

Predictive Maintenance with MATLAB and Simulink

Alec Stothert
Development Manager
Design Optimization and Identification

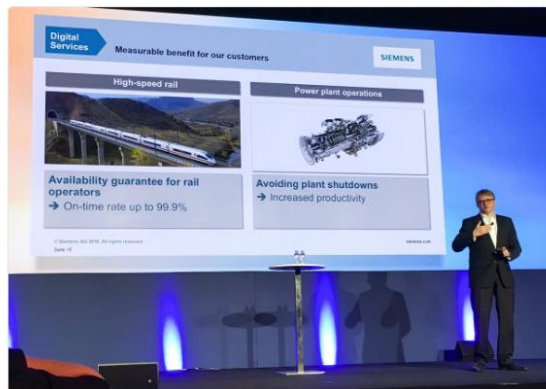


Why Predictive Maintenance?

- Improved operating efficiency
- New revenue streams
- Competitive differentiator



Siemens @Siemens
 Thanks to predictive maintenance the #Velaro E trains between Barcelona and Madrid run w/ 99.9% availability #GartnerSYM

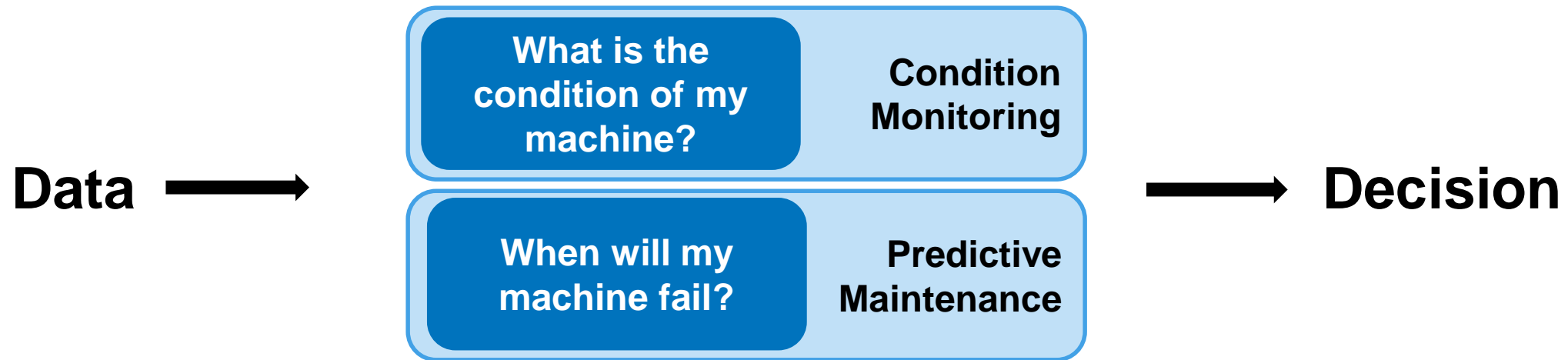


SAP IoT @SAP_IoT
 John Deere uses machine alerts using #telematics for predictive maintenance and to lower downtime of assets v3.co.uk/v3-uk/news/234 ... #IIoT



What does a Predictive Maintenance algorithm do?

Helps make maintenance decisions based on large volumes of complex data



Condition Monitoring

Process of monitoring sensor data from machines (vibration, temperature etc.) in order to identify significant changes which can indicate developing faults

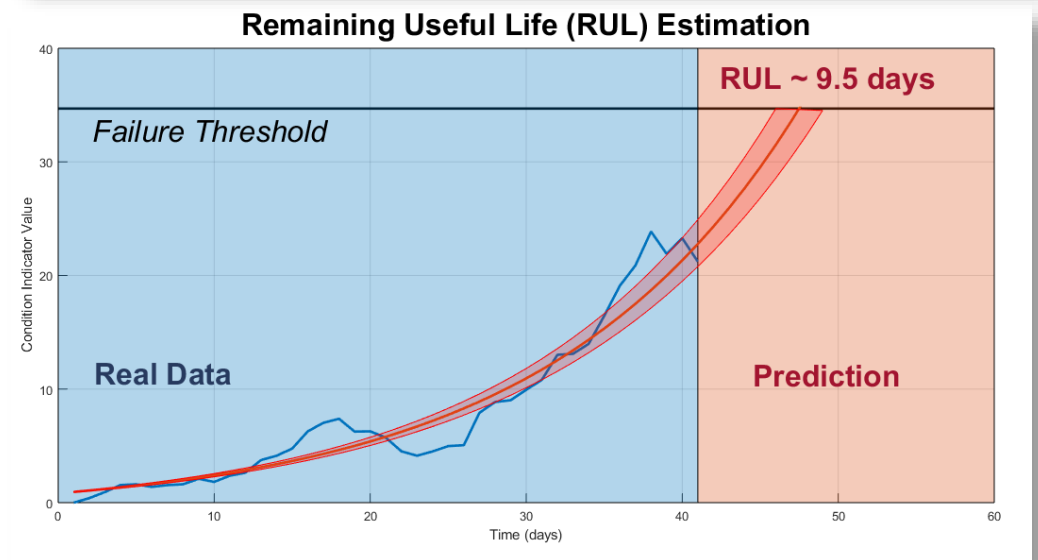
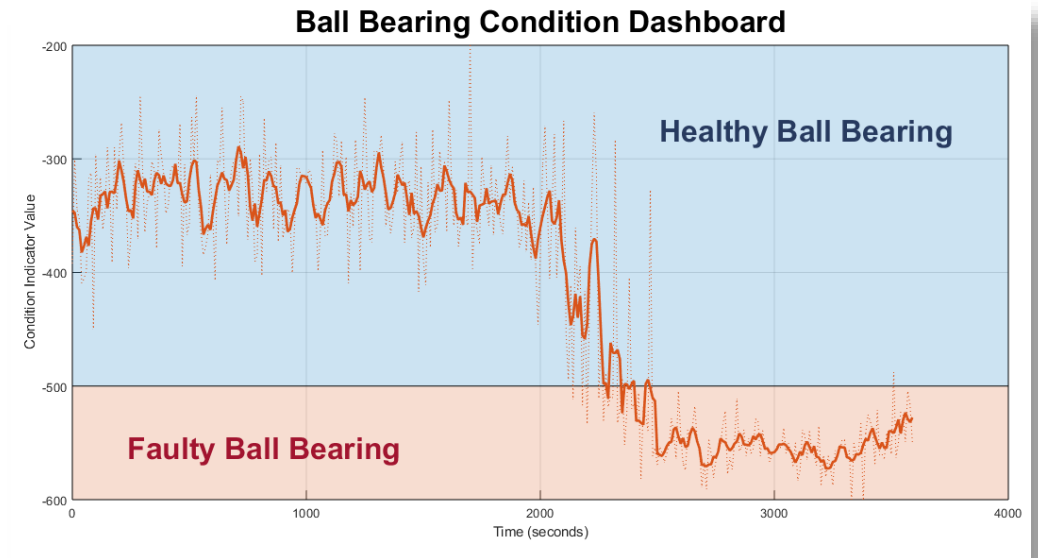
Predictive Maintenance

Technique that determines **time-to-failure/remaining useful life (RUL)** from sensor data & historical data in order to predict when maintenance should be performed

Predictive Maintenance Toolbox

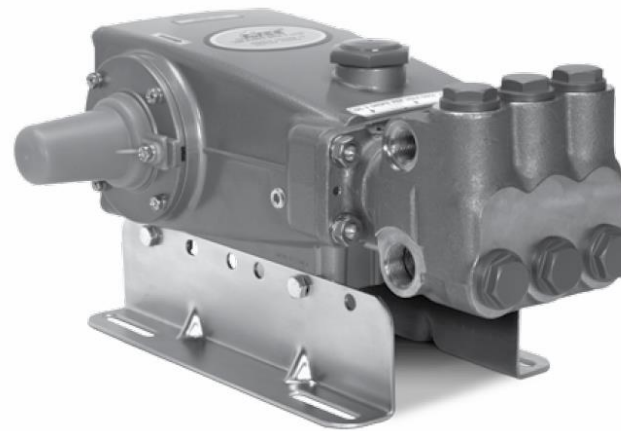
R2018a

- Develop and validate condition monitoring and predictive maintenance algorithms
- Apply signal processing and dynamic modeling techniques to extract features from your data to monitor machine health
- Train machine learning and time-series models to detect, classify, and predict machine failure

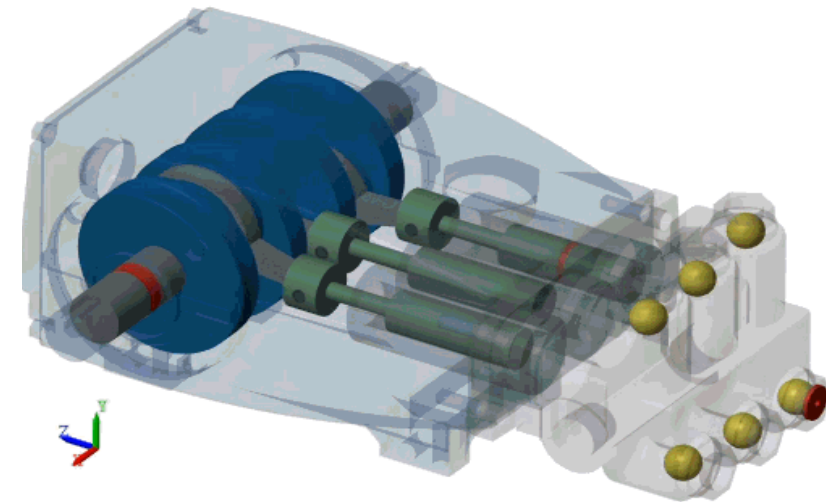


Reciprocating Pump Example

- Monitor pump condition and predict future condition
- Measure
 - Flow rate
 - Pressure
 - Engine current
- Faults
 - Leaks
 - Worn bearings
 - Blockages



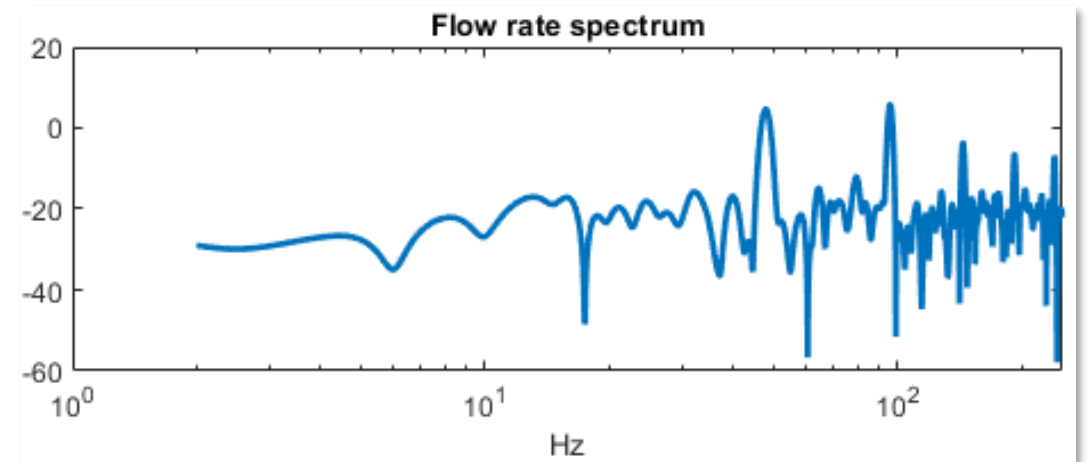
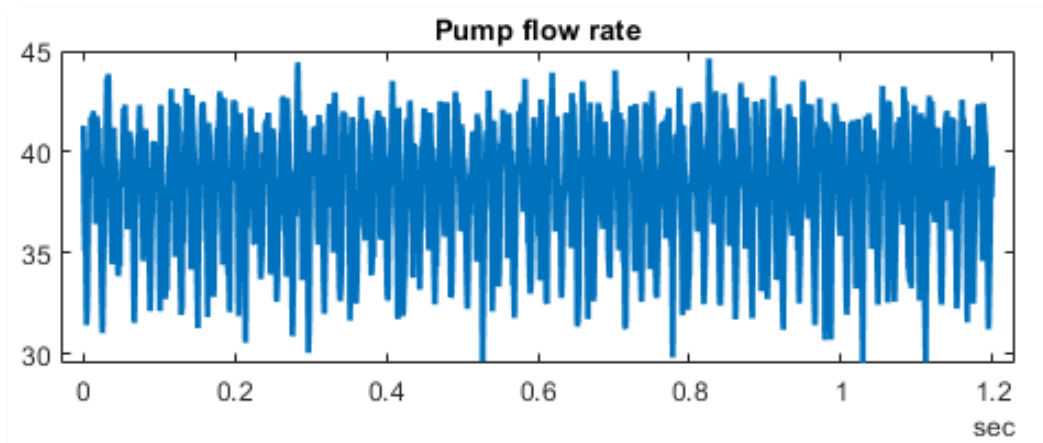
COMMON SPECIFICATIONS		
	U.S.	Metric
Bore	0.945"	24 mm
Stroke	1.18"	30 mm
Crankcase Capacity	42 oz.	1.26 l
Shaft Diameter	1.181"	30 mm



Frequency Domain Indicators

- Use spectral peaks and harmonics to understand condition of pump

```
% Remove the mean from the flow and compute the flow spectrum  
fA = flow;  
fA.Data = fA.Data - mean(fA.Data);  
[flowSpectrum,flowFrequencies] = pspectrum(fA,'FrequencyLimits',[2 250]);
```



Classify Faults Based on Condition Indicators

- Train and test a support vector machine

```
% Create and train the classifier
template = templateSVM(...
    'KernelFunction', 'polynomial', ...
    'PolynomialOrder', 2, ...
    'KernelScale', 'auto', ...
    'BoxConstraint', 1, ...
    'Standardize', true);
combinedClassifier = fitcecoc(...
    predictors(cvp.training(1,:), ...
    response(cvp.training(1,:), ...
    'Learners', template, ...
    'Coding', 'onevsone', ...
    'ClassNames', [0; 1; 2; 3; 4; 5; 6; 7]));
```

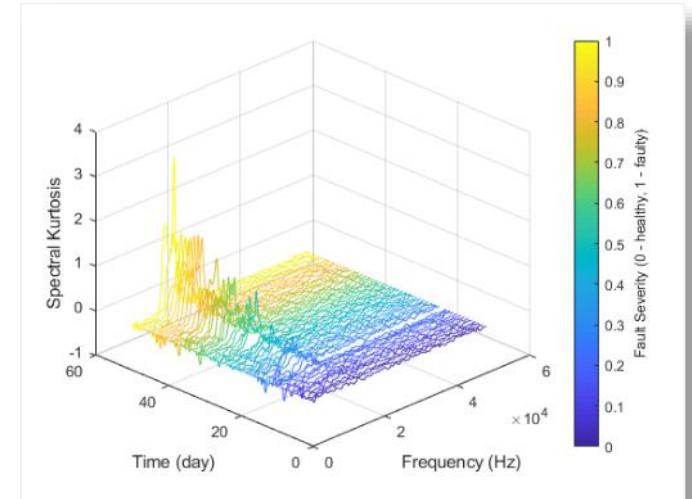
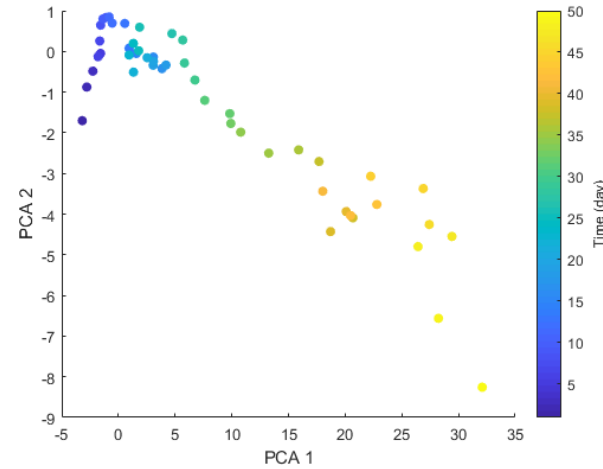
Actual leak fault	None	35	0	0	0	1	0	0	0
	Leak	0	24	0	5	0	1	0	0
	Blocking	2	0	32	0	0	0	1	0
	Leak & Blocking	0	1	0	14	0	0	0	0
	Bearing	2	0	0	0	26	1	2	0
	Bearing & Leak	0	1	0	2	0	10	0	1
	Bearing & Blocking	0	0	2	1	3	0	15	0
	All	0	0	0	0	0	4	1	13
		None	Leak	Blocking	Leak & Blocking	Bearing	Bearing & Leak	Bearing & Blocking	All

Other Condition Indicators

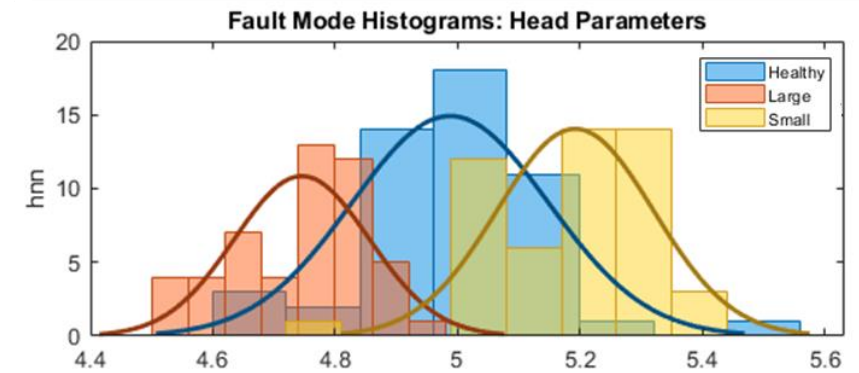
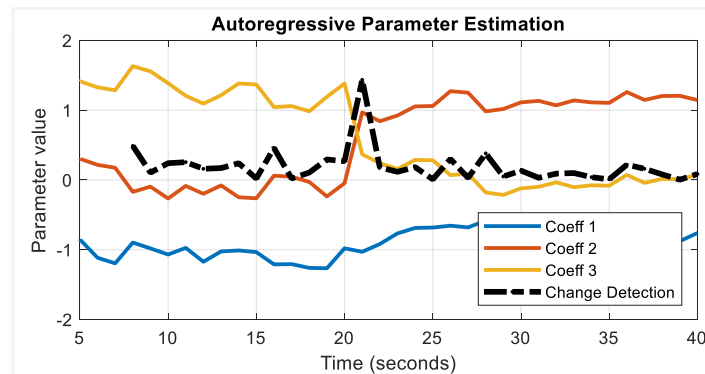
Extract features using signal-based and model-based methods to determine machine health

Condition Indicator

Signal-based Methods

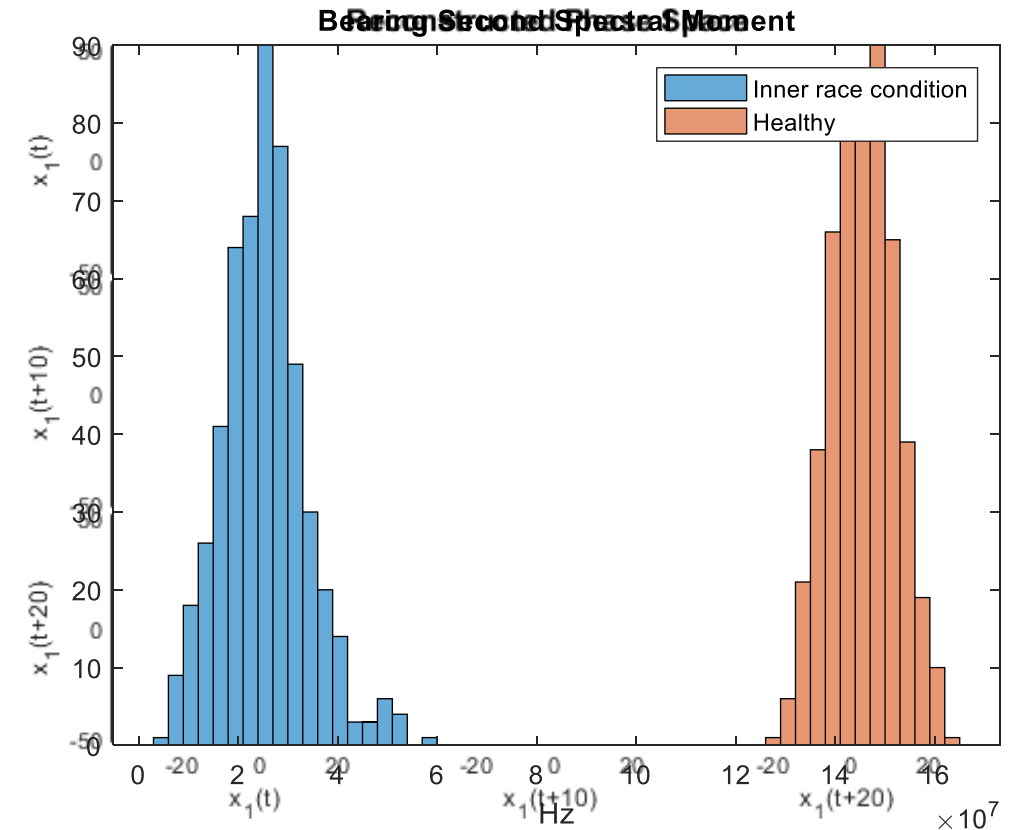


Model-based Methods



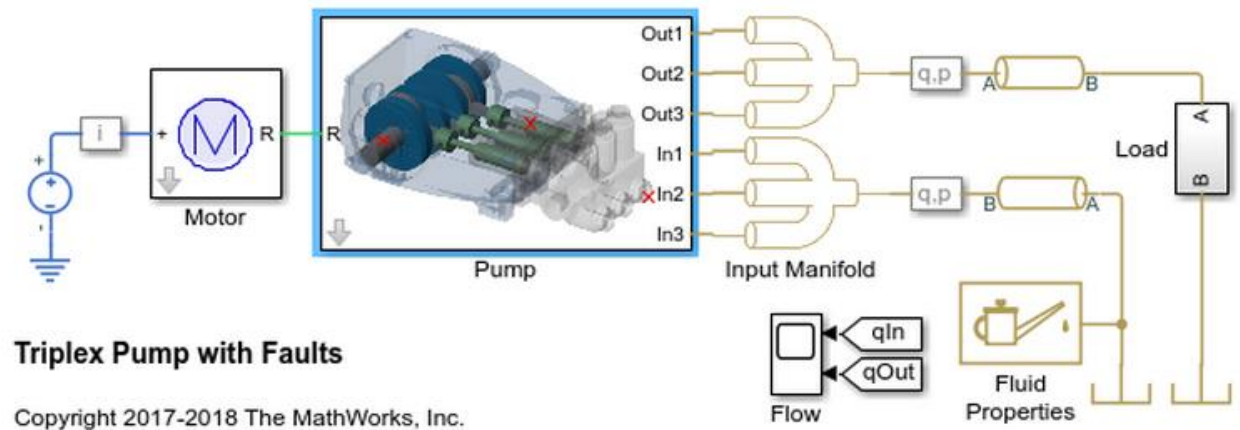
Advanced Condition Indicators

- Capture time-varying dynamics (e.g. vibration data) by computing time-frequency moments
- Detect sudden changes in nonlinear systems using phase-space reconstruction methods (correlation dimension, approximate entropy, Lyapunov exponent)



Tools for Managing and Analyzing Data

- Generate data using Simulink
- Analyze data using datastores

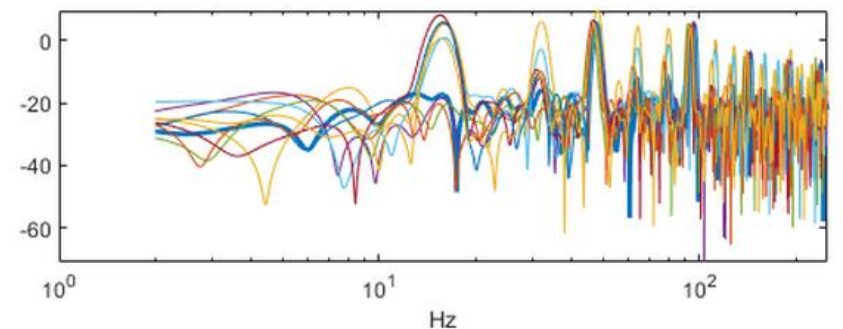
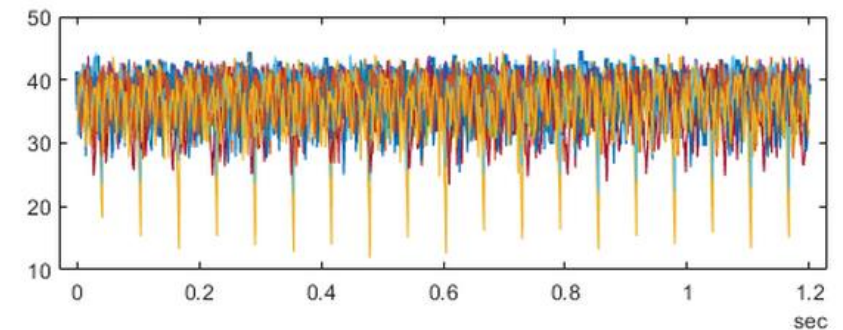


```
ens = simulationEnsembleDatastore('.\Data');
ens.SelectedVariables = ["qOut_meas", ...
    "pMid", "qVar", "qSkewness", "qKurtosis",...
    "LeakFault", "BlockingFault", "BearingFault"];
dataTable = tall(ens)
```

dataTable =

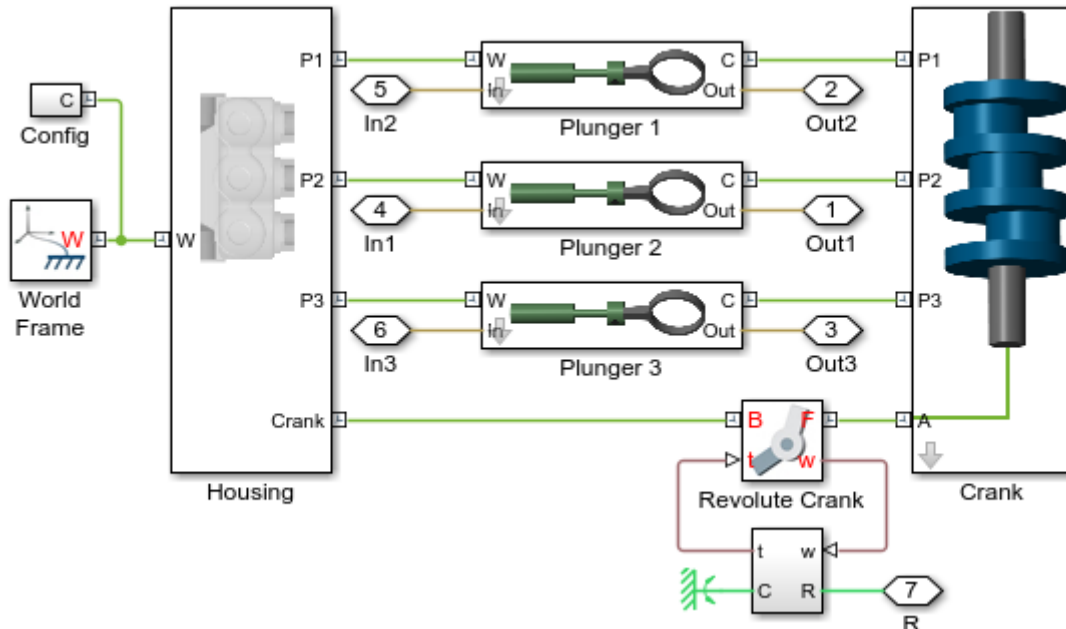
M×8 tall table

qOut_meas	pMid	qVar	qSkewness	qKurtosis	LeakFault	BlockingFault	BearingFault
[2001×1 timetable]	103.56	9.5309	-0.55515	2.4113	1e-09	0.8	0
[2001×1 timetable]	87.039	9.2684	-0.57675	2.4018	1e-09	0.8	0
[2001×1 timetable]	117.22	9.3804	-0.54379	2.3226	1e-09	0.8	0
[2001×1 timetable]	97.658	9.5407	-0.5538	2.4124	4e-07	0.74	0.0002
[2001×1 timetable]	106.05	9.232	-0.56334	2.4103	1e-09	0.8	0
[2001×1 timetable]	109.96	9.732	-0.53987	2.3798	1e-09	0.8	0
[2001×1 timetable]	105.06	9.4902	-0.56641	2.3461	1e-09	0.8	0
[2001×1 timetable]	105.1	9.2956	-0.56135	2.3623	1e-09	0.8	0
:	:	:	:	:	:	:	:
:	:	:	:	:	:	:	:

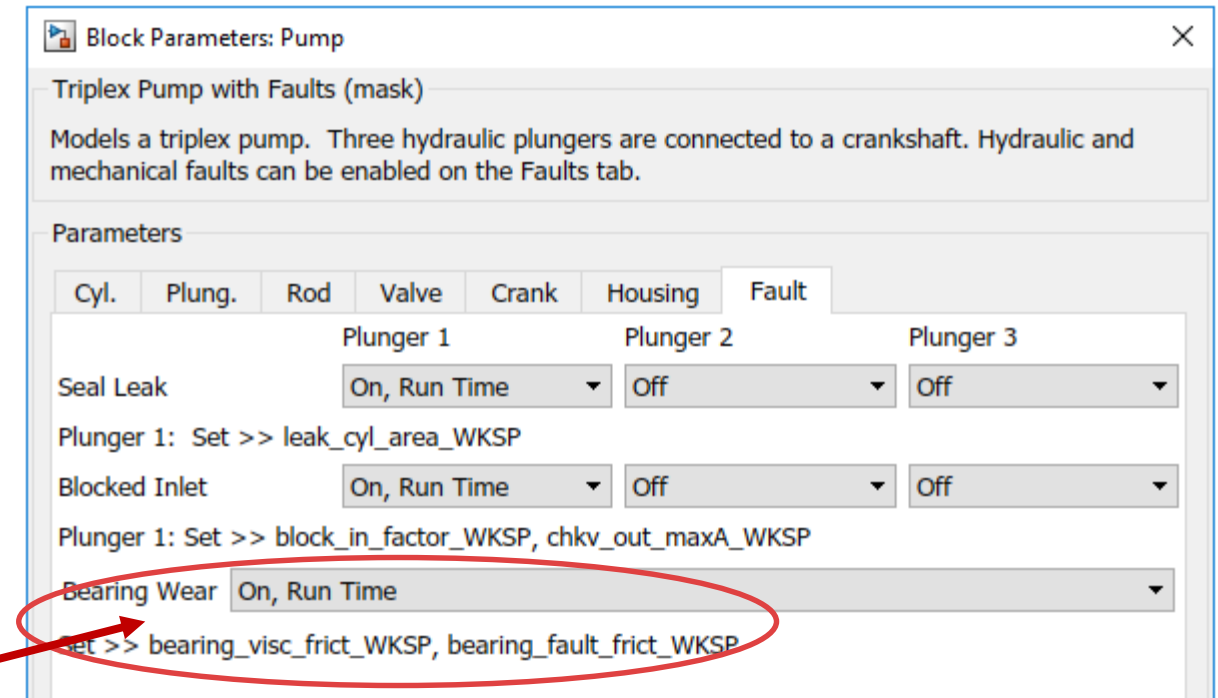
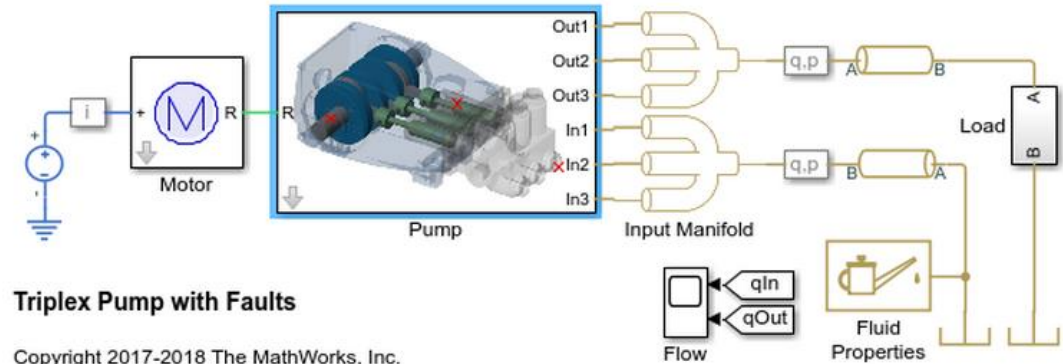


Modeling Faults In Simulink

- Parameterize Blocks



FEMA analysis to choose
parameters and failure modes



Datastore to Manage Data

M×9 [tall table](#)

Date	qOut_meas	pMid	qVar	qSkewness	qKurtosis	LeakFault	BlockingFault	BearingFault
16-Jan-2015	[2001×1 timetable]	103.56	9.5309	-0.55515	2.4113	1e-09	0.8	0
17-Jan-2015	[2001×1 timetable]	87.039	9.2684	-0.57675	2.4018	1e-09	0.8	0
18-Jan-2015	[2001×1 timetable]	117.22	9.3804	-0.54379	2.3226	1e-09	0.8	0
19-Jan-2015	[2001×1 timetable]	97.658	9.5407	-0.5538	2.4124	4e-07	0.74	0.0002
20-Jan-2015	[2001×1 timetable]	106.05	9.232	-0.56334	2.4103	1e-09	0.8	0
21-Jan-2015	[2001×1 timetable]	109.96	9.732	-0.53987	2.3798	1e-09	0.8	0
22-Jan-2015	[2001×1 timetable]	105.06	9.4902	-0.56641	2.3461	1e-09	0.8	0
23-Jan-2015	[2001×1 timetable]	105.1	9.2956	-0.56135	2.3623	1e-09	0.8	0
:	:	:	:	:	:	:	:	:
:	:	:	:	:	:	:	:	:

- Ensemble (whole table)
- Member (one row)
- Independent variables
- Data variables
 - Source variables
 - Derived variables
- Condition variables

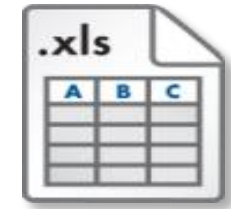
```
ens = simulationEnsembleDatastore('.\Data');
ens.SelectedVariables = [...
    "qOut_meas", "qVar", "qSkewness", "qKurtosis", ...
    "LeakFault",];
data = read(ens)
```

data = 1×5 table

	qOut_meas	qVar	qSkewness	qKurtosis	LeakFault
1	2001×1 tim...	9.5309	-0.5551	2.4113	1.0000e-09

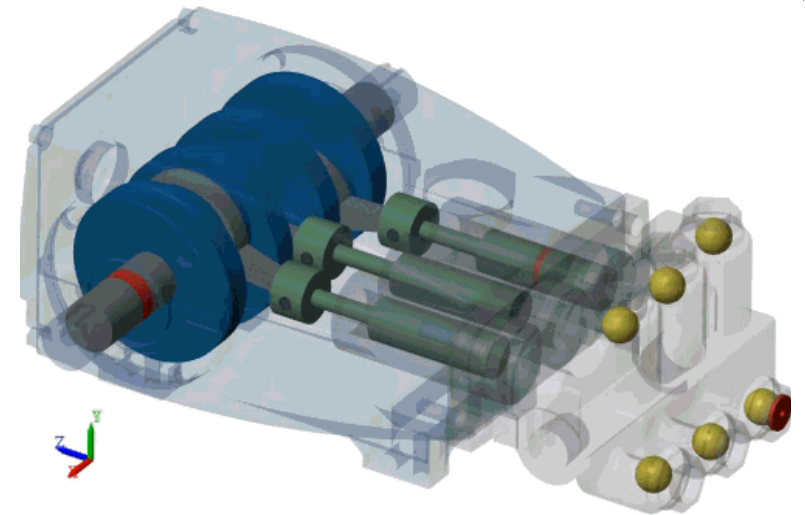
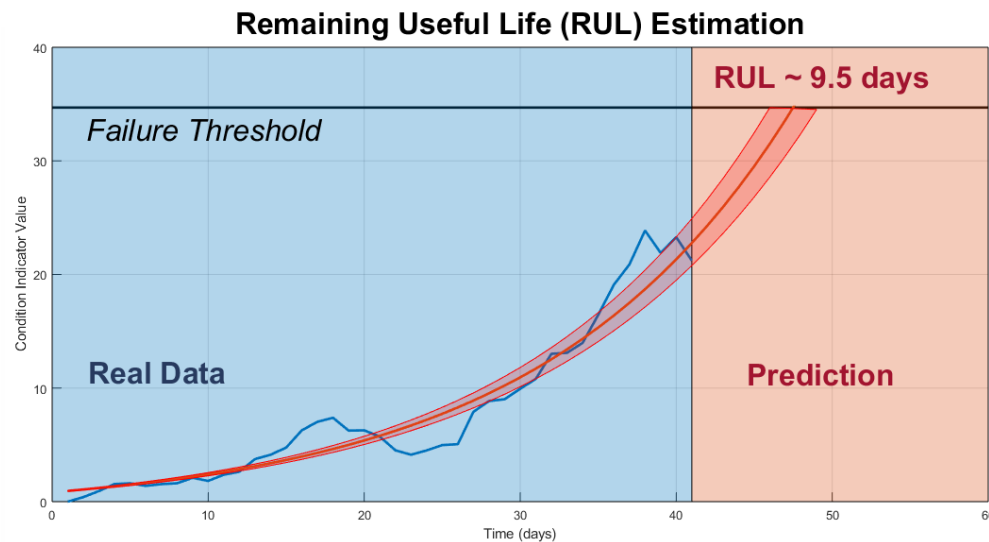
Sensor And Data Access in MATLAB

- Data sources accessible through MATLAB
 - Files (.xls, .csv, .txt, .mat, etc,)
 - Distributed file systems Azure Blob Storage
 - Amazon S3
 - Industrial Internet of Things



Predict Pump Failures

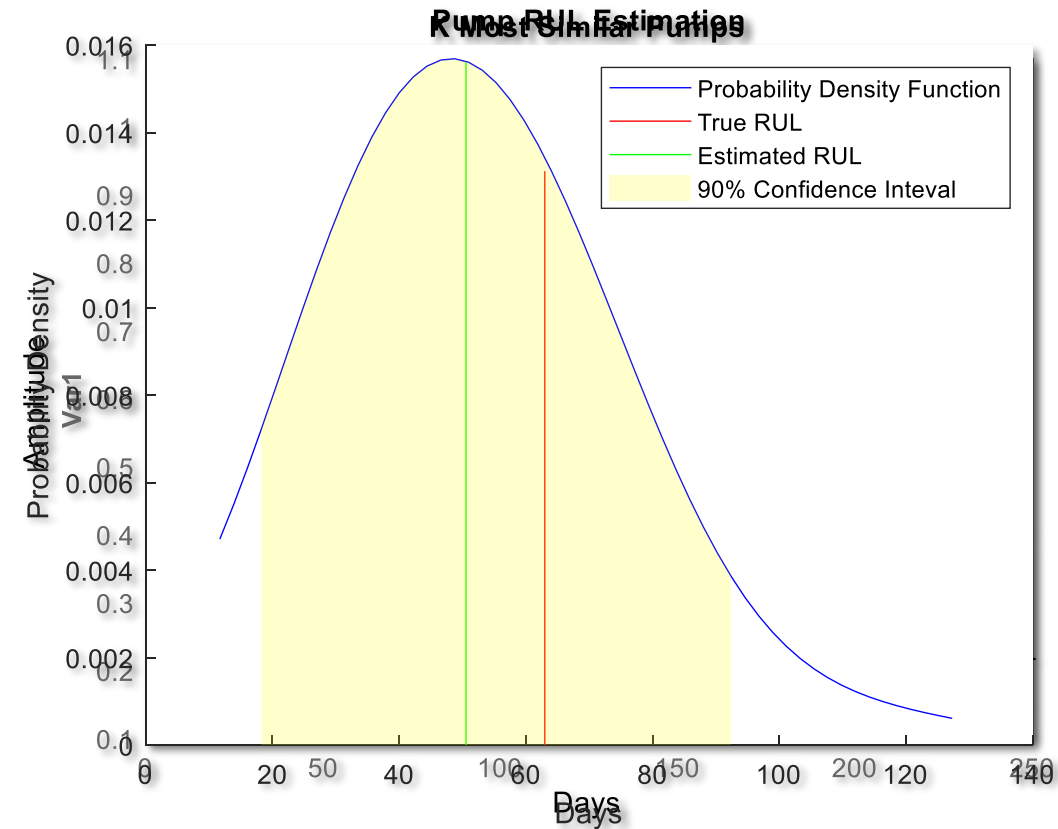
- Use condition indicators to predict future behavior – remaining useful life (RUL)



Using Fleet Data to Predict Remaining Useful Life

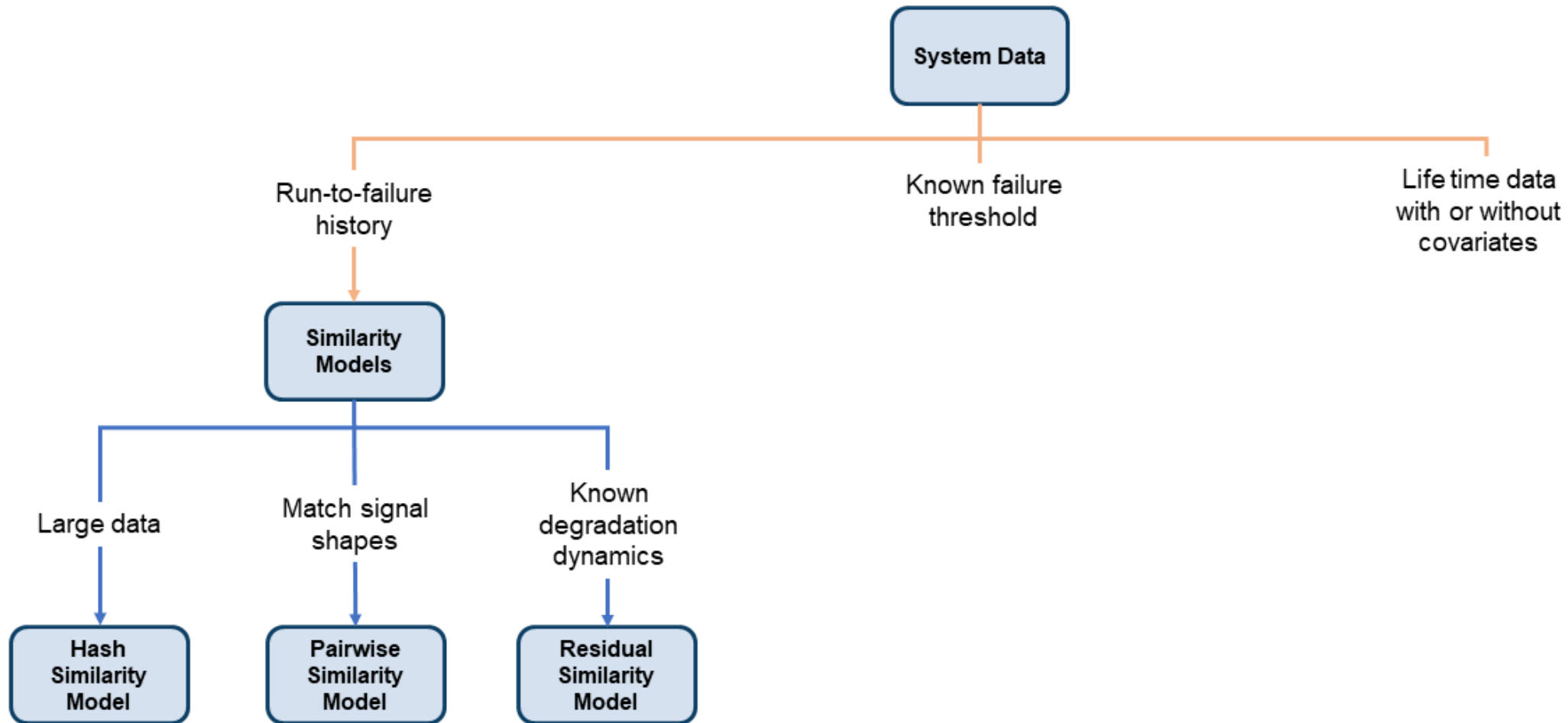
```
% Build and train the model
mdl = residualSimilarityModel(...
    'Method', 'poly2',...
    'Distance', 'absolute',...
    'NumNearestNeighbors', 50,...
    'Standardize', 1);
fit(mdl, trainData);

% Use the model to predict RUL
[estRUL,ciRUL,pdfRUL] = predictRUL(mdl, newData);
```



RUL Methods and when to use them

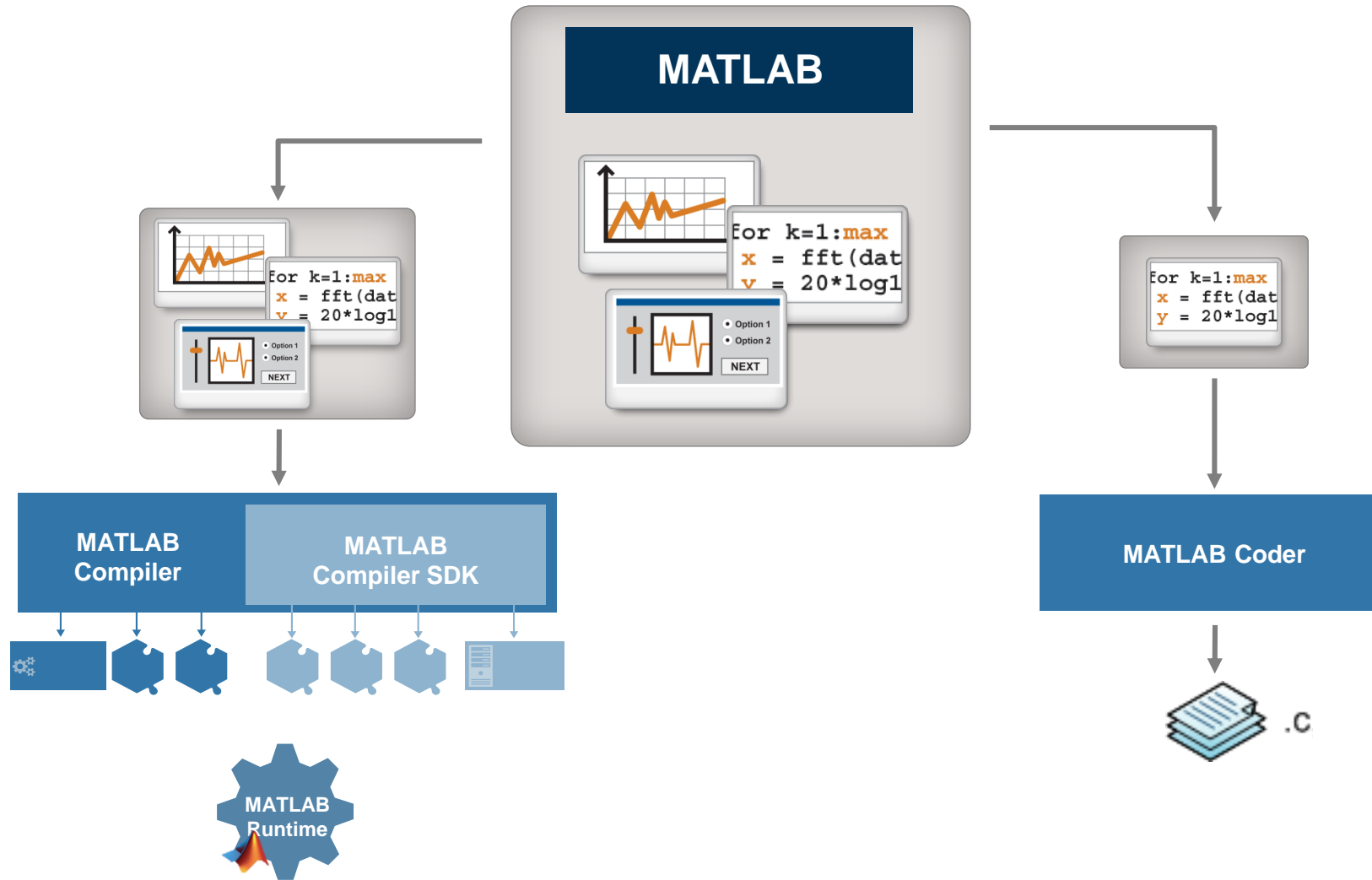
Requirement: Need to know what constitutes failure data



**Weibull
Distribution**

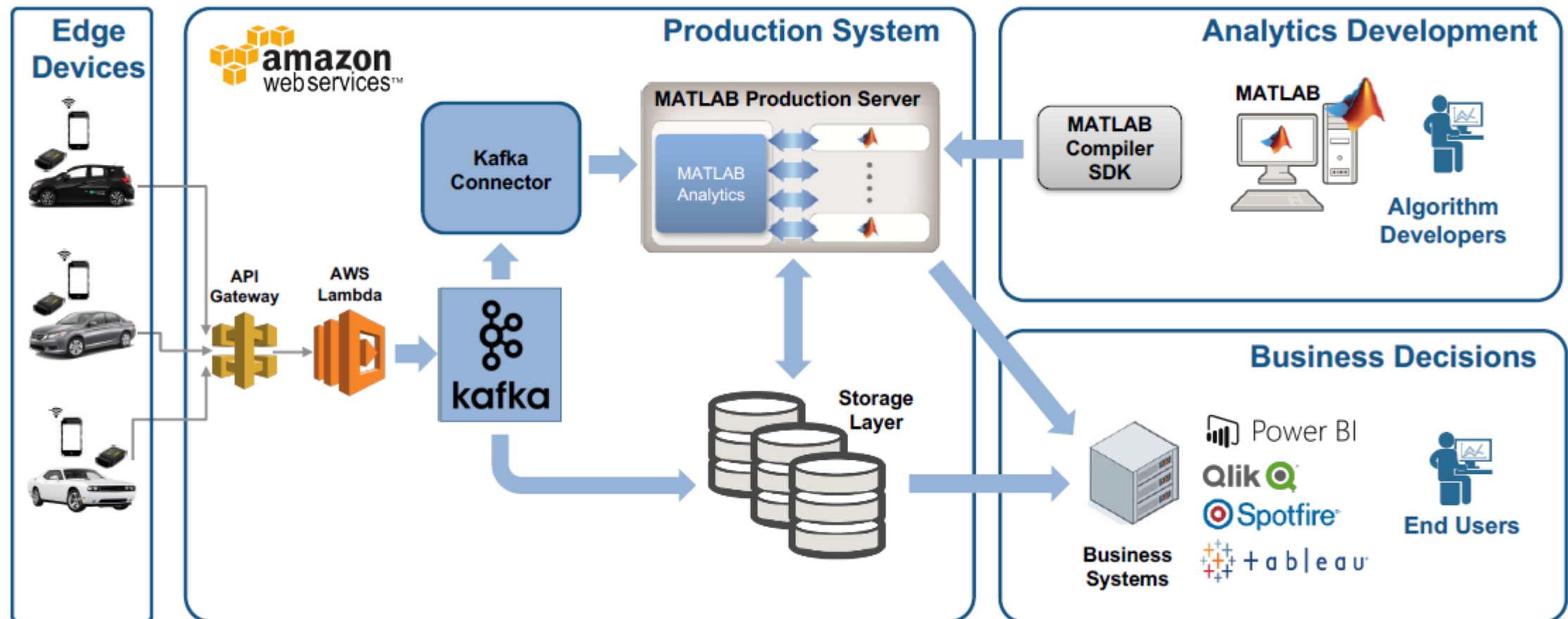
MATLAB Coder and Compiler

- Run predictive model on embedded devices



MATLAB Production Server and Enterprise Integration

- Integrate predictive model with your enterprise system and cloud platform



Predictive Maintenance Development - Common pains

- How do I get started with developing algorithms?
 - Reference examples
 - Documentation based on the workflow
- How do I manage data and what if I don't have any data?
 - Command line functions to manage and label data
 - Examples showing Simulink models generating failure data
- How do I chose condition indicators and estimate the RUL?
 - Functions provided for estimating RUL
 - Functions for computing condition indicators

The screenshot shows the MathWorks documentation page for the example 'Nonlinear State Estimation of a Degrading Battery System' (R2017b). The page includes a 'CONTENTS' section and a code snippet for running simulations. Below the code, there is a log of simulation progress showing 10 completed runs out of 208. At the bottom, there are sections for 'Deploy Predictive Maintenance Algorithms' and 'Applications'.

Documentation

CONTENTS

Nonlinear State Estimation of a Degrading Battery System R2017b

This example shows how to estimate the states of a nonlinear system using the Extended Kalman Filter (EKF). The example also shows how to use the EKF to estimate the Remaining Useful Life (RUL) of the system.

```
% Run the simulations and create an ensemble to manage the simulation results
mkdir('.\Data') % Create directory to store results
runAll = true;
if runAll
    ens = createSimulationEnsemble([gridSimulationInput, randomSimulationInput], ...
    [pwd '\Data'], 'UseParallel', true);
else
    ens = createSimulationEnsemble(gridSimulationInput(1:10), [pwd '\Data']);
end
```

[21-Nov-2017 09:06:31] Checking for availability of parallel pool...
 Starting parallel pool (parpool) using the 'local' profile ...
 connected to 6 workers.
 [21-Nov-2017 09:06:56] Loading Simulink on parallel workers...
 [21-Nov-2017 09:07:12] Configuring simulation cache folder on parallel workers...
 [21-Nov-2017 09:07:12] Loading model on parallel workers...
 [21-Nov-2017 09:07:18] Running simulations...
 Analyzing and transferring files to the workers ...done.
 [21-Nov-2017 09:07:37] Completed 1 of 208 simulation runs
 [21-Nov-2017 09:07:38] Completed 2 of 208 simulation runs
 [21-Nov-2017 09:07:38] Completed 3 of 208 simulation runs
 [21-Nov-2017 09:07:39] Completed 4 of 208 simulation runs
 [21-Nov-2017 09:07:39] Completed 5 of 208 simulation runs
 [21-Nov-2017 09:07:39] Completed 6 of 208 simulation runs
 [21-Nov-2017 09:07:46] Completed 7 of 208 simulation runs
 [21-Nov-2017 09:07:46] Completed 8 of 208 simulation runs
 [21-Nov-2017 09:07:47] Completed 9 of 208 simulation runs
 [21-Nov-2017 09:07:47] Completed 10 of 208 simulation runs

0 20 40 60 80 100 120 140
Cycle

From decision models for condition monitoring and fault detection, predict remaining useful life (RUL)

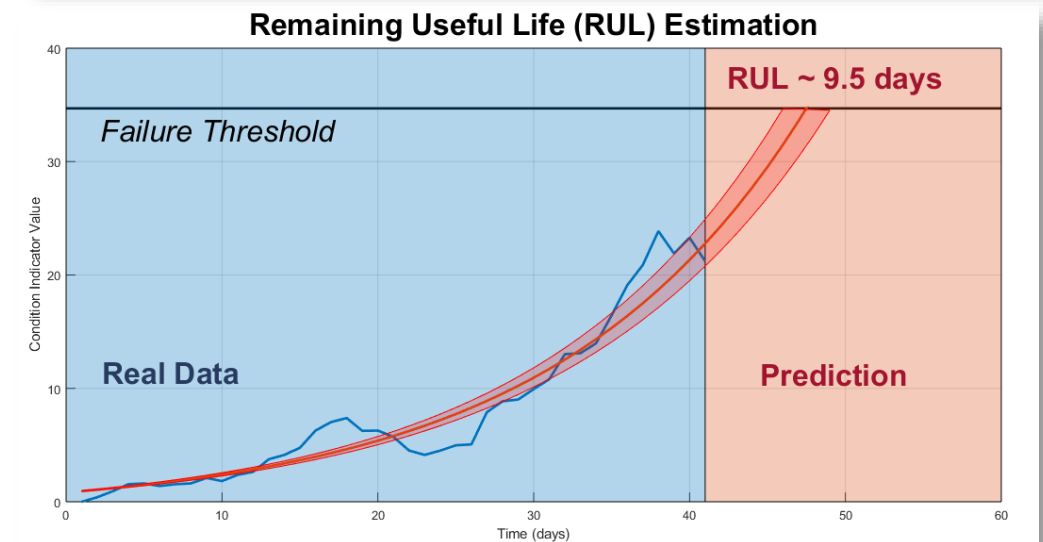
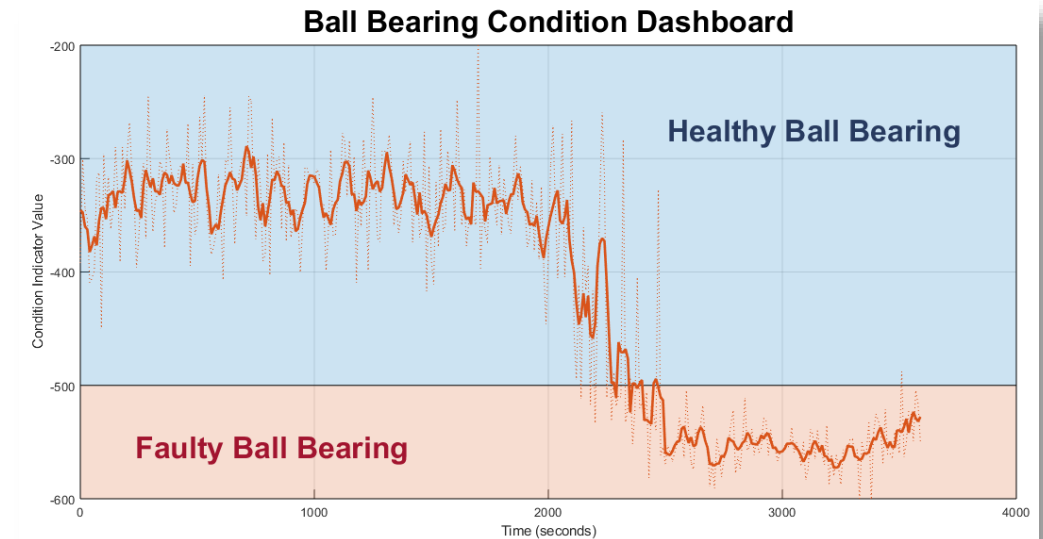
Deploy Predictive Maintenance Algorithms
 Implement and deploy condition-monitoring and predictive-maintenance algorithms

Applications
 Examples of predictive-maintenance algorithm development

Predictive Maintenance Toolbox

R2018a

- Develop and validate condition monitoring and predictive maintenance algorithms
- Apply signal processing and dynamic modeling techniques to extract features from your data to monitor machine health
- Train machine learning and time-series models to detect, classify, and predict machine failure
- Use functionality from Signal Processing, Statistics and Machine Learning and System identification



Thank you!

Predictive Maintenance Toolbox

R2018a

- Develop and validate condition monitoring and predictive maintenance algorithms
- Apply signal processing and dynamic modeling techniques to extract features from your data to monitor machine health
- Train machine learning and time-series models to detect, classify, and predict machine failure
- Use functionality from Signal Processing, Statistics and Machine Learning and System identification

