Predicting Diabetes Patient Hospital Readmission

Milestone Report

Pavel Zimin, PhD

Problem Statement

Hospital readmission is a highly preventable cause for high healthcare costs. In fact, hospitals are financially punished if they have higher readmission rate than the “expected” level. It is important for the hospitals to be able to predict which population of patients are at a higher risk for being readmitted. This knowledge will help prioritize patients that will most likely benefit from hospital discharge follow-up programs.

This analysis will be helpful to the hospital healthcare teams with prioritizing patient support program. Perhaps most importantly this analysis will benefit patients who will receive improved health care, decreased chances of readmission while incurring smaller cost.

The data that will be used for this project has been published in UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/datasets/diabetes+130-us+hospitals+for+years+1999-2008). The data have been collected for 10 years (1999-2008) from 130 hospitals located throughout the U.S. The data were initially obtained from a large clinical dataset which was filtered for such criteria as inpatient encounter (admitted to the hospital), diabetes diagnosis codes, the length of stay (in the range of 1-14 days), laboratory tests (were performed during the encounter) and administration of medications during an encounter (see the original publication by Strack et al. for further details). After applying these filters the dataset consists of 101,766 encounters (observations) with 55 attributes. Attributes include demographic data, medications, diagnostic results and information regarding their hospital stay. The outcome for this dataset is whether the patient was readmitted within 30 days from discharge.

This project is a classification problem. I will try multiple algorithms and will select the one that more accurately predicts the hospital readmission.

At the conclusion of this project I plan to produce a Jupyter Notebook containing code, narrative explanation, and a slide deck.

Description of the dataset and cleaning steps

The hospital readmission data consist of 2 files: the first file contains the data for the hospital encounters, their attributes and the hospital readmission outcomes, the second file contains mappings for 3 columns in the first file. Data were loaded in pandas DataFrame. Initial inspection revealed that there are 101,766 observations of unique hospital encounters with 50 variables: 13 columns of integer type, 37 columns of object type. The columns with the patient weights was eliminated from further analysis since it contained too many missing values. The dataset includes 3 columns that contain ICD9 codes (International Classification of Diseases, version 9) for primary diagnosis, secondary diagnosis and additional secondary diagnosis. It could be beneficial to have more general labels for the diagnoses since they could potentially be more predictive for hospital readmission than the very narrow diagnosis. In order to group diagnoses into larger groups on multiple levels, the script and data in the json format from the GitHub repository was used (https://github.com/sirrice/icd9.git). The json file was read and converted to a pandas DataFrame, then merged with the hospital readmission dataset providing 3 levels of grouping for each ICD9 code. This procedure resulted in additional 9 columns (3 levels for each of 3 original columns with ICD9 codes).

Missing values were encoded with various symbols: “?”, “None”, “NULL” and others. All these encodings were replaced by NaN values from numpy. All object types columns were converted to categorical type and replaced with integers. Missing values were coded as 0. The mapping of categorical labels to integer values were saved in a dictionary. This will allow an easy conversion to human-readable lables later in the analysis. In addition, for each column containing missing values a new column was created with values of 1 for a missing value and values of 0 otherwise.

To detect outliers 2 method were tried: 1) to detect univariate outliers, values below 25 percentile - 1.5 IQR (interquartile range) and values above 75 percentile + 1.5 IQR; 2) to detect multivariate outliers, principal component analysis (PCA) was used. First method produced outliers in many columns, however it did not make much sense for categorical columns. Some integer columns also produced outliers by this method. For example, times in hospital of greater than 12 days were outliers, number of diagnoses in the system greater than 13 were outliers. After careful consideration I decided not to remove these outliers since these values could be predictive of hospital readmission. PCA with 2 and 3 components was used to identify multivariate outliers since this method allows to reduce the dimensions of the original data and plot the principal components as a scatter plot in 2 dimensional space. Plotting either components of 2 component PCA or combinations of 2 components of 3 component PCA failed to reveal outliers that are grossly separated from the rest of the data points. Therefore I decided to keep data of all observations without removing the outliers.

Exploratory Data Analysis (EDA)

What variables are the most significant for predicting the readmitted variable?

To answer that question I created a cluster map of Pearson correlation coefficients (Figure 1). Due to the large number of variables I selected only the variables that have a higher degree of positive or negative correlation with the readmitted variable.

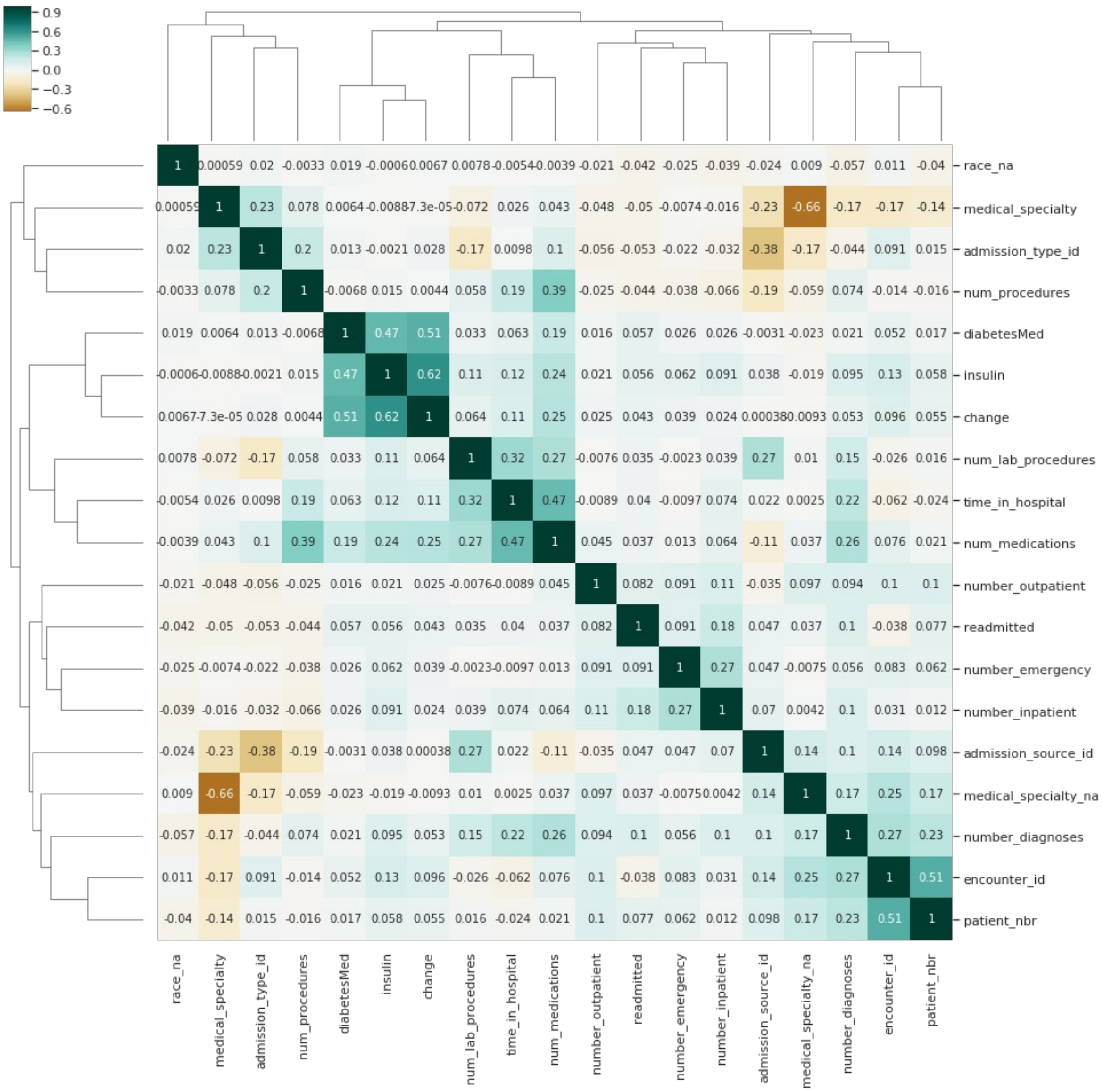

Figure 1. Correlation clastermap showing variables with the highest positive and negative correlations to the target variable.

Overall there is moderate to low correlation between the readmitted variable and other variables. The variables showing largest correlation are: number_inpatient, number_emergency, and number_outpatient. In the next steps I investigated their relationship with the dependent variable further.

Additional EDA of the variables correlated with the target variable

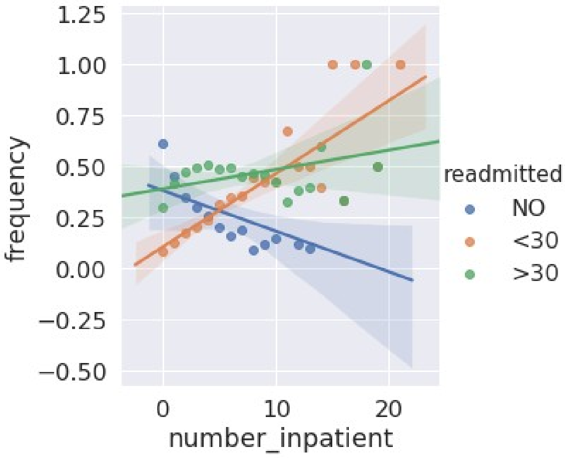

Figure 2. Frequencies of a number of innpatient visits in a previous year split by the values of readmitted column.

- Figure 2 shows the plot of the number of inpatient visits in the year preceding the encounter (number_inpatient) vs. its frequency split by the values of readmitted variable. It appears from the plot that there is a strong correlation between the number of inpatient visits and its frequency in the category of patients readmitted within 30 days (the Pearson correlation coefficient was 0.46). The slope of that relationship is 0.018. To test if we are confident that the correlation coefficient of the entire population is different from zero, I performed hypothesis testing using bootstrapping approach. One of the important advantages of the bootstrapping method is no requirements for the distribution. The null hypothesis was that there was no correlation between the number of inpatient visits and its frequency. The alternative hypothesis was that the correlation coefficient is different from zero. I selected the significance level for this and for further tests as 0.01. To simulate the assumption of the null hypothesis I permutated the order of number_inpatient and calculated the Pearson correlation coefficient of that permutated number_inpatient variable vs. original frequency. The expected value of that correlation coefficient is zero. I performed this procedure 10,000 times thereby obtaining 10,000 correlation coefficients. Then, I calculated the fraction of cases that produced the correlation coefficient whose absolute vvalue is equal or greater than the observed correlation coefficient. This fraction is the P-value that in this case was 0.0003 which is smaller than our previously set significance level. The interpretation of these findings is that as the number of inpatient visits in the year preceding the encounter increases the chances that the patien will be readmitted within 30 days of hospital discharge also increase. The slope of 0.018 indicates that for every additional inpatient visit during the preceding year the chances of being readmitted within 30 days increase by approximately 0.018.

- Next I plotted the number of emergency visits in the preceding year vs its frequency split by the readmitted group (Figure 3). The slope of this relationship is 0.013 with the observed Pearson correlation coefficient of 0.71 with the P-value of 0.0 calculated from the bootstrapping sample of 10,000. This is indicates that the actual P-value is close to 0.0001 or below which is smaller than our preselected significance level of 0.01. Therefore, I am confident that the Pearson correlation coefficient is different from zero. Further increasing the size of the bootstrapping sample would allow to estimate the P-value more precisely. My interpretation is that for every increase in the number of emergency visits in the preceding year there is an increase in the chances of being readmitted within 30 days of approximately 0.013.

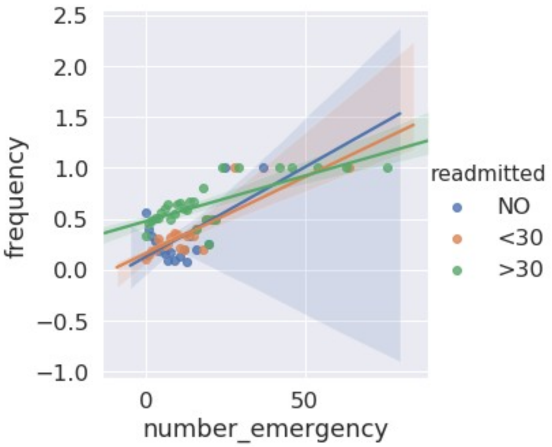

Figure 3. Frequencies of a number of emergency visits in a previous year split by the values of readmitted column.

- Next I explored the relationship of the number of outpatient visits in the year preceding the encounter to its frequency split by the readmitted category (Figure 4). The slope of the linear function was calculated to be 0.014, the Pearson correlation coefficient: 0.64 with the P-value calculated by the bootstrapping approach with 10,000 samples of 0.0. This indicates that for each additional outpatient visit during the year

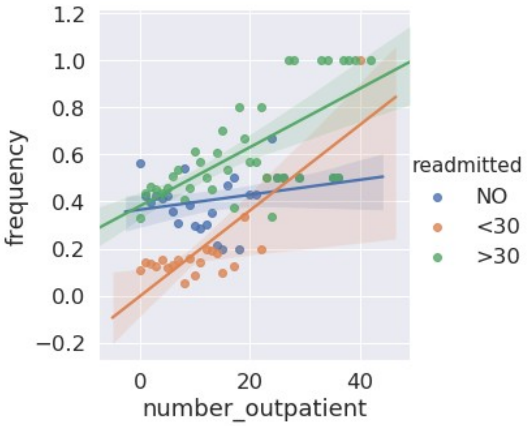

Figure 4. Frequencies of a number of outpatient visits in a previous year split by the values of readmitted column.

preceding the encounter there is an increase in chances of being readmitted within 30 days by approximately 0.014.

What pairs of independent variables show strong correlations?

These relationships help us figure out if one of the 2 variables could be removed from the model to help with reducing the number of dimensions. Another benefit of knowing these correlations is the sanity check of the dataset. For example, we can predict that the duration of a hospital stay would correlate with the number of lab procedures performed. If we find out that our data set does not show this property, it would raise serious concerns about the quality of the dataset. Otherwise, if we see the expected correlations, it increases our assurance about the dataset.

- One of the strongest correlations observed from the correlation plot is between the change of medications and the insulin feature. The plot in the Figure 5 shows the heatmap of the counts of the change and insulin variables. The computed Pearson's correlation coefficient for these two variables is 0.62. It is clear that all hospital encounters with the insulin levels of "Up" or "Down" cannot have change levels of "No". This is exactly what is confirmed in this plot as a sanity check. The encounters that did not change insulin regimen could have changes in other medications and therefore may be distributed between "No" and "Ch" values of change variable.

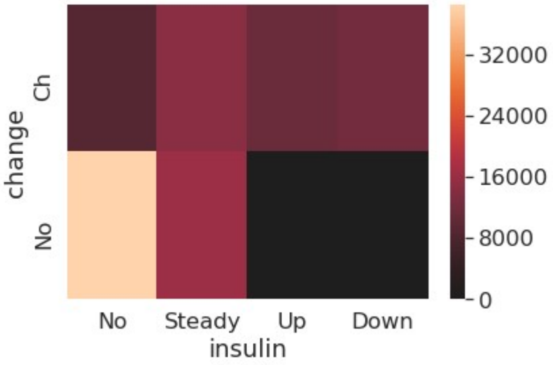

Figure 5. Correlation heatmap between medication change variable and an insulin dosage change.

- The count heatmap for number of medications versus number of days between admission and discharge shows that these variables are correlated (Figure 6). This finding serves as a sanity check since it is expected that the longer hospital stay

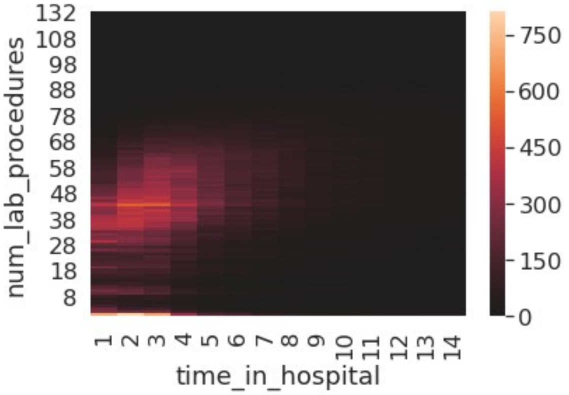

Figure 6. Correlation heatmap between the duration of hospital stay and the number of lab procedures.

would be associated with the higher number of medications.

- The number of lab tests performed during the encounter is also correlated with the number of days between admission and discharge (Figure 7).

These findings confirm our expectations and provide an additional support for the validity of the dataset.

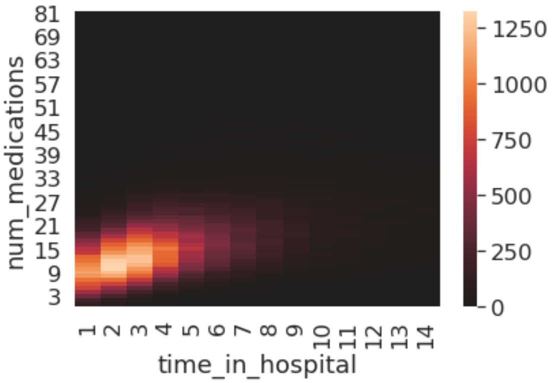

Figure 7. Correlation heatmap between the duration of hospital stay and the number of medications.

Is there a relationship between the age of the patient and the frequency of readmission within 30 days?

Figure 8 shows that the patients below 20 years old have the lowest frequencies of readmission within 30 days, all other age groups show higher frequencies. The age group with the highest frequency for the readmission is 20-30 years.

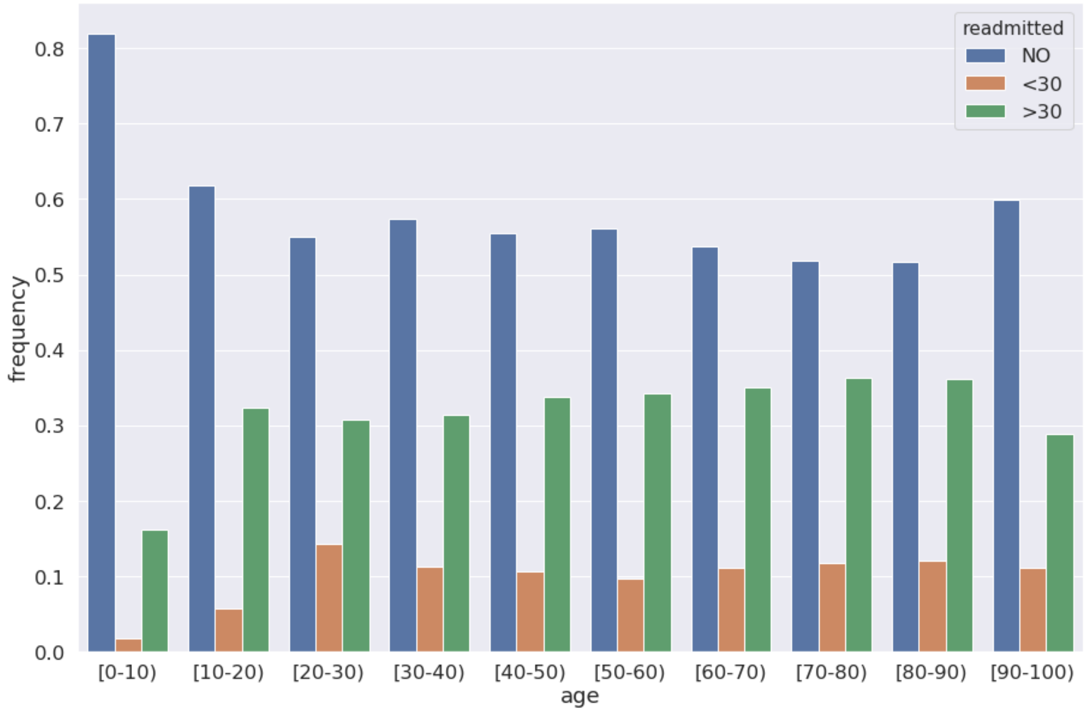

Figure 8. Distribution of frequencies of age groups split by the values of the readmitted variable.

How the changes in medications are associated with readmitted variable?

During their hospital encounter the patients' prescription medication doses were documented and reflected in the data setas either not prescribed, dose did not change, dose increased or dose decreased. The next question that I asked: is there a relationship between dose changes and the frequencies of being readmitted within 30 days? I found that the most dramatic changes in those frequencies were for the following medications for treating diabetes: repaglinide, nateglinide, and acarbose. The changes in the doses of these medications have differential effects on readmission frequency (Figure 9).

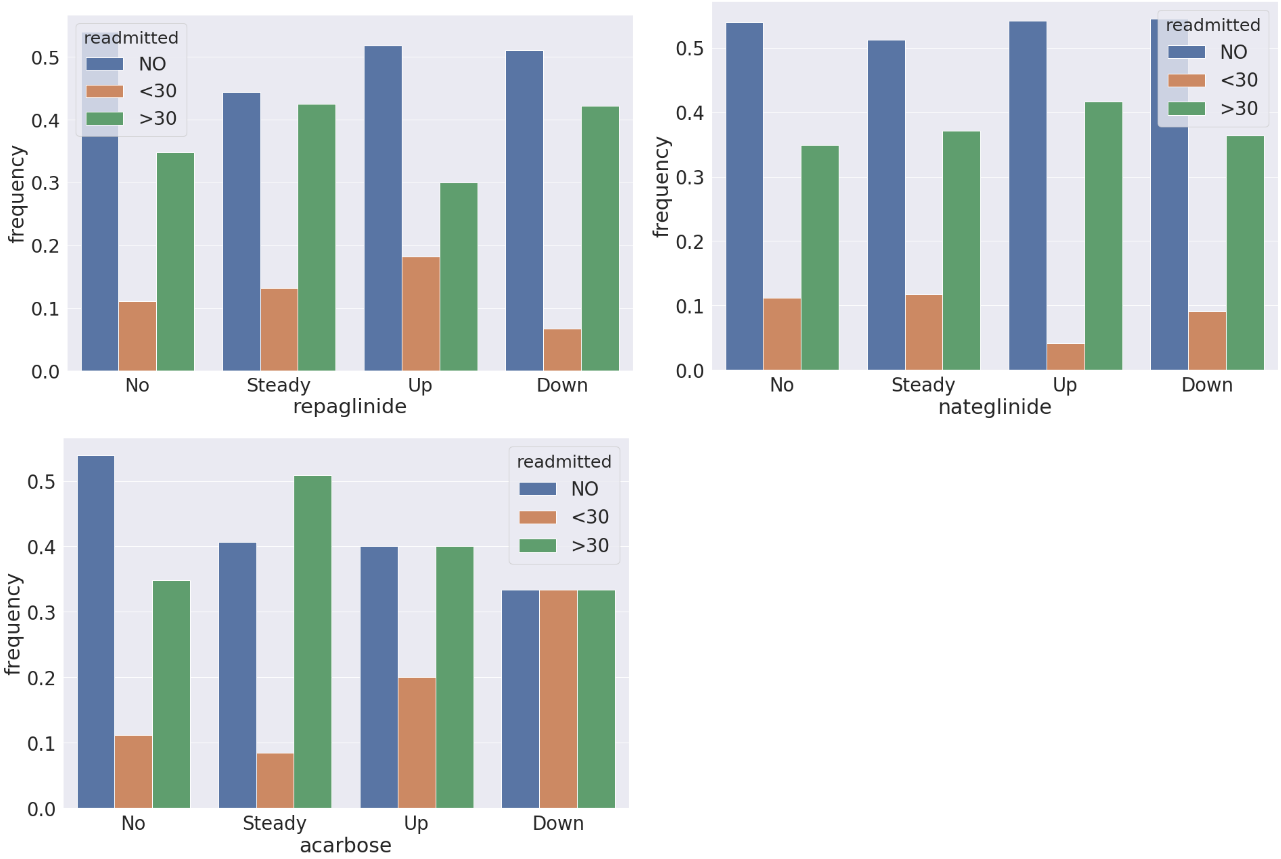

Figure 9. Distribution of frequencies of medication dosage changes split by the values of the readmitted variable.

- Specifically, patients with increased dosage of repaglinide are nearly twice as likely to by readmitted within 30 days than patients without prescription or patients with decreased dosage. These data could indicate that repaglinide could be toxic and an increased dose leads to the hospital readmission. Another possible explanation is that repaglinide is prescribed as a drug of last resort and its increased dosage is just an indication that the patient is severely sick.

- Patients with increase dosage of nateglinide decreased the frequency of being readmitted within 30 days by more than half in comparison to patients who did not change their dosage or who were not prescribed nateglinide. One possible interpretation is that an increase in nateglinide dosage could be reducing the frequency of hospital readmission by its increased therapeutic effect.
- Changes in medication acarbose have totally different effect: an increase or decrease of the dose led to the increase in the frequency of hospital readmission within 30 days in comparison to the patients who were not prescribed or patients who did not change their dosage of acarbose. Interpretation of acarbose data is more of a challenge. Acarbose could require more gradual changes in its dosages and drastic increase or decrease could lead to toxic effects by the same or different mechanisms.

In order to add more weight to these interpretations additional analyses are needed.

Are there primary diagnoses that are associated with higher chances of being readmitted within 30 days?

The box plot of distribution of frequencies by primary diagnosis (Figure 10) shows that frequencies of hospital encounters with readmission within 30 days are generally in the low range in comparison to not readmitted group. However there are primary diagnoses that show much higher frequency of readmission within 30 days. The primary diagnoses with the highest frequencies are 1) encounter for other and unspecified procedures and aftercare, 2) diabetes with renal manifestations, and 3) peritonitis and retroperitoneal infections.

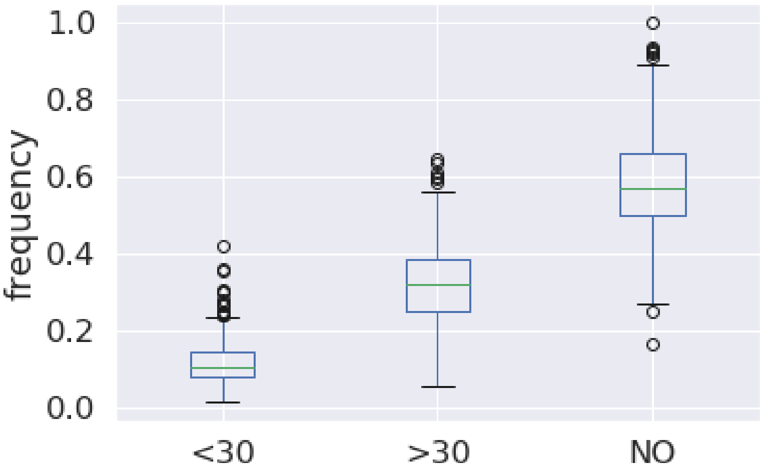

Figure 10. Distribution of frequencies of the primary diagnoses split by the readmitted variable.

Interestingly, patients without primary diagnoses (labeled as 1 in Figure 11) are twice as likely to be readmitted within 30 days than patients with primary diagnoses (labeled as 0). This could indicate that the lack of primary diagnosis is not independent of readmission status, e.g. if the diagnosis is not known, the specific treatment is not applied and as a result a patient does not get better which results in readmission within 30 days. However, this relationship would require further investigation.

Conclusions

In conclusion, Exploratory Data Analysis showed that overall there is a moderate to low correlation between the readmitted variable and other variables. The variables showing the largest correlations are: number_inpatient, number_emergency, and number_outpatient. In addition, the most dramatic changes in the frequencies of medication changes were for the following medications for treating diabetes: repaglinide, nateglinide, and acarbose. Distribution of the primary diagnoses shows that for some primary diagnoses the frequency of readmission within 30 days is much higher than the median frequency for that group. The top primary diagnoses in the group with readmission within 30 days are 1) encounter for other and unspecified procedures and aftercare, 2) diabetes with renal manifestations, and 3) peritonitis and retroperitoneal infections.

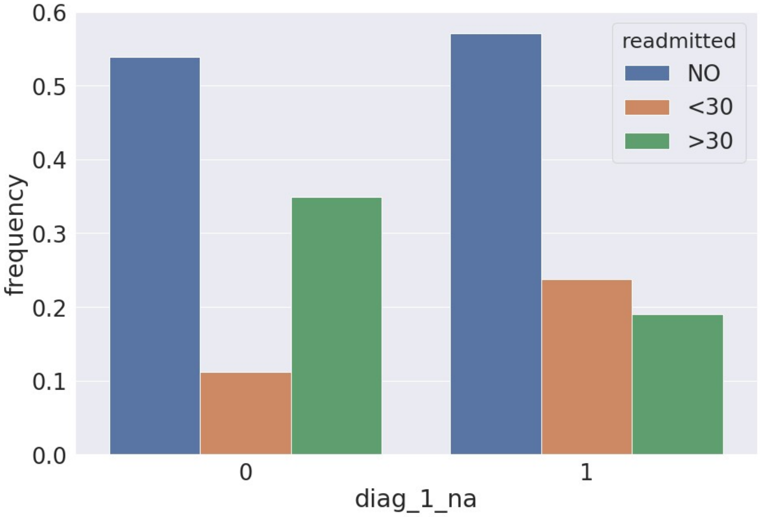

Figure 11. Distribution of frequencies of the lack of primary diagnosis split by the readmitted variable. Groups with missing primary diagnosis are labeled 1, groups with the primary diagnosis are labeled 0.

Machine Learning

Data splitting

To prepare the data for the machine learning the columns that are not useful for prediction ('encounter_id', 'patient_nbr', 'payer_code') were deleted. For model tuning data were split into a training set (70% of the data) and three hold-out sets (10% each). Hold-out set 1 was used for tuning the hyperparameters. Hold-out set 2 was used for validating the model. Hold-out set 3 was used for the final model test.

Dealing with the Imbalanced Data

The data set is highly inbalanced with the positive class representing only about 11% of all data. This presents a potential issue with training the classification models since they would tend to perform better on predictive the negative class. In fact, the fitted random forest model with default hyperparameters showed excellent scores on the training data, while performing much worse on the validation data. This is an indication of overfitting. To deal with that, the model needs to be tuned by adjusting the hyperparameters and validating on an unseen data set. Also the model works much better on predicting the negative class than predicting the positive class. This is likely due to the fact that the data is imbalanced towards the negative class. To account for that, I will be using the F1 score as a metrics for the hyperparameter tuning. Other available metrics for dealing with imbalanced data are: precision, recall, F beta, mattews correlation coefficient. Optimizing the F1 score will help to maintain a balance between precision and recall.

Three methods for dealing with the imbalanced data were tested: random undersampling (the negative class was randomly sampled to match size of the postive class), the Synthetic Minority Oversampling Technique (SMOTE) and the Adaptive Synthetic sampling (ADASYN). SMOTE and ADASYN methods generate new samples by interpolation. All three methods are implemented in imbalanced-learn. The F1 score of random forest model trained on the undersampled data was 0.22. This is double of the F1 scores produced by the random forest model trained on the oversampling data (both produced the F1 scores of about 0.11). Hence the random undersampling was selected to deal with imbalanced data and was used for training all remaining models.

Model tuning

To evaluate a model, the function was created that calculates precision score, recall score, F1 score, F beta score, mattews correlation coefficient, accuracy score and the out-ofbag score for random forest models. These scores are reported for the training and the test data sets. In addition, this function plots confusion matrices for prediction on the training data and the validation data. F1 score on predicting the validation data was used for tuning hyperparameters.

The following models were selected for training:

1) **Dummy model.** To establish a benchmark for the machine learning models, the dummy classifier was created using stratified strategy. The F1 score of this model on the validation set was 0.11.

2) **Random Forest.** Random Forest is a collection of decision trees that are fitted on the bootstrapped sample and using a subset of features. Random Forest is an efficient algorithm and runs efficiently. Here, Random Forest was fitted on the training set with different combinations of the hyperparameters: n_estimators in the range of [2, 5, 10, 20, 40, 60, 100], max_features in the range of [2, 5, 7, 10, 15, 20, 25, 30], min_samples_splits in the range of [2, 3, 5, 7, 10, 50] and the max_depth in the range of [2, 5, 10, 20, 40, None]. The best F1 score on the hold-out set 1 was 0.27. The metrics on the training data are much better than ones on the hold-out set 1 indicating that the model overfits.

3) **Logistic Regression.** Logistic regression is the most basic algorithm used to predict binary classification. It offers regularisation to deal with the overfitting and make models more generalizable. The data was scaled to make the ranges of features similar. Recursive feature elimination (RFE) available at scikit-learn was used to select the features most useful for training the logistic regression. These features are ['discharge_disposition_id', 'number_outpatient', 'number_emergency', 'number_inpatient', 'number_diagnoses', 'chlorpropamide', 'acarbose', 'miglitol', 'tolazamide', 'simple_diag_1', 'simple_diag_2', 'simple_diag_3', 'level_2_diag_1', 'level_3_diag_1', 'level_4_diag_1', 'level_4_diag_2', 'level_5_diag_2', 'level_3_diag_3', 'level_4_diag_3', 'level_5_diag_3']. Interestingly a large subset of these features was added from the ICD9 data set that provides variaous levels of grouping of the diagnoses from the hospital readmission data set. Logistic regression was fitted for the C_param in the range of [0.001, 0.01, 0.1, 1, 10, 100] and using L1 or L2 penalties. The tuned model's F1 score on the hold-out set 1 was 0.26.

4) **Logistic Regression using stochastic gradient descent.** Since Logistic Regression in scikit-learn does not offer elastic net option for regularisation, stochastic gradient descent was also used for training. The following hyperparameters were optimized: alpha from the range of [0.00001, 0.0001, 0.001,0.01,0.1,1,10,100], penalty functions

['l1', 'l2', 'elasticnet'], and l1_ratio for the elastic net penalty in the range of [.1, .3, .5, .7, .9]. The tuned model's F1 score on the hold-out set 1 was 0.25.

5) **AdaBoost.** AdaBoost uses a weak classifier, and then attempts to improve it by adjusting the weights of incorrectly classified instances. Deceision tree classifier was used as a base classifier for the AdaBoost algorithm. The following hyperparameters were tuned: max_depth in the range of [2, 5, 10, 20, 40, None], min_samples_split in the range of [2, 3, 5, 7, 10, 50], and n_estimators in the range of [2, 5, 10, 20, 40, 60, 100]. The tuned model's F1 score on the hold-out set 1 was 0.27.

6) **Gradient Boosting classifier.** Gradient Boosting classifiers work by iteratively minimizing loss function. Hyperparameter tuned were max_depth in the range of [2, 5, 10, 20, 40, None], min_samples_split in the range of [2, 3, 5, 7, 10, 50], n_estimators in the range of [2, 5, 10, 20, 40, 60, 100], max_features in the range of [2, 5, 7, 10, 15, 20, 25, 30]. The largest F1 score of the model evaluated on the hold-out set 1 was 0.27.

7) **Gradient Boosting classifier fitted to principal components.** In order to simplify the model, Gradient Boosting was applied to the features converted into 3 principal components. The F1 score was 0.20. This places this model about in the middle between the best perfroming model and the dummy model.

The best hyperparameters as judged by the F1 scorre are summarized in Table 1.

Model	Hyperparameters
Dummy	None
Random Forest	n_estimators=20, max_depth=5, min_samples_split=5, max_features=25
Logistic Regression	penalty='l2', solver='liblinear', C=1
Logistic Regression, stochastic	alpha=0.001, penalty='elasticnet', l1_ratio=0.3
AdaBoost	max_depth=2, min_samples_split=2, n_estimators=10
Gradient Boosting	n_estimators=100, max_depth=2, min_samples_split=2, max_features=20

Table 1. Hyperparameters optimized for the F1 score.

Model Selection

To compare these models, the training set and the hold-out set 1 were combined and used for training the models with the tuned hyperparameters. Then these models were tested on the hold-out set 2. The model that showed the highest F1 score on the hold-out set 2 was Gradient Boosting Classification with the F1 score of 0.27 (Figure 12). This model will be used to build the final model.

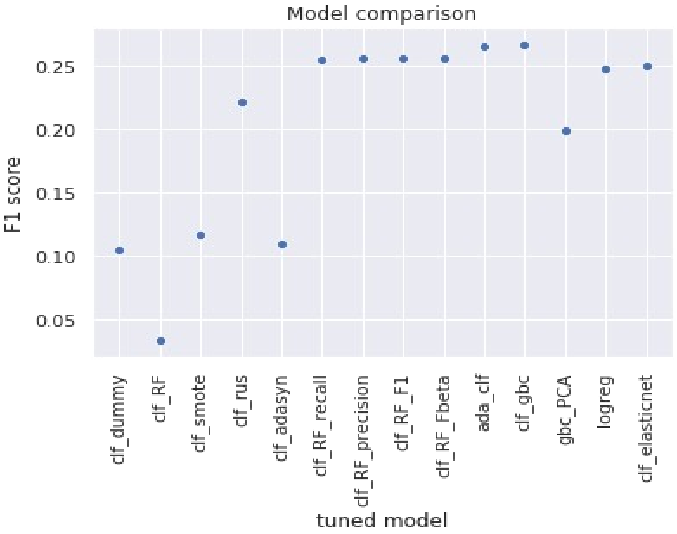

Figure 12. F1 scores of tuned models trained on the hold-out set 2. The abbreviated classifiers are described in the accompanied Jupyter Notebook.

Final Model Evaluation

To create the final model, Gradient Boosting classifier was trained on the data that includes the training set and 2 hold-out sets. This model was tested on up to this point unused data set (hold-out set 3) to estimate how the model would generalize to new data. The F1 score of the final model on the hold-out set 3 (test data) is about 0.26 (see Table 2 and Figure 13). This is slightly below the one obtained from the validation data, The precision score (the fraction of true positive observation out of the total predicted positive) is about 0.17. The recall score (the fraction of true positive observations out of total true positive) is about 0.57. The model's precision score still is very low, and the model is still overfitting.

metric	**training data**	**test data**
precision score	0.64	0.17
recall score	0.57	0.57
F1 score	0.60	0.26
Fβ score	0.63	0.17
Mattews correlation coefficient	0.24	0.16
accuracy score	0.62	0.67

Table 2. Performance metrics of the final model.

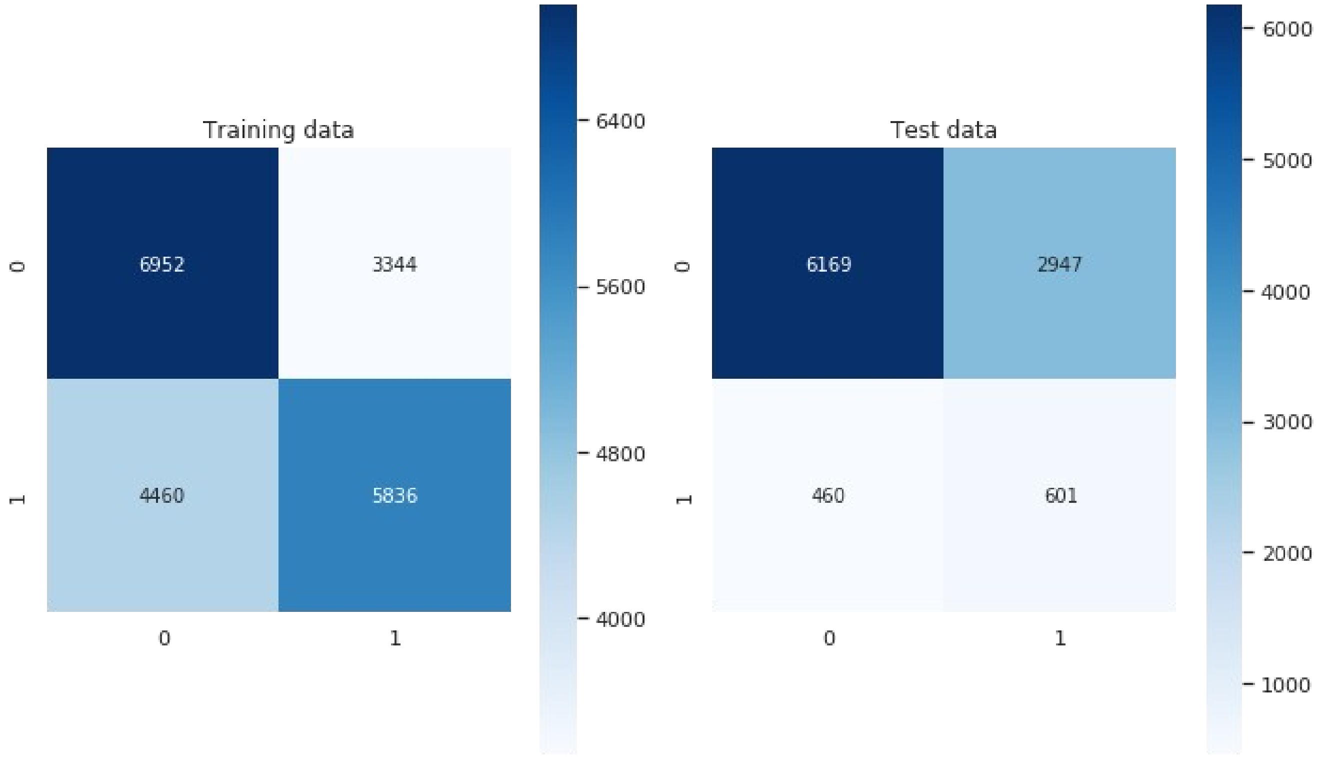

Figure 13. Confusion matrices showing prediction on the extended training data (training data, hold-out set 1 and hold-out set 2) and on the test data.

The important features that played a role in the prediction whether the patient will be readmitted within the 30 days following hospital discharge are number of inpatient visits in a year preceding the encounter, discharge disposition, number of emergency visits in a year preceding the encounter, duration of hospital stay, age.

The model was saved on the hard disk for future use.

Limitations

The model created for this report suffers from overfitting and poor precision. To significantly improve the model, more data is needed for machine learning.

Recommendations

There are 2 main recommendations from this:

1) Use the model as is. Only 17% of the predicted positive are true positive, but the model still allows to narrow down the number of patients that would benefit from the follow-up program aimed at minimazing the chances of readmission. This might result in significant reduction of the 30-day readmission rates. For example, if all patients identified as positive by the model do not return within 30 days of discharge as a result of follow-up program, the 30-

day readmission rate would drop by more than half. This is an upper estimate of the benefit of the current model.

2) If better precision is required, more data need to be collected. Larger number of features would also be helpful for building the model. This will help with building a model with better precision.

Acknowledgments

I would like to thank, Ramkumar Hariharan, for his mentorship that was so helpful for this project.