

Readmission to Hospital Predictor for Diabetic Patients

Group 29: Rachel Wong, Javairia Raza, Sukhdeep Kaur, and Zhiyong Wang

- Summary
- Introduction
- Methods
 - Data
 - Analysis
- Results and Discussion
- Future Improvements
- References

Summary

This project builds a classification model using logistic regression to analyze various features related to the care of the diabetic patients to predict whether or not the patient will be readmitted to the hospital again.

Introduction

For this project, we are trying to answer the predictive question: Given a diabetic patient's demographic, medication history and management of diabetes during hospital stay, can we predict if they will be readmitted to the hospital or not?

Due to COVID-19, it is critical to reduce the burden on the health care system and prevent readmission rates from increasing to make space for COVID-19 cases. Our predictor aims to look at the diabetes management and diagnosis during a patient's hospital stay to understand how much this affects their readmission. Analysis with machine learning models will identify features more likely to predict patient readmission so that hospital management has an idea of where to improve their protocols to better manage diabetic patients during their hospital stay to provide effective care and prevent readmission during this critical time.

Methods

The R programming language (R Core Team 2020) and Python programming language (Van Rossum and Drake 2009) were used to perform the analysis. The following R and Python packages were also used to perform the analysis:

- knitr (Xie 2020)
- tidyverse (Wickham et al. 2019)

- numpy (Harris et al. 2020)
- pandas (McKinney and others 2010)
- altair and altair_saver (VanderPlas et al. 2018)
- hashlib (Rossum and Drake Jr, n.d.b)
- matplotlib (Hunter 2007)
- iPython (Pérez and Granger 2007)
- scikit (Pedregosa et al. 2011)
- requests (Chandra and Varanasi 2015)
- zipfile (Rossum and Drake Jr, n.d.d)
- urllib (Rossum and Drake Jr, n.d.c)
- io (Rossum and Drake Jr, n.d.a)
- pandas profiling (Brugman 2019)

For statistical analysis specifically:

- StackOverflow (Whidden, n.d.)
- store_vals_function (Kolhatkar, n.d.)
- plot formatting (Ostblom, n.d.)

The code used to perform the project and create this report can be found here (<https://github.com/UBC-MDS/group29>).

Data

The data are submitted on behalf of the Center for Clinical and Translational Research, Virginia Commonwealth University, a recipient of NIH CTSA grant UL1 TR00058, and a recipient of the CERNER data. This dataset was collected from 1998-2008 among 130 hospitals and integrated delivery networks throughout the United States of America.

This data set was sourced from the UCI Machine Learning Repository (Strack et al. 2014a) and can be found here (<https://archive.ics.uci.edu/ml/datasets/diabetes+130-us+hospitals+for+years+1999-2008#>). Research from this collected data was used to assess diabetic care during hospitalization and determine if patients were likely to be readmitted or not. The paper by (Strack et al. 2014b) can be found here (<https://www.hindawi.com/journals/bmri/2014/781670/>). Each row corresponds a unique encounter with a diabetic patient, totaling 74,036,643 unique encounters. Details about each column feature of information collected during these unique encounters can be found here (<https://www.hindawi.com/journals/bmri/2014/781670/tab1/>).

A subset of the data (1000 samples) was used for exploratory data analysis and model testing for the purposes of improving the efficiency and speed of the analysis. We have noted in future improvements that with accessibility to more powerful machinery, we will take full advantage of the large dataset.

Analysis

After data cleaning, machine learning model logistic regression was tested against radial basis function kernel with support vector machine (RBF SVM) and a baseline dummy classifier method. Logistic regression was determined as the best model in terms of fit and score time, accuracy, and f1 score. Continuing with logistic regression, different parameters were optimized and our model was then used to predict diabetic patient readmission (found in the readmitted target column of the data set). The code used to perform the analysis can be found here (<https://github.com/UBC-MDS/group29/tree/main/src>).

Results and Discussion

Through exploratory data analysis, we determined that some of the features were not informative to answering our question or contained many missing values. Feature information was confirmed through Pandas Profiling which can be found here (https://github.com/UBC-MDS/group29/blob/main/reports/figures/pandas_profiling.html).

Before model testing, data cleaning was done to address the findings of non-informative features, class imbalance, NAN values, and duplicate encounters, and this code can be found here (<https://github.com/UBC-MDS/group29/blob/main/src/processingdata.py>). Non informative features included encounter id and patient number as those were entirely unique values. We removed weight, payer_code, medical_speciality, examide and citoglipton because much of the information was missing or all values in the column were the same. We were unsure about the race columns as majority of the patients were Caucasian and we did not want to add any racial biases to the data. As shown below, the distribution of race was reviewed to determine that there was no significant difference between different races in patient readmission and therefore the race column was removed from our analysis to prevent any bias.

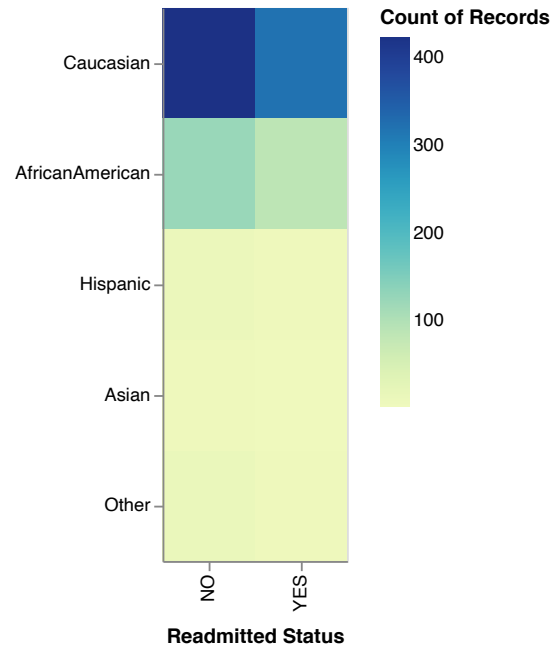


Figure 1. Distribution of Race and Readmission Status

Several of the numerical features were reviewed by plotting the distribution of the feature for the two different target classes against each other and found the primary diagnosis of the patient, days spent in hospital, number of medications taken and the frequency of lab procedures to be most informative of predicting readmission status. The importance of the primary diagnosis feature is that if the initial diagnosis is diabetes perhaps patients received better care during their first encounter and thus do not get readmitted (Strack et al. 2014b).

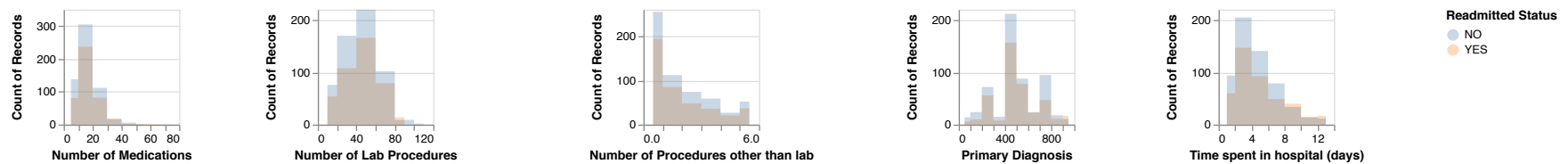


Figure 2. Distribution of Numeric Features and Readmission Status

Several of the categorical features were also reviewed in a similar format to the numerical features analysis, and found that metformin, insulin, and Hemoglobin A1C levels were most informative of predicting readmission status. Testing for hemoglobin levels may have resulted in applicable changes to the medications and thus better management of diabetes during their hospital stay, preventing readmission (“Insulin, Medicines, & Other Diabetes Treatments” 2016).

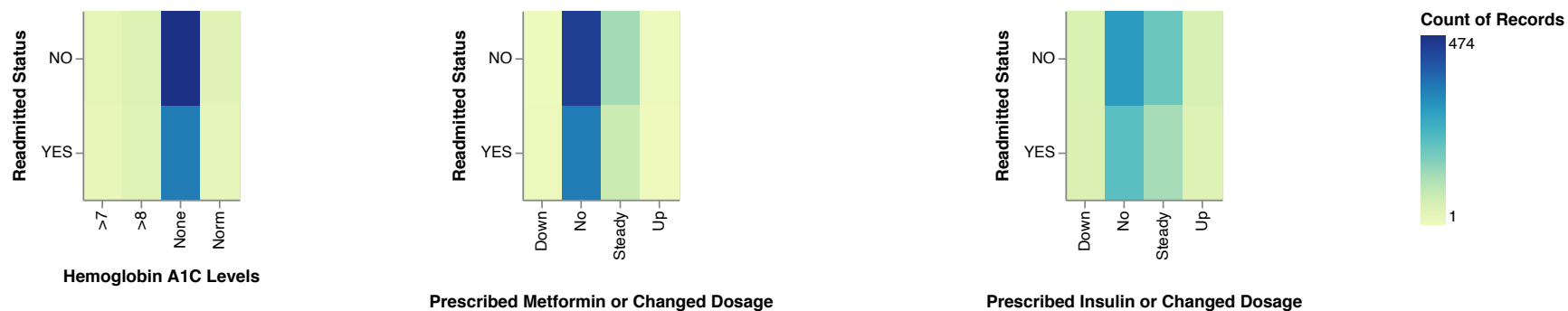


Figure 3. Distribution of Categorical Features and Readmission Status

After exploratory data analysis and data cleaning, machine learning models were tested with our data set, and logistic regression (with equal importance assigned to both classes) was determined as the best model to move forward with in our analysis based on fit and score time, accuracy, and f1 scores as shown in Table 1. Fit and score time were chosen as evaluation metrics because the data set was quite large although we took a subset, and it was important to ensure the analysis would fit and score promptly. Accuracy and f1-score were also metrics used to evaluate our models on because accuracy determined the number of predictions our model predicted correctly, and f1-score further analyzes accuracy by combining precision and recall scores. From the scores in Table 1, we can see that logistic regression (balanced) had the highest test and train accuracy scores, as well as test and train f1-scores. Although the fit and score times of logistic regression (balanced) were the slowest, they were only slightly slower from the other models tested. The code used to perform machine learning can be found here (https://github.com/UBC-MDS/group29/blob/main/src/explore_script4.py).

Table 1. Classifier Models and Their Associated Times and Scores

Models	Fit Time	Score Time	Test Accuracy	Train Accuracy	Test F1-score	Train F1-score
Dummy Classifier	0.0236	0.0238	0.5163	0.4903	0.4381	0.4002
RBF SVM	0.0651	0.0427	0.6200	0.6928	0.3738	0.4990
Logistic Regression	0.0734	0.0229	0.5738	0.8347	0.4581	0.7903
Logistic Regression (balanced)	0.0862	0.0243	0.5662	0.8416	0.5021	0.8170

Different parameters of the models were optimized and identified potential improvements for our model in future use. From the confusion matrix, we can see that there is a high amount of false positives and false negatives. From the receiver operating characteristic curve with area under the curve (ROC curve with AUC), and the red point showing our threshold at 0.5, our model is performing at 0.54 where AUC = 1.0 is a perfect classification. These results suggest that our model could be improved by choosing a different classifier or testing other scoring methods for our

optimization besides f1-score. Our model performed much lower on the testing data than on the training data, and will therefore require improvements before implementation on deployment data for use in clinical studies. Details about these improvements can be found in the Future Improvements section of this report.

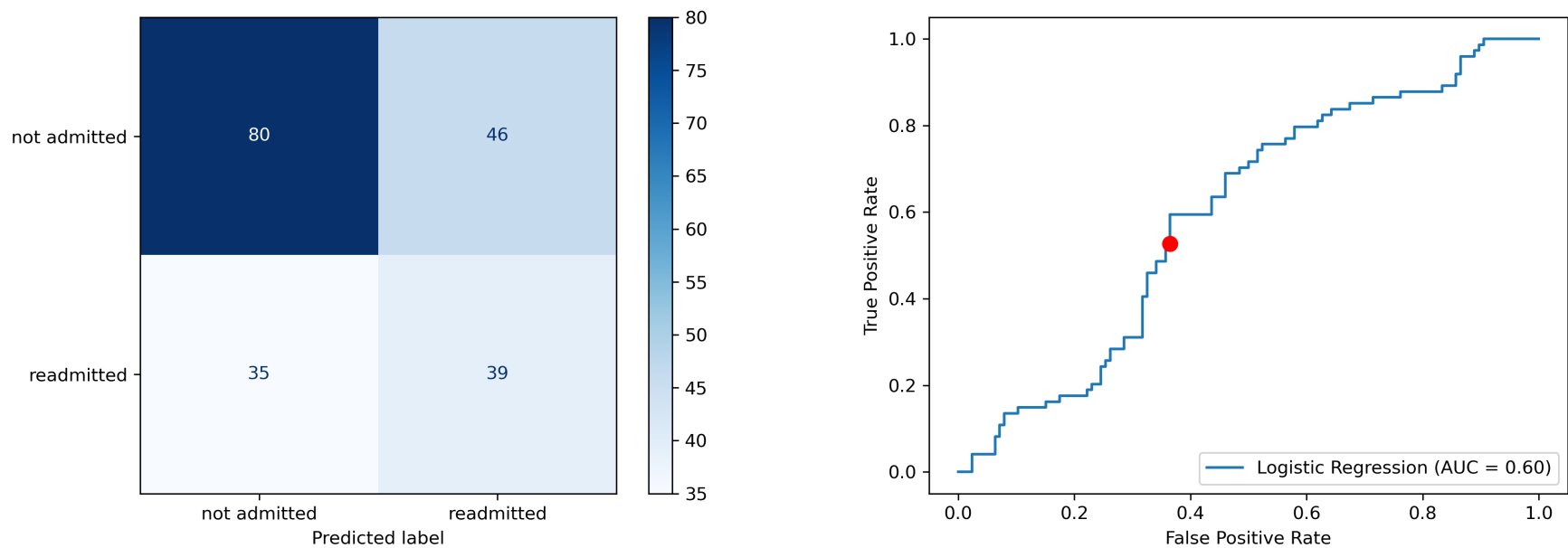


Figure 4. Confusion Matrix and ROC AUC Curve Results From Logistic Regression (Balanced)

Completing our analysis using our model on test data, the top 10 features most indicative of patient readmission were found by having the highest associated coefficients as shown in Table 2. These results suggest the health care system begin implementing changes in these specific features, particularly in admission type ID, in order to manage diabetic patient readmission more effectively.

Table 2. Top 10 Features Most Indicative of Patient Readmission

Features	Coefficients
admission_type_id	0.0116065
discharge_disposition_id	0.1415067
admission_source_id	0.0504247
time_in_hospital	0.1985486
num_lab_procedures	0.0289513

Features	Coefficients
num_procedures	0.1612403
num_medications	0.3981985
number_outpatient	0.1908310
number_emergency	0.0592797
number_inpatient	0.5405392

Future Improvements

To improve our model in the future to further analyze patient readmission, we can recommend three main suggestions. First, rather than subsetting our data to take a representative and random sample of 1000, we can take a larger subset or even use our entire data set of roughly 70,000 observations. In future models, the use of a larger data set will provide more accurate predictions, reduce bias, and identify outliers that may skew our results. Second, other classifier models such as random forest could be used to test against our logistic regression model to improve predictions. The use of logistic regression assumes there is a linear relationship between the independent and dependent variables, which may not be the case in our project. Random forest is non-parametric and will bypass this linearity assumption associated with logistic regression. Third, our model analyzes a binary classification of readmission although the original data set was a multi-classification of readmission (not readmitted, readmitted for less than 30 days, readmitted for more than 30 days). By using a multi-classification model instead, we can generate a more specific prediction and analyze the severity of features related to readmission time, making our conclusions more informative to improving the health care system.

References

- Brugman, Simon. 2019. “pandas-profiling: Exploratory Data Analysis for Python.” <https://github.com/pandas-profiling/pandas-profiling> (<https://github.com/pandas-profiling/pandas-profiling>).
- Chandra, Rakesh Vidya, and Bala Subrahmanyam Varanasi. 2015. *Python Requests Essentials*. Packt Publishing Ltd.
- Harris, Charles R., K. Jarrod Millman, St’efan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, et al. 2020. “Array Programming with NumPy.” *Nature* 585 (7825): 357–62. <https://doi.org/10.1038/s41586-020-2649-2> (<https://doi.org/10.1038/s41586-020-2649-2>).
- Hunter, John D. 2007. “Matplotlib: A 2D Graphics Environment.” *Computing in Science & Engineering* 9 (3): 90–95.

- “Insulin, Medicines, & Other Diabetes Treatments.” 2016. *National Institute of Diabetes and Digestive and Kidney Diseases*. U.S. Department of Health; Human Services. <https://www.niddk.nih.gov/health-information/diabetes/overview/insulin-medicines-treatments> (<https://www.niddk.nih.gov/health-information/diabetes/overview/insulin-medicines-treatments>).
- Kolhatkar, Varada. n.d. “DSCI: 571 Supervised Learning 1”. “https://github.ubc.ca/MDS-2020-21/DSCI_571_sup-learn-1_students” (%22https://github.ubc.ca/MDS-2020-21/DSCI_571_sup-learn-1_students%22).
- McKinney, Wes, and others. 2010. “Data Structures for Statistical Computing in Python.” In *Proceedings of the 9th Python in Science Conference*, 445:51–56. Austin, TX.
- Ostblom, Joel. n.d. “DSCI: 531 Data Visualization 1”. “https://github.ubc.ca/MDS-2020-21/DSCI_531_viz-1_students” (%22https://github.ubc.ca/MDS-2020-21/DSCI_531_viz-1_students%22).
- Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, et al. 2011. “Scikit-Learn: Machine Learning in Python.” *Journal of Machine Learning Research* 12 (Oct): 2825–30.
- Pérez, Fernando, and Brian E Granger. 2007. “IPython: A System for Interactive Scientific Computing.” *Computing in Science & Engineering* 9 (3).
- R Core Team. 2020. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/> (<https://www.R-project.org/>).
- Rossum, Guido "Van, and Fred L" Drake Jr. n.d.a. “Core Tools for Working with Streams, Python/Cpython”. “<https://github.com/python/cpython/blob/3.9/Lib/io.py>” (%22<https://github.com/python/cpython/blob/3.9/Lib/io.py>%22).
- — —. n.d.b. “Secure Hashes and Message Digests, Python/Cpython”. “<https://docs.python.org/3/library/hashlib.html>” (%22<https://docs.python.org/3/library/hashlib.html>%22).
- — —. n.d.c. “URL Handling Modules, Python/Cpython”. “<https://github.com/python/cpython/tree/3.9/Lib/urllib/>” (%22<https://github.com/python/cpython/tree/3.9/Lib/urllib/>%22).
- — —. n.d.d. “Work with Zip Archives, Python/Cpython”. “<https://github.com/python/cpython/blob/3.9/Lib/zipfile.py>” (%22<https://github.com/python/cpython/blob/3.9/Lib/zipfile.py>%22).
- Strack, Beata, Jonathan P. DeShazo, Chris Gennings, Juan L. Olmo, Sebastian Ventura, Krzysztof J. Cios, and John N. Clore. 2014a. “UCI Machine Learning Repository.” University of California, Irvine, School of Information; Computer Sciences. <http://archive.ics.uci.edu/ml> (<http://archive.ics.uci.edu/ml>).
- — —. 2014b. “Impact of Hba1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records.” *BioMed Research International*. Hindawi. <https://www.hindawi.com/journals/bmri/2014/781670/> (<https://www.hindawi.com/journals/bmri/2014/781670/>).
- VanderPlas, Jacob, Brian Granger, Jeffrey Heer, Dominik Moritz, Kanit Wongsuphasawat, Arvind Satyanarayan, Eitan Lees, Ilia Timofeev, Ben Welsh, and Scott Sievert. 2018. “Altair: Interactive Statistical Visualizations for Python.” *Journal of Open Source Software* 3 (32): 1057.

Van Rossum, Guido, and Fred L. Drake. 2009. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

Whidden, Christopher. n.d. "StackOverflow Question 39187875". "https://stackoverflow.com/questions/39187875/scikit-learn-script-giving-vastly-different-results-than-the-tutorial-and-gives" (%22https://stackoverflow.com/questions/39187875/scikit-learn-script-giving-vastly-different-results-than-the-tutorial-and-gives%22).

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. <https://doi.org/10.21105/joss.01686> (<https://doi.org/10.21105/joss.01686>).

Xie, Yihui. 2020. *Knitr: A General-Purpose Package for Dynamic Report Generation in R*. <https://yihui.org/knitr/> (<https://yihui.org/knitr/>).