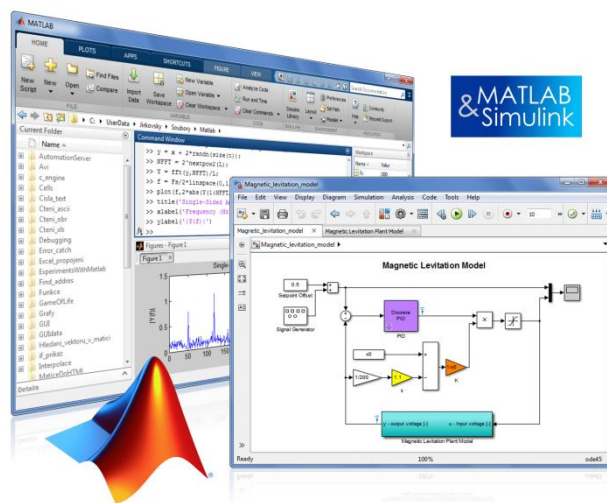


5.9.2019 Brno

## TCC 2019

# Prediktivní údržba v prostředí MATLAB



Jaroslav Jirkovský  
jirkovsky@humusoft.cz

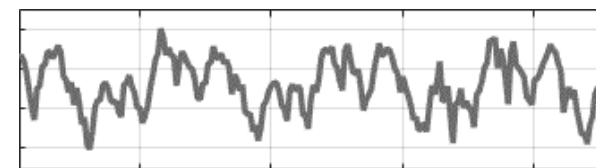
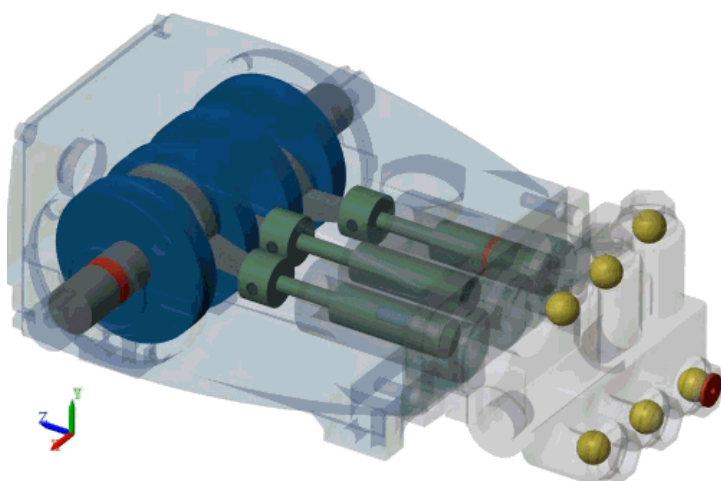
[www.humusoft.cz](http://www.humusoft.cz)  
[info@humusoft.cz](mailto:info@humusoft.cz)

[www.mathworks.com](http://www.mathworks.com)

# Types of Maintenance

- **Reactive – Perform maintenance once there's a problem**
  - Example: replace car battery when it has a problem
  - Problem: unexpected failures can be expensive and potentially dangerous
- **Scheduled – Perform maintenance at a regular rate**
  - Example: change car's oil every 5,000 miles
  - Problem: unnecessary maintenance can be wasteful; may not eliminate all failures
- **Predictive – Forecast when problems will arise**
  - Example: certain GM car models forecast problems with the battery, fuel pump, and starter motor
  - Problem: difficult to make accurate forecasts for complex equipment

# What is Predictive Maintenance?



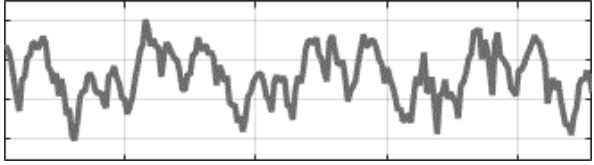


English Spanish French Pump - detected



English Russian Greek

Translate



1/5000

I need help.



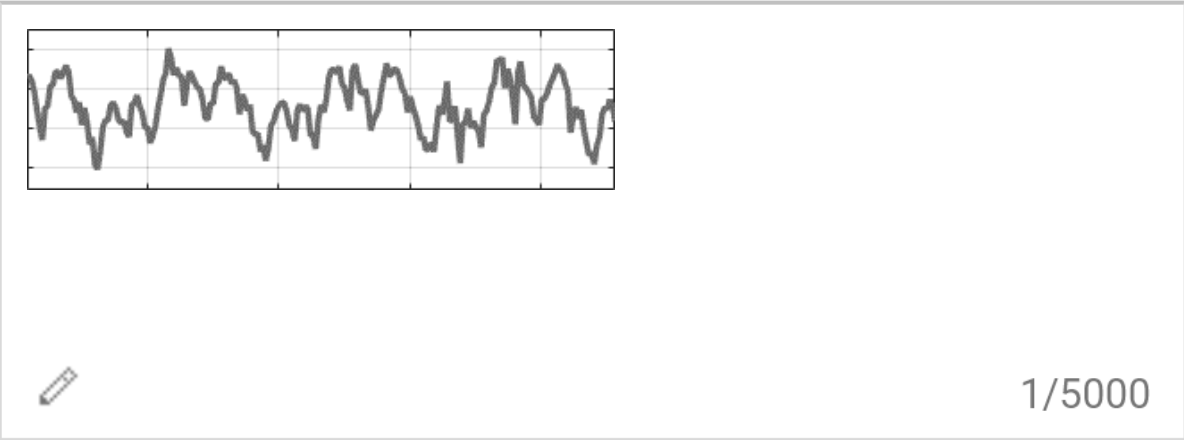


English Spanish French Pump - detected ▾



English Russian Greek ▾

Translate



**I need help. One of my cylinders is blocked. I will shut down your production line in 15 hours.**

# Benefits of Predictive Maintenance

**Increase “up time” and safety**

**Minimize maintenance costs**

**Optimize supply chain**

# Predictive Maintenance Algorithm

## ⇒ Answers These Questions

Is my machine  
operating  
normally?

Anomaly  
Detection

I need help.

Why is my  
machine behaving  
abnormally?

Condition  
Monitoring

One of my cylinders is blocked.

How much longer  
can I operate my  
machine ?

Remaining  
Useful Life  
Estimation

I will shut down your line in 15 hours.

# Predictive Maintenance Toolbox for Developing Algorithms

Is my machine operating normally?

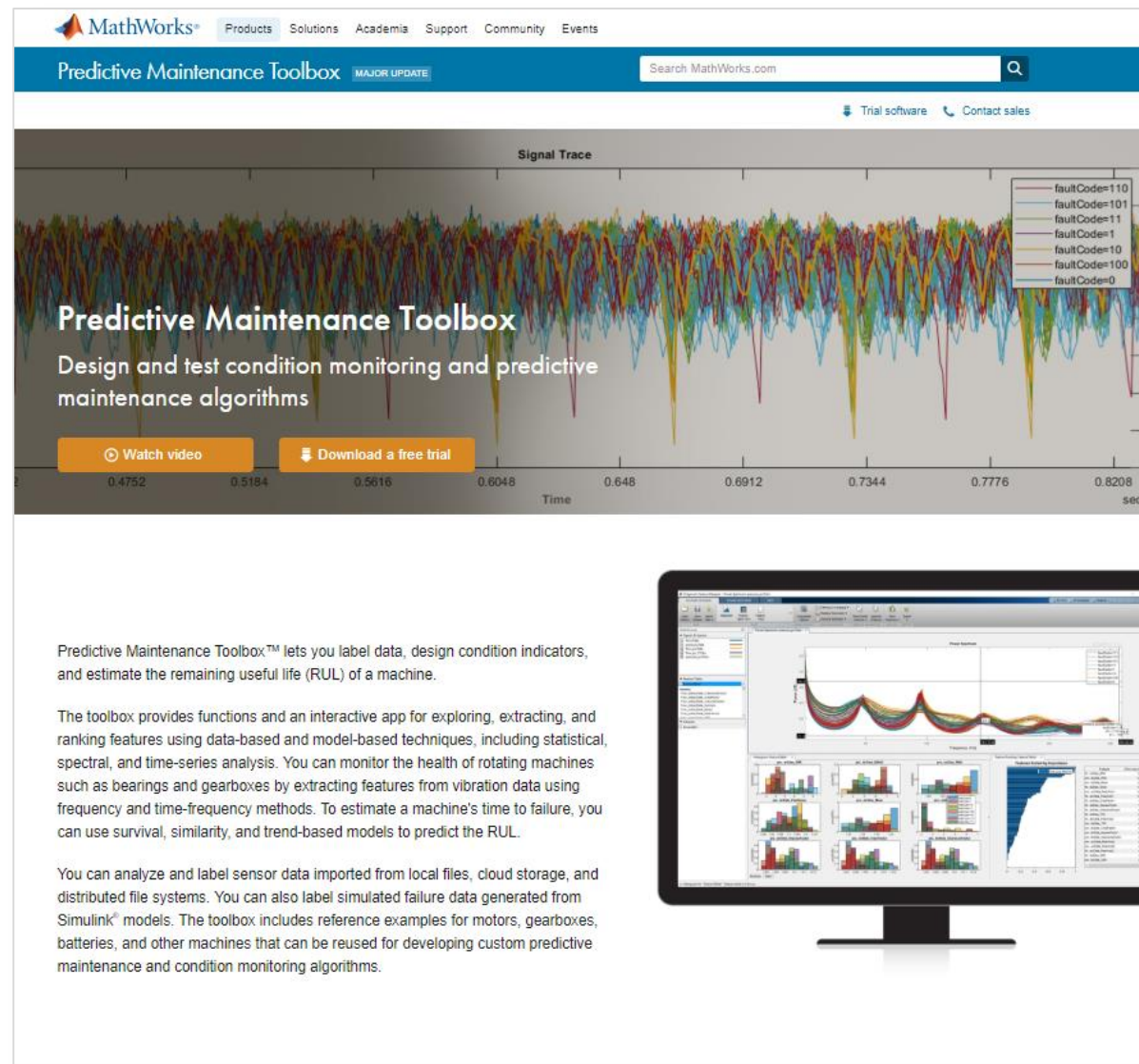
**Anomaly Detection**

Why is my machine behaving abnormally?

**Condition Monitoring**

How much longer can I operate my machine ?

**Remaining Useful Life Estimation**




The screenshot shows the MathWorks Predictive Maintenance Toolbox website. At the top, there is a navigation bar with links for Products, Solutions, Academia, Support, Community, and Events. Below this is a search bar and a 'MAJOR UPDATE' badge. The main content area features a 'Signal Trace' plot with multiple colored lines representing different fault codes. A legend on the right lists fault codes: faultCoder=110, faultCoder=101, faultCoder=11, faultCoder=1, faultCoder=10, faultCoder=100, and faultCoder=0. Below the plot, there are buttons for 'Watch video' and 'Download a free trial'. The x-axis is labeled 'Time' and has values from 0.4752 to 0.8208 seconds.

Predictive Maintenance Toolbox™ lets you label data, design condition indicators, and estimate the remaining useful life (RUL) of a machine.

The toolbox provides functions and an interactive app for exploring, extracting, and ranking features using data-based and model-based techniques, including statistical, spectral, and time-series analysis. You can monitor the health of rotating machines such as bearings and gearboxes by extracting features from vibration data using frequency and time-frequency methods. To estimate a machine's time to failure, you can use survival, similarity, and trend-based models to predict the RUL.

You can analyze and label sensor data imported from local files, cloud storage, and distributed file systems. You can also label simulated failure data generated from Simulink® models. The toolbox includes reference examples for motors, gearboxes, batteries, and other machines that can be reused for developing custom predictive maintenance and condition monitoring algorithms.



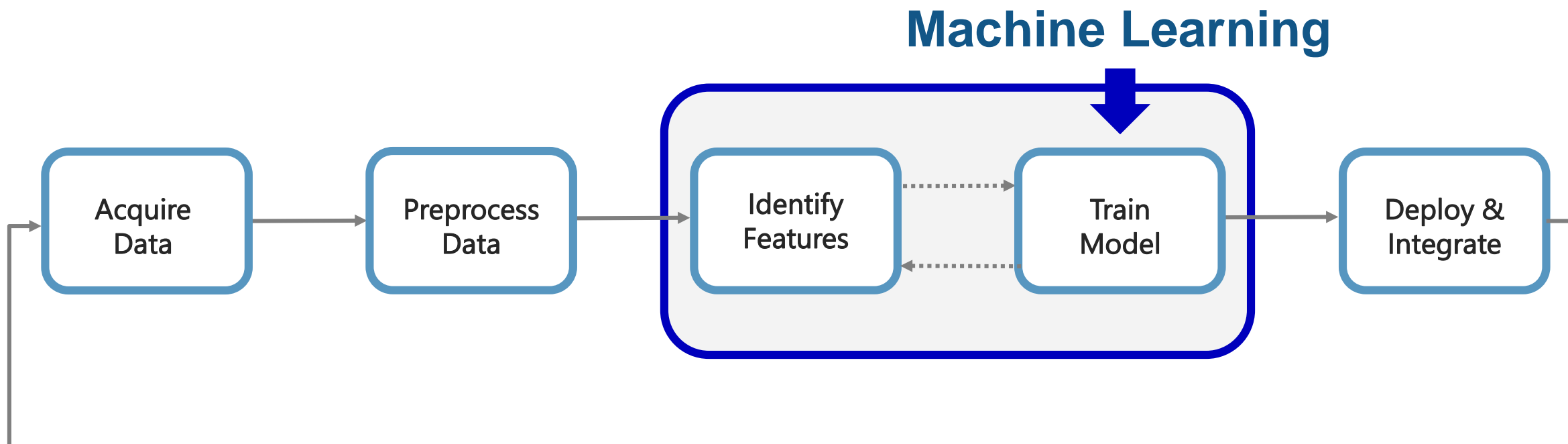
The software interface screenshot shows a dashboard with several charts and plots. There is a large plot at the top showing a signal trace, and several smaller bar charts and line graphs below it. The interface is clean and professional, with a dark theme.



# Approaches to Condition Monitoring and Predictive Maintenance

- **”Generic” Machine Learning (including Deep Learning)**
  - E.g. Classification of failure types (bearing fault, burnt diode, seal leak)
  - E.g. Binary classification of machine state (normal/abnormal)
  - E.g. Classification of service urgency (high, medium, low)
  - E.g. Regression for remaining life estimation (15, 16, 17, 18, ... days)
  - Tools for this: Statistics and Machine Learning toolbox, Deep Learning toolbox
- **Specialized models (in Predictive Maintenance toolbox)**
  - Degradation models
  - Similarity models
  - Survival models
  - All of the above answer to ”how much longer will the machine work?”

# Workflow for Developing a Predictive Maintenance Algorithm

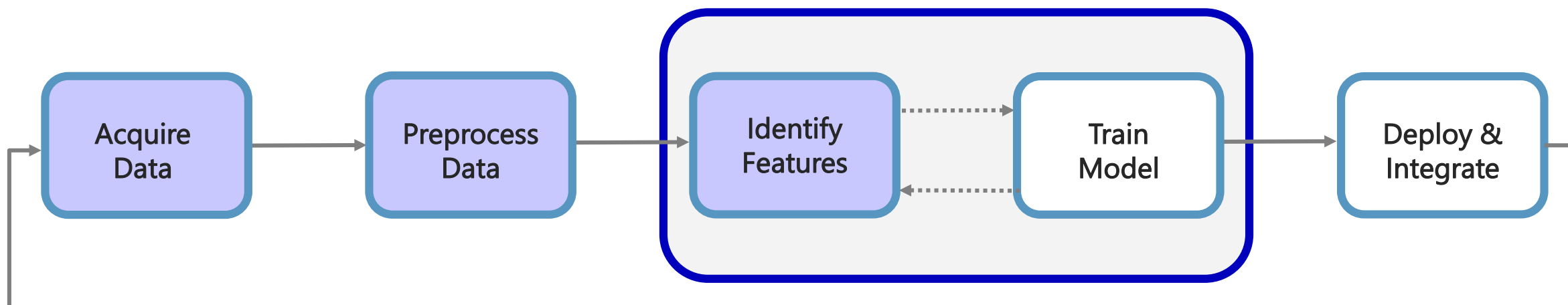


# Predictive Maintenance Challenges

- **Reduce the amount of data you need to store and transmit**
- **Explore approaches to feature extraction and predictive modeling**
- **Deliver the results of your analytics based on your audience**
- **Get started quickly...especially if you are an engineer**

# Predictive Maintenance Challenges

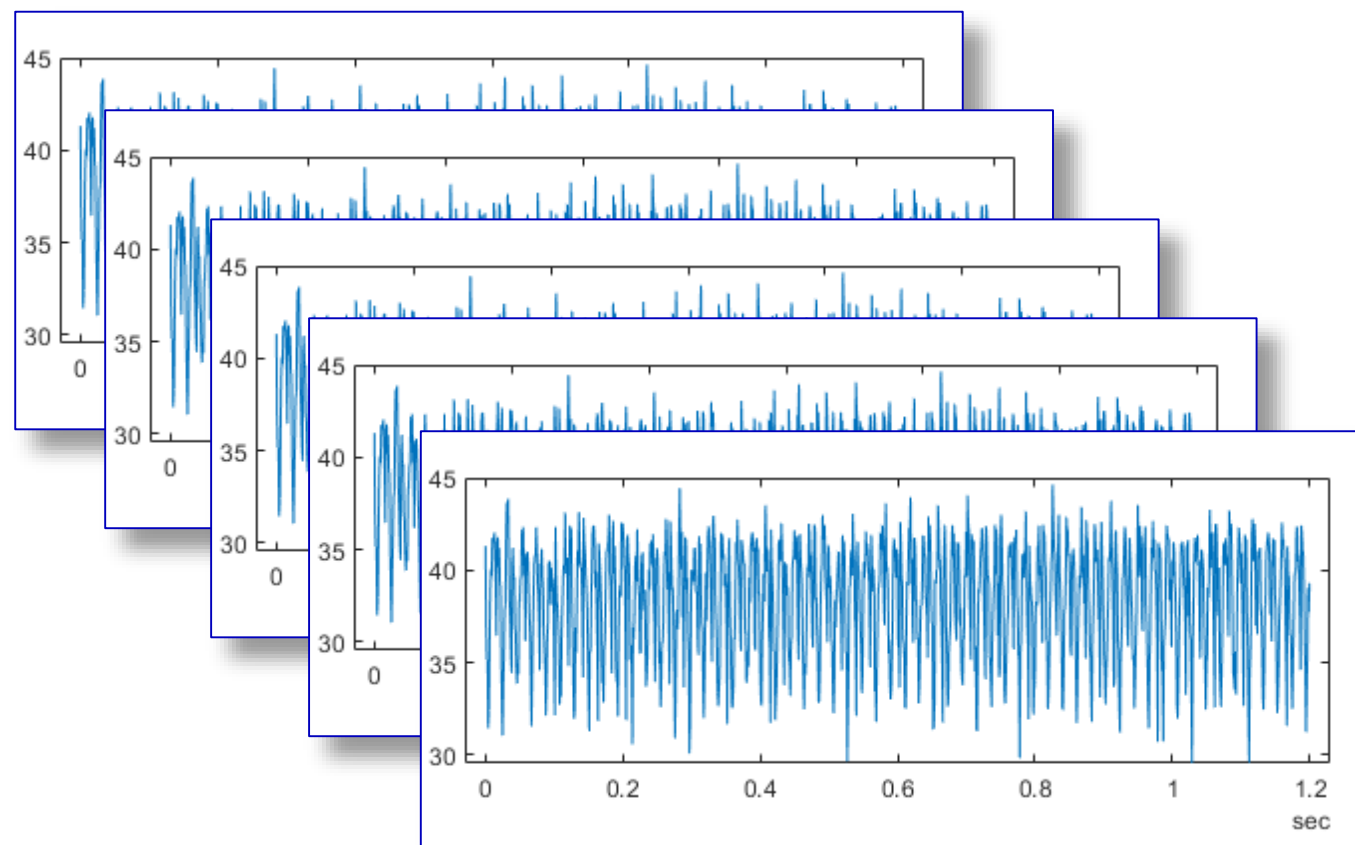
- **Reduce the amount of data you need to store and transmit**
- Explore approaches to feature extraction and predictive modeling
- Deliver the results of your analytics based on your audience
- Get started quickly...especially if you are an engineer



# How do you make sense of the ALL the data being collected?

- 1 day ~ 1.3 GB
- 20 sensors/pump ~26 GB/day
- 3 pumps ~ 78 GB/day
- **Satellite transmission**
  - Speeds approx. 128-150 kbps
  - Cost \$1,000/ 10GB of data
- **Needle in a haystack problem**

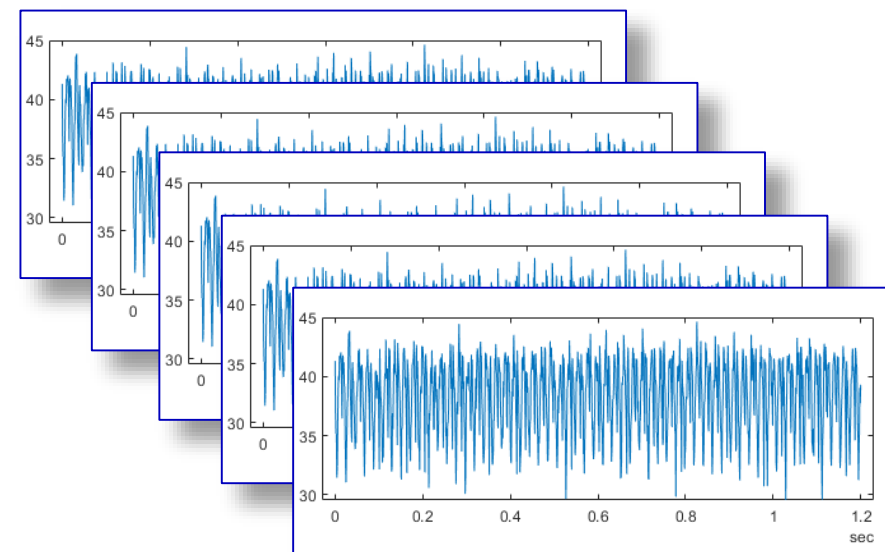
Pump flow sensor 1 sec ~ 1000 samples ~16kB



# Solution: Feature Extraction

Reduce the amount of data you need to store and transmit

- **How do you extract features?**
  - Signal processing methods
  - Statistics & model-based methods
  
- **Which features should you extract?**
  - Depends on the data available
  - Depends on the hardware available
  
- **How do I deal with streaming data?**
  - Determine buffer size
  - Extract features over a moving buffer window



qMean	qVar	qSkewness	qKurtosis
38.4945	9.2306	-0.5728	2.4662
qPeak2P...	qCrest	qRMS	qMAD
15.2351	1.1553	38.6141	2.5562

# 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



FEATURE DESIGNER | SIGNAL TRACE | VIEW

Open Session | Save Session | Import Data

Signal Trace

Computation Options

Filtering & Averaging | Residue Generation | Spectral Estimation

Time-Domain Features | Spectral Features

Rank Features | Export

FILE | PLOT | COMPUTATION | DATA PROCESSING | FEATURE GENERATION | RANKING | EXPORT

Data Browser

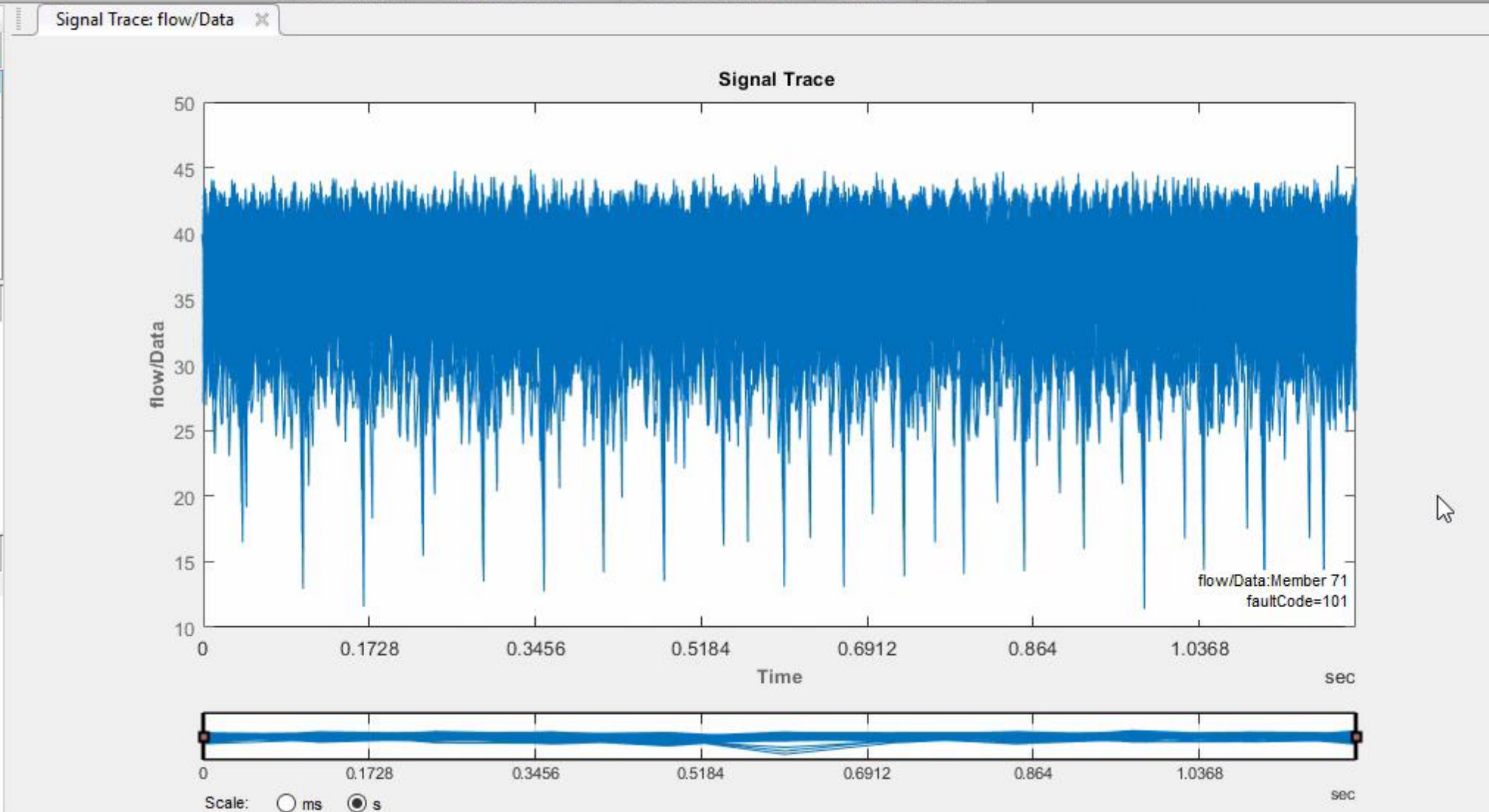
▼ Signals & Spectra

- flow/Data
- pressure/Data

▼ Feature Tables

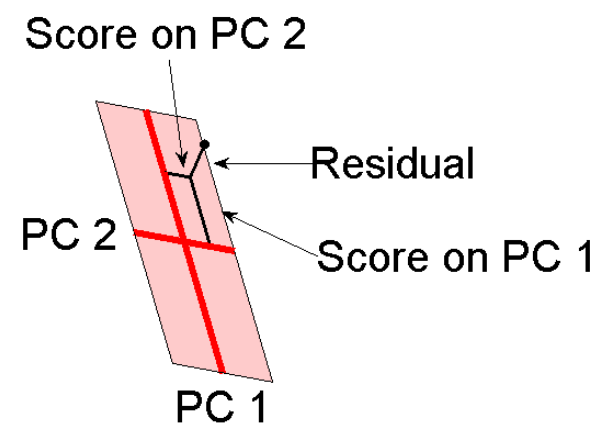
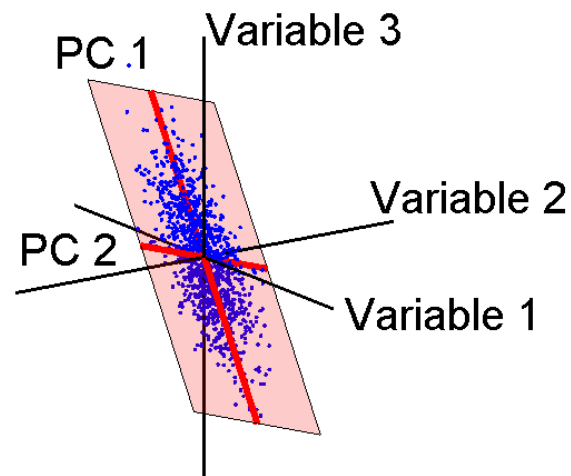
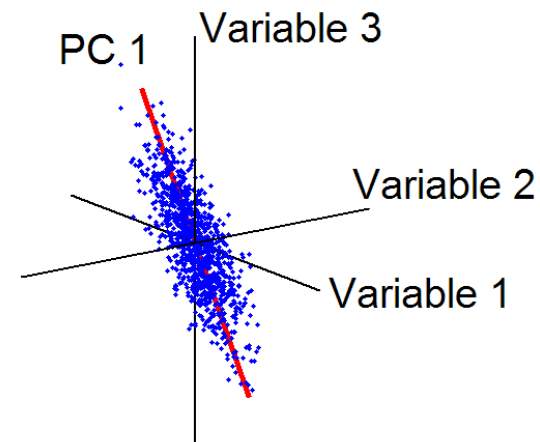
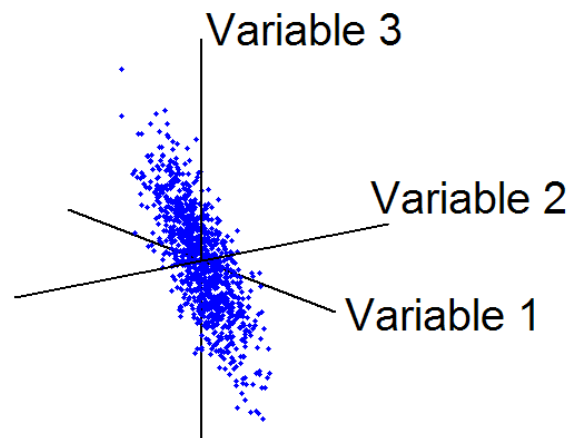
▼ Datasets

- Ensemble1



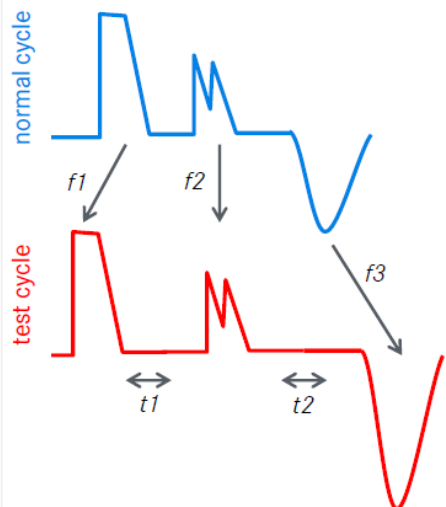


# Principal Components Analysis – what is it doing?



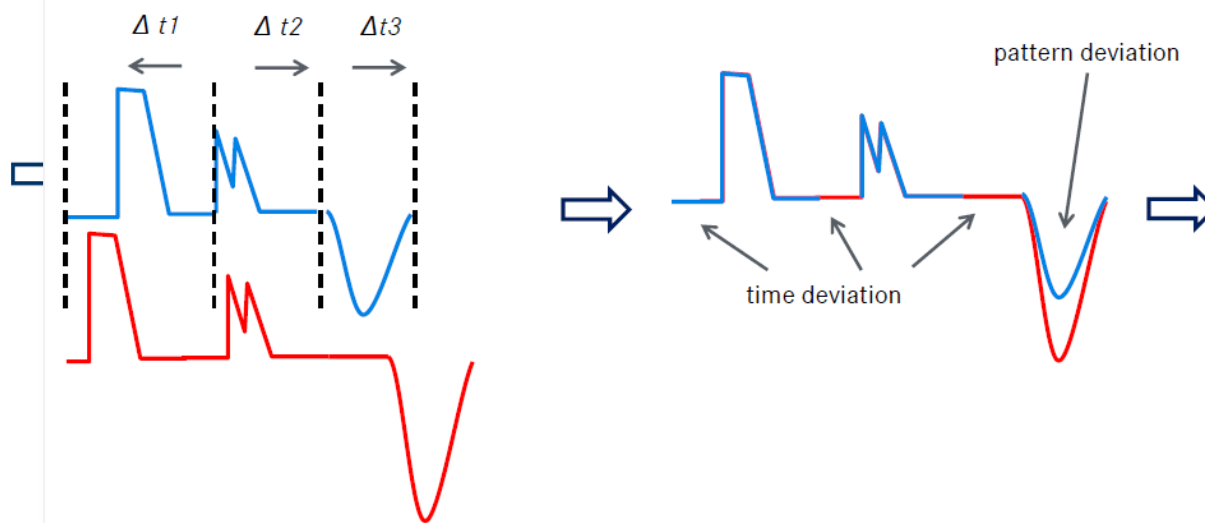
# Daimler are Using MATLAB Today for Anomaly Detection

## Algorithm principle



- Cycle can be described as sequence features  $f_1, f_2, f_3$
- Each cycle can show some delays in time  $t_1, t_2$

## Algorithm principle



- Pattern matching through shift of feature along time axis ( $\Delta t_1, \Delta t_2, \Delta t_3$ ): minimization of SRS

- Description of a cycle as feature sequence
- For each feature time and pattern deviation can be calculated

$f_1$	$f_2$	$f_3$	
$\Delta t_1$	$\Delta t_2$	$\Delta t_3$	Time deviation
No	No	Yes	Pattern deviation

- Time and pattern deviation for each feature are used as characteristic numbers for test cycle

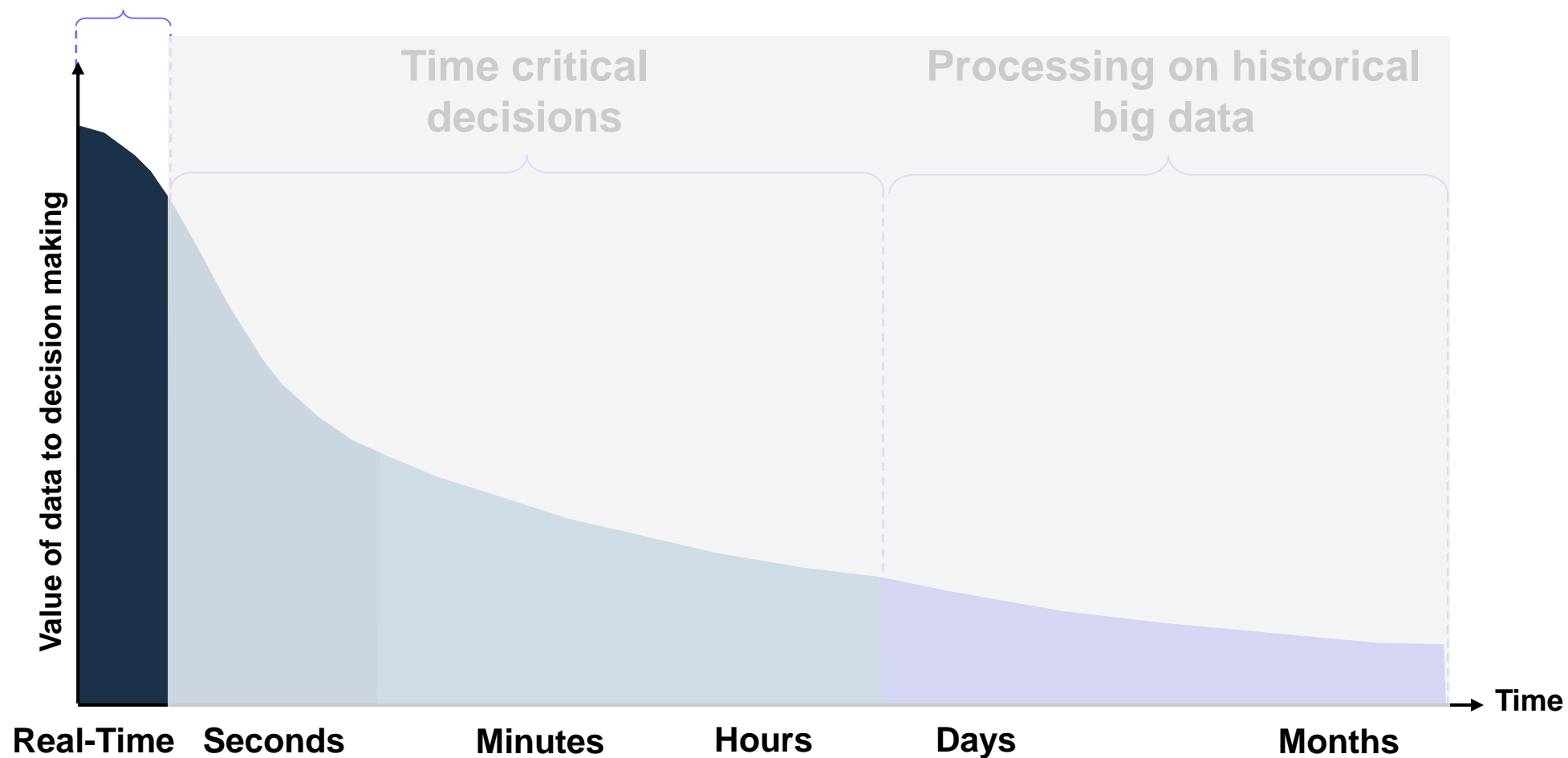


Data reduction!

**Data reduction of time series by a factor of 250x without a significant loss of information**

# When is Your Data Most Valuable?

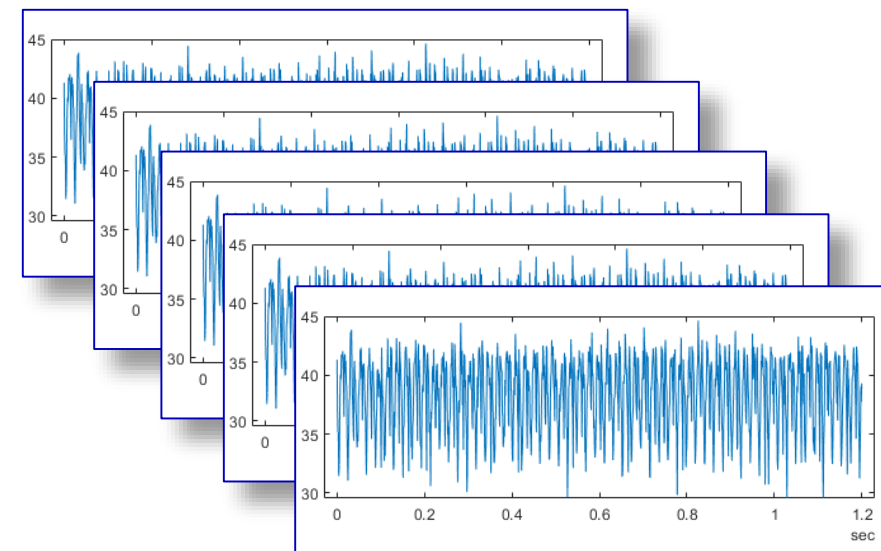
Near real-time decisions



# Solution: Feature Extraction at the Edge

Reduce the amount of data you need to store and transmit

- Design your feature extraction algorithm in MATLAB, then automatically convert MATLAB to C/C++
  - Eliminate chance for coding-errors
  - Implement new versions quicker
  - Maintain only one source (MATLAB)
  - Process data in real-time



qMean	qVar	qSkewness	qKurtosis
38.4945	9.2306	-0.5728	2.4662
qPeak2P...	qCrest	qRMS	qMAD
15.2351	1.1553	38.6141	2.5562

FILE NAVIGATE EDIT BREAKPOINTS RUN

New Open Save Find Files Compare Print Go To Find Insert Comment Indent Breakpoints Run Run and Advance Run Section Advance Run and Time

C:\Users\abaru\Desktop\Expo 2018\FinalDemo\Demo\_Files\Data\_Reduction

Current Folder

- Folder
  - codegen
  - Copy\_of\_Data
  - Data
- Function
  - featureExtraction.m
  - featureExtractionBuffer.m
  - helperSortedBarPlot.m
  - monotonicity.m
- MEX-file
  - featureExtraction\_mex.mexw64
  - featureExtractionBuffer\_mex.mexw64
- Live Script
  - Expo\_Data\_Preprocessing\_CodeGe...
- MATLAB Code Project
  - featureExtraction.prj
  - featureExtractionBuffer.prj

featureExtractionBuffer.m (Function)

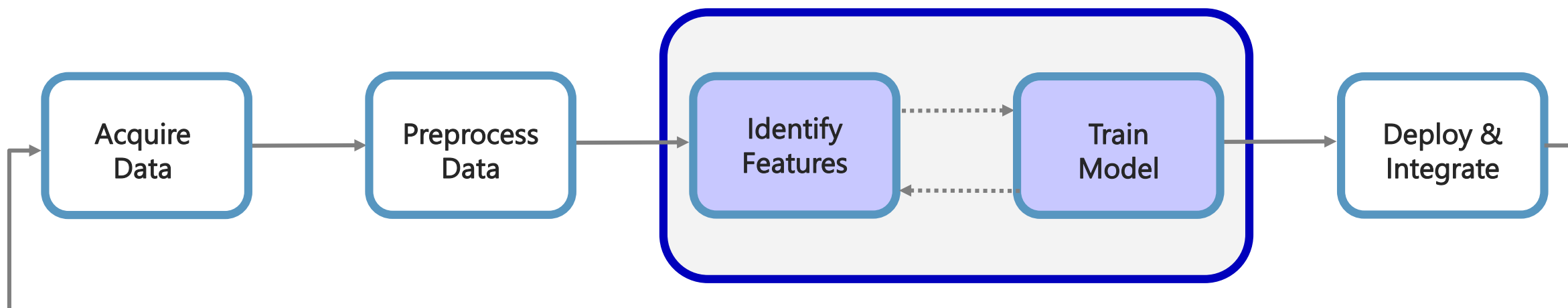
Editor - C:\Users\abaru\Desktop\Expo 2018\FinalDemo\Demo\_Files\Data\_Reduction\featureExtractionBuffer.m

Expo\_Data\_Preprocessing\_CodeGen.mlx featureExtractionBuffer.m

```
1 function [feature_list] = featureExtractionBuffer(data,timestamp)
2
3 persistent flow_array
4 persistent time_array
5 Np = 1000;
6
7 if isempty(flow_array)
8     flow_array = nan(Np,1);
9 end
10
11 if isempty(time_array)
12     time_array = nan(Np,1);
13 end
14
15 flow_array = [data; flow_array(1:Np-1)];
16 data = flow_array;
17
18 time_array = [timestamp; time_array(1:Np-1)];
19 timestamp = time_array;
20
21
22 if isempty(find(isnan(data),1))
23
24     flow = data;
25
26     % Ensure the flow is sampled at a uniform sample rate
27     t_flow = timestamp;
```

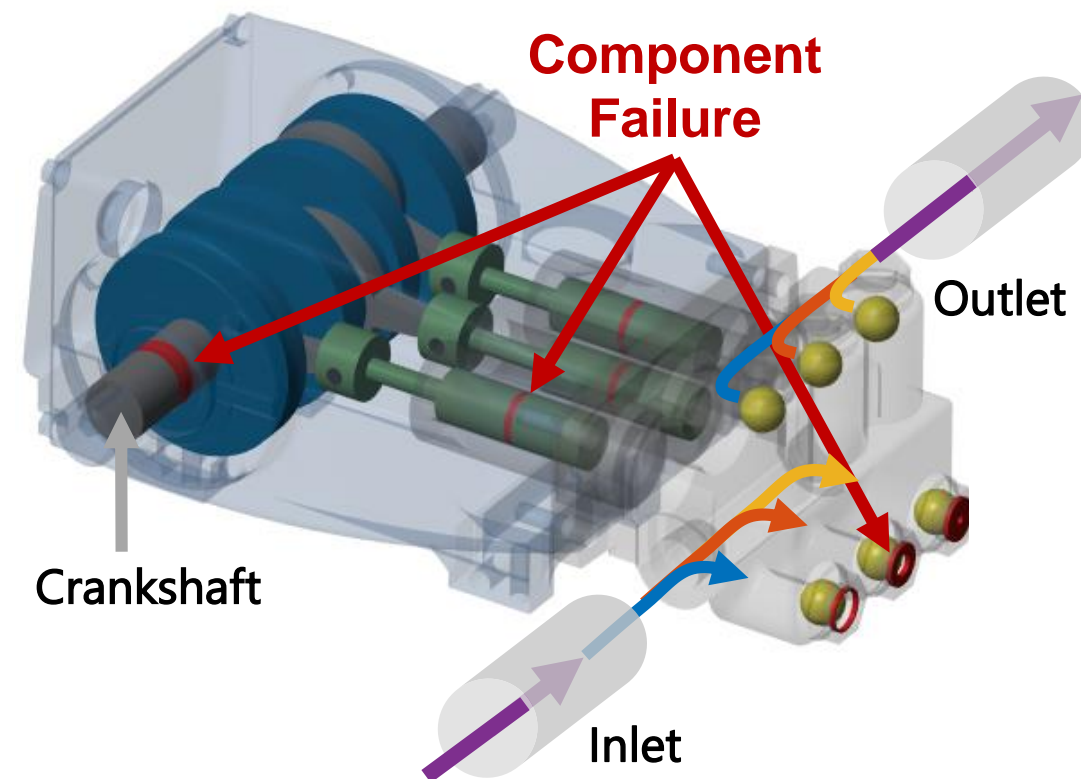
# Predictive Maintenance Challenges

- Reduce the amount of data you need to store and transmit
- **Explore approaches to feature extraction and predictive modeling**
- Deliver the results of your analytics based on your audience
- Get started quickly...especially if you are an engineer



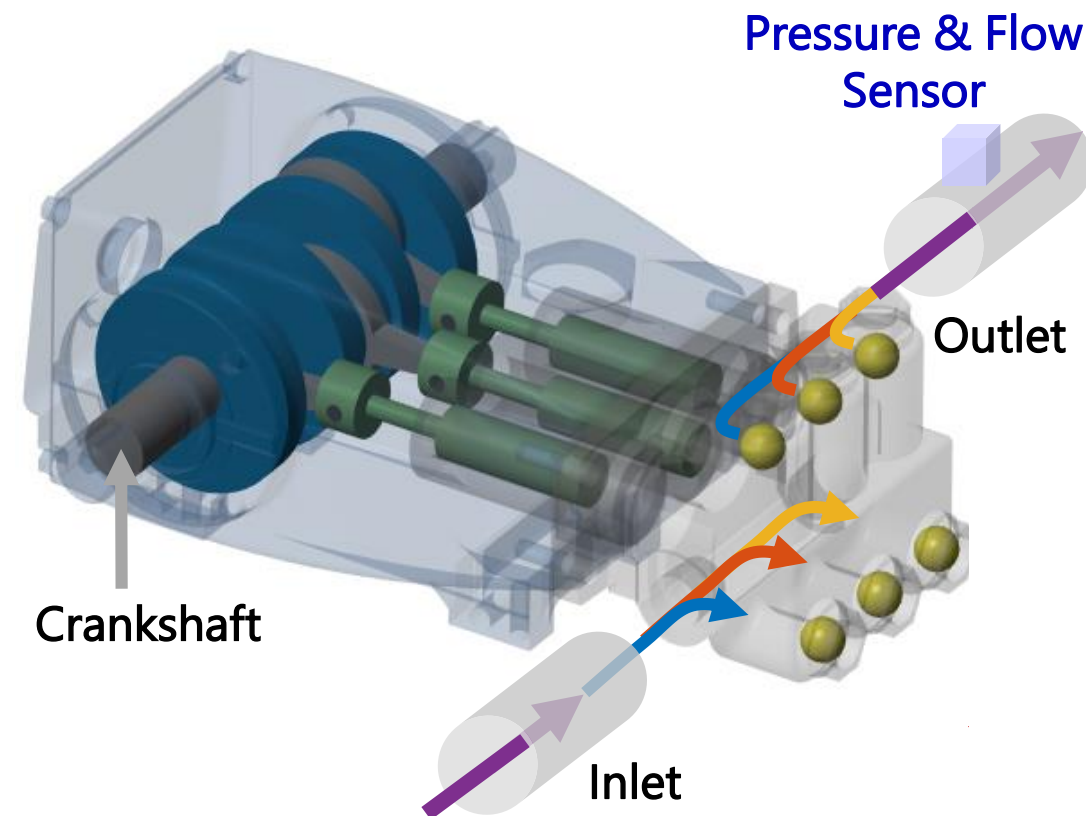
# Fault Classification Algorithms Allow You to Identify the Root Cause of Anomalous Behavior

- Three-phase pump commonly used for drilling and servicing oil wells
  - Three plungers try to ensure a uniform flow
- Condition monitoring to detect:
  - Seal leak
  - Inlet blockage
  - Bearing degradation



# Fault Classification Algorithms Allow You to Identify the Root Cause of Anomalous Behavior

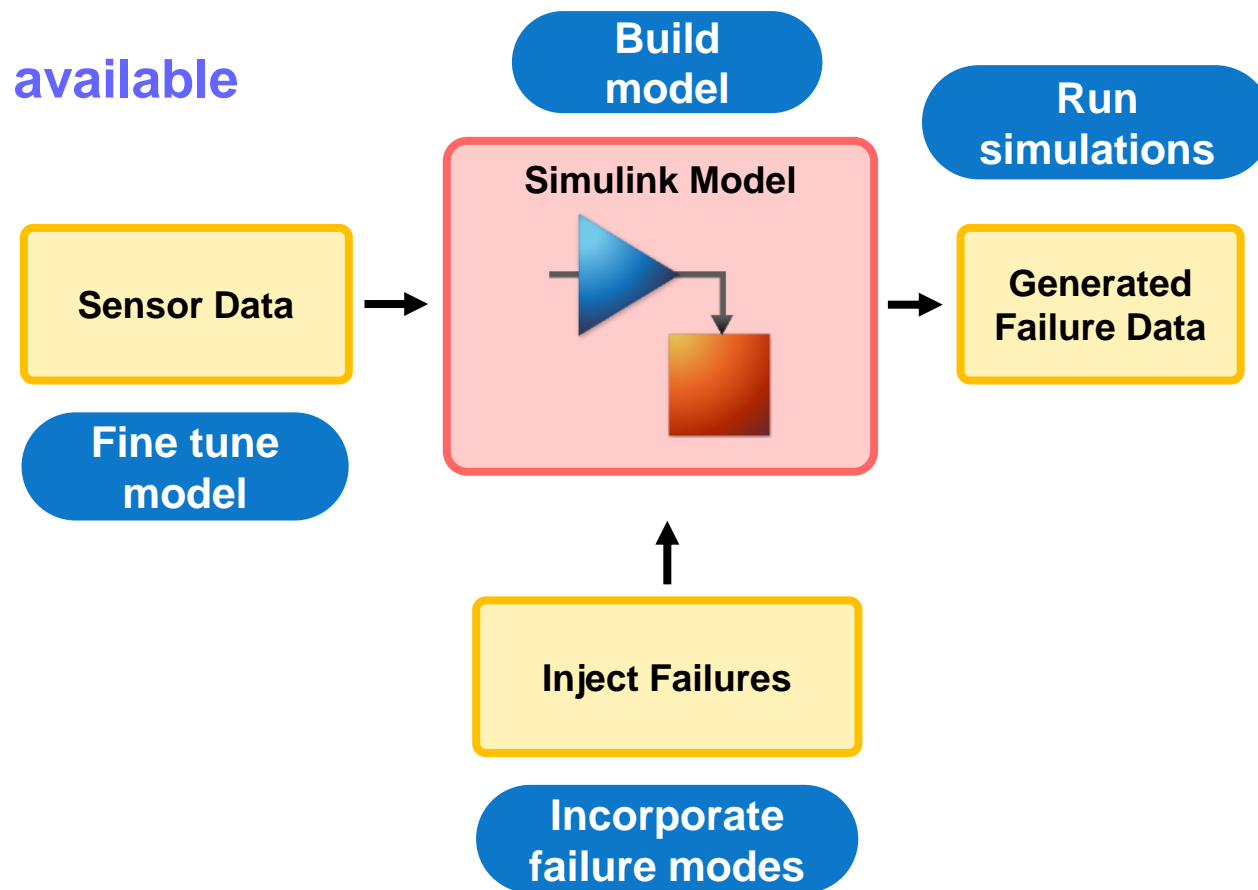
- Three-phase pump commonly used for drilling and servicing oil wells
  - Three plungers try to ensure a uniform flow
- Condition monitoring to detect:
  - Seal leak
  - Inlet blockage
  - Bearing degradation
- Identify fault present in system using only pressure and flow sensor data





# Generate Synthetic Failure Data from Simulink Models if Real Failure Data is Unavailable

- **Model failure modes**
  - Work with domain experts and the data available
  - Vary model parameters/components
- **Customize a generic model to a specific machine**
  - Fine tune models based on real data
  - Validate performance of tuned model



FEATURE DESIGNER | SIGNAL TRACE | VIEW

Open Session | Save Session | Import Data

Select data to plot

Signal Trace | Power Spectrum | Order Spectrum | Histogram

Computation Options

Filtering & Averaging | Residue Generation | Spectral Estimation

Time-Domain Features | Spectral Features

Rank Features | Export

FILE | PLOT | COMPUTATION | DATA PROCESSING | FEATURE GENERATION | RANKING | EXPORT

Data Browser

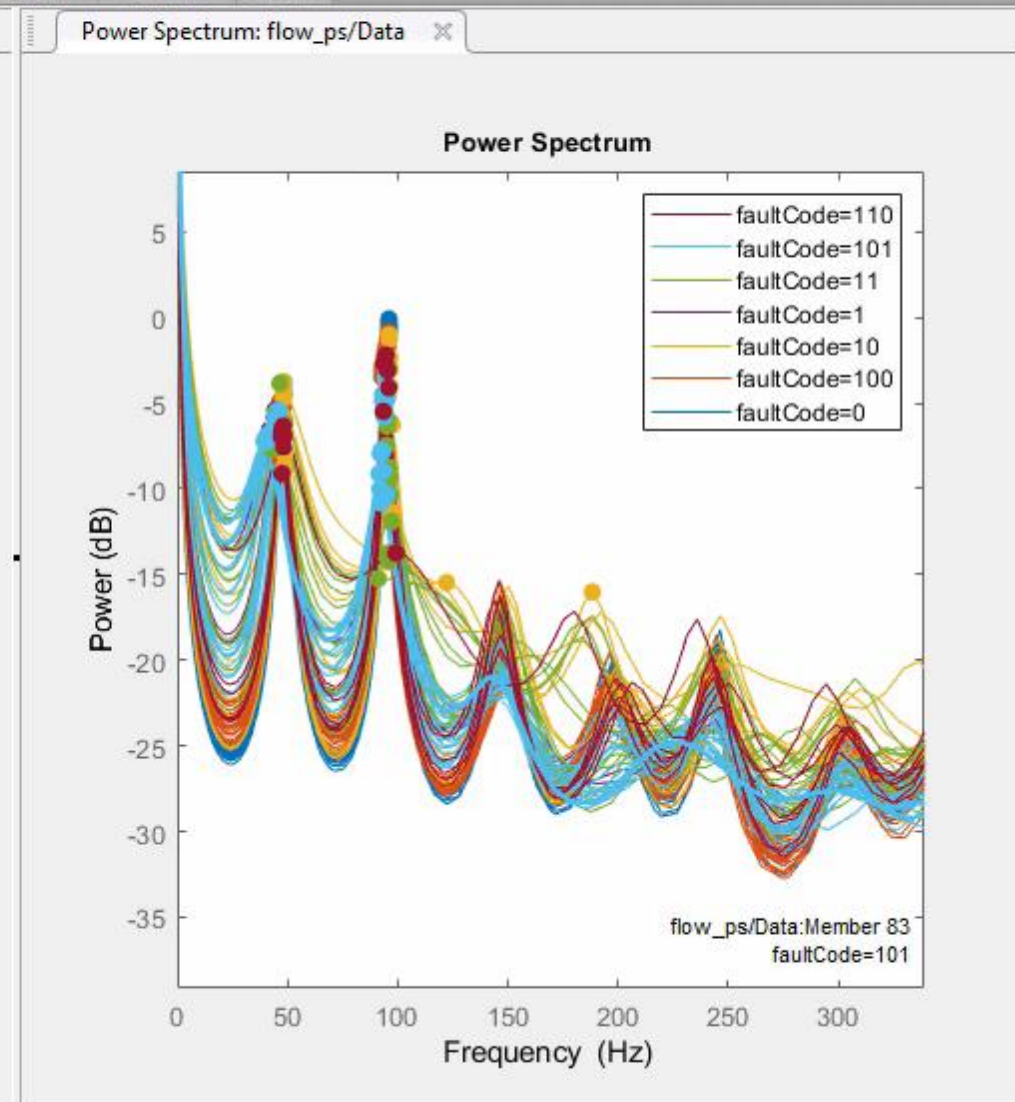
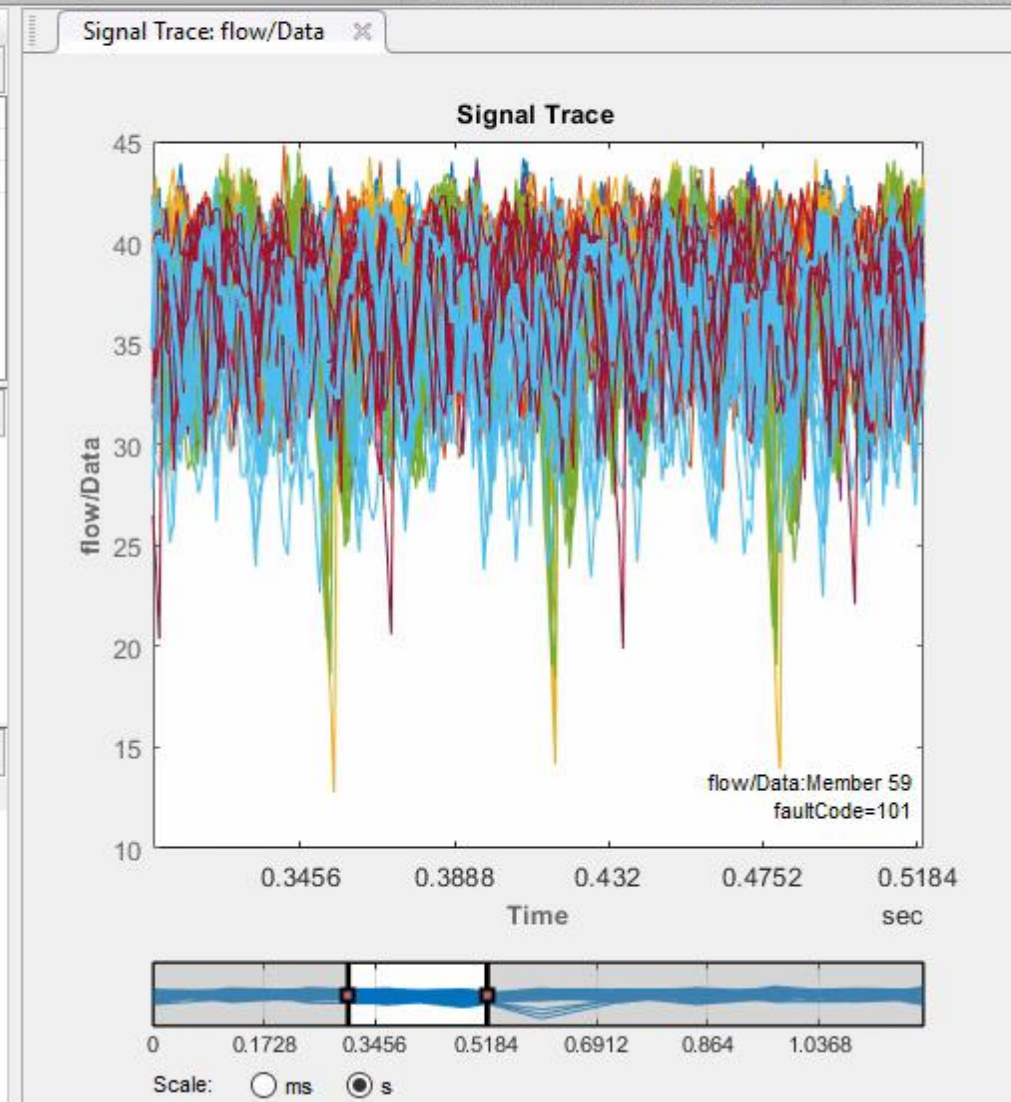
▼ Signals & Spectra

- flow/Data
- pressure/Data
- flow\_ps/Data

▼ Feature Tables

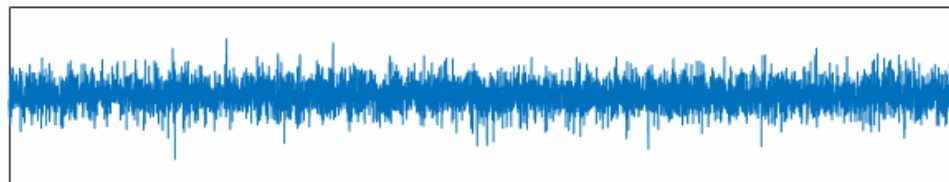
▼ Datasets

- Ensemble1

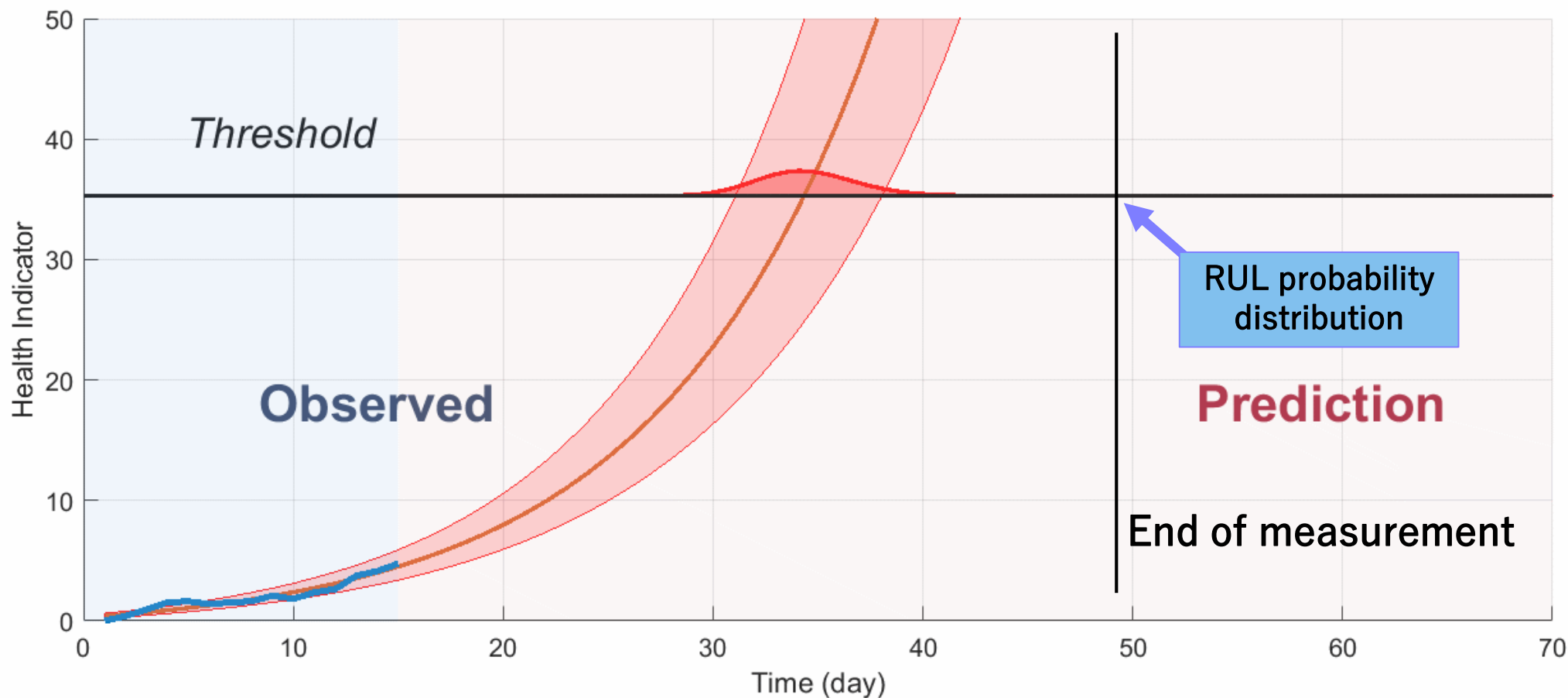


# Estimate Remaining Useful Live (RUL)

to Determine When You Should Perform Maintenance

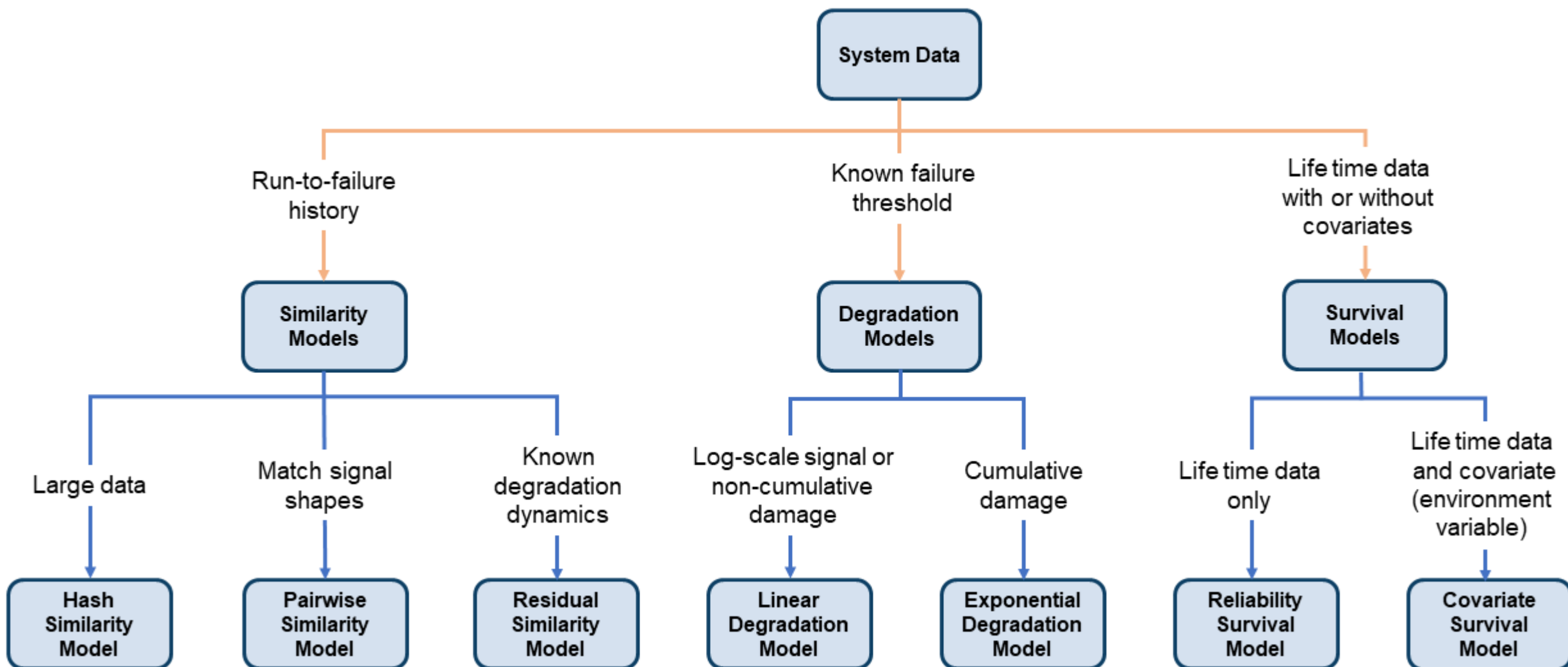


RUL: 459 hours  
(95%CI: 374-558 hours)



# Estimate Remaining Useful Live (RUL)

to Determine When You Should Perform Maintenance



# Estimate Remaining Useful Live (RUL)

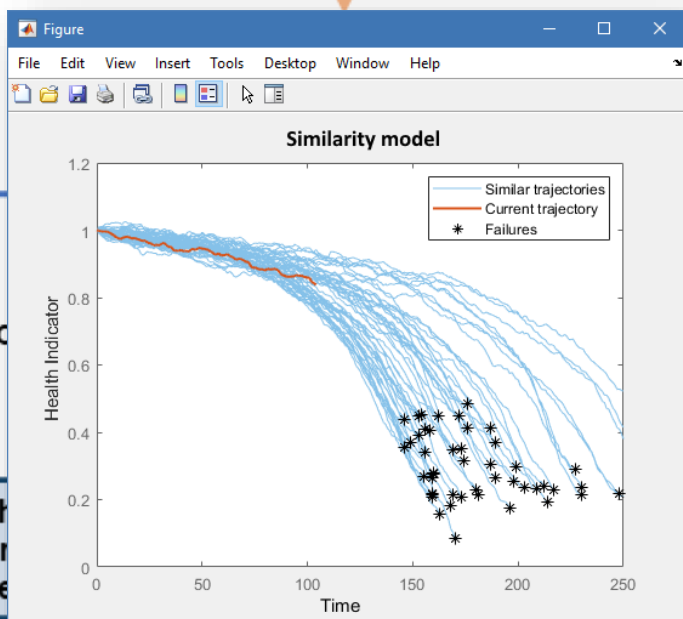
to Determine When You Should Perform Maintenance

System Data

Run-to-failure history

Known failure threshold

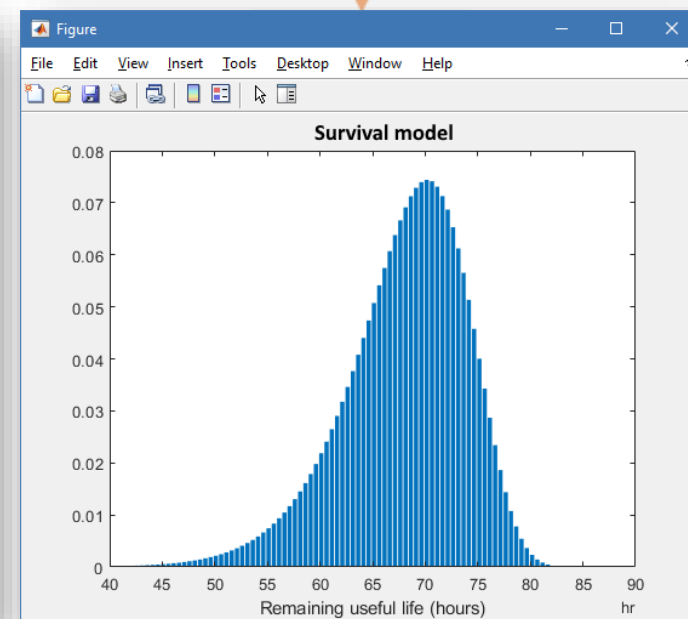
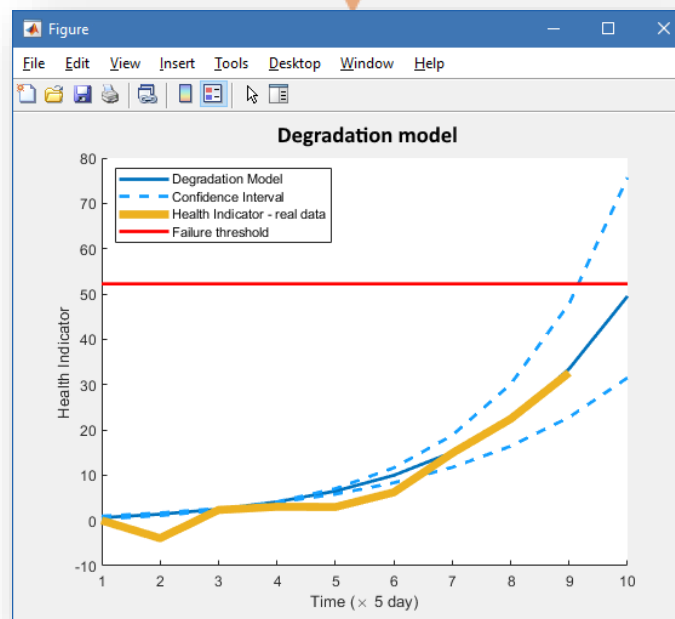
Life time data with or without covariates



Large c

Known degradation dynamics

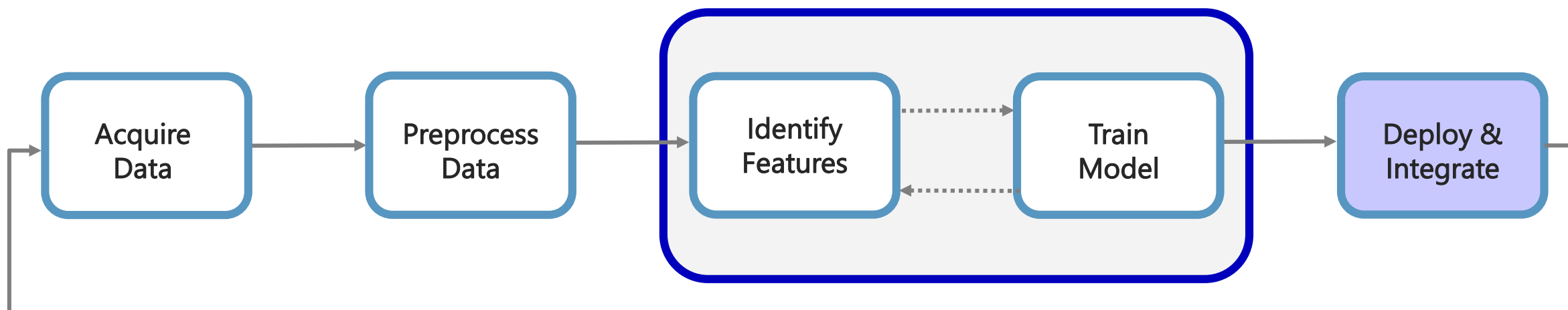
Residual similarity model



Hash Similarity Model

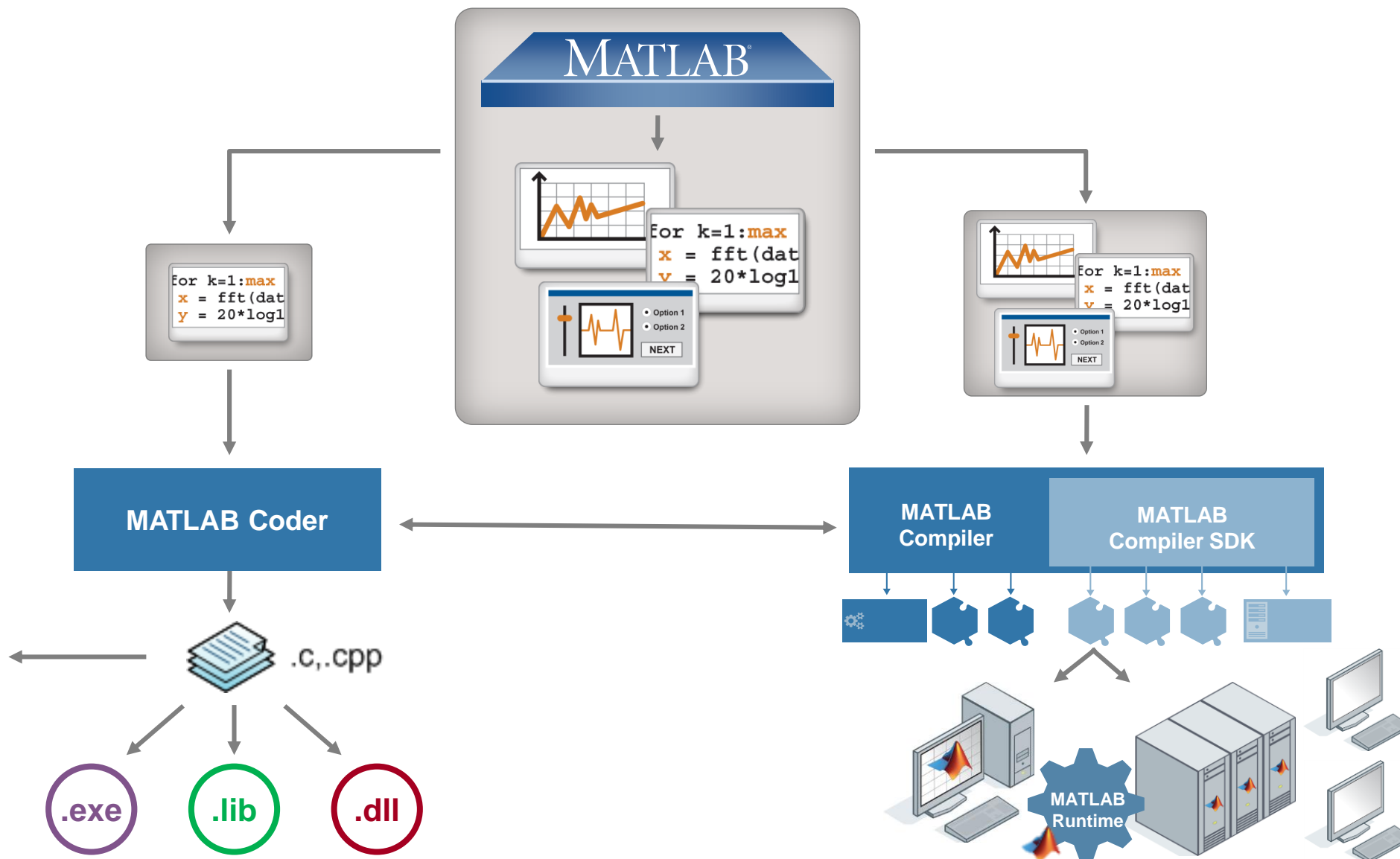
# Predictive Maintenance Challenges

- Reduce the amount of data you need to store and transmit
- Explore approaches to feature extraction and predictive modeling
- **Deliver the results of your analytics based on your audience**
- Get started quickly...especially if you are an engineer

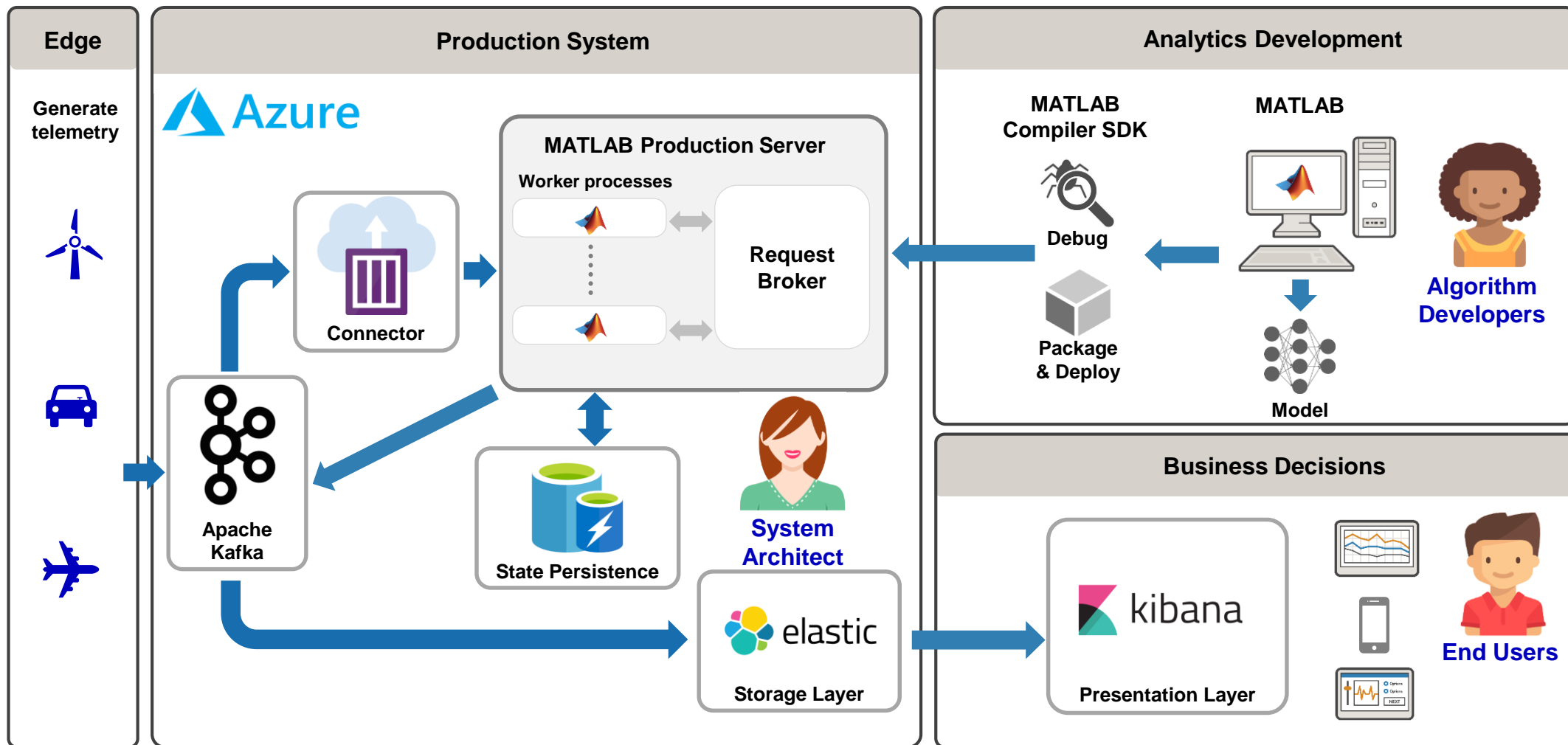


# Integrate Analytics with Your Enterprise Systems

## MATLAB Compiler and MATLAB Coder

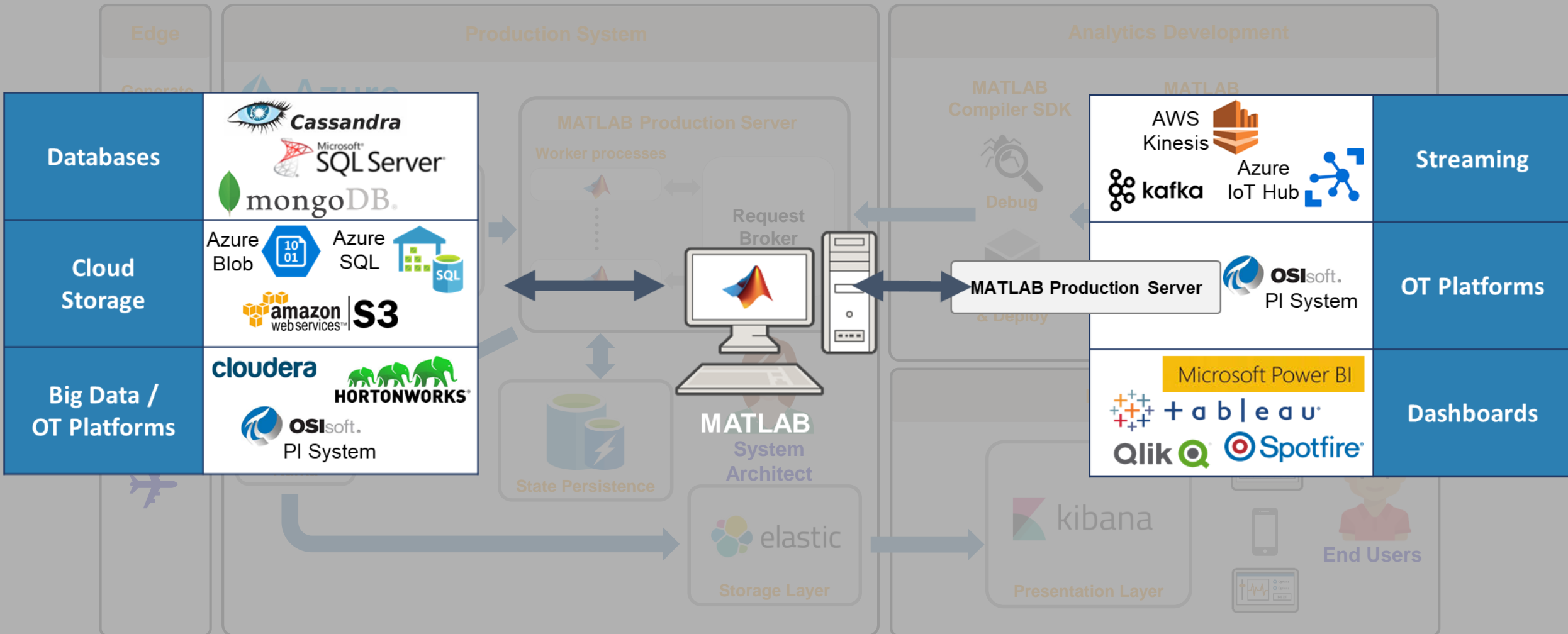


# Predictive Maintenance Architecture on Azure





# Predictive Maintenance Architecture on Azure

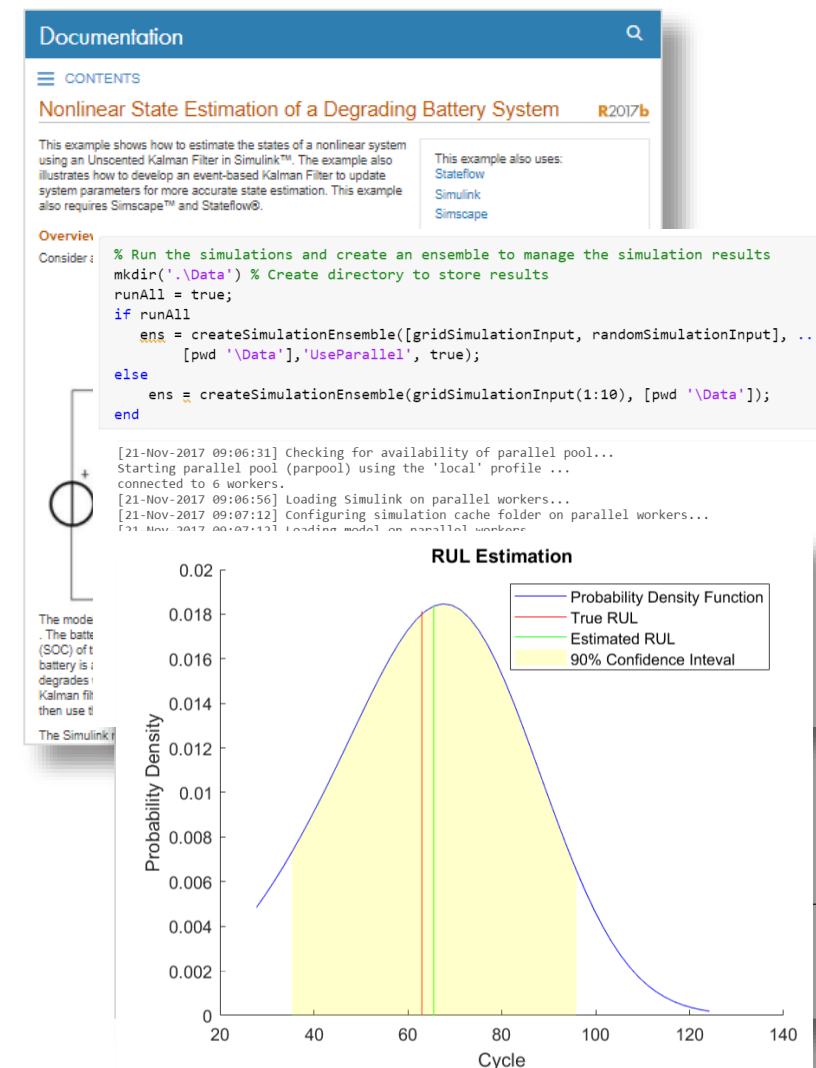


# Predictive Maintenance Challenges

- Reduce the amount of data you need to store and transmit
- Explore approaches to feature extraction and predictive modeling
- Deliver the results of your analytics based on your audience
- **Get started quickly...especially if you are an engineer**

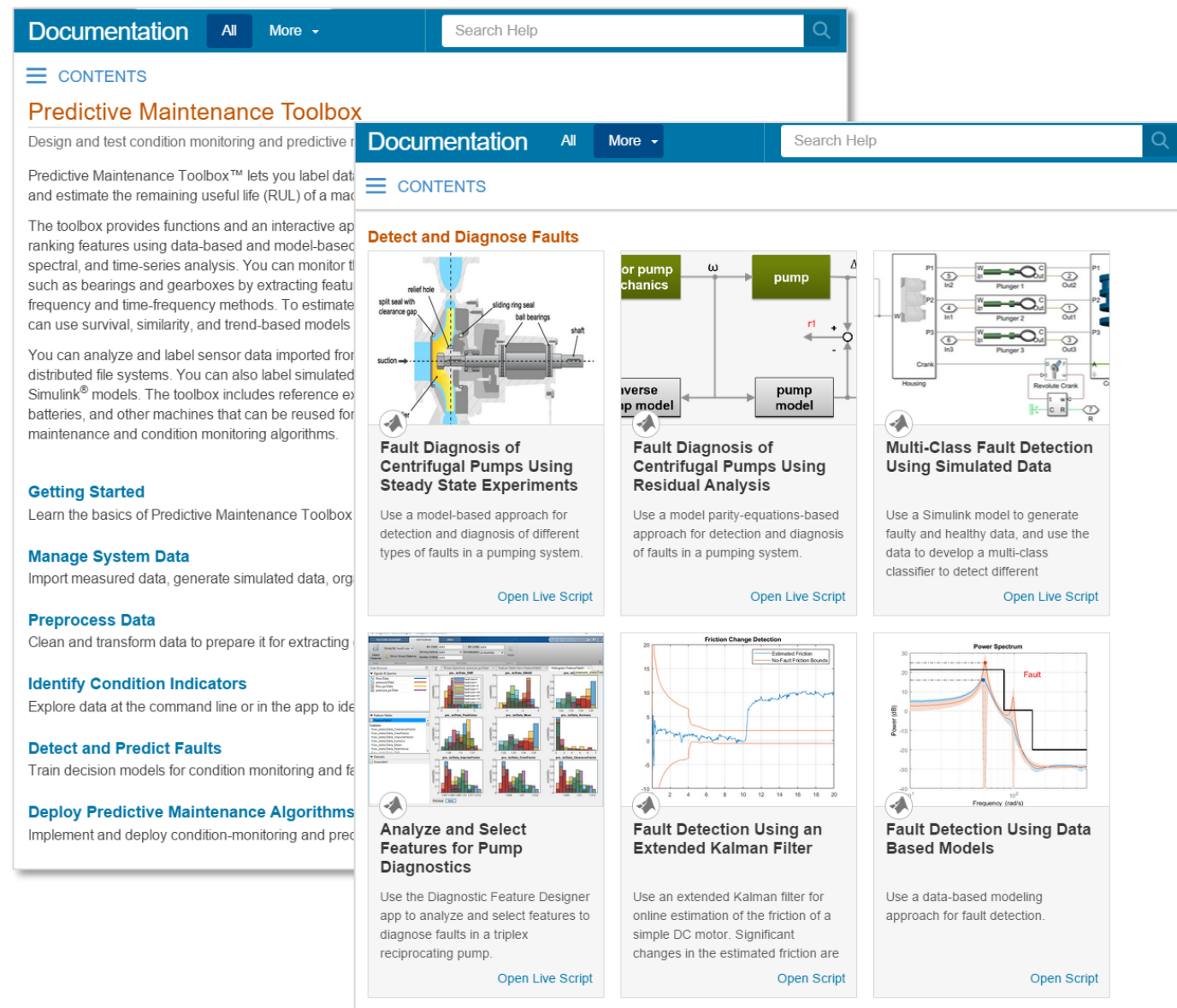
# How can the Predictive Maintenance Toolbox help you?

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



# MATLAB can help you get started TODAY

- Examples
- Documentation
- Tutorials & Workshops
- Consulting
- Tech Talk Series

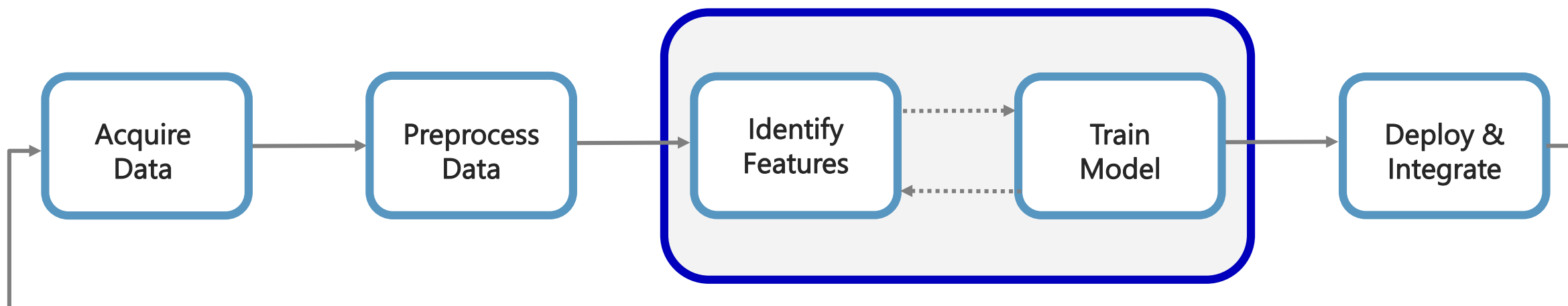


The screenshot displays the documentation for the Predictive Maintenance Toolbox, organized into several sections:

- Predictive Maintenance Toolbox**: Overview of design and test condition monitoring and predictive maintenance.
- Detect and Diagnose Faults**: A collection of diagnostic methods:
  - Fault Diagnosis of Centrifugal Pumps Using Steady State Experiments**: Uses a model-based approach for detection and diagnosis of different types of faults in a pumping system.
  - Fault Diagnosis of Centrifugal Pumps Using Residual Analysis**: Uses a model parity-equations-based approach for detection and diagnosis of faults in a pumping system.
  - Multi-Class Fault Detection Using Simulated Data**: Uses a Simulink model to generate faulty and healthy data, and use the data to develop a multi-class classifier to detect different faults.
  - Analyze and Select Features for Pump Diagnostics**: Uses the Diagnostic Feature Designer app to analyze and select features to diagnose faults in a triplex reciprocating pump.
  - Fault Detection Using an Extended Kalman Filter**: Uses an extended Kalman filter for online estimation of the friction of a simple DC motor. Significant changes in the estimated friction are detected.
  - Fault Detection Using Data Based Models**: Uses a data-based modeling approach for fault detection.
- Getting Started**: Learn the basics of Predictive Maintenance Toolbox.
- Manage System Data**: Import measured data, generate simulated data, organize data.
- Preprocess Data**: Clean and transform data to prepare it for extracting features.
- Identify Condition Indicators**: Explore data at the command line or in the app to identify condition indicators.
- Detect and Predict Faults**: Train decision models for condition monitoring and fault prediction.
- Deploy Predictive Maintenance Algorithms**: Implement and deploy condition-monitoring and predictive maintenance algorithms.

# Predictive Maintenance Challenges

- Reduce the amount of data you need to store and transmit
- Explore approaches to feature extraction and predictive modeling
- Deliver the results of your analytics based on your audience
- Get started quickly...especially if you are an engineer



**Děkuji za pozornost**