Predicting early readmissions for diabetes patients using clinical records from 130 US hospitals

Part A. Problem statement

In 2012, patients with diabetes incurred approximately $124 billion in annual expenditure for hospital care in the United States. Although diabetic patients represent about 8% of the US population, they account for 23% of hospitalizations (8.8 million) each year. Among all the healthcare utilizations, hospitalization is major cost driver for diabetes patients. Readmission within 30 days after a hospital discharge is a high-priority healthcare quality measure and target for cost reduction. It is a more severe situation among patients admitted for diabetes as the rate in those patients is 14.4–21.0%[1], as compared to the overall early readmission rate, which is 8.5–13.5%[2,3]. Therefore, this study aims to build a machine learning model to precisely predict the likelihood of early readmissions among inpatient diabetes patients using a large hospital database from 1999 to 2008.

Who might care?

Physicians, healthcare researchers, and health plan insurance companies might be interested in using this model to predict the readmission outcome for diabetic patients. Based on the predicted probability of early readmission, early interventions can be done to patients such as discharge education, further lab testing to ensure stabilized glucose level, and home care reminders of medications.

Data Source

The hospital dataset for this study was originally obtained from the Health Facts database, which was an extract representing 10 years (1999–2008) of clinical care at 130 hospitals and

---

[1] Rubin, D. J., E. Handorf, and M. McDonnell. "Predicting early readmission risk among hospitalized patients with diabetes (7796)." *ENDO* (2013): 95th.
[2] “Hospital Readmissions in Pennsylvania 2010”. Pennsylvania Health Care Cost Containment Council. Accessed 21/06/2018
[3] Friedman, Bernard, H. Joanna Jiang, and Anne Elixhauser. "Costly hospital readmissions and complex chronic illness." INQUIRY: The Journal of Health Care Organization, Provision, and Financing 45.4 (2008): 408-421.

integrated delivery networks throughout the United States. It was later adjusted to suit for a diabetes research and was released on behalf of the Center for Clinical and Translational Research, Virginia Commonwealth University on UCI Repository[4]. The adjusted data contains 100000 of encounters with 55 features representing patient and hospital characteristics. All encounters in the dataset are for diabetes admissions.

Modeling approach

Since we want to predict the likelihood of an early readmission, a supervised classification algorithm is a perfect choice to build the predictive model. The classification algorithm not only classifies classes (in our case, two classes: “early-readmitted” and “not early-readmitted”) but also predict the probability of each class. This data set is supposed to have imbalanced classes (only less than 10% of the patients will be re-admitted), which makes the project challenging. We plan to use different approaches to tackle the imbalanced class issue and choose the best one. Also, we will try various classification algorithms and pick the one which performs best.

Part B. Description of the dataset

There are totally 50 features and 101,766 observations in the original csv file. Of those features, 2 are the identification ids including unique numbers of medical encounters and corresponding individuals; 5 features are related to patient demographic characteristics such as race and gender; 3 features inform the admission and discharge status; 7 features are numeric values of events (e.g., lab tests, procedures) and time during inpatient stay; 24 features described the use of 24 specific diabetes medications; other features include the change of medication use. The target outcome is re-admitted or not.

After checking each feature including their data types (numerical or categorical), number of missing observations, and counts of unique values for categorical variables, several steps were conducted to clean and process the data for later analysis.

[4] Beata Strack, Jonathan P. DeShazo, Chris Gennings, Juan L. Olmo, Sebastian Ventura, Krzysztof J. Cios, and John N. Clore, “Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records,” BioMed Research International, vol. 2014, Article ID 781670, 11 pages, 2014.

Data Wrangling Steps

1. Missing Values

Before using the Pandas library to explore the metadata for each column, several variables are filtered out using Excel due to too many missing values:

- Weight (numeric; 97% values missing);
- Payer code (17 categories; 40% values missing); and

2. Outliers

Majority of the features are categorical and no extreme values were seen for numeric variables.

3. Duplicate Patients and Deaths

Each encounter id is unique but there are duplicate patient ids so the same patient may have multiple encounters in the dataset. Using results of same individuals tend to bias the true association so duplicates were dropped. Additionally, patients who were discharged to hospice or discharged to a death were also deleted. Finally, 69,970 encounters were left for the study.

4. Transformation of Categorical Features

- Race: Caucasian, African American, Hispanic, Other, and Missing (dummy features created);
- Gender: Female and Male (dummy feature created);
- Age: each class is a range of 10 years (numerical mean age replaced the range);
- Admission_type & Admission_source: overlapped information; Admission_source was re-grouped as “emergency”, “referral”, and “other”;
- Discharge_disposition: was grouped as “discharged to home” and “other”;
- Diagnostic codes: CCI score may be helpful to translate this variable into clinical characteristics of the patients
- Glucose test and A1C test: despite original categories, dummy variables indicating whether the test was performed or not were created
- 24 features of medication: replaced by the feature of diabetes medication (Yes/No)
- Readmitted (Target): No, (readmitted)>30, (readmitted)<30

5. Further Process

Based on the ICD-9 diagnostic codes in the dataset, I created the Charlson Comorbidity Score (CCI) to take into account the clinical characteristics (disease severity) of the patients. It will apply the grouping and mapping of ICD-9 codes to a CCI scores.

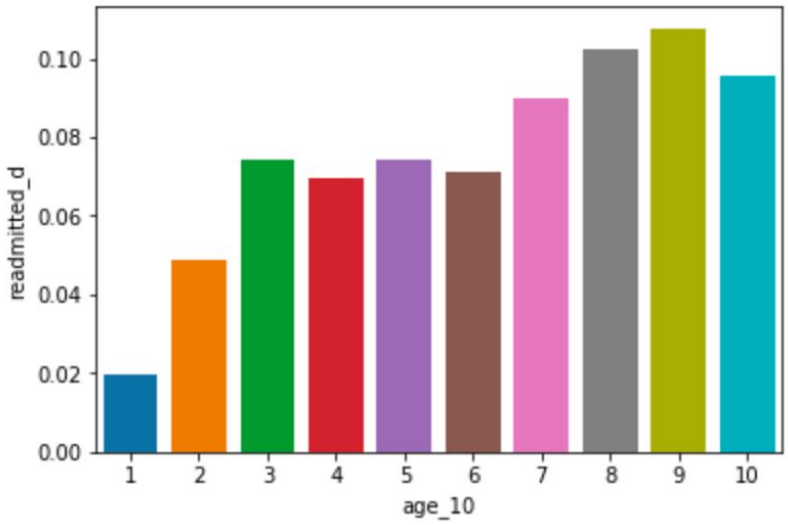
Part C. Initial findings from exploratory analysis

After the data wrangling process, there are 25 columns (24 predictors and 1 outcome) left for the exploratory data analysis. Most features are categorical and have been transformed to be numerical dummy variables.

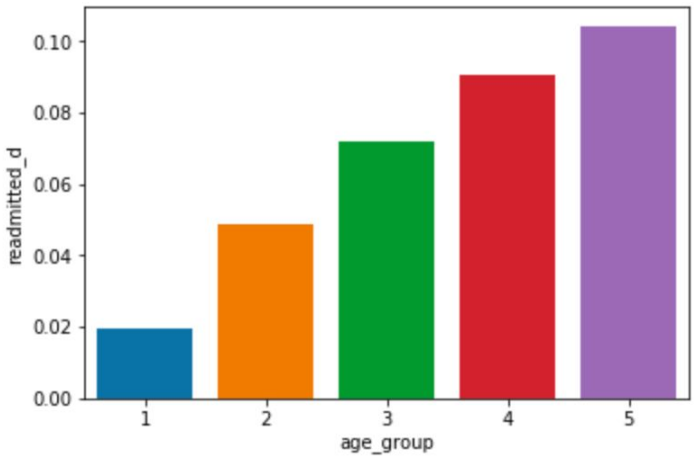
Exploratory Data Analysis with Plots

To get a preliminary sense and how the data looks like, especially how the outcome varies among different classes of each feature, bar plots for percentages of early readmission (1 if readmitted within 30 days) among features were generated. Below are two sample plots.

Figure 1. Example of Early Readmission Rates among Different Age Groups



According to the distribution of readmission rates, I further grouped several age categories together if they have similar rates. And after grouping, the readmission barplot looks like this:



Same grouping was applied to other features if needed.

Inferential Statistics

Since I have different types of independent variables, I used different statistical tests. For all tests, the general null hypothesis is that there is no relationship between the feature and early-readmission. All features were separated into dummy features, categorical features, and continuous features based on their data type. Since the samples in this dataset should be randomly selected, the number of success or failure is large enough (>10), and samples are independent of each other, we can assume the conditions of randomness, normal approximation, and independence are met, and many tests can be applied.

- **Dummy features**

2 sample proportion z-tests were applied using the *proportions_ztest* from statsmodels.stats.proportion. 2 sample Z-test for proportion was chosen since the standard deviation of the true proportion was known (unbiasedly estimated by the pooled standard deviation).

- **Categorical features**

For categorical features with multiple classes, chi-squared test was chosen since we can compare if distribution of the observed counts in each class is significantly different than the expected counts (proportional to class counts).

- **Continuous features**

To test for significance impact of continuous features on binary outcomes, *Logit* model from *statsmodels.discrete.discrete_model* was applied.

Table 1. Features and their p-values

Features	p-values	Features	p-values
race_?	0.012	glipizide_Down	0.007
race_Caucasian	0.001	glipizide_No	0.006
race_Other	0.012	dischar_group	0.000
max_glu_dummy	0.014	ad_type_group	0.021
A1Cresult_dummy	0.011	med_spec_group	0.000
diabetesMed_d	0.000	admis_source_group	0.006
Alc_nonch	0.008	diag_group	0.000
med_up	0.004	time_in_hospital	0.000
med_down	0.000	num_lab_procedures	0.000
insulin_Down	0.000	num_procedures	0.000
insulin_No	0.000	num_medications	0.000
insulin_Up	0.008	number_emergency	0.008
metformin_No	0.001	number_inpatient	0.000
metformin_Steady	0.003	number_diagnoses	0.000
metformin_Up	0.040	CCI	0.000

All statistical significant features with p value less than 0.05 were included in a list and will be further compared and analyzed in the machine learning models.

Summary

To precisely predict the early readmission outcome, machine learning techniques using Scikit Learn will be further implemented. We can see for now that several features are significantly correlated with the outcomes in the multivariate logistic regression model.

Part D. Machine Learning Process

Since the target value is 1 (early readmission) vs 0 (not early-readmitted), a classifier is needed to train the data and predict the binary outcome. To predict 1/0 for early readmission,

classification machine learning models can be applied. In case the imbalanced data would affect the model performance especially in this study we are more interested in the minority outcome, over sampling will be applied.

Metrics selection

Considering the results in a real-world practice, people in healthcare industries would prefer a prediction model that can identify most of the early-readmission cases, and tolerate some extent of low prediction score. For example, if a prediction model can identify over half of all the early readmitted patients with a 0.1 prediction score, this model would be better than another model that has a 0.9 prediction score with low capability (say, only 0.01 of recall) to identify early readmitted patients.

Therefore, AUC score will be used to train models and recall will be evaluated among different probability thresholds

Machine learning pipeline

In a machine learning task, it is very common to set up a customized processing pipeline that can facilitate the task. The pipeline can starts from feature extraction/preprocessing, proceeds to feeding features into the machine learning algorithms, and ends with model evaluation with specified metrics. A machine learning pipeline can be simply set up with scikit learn's *sklearn.pipeline* modules, but sometimes a self-defined pipeline function can also works especially there is some step that is hard to fit in to the *sklearn.pipeline* settings.

I am essentially interested in if polynomial terms and resampling (oversampling) can be beneficial, so to serve for my goal, I wrote a function that can allow me to apply polynomial and resampling as an option/argument when processing the data, and also to apply model fitting and evaluation.

Applying different machine learning models

With the pipeline settled, it is very easy to try different models without worrying about transforming the data. Among all the classifiers, I picked the Decision Tree technique as my first simple classifier for the following reasons:

- It is naturally a good algorithm for mixed type data, which is the case in my project;
- I can obtain the embedded feature importances to select best features;
- It is explainable and cheap to run

Other classifiers like K Nearest Neighbors (KNN), Naive Bayes (NB), Support Vector Machine (SVM), and ensemble approach like Random Forest (RF) were also be tested later.

Cross-validation scores and test score were returned for each model fit. For the first DecisionTree model, the results are as followed:

CV AUC Scores: 0.66091814, 0.64988475, 0.63991324, 0.64551209, 0.63448754
Test AUC Score: 0.60903298

RandomForest did better as both the CV AUC and test AUC increased. But for both classifier, test score is lower than cv score which means that there is over-fitting.
**It is weird to see that there is a big gap between cross-validation scores and test score. One reason for that could be the oversampling approach applied to address the minority group issue. So for the logistic regression model I just used the original training sample without oversampling. Again, there is still the gap between CV and test scores so it remains a question for this project.

Parameter Tuning

Parameter tuning was applied for the logistic regression and Xgboost regression models with the `GridSearchCV` approach.

Exploring the threshold

Since we are more interested in identifying patients that would be early readmitted, we can explore the probability threshold of the model to see if we can have more early readmission included without reducing too much AUC value. Some threshold value with their AUC and confusion matrix are shown below:

******** For i = 0.46 ******
Our AUC is 0.614710737177228
[[9005 6930]
 [523 1035]]

******** For i = 0.47 ******

Our AUC is 0.619523795522004
[[9455 6480]
 [552 1006]]

******** For i = 0.48 ******
Our AUC is 0.6223100263304914
[[9912 6023]
 [588 970]]

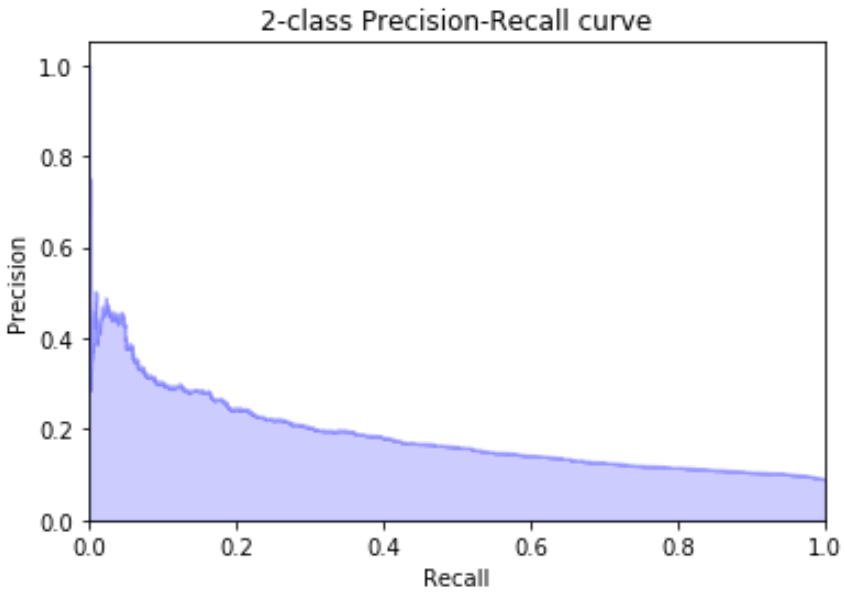
******** For i = 0.49 ******
Our AUC is 0.6212209380776285
[[10358 5577]
 [635   923]]

******** For i = 0.5 ******
Our AUC is 0.6214298459764939
[[10784 5151]
 [676   882]]

******** For i = 0.51 ******
Our AUC is 0.6199400001530608
[[11207 4728]
 [722   836]]

******** For i = 0.52 ******
Our AUC is 0.623480881292059
[[11596 4339]
 [749   809]]

It is obvious that if the threshold is lower than 0.5, than more patients will be predicted as early readmission and recall will be higher. The Precision-Recall curve is shown below:



We can see that even the recall is 0, the precision cannot be improved much. So we can bear to lower the precision, which means we can bear with predicting more patients as early-readmitted who actually would not, so that the recall will be higher.

Conclusions

If we can bear lower AUC score or we don't care about it at all, we can use a lower threshold (<0.5), e.g., a threshold of 0.45 would identify around 2/3 of the early-readmitted patients with this model; among all the patients predicted with this model as early-readmitted in the future, 7/8 of them will actually not be early-readmitted and further screening process can be applied to these patients to filter out the right person that will have early-readmission.