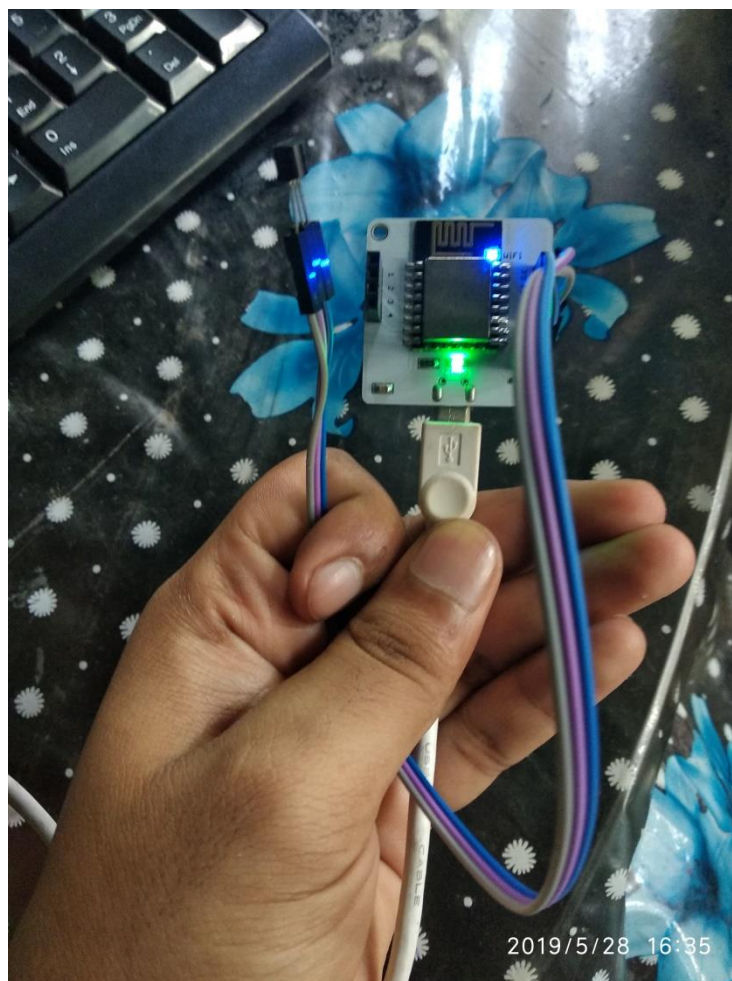


Project Objectives:

The pharmaceutical companies use a cooling chamber which is similar to a refrigerator to keep the tablets and maintain the temperature in the required limits. However, since you most probably don't have a cooling chamber which can maintain a temperature in the range, of -40 to -30 degrees Celsius, you can instead use a regular refrigerator at your home for this project.

The objectives of the Capstone project are as follows.

- a) Build the circuit for temperature monitoring system, using the Bolt and LM35 sensor.



b) Create a product on the Bolt Cloud, to monitor the data from the LM35, and link it to your Bolt.

Products

TAKE A TOUR

2spandan2

Plant_Monitor 0

IoT_LED 0

IoT_LED_using_PW... 0

IoT_Buzzer 0

LM_35_with_ML 1

+ Add Product

ID: BOLT1117472

STATUS

PRODUCT

ACTIONS

BOLT1117472

ONLINE

LM_35_with_ML

Help

c) Write the product code, required to run the polynomial regression algorithm on the data sent by the Bolt.

Products: Setup

LM_35_with_ML

Input device • Connected with GPIO

Hardware Code

Write your code in the code window below.

predict

```
1 setChartLibrary('google-chart');
2 setChartTitle('Polynomial Regression');
3 setChartType('predictionGraph');
4 setAxisName('time_stamp','temp');
5 mul(0.0977);
6 plotChart('time_stamp','temp');
7
```

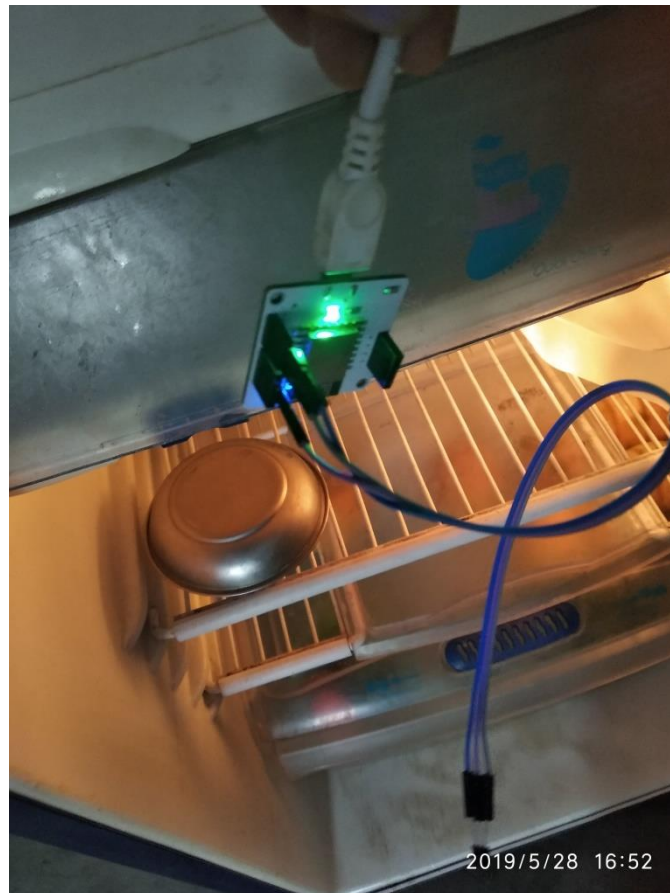
Click here to access the Documentation

Note: Variable name can only contain lowercase alphanumeric characters and underscore and should start with an alphabet.

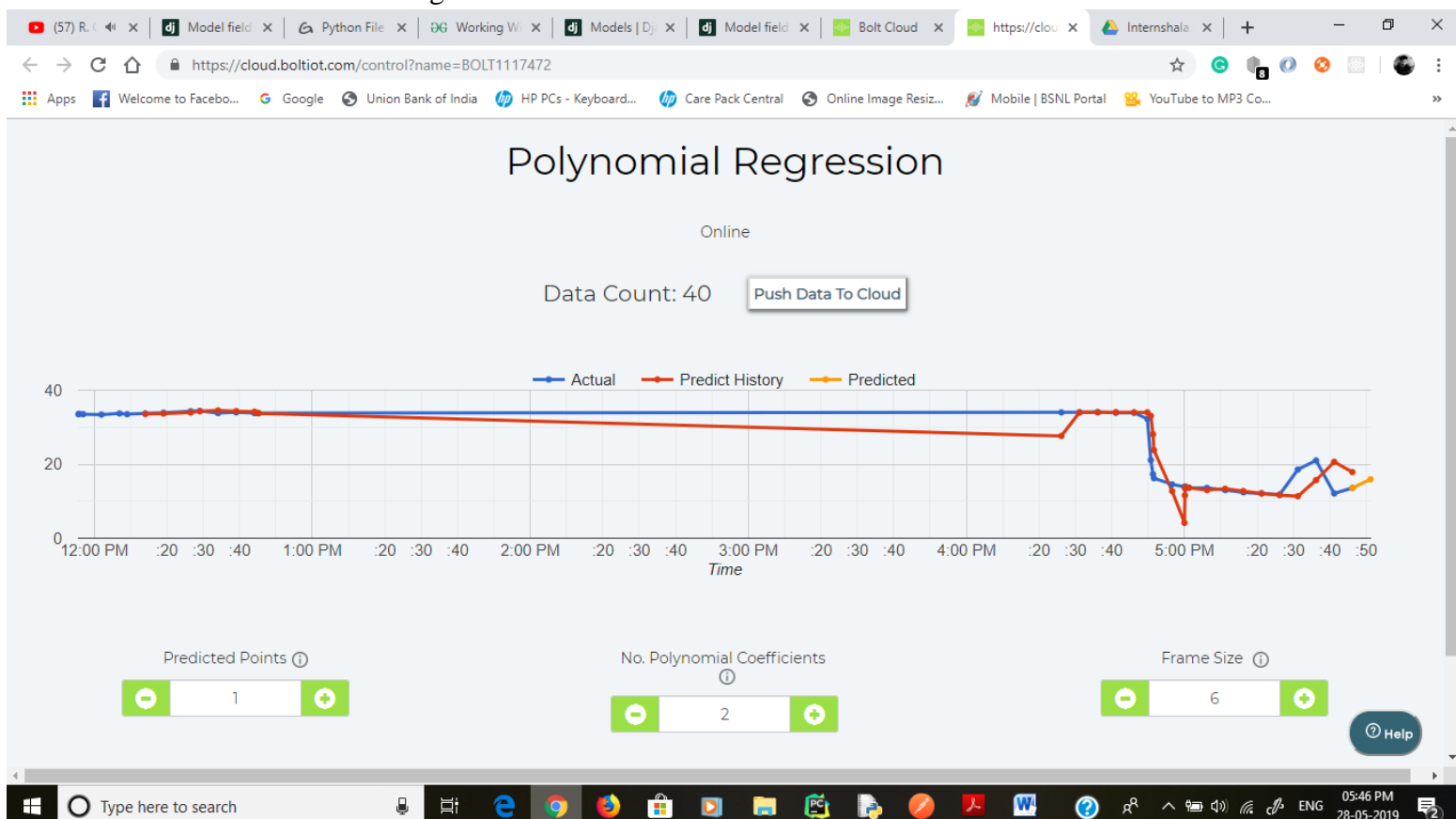
Pin	Variable Name
A0 Analog	temp

Help

- d) Keep the temperature monitoring circuit inside your fridge with the door of the fridge closed, and let the system record the temperature readings for about 2 hours.

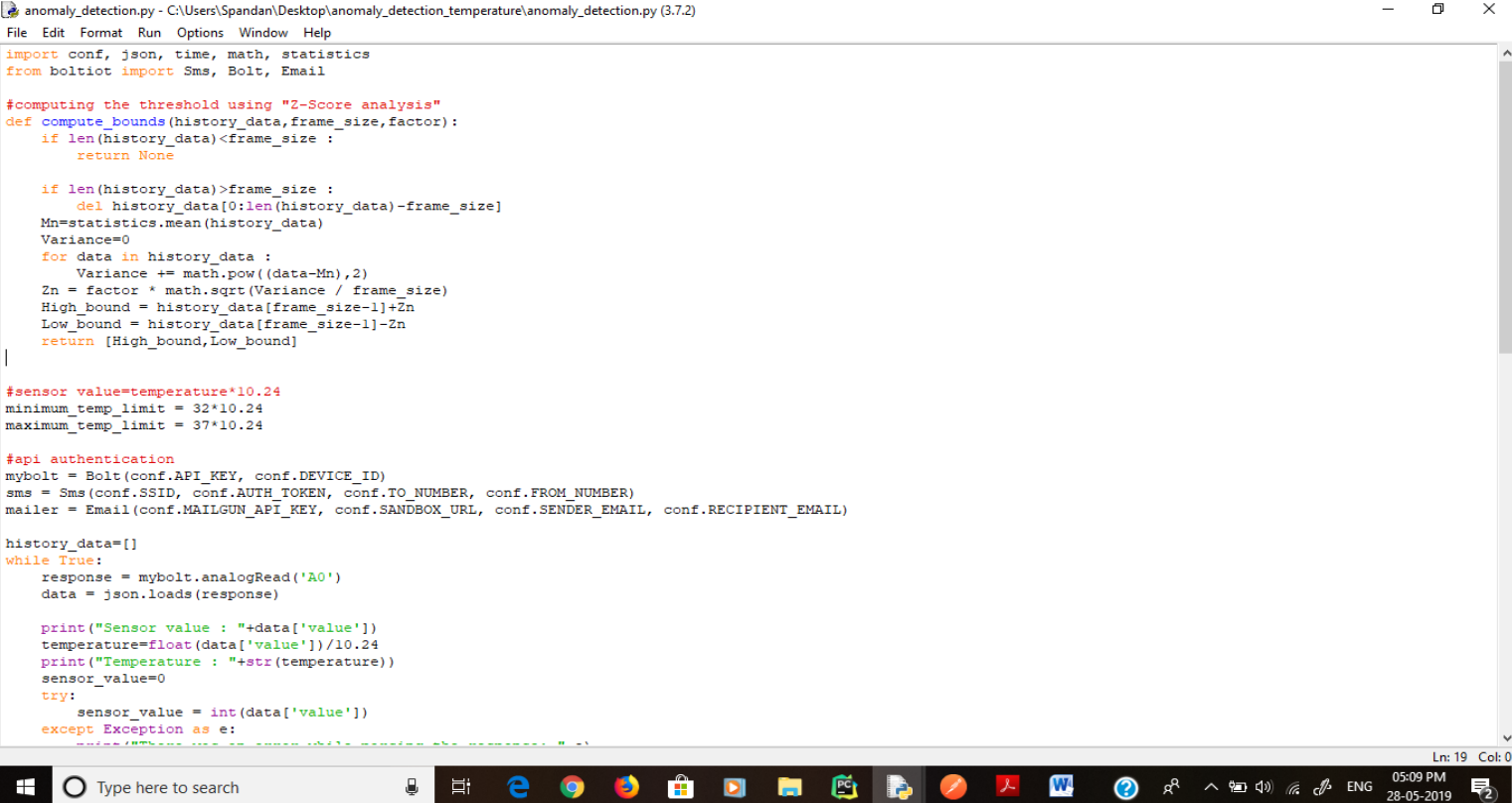


- e) Using the reading that you received in the 2 hours, set boundaries for the temperature within the fridge



- f) Write a python code which will fetch the temperature data, every 10 seconds, and send out an email alert, if the temperature goes beyond the temperature thresholds you decided on in objective 'e'.
- g) Modify the python code, to also do a Z-score analysis and print the line "Someone has opened the fridge door" when an anomaly is detected.
- h) Tune the Z-score analysis code, such that, it detects an anomaly when someone opens the door of the fridge.

Below are the screenshots of my Python Code



```
anomaly_detection.py - C:\Users\Spandan\Desktop\anomaly_detection_temperature\anomaly_detection.py (3.7.2)
File Edit Format Run Options Window Help
import conf, json, time, math, statistics
from boltiot import Sms, Bolt, Email

#computing the threshold using "Z-Score analysis"
def compute_bounds(history_data, frame_size, factor):
    if len(history_data) < frame_size:
        return None

    if len(history_data) > frame_size:
        del history_data[0:len(history_data)-frame_size]
    Mn = statistics.mean(history_data)
    Variance = 0
    for data in history_data:
        Variance += math.pow((data-Mn),2)
    Zn = factor * math.sqrt(Variance / frame_size)
    High_bound = history_data[frame_size-1]+Zn
    Low_bound = history_data[frame_size-1]-Zn
    return [High_bound, Low_bound]

#sensor value=temperature*10.24
minimum_temp_limit = 32*10.24
maximum_temp_limit = 37*10.24

#api authentication
mybolt = Bolt(conf.API_KEY, conf.DEVICE_ID)
sms = Sms(conf.SSID, conf.AUTH_TOKEN, conf.TO_NUMBER, conf.FROM_NUMBER)
mailer = Email(conf.MAILGUN_API_KEY, conf.SANDBOX_URL, conf.SENDER_EMAIL, conf.RECIPIENT_EMAIL)

history_data=[]
while True:
    response = mybolt.analogRead('A0')
    data = json.loads(response)

    print("Sensor value : "+data['value'])
    temperature=float(data['value'])/10.24
    print("Temperature : "+str(temperature))
    sensor_value=0
    try:
        sensor_value = int(data['value'])
    except Exception as e:
        pass

Ln: 19 Col: 0
```

```

except Exception as e:
    print("There was an error while parsing the response: ",e)
    continue

bound = compute_bounds(history_data,conf.FRAME_SIZE,conf.MUL_FACTOR)
if not bound:
    required_data_count=conf.FRAME_SIZE-len(history_data)
    print("Not enough data to compute Z-score. Need ",required_data_count," more data points")
    history_data.append(int(data['value']))
    time.sleep(10)
    continue

try:
    if sensor_value > bound[0]:
        print ("The temperature level increased suddenly. Sending an SMS and Email Alert.")
        response = sms.send_sms("The refrigerator has been opened")
        print("This is the response from Twilio ",response)

        print("Making request to Mailgun to send an email")
        response = mailer.send_email("Alert", "The Current temperature sensor value is " +str(sensor_value)+ " and temperature is " + str(
            temperature) + " degree Celsius.")
        response_text = json.loads(response.text)
        print("Response received from Mailgun is: " + str(response_text['message']))
    elif sensor_value < bound[1]:
        print ("The temperature level decreased suddenly. Sending an SMS and Email Alert.")
        response = sms.send_sms("The refrigerator has been closed")
        print("This is the Response from Twilio ",response)

        print("Making request to Mailgun to send an email")
        response = mailer.send_email("Alert", "The Current temperature sensor value is " +str(sensor_value)+ " and temperature is " + str(
            temperature) + " degree Celsius.")
        response_text = json.loads(response.text)
        print("Response received from Mailgun is: " + str(response_text['message']))
    elif sensor_value>maximum_temp_limit:
        print ("The temperature level crossed the maximum limit. Sending an SMS and Email Alert.")
        response = sms.send_sms("The temperature level crossed the maximum limit")
        print("This is the response from Twilio ",response)

        print("Making request to Mailgun to send an email")
        response = mailer.send_email("Alert", "The Current temperature sensor value is " +str(sensor_value)+ " and temperature is " + str(
            temperature) + " degree Celsius.")
        response_text = json.loads(response.text)
        print("Response received from Mailgun is: " + str(response_text['message']))

```

Ln: 19 Col: 0

```

        print("This is the response from Twilio ",response)

        print("Making request to Mailgun to send an email")
        response = mailer.send_email("Alert", "The Current temperature sensor value is " +str(sensor_value)+ " and temperature is " + str(
            temperature) + " degree Celsius.")
        response_text = json.loads(response.text)
        print("Response received from Mailgun is: " + str(response_text['message']))
    elif sensor_value < bound[1]:
        print ("The temperature level decreased suddenly. Sending an SMS and Email Alert.")
        response = sms.send_sms("The refrigerator has been closed")
        print("This is the Response from Twilio ",response)

        print("Making request to Mailgun to send an email")
        response = mailer.send_email("Alert", "The Current temperature sensor value is " +str(sensor_value)+ " and temperature is " + str(
            temperature) + " degree Celsius.")
        response_text = json.loads(response.text)
        print("Response received from Mailgun is: " + str(response_text['message']))
    elif sensor_value>maximum_temp_limit:
        print ("The temperature level crossed the maximum limit. Sending an SMS and Email Alert.")
        response = sms.send_sms("The temperature level crossed the maximum limit")
        print("This is the response from Twilio ",response)

        print("Making request to Mailgun to send an email")
        response = mailer.send_email("Alert", "The Current temperature sensor value is " +str(sensor_value)+ " and temperature is " + str(
            temperature) + " degree Celsius.")
        response_text = json.loads(response.text)
        print("Response received from Mailgun is: " + str(response_text['message']))
    elif sensor_value<minimum_temp_limit:
        print ("The temperature level went below the minimum limit. Sending an SMS and Email Alert.")
        response = sms.send_sms("The temperature level went below the minimum limit")
        print("This is the response from Twilio ",response)

        print("Making request to Mailgun to send an email")
        response = mailer.send_email("Alert", "The Current temperature sensor value is " +str(sensor_value)+ " and temperature is " + str(
            temperature) + " degree Celsius.")
        response_text = json.loads(response.text)
        print("Response received from Mailgun is: " + str(response_text['message']))
    history_data.append(sensor_value)
except Exception as e:
    print ("Error",e)
time.sleep(10)

```

Ln: 19 Col: 0

Below are the screenshots after running the code

```
*Python 3.7.2 Shell*
File Edit Shell Debug Options Window Help

RESTART: C:\Users\Spandan\Desktop\anomaly_detection_temperature\anomaly_detection.py
Sensor value : 142
Temperature : 13.8671875
Not enough data to compute Z-score. Need 10 more data points
Sensor value : 141
Temperature : 13.76953125
Not enough data to compute Z-score. Need 9 more data points
Sensor value : 139
Temperature : 13.57421875
Not enough data to compute Z-score. Need 8 more data points
Sensor value : 138
Temperature : 13.4765625
Not enough data to compute Z-score. Need 7 more data points
Sensor value : 138
Temperature : 13.4765625
Not enough data to compute Z-score. Need 6 more data points
Sensor value : 136
Temperature : 13.28125
Not enough data to compute Z-score. Need 5 more data points
Sensor value : 136
Temperature : 13.28125
Not enough data to compute Z-score. Need 4 more data points
Sensor value : 134
Temperature : 13.0859375
Not enough data to compute Z-score. Need 3 more data points
Sensor value : 132
Temperature : 12.890625
Not enough data to compute Z-score. Need 2 more data points
Sensor value : 131
Temperature : 12.79296875
Not enough data to compute Z-score. Need 1 more data points
Sensor value : 130
Temperature : 12.6953125
Sensor value : 130
Temperature : 12.6953125
Sensor value : 128
Temperature : 12.5
Sensor value : 137
Temperature : 13.37890625
Sensor value : 232
Temperature : 22.65625

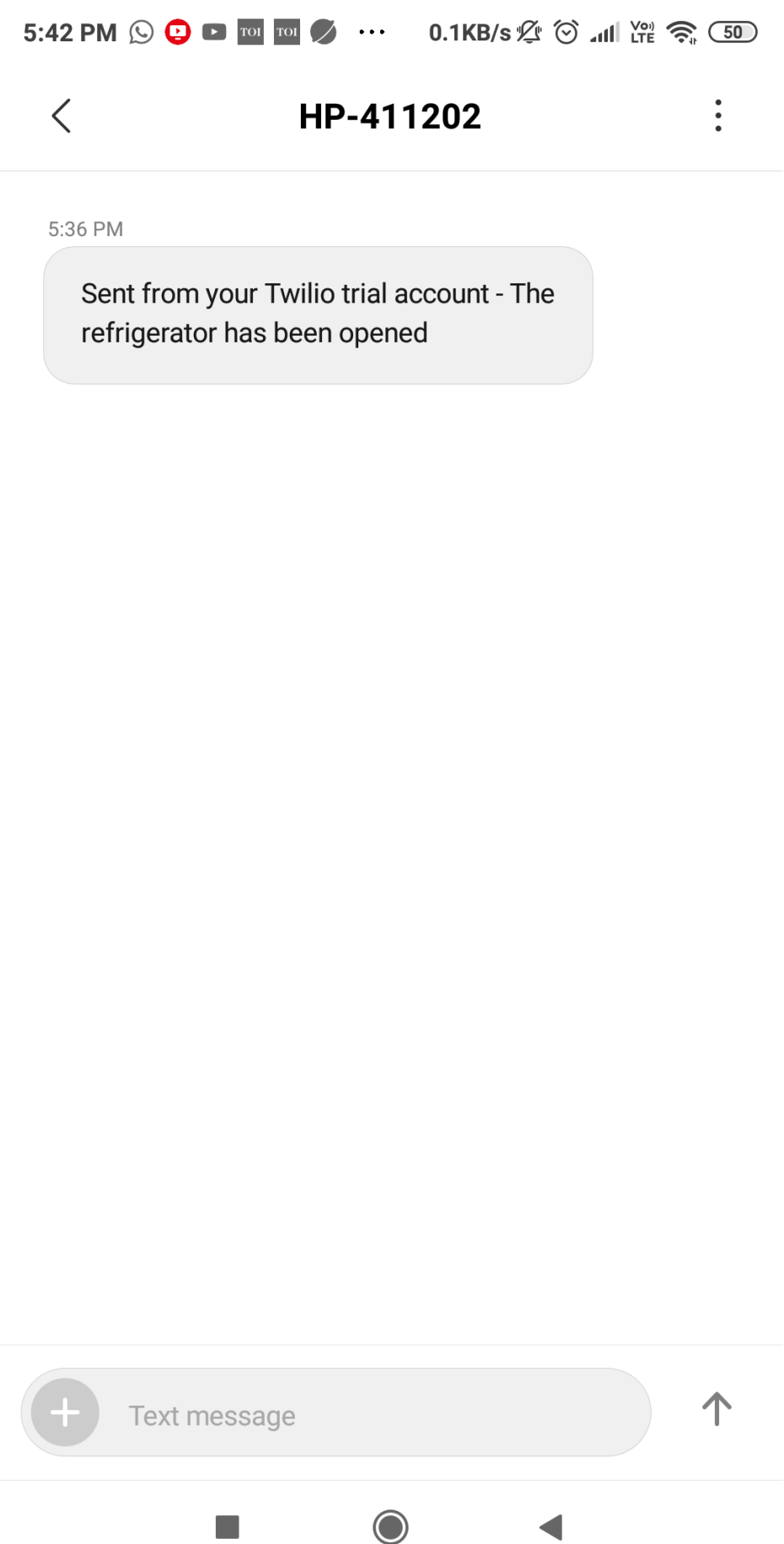
Ln: 740 Col: 0
```

```
*Python 3.7.2 Shell*
File Edit Shell Debug Options Window Help

Sensor value : 130
Temperature : 13.4765625
Not enough data to compute Z-score. Need 7 more data points
Sensor value : 138
Temperature : 13.4765625
Not enough data to compute Z-score. Need 6 more data points
Sensor value : 136
Temperature : 13.28125
Not enough data to compute Z-score. Need 5 more data points
Sensor value : 136
Temperature : 13.28125
Not enough data to compute Z-score. Need 4 more data points
Sensor value : 134
Temperature : 13.0859375
Not enough data to compute Z-score. Need 3 more data points
Sensor value : 132
Temperature : 12.890625
Not enough data to compute Z-score. Need 2 more data points
Sensor value : 131
Temperature : 12.79296875
Not enough data to compute Z-score. Need 1 more data points
Sensor value : 130
Temperature : 12.6953125
Sensor value : 130
Temperature : 12.6953125
Sensor value : 128
Temperature : 12.5
Sensor value : 137
Temperature : 13.37890625
Sensor value : 232
Temperature : 22.65625
The temperature level increased suddenly. Sending an SMS and Email Alert.
This is the response from Twilio <Twilio.Api.V2010.MessageInstance account_sid=AC2b76b5dd583676e04f169826ea471a13 sid=SMb155860bad274e49a69c39ba49530331>
Making request to Mailgun to send an email
Response received from Mailgun is: Queued. Thank you.
Sensor value : 196
Temperature : 19.140625
Sensor value : 176
Temperature : 17.1875
Sensor value : 167
Temperature : 16.30859375

Ln: 790 Col: 0
```


Below are the screenshots of the alerts that I received on my mobile





Alert Inbox



temperature_alert@sandbox48... 12:39 PM

The Current temperature sensor value is 349
and temperature is 34.08203125 degree

9



temperature_alert@sa... 5:36 PM

to me ▾



The Current temperature sensor value is 232 and
temperature is 22.65625 degree Celsius.

 Reply

 Reply all

 Forward

