SMARTBRIDGE
Let's Bridge the Gap

PREDICTING THE READMISSION TO HOSPITALS FOR DIABETIC PATIENTS

Members Involved:

1. MOGILI VISHAL GOUD
2. PRUTHVINATH REDDY SIREDDY
3. VARADA SAKETA YASASVI
4. EMERALD SAGAR KALI

INTRODUCTION:

Python along with packages like NumPy, scikit-learn, iPython Notebook, and matplotlib form the basis to start your AI project.

NumPy is used as a container for generic data comprising of an N-dimensional array object, tools for integrating C/C++ code, Fourier transform, random number capabilities, and other functions.

Another useful library is pandas, an open source library that provides users with easy-to-use data structures and analytic tools for Python.

Matplotlib is another service which is a 2D plotting library creating publication quality figures. You can use matplotlib to up to 6 graphical users interface toolkits, web application servers, and Python scripts.

Your next step will be to explore k-means clustering and also gather knowledge about decision trees, continuous numeric prediction, logistic regression, etc.

Some of the most commonly used Python AI libraries are AIMA, pyDatalog, SimpleAI, EasyAi, etc. There are also Python libraries for machine learning like PyBrain, MDP, scikit, PyML.

OBJECTIVES OF RESEARCH:

In this project we will demonstrate how to build a model predicting readmission in Python using the following steps

- data exploration
- feature engineering
- building training/validation/test samples
- model selection
- model evaluation

PROBLEM STATEMENT:

One patient population that is at increased risk of hospitalization and readmission is that of diabetes. Diabetes is a medical condition that affects approximately 1 in 10 patients in the United States. Patients with diabetes have almost double the chance of being hospitalized than the general population. Therefore, in this project, we will focus on predicting hospital readmission for patients with diabetes. We are going to predict if a patient with diabetes will be readmitted to the hospital within 30 days. For this we are going to create a model that is able to predict the patients with diabetes with highest risk of being readmitted within 30 days.

REVIEW OF LITERATURE :

As the healthcare system moves toward value-based care, The Centers for Medicare and Medicaid Services (CMS) has created many programs to improve the quality of care of patients. One of these programs is called the Hospital Readmission Reduction Program (HRRP), which reduces reimbursement to hospitals with above average readmissions. For those hospitals which are currently penalized under this program, one solution is to create interventions to provide additional assistance to patients with increased risk of readmission. But how do we identify these patients? We can use predictive modelling from data science to help prioritize patients.

DATA COLLECTION:

The dataset represents 10 years (1999-2008) of clinical care at 130 US hospitals and integrated delivery networks. It includes 50 features representing 101766 diabetes patients and hospital outcomes. Information was extracted from the database for encounters that satisfied the following criteria:

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

The data contains such attributes as patient number, race, gender, age, admission type, time in hospital, medical specialty of admitting physician, number of lab test performed, HbA1c test result, diagnosis, number of medications, diabetic medications, number of outpatients, inpatient, and emergency visits in the year before the hospitalization, etc.

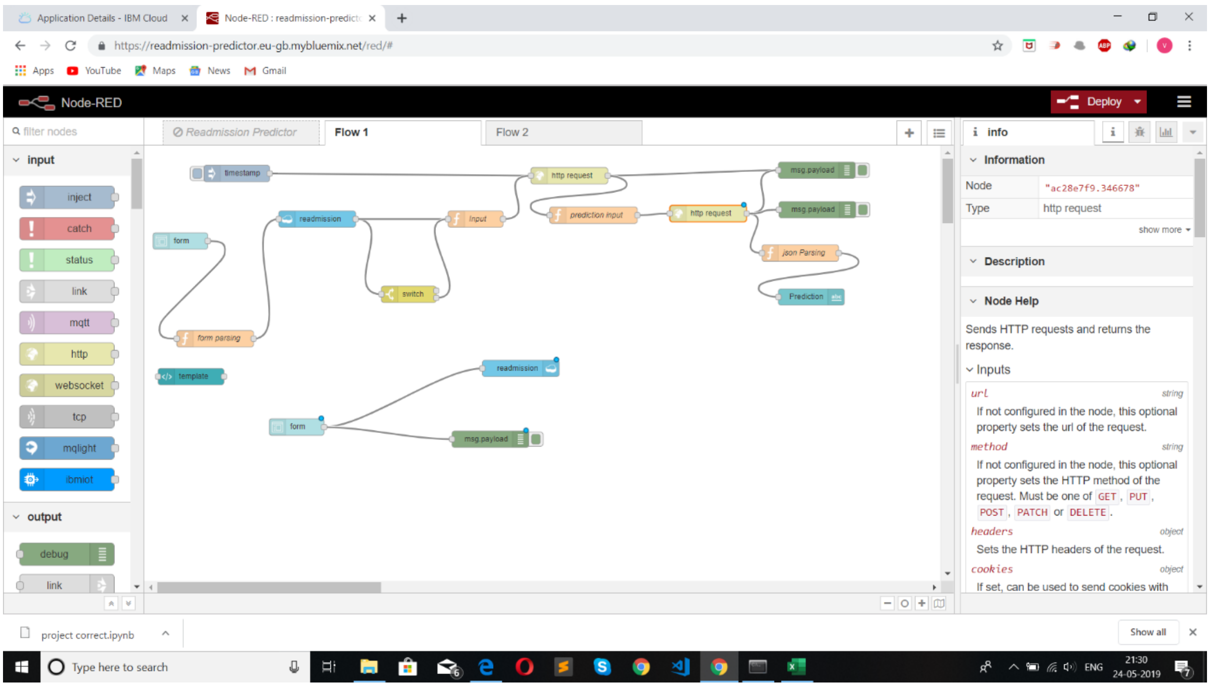
Source: UCI Machine Learning Repository, https://archive.ics.uci.edu/ml/datasets/Diabetes+130-US+hospitals+for+years+1999-2008

METHODOLOGY:

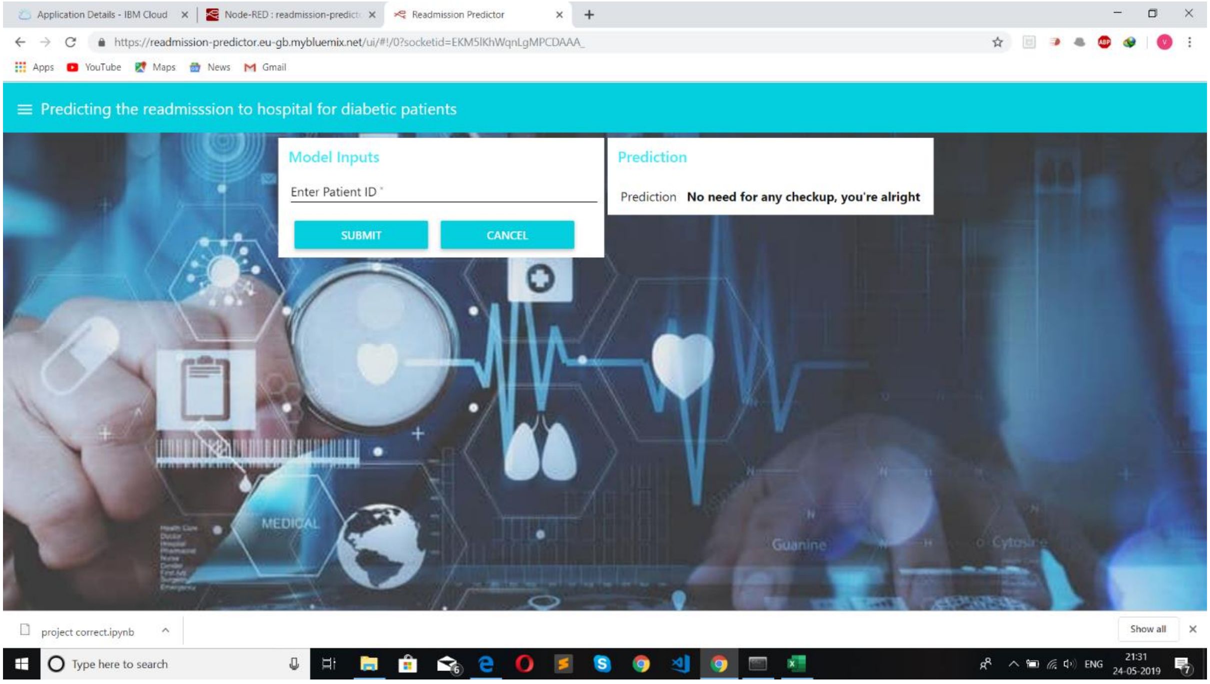
Exploratory Data Analysis:

Figures and Tables:

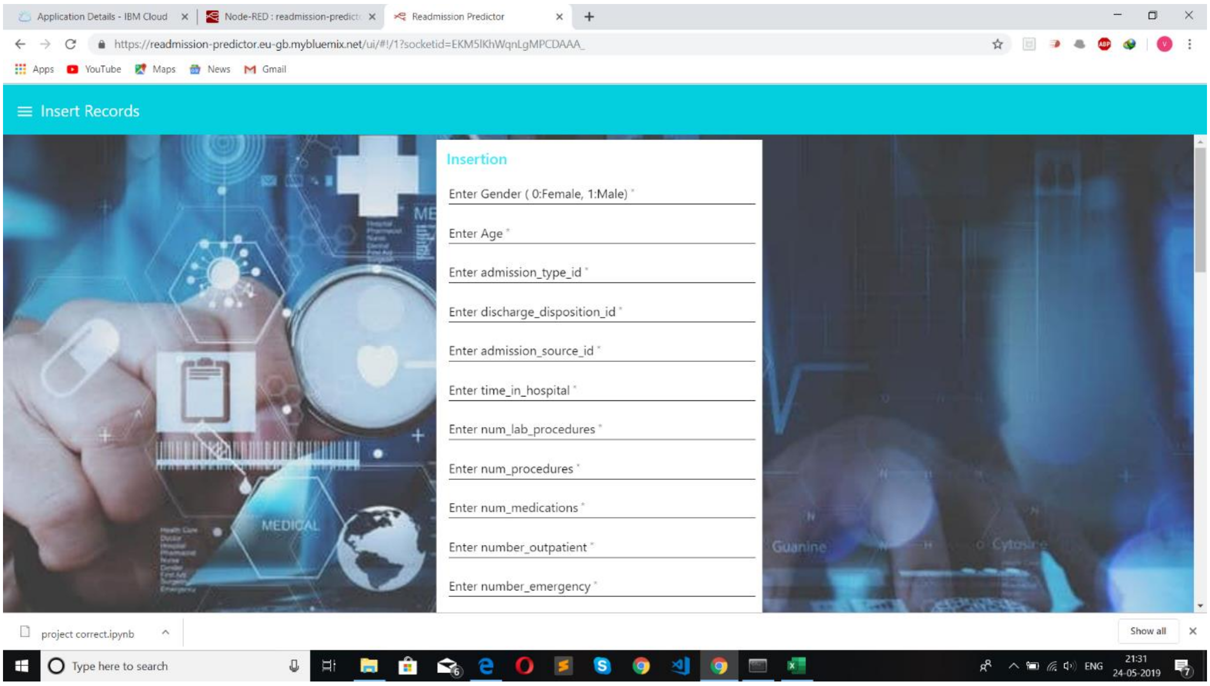
Node-RED Flow:



User Interface:



Input to Database:



Data Modelling:

Importing packages

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import keras
```

```
Using TensorFlow backend.
```

If we look at the IDs_mapping.csv we can see that 11,13,14,19,20,21 are related to death or hospice. We should remove these samples from the predictive model

```
df = df.loc[~df.discharge_disposition_id.isin([11,13,14,19,20,21])]
```

Label encoding columns

```
from sklearn.preprocessing import LabelEncoder
```

```
lb = LabelEncoder()
```

```
x = df.iloc[:,0:38].values
```

```
x[:,0] = lb.fit_transform(x[:,0])  ##gender
```

```
x[:,13] = lb.fit_transform(x[:,13])
```

```
x[:,14] = lb.fit_transform(x[:,14])  ###
```

```
x[:,15] = lb.fit_transform(x[:,15])  ##discharge
```

```
x[:,16] = lb.fit_transform(x[:,16])  ##admission_source_id
```

```
x[:,17] = lb.fit_transform(x[:,17])  ##time in hourd
```

```
x[:,18] = lb.fit_transform(x[:,18])   ##payer_code
```

```
x[:,19] = lb.fit_transform(x[:,19])   ###medical_speciality
```

```
x[:,20] = lb.fit_transform(x[:,20])    ##num_lab_procedure
```

Splitting train and test data

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.2,random_state=0)
```

Standardization of dataset

```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
x_train = sc.fit_transform(x_train)  ## finding out the best value and transforming
x_test = sc.transform(x_test)   ## here again we use fit then the best value which we found out in the x_train wich is stored in
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\utils\validation.py:595: DataConversionWarning: Data with input dtype object
was converted to float64 by StandardScaler.
  warnings.warn(msg, DataConversionWarning)
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\utils\validation.py:595: DataConversionWarning: Data with input dtype object
was converted to float64 by StandardScaler.
  warnings.warn(msg, DataConversionWarning)
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\utils\validation.py:595: DataConversionWarning: Data with input dtype object
was converted to float64 by StandardScaler.
  warnings.warn(msg, DataConversionWarning)
```

Creating layers

```
import keras
from keras.models import Sequential
from keras.layers import Dense
```

```
model=Sequential()
model.add(Dense(10,input_dim=38))
model.add(Dense(100,activation = 'softmax'))
model.add(Dense(100,activation = 'softmax'))

model.add(Dense(3,activation = 'softmax'))
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
       # for regression no need of activation function
```

```
WARNING:tensorflow:From C:\ProgramData\Anaconda3\lib\site-packages\tensorflow\python\framework\op_def_library.py:263: colocate_
with (from tensorflow.python.framework.ops) is deprecated and will be removed in a future version.
Instructions for updating:
Colocations handled automatically by placer.
```

Training the model

```
In [59]: from keras.utils import to_categorical
         y_train = to_categorical(y_train)
```

```
In [60]: model.fit(x_train,y_train,epochs= 100,batch_size = 32)
```

```
Epoch 92/100
79474/79474 [==============================] - 6s 70us/step - loss: 0.8784 - acc: 0.5879
Epoch 93/100
79474/79474 [==============================] - 6s 70us/step - loss: 0.8782 - acc: 0.5874
Epoch 94/100
79474/79474 [==============================] - 6s 70us/step - loss: 0.8780 - acc: 0.5880
Epoch 95/100
79474/79474 [==============================] - 6s 71us/step - loss: 0.8780 - acc: 0.5883
Epoch 96/100
79474/79474 [==============================] - 6s 70us/step - loss: 0.8779 - acc: 0.5879
Epoch 97/100
79474/79474 [==============================] - 6s 70us/step - loss: 0.8777 - acc: 0.5882
Epoch 98/100
79474/79474 [==============================] - 6s 71us/step - loss: 0.8774 - acc: 0.5885
Epoch 99/100
79474/79474 [==============================] - 6s 70us/step - loss: 0.8773 - acc: 0.5886
Epoch 100/100
79474/79474 [==============================] - 6s 70us/step - loss: 0.8774 - acc: 0.5881

Out[60]: <keras.callbacks.History at 0x20cb29e2ba8>
```

Model result

```
y_train
```

```
array([[0., 0., 1.],
       [0., 0., 1.],
       [0., 1., 0.],
       ...,
       [0., 0., 1.],
       [0., 1., 0.],
       [0., 0., 1.]], dtype=float32)
```

```
y_pred = model.predict(x_train[0:1])
```

```
y_pred = model.predict_classes(x_train[0:1])
```

```
y_pred
```

```
array([2], dtype=int64)
```

FINDINGS AND SUGGESTIONS:

The best model for predicting the patients with diabetes with highest risk of being readmitted with 30 days was a gradient boosting classifier with optimized hyperparameters. The model was able to catch 58% of the readmissions and is about 1.5 times better than just randomly picking patients.

CONCLUSION :

Hospital readmission of patients with diabetes is an important health care quality measure and driver of costs. Major risk factors for readmission include lower socioeconomic status, racial/ethnic minority, greater burden of comorbidities, public insurance, emergent or urgent admission, and a history of recent prior hospitalization. Certain hospitalized patients with diabetes may be at higher risk of readmission than those without diabetes. Multiple health system and patient-related barriers to reducing readmission rates exist. A mix of expert opinion and a handful of mostly small studies provide a number of potential strategies for reducing readmission risk, including inpatient education, specialty care, better discharge instructions, coordination of care, and post-discharge support. The diabetes-specific strategies such as diabetes education, intensifying therapy, and outpatient diabetes care tend to be more effective in poorly controlled patients and tend to reduce. Through this project, we created a model that is able to predict the patients with diabetes with highest risk of being readmitted within 30 days. The model was able to catch 58% of the readmissions and is about 1.5 times better than just randomly picking patients. Overall, we believe many healthcare data scientists are working on predictive models for hospital readmission.