Performance Testing

Team ID	PNT2022TMID47935
Project Name	Real-time river water quality monitoring and control system

Model Performance

Classification Accuracy

Classification accuracy is the number of correct predictions made as a ratio of all predictions made.

This is the most common evaluation metric for classification problems, it is also the most misused. It is really only suitable when there are an equal number of observations in each class (which is rarely the case) and that all predictions and prediction errors are equally important, which is often not the case. `#include <WiFi.h>`

```
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "ks8pti"
#define DEVICE_TYPE "ESP32"
#define DEVICE_ID "143143"
#define TOKEN "123456789"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char topic[] = "iot-2/cmd/command/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);
void publishData();


const int trigpin=5;
const int echopin=18;
String command;
String data="";

long duration;
float dist;
```

Note: given the stochastic nature of the algorithm or evaluation procedure, or differences in numerical precision. Consider running the example a few times and compare the average outcome.
This can be converted into a percentage by multiplying the value by 100, giving an accuracy score of approximately 77% accurate.

```
1Accuracy: 0.770 (0.048)
```


2. Log Loss

[Logistic loss](#) (or log loss) is a performance metric for evaluating the predictions of probabilities of membership to a given class.
The scalar probability between 0 and 1 can be seen as a measure of confidence for a prediction by an algorithm. Predictions that are correct or incorrect are rewarded or punished proportionally to the confidence of the prediction.

Below is an example of calculating log loss for Logistic regression predictions on the Pima Indians onset of diabetes dataset.

```
void setup()
{
  Serial.begin(115200);
  pinMode(led, OUTPUT);
  pinMode(trigpin, OUTPUT);
  pinMode(echopin, INPUT);
  wifiConnect();
  mqttConnect();
}

void loop() {
  bool isNearby = dist < 100;
  digitalWrite(led, isNearby);

  publishData();
  delay(500);

  if (!client.loop()) {
    mqttConnect();
  }
}

void wifiConnect() {
  Serial.print("Connecting to
"); Serial.print("Wifi");
  WiFi.begin("Wokwi-GUEST", "",
6);
```

```
  while (WiFi.status() !=
WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.print("WiFi connected,
IP address: ");
Serial.println(WiFi.localIP());
}

void mqttConnect() {
  if (!client.connected()) {
    Serial.print("Reconnecting
MQTT client to ");
Serial.println(server);
    while
(!client.connect(clientId,
authMethod, token)) {
      Serial.print(".");
      delay(500);
    }
    initManagedDevice();
    Serial.println();
  }
}
```

Smaller log loss is better with 0 representing a perfect log loss.

Note: given the stochastic nature of the algorithm or evaluation procedure, or differences in numerical precision. Consider running the example a few times and compare the average outcome.

As mentioned above, the measure is inverted to be ascending when using

3. Area Under ROC Curve

Area Under ROC Curve (or ROC AUC for short) is a performance metric for binary classification problems.

The AUC represents a model's ability to discriminate between positive and negative classes. An area of 1.0 represents a model that made all predictions perfectly. An area of 0.5 represents a model as good as random.

A ROC Curve is a plot of the true positive rate and the false positive rate for a given set of probability predictions at different thresholds used to map the probabilities to class labels. The area under the curve is then the approximate integral under the ROC Curve.

The example below provides a demonstration of calculating AUC.

```
void initManagedDevice() {
  if (client.subscribe(topic)) {
    //
Serial.println(client.subscribe(topic));
    Serial.println("IBM subscribe to cmd
OK");
  } else {
    Serial.println("subscribe to cmd
FAILED");
  }
}
void publishData()
{
  digitalWrite(trigpin,LOW);
  digitalWrite(trigpin,HIGH);
  delayMicroseconds(10);
  digitalWrite(trigpin,LOW);
  duration=pulseIn(echopin,HIGH);
  dist=duration*speed/2;
  if(dist<100){
    String payload = "{\"Alert Distance
is\":";
    payload += dist;
    payload += "}";
```

Note: given the stochastic nature of the algorithm or evaluation procedure, or differences in numerical precision. Consider running the example a few times and compare the average outcome.
You can see the the AUC is relatively close to 1 and greater than 0.5, suggesting some skill in the predictions.

```
AUC: 0.824 (0.041)
```


4. Confusion Matrix

The confusion matrix is a handy presentation of the accuracy of a model with two or more classes.
The table presents predictions on the x-axis and accuracy outcomes on the y-axis. The cells of the table are the number of predictions made by a machine learning algorithm.

For example, a machine learning algorithm can predict 0 or 1 and each prediction may actually have been a 0 or 1. Predictions for 0 that were actually 0 appear in the cell for prediction=0 and actual=0, whereas predictions for 0 that were actually 1 appear in the cell for prediction = 0 and actual=1. And so on.

Below is an example of calculating a confusion matrix for a set of prediction by a model

```
    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
      if(client.publish(publishTopic, (char*) payload.c_str())) {
      Serial.println("Warning crosses 110cm -- it automaticaly of the loop");
      digitalWrite(led,HIGH);
    }

  }
    if(dist>101 && dist<111){
    String payload = "{\"Normal Distance\":";
    payload += dist;
    payload += "}";

    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);

    }

  }
```

on a test set.

Note: Your results may vary given the stochastic nature of the algorithm or evaluation procedure, or differences in numerical precision. Consider running the example a few times and compare the average outcome.
Although the array is printed without headings, you can see that the majority of the predictions fall on the diagonal line of the matrix (which are correct predictions).

```
[[141  21]
 [ 41  51]]
```


5. Classification Report

Scikit-learn does provide a convenience report when working on classification problems to give you a quick idea of the accuracy of a model using a number of measures.

The *classification_report()* function displays the precision, recall, f1-score and support for each class.

The example below demonstrates the report on the binary classification problem.

```
void initManagedDevice() {
  if (client.subscribe(topic)) {
    // Serial.println(client.subscribe(topic));
    Serial.println("IBM subscribe to cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}
```

Note: given the stochastic nature of the algorithm or evaluation procedure, or differences in numerical precision. Consider running the example a few times and compare the average outcome.