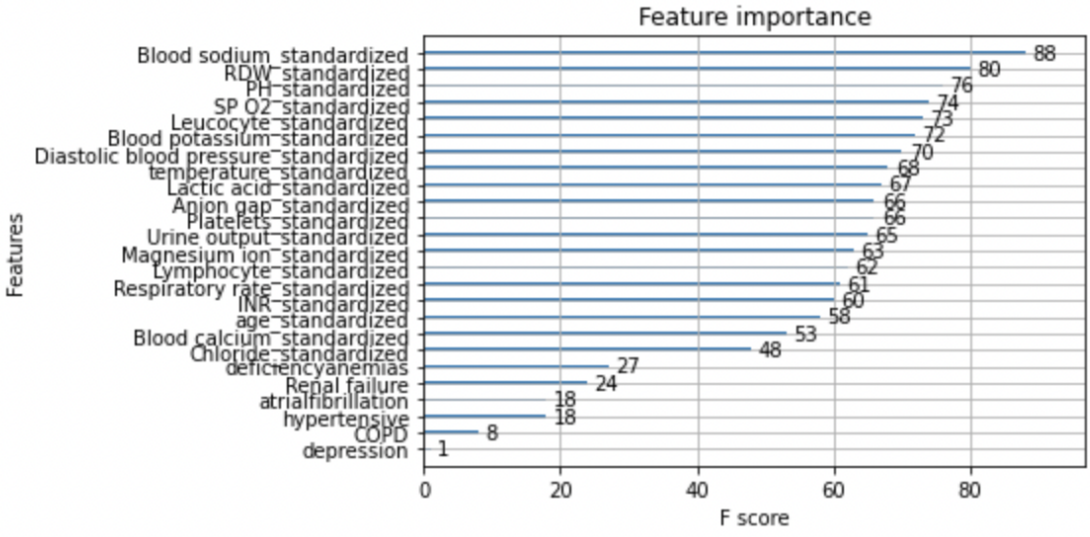


Figure 10: XGBoost with SMOTE Confusion Matrix

Use in Production: The author would caution against using this model in production in its current form. One system could use this model to generate a prediction and prioritize

care based on the likelihood of mortality. This has ethical implications as it would result in preferential treatment that while based on good intentions, may result in clinical inequality. If more data were collected from different geographic areas and trained a model on a larger population, the author would feel more comfortable for use in a production physician alert system. It would also be a benefit to have clinician input to sniff tests results so the appropriate questions and considerations will be raised that accompany clinical training.

8. Potential Next Steps

Further research is needed to determine the strength of the relationship of blood sodium and mortality in heart failure patients. This research could be expanded with the collection of more data from diverse populations and geographical locations to improve the strength of the model, as mentioned in the ethical considerations.

From a modeling perspective, different combinations of features could be tried in the model using methods such as forward selection. An expansion could also be to include interaction terms between some of the features.

9. References

1. Johnson, A., Pollard, T., amp; Mark, R. (2019, April 24). Mimic-III clinical database demo. MIMIC-III Clinical Database Demo v1.4.
Retrieved October 1, 2021 from https://physionet.org/content/mimiciii-demo/1.4/.

2. Abebe, T. B., Gebreyohannes, E. A., Tefera, Y. G., Bhagavathula, A. S., Erku, D. A., Belachew, S. A., Gebresillassie, B. M., amp; Abegaz, T. M. (2018, November 8). The prognosis of heart failure patients: Does sodium level play a significant role? PloS one.
Retrieved November 5, 2021, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6224129/.

3. Pedregosa, F., Profile, V., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É., amp; Metrics, O. M. V. A. (2011, February 1). Scikit-Learn: Machine learning in Python. The Journal of Machine Learning Research.
Retrieved October 13, 2021, from https://dl.acm.org/doi/10.5555/1953048.2078195.

4. Brownlee, J. (2021, March 16). Smote for imbalanced classification with python. Machine Learning Mastery. Retrieved November 1, 2021, from https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification.

5. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. An Introduction to Statistical Learning : with Applications in R. New York :Springer, 2013.