Predicting Hospital Readmittance Via Machine Learning Classification   
Techniques Using Structured and Unstructured Data

Evan Canfield & Christine Personett

DSBA 6156: Applied Machine Learning

**Introduction**

Hospital readmission rates are an important consideration for hospitals in the United States. Readmission rates are directly connected to Medicare reimbursement, with Medicare being the largest payer of hospital services in the country. Hospitals with high readmission rates incur financial penalties. Readmissions are defined as any return to the hospital within 30 days of being discharged. Currently, the LACE Index is to predict readmissions and provides a score based on four features: the patient’s length of stay, acuity, co-morbidities, and emergency department visits within the last 6 months.  This tool is used by physicians currently, and within some studies show poor discrimination in distinguishing between patients that will and will not be readmitted.

We will be exploring the prediction of hospital readmissions through machine learning methods applied to both structured and unstructured data. In analysis of structured data, we will be employing modeling methods such as logistic regression, Naive Bayes, support vector machines, and random forest. In analyzing unstructured data, we will process digitized doctor reports through natural language processing methods. Through the analysis of both structured and unstructured methods, we plan to improve on the prediction of patients likely to be readmitted.

**Related Work**

**Methods**

The analysis of hospital readmissions will consist of two parts: analysis of structured and unstructured data. After processing, the structured data provided in the MIMIC-III dataset will be analyzed with logistic regression, naive bayes, random forest, and support vector machine in order to predict hospital readmittance.

The unstructured data in the data set is in the form of digitized doctor’s reports. These reports will be processed with natural language processing methods. Once processed, the notes will be analyzed with similar classification models to predict hospital readmittance.

The outputs of each method, structured and unstructured, will then be scored and compared to determine which model provides the most desirable and robust results. The Lace Index is currently one method used to identify patients likely to be readmitted. Multiple studies have analyzed the LACE method. The predictive scoring capabilities of the models developed for this project will be compared to the available output of such studies.

**Datasets**

The MIMIC-III data is a relational database of patient information gathered from Beth Israel Deaconess Medical Center.  It contains twenty-six tables with information on patient stays, diagnoses, procedures, prescriptions, lab results and vital signs among other attributes.  There are some missing measurements in the data and some measurements that have different values assigned in across different tables. We will be joining tables on keys provided to access all of the required data.

The CITI Program course, “Data or Specimens Only Research” was completed on October 16th and 17th.  A data use agreement and credentialing application were both submitted to Physionet.  Access was granted on October 24, 2019.

There are 26 tables in the database, they vary in size from hundreds to millions of rows.  The largest table has 330,712,483 rows. In analyzing the available structured data, we will primarily be using the Admissions table to access individual patients and their unique hospital visit. It has several data types including integers, date-time, strings and binary values.

For unstructured data analysis, we will be analyzing the NoteEvent table, which collects digitized doctor reports for each patient. The NoteEvent table contains 2,083,180 observations, with the digitized reports stored as free text.

**Structured Data Preprocessing**

The Admissions table needed some cleaning prior to modeling. The subject id column values were given leading zeros if they had five digits, to standardize the values. Columns that were not needed were dropped; this included the row id column. Missing values were initially stored as a space; these were replaced with nan values for further processing later. Date values were converted from strings to datetime, this included time of admission, discharge, death, emergency department (ED) registration, and ED discharge. The patients table needed birth time converted from a string to datetime and standardization of subject id before it could be joined to the admissions table.

The admissions table was joined with the patient table in a left join using the subject id as a key. An admission type column value of newborn was dropped. If there was a value for deathtime, the subject was dropped. Language, religion, diagnosis and ED dates were also dropped. There were a significant number of missing values for marital status, they were treated as an additional category instead of dropping or imputing since they may not have been missing at random. Age at admission was a calculated field taken from the difference between admission date and birthdate. Thirty-day readmissions were calculated and stored in the column readmit\_30. Patients were grouped together using subject id and if the difference between a discharge date and admission date was less than or equal to 30 days, then they were counted as having a readmission. Patients with a readmission category of elective were excluded.

The ICU stay table was joined with the admissions table on hadmid with a right join. This determined whether the patient went to the intensive care unit (ICU) during their hospital stay. If there was nan for the icu stay value, then the hospital visit did not involve the ICU.

The lab events table was also used to gather specific lab values during the hospital visit. The lab events table also included outpatient lab values, therefore if there was not a hadmid associated with the lab, it was excluded. Lab events were filtered by the item id, then the chart time, value and units were extracted for blood urea, platelets, magnesium, calcium and albumin. The hadmid and lab item ids were grouped, and total flags were counted from the flag column in the lab events table. This indicated that there was an abnormal value for the specific lab during the hospital stay. Presence of an abnormal lab value was recorded instead of specific values due to the number of missing lab values. The missing values were not imputed with a normal value because the absence of a lab test, especially in the hospital, does not imply that it would be a normal value. The calculated flag for the lab was categorical with three values including abnormal, no abnormal values and no lab drawn. Each flag value was recorded for the columns of blood urea, platelets, magnesium, calcium and albumin. Then the admissions table was left joined with the modified lab events table.

The microbiology events table was processed similarly to the lab events table. The hadmid column and a calculated column of whether a microbiology test was ordered. The admissions table was left joined to the microbiology events table on hadmid. If there was a lab drawn for the hadmid, then microbiology column was flagged as a one, if there was a null value it was flagged as a zero. The order of a microbiology test indicates that the physician suspected an infection and was sufficient to include regardless of the outcome of the culture.

After the complete data frame was created using the admissions table, patients table, icu table, lab events table, and microbiology events table, additional processing was needed. The length of stays were transformed to the log scale to create a distribution closer to normal. The ethnicity variable was condensed to fewer categories. Spaces were also converted to a text character for strings like insurance, marital status, discharge location and admission location. All categorical variables were converted to an n-1 set of dummy variables, these included: admission type, admission location, discharge location, insurance, marital status, ethnicity and gender.

**Preprocessing Structured Data Plus Unstructured Text**

The admissions table was stripped to the hadmid and readmit\_30 columns and joined with the note events table using a left join. In the NoteEvents table, five columns were evaluated: HADMID, Iserror, Category, Description, and Text.  First all rows with an Inerror value of 1 were dropped to eliminate notes that were recorded in error.  Two datasets were made from the unstructured text data, one including all categories of text and one limited to the category of discharge notes.

The majority of the dataset was comprised of patients that had not been readmitted.  To adjust for the imbalance, patients that were not readmitted were under-sampled for training the model.  This reduced the size of the training data. Another technique used for the imbalanced dataset was ADASYN. This is a modification of the SMOTE function which randomly synthesizes new instances of the minority class (readmissions) and increases the size of our training set.

Truncated singular value decomposition (SVD) was used as dimensionality reduction because of the sparse dataset.  This helped the models run more efficiently while retaining most of the information. The number of dimensions chosen for SVD was 300 which was 10% of the columns after vectorizing. Logistic regression was run with and without SVD and no reduction in performance was seen.

The data was preprocessed for Natural Language Processing (NLP) by replacing punctuation and newline characters with spaces.  All of the words were converted to lowercase then tokenized and vectorized using CountVectorizer and TfidfVectorizer. The most common words were viewed using a bar chart and added as stop words.

**Predicting Readmissions Using Unstructured Text**

Logistic regression was performed to predict whether a patient would be readmitted to the hospital within 30 days. The model trained with 60% of the dataset while 20% was reserved for both validation and testing. Under sampling was used to balance the dataset and resulted in an accuracy of 0.648, precision of 0.126, recall of 0.690 and f1-score of 0.107.

We plan to tune the model to achieve the optimal f1-score or recall and run other models to compare their performance to logistic regression.