MathWorks®

Japan Advanced Institute of Science and Technology

Navigating the AI Landscape with MATLAB

HIROTA Koji - khirota@mathworks.com
NGUYEN Dinh Mau - maudn@jaist.ac.jp
2023, December 15th

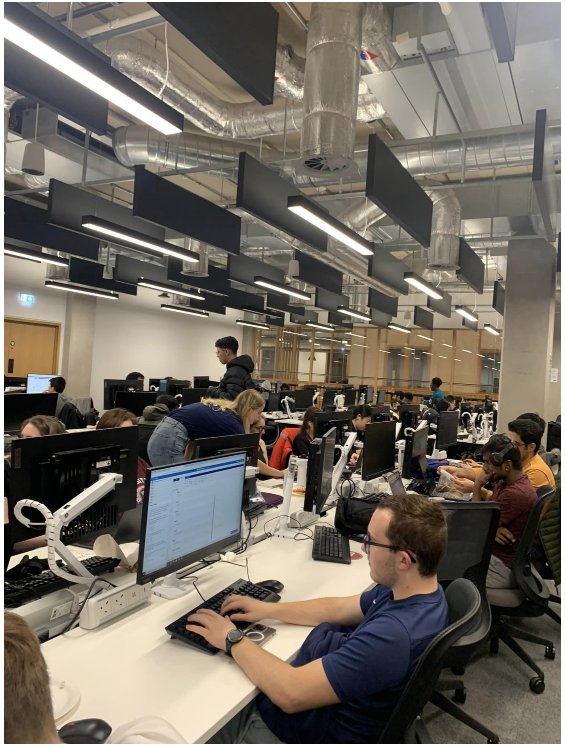
1. INTRODUCTION

- Koji HIROTA
  - Customer Success Engineer at MathWorks
  - Phd in Bioengineering and Biomedical Engineering at University of California, Riverside

- Mau NGUYEN
  - MathWorks Student Ambassador at JAIST
  - M2 in HA3CI Lab at School of Information Science, JAIST

MATLAB Communities around the World

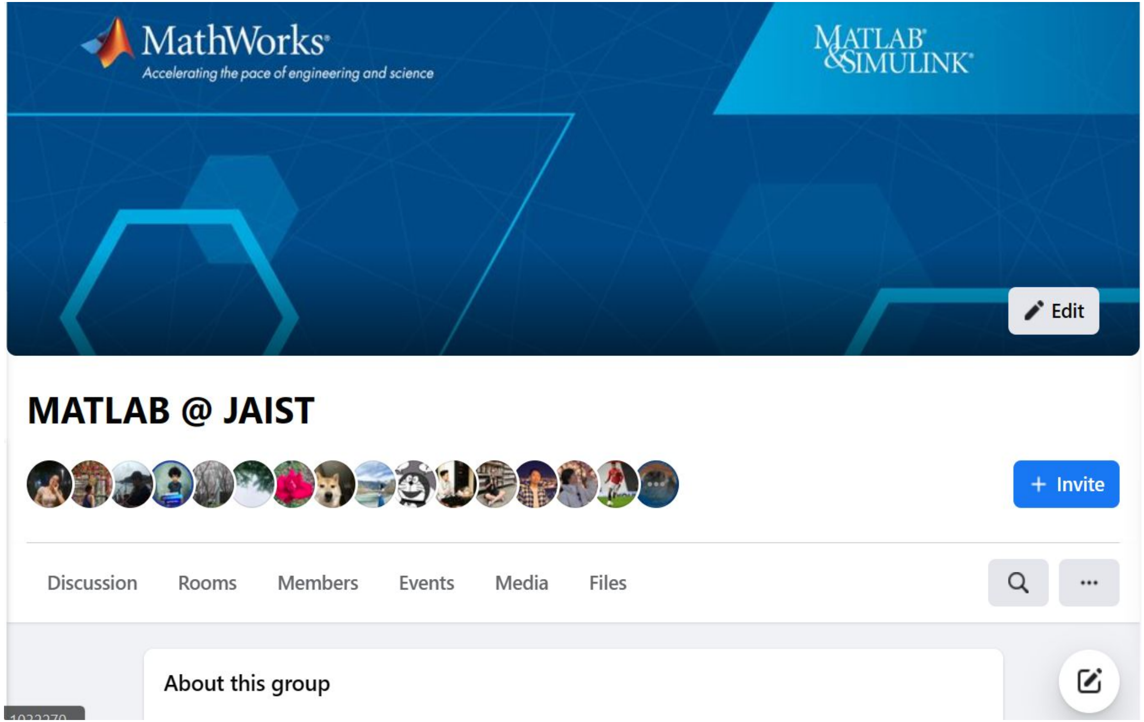
MATLAB Communities around the World



MATLAB Community in Japan

- JAIST
- NAIST
- Keio University
- Kyushu University
- Waseda University
- University of Tokyo
- University of Tsukuba
- Tokyo Denki University
- University of Electro-Communications

MATLAB Community at JAIST



Objectives

- Have a deeper understanding of what we can do with MATLAB
- Especially in Artificial Intelligence

*Beside is the link to download ebook "**Machine Learning with MATLAB**":

Seminar structure

1. Introduction
2. MATLAB essentials for AI
3. Hands-on Machine Learning with MATLAB
4. Deep Learning Capabilities in MATLAB
5. Data Handling and Visualization
6. Q&A

2. MATLAB Essentials for AI

- Data-driven AI
    - At the center of most artificial intelligence applications is data. Taking raw data and making it useful for an accurate and meaningful model will most likely represent significant time spent on your AI effort. Data preparation requires domain expertise to understand the data's critical features, which ones are unimportant, and what rare events to consider.
- AI Modeling
    - Choose a set of algorithms: Are you looking at machine learning or deep learning? Starting with a complete set of algorithms and pre-built models means you are already ahead of the game by taking advantage of the broader work in the artificial intelligence community and not starting from scratch.
- Simulation in AI Design
    - AI models typically exist within large, complex systems. For example, in automated driving systems, artificial intelligence for perception must integrate with algorithms for localization and path planning and controls for braking, acceleration, and other components. These pieces work together to create a complete system. Complex, AI-driven systems like these require integration and simulation.

AI things that you can do (or BMW) with MATLAB

Detecting Oversteering in BMW Automobiles with Machine Learning

Oversteering is an unsafe condition in which a vehicle's rear tires lose their grip while navigating a turn (Figure 1). It can be caused by worn tires, slippery road conditions, taking a turn too fast, braking abruptly while turning, or a combination of these factors.

AI things that you can do (or BMW) with MATLAB

Collecting Data and Extract Features

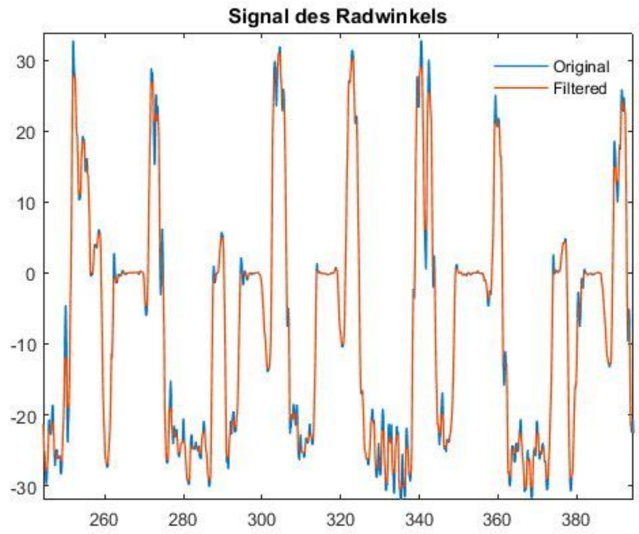
“During the tests, we captured signals commonly used in oversteer detection algorithms: the vehicle’s forward acceleration, lateral acceleration, steering angle, and yaw rate.”

AI things that you can do (or BMW) with MATLAB

Data Cleaning/Filtering

The results produced by models trained on this raw data were not outstanding—the accuracy was between 75% and 80%. To achieve more accurate results, we cleaned and reduced the raw data. First, we applied filters to reduce noise on the signal data.

Next, we used peak analysis to identify the peaks (local maxima) on our filtered input signals.



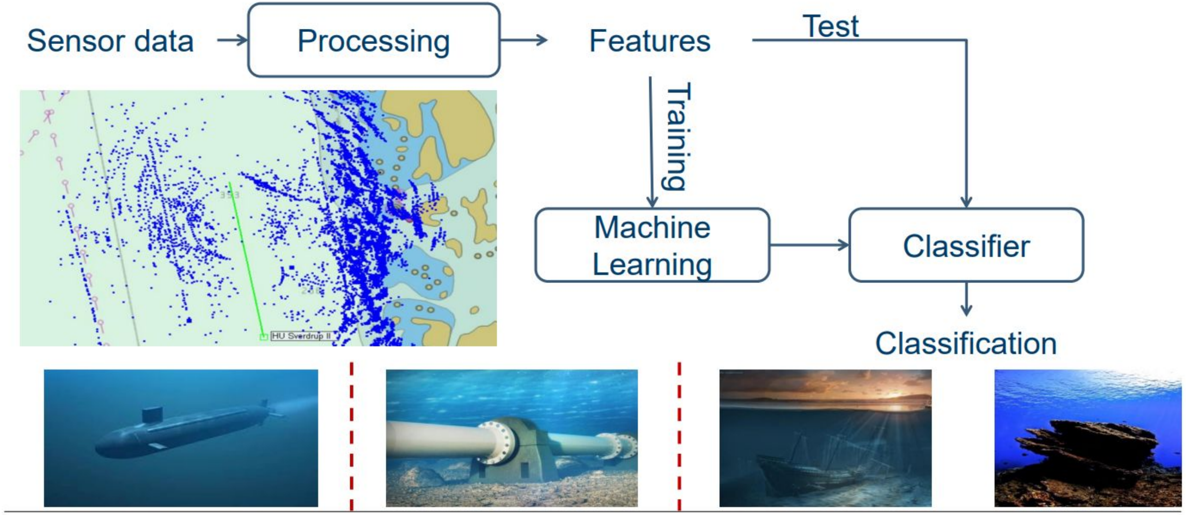
AI things that you can do (or BMW) with MATLAB

	True Positive (%)	True Negative (%)	False Positive (%)	False Negative (%)
K-Nearest Neighbor with PCA	94.74	90.35	5.26	9.65
Support Vector Machine	98.92	73.07	1.08	26.93
Quadratic Discriminant Analysis	98.83	82.73	1.17	17.27
Decision Trees	98.16	95.86	1.84	4.14

Statistics and Machine Learning Toolbox™

AI things that you can do with MATLAB

Classification of anti-submarine warfare sonar targets using a deep neural network (**Norwegian Defense Research Establishment**).



Deep Learning Toolbox

AI things that you can do with MATLAB

AI-Based Predictive Maintenance Models for Offshore Wind Power (Korea Institute of Energy Research)

“If we had used an open-source alternative such as Python, it would have taken more time to preprocess data, develop sound diagnostic algorithms, and create a dashboard.”

Deep Learning Toolbox + Statistics and Machine Learning Toolbox

Hands-on Machine Learning with MATLAB

Handwriting Recognition Using Bagged Classification Trees

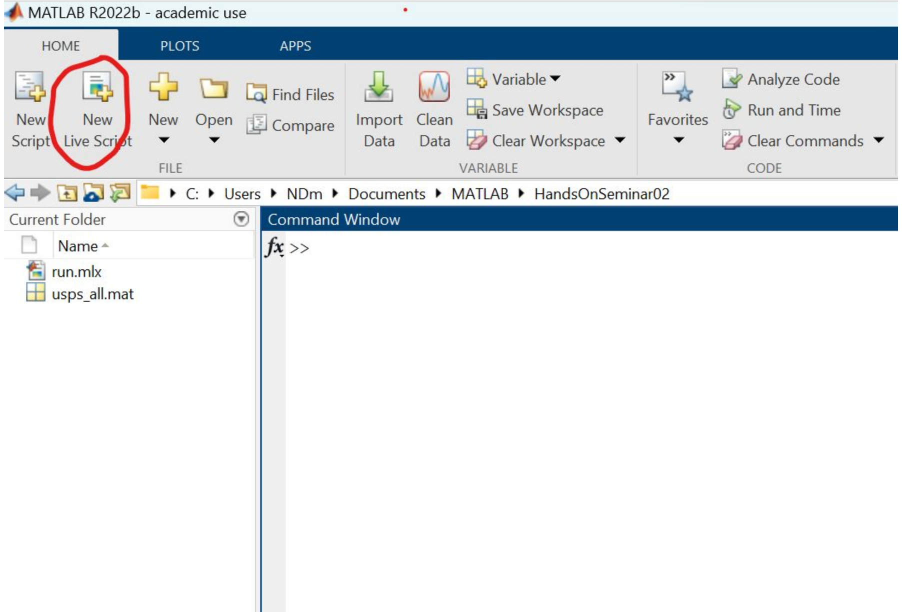
Data & code download: https://github.com/NgDMau/matlab

Description:

This example shows how to recognize handwritten digits using an ensemble of bagged classification trees. Images of handwritten digits are first used to train a single classification tree and then an ensemble of 200 decision trees. The classification performance of each is compared to one another using a confusion matrix.

Hands-on Machine Learning with MATLAB

- Open folder that you put the downloaded data (up to you)
- Create a new Live Script



Hands-on Machine Learning with MATLAB

Load training and test data

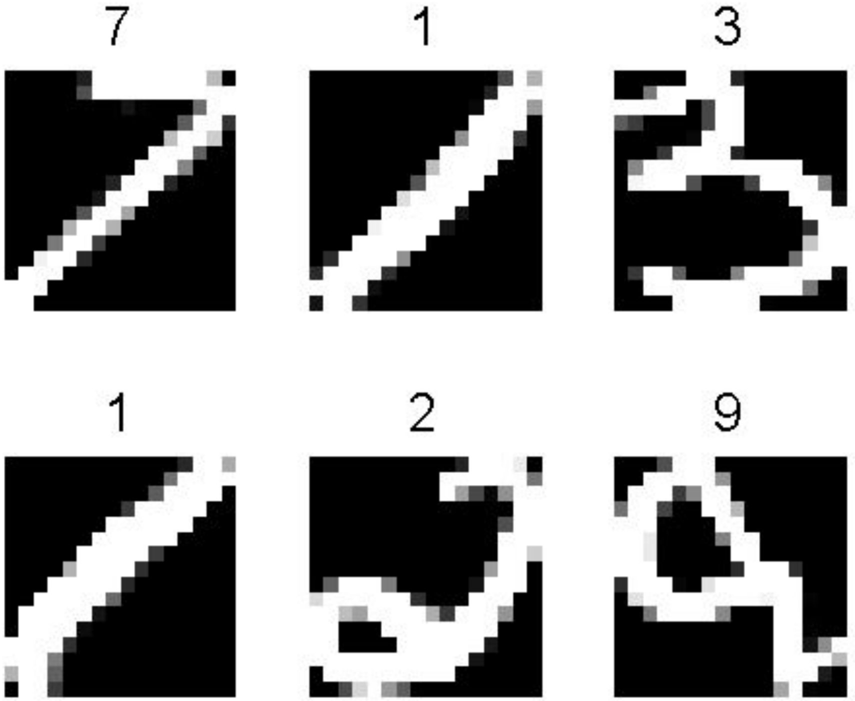
```
clear
load('usps_all');

reduce_dim = false;
X = double(reshape(data,256,11000)');
ylabel = [1:9 0];

y = reshape(repmat(ylabel,1100,1),11000,1);

clearvars data
```

Hands-on Machine Learning with MATLAB

Visualize Six Random Handwritten Samples

```
figure(1)
for ii = 1:6
    subplot(2,3,ii)
    rand_num = randperm(11000,1);
    image(reshape(X(rand_num,:),16,16))
    title((y(rand_num)),'FontSize',20)
    axis off
end
colormap gray
```

Hands-on Machine Learning with MATLAB



Hands-on Machine Learning with MATLAB

Randomly Partition the Data into Training and Validation Sets

```
cv = cvpartition(y, 'holdout', .5);
Xtrain = X(cv.training,:);
Ytrain = y(cv.training,1);

Xtest = X(cv.test,:);
Ytest = y(cv.test,1);
```

Hands-on Machine Learning with MATLAB

Train and Predict Using a Single Classification Tree

```
mdl_ctree = ClassificationTree.fit(Xtrain,Ytrain);
ypred = predict(mdl_ctree,Xtest);
Confmat_ctree = confusionmat(Ytest,ypred);
```

Hands-on Machine Learning with MATLAB

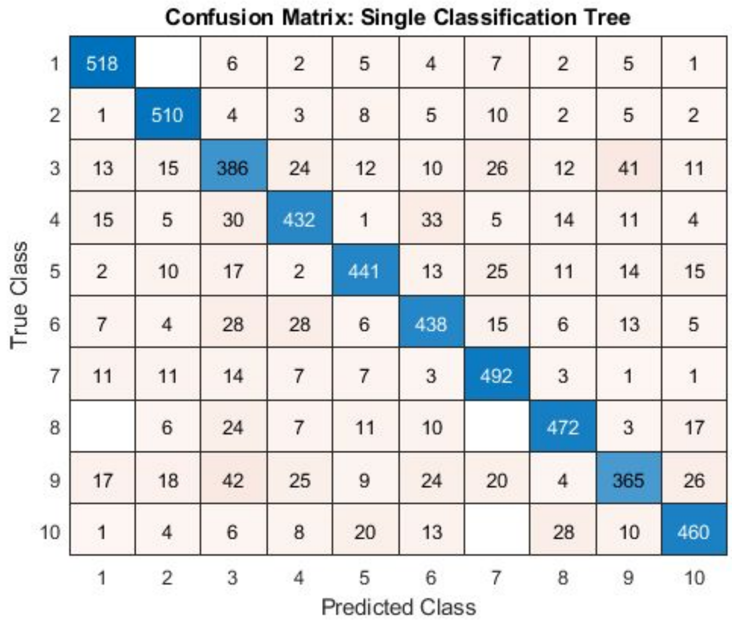
Train and Predict Using Bagged Decision Trees

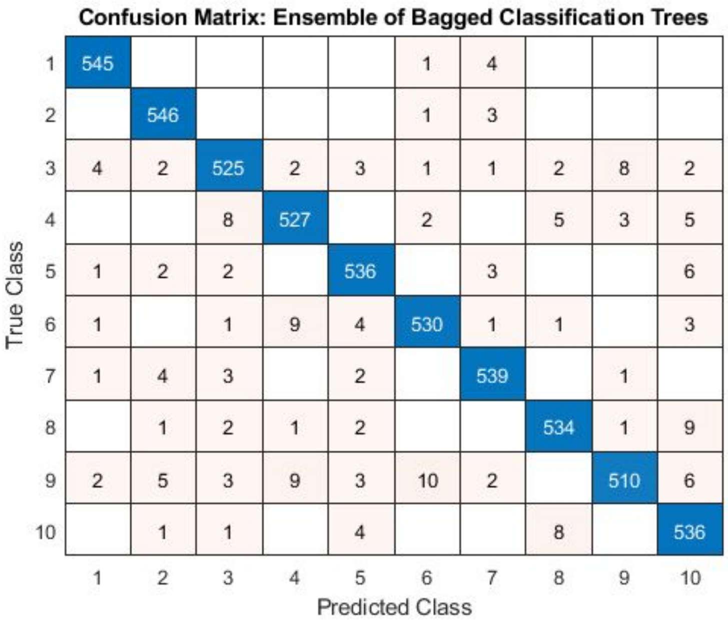
```
mdl = fitensemble(Xtrain,Ytrain,'bag',200,'tree','type','Classification');
ypred = predict(mdl,Xtest);
Confmat_bag = confusionmat(Ytest,ypred);
```

Compare Confusion Matrices

```
figure
confusionchart(Confmat_ctree)
title('Confusion Matrix: Single Classification Tree')
figure
confusionchart(Confmat_bag)
title('Confusion Matrix: Ensemble of Bagged Classification Trees')
```

Hands-on Machine Learning with MATLAB





Let’s take a break!