URAI Autumn School dedicated to time series classification from robotics applications: a tutorial in a trinational context

Thomas Josso-Laurain[1], Maxime Devanne[1], Quentin Boerlen[1], Jonathan Ledy[1], Jean-Philippe Lauffenburger[1], Jonathan Weber[1]

[1] IRIMAS, University of Haute-Alsace, Mulhouse, France
Firstname.lastname@uha.fr

Abstract. URAI Symposium is a trinational research symposium organized by university partners inside the Upper Rhine region. This symposium aims to bring together researchers that are working on and/or with artificial intelligence. For the 5th edition in 2023, this symposium is taking place in Mulhouse, France. It is preceded by a one-day Autumn School. The participants are following a tutorial about how artificial intelligence can be used for time series from a robotic platform. The design of the tutorial is presented as well as the collected feedback. The results show that the tutorial is well tuned for AI beginners with an increase on the technical skills. All the participants were globally happy with the organization of the day.

Keywords: Artificial intelligence; Time series; Robotics; tutorial

1 Introduction

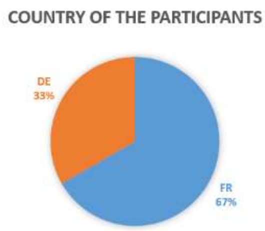
Artificial intelligence has become more and more popular in the research community, in computer science domain but also in every other applicative domain. Particularly, robotics have a growing interest in artificial intelligence algorithms since they are able to analyze the large number of information coming from the heterogeneous sensors.

Since 2019, an annual symposium is held in the Upper Rhine region, gathering researchers from universities from France, Germany and Switzerland. Organized by TriRhenaTech, this symposium aims to bring together scientists working in or with Artificial Intelligence. Every year, an application is targeted by TriRhenaTech and the organizing university to show the interest of AI for applicative research: industries (2020), life science (2021), medicine (2022) and robotics (2023).

This 2023 edition dedicated to AI for time series and robotics, is organized by the University of Haute-Alsace (UHA), France, mainly by the ENSISA engineering school and the IRIMAS institute of research. Taking place on the 17/11/23, an Autumn School is proposed on the 16th.

The Autumn School is a 1-day tutorial for Master and PhD students from all the TriRhenaTech partners. It has received a great interest with 12 registrations coming from people from the Upper Rhine region:

Hochschule Offenburg	2	Karlsruhe Univ. of Applied Sciences	1
French-German Research institute of Saint-Louis	2	Hochschule Furtwangen	1
ICAM Strasbourg	1	Univ. of Haute-Alsace	5



The program of the Autumn School is the following:

Thursday, 16th of November 2023

08:30 – 09:00: welcoming
09:00 – 10:00: AI basics
10:00 – 10:30: presentation to the SMART-UHA platform
10:30 – 12:00: machine learning classifiers
12:00 – 13:30: lunch
13:30 – 15:00: deep learning classifiers
15:00 – 15:30: measure acquisition with the SMART-UHA robot
15:30 – 16:30: classifiers validation

First, we will quickly present the robotics platform on which the Autumn School is based. Then, we will detail the work sessions with the objectives and examples of the code and results. Finally, the collected feedback from the participants is analyzed and presented.

2 SMART-UHA robot

The main application for this day is a robotic system part of the SMART-UHA project. A quick description of the project and the robot is provided to introduce the use-case of the tutorial day.

2.1 SMART-UHA project

The SMART-UHA project is a global project gathering all the research institutes and teaching facilities inside the University of Haute-Alsace (UHA). The IRIMAS institute is the project leader. It receives financial support from M2A, SGARE Grand Est, UHA and the FEDER founds.

This project is divided into two parts: one part dedicated to the electric energy management with solar panels and smart measurement units recording current, voltage, weather conditions and so on; a second part dedicated to the implementation of a robotic platform travelling through the campus in order to deliver letters and packages.

2.2 SMART-UHA robot

The robotic platform coming from the SMART-UHA project is equipped with a large number of sensors in order to ensure the safety of the university users. The environment perception is realized through stereovision cameras, LiDARs sensors and ultrasound sensors.

The localization of the robot remains on the GPS with RTK accuracy (a fixed antenna provides a triangulation within a centimeter precision). This ensures safety since the robot is able to track the desired trajectory with a very high precision.

In case of failure in the GPS cover (close to buildings for instance), the robot can navigate using odometry based on cameras (visual odometry) or the Inertial Measurement Unit (IMU). This IMU records a large variety of measures, including the accelerations and the angular velocities in the three axes of the robot (x pointing to forehead, y to lateral left and z to the sky).

The robot is communicating through Wi-Fi network within the UHA cover. An additional network, based on LoRAWAN technology, is providing a safety signal, sending periodically GPS coordinates on a huge range for small energy.

The robot has two control units: one "high-level" based on a NVIDIA Jetson GPU unit and another one "low-level" based on a EMTRION embedded computer. While the Jetson is processing most of the data coming from the LiDARs and the cameras, the EMTRION is controlling the four wheels with brakes and motors (propulsion and direction). Consequently, an overload of the GPU does not endanger the navigation of the robot which can stop at any moment in case of a failure.

2.3 Robotics use-case

The use-case we consider for this "AI for time series applied to robotics" tutorial is the automatic classification of the nature of the surface on which the SMART-UHA robot is navigating. The nature of the surface is a sensitive criterion when it comes to lateral stability since it highly modifies the adhesive coefficient of the tire-ground contact. This parameter is very hard to measure in real time.
In order to assess the nature of the surface, one can rely on the camera and use AI-based computer vision. However, the position of the camera and the technology make this sensor particularly sensitive to noisy environmental conditions (rain, fog, etc.) Another idea is to consider the IMU which is more trustable since it is stable and safely installed inside the robot body. The signals coming from the IMU, i.e. accelerations, are then considered as time series.
Therefore, the use-case for the tutorial day is to classify the nature of the surface between two classes: "concrete" and "grass". First approaches will consider univariable algorithms based on the x-acceleration. More approaches are then proposed along the day, considering multivariable algorithms with acceleration on x and z axes.

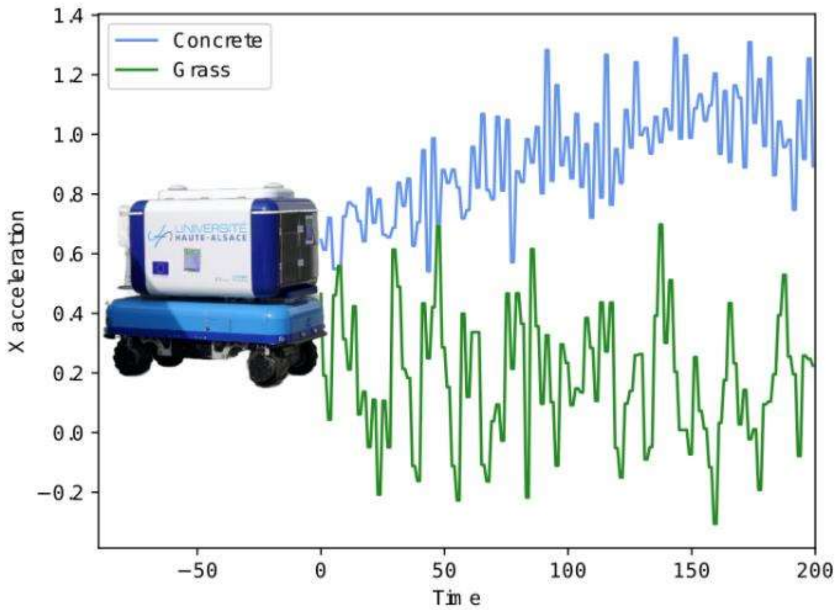


Fig. 1. Use-case: x-acceleration analysis in order to classify the nature of the ground (grass/concrete)

3 Tutorial details

The tutorial is based on Google colab codes completed with full comments, texts and videos to illustrate particular elements such as data normalization of k-nn algorithm.
The tutorial is divided into 5 sessions, 3 in the morning and 2 in the afternoon. For each session, the technical objectives are presented as well as some examples of the code and the results.

3.1 Basics in AI

The first session is dedicated to basics in AI since the level of knowledge of the participants is unknown.

Objectives:

- Using data normalization (z-norm)
- Importing Python libraries
- Visualizing time series
- Computing Euclidean distance between time series (non-normalized and normalized)
- Splitting a dataset
- Understanding Cross-validation

First, a synthetic dataset is considered with two classes: rectangles and triangles. Below is a display of the raw dataset proposed to the participants:

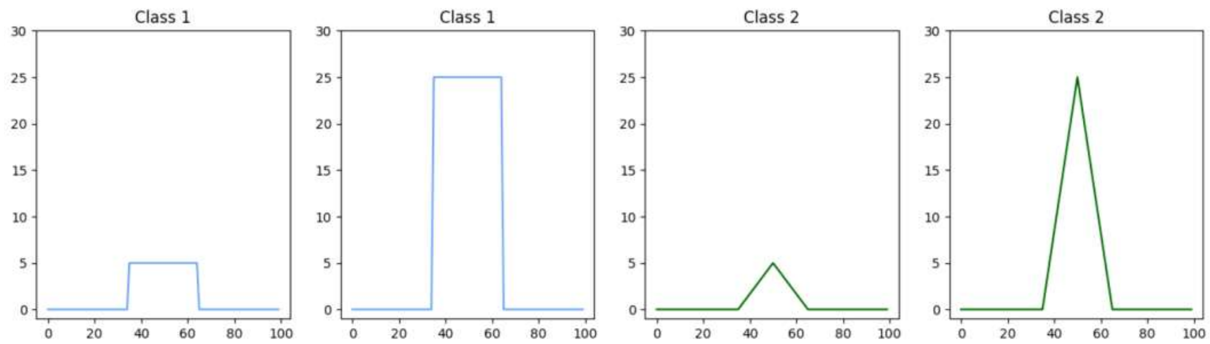


Fig. 2. Synthetic dataset

Computing the Euclidean distance, we can see that the first time series are closer to the small triangle of Class 2. A z-normalization is then applied on the dataset. The Euclidean distance is computed to show that the first time series belongs to Class 1.

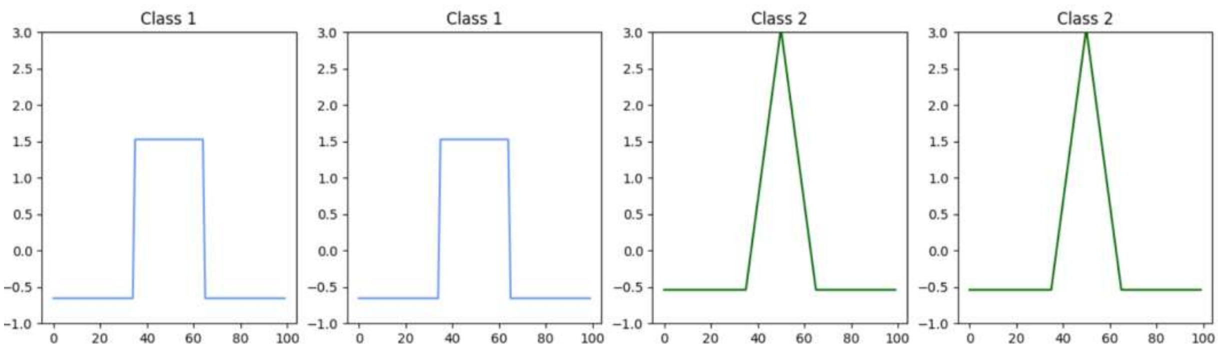


Fig. 3. Normalized synthetic dataset

The second part of this first session of the morning is dedicated to the study of the SMART-UHA dataset including x-acceleration raw data from the robot, with a split between train data and validation data.

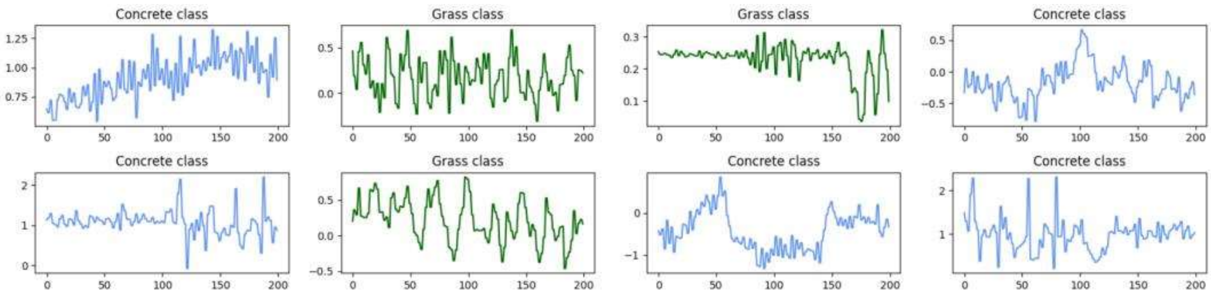


Fig. 4. Examples of SMART-UHA dataset (raw data + label)

3.2 Machine Learning classifiers – part 1

The next session of the morning is dedicated to Machine Learning classifiers.
Objectives:

- Applying k-NN (Nearest Neighbor) algorithms with Euclidean distance
- Reading a Confusion Matrix
- Computing the Dynamic Time Warping (DTW) distance

The Colab tutorial is composed on commented code but also schematic videos uploaded on Youtube to get the main idea of a concept. Below is an example with the 1-NN algorithm:

Fig. 5. Example of the 1-NN video on Youtube (@Maxime Devanne)

The participants have the choice to either compute the commented code that is already written, or to write again the code. In some parts, they have to write from scratch: for instance, the 1-NN algorithm is already written, but not the 3-NN they have to test further.

```
[7] # counter of good classifications initialized at 0
    good_classification = 0
    # a for loop over all predictions
    for i in range(y_pred.shape[0]):
      # For each prediction, if the predicted label corresponds to the true one, we increment the counter by 1
      if y_pred[i] == y_test[i]:
        good_classification = good_classification + 1

    # The rate of good classifications (accuracy) corresponds to the number of good classification divided by the total number of predictions
    accuracy = good_classification / y_pred.shape[0]
    # printing the classification accuracy in percentage
    print('Percentage of good classifications:', accuracy*100)

    Percentage of good classifications: 50.0
```

Fig. 6. Example of the code provided on Google Colab – here the 1-NN prediction result

At the end, they reach a prediction based on the 3-NN algorithm with Dynamic Time Warping distance. The prediction can be assessed using a Confusion Matrix stating the probability for each class to be detected as it is.

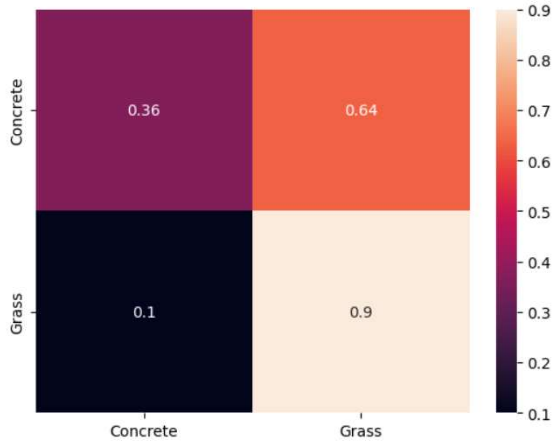


Fig. 7. Final Confusion Matrix obtained for a 3-NN algorithm with DTW distance

3.3 Machine Learning classifiers – part 2

After investigating some simple algorithms based on k-NN, the participants are considering linear classifiers.

Objectives:

- Using Softmax function to get output probabilities
- Computing a loss function based on binary cross entropy
- Optimizing the training of the model by gradient descent
- Applying linear classifiers to make predictions

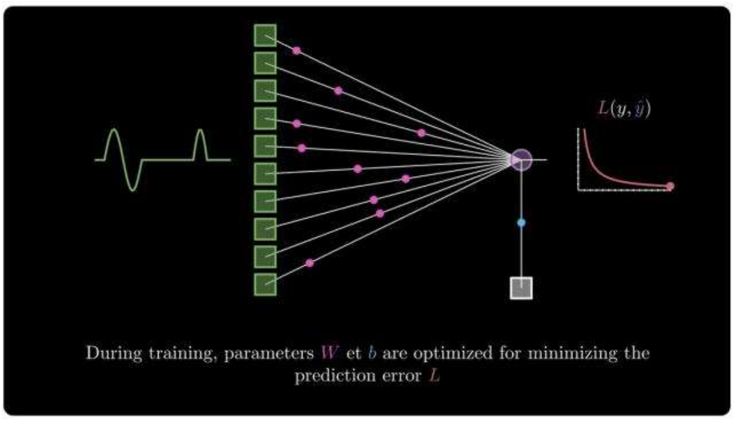


Fig. 8. Example of the video about linear classifier training (Maxime Devanne – Youtube)

```
[6]  # initializing an array for storing Softmax values
     softmax_outputs_serie0 = np.zeros(outputs_serie0.shape)
     # a for loop over the scores
     for i in range(outputs_serie0.shape[0]):
       # applying the Softmax function
       softmax_outputs_serie0[i] = np.exp(outputs_serie0[i]) / np.sum(np.exp(outputs_serie0))

     # printing the Softmax output (probabilities)
     print(softmax_outputs_serie0)

     [0.13192183 0.86807817]
```

Fig. 9. Example of the commented code of the Softmax function

The goal of the morning is to achieve classification on the SMART-UHA dataset using the Machine Learning technique of linear classifiers. The weights of the linear classifier are computed by optimizing a loss function based on the binary cross entropy and the gradient descent. The cross entropy presented in a text element of the Google Colab such as:

In the case of a classification, rather than a simple difference, we use cross entropy which is more suitable. In the case of binary classification (as in our case), the binary cross entropy is defined as follows:

$L = -y * log(\hat{y}) + (1-y) * log(1-\hat{y})$,

with y being the true label and $\hat{y}$ is the predicted label.

Fig. 10. Cross entropy explanation in the Google Colab document

The participants learn then the importance of the weights initialization. A 10-fold prediction provides an average performance of the linear classifier on the SMART-UHA dataset:

```
Accuracy of 10 runs: [0.21875 0.34375 0.25    0.3125  0.25    0.34375 0.34375 0.28125 0.21875
 0.3125 ]
Average accuracy: 0.2875+-0.0480071609241788
```

Fig. 11. Linear classifier accuracy on classifying concrete and grass on the SMART-UHA dataset

It is not very performant! The use-case that has been designed makes simple machine learning algorithms fail so there is a need to investigate deep learning classifiers.

3.4 Deep Learning classifiers

The afternoon is dedicated to deep learning classifiers, particularly to the family of Convolutional Neural Networks (CNN)

Objectives:

- Understanding the process of a convolution
- Building a CNN architecture with layers of convolution, ReLU and Global Average Pooling 1D using the Keras framework
- Training a CNN model using Adam optimizer
- Reading a Class Activation Map (CAM)
- Improving the performance with multivariable CNN

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d (Conv1D)             (None, 200, 32)           192

 global_average_pooling1d (  (None, 32)                0
 GlobalAveragePooling1D)

 dense (Dense)               (None, 2)                 66

=================================================================
Total params: 258 (1.01 KB)
Trainable params: 258 (1.01 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Fig. 12. Simple CNN model built by the participants

```
[12] keras.utils.plot_model(simple_cnn_model, show_shapes=True)
```

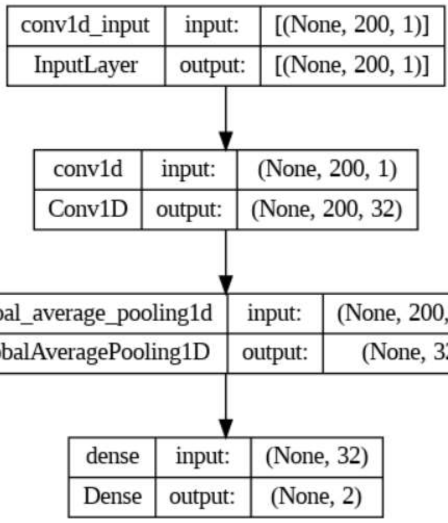


Fig. 13. Display of the simple CNN model using Keras

Once the CNN model has been built, the participants are training it with the SMART-UHA dataset:

```
4/4 [==============================] - 0s 7ms/step - loss: 0.4922 - accuracy: 0.7619
Epoch 997/1000
4/4 [==============================] - 0s 6ms/step - loss: 0.4922 - accuracy: 0.7619
Epoch 998/1000
4/4 [==============================] - 0s 6ms/step - loss: 0.4916 - accuracy: 0.7778
Epoch 999/1000
4/4 [==============================] - 0s 5ms/step - loss: 0.4916 - accuracy: 0.7778
Epoch 1000/1000
4/4 [==============================] - 0s 5ms/step - loss: 0.4914 - accuracy: 0.7619
Training time (s): 29.6695294380188
```

Fig. 14. Training the simple CNN model over 1000 epochs

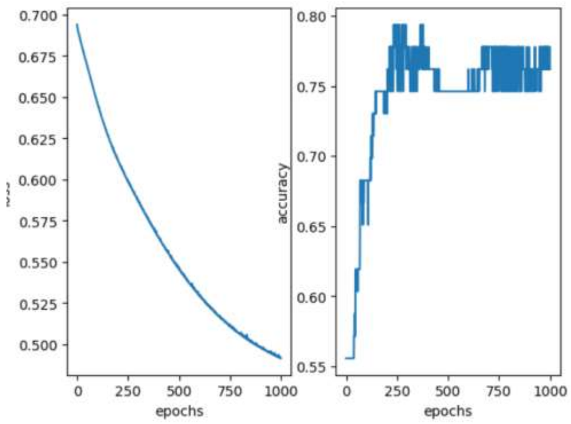


Fig. 15. Loss and accuracy through the training process

The trained CNN model is then used to predict over the test part of the SMART-UHA dataset. The results are (hopefully) better than machine learning algorithms:

```
Accuracy on the test set: 78.125
```

Fig. 16. Result of the 1-Layer CNN prediction

The participants are then asked to build more complex CNNs with more layers. In addition, the hot topic of explainability is introduced. Right now, explaining the results of AI algorithms has received a growing interest from the research community.
First, the activated features are studied layer-by-layer:

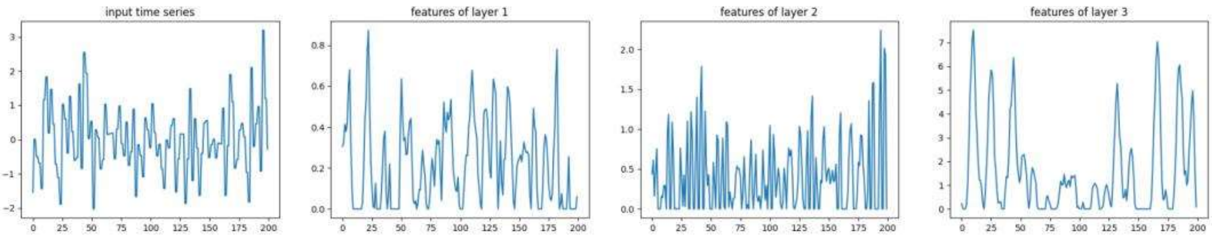


Fig. 17. Features of each layer of a 3-layers CNN

And then by the Class Activation Map:

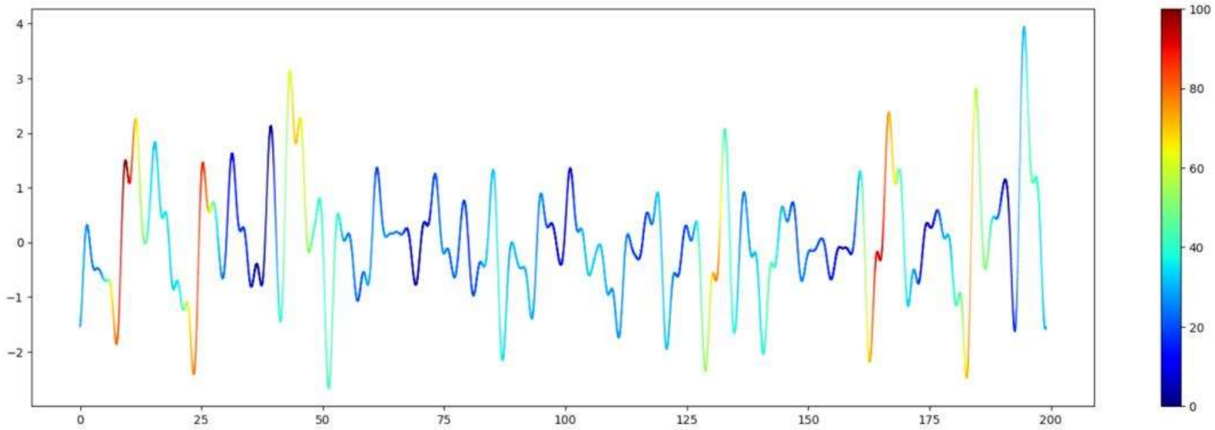


Fig. 18. Class Activation Map of the 3-layers CNN

where the red parts corresponding to the most relevant features for the prediction.
The next step for the participants is to improve the performance of the prediction by investigating multivariable neural networks. To reach that, not only the acceleration on the x-axis is considered but also on the z-one (pointing the sky). After the training, we can see that the multivariable CNN model performs better than the univariable one:

```
loss: 0.4774 - accuracy: 0.9062
```

Fig. 19. Accuracy and loss of the multivariable (x,z) 3-layer CNN model

3.5 Validation on a real-field complex dataset

The final stage of the afternoon is to evaluate the performance of the previously designed univariable neural networks over a complex dataset recorded during a real-field use; therefore, it alternates between the grass sections and the concrete sections, changing at different and variable times.
Objectives:

- Training and evaluating a CNN over long complex time series

The considered model is a univariable (x-acceleration) 3-layers CNN. It is trained over 1000 epochs with a learning rate of 0.001.

Below are the ground truth (first plot) and the prediction classes (second plot) where green is grass and blue is concrete:

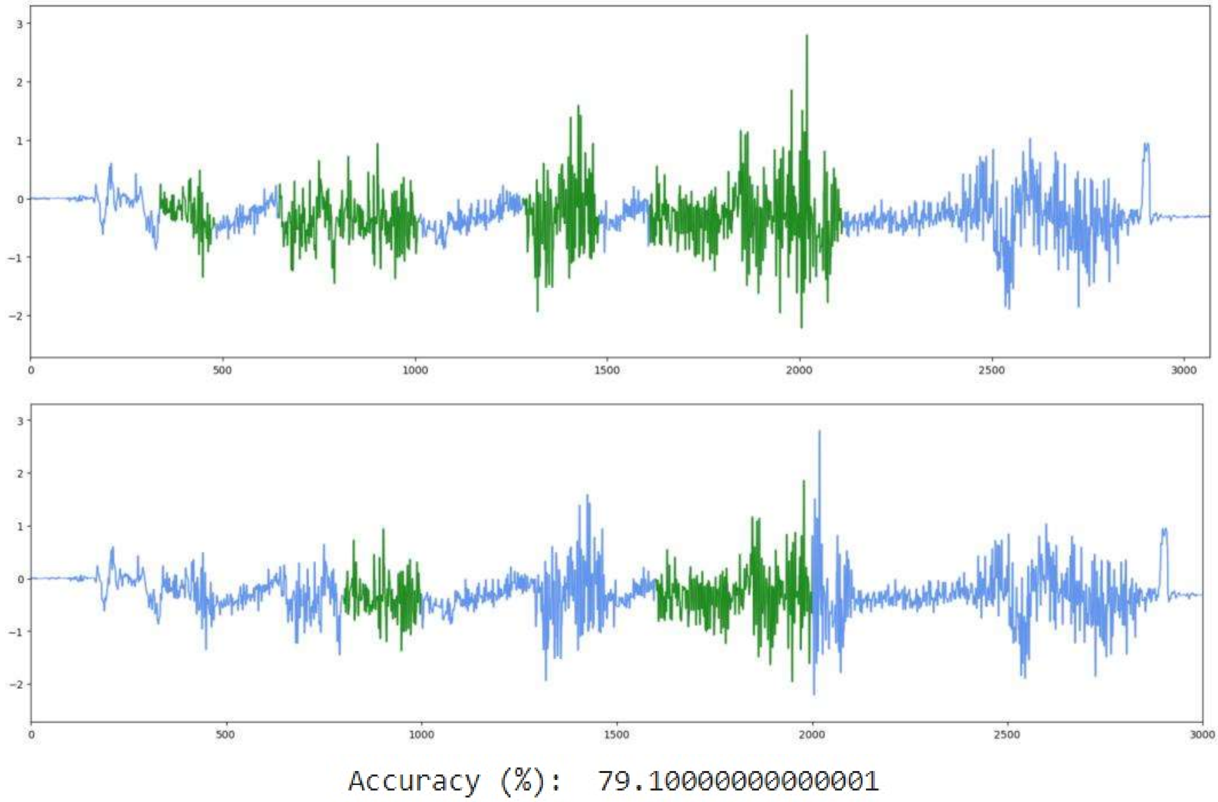


Fig. 20. SMART-UHA real-case validation sequence (up) and 3-layers CNN prediction results (bottom)

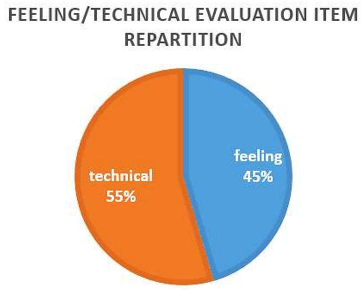
4 Participants feedback

4.1 Feedback evaluation protocol

As presented in the previous section, the tutorial content has been designed to address the defined technical objectives. Moreover, it can be interesting to also evaluate the feeling of the participants regarding the organization of the day, from a global point of view (lunch, coffee breaks…) to a scientific content point of view (speed of the tutorial, etc.).
At the end of the tutorial day, a form has been sent to the participants to evaluate if the objectives have been reached. The form is anonymous to ensure liberty of feedback. The survey is composed on the following questions:

Questions	**Assessed index**	**Type**
1 Did you enjoy?	Feeling	1-5 scale
2 How was the density of the knowledge?	Feeling/Technical	1-5 scale
3 How was the speed of the tutorial?	Feeling/Technical	1-5 scale
4-1 Level of skill in AI before the tutorial	Technical	1-10 scale
4-2 Level of skill in AI after the tutorial	Technical	1-10 scale
5 Reusability of what you have learnt	Technical	1-3 scale
6 What was the best moment of the day?	Feeling	Multiple Choice
7 Any free comments?	Feeling/Technical	Free comments

The assessed items cover an equal part of feeling and technical as we can see:



4.2 Feedback analysis

Over the 12 participants, 6 accepted to answer to the feedback form.
First of all, the participants were globally happy and enjoyed very much the tutorial day (Q1).

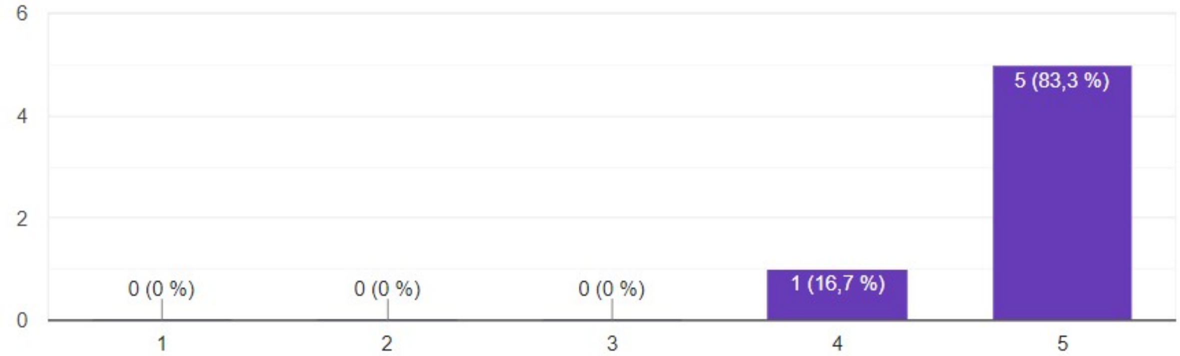


Fig. 21. Happiness assessment results with subjective scale (1 : not happy)

Regarding the tutorial organization, we assessed both the density of the knowledge presented in the tutorial supports, and the speed of the training for the whole day. For the density, we designed a Gaussian-like scale, with 1 = no dense at all, and 5 = way too dense. This means that 3 is a good score. The same scale is used for the speed, with 1 = too slow and 5 = too fast. The results are exactly the same for the two assessments (Q2 and Q3)

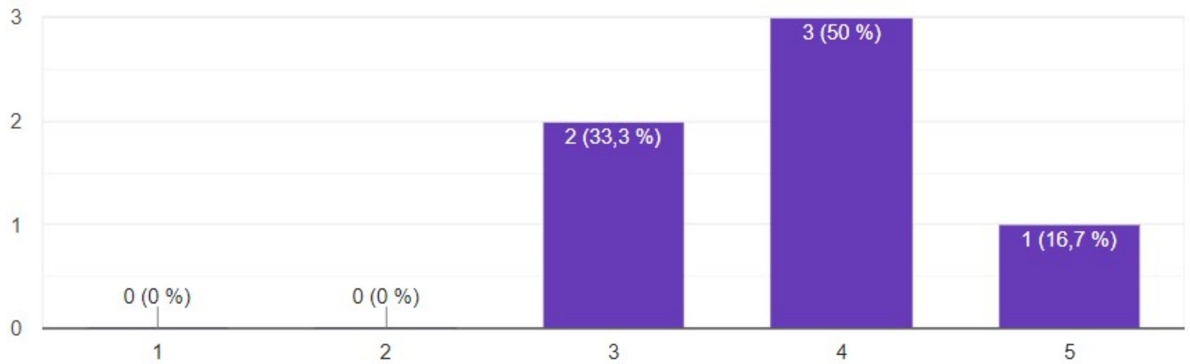


Fig. 22. Density and speed assessment results with Gaussian-like scale (1 = no dense ; 5 = too dense)

The next two questions Q4-1 and Q4-2 allow us to evaluate the technical skills acquired during the day since we asked the participants to assess their mastery of AI before and after the day.

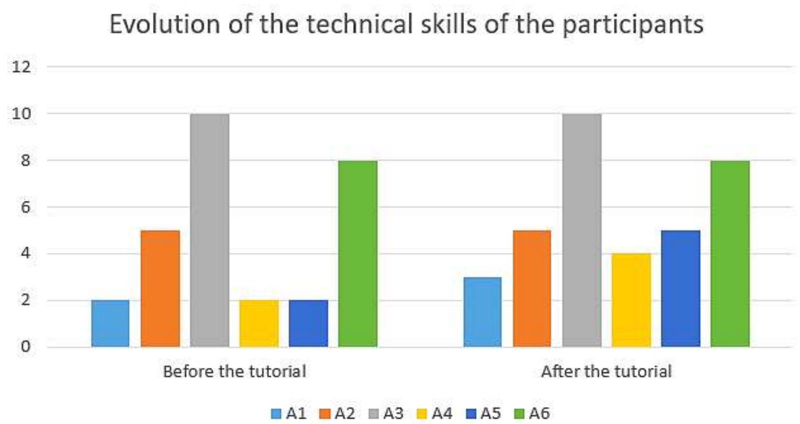


Fig. 23. Density and speed assessment results with Gaussian-like scale (1 = no dense ; 5 = too dense)

The results show different interesting things: first of all, the heterogeneous caracteristic of the audience. We can distinguish three categories of participants: those who do not know AI (starting around 2/10), those who have knowledge about it (around 5/10) and those who already master AI (over 8/10)
For the AI masters, there is no difference between and after since they are daily working with AI. They did not criticize the rhythm of the day, so we can conclude that they were not the targeted audience for a tutorial.
For the AI knowers, we notice no difference between before and after, meaning that they did not improve their technical skills. However, they still enjoyed the day and none of them complained about being bored or the tutorial being too slow. We can conclude that for this category, they enjoyed the tutorial by consolidating their knowledge and discovering the robotics application.
For the AI beginners, the difference is very important, with an average gain of technical skills of +2/10. That means that the tutorial is well designed since it starts with AI basics and goes up the to deep learning. For AI beginners, it is a very good introduction to the world of classification.

Another point we wanted to assess is the reusability of the acquired technical skills in the participants research topics.

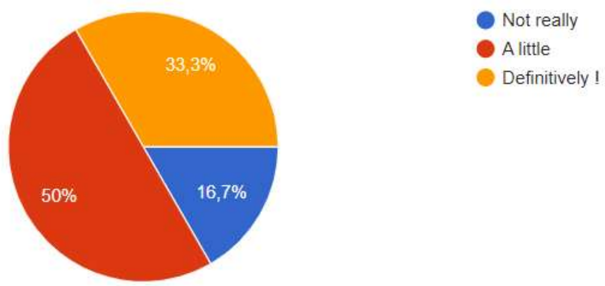


Fig. 24. Reusability results

We can see on the figure that the audience was composed of persons in need for AI as well as curious people. The limits of this one-question result is that is to not possible to determine if the Definitively answers are because the participants already work with AI or if they clearly learnt how to use it within their own research.
The last question results illustrate that the participants liked learning autonomously with the tutorial support as well as discovering the robotics platform of the lab.
The free feedback section contains elements that highlight the interest of the participants in such kind of tutorials, particularly in the Upper Rhine region. This is a good hint for the future!

"Thank you for today's tutorial. It is very good for AI beginners. I hope there will be similar workshops or schools at the next URAI Symposium where people can share knowledge about their research areas."

5 Conclusion

In conclusion, the URAI Autumn School was designed to propose a tutorial based on artificial intelligence for time series applied to robotics. We received 12 participants coming from France and Germany.
The tutorial covered a large panel of AI tools, starting from the basics as data normalization until Convolutional Neural Networks. The algorithms were applied on synthetic data as well as real-field data coming from the SMART-UHA robotic platform (raw data of x-acceleration from the Inertial Measurement Unit)
The feedback of the participants was collected with a form. The results analysis show that the tutorial is well tuned for AI beginners. Participants that already quite master AI are not feeling bored over the day since they are able to apply the tools, they know for time series in robotics. The rhythm allows the participants to work autonomously, alternating with coffee breaks and platform presentation.
For the future, the level of the participants should be assessed during the registration in order to create different groups in function of their mastery level. The tutorial experience should be definitively reconduct in the next URAI Symposiums as highlighted by the participants.

6 Acknowledgment

The authors would like to thank the University of Haute-Alsace, the Ecole Nationale Supérieure d'Ingénieurs Sud-Alsace (ENSISA) for hosting the event, AlsaceTech and TriRhenaTech for the organization of the symposium, particularly Katrin Wenzel, Florent Vallier and Anna Dister.
Finally, we would like to thank our nice participants for attending: Javidan Abdullayev, Mohamed Arebi, Augustin Borne, Prabin Dahal, Gandorj Darambazar, Rauf Fatali, Ali Ismail-Fawaz, Anis Koliai, Gauthier Miguet, Saptadi Nugroho, Alain Uwadukunze, Viktor Walter.

Tri Rhena Tech

ensisa
École nationale supérieure d'ingénieurs sud Alsace

IRIMAS