|  |
| --- |
| **Data Analytics** |
| Predictive Analysis of Hospital Readmissions |
| CSCI-4957 |

|  |
| --- |
| Koi Stephanos  4-18-2020 |

Contents

[Introduction 2](#_Toc38126908)

[Preprocessing 3](#_Toc38126909)

[Checking and Reading the Data 3](#_Toc38126910)

[Dealing with Missing Values 4](#_Toc38126911)

[Exploratory Analytics 5](#_Toc38126912)

[Data Visualization 5](#_Toc38126913)

[Patients Readmitted vs Non-Readmitted: 5](#_Toc38126914)

[Rate of Patients Currently on Diabetes Medication: 5](#_Toc38126915)

[Breakdown of Patients by Race: 6](#_Toc38126916)

[Breakdown of Patients by Gender: 6](#_Toc38126917)

[Breakdown of Patients by Age: 6](#_Toc38126918)

Introduction

This document catalogs the creation and results of a classification model to predict the re-admittance rate for hospitals patients. The initial dataset provided was a semi-cleaned list of 10k patients that had been admitted to the hospital. Features include basic information about the patient including their medical background and the admitting physician, a series features for current medications, previous diagnoses, test results and the target of the prediction, whether that patient was readmitted. The framework will be built in python using popular data analytics libraries such as pandas, numpy, sklearn and matplotlib.

We begin the process by examining the provided data set and apply several preprocessing techniques to clean the data and interpolate missing values when appropriate. Next we begin exploratory analysis by producing plots of several key feature points to understand the relationships constituent in the data. To further explore these relationships, we will perform pattern mining and association rule learning specifically on the factors that contribute to a patient dying or being readmitted. Finally, we will attempt to perform some predictive analysis by building a support vector machine model as well as a logistic regression model both trained on the cleaned data set. We will evaluate both learning algorithms with several training:testing splits, providing several metrics to gauge the accuracy and fit of the models, ultimately providing graphs to summarize our success.

All code and data files used throughout this course of this report may be found at the following repo: [github/predictingHospitalReadmissions](https://github.com/dkStephanos/predictingHospitalReadmissions).

Preprocessing

# Checking and Reading the Data

We begin by reading in the raw data file containing entries for ten thousand patients. Initially we want to test this sample to determine the overall health of the data, determining how much data is present, and reflect believable values. To begin, we had to determine what columns had missing values. The data was not consistent in how it represented this and would vary between leaving cells blank or recording ‘?’ or ‘None’, etc. As a result, the first thing we did was replace all those values with Nan and then determine counts per column. Here are columns with a significant count of missing values:

# Dealing with Missing Values

Looking at these counts, columns like *weight*, *payer\_code*, *medical\_specialty*, *max\_glu\_serum* and *A1Cresult* should be dropped immediately. Thousands of missing examples just aren’t recoverable. For race, we just set missing values to the Other class. We decided to keep missing values for *diag\_1*, *diag\_2* and *diag\_3*, as an absence in this field could serve predictive value. We chose, however, to drop the diagnosis descriptions, as they would have required natural language processing to contribute and that was beyond the scope of this project.

The remaining columns all have more than 85% of values present, so attempts may be made to overcome those gaps. To do this, we used mode-based imputation on the missing values. Since these fields were categorical, we felt it was best to just assume the most frequently occurring type for each of these columns. Finally, all rows containing numeric features that were outliers outside a standard deviation of 4 were dropped. Ultimately, our resulting data contained **9646** examples with **44** features each. Some missing fields were represented in the remaining dataset, but otherwise, no missing values remain. Below is a flowchart summarizing this process:

Exploratory Analytics

# Data Visualization

Once our data is clean, the next step is to beginning plotting various features to derive patterns. Since our priority is to predict whether a patient will be readmitted to the hospital, we will prioritize data plots related to that information.

## Patients Readmitted vs Non-Readmitted:

## Rate of Patients Currently on Diabetes Medication:

## Breakdown of Patients by Race:

## Breakdown of Patients by Gender:

## Breakdown of Patients by Age:

## Percentage of Deceased Patients:

## Analyzing Visualizations

Overall by looking at the breakdown of our patient data in multiple dimension, we can get a feel for the skewness, or how much the data reflects reality. Based on the above graphs, we can see for example the majority of our patients are female, perhaps a bit more than we might expect for an average sample. The amount of patients readmitted looks good, having a healthy amount of each in the data set will allow our predictive model to perform better. We are noticeably skewed towards older Caucasians on some form a diabetic medication. This could be problematic, but more than likely just represents the natural distribution found in the problem domain.

# Pattern Mining

The next step in our analysis is to look for patterns and associations that are prevalent within the data. To do this, we will use the apryori algorithm.