Predicting Hospital Length of Stay from Imbalanced Data
A Classification-Regression Approach

Stanford CS229 Project

Miguel Fuentes
Stanford University
migufuen@stanford.edu

Pinlin [Calvin] Xu
Stanford University
pinlinxu@stanford.edu

Lisa Liu
Stanford University
lisacliu@stanford.edu

1 Introduction

Accurately predicting the length of stay (LoS) of admitted patients to the intensive care unit (ICU) from electronic health record (EHR) data is a relevant and growing area of research. The COVID-19 pandemic placed unprecedented stress on hospital-based care across the United States and called to attention the need to efficiently manage healthcare capacities and resources (Wu et al., 2020). Recognizing LoS as a crucial random variable that has been investigated in Statistics and Operational Research since the 1970s to aid such planning (Stone et al., 2022), this project applies machine learning methods to the Medical Information Mart for Intensive Care, version 4 (MIMIC-IV), a popular EHR dataset available containing de-identified health information from patients admitted to Beth Israel Deaconess Medical Center (Johnson et al., 2023a,b).

To predict a patient's LoS, we use both categorical and numerical characteristics of patients (e.g., sex, age, diagnoses, laboratory results) as input, and apply a cohort classifier and then respective regressors in a step-wise approach. Different algorithms such as logistic regression, decision tree, ensemble, random forest, and XGBoost, are used and evaluated. We also aimed at making the training process of our predictor quick on consumer-level desktop computers and laptops, such that our methodology will be accessible to researchers and providers of all scales. Finally, we recognize the importance of fairness in our LoS prediction and evaluate our model across different demographic sub-groups.

2 Related Work

Many recent studies use the MIMIC raw data and machine learning methods for hospital-based predictions. A 2019 paper in *Nature* presents "strong linear and neural baselines" for four clinical prediction benchmarks on MIMIC-III data, one of which is LoS forecasting. The mean absolute deviation (MAD) in unit of days after conversion is 4.85 for baseline linear regression and at best 3.92 from a channel-wise long short-term memory (LSTM) neural network with deep supervision (Harutyunyan et al., 2019). The authors report that their results for LoS forecasting are the worst among the four benchmarks due to the intrinsic difficulty of the task, and note that even small LSTMs easily overfit on this task.

Due to the difficulty of predicting LoS directly, many studies instead seek to separate long stays (that present most stress on healthcare providers) from short stays or do so first in a multistep approach. Hempel et al. (2023) predict whether the stay will be short or long (more than 4 days) with 81% classification accuracy (F1 score 0.442) using random forest and predict LoS only for data classified as short stays with RMSE of 1.13; predictions are poor when the actual LoS is longer. Harerimana et al. (2021) uses a deep attention model with pre-trained medical embeddings to classify LoS into three classes with 86% accuracy (F1 score 0.2441). These works note that due to the highly imbalanced nature of the data, accuracy is an inadequate metric: even a 95% classification accuracy does not enable good predictions of long stays, which is the minority class. The state-of-the-art for LoS prediction lies in NLP understanding using pre-trained large language models (LLMs). van Aken et al. (2021) reports that BioBERT performs well on a 4-classes LoS classification task while fine-tuning yields further gains. However, we limit our research to traditional ML models in this project to exploit their cost advantage over more advanced methods.

Other works using MIMIC illustrate good approaches for working with the data in general. (Sun et al., 2023), focusing on mortality prediction, identify that LASSO and XGBoost as useful for feature selection. Gupta et al. (2022) note the issues in unprocessed and uncleaned MIMIC data and offer a configurable pipeline that prepares MIMIC-IV for downstream tasks. As it was built for an older version of the data, their code is not used in this project. Nevertheless, ideas such as outlier removal, dimensionality reduction, etc. informed our approach as did other insights reported in this section.

3 Dataset & Features

MIMIC-IV v2.2 has 32 separate CSV files in its raw form, each containing information such as admission records, microbiology cultures, medication administration, and billed diagnoses. To build a dataset we can work with, we first inner joint tables by both patient and admission ID (eliminating most incomplete entries in the process) and preliminarily kept most features based on domain knowledge.

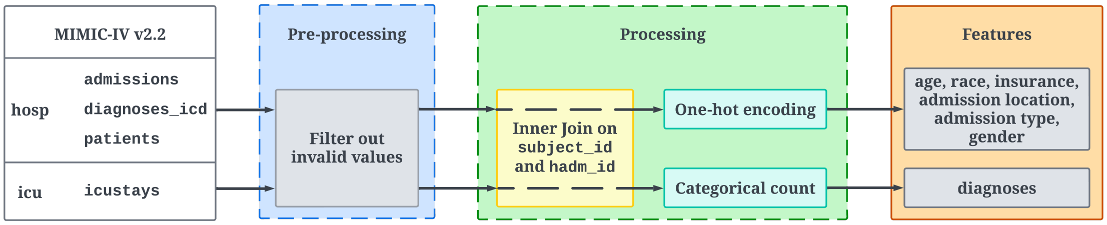

Figure 1: Feature processing on MIMIC-IV v2.2 dataset.

Categorical variables such as gender, race, and admission type are encoded in either binary or one-hot form. In particular, individual patient diagnoses in ICD-9 and ICD-10 codes are converted to 19 categories based on the International Classification of Diseases and Related Health Problems standard (WHO, 1977; Bailey et al., 2019). We also calculated desensitized information such as age from “anchored” entries and the admitted time (also desensitized). These are converted into age groups based on definitions from NLM (1998). Though Multiple Correspondence Analysis (MCA) was used at one point to reduce excess dimensionality in categorical data, ultimately the features selected are not as sparse as we expected. Because we want to predict LoS from mostly the initial information gathered at admission, microbiology or lab events after admission were excluded.

Our experiments with baseline algorithms significantly guided feature engineering. After running tests with LASSO and ElasticNet, which is a regularized GLM combining LASSO and Ridge where $\theta^* := \arg\min_\theta \|y - X\theta\|^2 + \lambda_2 \|\theta\|^2 + \lambda_1 \|\theta\|_1$, features with regression coefficients of zero are dropped after verifying they caused no performance degradation to other algorithms (e.g., marital status). Invalid entries such as deaths during hospitalization are removed. Furthermore, continuous numerical features such as temperature and heart rate are determined to contain impossible outliers, thus values above the 95th or below the 5th percentile are clamped to the range. We then compute the length of time where heart rate, respiratory rate, and temperature are measured to be normal or abnormal in the first 24 hours and normalize each category to sum to 1, also balancing out the numerical data with the categorical one-hot encodings.

We perform 10-fold cross-validation on the MIMIC dataset, using an 80%-20% training-validation split, and additionally randomly select 10% of the training data as test data to report our final results. In the end, we have 249772 training examples, 30836 validation examples, and 27753 test examples.

4 Methods

Table 1: Baseline Regression Performance (RMSE)

Baseline	OLS (with Ridge)	LASSO	ElasticNet	SGD Regressor	NN	XGBoost
Validation	3.1416	3.1692	3.1544	3.1499	3.0868	3.04795

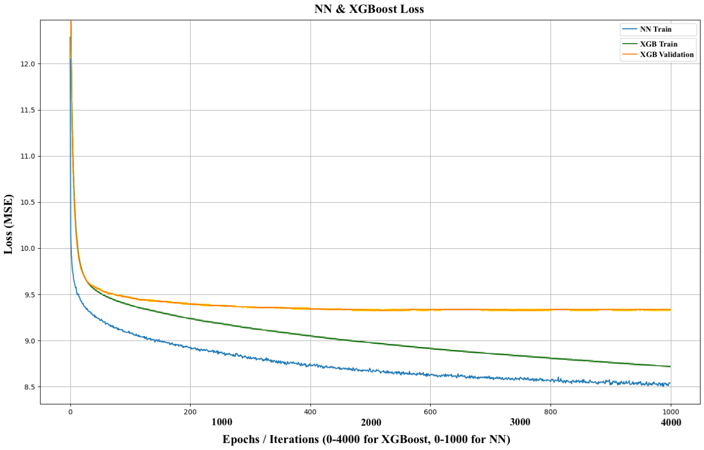

Figure 2: Learning curves of baselines.

From preliminary experiments, simply applying common regression algorithms and a neural network with 4 hidden layers we designed performed poorly as shown in Table 1. Moreover, though some learners could achieve lower asymptotic training loss, we found that validation loss stopped decreasing long before, indicative of overfitting on training data shown in Figure 2.

We surmised that the reason for poor regression performance was that the available examples are highly imbalanced as discussed in Hempel et al. (2023) (shown by Figure 3). Regressors fitted directly on the whole dataset fail badly on the minority long-stay cases, as shown in Figure 4.

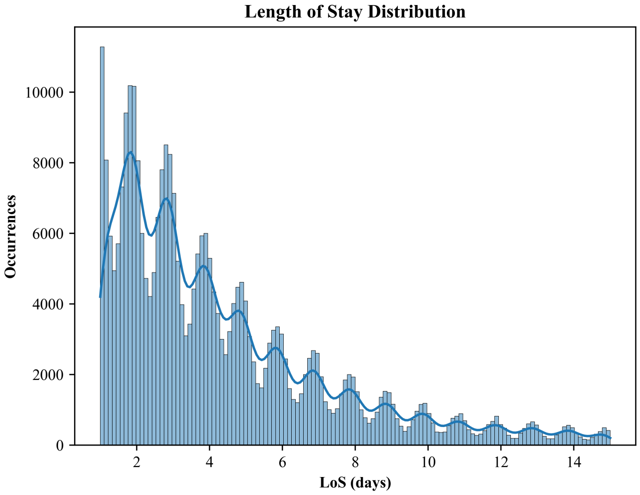

Figure 3: Histogram of LoS

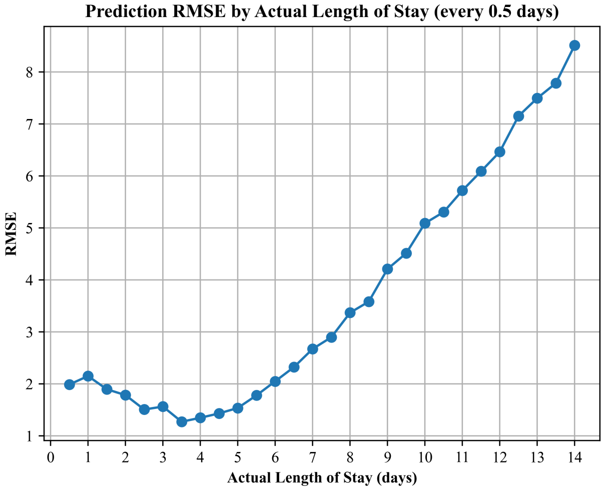

Figure 4: Linear Regression RMSE by LoS

4.1 Addressing Imbalance via Synthetic Oversampling

Aiming to overcome these challenges and go beyond the literature to predict LoS across the entire range, we initially reweighted the minority class through oversampling motivated by Problem 4 on Problem Set 2. Out of S short-stay (≤ 4 days) examples and L long-stay (> 4 days) examples, we can oversample the long-stays to achieve better balanced accuracy $\bar{A} := \frac{1}{2}(A_0 + A_1) := \frac{1}{2}(\frac{TP}{TP+FN} + \frac{TN}{TN+FP})$ where TP, FP, TN, FN stand for the true positive, false positive, true negative, and false negative count, respectively. In our case, the long stay category is positive and the short stay is negative.

We then adopted a more sophisticated method that improves the balanced accuracy and the recall for the minority class. Simply duplicating examples from the minority class does not provide new information about the data. Instead, SMOTE (Synthetic Minority Oversampling Technique) synthesizes new examples from the existing ones (Chawla et al., 2002). Modifying Chawla et al.'s original algorithm, we take $k = 5$ and consider the k nearest neighbors for each example $x^{(i)}$ in the minority class. Let $N = \lfloor \frac{S}{L} \rfloor$. For $R = S - N \cdot L$ iterations we use N and we use $N' = N - 1$ for the remaining. Now, we randomly select N (or N') of the k neighbors considered, denoted $x^{(i_n)}$ for $n = 1, \ldots, N$. For each selection, we generate a new example $\tilde{x}^{(i_n)} = \alpha \cdot x^{(i)} + (1 - \alpha)x^{(i_n)}$ for some $\alpha \in (0, 1)$. In other words, we take some point on the line segment joining $x^{(i)}$ and $x^{(i_n)}$ and add it to our sample space. In the end, we get a balanced distribution.

4.2 The Classification-Regression Pipeline

Motivated by previous literature focusing on classification, we postulated that prediction accuracy could be improved if we obtain a robust classifier separating longer stays from shorter stays, allowing the use of multiple accurate regressors each trained on one class of examples.

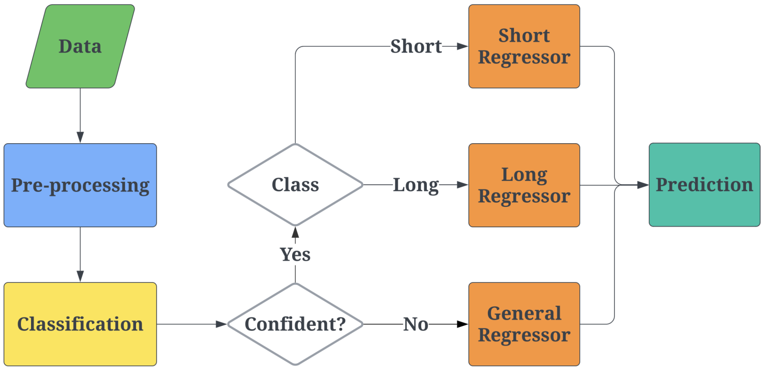

Figure 5: Classification-Regression model pipeline.

We evaluated multiple classifiers (logistic regression, XGBoost, Ensemble, and neural network) for short stays (0) vs. long stays (1). For all models, we use the binary logistic (cross-entropy) loss function

$$L(y, p) = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)],$$

where p contains the predicted probabilities and y contains the corresponding true labels in $\{0, 1\}$. XGBoost is a gradient boosting algorithm with L_1 regularization weight $\alpha = 10$. We end up with a 30% sampling of features used in building each tree, a maximum depth of 30, and 100 trees to be built as our hyperparameters. Ensemble learns the best weight for $F(x) = w_1 f_1(x) + w_2 f_2(x) + w_3 f_3(x)$ where f_1 is a random forest model, f_2 is XGBoost, and f_3 is a soft voting classifier. The neural net we use has an output activation function using sigmoid $f(x) = \frac{1}{1+\exp(-x)}$ and three layers of ReLU $g(x) = \max(0, x)$.

Using the best classification model, we assign our input to go through different regressors, as outlined in Figure 5. The three regressors considered are SGD (stochastic gradient descent) with MSE loss, ElasticNet as previously introduced, and XGBoost using MSE and L_1 regularization. To perform regression, each tree in XGBoost predicts a continuous value by averaging those given by the confines of each leaf node and each successive tree learns from the combined errors of the previous ones. We obtain a short, long, and general regressor by training these models on the examples with short stays, those with long stays, and the whole dataset. For each type of regressor, we take the probabilities p rather than the classes themselves and send inputs with $p > 0.7$ to the long regressor, inputs with $p < 0.3$ to the short regressor, and everything else to the general regressor.

5 Experiments & Results

We first identify the best classifier for predicting short (≤ 4 days) or long (> 4 days) stays. Because our original dataset is highly unbalanced, our main evaluation metric is balanced accuracy and recall, though accuracy, precision, and F1 score are also presented in Table 2. Based on this, the ensemble classifier outperformed the rest on four of the five metrics (in particular, on balanced accuracy and recall). The ensemble also displayed the highest AUC as shown in Figure 7, so the model of choice for the complete pipeline was the ensemble model.

Table 2: Classification Performance

Model	**Accuracy**	**Balanced Accuracy**	**Precision**	**Recall**	**F1 Score**
Logistic	0.71	0.70	0.75	0.74	0.74
XGBoost	**0.72**	**0.71**	0.74	0.80	**0.77**
Ensemble	**0.72**	**0.71**	0.74	**0.81**	**0.77**
NN	**0.72**	**0.71**	**0.76**	0.77	0.76

Table 3: Regression Metrics. Test and (train) results

Regressor	**RMSE**	**MAE**	**R^2**
SGD	2.53 (2.33)	1.84 (1.68)	**0.29** (0.39)
ElasticNet	2.56 (2.29)	1.87 (1.68)	0.27 (0.41)
XGBoost	**2.52** (2.28)	**1.83** (1.65)	**0.29** (0.42)

Our accuracy is lower than that presented by Harerimana et al. (2021) as we opted to increase the balanced accuracy that supports the downstream regression task, specifically aiming to reduce false negatives (FN), long stay examples incorrectly classified as short stays, which will suffer from large regression errors. Our F1 score, on the other hand, is way higher than that of Harerimana et al. (2021).

When combined, the best Classification-Regression predictor obtained via 10-fold cross-validation is the ensemble classifier combined with an XGBoost regressor as shown in Table 1. This model outperforms baseline models presented earlier across the input range, especially for long stays, as compared in Figure 6. In addition, the root mean square error (RMSE) for short stays remains stable under two days. Error in predicting LoS increases almost monotonically as actual LoS increases, aligning with the expected randomness of the stochastic process.

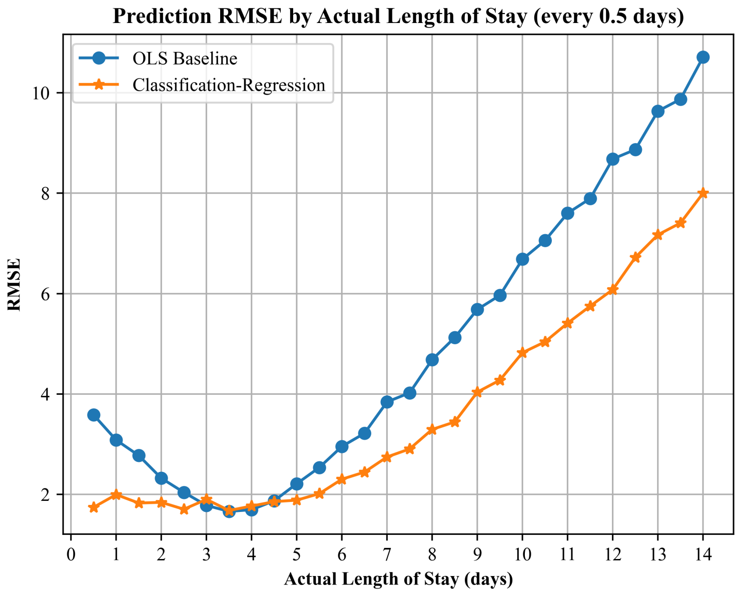

Figure 6: Classification-Regression's RMSE by LoS, lower is better

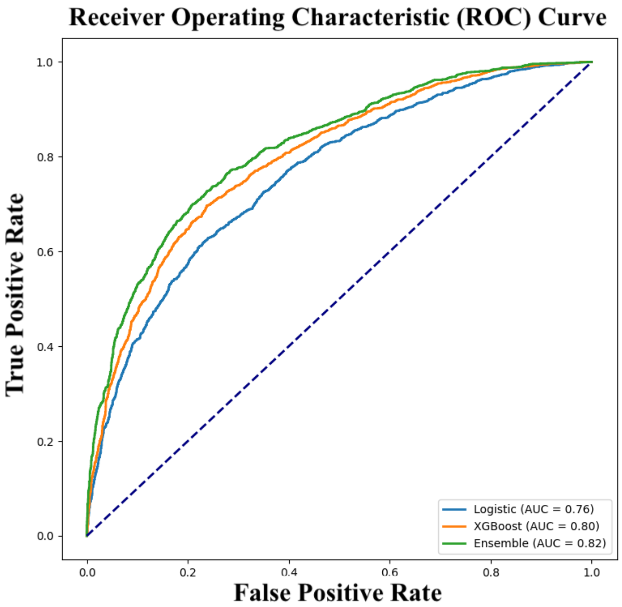

Figure 7: ROC Curves of each classifier

Our pipeline's best mean absolute error (MAE) of 1.80 shows a great improvement over the performance of the strongest neural baseline (MAE 3.92) reported by Harutyunyan et al. (2019). However, it should be noted that the multitask LSTM was learning four tasks at once and did not address highly skewed data that we investigated and focused on.

We observed consistent performance from Classification-Regression on different subsets of data during cross-validation, suggesting that the model is not overfitting onto any particular training data; additionally, weight values obtained were of varying reasonable magnitudes.

6 Fairness Analysis

We evaluate the ensemble classifier's true positive rate, false positive rate, true negative rate, false negative rate, and overall positive rate (TPR, FPR, TNR, FNR, and PR) for each sub-group separated by gender and race, presented in Table 4 to discuss algorithmic fairness. We also have the confusion matrix from the classification in Figure 6 for visualization.

Table 4: Prediction Rates by Gender and Race

Groups	TPR	FPR	TNR	FNR	PR
Female	0.577	0.186	0.814	0.423	0.344
Male	0.645	0.222	0.778	0.355	0.412
Asian	0.510	0.161	0.839	0.490	0.310
Black/African-American	0.587	0.171	0.829	0.413	0.340
Hispanic/Latino	0.534	0.161	0.839	0.466	0.295
White	0.618	0.214	0.786	0.382	0.387
Other/Unknown	0.675	0.214	0.786	0.325	0.431

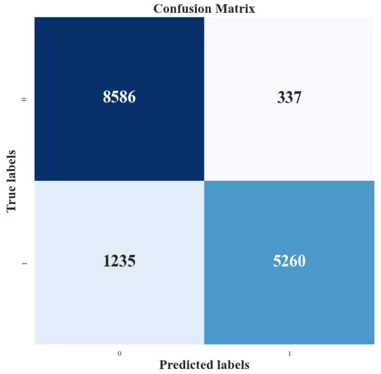

Figure 8: Classifier Confusion Matrix.

For reference, we use the formulas $TPR = \frac{TP}{TP+FN}$, $FPR = \frac{FP}{FP+TN}$, $TNR = \frac{TN}{TN+FP}$, $FNR = \frac{FN}{FN+TP}$, and $PR = \frac{TP+FP}{TP+FP+TN+FN}$. Demographic parity holds when PR is comparable among different sub-groups separated by a particular attribute. This means that our model is equally likely to predict a long stay for any given group. The PR is indeed pretty close across genders. There is more variation among racial sub-groups but it's not so significant. We also consider equalized odds demonstrated by equal TPR and FPR values. This means that our model performs equally well for any of the sub-groups given. As seen in the table, both TPR and FPR are comparable among the gender sub-groups and racial sub-groups. Another fairness metric is equalized opportunity, which measures how well the model performs on just examples with positive labels by only considering the TPR, so it is implied by equalized odds. Overall, the ensemble classifier aligns with the three fairness metrics considered.

7 Conclusion & Future Work

We presented a two-stage predictor for a patient's LoS in a hospital that outperforms common baselines and more complex models in this paper. We achieved this by recognizing that LoS distribution is highly skewed or unbalanced and addressed the issue using oversampling and SMOTE. Inspired by existing studies, we broke the task into a classification step using an ensemble of random forest and XGBoost, the best classifier found, and a regression step using XGBoost. It is reasonable that the ensemble gives the best result over any single model by aggregating different types of errors and potentially offering a more balanced bias-variance trade-off. Our classifier is also notable in its balanced accuracy which benefits the regression step by guiding examples based on model confidence, mitigating prediction errors for longer stays when we go through the pipeline. We also comprehensively examined the ensemble classifier's performance across different demographic sub-groups, assessing algorithmic fairness and ensuring that our model treats all patients equitably across genders and racial backgrounds. Finally, our model takes relatively minimal time and computation to train and run, making it accessible and cost-efficient, an important characteristic for a practical implementation.

For future work, we want to explore other ways of reducing prediction errors for long stays. Given more computation power, we may apply both more sophisticated statistical methods that address imbalanced data and more complex deep learning approaches that use additional features such as natural language, being pursued by state-of-the-art models such as van Aken et al. (2021).

8 Contributions

All authors worked on cleaning the data and literature review. Given overlap over each member's work during the collaborative process, the following describes the respective main contributions:

Pinlin [Calvin] Xu worked on researching and running experiments on ideas that did or did not work to improve upon the baselines (oversampling, dimensionality reduction, outlier removal, etc.), producing plots/graphs, and managing project progress, next ideas to implement, and assigning tasks. He is also responsible for much of the write-up leading up to 4.1 detailing the rationale behind adopting the methodology.

Miguel Fuentes contributed with most of the code infrastructure to run all experiments, cleaning and combining the datasets and creating the pipeline for the classification-regression model. He is also responsible for much of the experiments and results section, the model performance results, graphs, and all diagrams in the write-up.

Lisa Liu worked on cleaning up the dataset in feature selection and deriving the SMOTE algorithm. She is responsible for most of the mathematical derivations and the exposition of the write-up starting in section 4. She is also responsible for researching and analyzing the algorithmic fairness of the models.

References

Betty van Aken, Jens-Michalis Papaioannou, Manuel Mayrdorfer, Klemens Budde, Felix A. Gers, and Alexander Löser. 2021. Clinical outcome prediction from admission notes using self-supervised knowledge integration. *EACL*.

M.K. Bailey, A.J. Weiss, M.L. Barrett, and et al. 2019. Characteristics of 30-day all-cause hospital readmissions, 2010–2016. *Healthcare Cost and Utilization Project (HCUP) Statistical Brief*.

N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. 2002. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Mehak Gupta, Brennan Gallamoza, Nicolas Cutrona, Pranjal Dhakal, Raphael Poulain, and Rahmatollah Beheshti. 2022. Extensive data processing pipeline for mimic-iv. *Proceedings of machine learning research*.

Gaspard Harerimana, Jong Wook Kim, and Beakcheol Jang. 2021. A deep attention model to forecast the length of stay and the in-hospital mortality right on admission from icd codes and demographic data. *Journal of Biomedical Informatics*, 118:103778.

Hrayr Harutyunyan, Hrant Khachatrian, David Kale, Greg Ver Steeg, and Aram Galstyan. 2019. Multitask learning and benchmarking with clinical time series data. *Sci Data*.

Lars Hempel, Sina Sadeghi, and Toralf Kirsten. 2023. Prediction of intensive care unit length of stay in the mimic-iv dataset. *Applied Sciences*, 13(12).

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2023a. Mimic-iv (version 2.2). *PhysioNet*.

Alistair Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Li-wei Lehman, Leo Anthony Celi, and Roger Mark. 2023b. Mimic-iv, a freely accessible electronic health record dataset. *Sci Data*.

NLM. 1998. Medical subject headings: Age group.

Kieran Stone, Reyer Zwiggelaar, Phil Jones, and Neil Mac Parthaláin. 2022. A systematic review of the prediction of hospital length of stay: Towards a unified framework. *PLOS digital health*.

Yiwu Sun, Zhaoyi He, Jie Ren Ren, and Yifan Wu. 2023. Prediction model of in-hospital mortality in intensive care unit patients with cardiac arrest: a retrospective analysis of mimic -iv database based on machine learning. *BMC Anesthesiol*.

WHO. 1977. Icd-9 title & code cross reference list. *Manual of the International Classification of Diseases, Injuries, and Causes of Death, Ninth Revision*.

Hsiu Wu, Minn M Soe, Rebecca Konnor, Raymund Dantes, Kathryn Haass, Margaret A Dudeck, Cindy Gross, Denise Leaptrot, Mathew R P Sapiano, Katherine Allen-Bridson, Lauren Wattenmaker, Kelly Peterson, Kent Lemoine, Sheri Chernetsky, Jonathan Edwards, Daniel Pollock, and Andrea Benin. 2020. Hospital capacities and shortages of healthcare resources among us hospitals during the coronavirus disease 2019 (covid-19) pandemic. *National Healthcare Safety Network*.