614 Project
Hospital Readmission Prediction Based on Health Facts Data

Xingyu Zhu
(xzhu76)

Yang Ye
(yye36)

Zhenggang Tan
(ztan11)

Abstract

In this project, our objective is to predict the likelihood of a patient's readmission to the hospital based on an extensive dataset consisting of over 50 features and 100,000 samples. We adopted a comprehensive approach to our research, starting with handling missing values through imputation and dropping columns with excessive NAs. Next, we conducted exploratory data analysis and extensively engineered the features. This included redefining columns with too many categories, aggregating some columns for cumulative counts, combining correlated columns into new features, and transforming categorical columns using one-hot encoding.

After preparing the data, we evaluated several classification models to predict the readmission type. The models we tested included Gaussian naive Bayes, linear discriminant analysis, logistic regression, decision tree, random forest, CatBoost, LightGBM, and XGBoost. Among these models, logistic regression and CatBoost provided the best predictions, with an accuracy of 0.78 and F-1 score around 0.75.

Our project could potentially assist healthcare providers to understand the importance of each factor and implement targeted interventions to prevent readmission, it can also help reduce the cost associated with hospital readmission.

1 Introduction

Background Hospital readmission is an essential indicator for the treatment effect. Generally, a patient who does not readmit after leaving the hospital seems to have a good curative effect. Proper prediction of readmission based on historical care data for patients not only can help hospitals evaluate whether a patient should leave the hospital or not but also can reduce a patient's health risk.

2 Data Preprocessing

Dataset This project used the clinical database of 130 US hospitals and integrated delivery networks from 1999 to 2008, which has 101,766 observations of 50 variables from 71,518 patients. (1)

Setup

- **Handling duplicate patient_nbr** In the dataset, it happens that an individual patient would be recorded more than once for their encounter in the hospital. This makes sense because it is likely that patients would go to the hospital more than once. However, this is no good for later analysis in that such consecutive encounters would be more or less dependent on each other. When predicting a patient's readmission to the hospital, it would make no sense if we treat multiple encounters as independent ones and make decisions on that basis. Rather, what would be more logical to do is to accumulate features for such individuals, since patients with more days in hospital(in a fixed range of time) and more symptoms or diagnoses should have higher risk of being readmitted into the hospital sooner. For all other categorical features, the most up-to-date values should be kept.

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

Thus, we made accumulative variables on the following list of variables: time_in_hospital, number_diagnoses, num_medications, num_pro cedures, num_lab_procedures, number_inpatient, number_emergency and number_outpatient. For the number of times a patient was encountered in the hospital, we constructed a new variable number_visits to keep track. For all other categorical variables, we kept the most recent values.

- **Train Test Split of the dataset** For later use on machine learning models, it is necessary that we at least have a held-out test set for evaluation. For this task, we chose the split such that 95% of the data consists of the training set, and 5% the test set. The reason that such a split was chosen is that since the data have 71,518 patients in total, 5% would be 3500 patients which seem sufficient for the purpose of testing.

- **Missing value** After cumulating the data, we try to deal with missing values, which can lead to the failure of training models.

  [Table. Missing Value]

  As we can see, 'race', 'weight', 'payer_code', 'medical_specialty', 'diag_1', 'diag_2' and 'diag_3', these 7 features contain NAs, of which:

  - 'weight' has 96.87% as NA
  - 'payer_code' has 39.58% as NA
  - 'medical_specialty' has 49.1% as NA

  We drop these 3 columns since there are large percent of missing values. For the rest, we will impute with mode.

- **One-hot encoding for response variable**

  For the response variable 'readmitted', we have three categories 'No', < 30, > 30. 'No' means the patient does not readmit after leaving the hospital, while < 30 means the patient returns to the hospital within 30 days. Thus, we assign numeric 0 to 'No', 1 to > 30, and 2 to < 30, which is convenient for the model construction. Logically, 2 would be the most severe since patients were readmitted in shorter time period and 0 with lightest severity.

3 Exploratory Data Analysis

Race There are five race group in our dataset: Caucasian, African American, Hispanic, Other, Asian, in which Caucasian and African American account for most of our data and we assign them into three major categories: Caucasian, African American, Other.

Gender There are slightly more female than male (5000 more) in the training dataset and the distribution of gender indicates that it does not have strong impact on readmitted status

Age Although the distribution of 'age' is skewed, we can see that the distributions of 'readmitted' over 'age' seem similar over each of the 'age' groups. Thus, this variable might not be so useful when building models. Figure 1

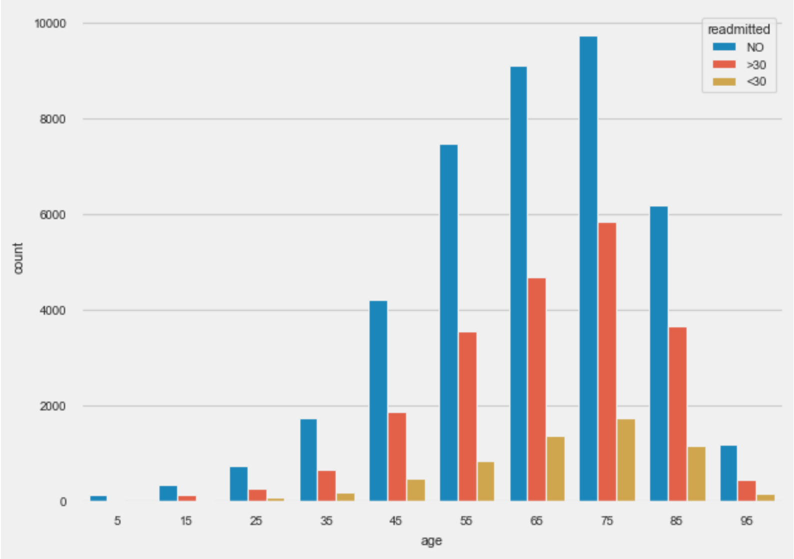

Figure 1: Age vs readmitted

Admission_type_id From the id table provided by the dataset, we know that there are 8 categories corresponding to this feature: 1. Emergency 2. Urgent 3. Elective 4. Newborn 5. Not Available 6. NULL 7. Trauma Center 8. Not Mapped.

We then map all the names to the ids in the data frame, which is shown in Figure 2.

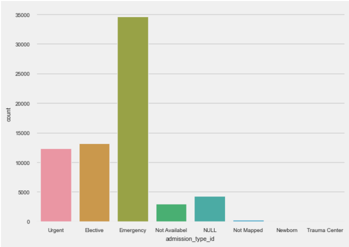

Figure 2: before transformation

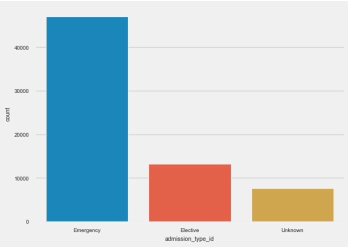

Figure 3: after transformation

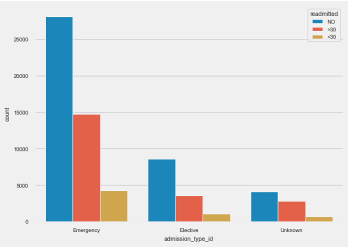

Figure 4: type vs readmitted

We can see that: ‘Emergency‘, ‘Urgent‘ and ‘Trauma Center‘ are actually the same thing! We will put them all under ‘Emergency‘. ‘Not Available‘, ‘NULL‘ and ‘Not Mapped‘ are also in the same category. We will put them under ‘Unknown‘. ‘Newborn‘ have very little data and for simplicity, we will also put it under the category of ‘Unknown‘. Now we map them to the 3 categories as follows:

- ‘Emergency‘ for ‘Emergency‘, ‘Urgent‘, and ‘Trauma Center‘
- ‘Elective‘ for ‘Elective‘
- ‘Unknown‘ for ‘Not Available‘, ‘NULL‘, ‘Not Mapped‘ and ‘Unknown‘

We can see that there are a few differences among each of the ‘admission_type_id‘ over ‘readmitted‘.

Discharge_disposition_id From the documentation, we know that the ‘discharge_disposition_id‘ has 30 categories. There are just too many categories to throw into ML models. However, it is possible that we define some larger categories to summarize a little bit based on the specific features. We define the following seven categories:

- ‘Self Care‘ for Discharged to home, and Neonate discharged to another hospital for neonatal aftercare

- ‘Home Care‘ for Discharged/transferred to home with home health service, Left AMA, and Discharged/transferred to home under the care of Home IV provider
- ‘Hospice‘ for Hospice / home, and Hospice / medical facility
- ‘Expired‘ for Expired, Expired at home. Medicaid only, hospice, and Expired, place unknown. Medicaid only, hospice
- ‘Hospital Care‘ for Admitted as an inpatient to this hospital, Discharged/transferred within this institution to Medicare approved swing bed, Discharged/transferred/referred to this institution for outpatient services, and Still patient or expected to return for outpatient services
- ‘Unknown‘ for NULL, Not Mapped, and Unknown/Invalid
- ‘Other‘ for Discharged/transferred to another short term hospital, Discharged/transferred to SNF, Discharged/transferred to ICF, Discharged/transferred to another type of inpatient care institution, Neonate discharged to another hospital for neonatal aftercare, Discharged/transferred/referred another institution for outpatient services, Discharged/transferred to another rehab facility including rehab units of a hospital, Discharged/transferred to a long term care hospital, Discharged/transferred to a nursing facility certified under Medicaid but not certified under Medicare, Discharged/transferred to another Type of Health Care Institution not Defined Elsewhere, Discharged/transferred to a federal health care facility, Discharged/transferred/referred to a psychiatric hospital of psychiatric distinct part unit of a hospital, and Discharged/transferred to a Critical Access Hospital (CAH)

Figure 5 shows the result of classification.

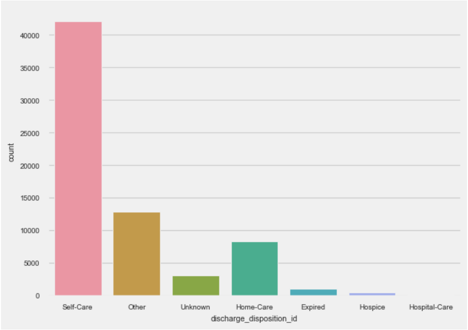

Figure 5: after transformation

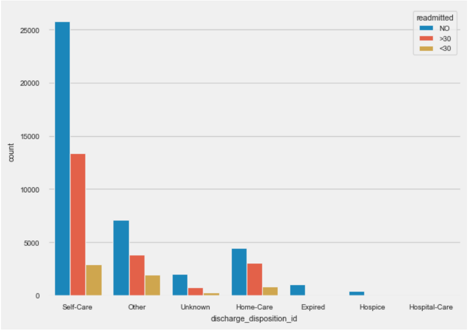

Figure 6: discharge type vs readmitted

We can see that there is some kind of relationship between the grouped ‘discharge_disposition_id‘ and ‘readmitted‘.

Admission_source_id From the documentation, the ‘admission_source_id‘ has 25 categories,which seems too much for model construction. However, it is possible that we define some larger categories to summarize a little bit based on some essential features. We define the following five categories as follows:

- ‘Referral‘ for Physician Referral, Clinic Referral, and HMO Referral
- ‘Transfer‘ for Transfer from a hospital, Transfer from a Skilled Nursing Facility (SNF), Transfer from another health care facility, Transfer from critical access hospital, Readmission to Same Home Health Agency, Transfer from hospital inpt/same fsc result in a sep claim, Transfer from Ambulatory Surgery Center, and Transfer from Hospice
- ‘Readmission‘ for Readmission to Same Home Health Agency
- ‘Delivery‘ for Emergency Room, Court/Law Enforcement, Normal Delivery, and Premature Delivery
- ‘Unknown‘ for Not Available, Sick Baby, Extramural Birth, Not Available, NULL, Not Mapped, Unknown/Invalid, Born inside this hospital, and Born outside this hospital

The following plot Figure 7 shows the result.

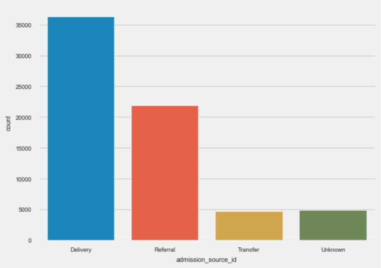

Figure 7: after transformation

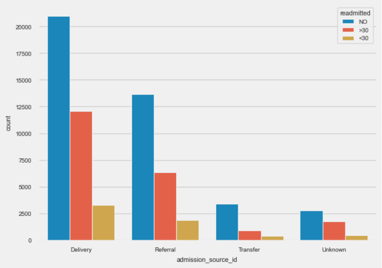

Figure 8: admission source vs readmitted

'Admission_source_id' does not seem to be a very good feature for classification of 'readmitted'.

Feature Engineering (Cumulative) : Because we want to understand the impact of variables to readmission status, We feature engineerd the following variables into cumulative form: time_in_hospital (Figure 9), number_of_lab_procedures (Figure 10), number_of_procedures (Figure 11), number_of_medications (Figure 12), number_of_inpatient (Figure 13), number of outpatient (Figure 14), number_of_emergency (Figure 15), visiting_to_hospital (Figure 16), number_of_diagnosis (Figure 17)

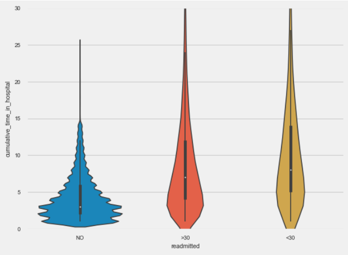

Figure 9: time_in_hospital

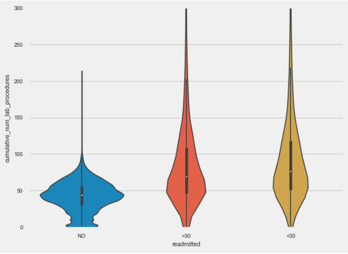

Figure 10: num_lab_procedures

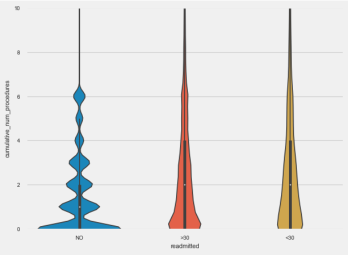

Figure 11: num_procedures

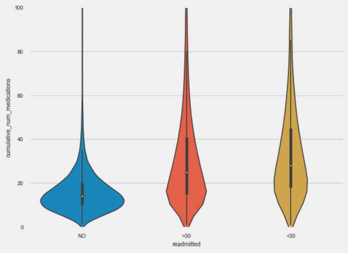

Figure 12: num_of_medications

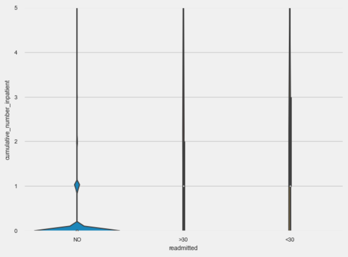

Figure 13: num_inpatient

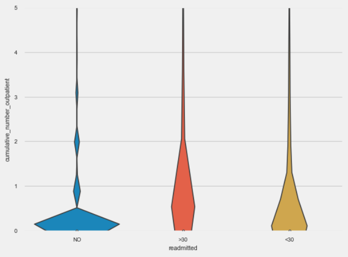

Figure 14: num_outpatient

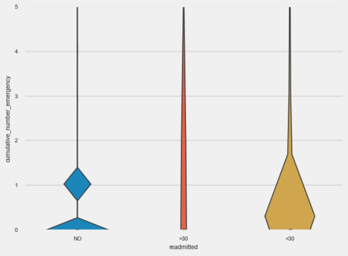

Figure 15: num_emergency

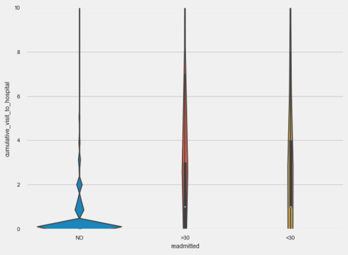

Figure 16: num_visit_hospital

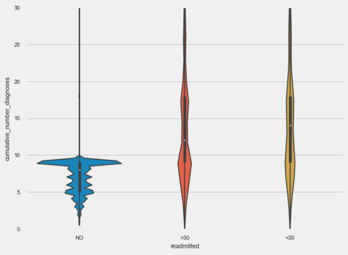

Figure 17: num_diagnosis

Diagnostics: ‘diag_1‘, ‘diag_2‘, and ‘diag_3‘ These three variables stand for the 1st, 2nd, and 3rd diagnosis doctor makes about a patient, which has 848, 923, and 954 unique values in each feature separately. At first, we thought that we would have no other option but to delete these features from our analysis. However, after examining the documentation of the dataset, we found that the diagnoses are recorded according to the International Classification of Disease 9th Revision(ICD9), for which we could group the categories as follows:

- ‘circulatory‘ for icd9: 390–459, 785
- ‘digestive‘ for icd9: 520–579, 787
- ‘genitourinary‘ for icd9: 580–629, 788
- ‘diabetes‘ for icd9: 250.xx
- ‘injury‘ for icd9: 800–999
- ‘musculoskeletal‘ for icd9: 710–739
- ‘neoplasms‘ for icd9: 140–239
- ‘respiratory‘ for icd9: 460–519, 786
- ‘other‘ for otherwise

The following Figure 18 Figure 19 Figure 20 shows the result of the transformation for diagnosis 1, 2, and 3 separately.

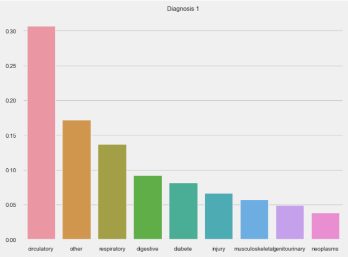

Figure 18: diagnosis 1

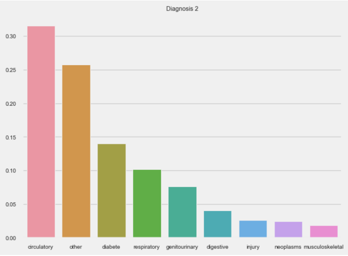

Figure 19: diagnosis 2

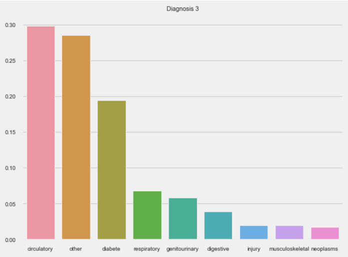

Figure 20: diagnosis 3

We can see that after transformation, each of the diagnoses now has 9 categories. As of now, we should be able to include these 3 variables in our model construction.

Maximum_glu_serum Maximum_glu_serum is another feature that was included in modeling. Below we show the distribution of Maximum_glu_serum and Maximum_glu_serum vs readmitted in Figure 21.

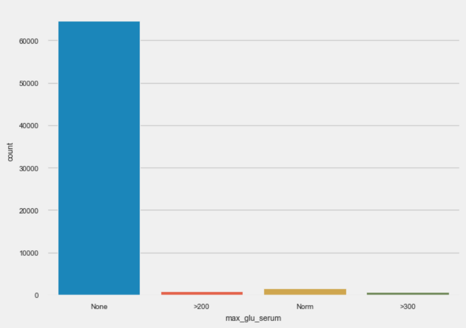

Figure 21: max glu serum

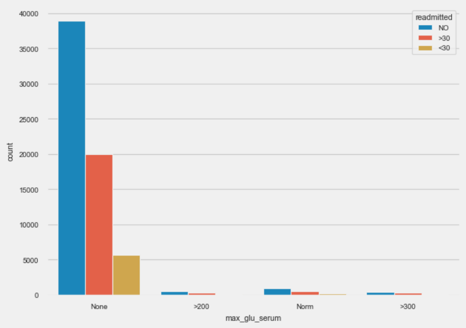

Figure 22: max glu serum vs readmitted

HbA1C_result HbA1C_result is another feature that was included in modeling. Below we show the distribution of HbA1C_result and HbA1C_result vs readmitted in Figure 23.

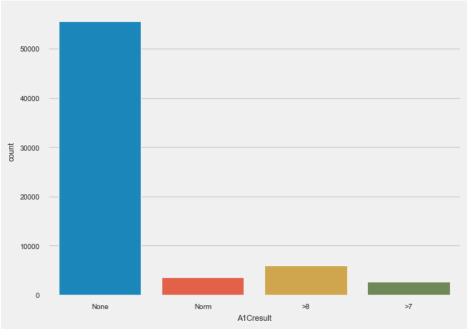

Figure 23: A1C result

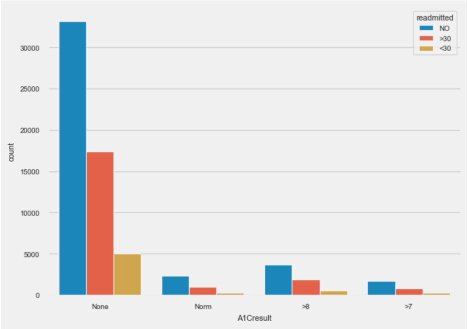

Figure 24: A1C result vs readmitted

Medication_change Medication_change refers to changes in dosage or generic of medication. As can be seen from Figure 25, this feature does not seem like seperate well with the response variable, so we took it out from modeling.

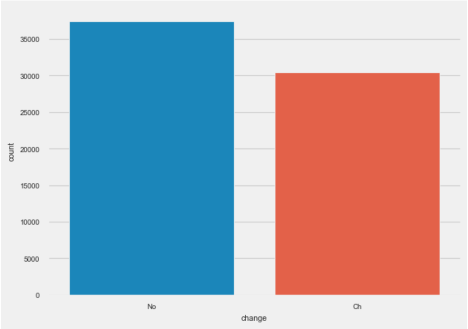

Figure 25: Change in medication

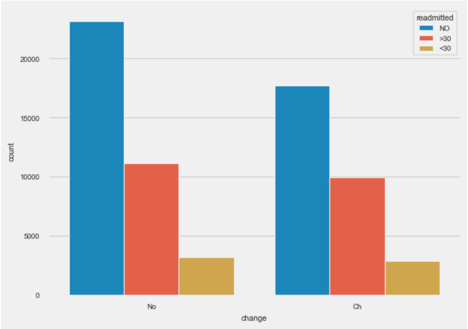

Figure 26: Change in medication vs readmitted

23 types of diabetes medication There are two features in ‘diabetesMed‘: ‘Yes‘ and ‘No‘. ‘Yes‘ means the patient uses glycaemic drugs while ‘No‘ means the patient does not use this kind of drug. There are also 23 variables specifically indicating what diabetes medicine that the patient has been used. The categories for these 23 variables are:

- “up” if the dosage was increased during the encounter,
- “down” if the dosage was decreased, “steady” if the dosage did not change, and
- “no” if the drug was not prescribed.

The following plot shows the distributions of these features, we can see that ‘examide‘ and ‘citoglipton‘ do not have any changes. So, we can drop these 2 columns. Figure. Distributions of 23 diabetes medicines. But we still have 21 diabetes medications left, a lot of which are not used for the majority of patients. The following table shows the distribution of these variables numerically. Table. Numerical distributions of 21 diabetes medicines. We will just focus on drugs with more than 10% patients prescribed the medications, which will be metformin, glipizide, glyburide and ‘insulin‘. We will drop all other medications. The following Figure 27 shows the transformation from 23 features to 4 features.

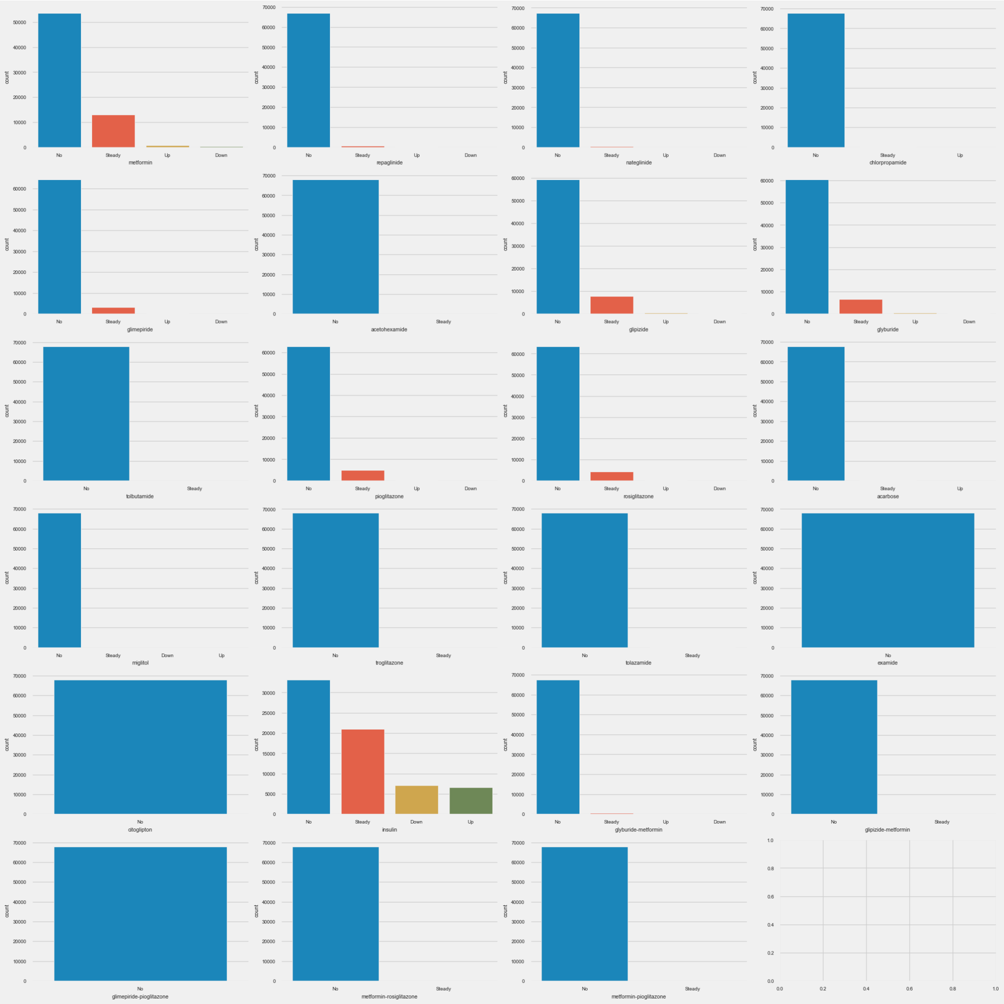

Figure 27: 23 medications

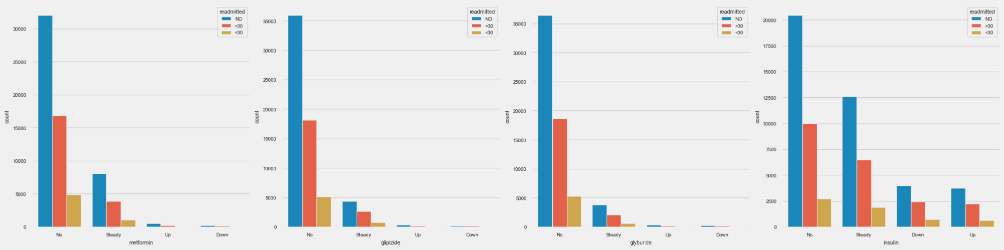

Figure 28: 4 remain medication vs readmitted

number_of_medication_change Using the 4 remaining medications from above, we construct a new feature called number_of_medication_change which only takes dosage change of medication into consideration. This Figure 29 shows number_of_medication_change vs readmitted.

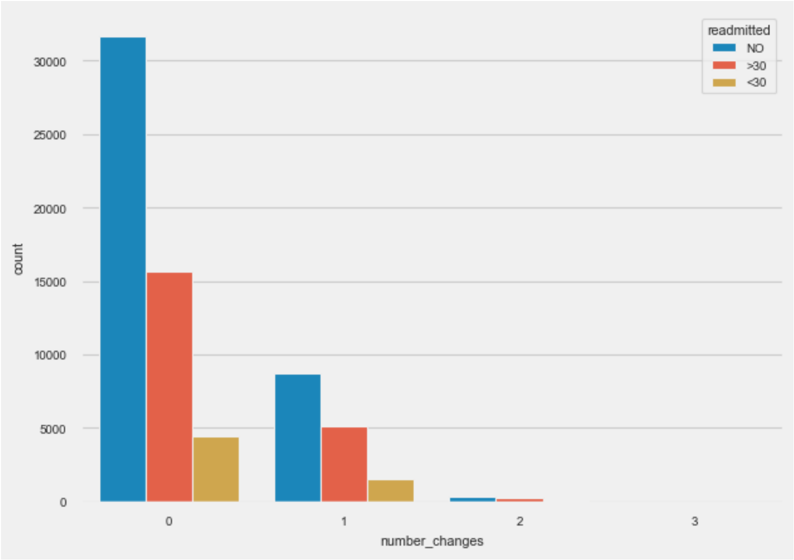

Figure 29: num change in medication

4 Results

Dummy variables and correlation map After doing the EDA, there are 25 remaining variables, 17 of which are categorical and 8 of which are numerical. For python modeling using sklearn and other handy packages, it is required to transform categorical variables to dummy variables so that the process could proceed without error. The correlation between variables is shown in Figure 30 below.

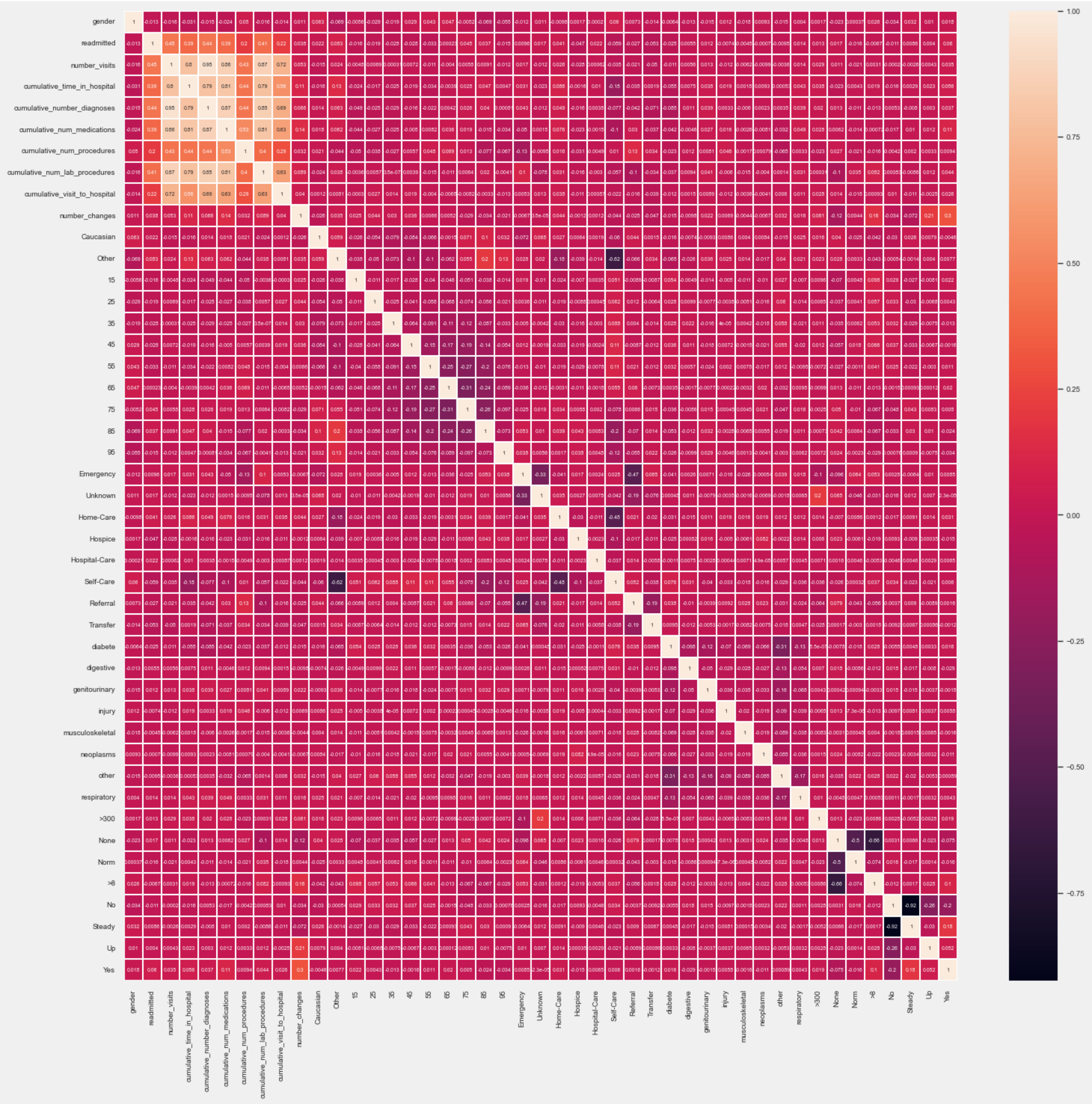

Figure 30: correlation

Modeling

- **types of models** Based on the above EDA, both traditional statistical models and machine learning models were used in model construction. For traditional statistical models, we mainly used one vs all logistic regression. For machine learning models, we tried Gaussian Naive Bayes model, Linear Discriminant Analysis model, Decision Tree model and Random Forest model. For more advanced gradient-boosting models, we tried catboost model, lightgbm model and xgboost model. Below we show the final results from our modeling process.
- **evaluation** 4 different metrics, accuracy, precision, recall and F1-score were applied to evaluation of our models. Accuracies were calculated over all samples, while precision, recall and F1-score were first calculated w.r.t. each pair of classes and then average with respect to weights determined by percentage of the category in the total population.

	Train				Test			
Model	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
gnb	0.74	0.71	0.74	0.71	0.76	0.74	0.76	0.73
lda	0.73	0.71	0.73	0.68	0.74	0.73	0.74	0.69
lr	0.77	0.73	0.77	0.72	**0.78**	**0.78**	**0.78**	0.74
tr	0.75	0.72	0.75	0.73	0.72	0.70	0.72	0.71
forest	0.78	0.76	0.78	0.74	**0.78**	0.75	**0.78**	0.74
catboost	0.78	0.79	0.78	0.75	**0.78**	0.77	**0.78**	**0.75**
lgb	0.79	0.79	0.79	0.75	**0.78**	0.76	**0.78**	**0.75**
xgb	0.79	0.79	0.79	0.75	**0.78**	0.76	**0.78**	0.74

Statistical Model: Among all the traditional model, one vs all logistic regression gives us the best result, with an test accuracy of 0.78, precision of 0.78 and recall of 0.78. At the lr classification table, we can see that for readmission type 2 (< 30), we only successfully predicted 9 true positive values but 194 type 2 (< 30), which suggests our classification is not too bad.

Model	Train			Test		
logistic regression	Precision	Recall	F1-Score	Precision	Recall	F1-Score
0	0.78	0.99	0.87	0.79	0.99	0.88
1	0.73	0.57	0.64	0.75	0.58	0.65
2	0.4	0.01	0.01	0.80	0.01	0.03
Weighted Average	0.73	0.77	0.72	**0.78**	**0.78**	**0.74**

Machine Learning Model We choose Catboosting as our recommended machine learning model. Its performance is almost the same as logistic regression. From the confusion matrix, we can tell Catboosting predict 27 true positives for type 2 admission (< 30) , which is somewhat better than logistic regression.

Model	Train			Test		
catboost	Precision	Recall	F1-Score	Precision	Recall	F1-Score
0	0.80	0.98	0.88	0.80	0.98	0.88
1	0.74	0.60	0.66	0.74	0.59	0.66
2	0.88	0.08	0.15	0.69	0.06	0.12
Weighted Average	0.79	0.78	0.75	**0.77**	**0.78**	**0.75**

Some Limitations Due to data imbalance, our models tend to predict patient that were not readmitted into hospital most accurately and tend to predict other two classes also toward not readmitted. This means that our model have high recall for not readmitted patients. Our models would be really useful if the goal is to classify patients who would not be readmitted. If the task is classify patients who would be readmitted and how fast their readmission would then our models would not perform so well.

Future direction for the analysis One potential area we can ascertain in the future is to study the impact of these variables on time in hospital. Which variable will make a patient study in hospital for a longer period, issuance, disease type, number of medications, etc. To study the impact, we can apply regression techniques to our data, and GAM could be a good model for further analysis.

5 Discussion

This project provides an extensive and comprehensive study for this clinical dataset. We try to use all popular traditional statistical methods and machine learning methods to fit the dataset and the result seems pretty well. We also compare the difference between the statistical model and machine learning model and find that ML model may do better for classification when the number of statistics in the category is small (unbalanced data). This actually gives us some insight when we try to use different models for prediction. If the data is really unbalanced and we want to find some features for the smaller category, we may prefer a machine learning model. On the other side, this project only provides a prediction for general readmission rate for all patients. In the future, researchers can focus on different types of patients' readmission rate specifically. For instance, diabetes patients, respiratory patients, and so on, which may have some more conclusion in the given context.

References

[1] 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