Advanced Anomaly Detection with Edge Impulse® and Arduino®

José Antonio Bagur Nájera

Universidad del Valle de Guatemala

Scientific Use of Machine Learning on Low-Power Devices 2024

Agenda

In this session, we will cover:

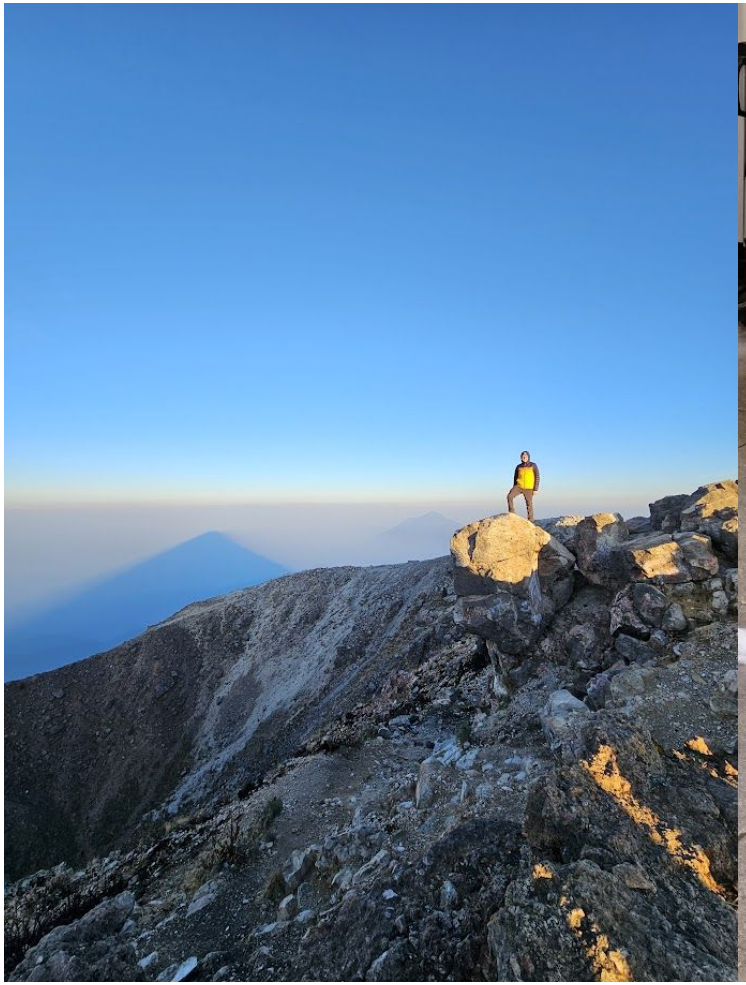
- Defining advanced anomaly detection
- Practical use cases
- Hands-on with Edge Impulse and Arduino

About me

Hello, my name is José, and I am from Guatemala. I work as a lecturer in the Mechanical Engineering Department at Universidad del Valle de Guatemala (UVG), where I am also the coordinator of the Aerospace Laboratory. Additionally, I work for the Product Experience team at Arduino as a technical content creator.

Outside of my professional life, I enjoy exploring the beautiful mountains and volcanoes of Guatemala through hiking. I am an avid coffee drinker and spend my free time with my dog, reading, and listening to music.

About me (2)



Anomaly detection

Anomaly detection is the process of **identifying patterns, events, or data points that deviate significantly from expected norms**. This is crucial for spotting critical issues in various fields like finance, cybersecurity, and healthcare.

Our primary focus in this session will be on its application in **industrial equipment monitoring**.

Anomaly detection (2)

Importance of anomaly detection in industrial equipment:

- Early detection
- Efficiency improvement
- Safety
- Predictive maintenance

Do you know more important aspects of anomaly detection for industrial equipment? Please share them in the Zoom chat 😊

Anomaly detection use cases

Use cases of anomaly detection in industrial equipment

- Vibration analysis and bearing fault detection
- Temperature monitoring and overheating detection
- Pressure monitoring in pipelines and vessels
- Energy consumption analysis
- Acoustic emissions and ultrasonic testing

Hands-on focus: vibration analysis.

Hands-on with Edge Impulse and Arduino

In this practical session, we will develop an anomaly detection model using Edge Impulse and an Arduino-supported board.

Our focus will be on creating a model to **detect anomalies in fan operation**, specifically targeting **vibration anomalies**.

Hands-on with Edge Impulse and Arduino (2)

We will detect vibrations in a fan using an accelerometer mounted on it.

For demonstration purposes, we'll use an affordable computer fan commonly available in electronics stores.

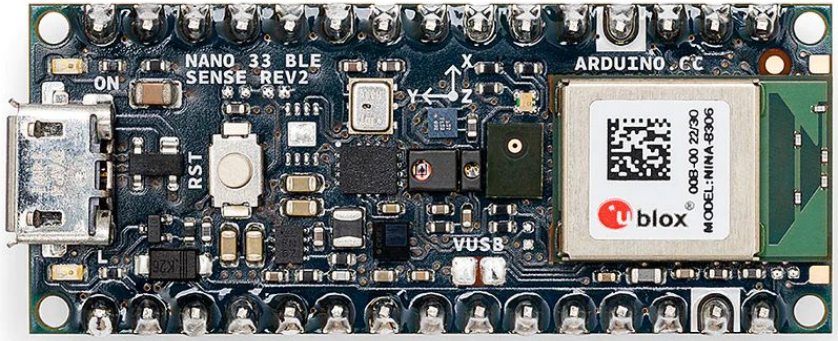
Does the placement of the accelerometer on the fan matter? Please share your thoughts in the Zoom chat 😊

Accelerometer placement in the fan

Hands-on with Edge Impulse and Arduino (3)

We will use the Arduino Nano 33 BLE Sense for our project.

This compact and versatile development board is based on Nordic Semiconductor's nRF52840 microcontroller, designed for low-power applications.



It features Bluetooth Low Energy (BLE) connectivity, making it ideal for a variety of wireless projects.

Hands-on with Edge Impulse and Arduino (4)

Key Features of the Nano 33 BLE Sense Board:

- **Microcontroller**: 32-bit ARM Cortex-M4 processor (nRF52840) running at 64 MHz, with 1 MB of Flash and 256 kB of SRAM.
- **Connectivity**: BLE for easy communication with other BLE-enabled devices.
- **Onboard Sensors**:
  - 9-axis Inertial Measurement Unit (IMU).
  - Temperature, humidity, and barometric pressure sensors.
  - Digital microphone.
  - Gesture, proximity, and ambient light sensors.
  - Color sensor.

Hands-on with Edge Impulse and Arduino (5)

Important note: There are two versions of the Nano 33 BLE Sense board available, **Rev1** and **Rev2**. Key differences between the two versions include:

- Different IMU's.
- Different temperature and humidity sensors.
- Different microphones.

While the Edge Impulse firmware works with both versions, you need to **be careful about choosing the correct version when working with the Arduino IDE**, as the libraries differ for each board.

Hands-on with Edge Impulse and Arduino (6)

Follow these instructions to configure the Nano 33 BLE Sense board with Edge Impulse. **Edge Impulse configuration can be divided into four main steps**:

1. Edge Impulse CLI installation.
2. Edge Impulse Studio account registration.
3. Arduino CLI installation.
4. Latest Edge Impulse firmware upload to the board.

Installation process can be tricky, don't hesitate to send your questions to the **network Discord server**!

Hands-on with Edge Impulse and Arduino (7)

A general **workflow** we can follow for developing an anomaly detection model is the following:

1. **Data collection**: aim for at least several hundreds of samples per class to start.
2. **Data preprocessing**: ensure the data is suitable for training a machine learning model.
3. **Model training**: split your data into training, test and validation sets. **A common split ratio is 70% for training and 30% for testing.**
4. **Model evaluation**: understand how well the model is performing.
5. **Model deployment**: deploy your model to your target hardware.

Hands-on with Edge Impulse and Arduino (8)

Comments about data collection:

For the hands-on, we will gather **two different types of samples** or **classes**: **"idle"** and **"nominal"**. Since this is a simple binary classification problem, the required sample size may be smaller compared to complex scenarios.

Other classes like **"broken"** or **"obstructed"**, for example, are not necessary but can be added to improve the model significantly.

Hands-on with Edge Impulse and Arduino (9)

Comments about data collection (2):

In this example, we are going to collect **6 minutes of data** ("not great, not terrible"). In this case, we need to allocate approximately the following amounts of data for each set:

- **Training**: 4 minutes (~70% of 6 minutes).
- **Testing**: 2 minutes (~30% of 6 minutes).

Hands-on with Edge Impulse and Arduino (10)

Important note: The **number of samples required for an anomaly detection model depends on several factors**. **There is no one-size-fits-all answer**, but some general guidelines to determine the appropriate sample size are the following:

- **Data complexity**: with complex patterns, larger sample size.
- **Type of anomaly detection technique**: supervised methods require more samples than unsupervised methods.
- **Model performance**: more data is not always better.
- **Rarity of anomalies**: a larger sample size is needed if anomalies are rare events.

As a starting point, **consider collecting at least several hundred to a few thousand samples** to train an anomaly detection model.

Hands-on with Edge Impulse and Arduino (11)

Comments about model training:

In this example we are going to use an **Spectral Analysis** processing block, which is great for analyzing repetitive motion, focusing only in the accelerometer readings.

Edge Impulse can also combine different sensor readings or outputs to get a new outcome of collected data.

Hands-on with Edge Impulse and Arduino (12)

Comments about model training (2):

For this example we are going to use **two learning blocks**: a **Classification** block which learn patterns from data and an **Anomaly Detection** (K-means) block which can recognize unknown states.

For the Anomaly Detection block is important to calculate the feature importance since this can tell us which aspects are contributing to the original classes (so we can focus on those features).

Hands-on with Edge Impulse and Arduino (13)

Comments about preprocessing data:

We can achieve a more accurate model in less time by **standardizing** (i.e. to a mean of 0 and unit variance) and decorrelating input data. Great paper about this here.

The standardization process assumes our data is normally distributed (i.e. Gaussian), if your data does not follow a Gaussian distribution, we should perform a normalization.

Take a look into this Google Colab notebook from Shawn Hymel and Edge Impuse.

Hands-on with Edge Impulse and Arduino (14)

The complete anomaly detection demonstration can be found here (it is **public**, you can clone it to you Edge Impulse account and use it freely).

Conclusions

Anomaly detection is a powerful technique for identifying abnormal behavior in industrial equipment, helping to prevent costly failures, improve efficiency and enhance safety in the workplace.

The combination of Edge Impulse and Arduino provides a practical and accessible platform for implementing and deploying real-world TinyML applications. These tools enable rapid prototyping, streamlined data collection, and seamless integration with various supported boards, sensors and actuators.

Thank you for your attention

Let’s keep in touch!

- **#icpt-workshop** at the network Discord server
- Email: **jabagur@uvg.edu.gt**