Machine Learning for Motor Control

Brenda Zhuang, PhD.
MathWorks

BREAK THROUGH TECH

August 2024

Welcome back for our meeting!

Brenda Zhuang (she/her)
MathWorks
Consulting Engineer
bzhuang@mathworks.com

AI Studio Challenge Project Overview

CHALLENGE SUMMARY

Motor control is one of the core skillsets in robotics and electrification areas which are becoming more and more widely used in the industry. Currently, many industrial motor applications are driven by classical and robust control-based methods. In this project, you will implement **Machine Learning-based motor control methods** as an alternate pathway to overcome the real-world challenges.

Project Goals and Outcome

YOUR TEAM'S OBJECTIVE

Conventional control approaches are effective when the system can be modelled predictably. It can be difficult to predict system nonlinearities due to motor parameter changes caused by aging and temperature variation.

- Define application-specific pipeline: systems, data, controls and simulation
- Study machine learning models for classification and regression tasks
- Deploy ML models in close-loop systems (software and hardware)
- * Use of reinforcement learning in motor control applications

DESIRED OUTCOMES

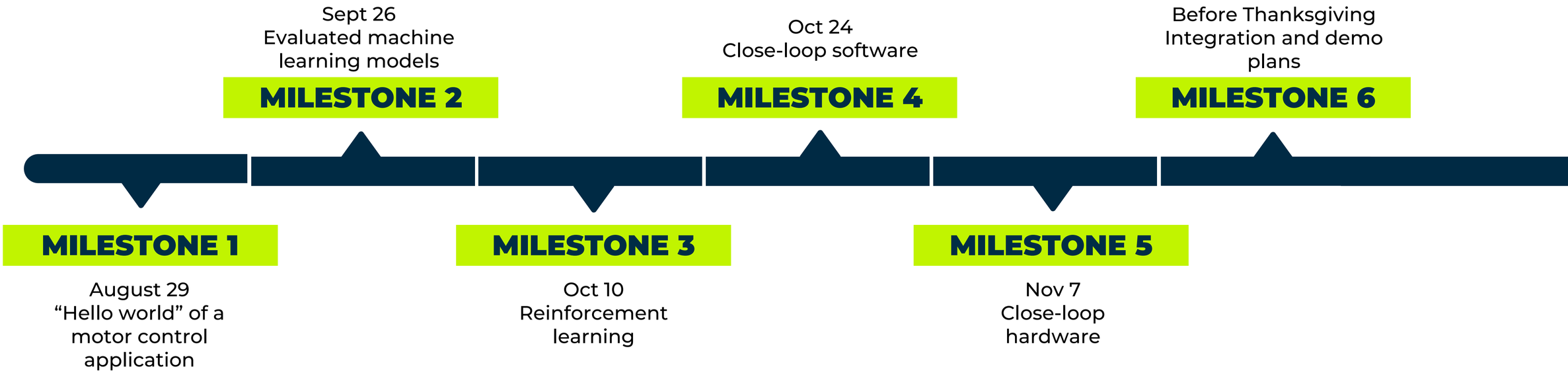
Develop and evaluate workflows that demonstrates controller design and optimization using classical control theory and machine learning-based approaches.

Explore multiple ML approaches

- Start with supervised learning and simpler machine learning models
- Reinforcement learning (based on deep learning approach) is the state-of-art approach
- Learn about classic methods

Project milestones and timeline

These are the milestones for your Challenge Project. They are roughly aligned to the CRISP-DM process you learned about in your ML Foundations course.



Let’s take a closer look at the milestone 2

Data

System

Training

Feature selection

Validation

Trained Models

What can be traditional methods to identify fault in a rotor in the motor?

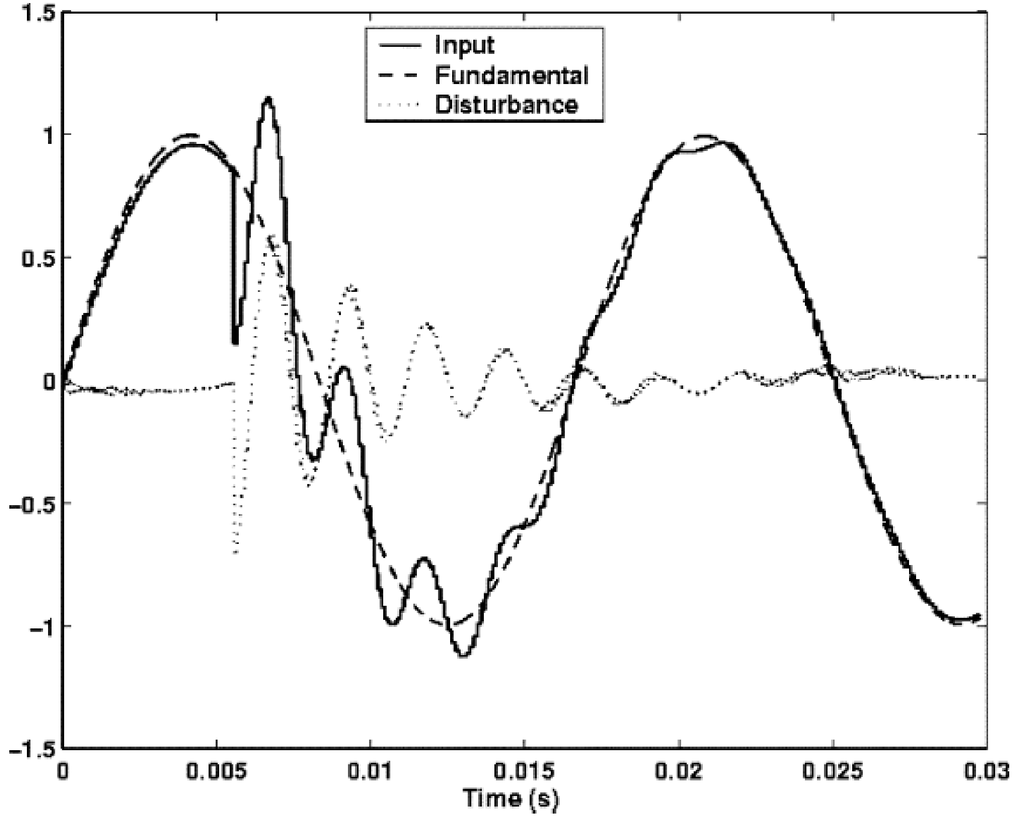


Fig. 5. Illustration of the performance of the algorithm in the detection of a disturbance imposed on a sinusoidal signal.

https://ieeexplore.ieee.org/document/1396092

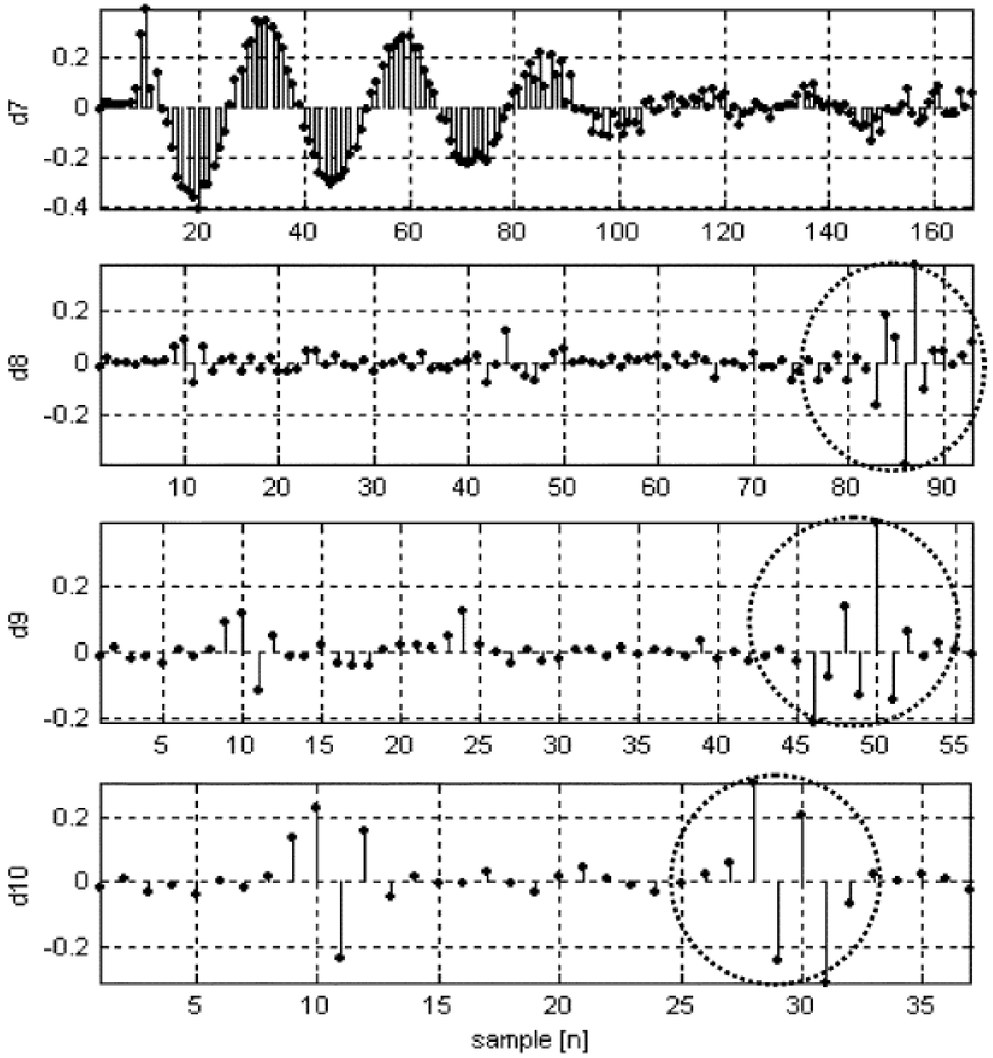
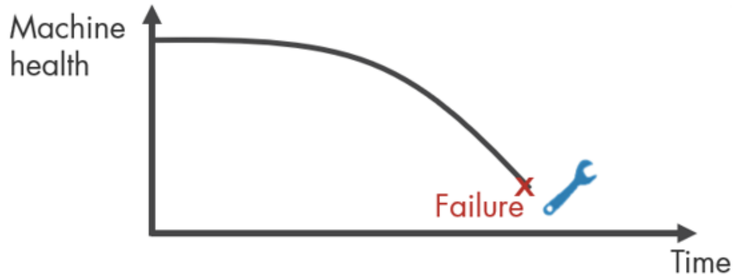


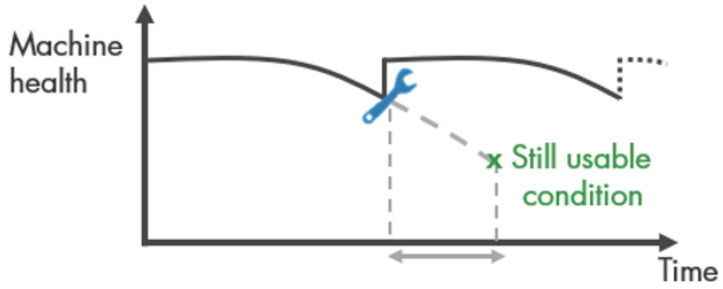
Fig. 10. Wavelet decomposition levels d7-d10 of a fully loaded damaged machine.

What Makes Maintenance “Predictive”?

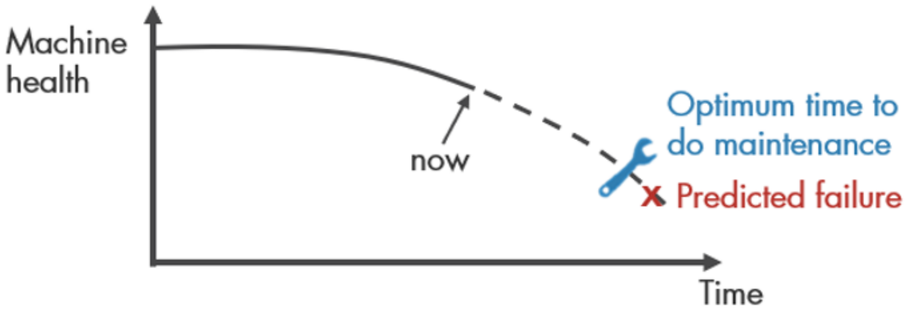
- **Reactive** – Do maintenance once there’s a problem
  - Problem: unexpected failures can be expensive and potentially dangerous



- **Scheduled** – Do maintenance at a regular rate
  - Problem: unnecessary maintenance can be wasteful; may not eliminate all failures



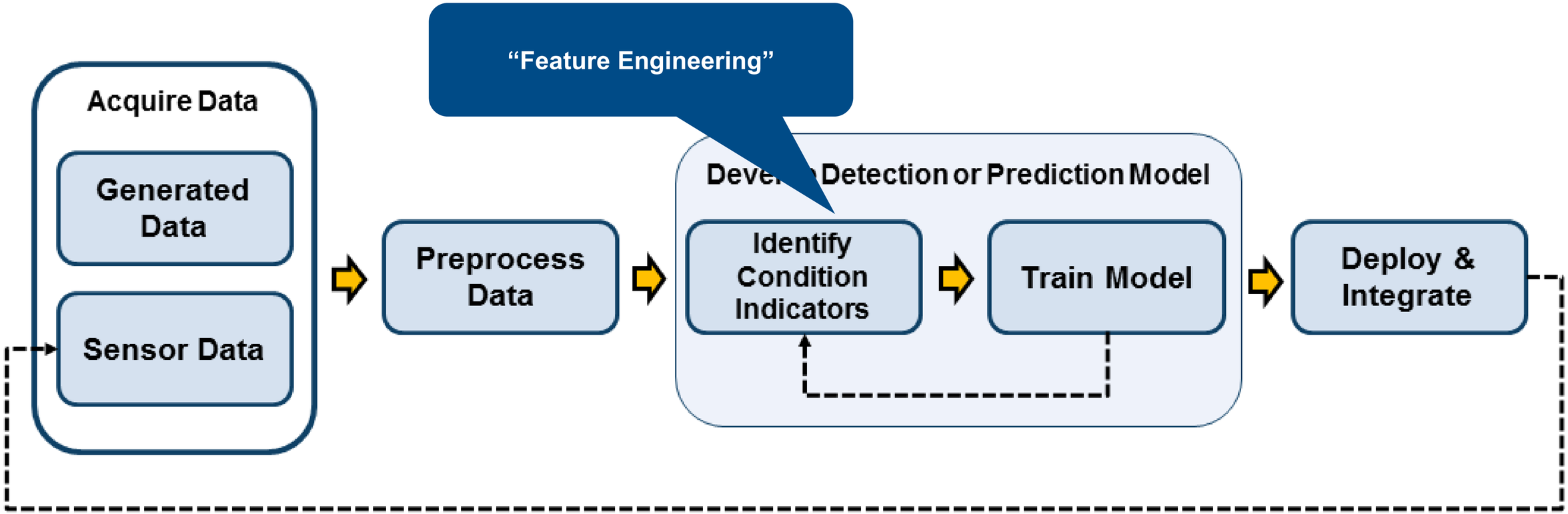
- **Predictive** – Forecast when problems will arise
  - Problem: difficult to make accurate forecasts for complex equipment



What does a predictive maintenance algorithm do?

Is my machine operating normally?	**Anomaly Detection**	**I need help.**
Why is my machine behaving abnormally?	**Fault Detection (Diagnostics)**	**One of my cylinders is blocked.**
How much longer can I operate my machine?	**Remaining Useful Life Estimation (Prognostics)**	**I will shut down your line in 15 hours.**

Predictive Maintenance Algorithm Development Workflow



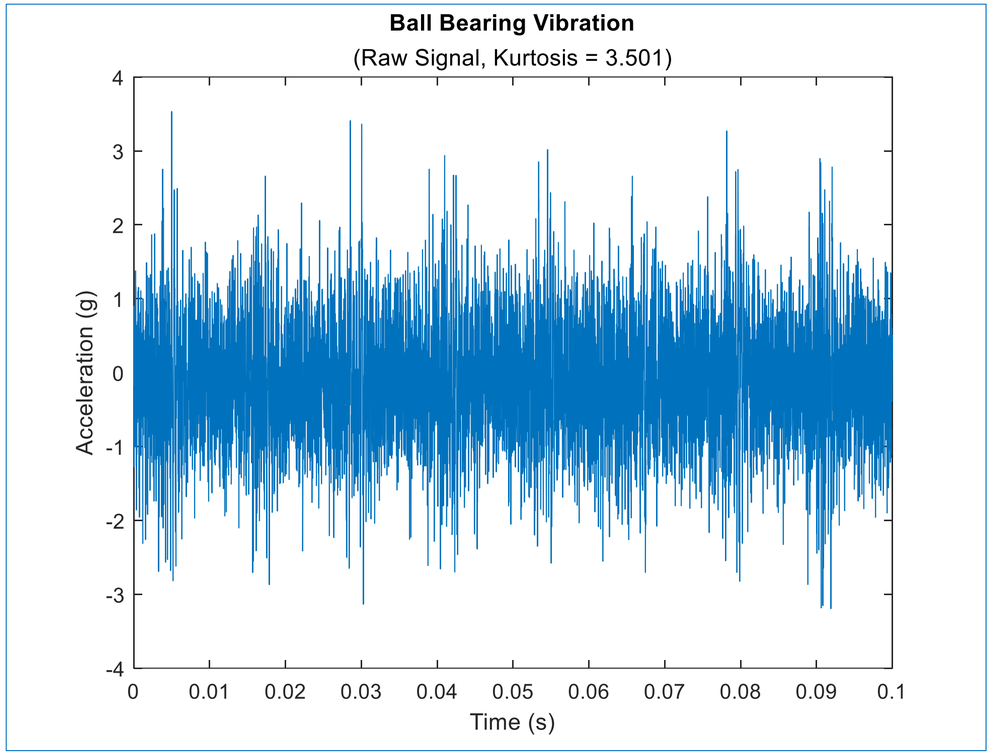
Engineering systems produce a lot of rich sensor data

- Current, voltage, magnetic field
- Position, orientation, proximity
- Pressure, flow rate
- Gas, smoke, air quality
- Image, sound

- Vibration, acceleration
- Angular position & velocity
- Force, torque
- Temperature
- Humidity

Sensors capture data on condition of engineering systems, but …

- Useful information (features) is often hidden in raw signals



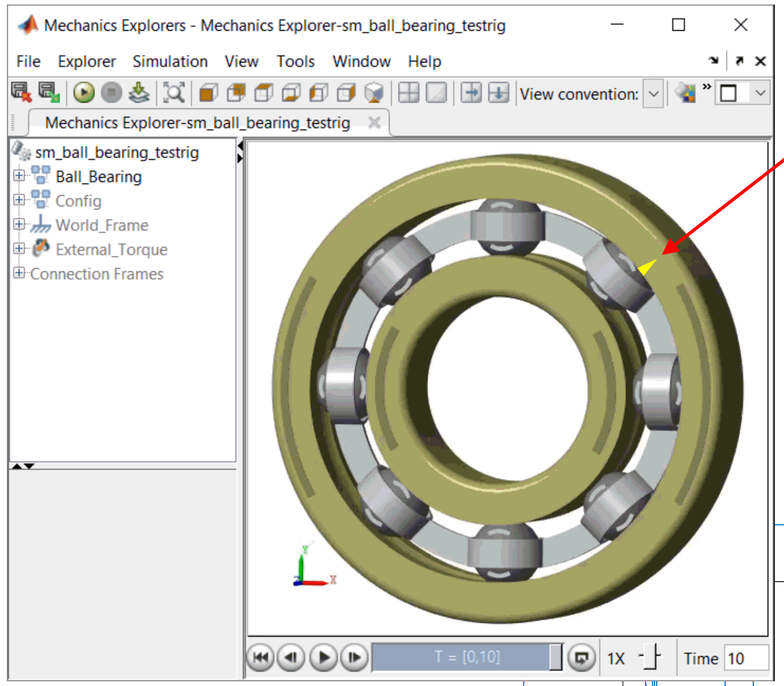
- Simple signal statistics may not produce meaningful features

Data Statistics for: Raw Data

Select statistics to display on the figure:

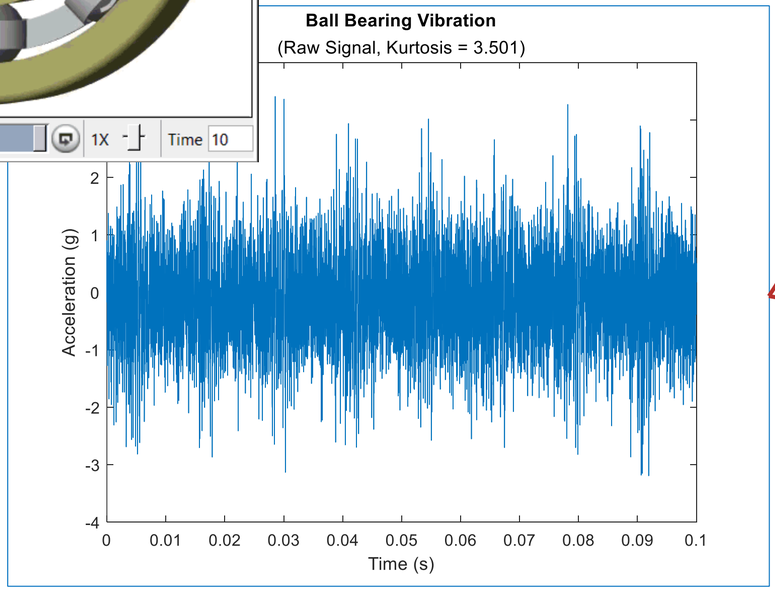
	X		Y	
min	0	☐	-4.401	☐
max	6	☐	4.349	☐
mean	3	☐	-0.1172	☐
median	3	☐	-0.1228	☐
mode	0	☐	-0.4952	☐
std	1.732	☐	0.8145	☐
range	6		8.75	

Better features can be extracted when you leverage application-specific characteristics

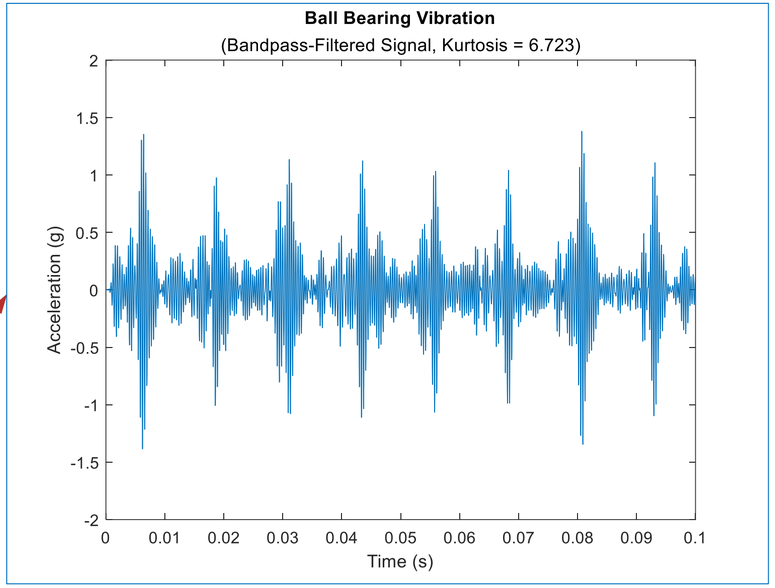


Outer race crack

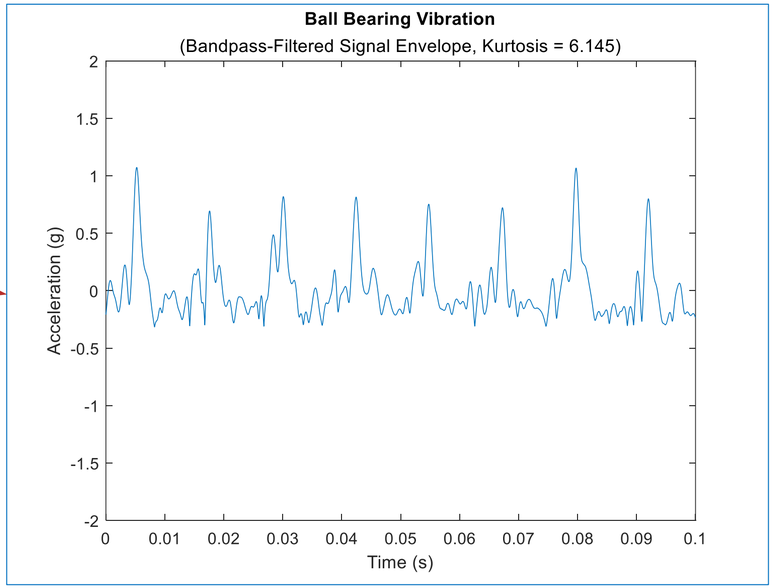
$$f_o = \frac{N_b}{2} f_r \left(1 - \frac{d_b}{d_p} \cos(\beta)\right)$$



Bandpass Filtering

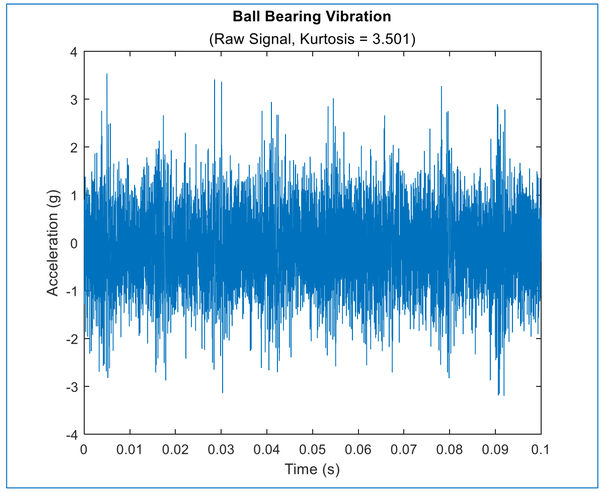


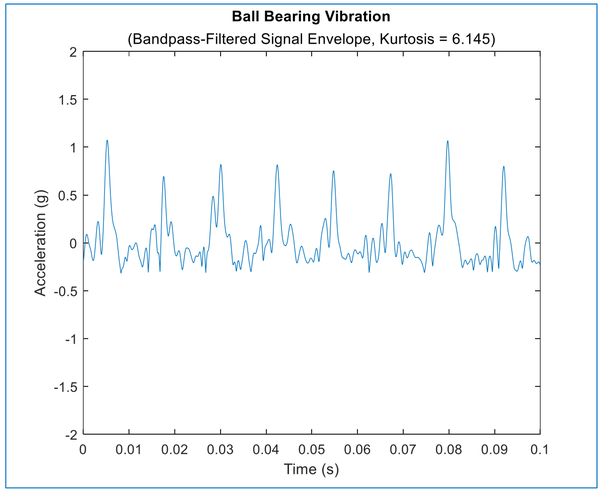
Signal Envelope

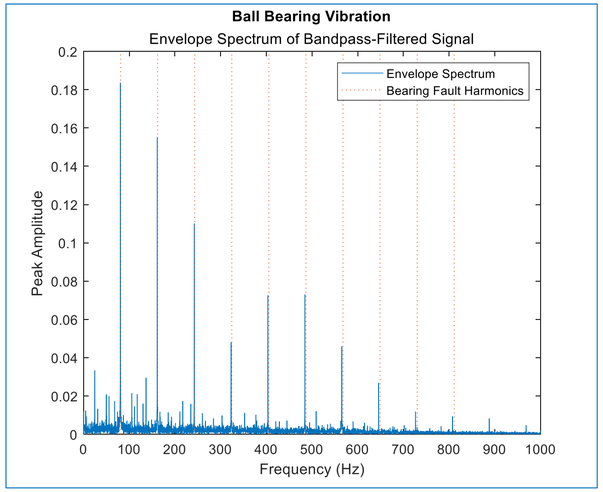


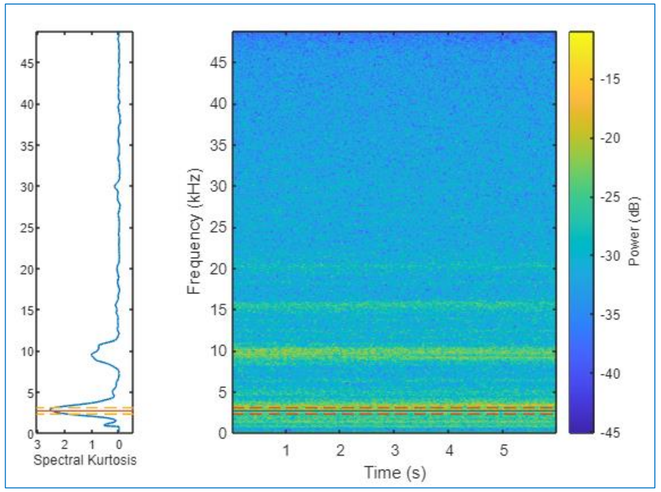
Better features can be extracted when you apply domain expertise

- Not just time-domain data and simple statistical features
- Features from other **signal domains** often make better condition indicators
  - Signal envelope (demodulation)
  - Frequency-domain (power spectrum)
  - Time-frequency domain
  - Time-synchronous averaging (rotating machinery)





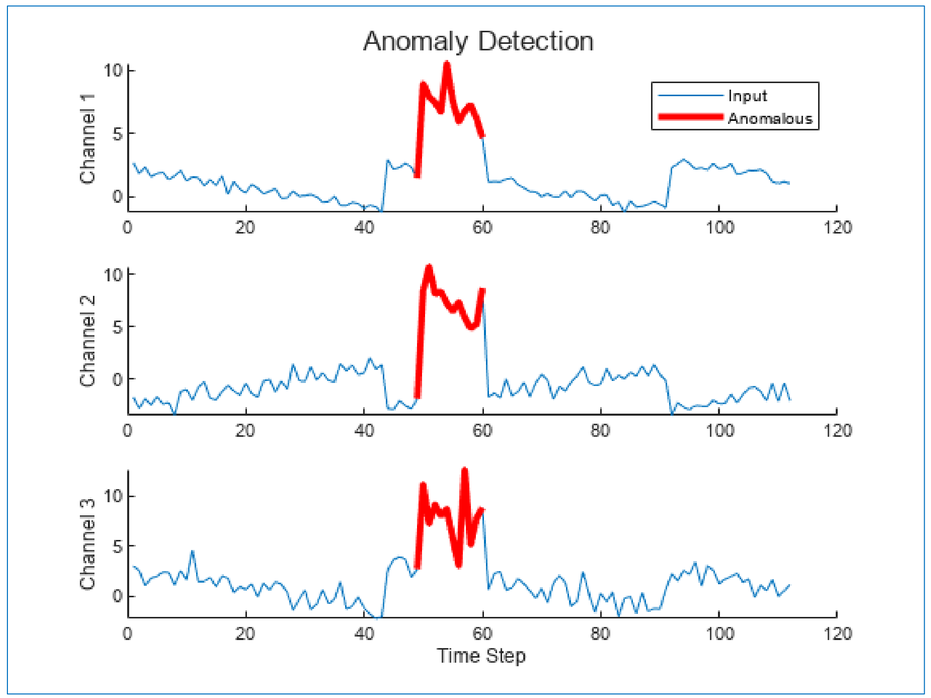




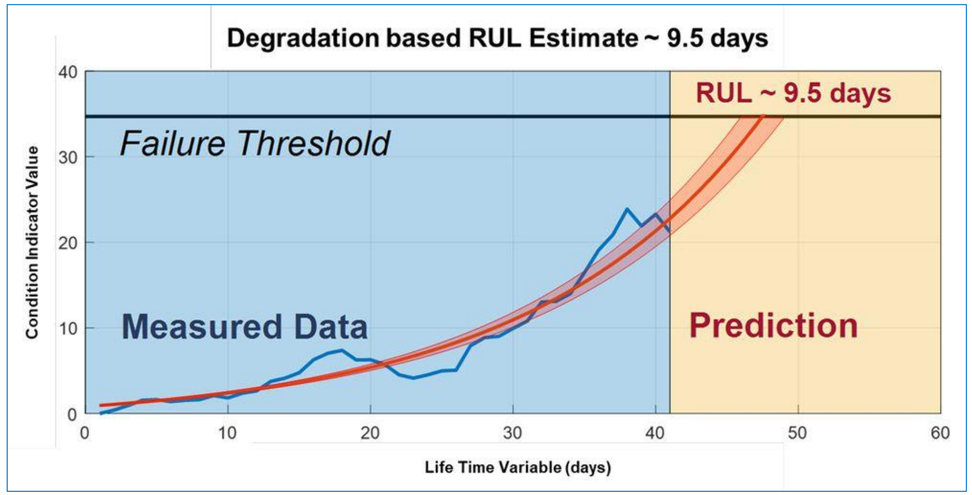
Raw Signal (time-domain) → Signal Envelope (demodulation) → Power Spectrum (frequency-domain) → Spectral Kurtosis (time-frequency-domain)

Effective features improve performance of predictive maintenance algorithms

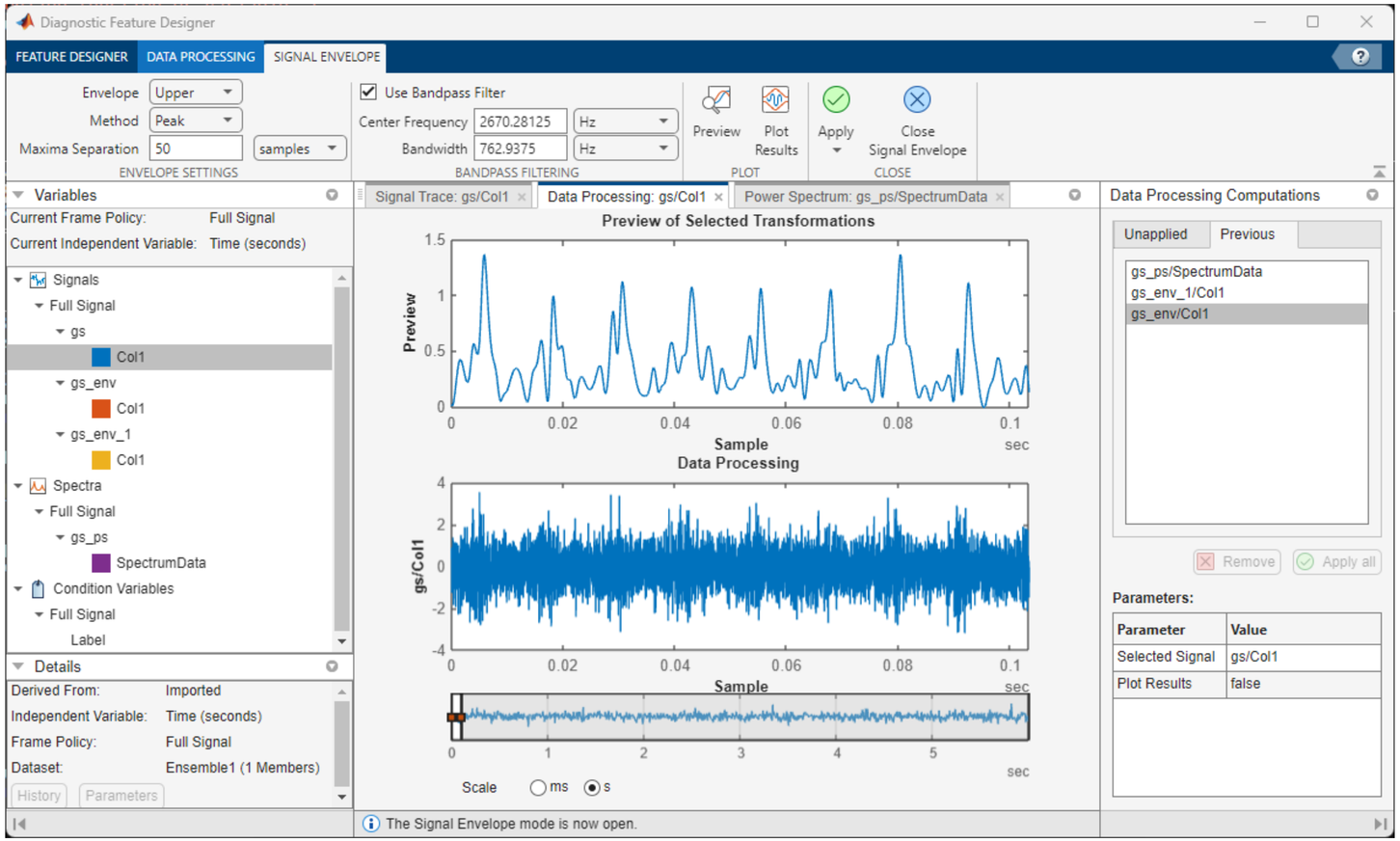
- Anomaly detection
  - Changepoint detection
  - Thresholding
  - Autoencoders



- Remaining Useful Life (RUL) estimation
  - Condition indicator + threshold
  - Linear or exponential degradation model using runtime data



Feature Engineering with Diagnostic Feature Designer app



- Enables low-code feature engineering
  - MATLAB code generation
- Data processing
- Features extraction
- Feature ranking
- Application-specific features
  - Rotating machinery
  - Bearings & gears
  - Custom features (spectral, user defined)

Milestone #2 items to address:

- What questions would you ask your data?
- What features to choose?
- What pre-processing can help?
- What new models can you train?
- What new insights if we use the full dataset?
- Deep learning models?

How we'll work together this semester

Check-in meetings	• Evening hours 7:30 – 8:30
Reporting	• Weekly huddle update (agile)
Communication	• Work email with 12-hour turn-around time
Tools and platforms	• GitHub, MATLAB (optional)
Other project norms	• Summary report on milestones with contribution from every team member

Questions?

What questions do you have?

Anything I can help clarify?

What are you most excited about?

Anything you’re unsure about?