Hospital Readmission of Diabetic Patients

Sydney Walter
sydneyrwalter@lewisu.edu
DATA-51000-[02], [Summer 2023]
Data Mining and Analytics
Lewis University

I. Introduction

The data that was found was 10 years (1999-2008) of clinical care data from 130 United States hospitals and integrated delivery networks [1]. This specific dataset was focused on patients who had been admitted to the hospital for diabetic reasons. The data was originally extracted from the clinical database, Health Facts database (Cerner Corporation, Kansas City, MO). The database contained 74 million unique encounters with 17 million unique patients. From that database, the inpatient diabetes encounters were identified. Information was extracted from the database if the encounters satisfied the following criteria:

(1.) It is an inpatient encounter (a hospital admission)
(2.) It is a "diabetic" encounter, that is, one during which any kind of diabetes was entered to the system as a diagnosis.
(3.) The length of the stay was at least 1 day and at most 14 days.
(4.) Laboratory tests were performed during the encounter.
(5.) Medications were administered during the encounter.

The purpose of this assignment was to create a predictive model using different methods to aid in the classification of the class label. The problem that I was trying to solve for this specific dataset was whether or not the diabetic patient will be readmitted after the initial recorded hospital visit. This predictive model allows patterns to be created and identified to predict which patients are at a higher risk for readmission to the hospital. This is important because if it is known that they are predicted to be a higher risk, there are certain precautions that can be taken. As this predictive model was created, certain attributes were identified that play an influential role in the probability that the patient will be readmitted: troglitazone, glipizide-pioglitazone, number_inpatient, number_emergency, and tolazomide.

As a brief overview of the step that were taken to generate this predictive model, I first preprocessed the data. I normalized the values of the attributes, and I changed the values of the target variable to be binary. Then I conducted gain of information ratios for the attributes to determine which ones deemed relevance to the problem I was trying to solve with the data. Then with the relevant attributes, I split my data into test and train sets. I used k Nearest Neighbors, Random Forest, and Logistic Regression to build different models. I then used evaluation metrics to determine the best model for this data which was the Random Forest.

The following sections of this report contain attributes' descriptions, the methodology, results and discussion, and the conclusion. Section II contains the description of the dataset used for this analysis. The methodology for analysis is presented in section III. The results and discussion are reported in section IV. In the last section, section V, I provided the conclusions of this analysis.

II. Data Description

The data that was used represented 10 years of diabetic patient encounter information from over 130 different hospitals in the United States. All of the dataset characteristics can be viewed in Table I. Since the goal was to determine if the diabetic patient was going to be readmitted after their initial hospital encounter, the attributes pertained to characteristics of the patient and the care they received prior and during the encounter. Therefore, some of the attributes contained demographic information on the patients. The data supplied information, such as, gender, race, and age. The gender was classified as male, female, or unknown/invalid. The race was identified as Caucasian, Asian, African American, Hispanic, and other. The age was grouped into 10-year intervals including the lower bound.

The data also included information of the patients' hospital encounters. It included the admission type which was an integer that corresponded to 9 different values: emergency, urgent, elective, newborn, etc. There was a discharge attribute which was another integer identifier that corresponded to 29 distinct values. Some examples would be 'discharged to home' and 'expired,' There was the admission source included which as well identified how the patient was admitted, whether that was from physician referral, emergency room, or transfer from a hospital. There were 21 distinct values for that attribute. The dataset included how long the patient was in the hospital for, as well as how the patient paid for the encounter. The medical specialty attribute was included in the dataset as this was the specialty of the admitting physician, and there were 84 distinct values identifying the specialties.

There was also information on the care and work that was done on the patient while they were admitted. The data contained the number of lab procedures performed, the number of procedures (other than lab tests), and the number of medications that were prescribed during the encounter. There were the results of the glucose serum test which indicated the range of the result or if the test was not taken. For many of the patients, there value was none indicating that the test was not performed. There was also the A1c test result, and the values indicated the range or 'none' if the test was not taken. Again, for a lot of the patients, the result was none. That meant that the hospital was not measuring their hyperglycemia levels through this test. Not monitoring the levels close enough could lead to improper treatment and consequent readmission to the hospital. The data included a change in medications attribute which indicated if there was a change to the patient's diabetes medications. The diabetes medications attribute indicated, 'yes' or 'no' to whether diabetes medication was prescribed. Then there were 24 features for the different types of diabetes medications. The values for those features were either up, down, steady, or no. The diabetic medications that were recorded in this dataset were metformin, repaglinide, nateglinide, chlorpropamide, glimepiride, acetohexamide, glipizide, glyburide, tolbutamide, pioglitazone, rosiglitazone, acarbose, miglitol, troglitazone, tolazamide, examide, sitagliptin, insulin, glyburide-metformin, glipizide-metformin, glimepiride-pioglitazone, metformin-rosiglitazone, and metformin-pioglitazone.

The data also included information pertaining to the history of the patient of the year before the documented hospital encounter. This data included the number of outpatient visits in the preceding year, number of inpatient visits in the preceding year, and the number of emergency visits in the preceding year. The last attribute of the dataset was the readmitted attribute which determined if the patient was readmitted <30 days after initial encounter, >30 days, or no for no readmission recorded.

TABLE I. DATASET CHARACTERISTICS

Attribute	Type	Example Value	Descripton
ENCOUNTER ID	Numeric	2278392	Unique identifier of an encounter
PATIENT NUMBER	Numeric	8222157	Unique identifier of patient
RACE	Nominal	Caucasian	Race of patient
GENDER	Nominal	Male	Gender of patient
AGE	Nominal	[10,20)	Age grouped in ten-year intervals
ADMISSION TYPE	Numeric	6	integer identifier corresponding to 9 distinct values, for example, emergency, urgent, elective, newborn, and not available
DISCHARGE DISPOSITION	numeric	25	Integer identifier corresponding to 29 distinct values, for example, discharged to home, expired, and not available.
ADMISSION SOURCE	Numeric	7	Integer identifier corresponding to 21 distinct values, for example, physician referral, emergency room, and transfer from a hospital
TIME IN HOSPITAL	Numeric	3	Days between admission and discharge
PAYER CODE	Nominal	Blue Cross Blue Shield, self-pay	Corresponding to 23 values of way to pay
NUMBER OF LAB PROCEDURES	Numeric	3	Number of lab test performed during the encounter
NUMBER OF MEDICATIONS	Numeric	2	Number of distinct generic names administered during the encounter
NUMBER OF OUTPATIENT VISITS	Numeric	1	Number of outpatient visits in the year preceding the encounter
NUMBER OF EMERGENCY VISITS	Numeric	0	Number of emergency visits in the year preceding the encounter
NUMBER OF INPATIENT VISITS	Numeric	1	Number of inpatient visits in the year preceding the encounter
NUMBER DIAGNOSES	Numeric	3	Number of diagnoses entered in the system
GLUSCOSE SERUM TEST RESULT	Nominal	>200, >300, none	Indicates the range of the result or if not taken
A1C TEST RESULT	Nominal	>8, none, >7	Indicates the range of the result or if the test was not taken
CHANGE OF MEDICATIION	Nominal	Yes	Indicates if there was a change in diabetic meds
DIABETES MEDICATIONS	Nominal	Yes	Indicates if any diabetic meds were prescribed
24 FEATURES FOR MEDICATIONS	Nominal	Up, down, steady, no	Values: “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
READMITTED	Nominal	No	Readmission of patient

To select the attributes that were most influential on determining whether the patient would be readmitted, I looked at the distributions of the attributes split by the target feature: readmitted. The attributes that were selected were the attributes that demonstrated a pattern. For example, there had to be a difference in the outcome of whether they were readmitted based upon the attribute value. Rather there had to be a difference in probability of ‘readmitted’ given the attribute value. As seen in Fig. 1, the probability of being readmitted increased tremendously if the patient had any prior inpatient encounters in the year prior to the recorded encounter. Therefore, this attribute was used for the analysis of predicting which patients would be readmitted. I also looked the frequency of the attribute in a distribution chart split by the target variable. I did this to determine how many patients were in each interval. As seen in Fig. 1., the probability of being readmitted was really high if the patient had more than three inpatient visit in the preceding year. However, if you look at Fig.2., there were very few patients that had more than three visits. I think this attribute was still useful because there was still a pattern seen; if there was at least one inpatient visit in the last year, the probability of being readmitted was higher than no readmission. I used this strategy for all the attributes to identify the most influential attributes that would have the greatest impact on the probability of readmission.

The next thing I did just to confirm that the attributes I felt were significant actually were, was conduct a gain of information test on all of the attributes. The attributes that were used for this predictive analysis are seen in Table II. in the results section with their gain of information score. The gain of information provided information about which attributes were relevant and provided any insight to whether the patient would be readmitted. I performed the information gain and information gain ratio, and the information gain ratio was more significant because the ratio provides normalized measure that considers the attribute’s intrinsic

characteristics. That way it can rank fairly even for attributes with varying scales and number of distinct values. Based on the information gain ratios, I kept all the attributes that had an information gain greater than 0. Consequently, I had 26 attributes that I used for my predictive analysis. I used all the attributes in Table I. except patient_nbr, race, gender, lab num procedures, num procedures, medical_specialty, glimepiride, acetohexamide, glipizide, glyburide, examide, citoglipton, glimepiride-pioglitazone, metformin-rosiglitazone, metformin-pioglitazone, payer_code, diag_1, diag_2, and diag_3.

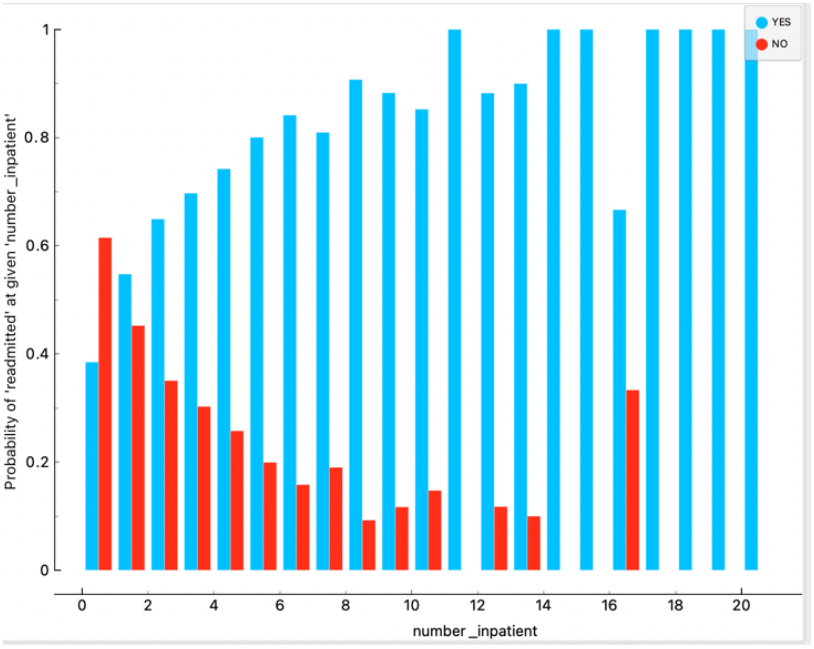

Fig. 1. The probability of readmission given the number of inpatient visits in the preceding year.

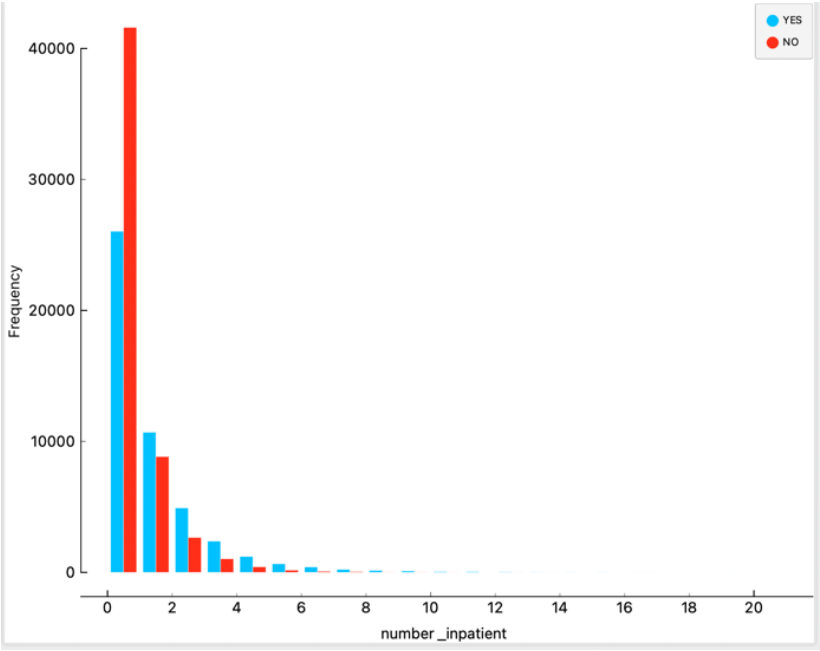

Fig. 2. The frequency of the number of inpatient visits in the preceding year split by readmission.

III. Methodology

I used Orange Canvas to create my predictive model to classify the readmission status of a diabetic patient as well as to conduct my analysis. I started my analysis by importing the diabetic data csv file. In the file widget, I was able to select the readmitted at the target variable. The next step was to view my data in a data table. This data was not processed yet, attributes that had no relevance to the problem I was trying to solve. My next step was using the Edit Domain widget. I wanted readmission status to either be 'yes' or 'no.' I changed the values that were <30 to yes, and I changed the values that were >30 to yes as well. I did this because I wanted a binary classification, and I thought a binary classification would yield more accurate and precise results. I also connected the distributions widget to the edit domain so I could visualize the probabilities of the readmission based on the value of the attribute. Then I connected the Preprocess widget to the Edit Domain widget as well. In the Preprocess widget, I normalized all the values to be between 0 and 1. I did this because the numeric values were all on different scales. Normalizing the values allowed the numeric attributes to be on a similar scale. This is important when working with variables with different units and varying ranges. It prevented the attributes with larger magnitudes from dominating the analysis and modeling process. Scaling the variables in this particular way ensured that each attribute contributed proportionately to the model fitting and analysis. Also, it helped with the model performance. As mentioned previously, the machine learning algorithm, kNN relies on distance-based calculations. Normalizing the attributes allows for the algorithm to converge faster as well as avoid any biases toward variables with larger values.

I also viewed the preprocessed data in a data table just to ensure the data was processed in the appropriate way. I attached the Data Table widget to the Preprocess widget. I connected the Preprocess domain to the Rank widget to determine the information gain ratios for each attribute [2]. Based on the gain of information ratios, I selected all the attributes that had a gain of information that was greater than zero. The gain of information ratios for the selected attributes are seen in Table II. There were 26 that provided insight to whether the patient would be readmitted. That meant that there were about 20 that provided zero information gain. The top information gain ratio was 0.073. That value would serve as the reference value to compare the other values to when determining if the gain of information provided by that attribute was significant enough. Since the reference ratio was only 0.073, I reasoned that 0.001 was provided enough insight to be included. I went back to the File widget, and all the attributes that had a gain of information ratio of 0 were skipped.

I returned to the Preprocess widget, and I attached the Data Sampler widget. I set the proportion of train data to be 70%. I connected the Data Sampler to the Test and Score widget. The Test and Score widget was set to cross validation. I also connected the k Nearest Neighbor (kNN), Random Forest, and Logistic Regression to the Test and Score widget. I chose these three statistical methods and algorithms all for specific reasons. I chose the kNN because it groups similar instances together based on feature similarity, and the classification is based on the majority class among the k nearest neighbors. This method can capture complex relationships that are not linear, and it can handle numeric and categorical features. I chose the Random Forest method for its ability to handle complex datasets and interactions between the features. I chose Logistic Regression because it is widely used for binary classification problems, so it would be good to classify the readmission status of the patients. It also provides interpretable coefficients which allowed for the understanding of which predictor variables had an effect on the readmission status. I viewed the results of these three methods, and then I experimented with changing the parameters of each model.

The next step was to examine the different models to determine which one was the best model for this problem and dataset. The k Nearest Neighbor algorithm functions by predicting the label of the new data point based on the labels of its k nearest neighbors in the training dataset. I sampled different k values to find the optimal number. The optimal number was 7. I used the Euclidian metric. Random Forest was another algorithm attached to the Test and Score widget, and this is a learning method that combines multiple decision trees to improve the predictive accuracy [3]. The number of trees was altered to determine the optimal number of trees. The number of trees I decided on was 100 trees based on experimentation with different values. The last model that I used was Logistic Regression. This statistical technique models the relationship between predictor variables and the log-odds of the outcome, assuming a linear relationship. Coefficients are estimated using maximum likelihood estimation. The sigmoid function is applied to convert log-odds to probabilities [4]. I then experimented with the regularization parameter. I tried 1, 0.5, and 40. Anything above 1 resulted in the same thing as the results from 1. Anything lower than 1 resulted in lower quality results. Then I experimented with the number of folds in the cross-fold examination. I looked at 10 folds and 20 folds. The right number of folds that provided the best results was 10-fold. I also then experimented with the proportion of training data. I tried 80-20, and the results were indicating better performance. I changed the split back to 70-30.

After fine tuning the different models, I examined the area under the ROC curve, precision, the F1, the classification accuracy, and recall. The model that resulted in the highest of all the evaluations metrics was the Random Forest model. I connected the Confusion Matrix widget and the ROC analysis widget to the Test and Score widget. After visualizing the results from the models, I went back to the Test and Score widget. I changed the selection from cross validation to test on test data. This data had not been tested on yet. This would prove how well the model performed at classifying the readmission status of the patients. The evaluation metrics were analyzed in the Test and Score widget, and the results were visualized in the confusion matrix and the ROC analysis.

IV. Results and Discussion

The task of this predictive model was to determine the readmission status of the diabetic patients based on their medical history, medical care in the hospital, and their demographics. I calculated the gain of information ratio for each attribute to determine the ones that were most influential to the class label. The results are seen below in Table II. All of the included attributes had a gain of information ratio greater than 0 and were deemed relevant and significant to the classification of the readmission status of the patient. The lowest ratio was 0.001 which might seem like it is quite small, but small is subjective. The largest value was only 0.073. Upon comparing these values, it was decided that a ratio of 0.001 was influential enough to keep. There were 7 attributes with a ratio of 0.001. If I chose not to include the attributes with that ratio, it would have been a 0.007 loss of information.

TABLE II. Gain of Information Ratios for Attributes

Attribute	**Information Gain Ratio**
Troglitazone	0.073
Glipizine-pioglitazone	0.050
Number_inpatient	0.029
Number_emergency	0.02
Tolazomide	0.02
Miglitol	0.010
Acarbose	0.010
Number_outpatient	0.009

Number_diagnoses	0.007
Diabetes_Med	0.005
Repaglinide	0.005
Medical_specialty	0.004
Admission_source_id	0.004
Chlorpropamide	0.003
Dishcarge_disposition_id	0.003
Insulin	0.003
Tolbutamide	0.003
Change	0.002
Num_medications	0.002
Time_in_hospital	0.002
Encounter_id	0.002
Max_glu_serum	0.002
Nateglinide	0.002
Glyburide-metofrmin	0.001
Admission_type_id	0.001
Rosiglitazone	0.001
Age	0.001
Pioglitazone	0.001
Metofrmin	0.001
A1Cresult	0.001

Then I experimented with a different number of k neighbors to determine the optimal number. The optimal number was 7 k neighbors. As seen in Fig. 3., the AUC is higher with 8 k neighbors, but the accuracy starts to decrease, along with the Fl and the recall. 7 k Neighbors provided higher evaluation metrics than any of the results that had a lesser number of neighbors. For these evaluation metric value, a higher value means the model performed better. The area under the ROC curve (AUC) is a metric used to evaluate the performance of a binary classification model. The ROC curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various classification thresholds. The AUC represents the area under this curve. An example of this type of plot is seen in Fig. 4. A value that is closer to 0.5 indicates random guessing. That means the model has no discriminatory power and performs the same as just a random guess to classify. If the value is closer to 1, it indicates that the model can distinguish better between the positive and negative instances. The model would achieve a higher TPR while still maintaining a lower FPR across the various thresholds. The model with 8 k neighbors had the highest AUC, but it was only 0.002 higher than the model with 7 neighbors, and other values were starting to decrease. The other performance metrics were crucial as well to examine. The classification accuracy (CA) was determined by (1). TP means true positives, TN is true negatives, FP is false positives, and FN is false negatives.

$$Accuracy = (\#TP + \#TN)/ (\#TP + \#TN + \#FP + \#FN) \quad (1)$$

A higher CA value means that the model was able to correctly classify a higher percentage of the instances correctly. That is part of the reason I chose the model with 7 neighbors over the other kNN models because with 7 neighbors, the model had the highest accuracy. However, accuracy alone should not be used on its own to determine the model's performance because there was a different number of instances in the 'yes' and 'no' classes. This can cause misleading information as the model could just predict the class label that has the majority of instances every time and get a high CA value. Therefore, I also looked at the precision, recall, and F1 as well. The precision is the percentage of true positives predictions among all positive predictions. The precision is calculated in (2). For the model with 7 neighbors, the precision was 0.565 which meant that out of the instances that were classified as positive, only 56.5% of the instances were actually positive. This was not a very precise result, especially in comparison to the other algorithms.

$$Precision = \#TP / (\#TP + \#FP) \quad (2)$$

I also looked at the recall which is the true positivity rate. It. measures the proportion of correctly predicted positive instances out of all the actual positive instances in the dataset. A higher recall indicates better performance because that means that a higher percentage of the positive instances were correctly classified by the model. The recall for this model was 0.565 which meant that 56.5% of the actual positive instances were identified. This was not a significant number again compared to the other model types.

The next performance measure that looked at was the F1, which is the harmonic mean of the precision and recall. The formula for the F1 is seen in (3).
The F1 was important to look at especially for this analysis because both the false positives and false negatives were important to consider. A higher F1 score means that the model is capable of capturing a significant portion of the positives as well as maintaining a high accuracy rate for positive predictions.

F1 = 2(precision * recall) / (precision + recall)* (3)

The F1 value for 7 k neighbors model was 0.567 which meant that it only had a moderate balance of precision and recall. The model is reasonable effective at making accurate predictions while capturing a reasonable proportion of actual positives instances.

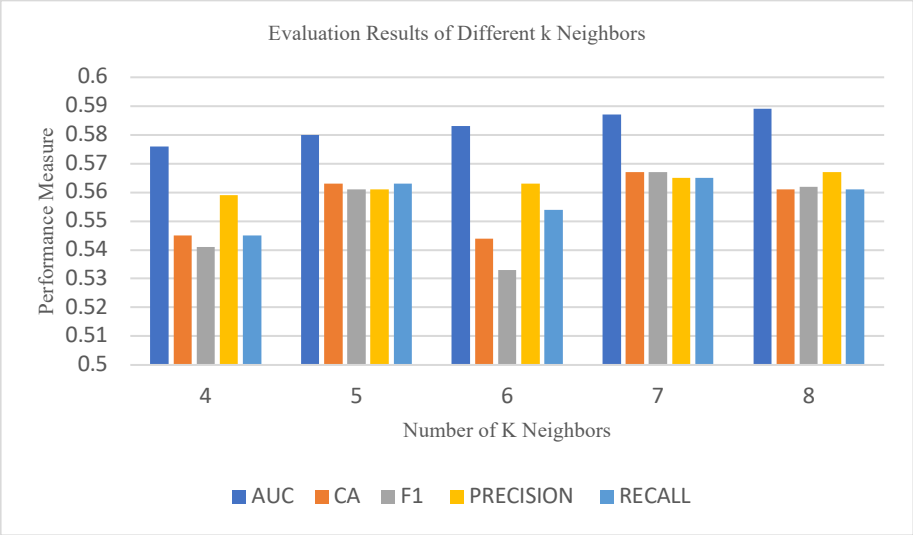

Fig. 3. The Evaluation metric result values for varying number of k Neighbors.

I then experimented with the number of decision trees in the Random Forest. The evaluation metric results for the values of the different number of trees are shown below in Fig. 4. The optimal number of trees was found to be 100. Anything over 100 did not improve the results of the evaluation metrics, but rather just stayed the same. If a number of trees less than 100 were used, then they yielded in lower quality results for the model. With 100 trees, the area under the ROC curve was 0.687, the classification accuracy was 0.64, the F1 was 0.637, the precision was 0.639, and the recall was 0.64.

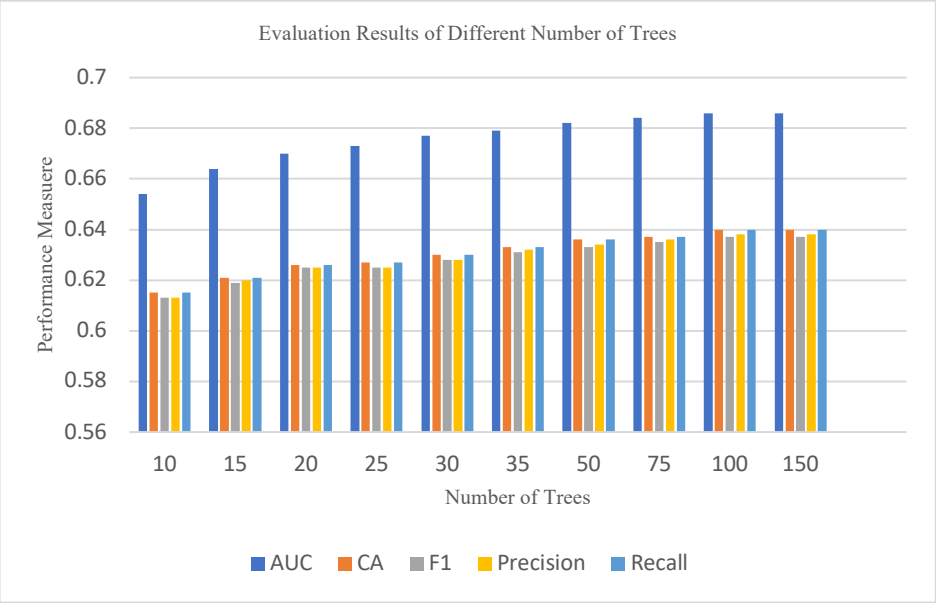

Fig. 4. The evaluation metric results of the Random Forest method with varying number of trees used.

For the Logistic Regression model, I experimented with the type of regularization. L1 and L2 resulted in the same values. I manipulated the regularization parameters to visualize the variation in the evaluation metrics as see below in Table III. The optimal regularization parameter was a value of 1. As you can see if you use a value higher than that, the values stay the same meaning that the quality of the results are not increasing. If you use a value lower than 1, such as 0.5, the quality of the results decreases.

TABLE III. EVALULUATION METRIC RESULTS OF DIFFERENT REGULARIZATON PARAMETERS

Regularization	Area under ROC Curve	Classification Accuracy	F1	Precision	Recall
0.5	0.661	0.6	0.606	0.623	0.62
1	0.662	0.622	0.608	0.624	0.622
10	0.662	0.622	0.608	0.624	0.622

Once I had tuned all the hyperparameters to the different models to generate the best performing models, I experimented with the number of fold and train-test split. However, my original parameters worked the best. I decided to stick with 10-fold cross validation, and a 70-30 split. I tried using a 5-fold cross validation, as well as a 20-fold, and the quality of the results plummeted. I tried to conduct an 80-20 split between train and test data, and again the quality of the results decreased. In Table IV, the evaluation results for the models that were created using the training data are displayed. Based upon the results in Table V, it was evident that the Random Forest was the optimal model to use for this analysis. The Random Forest model had the highest area under the ROC curve, and this is evident in Table IV. and in Fig. 5. The area under the curve was 0.687 in comparison to the other values of kNN with 0.589, and Logistic Regression with a value of 0.622. In Fig. 5., the orange line represents the Random Forest model, and since it has the largest area under the curve that means the model could better distinguish between the positive and negative instances. Random Forest also provided the highest classification accuracy with a value of 0.64 which means that the model was able to correctly classify 64% of the actual positives as positives. The precision was the highest value as well with a value of 0.639 which means the model has a moderate level of predicting positives. Approximately 63.9% of all the precited positives were actually positive. The recall was the highest with a value of 0.64 meaning that the model was able to capture 64% of the positive instances. Lastly, it also had the highest F1 value meaning it had the best balance between the recall and precision values.

TABLE IV. EVALUATION RESULTS OF MODELS FOR TARGET USING CROSS-VALIDATION

Model	Area under ROC Curve	Classification Accuracy	F1	Precision	Recall
K Nearest Neighbor	0.589	0.561	0.562	0.567	0.561
Random Forest	0.687	0.64	0.637	0.639	0.64
Logistic Regression	0.662	0.622	0.608	0.624	0.622

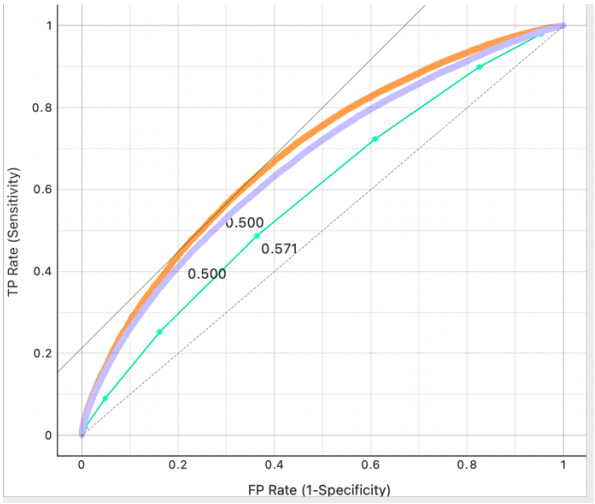

Fig 5. Area under the ROC Curve; orange represents Random Forest; purple corresponds to Logistic Regression and green goes with kNN.

After the best suited model was chosen, it was time to test it with the testing data. The performance results of the models are seen in Table V when used with the testing data. Again, it was proved that the Random Forest model was the best to use for this analysis. The area under the ROC curve was 0.691, the CA was 0.64, the F1 was 0.638, the precision was 0.639, and the recall was 0.64. All of these values were similar to the values from the training data, if not better. Another way to visualize the results was in the confusion matrix as seen in Table VI. The data highlighted in green in the confusion matrix shows the instances that were correctly classified either as yes or no. The green box on the left shows the true positives, and the green box on the right shows the true negatives. The salmon-colored boxes show the false predictions. The salmon box on the left shows the false positives where the classification was no, but it was predicted to be yes. The salmon-colored box on the right shows the false negatives where it was actually yes, but it was classified as no. Overall, this model demonstrated reasonable accuracy, precision, recall, and F1. Based on the selected attributes for this dataset, this model was able to predict with 64% accuracy, 63.9% precision, and 64% recall the readmission status of a diabetic patient.

TABLE V. EVALUATION RESULTS OF MODELS FOR TARGET USING TEST DATA

Model	Area under ROC Curve	Classification Accuracy	F1	Precision	Recall
K Nearest Neighbor	0.593	0.571	0.569	0.569	0.571
Random Forest	0.691	0.64	0.638	0.639	0.64
Logistic Regression	0.661	0.617	0.603	0.62	0.617

TABLE VI. CONFUSON MATRIX OF RANDOM FOREST MODEL WITH TEST DATA

		Predicted		
		Yes	No	Totals
Actual	Yes	7833	6261	14094
	No	4727	11708	16435
	Totals	12560	17969	30529

V. CONCLUSIONS

This dataset consisted of demographic, medical history, and medical care information on diabetic patients who were admitted to various United States hospitals over a 10-year span. The goal of the assignment was to create a predictive model to determine if the diabetic patient will be readmitted into the hospital after the initial documented encounter based on certain attributes that were deemed relevant. To build to predictive model, I used Orange Canvas, and I first preprocessed the data which consisted of normalizing the numeric attribute values and converting the readmitted class labels to be binary. I then ranked the attributes based on the gain of information ratios. I selected the attributes that provided a gain of information ratio greater than 0. The attributes that played the biggest role in determining which patients would be readmitted were troglitazone, glipizide-pioglitazone, number_inpatient, number_emergency, and tolazomide.

Then I split the data into 70% training data and 30% testing data. I experimented with various machine learning algorithms and methods and experimented with the hyperparameters of each model using 10-fold cross validation. I tested kNN, Random Forest, and Logistic Regression. I found the model that provided the best performance results based on the AUC, CA, F1, precision, and recall. The model that functioned best for this task was the Random Forest. It had the highest value for each metric indicating that it provided the most accurate and precise results for the classification of readmission status. Once the Random Forest model was selected, I tested the model with the test data. All of the results were greater than or equal to 0.638. That meant that this model could provide reasonably accurate classification results given the relevant attributes used. This predictive model demonstrated that there was a pattern and association between certain attributes and whether the patient was readmitted. This information could be implemented in hospitals, so healthcare workers can look at the patterns and figure out which of their patients need more care and are at a higher risk for readmission. For example, if the patient had a high number of inpatient visits and high number of emergencies in the preceding year, then they were more at risk of readmission. Or they could use this predictive model to incorporate all the patient attributes that were significant to predict if the patient was going to be readmitted. If the classification was yes, then they could take specific precautions to decrease the patient's chances of readmission.

REFERENCES

[1] B. Strack, J. P. DeShazo, C. Gennings, J. L. Olmo, S. Ventura, K. J. Cios, and J. 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, pp. 1-11, 2014.

[2] C.E. Shannon, "A mathematical theory of communication," The Bell System Technical Journal, vol 27, pp. 379-423, 623-656, July, October, 1948.

[3] L.Breiman, "Random Forest," Statistics Department, University of California, Berkely, CA 94720, January 2001.

[4] D.R.Cox, "The estimation of probability density function and its cumulative distribution function, with application in discriminant analysis," Biometrika, vol 45, no. 3-4, pp. 515-528, Dec 1958.