PROJECT BUILD–A–THON

ON

DIABETES PREDICTION SYSTEM BASED ON LIFE STYLE

BY

Dr. M. VIJAYA LAXMI

(Associate Professor, Dept. of ECE, SRIKALAHASTEESWARA INSTITUTE OF TECHNOLOGY, SRIKALAHASTI, A.P.)

All India Council for Technical Education

&

IBM in collaboration with Smart Bridge

October, 2020

ACKNOWLEDGEMENT

I take this opportunity to express my profound gratitude and deep regards to the Organizers of All India Council for Technical Education (AICTE) and IBM in collaboration with Smart bridge to share a unique IBM Hack Challenge and Academic Initiative program-"GuruCool" for faculty members.

I express my heartfelt thanks to all the experts and mentors for providing us excellent training.

I extend my special thanks to Mr.Hemanth for his constant help in completing Project Build-a-Thon.

Abstract

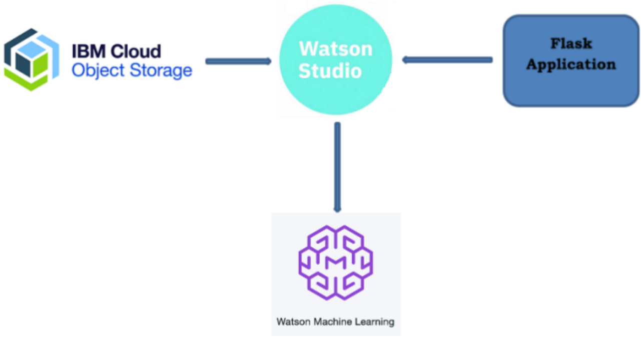
Diabetes is a global pandemic. In today’s digital world people are prone to many health issues due to the sedentary lifestyle. Evidence-based medicine is a powerful tool to help minimize treatment variation and unexpected costs. Large amount of healthcare data such as physician notes, medical history, medical prescription, lab and scan reports generated is useless until there is a proper method to process this data interactively in real - time. In this world filled with the latest technology, healthcare professionals feel more comfortable to utilize the social network to treat their patients effectively. To achieve this, an effective framework is needed which is capable of handling large amount of structured, unstructured and live streaming data about the patients from their social network activities.

In this project Machine learning is applied to extract insights from the heterogeneous data of the patients. It provides individual recommendations based on the past learning experience and the patterns extracted from clinical data. Combination of information retrieval and machine learning can be used for medical database classification.

In this, we need to diagnostically predict whether or not a patient has diabetes, based on certain diagnostic measurements included in the dataset. Predictor variables include the number of pregnancies the patient has had, their BMI, insulin level, age, and so on. In particular, all patients here are females at least 21 years old of Pima Indian heritage.

Developed an end-to-end web application that predicts the probability of females having diabetes. The application must be built with Python-Flask or Django framework with the machine learning model trained and deployed on IBM Watson Studio.

Figure 1: Proposed Technical Architecture:



CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

There are three types of diabetes Type-I diabetes, Type-II diabetes and gestational diabetes. Type-I diabetes is occurred when pancreas, due to some abnormality, does not produce insulin or produces very little insulin. This is called as juvenile diabetes or insulin-dependent diabetes. The treatment does not cure Type-I diabetes but it aims to control blood sugar levels with insulin, diet and lifestyle to prevent complications. Gestational diabetes occurs in pregnant women where it is observed high sugar levels during pregnancy in them. Later there are chances that gestational diabetes can be converted into Type-II diabetes.

Type-II diabetes also called as diabetes mellitus is a metabolic disorder which causes sugar levels in body to raise up. Unlike Type-I diabetes, Type-II diabetes can be reversible. The treatment can be different from one person to other person. Some need only lifestyle changes like reducing weight, leading healthy lifestyle etc. and doesn’t require taking insulin. While others need to take medical treatment which involves medicines, insulin injections and following good lifestyle to maintain sugar levels. During metabolic activity intake of food is converted to energy. This process requires hormone called insulin which helps in converting sugars to energy. Type-II diabetes is caused when body slowly loses its capacity to absorb insulin, which actually controls the sugar level, thereby sugar levels in body will not be controlled and hence higher sugar values in Type-II diabetes patients. This is also referred as “adultonset” diabetes as this is developed in later stages of life. This has other name “insulin resistance” as body is showing resistance to absorb insulin. This is more common when compared to other types of diabetes. Statistics show that out of 100 diabetes patients, 90 have Type-II diabetes. An estimated 1.6 million deaths were directly caused by diabetes. Another 2.2 million deaths because of high glucose in blood. World Health Organization (WHO) estimates that diabetes was the 7th leading cause of deaths. India is in second position with highest number of diabetes patients. With the help of data mining, machine learning techniques and technology the risks for Type-II diabetes can be identified early and with proper treatment Type-II diabetes can be controlled there by reducing negative impacts of diabetes.

Machine Learning

The main purpose of machine learning is to build the system that should learn from previous experiences and complete the tasks by itself without any need of external instruction.

Important part of machine learning is algorithms with which different models can be established. Machine learning models will predict output for a particular given input. The inputs to generate machine learning model are sample dataset and machine learning algorithms are chosen in such a way that they suit the attributes in the sample data. The steps in the process are:

1. Training data set (Pima Indian Diabetes Dataset) is given as input.

The objective of the dataset is to diagnostically predict whether or not a patient has diabetes, based on certain diagnostic measurements included in the dataset.

The dataset contains seven sixty eight instances and nine features. The dataset features are:

- Total number of times pregnant
- Glucose/sugar level
- Diastolic Blood Pressure
- Body Mass Index (BMI)
- Skin fold thickness in mm
- Insulin value in 2 hour
- Heredity factor – Pedigree function
- Age of patient in years

2. Selected machine learning algorithm is applied on the data. In this stage algorithm learns the patterns in the data.

3. The model is built after algorithm completes the learning of patterns in given data. After the model is ready, whenever new data is given for prediction to the model, the model predicts the output for that particular input.

1. **The IBM Watson Studio** learning path demonstrates various ways of using IBM Watson Studio to predict customer churn. It ranges from a semi-automated approach using the AutoAI Experiment tool to a diagrammatic approach using SPSS Modeler Flows to a fully programmed style using Jupyter notebooks for Python.
2. **Node-RED:** It uses a visual flow-based programming paradigm and is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways. It provides a browser-based editor that makes it easy to wire together flows using the wide range of **nodes** in the palette that can be deployed to its runtime in a single-click.

There are three types of nodes:

- Input **Nodes** (e.g. inject)
- Ouput **Nodes** (e.g. debug)
- Processing **Nodes** (e.g. function)

Working of the Proposed System:

The proposed system focuses using algorithms combinations. The basic classification algorithms and steps followed are:

Step – 1: Data collection and dataset preparation

Step – 2: Developing a recommender system based on predictions using **AutoAI** automatically prepares data, applies algorithms, and attempts to build model pipelines best suited for the data and use case.

Rule 1: No Diabetes Range

If FPG has a level between 70 and 100 mg/dL (3.9 and 5.6 mmol/L), then it indicates- no diabetes range. If the blood glucose level below 125 mg/dL in CGTT, then it indicates- no diabetes range. If HBA1C value is below 97 mg/dL, then it indicates- no diabetes range.

Rule 2: Pre-diabetes Range

If FPG ranges from 100 mg/dl to 125 mg/dl and CGTT ranges from 140 mg/dl to 199 mg/d and HBA1C test values lie in range 97-154 mg/dL, it indicates pre-diabetes range.

Rule 3: Diabetes Range

If FPG is 126 mg/dl or more and CGTT is 200 mg/dl or more and HBA1C is greater than 180 mg/dL, it indicates diabetes.

i. Preparing training data: Collect model data in a CSV file that is less than 100MB. AutoAI will transform the data and impute missing values.
ii. Open the AutoAI tool
iii. Create or open a project
iv. Specify details of the model and training data and launch AutoAI:
Training data is used to train the model and used to measure the performance of the model.

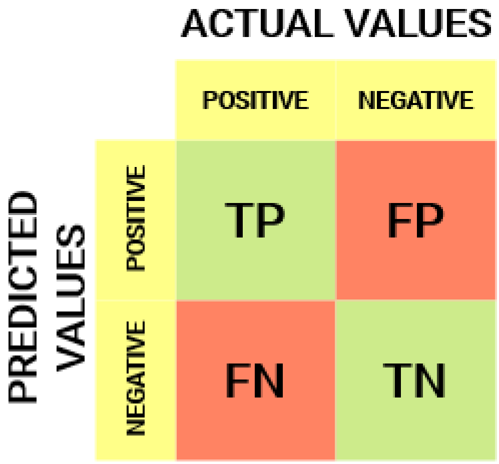
We have datasets in which different columns have different units – like one column can be in kilograms, while another column can be in years etc. So, to give importance to different units, we need feature scaling. Feature transformation (FT) refers to family of algorithms that create new features using the existing features. These new features may not have the same interpretation as the original features, but they may have more discriminatory power in a different space than the original space.

Prediction settings, to optionally specify which algorithms AutoAI should consider for pipeline creation. Only checked algorithms will be considered during the model selection phase of the experiment. For binary classification models you can also edit the positive class.

Confusion Matrix: There are plenty of ways to gauge the performance of classification model but none have stood the test of time like the confusion matrix. It helps us evaluate how our model performed, where it went wrong and offers us guidance to correct our path.

A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model. This gives us a holistic view of how well our classification model is performing and what kinds of errors it is making.

For a binary classification problem, we would have a 2 x 2 matrix as shown below with 4 values:



		ACTUAL VALUES	
		POSITIVE	NEGATIVE
PREDICTED VALUES	POSITIVE	TP	FP
	NEGATIVE	FN	TN

True Positive (TP)

- The predicted value matches the actual value
- The actual value was positive and the model predicted a positive value

True Negative (TN)

- The predicted value matches the actual value
- The actual value was negative and the model predicted a negative value

False Positive (FP) – Type 1 error

- The predicted value was falsely predicted
- The actual value was negative but the model predicted a positive value
- Also known as the **Type 1 error**

False Negative (FN) – Type 2 error

- The predicted value was falsely predicted
- The actual value was positive but the model predicted a negative value
- Also known as the **Type 2 error**

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Area Under Curve

Area Under Curve(AUC) is one of the most widely used metrics for evaluation. It is used for binary classification problem. *AUC* of a classifier is equal to the probability that the classifier will rank a randomly chosen positive example higher than a randomly chosen negative example.

True Positive Rate (Sensitivity)

$$TruePositiveRate = \frac{TruePositive}{FalseNegative + TruePositive}$$

True Negative Rate (Specificity)

$$TrueNegativeRate = \frac{TrueNegative}{TrueNegative + FalsePositive}$$

False Positive Rate :

$$FalsePositiveRate = \frac{FalsePositive}{TrueNegative + FalsePositive}$$

AUC has a range of [0, 1]. The greater the value, the better is the performance of our model.

Precision vs. Recall

Precision tells us how many of the correctly predicted cases actually turned out to be positive.

Precision Calculation

$$Precision = \frac{TP}{TP + FP}$$

This would determine whether our model is reliable or not. Precision is important in music or video recommendation systems, e-commerce websites, etc. Wrong results could lead to customer churn and be harmful to the business.

Recall tells us how many of the actual positive cases were able to predict correctly with our model.

Recall calculation

$$Recall = \frac{TP}{TP + FN}$$

Recall is important in medical cases where it doesn't matter whether we raise a false alarm but the actual positive cases should not go undetected!

F1-Score

In practice, when we try to increase the precision of our model, the recall goes down, and vice-versa. The F1-score captures both the trends in a single value:

$$F1 - score = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$$

F1-score is a harmonic mean of Precision and Recall, and so it gives a combined idea about these two metrics. It is maximum when Precision is equal to Recall.

Log loss

Log loss is often used as the objective function that is optimized under the hood of machine learning models. Yet, it can also be used as a performance metric.

After Run experiment, an info graphic shows the creation of pipelines for the data.

v. View the results

Table 1: The eight medical predictor features used in the model are:

S.No.	Attribute
1	**Pregnancies:** Number of times Pregnant
2	**Glucose:** Plasma glucose concentration a 2 hours in an oral glucose tolerance
3	**Blood Pressure:** Diastolic blood pressure (mm Hg)
4	**Skin Thickness:** Triceps skin fold thickness (mm)
5	**Insulin:** 2-Hour serum insulin (µU/ml)
6	**BMI:** Body Mass Index (weight in kg/(height in m)^2)
7	**DiabetesPedigreeFunction** : Diabetes pedigree function on genetic influence and hereditary risk
8	**Age** (years)

Step 3: Deployment and analysis on real life scenario.

Node red is a Open Source flow based tool and IOT platform and Dashboard developed by IBM and written in Node.js.

Node-red lets easily applications by joining together black box functions (**nodes**) using a web interface and requires very little, if any, programming knowledge. Node-red nodes pass the **msg object** between nodes.

Node-Red provides three mechanisms:

- The **context object** -stores data for a node
- The **Flow object** – stores data for a flow
- The **global object** -stores data for the canvas

If the flow needs to store variables then use a **JSON data file**.

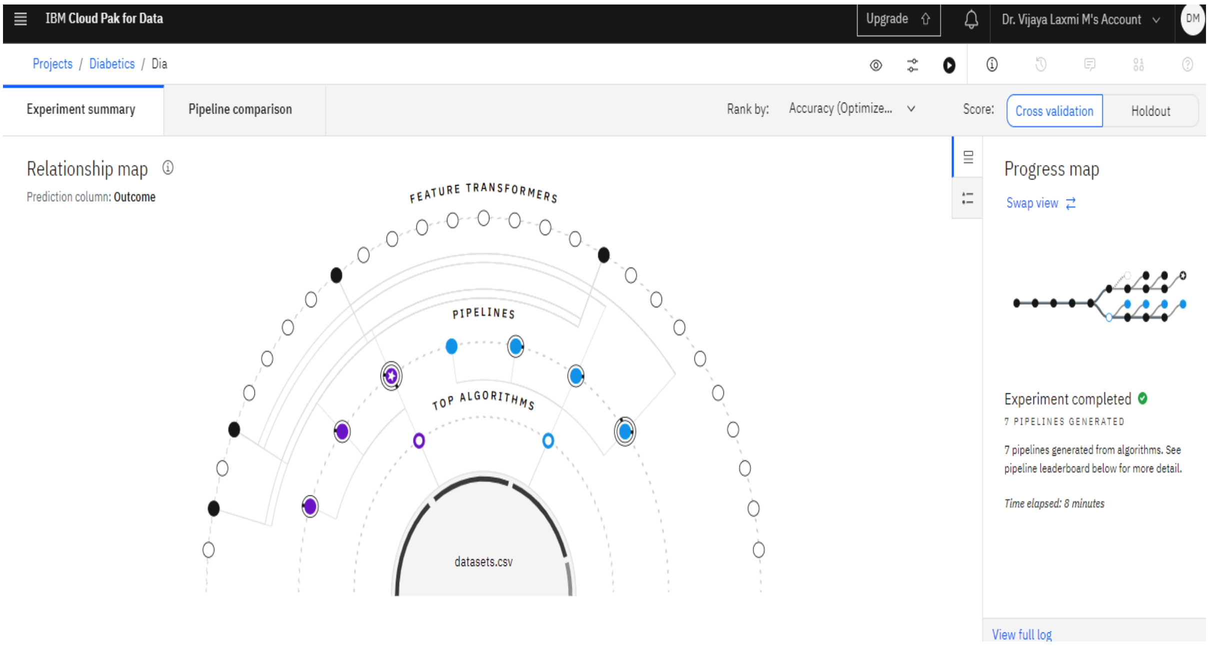
CHAPTER III

Results:

1. Experimental summary using Auto AI

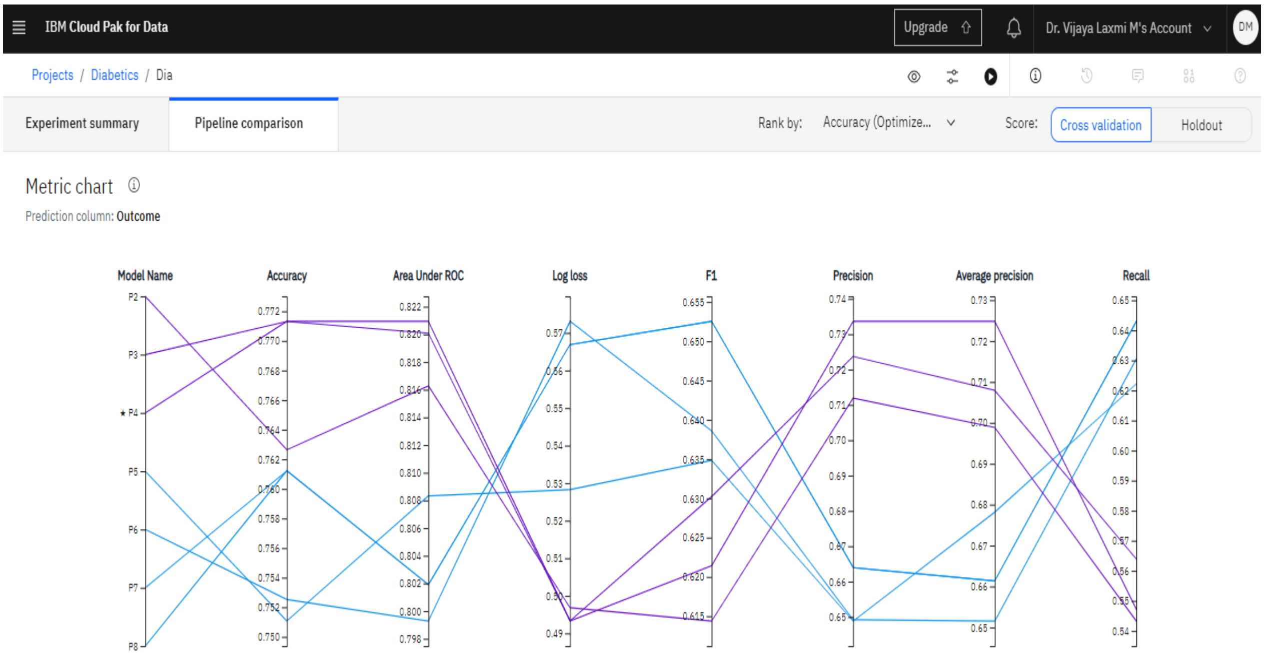
https://dataplatform.cloud.ibm.com/ml/auto-ml/4f65d9f0-eb29-453d-b6fc-3eb219a17d20/train?projectid=5d10e686-c4ab-4c14-8bc2-42f8dca9fefa&mlInstanceGuid=2465a2fd-2595-4597-813a-d44cba01dee8&context=cpdaas

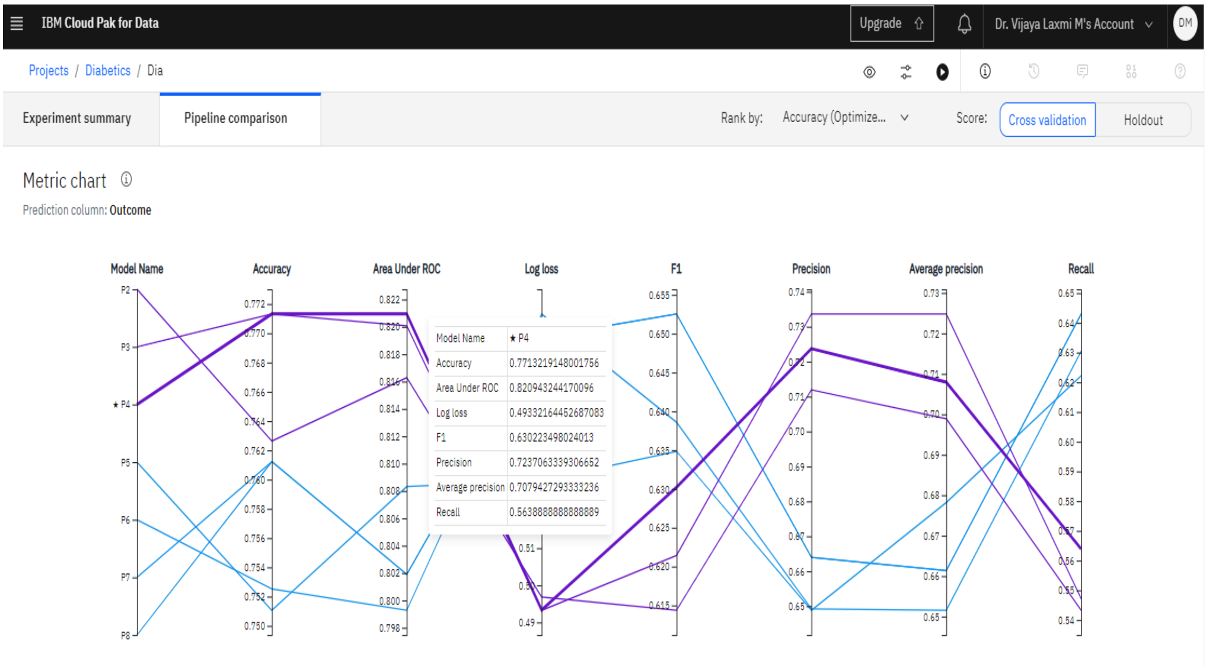
Figure 2 : Info graphic shows the creation of pipelines for the data



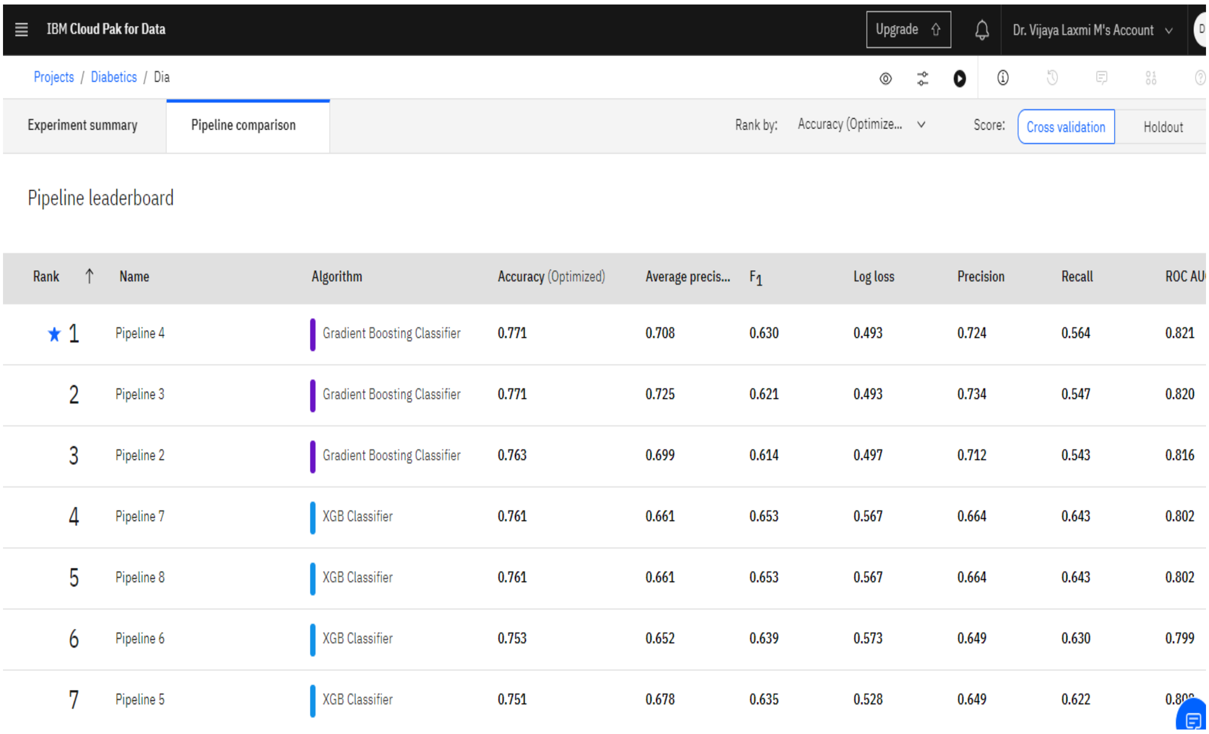
7 pipelines generated from algorithms. See pipeline leaderboard below for more detail.

Time elapsed: 8 minutes





Model Name	★ P4
Accuracy	0.7713219148001756
Area Under ROC	0.820943244170096
Log loss	0.49332164452687083
F1	0.630223498024013
Precision	0.7237063339306652
Average precision	0.7079427293333236
Recall	0.5638888888888889

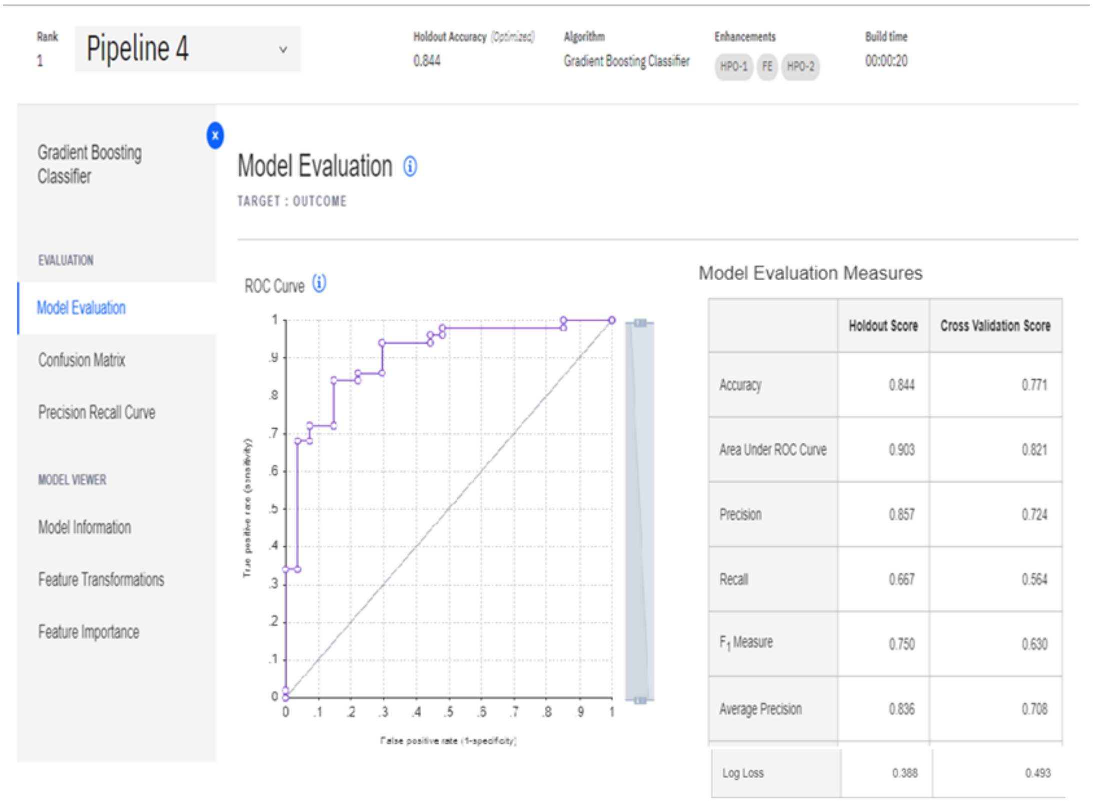


Pipeline leaderboard

Rank ↑	Name	Algorithm	Accuracy (Optimized)	Average precis...	F_1	Log loss	Precision	Recall	ROC AU
★ 1	Pipeline 4	Gradient Boosting Classifier	0.771	0.708	0.630	0.493	0.724	0.564	0.821
2	Pipeline 3	Gradient Boosting Classifier	0.771	0.725	0.621	0.493	0.734	0.547	0.820
3	Pipeline 2	Gradient Boosting Classifier	0.763	0.699	0.614	0.497	0.712	0.543	0.816
4	Pipeline 7	XGB Classifier	0.761	0.661	0.653	0.567	0.664	0.643	0.802
5	Pipeline 8	XGB Classifier	0.761	0.661	0.653	0.567	0.664	0.643	0.802
6	Pipeline 6	XGB Classifier	0.753	0.652	0.639	0.573	0.649	0.630	0.799
7	Pipeline 5	XGB Classifier	0.751	0.678	0.635	0.528	0.649	0.622	0.80[illegible]

From the above pipeline comparison, the results of pipeline 4 are shown below:

Figure 3 : Pipeline 4 ROC curve and Model Evaluation Measures



Rank 1 Pipeline 4

Holdout Accuracy (Optimized): 0.844

Algorithm: Gradient Boosting Classifier

Enhancements: HPO-1 FE HPO-2

Build time: 00:00:20

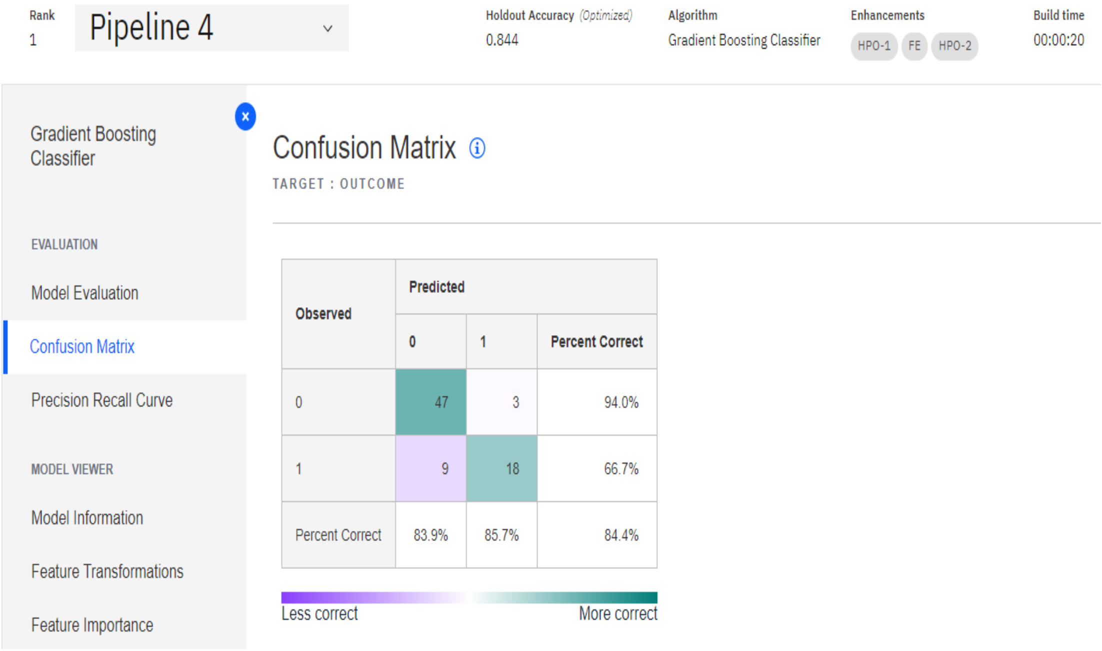
Model Evaluation

TARGET : OUTCOME

Model Evaluation Measures

	Holdout Score	Cross Validation Score
Accuracy	0.844	0.771
Area Under ROC Curve	0.903	0.821
Precision	0.857	0.724
Recall	0.667	0.564
F_1 Measure	0.750	0.630
Average Precision	0.836	0.708
Log Loss	0.388	0.493

Figure 4 : Pipeline 4 confusion matrix



Rank 1 Pipeline 4

Holdout Accuracy (Optimized): 0.844

Algorithm: Gradient Boosting Classifier

Enhancements: HPO-1 FE HPO-2

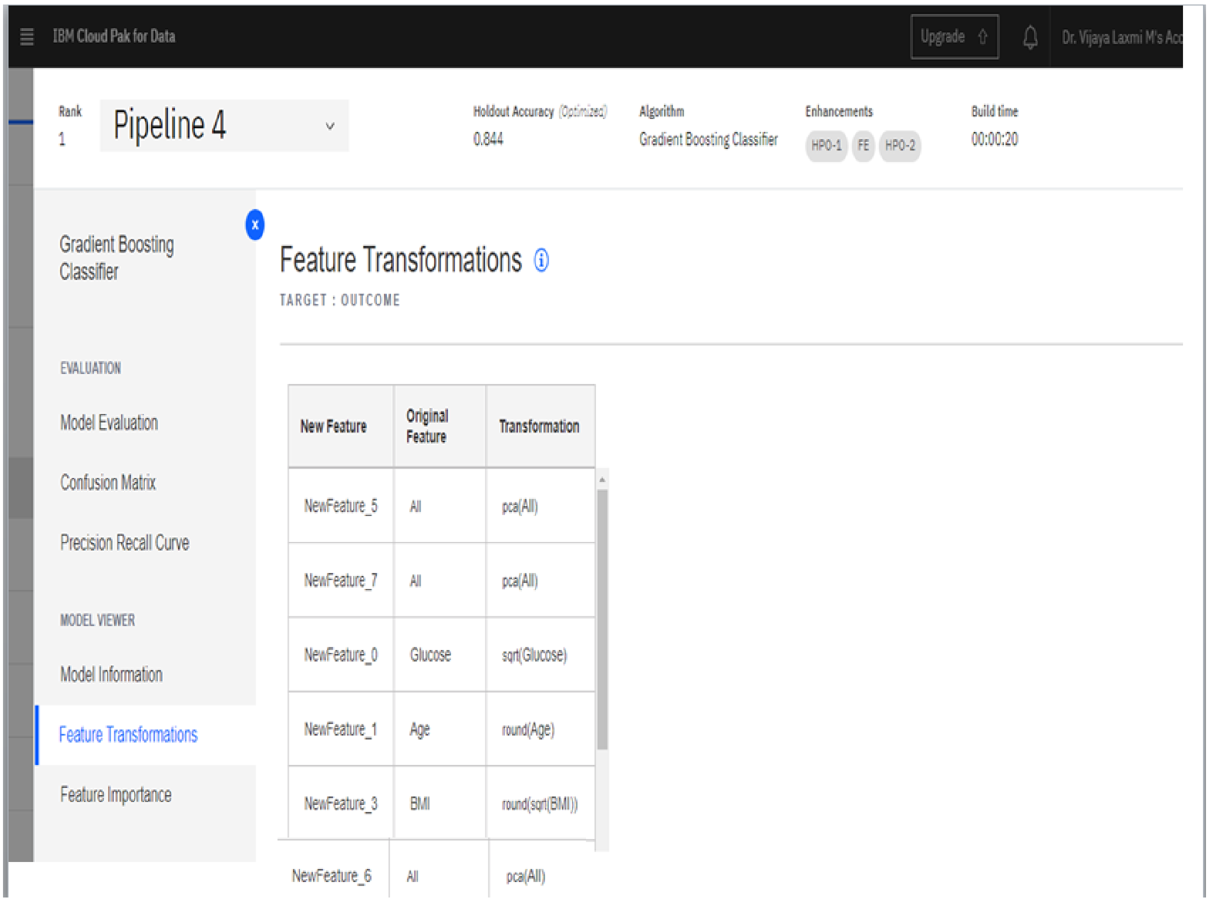
Build time: 00:00:20

Confusion Matrix

TARGET : OUTCOME

Observed	Predicted		
	0	1	Percent Correct
0	47	3	94.0%
1	9	18	66.7%
Percent Correct	83.9%	85.7%	84.4%

Figure 5 : Pipeline 4 Feature Transformations



Rank 1 Pipeline 4

Holdout Accuracy (Optimized) 0.844

Algorithm Gradient Boosting Classifier

Enhancements HPO-1 FE HPO-2

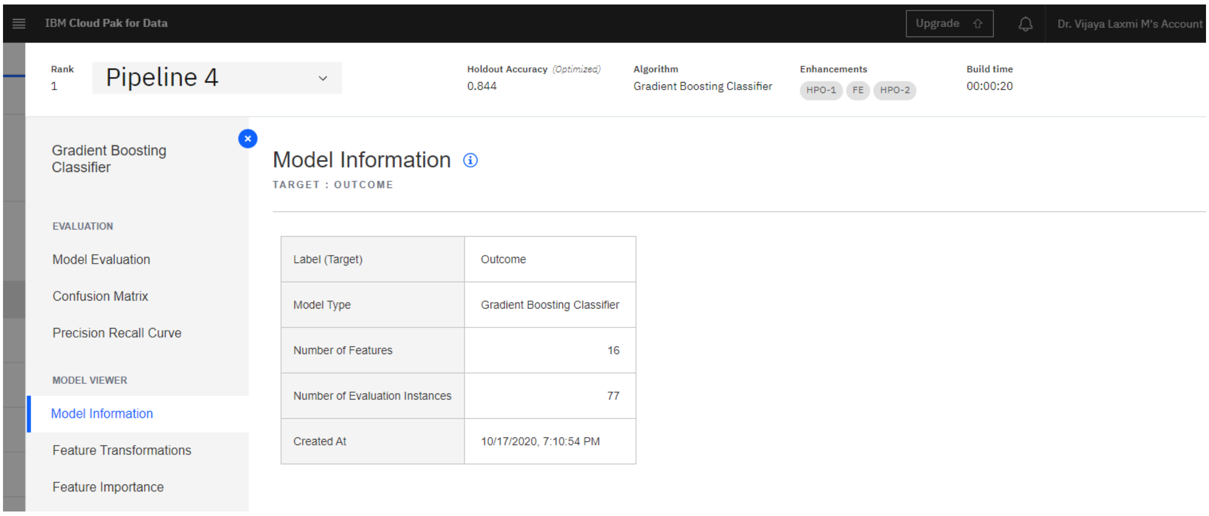
Build time 00:00:20

Feature Transformations

TARGET : OUTCOME

New Feature	Original Feature	Transformation
NewFeature_5	All	pca(All)
NewFeature_7	All	pca(All)
NewFeature_0	Glucose	sqrt(Glucose)
NewFeature_1	Age	round(Age)
NewFeature_3	BMI	round(sqrt(BMI))
NewFeature_6	All	pca(All)

Figure 6 : Pipeline 4 Model Information



Rank 1 Pipeline 4

Holdout Accuracy (Optimized) 0.844

Algorithm Gradient Boosting Classifier

Enhancements HPO-1 FE HPO-2

Build time 00:00:20

Model Information

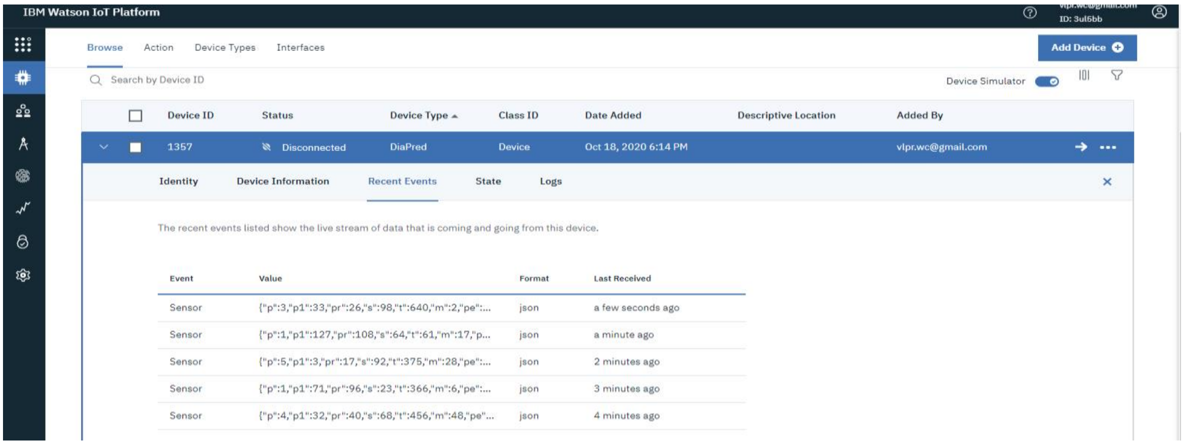
TARGET : OUTCOME

Label (Target)	Outcome
Model Type	Gradient Boosting Classifier
Number of Features	16
Number of Evaluation Instances	77
Created At	10/17/2020, 7:10:54 PM

2. IBM Watson IoT Platform is a managed, cloud-hosted service designed to make it simple to derive value from IoT devices. The device communicates by using HTTP (Hyper Text Transfer Protocol) or MQTT (Message Queuing Telemetry Transport) protocols. The device messages must conform to the Watson IoT Platform message payload requirements.

   Link: https://3ul5bb.internetofthings.ibmcloud.com/dashboard/devices/browse

Figure 7: IBM in-built simulator to generate sensor value

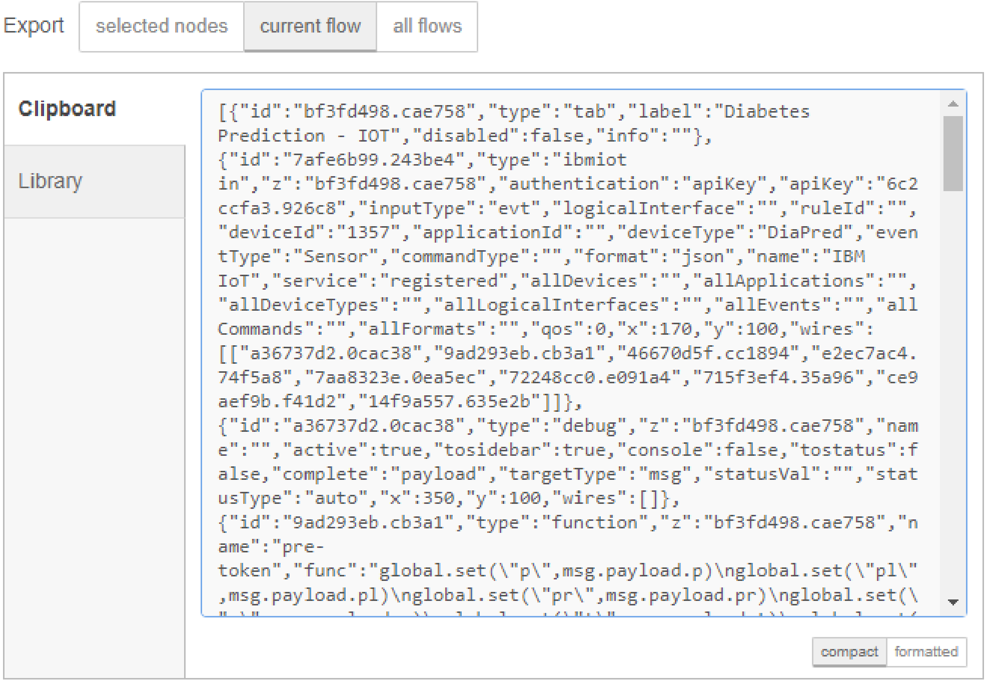


□	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By
□	1357	Disconnected	DiaPred	Device	Oct 18, 2020 6:14 PM		vlpr.wc@gmail.com

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
Sensor	{"p":3,"p1":33,"pr":26,"s":98,"t":640,"m":2,"pe":...	json	a few seconds ago
Sensor	{"p":1,"p1":127,"pr":108,"s":64,"t":61,"m":17,"p...	json	a minute ago
Sensor	{"p":5,"p1":3,"pr":17,"s":92,"t":375,"m":28,"pe":...	json	2 minutes ago
Sensor	{"p":1,"p1":71,"pr":96,"s":23,"t":366,"m":6,"pe":...	json	3 minutes ago
Sensor	{"p":4,"p1":32,"pr":40,"s":68,"t":456,"m":48,"pe"...	json	4 minutes ago

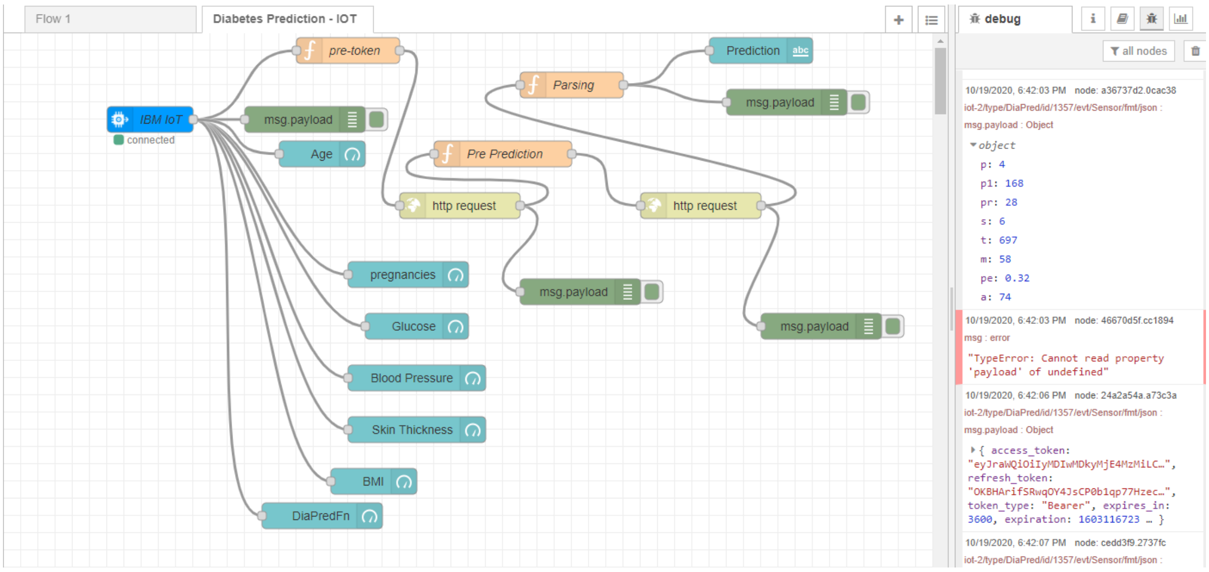
3. The flow created is represented by the json.



```
[{"id":"bf3fd498.cae758","type":"tab","label":"Diabetes
Prediction - IOT","disabled":false,"info":""},
{"id":"7afe6b99.243be4","type":"ibmiot
in","z":"bf3fd498.cae758","authentication":"apiKey","apiKey":"6c2
ccfa3.926c8","inputType":"evt","logicalInterface":"","ruleId":"",
"deviceId":"1357","applicationId":"","deviceType":"DiaPred","even
tType":"Sensor","commandType":"","format":"json","name":"IBM
IoT","service":"registered","allDevices":"","allApplications":"",
"allDeviceTypes":"","allLogicalInterfaces":"","allEvents":"","all
Commands":"","allFormats":"","qos":0,"x":170,"y":100,"wires":
[["a36737d2.0cac38","9ad293eb.cb3a1","46670d5f.cc1894","e2ec7ac4.
74f5a8","7aa8323e.0ea5ec","72248cc0.e091a4","715f3ef4.35a96","ce9
aef9b.f41d2","14f9a557.635e2b"]]},
{"id":"a36737d2.0cac38","type":"debug","z":"bf3fd498.cae758","nam
e":"","active":true,"tosidebar":true,"console":false,"tostatus":f
alse,"complete":"payload","targetType":"msg","statusVal":"","stat
usType":"auto","x":350,"y":100,"wires":[]},
{"id":"9ad293eb.cb3a1","type":"function","z":"bf3fd498.cae758","n
ame":"pre-
token","func":"global.set(\"p\",msg.payload.p)\nglobal.set(\"pl\"
,msg.payload.pl)\nglobal.set(\"pr\",msg.payload.pr)\nglobal.set(\
```

Link: https://node-red-gnpdc-2020-10-18.eu-gb.mybluemix.net/red/#flow/bf3fd498.cae758

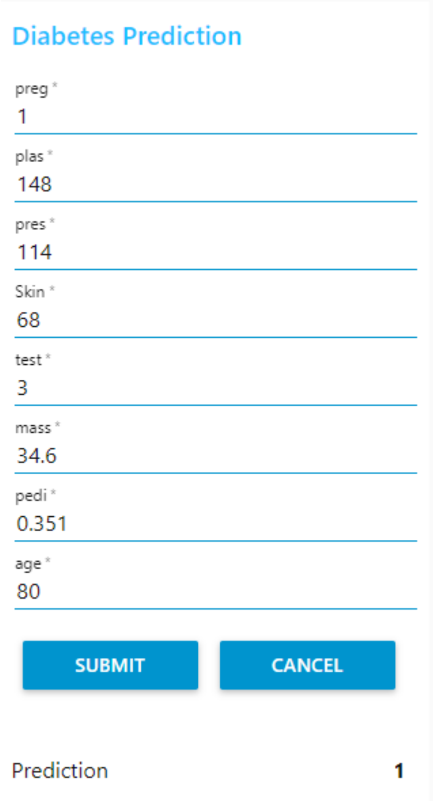
Figure 8: Diabetic Prediction Flow diagram using Node-RED



```
10/19/2020, 6:42:03 PM   node: a36737d2.0cac38
iot-2/type/DiaPred/id/1357/evt/Sensor/fmt/json :
msg.payload : Object
▾object
  p: 4
  p1: 168
  pr: 28
  s: 6
  t: 697
  m: 58
  pe: 0.32
  a: 74
10/19/2020, 6:42:03 PM   node: 46670d5f.cc1894
msg : error
"TypeError: Cannot read property
'payload' of undefined"
10/19/2020, 6:42:06 PM   node: 24a2a54a.a73c3a
iot-2/type/DiaPred/id/1357/evt/Sensor/fmt/json :
msg.payload : Object
▸{ access_token:
"eyJraWQiOiIyMDIwMDkyMjE4MzMiLC…",
refresh_token:
"OKBHArifSRwqOY4JsCP0b1qp77Hzec…",
token_type: "Bearer", expires_in:
3600, expiration: 1603116723 … }
10/19/2020, 6:42:07 PM   node: cedd3f9.2737fc
iot-2/type/DiaPred/id/1357/evt/Sensor/fmt/json :
```

4. Link: https://3ul5bb.internetofthings.ibmcloud.com/dashboard/devices/browse

a)


Diabetes Prediction

preg *
1

plas *
148

pres *
114

Skin *
68

test *
3

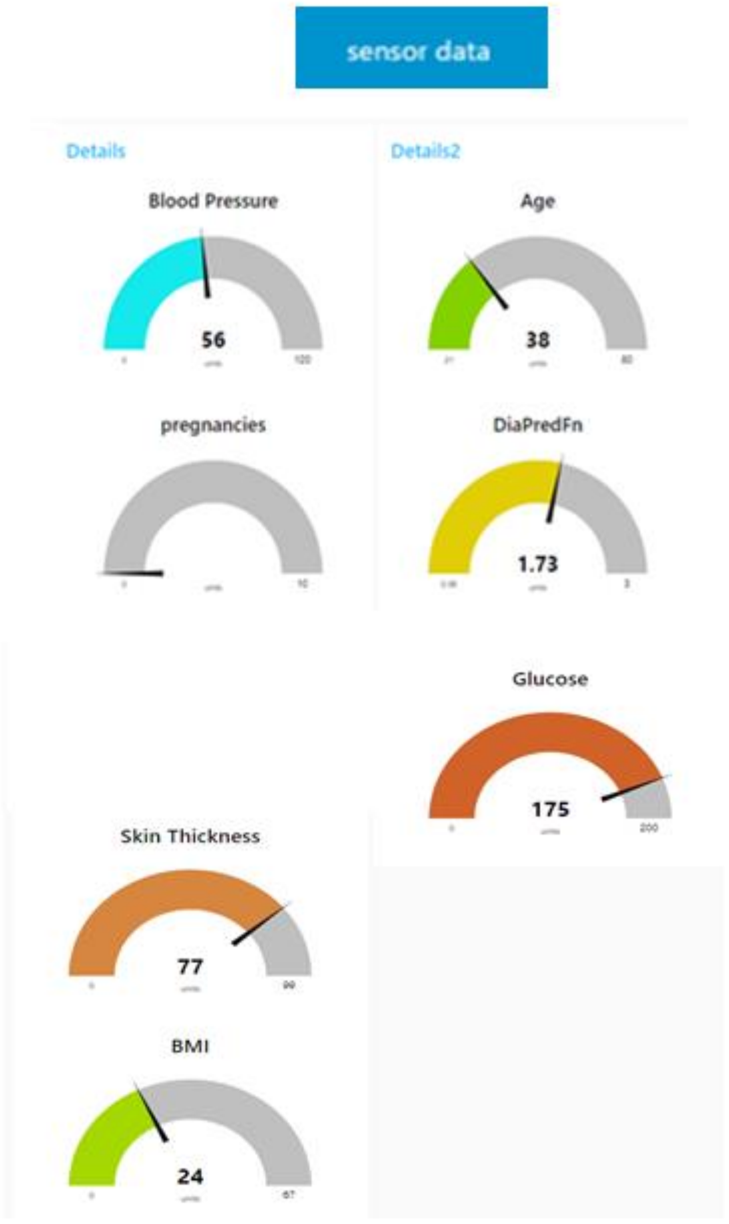
mass *
34.6

pedi *
0.351

age *
80

Prediction 1

b) Figure 9: Display of Sensor data on dashboard



CONCLUSION:

In this project, an end-to-end web application that predicts the probability of females having diabetes is developed. The application framework with the machine learning model trained and deployed on IBM Watson Studio.

REFERENCES

1. “GuruCool” training videos
2. IBM Machine Learning tutorials and videos