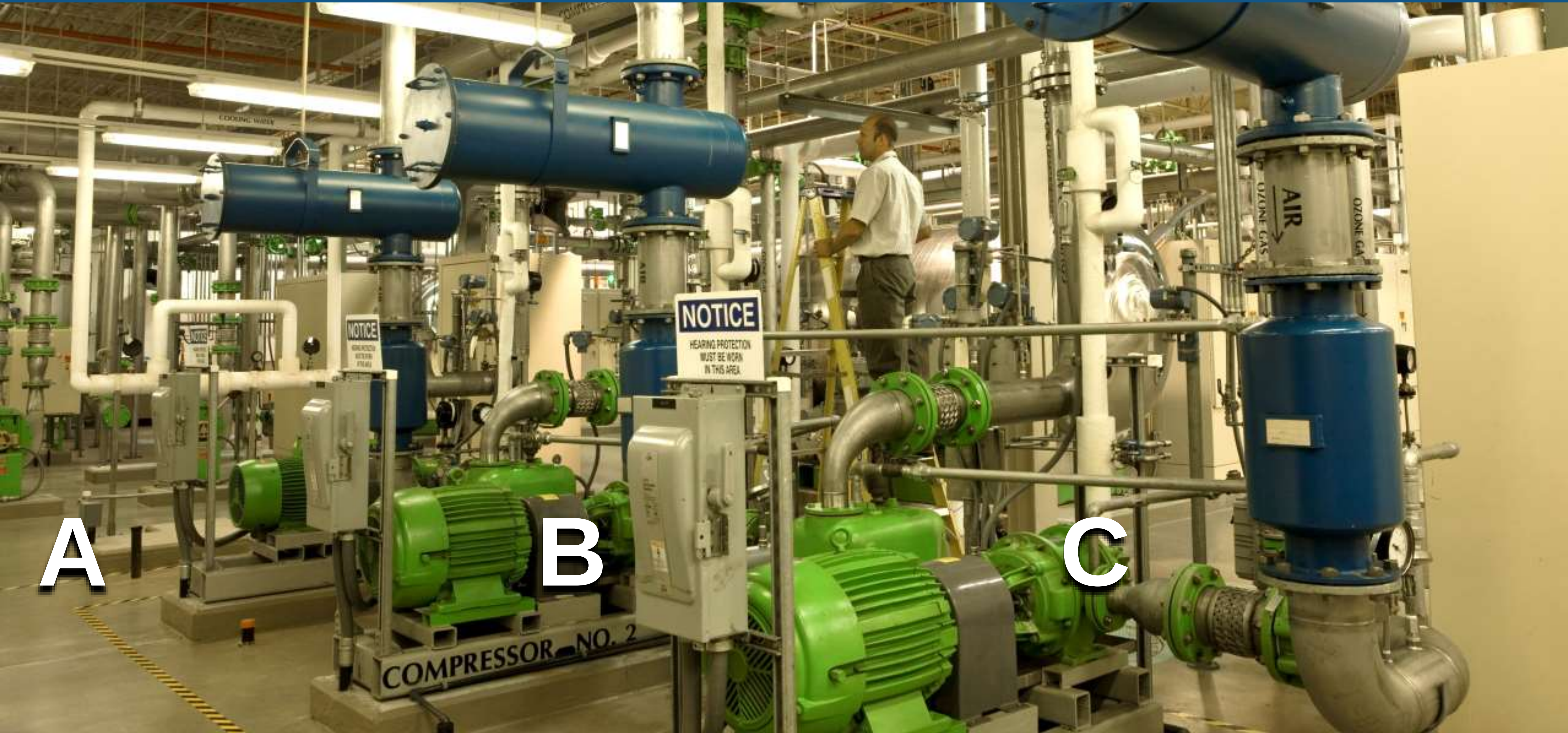
MATLAB EXPO
2021

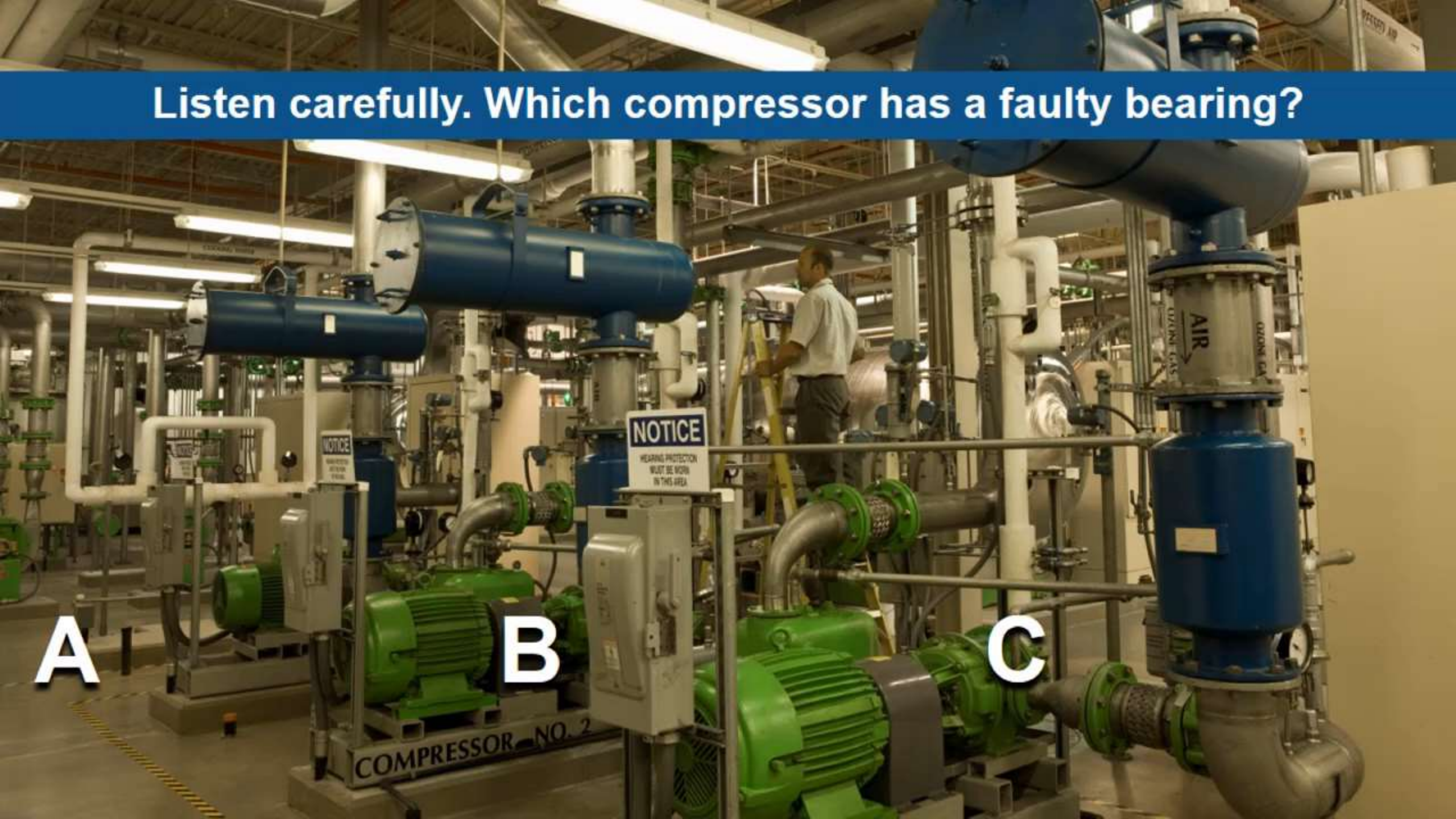
건전성 예측관리를 위한 산업용 AI 솔루션 개발

엄준상 차장

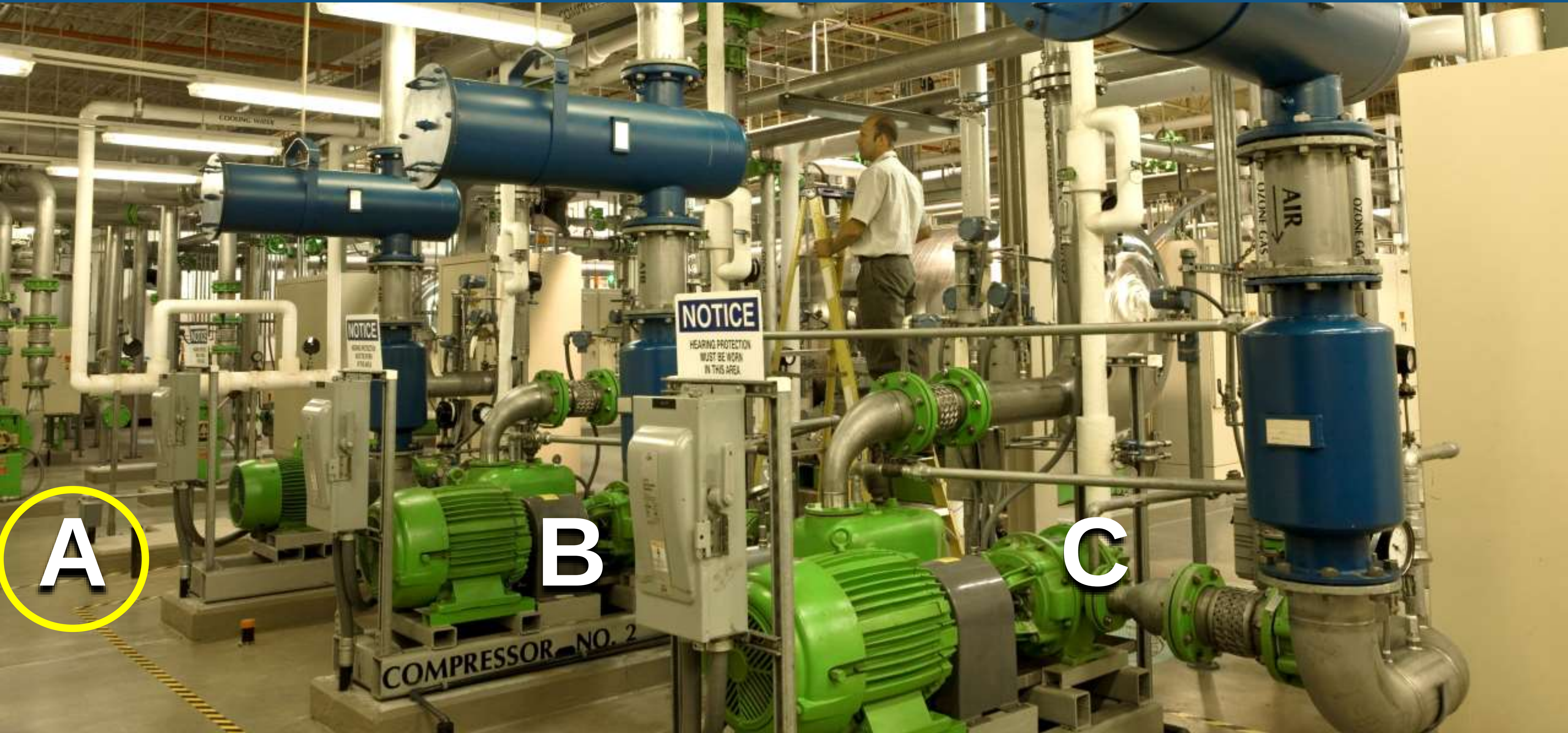
MathWorks

Listen carefully. Which compressor has a faulty bearing?




Listen carefully. Which compressor has a faulty bearing?
A
B
C
NOTICE
HEARING PROTECTION MUST BE WORN IN THIS AREA
COMPRESSOR NO. 2
AIR
OZONE GAS

Listen carefully. Which compressor has a faulty bearing?



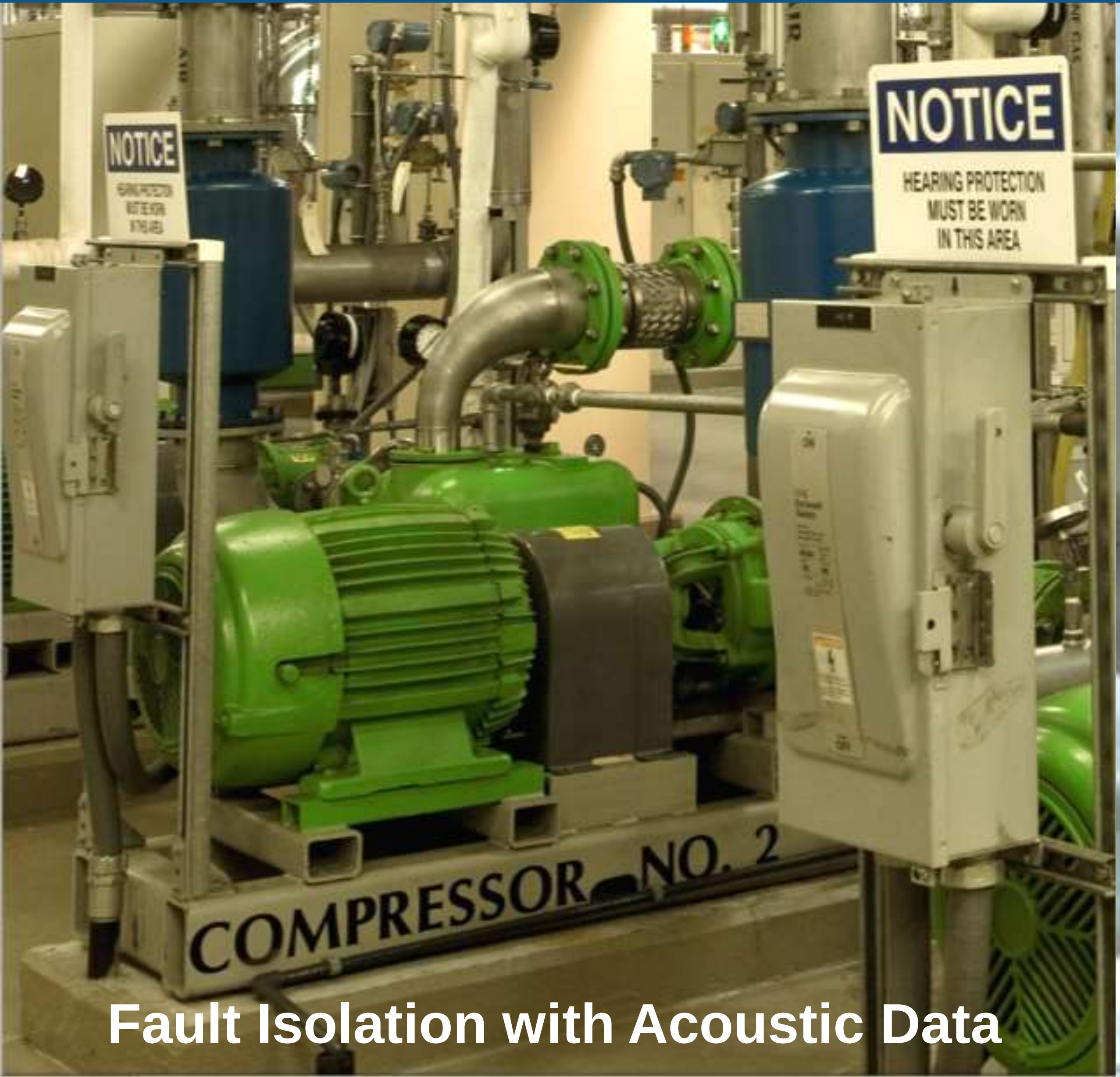
Key Takeaways for Predictive Maintenance

Small gains can yield big rewards.
Try different approaches, including deep learning.

You need AI *and* domain expertise.
MATLAB helps you do both.

MATLAB can automate your entire workflow

Journey 1:
Do you speak air compressor?



Fault Isolation with Acoustic Data

Journey 2:
Prognostics and Health management

RUL estimation using Deep Learning

Equipment Operation Manager

- Mechanical Engineer at Membrane Manufacturing*
- Responsible for a fleet of industrial machines
- New company AI initiative
- No deep learning experience

Membrane Manufacturing

*Not a real company

Predictive Maintenance Workflow

DATA PREPARATION

- Data access and preprocessing
- Simulation-based data generation
- Feature engineering

Predictive Maintenance Workflow

DATA PREPARATION

- Data access and preprocessing
- Simulation-based data generation
- Feature engineering

⟷

AI MODELING

- Model design and tuning
- Hardware-accelerated training
- Model exchange across frameworks

Predictive Maintenance Workflow

DATA PREPARATION

- Data access and preprocessing
- Simulation-based data generation
- Feature engineering

AI MODELING

- Model design and tuning
- Hardware-accelerated training
- Model exchange across frameworks

DEPLOYMENT

- Embedded Devices
- Enterprise Systems
- Edge, cloud, desktop

ITERATION & REFINEMENT

Journey 1: Do you speak air compressor?

Journey 1: Do you speak air compressor?

Goal | Data | Approach | Result

- **Fault detection:** Identify specific faults to enable maintenance staff to respond more quickly

?

Journey 1: Do you speak air compressor?

Goal | **Data** | **Approach** | **Result**

- Acoustic time series data from sensors
- Labeled faults from maintenance logs

1. Healthy
2. Leakage Inlet Valve fault
3. Leakage Outlet Valve fault
4. Non-Return Valve fault
5. Piston Ring fault
6. Flywheel fault
7. Rider Belt fault
8. Bearing fault

Journey 1: Do you speak air compressor?

Goal | Data | **Approach** | Result

Method	Validation Accuracy
Ensemble Bagged Trees	88%
Deep Neural Network	?

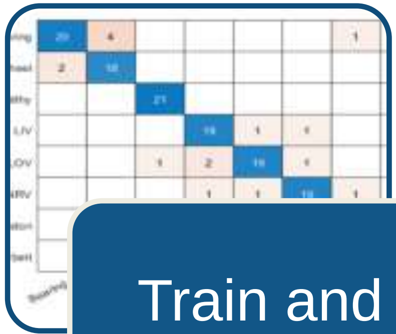
Journey 1: Do you speak air compressor?

Goal | **Data** | **Approach** | **Result**

Access data with datastore

```
audioFeatureExtractor('SampleRate
Window',hamming(windowLength,'peric
OverlapLength',overlapLength,...
spectralCentroid',true, ...
spectralCrest',true, ...
spectralDecrease',true, ...
spectralEntropy',true,...
spectralFlatness',true,...
```

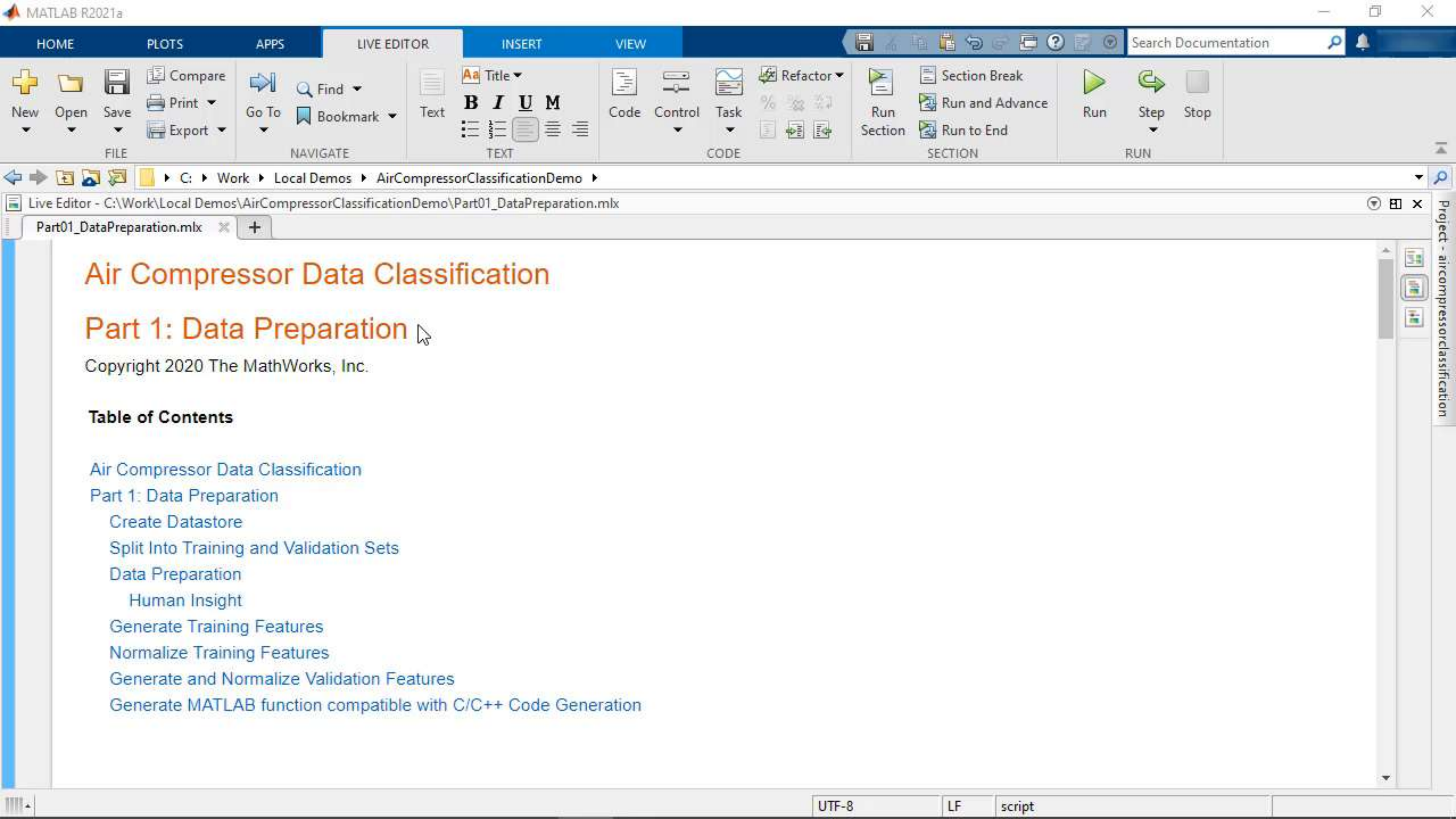
Extract features with Audio Toolbox



Train and validate LSTM

```
d extractFeatures_init
onfig.linearSpectrum.N
or (int i = 0; i < 257
 config.OneSidedSpectr
+ 1U);
 config.linearSpectrum
```

Generate C code for edge deployment

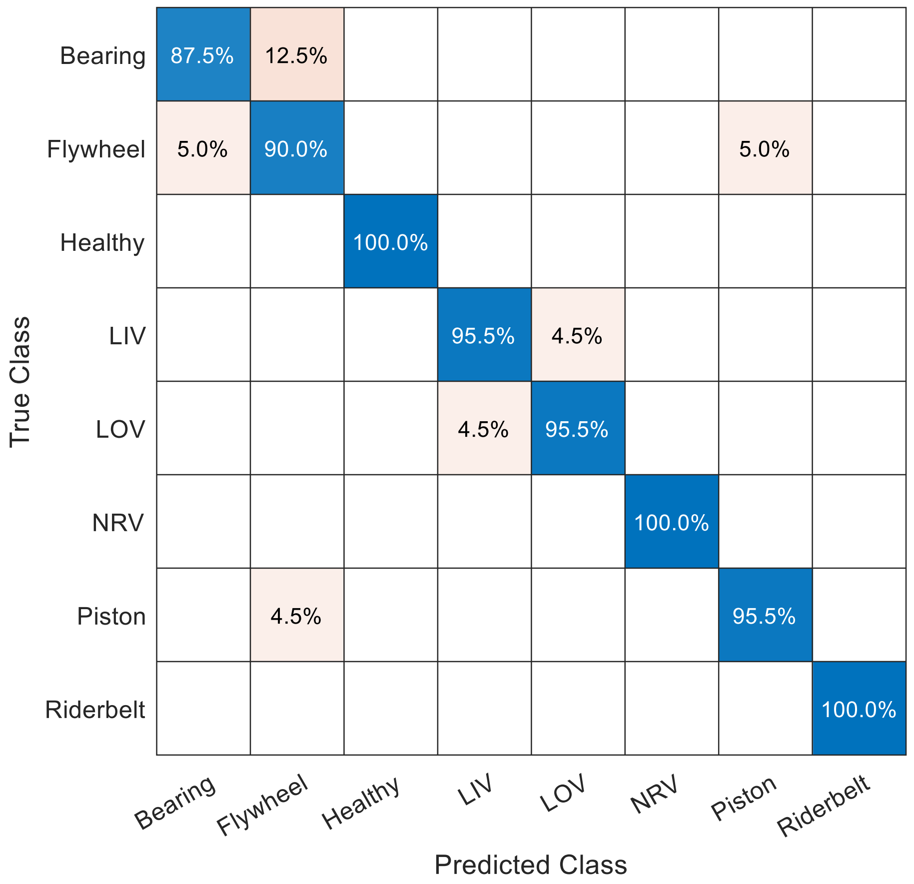

MATLAB R2021a
HOME
PLOTS
APPS
LIVE EDITOR
INSERT
VIEW
Search Documentation
New
Open
Save
Compare
Print
Export
FILE
Go To
Find
Bookmark
NAVIGATE
Text
Title
TEXT
Code
Control
Task
Refactor
CODE
Run Section
Section Break
Run and Advance
Run to End
SECTION
Run
Step
Stop
RUN
C: ▸ Work ▸ Local Demos ▸ AirCompressorClassificationDemo ▸
Live Editor - C:\Work\Local Demos\AirCompressorClassificationDemo\Part01_DataPreparation.mlx
Part01_DataPreparation.mlx
Air Compressor Data Classification
Part 1: Data Preparation
Copyright 2020 The MathWorks, Inc.
Table of Contents
Air Compressor Data Classification
Part 1: Data Preparation
Create Datastore
Split Into Training and Validation Sets
Data Preparation
Human Insight
Generate Training Features
Normalize Training Features
Generate and Normalize Validation Features
Generate MATLAB function compatible with C/C++ Code Generation
Project - aircompressorclassification
UTF-8
LF
script

Journey 1: Do you speak air compressor?

Goal | Data | Approach | **Result**

- Successfully identified faults with 95% validation accuracy

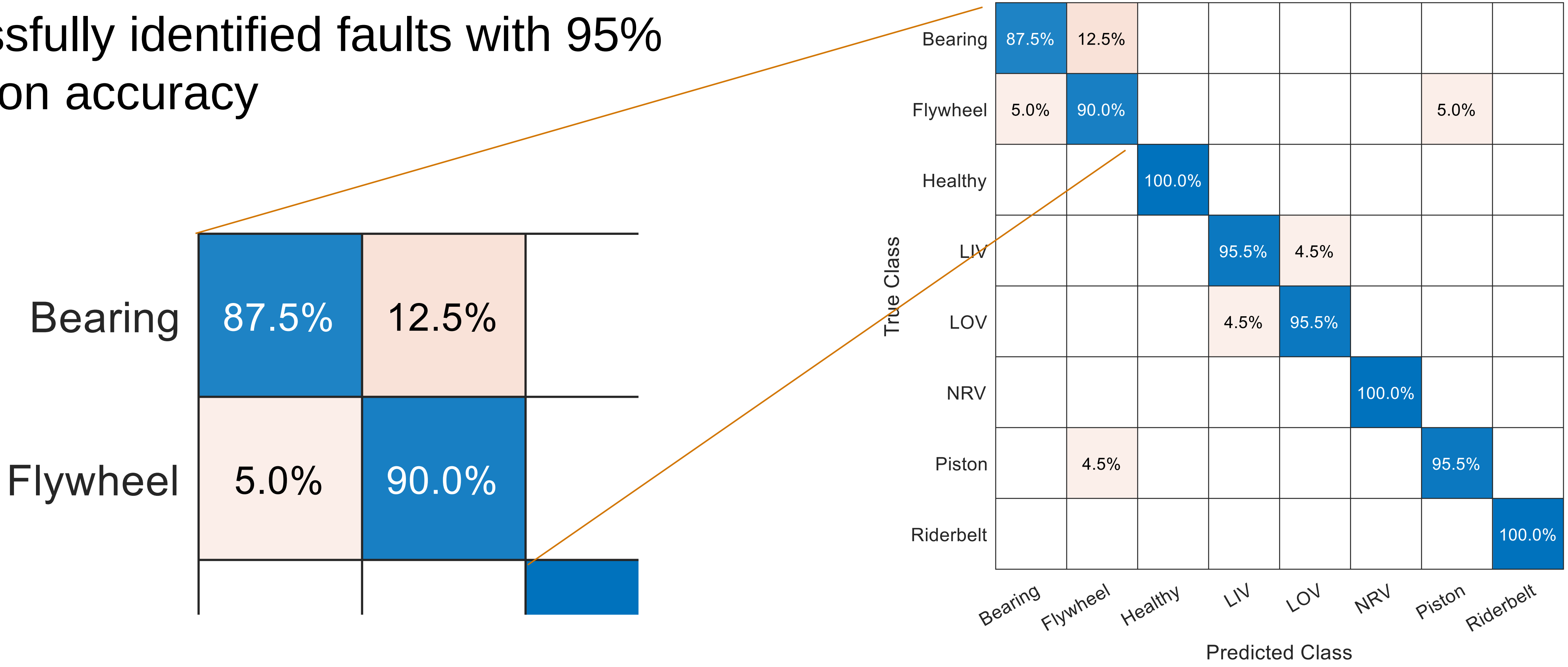
Method	Validation Accuracy
Ensemble Bagged Trees	88%
Deep Neural Network	95%



Journey 1: Do you speak air compressor?

Goal | **Data** | **Approach** | **Result**

- Successfully identified faults with 95% validation accuracy



Journey 1: Do you speak air compressor?

Goal | Data | Approach | **Result**

Poll: How could we improve the results?

- ❑ Collect more data
- ❑ Tune network hyperparameters
- ❑ Try a different feature set
- ❑ Try a different algorithm
- ❑ Buy more GPUs

Journey 1: Do you speak air compressor?

DATA PREPARATION

- Data access and preprocessing
- Simulation-based data generation
- Feature engineering

AI MODELING

- Model design and tuning
- Hardware-accelerated training
- Model exchange across frameworks

DEPLOYMENT

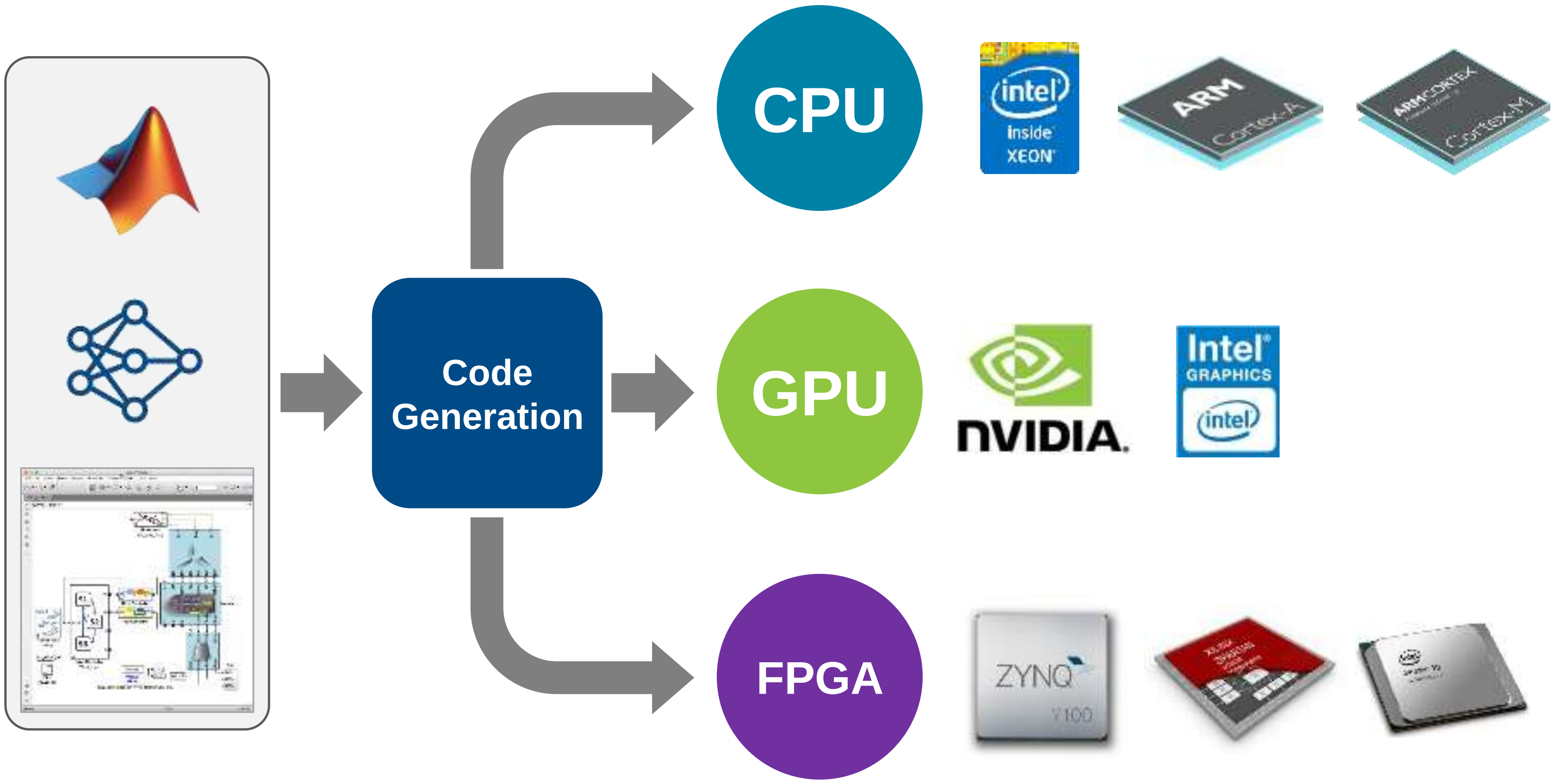
- Embedded Devices
- Enterprise Systems
- Edge, cloud, desktop

ITERATION & REFINEMENT

Journey 1: Do you speak air compressor?

DEPLOYMENT

- Embedded Devices
- Enterprise Systems
- Edge, cloud, desktop



Journey 1: Do you speak air compressor?

Goal | **Data** | **Approach** | **Result**

- What's Next?

MATLAB EXPO

MATLAB을 활용한 임베디드 및 프로덕션 시스템으로의 AI 배포

Journey 1: Do you speak air compressor?

Goal | Data | Approach | **Result**

Poll: How could we improve the results?

- [x] Collect more data
- [x] Tune network hyperparameters
- [x] Try a different feature set
- [x] Try a different algorithm
- [ ] Buy more GPUs

Journey 2: RUL estimation using Deep Learning

Journey 2: RUL estimation using Deep Learning

Goal | Data | Approach | Result

- **Predict Remaining Useful Life(RUL) of engines by CNN**

- Run to failure 100 sequence data
- No prior knowledge of machine health prognostics and signal processing

Journey 2: RUL estimation using Deep Learning

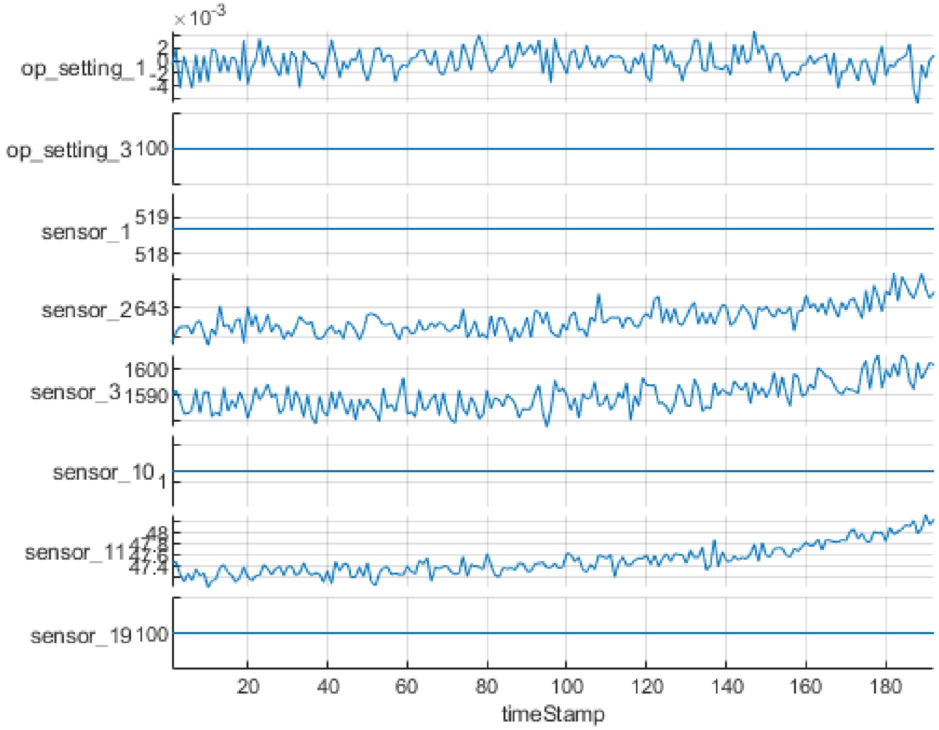
Goal | **Data** | **Approach** | **Result**

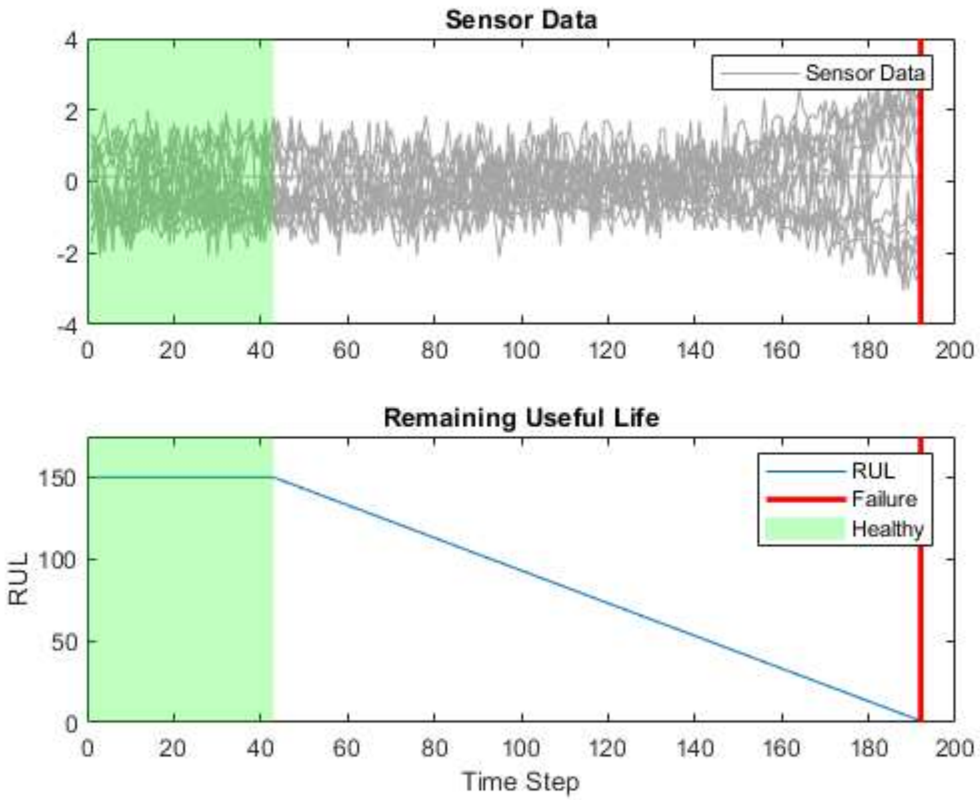
- 100 engine with 21 sensor data

- Each row is a snapshot of data taken during a single operational cycle, and each column represents a different variable:
- Column 1: Unit number
- Column 2: Time-stamp
- Columns 3–5: Operational settings
- Columns 6–26: Sensor measurements 1–21

Journey 2: RUL estimation using Deep Learning

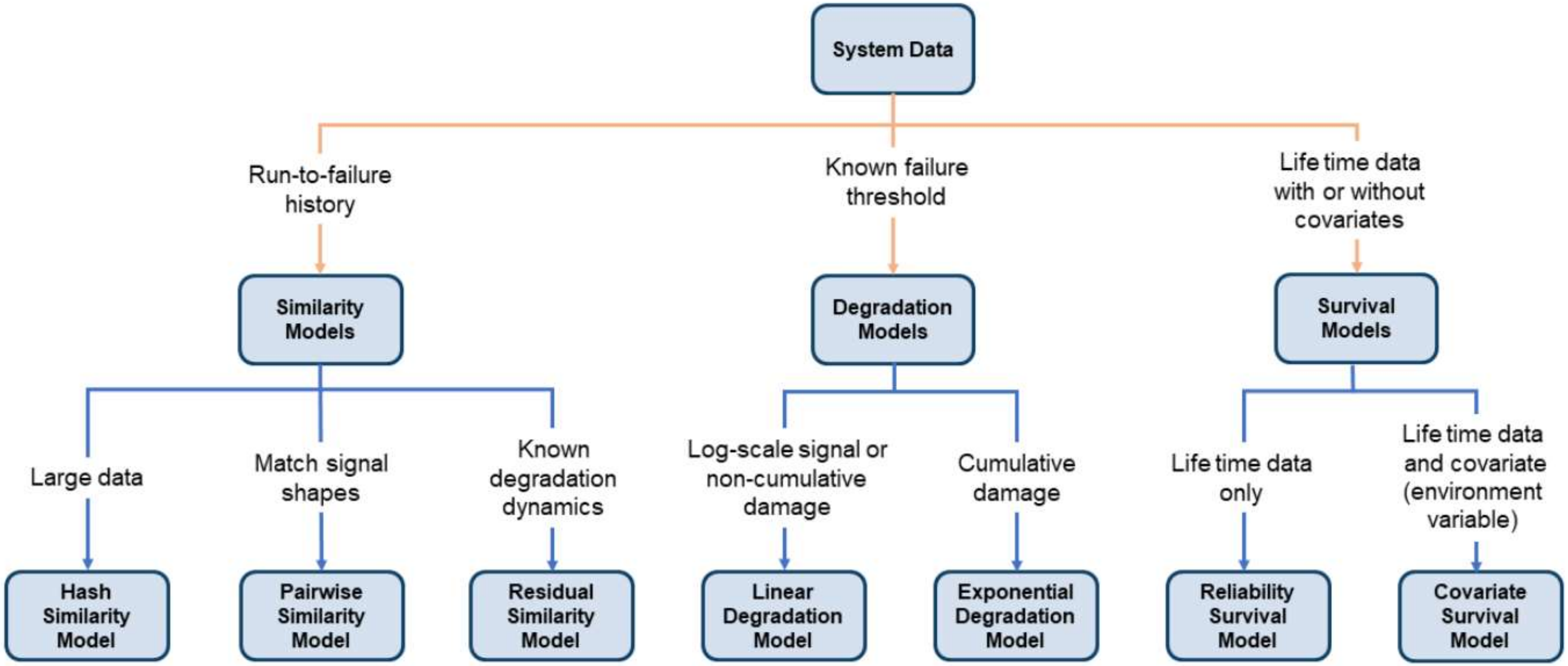
Goal | **Data** | **Approach** | **Result**





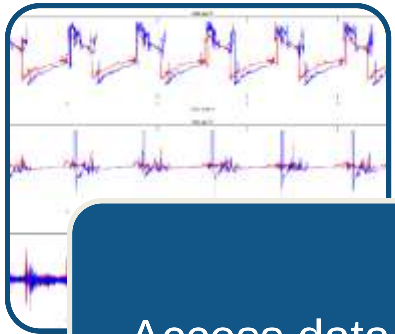
Journey 2: RUL estimation using Deep Learning

Goal | Data | **Approach** | Result

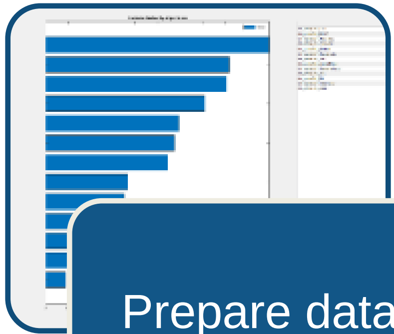


Journey 2: RUL estimation using Deep Learning

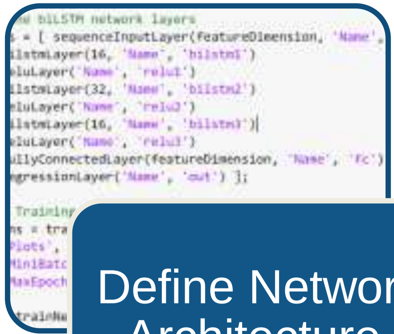
Goal | **Data** | **Approach** | **Result**



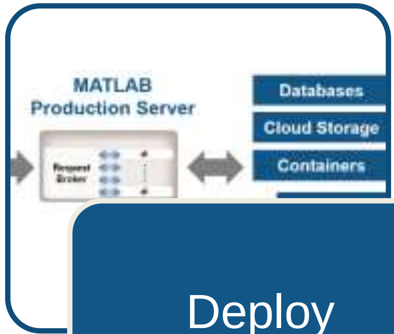
Access data from files



Prepare data for CNN Training



Define Network Architecture, Train the Network

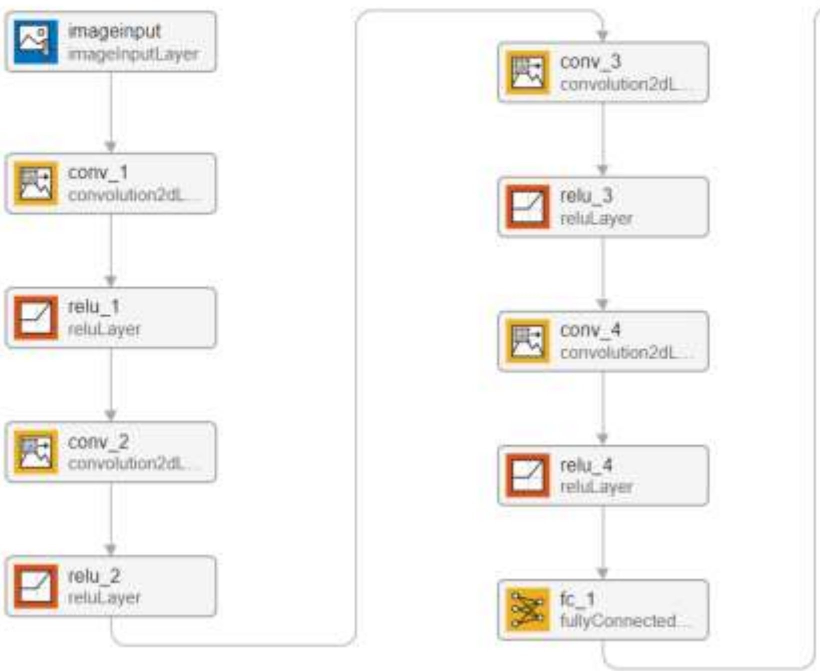


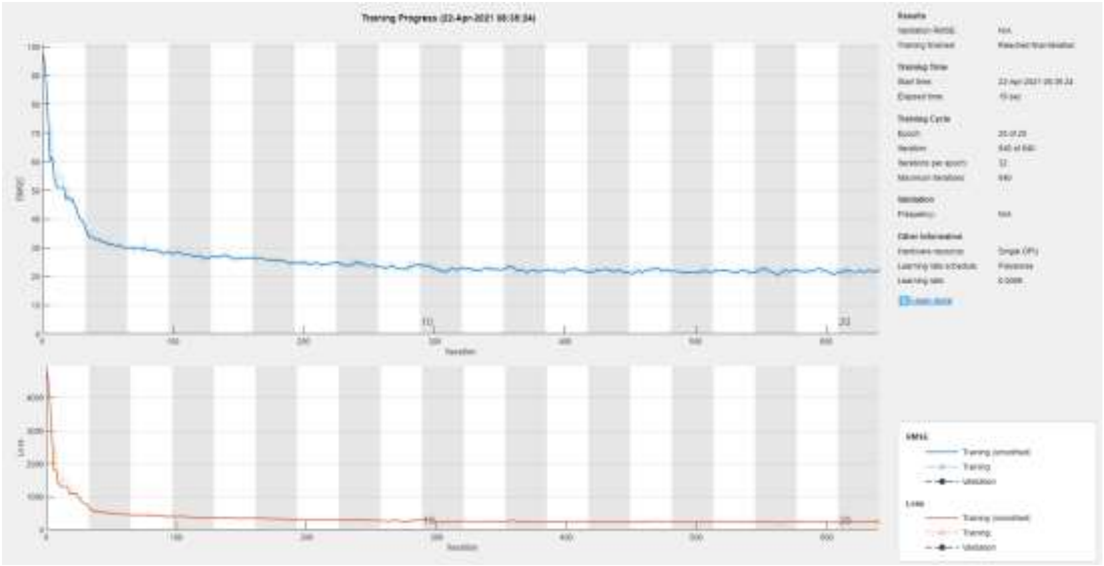
Deploy algorithms to the cloud

Journey 2: RUL estimation using Deep Learning

Goal | Data | **Approach** | Result

Deep Convolutional Neural Network





Remaining Useful Life Estimation using Convolutional Neural Network

This example shows how to predict the remaining useful life (RUL) of engines by using deep convolutional neural networks (CNN) [1]. The advantage of a deep learning approach is that there is no need for manual feature extraction or feature selection for your model to predict RUL. Furthermore, prior knowledge of machine health prognostics and signal processing is not required for developing a deep learning based RUL prediction model.

Download Dataset

This example uses the Turbofan Engine Degradation Simulation Dataset (C-MAPSS) [2]. The ZIP-file contains run-to-failure time-series data for four different sets (namely FD001, FD002, FD003, FD004) simulated under different combinations of operational conditions and fault modes.

This example uses only the FD001 dataset which is further divided into training and test subsets. The training subset contains simulated time series data for 100 engines. Each engine has several sensors whose values are recorded at a given instance in a continuous process. Hence the sequence of recorded data varies in length and corresponds to a full run-to-failure (RTF) instance. The test subset contains 100 partial sequences and corresponding values of the remaining useful life at the end of each sequence.

Download the Turbofan Engine Degradation Simulation dataset to a file named "CMAPSSData.zip" and unzip it to a folder called `"data"` in the current directory.

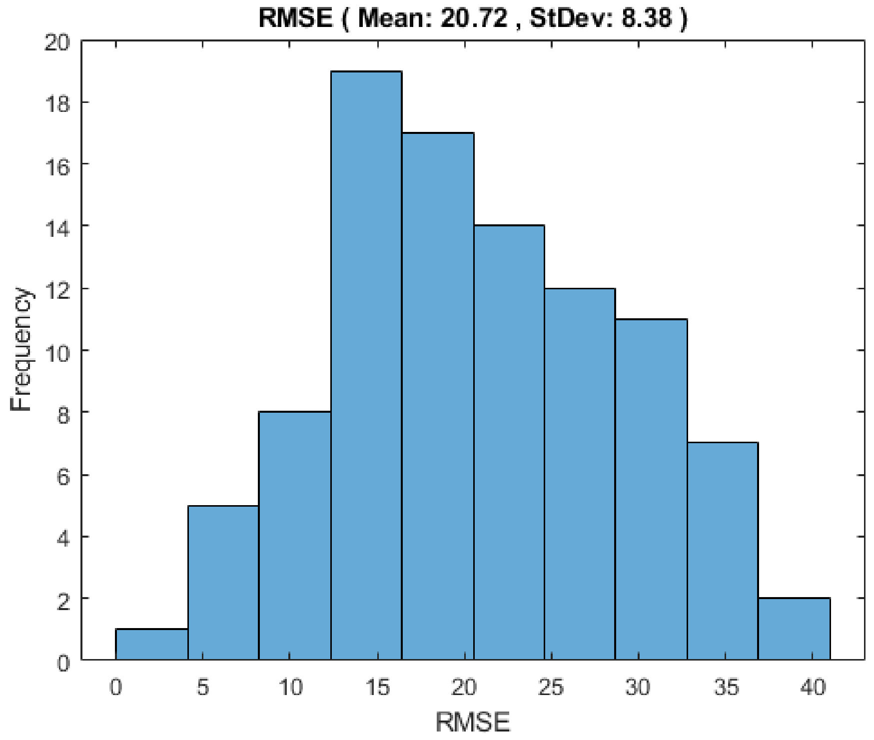
```
% filename = "CMAPSSData.zip";
% if ~exist(filename,'file')
%     url = "https://ti.arc.nasa.gov/c/6/";
%     websave(filename,url);
% end
%
dataFolder = "data";
% if ~exist(dataFolder,'dir')
%     mkdir(dataFolder);
% end
% unzip(filename,dataFolder)
```

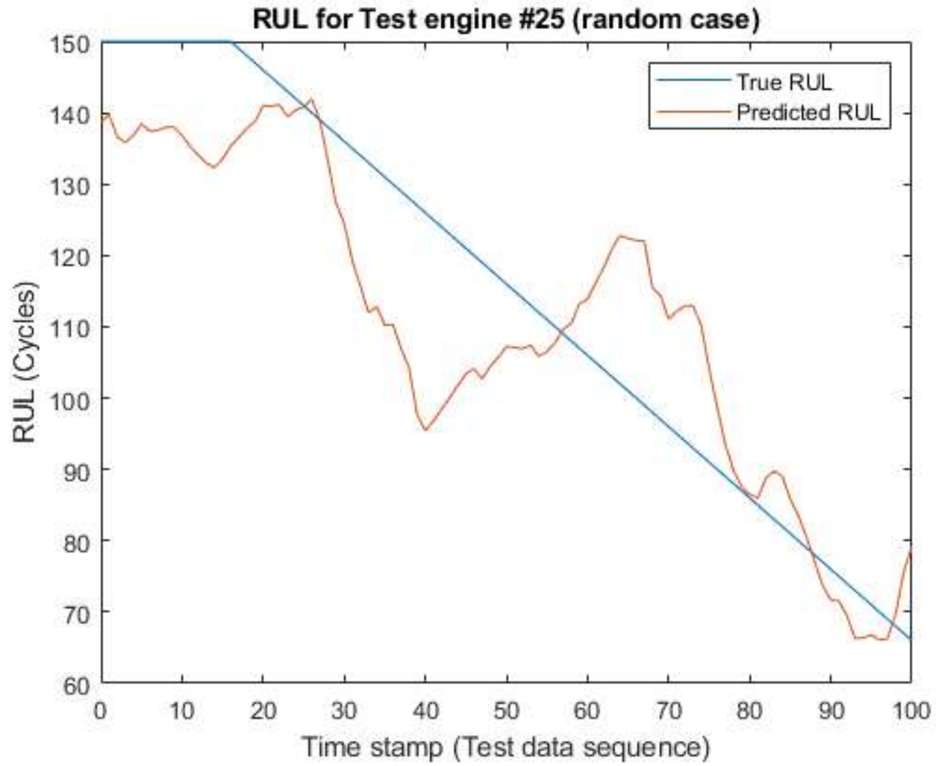
The data folder now contains text files with 26 columns of numbers, separated by spaces. Each row is a snapshot of data taken during a single operational cycle, and each column represents a different variable:

- Column 1: Unit number
- Column 2: Time stamp

Journey 2: RUL estimation using Deep Learning

Goal | Data | Approach | **Result**





Journey 2: RUL estimation using Deep Learning

Goal | Data | Approach | **Result**

- What's Next?

MATLAB EXPO

MATLAB과 Simulink를 활용하여 지속적 환경에 통합
(CI: Continuous Integration)하는 방법

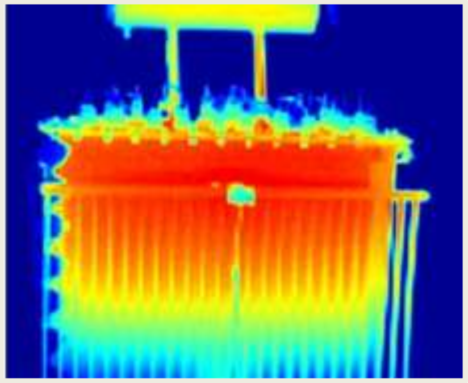
Six Months Later

- Increased uptime by 10%
- Want to expand to entire fleet, multiple locations
- Next project: Deploy Embedded AI model and Enterprise System
- Got a promotion! ☺

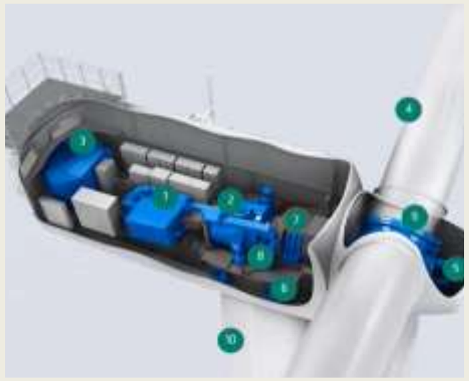
Membrane Manufacturing

Companies are succeeding with MATLAB for Predictive Maintenance

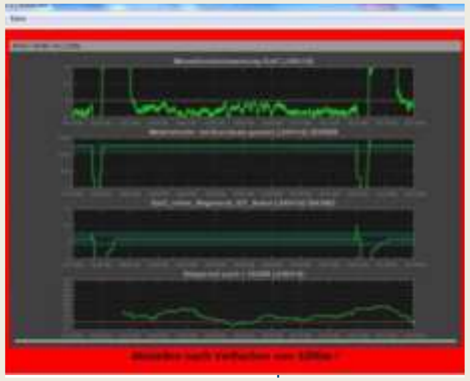
Airbus detects defects in aircraft pipes with semantic segmentation



Siemens develops health monitoring system for distribution transformers



RWE Renewables detects anomalies in wind turbine bearings using neural networks



Mondi develops and deploys algorithms to predict plastic production machine failures

LG Energy Solution used Deep Learning for Predictive Maintenance on industrial cutter

Challenge

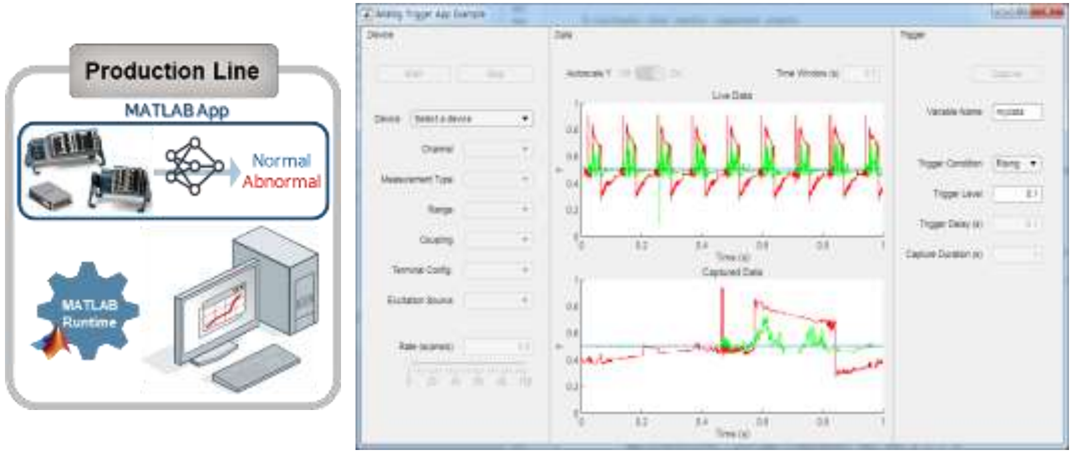
Maintenance of equipment in the factory also depends on the site engineer’s opinion, and sometimes those are a bit conservative

Solution

Developed a condition monitoring system and deployed standalone executable which can acquire raw data from NI device directly, make a prediction and display the result in GUI

Advantages of using MATLAB and Simulink

- Interactive Apps for generating features and training various AI models
- Capabilities of entire workflow from data acquisition to deployment
- Leveraged MathWorks engineer’s support for fast prototyping



Condition monitoring system using Deep Learning

“3 advantages of MATLAB that lead our project to success: App-based AI development workflow, compatibility with 3rd party hardware and short test cycle with rapid prototyping.”

Junghoon Lee, ***LG Energy Solution***

>> [MATLAB for deep learning]

Korea Institute of Energy Research uses MATLAB for Wind Turbine Health Monitoring System

Challenge

Develop Wind Turbine Predictive Maintenance algorithm within limited sensor data, Lack of experience on Industrial AI, real time monitoring solution.

Solution

Use MATLAB to develop, train, and evaluate a variety of supervised machine learning and deep learning diagnostic model.

Results

- Data aggregating, pre-processing from edge device
- Correlation analysis for 3K component based on 8K sensor data
- App Designer is great environment for monitoring system
- AutoML is easy to optimize diagnostic model

KIER Wind Turbine Monitoring System

"Working in MATLAB, we developed a diagnostics model as a proof of concept. Despite having little previous experience with AI, within limited budget and timebound, we completed a prototype capable of detecting failure with over 90~95% accuracy."

- Jung Chul, Choi, Korea Institute of Energy Research

>> [KIER Customer Presentation]

Key Takeaways for Predictive Maintenance

Small gains can yield big rewards.
Try different approaches, including deep learning.

You need AI *and* domain expertise.
MATLAB helps you do both.

MATLAB can automate your entire
Predictive Maintenance workflow

MATLAB EXPO

2021

Thank you

MathWorks