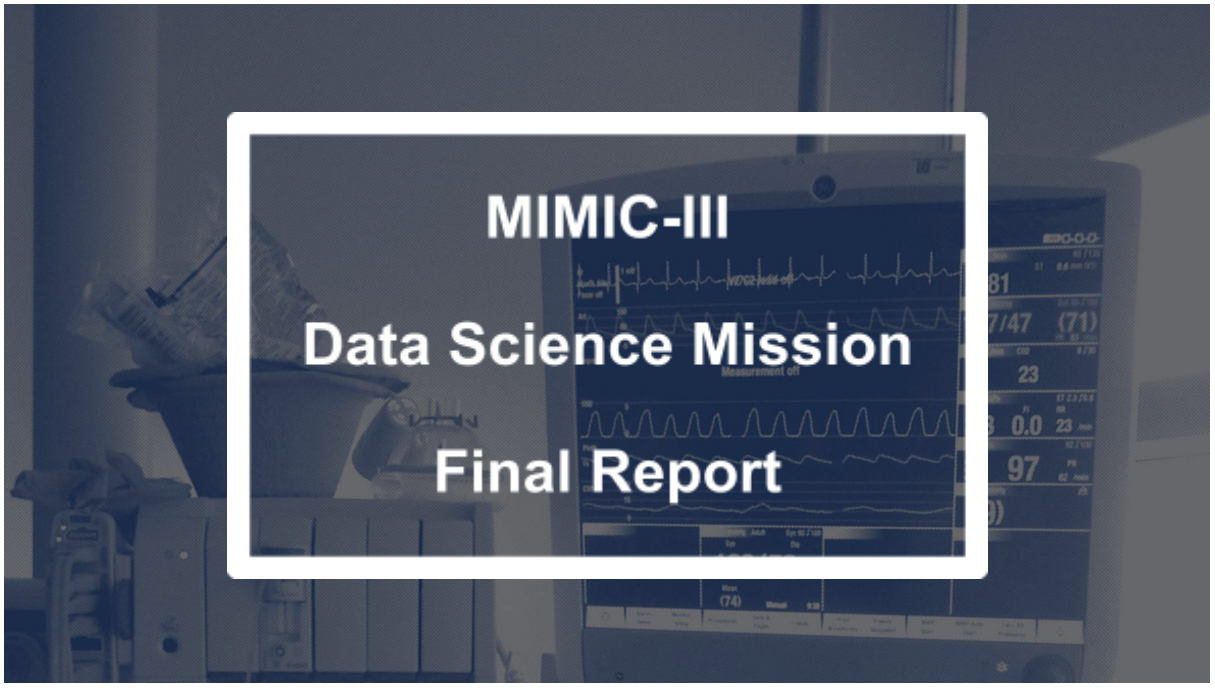


Team Members :

Sugumaran BALASUBRAMANIYAN, Trisha KUMAR, Qian LIU, Mingfei LI, Christophe GUIBOURD de LUZINAIS

Under the supervision of emlyon Professor Franck JAOTOMBO

Data Science and Artificial Intelligence - March 2023

Acknowledgements

First of all, we would like to express our deepest gratitude to our dedicated team for their exceptional work on the analysis of the MIMIC-III dataset.
The creativity, and expertise of each one have contributed immensely to this project.
We also want to express my sincere appreciation to our esteemed professor at emlyon, especially Mr. Franck JAOTOMBO, whose guidance, support, and invaluable insights have been instrumental in shaping our understanding and driving us towards excellence.
Together, we have made significant strides in advancing our knowledge in the data science field, and we were truly proud to be a part of such a remarkable team.

Table of contents

Introduction

Our mission

The goal of this project was to predict critical health outcomes using patient data and specifically, a fusion of different types of data : structured tabular and free text. For our project, we chose to focus on building a model that can be trained on a fusion of structured information of patients and clinical notes written by doctors, and ultimately, predict patient mortality.

Fusion models, while still fairly new in the field of data science, is becoming an increasingly popular topic for researchers. To put it simply, when humans make decisions, there are a multitude of factors that are taken into consideration when making a final decision. Similarly, if machines were to rely on one type of data, their potential to predict an output would be limited. Therefore, fusion models are valuable as they are able to integrate data from diverse sources and provide more accurate and reliable predictions than traditional models.

In the field of healthcare specifically, there are so many different types of data that are collected from different sources, however, the problem is that merging this type of data can be challenging due to their heterogeneity and complexity. When adopting new technologies, hospitals tend to remain cautious, as the two most important factors for them in a model would be performance and explainability. Fusion models offer a solution to this challenge by combining multiple data sources to create a more comprehensive overview of patient health.

Fusion models have the potential to revolutionize the healthcare industry by enabling more personalized and effective treatments. By leveraging the power of multiple data sources, fusion models can provide more accurate and reliable predictions, leading to better health outcomes for patients.

The Data

In order to achieve our goal, we used patient data from the The MIMIC-III or as it is referred to as Medical Information Mart for Intensive Care third iteration dataset, which is a comprehensive, publicly available collection of de-identified, critical care patient data gathered from the Beth Israel Deaconess Medical Center in Boston, Massachusetts between 2001 and 2012. This rich dataset, which encompasses 53,423 distinct critical ICU admissions, serves as a valuable resource for researchers and clinicians seeking to improve patient care, enhance clinical decision-making, and advance our understanding of critical care medicine.

The MIMIC-III dataset comprises various types of data, including demographics, vital signs, laboratory tests, medications, diagnoses, and procedure codes, among others. The integration of these diverse data sources enables researchers to explore complex relationships, develop predictive models, and validate new methodologies in the realm of healthcare.

As a result of its open access nature and the wealth of information it contains, the MIMIC-III dataset has become an indispensable tool for researchers worldwide, fostering collaboration and innovation in the critical care community.

Project Overview

The Massachusetts Institute of Technology (MIT) maintains the MIMIC-III dataset in a static repository. Every patient data has undergone a comprehensive identification process. Anybody seeking access to the MIMIC dataset must complete training on patient data safety, according to MIT.

Getting access to the data

To gain access to the MIMIC-III dataset, we needed to follow a few steps, which include completing a required training course on human research subjects.

First we created an account on the PhysioNet website (https://physionet.org/), then we had to complete a Collaborative Institutional Training Initiative (CITI) course on "Data or Specimens Only Research." The aim of this course covers the ethical aspects of using de-identified patient data for research purposes. Completing the course gave us a completion certificate as a PDF file.

Once the completion of the CITI training, we log back into the PhysioNet account, to the "Request Access" section in the "Databases" page. We had to complete the access request form, which includes providing name, contact details, research interests, and a brief description of how we intended to use the MIMIC-III dataset, then we upload the CITI training certificate PDF file.

After review and acceptance of the terms and conditions, we waited for approval.

Upon waiting for one week, we finally received an email confirmation granting us access to the dataset.

Once getting access, the data was received as 26 archives containing csv files, which correspond to the 26 tables in the MIMIC-III database, namely :
ADMISSIONS, CALLOUT, CAREGIVERS, CHARTEVENTS, CPTEVENT, D_CPT, D_ICD_DIAGNOSES, D_ICD_PROCEDURES, D_ITEMS, D_LABITEMS, DATETIMEEVENTS, DIAGNOSES_ICD, DRGCODES, ICUSTAYS, INPUTEVENTS_CV, INPUTEVENTS_MV, LABEVENTS, MICROBIOLOGYEVENTS, NOTEEVENTS, OUTPUTEVENTS, PATIENTS, PRESCRIPTIONS, PROCEDUREEVENTS_MV, PROCEDURES_ICD, SERVICES, TRANSFERS

Theses files containing the following informations :

Demographics: Patient age, gender, ethnicity, and admission and discharge dates, among other details.
Vital signs: Continuous and intermittent measurements of heart rate, blood pressure, respiratory rate, temperature, and oxygen saturation, among others.
Laboratory tests: Results from various laboratory tests, including blood chemistry, hematology, microbiology, and coagulation profiles.
Medications: Information on medications administered to patients during their ICU stay, including drug names, dosages, routes of administration, and start and end times.
Diagnoses: ICD-9 codes for diagnoses assigned to patients during their hospital stay.
Procedures: ICD-9 procedure codes representing surgical and non-surgical interventions performed on patients.
Clinical notes: Free-text notes authored by healthcare providers, including nursing notes, physician progress notes, radiology reports, and discharge summaries. These notes have been de-identified and scrubbed of personal health information (PHI) to maintain patient privacy.
Imaging data: Although the MIMIC-III dataset does not include actual radiographic images, it contains textual descriptions and findings from radiology reports.
Fluid balance: Data related to fluid input and output, including intravenous fluids, oral intake, and urine output, among other measurements.
Severity scores: Scores calculated using established severity-of-illness scoring systems such as SAPS-II (Simplified Acute Physiology Score II) and SOFA (Sequential Organ Failure Assessment).

Project Scope and Functionality

Data Host and Processing

One of the first goals for this project was to construct a database that could be searched to carry out the activities of data exploration, and other types of data mining due to the complexity of the MIMIC-III database and the enormous size of the source data (nearly 40 gigabytes in total). It was necessary to use a powerful infrastructure in order to complete the necessary computations.

Our personal computers systems were struggling to handle the size and complexity of the Mimic-III dataset.Thus we chose the Amazon S3 servers solution which provides us with the storage capacity to easily store the entire dataset and the additional data generated during our research.

In addition we also use Amazon Athena service as it is designed to handle big data, it was the ideal choice for querying and analyzing large datasets like MIMIC-III directly on data stored in Amazon S3. This flexibility has made it easy to analyze the MIMIC-III dataset on-demand, without incurring the overhead of managing a full-fledged local database system. Storing the MIMIC-III dataset on Amazon Servers has enabled seamless collaboration among every team member to access the dataset and run queries from anywhere and any devices.

Amazon robust data security features such as encryption, access control policies, and versioning, has ensured that the data was protected and compliant with regulatory requirements. One other benefit was that AWS services are already compliant with the Health Insurance Portability and Accountability Act (HIPAA), which is crucial when handling sensitive healthcare data.

Data Import

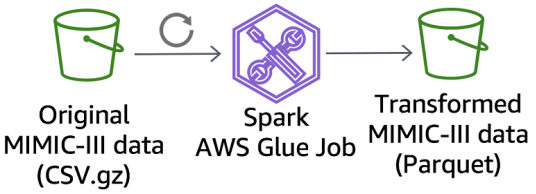
To ensure efficiency we decided to convert the original csv files into Parquet.
Parquet is a columnar storage file format that is optimized for use with big data processing frameworks like Spark, providing efficient compression and encoding techniques for faster query performance which was a strong beneficial argument.

We first uploaded the MIMIC-III dataset CSV files to an Amazon S3 bucket. Ensuring that all files were organized into their appropriate folders for easier processing.

We then set up an AWS Glue Data Catalog to serve as a central metadata repository for all the table data. This catalog stores table definitions and other metadata required for processing the MIMIC-III dataset. In the AWS Glue Console, we created a new Crawler to automatically discover and classify the MIMIC-III CSV files in the S3 bucket. The Crawler was pointed to the S3 bucket containing the CSV files, and configured to store the discovered table schema in the Glue Data Catalog.

The Crawler was then executed to create table definitions for each MIMIC-III CSV file in the Glue Data
Catalog. The Crawler was able to infer the schema of each CSV file, automatically detecting column names and data types.

The next step was to create a new Glue job using Apache Spark. The Glue job uses Apache Spark to read the MIMIC-III CSV files, convert them to Parquet format, and write the output back to a new S3 bucket.



Finally we check the destination S3 bucket to ensure the Parquet files were created successfully. We also created another Glue Crawler to catalog the newly created Parquet files, making them available for further analysis in Amazon Athena and Amazon SageMaker Notebook services.

Data cleaning

The Admission table

The data cleaning of the admission table in the MIMIC-III dataset starts with filtering out records that do not have chart events data. This is done by checking the 'HAS_CHARTEVENTS_DATA' column and retaining only rows with a value of 1.

Next, the admission and discharge time columns were converted to datetime format, and a new column, 'ADMISSIONS_DURATION_DAYS,' was created to store the duration of the hospital stay in days.

The dataset was then checked for missing values (NA values). The 'EDREGTIME' and 'EDOUTTIME' columns, which contain the emergency department registration and departure times, had missing values filled with zeros. The duration of the patient's stay in the emergency department is then calculated and stored in a new column, 'ADMISSIONS_ED_DURATION_HOURS.'

Several unwanted columns are dropped from the dataset, including 'ROW_ID', 'DISCHTIME', 'DEATHTIME', 'LANGUAGE', 'EDREGTIME', 'EDOUTTIME', and 'HAS_CHARTEVENTS_DATA.' Missing values in the 'RELIGION' and 'MARITAL_STATUS' columns were replaced with appropriate default values. Rows containing missing values in any other columns were also dropped.

The 'RELIGION' column was further cleaned by replacing the 'UNOBTAINABLE' category with 'RELIGION_NOT_SPECIFIED.' The 'MARITAL_STATUS' column is cleaned by replacing the 'UNKNOWN (DEFAULT)' and 'LIFE PARTNER' categories with 'MARITIAL_STATUS_NOT_SPECIFIED' and 'MARRIED,' respectively.

The 'ETHNICITY' column was standardized by replacing 'UNKNOWN/NOT SPECIFIED', 'PATIENT DECLINED TO ANSWER', and 'UNABLE TO OBTAIN' categories with 'ETHNICITY_NOT_SPECIFIED.'

The 'DIAGNOSIS' column was also standardized by consolidating various categories related to coronary artery bypass graft into a single category named 'CORONARY ARTERY BY PASS GRAFT,' along with several other replacements to ensure consistency in diagnosis categories.

Finally, the 'ADMITTIME' column was converted to datetime format using the '%Y-%m-%d' format. This cleaned admissions table can now be used for further analysis or modeling.

The Chartevents table

The cleaning of the Chartevents table in the MIMIC-III dataset started with importing the required columns from the CHARTEVENTS table for multiple variables, such as arterial blood pressure systolic (itemid: 220050), arterial blood pressure diastolic (itemid: 8368), respiratory rate (itemid: 220210), temperature (itemid: 223762), O2 saturation (itemid: 220277), and heart rate (itemid: 220045).

We define a function “compute_stats” to take the DataFrame and a prefix as input arguments and to compute the mean, min, and max of the 'valuenum' column after grouping by 'subject_id'.

Dealing with outliers: For each variable, we created a boxplot to visualize and identify the outliers values. Then, we removed the outliers using the remove_outliers custom function.

Handling missing values: The code checks for NaN values using the na_percentage function and dropping the rows containing NaN values using the .dropna method considering we have enough data to work with.

After cleaning the data, we computed the mean, min, and max for each variable using the compute_stats function. This results in a DataFrame with columns ['subject_id', '{prefix}_mean', '{prefix}_min', '{prefix}_max'] for each variable.

At the end of this process, the cleaned data for each variable was stored in separate DataFrames (grouped_abpsys_chartevents, grouped_abpdiastolic_chartevents, grouped_resp_rate_chartevents, grouped_temperature_chartevents, grouped_spO2_chartevents, and grouped_hr_chartevents).

The chartevents dataframes were then merged iteratively using outer join on the subject_id field to create a combined dataframe named “merged_chartevents”.

The merged_chartevents was then cleaned by dropping columns with a high percentage of missing values and dropping rows with missing values in specified columns.

The D_Item table

To clean the D_Item table, we once again use the na_percentage function to calculate the percentage of missing values (NAs) for each column. The output showed us that columns 'ABBREVIATION', 'CATEGORY', 'UNITNAME', 'PARAM_TYPE', and 'CONCEPTID' were having a significant amount of missing data.

We proceeded to drop all unnecessary columns or columns with a high percentage of missing values such as 'ROW_ID', 'ABBREVIATION', 'DBSOURCE', 'LINKSTO', 'CATEGORY', 'UNITNAME', 'PARAM_TYPE', and 'CONCEPTID'

We finally used the .dropna function to drop any rows with missing values from the remaining columns in the d_items dataframe.

The Icustays table

As for the previous tables we made use of the ‘na_percentage’ and ‘remove_outliers’ to identify and remove the NaN. The output shows that columns 'OUTTIME' and 'LOS' had a small percentage of missing values (0.02%).

The cleaned icustays table has unnecessary columns removed and rows with missing values or outliers dropped.

The labevents table

For this table we first use our functions to get the na_percentage and then use drop duplicates to remove any duplicate rows from the labevents table using the drop_duplicates() function.

We dropped the unwanted columns: "ROW_ID", "HADM_ID", 'CHARTTIME', 'VALUENUM', "VALUEUOM", and "FLAG".

Then we merged the labevents and d_labitems tables on the 'ITEMID' column, creating a new DataFrame called labevents_merged.

We replaced specific non-numeric values in the VALUE column with numeric values using the replace() function. Then converting the VALUE column to numeric, replacing any remaining non-numeric values with NaN using the pd.to_numeric() function with errors='coerce and deleting any other missing values in the VALUE column using the dropna() function.

The labevents_merged DataFrame was then grouped by 'SUBJECT_ID' and 'ITEMID', and the minimum, maximum, and average of the VALUE column were computed for each group. The first LABEL and FLUID values were also retained.

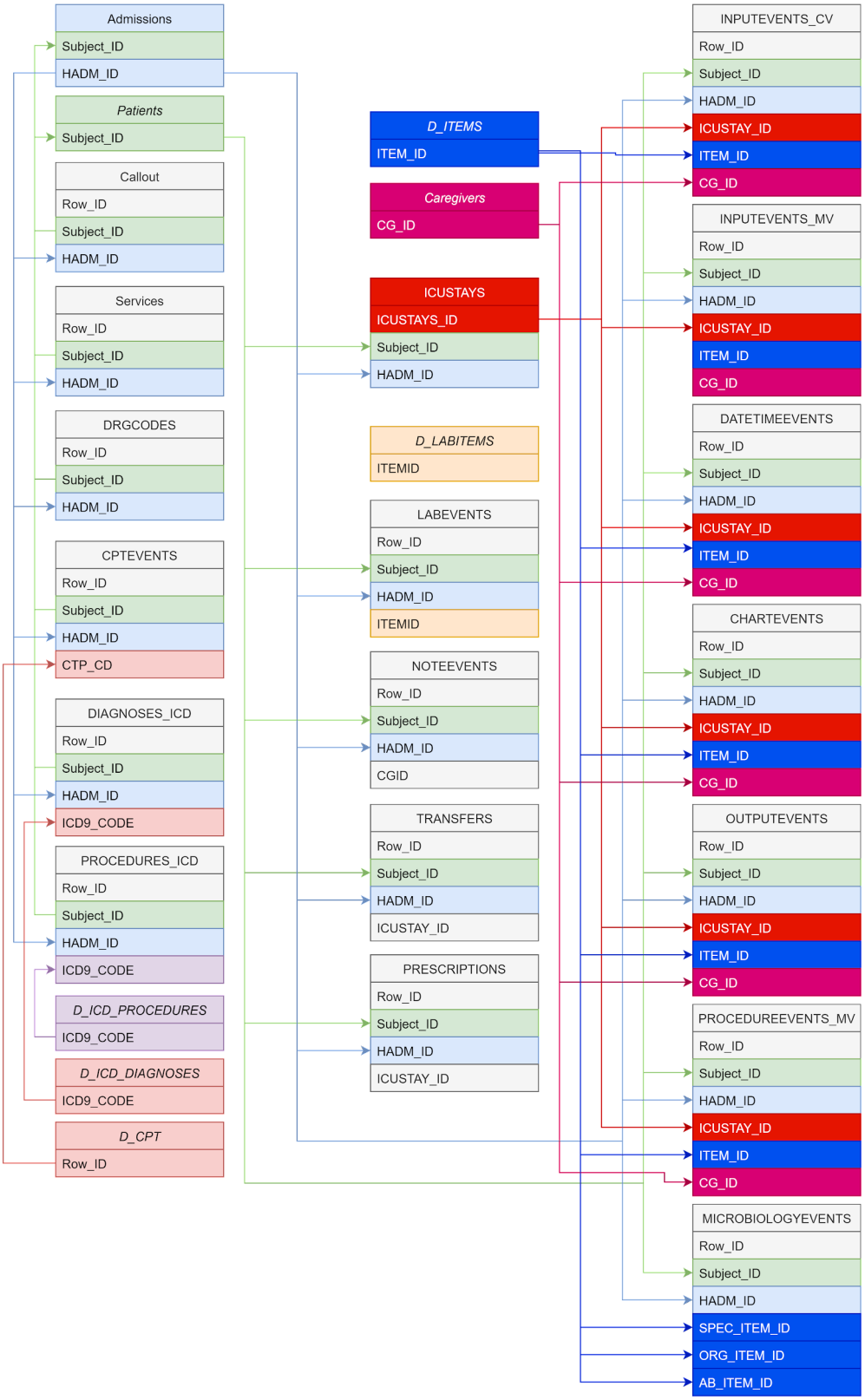
The resulting DataFrame's column names are flattened, and the index is reset to make 'SUBJECT_ID' and 'ITEMID' regular columns.We created a copy of the grouped DataFrame called labevents_merged_stats and drops the "ITEMID" column from the labevents_merged_stats DataFrame.

The columns of the labevents_merged_stats DataFrame are renamed to more descriptive names, such as 'Lab_VALUE_min', 'Lab_VALUE_max', 'Lab_VALUE_mean', 'Lab_LABEL_first', and 'Lab_FLUID_first'.

The resulting labevents_merged_stats DataFrame has the shape (1677746, 6), indicating it has 1,677,746 rows and 6 columns after the data cleaning process

Exploratory Data Analysis

Conceptual Data Model



Data merging

In our process, we successfully merged the admissions, ICU, lab events, and chart events tables to create a comprehensive dataset for further examination. First, we merged the admissions and patients tables using a 'left' join, based on the 'SUBJECT_ID' column, resulting in the 'admissions_patients' DataFrame. Next, we combined this DataFrame with the 'icustays' table using a 'left' join on the 'HADM_ID' column, creating the 'adm_pat_icu' DataFrame. To account for missing ICU stay lengths, we filled any missing values in the 'LOS_icustays' column with zeros.

Subsequently, we integrated the 'labevents_clean' table by merging it with the 'adm_pat_icu' DataFrame using a 'left' join on the 'HADM_ID' column, generating the 'adm_pat_icu_lab' DataFrame. To merge the 'chartevents' table, we first renamed its 'hadm_id' column to 'HADM_ID' for consistency. Finally, we combined the 'adm_pat_icu_lab' and 'chartevents' DataFrames using a 'left' join on the 'HADM_ID' column, resulting in the 'adm_pat_icu_lab_chart' DataFrame.

After merging all the relevant tables, we saved the final, comprehensive dataset as a CSV file for further processing. This approach allowed us to create a unified dataset containing a wealth of information on patient demographics, admissions, ICU stays, laboratory events, and chart events.

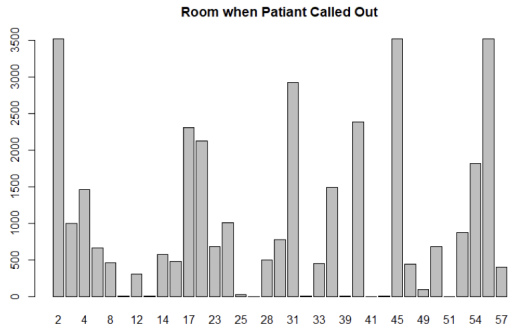
In order to prepare the new dataset for analysis, we performed several data processing steps, as outlined in the provided code. First, we converted the 'ADMITTIME' and 'DOB' columns to datetime format and extracted the respective years. Next, we calculated the age of each patient by subtracting the 'DOB' from the 'ADMITTIME' and created a new column, 'AGE', to store this information. We then calculated the minimum, maximum, mean, and standard deviation of patient ages to get a sense of the distribution

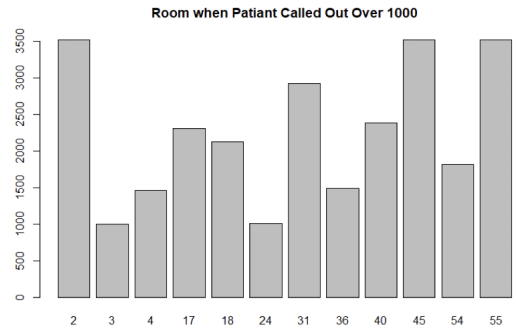
To streamline the dataset, we dropped the 'ADMITTIME' and 'DOB' columns and removed any outliers in the 'AGE' column using a custom 'remove_outliers' function. After removing any rows containing missing values using the 'dropna' method, we were left with a dataset of 18,151 rows and 35 columns. To further refine the dataset, we removed several unnecessary columns, including 'Unnamed: 0', 'SUBJECT_ID_x', 'SUBJECT_ID_y', 'HADM_ID', 'ROW_ID', 'DISCHARGE_LOCATION', and 'EXPIRE_FLAG'. We then removed any duplicate rows from the dataset to ensure data integrity.

Lastly, we replaced any negative values in the 'ADMISSIONS_DURATION_DAYS' and 'ADMISSIONS_ED_DURATION_HOURS' columns with zeros, as negative durations are not meaningful in this context. Once the dataset was cleaned and preprocessed, we saved the final version as a Parquet file named 'Cleaned_Dataset.parquet' for efficient storage and further analysis, providing a solid foundation for our subsequent computing.

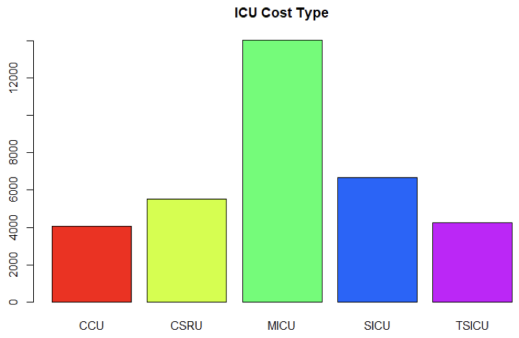
Data Analysis

EXPLORATORY DATA ANALYSIS



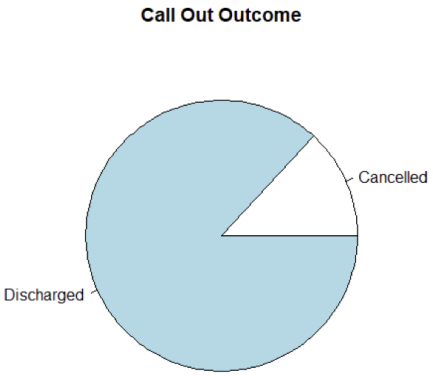


In these charts, we can understand that rooms 2,3,4,17,18,24,31,36,40,46,54,55 are several rooms that have patients called out frequently. Room 2, 45, and 55 are the top 3 frequently used. This can be used to better arrange patient rooms.



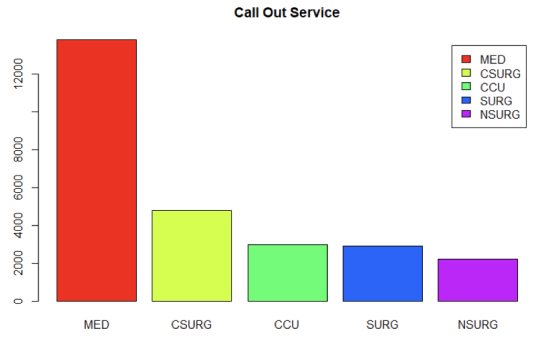
CCU – Coronary Care Unit; CSRU - Cardiac Surgery Recovery Unit; MICU - Medical Intensive Care Unit; SICU - Surgical Intensive Care Unit; TSICU - Trauma Surgical Intensive Care Unit

From this chart we can understand there are five ICU costs in total and most of the ICU cost is MICU.

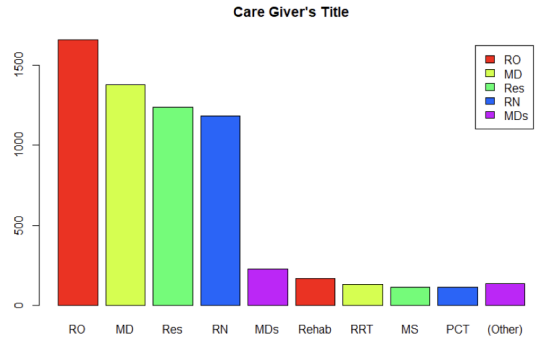


From this pie chart, we can see that most of the time, the outcome of a call out will be discharged. Only 10% will be canceled.

We can focus on those canceled rows to understand why they are canceled.



The best service provided for call out is MED, then CSURG, then CCU, then SURG, and NSURG at last. It might be important to understand why most of them are MED, and why they need CSURG, CCU, SURG, or NSURG if not MED



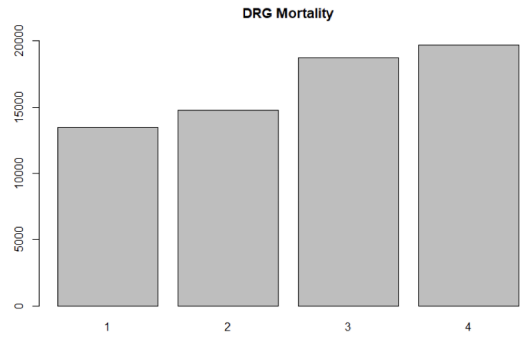
Here is the situation of the care giver's title. Most of them are RO., MD., Res., And RN. It might be important to understand the result for different groups of caregivers.

```
> summary(x4)
      0       1    NA's
2686689     231 1799017
> summary(x5)
      0       1    NA's
2686325     595 1799017
>
```

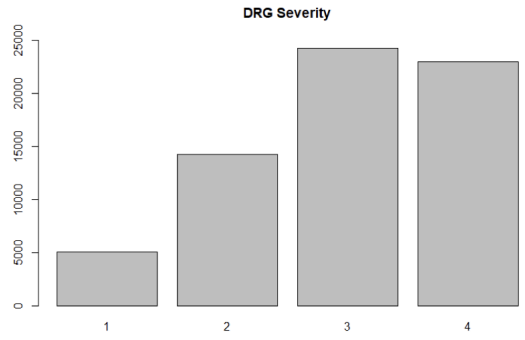
x4 – warning, x5 – error

```
> summary(DATETIMEEVENTS$STOPPED)
   D/C'd NotStopd     NA's
   32623  1766394  2686920
```

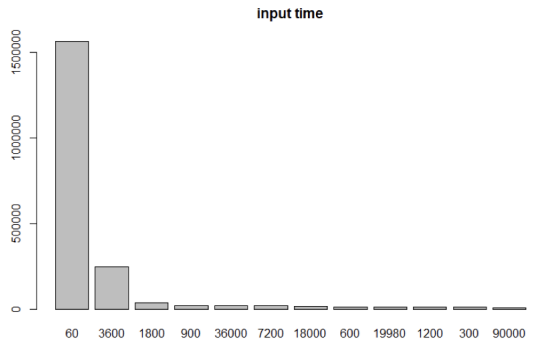
#If a row has a warning, then it has an error value. Otherwise, it will have a stopped value.
It might be important to understand why this happened.



The level of DRG (Diagnosis Related Groups) mortality. The Diagnosis Related Groups (DRGs) are a patient classification scheme which provides a means of relating the type of patients a hospital treats (i.e., its case mix) to the costs incurred by the hospital. This might be an important factor to predict mortality.
Higher mortality rates for a particular DRG may indicate issues with the quality of care provided or with the patient population being treated.



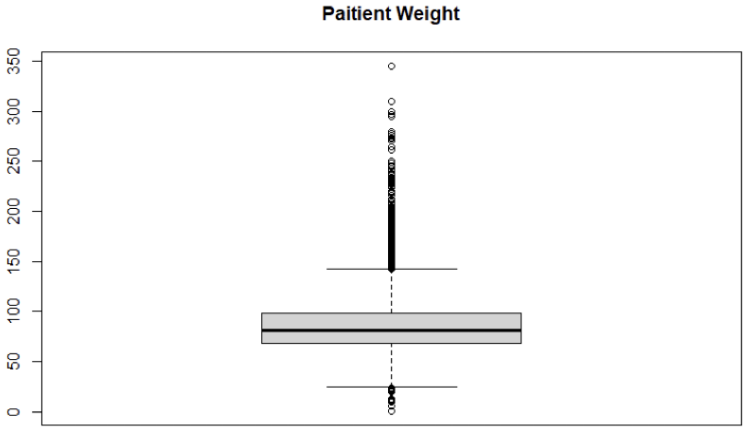
The level of DRG severity. This might also be an important factor to predict mortality. DRG severity refers to the level of illness or severity of a patient's medical condition within a specific DRG.
DRG severity is important because it can impact reimbursement rates, as well as the resources and level of care required to treat patients. Hospitals and healthcare providers use DRG severity to ensure that patients receive appropriate care and resources based on their level of illness and severity.
P.S. There are NAs in both mortality and severity. Both of those NAs might be interesting in terms of why. Some DRGs may be assigned to patients who are not at risk for mortality. Some healthcare systems may choose not to report DRG Mortality for certain DRGs, either because of data limitations or for confidentiality reasons.



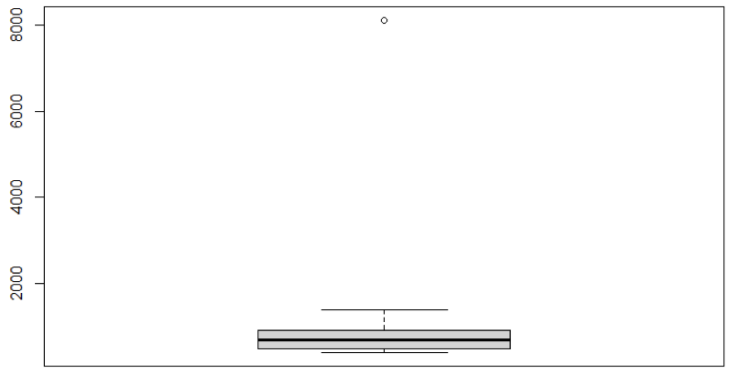
From this chart, we can know that the input time will usually be 60. Some of them will be 3600. The longer input, the less case it is.

```
> summary(INPUTEVENTS_MV$PATIENTWEIGHT)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00   68.40   81.40   85.56   98.00 8106.00
```

Here is the weight box plot for those under 400



Here is the weight box plot for those above 400.



The patient weight can be used to calculate the BMI

This is a box plot explaining input amount in mL. organic matter Name :

STAPH AUREUS COAG +	ESCHERICHIA COLI	STAPHYLOCOCCUS, COAGULASE NEGATIVE
63947	60133	32777
KLEBSIELLA PNEUMONIAE	PSEUDOMONAS AERUGINOSA	ENTEROCOCCUS SP.
30628	28926	16429
YEAST	PROTEUS MIRABILIS	ENTEROBACTER CLOACAE
14182	9605	8709
NA's		
303710		

Model Building for Mortality Prediction

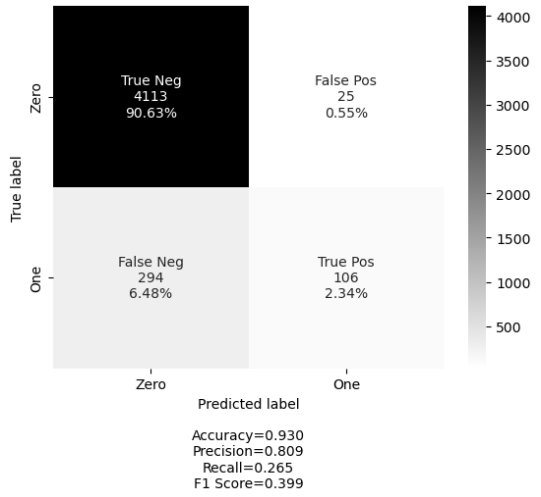
Structured data

Our structured dataset was ultimately an amalgamation of chosen variables that existed in multiple tables in the MIMIC 3 dataset. This process required extensive data wrangling in order to extract the relevant variables and create a cohesive feature set. Our approach involved merging data from the admissions, patients, and ICU stays tables. We also collected vital sign measurements and lab test results from the Labevents and Chartevents tables and calculated summary statistics such as the minimum, maximum, and mean values for each patient. To include age as a variable, we calculated the difference between the patient's date of birth and admission date. Additionally, we used one-hot encoding to handle categorical variables and converted our columns into numerical ones. After this process, we ended up with a dataset containing 18151 rows and 5725 columns, with the target column being "HOSPITAL_EXPIRE_FLAG".

To build the model, we first split the data and utilized stratified sampling to divide the dataset into training and testing sets, with a 75:25 split. This stratified approach ensured that the distribution of the target variable remained consistent across both sets. After splitting the data, we printed the shapes of the training and testing sets, which confirmed that our training set contained 13,613 samples and our testing set had 4,538 samples. To develop the prediction model, we chose a Random Forest Classifier and set the number of trees ('n_estimators') to 10. In this case, our Mortality Prediction Model achieved an accuracy of 92.55%.

To perform feature selection on our dataset, we utilized the feature importances provided by the trained Random Forest Classifier. We trained several machine learning models on our dataset and evaluated their performance using various metrics. Our initial model was a Random Forest Classifier, which achieved an accuracy score of 92.55%. However, upon inspecting the feature importance, we discovered data leakage in the "ADMISSION_DURATION_DAYS" and "LOS_icustays" columns, which contained information on patient expiration. We fixed this issue by replacing the admission duration values that were “-1” (for patients who expired) with a zero. We also removed the outliers in the length of stay in the ICU, by using the interquartile range. Following the changes, our accuracy score decreased slightly but our F1 score remained the same. We then used feature selection to ensure that our model focused on the most relevant features, we decided to select only the features with an importance score greater than 0.005, resulting in 30 highly important features. We aimed to simplify our model and reduce the risk of overfitting while still maintaining its predictive power.

After identifying the most important features (those with an importance score greater than 0.008) from the previous model, we built a new Mortality Prediction Model using only these 12 selected features. The Random Forest Classifier was retrained on the reduced feature set, and the performance of the new model was evaluated using accuracy as the metric. The accuracy of the model with the selected features was 92.97%, which is slightly higher than the previous model's accuracy of 92.55%. This improvement indicates that the selected features were indeed relevant to the model's predictions and that the model's performance was not negatively impacted by reducing the number of features. We generated a confusion matrix, which showed that the model correctly classified 4123 true negatives and 106 true positives, while making 25 false positive and 294 false negative errors.



In order to compare the performance of different classification methods, we built a Mortality Prediction Model using various classifiers. The classifiers used include Generalized Linear Models (Logistic Regression and Ridge Classifier), a Decision Tree, k-Nearest Neighbors, Naive Bayes, Ensemble Methods (Gradient Boosting and Random Forest), and a Neural Network (Multi-Layer Perceptron).Each classifier was fitted to the training data, and predictions were made on both the training and testing sets. The models were evaluated using various metrics such as accuracy, recall, ROC AUC score, F1 score, and Matthews correlation coefficient.

Based on the ROC AUC score, the Gradient Boosting Classifier has the highest performance (0.841) followed by the Random Forest Classifier (0.838). It's important to consider other metrics as well, such as accuracy, recall, F1 score, and Matthews correlation coefficient, to make a comprehensive evaluation of the models' performances. In this case, the Gradient Boosting Classifier and Random Forest Classifier appear to have the best overall performance among the various classification methods used. The random forest model had a better MCC score and F1 score on the test data as well as the highest training accuracy of 0.974, but a test accuracy of 0.911, indicating that the model was overfitting. While the random forest model had better metrics compared to gradient boosting, we want to avoid a model that is overfitting

	Accuracy_train	Accuracy_test	Recall_train	Recall_test	ROC_AUC_test	F1_test	MCC_test
gradient_boosting	0.936	0.923	0.330	0.253	0.841	0.366	0.378
random_forest	0.974	0.911	0.918	0.407	0.838	0.446	0.400
logistic	0.919	0.918	0.130	0.130	0.776	0.219	0.277
naive_bayes	0.884	0.879	0.325	0.305	0.769	0.308	0.241
mlp	0.916	0.914	0.097	0.090	0.765	0.156	0.204
knn	0.926	0.917	0.219	0.165	0.739	0.260	0.290
decision_tree	0.932	0.914	0.274	0.170	0.736	0.259	0.271
ridge	0.733	0.721	0.701	0.695	nan	0.305	0.256

SHAP

By using SHAP (SHapley Additive exPlanations) to analyze our model, we gained further insights into the contributions of individual features towards predicting the output. The SHAP values revealed that Age was an important predictor in our model and indicated that younger patients had a higher likelihood of survival compared to the older patients. Moreover, we were able to identify specific vital signs that strongly correlated with survival or expiration. SHAP provided us with a deeper understanding of our model's inner workings, and would be especially valuable to hospitals, to understand in more detail how the features in the model are behaving.

Model with imputation

In our original dataset, we observed that some of the features had missing values which were dropped prior to model building instead of being imputed. To account for this, we built a new model using imputed values for the missing data. We experimented with two imputation methods: SimpleImputer and KNN Imputer. After imputation, we found that the model's accuracy score decreased compared to the original model that dropped the null values. Specifically, SimpleImputer showed marginally better performance compared to KNN Imputer. Since we already had a large size of data, we realized it would be more computationally efficient to simply drop values, rather than impute them.

Data Fusion

Unstructured data (Clinical Notes)

The objective of our project was to investigate the potential of combining unstructured data, such as clinical notes, with structured data for predicting patient mortality. Clinical notes written by doctors contain valuable information that is often not utilized in data analysis. Therefore, we utilized natural language processing and machine learning techniques to process and analyze the clinical notes from the Noteevents table, specifically the "TEXT" column. The pipeline to manage this text column included text preprocessing and feature extraction using Word2Vec.

Text Preprocessing

The preprocessing functions used to clean the text involved several steps. Firstly, the text was transformed to lowercase and new line characters and dates/times were removed. Stopwords were removed using the standard NLTK list, however we extended this list to include medical-specific stopwords that we noticed had a high occurrence in the notes. The Gensim library was then used for additional preprocessing, including punctuation removal and tokenization. Bigram and trigram models were applied to combine frequently co-occurring words into single terms, further reducing the dimensionality of the data. Finally, the resulting corpus was created using a document-term matrix with an id2word mapping, which maps words in the corpus to unique integer IDs. The id2word mapping is useful for flexible data manipulation, such as removing rare words or selecting specific words to include in the model. The final output of the function was a cleaned, tokenized list of words, along with the HADM_ID, SUBJECT_ID and HOSPITAL_EXPIRE_FLAG columns.

Language Model

After exploring a few language models, we chose Word2Vec as it is an effective language model that is often used for text classification, semantic similarity, and information retrieval. The model works by capturing the semantic meaning of words by analyzing the co-occurrence of words within the corpus and generating dense vector representations (embeddings) of words in a high-dimensional space. Therefore, we found it particularly useful for this project, as clinical notes are known to contain complex medical terminology and jargon, and Word2Vec was ideally capturing the nuanced meaning of these terms, when predicting mortality of a patient.

We did evaluate other language models for this project. TF-IDF was considered due to its simplicity and interpretability, however since it focuses on term frequency, this model can't capture the semantic meaning of words as effectively. We found that since our text data was so large, using TF-IDF resulted in a high-dimensional sparse matrix, which was difficult to work with and more computationally expensive. Furthermore, we also looked at Doc2Vec and LDA (Latent Dirichlet Allocation), which are our more complex models. While these models are more suitable for large datasets, they did require more computation and we found them to be slightly less interpretable. LDA generates topic distributions based on the frequency of words in the document and it can be challenging to identify how specific words contribute to each topic. Doc2Vec generates document embeddings by training a neural network which can be difficult to interpret without extensive knowledge of the underlying model architecture.

Word2Vec

To generate embeddings for our text data, we used the Gensim library to train a Word2Vec model. Word2Vec generates vector representations of words based on their context in a corpus. We set the 'vector_size' parameter to 100, so that each word was represented by a 100-dimensional vector. The 'window' parameter was set to 5, which means that Word2Vec considers the five words to the left and right of the target word as its context. The 'min_count' parameter was set to 5, which means that any word that appears less than five times in the corpus is ignored. Finally, the 'workers' parameter was set to 4, which enables Word2Vec to use four CPU cores for parallel processing.

To apply the trained Word2Vec model to our preprocessed text data, we created a function that iterates through the tokenized text and for each token that exists in the Word2Vec model vocabulary, it adds the relevant embedding to a list. If no embeddings exist for a given token, a zero vector is used as a placeholder. We then took the mean of all token embeddings and added a new column to the data frame for each feature in the embedding. This resulted in a data frame with all the new embedding columns.

Before merging this data with the structured data, we wanted to try to predict mortality using just the text data. We trained multiple classification models using the text data embeddings and evaluated their performance.

Results

	Accuracy_train	Accuracy_test	Recall_train	Recall_test	ROC_AUC_test	F1_test	MCC_test
logistic	0.969	0.967	0.710	0.722	0.980	0.772	0.757
mlp	0.967	0.965	0.728	0.725	0.975	0.759	0.741
gradient_boosting	0.980	0.954	0.760	0.546	0.961	0.644	0.631
random_forest	0.979	0.942	1.000	0.630	0.946	0.624	0.592
knn	0.952	0.941	0.429	0.309	0.869	0.443	0.470
naive_bayes	0.806	0.801	0.703	0.738	0.856	0.362	0.341
decision_tree	0.958	0.926	0.611	0.410	0.824	0.457	0.421
ridge	0.905	0.899	0.955	0.944	nan	0.589	0.596

The table provides the performance metrics of different machine learning models for predicting mortality using word embeddings from clinical notes. From the models evaluated, we can see Logistic regression and multilayer perceptron models performed the best overall on both training and test sets specifically looking at the accuracy, roc_auc and Matthew correlation coefficient. K-nearest neighbors and ridge regression models had relatively low scores overall, while the naive Bayes model had lower accuracy but higher recall scores, suggesting it is better at identifying the positive class. The decision tree model had overfitting issues, with high accuracy on the training set but much lower accuracy and recall scores on the test set.

Fusion Model

Merging Data

To combine the unstructured text data with the structured data, we used the "HADM_ID" as the common key to join the two datasets. We then merged the datasets to create a single, comprehensive dataset that contains both structured and unstructured data. The resulting dataset had a total of 120 columns and 18027 rows, with the unstructured text data represented as word embeddings in multiple columns, along with the selected (most important) features that we chose from our structured dataset.

After merging the datasets, we trained fusion classification models on this merged dataset on the combined information. By leveraging both types of data, we were able to improve the predictive power of our models and gain new insights into the data.

Results

	Accuracy_train	Accuracy_test	Recall_train	Recall_test	ROC_AUC_test	F1_test	MCC_test
gradient_boosting	0.982	0.964	0.813	0.642	0.979	0.744	0.736
mlp	0.966	0.965	0.733	0.720	0.976	0.772	0.755
random_forest	0.980	0.947	1.000	0.663	0.956	0.671	0.642
logistic	0.937	0.936	0.359	0.358	0.906	0.480	0.482
naive_bayes	0.811	0.815	0.732	0.685	0.863	0.379	0.342
decision_tree	0.956	0.933	0.579	0.456	0.828	0.526	0.498
knn	0.928	0.921	0.164	0.108	0.760	0.183	0.230
ridge	0.905	0.913	0.964	0.949	nan	0.644	0.643

Above, are the results of the fusion models which show that gradient boosting, MLP, and random forest had the highest accuracy on the test set, with values ranging from 0.947 to 0.965. These models also had relatively high recall values on the test set, which indicates that they were able to correctly identify a significant number of positive cases (patients who died).

In comparison, the results of the models trained on unstructured data showed that logistic regression and MLP had the best performance and these models also had relatively high recall values. However, the models trained on only structured data performed worse in comparison, with the highest accuracy on the test set ranging from 0.911 to 0.923. These models had relatively low recall values on the test set, which indicates that they were not able to correctly identify as many positive cases as the fusion models were able to.

Furthermore, since we were working with an imbalance dataset, we wanted to prioritize the ROC AUC and MCC score, rather than just looking at model accuracy. ROC AUC is particularly useful when working with imbalanced classes, as it is able to measure the ability of a classification model to distinguish between positive and negative classes and MCC is a metric that takes into account both true positive and true negative rates and is able to capture the quality of the model's predictions regardless of the class distribution. From our fusion models, Gradient boosting and MLP both had strong scores for these two metrics, indicating these models performed best, despite our data imbalance. As well as this, fusion models may also help reduce the risk of overfitting, as they have more data available for training and are less likely to memorize the noise in the data.

The results indicate that the fusion classification models outperformed the models trained on only one type of data. This is likely as fusion models can leverage the strengths and insights available in both types of data. The structured data, such as lab results and vital signs, provides a standardized, quantitative representation of patient health, while the clinical notes contain detailed insights that cannot be easily captured in structured data. By combining these two types of data, the fusion models can better capture the complexity of patient health and improve their predictive performance.

The current state of imbalance in the Dataset

The MIMIC-III dataset has a diverse distribution of data resulting from its size, but the number of records as big as it is doesn't represent a balanced set of individual characteristic in every groups of patients, first, the age distribution, it is important to note that patients aged below 18 are not included. The dataset is skewed towards older patients, with the majority of patients being above 50 years old. This reflects the higher prevalence of critical illnesses among older populations but may limit the dataset's applicability for studying pediatric populations or younger adults. The dataset is however fairly balanced in terms of gender distribution, with roughly 44% female and 56% male patients. This balance is relatively reflective of the general population and ensures that both genders are well-represented for most research purposes.

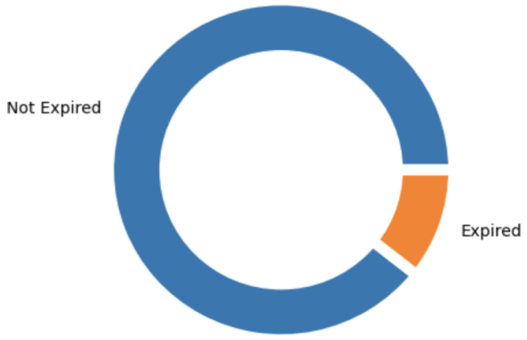
The ethnic distribution is not as well balanced, with a majority of patients being White (around 74%), followed by African American (around 15%), Hispanic (around 6%), Asian (around 3%), and other ethnicities. This imbalance may lead to potential biases when developing models or studying health disparities among different ethnic groups. Researchers should be cautious in generalizing findings to other ethnic populations.

During the data analysis we saw that some diagnoses are more common than others, such as sepsis, pneumonia, and acute respiratory failure. Less common diagnoses may not have enough data points for robust analysis or model development, which could lead to imbalanced representation of certain diseases or conditions. Also the data consists of patients from critical care units, which inherently represents a more severe and acute patient population. While this is useful for studying critical care, it limits the dataset's applicability for studying less severe or chronic conditions.

On this subject certain medications are more commonly prescribed, such as antibiotics and vasopressors. This can lead to an imbalance in the representation of medication usage and may impact the generalizability of findings related to medication effectiveness or safety

Addressing the data imbalance for our project

One of our first observations during the Exploratory Data Analysis, was the imbalance in the dataset, particularly in relation to the topic we were predicting, patient mortality.



As seen in the figure, a significant 89% of the data was attributed to class 0, signifying patients who did not expire. Conversely, only 10.6% of the data belonged to class 1, representing patients who expired. This posed a challenge when trying to predict patient mortality, as the model can easily become biased towards predicting survival. However, imbalanced datasets are not always an issue, as the data often is a true reflection of what is really happening in hospitals. This is an on-going debate in data science, and while our models were still performing well with an unbalanced dataset, we decided to still try a few techniques to address the imbalance and compare the outcomes.

Over-Sampling the Minority Class

The first method we used was oversampling the minority class using Synthetic Minority Over-sampling Technique (SMOTE), Adaptive Synthetic Sampling Approach (ADASYN) along with other hybrid sampling techniques such as SMOTE +Tomek Links and SMOTE + ENN. With each method however, the improvement in the model's performance was minimal. Additionally, these methods essentially create synthetic values to represent the minority class and we concluded that artificially balancing the data could potentially lead to a biased model, which would probably not accurately reflect real-world scenarios. If we were to use our model on fresh data, the new data is probably going to be largely imbalance as well as in real life, more patients in a hospital survive, and so the model may not perform well, Therefore, we made a conscious decision not to artificially balance the data, and instead focused on optimizing our models to better handle imbalanced datasets.

Threshold Optimisation

Threshold optimization was another method we explored to address the imbalanced nature of our dataset. Specifically, we used two common methods, G-mean and Youden J index, to determine the optimal threshold for our models. We then calculated the ROC AUC using this optimal threshold and compared it to the ROC AUC obtained using the default threshold. However, the

results showed that the ROC AUC scores actually dropped slightly when using the optimal threshold, for both methods. The result, while unexpected, could be due to several factors, such as the complexity of the dataset or the limitations of the threshold optimization methods. Nevertheless, this outcome was another reason we did not attempt to balance the dataset, as it's not always clear how to optimally balance the classes, and doing so can sometimes lead to unexpected results.

Overall, we acknowledged that balancing a dataset can be beneficial for many reasons. For example, it can improve model performance by giving equal importance to all the classes. It can also reduce the risk of predicting false positives, which in our case, of predicting patient expiration, would be very important for hospitals. In scenarios where classes carry equal importance, balancing a dataset can ensure classes are property represented.

Nevertheless, in future projects involving the Mimic 3 dataset, depending on the outcome we are predicting, it could be valuable to improve the representation of different age groups, genders, and ethnicities. Data from additional hospitals or healthcare systems can be collected and integrated into the dataset. By diversifying the data sources, the dataset's balance can be enhanced, providing more comprehensive insights into various demographic groups. For specific research questions or modeling tasks, we could also use stratified sampling

On the other hand, depending on your project, balancing a dataset may not be a reasonable idea. For our project specifically, the patient data is a reflection of real word distributions, in reality, mortality and survival are not balanced in most hospitals and so attempting to train models on a balance the dataset would ultimately result in overgeneralization. Altering classes could also impact the data negatively; undersampling the majority class could lead to a loss in important information and alternatively, oversampling the minority class could result in overfitting. Finally, balancing a dataset often requires time and additional resources, such as money and computational power. As we saw in our case, the oversampling methods only had a minimal improvement on the model, and therefore, since we want our model to maintain reflection on the real word distribution, as well as being efficient with our resources, we decided to keep the data in its unbalanced form.

techniques to ensure a more balanced distribution of key variables, such as diagnoses, interventions, or demographic characteristics. This method can help mitigate the impact of imbalances in the original dataset.

Promoting a culture of data sharing within the research community will facilitate the integration of various datasets, ultimately leading to a more comprehensive and balanced dataset. This can be achieved through the development of data-sharing platforms, policies, and incentives

Discussion

The future use for our model

Integrating our machine learning model in a hospital could offer several potential benefits for both healthcare providers and patients. Early and accurate prediction of patient mortality risk enables clinicians to prioritize resource allocation and triage, ensuring that high-risk patients receive the appropriate level of care and timely interventions. This can lead to improved patient outcomes, as timely treatments can potentially prevent further complications or deterioration.

Additionally, the model could assist physicians in making informed decisions regarding treatment plans, facilitating more personalized and effective care. It could also support communication with patients and their families, providing them with realistic expectations about prognosis and facilitating discussions around end-of-life care and advanced care planning.

The integration of the machine learning model for predicting patient mortality could overall contribute to enhancing the quality of care, optimizing resource management, and improving patient satisfaction in a hospital environment.

The legal procedures to deploy the model into an existing hospital

To deploy the model into a hospital, we would first need to obtain ethical and legal approvals. This will involve ensuring that the model is in compliance with regulations like the General Data Protection Regulation (GDPR) and the Health Insurance Portability and Accountability Act (HIPAA) in the US.
Then we could integrate the model into the hospital's information system using a suitable interface, such as a web API. Ensuring that the interface is secure, robust, and scalable.
After this, we need to continuously monitor the model's performance and retrain it periodically to improve its accuracy and reliability. Ensure that the model is auditable, and its outputs can be traced to the inputs. Besides, we need to communicate with stakeholders, including hospital staff and patients, about the model's capabilities and limitations. Ensure that they understand how the model works and how to interpret its results.

What were the issues and limitations we faced ?

Embarking on our project with the MIMIC-III dataset presented several challenges, given that our team had no prior experience with big data projects. Initially, we struggled to navigate and manage the vast amount of data, as we were unfamiliar with the techniques and tools required to efficiently handle such large datasets.

Understanding the relationships between the numerous tables proved to be a daunting task, as we needed to comprehend the intricate connections between various data elements to perform meaningful analyses. Furthermore, our personal computers were ill-equipped to process the data, often resulting in kernel crashes and significant delays in our progress.

Establishing a suitable server environment to overcome these limitations was time-consuming, and we faced a steep learning curve. Compounding these issues, we had to juggle multiple courses simultaneously, which limited the time and resources we could devote to this project.

Despite these challenges, our experience working with the MIMIC-III dataset provided valuable insights into the complexities of big data and healthcare analytics, ultimately contributing to our growth as researchers and data scientists.

The challenge we faced working on the data

The primary issue we encountered while analyzing the dataset was its sheer size and the intricate relationships between the numerous tables containing the data. The complexity of these relationships posed a significant challenge, as understanding the connections between different data elements was crucial for conducting meaningful analyses.

To further complicate matters, our team lacked prior medical knowledge, which hindered our ability to interpret the data and draw relevant insights. The combination of the dataset's size, the intricacy of the relationships between tables, and our limited medical background made it particularly difficult for us to navigate and effectively analyze the dataset.

What could we have improved ?

Prior to beginning the project, we could have invested more time in learning the fundamentals of healthcare analytics to better prepare the team for handling the complexities of the dataset.

Knowing better how to employ specialized tools and libraries, such as Apache Spark, Dask, Hadoop or PostgreSQL could have helped speed up the process and mitigate the risk of kernel crashes in our personal computers.

In the process of optimizing the model, we encountered significant obstacles in tweaking the hyperparameters. Firstly, training the model proved to be highly time-consuming when executed on our personal computers. The computational requirements for processing such a large and complex dataset demanded powerful hardware, which was not readily available to us. This time constraint limited our ability to test and iterate over different hyperparameter configurations, ultimately hindering the model's performance. Secondly, resorting to cloud-based solutions, such as Amazon servers, presented financial challenges. Although these services offered the required computational resources, each new training session incurred substantial costs. Balancing the need for optimal model performance with financial constraints, we could not afford to explore a wide range of hyperparameter settings, thus further restricting our ability to fine-tune the model.

Balancing the workload between the different courses and the data mission has required us to effectively manage time. Prioritizing tasks, setting realistic goals, and allocating dedicated time slots for working on the project had helped us maintain progress even if a better schedule of work could have helped us less compromise on others academic responsibilities.

Having more tutoring sessions or engaging with others experienced researchers, data scientists, or healthcare professionals could have provided valuable insights and guidance in navigating the dataset, understanding the relationships between tables, and addressing specific challenges.

Conclusion

The project aimed to create a model that predicts a critical health outcome, using a fusion of data types. For this project, we aimed to build a model that predicts patient mortality, focusing on both structured and unstructured data of patients. We trained a range of classifiers including Generalized Linear Models, Decision Trees, k-Nearest Neighbors, Naive Bayes, Ensemble Methods, and Neural Networks, separately on both the structured and unstructured data and evaluated the performance. We then trained fusion models after merging the data, to see if we could develop a model with more accuracy and better performance.

From our investigation, we can conclude that when predicting patient mortality, a fusion model resulted in better scores and is therefore more impactful than predicting this outcome with just one type of data. This finding highlights the importance of utilizing diverse data sources that exist in hospitals, and leveraging fusion methods to achieve high performing models that are interpretable.

This project demonstrates the potential of using multimodal machine learning techniques to predict patient mortality in a hospital setting. The developed models can aid medical professionals in identifying high-risk patients, optimizing resource allocation, and making more informed decisions regarding patient care. However, it is crucial to continue refining these models, incorporating additional data sources, and evaluating their real-world performance through further testing and validation before deploying them in clinical settings.

Appendix

Data description of the tables

(PK = Primary Key, FK = Foreign Key, TS = Timestamp, VAR = Variable Character, NUM = Number, DB = Double)

Admissions	PK	FK	Type	Hospital admission associated with ICU stay
ROW_ID			INT	Unique row identifier
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID	X		INT	Unique identifier for the hospital admission
ADMITTIME			TS	Date and time of admission
DISCHTIME			TS	Date and time of discharge
DEATHTIME			TS	date and time of death, if applicable
ADMISSION_TYPE			VAR	Type of admission (emergency, urgent, elective, newborn)
ADMISSION_LOCATION			VAR	Location where the patient was admitted (e.g., emergency department, transfer from another hospital)
DISCHARGE_LOCATION			VAR	Location where the patient was discharged (e.g., home, nursing home)
INSURANCE			VAR	Type of insurance
LANGUAGE			VAR	Patient's preferred language
RELIGION			VAR	Patient's religious affiliation
MARITAL_STATUS			VAR	Patient's marital status
ETHNICITY			VAR	Patient's ethnicity
DIAGNOSIS			VAR	Primary diagnosis for the hospital admission
HOSPITAL_EXPIRE_FLAG			INT	Binary flag indicating whether the patient died during the hospital admission
HAS_CHARTEVENTS_DATA			INT	Binary flag indicating whether there are any chart events recorded for the hospital admission

Callout	PK	FK	Data Type	Contains information about requests made by healthcare providers to specialty services, such as consultations or procedures.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
SUBMIT_WARDID			INT	Ward ID where the callout was submitted
SUBMIT_CAREUNIT			VAR	Care unit where the callout was submitted
CURR_WARDID			INT	Current ward ID of the patient
CURR_CAREUNIT			VAR	Current care unit of the patient
CALLOUT_WARDID			INT	Ward ID for the callout
CALLOUT_SERVICE			VAR	Service for the callout
REQUEST_TELE			INT	Flag indicating if the callout is for a teleconsultation
REQUEST_RESP			INT	Flag indicating if the callout is for a respiratory therapist
REQUEST_CDIFF			INT	Flag indicating if the callout is for a C. difficile consult
REQUEST_MRSA			INT	Flag indicating if the callout is for a MRSA consult
REQUEST_VRE			INT	Flag indicating if the callout is for a VRE consult
CALLOUT_STATUS			VAR	status of the callout
CALLOUT_OUTCOME			VAR	outcome of the callout

Caregivers	PK	FK	Data Type	Contains information about requests made by healthcare providers to specialty services, such as consultations or procedures.
ROW_ID	X		INT	Unique identifier for the row
CGID	X		INT	The unique identifier for each caregiver
LABEL			VAR	Describes the role of the caregiver
DESCRIPTION			VAR	Detailed description of the caregiver's role

Chartevent	PK	FK	Data Type	Contains every events occurring on a patient chart
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
ICUSTAY_ID		X	NUM	The unique identifier for the ICU stay
ITEMID			NUM	Unique identifier for the type of measurement
CHARTTIME			DATE	Date and time at which the measurement or observation was recorded
VALUE			VAR	Value of the measurement or observation
VALUENUM			NUM	Numeric value of the measurement or observation
VALUEUOM			VAR	The unit of measurement for the value
WARNING			NUM	Flag that indicates whether the measurement or observation triggered a warning or alert
ERROR			NUM	Flag that indicates whether the measurement or observation was recorded in error

Cptevent	PK	FK	Data Type	Contains information about Current Procedural Terminology (CPT) codes recorded during a patient's hospital stay
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
COSTCENTER			VAR	Text label that describes the cost center responsible for the procedure
CHARTDATE			TS	Date on which the procedure was recorded in the patient's chart
CPT_CD		X	VAR	Unique CPT code for the procedure
CPT_NUMBER			INT	The numerical part of the CPT code
CPT_SUFFIX			VAR	The suffix part of the CPT code
TICKET_ID_SEQ			INT	Unique identifier for the ticket associated with the procedure

D_cpt	PK	FK	Data Type	Contains information about Current Procedural Terminology (CPT) codes used in medical procedures and services
ROW_ID	X		INT	Unique identifier for the row
CATEGORY			INT	The broad category of the procedure or service described by the CPT code.
SECTIONRANGE			VAR	The range of sections to which the CPT code belongs
SECTIONHEADER			VAR	The name of the section
SUBSECTIONRANGE			VAR	The range of subsections
SUBSECTIONHEADER			VAR	The name of the subsection
CODESUFFIX			VAR	The suffix used in the CPT code
MINCODEINSUBSECTION			INT	The minimum CPT code within the subsection
MAXCODEINSUBSECTION			INT	The maximum CPT code within the subsection

D_ICD_Diagnoses	PK	FK	Data Type	Contains information about Current Procedural Terminology (CPT) codes used in medical procedures and services
ROW_ID			INT	Unique identifier for the row
ICD9_CODE	X		VAR	Unique ICD-9 code for the diagnosis
SHORT_TITLE			VAR	A short title for the diagnosis
LONG_TITLE			VAR	A longer description of the diagnosis

D_ICD_Procedures	PK	FK	Data Type	Reference table that contains information about International Classification of Diseases
ROW_ID			INT	Unique identifier for the row
ICD9_CODE	X		VAR	Unique ICD-9 code for the diagnosis
SHORT_TITLE			VAR	A short title for the diagnosis
LONG_TITLE			VAR	A longer description of the diagnosis

D_Items	PK	FK	Data Type	Reference table that contains information about International Classification of Diseases
ROW_ID			INT	Unique identifier for the row
ITEMID	X		INT	Unique identifier for the item
LABEL			VAR	Short name for the item
ABBREVIATION			VAR	More detailed description of the item
DBSOURCE			VAR	The source database for the item
LINKSTO			VAR	Table which contains data for the given ITEMID
CATEGORY			VAR	The category that the item belongs to
UNITNAME			VAR	The name of the unit of measurement used for the item

D_Labitems	PK	FK	Data Type	Information about all of the laboratory tests that were performed on the patients
ITEMID	X		INT	Unique identifier for the item
LABEL			VAR	Short name for the item
FLUID			VAR	More detailed description of the fluid that was tested
CATEGORY			VAR	The category that the item belongs to
LOINC_CODE			VAR	The LOINC code associated with the test

Datatimeevents	PK	FK	Data Type	Contains information about Current Procedural Terminology (CPT) codes recorded during a patient's hospital stay
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
ITEMID			VAR	Text label that describes the cost center responsible for the procedure
CHARTTIME			TS	The time that the event occurred
VALUE		X	VAR	The value associated with the event

Diagnoses_icd	PK	FK	Data Type	Contains information about the International Classification of Diseases,
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
SEQ_NUM			VAR	Sequence number assigned to the diagnosis
ICD9_CODE			TS	The ICD-9-CM code assigned to the diagnosis

Drgcodes	PK	FK	Data Type	Contains information about the Diagnosis Related Group (DRG)
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
DRG_TYPE			VAR	The version of the DRG classification system used
DRG_CODE			TS	The DRG code assigned to the hospital stay

Icustays	PK	FK	Data Type	Contains information on each patient's ICU stay
ICUSTAY_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
INTIME			TS	Date and time the patient was admitted to the ICU
OUTTIME			TS	Date and time the patient was discharged from the ICU
LOS			DB	Length of stay in the ICU for the patient
ICUType			VAR	Categorical variable indicating the type of ICU the patient was admitted
FIRST_CAREUNIT			VAR	First ICU where the patient was treated during the hospitalization
LAST_CAREUNIT			VAR	Last ICU where the patient was treated
FIRST_WARDID			INT	Identifier for the first ICU where the patient was treated
LAST_WARDID			INT	Identifier for the last ICU where the patient was treated
DBSOURCE			VAR	Indicates the source of the data

Inputevents_cv	PK	FK	Data Type	contains information on medications and other inputs administered to patients during their ICU stay
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
ICUSTAY_ID		X	INT	Unique identifier for each ICU stay
ITEMID		X	INT	Identifier for the medication or other input administered to the patient
CHARTTIME			TS	Date and time the medication or other input was administered to the patient
AMOUNT			DB	The amount of the medication or other input administered to the patient
AMOUNTUOM			VAR	The unit of measure for the amount variable
RATE			DB	The rate at which the medication or other input was administered to the patient (if applicable)
RATEUOM			VAR	The unit of measure for the rate variable
STORETIME			TS	Date and time the data point was entered into the electronic medical record
CGID		X	INT	Unique identifier for the caregiver who entered the data point into the electronic medical record
ORDERID			INT	Unique identifier for the order associated with the medication or other input
LINKORDERID			INT	Unique identifier for the parent order associated with the medication or other input
STOPPED			VAR	Binary variable indicating whether the medication or other input was stopped (1 = stopped, 0 = not stopped)
NEWBOTTLE			INT	Binary variable indicating whether a new medication or input bottle was started (1 = new bottle started, 0 = bottle not started)
ORIGINALAMOUNT			DB	The original amount of the medication or other input ordered by the clinician
ORIGINALAMOUNTUOM			VAR	The unit of measure for the original amount variable
ORIGINALROUTE			VAR	The route of administration originally ordered by the clinician
ORIGINALRATE			DB	The original rate of administration ordered by the clinician
ORIGINALRATEUOM			VAR	The unit of measure for the original rate variable

Inputevents_mv	PK	FK	Data Type	Contains information on medications and other inputs administered to patients during their ICU stay, specifically for mechanically ventilated patients.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
ICUSTAY_ID		X	INT	Unique identifier for each ICU stay
STARTTIME			TS	Date and time the medication or other input was started
ENDTIME			TS	ate and time the medication or other input was stopped
ITEMID			INT	Numeric identifier for the medication or other input administered to the patient
AMOUNT			DB	The amount of the medication or other input administered to the patient
AMOUNTUOM			VAR	The unit of measure for the amount variable
RATE			DB	The rate at which the medication or other input was administered to the patient
RATEUOM			VAR	The unit of measure for the rate variable
STORETIME			TS	Date and time the data point was entered into the electronic medical record
CGID			INT	Unique identifier for the caregiver who entered the data point into the electronic medical record
ORDERID			INT	Unique identifier for the order associated with the medication or other input
LINKORDERID			INT	Unique identifier for the parent order associated with the medication or other input
STOPPED			INT	Binary variable indicating whether the medication or other input was stopped
NEWBOTTLE			VAR	Binary variable indicating whether a new medication or input bottle was started
ISERROR			INT	Binary variable indicating whether there was an error in the administration of the medication or other input
ERRORDESCRIPTION			VAR	Description of the error that occurred
RESULTSTATUS			VAR	Status of the medication or other input administration
STOPPEDBY			TS	Unique identifier for the caregiver who stopped the medication or other input administration (if applicable)
COMMENTS_EDITEDBY			VAR	Unique identifier for the caregiver who edited the medication or other input administration comments
COMMENTS_CANCELEDBY			VAR	Unique identifier for the caregiver who canceled the medication or other input administration comments
COMMENTS_DATE			TS	Date and time the medication or other input administration comments were entered
COMMENTS			VAR	Comments associated with the medication or other input administration

Labevents	PK	FK	Data Type	Contains information on laboratory test results for each patient during their ICU stay.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
ITEMID			INT	project identifier
CHARTTIME			TS	measure time
VALUE			VAR	Measurement items
VALUENUM			DB	Measure numerical data
VALUEUOM			VAR	Units of measurement
FLAG			VAR	Is the measured value abnormal or not

Microbiologyevents	PK	FK	Data Type	Contains information on microbiology test results for each patient during their ICU stay.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
CHARTDATE			TS	Date the microbiology test was performed
CHARTTIME			TS	Time the microbiology test was performed
SPEC_ITEMID			INT	Numeric identifier for the specific item tested
SPEC_TYPE_DESC			VAR	Description of the type of specimen tested
ORG_ITEMID			INT	Numeric identifier for the specific organism identified
ORG_NAME			VAR	Name of the organism identified
ISOLATE_NUM			INT	Numeric identifier for the specific isolate identified
AB_ITEMID			INT	Numeric identifier for the specific antibiotic tested
AB_NAME			VAR	Name of the antibiotic tested
DILUTION_TEXT			VAR	Text description of the dilution factor used
DILUTION_COMPARISON			VAR	Comparison value used to interpret the dilution factor
DILUTION_VALUE			DB	Numeric value of the dilution factor
INTERPRETATION			VAR	Interpretation of the microbiology test result

Noteevents	PK	FK	Data Type	Contains free-text clinical notes recorded by healthcare providers during a patient's ICU stay.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
CHARTDATE			TS	The date the note was recorded
CHARTTIME			TS	record the date and time of the note
STORETIME			TS	Record the date and time when the note was saved to the system
CATEGORY			VAR	Record Type 'Discharge'
DESCRIPTION			VAR	Record category 'Summary'
CGID			INT	Nursing staff identifier
ISERROR			CHAR	'1' means the record is marked as error
TEXT			TEXT	Content of doctor's order

Outputevents	PK	FK	Data Type	Contains information on the output measurements for each patient during their ICU stay.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
ICUSTAY_ID			INT	Unique identifier for each ICU stay
CHARTTIME			TS	Time the output measurement was recorded.
ITEMID			INT	Unique identifier for the type of output measurement
VALUE			DB	Numeric value of the output measurement
VALUEUOM			VAR	Unit of measurement for the output value
STORETIME			TS	Time the output measurement was stored in the database
CGID			INT	Unique identifier for the healthcare provider who recorded the output measurement

Patients	PK	FK	Data Type	Patients associated with an ICU admission
SUBJECT_ID	X		INT	Unique identifier for the patient
GENDER			VAR	Sex of the patient
DOB			TS	Date of Birth
DOD			TS	Date of Dead
DOD_HOSP			TS	Date of death registered in hospital
DOD_SSN			TS	Social Security Number registered date of death
EXPIRE_FLAG			VAR	Marker for death

Prescriptions	PK	FK	Data Type	Contains information about medications that were prescribed to patients during their hospital stay.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
ICUSTAY_ID			INT	Unique identifier for each ICU stay
CHARTTIME			TS	Time the output measurement was recorded.
STARTDATE			TS	Medication start time
ENDDATE			TS	Medication end time
DRUG_TYPE			VAR	Drug type
DRUG			VAR	Drug name
DRUG_NAME_GENERIC			VAR	Drug description
FORMULARY_DRUG_CD			VAR	Prescription drug code
GSN			VAR	Universal serial number
NDC			VAR	National Drug Code
PROD_STRENGTH			VAR	The prescribed dose of the drug
DOSE_VAL_RX			VAR	The units of the prescribed dose
DOSE_UNIT_RX			VAR	The dispensed amount of the drug
FORM_UNIT_DISP			VAR	The units of the dispensed amount
ROUTE			VAR	The route of administration for the drug (e.g., oral, intravenous)
GSN_FLAG			INT	A flag that indicates whether the drug is listed in the National Drug Code (NDC) directory

Procedureevents_mv	PK	FK	Data Type	contains information on procedures performed on patients during their ICU stay. The table includes both invasive and non-invasive procedures, such as intubation, central line placement, and bronchoscopy.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
ICUSTAY_ID		X	INT	Unique identifier for each ICU stay
STARTTIME			TS	Date and time the procedure was initiated
ENDTIME			TS	Date and time the procedure was completed
ITEMID		X	INT	Unique identifier for the type of output measurement
VALUE			DP	Numeric value associated with the procedure
VALUEUOM			VAR	Unit of measurement for the procedure value
LOCATION			VAR	Location of the procedure
LOCATIONCATEGORY			VAR	Categorization of the location of the procedure
STORETIME			TS	Time the procedure information was stored in the database
CGID		X	INT	Unique identifier for the healthcare provider who recorded the output measurement

Procedures_icd	PK	FK	Data Type	Contains information on procedures performed on patients during their hospital stay. The table includes both invasive and non-invasive procedures, such as surgeries and diagnostic tests.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
SEQ_NUM			INT	Numeric sequence number indicating the order of the procedures performed during the hospital stay
ICD_CODE		X*	VAR	International Classification of Diseases, Ninth Revision (ICD-9) code indicating the type of procedure perform

Services	PK	FK	Data Type	Contains information on the different services that a patient receives during their hospital stay. A service in this context refers to a specific medical team responsible for the care of a patient, such as the cardiology service or the surgical service.
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
TRANSFERTIME			TS	Date and time the patient was transferred to the service
PREV_SERVICE			VAR	Service the patient was previously receiving
CURR_SERVICE			VAR	Service the patient is currently receiving

Transfers	PK	FK	Data Type	Contains information on the location of patients during their hospital stay
ROW_ID	X		INT	Unique identifier for the row
SUBJECT_ID		X	INT	Unique identifier for the patient
HADM_ID		X	INT	Unique identifier for the hospital admission
ICUSTAY_ID		X	INT	Date and time the patient was transferred to the service
DBSOURCE			VAR	Source database of the item
EVENTTYPE			VAR	Type of event
PREV_CAREUNIT			VAR	Previous careunit
CURR_CAREUNIT			VAR	Current careunit
PREV_WARDID			INT	Identifier for the patient's previous ward
CURR_WARDID			INT	Identifier for the patient's current ward
INTIME			TS	Time when the patient was transferred into the unit
OUTTIME			TS	Time when the patient was transferred out of the unit
LOS			INT	Length Of Stay in the unit in minutes

The current status of the application of the MIMIC-III database

Literature Review

Summary

The Digital well-being system has developed rapidly in recent years and is widely used by major hospitals. Even so, due to some reasons such as security, this information is challenging to integrate and apply to scientific research. The release of MIMIC-III (Medical Information Mart for Intensive Care) solves this problem. It integrates patient data from Beth Israel Deaconess Medical Center in Boston and is free to access and use. Since its release, it has been widely used in scientific research, contributing to research and development in patient outcome prediction and entity identification. The application of MIMIC-III in the field of machine learning is listed and analyzed here, and some questions are raised.

Introduction

Digital well-being recording systems have been widely used in major hospitals in recent years. In the 7 years from 2008 to 2014, the number of non-federal acute care hospitals with basic digital systems increased from 9.4% to 75.5% [1]. Nonetheless, the interoperability of digital systems remains an issue, posing no small challenge for the Data Transmission Service. In addition, in scientific research, experiments for medical Data Analysis lack reproducibility. Therefore, an open, integrated, and informative repository of medical information is needed to provide researchers with information. As a result, Johnson et al. released the MIMIC-III (Medical Information Mart for Intensive Care) database, which is also an update to the MIMIC-II (Multiparameter Intelligent Monitoring in Intensive Care) database [2].

Introduction to MIMIC Database

MIMIC-III integrates clinical data from identified patients at Beth Israel Deaconess Medical Center in Boston, including different aspects of patient baseline information, laboratory information, diagnostic reports, etc., and can be used to explore topics such as Machine Learning methods for predicting patient outcomes, blood pressure The clinical implications of monitoring technology and the semantic analysis of unstructured patient notes. And under the data usage protocol, it enables international researchers to obtain this data and conduct research at no cost. Kurniati et al. [3] evaluated the quality of the data provided by MIMIC-III and found that the comprehensive data provided in the database can effectively help researchers conduct research. Better data cleaning and integration can make the best use of this effect.

In addition, in the field of scientific research, researchers are increasingly worried about the reproducibility of scientific results [4]. To this end, Johnson et al. developed a matching code repository (Mimic Code Repository) based on the MIMIC-III database. The codebase is open source and includes standardized scripts for languages such as SQL, Python, and R. It provides a communication community for researchers, researchers can upload code to communicate, and other researchers can download copies to ensure that research using MIMIC-III is comparable and reproducible [5].

Data Analysis Based on Machine Learning

Patient outcomes, such as length of stay, readmission, and type of discharge, are considered important indicators that need to be evaluated in the clinical treatment process [9]. Most of the existing studies use data mining, machine learning, or deep learning methods to generate prediction models for specific types of clinical outcomes [10]. At present, many researchers have used machine learning, deep learning, and other methods to predict patient outcomes, with good results. Lee et al. [11] have shown that predicting patient outcomes using methods such as Machine Learning can help clinicians make better clinical decisions. Sanjay et al. [12] also demonstrated the feasibility of using Machine Learning algorithms for predicting patient outcomes on the MIMIC-III datase

Prediction of death risk

Death is very common and the most serious ICU patient outcome and an accurate assessment of mortality risk facilitate timely clinical intervention and quotas [13]. For the prediction of death risk, the vast majority of prediction models initially used are based on overall baseline patient characteristics. These systems typically rely on a weighted linear combination of characteristics such as age, type of admission, and vital sign measurements. Such as Modifide Early Warning Score (MEWS) [14], Sepsis-related Organ Failure Assessment (SOFA) [15] and Simplified Acute Physiology Score (SAPS II) [16]. Davoodi et al. [17] proposed a deep rule-based fuzzy classification system (Deep Rule-Based Fuzzy System, DRBFS) to extract the data used in MIMIC-III, and use a large number of input variables to estimate the risk of hospitalization in ICU patients. accurate prediction. The Naive Bayesian Model (Naive Bayes, NB), Decision Tree (Decision Tree, DT), Fradient Boosting (GB), Deep Belief Nets (DBN), and other commonly used classifiers were used to evaluate the method, which proved the feasibility of the method. However, the patient's various indicators are not fixed during hospitalization, so these analyses based on baseline data are not ideal for clinical application [18]. To solve this problem, Jensen et al. [19] proposed the concept of temporal disease trajectories to simulate the expected progression of patients over time, thereby mapping patient trajectories in time for other predictions. On the basis of predicting patient trajectories, Jones et al. [20] applied MIMIC-III data and used two deep learning techniques, namely Unsupervised Autoencoders and long short-term memory network (Long Short-term Memory, LSTM) to predict ICU care outcomes and survival rates, and applied Time Series to predict more accurate results than traditional Machine Learning methods.

Readmission threat and risk assessment

Intensive care unit (ICU) readmissions are an important clinical issue because they are associated with patient harm, inefficiency, and higher costs [21]. Moreover, patients who are readmitted to the ICU experience more adverse events, and the in-hospital mortality rate can be up to six times that of patients who are not readmitted [22]. Therefore, predicting patient readmissions and intervening can reduce the chance of readmission and reduce mortality. McWilliams et al. [23] used a random forest (Random Forest,

RF) [24] and a Logistic Classifier (LC) [25] algorithm to establish a patient discharge decision model using MIMIC-III data to help doctors decide whether to discharge patients.

The traditional method for predicting readmission is to use regression models to predict the probability of readmission, while in recent years Churpek et al. [26] used Machine Learning methods to analyze readmission and obtained better results than regression models. On this basis, et al. improved the Machine Learning algorithm, using a series of patient characteristics extracted from MIMIC-III, such as patient characteristics, nursing assessment, drugs, intensive care unit interventions, and diagnostic tests, to establish a layer-enhanced machine model, which obtained better results than the previous model. prediction results.

Prediction of disease

Sepsis is a general term for some complex diseases. It is defined in Sepsis-3 [27] as life-threatening organ dysfunction due to a dysfunctional host response to infection. Due to the heterogeneity of the disease and the diversity of host responses, these diseases have long been difficult for doctors to identify and diagnose. Therefore, if sepsis can be accurately predicted, clinical decisions can be made in a targeted manner. There are also many scoring systems for predicting sepsis, such as SOFA score [16], MEWS score [17], etc. Desautels et al. [28] proposed the insight Machine Learning model on the basis of traditional methods, extracted data in MIMIC-III, and applied insight scoring and traditional scoring methods such as SOFA and MEWS scoring to predict whether sepsis will occur in a fixed time. The results show that insight has better performance. In order to improve the prediction performance, Nemati et al. [29] proposed a Machine Learning model that applies dynamic time series to predict sepsis. The model was built using data from Emory University Hospital, and verified with MIMIC-III data, proving the availability of the algorithm.

Acute kidney injury is also a complex disease commonly seen in the ICU and is closely related to patient outcomes such as readmission and death [30]. Zimmerman et al. [31] used MIMIC-III data to extract features including patient age, creatinine, and urine output after excluding patients with pre-existing kidney injury at admission. Using Logistic Regression (LR), RF and Machine Learning models including Artificial Neural Networks (ANN) were analyzed, proving the utility of the algorithm in predicting acute kidney injury in patients.

Data Analysis Based on natural language processing (NLP)

In the healthcare system, patient medical records are a Big data source. But in many cases, doctor's notes, image reports, etc. are composed of unstructured text. This data cannot be analyzed directly using statistical tools, so it needs to be processed using the Named Entity Recognition (Named Entity Recognition, NER) method.
Embeddings based on neural networks have greatly advanced the development of natural language processing (Natural Language Processing, NLP). Devlin et al. [32] used long short-term memory network (LSTM) and conditional random field (conditional random field, CRF) Machine Learning method to extract labels in the MIMIC-III report, and achieved good results. Recently, more advanced embedding methods and representations (such as ELMo [33], and BERT [34]) have further promoted the development of NLP. However, these methods are not well practiced in clinical concept extraction. Si et al. [35] applied traditional word embedding (Word Embedding) and context embedding (Contextual Embedding) methods to the MIMIC-III dataset to demonstrate their feasibility in clinical concept extraction.

In addition, the International Classification of Diseases (ICD) coding has been widely used to describe the diagnosis of patients [36]. Manual coding is inefficient and cumbersome, and if deep learning methods are used, coding efficiency can be greatly improved. Li et al. [37] applied the deep learning method to extract features in MIMIC-III for ICD-9 coding and verified its reliability.

Summary

MIMIC-III provides information on all aspects of ICU patients and is free and open to researchers. Since its release, due to the richness of patient information it provides, it has been widely used to establish models for predicting patient outcomes, establish entity recognition models that can be applied to clinical medical cases, and conduct retrospective studies to explore the relationship between patient attributes. Among them, it is more used in the prediction of patient outcomes, and death is the most important outcome.

References

[1] Charles D,King J, Patel V,Furukawa M. Adoption of Electronic Health record Systems among U.S[J]. ONC Data Brief ,2013,9: 1–9.

[2] Johnson A E W ,Pollard T J ,Shen L ,et al. MIMIC-III,a freely accessible critical care database[J]. Scientific Data,2016,3:160035.

[3] Kurniati A P ,Rojas E ,Hogg D ,et al. The assessment of data quality issues for process mining in healthcare using Medical Information Mart for Intensive Care III,a freely available e-health record database[J]. Health Informatics Journal,2019,25(4):1878-1893.

[4] Baker Monya. 1,500 scientists lift the lid on reproducibility[J]. Nature,2016,533(7604):452-454.

[5] Alistair-E-W Johnson,Stone David-J,Celi Leo-A,et al. The MIMIC Code Repository: enabling reproducibility in critical care research[J]. Journal of the American Medical Informatics Association,2018,25(1): 32-39.

[6] Wang H,Yang H. Statistical Analysis of Inter-attribute Relationships in Unfractionated Heparin Injection Problems[J]. Annu Int Conf IEEE Eng Med Biol Soc,2020,2020:5374-5377.

[7] Vincent J,Nielsen N D,Shapiro N I,et al. Mean arterial pressure and mortality in patients with distributive shock: a retrospective analysis of the MIMIC-III database[J]. Annals of Intensive Care,2018,8(1):107.

[8] Neto A S ,Deliberato R O ,Johnson A ,et al. Mechanical power of ventilation is associated with mortality in critically ill patients: an analysis of patients in two observational cohorts[J]. Intensive Care Medicine,2018,44:1914–1922

[9] Huang Z ,Juarez J M ,Duan H ,et al. Length of stay prediction for clinical treatment process using temporal similarity[J]. Expert Systems with Applications,2013,40(16):6330–6339.

[10] Outcome Prediction in Clinical Treatment Processes[J]. Journal of Medical Systems,2016,40(1):1-13.

[11] Lee J . Is Artificial Intelligence Better Than Human Clinicians in Predicting Patient Outcomes?[J]. Journal of Medical Internet Research,2020,22(8):e19918.

[12] Sanjay P ,Chuizheng M ,Zhengping C ,et al. Benchmarking deep learning models on large healthcare datasets[J]. Journal of Biomedical Informatics,2018,83:112-134.

[13] Siontis G C M ,Tzoulaki I ,Ioannidis J P A . Predicting death: an empirical evaluation of predictive tools for mortality.[J]. Archives of Internal Medicine,2011,171(19):1721-1726.

[14] Subbe C P ,Slater A ,Menon D ,et al. Validation of physiological scoring systems in the accident and emergency department[J]. Emergency Medicine Journal Emj,2006,23(11):841.

[15] Vincent J L ,Moreno R ,Takala J ,et al. The SOFA (Sepsis-related Organ Failure Assessment) score to describe organ dysfunction/failure[J]. Intensive Care Medicine,1996,22(7):707-710.
[16] Le,Gall,J,et al. A new Simplified Acute Physiology Score (SAPS II) based on a European/North American multicenter study[J]. JAMA: The Journal of the American Medical Association,1993,270(24):2957-2963 .
[17] Davoodi R ,Hassan Moradi M . Mortality Prediction in Intensive Care Units (ICUs) Using a Deep Rule-based Fuzzy Classifier[J]. Journal of Biomedical Informatics,2018:48-59.
[18] Calvert J ,Mao Q ,Hoffman J L ,et al. Using electronic health record collected clinical variables to predict medical intensive care unit mortality[J]. Annals of Medicine and Surgery,2016,11:52-57.
[19] Jensen A B ,Moseley P L ,Oprea T I ,et al. Temporal disease trajectories condensed from population-wide registry data covering 6.2 million patients[J]. Nature Communications,2014,5:4022.
[20] Beaulieu-Jones B K ,Orzechowski P ,Moore J H . Mapping Patient Trajectories using Longitudinal Extraction and Deep Learning in the MIMIC-III Critical Care Database[J]. Pacific Symposium on Biocomputing Pacific Symposium on Biocomputing,2018,23:123-132.
[21] Kramer A A ,Higgins T L ,Zimmerman J E . The association between ICU readmission rate and patient outcomes[J]. Critical Care Medicine,2013,41(1):24-33.
[22] Van Sluisveld N, Bakhshi-Raiez F, de Keizer N, et al. Variation in rates of ICU readmissions and post-ICU in-hospital mortality and their association with ICU discharge practices.[J]. BMC Health Services Research, 2017,17(1):281.
[23] McWilliams C J, Lawson D J, Santos-Rodriguez R, et al. Towards a decision support tool for intensive care discharge: machine learning algorithm development using electronic healthcare data from MIMIC-III and Bristol, UK[J]. BMJ Open, 2019,9(3):e25925.
[24] Liaw A ,Wiener M . Classification and Regression by randomForest[J]. R News,2002,2:18-22.
[25] Dreiseitl S, Ohno-Machado L. Logistic regression and artificial neural network classification models: a methodology review[J]. JOURNAL OF BIOMEDICAL INFORMATICS, 2002,35(5-6):352-359.
[26] Churpek M M ,Yuen T C ,Winslow C ,et al. Multicenter Comparison of Machine Learning Methods and Conventional Regression for Predicting Clinical Deterioration on the Wards[J]. Critical care medicine,2016,44(2):368-374.
[27] Rather A R ,Kasana B . The Third International Consensus Definitions for Sepsis and Septic Shock (Sepsis-3)[J]. J Med,2015,18(2):162-164.
[28]Desautels T ,Calvert J ,Hoffman J ,et al. Prediction of Sepsis in the Intensive Care Unit

With Minimal Electronic Health Record Data: A Machine Learning Approach[J]. JMIR Medical Informatics,2016,4(3).

[29] Nemati S ,Holder A ,Razmi F ,et al. An Interpretable Machine Learning Model for Accurate Prediction of Sepsis in the ICU[J]. Critical Care Medicine,2017:1.

[30]Ali T, Khan I, Simpson W, et al. Incidence and outcomes in acute kidney injury: a comprehensive population-based study[J]. Journal of the American Society of Nephrology : JASN, 2007,18(4):1292-1298.

[31] Zimmerman L P ,Reyfman P A ,Smith A D R ,et al. Early prediction of acute kidney injury following ICU admission using a multivariate panel of physiological measurements[J]. BMC Medical Informatics and Decision Making,2019,19(S1):6.

[32] Jauregi Unanue I ,Zare Borzeshi E ,Piccardi M . Recurrent neural networks with specialized word embeddings for health-domain named-entity recognition[J]. Journal of Biomedical Informatics,2017,76:102-109.

[33] Peters ME, Neumann M, Iyyer M., et al. Deep contextualized word repre-Sentations[J] Proceedings of NAACL-HLT, 2018: 2227–2237.

[34] Devlin J, Chang M-W, Lee K, Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of NAACL-HLT, 2019: 4171–4186.

[35] Si Y, Wang J, Xu H, et al. Enhancing clinical concept extraction with contextual embeddings[J]. Journal of the American Medical Informatics Association, 2019,26(11):1297-1304.

[36]Peter B. Jensen,Lars J. Jensen,Søren Brunak. Mining electronic health records: towards better research applications and clinical care[J]. Nature Reviews Genetics,2012,13(6):395-405.

[37] Li M, Fei Z, Zeng M, et al. Automated ICD-9 Coding via A Deep Learning Approach[J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2019,16(4):1193-1202.

PhysioNet website (https://physionet.org/)

MIMIC-III dataset page (https://mimic.physionet.org/)

CITI Program website (https://www.citiprogram.org/)