MATLAB EXPO 2019

Predictive Maintenance with MATLAB

Amit Doshi,
Senior Application Engineer – Data Analytics
MathWorks India
adoshi@mathworks.com
Agenda:

1. What is Predictive Maintenance? Who is benefiting by doing it?

2. How can you develop a predictive maintenance algorithm using MATLAB?

3. How can you get started quickly?
Types of maintenance

- Reactive maintenance
- Preventive maintenance
- Predictive maintenance
Reactive maintenance

Machine health

Time

Failure
Reactive maintenance

Machine health

Time

Failure

Aircraft engine
Reactive maintenance

Preventive maintenance

Machine health

Failure

Time

Machine health

Time
Reactive maintenance

Preventive maintenance

Machine health

Time

Failure

Still usable condition
Reactive maintenance

Preventive maintenance

Predictive maintenance

Machine health

Time

Failure

Still usable condition

Optimum time to do maintenance

Predicted failure

MATLAB EXPO 2019
A Predictive Maintenance Algorithm Answers These Questions

- **Is my machine operating normally?**  
  **Anomaly Detection**

- **Why is my machine behaving abnormally?**  
  **Condition Monitoring**

- **How much longer can I operate my machine?**  
  **Remaining Useful Life Estimation**

For example:

- I need help.

- One of my cylinders is blocked.

- I will shut down your line in 15 hours.
Predictive Maintenance Success Stories

Pump Health Monitoring System
- Spectral analysis and filtering on binary sensor data and neural network model prediction
- More than $10 million projected savings

Online engine health monitoring
- Real-time analytics integrated with enterprise service systems
- Predict sub-system performance (oil, fuel, liftoff, mechanical health, controls

Production machinery failure warning
- Reduce waste and machine downtime
- MATLAB based HMI warns operators of potential failures
- > 200,000 € savings per year
Blowout Preventer Control System: Condition and Performance Monitoring
Mete Mutlu, John Kozicz
Transccean Inc.

Condition and Performance Monitoring of Blowout Preventer (BOP) at Transccean
Mete Mutlu, John Kozicz, Transccean, Inc.
Transocean uses MATLAB tools to transition from preventative maintenance to CPM* for a critical deep sea drilling component

*Condition and Performance Monitoring

Challenges

- Minimize unplanned downtime of the component
- Use as-close-to-real-time data for CPM
- Deploy CPM solution to components in other locations

Solution

- Monitoring data from the drill is collected in a PI archive. MATLAB is used to create an app that takes in the data, preprocesses the data, and generates a quality indicator.
- The app is deployed onto MATLAB Production Server for real-time use; results are stored back on the PI archive.

Results

Able to move to CPM for their component(s) with improved decision-making capabilities through faster access to data and quicker analytics deployment.

MATLAB EXPO 2019

Interfacing directly with data in PI gives you increased performance at the cost of having to write/deal with low-level code (“getting under the hood”). It was noted that using MATLAB parallel tools (which entails a higher-level interface) was not only easier but also it provided “hands-down” faster performance than interfacing with PI directly.
Predictive Maintenance Toolbox for Developing Algorithms

Is my machine operating normally?

Why is my machine behaving abnormally?

How much longer can I operate my machine?

Anomaly Detection

Condition Monitoring

Remaining Useful Life Estimation
Workflow for Developing a Predictive Maintenance Algorithm

Machine Learning

Acquire Data → Preprocess Data → Identify Features → Train Model → Deploy & Integrate
Agenda:

1. What is Predictive Maintenance? Who is benefiting by doing it?

2. How can you develop a predictive maintenance algorithm using MATLAB?

3. How can you get started quickly?
## Develop Predictive Maintenance Algorithm: Use cases

<table>
<thead>
<tr>
<th>Question</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is my machine operating normally?</td>
<td>Anomaly Detection</td>
</tr>
<tr>
<td>Why is my machine behaving abnormally?</td>
<td>Condition Monitoring</td>
</tr>
<tr>
<td>How much longer can I operate my machine?</td>
<td>Remaining Useful Life Estimation</td>
</tr>
</tbody>
</table>
Develop Predictive Maintenance Algorithm for -

Use case 1: Fault Classification

Condition indicators

Predictive maintenance algorithm

Healthy
Blocked Inlet
Seal
Leakage
Worn Bearing
Blocked Inlet & Worn Bearing
Seal Leakage & Worn Bearing
Seal Leakage & Blocked Inlet

Why is my machine behaving abnormally?
Condition Monitoring

Identify fault type

Determine what needs to be fixed
Develop Predictive Maintenance Algorithm for -

Use case 2: Remaining useful life

Condition indicators

Predictive maintenance algorithm

Estimate remaining useful life

Current condition

Machine health

Time

Remaining useful life

How much longer can I operate my machine?

Remaining Useful Life Estimation
Collect data using sensors

Triplex pump

healthy + faulty

Sensor data

Temperature
Flow
Pressure

*For simplification purposes, healthy and faulty operation are represented by single measurements. In a realistic scenario, there may be hundreds of measurements for both types of operations.
Acquire Data → Preprocess Data → Identify Condition Indicators → Train Model → Deploy & Integrate

Triplex pump

healthy + faulty

Collect data using sensors

Temperature
Flow
Pressure

Sensor data

Time

*Quality data→Robust Algorithms

MATLAB EXPO 2019
Predictive Maintenance Algorithm

1. Acquire Data
2. Preprocess Data
3. Identify Condition Indicators
4. Train Model
5. Deploy & Integrate

Collect data using sensors

Triplex pump

healthy + faulty

Temperature
Flow
Pressure

Sensor data

Time
Predictive Maintenance Algorithm

1. Acquire Data
2. Preprocess Data
3. Identify Condition Indicators
4. Train Model
5. Deploy & Integrate

Collect data using sensors

Triplex pump

Sensor data

Temperature
Flow
Pressure

Operating temperature: 30°F
Fluid viscosity: Low

Healthy + faulty

Operating temperature: 90°F
Fluid viscosity: High

MATLAB EXPO 2019
What if real failure data is not available?
Predictive Maintenance Algorithm

Acquire Data → Preprocess Data → Identify Condition Indicators → Train Model → Deploy & Integrate

- Sensor data
- Time

Triplex pump

Refine model

- Seal leakage
- Blocked inlet
- Worn bearing

Inject faults

Mathematical model of the triplex pump (Digital Twin)

Sensor data → Synthetic data

MATLAB EXPO 2019
240 measurements of flow and pressure with a duration of 1.2 seconds.

The measurements cover conditions where none, one, or multiple faults occur. They are stored in a table where each row is a different measurement.

```
load('savedPumpData')
pumpData
```

```
pumpData = 240x3 table

<table>
<thead>
<tr>
<th>Flow</th>
<th>Pressure</th>
<th>faultCode</th>
</tr>
</thead>
<tbody>
<tr>
<td>1201x1 time...</td>
<td>1201x1 time...</td>
<td>0</td>
</tr>
<tr>
<td>1201x1 time...</td>
<td>1201x1 time...</td>
<td>0</td>
</tr>
<tr>
<td>1201x1 time...</td>
<td>1201x1 time...</td>
<td>0</td>
</tr>
<tr>
<td>1201x1 time...</td>
<td>1201x1 time...</td>
<td>100</td>
</tr>
<tr>
<td>1201x1 time...</td>
<td>1201x1 time...</td>
<td>100</td>
</tr>
<tr>
<td>1201x1 time...</td>
<td>1201x1 time...</td>
<td>100</td>
</tr>
<tr>
<td>1201x1 time...</td>
<td>1201x1 time...</td>
<td>100</td>
</tr>
<tr>
<td>1201x1 time...</td>
<td>1201x1 time...</td>
<td>100</td>
</tr>
</tbody>
</table>

Health condition (fault code):
- Healthy (0)
- Blocked inlet (010)
- Worn bearing (001)
- Seal Leakage (100)
- Blocked inlet, worn bearing (011)
- Seal leakage, worn bearing (101)
- Seal leakage, blocked inlet (110)
- Seal leakage, blocked inlet, worn bearing (111)

Learn further: Data Ensembles
Predictive Maintenance Algorithm

**Acquire Data**
- Sensor Data
- Generated Data

**Preprocess Data**

**Identify Condition Indicators**

**Train Model**

**Deploy & Integrate**

**Challenges**

- Data clean up
  - Poorly formatted files
  - Irregularly sampled data
  - Redundant data, outliers, missing data etc.

- Data specific processing
  - Signals: Smoothing, resampling, denoising, Wavelet transforms, etc.
  - Images: Image registration, morphological filtering, deblurring, etc.
Plot time vs pressure data for all the 240 runs

```matlab
f = @(x) plot(pumpData.pressure{x,1}.Time,pumpData.pressure{x,1}.Data);
for i=1:240, f(i), end
```

Raw data

Spikes to sensor's maximum value

Offset in time
```matlab
function [dataToWrite] = preprocess(data)

tMin = seconds(0.8);

flow = data.qOut_meas{1};
flow = flow(flow.Time >= tMin,:);
flow.Time = flow.Time - flow.Time(1);
pressure = data.pOut_meas{1};
pressure = pressure(pressure.Time >= tMin,:);
pressure.Time = pressure.Time - pressure.Time(1);

% Ensure the flow and pressure is sampled at a uniform sample rate
flow = retime(flow,'regular','linear','TimeStep',seconds(1e-3));
pressure = retime(pressure,'regular','linear','TimeStep',seconds(1e-3));
```

MATLAB EXPO 2019
Predictive Maintenance Algorithm

Acquire Data
- Sensor Data
- Generated Data

Preprocess Data

Identify Condition Indicators
• What are Condition Indicators?
• Why are they important?
• How to identify significant ones?

Train Model

Deploy & Integrate
A condition indicator can be any feature that is useful -
• for distinguishing normal from faulty operation or
• for predicting remaining useful life
Signal-Based Condition Indicators

- **Time-domain features**
  - Mean
  - Standard deviation
  - Skewness
  - Root-mean square
  - Kurtosis

- **Frequency-domain features**

- **Time-frequency domain features**
Visualizing the effects of different failure conditions on pressure over time:

- **Seal Leakage**
- **Blocked Inlet**
- **Worn Bearing**
- **Seal Leakage, Blocked Inlet**
- **Seal Leakage, Worn Bearing**
- **Blocked Inlet, Worn Bearing**

Pressures are measured in bars and time in seconds. The diagram illustrates how fault severity increases over time for each condition.
Blocked Inlet

Pressure (bar)

Blocked Inlet

Healthy

MEAN

No overlap

Minimum

25th percentile

Median

75th percentile

Maximum

Minimum

25th percentile

Median

75th percentile

Maximum

Blocked Inlet

Healthy
Signal-Based Condition Indicators

- **Time-domain features**
  - Mean
  - Standard deviation
  - Skewness
  - Root-mean square
  - Kurtosis

- **Frequency-domain features**

- **Time-frequency domain features**

Learn more about Condition Indicators
Three different vibration sources:
- Bearing
- Motor shaft
- Disc

- In time-domain, we observe the combined effect of different sources of vibration.

- Using frequency-domain analysis, we can distinguish different sources of vibration.
Frequency-domain features:

- Peaks
- Peak frequencies
Signal-Based Condition Indicators

**Time-domain features**
- Mean
- Standard deviation
- Skewness
- Root-mean square
- Kurtosis

**Frequency-domain features**
- Power bandwidth
- Mean frequency
- Peak values
- Peak frequencies
- Harmonics

**Time-frequency domain features**
- Spectral entropy
- Spectral kurtosis

Learn more about Condition Indicators
Diagnostic Feature Designer App
Predictive Maintenance Toolbox R2019a

- Extract, visualize, and rank features from sensor data
- Use both statistical and dynamic modeling methods
- Work with out-of-memory data
- Explore and discover techniques without writing MATLAB code
Upon saved selection or import data to begin.
Machine Learning Workflow

Machine learning uses data and produces a program to perform a task

**Train:** Iterate till you find the best model using historical data

**Predict:** Integrate trained models into applications
Machine Learning: Types

**Condition Indicator**

- Machine Learning

**Type of Learning**

- Supervised Learning
  - Develop *predictive model* based on both input and output data
- Unsupervised Learning
  - Group and interpret data based only on input data

**Categories of Algorithms**

- Regression
- Classification
MATLAB EXPO 2019
Video showing App in action
Summary: Develop Predictive Maintenance Algorithm:

Use case 1: Fault Classification

Condition indicators

Predictive maintenance algorithm

Healthy
Blocked inlet
Seal Leakage
Worn Bearing
Blocked Inlet & Worn Bearing
Seal Leakage & Worn Bearing
Seal Leakage & Blocked inlet

Identify fault type

Determine what needs to be fixed

Acquire Data

Sensor Data
Generated Data

Preprocess Data

Identify Condition Indicators

Train Model

Deploy & Integrate

Diagnostic Feature Designer

Classification Learner App
Develop Predictive Maintenance Algorithm: Use case 2

How much longer can I operate my machine?

Condition indicators

Predictive maintenance algorithm

Estimate remaining useful life

Machine health

Current condition

Remaining useful life

Remainig Useful Life Estimation
What is RUL?

Machine deterioration profile

Condition indicator

Current condition
Failure condition

Time

Remaining useful life (RUL)

[ Number of days ]
[ Miles ]
[ Cycles ]

MATLAB EXPO 2019
RUL Estimator Models

Similarity model

Degradation model

Survival model
RUL Methods and when to use them

**Requirement:** Need to know what constitutes failure data

- **System Data**
  - Run-to-failure history
    - **Similarity Models**
      - Large data: Hash Similarity Model, Pairwise Similarity Model
      - Match signal shapes: Residual Similarity Model
      - Known degradation dynamics
  - Known failure threshold
    - **Degradation Models**
      - Log-scale signal or non-cumulative damage: Linear Degradation Model
      - Cumulative damage: Exponential Degradation Model
  - Life time data with or without covariates
    - **Survival Models**
      - Life time data only: Reliability Survival Model
      - Life time data and covariate (environment variable): Covariate Survival Model
Predictive Modeling and Deployment

```matlab
mdl = exponentialDegradationModel('LifeTimeUnit','hours','SlopeDetectionLevel',0.5);
tt = table([data_40_pca_tt;data_45_pca_tt;data_50_pca_tt;data_60_pca_tt]);
fit(mdl,tt,'Time','PCA1');
mdl

healthIndicator = data_55_pca_tt.PCA1;
threshold = -9;
timeUnit = 'hours';

Keep records at each iteration

totalDay = length(healthIndicator) - 1;
estRULFused = zeros(totalDay, 1);
trueRULFused = zeros(totalDay, 1);
CIRULFused = zeros(totalDay, 2);
poRULFused = cell(totalDay, 1);

% Create figures and axes for plot updating
figure('Visible','on')
ax1 = gca;```
Acquire Data → Preprocess Data → Identify Condition Indicators → Train Model → Deploy & Integrate

Edge device

Cloud
Deploy & Integrate analytics using MATLAB:
Feature Extraction Algorithm at the Edge

Pump flow sensor 1 sec ~ 1000 samples ~16kB

- 1 day ~ 1.3 GB
- 20 sensors/pump ~26 GB/day
- 3 pumps ~ 78 GB/day

Challenge:
Data transmission cost is pretty high

Solution:
Extract only relevant information and send it to predictive model
function [feature_list] = featureExtractionBuffer(data, timestamp)

persistent flow_array
persistent time_array
Np = 1000;

if isempty(flow_array)
    flow_array = nan(Np,1);
end

if isempty(time_array)
    time_array = nan(Np,1);
end

flow_array = [data; flow_array(1:Np-1)];
data = flow_array;

time_array = [timestamp; time_array(1:Np-1)];
timestamp = time_array;

if isempty(find(isnan(data),1))
    flow = date;
    % Ensure the flow is sampled at a uniform sample rate
    t_flow = timestamp;
function feature_list = featureExtractionBuffer(data, timestamp)

persistent flow_array
persistent time_array
Np = 1000;

if isempty(flow_array)
    flow_array = nan(Np,1);
end

if isempty(time_array)
    time_array = nan(Np,1);
end

flow_array = [data; flow_array(1:Np-1)];
data = flow_array;

time_array = [timestamp; time_array(1:Np-1)];
time = time_array;

if isempty(find(isnan(data), 1))
    flow = data;

    % Ensure the flow is sampled at a uniform sample rate
    t_flow = timestamp;
end
Code Deployment for Machine Learning

*Deploy trained models as standalone C/C++ code*

- SVM Classification
- Linear Classification

- Linear Regression
- Generalized Linear Regr.
- Decision trees
- Ensembles for Class.

- Ensembles for Regr.
- SVM Regression
- KNN Classification
- Gaussian Process Regr.
- Discriminant Analysis

- Non-tree Ensembles
- KNN with kd-tree
What do your end users want?

Flexible Deployment

▪ Maintenance needs simple, quick information
  – Hand held devices, Alarms

▪ Operations needs a birds-eye view
  – Integration with IT & OT systems

▪ Customers expect easy to digest information
  – Automated reports
Agenda:

1. What is Predictive Maintenance? Who is benefiting by doing it?

2. How can you develop a predictive maintenance algorithm using MATLAB?

3. How can you get started quickly?
MathWorks can help you get started TODAY

- Examples
- Documentation
- Tutorials & Workshops
- Consulting
- Tech Talk Series
Training Services  
Exploit the full potential of MathWorks products

Flexible delivery options:
- Public training available in several cities
- Onsite training with standard or customized courses
- Web-based training with live, interactive instructor-led courses

More than 48 course offerings:
- Introductory and intermediate training on MATLAB, Simulink, Stateflow, code generation, and Polyspace products
- Specialized courses in control design, signal processing, parallel computing, code generation, communications, financial analysis, and other areas

www.mathworks.in/training
Machine Learning with MATLAB

- This two-day course focuses on data analytics and machine learning techniques in MATLAB. The course demonstrates the use of unsupervised learning to discover features in large data sets and supervised learning to build predictive models. Topics include:
  - Organizing and preprocessing data
  - Clustering data
  - Creating classification and regression models
  - Interpreting and evaluating models
  - Simplifying data sets
  - Using ensembles to improve model performance
Statistical Methods in MATLAB

After this 2-day course you will be able to:

- Import, visualize, explore, and model data
- Fit probability distributions to data, and perform hypothesis tests
- Develop and fit regression models to data
- Generate random numbers and perform simulations
Summary: Why MATLAB for Predictive Maintenance?

- Dedicated toolbox for data preprocessing and feature extraction and developing predictive models
- Apps to make the task simple
- Support for taking these algorithm to edge and enterprise
- Get started quickly…examples, training and consulting
Please provide feedback for this block of sessions

- Scan this QR Code or log onto link below (link also sent to your phone and email)
- Enter the registration id number displayed on your badge
- Provide feedback for this session
MATLAB EXPO 2019

Speaker Details:
Email: adoshi@mathworks.com
LinkedIn: https://www.linkedin.com/in/amit-doshi/