Developing Prognostics Algorithms: Data-Based and Model-Based Approaches

Seth DeLand

May 9, 2017

MathWorks Automotive Conference

What is Prognostics?

Prognostics

From Wikipedia, the free encyclopedia

This article is about the engineering discipline. For the medical term, see prognosis.

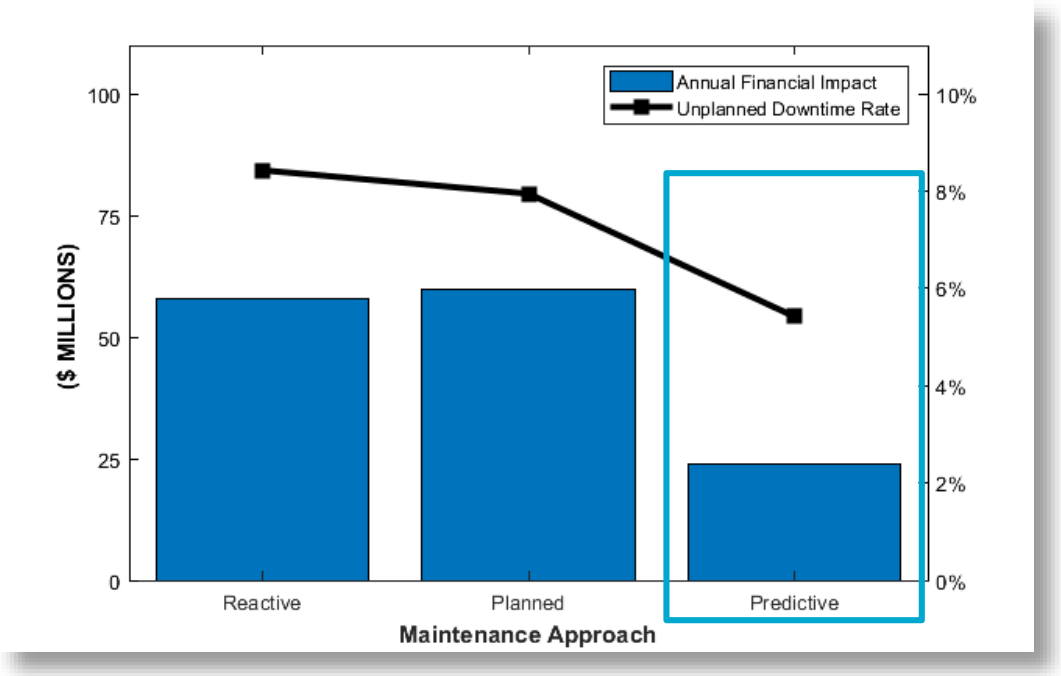
Prognostics is an engineering discipline focused on predicting the time at which a system or a component [1] will no longer perform its intended function.[2] This lack of performance is most often a failure beyond which the system can no longer be used to meet desired performance. The predicted time then becomes the **remaining useful life** (**RUL**), which is an important concept in decision making for contingency mitigation. Prognostics predicts the future

....

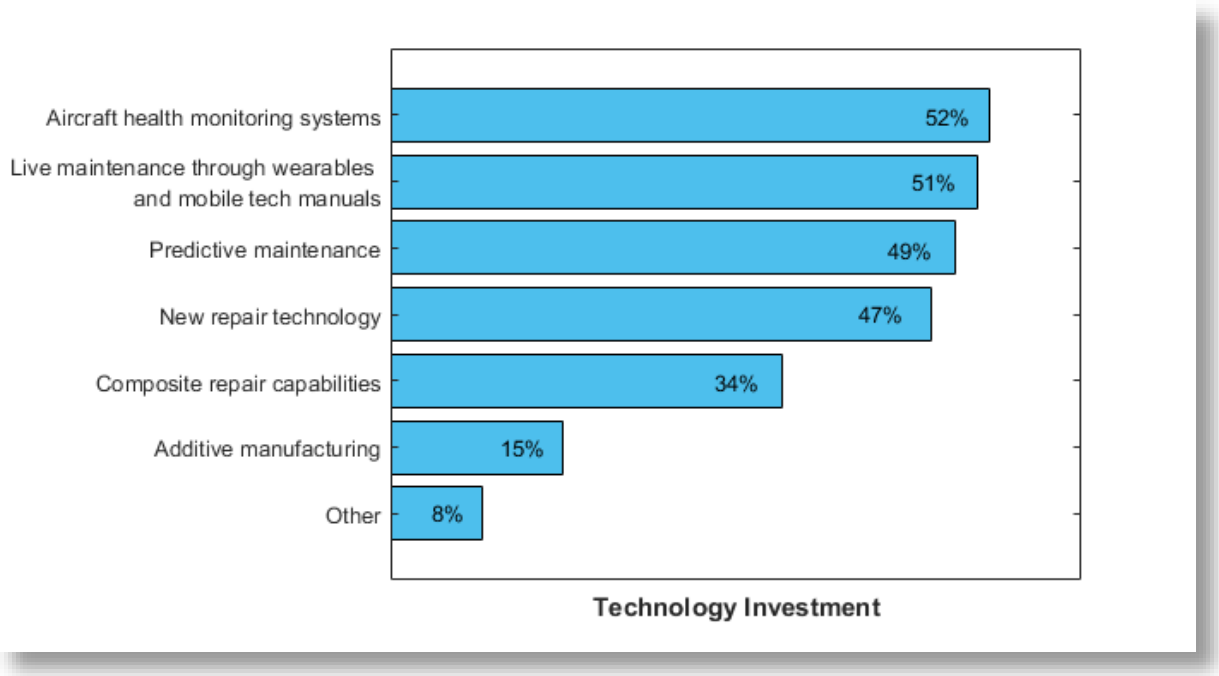
conditions.[3] The science of prognostics is based on the analysis of failure modes, detection of early signs of wear and aging, and fault conditions. An effective prognostics solution is implemented when there is sound knowledge of the failure mechanisms that are likely to cause the degradations leading to eventual failures in the system. It is therefore necessary to have initial

Why Prognostics?

- Improved operating efficiency
- New revenue streams
- Competitive differentiator



Source: GE Oil & Gas



Source: Oliver Wyman 2015 MRO Survey

How does it work? Prognostics Algorithm Workflow

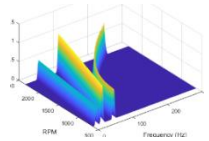
Access and Explore Data

Business Data

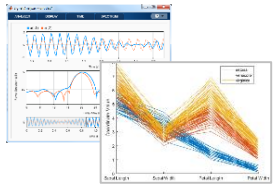
Sensor Data

Preprocess Data

Data Reduction/ Transformation



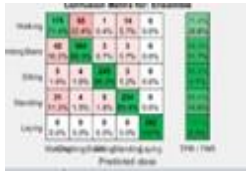
Feature Extraction



Develop Predictive Models

Model Creation

Model Validation

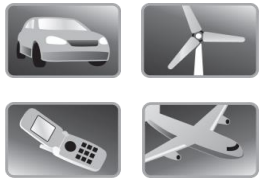


Integrate Analytics with Systems

Enterprise Systems

MATLAB Excel .NET .exe C/C++ Java .dll

Embedded Devices

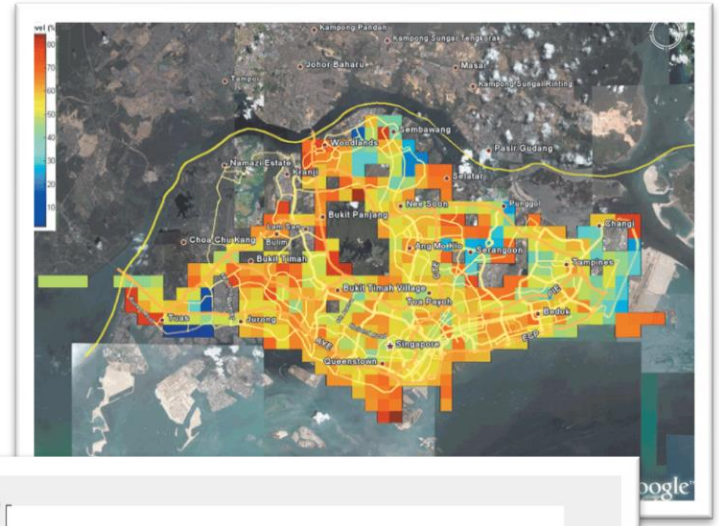


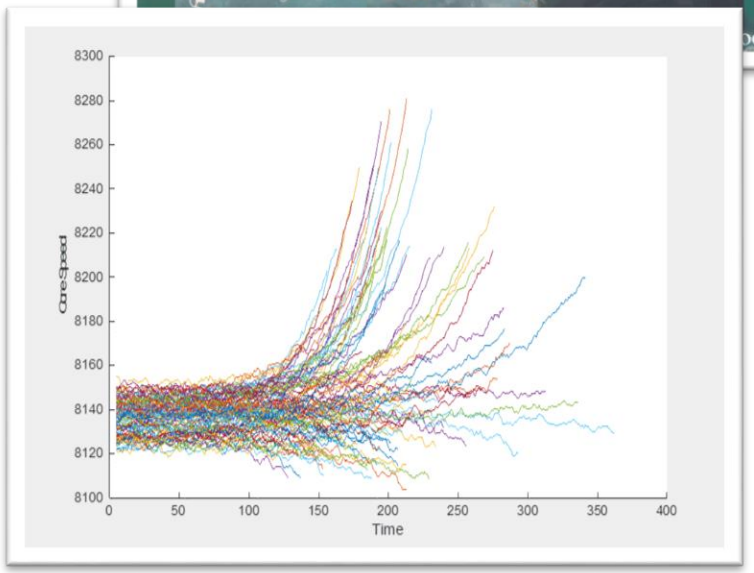
Challenges for Prognostics Development

- How long will it take to collect fault data?
- How expensive is it to collect?
- How complex is the system?

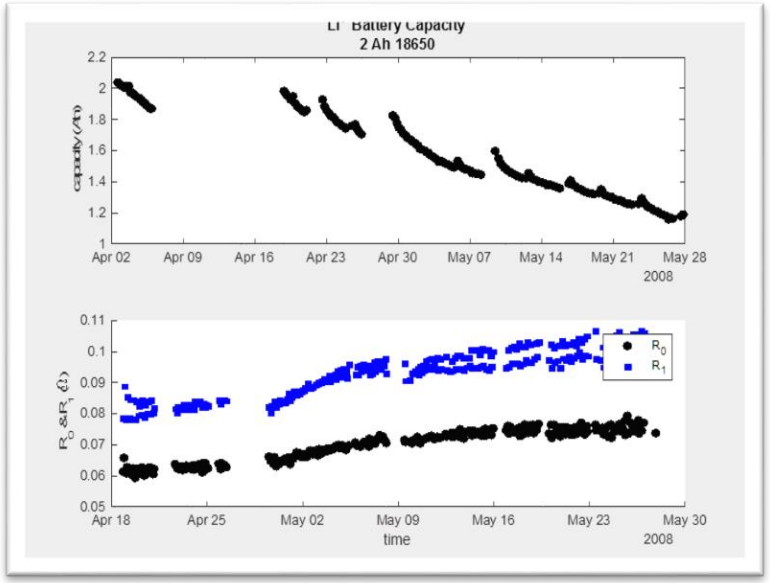
Sources of Data for Prognostics Development

Fleet

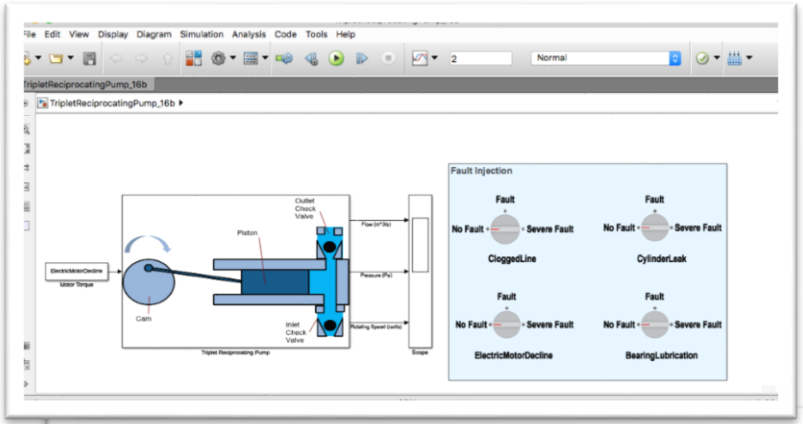


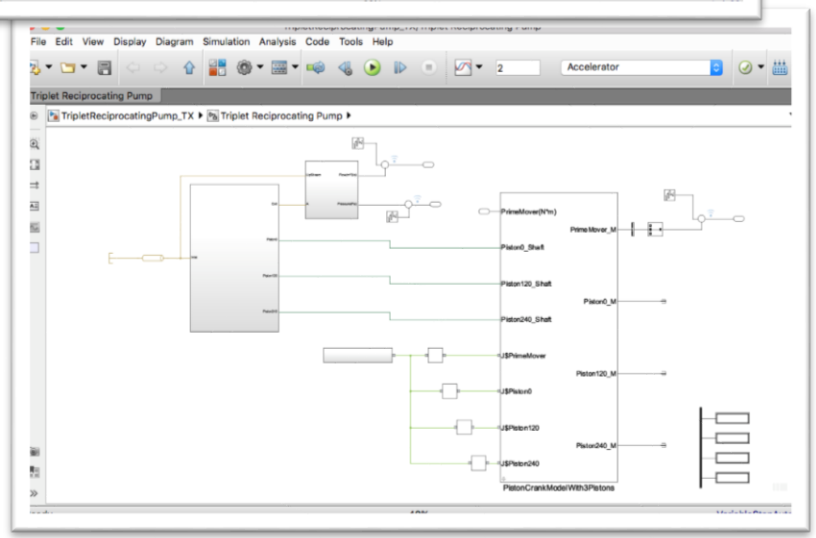


Experiments

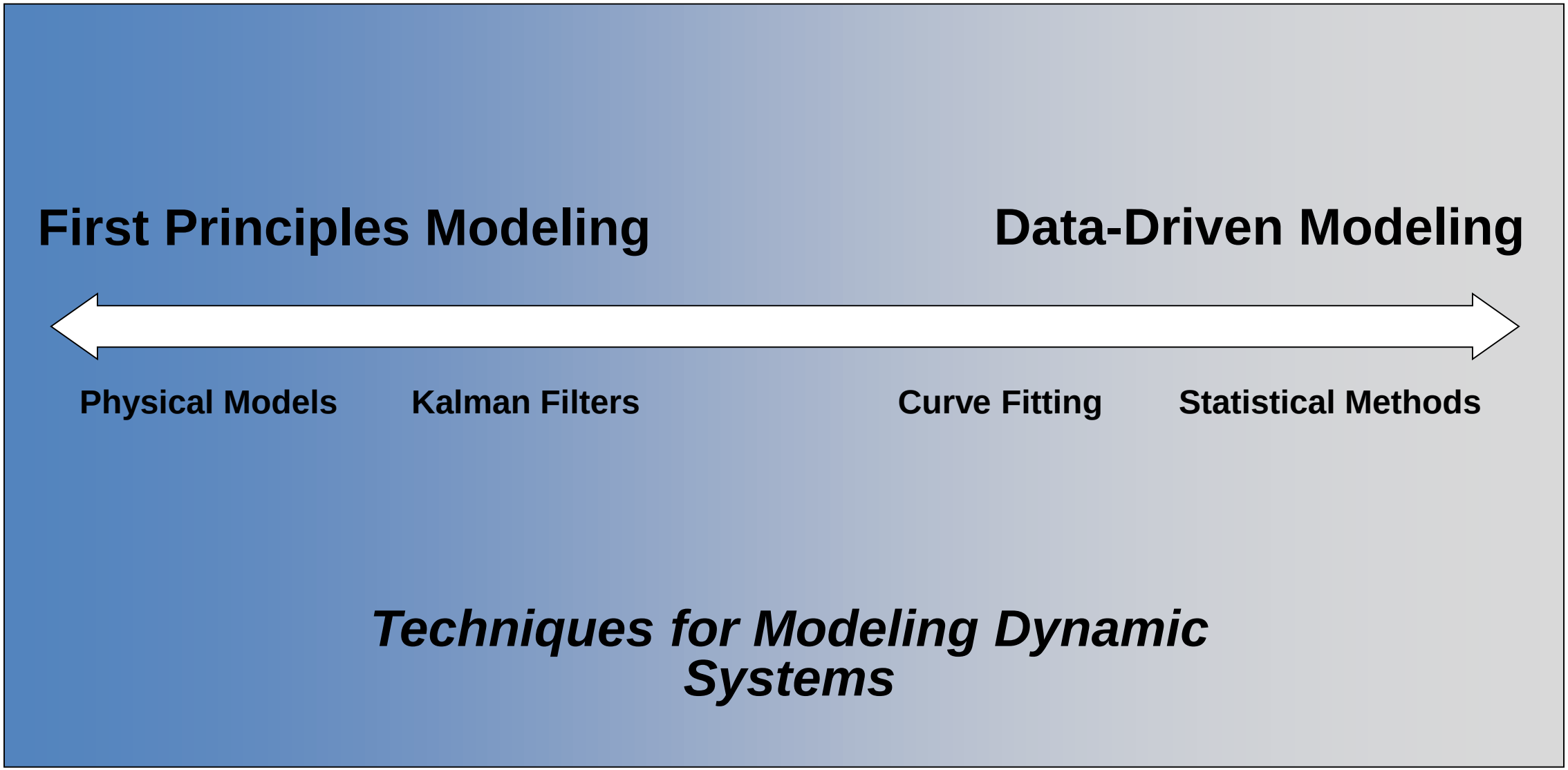


Simulation





Spectrum of Approaches for Prognostics Algorithms



Examples

1. Data-based prognostics using machine learning

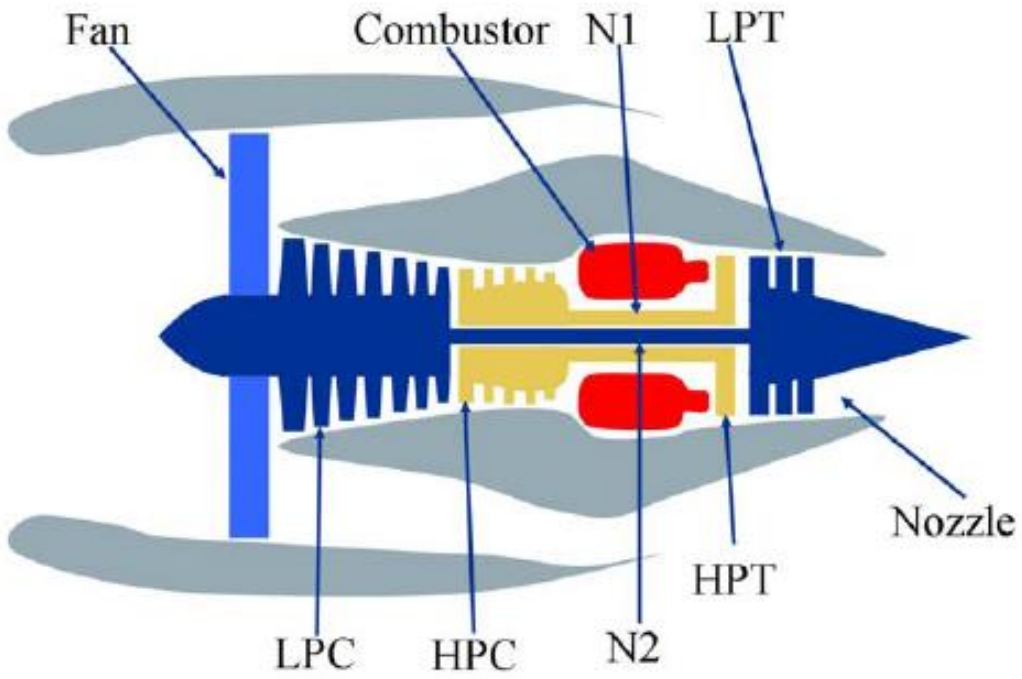
2. Fault injection and failure analysis using simulation

Examples

1. Data-based prognostics using machine learning

2. Fault injection and failure analysis using simulation

Example 1: Data-based Prognostics with Machine Learning



Data provided by NASA PCoE
http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/

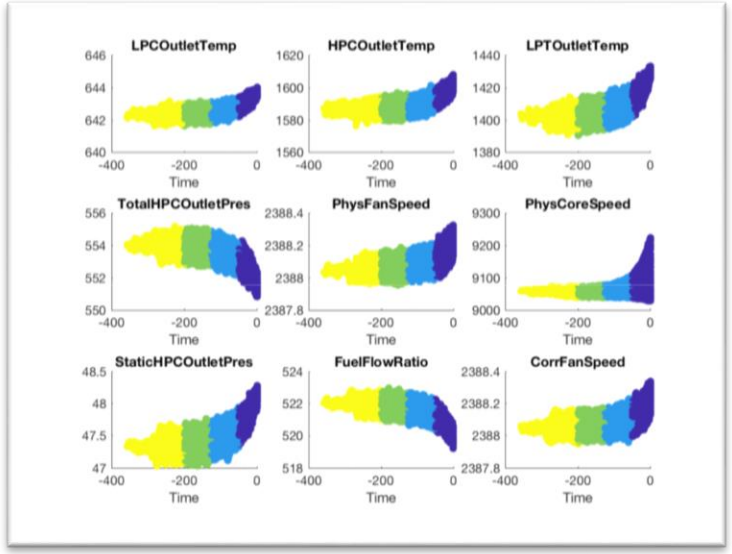
Different Types of Learning

Type of Learning

Machine Learning

Supervised Learning

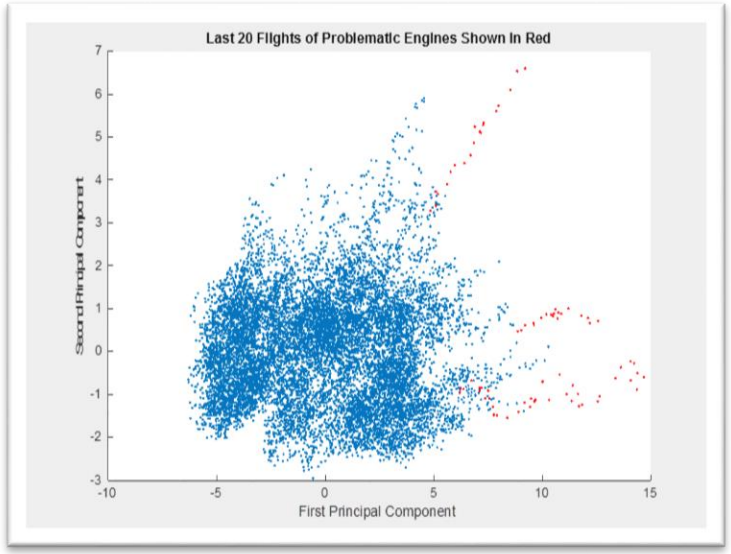
Develop **predictive model** based on both **input and output** data



Predicting Failures

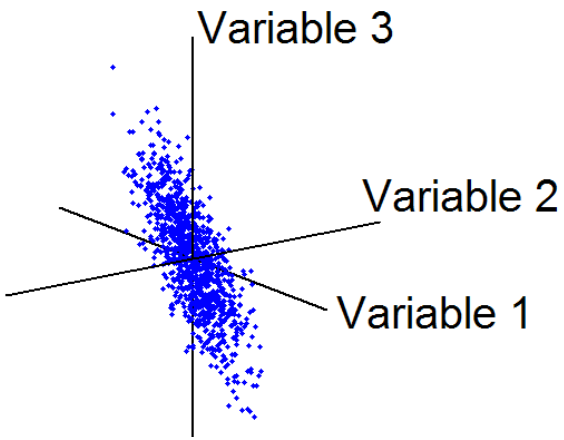
Unsupervised Learning

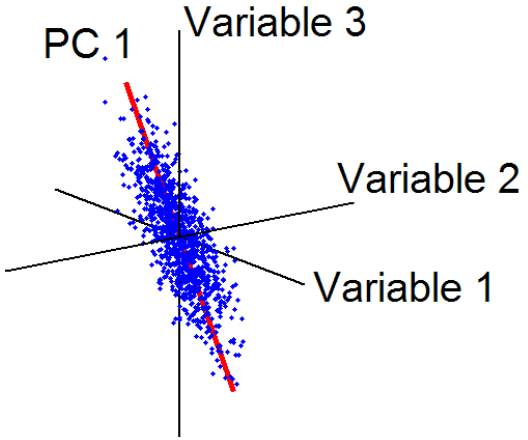
Discover an **internal representation** from **input data** only

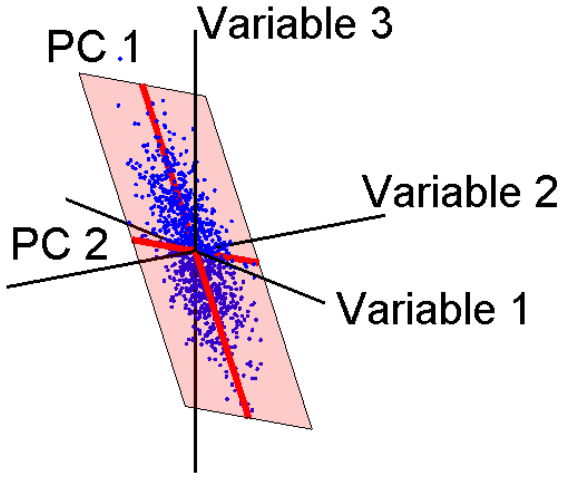


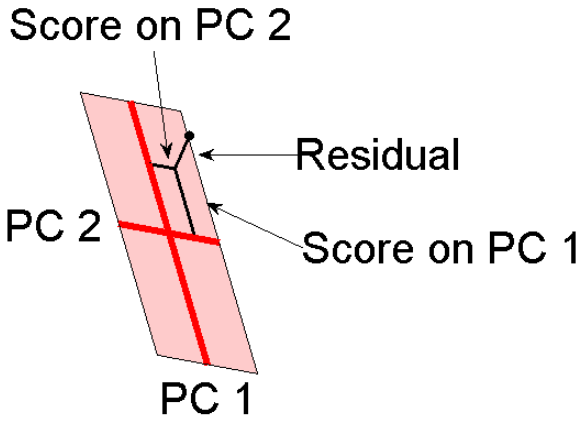
Anomaly Detection

Principal Components Analysis – what is it doing?









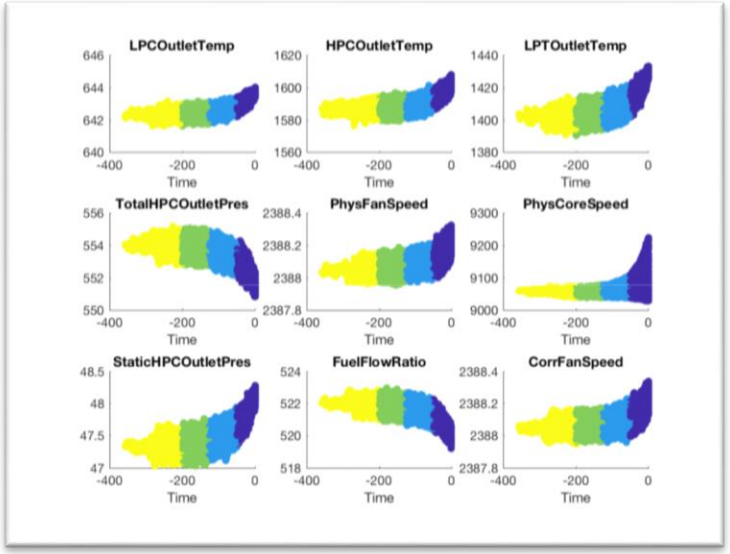
Different Types of Learning

Machine Learning

Type of Learning

Supervised Learning

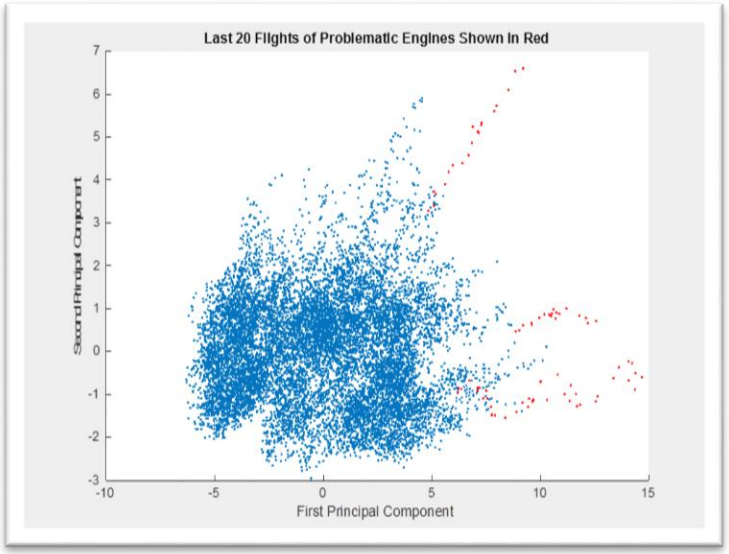
Develop **predictive model** based on both **input and output** data



Predicting Failures

Unsupervised Learning

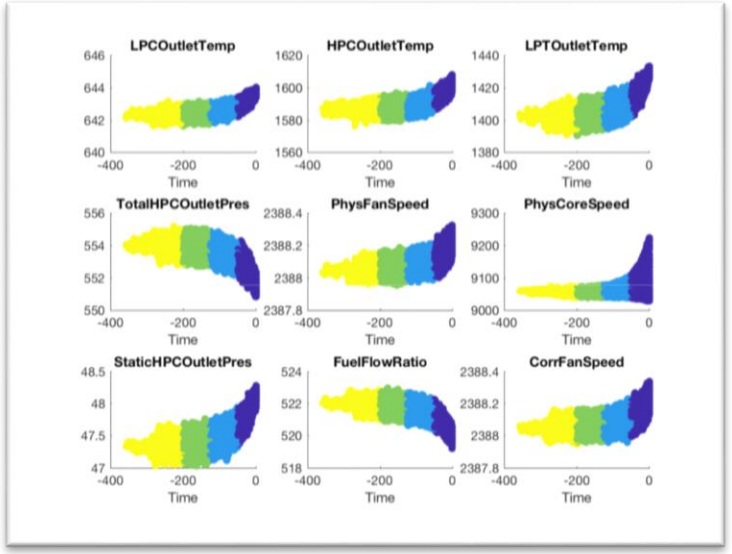
Discover an **internal representation** from **input data** only

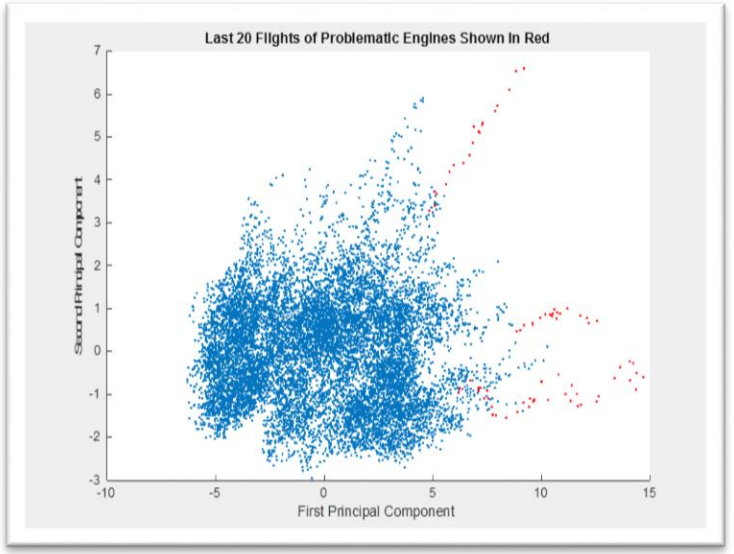


Anomaly Detection

Data-based Prognostics with Machine Learning - Takeaways

- Use machine learning to identify outliers and build predictive models
- Many choices for algorithms, apps make it easy to compare options
- Workflow-focused tools help you fine-tune the model to your particular data



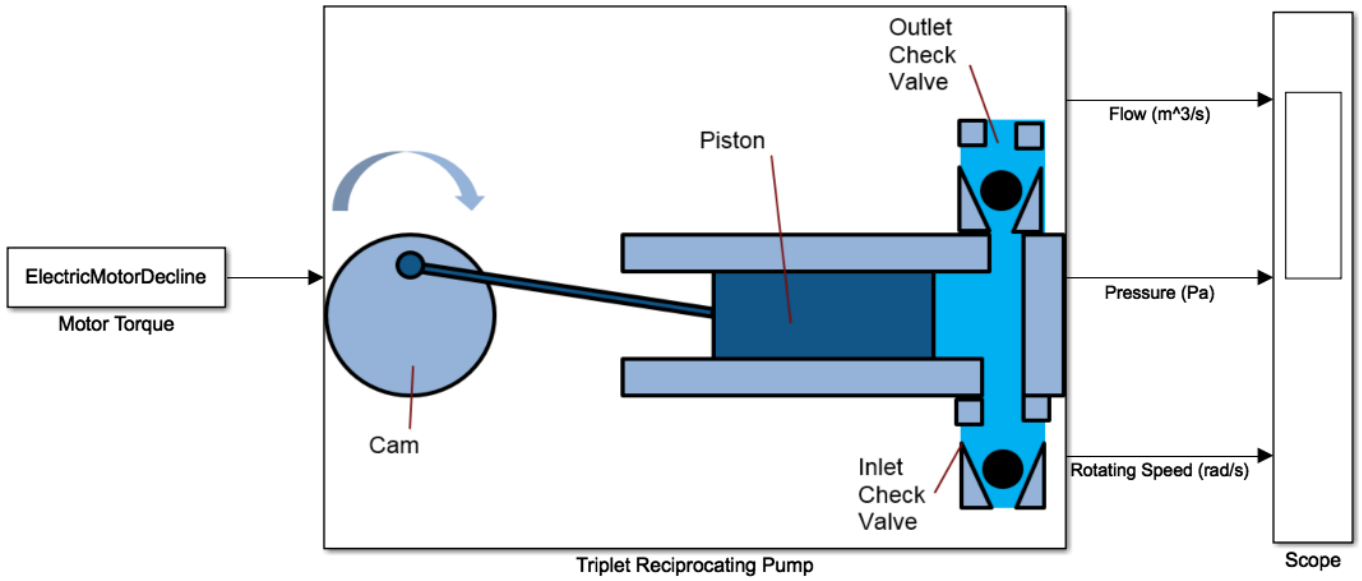


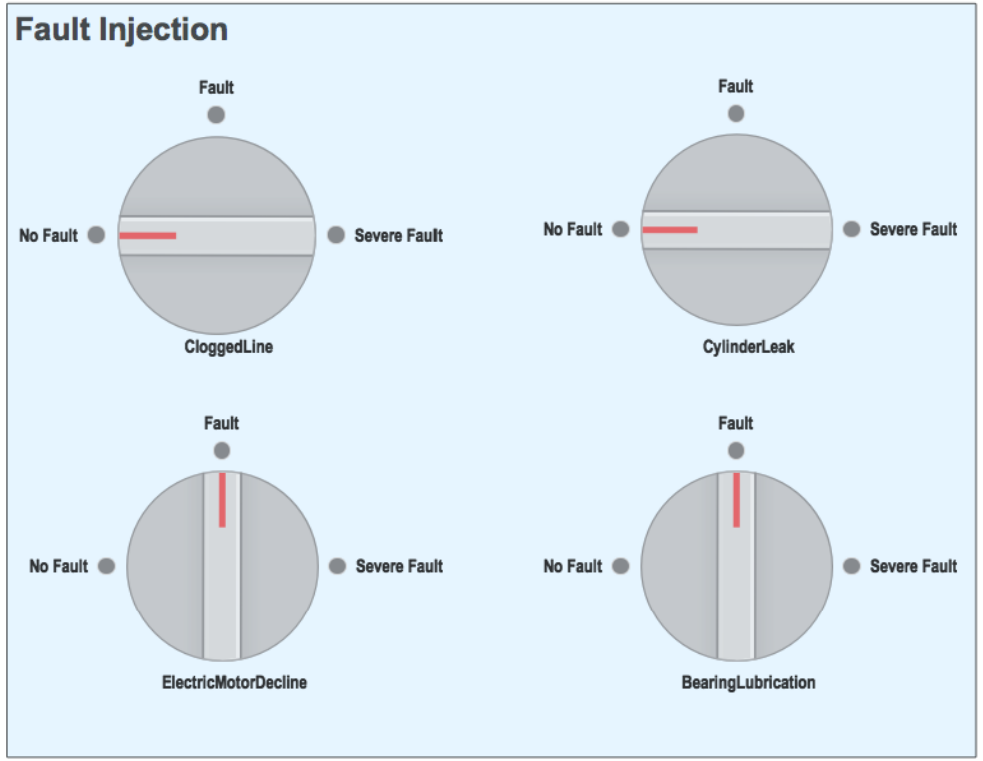
Examples

1. Data-based prognostics using machine learning

2. Fault injection and failure analysis using simulation

Example 2: Fault injection and failure analysis using simulation

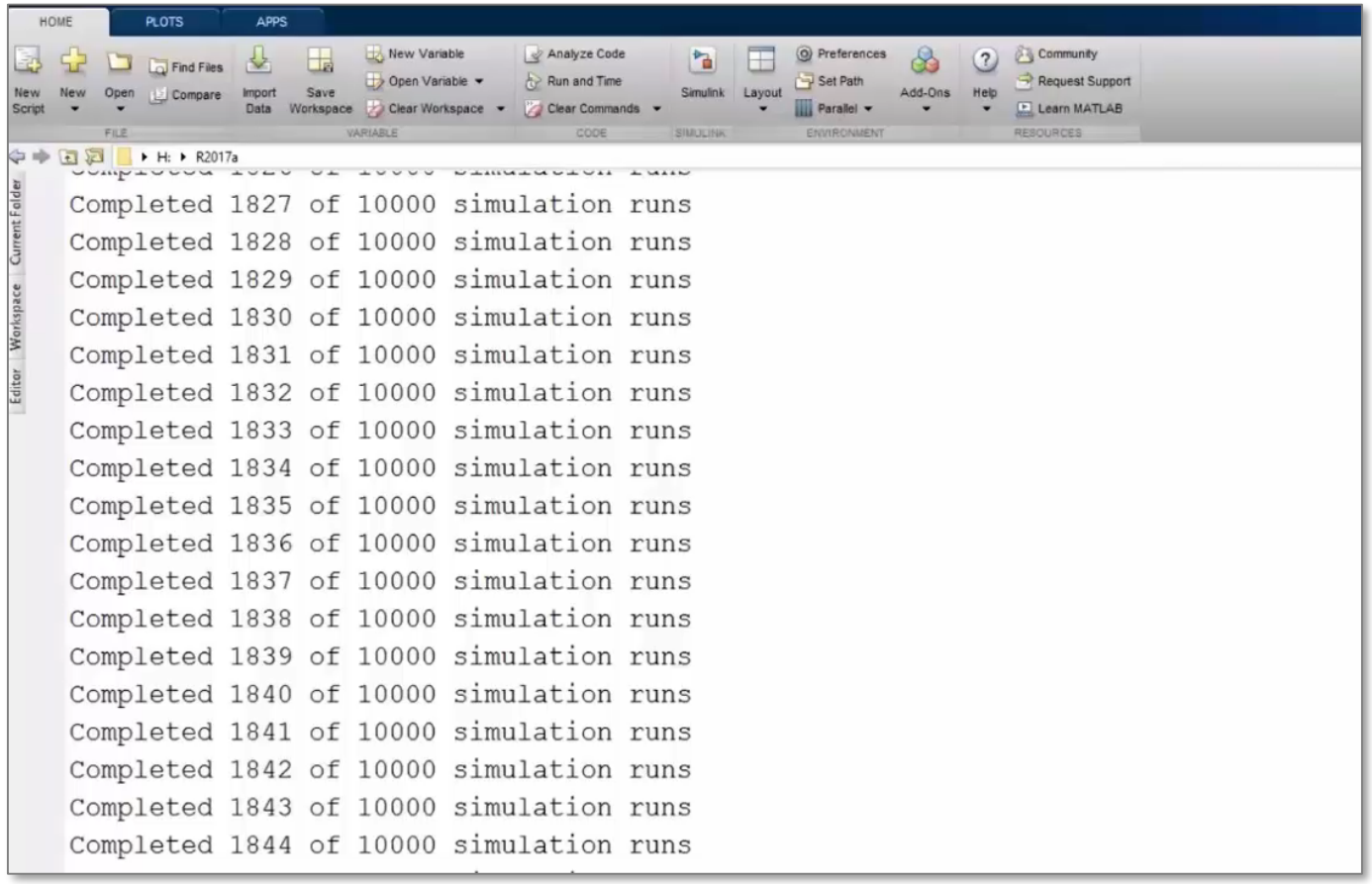




FrequencyAnalysisButton

Double-click here to turn the Mechanics Explorer ON

Run multiple parallel simulations from the `parsim` command

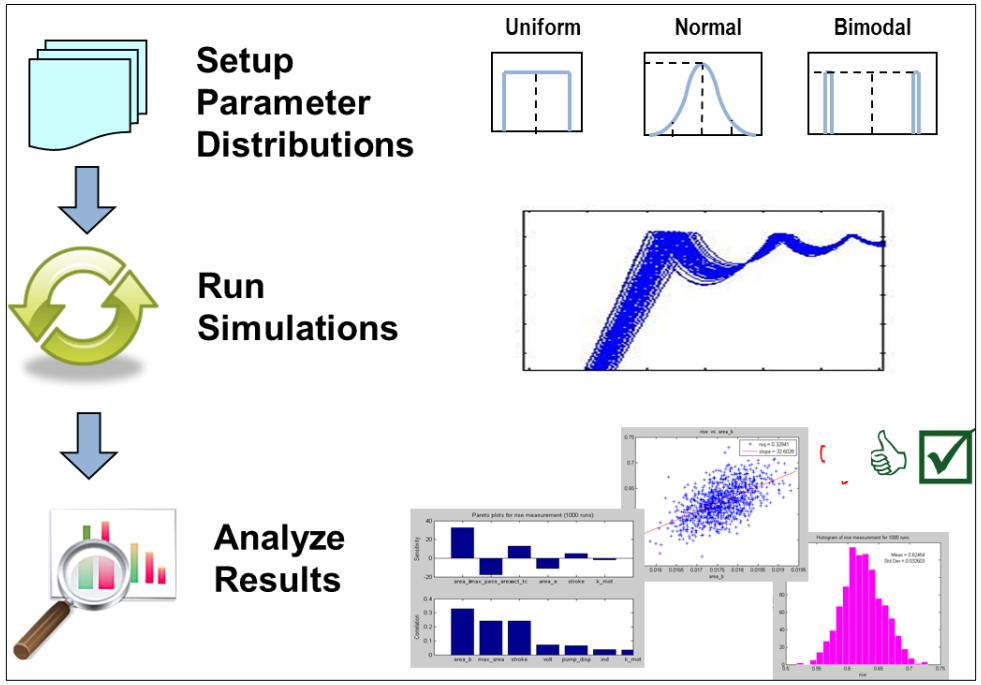


- Speed up simulations and simplify workflow
- Simplifies large simulation runs

Leverage Parallel Computing with Simulink

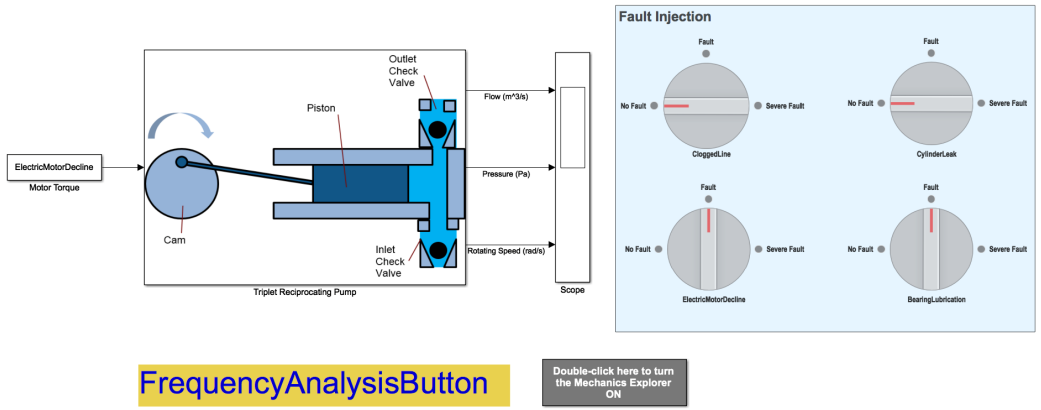
Reduce the total amount of time it takes to…

Run multiple independent simulations (E.g. Parameter sweeps, Monte Carlo Analysis)



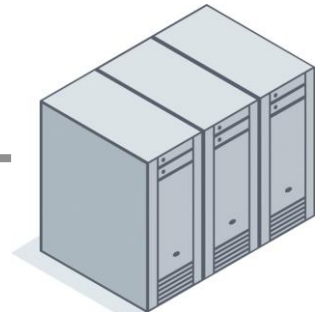
Fault injection and failure analysis – Takeaways

- Use simulation when measured data is not available
- Run what-if analyses to explore scenarios that are difficult to recreate
- Comparing field data to simulation data can help diagnose cause-of-failure



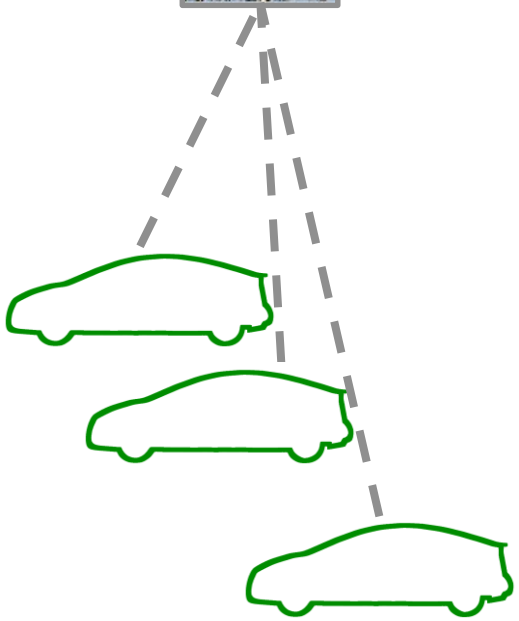
Fleet Data for Prognostics Development

Server



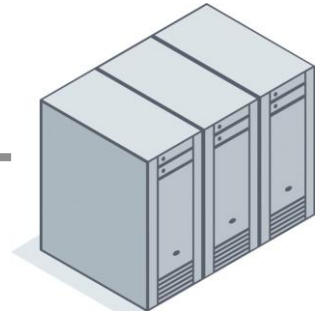
Data Storage

Vehicle Data



Deploying Prognostics Algorithms

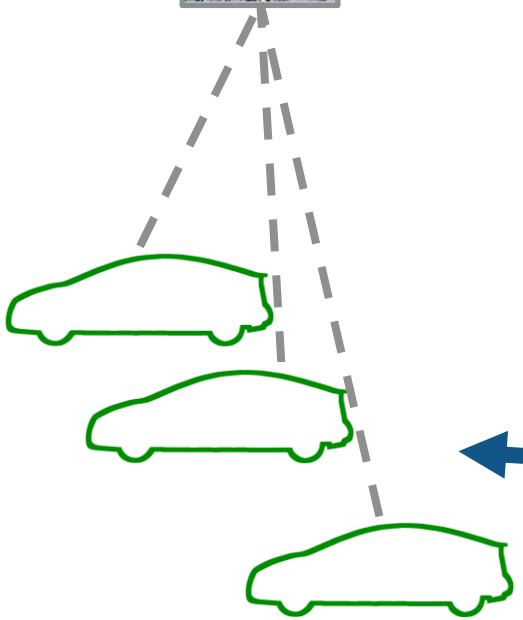
Server



Data Storage

Vehicle Data

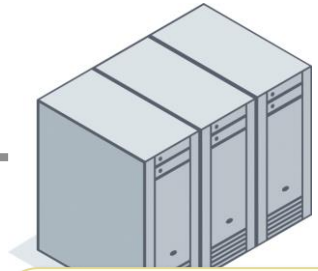
Server-side prognostics



Onboard prognostics

Considerations for System Architecture

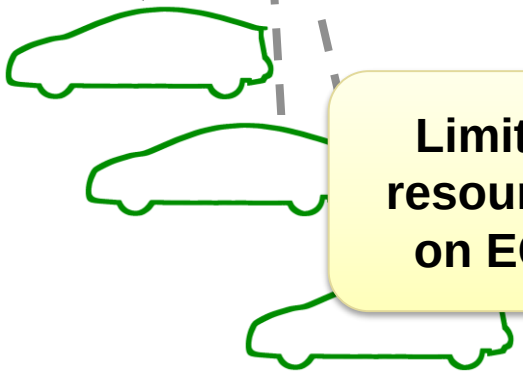
Server



Data Storage

“Infinite” resources on server

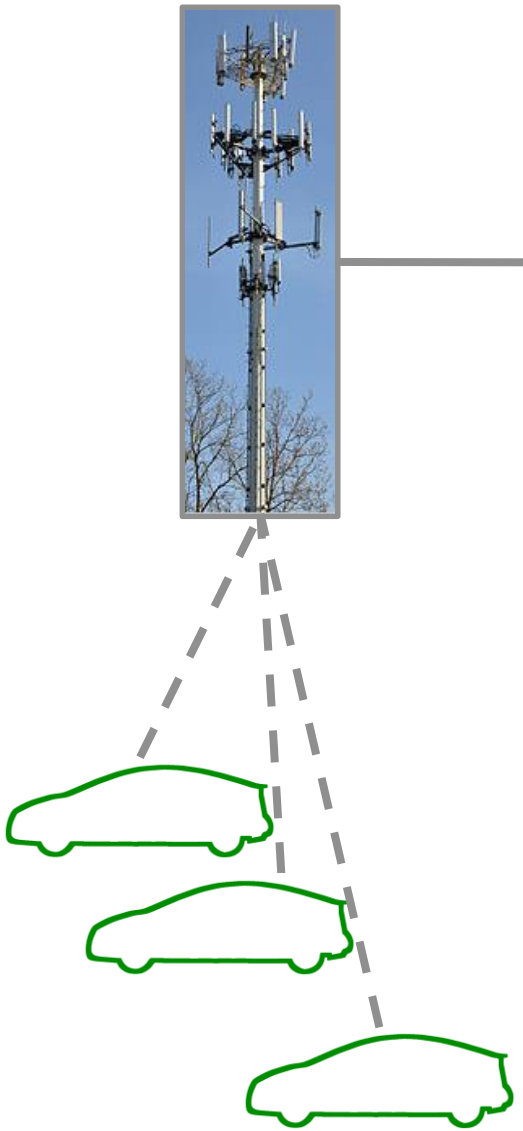
Connectivity is fragile



Limited resources on ECU

Some algorithms more viable than others

Considerations for System Architecture



Reasonable Compromise

- Hybrid approach – split between onboard and server-side
- Onboard: buffer, preprocess
- Server-side: classify, take action
- **Recommendation:** Think about modularity when designing

Server-side Prognostics

Server

RESTful

Java

.NET

Python

C/C++

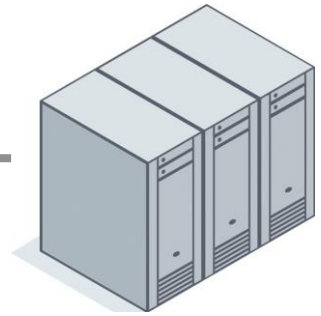
Data Storage

Vehicle Data

Package algorithms for MATLAB Production Server

Onboard Prognostics

Server

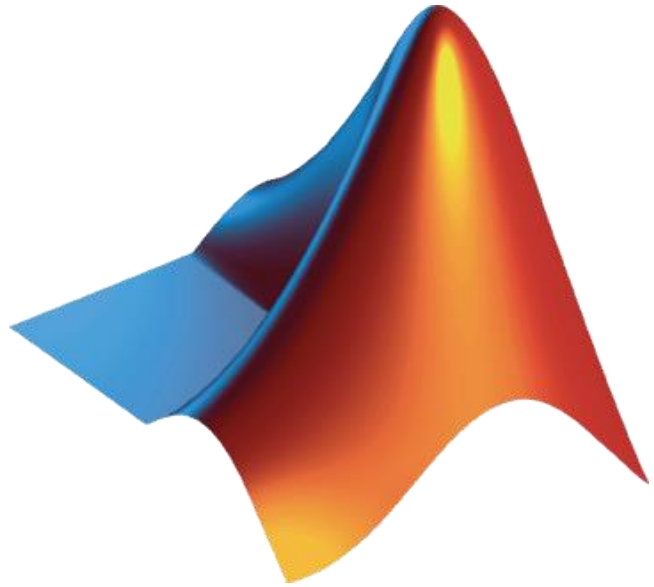


Data Storage

Vehicle Data

Generate standalone C/C++ code using Simulink Coder

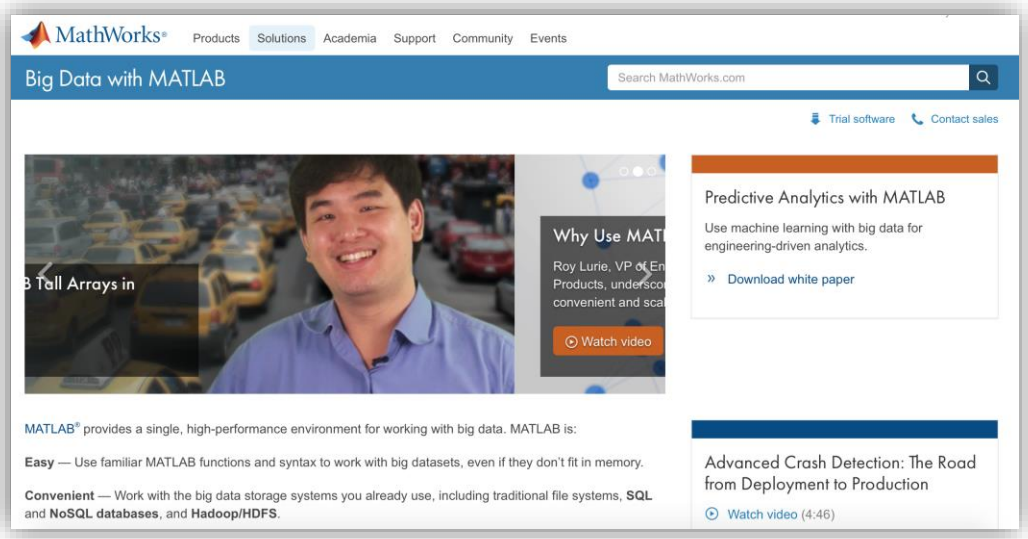
Key Takeaways



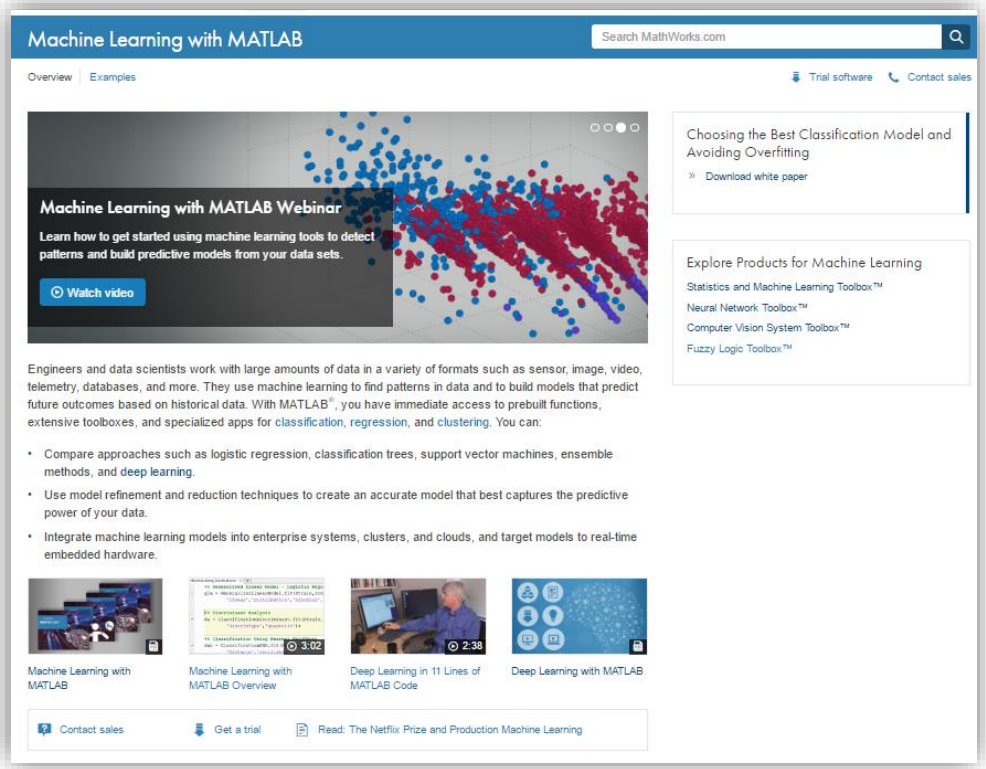
- No “one-size-fits-all” approach to prognostics.
- Prognostics system architecture is evolving.
- MATLAB and Simulink provide a platform for developing prognostics algorithms.

Learn More

mathworks.com/big-data



mathworks.com/machine-learning



Example: Model-based approach