The trauma severity model: An ensemble machine learning approach to risk prediction

Michael T. Gorczyca[a*], Nicole C. Toscano[b], Julius D. Cheng[b]

[a]Department of Biological and Environmental Engineering, Cornell University, Ithaca, NY 14853

[b]Department of Surgery, University of Rochester Medical Center, Rochester, NY 14642

[*]Corresponding author. E-mail address: mtg62@cornell.edu. Address: 321 Riley-Robb Hall, Cornell University, Ithaca, NY 14853. Phone: +1(607)255-2173.

Abstract

Statistical theory indicates that a flexible model can attain a lower generalization error than an inflexible model, provided that the setting is appropriate. This is highly relevant in the context of mortality risk prediction for trauma patients, as researchers have focused exclusively on the use of generalized linear models for risk prediction, and generalized linear models may be too inflexible to capture the potentially complex relationships in trauma data. Due to this, we propose a machine learning model, the Trauma Severity Model (TSM), for risk prediction. In order to validate TSM's performance, this study compares TSM to three established risk prediction models: the Bayesian Logistic Injury Severity Score, the Harborview Assessment for Risk of Mortality, and the Trauma Mortality Prediction Model. Our results indicate that TSM has superior performance, and thereby provides improved risk prediction.

Keywords: Risk Prediction, Trauma Quality Improvement, Machine Learning, Stacked Generalization, Electronic Health Records

Highlights:

- We propose an ensemble machine learning model for trauma risk prediction.
- A hyper-parameter search scheme is proposed for model development.
- We compare our model to established models for trauma risk prediction.
- Our model improves over established models for each performance metric considered.

1. Background

Trauma is a global healthcare epidemic, accounting for 9.2% of all deaths and 10.9% of disability-adjusted life-years [1]. The potential impact of trauma injuries on one's quality of life has inspired several studies on how we can improve the quality of trauma care, and consequently improve trauma patient outcomes. However, several of these studies require that we take a trauma patient's injury severity (risk of mortality) into account, and there is no consensus as to which risk prediction model is most appropriate for use [2-9].

Interestingly, careful consideration of the methodologies used to develop these risk prediction models indicates that regardless of which model is most appropriate, there may be room for substantial improvement in the quality of risk prediction. One reason for this is that several risk prediction models have been developed from small data sets [2-5], which implies that these models may not represent the population appropriately [10]. Another reason is that every model developed from a large data set thus far has been a generalized linear model [6-9], and generalized linear models may be insufficient for capturing the potentially complex relationships that exist within trauma data.

For these reasons, our objective is to develop a risk prediction model from machine learning algorithms with data from the National Trauma Data Bank (NTDB) and to compare its performance to the performance of other established risk prediction models. This is achieved by comparing three established risk prediction models – the Bayesian Logistic Injury Severity Score (BLISS) [7], the Harborview Assessment for Risk of Mortality (HARM) [8], and the Trauma Mortality Prediction Model (TMPM) [9] – to a new machine learning model for risk prediction. This machine learning model is the Trauma Severity Model (TSM).

2. Methods

2.1. Data Summary and Processing

This study was performed using data from the NTDB for patients hospitalized in 2008, 2009, 2010, and 2012. The NTDB is currently the largest aggregation of trauma data in the United States and provides patient demographics, hospital demographics, ICD-9-CM diagnoses codes (ICD-9 codes), general trauma assessments, hospital identifiers, physiology values, and in-hospital mortality [11]. The data set initially consisted of 2,865,867 patient records from 884 hospitals and 7,283 ICD-9 codes.

Risk prediction with this data set was formalized as a binary classification task. The input variables considered were patient demographics (age and gender), ICD-9 codes, and general trauma assessments (comorbidities, Glasgow Coma Scale response scores prescribed by a physician, injury mechanism, injury type, and intent of trauma). All input variables except age are treated as binary indicators specifying whether or not a patient had that particular condition. For Glasgow Coma Scale response scores, the eye response score is represented by 5 binary indicators (4 for response scores and 1 for not provided in the dataset), the verbal score is represented by 6 binary indicator variables (5 for response scores and 1 for not provided in the dataset), and the motor score is represented by 7 binary indicators (6 for response scores and 1 for not provided in the dataset). Response scores are treated as binary indicators to improve the performance of the linear models. The output variable is a binary indicator specifying whether or not the patient died prior to discharge from a hospital.

To ensure that model comparison is fair, we closely followed the data cleaning procedure from TMPM's study, which is in accordance with the data cleaning procedures from BLISS and HARM's studies. For patient selection, this involved excluding patients that had burns or an ICD-

9 code unrelated to trauma (e.g., poisoning, drowning, or suffocation) (193,606), were admitted to a hospital that did not maintain complete documentation of relevant trauma diagnoses (655,440), were missing data (for age, comorbidities, gender, injury mechanism, injury type, intent of trauma, and outcome) (335,980), had pre-hospital mortality (60,234), were transferred to another hospital (848,885), were discharged to hospice care or another acute care hospital (16,429), withdrew care (18,395), or were less than one year old (47,693).

There are 2 differences between the patient selection process in this study and that in TMPM's study. One difference is how we selected hospitals from which we selected patients. In TMPM's study, the data set consisted of patients from hospitals that admitted at least 500 patients during at least 1 year of the study (hospitals with "substantial trauma experience") [9]. We instead used all patients that were admitted to any hospital that kept complete records of all ICD-9 codes that were considered relevant in TMPM's study. The reasoning for this is that some trauma centers that would qualify as having substantial trauma experience omitted relevant ICD-9 codes from their registry, and this could harm each model's ability to provide accurate risk predictions [12]. Another difference is that TMPM's study ensured complete documentation only for age, gender, and outcome when determining which patients to include. We extended this to also ensure complete documentation for comorbidities, injury mechanism, injury type, and intent of trauma. The reasoning for this is that (1) no additional patients were excluded because of these criteria, (2) this information is typically known at the time of admission and is relevant in determining patient outcome, and (3) no risk prediction model has ever given consideration to such a combination of variables.

Once the patient selection process was completed, an ICD-9 code combining procedure was performed, which followed the ICD-9 code combining procedure in TMPM's study exactly

(please see [9] for an overview of this procedure). This ICD-9 code cleaning procedure was followed by an additional pre-processing step that combined any ICD-9 code that appeared fewer than 5 times with the closest corresponding ICD-9 code (based on expert consensus). This consisted of combining a specific injury with a more general injury; an open injury with a closed injury; or a group of highly similar injuries that were poorly represented to one single injury. This additional pre-processing step improved the performance of all models in this study.

The patient selection process kept 1,385,795 patient records out of 2,865,867 patient records and 2,033 ICD-9 codes out of 7,283 ICD-9 codes. The ICD-9 code cleaning procedure from TMPM's study collapsed these 2,033 ICD-9 codes into 1,272 binary indicators representing ICD-9 codes. Combining ICD-9 codes that appeared fewer than five times with what was determined to be the closest corresponding ICD-9 code collapsed these 1,272 binary indicators into 1,234 binary indicators. There are 74 other variables that represent patient demographics, general trauma assessments, and patient outcome, which leaves us with a sparse 1,385,795 by 1,308 matrix (all variables are binary indicators except age, which is numeric). Table 1 provides a brief summary of the demographics for this processed data set.

2.2. Experimental Setup

We considered two experiments for this study. The first experiment developed TSM, BLISS, HARM, and TMPM using information pertaining to ICD-9 codes in the processed data set as input variables (HARM and TMPM utilize dimensionality reduction procedures on ICD-9 codes). This first experiment will be referred to as the "ICD-9 experiment." The second experiment developed TSM, BLISS, HARM, and TMPM using patient demographics (age and gender), information pertaining to ICD-9 codes, and general trauma assessments (comorbidities, Glasgow

Coma Scale response scores, injury mechanism, injury type, and intent of trauma) in the processed data set as input variables. The second experiment will be referred to as the "augmented experiment."

To ensure appropriate model development and assessment for these experiments, the processed data set was randomly divided into a training set for model development (60% of the entire data set), a validation set for optimizing model performance (20% of the entire data set), and a test set for model assessment (20% of the entire data set). Model development was performed using the h2o [13] and sandwich [14] packages in the R statistical software (Version 3.3.1) [15]. Optimizing model performance concerns minimizing log-loss (LL) on the validation set for these experiments [16].

2.3. Model Development

2.3.1. BLISS

BLISS utilizes Bayesian logistic regression for risk prediction. To re-develop BLISS for this study, two different Bayesian logistic regression models for each experiment, where the prior distribution (a Laplace prior or a Gaussian prior) was varied. The model that had the lowest LL on the validation set was selected as BLISS for that experiment.

2.3.2. HARM

HARM is a logistic regression model that takes advantage of the hierarchical structure of ICD-9 codes to reduce dimensionality. Specifically, ICD-9 codes and patient demographics are combined together to create new variables (based on expert consensus) that replace ICD-9 codes and patient demographics. These new variables are selected as the input variables for HARM

using forward selection [16].

To develop HARM for this study, HARM's variable combining procedure was followed as closely as possible with the processed data set – the NTDB does not account for diagnoses related to chronic obstructive pulmonary disease and ischemic heart disease, which correspond to three input variables in the original HARM model. Input variables in the data set that were not dealt with in HARM's original study were still considered for the forward selection procedure, but these input variables were not combined with any other input variable. For the ICD-9 experiment, ICD-9 codes that were not replaced and new variables for ICD-9 codes were considered as input variables. For the augmented experiment, new variables for ICD-9 codes and patient demographics; ICD-9 codes and patient demographics not replaced; and general trauma assessments were considered as input variables. Forward selection was performed until LL on the validation set no longer improved for each experiment.

2.3.3. TMPM

TMPM is a probit regression model that maps ICD-9 codes to numeric severity values ("MARC values") in order to reduce dimensionality (please see [9] for an overview of TMPM's model development). There is only one difference between our model development procedure for TMPM and that specified in its original study. TMPM was originally developed using information pertaining to a patient's five largest MARC values as input variables. We instead used forward selection to select the input variables for TMPM. For the ICD-9 experiment, every MARC value a patient may have as well as first-order interactions between the five largest MARC values were considered for forward selection. For the augmented experiment, the same input variables in the ICD-9 experiment as well as patient demographics and general trauma

assessments were considered for forward selection. Forward selection was performed until LL did not improve on the validation set. We found that this forward selection procedure improved LL of TMPM relative to following TMPM's model development procedure exactly.

2.3.4. TSM

TSM was developed using stacked generalization [17, 18]. Our approach to stacked generalization followed this sequence. First, several machine learning models, or base models, are created from four machine learning algorithms: logistic regression with the elastic net penalty [20], random forests [21], gradient boosted machines [22], and feed-forward neural networks [23]. The feed-forward neural networks were developed with the AdaDelta optimizer [24] and the Hogwild stochastic gradient update scheme [25]. During the training process, five-fold cross-validation was used to gather approximate out-of-sample risk predictions (cross-validated risk predictions) from each base model. Each base model's cross-validated risk predictions are then combined to create a "meta-learner training set," which is used to develop a higher-level model (a meta-learner). For clarity, the meta-learner training set consists of each base model's cross-validated risk predictions as the input variables, and a binary indicator specifying whether or not the corresponding patient died prior to discharge as the output variable.

The meta-learner for TSM is a gradient boosted machine, which was developed using an exhaustive grid search where the only hyper-parameter varied was the maximum depth the trees in a gradient boosted machine were allowed to grow (from 1 to 16 with an increment of 1). All other hyper-parameters were set to their default values in h2o except the learning rate (which was set to 0.05), the annealing parameter for the learning rate (which was set to 0.99), and the number of trees developed (the default early stopping protocol in h2o was used with the validation set to

determine how many trees to develop) [25]. The gradient boosted machine with the lowest LL on the validation set was selected as the meta-learner for TSM.

2.3.5. TSM Hyper-Parameter Search Procedure

A benefit of using stacked generalization with cross-validation is that the meta-learner for TSM is developed from the same patients used to develop its base models. This allows appropriate comparison between TSM's base models, TSM's meta-learner, BLISS, HARM, and TMPM. However, in order to ensure appropriate model comparison, strong performing base models must be developed, which depends on the configuration of the hyper-parameter space for a hyper-parameter search procedure. This can be problematic in practice, as a hyper-parameter space is user-defined, and the user may configure the hyper-parameter space inappropriately [26]. To avoid this potential issue, we propose the following search procedure for hyper-parameter optimization.

First, a manual search is performed to determine an initial hyper-parameter space configuration for a machine learning algorithm. Then, machine learning models are developed using a random search for hyper-parameters within this initial configuration [27]. After 5 models are developed from this initial hyper-parameter space (10 for neural networks due to its larger number of hyper-parameters), a checking procedure is performed. For clarity when describing the sequence of this checking procedure, hyper-parameters denote the inputs of a machine learning algorithm, input values denote the hyper-parameters used to develop a machine learning model, and hyper-parameter interval denotes a dimension of a hyper-parameter space. (1) The top 2 performing models are selected based on their LL on the validation set (3 for neural networks). (2) The input values of these selected models are examined to determine where they lie on their corresponding

hyper-parameter intervals. (3) For every hyper-parameter interval where the corresponding input values of all selected models are in the top (or bottom) quarter of their hyper-parameter intervals, shift these hyper-parameter intervals such that the top (or bottom) quarter of these original hyper-parameter intervals now represents the bottom (or top) quarter of new hyper-parameter intervals.

If none of the hyper-parameter intervals shift after 5 (10) models are developed, then this checking procedure is performed after each subsequent model is developed and the checking procedure will give consideration to all models developed in this hyper-parameter space. If a hyper-parameter interval does shift, then the checking procedure is not performed until 5 (10) new models are developed, and the checking procedure will only give consideration to the models developed in this new hyper-parameter space. For this study, this search procedure was performed until 40 models were developed from each machine learning algorithm except neural networks, from which 80 models were developed. Table 2 provides the initial hyper-parameter space configuration of each algorithm.

2.3.6. Machine Learning Model Calibration

Naively assessing the probabilistic calibration of each model in this study may be problematic, as non-linear machine learning models (random forests, gradient boosted machines, and neural networks) can have poor probabilistic calibration when the outcome event is rare, which Table 1 indicates [28, 29]. To avoid this potential issue, a balanced training set was developed for creating these non-linear models (all non-linear base models and meta-learner for TSM). This involved randomly over-sampling patients in the training set until the balanced training set was approximately 5 times the size of the original training set, and the number of patients who survived care was approximately the same as the number of patients who did not survive care

[30]. Further, the base model from each non-linear algorithm that had the lowest LL on the validation set as well as the meta-learner for TSM were re-calibrated using isotonic regression before assessing their performance on the test set [28]. These isotonic regression models were developed with the prediction outputs of these models on the validation set.

2.4. Model Assessment

The performance of the established risk prediction models; TSM; and the logistic regression model developed with the elastic net penalty, random forest, gradient boosted machine, and neural network in TSM's ensemble that has the lowest log-loss on the validation set (selected base models) are evaluated with six performance metrics, which may be divided into three groups: threshold metrics, rank metrics, and probabilistic calibration metrics (calibration metrics). The threshold metrics are classification accuracy (ACC) and F-score (FSC). These metrics are computed based on whether or not a risk prediction is above a user-specified threshold value. ACC and FSC range from 0 to 1, where larger values indicate better performance. A threshold of 0.5 was used when computing these metrics [31].

The rank metrics used in this study are the area under the receiver operating characteristic curve (ROC) [32] and the area under the precision-recall curve (APR) [33]. Rank metrics depend on the ordering of outcomes, and not the actual risk predictions. Provided that this ordering is preserved, the range of a model's risk predictions does not affect its rank metric. These metrics measure how well positive cases (survival) are ordered before negative cases (mortality) and can be viewed as a summary of model performance across all possible thresholds. The ROC statistic may range from approximately 0.5 to 1, and APR may range from 0 to 1. Larger values indicate better performance.

Calibration metrics assess how well a risk prediction corresponds to a patient's true risk of mortality. The calibration metrics considered in this study are log-loss (LL) [16] and the Hosmer-Lemeshow statistic (HL) [34]. For these metrics, smaller values indicate better performance, where 0 represents perfect probabilistic calibration.

In addition to assessing the models with these performance metrics, calibration curves [34] and precision-recall curves [35] were developed for model assessment. The 10 largest variable importance measures from the selected base models of the augmented experiment were also compared [15, 36]. Model assessment was performed using the boot [37, 38], Metrics [39], and ResourceSelection [40] packages in the R statistical software. Variable importance measures were gathered using the h2o package [13].

3. Results

3.1. Model Performance and Variable Importance

The performance metrics of each model from the ICD-9 experiment (where only information pertaining to ICD-9 codes in the processed data set was considered as input variables) are displayed in Table 3. For the ICD-9 experiment, TSM demonstrates an improvement over BLISS, HARM, and TMPM for each performance metric. TSM also demonstrates an improvement over its base models for nearly every performance metric (the selected random forest model has better HL). The performance metrics of each model from the augmented experiment are displayed in Table 4. Every model greatly improved in performance when augmented to account for patient demographics (age and gender) as well as general trauma assessments (comorbidities, Glasgow Coma Scale response scores, injury mechanism, injury type, and intent of trauma). But, TSM outperforms every other model for each performance metric. No single base model in TSM's

ensemble consistently outperforms all other base models for every performance metric in each experiment. BLISS outperforms TSM's base models for most performance metrics in each experiment.

The 10 largest variable importance measures from the selected base models of the augmented experiment are displayed in Figure 1. Each model ranks the significance of their input variables differently. But, a Glasgow Coma Scale eye response score of 1 and a patient's age were amongst the 10 largest variable important measures for all selected base models. In general, each selected base models heavily relies on information pertaining to head trauma at the time of admission when predicting patient outcomes. The models differed in that the selected logistic regression model developed with the elastic net penalty placed a large variable importance measure on neck sprains; the random forest placed a large variable importance measure on congestive heart failure; the gradient boosted machine placed a large variable importance measure on lung injury; and the neural network placed a large variable importance measure on being physically struck by a person or object.

The calibration curves of TSM, BLISS, HARM, and TMPM models from the ICD-9 experiment are displayed in Figure 2; the calibration curves of TSM, BLISS, HARM, and TMPM models from the augmented experiment are displayed in Figure 3. TSM and BLISS consistently provide well-calibrated prediction outputs, whereas TMPM and HARM do not provide well-calibrated prediction outputs for the ICD-9 experiment. The precision-recall curves of TSM, BLISS, HARM, and TMPM models from the ICD-9 experiment are displayed in Figure 4; the precision-recall curves of TSM, BLISS, HARM, and TMPM models from the augmented experiment are displayed in Figure 5. TSM generally displays higher precision and recall than BLISS, HARM, and TMPM for all thresholds in both experiments.

Figure 6 shows the performance of our hyper-parameter search scheme for the ICD-9 experiment, and Figure 7 shows the performance of our hyper-parameter search scheme for the augmented experiment. Figure 6 demonstrates that our hyper-parameter search scheme was particularly successful when developing random forest models, as hyper-parameter space shifts correspond to decreasing LL on the validation set. Figure 7 shows that no hyper-parameter space shifting occurred during this experiment. Table 5 displays the hyper-parameters of the selected machine learning base models from each experiment (the models with lowest LL on the validation set). The maximum tree depth hyper-parameter of the meta-learner selected for TSM was 1 in both experiments.

3.2. Discussion

Trauma is the leading cause of death for people younger than 44, and the fourth leading cause of death for all age groups in the United States [41]. As healthcare spending has grown to 17.8% of the Gross Domestic Product, it is increasingly important to take the cost of care into consideration when improving the quality of trauma care [42]. But in order to achieve the goals of improving the quality of trauma care while controlling the cost of care, we must utilize the best possible risk prediction models in trauma system evaluations. If risk prediction can be improved, so too can the quality of trauma care, as better risk prediction models allow for a better evaluation of novel treatments, interventions, and policies. This study demonstrates that TSM, a machine learning model, outperforms established risk prediction on every performance metric considered in this study.

There is controversy regarding the utility of machine learning in healthcare [43]. This is in part motivated by several studies that compared generalized linear models to individual machine

learning models for risk prediction, often with contradictory results [44-47]. This phenomenon is in part due to the fact that no single algorithm is inherently better than all others – depending on the performance metric and the complexity of the data, the best predictive model may be developed from any algorithm [48, 49]. This claim is evidenced by the results of this study, as TSM's base models outperform established risk prediction models on some performance metrics, while established risk prediction models outperform TSM's base models on other performance metrics.

What separates an ensemble machine learning approach, such as stacked generalization, from a methodology where a single model is selected and assessed is that, if performed appropriately, stacked generalization will utilize its base models' strengths while compensating for their weaknesses. As a result, it is likely that a well-designed ensemble machine learning model developed from stacked generalization will obtain better predictive performance than any base model in its ensemble [17, 18, 50-52]. This is also indicated by the results in this study, as TSM outperforms its base models on nearly every performance metric.

The challenge with developing a well-designed ensemble machine learning model is that the ensemble must consist of base models that have strong predictive performance (ensemble strength) as well as base models that provide different prediction outputs for the same conditions (ensemble diversity). Our hyper-parameter search scheme attempts to address both of these, as a random search can provide both ensemble strength and ensemble diversity with regards to a hyper-parameter space, and hyper-parameter space shifting attempts to improve hyper-parameter space configuration if the optimal hyper-parameters lie beyond the initial hyper-parameter space configured.

While our hyper-parameter search scheme worked for this study, our search scheme is not

guaranteed to work for all settings, as it is dependent on the sensitivity of the initial hyper-parameter space configured as well as the state of the random number generator. Issues pertaining to sensitivity may be addressed by taking careful measures to configure an appropriate initial hyper-parameter space. Issues pertaining to random number generation may be addressed by developing a large number of models, examining a large number of models with regards to a small portion of each hyper-parameter interval, and specifying a small distance to shift a hyper-parameter interval.

A potential concern with the results of this study is that our hyper-parameter search procedure may have been insufficient due to BLISS outperforming the non-linear base models on most performance metrics. But, this is a consequence of re-calibrating these non-linear models. In particular, due to the tradeoff between discrimination and probabilistic calibration [53], the ROC of the random forest, gradient boosted machine, and neural network base models selected for model assessment diminished. Although this improved the probabilistic calibration of these models, some models, such as the selected gradient boosted machine in the ICD-9 experiment, were so poorly calibrated that their overall performance appeared poor when re-calibrated.

While this study highlights the strengths of developing an ensemble machine learning model, most medical studies do not require anything more sophisticated than a generalized linear model. This is due to their low computational cost, their simple functional form (which captures the underlying relationships in most medical data sets), and the interpretability as well as consistency of their weights. Arguably, generalized linear models could still be considered most appropriate for mortality risk prediction with trauma patients, as the established risk prediction models have strong predictive performance, and it is currently unknown whether or not TSM will display a clinically significant improvement over these established risk prediction models in other settings.

However, as modeling problems with healthcare data become increasingly complex, non-linear machine learning algorithms should be considered, as they can automatically find non-linear relationships in data [16]. Generalized linear models, on the other hand, would require extensive feature engineering for such data, and depending on the setting such measures will not result in the development of a model that performs as well as a model developed from machine learning algorithms. This claim is partly validated by our results, as HARM, which is developed using extensive feature engineering based on clinical intuition and expert consensus, generally had worse performance metrics than TSM's base models. Further, the variable importance measures of the non-linear models from Figure 1 reflect reality, as firearm injuries (ranked highly by the random forest) as well as vehicular accidents (ranked highly by the neural network) are considered significant variables in predicting patient outcome [11]. This indicates that the variable importance measures of non-linear machine learning models can also have value in studies necessitating interpretability.

To address a major concern with the use of machine learning algorithms in healthcare, these results do not imply that the prediction outputs of machine learning models should replace expert opinion. But, the use of machine learning models with expert opinion can greatly improve patient care in a variety of settings, as indicated in [54].

4. Conclusions

The Trauma Severity Model improves over established risk prediction models for every performance metric considered in this study, which gives it prognostic value in trauma system evaluations. The hyper-parameter search scheme proposed for this study performed well and developed strong performing machine learning models. The performance of an ensemble machine

learning model on a well-studied problem in epidemiology indicates that ensemble machine learning approaches may be fruitful for other complex problems in healthcare.

Conflict of Interest

The authors declare that they have no conflicts of interest in regards to the content in this article.

Acknowledgements

The authors would like to thank Dr. Laurent Glance and Dr. Turner Osler for their invaluable assistance on this project. The authors would also like to thank Ph.D. student Hugo Milan for conversations concerning this manuscript. These results were presented at the Eastern Association for the Surgery of Trauma Annual Scientific Assembly in 2017.

References

[1] T. Mathes, C. Mosch, M. Eikermann, Economic Aspects of Trauma Care, Springer, 2016, pp. 9-14.

[2] S. P. Baker, B. O'Neill, W. Haddon, W. B. Long, The injury severity score: A method for describing patients with multiple injuries and evaluating emergency care, Journal of Trauma and Acute Care Surgery, 1974.

[3] H. R. Champion, W. J. Sacco, A. J. Carnazzo, W. Copes, W. J. Fouty, Trauma score, Critical Care Medicine, 1981.

[4] C. R. Boyd, M. A. Tolson, W. S. Copes, Evaluating trauma care: the triss method, Journal of Trauma and Acute Care Surgery, 1987.

[5] H. R. Champion, W. S. Copes, W. J. Sacco, M. M. Lawnick, L. W. Bain, D. S. Gann, T.

Gennarelli, E. Mackenzie, S. Schwaitzberg, A new characterization of injury severity, Journal of Trauma and Acute Care Surgery, 1990.

[6] T. Osler, R. Rutledge, J. Deis, E. Bedrick, Iciss: An international classification of disease-9 based injury severity score, Journal of Trauma and Acute Care Surgery, 1996.

[7] R. S. Burd, M. Ouyang, D. Madigan, Bayesian logistic injury severity score: A method for predicting mortality using international classification of disease-9 codes, Academic Emergency Medicine, 2008.

[8] T. A. West, F. P. Rivara, P. Cummings, G. J. Jurkovich, R. V. Maier, Harborview assessment for risk of mortality: An improved measure of injury severity on the basis of icd-9cm, Journal of Trauma and Acute Care Surgery, 2000.

[9] L. G. Glance, T. M. Osler, D. B. Mukamel, W. Meredith, J. Wagner, A. W. Dick, Tmpm-icd9: A trauma mortality prediction model based on icd-9-cm codes, Annals of Surgery, 2009.

[10] A. Banerjee, S. Chaudhury, Statistics without tears: Populations and samples, Industrial Psychiatry Journal, 2010.

[11] American college of surgeons, Ntdb annual report 2016, https://www.facs.org/~/media/files/quality%20programs/trauma/ntdb/ntdb%20annual%20report%202016.ashx

[12] W. H. Greene, Econometric Analysis, 5th Edition, 1993, Ch. 21.

[13] The H2O.ai team. h2o: R Interface for H2O, version 3.16.0.2, 2017.

[14] A. Zeileis, Econometric computing with hc and hac covariance matrix estimators, Journal of Statistical Software, 2004.

[15] R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, 2017.

[16] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Springer, 2003.

[17] D. H. Wolpert, Stacked generalization, Neural Networks, 1992.

[18] M. J. van der Laan, E. C. Polley, A. E. Hubbard, Super learner, Statistical Applications in Genetics and Molecular Biology, 2007.

[19] H. Zou, T. Hastie, Regularization and variable selection via the elastic net, Journal of the Royal Statistical Society: Series B, 2005.

[20] L. Breiman, Random forests, Machine Learning, 2001.

[21] J. H. Friedman, Greedy function approximation: A gradient boosting machine. The Annals of Statistics, 2001.

[22] Y. Bengio, I. J. Goodfellow, A. Courville, Deep learning, MIT Press, 2015.

[23] M. D. Zeiler, Adadelta: An adaptive learning rate method, arXiv:1212.5701, 2012.

[24] B. Recht, C. Re, S. Wright, F. Niu, Hogwild: A lock-free approach to parallelizing stochastic gradient descent, NIPS, 2011.

[25] R. Caruana, S. Lawrence, C. L. Giles, Overfitting in neural networks: Backpropagation, conjugate gradient, and early stopping, NIPS, 2001.

[26] Y. Bengio, Practical Recommendations for Gradient-Based Training of Deep Architectures, Springer, 2012.

[27] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, Journal of Machine Learning Research, 2012.

[28] A. Niculescu-Mizil, R. Caruana, Predicting good probabilities with supervised learning, ICML, 2005.

[29] C. Guo, G. Pleiss, Y. Sun, K. Q. Weinberger, On calibration of modern neural networks, arXiv preprint arXiv:1706.04599, 2017.

[30] G. E. A. P. A. Batista, R. C. Prati, M. C. Monard, A study of the behavior of several methods for balancing machine learning training data, SIGKDD Explorations, 2004.

[31] R. Caruana, A. Niculescu-Mizil, G. Crew, A. Ksikes, Ensemble selection from libraries of models, ICML, 2004.

[32] J. A. Hanley, B. J. McNeil, The meaning and use of the area under a receiver operating characteristic (roc) curve, Radiology. 1982.

[33] J. Davis, M. Goadrich, The relationship between precision-recall and roc curves, ICML, 2006.

[34] D. W. Hosmer, S. Lemeshow, Applied Logistic Regression, 2nd Edition, Wiley, 2000, Ch. 5.

[35] T. Saito, M. Rehmsmeier, The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets, Plos One, 2015.

[36] T. D. Gedeon, Data mining of inputs: Analysing magnitude and functional measures, International Journal of Neural Systems, 1997.

[37] A. Canty, B. Ripley, boot: Bootstrap R (S-plus) functions, https://cran.r-project.org/web/packages/boot/boot.pdf

[38] A. C. Davison, D. V. Hinkley, Bootstrap Methods and Their Applications, Cambridge University Press, 1997.

[39] B. Hamner, M. Frasco, Metrics: Evaluation metrics for machine learning, https://CRAN.R-project.org/package=Metrics(2018)

[40] S. R. Lele, J. L. Keim, P. Solymos, ResourceSelection: Resource selection (probability) functions for use-availability data, https://CRAN.R-project.org/package=ResourceSelection(2017)

[41] Center for disease control. ten leading causes of death by age group, united states -2014, https://www.cdc.gov/injury/wisqars/pdf/leading_causes_of_death_by_age_group_2014-a.pdf

[42] A. B. Martin, M. Hartman, B. Washington, A. Catlin, National health expenditure accounts team. National health spending: Faster growth in 2015 as coverage expands and utilization increases, Health Affairs, 2017.

[43] A. Verghese, N. H. Shah, R. A. Harrington, What this computer needs is a physician - humanism and artificial intelligence, JAMA, 2017.

[44] R. Dybowski, P. Weller, R. Chang, V. Gant, Prediction of outcome in critically ill patients using artificial neural network synthesised by genetic algorithm, The Lancet Respiratory Medicine, 1996.

[45] G. Clermont, D. C. Angus, S. M. DiRusso, M. Griffin, W. T. Linde-Zwirble, Predicting hospital mortality for patients in the intensive care unit: A comparison of artificial neural networks with logistic regression models, Critical Care Medicine, 2001.

[46] V. J. Ribas, J. C. Lopez, A. Ruiz-Sanmartin, J. C. Ruiz-Rodriguez, J. Rello, A. Wojdel, A. Vellido, Severe sepsis mortality prediction with relevance vector machines, EMBC, 2011.

[47] S. Kim, W. Kim, R. W. Park, A comparison of intensive care unit mortality prediction models through the use of data mining techniques, Health Informatics Research, 2011.

[48] R. Pirracchio, M. L. Petersen, M. Carone, M. R. Rigon, S. Chevret, M. J. van der Laan, Mortality prediction in intensive care units with the super icu learner algorithm (sicula): A population-based study, The Lancet Respiratory Medicine, 2015.

[49] D. H. Wolpert and W. G. Macready, No free lunch theorems for optimization, IEEE Transactions of Evolutionary Compution, 1997.

[50] D. Opitz, R. Maclin, Popular ensemble methods: An empirical study, Journal of Artificial Intelligence Research, 1999.

[51] R. Polikar, Ensemble based systems in decision making, IEEE, 2006.

[52] L. Rokach, Ensemble-based classifiers, Artificial Intelligence Review, 2010.

[53] G. A. Diamond, What price perfection? Calibration and discrimination of clinical prediction models, Journal of Clinical Epidemiology, 1992.

[54] D. Wang, A. Khosla, R. Gargeya, H. Irshad, A. H. Beck Wang, Deep learning for identifying metastatic breast cancer. arxiv:1606.05718, 2016.

Demographic Characteristics	
Age (Interquartile Range)	(23, 61)
Male (%)	63.93
Number of Hospitals*	713
Mortality before Discharge (%)	3.77
Racial Characteristics	
White (%)	64.52
Black or African American (%)	16.17
Hispanic‖ (%)	12.74
Asian (%)	1.93
Native American/Native Hawaiian/Pacific Islander (%)	0.79
Other (%)	10.82
Not Recorded (%)	5.77

(*) Specific hospital demographics not displayed due to this information changing each year in NTDB.

(‖) Hispanic is denoted as an ethnicity in NTDB data, not race.

Table 1: Demographics for the processed data set.

Hyper-parameter	ICD-9 experiment	Augmented experiment
Logistic Regression with Elastic Net Penalty		
α	$\mathcal{U}(0, 1)$[a]	$\mathcal{U}(0, 1)$
λ	$10^{\mathcal{U}_d(-10, -1)}$	$10^{\mathcal{U}_d(-10, -1)}$
Random Forest		
#Trees	$\mathcal{U}_d(50, 150)$[a]	$\mathcal{U}_d(50, 150)$
MNL[b]	$\mathcal{U}_d(1, 75)$	$\mathcal{U}_d(1, 75)$
NVS[c]	$\mathcal{U}_d(40, 150)$	$\mathcal{U}_d(40, 150)$
Max. Tree Depth	$\mathcal{U}_d(1, 50)$	$\mathcal{U}_d(60, 180)$
Gradient Boosted Machine		
#Trees	$\mathcal{U}_d(10, 80)$	$\mathcal{U}_d(10, 80)$
Max. Tree Depth	$\mathcal{U}_d(1, 15)$	$\mathcal{U}_d(1, 15)$
Learning rate	$\mathcal{U}(0.05, 0.50)$	$\mathcal{U}(0.05, 0.50)$
Annealing	$\mathcal{U}(0.850, 0.999)$	$\mathcal{U}(0.850, 0.999)$
Neural Networks		
#Hidden layers	$\mathcal{U}_d(1, 4)$	$\mathcal{U}_d(1, 4)$
#Neurons	$\mathcal{U}_d(1, 2^{11-\#\text{Hidden Layers}})$	$\mathcal{U}_d(1, 2^{11-\#\text{Hidden Layers}})$
Activation function	ReLU or Hyperbolic Tangent	ReLU or Hyperbolic Tangent
Dropout rates[d]	$\mathcal{U}(0, 0.33)$	$\mathcal{U}(0, 0.33)$
Epochs	$\mathcal{U}_d(10, 10{,}000)$	$\mathcal{U}_d(10, 10{,}000)$
ρ[e]	$\mathcal{U}(0.75, 0.999)$	$\mathcal{U}(0.75, 0.999)$
ε[e]	$10^{\mathcal{U}(-12, -3)}$	$10^{\mathcal{U}(-12, -3)}$

(a) $\mathcal{U}(a, b)$ denotes uniform continuous distribution from *a* to *b*, $\mathcal{U}_d(a, b)$ denotes uniform discrete distribution from *a* to *b*.
(b) MNL: minimum number of observations in a leaf.
(c) NVS: number of variables used in each split.
(d) The dropout rate was allowed to differ for each hidden layer, as this improved predictive performance.
(e) Hyper-parameters from the AdaDelta optimizer.

Table 2: Initial hyper-parameter space configured for each machine learning algorithm. The number of trees in a random forest was not allowed to shift during our hyper-parameter search procedure.

Model	ACC	FSC	ROC	APR	HL	LL
TSM	**0.968 ($3.368\cdot10^{-4}$)**	**0.404 ($5.172\cdot10^{-3}$)**	**0.912 ($1.556\cdot10^{-3}$)**	**0.489 ($5.387\cdot10^{-3}$)**	**84.400 (17.183)**	**0.098 ($8.597\cdot10^{-4}$)**
BLISS	0.967 ($3.420\cdot10^{-4}$)	0.369 ($5.097\cdot10^{-3}$)	0.900 ($1.747\cdot10^{-3}$)	0.448 ($5.518\cdot10^{-3}$)	556.357 (38.350)	0.108 ($9.439\cdot10^{-4}$)
HARM	0.965 ($3.426\cdot10^{-4}$)	0.299 ($4.798\cdot10^{-3}$)	0.866 ($2.027\cdot10^{-3}$)	0.378 ($5.021\cdot10^{-3}$)	140.257 (22.619)	0.114 ($9.230\cdot10^{-4}$)
TMPM	0.966 ($3.421\cdot10^{-4}$)	0.336 ($5.135\cdot10^{-3}$)	0.898 ($1.753\cdot10^{-3}$)	0.435 ($5.450\cdot10^{-3}$)	154.692 (40.537)	0.105 ($9.221\cdot10^{-4}$)
PLM	0.966 ($3.419\cdot10^{-4}$)	**0.371 ($5.076\cdot10^{-3}$)**	0.899 ($1.749\cdot10^{-3}$)	0.448 ($5.511\cdot10^{-3}$)	529.964 (37.479)	0.108 ($9.462\cdot10^{-4}$)
RF	**0.967 ($3.369\cdot10^{-4}$)**	0.364 ($5.132\cdot10^{-3}$)	0.899 ($1.727\cdot10^{-3}$)	**0.452 ($5.224\cdot10^{-3}$)**	**77.420 (16.719)**	**0.104 ($8.880\cdot10^{-4}$)**
GBM	0.966 ($3.475\cdot10^{-4}$)	0.313 ($5.254\cdot10^{-3}$)	0.887 ($1.896\cdot10^{-3}$)	0.420 ($5.439\cdot10^{-3}$)	2510.427 (60.227)	0.114 ($7.794\cdot10^{-4}$)
NN	0.966 ($3.452\cdot10^{-4}$)	0.291 ($5.111\cdot10^{-3}$)	**0.902 ($1.678\cdot10^{-3}$)**	0.434 ($5.419\cdot10^{-3}$)	305.524 (29.035)	0.104 ($8.563\cdot10^{-4}$)

Table 3: Model comparison for the ICD-9 experiment. Standard error of the metric is denoted in the parenthesis. TSM consistently has superior performance under each performance metric except for the HL statistic, which was attained by the random forest base model. PLM, RF, GBM, and NN denote logistic regression with the elastic net penalty, random forest, gradient boosted machine, and neural network, respectively.

Model	ACC	FSC	ROC	APR	HL	LL
TSM	**0.976 ($2.964\cdot10^{-4}$)**	**0.621 ($4.277\cdot10^{-3}$)**	**0.965 ($7.936\cdot10^{-4}$)**	**0.696 ($4.485\cdot10^{-3}$)**	**23.341 (14.147)**	**$6.889\cdot10^{-2}$ ($7.165\cdot10^{-4}$)**
BLISS	0.975 ($2.925\cdot10^{-4}$)	0.601 ($4.235\cdot10^{-3}$)	0.957 ($9.828\cdot10^{-4}$)	0.665 ($4.795\cdot10^{-3}$)	95.063 (17.530)	$7.498\cdot10^{-2}$ ($7.676\cdot10^{-4}$)
HARM	0.973 ($2.972\cdot10^{-4}$)	0.564 ($4.353\cdot10^{-3}$)	0.955 ($9.914\cdot10^{-4}$)	0.631 ($5.086\cdot10^{-3}$)	115.840 (16.765)	$7.810\cdot10^{-2}$ ($7.416\cdot10^{-4}$)
TMPM	0.973 ($2.992\cdot10^{-4}$)	0.573 ($4.378\cdot10^{-3}$)	0.958 ($9.118\cdot10^{-4}$)	0.643 ($4.771\cdot10^{-3}$)	135.461 (84.870)	$7.577\cdot10^{-2}$ ($7.472\cdot10^{-4}$)
PLM	**0.974 ($2.949\cdot10^{-4}$)**	**0.593 ($4.291\cdot10^{-3}$)**	0.955 ($1.007\cdot10^{-4}$)	0.653 ($4.862\cdot10^{-3}$)	96.256 (16.654)	**$7.599\cdot10^{-2}$ ($7.719\cdot10^{-4}$)**
RF	0.974 ($2.996\cdot10^{-4}$)	0.577 ($4.456\cdot10^{-3}$)	**0.957 ($8.913\cdot10^{-4}$)**	**0.653 ($4.641\cdot10^{-3}$)**	40.772 (21.683)	$7.608\cdot10^{-2}$ ($7.449\cdot10^{-4}$)
GBM	0.974 ($3.063\cdot10^{-4}$)	0.561 ($4.569\cdot10^{-3}$)	0.957 ($9.171\cdot10^{-4}$)	0.641 ($5.060\cdot10^{-3}$)	135.050 (18.694)	$7.644\cdot10^{-2}$ ($7.136\cdot10^{-4}$)
NN	0.971 ($3.193\cdot10^{-4}$)	0.540 ($4.528\cdot10^{-3}$)	0.955 ($1.052\cdot10^{-4}$)	0.598 ($5.322\cdot10^{-3}$)	**39.630 (12.818)**	$7.780\cdot10^{-2}$ ($7.608\cdot10^{-4}$)

Table 4: Model performance comparison for the augmented experiment. Standard error of the metric is denoted in the parenthesis. TSM consistently has superior performance under each metric. PLM, RF, GBM, and NN denote logistic regression with the elastic net penalty, random forest, gradient boosted machine, and neural network, respectively.

Hyper-parameter	ICD-9 Experiment	Augmented Experiment
Logistic Regression with Elastic Net Penalty		
λ	10^{-4}	10^{-4}
α	0.845	0.720
Random Forest		
#Trees	83	138
MNL[a]	29	70
NVS[b]	70	107
Max. Tree Depth	157	71
Gradient Boosted Machine		
#Trees	80	66
Max. Tree Depth	15	8
Learning rate	0.670	0.250
Annealing	0.962	0.984
Neural Network		
#Neurons	(41, 42, 12, 49)	(13, 60, 35, 62)
Activation function	ReLU	ReLU
Dropout rates	(0.15, 0.03, 0.28, 0.31)	(0.02, 0.01, 0.23, 0.32)
Epochs	163	91
ρ	0.978	0.991
ε	$3.162 \cdot 10^{-10}$	10^{-10}

(a) MNL: minimum number of observations in a leaf.
(b) NVS: number of variables used in each split.

Table 5: Hyper-parameters of the selected base models from each experiment.

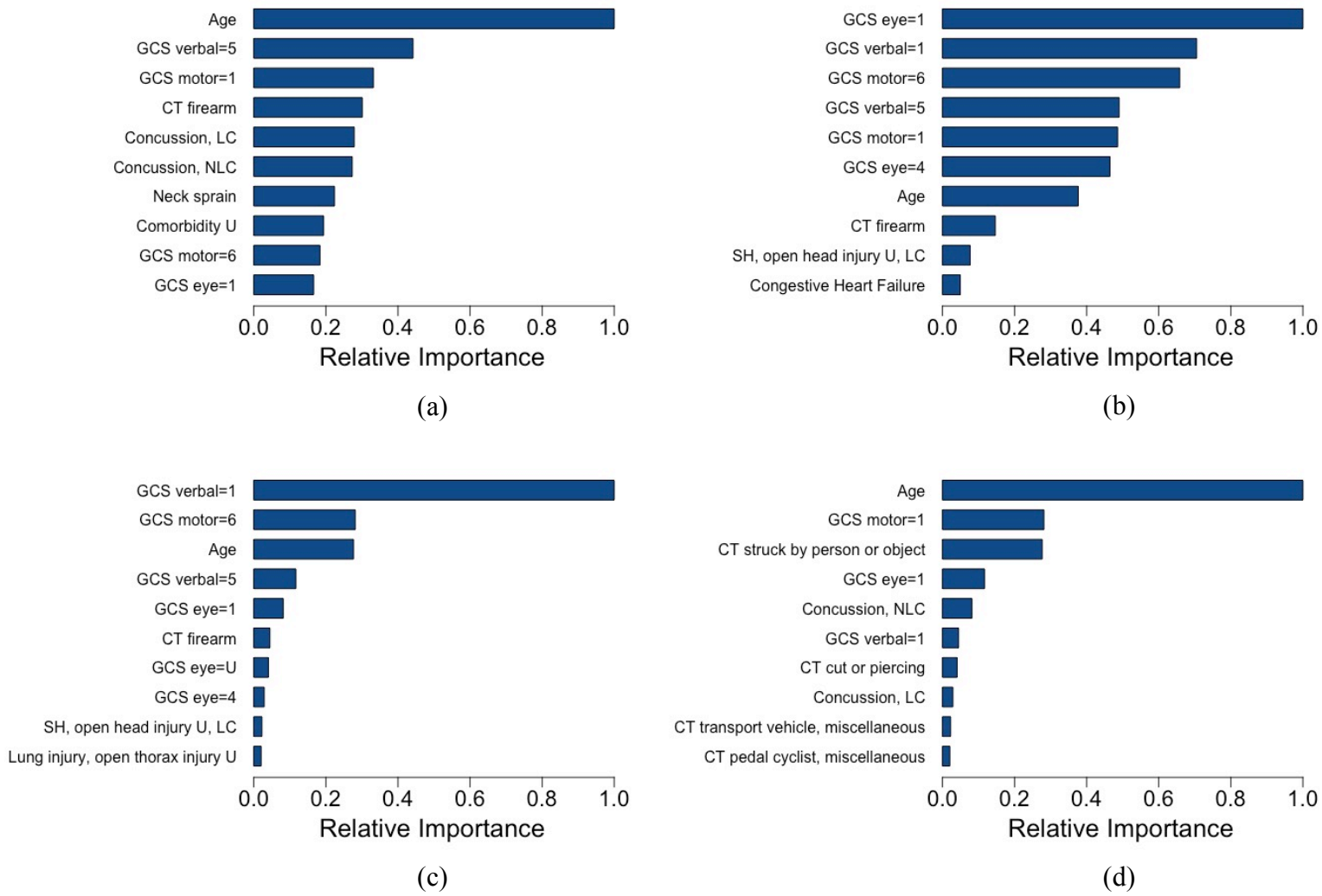

Figure 1: 10 largest variable importance measures of the selected base models from TSM's ensemble for the augmented experiment. The selected logistic regression model developed with the elastic net penalty is denoted by (a), random forest by (b), gradient boosted machine by (c), and neural network by (d). Further, GCS denotes Glasgow Coma Scale, CT denotes cause of trauma, LC denotes loss of consciousness, NLC denotes no loss of consciousness, U denotes unknown, and SH denotes subdural hemorrhage.

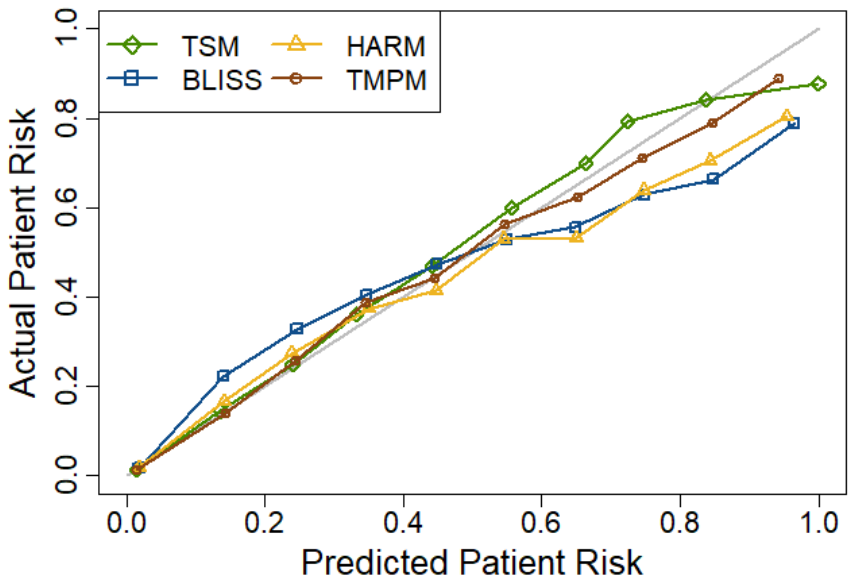

Figure 2: Calibration curves for TSM, HARM, BLISS, and TMPM models from the ICD-9 experiment. The grey line represents perfect probabilistic calibration. TSM and TMPM provide well-calibrated prediction outputs.

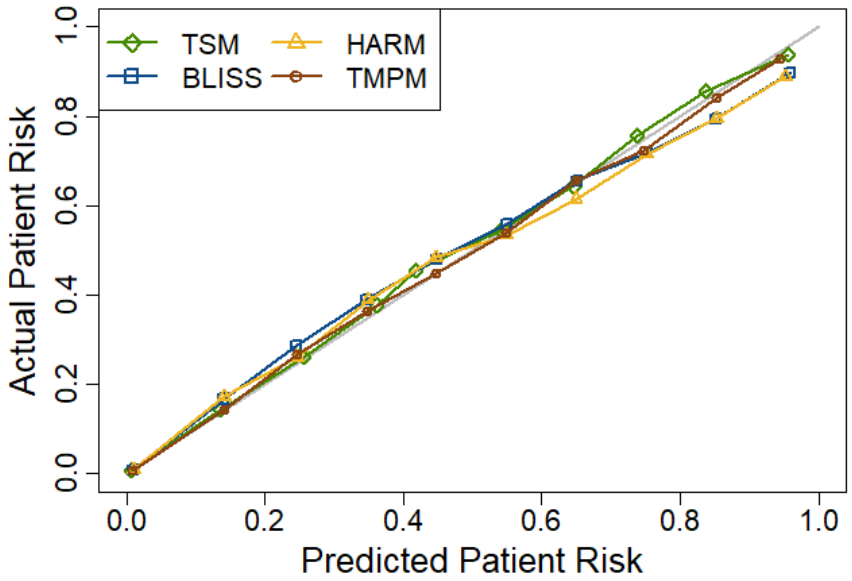

Figure 3: Calibration curves for TSM, HARM, BLISS, and TMPM models from the augmented experiment. The grey line represents perfect probabilistic calibration. Every model provides well-calibrated prediction outputs.

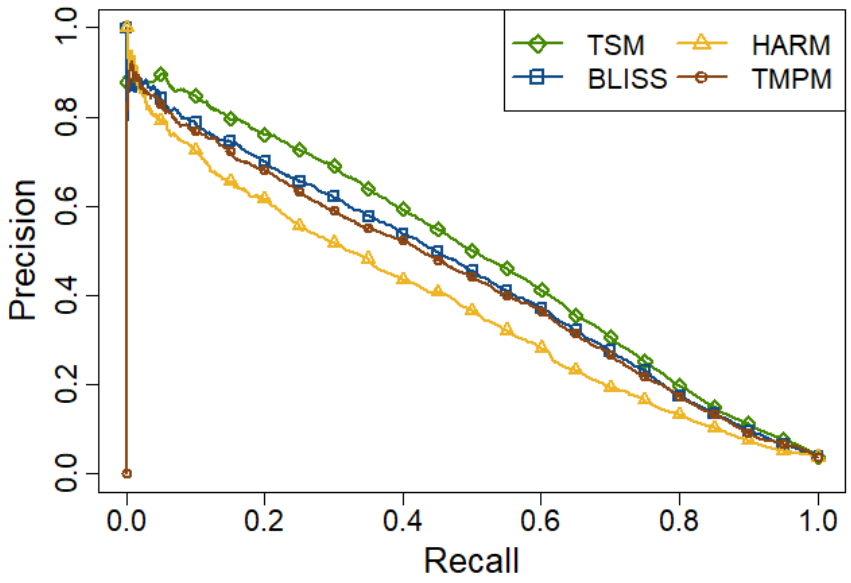

Figure 4: Precision-recall curves for TSM, BLISS, HARM, and TMPM models from the ICD-9 experiment. TSM generally had higher precision and recall for all thresholds.

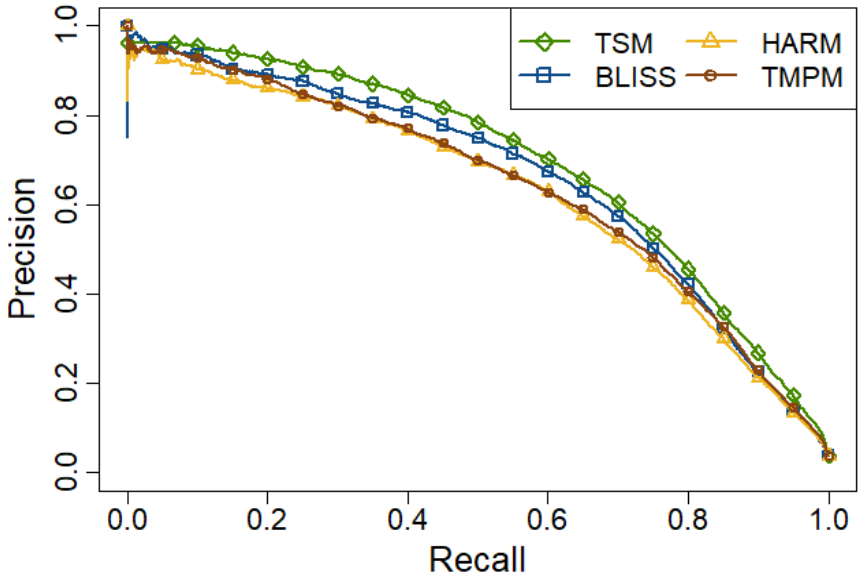

Figure 5: Precision-recall curves for TSM, BLISS, HARM, and TMPM models from the augmented experiment. TSM generally had higher precision and recall for all thresholds.

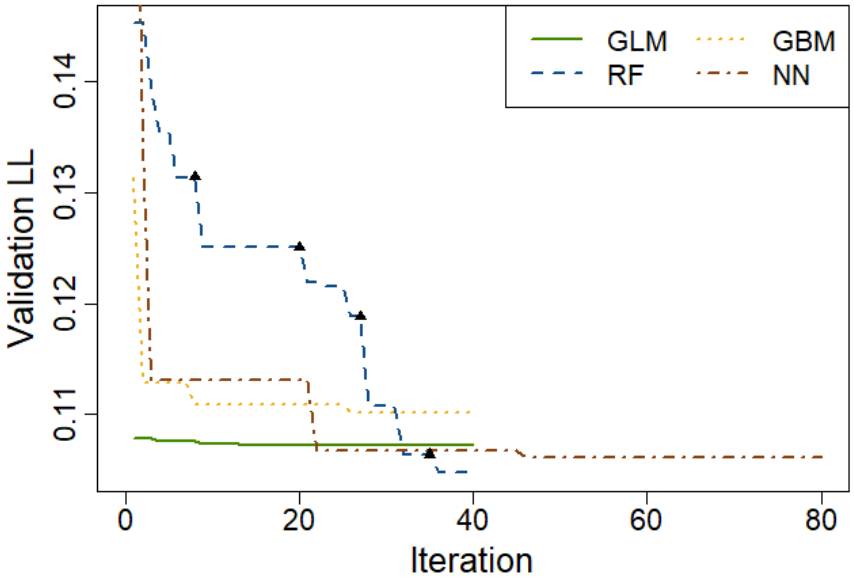

Figure 6: Log-loss on the validation set (validation LL) for the model that had the lowest validation LL after each iteration of our hyper-parameter search procedure from the ICD-9 experiment. The hyper-parameter space shifted four times for the random forest algorithm, and each shift is associated with a decrease in validation LL. Shifts are denoted by a black triangle (▲).

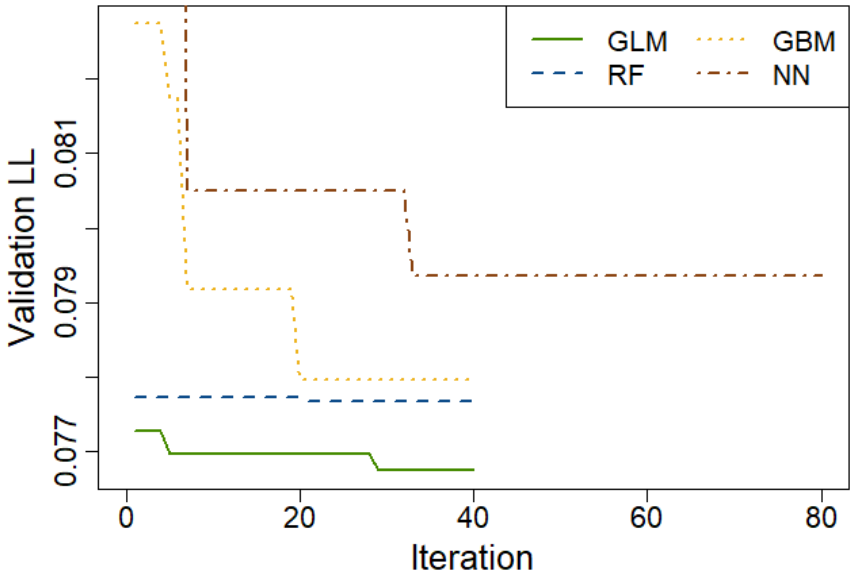

Figure 7: Log-loss on the validation set (validation LL) for the model that had the lowest validation LL after each iteration of our hyper-parameter search procedure from the augmented experiment. The hyper-parameter space did not shift for any hyper-parameter search.

Summary:

Previous studies on trauma mortality prediction from ICD-9 codes (800-959.9) have focused on the use of generalized linear models for risk prediction. Although this has resulted in beneficial mortality prediction models, there are a variety of other algorithms that may lead to the development of a model with even better predictive performance. In this study, we have developed several predictive models from different machine learning algorithms, and we have compared their predictive performance to the performance of established, widely used trauma risk prediction models. Our results indicate that these individual machine learning models have comparable performance to the established trauma risk prediction models. However, combining these machine learning models into an ensemble (using stacked generalization) leads to the development of a model with better predictive performance than the established trauma risk prediction models for each performance metric considered. Previously, stacked generalization has seldom been considered in this setting. This study indicates that intensive data-driven approaches can improve our ability to predict mortality risk of trauma patients, and thereby delineate patients in need of aggressive care.