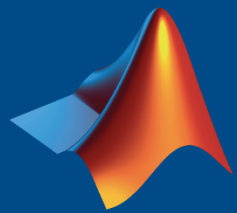
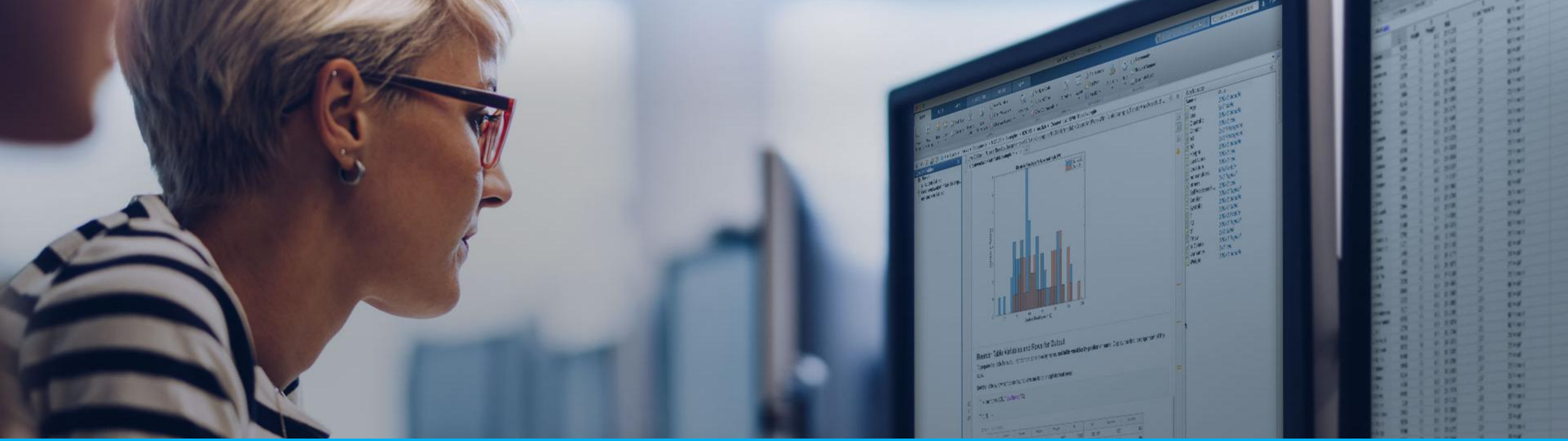


AI Workflow with MATLAB: From Feature Extraction to Deployment

Yueyi Xu
MathWorks China





MathWorks®

Accelerating the pace of engineering and science

The leading developer of mathematical computing software
for engineers and scientists.

Our software is used to design the products we rely on every day



Automobiles



Commercial Aircraft



Smartphones



Consumer Goods

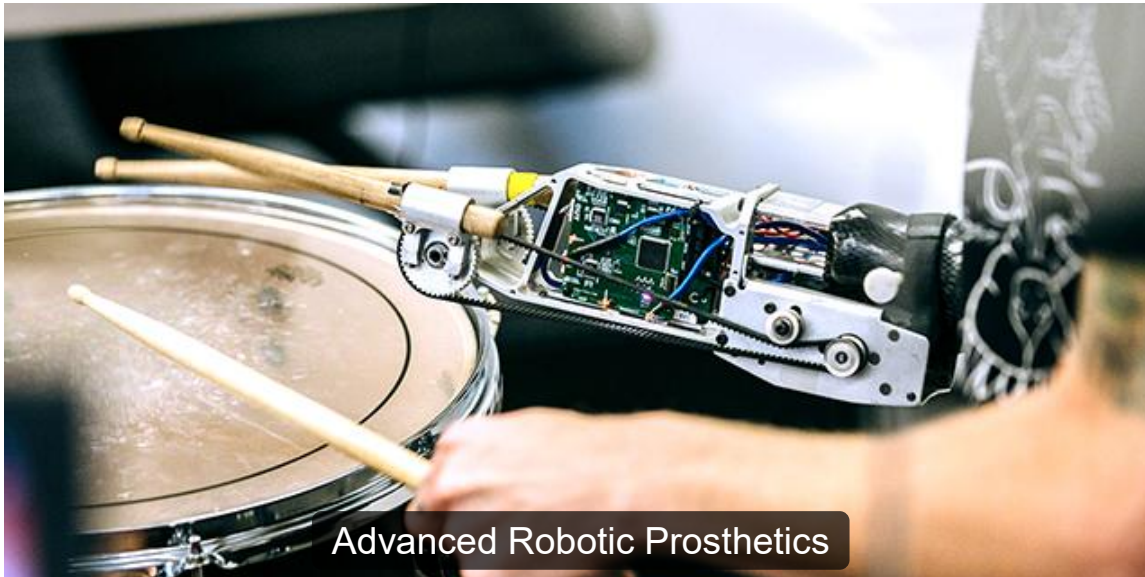
And breakthroughs that transform the way we live, learn, and work



Autonomous Vehicles



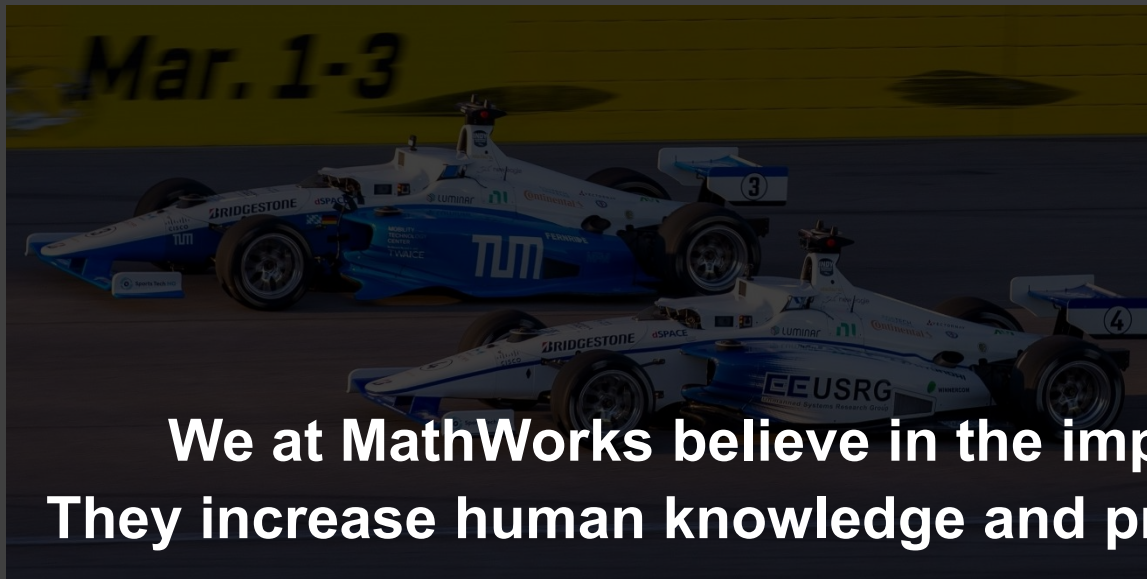
Unlimited Clean Energy



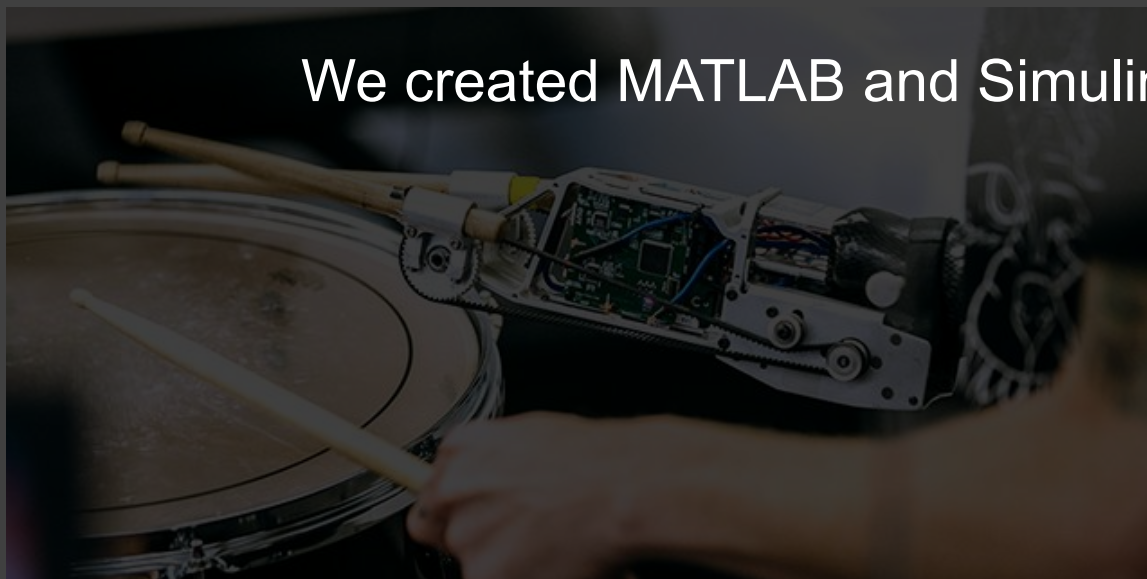
Advanced Robotic Prosthetics



Space Exploration



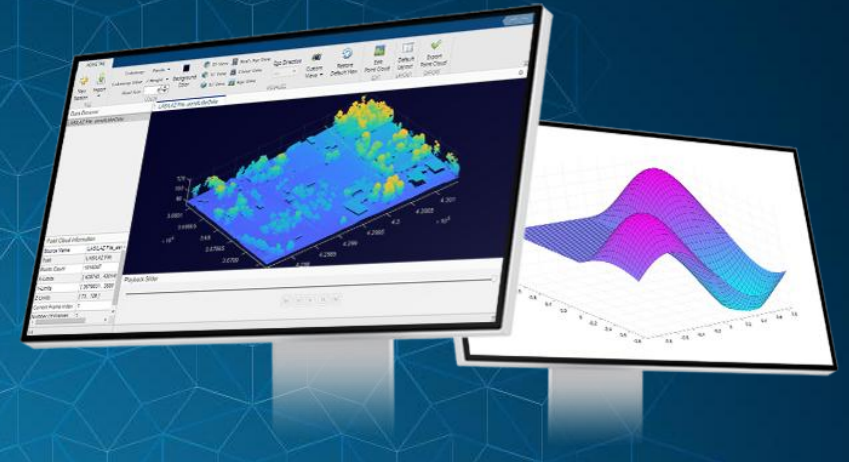
**We at MathWorks believe in the importance of engineers and scientists.
They increase human knowledge and profoundly improve our standard of living.**



We created MATLAB and Simulink to help them do their best work.

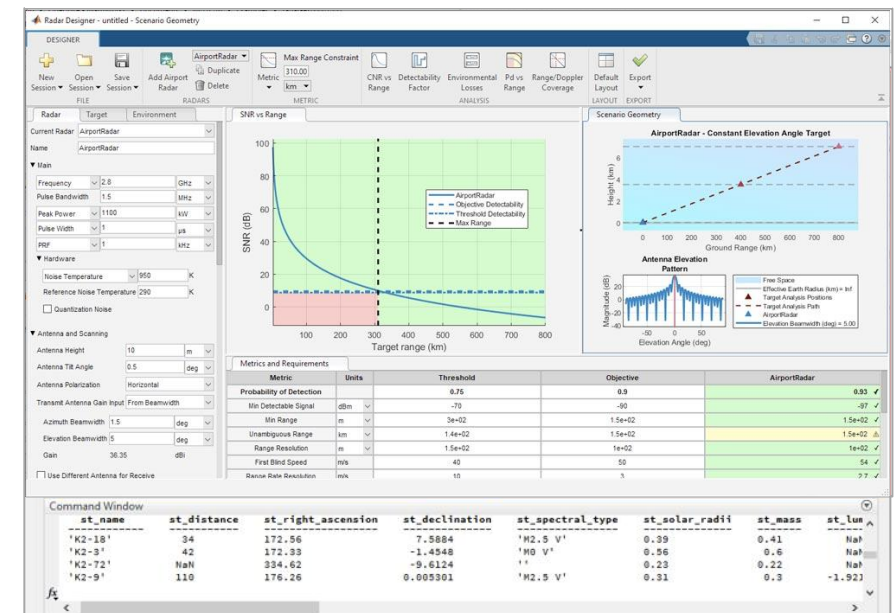


MATLAB® & SIMULINK®



- **MATLAB** – Programming environment for algorithm development, data analysis, visualization, and numeric computation.
- **Simulink** – Block diagram environment for simulation and Model-Based Design of multidomain and embedded engineering systems.
- **130+ add-on products** for specialized tasks.

Radar Toolbox



Campus Wide License

Anytime, Anywhere Access for Faculty, Staff, Students, and Visitors



MATLAB for Desktops

Access MATLAB on personal and university-owned machines



MATLAB Online

Access MATLAB with a web browser



MATLAB Mobile

Access MATLAB on iOS/Android devices

Visit your university [MATLAB portal](#)

Visit matlab.mathworks.com



Agenda

- I. AI adoption stories using MATLAB
- II. Complete AI workflow with MATLAB
- III. Online resources to fuel your AI journey

Agenda

- I. AI adoption stories using MATLAB
- II. Complete AI workflow with MATLAB
- III. Online resources to fuel your AI journey



AI is Transforming Engineered Systems



Engineered systems are designed to sense and interact with their environment to perform within operational requirements



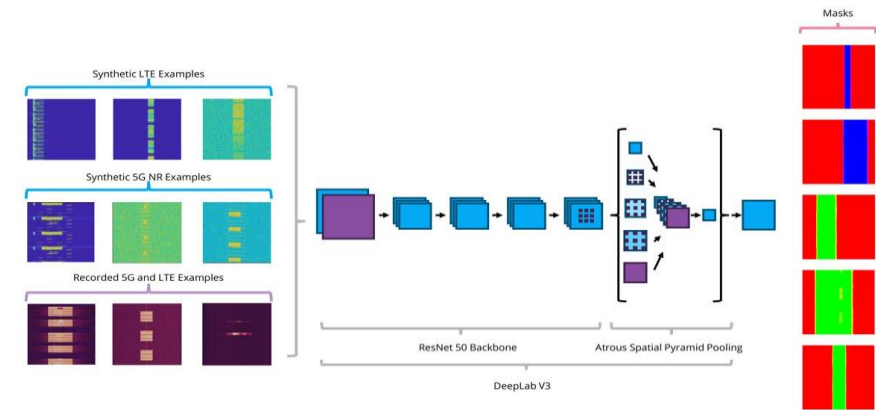
Success Stories



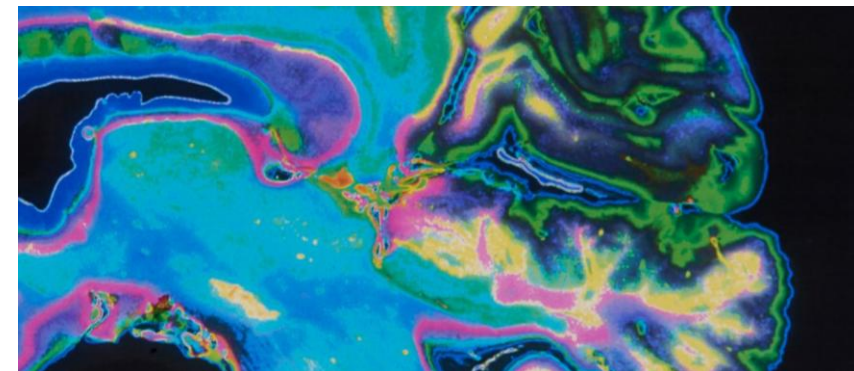
Health Monitoring & Predictive Maintenance
Mondi



Fall Risk Assessment in Older Adults
Kinesis Health Technologies



Accelerate Research on Next-Generation AI for Wireless
Qoherent

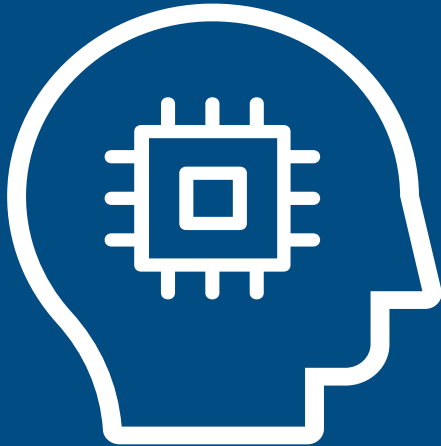


Machine Learning Aids Neurosurgeon During Brain Surgery
National Research Institute in Poland (NASK)

AI models for engineered systems

ARTIFICIAL INTELLIGENCE

Any technique that enables machines to mimic human intelligence

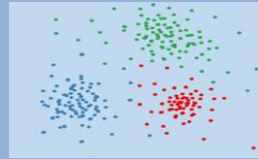


MACHINE LEARNING

Statistical methods that enable machines to “learn” tasks from data without explicitly programming

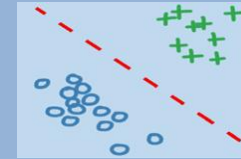
UNSUPERVISED LEARNING

(No Labeled Data)



SUPERVISED LEARNING

(Labeled Data)

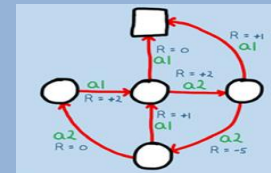


DEEP LEARNING
(Neural networks with many layers)



REINFORCEMENT LEARNING

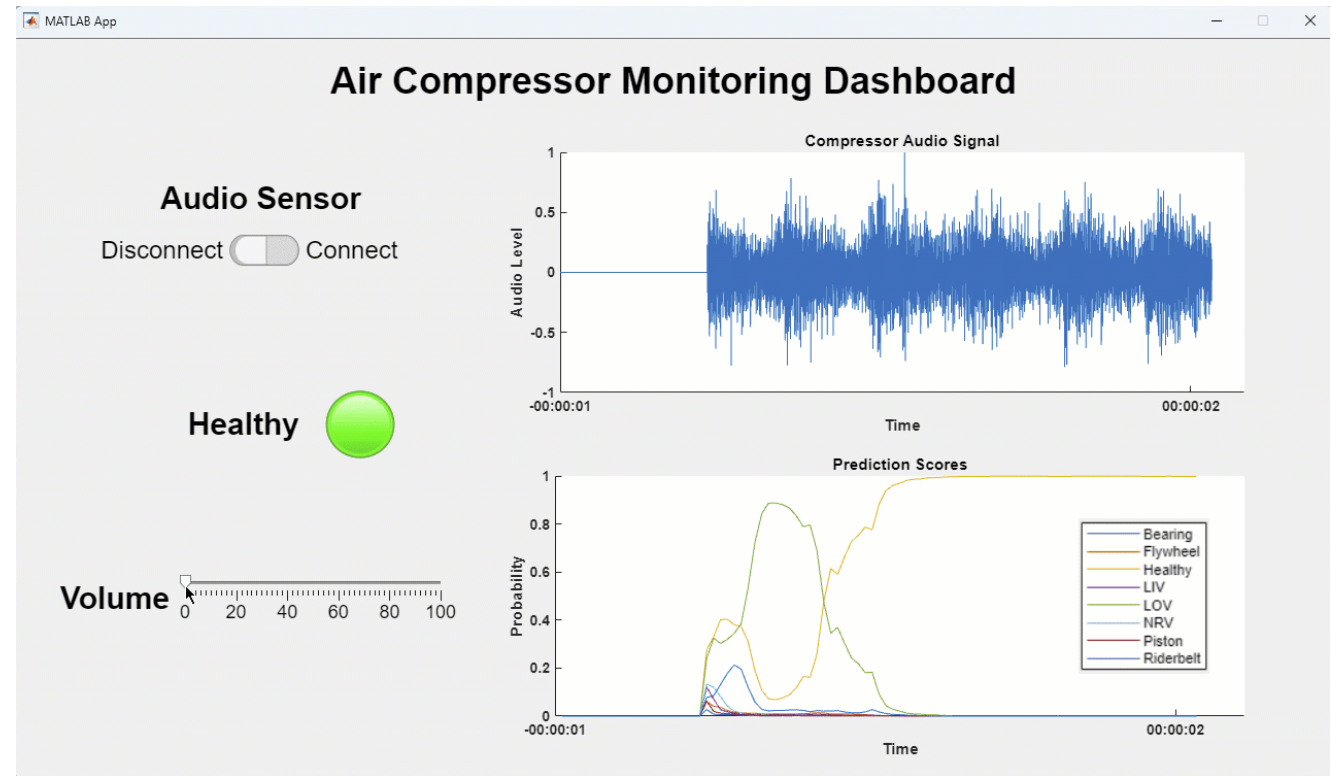
(Interaction Data)



Example: Classifying Sounds

Build, test, and deploy a deep learning solution that can classify sequences in signal data.

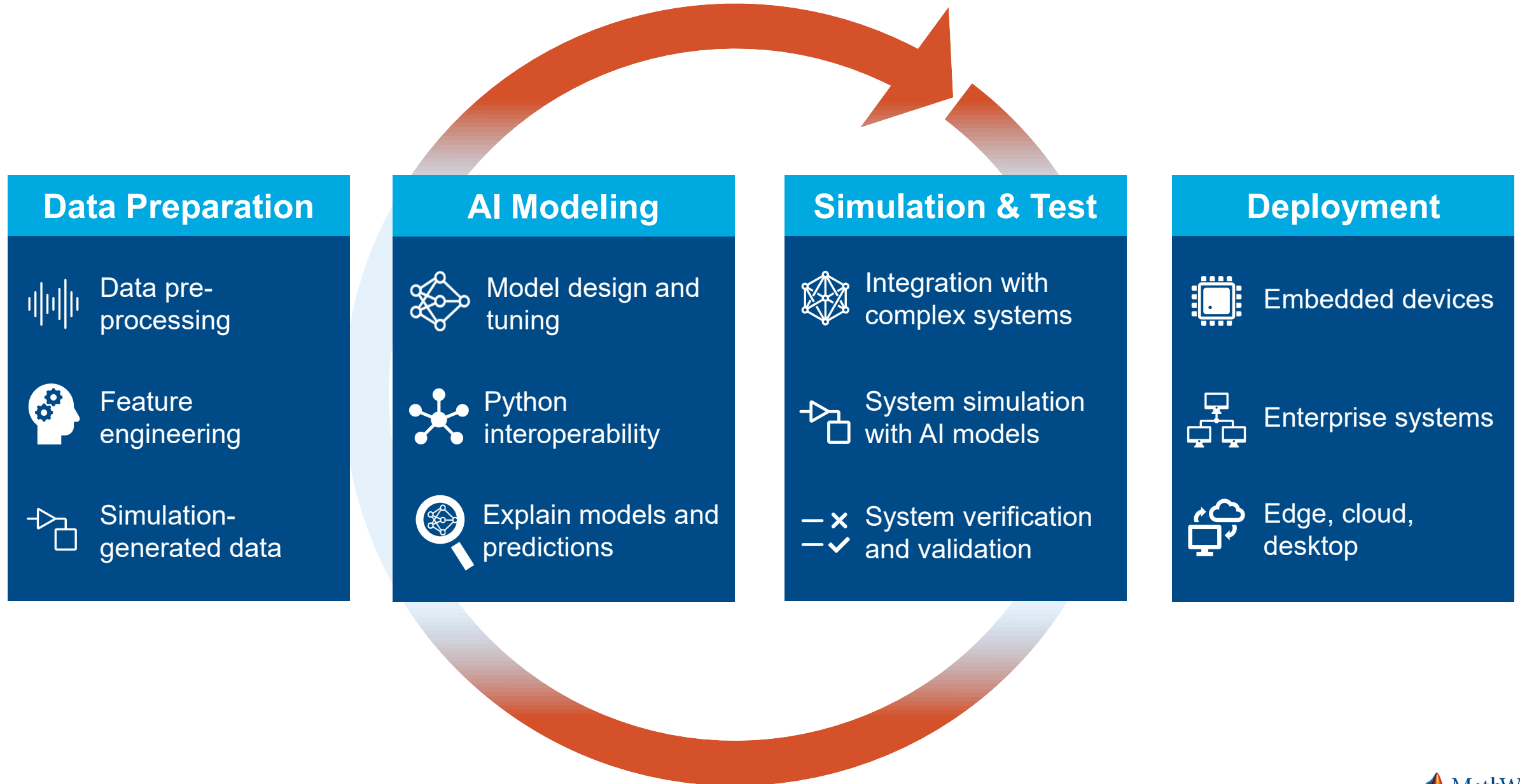
- **Goal:** Detect faults in an air compressors based on their noise



Agenda

- I. AI adoption stories using MATLAB
- II. Complete AI workflow with MATLAB
- III. Online resources to fuel your AI journey

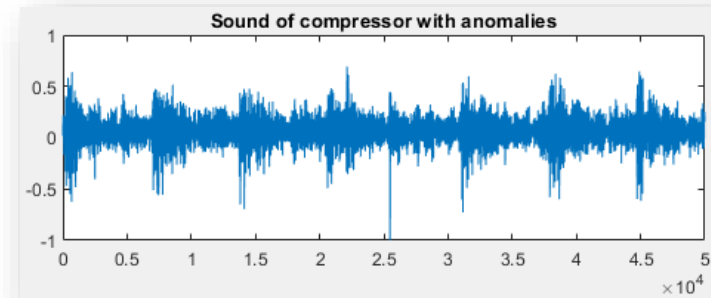
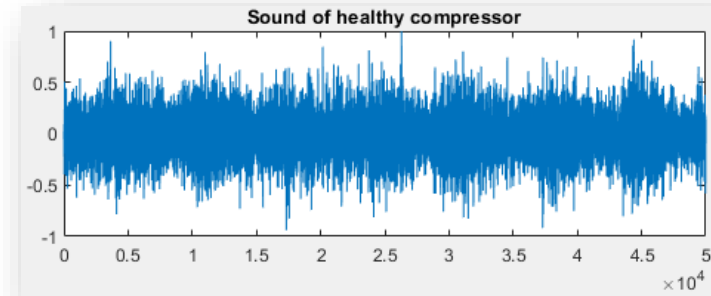
AI workflow for engineered systems



Example: Classifying Air Compressor Sounds

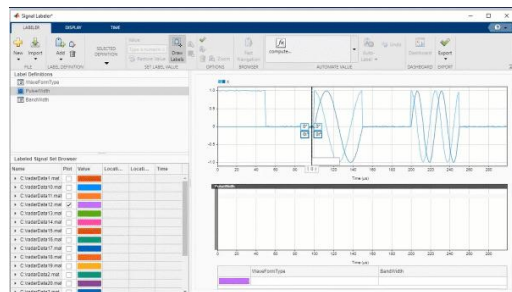
Build, test, and deploy a deep learning solution that can classify sequences in signal data.

- **Goal:** Detect faults in an air compressors based on their noise
- Have dataset with labeled sound recordings
- One “healthy” class
- 7 different classes of faults
- 1800.wav files, 225 per class



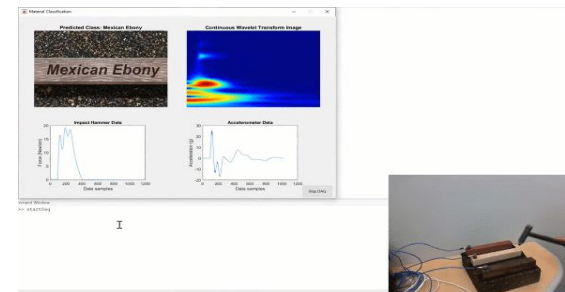
Enhance the quality of your training signal data

Define accurate
data labels



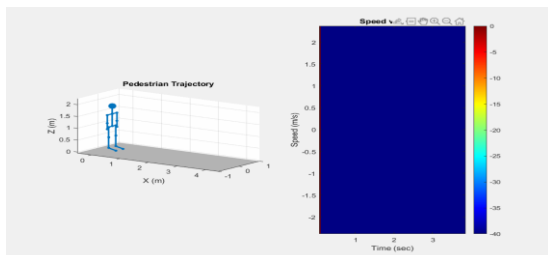
signalLabeler

Record and label new data via
Apps and Hardware

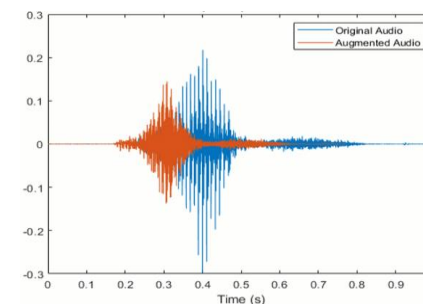


App Designer

Synthesize data via simulation



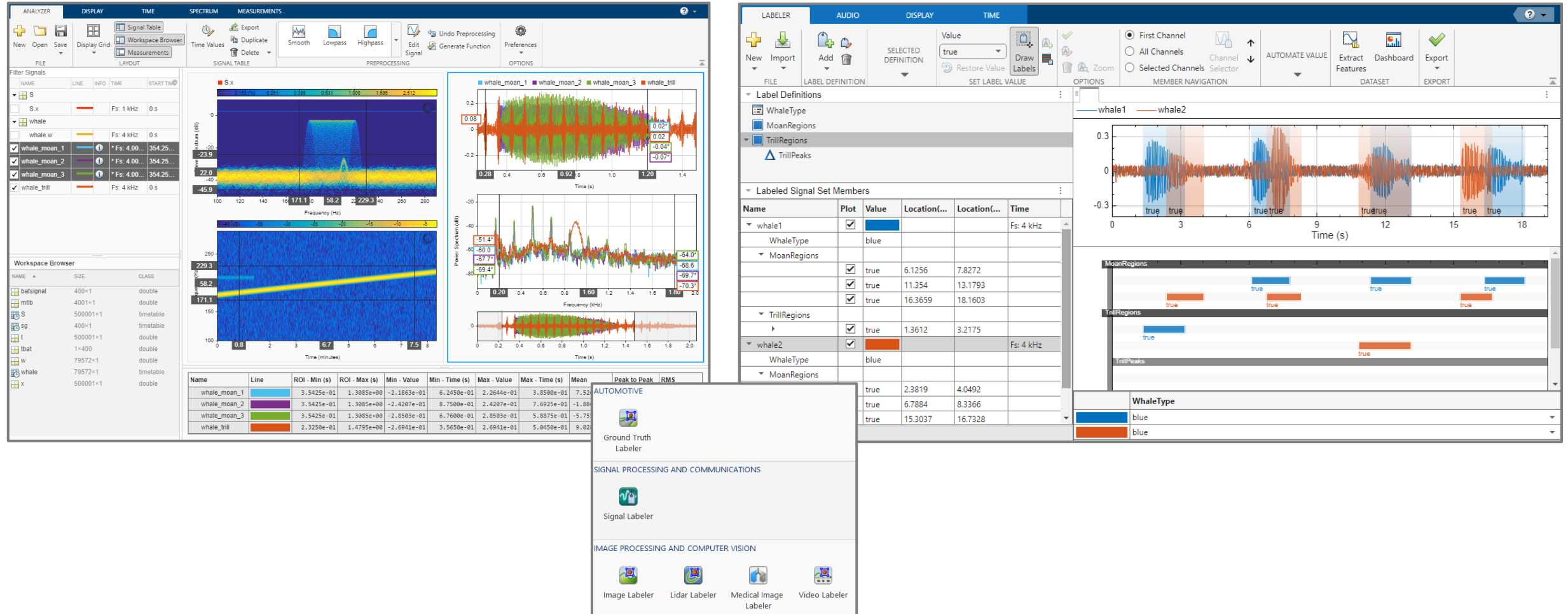
Augment data via signal processing



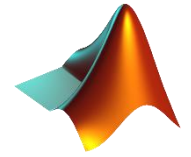
audioDataAugmenter

Spend less time preprocessing and labeling data

Synchronize disparate time series, filter noisy signals, automate labeling of signals, and more.

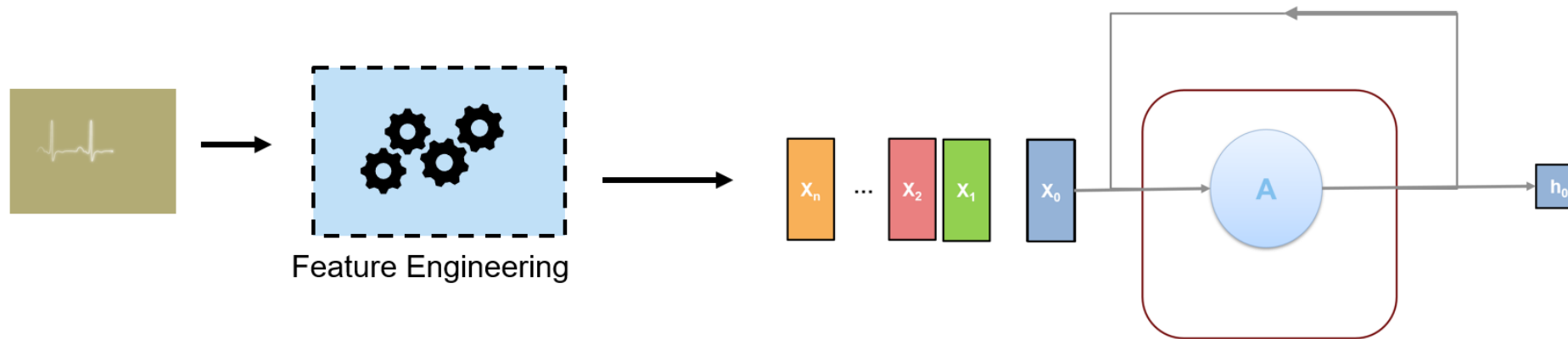


Data Preparation Demo

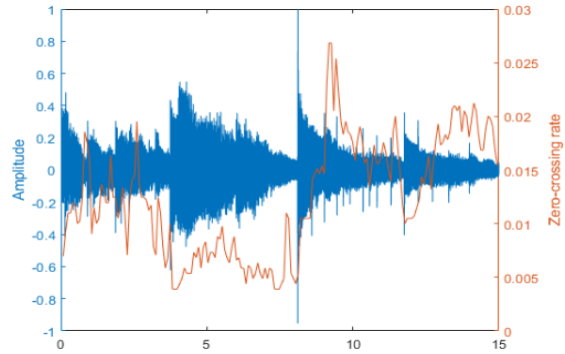


**Open Script
Part 1**

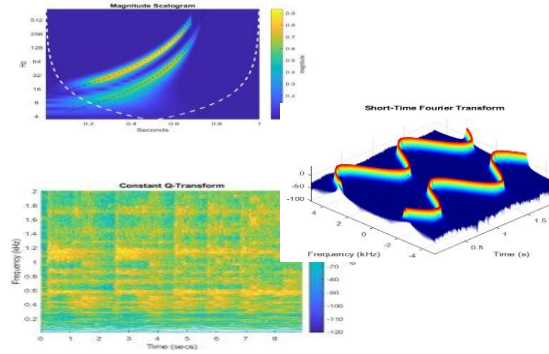
Long Short Term Memory (**LSTM**) Networks



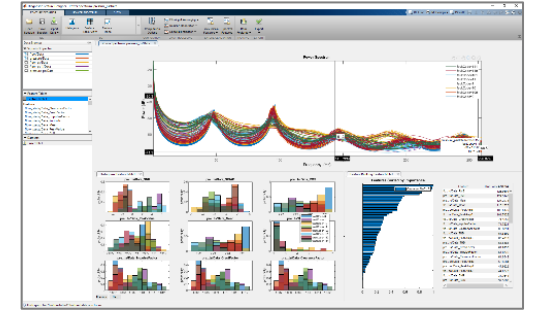
Transform raw data for useful modeling and analysis



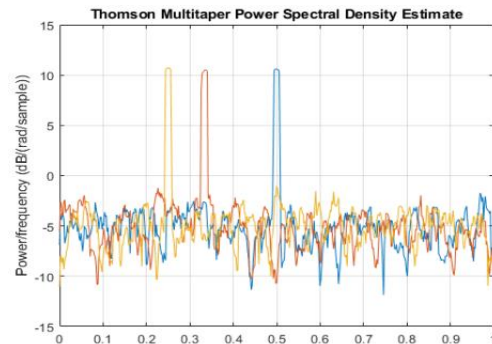
signalTimeFeatureExtractor



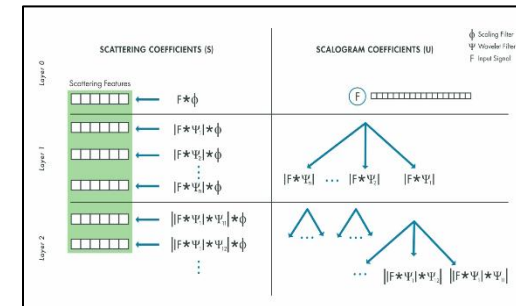
signalFrequencyFeatureExtractor



diagnosticFeatureDesigner



signalTimeFrequencyFeatureExtractor

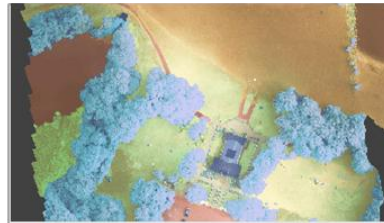


Wavelet based features

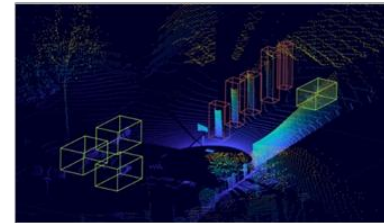
Leverage application specific reference examples



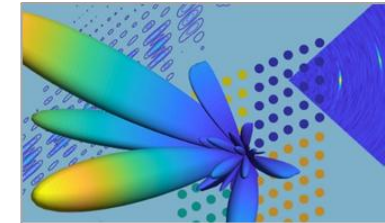
Predictive Maintenance
Anomaly Detection and Condition Monitoring



Geospatial Analysis
Hyperspectral Image Classification



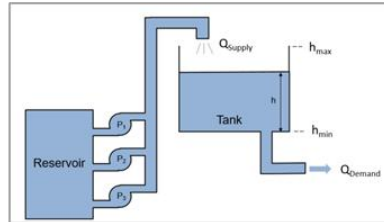
Lidar
3-D Point Cloud Object Detection



Radar
Waveform Classification



Wireless Comms
Data Synthesis for 5G Channel Estimation



Controls Systems
PID Tuning & System Scheduling



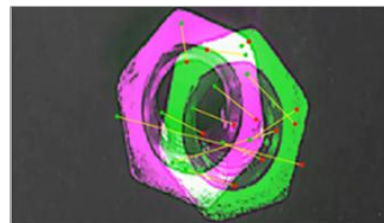
Computational Finance
Trading & Risk Management



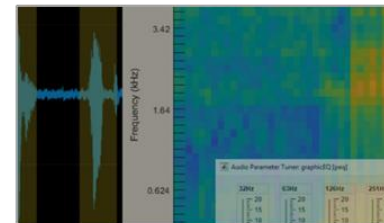
Automated Driving
Pedestrian & Vehicle Detection



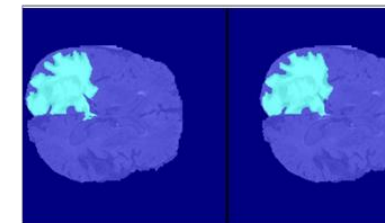
Robotics
Path Planning & Process Optimization



Visual Inspection
Defect Detection

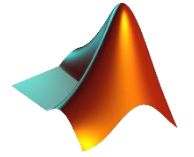


Audio
Speech Recognition



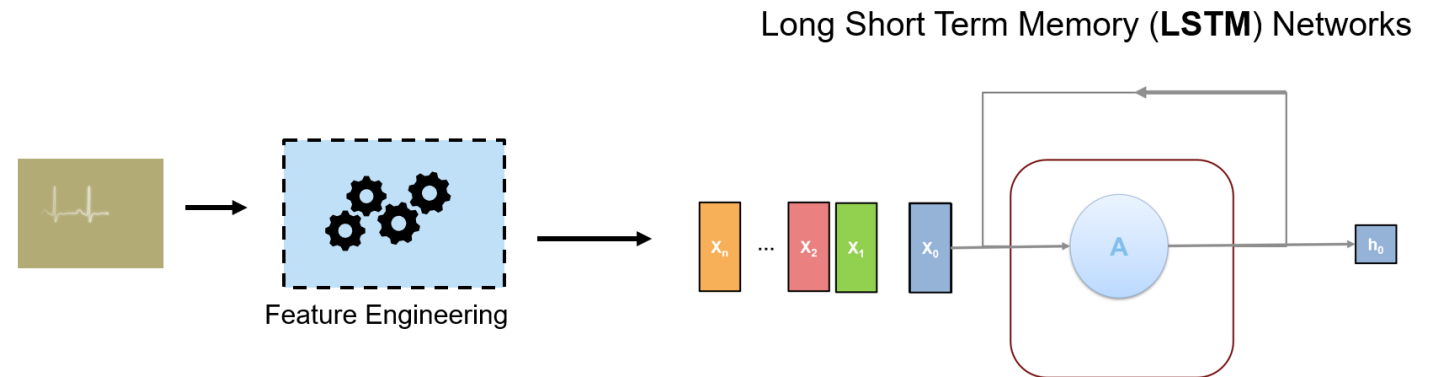
Medical Imaging
Tumor Detection

Modeling Demo



**Open Script
Part 2**

- Feature extraction with simpler and smaller AI models
 - Audio Specific Features



Modeling Demo

The screenshot displays the MATLAB R2025b environment. The top ribbon shows tabs for HOME, PLOTS, APPS, PROJECT, LIVE EDITOR, INSERT, and VIEW. The PROJECT tab is active, showing a project named 'aircompressorclassification'. The left pane lists the project's contents, including folders like 'AirCompressorData', 'codegen', 'HelperFiles', 'resources', 'Results', 'SavedData', 'slprj', and 'Tests', as well as files like 'Aircompressorclassification.prj', 'AirCompressorDataset.rights', 'Experiment1.mat', 'Experiment1_setup1.mlx', 'Experiment1_setup2.mlx', 'Experiment1_setup3.mlx', 'Experiment2.mat', 'Experiment2_setup1.mlx', 'Experiment3Function1.mlx', 'Experiment3.mat', 'Experiment3_setup1.mlx', '.git', '.gitignore', 'LICENSE', 'Part01_DataPreparation.asv', 'Part01_DataPreparation.mlx', 'Part02_Modeling.mlx', 'Part03_Simulation.slx', 'Part03_Simulation.slx', 'Part03_Simulation.slx', 'Part03_Simulation.slx', 'Part04a_CodeGeneration.mlx', 'Part04b_AppDeployment.mlapp', and 'README.md'. The main workspace shows a 'Configuration' section with a 'doTraining' checkbox checked, and a 'Load Data' section. The Command Window at the bottom shows the execution of 'clear' and 'Dataset found.' commands.

Configuration

Click on the checkboxes below to choose options for how to run this script.

1 ☒ doTraining = ☒ ;

Make sure we run this as a project. We then reset the random number generator (for consistent results).

```
2 try
3   prj = currentProject;
4 catch
5   open("Aircompressorclassification.prj");
6   OpenPart2;
7   prj = currentProject;
8 end
9
10 rng(3);
```

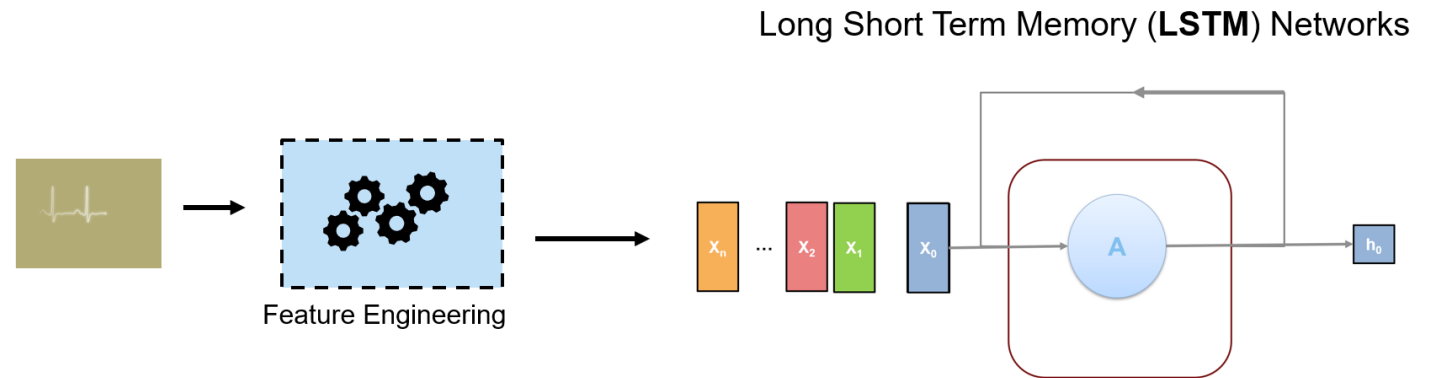
Load Data

Command Window

```
Checking for dataset...
Dataset found.
>> clear
Checking for dataset...
Dataset found.
>> clear
>>
```

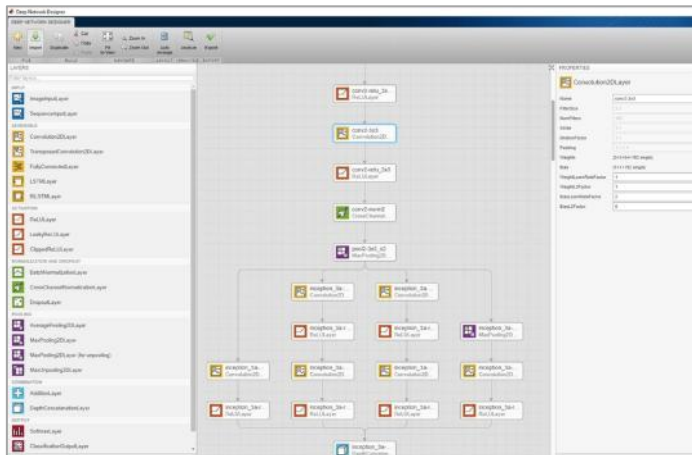
Modeling Demo

- Feature extraction with simpler and smaller AI models
 - [Audio Specific Features](#)
 - [Wavelet Scattering](#)
- Reduce training time or have small amount of data
 - [Adapt a pretrained network to classify new audio signals](#)

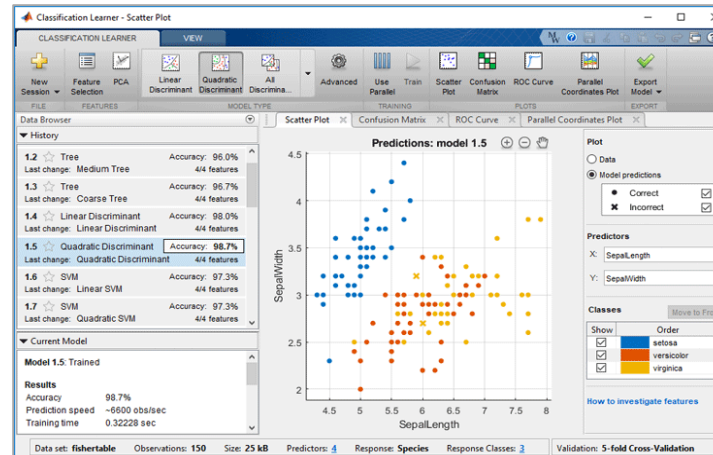


Increase productivity using apps for design and analysis

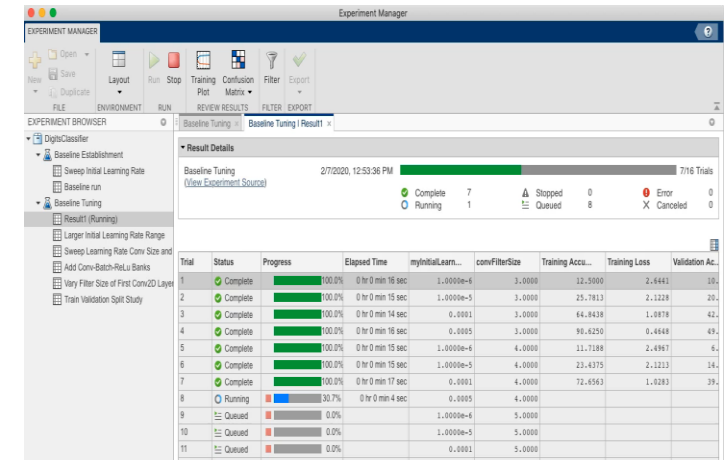
Use MATLAB Apps to design deep learning networks, explore a wide range of classifiers, train regression models, train an optical character recognition model, and more.



Deep Network Designer app to build, visualize, and edit deep learning networks.



Learner apps to try different algorithms and find the best fit for data sets.



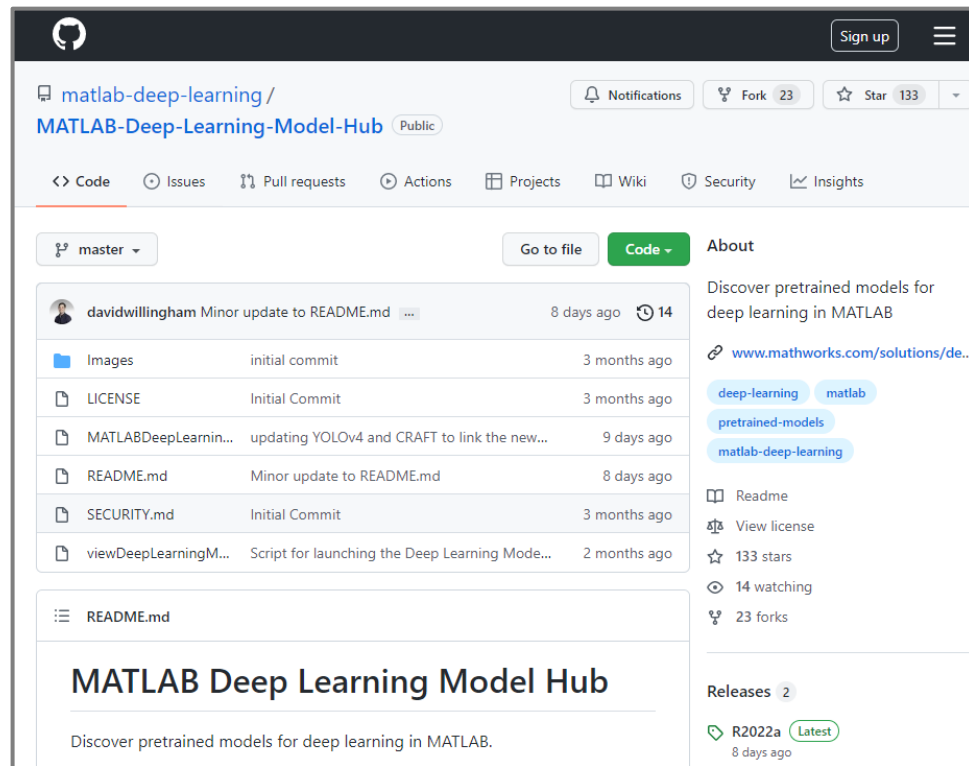
Experiment Manager app to run AI experiments to train models and compare results.

Access pretrained models

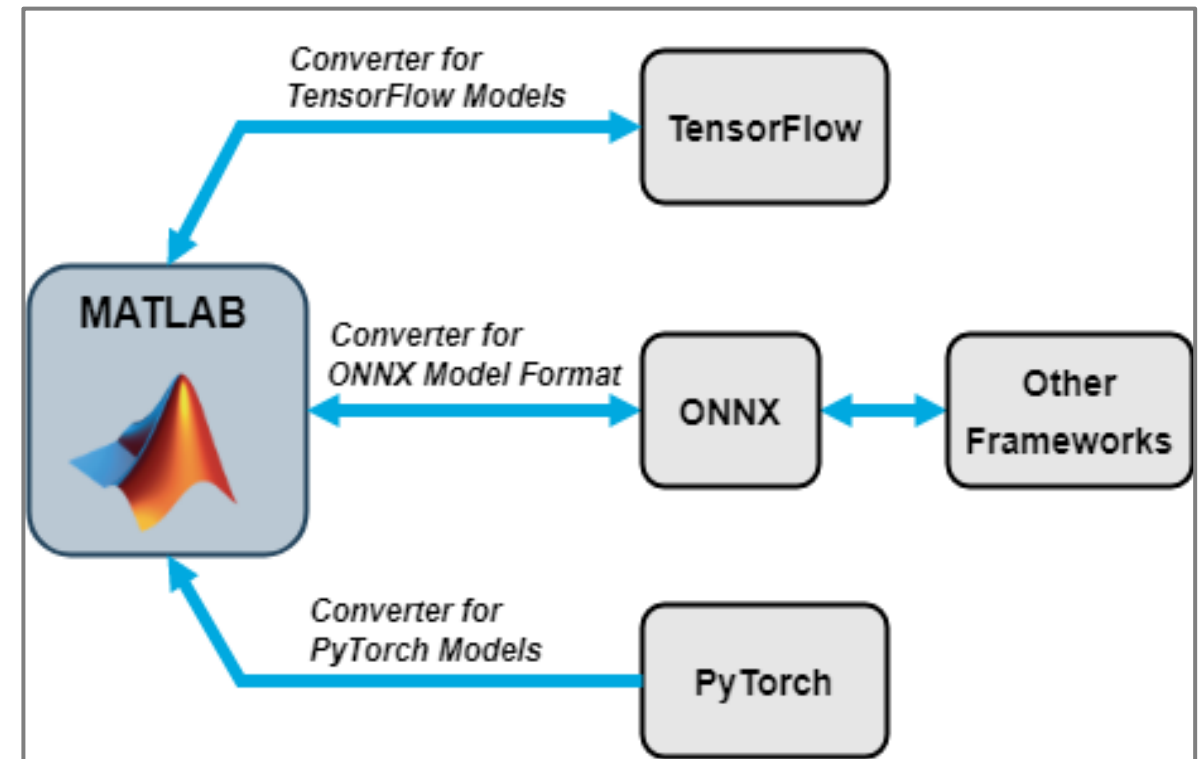
Take advantage of the knowledge provided by pretrained networks to learn new patterns in new data

Find one directly in MATLAB

<https://github.com/matlab-deep-learning/MATLAB-Deep-Learning-Model-Hub>



Import it from other platforms



Import models from PyTorch

The image shows the MATLAB R2025b interface with a script titled 'importNetwork_DND.mlx' open in the Live Editor. The script is titled 'Import Network from PyTorch and Classify Image' and describes the process of importing a pretrained PyTorch model (MNASNet) into MATLAB and using it to classify an image.

The script contains the following steps:

1. `deepNetworkDesigner`
The `DeepNetworkDesigner` automatically added an `imageInputLayer` with an arbitrary Input Size. The next step is go into the `Designer` and modify the `imageInputLayer`'s Input Size in the Properties to `[224,224,3]`.
2. The export the network to the MATLAB Workspace, and re-name the network.
`net = net_1`
3. Read the image you want to classify.
`Im = imread("peppers.png");`
Resize the image to the input size of the network. Show the image.
`InputSize = [224 224 3];`
`Im = imresize(Im,InputSize(1:2));`
`imshow(Im)`
6. The inputs to MNASNet require further preprocessing. Rescale the image. Then, normalize the image by subtracting the training images mean and dividing by the training images standard deviation. For more information, see [Input Data Preprocessing](#).

The Command Window at the bottom shows the prompt `>>`.

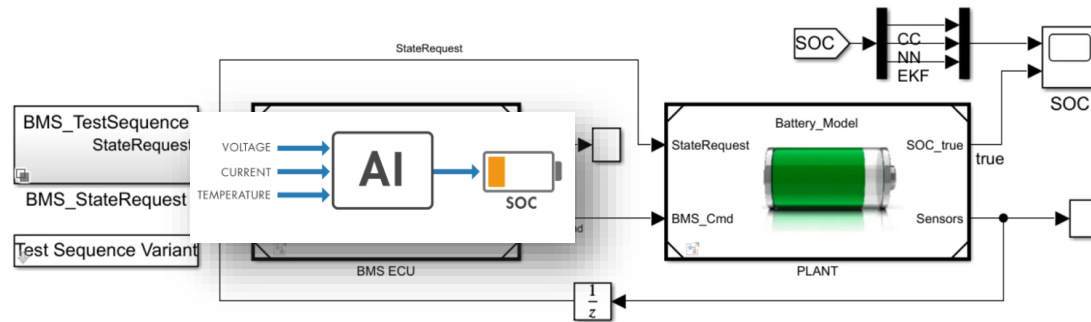
The MATLAB interface includes a top toolbar with tabs for HOME, PLOTS, APPS, LIVE EDITOR, INSERT, and VIEW. The LIVE EDITOR tab is active, showing the script. The left sidebar displays the File Explorer and Workspace. The File Explorer shows the current directory structure, including 'webcam_demo.gif', 'importNetwork_DND.mlx', and 'mnasnet1_0.pt'. The Workspace shows the variables 'Name', 'Value', and 'Size'.

AI models drive advancements in engineered systems

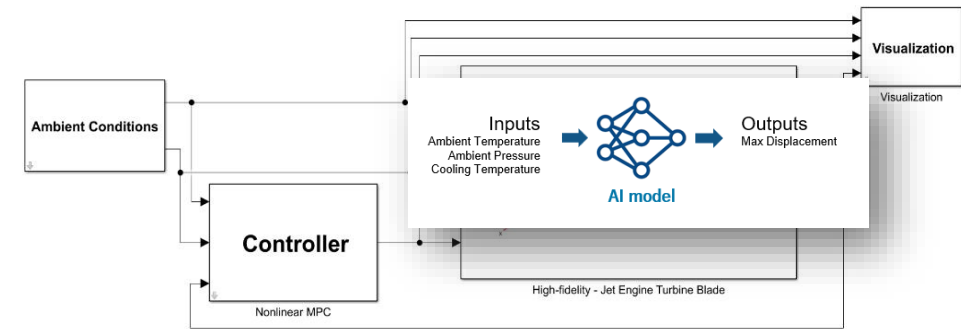
Adding data-driven components using AI can allow new capabilities, reduce costs, increase predictive accuracy, and allow earlier, more robust system testing



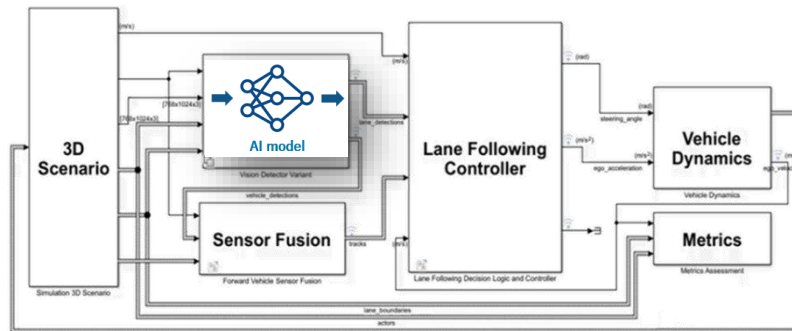
In Battery Management System:
Virtual sensor for battery state of charge (SOC) estimation



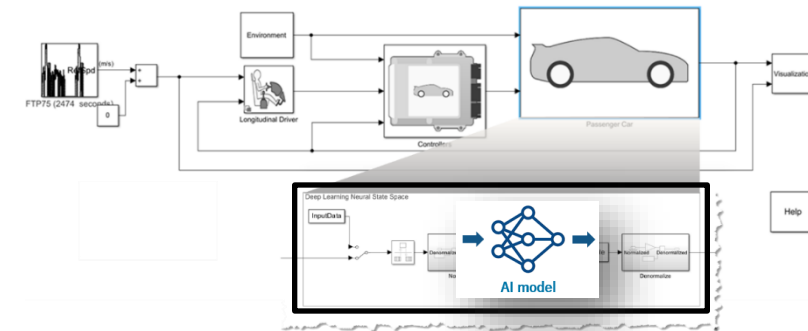
In Jet Engine Closed-Loop Control:
Reduced-order model (ROM) of blade structural behavior (FEA model)



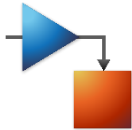
In Highway Lane Following System:
Computer vision model detects lanes for controller



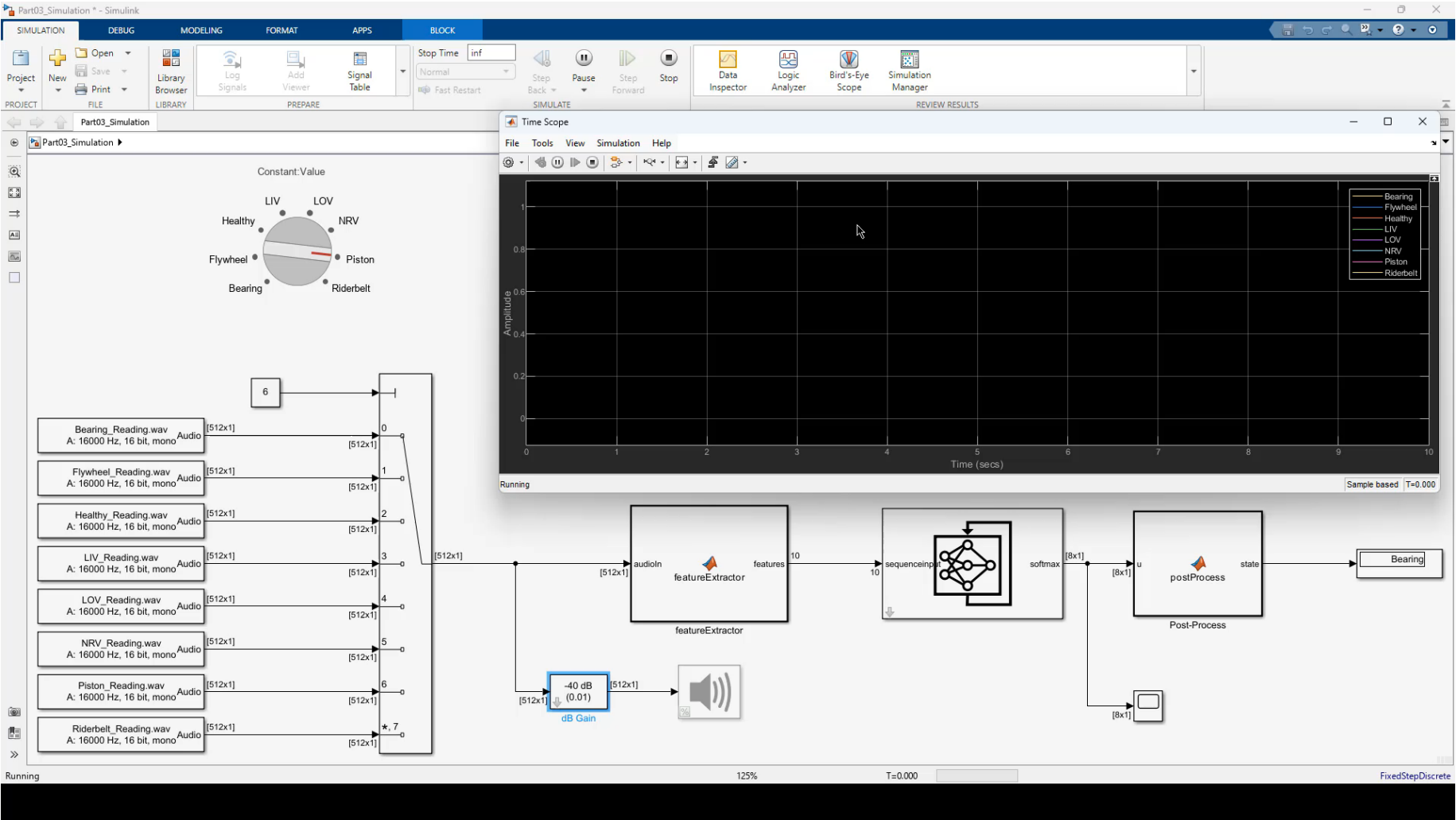
In Vehicle Drive-Cycle Virtual Test Bench:
ROM replaces the vehicle for rapid design and testing of control systems



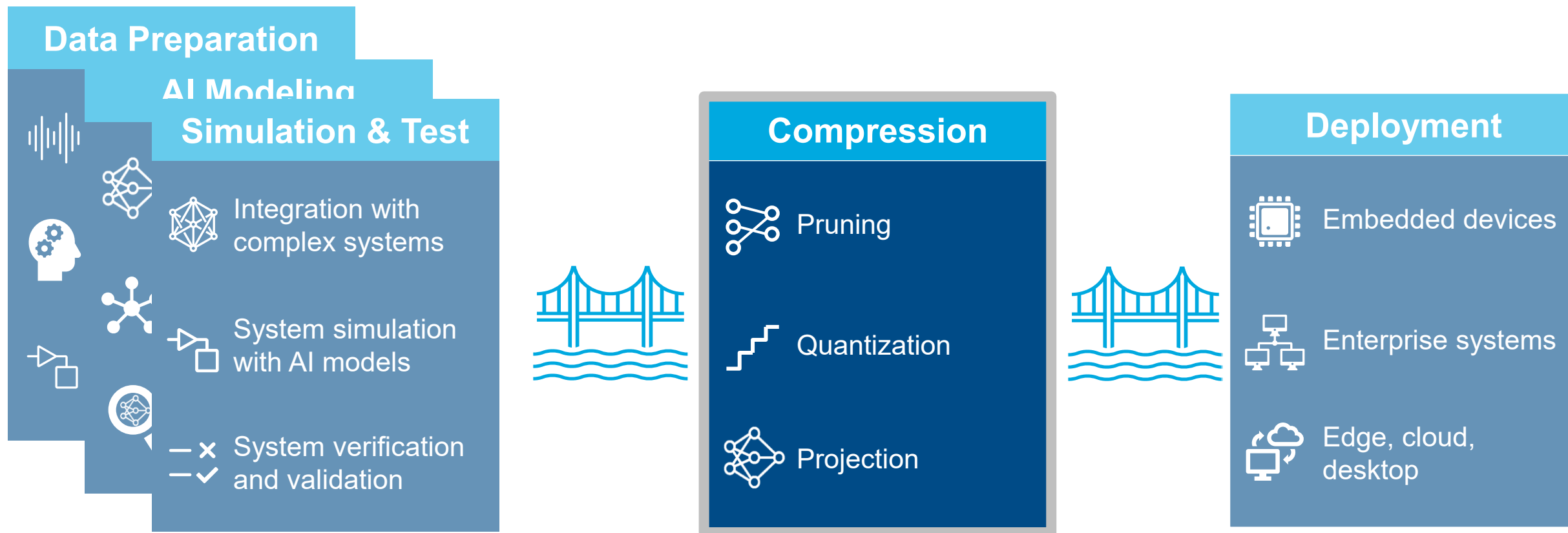
Simulation & Test Demo



Open Script
Part 3

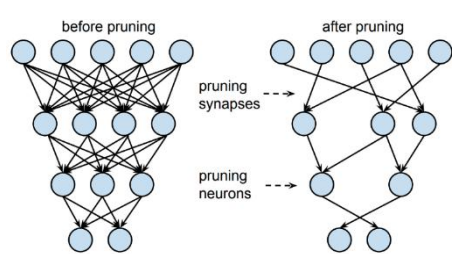


Model compression bridges the gap between design & deployment

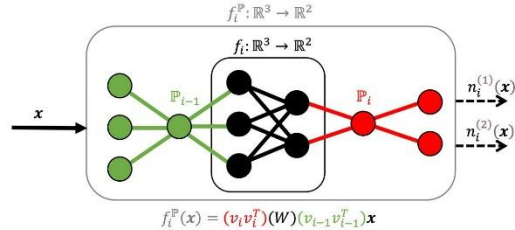


Model Compression

Structural Compression



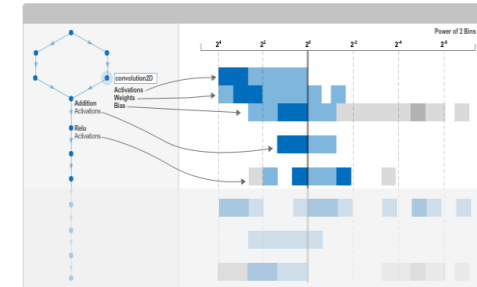
Pruning
convolutional neural
networks



Projection of deep
neural networks

[Quantization, Projection, and Pruning](#)

Datatype Compression

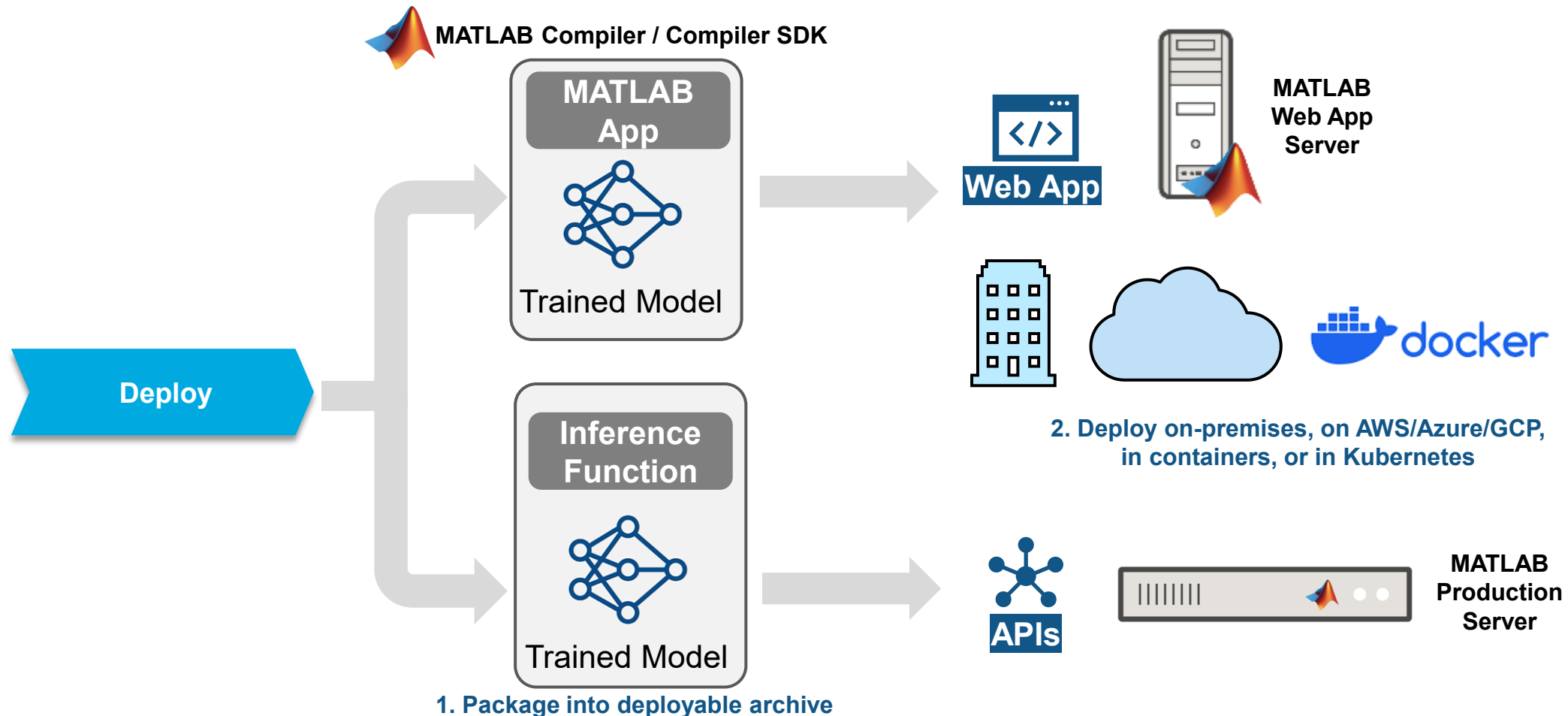


Quantization of network
weights to lower precision
datatypes (bfloat16, int8)

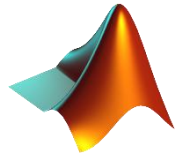
Access models *anywhere*

Provide access to trained models for inference

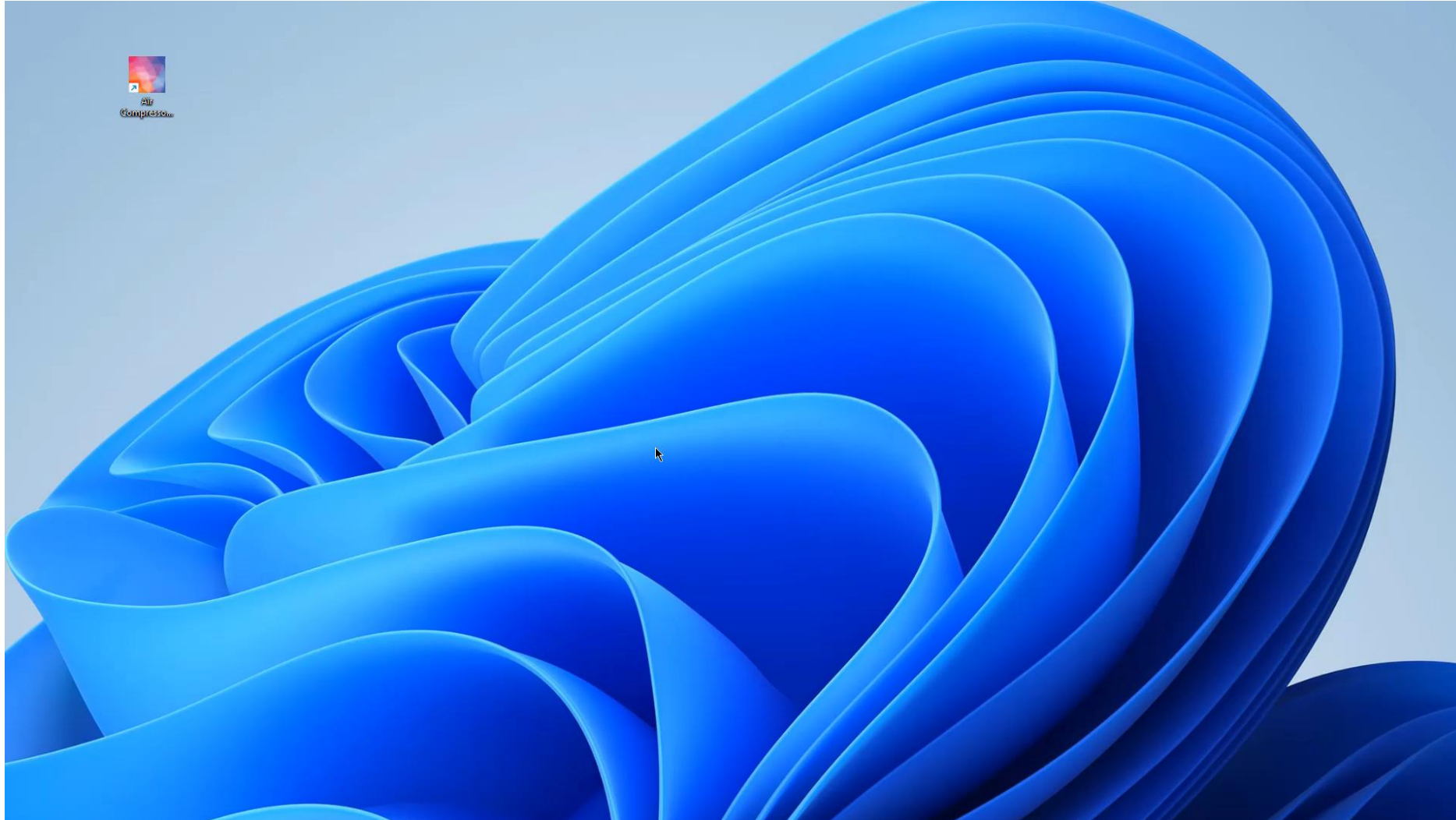
Package and deploy your trained models to enterprise IT infrastructure



Deployment Demo

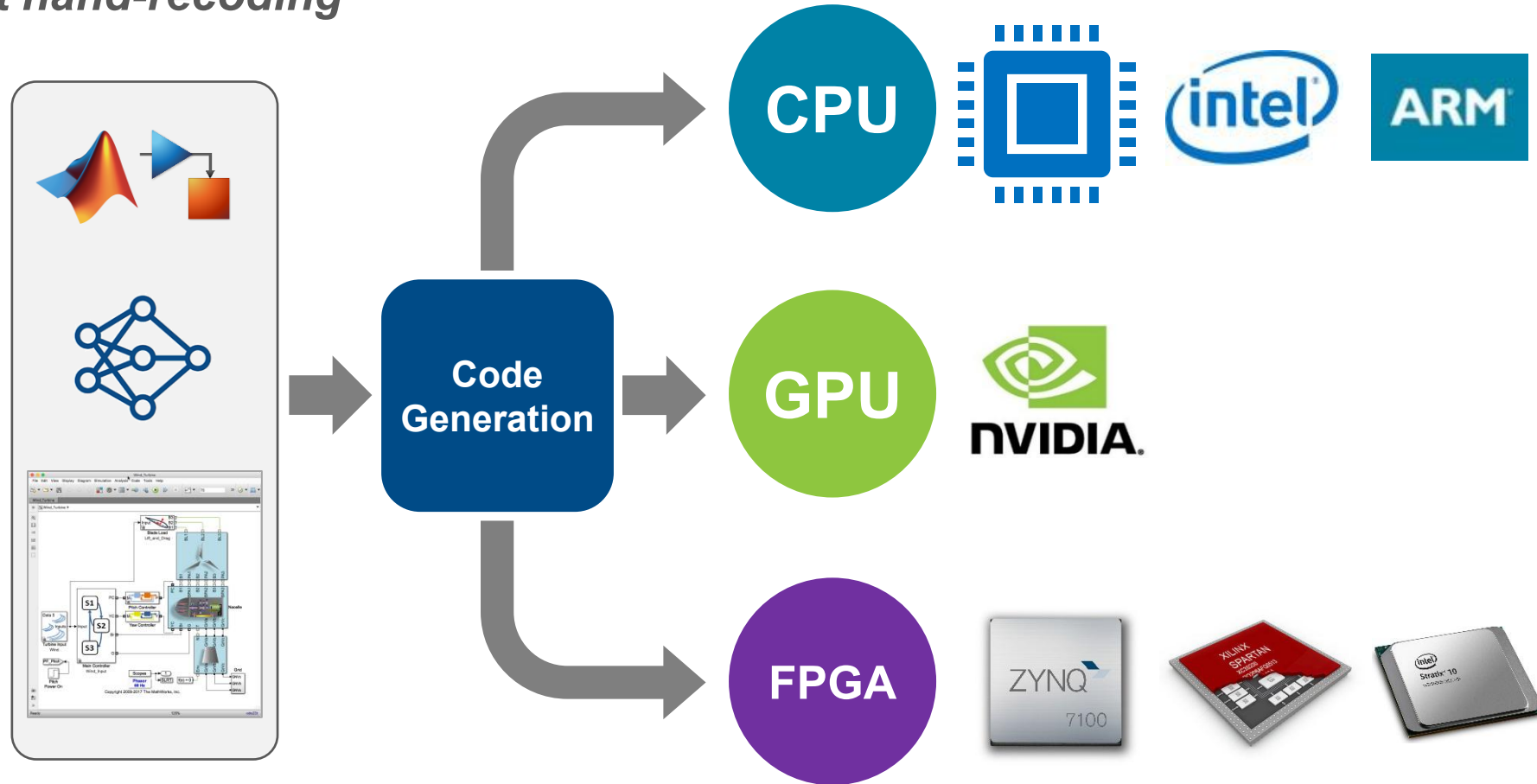


**Open Script
Part 4**

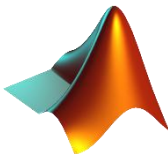


Automatically generate code for embedded Devices

Test trained models on low-cost prototyping boards or take models to production *without hand-recoding*



Deployment Demo



Open Part 4b

REPORT

Back Forward Find

Trace Code

Edit In MATLAB

Package Code

Export Report Information

NAVIGATE

MATLAB Source

Function List Call Tree

streamingClassifier.m

fx streamingClassifier

extractFeatures.m

fx extractFeatures

fx getConfig

fx getProps

Generated Code

FF Implementation

SoftmaxLayer.c

SoftmaxLayer.h

callPredict.c

callPredict.h

elementwiseOperatic

elementwiseOperatic

extractFeatures.c

extractFeatures.h

internal_softmax.c

internal_softmax.h

Function: streamingClassifier

```
1 % Copyright 2020 The MathWorks, Inc.
2
3 function [scores,features] = streamingClassifier(audioIn)
4 % This is a streaming classifier function
5
6 persistent airCompNet;
7
8 if isempty(airCompNet)
9     airCompNet = coder.loadDeepLearningNetwork('TrainedModel.mat');
10 end
11
12 % Extract features using function
13 features = extractFeatures(audioIn);
14
15 % Classify
16 [scores, state] = predict(airCompNet,dllarray(features,'CTB'));
17 airCompNet.State = state;
18 end
19
20
```

Summary

All Messages (0)

Build Logs

Code Insights (4)

Variables

Code generation successful

Generated on:

12-Mar-2024 14:36:48

Build type:

Static Library

Toolchain:

MinGW64 | gmake (64-bit Windows)

Deployment Demo

The screenshot displays the MATLAB R2025b environment with the 'PROJECT' tab selected. The project is named 'aircompressorclassification' and is located in the 'OneDrive > TAH Account > CPCE_PolyU > AirCompressorClassificationDemo' folder. The project structure is visible in the left pane, showing folders like 'AirCompressorData', 'codegen', 'HelperFiles', 'resources', 'Results', 'SavedData', 'slprj', 'Tests', and files like 'Aircompressorclassification.prj', 'AirCompressorDataset.rights', 'Experiment1.mat', 'Experiment1_setup1.mlx', 'Experiment1_setup2.mlx', 'Experiment1_setup3.mlx', 'Experiment2.mat', 'Experiment2_setup1.mlx', 'Experiment3Function1.mlx', 'Experiment3.mat', 'Experiment3_setup1.mlx', '.git', '.gitignore', 'LICENSE', 'Part01_DataPreparation.asv', 'Part01_DataPreparation.mlx', 'Part02_Modeling.mlx', 'Part03_Simulation.slx', 'Part03_Simulation.slxc', 'Part03_Simulation.slx.r2024a', 'Part04a_CodeGeneration.mlx', 'Part04b_AppDeployment.mlapp', and 'README.md'.

The main workspace shows the 'Air Compressor Data Classification' project with 'Part 3: Deployment' selected. The deployment process is outlined in four stages:

- Data Preparation:** Data cleansing and preparation, Human insight, Simulation-generated data.
- AI Modeling:** Model design and tuning, Hardware accelerated training, Interoperability.
- Simulation & Test:** Integration with complex systems, System simulation, System verification and validation.
- Deployment:** Embedded devices, Enterprise systems, Edge, cloud, desktop.

The 'Configuration' section shows a checkbox for 'doCodeGeneration' which is checked. Below it, a text box indicates 'Make sure we run this as a project.'

The 'Command Window' at the bottom shows the following output:

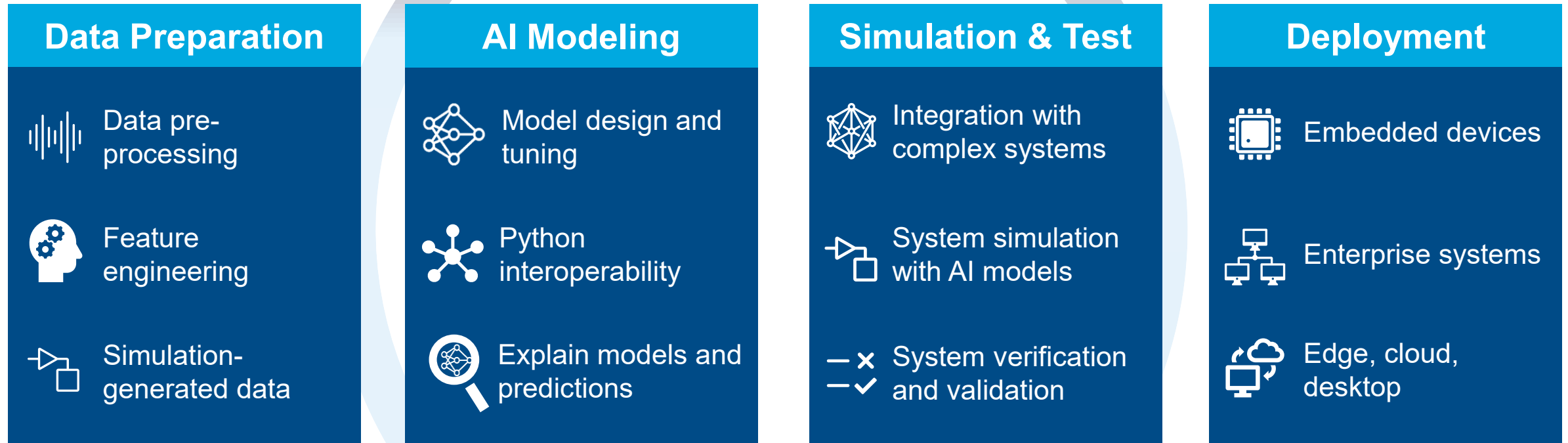
```
Checking for dataset...
Dataset found.
>> clear
Checking for dataset...
Dataset found.
>> clear
>>
```

Things we didn't talk about today...

Follow Up Topics

- Model Tuning
- Explainable AI
- Model Compression (Pruning, Projection, Quantization)
- Model Verification and Validation
- AI on the cloud workflows
- and more!

AI-driven system design



Key Take-aways

- Empower domain experts, including ones with limited AI experience
- Build better data sets with domain-specific tools
- Access models available in open-source repositories
- Use modeling and simulation to tackle integration challenges and reduce risk
- Deploy AI models to wherever you need them

Data Preparation



Data pre-processing



Feature engineering



Simulation-generated data

AI Modeling



Model design and tuning



Python interoperability



Explain models and predictions

Simulation & Test



Integration with complex systems



System simulation with AI models



System verification and validation

Deployment



Embedded devices



Enterprise systems



Edge, cloud, desktop

Demo materials

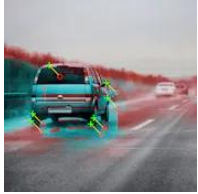
- https://mathworks-my.sharepoint.com/:f:/p/yueyixu/IgCVb7yfYSvhSJnHmFnq_u4GASXtVwuvForz09-vusZUydU?e=H656Mb
- Link is valid through July 12th, 2026

Agenda

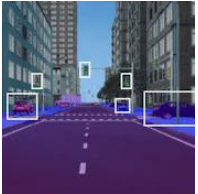
- I. AI adoption stories using MATLAB
- II. Complete AI workflow with MATLAB
- III. Online resources to fuel your AI journey

Get Started with AI in MATLAB and Simulink at your desk

No download, no installation, just your browser and you...



[Computer Vision for Engineering and Science](#)



[Deep Learning for Computer Vision](#)



[Image Processing for Engineering and Science](#)



[Practical Data Science with MATLAB](#)

coursera

Specializations – multiple-course series



[Machine Learning Onramp](#)



[Machine Learning with MATLAB](#)



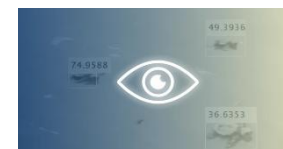
[Deep Learning Onramp](#)



[Deep Learning with MATLAB](#)



[Reinforcement Learning Onramp](#)



[Computer Vision Onramp](#)

<https://matlabacademy.mathworks.com/#ai>

Call to Action – Try out focused learning paths



Machine Learning Techniques in
MATLAB



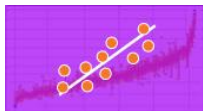
Deep Learning Techniques in
MATLAB for Image Applications



Classification Methods with Machine Learning
2 hours



Explore Convolutional Neural Networks
1 hour



Regression Methods with Machine Learning
1.5 hours



Tune Deep Learning Training Options
1 hour



Cluster Analysis with Machine Learning
1.5 hours



Regression with Deep Learning
0.5 hours



Dimensionality Reduction Techniques
1 hour



Object Detection with Deep Learning
1.5 hours



Learn more about MathWorks, our products, and our services at
mathworks.com and on social media:



Please fill out the survey to get access to today's presentation slides

Lingnan University MATLAB Event

