life.augmented

Machine Learning Core in ST MEMS sensor

ST Korea

지준영 차장 (Jerry JI)

Agenda

1. Artificial Intelligence introduction
2. AI on the edge
3. Machine Learning Core in ST sensor
4. [Demo] Application state estimation with MLC

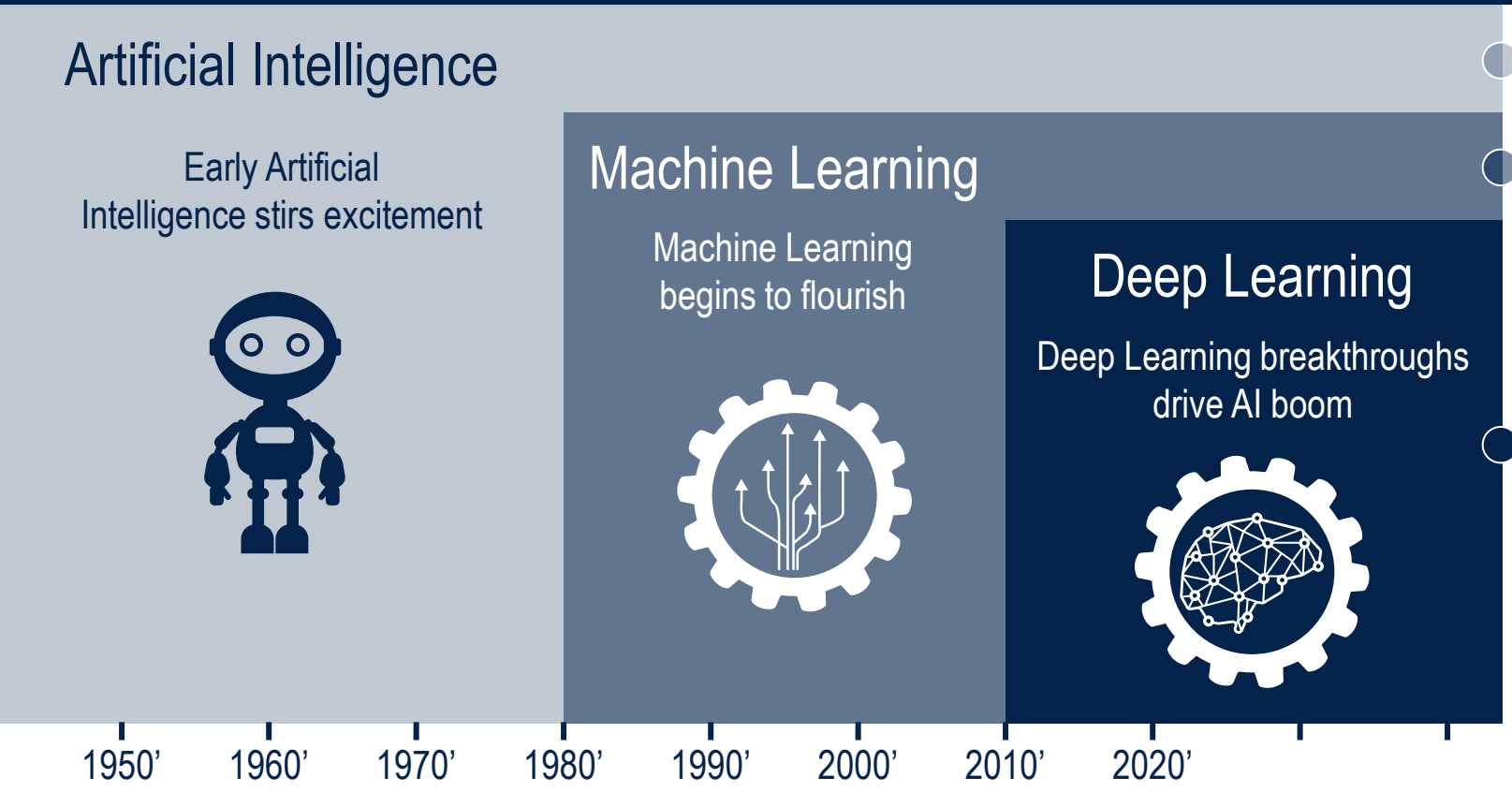
life.augmented

Artificial Intelligence introduction

life.augmented

What is AI?

The evolution of AI



Any technique that enables computer to mimic **human behavior**

Subset of AI. Algorithms and methodologies that improve over time through **learning from data**

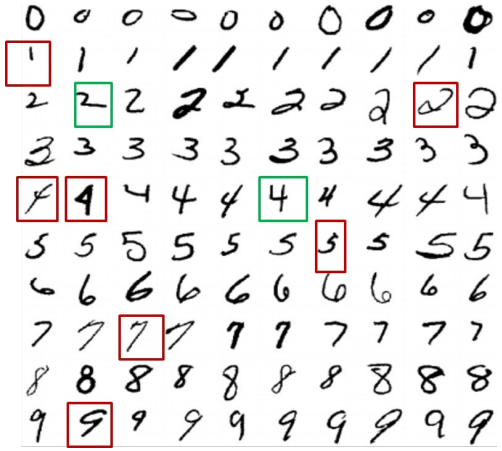
Subset of ML. Learning algorithms that derive meaning out of data, by using a hierarchy of multiple layers **that mimic the neural networks of the human brain**

life.augmented

Machine Learning, why do we need it?

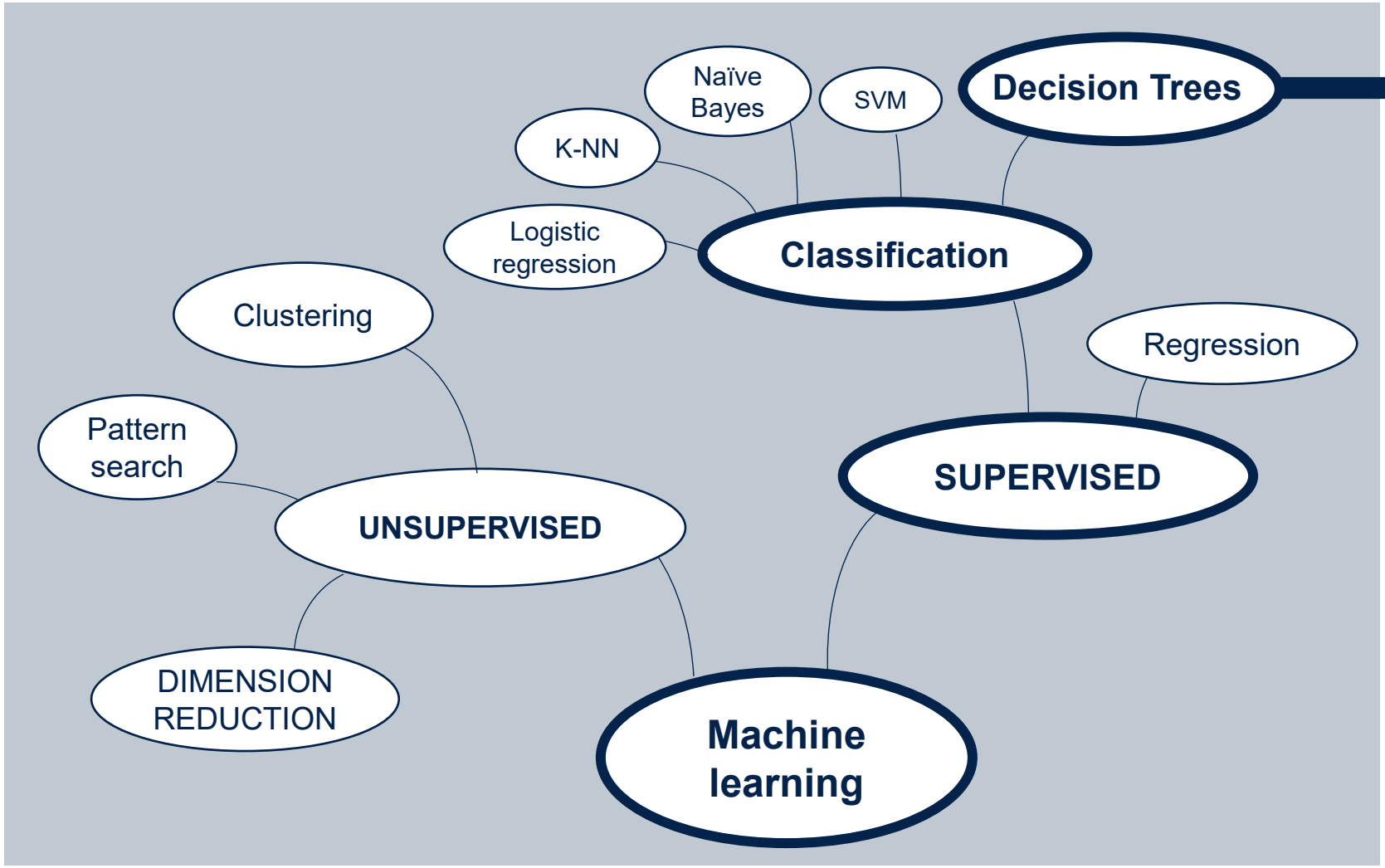
When a complex task or problem involves a large amount of data and lots of variables**, but no existing formula or equation can solve it**

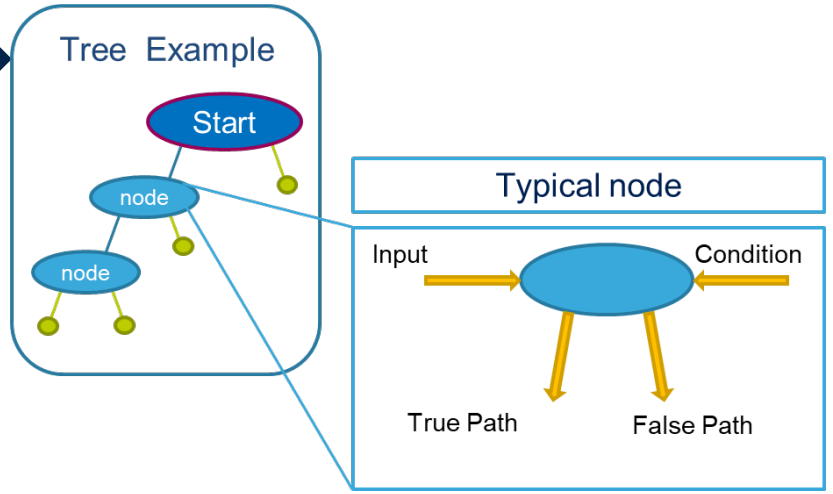
- An example of difficult program
  - How to recognize the hand written digits?
  - Very difficult to define the rules!
  - What makes all these numbers to be identifiable?
  - Is there a pattern?
  - What is it that makes a 2 to be identified as a 2?



life.augmented

Machine Learning Core(MLC) in ST MEMS sensor





ST sensors embed **Decision trees**

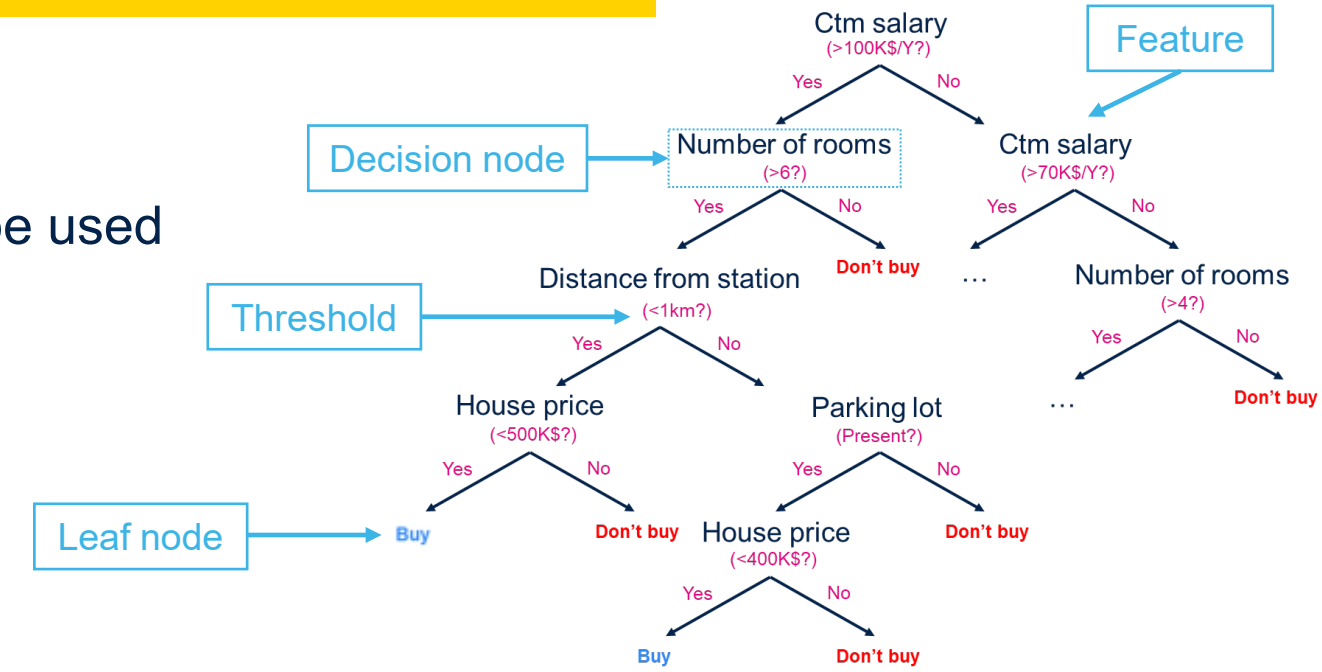
Decision tree runs **Classification**, i.e algorithm that splits objects into classes based on attributes known beforehand

Supervised machine learning technique is used to create decision trees

ML model training Decision tree algorithm

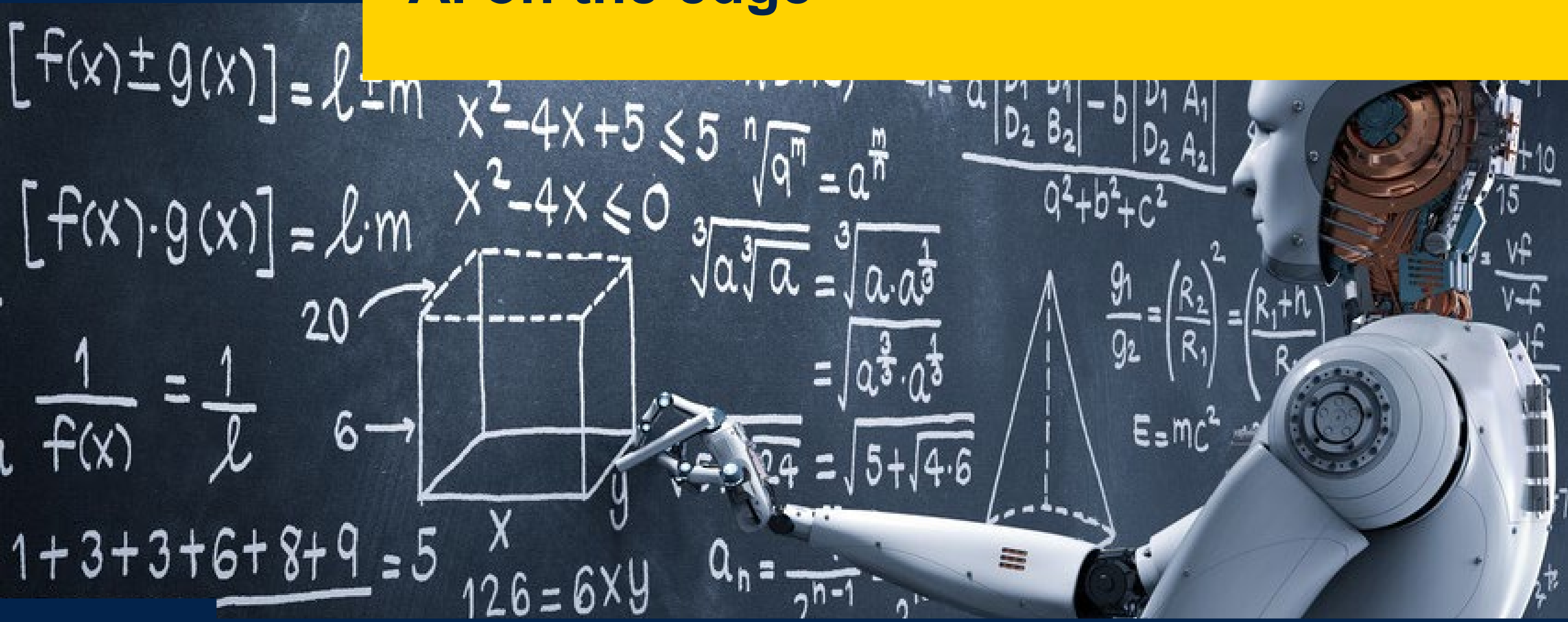
What the ML model learn from the data?

- **Which Features** will be used
- **In which Decision Nodes** the Features will be used
- **Threshold values** for Decision Nodes
- **Leaf Nodes**



Training the same Machine Learning Model with another dataset (i.e. Houses in Milan), the resulting Decision Tree is different to "best fit" the new data given for training.

AI on the edge

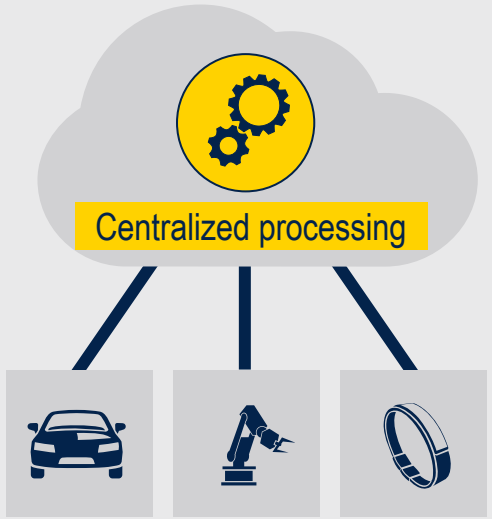


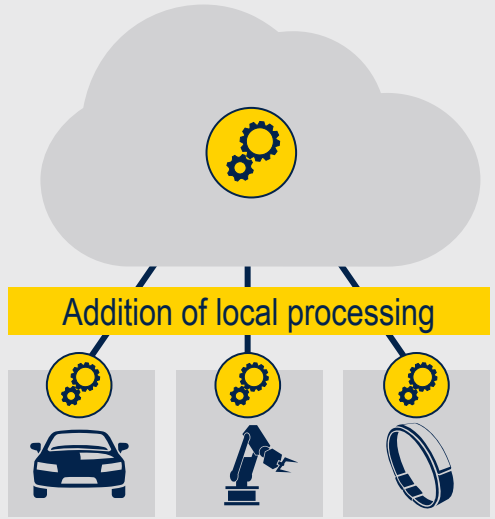
life.augmented

Moving AI to edge computing

Move from Cloud Computing to Edge Computing Now!

Cloud computing	Edge computing
Heavy and expensive computation in the cloud	Optimized computation in the cloud
Large amount of data sent over the network	Smaller amount of data sent over the network
Time-sensitive applications are limited by remote cloud computing	Time-sensitive applications are not limited: locally processed





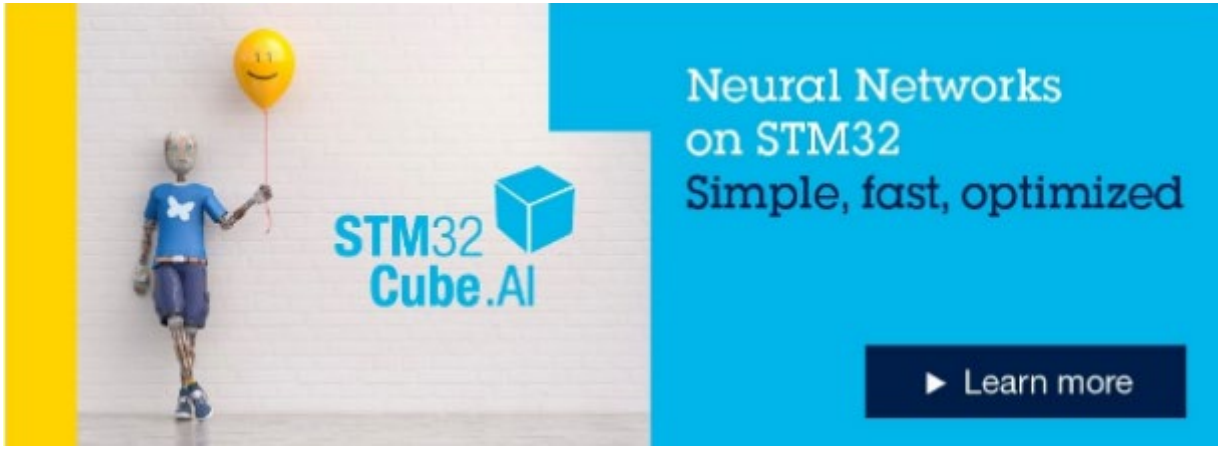
Market needs: Smart sensors with local processing (AI) for real time elaboration and best power efficiency

ST MEMS sensor with MLC

life.augmented

Artificial Intelligence at STMicroelectronics

- Thanks to STM32Cube.AI, pre-trained **Artificial Neural Networks** (ANN) can be run on STM32 microcontrollers.



- Advanced sensors contain a **Machine Learning Core (MLC)**, a Finite State Machine (FSM), and advanced digital functions. They run custom algorithms on the IMU and share the workload from the main processor enabling system functionality while significantly saving power.

life.augmented

Machine Learning Core in ST sensor

life.augmented

New X-Factor sensor products with MLC
LSM6DSOX, LSM6DSRX, ISM330DHCX , IIS2ICLX

We create the new generation of **sensors** to allow **developers** exploiting their potential by improving the overall system **efficiency**

- ST has **all the building blocks** for the **Machine Learning Core** for tools and software

- Our constantly expanding development ecosystem makes **design fast** and help you **reduce the time-to-market**

Consumer grade

LSM6DSOX

LSM6DSRX

Industrial grade

ISM330DHCX

IIS2ICLX

life.augmented

Basic process of MLC development

How it works in 5 simple steps and with an intuitive use case

1

User defines **classes** to be recognized and **collects data logs**.

2

Clean and **label** logs. Define **features** best characterizing the identified classes.

3

Machine Learning tools **build tree** based on logs and features.

4

ST tool generates sensor configuration with **embedded decision tree.**

5

Configure the sensor and **run** the application.

Capture data

Label data & extract features

Build decision tree

Embed decision tree

Process new data

life.augmented

Data collection – ST HW and SW tools

Form Factor

SensorTile.Box
STEVAL-MKSBOX1V1

CONSUMER

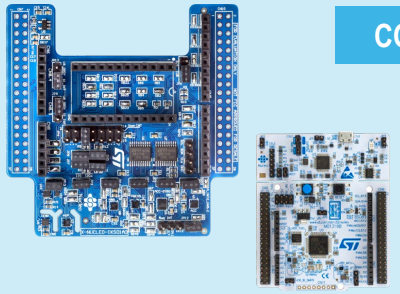
STWIN: Wireless Industrial Node
STEVAL-STWINKT1

INDUSTRIAL

Evaluation board

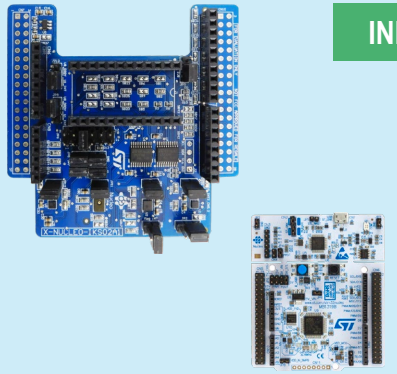
STM32Nucleo expansion
X-NUCLEO-IKS01A3

CONSUMER



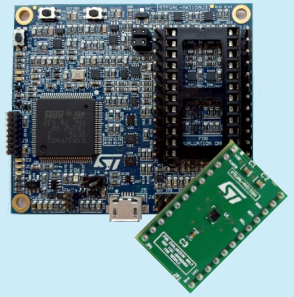
STM32Nucleo expansion
X-NUCLEO-IKS02A1

INDUSTRIAL



Professional board

Profi MEMS tool
STEVAL-MKI109V3



AUTOMOTIVE

INDUSTRIAL

CONSUMER

Software solutions

AlgoBuilderSuite

Algo Builder

NUCLEO

UNICO

All-in-one software package facilitate the programming of sensors for an easy and intuitive experience without writing any single line of code.

life.augmented

Features calculation inside MEMS

Raw data

Acc_X, Acc_Y, Acc_Z
Acc_V, Acc_V2

Gyro_X, Gyro_Y, Gyro_Z
Gyro_V, Gyro_V2

ExtSens_X, ExtSens_Y, ExtSens_Z
ExtSens_V, Ext_Sens_V2

Filters

Filtered data
(High-pass, Band-pass, IIR1 and IIR2 filters)

Features

Mean

$$\frac{1}{WL}\sum_{k=0}^{WL-1} I_k$$

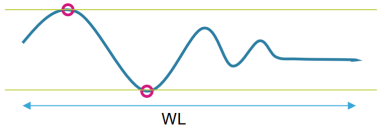
Variance

$$\left(\frac{\sum_{k=0}^{WL-1} I_k^{\,2}}{WL}\right) - \left(\frac{\sum_{k=0}^{WL-1} I_k}{WL}\right)^2$$

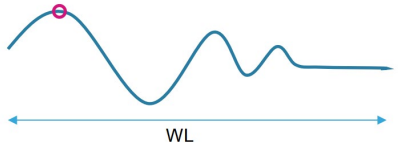
Energy

$$\sum_{k=0}^{WL-1} I_k^{\,2}$$

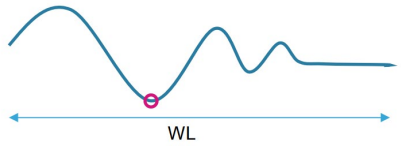
Peak-to-Peak



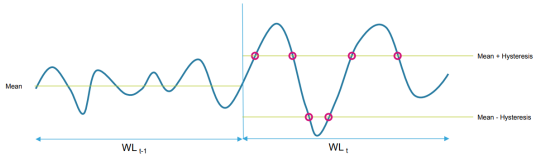
Maximum



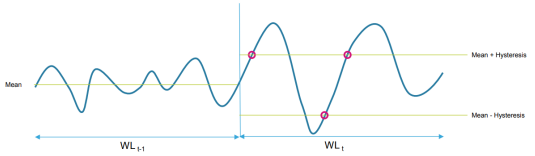
Minimum



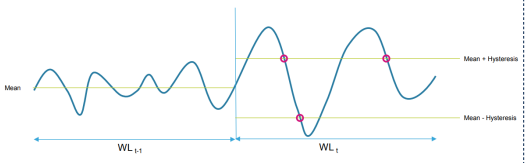
Zero crossing



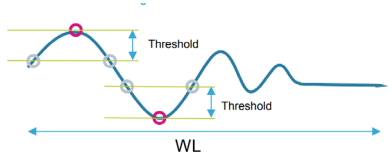
Positive Zero crossing



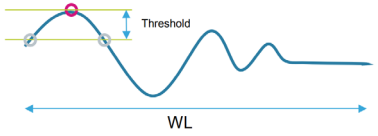
Negative Zero crossing



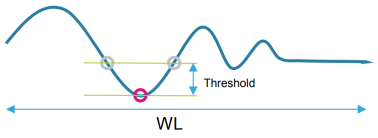
Peak detector



Positive Peak detector



Negative Peak detector



Note: the window for features computation (WL) is configurable (from 1 to 255 samples). It is not a moving window

life.augmented

Decision tree algorithm
Failed motor detection example

PROBLEM:

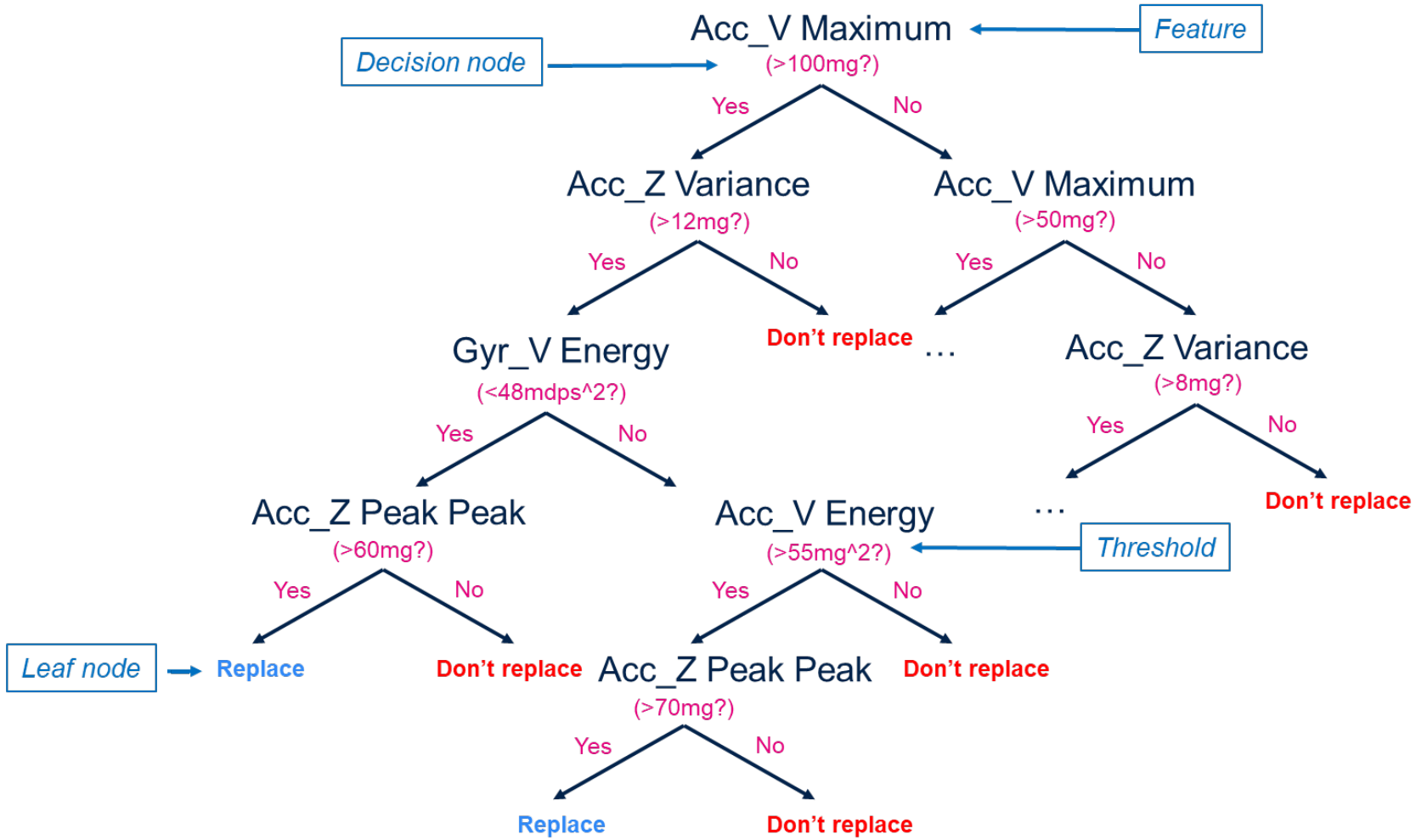
- Does the motor need replacement?

FEATURES (Inputs):

- Acc_X Peak Peak
- Acc_Z Peak Peak
- Acc_Z Variance
- Acc_V Energy
- Acc_V Maximum
- Gyr_Z Variance
- Gyr_V Energy
- Gyr_V Maximum
- …

OUTPUT:

- Replace or Don’t replace



life.augmented

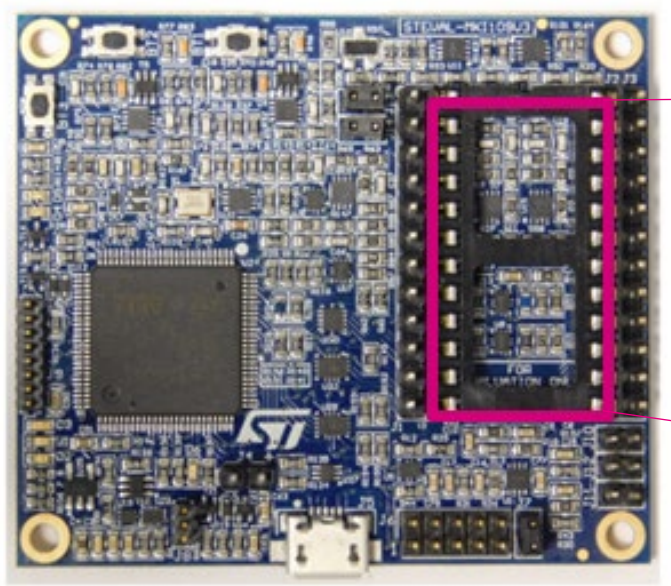
[Demo] Application state estimation with MLC

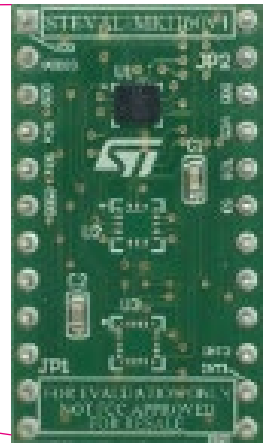
life.augmented

H/W: ProfiMEMS board with LSM6DSOX

General-purpose MEMS sensor evaluation tool

Professional MEMS mother board	Sensor adapter board





ST MEMS motherboard with STM32F401VE and compatible with all ST MEMS adapters

STEVAL-MKI109V3

Sensor adapter board for a standard DIL24 socket

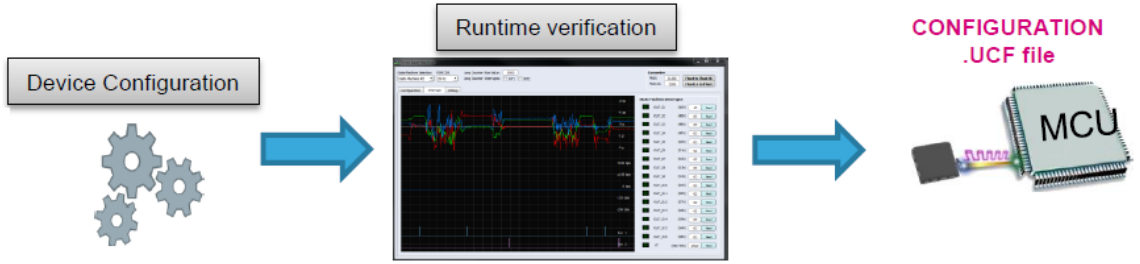
LSM6DSOX（STEVAL-MKI197V1）

life.augmented

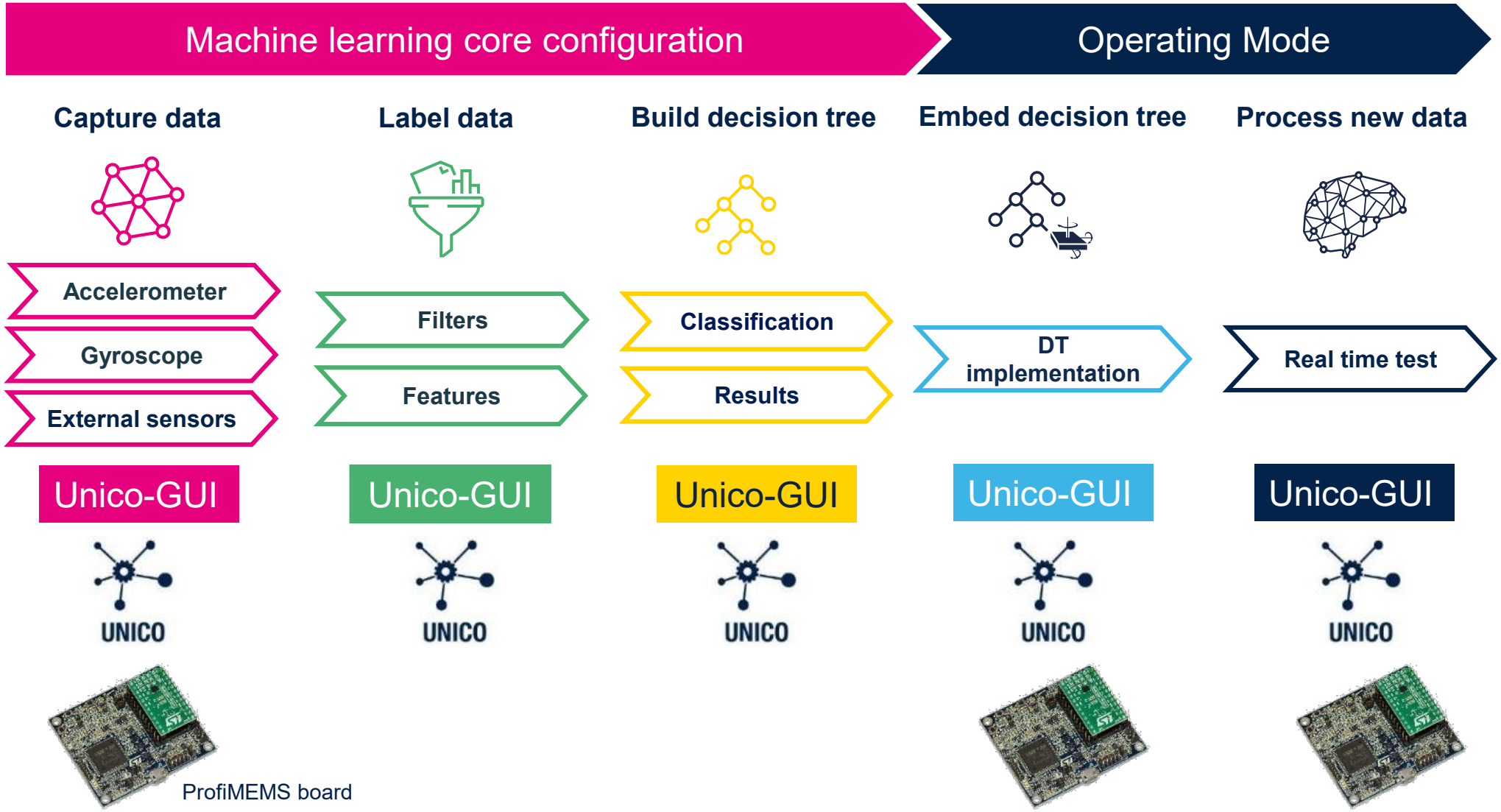
S/W: Unico-GUI(PC application)

PC application for sensor evaluation

- All **Motion/ Environmental MEMS** adapter boards (STEVAL-MKIxxx) supported
  - Compatible with Profi MEMS tool (STEVAL-MKI109V3)
- Cross-platform GUI for **Windows**, **Linux** and **MacOS X**
  - STSW-MKI109W for Windows OS
  - STSW-MKI109L for Linux OS
  - STSW-MKI109M for Mac OS
- **Features**:
  - Device registers configuration, Data logging
  - Configuration file (.ucf) creation & C code generation
  - **FSM / MLC development support**



Creating a decision tree with Unico-GUI



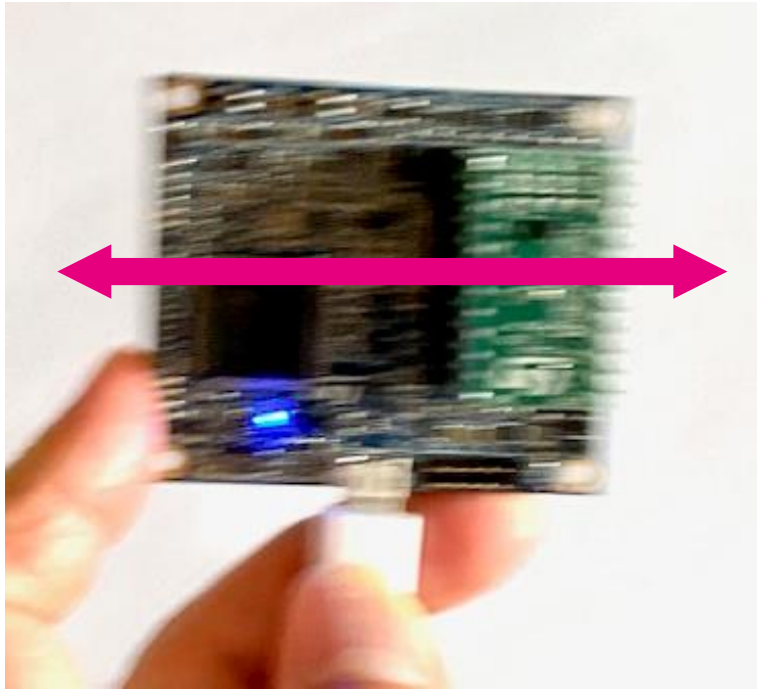
Application state estimation demo with MLC

Detect the following 3 states with the Machine Learning Core!

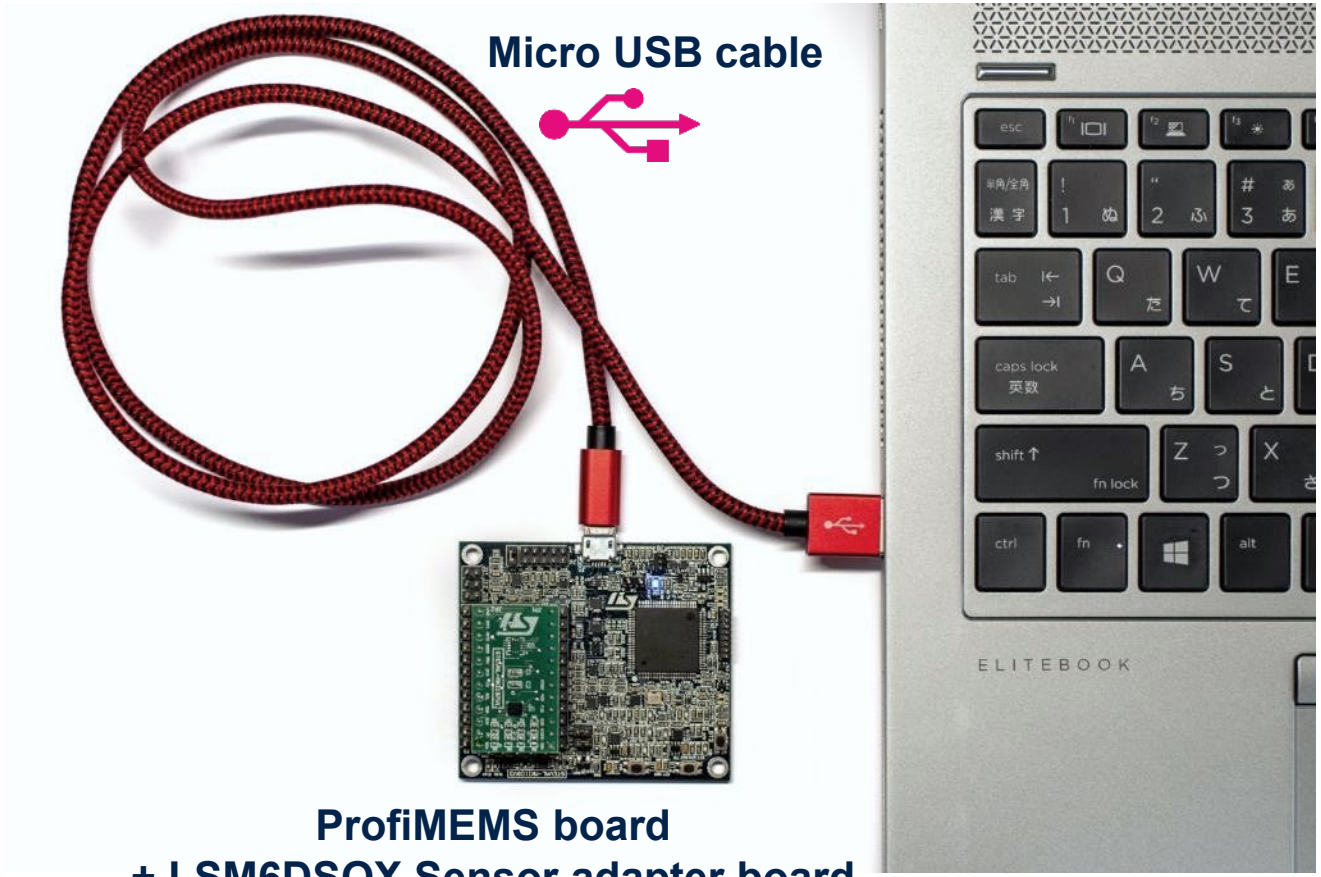
State #1: Face-up

State #2: Face-down

State #3: Shaking



STEP 1- (a): Connect the board to the PC

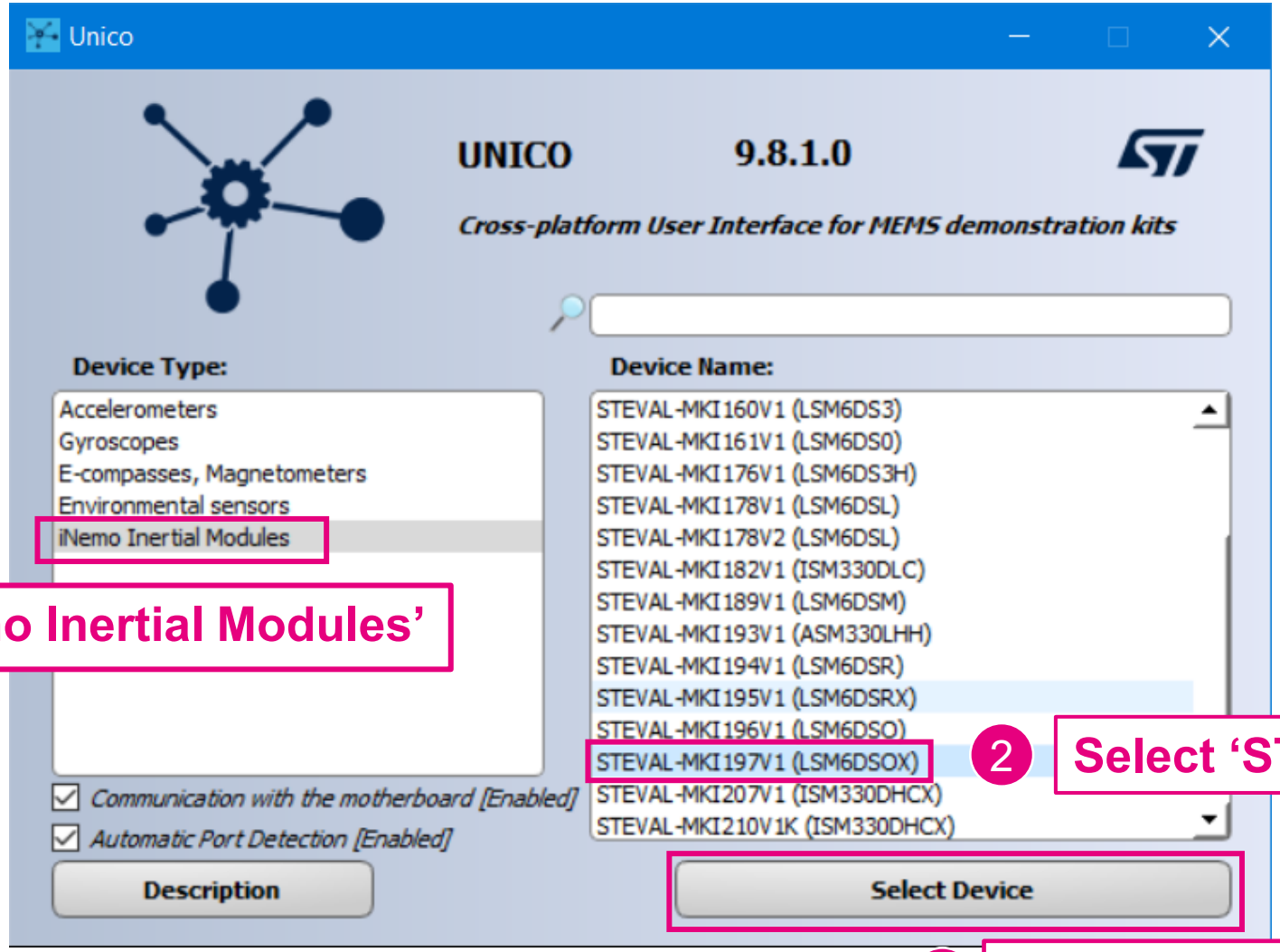


ProfiMEMS board
+ LSM6DSOX Sensor adapter board

Computer

1. **Connect ProfiMEMS board with LSM6DSOX sensor adapter board to PC USB port**

STEP 1-(b): Execute Unico-GUI tool

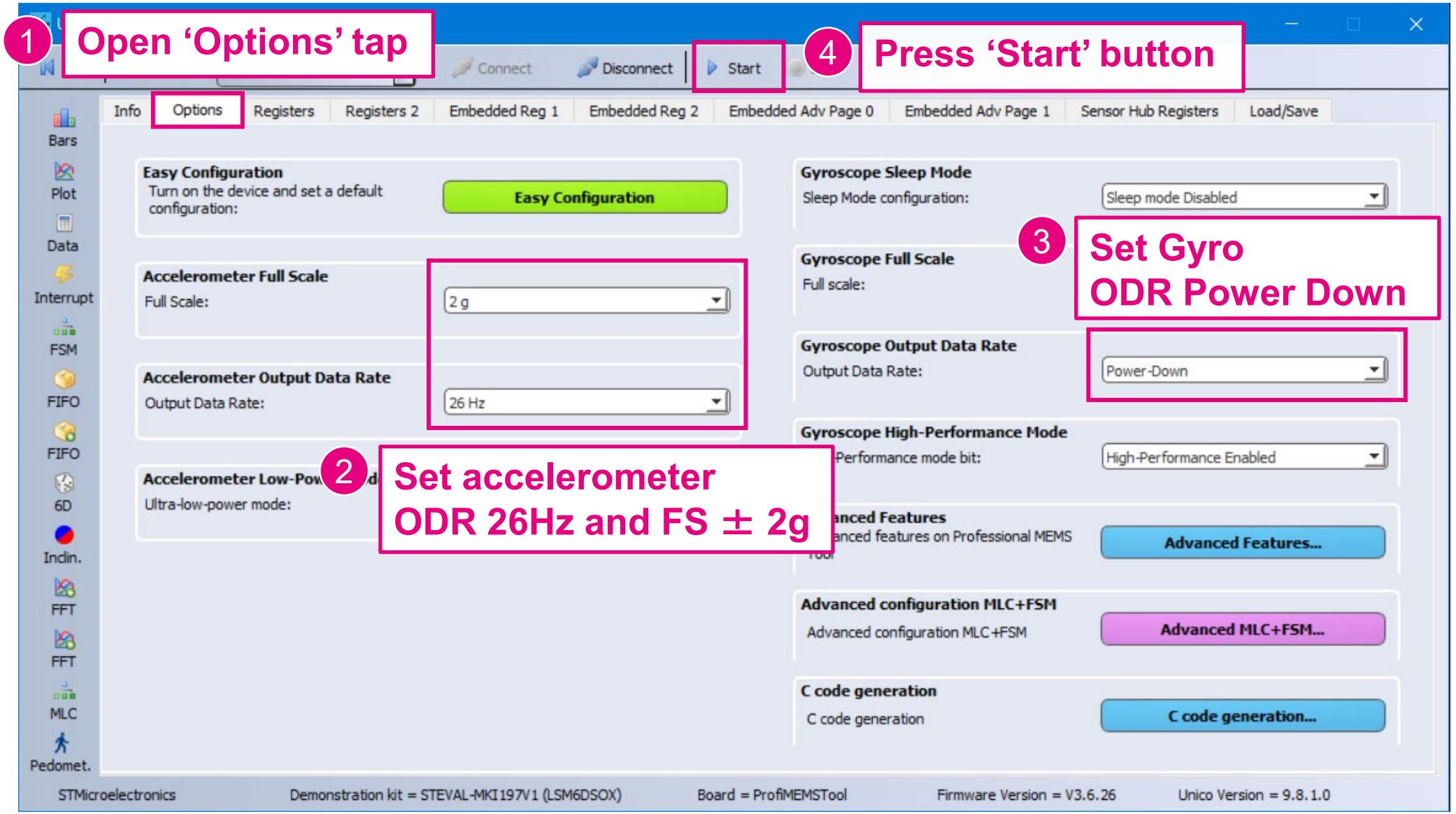


1 Select ‘iNemo Inertial Modules’

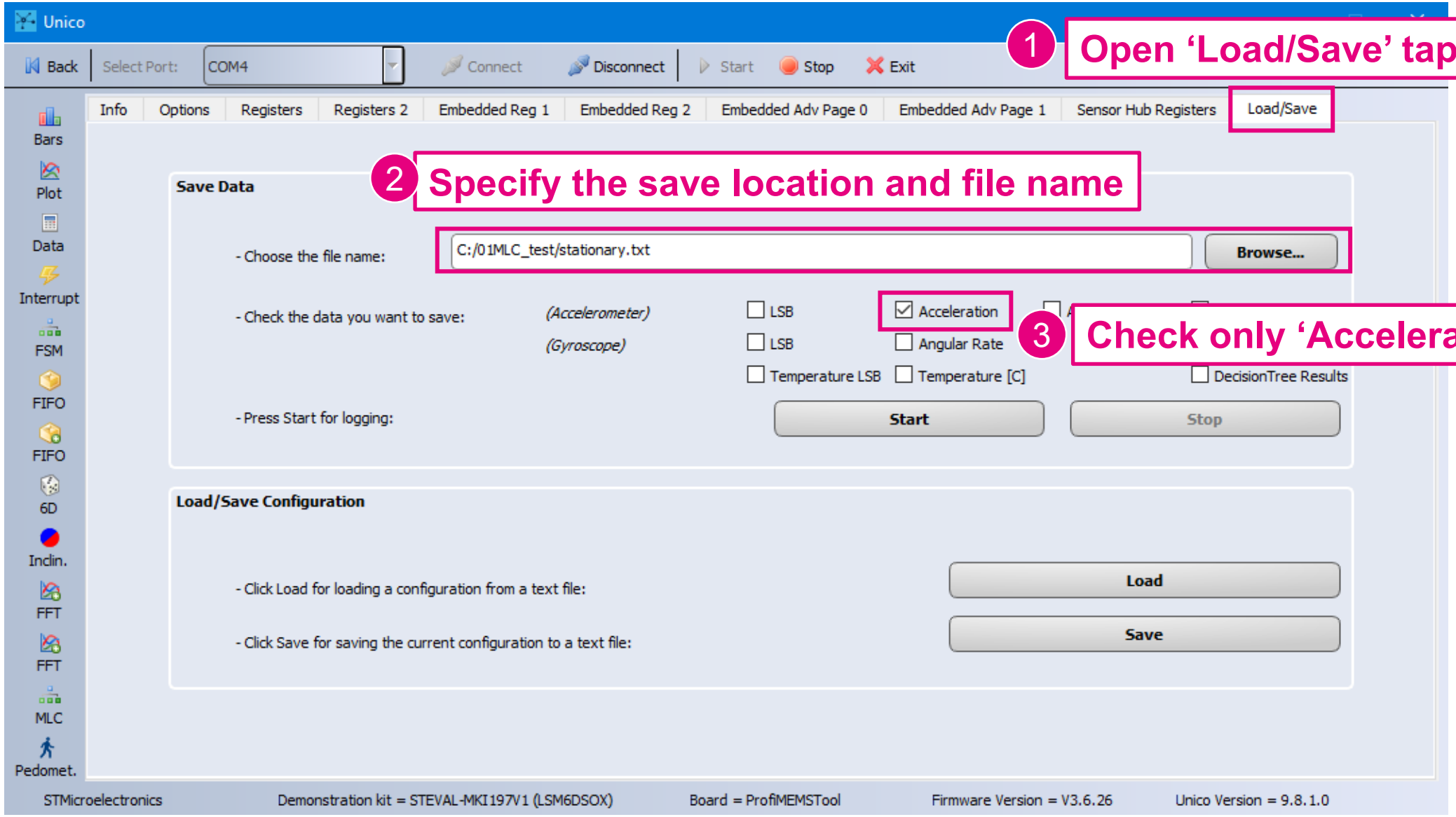
2 Select ‘STEVAL-MKI197V1’

3 Press ‘Select Device’ button

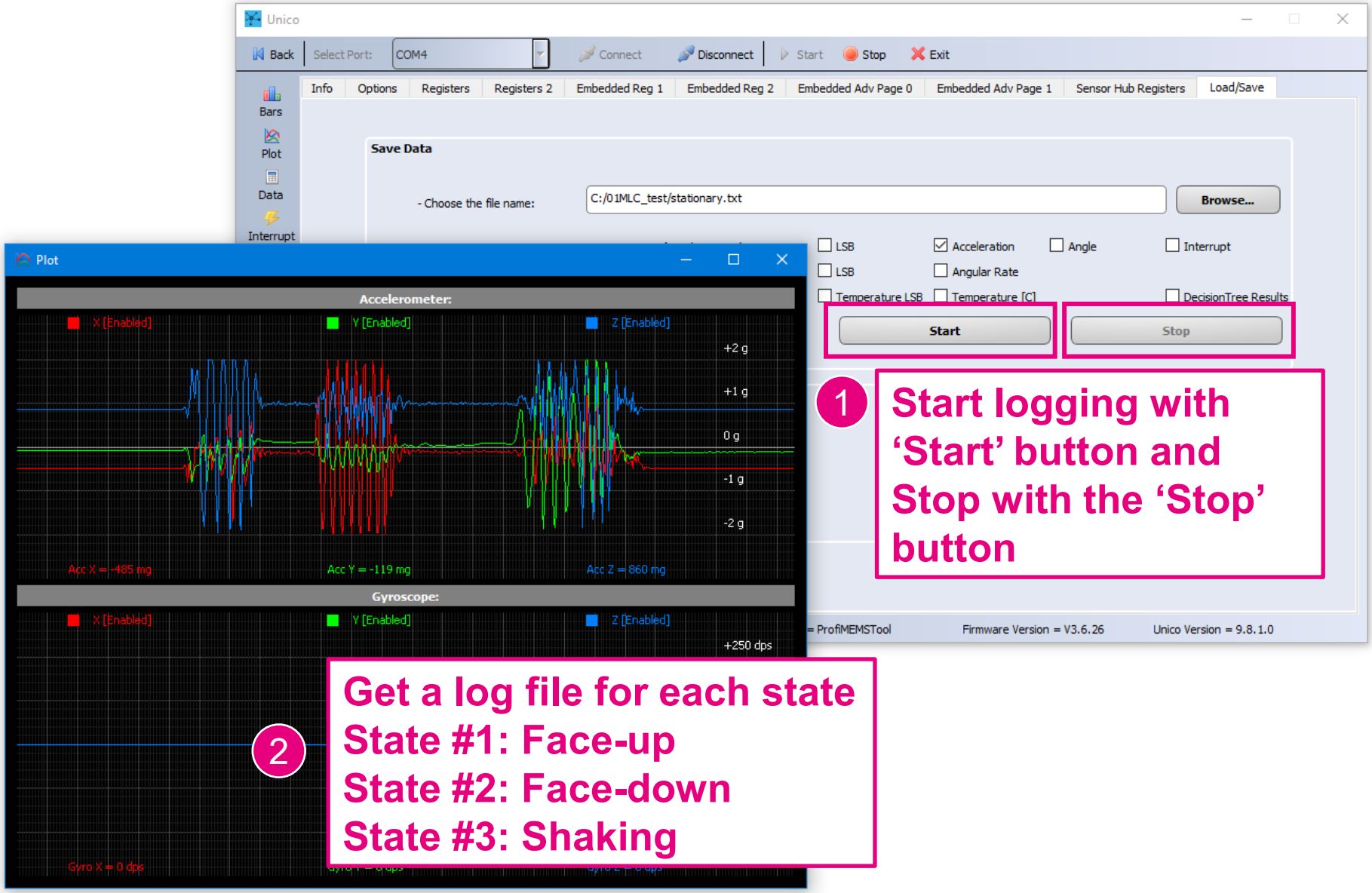
STEP 2: Set ODR and Full scale



STEP 3- (a): Get the log file



STEP 3- (b): Get the log file

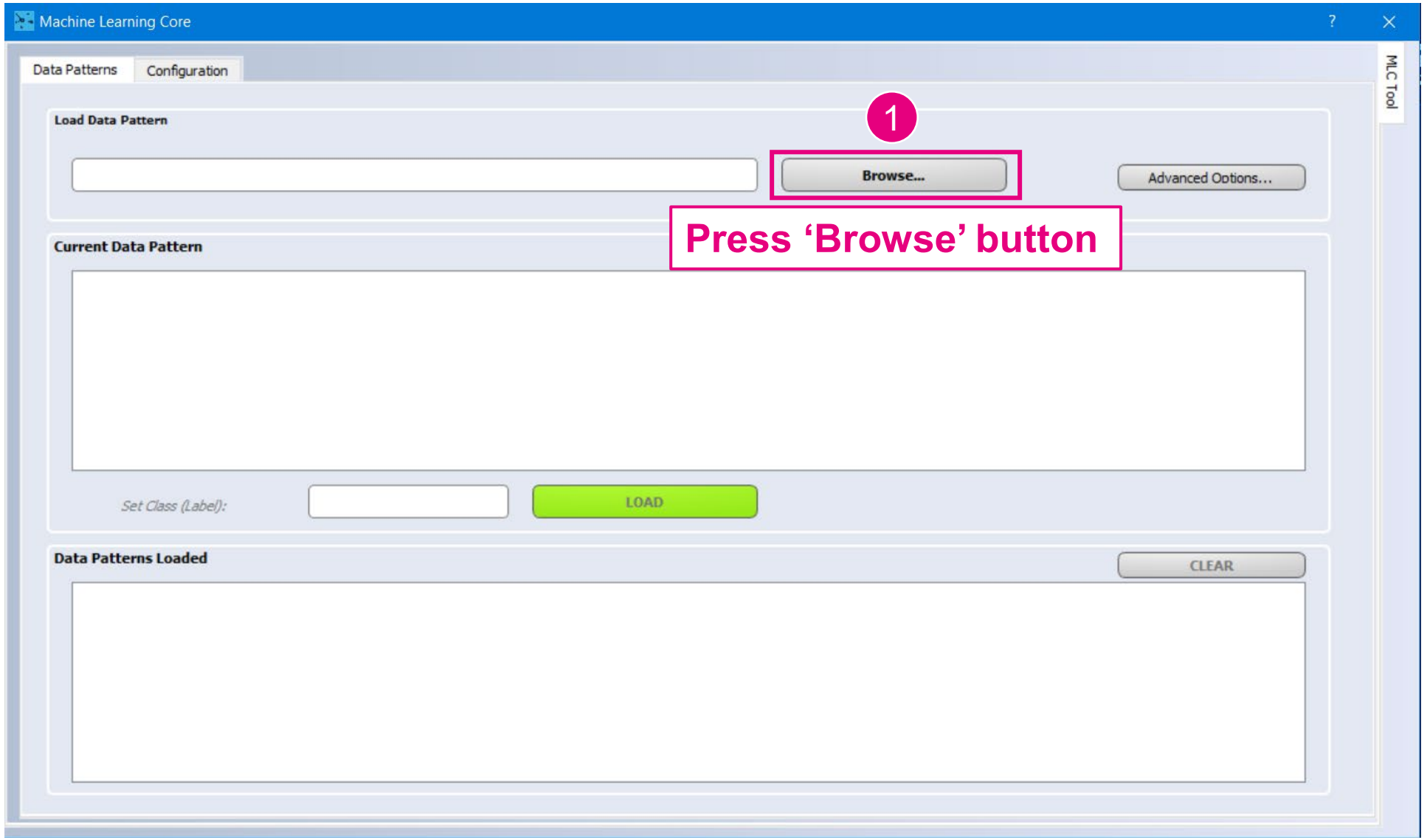


Labeling

STEP 4: MLC settings and decision tree creation

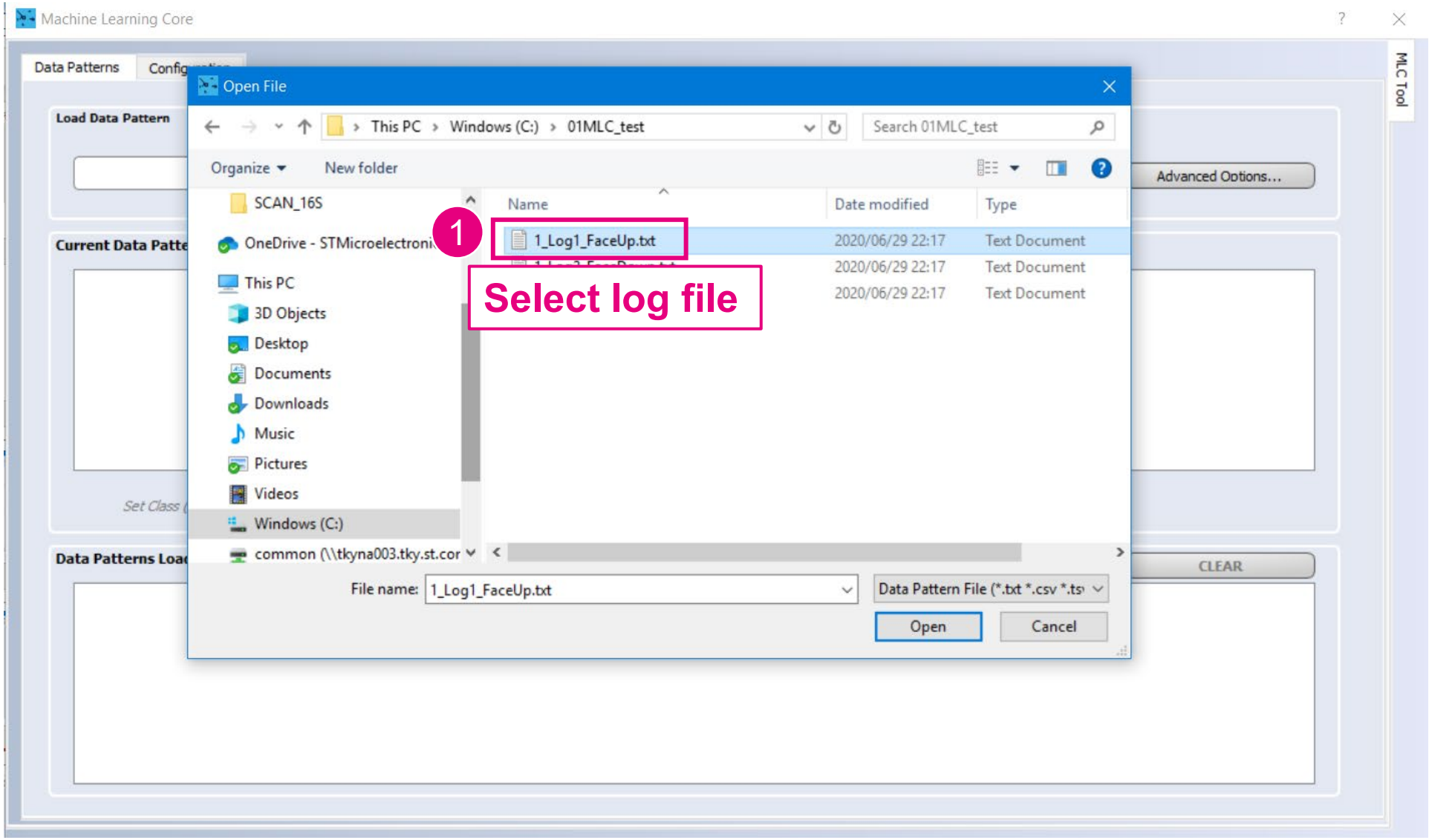
1 **Launch MLC tool**

STEP 4: MLC settings and decision tree creation

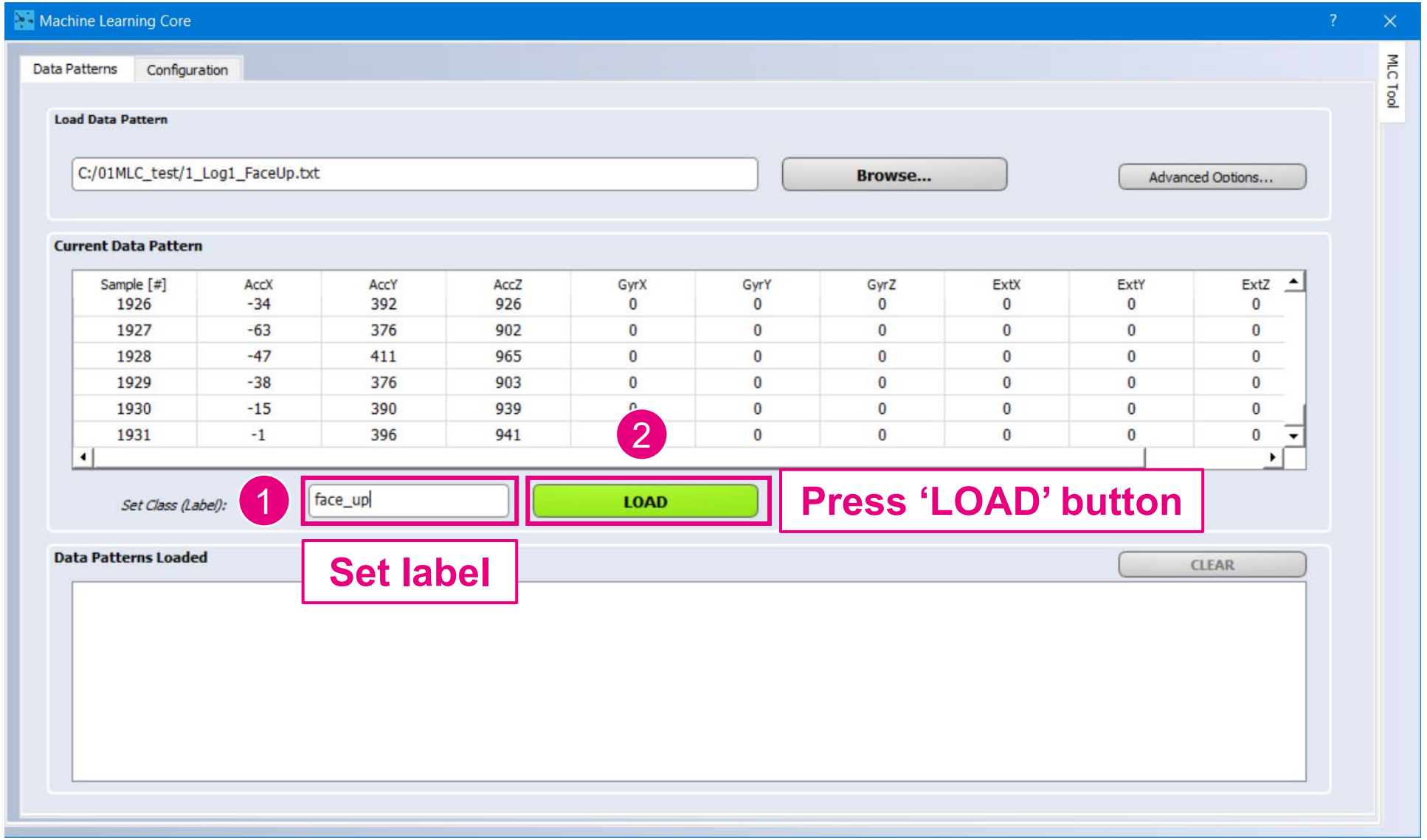


Labeling

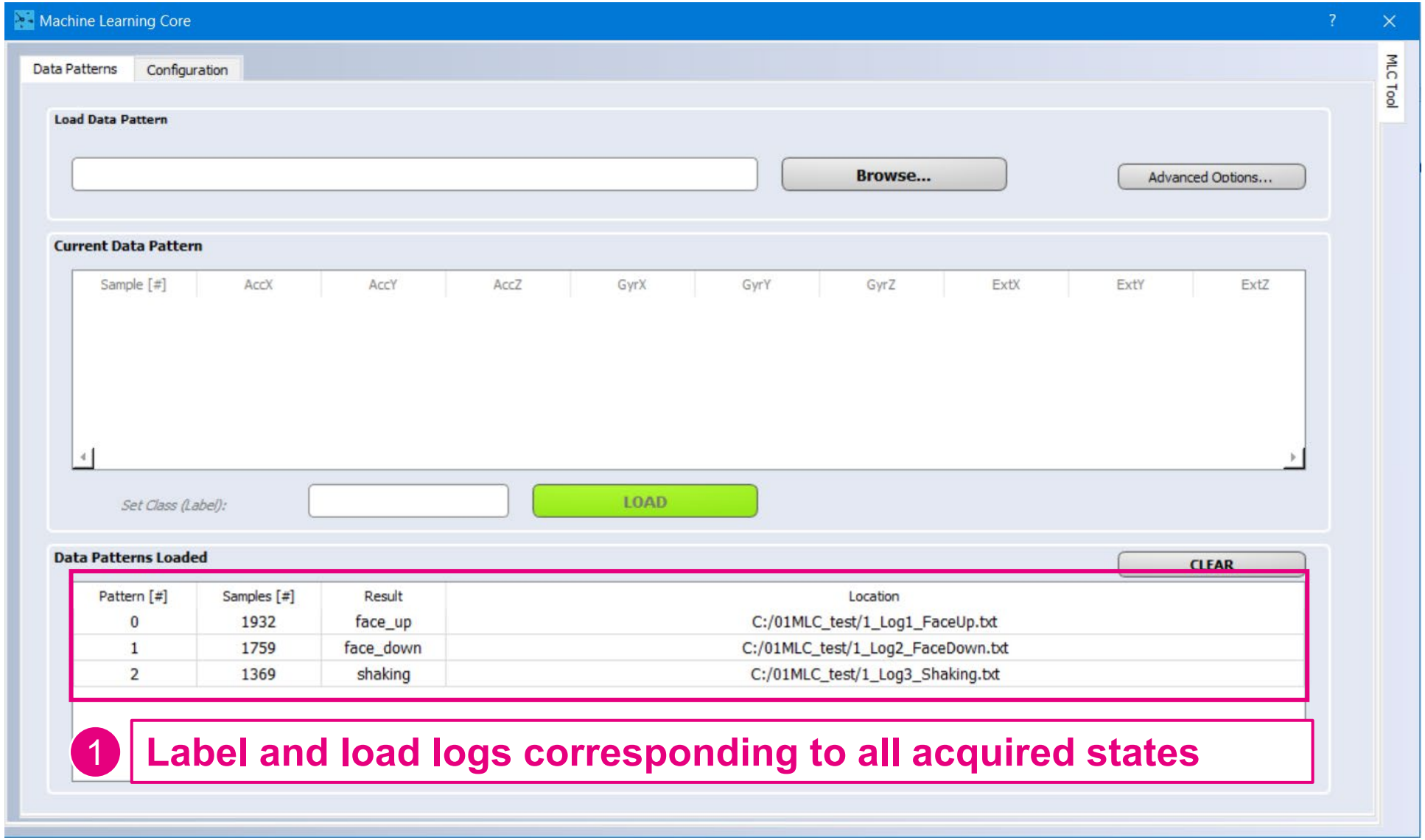
STEP 4: MLC settings and decision tree creation



STEP 4: MLC settings and decision tree creation



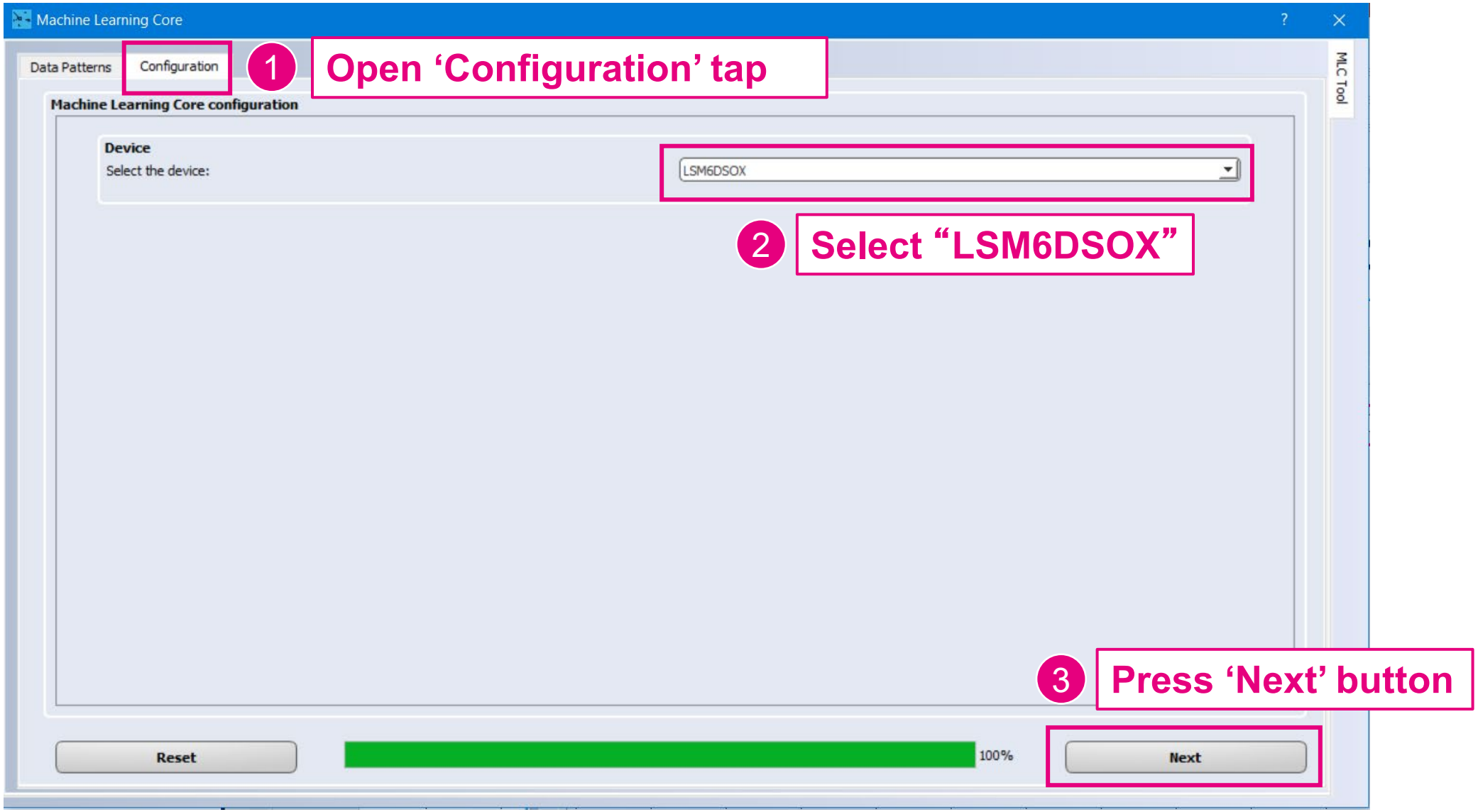
STEP 4: MLC settings and decision tree creation



UNICO

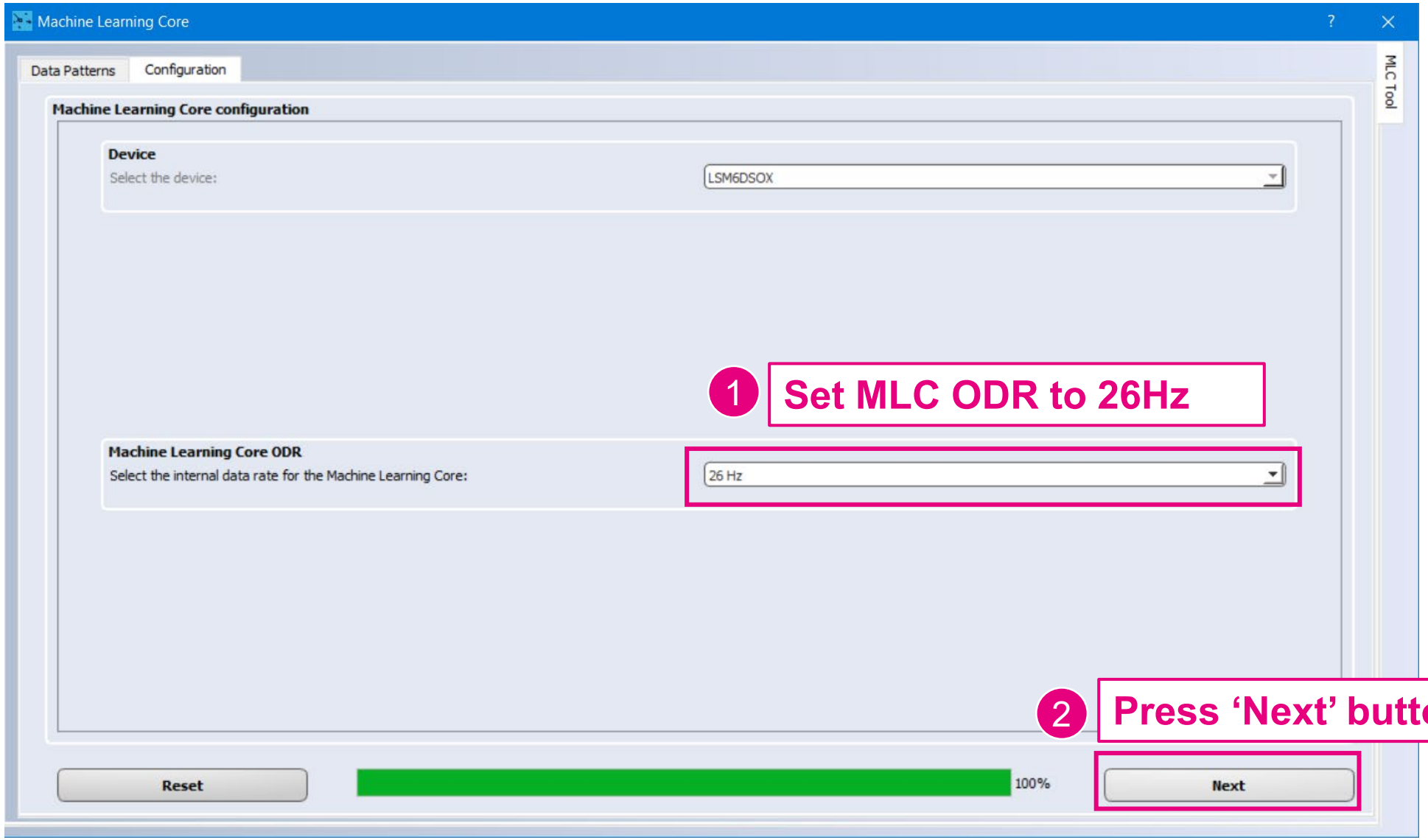
Labeling

STEP 4: MLC settings and decision tree creation



life.augmented

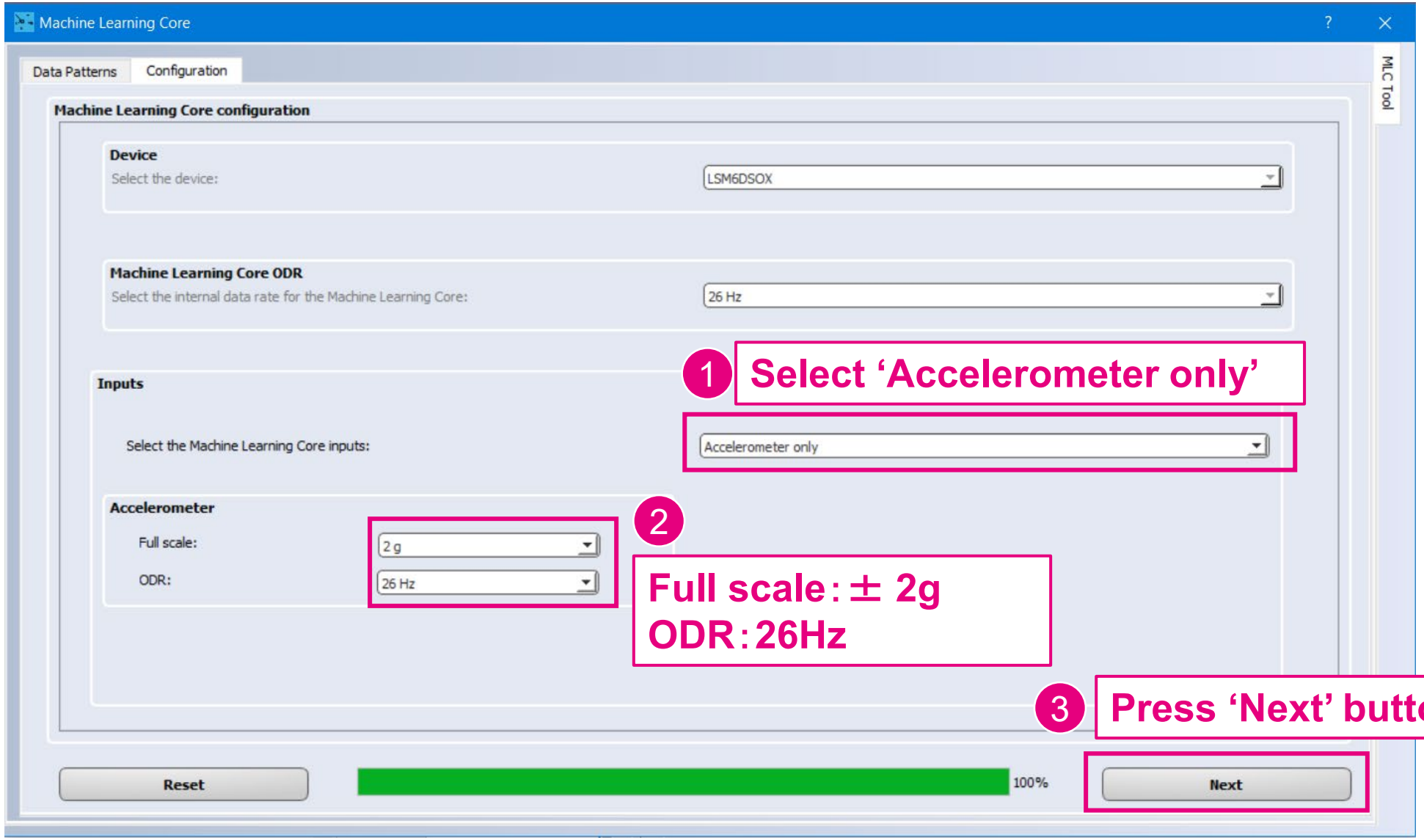
STEP 4: MLC settings and decision tree creation



UNICO

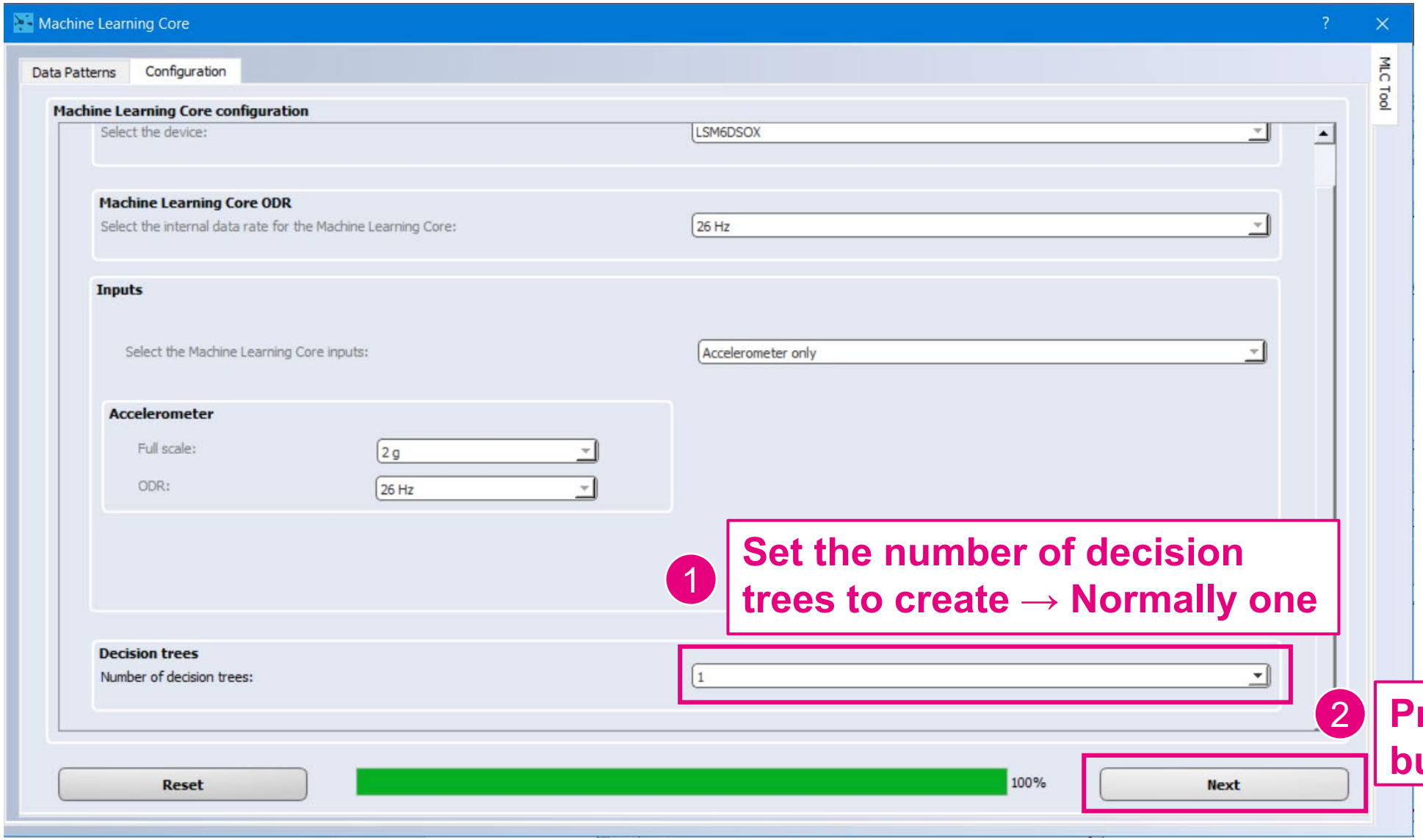
Labeling

STEP 4: MLC settings and decision tree creation

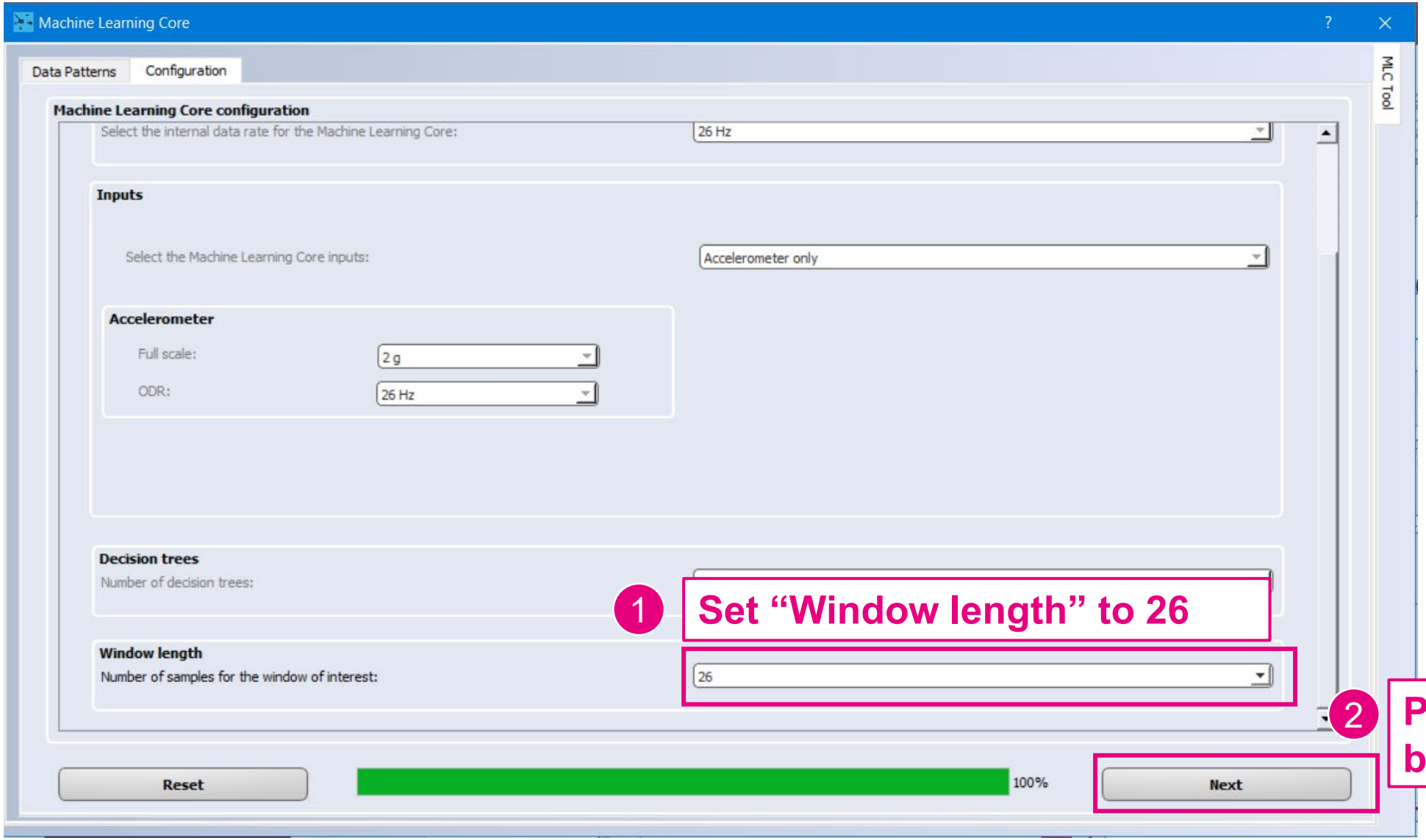


life.augmented

STEP 4: MLC settings and decision tree creation



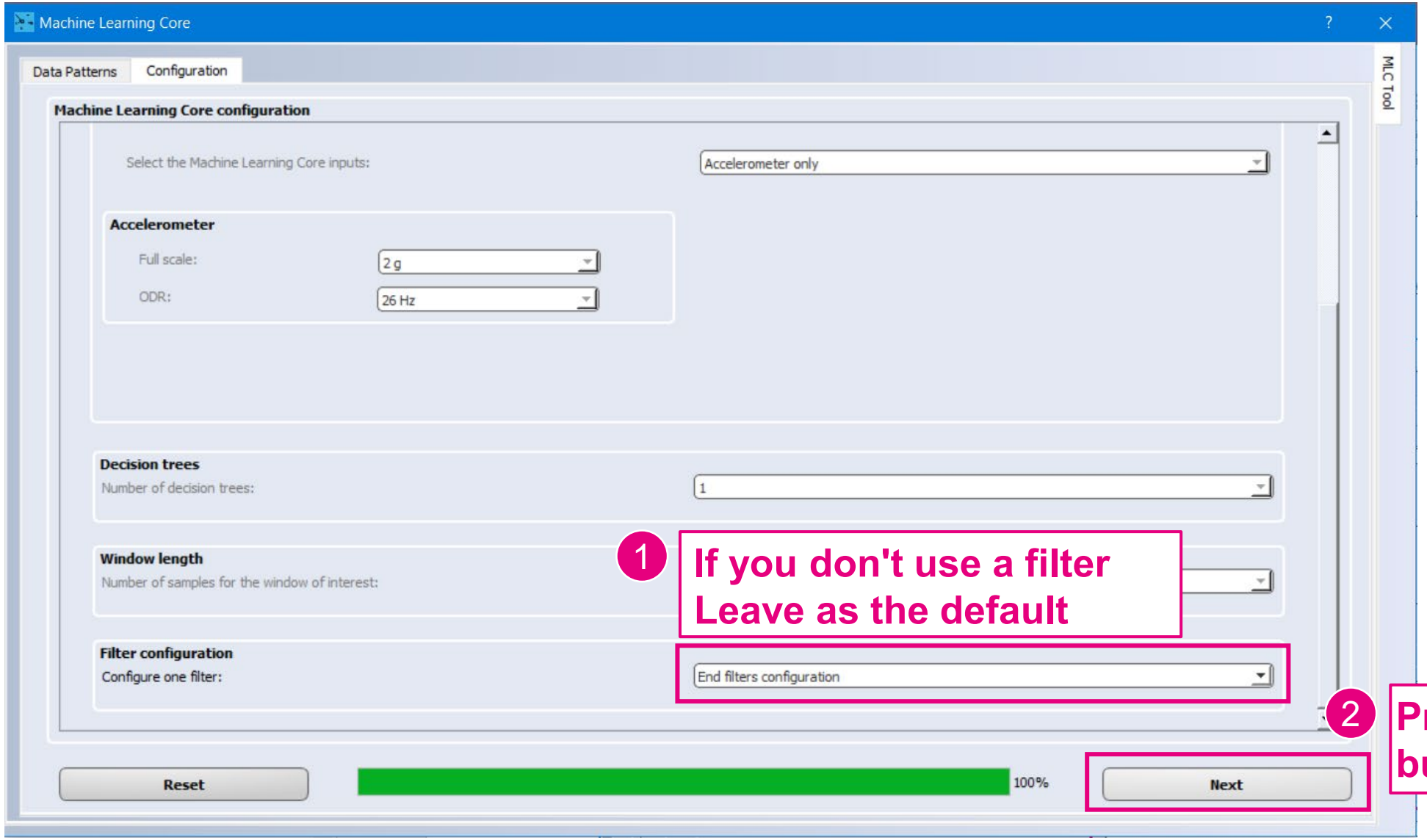
STEP 4: MLC settings and decision tree creation



UNICO

Labeling

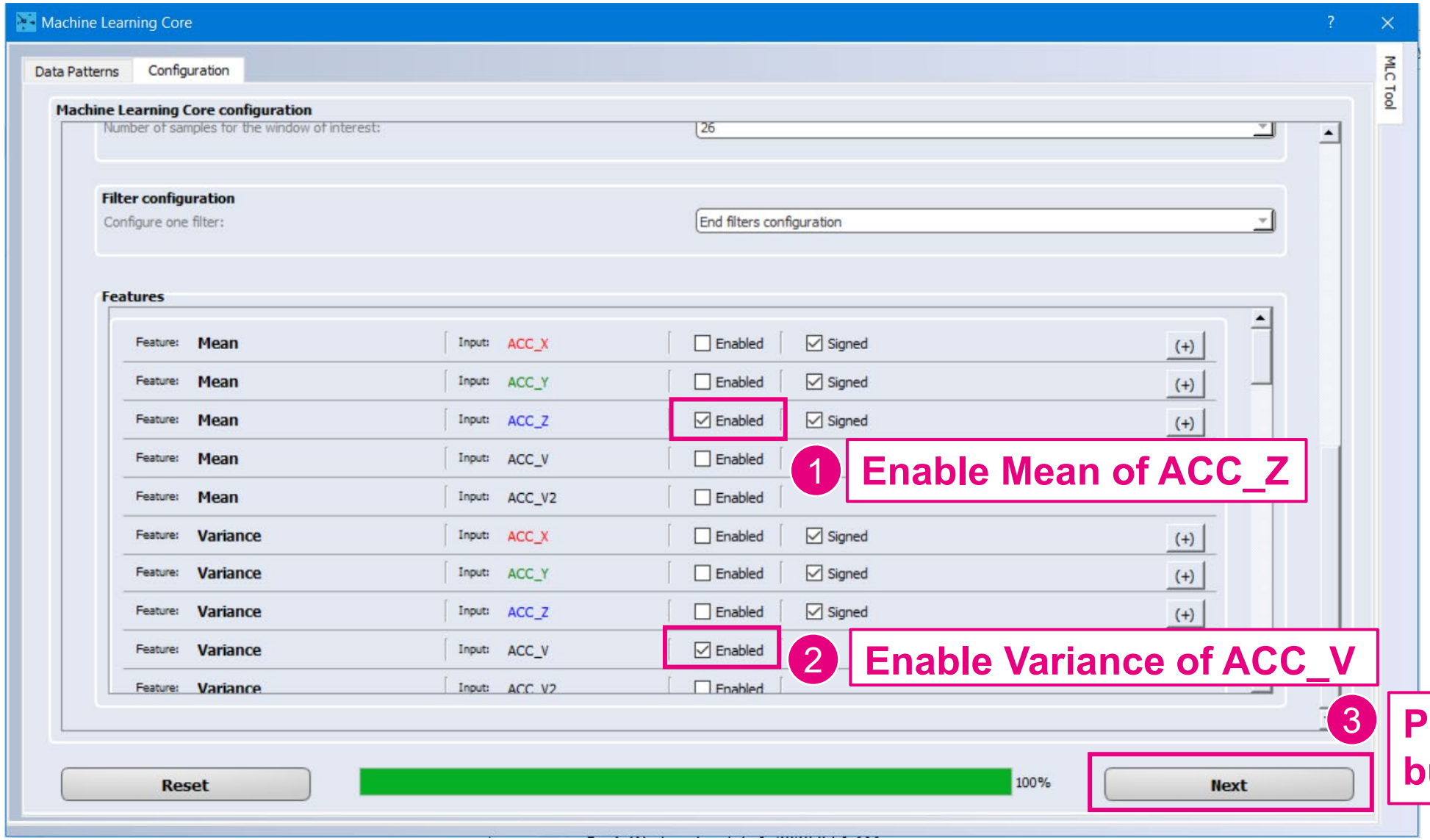
STEP 4: MLC settings and decision tree creation



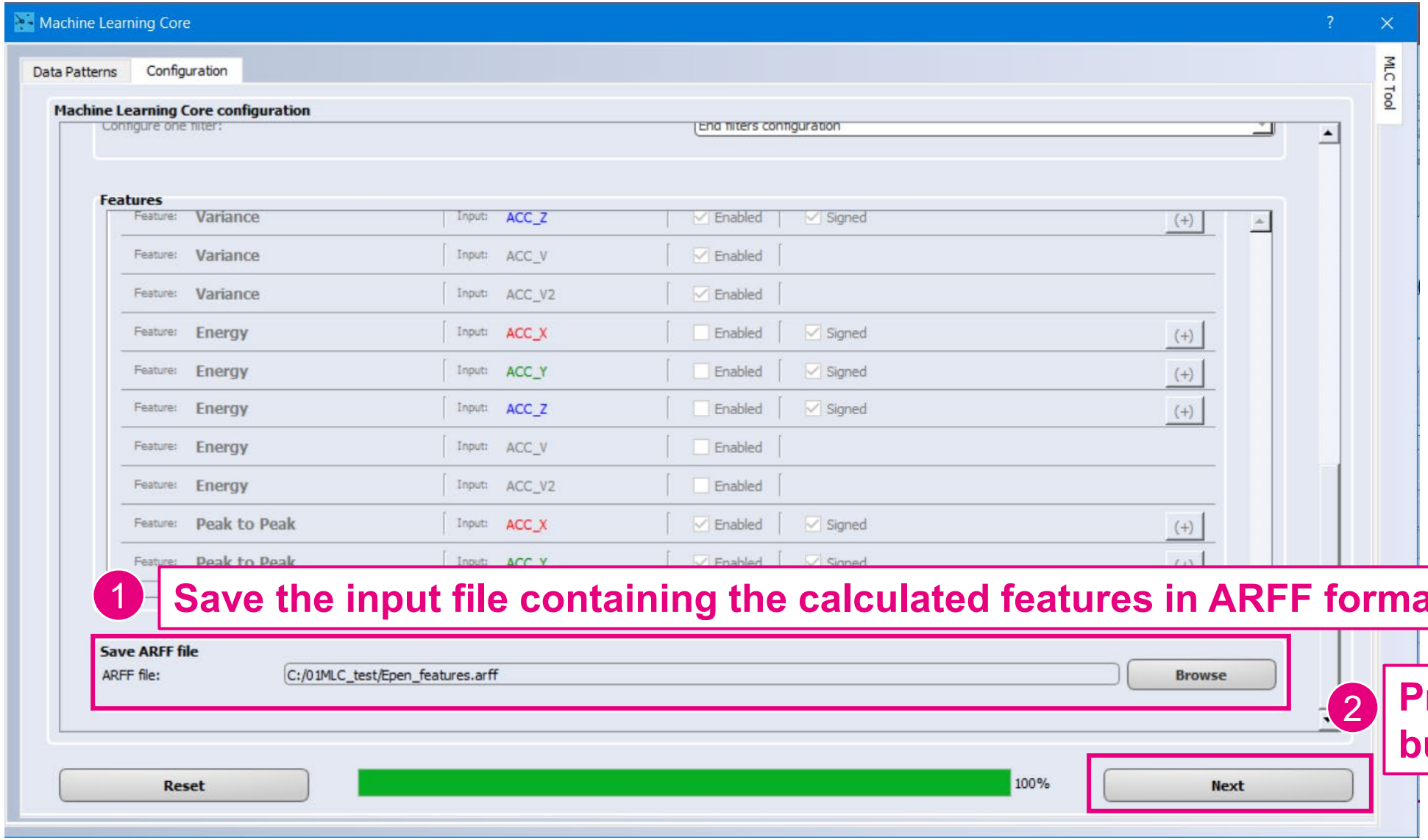
life.augmented

Labeling

STEP 4: MLC settings and decision tree creation



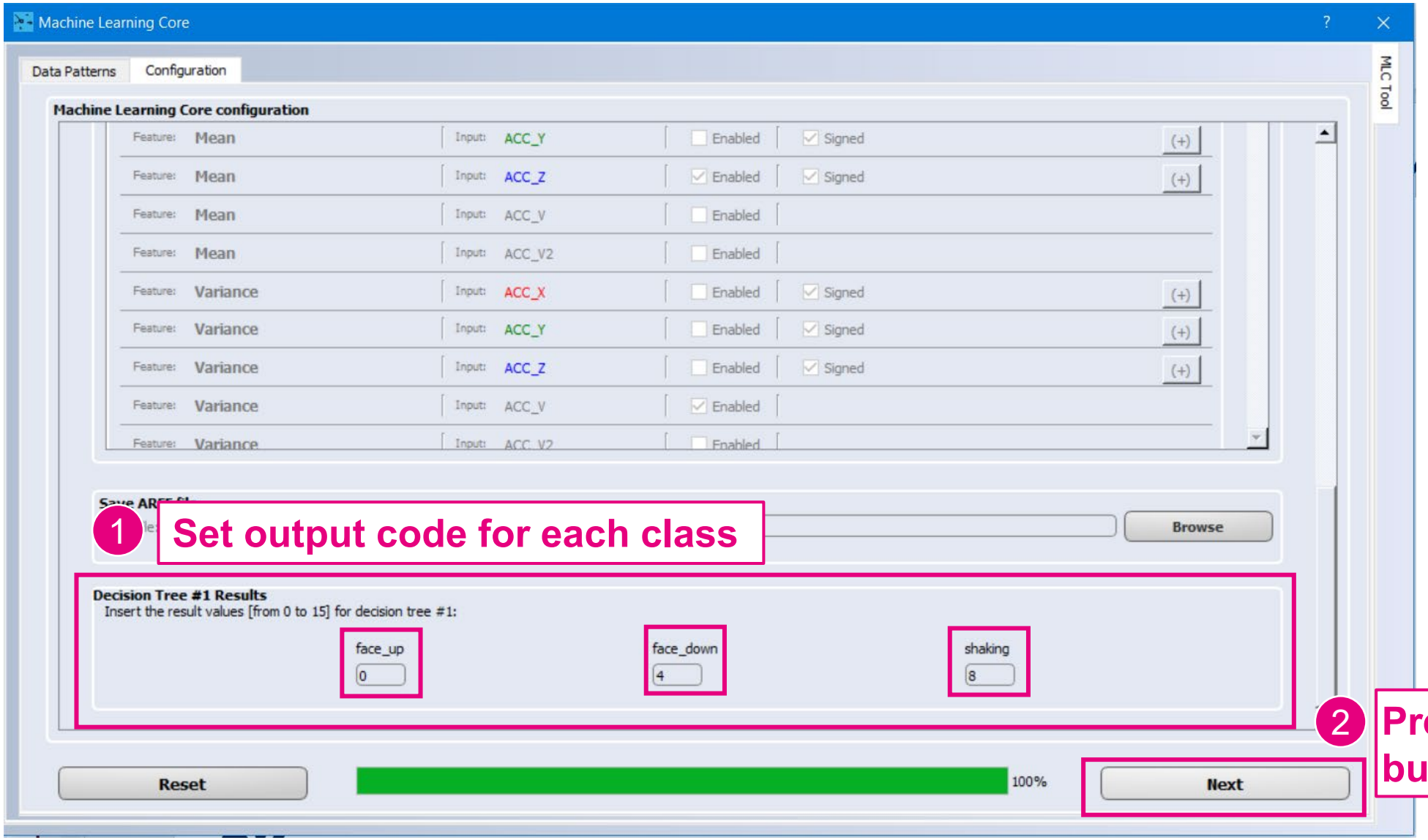
STEP 4: MLC settings and decision tree creation



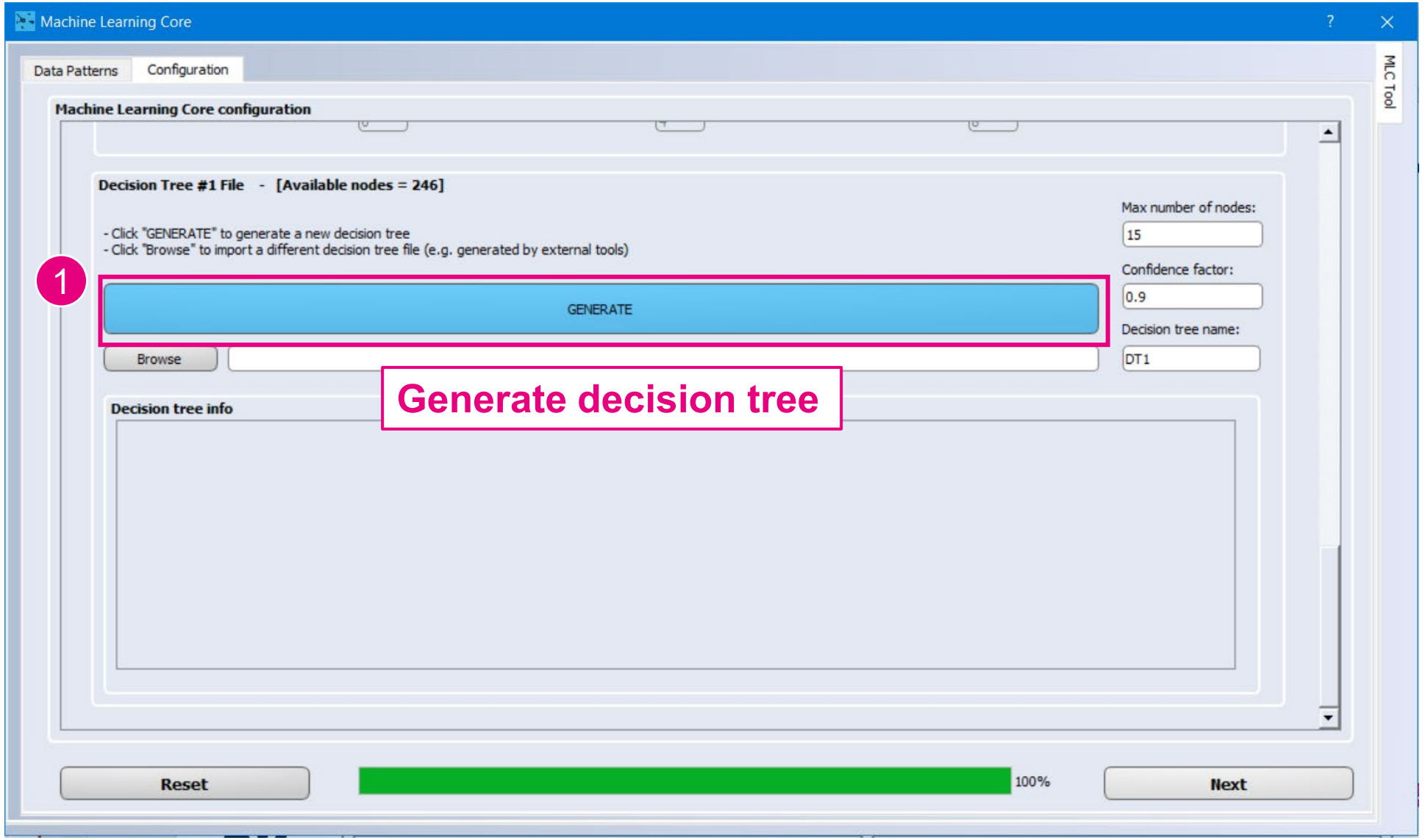
1 Save the input file containing the calculated features in ARFF format

2 Press 'Next' button

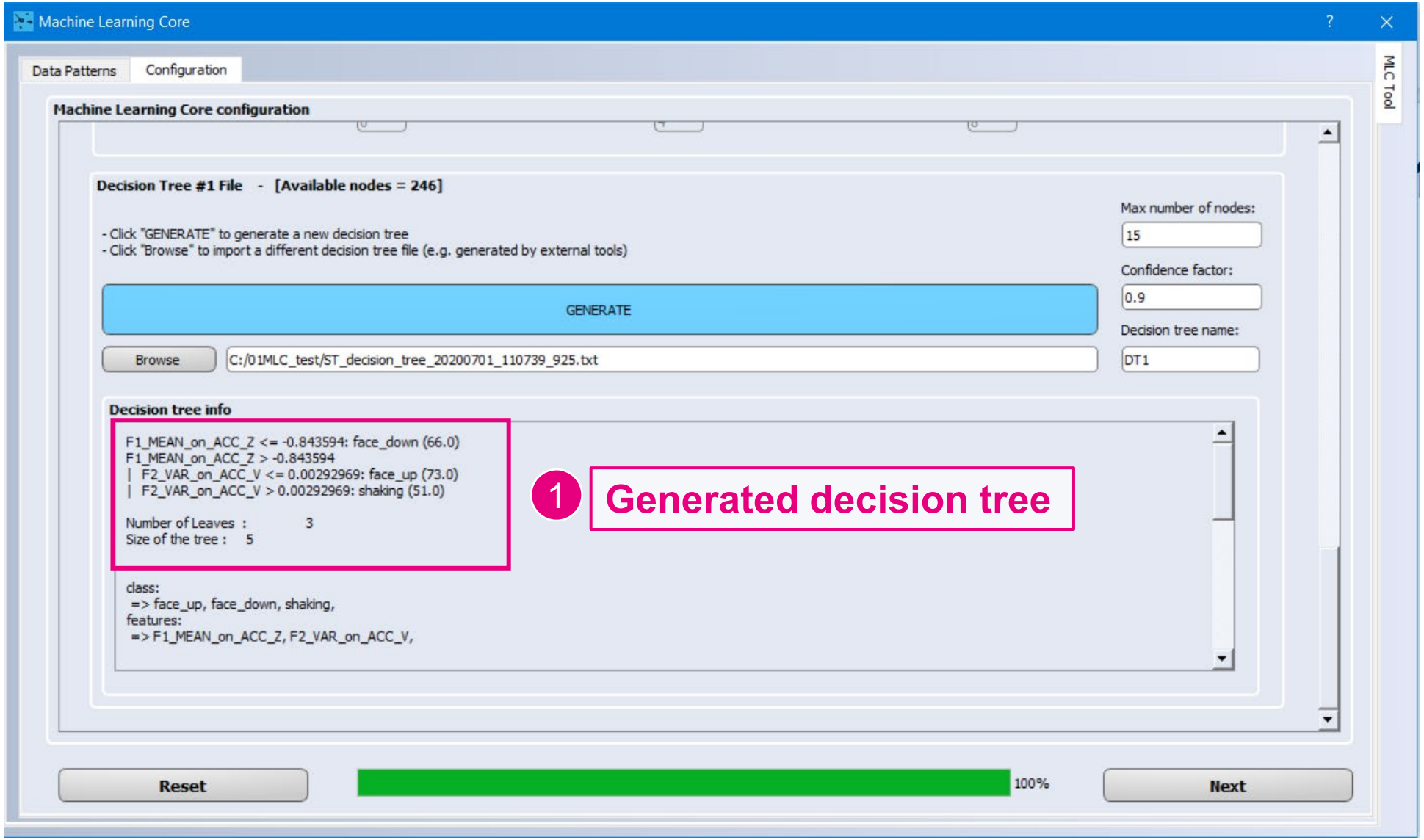
STEP 4: MLC settings and decision tree creation



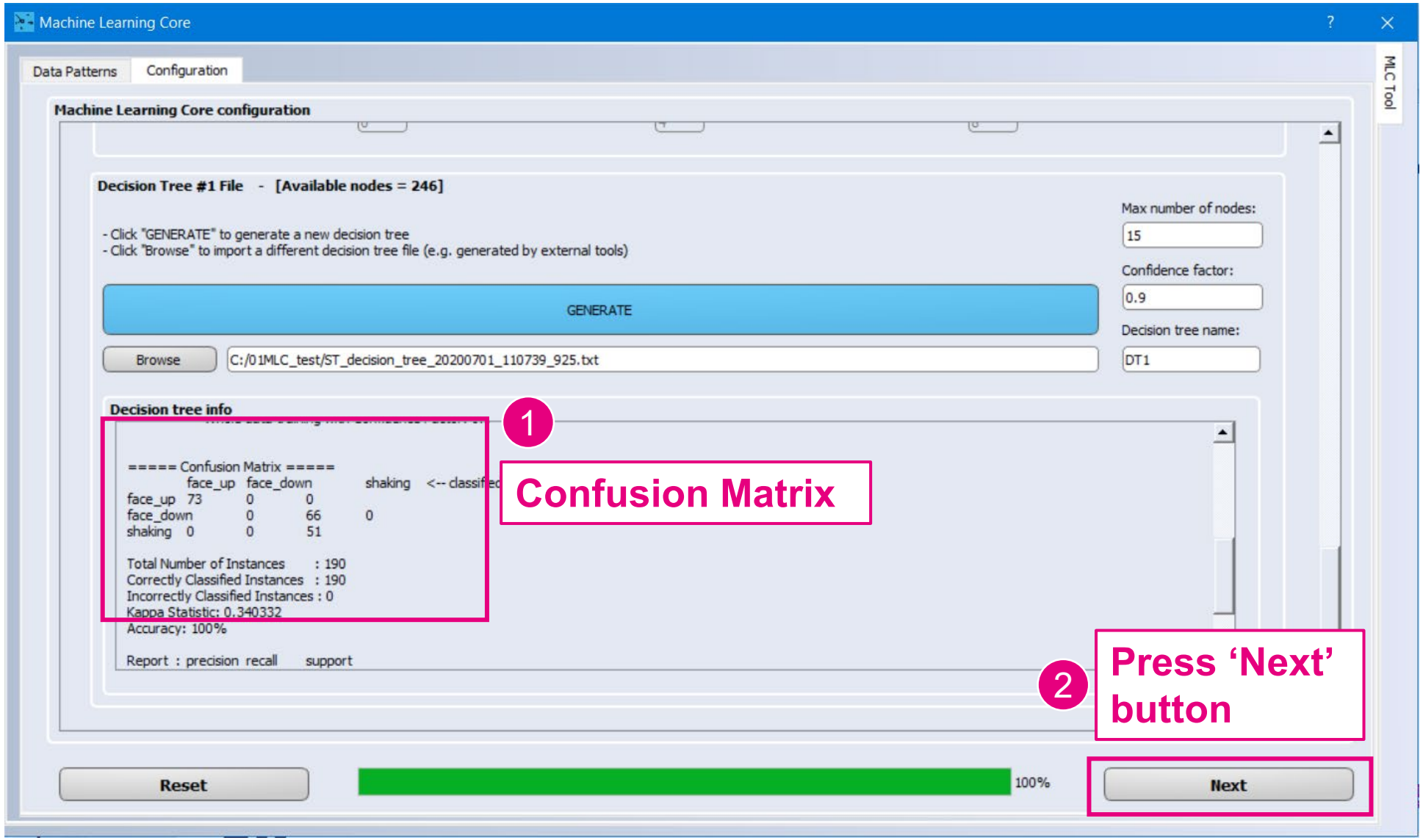
STEP 4: MLC settings and decision tree creation



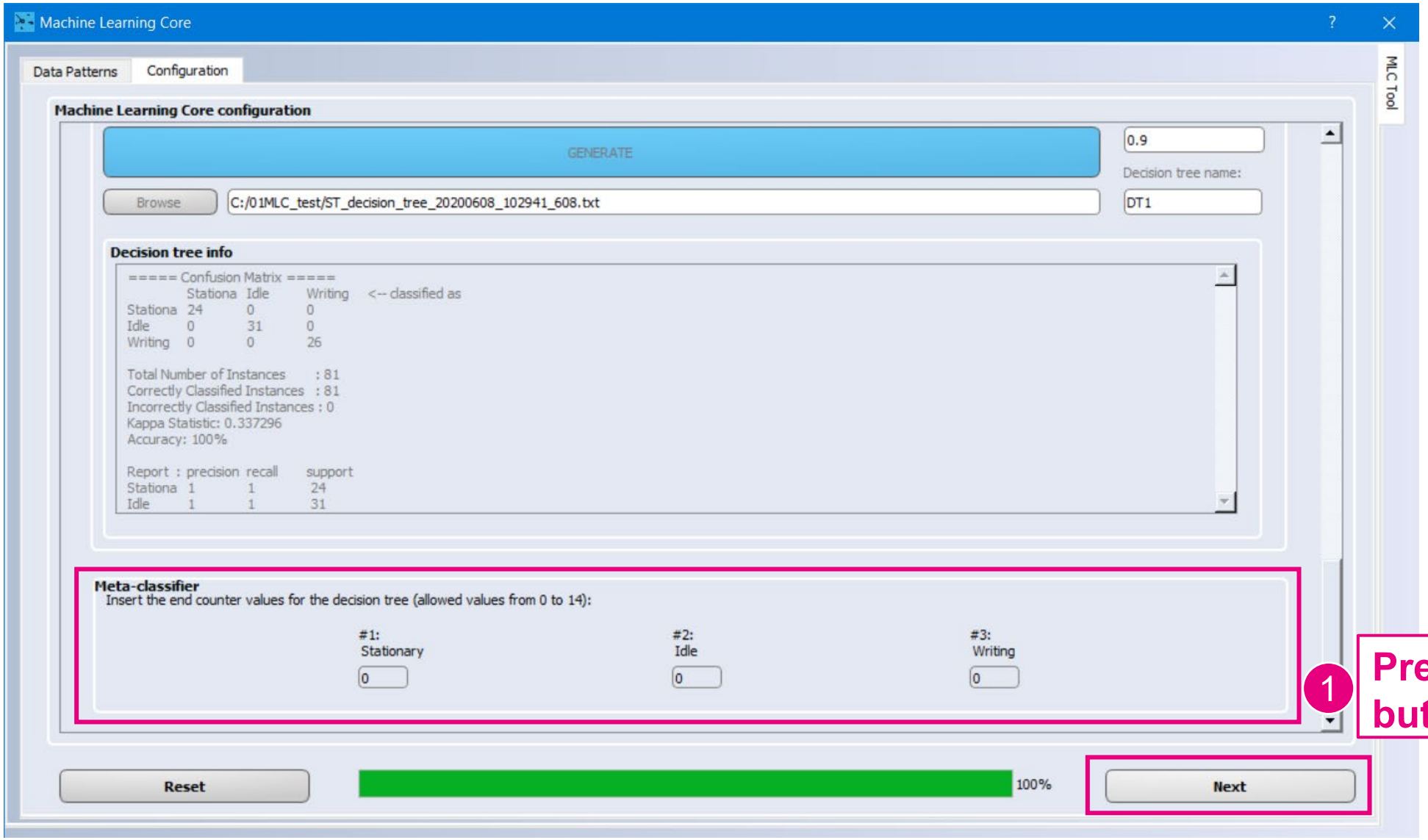
STEP 4: MLC settings and decision tree creation



STEP 4: MLC settings and decision tree creation

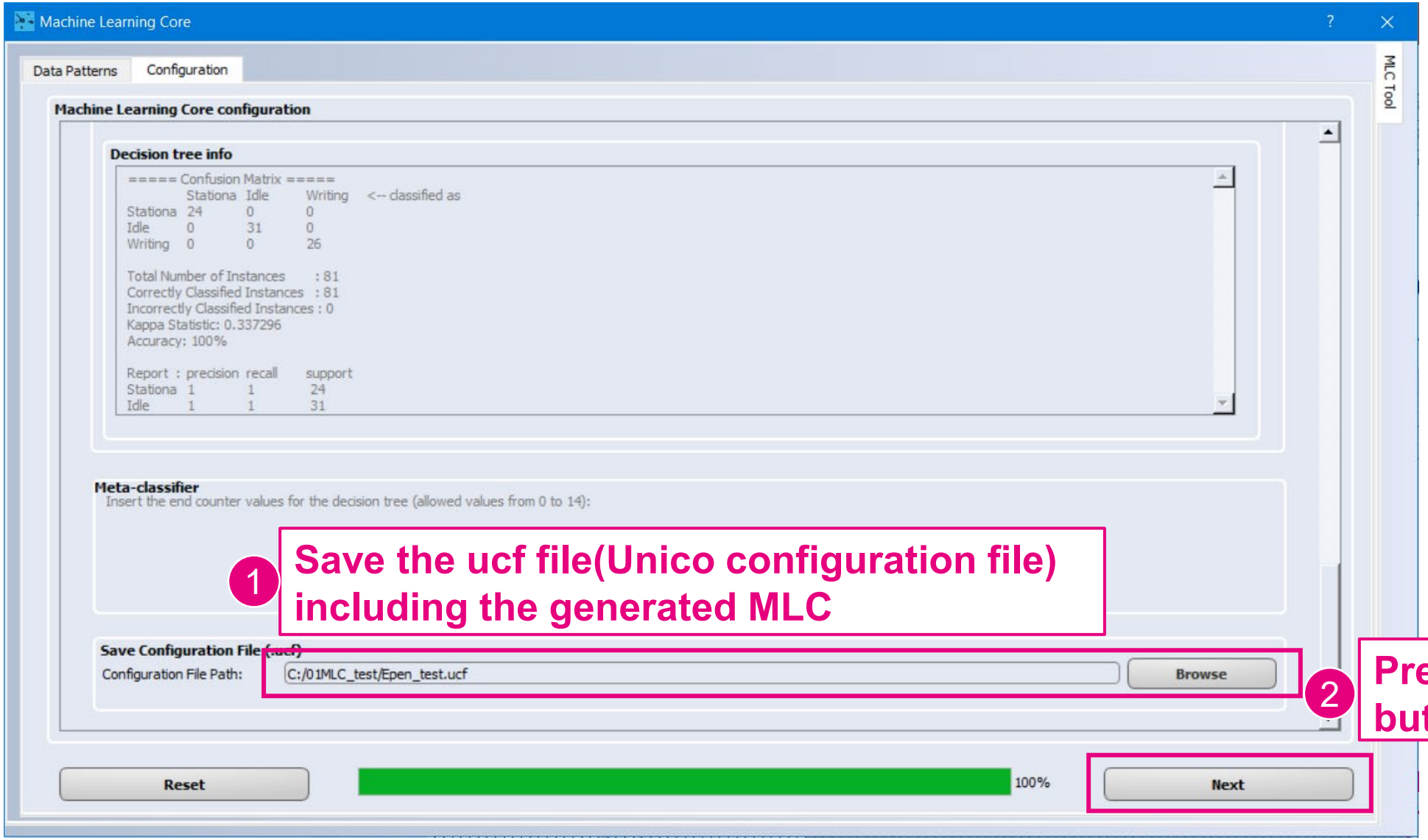


STEP 4: MLC settings and decision tree creation

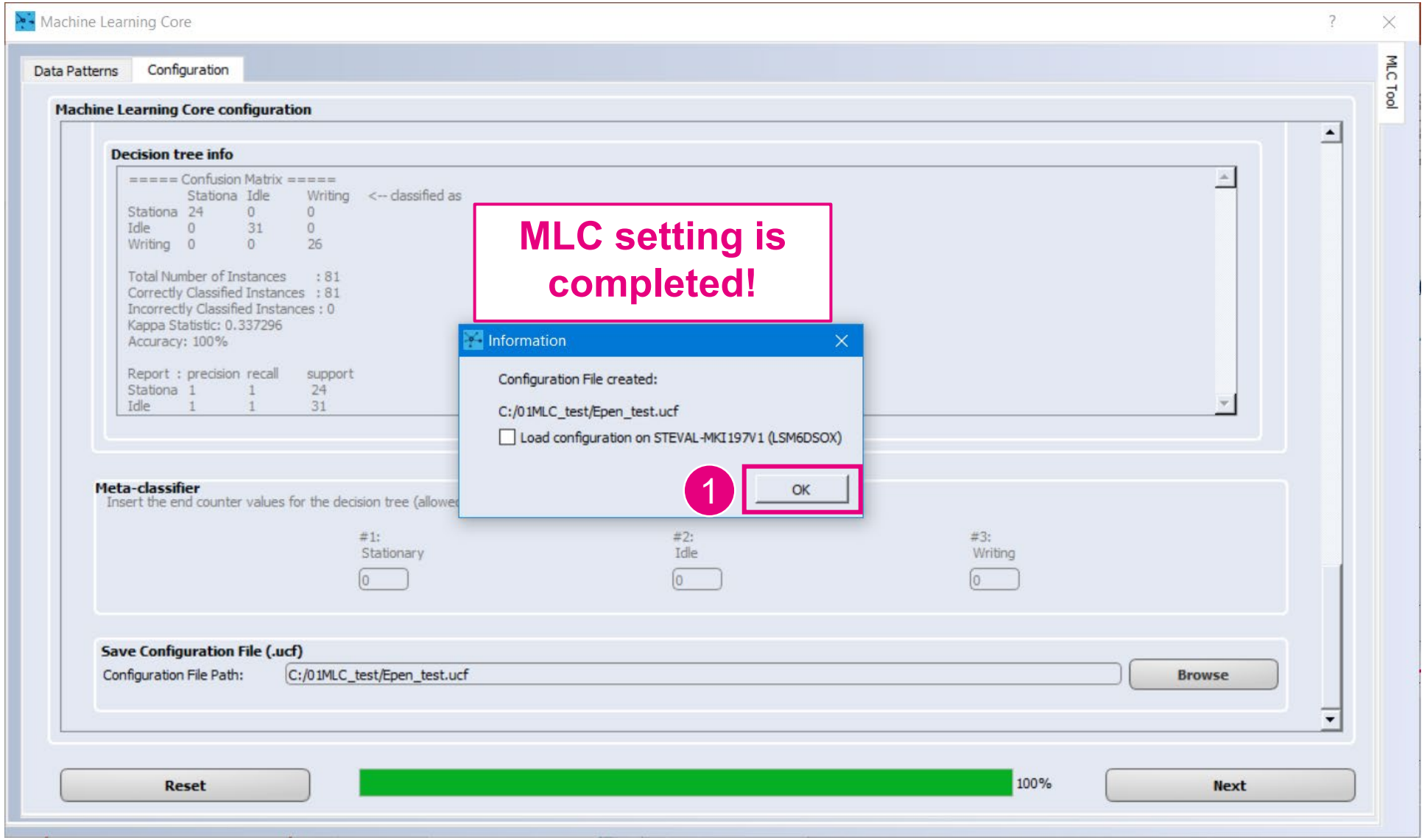


1 Press ‘Next’ button

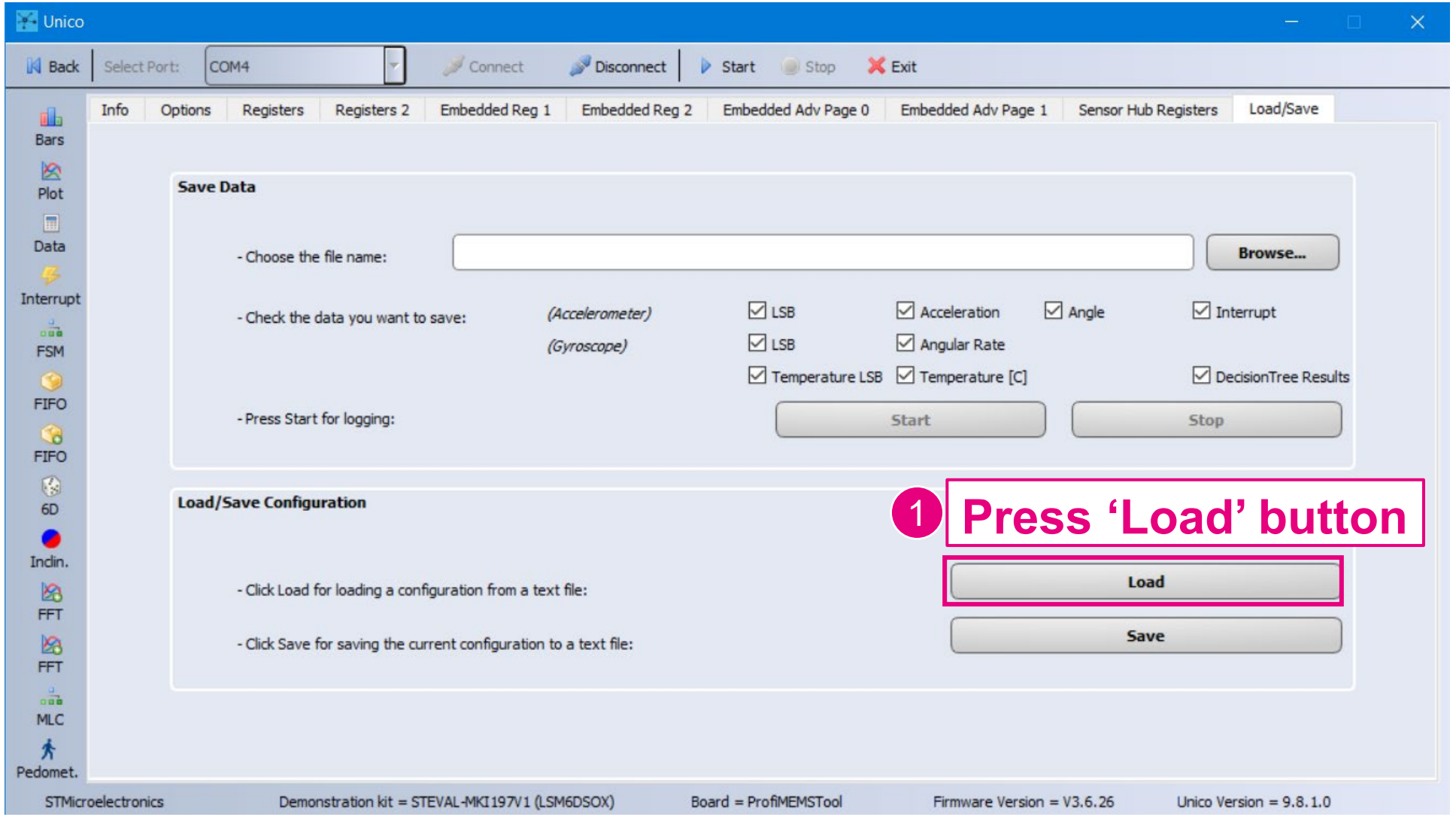
STEP 4: MLC settings and decision tree creation



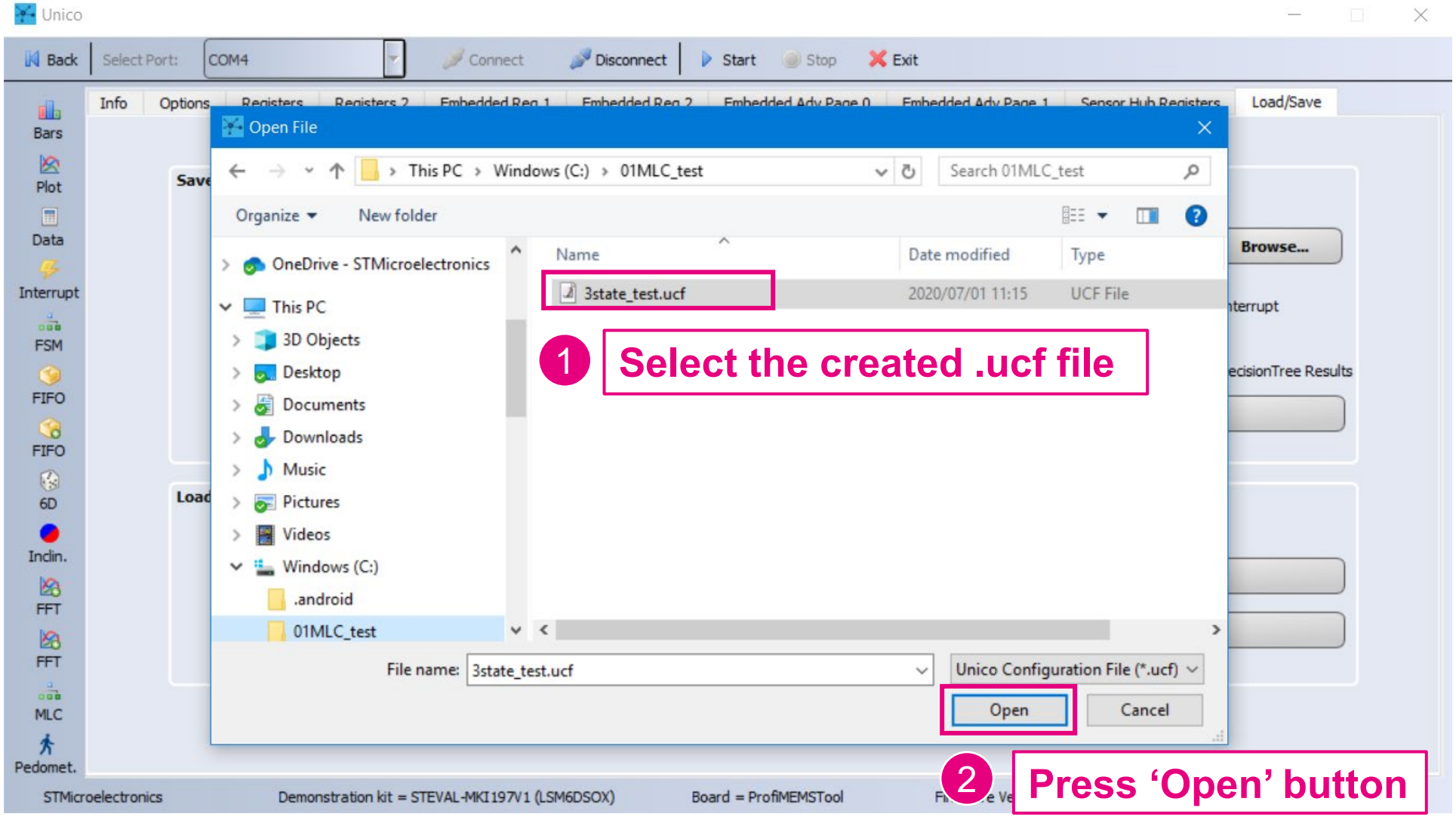
STEP 4: MLC settings and decision tree creation



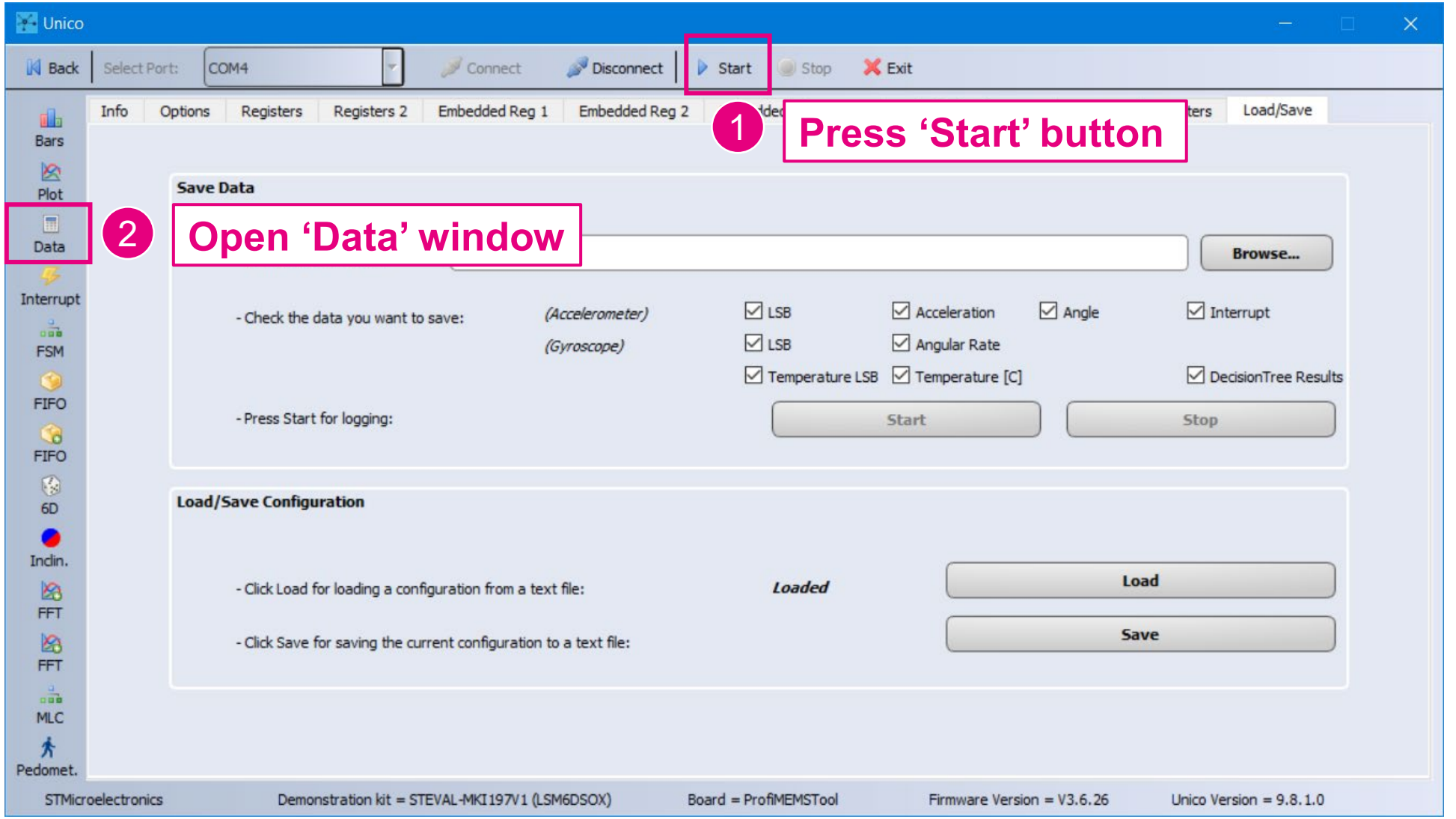
STEP 5: Check the operation of the created MLC



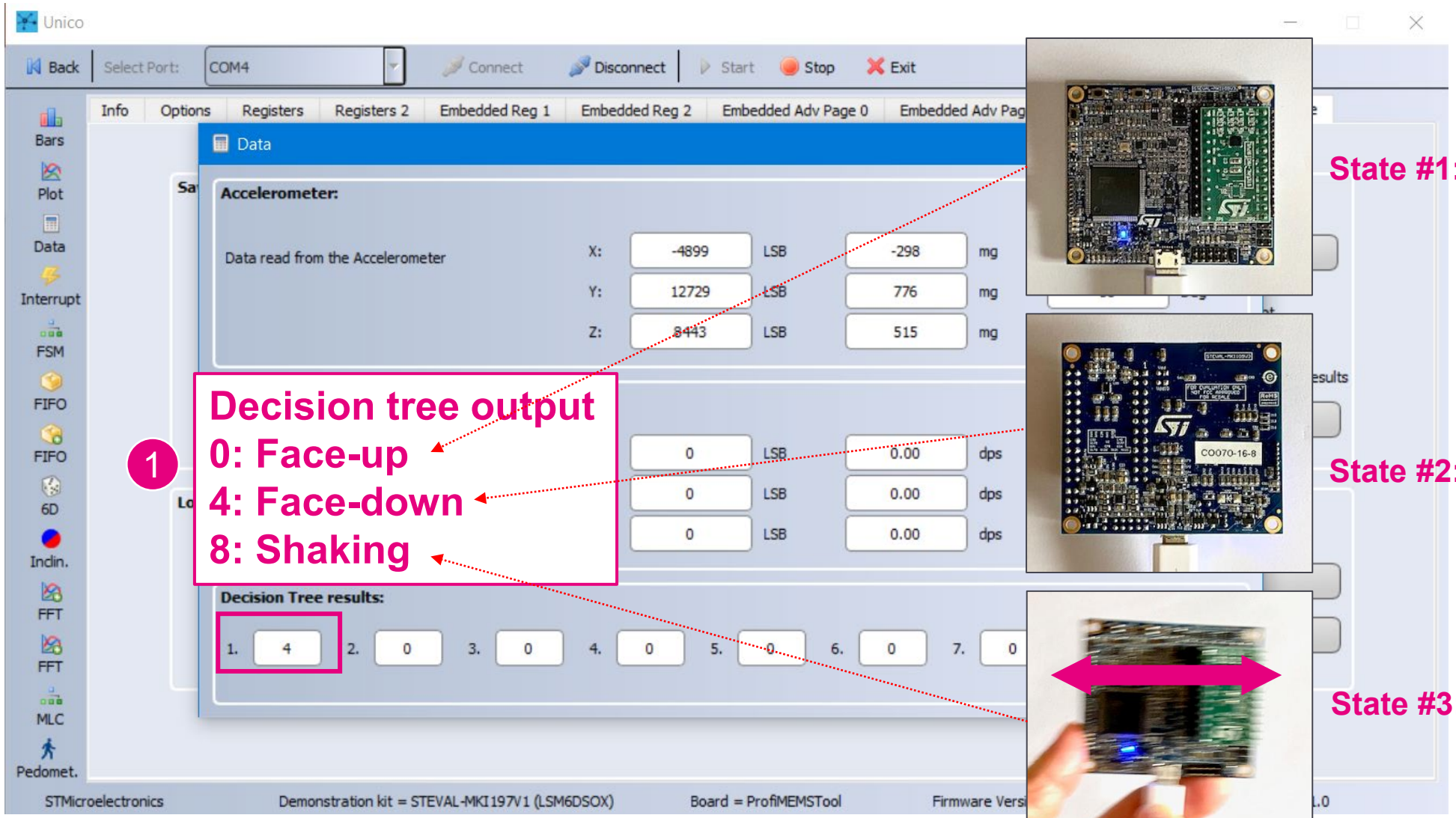
STEP 5: Check the operation of the created MLC



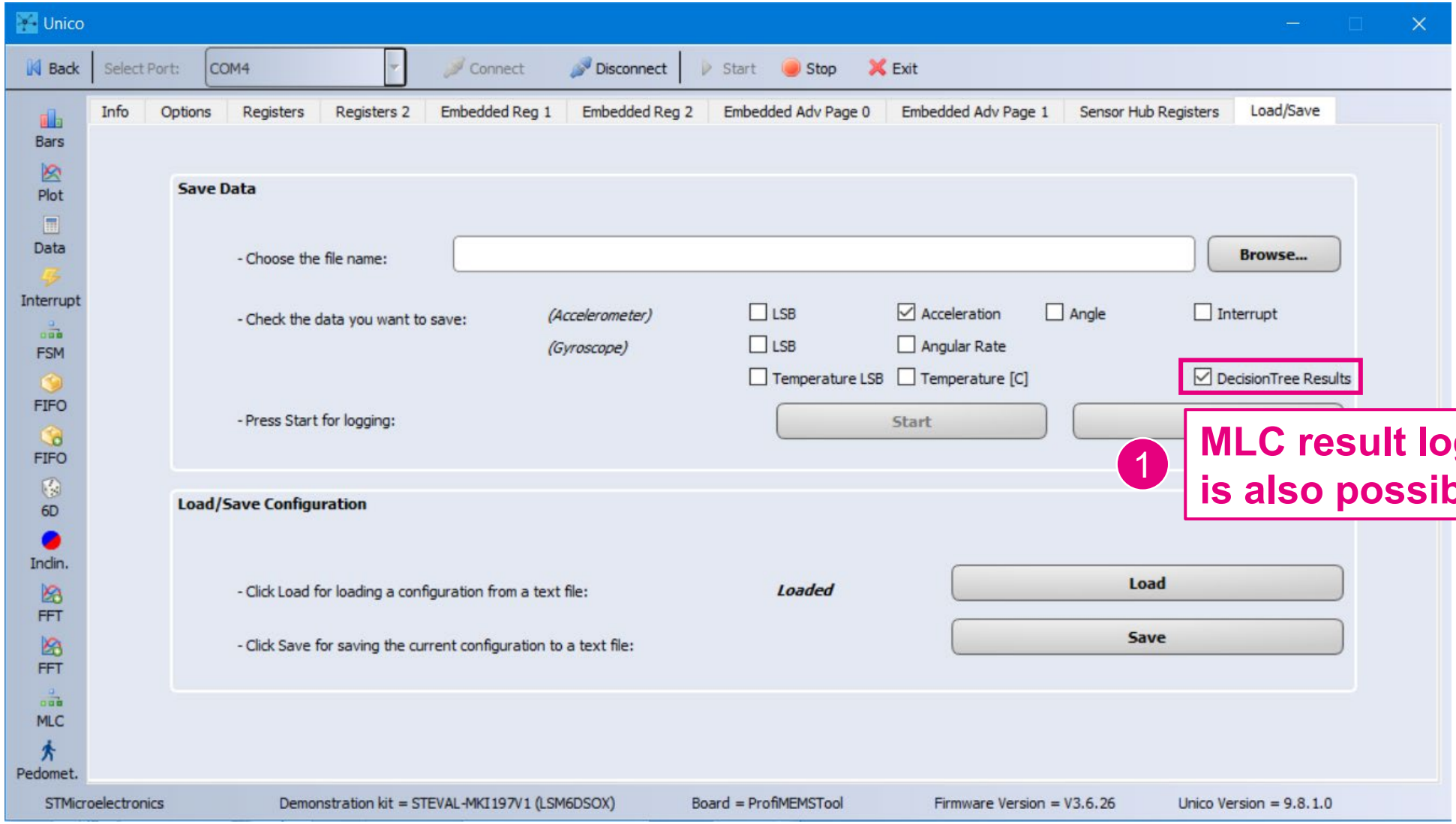
STEP 5: Check the operation of the created MLC



STEP 5: Check the operation of the created MLC



STEP 5: Check the operation of the created MLC



Q&A

life.augmented

Thank you

life.augmented