

# 2020 MathWorks 中国汽车年会

## 工程数据的特征提取

朱黎明, 泛亚汽车技术中心

刘海伟, MathWorks 中国区高级应用工程师



# 主要内容

- 常见的工程应用场景与案例分析
  - 场景一与场景二及案例分享
  - 基于Benchmark数据分析研究
    - 背景
    - Benchmark数据分析现状
    - 数据特征提取识别
    - 结论和展望
  - 场景三与场景四
  - 总结

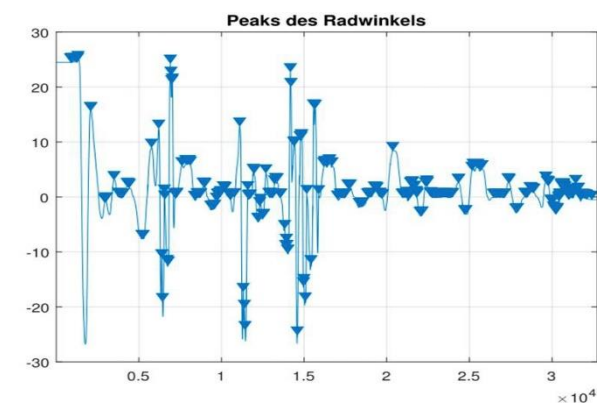
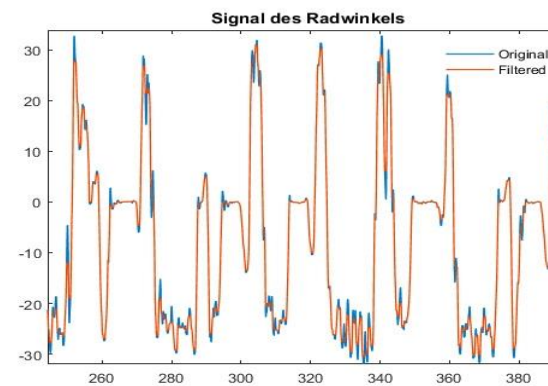
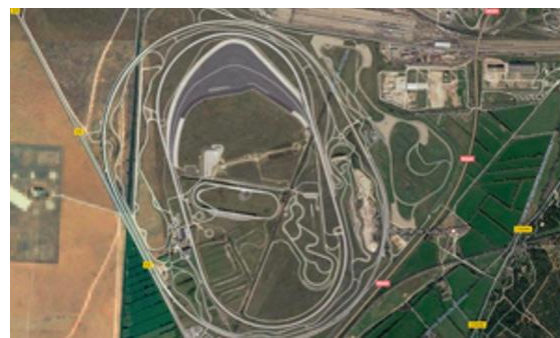
# 工程数据的特征提取

- 场景一：基于信号特征，参量已知（领域知识/原理）
- 场景二：基于状态量，参量已知（业务需求）
- 场景三：基于模型特征，参量已知（领域知识/原理）
- 场景四：参量未知/简化模型

# 场景一：基于信号特征，参量已知（领域知识/原理）

## BMW 使用机器学习检测过度转向

- 收集数据和特征提取



- 评估机器学习方法

	真阳率 (%)	真阴率 (%)	假阳率 (%)	假阴率 (%)
PCA降维后 的使用 K-最近邻	94.74	90.35	5.26	9.65
支持向量机	98.92	73.07	1.08	26.93
二次判别分析	98.83	82.73	1.17	17.27
决策树	98.16	95.86	1.84	4.14

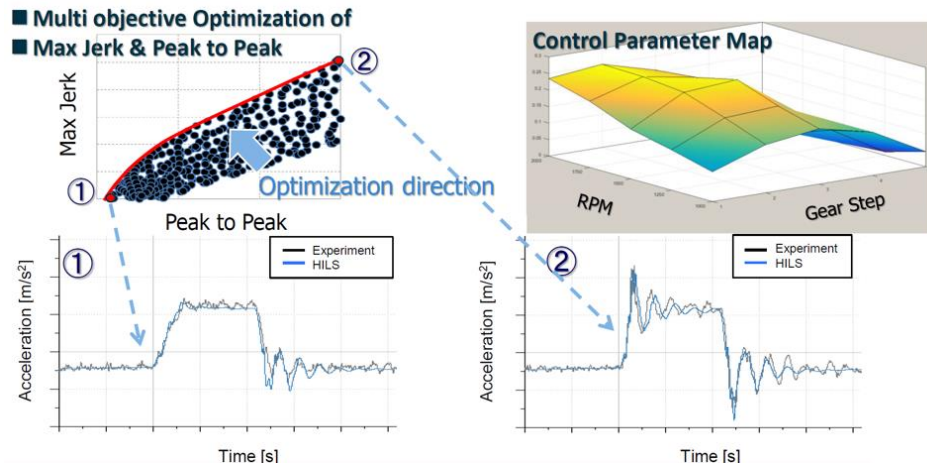
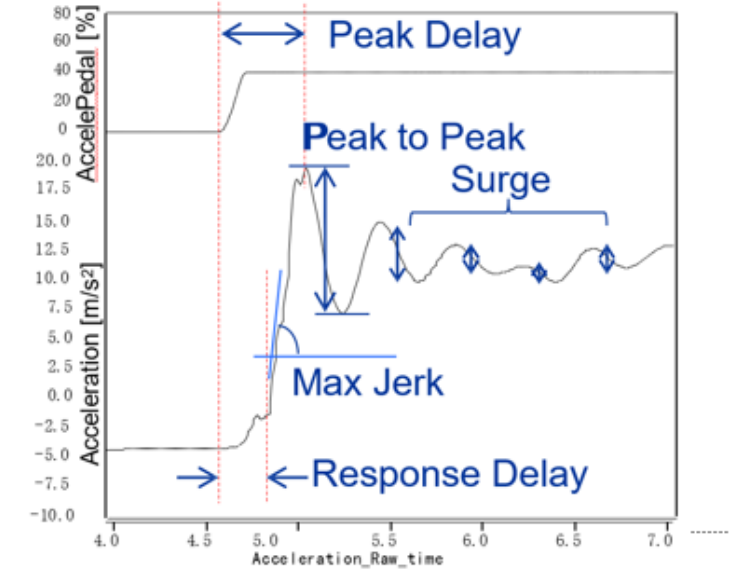
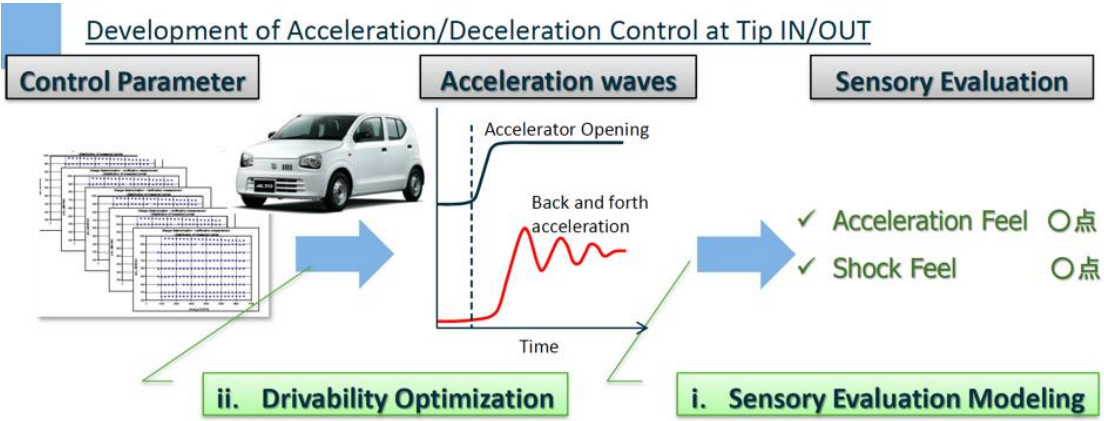
- 生成车载装置测试所需的代码

- 使用 MATLAB Coder™ 从模型生成代码，并为安装在宝马 5 系的目标 ECU 编译代码
- 在 ECU 上实时运行的分类器表现非常出色，准确率约为 95%

[Link to article](#)

# 场景一：基于信号特征，参量已知（领域知识/原理）

Suzuki Motor Corporation “使用机器学习和优化技术提升驾驶操控性”



$$\min_{RL^*} J = 0.5(t_{resp}^*) \text{ with constraints}$$

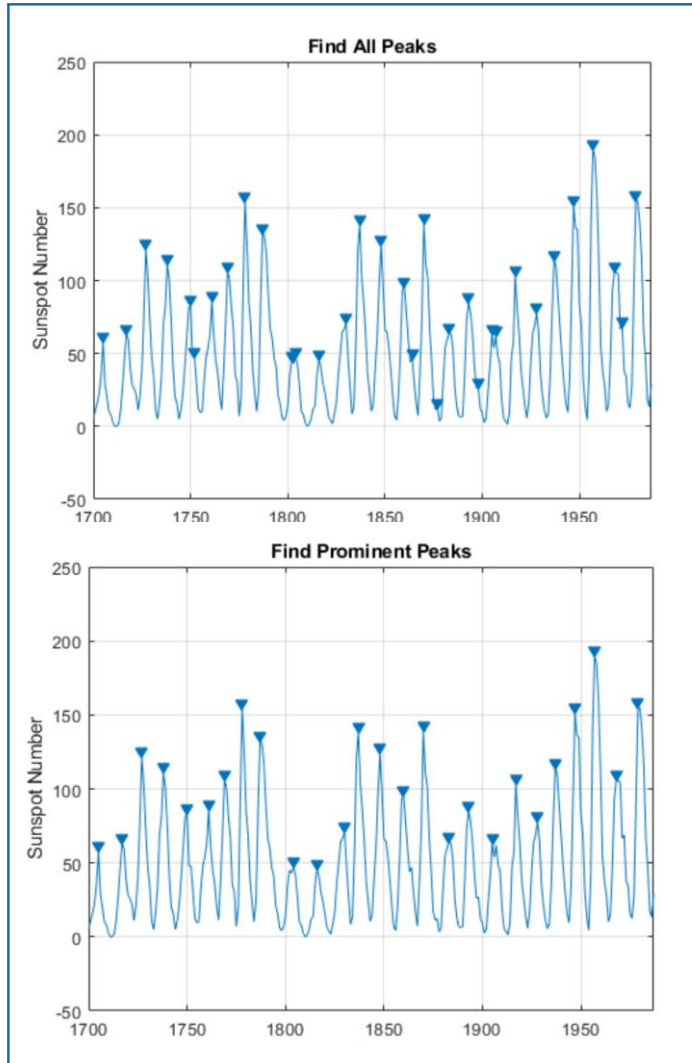
$$\min_{RL^*} J = 0.5(jerk_{max}^*) \text{ with constraints}$$

$$\min_{RL^*} J = 0.5(VDV^*) \text{ with constraints}$$

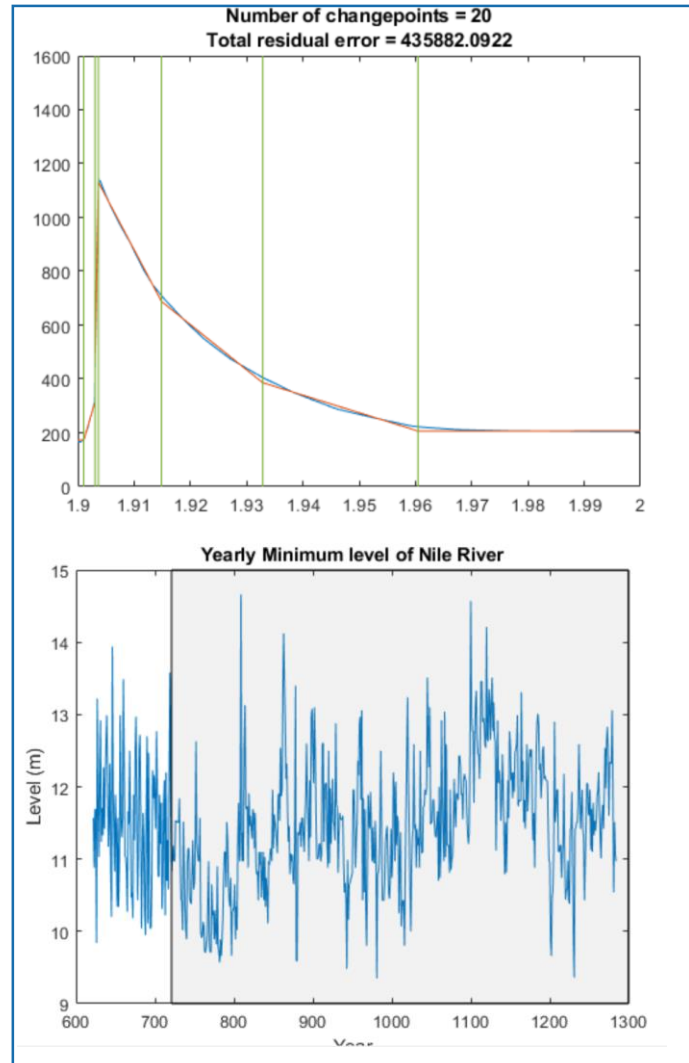
$$\min_{RL^*} J = 0.5(t_{resp}^* + jerk_{max}^*) + 0.5(VDV^*) + constraints$$

# 测量和特征计算

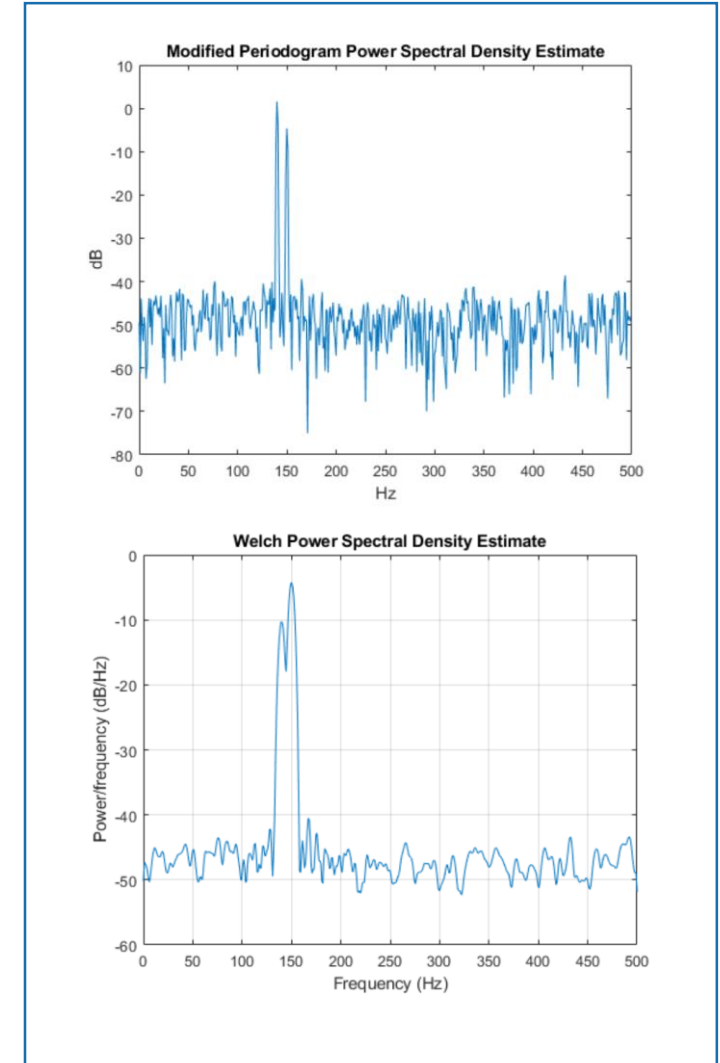
## 峰值分析



## 查找突变点

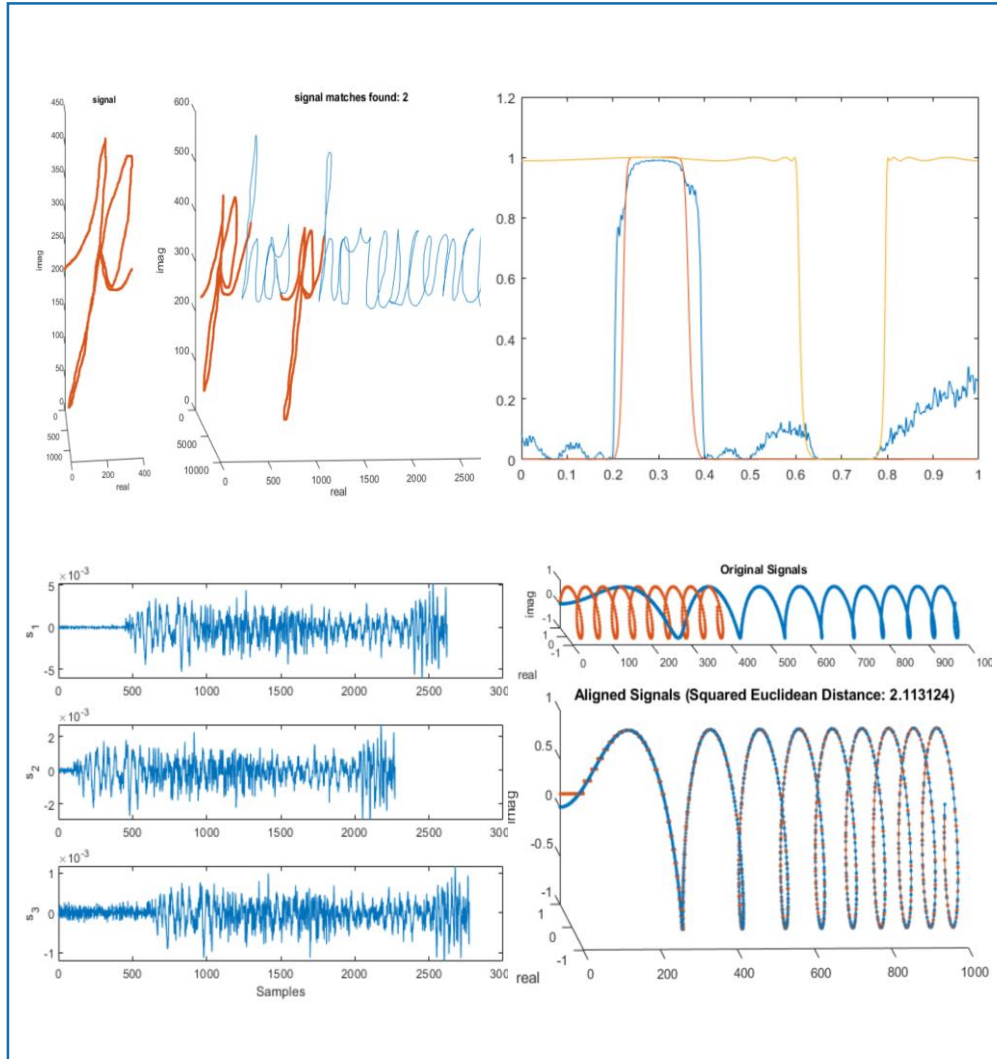


## 频谱估计

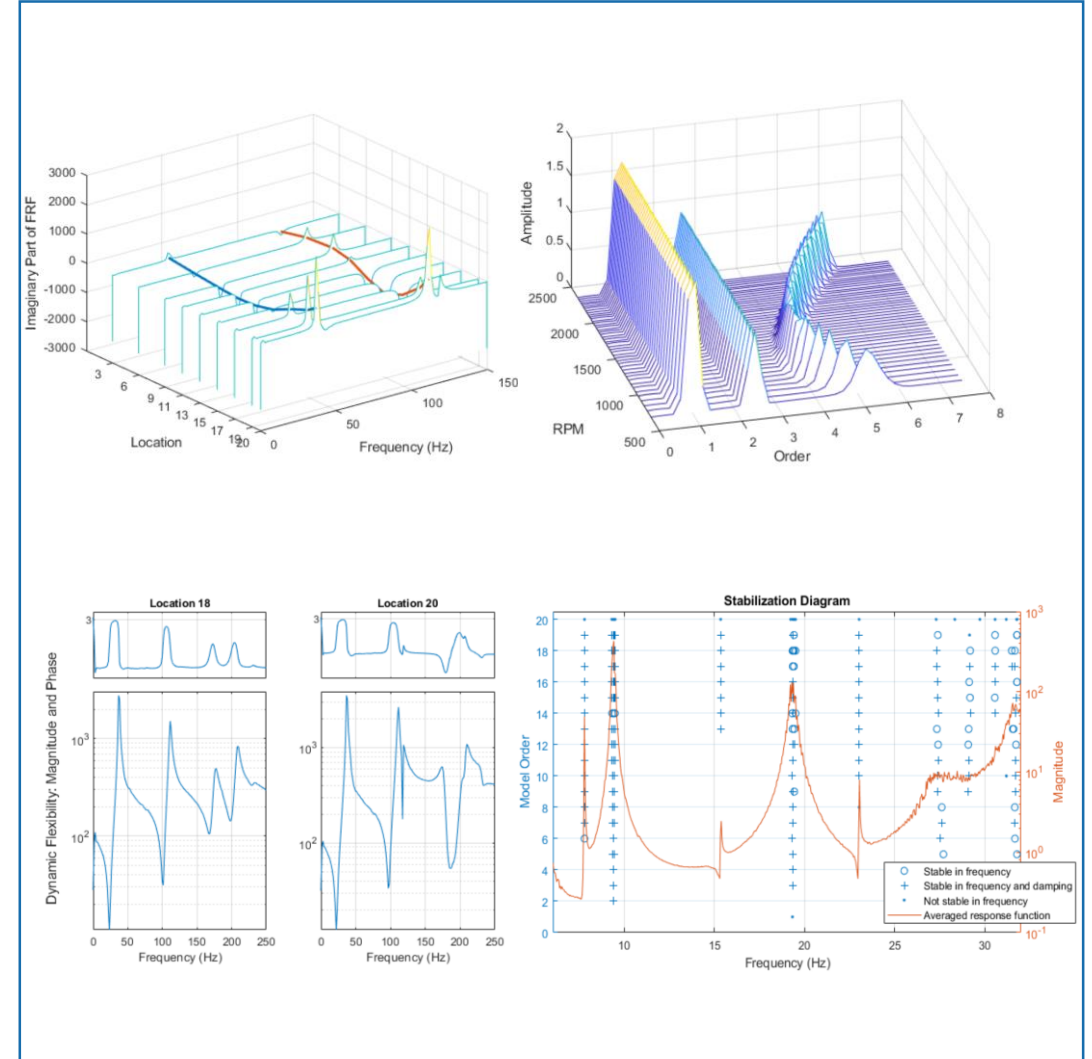


# 测量和特征计算

## 信号匹配与相关分析



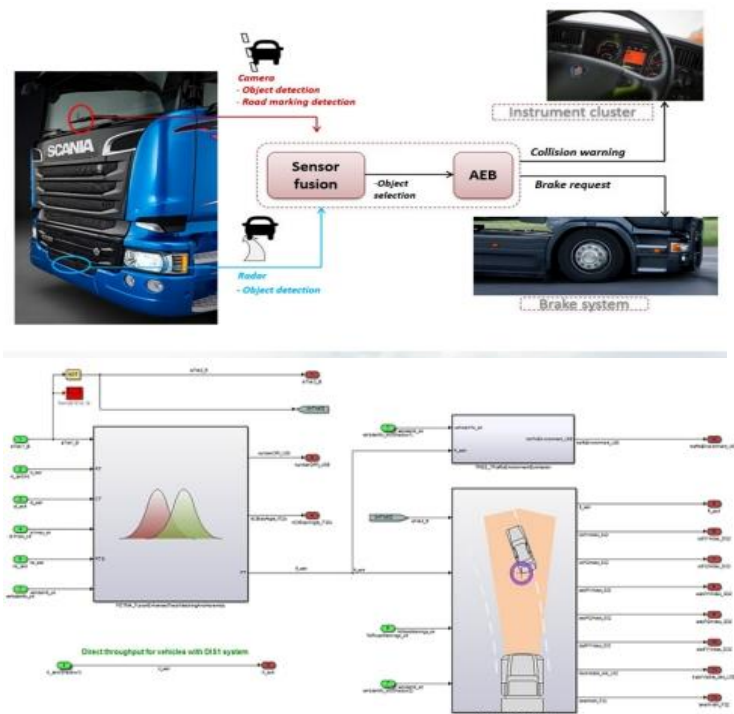
## 振动分析



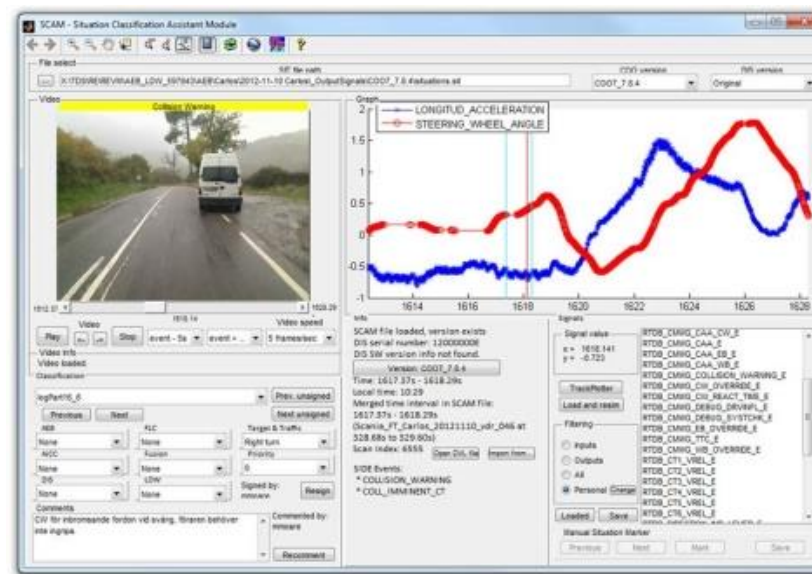
# 场景二：基于状态量，参量已知（业务需求）

## Scania 开发AEB(Advanced Emergency Braking)系统

- 开发传感器融合算法



- 验证和修正设计



- 系统实现并优化代码性能

- 代码生成并部署

[link to the story](#)



# 场景二：基于状态量，参量已知（业务需求）

## 自动驾驶/车队数据分析

- CAN数据访问
  - 数据解析: blf, mdf, cdf
  - 数据存储: parquet, csv, mat
  - 数据量大: TB/~100GB
- 整车实验数据分析: 不同工况的台架/道路试验
- 筛选特征工况数据（离合器动作/发动机介入/AEB触发/车辆变道）以及与有些特定功能有关的数据（AEB,变速器）
  - 矩阵索引
  - 高效的计算函数
    - cellfun
    - arrayfun
    - rowfun
    - structfun
  - 大数据的支持
    - Map Reduce
    - Tall Array
    - API for Spark

```
%小数据的矩阵索引
dataSelected = data(abs(data.BrakePedalPosition)>criteria)
%小数据的矩阵计算
[AEBResult] = arrayfun(@(x) SetParameterforModelandSim(x,...
    MDL), dataSelected);
%大数据的矩阵数据提取与并行仿真 在map中把小数据的矩阵索引用上
outdsAEB = mapreduce(ds, @mapAEBfun, @reduceAEBfun);
```

# 基于 Benchmark 数据分析研究

朱黎明, PATAC

# 演讲内容

- 背景
- Benchmark 数据分析现状
- 数据特征提取识别
- 结论和展望

# 背景

- Benchmark对标过程中需要多角度、全方位设计试验和分析数据，特别是针对复杂的多能源动力系统需要更加科学的方法来分析数据；
- 重点是如何快速识别潜在的影响因素和逻辑，区分不同控制逻辑的合理性和必要性，发现潜在的改进的机会和方向；
- 如何迅速高效的获取和识别数据成为了最关键的问题；

# Benchmark 数据分析现状

## ➤ 当前方法:

基于目前的可视化工具，同时显示多个传感器数据并做比例尺归一，手动判断对应的工况并进行查找;

## ➤ 存在问题:

- a) 数据的获取和分析高度依赖个人的经验和判断，存在偏差;
- b) 对于多能源动力系统，结构和控制都更加复杂，无法统一化和精确化;
- c) 手动识别效率很低，工作量大，无法高效对大规模数据进行特征识别和数据处理;

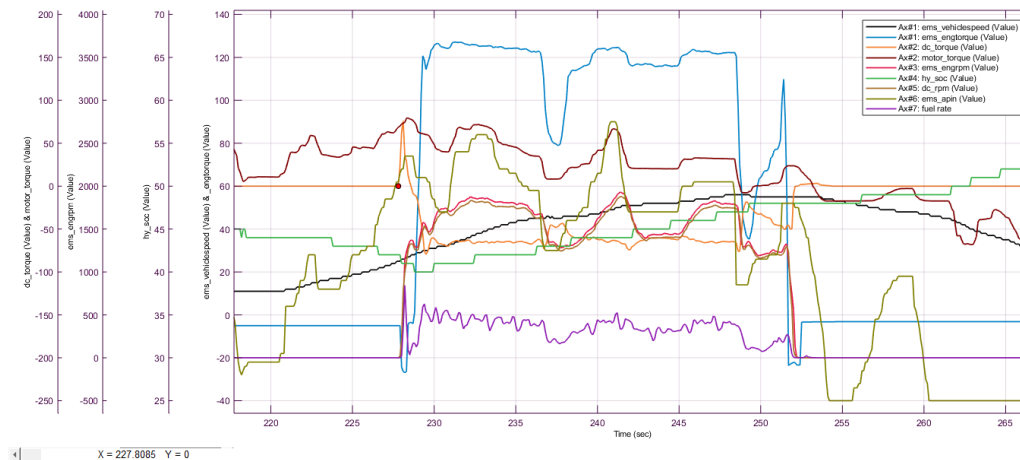


图1 试验数据

# 数据特征识别提取

- -通过单一传感器的阈值判断容易漏点或错误点
- -判断逻辑太多，每个传感器都可以作为一个判断标准
- -多个逻辑通过矩阵 $\cap \cup$ ，计算效率高

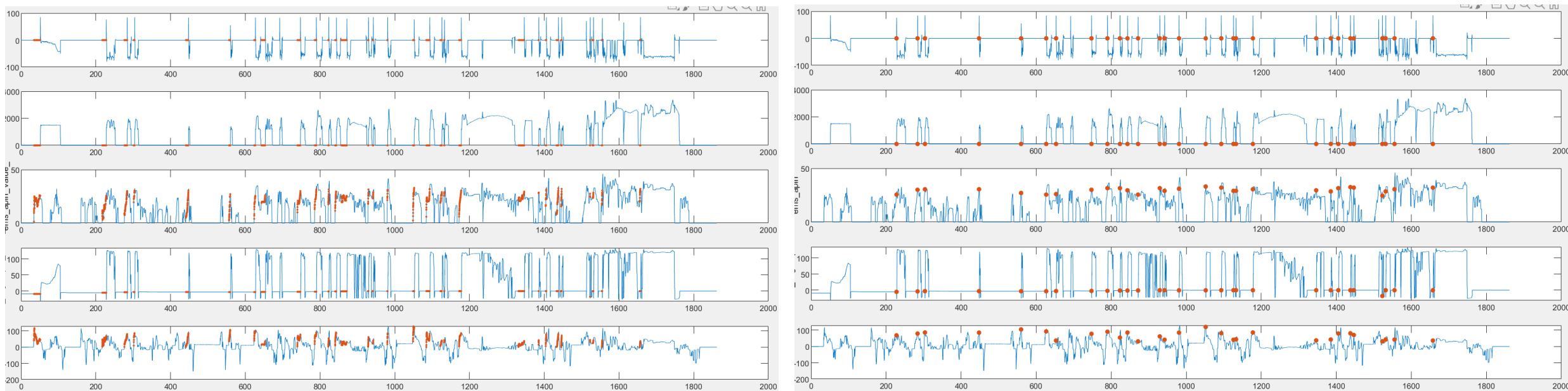


图2 试验数据

# 特征识别：模式匹配

- **优势：** 数据处理可以使用现成的算法库，简单高效；
- **劣势：** 准确性存疑，存在漏报或误报的情况，不够灵活；

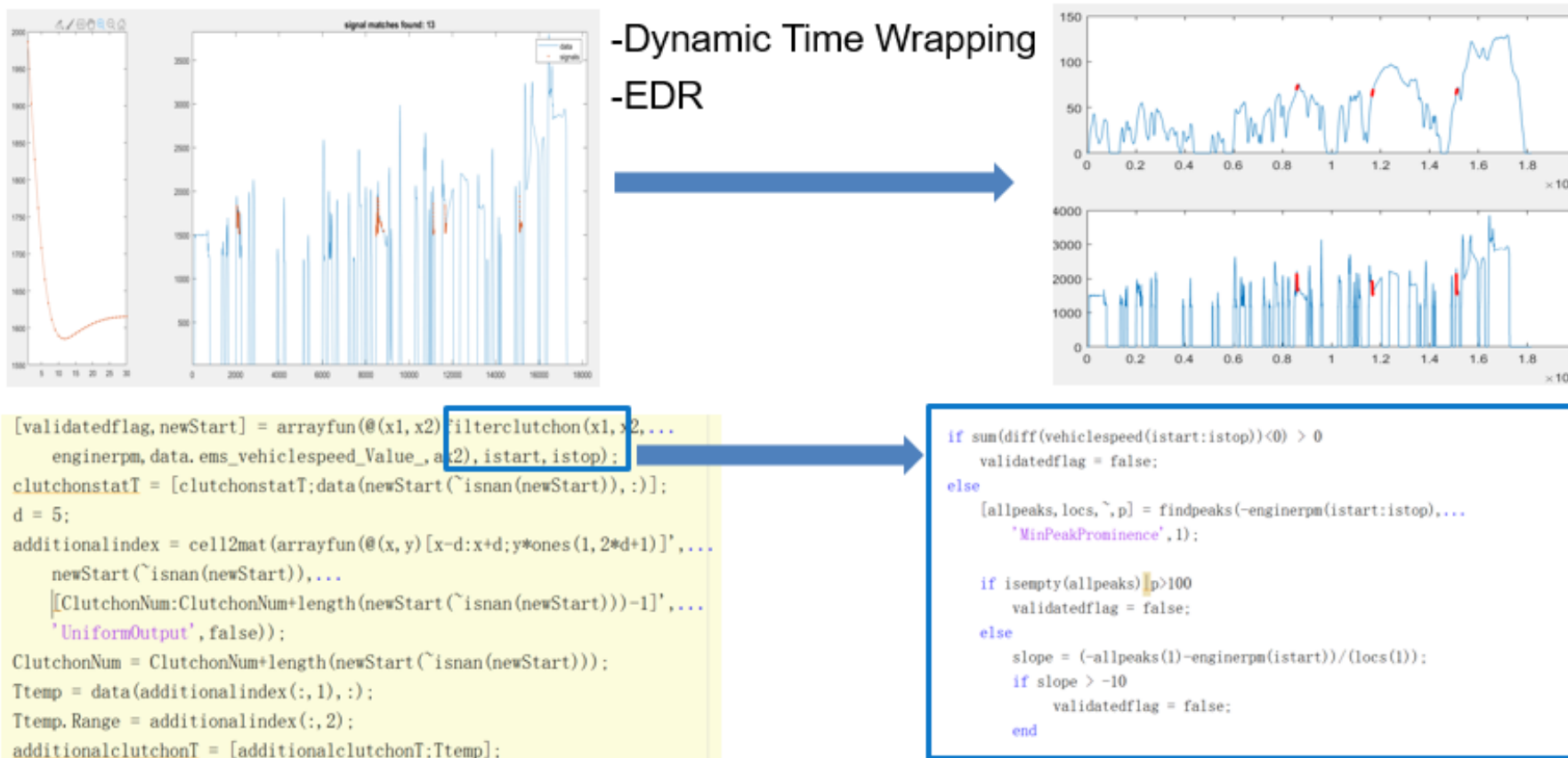


图3 离合器结合过程

# 特征识别：统计和数值计算

- 统计+Savitzky-Golay/Gaussian平滑：计算稳态值；
- 微分：计算变量加速度；
- 领域+时域：计算反推目标点；
- 领域+相关分析：防止误判，提高准确度；

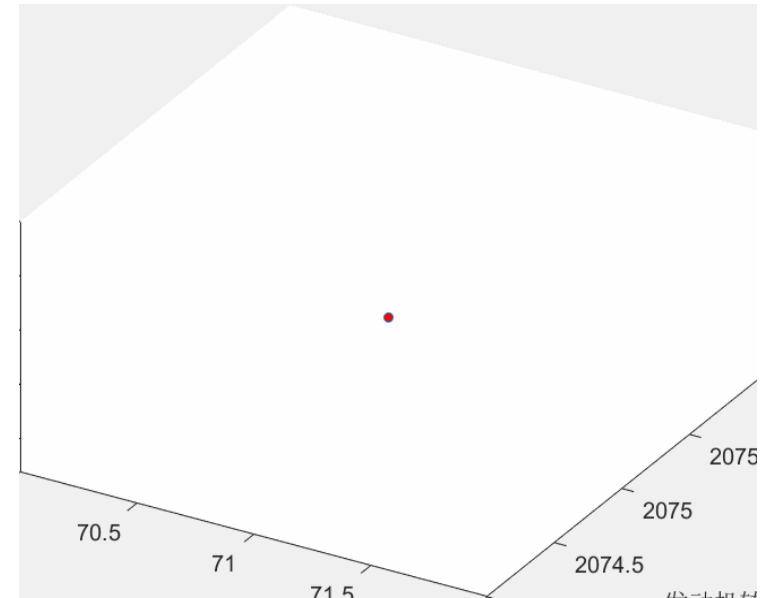
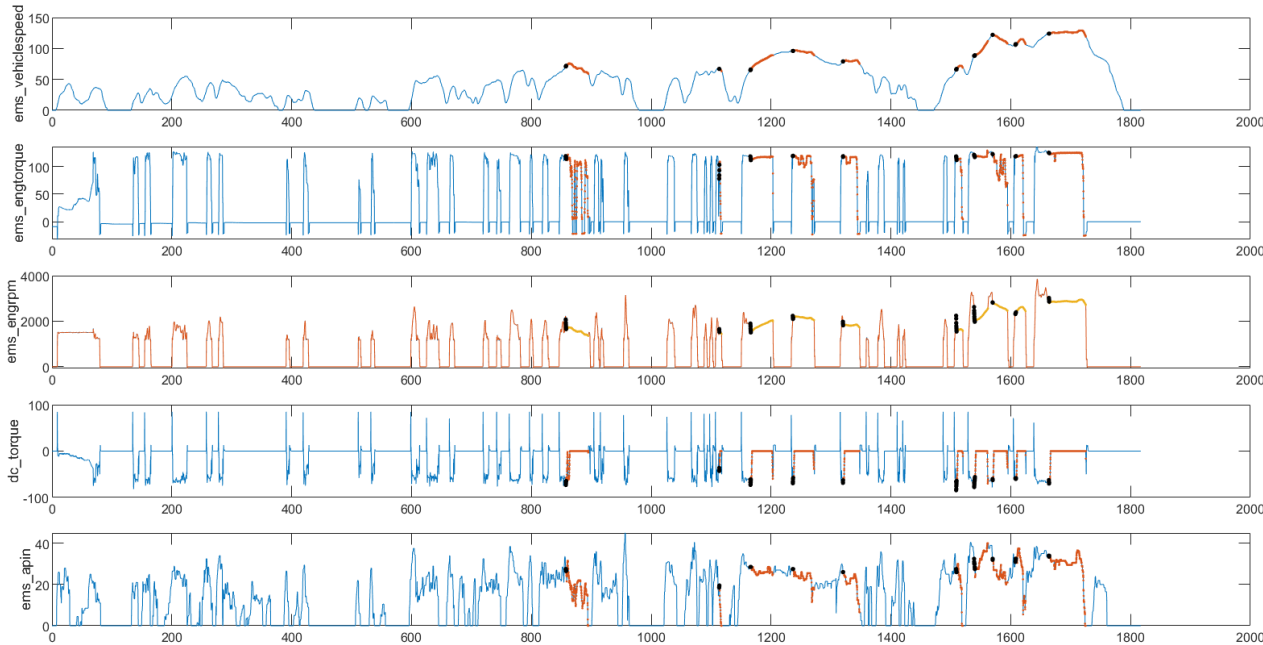


图4 离合器结合过程



# 结论和展望

➤ 目前的输出处理结果表明：

- 识别准确率高，特别是针对复杂工况能够高效识别潜在的影响因素，大大简化工作量；
- 特定工况，数据识别出来的准确率和一致性存在差异，特别是对于动态变化过程需要进一步优化（传感器，算法...）；

➤ 潜在影响因素和控制逻辑寻优，推广应用到更大的范围；

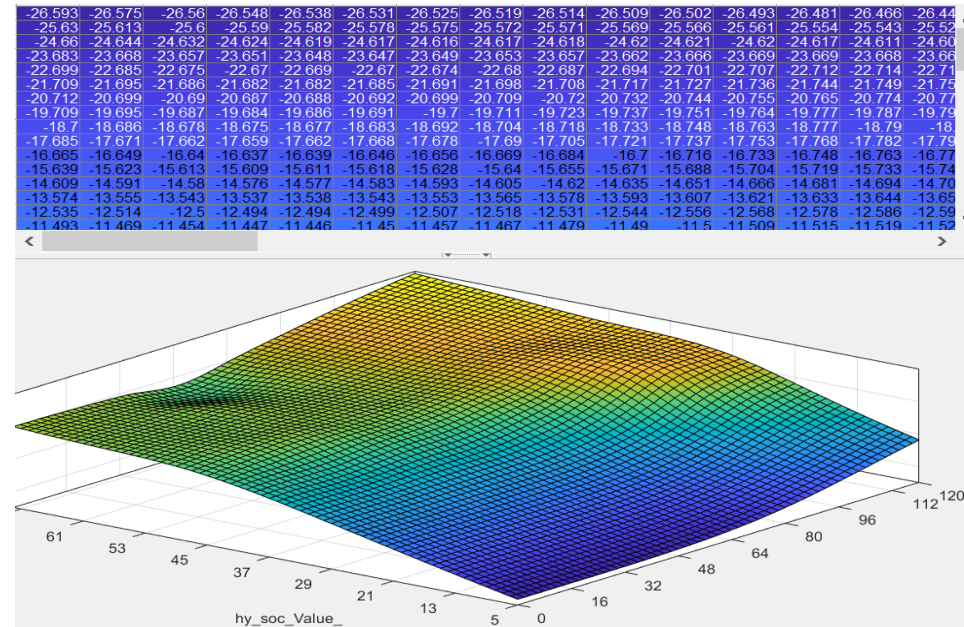
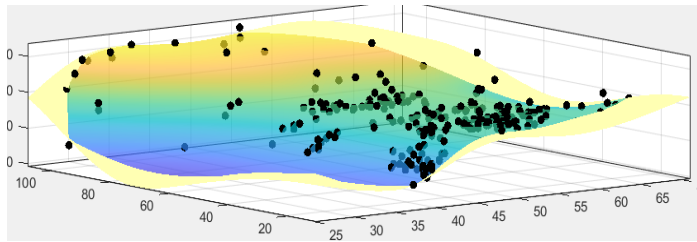
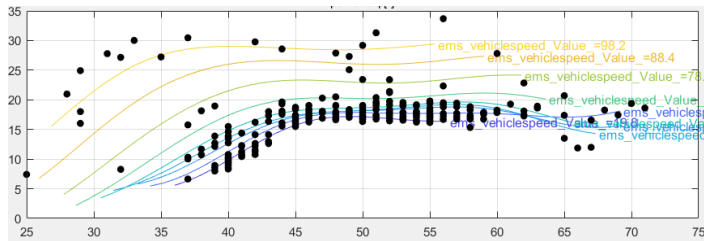


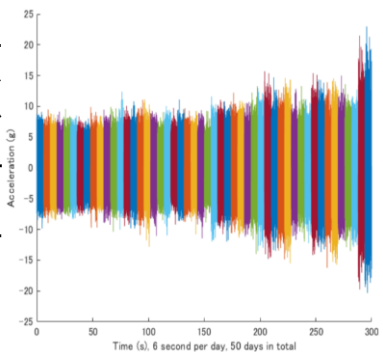
图6 发动机启动控制

# 场景三：基于模型特征，参量已知（领域知识/原理）

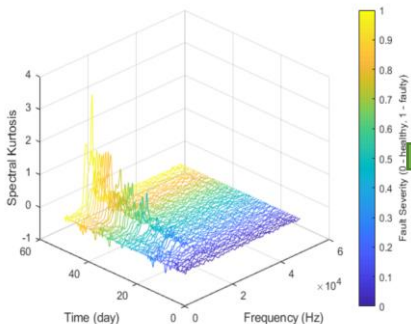
## 健康管理 健康指标设计

### 基于信号特征

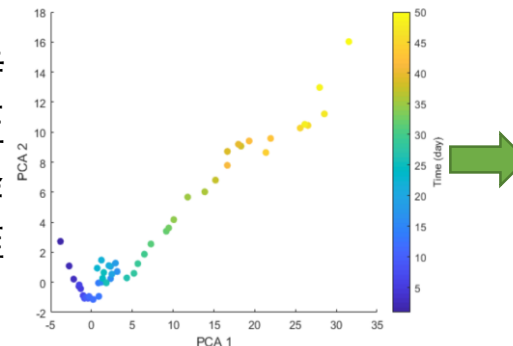
原始信号



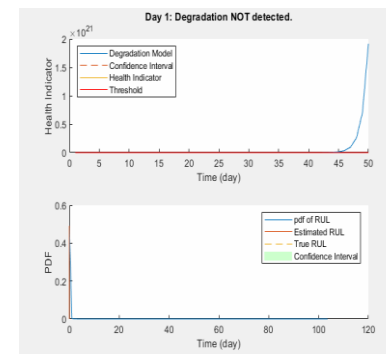
计算信号特征



特征降维



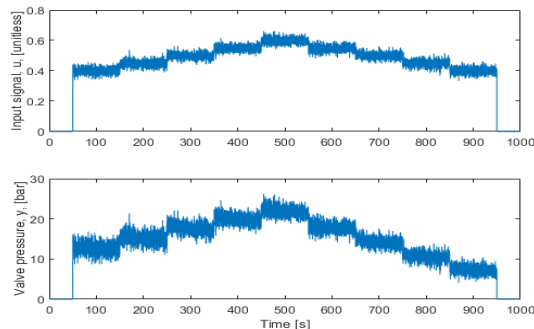
创建寿命模型



### 基于模型特征

#### — 无级变速箱 (CVT)

- 物理机理系统:  $y(t) = k(t)u(t) + b(t)$  其中  $\frac{-b(t)}{k(t)} < u \leq 1$

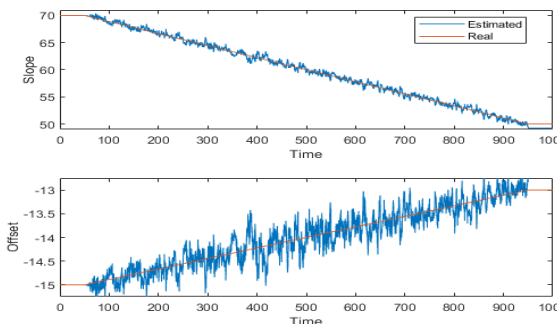


在线参数估计:

$$y(t) = k(t)u(t) + b(t) + e(t)$$

$$= [k(t) \ b(t)][u(t) \ 1]T + e(t)$$

$$= H(t)x(t) + e(t)$$



- [ssest](#) — Estimate a state-space model
- [arx](#), [armax](#), [ar](#) — Estimate an autoregressive or moving-average (AR or ARMA) model
- [nlrx](#) — Model nonlinear behavior using dynamic nonlinearity estimators
- [recursiveARX](#) and [recursiveAR](#) --- recursive estimators
- [unscentedKalmanFilter](#), [extendedKalmanFilter](#), and [particleFilter](#) state estimators

# 场景四：参量未知/简化模型

## Dimensionality Reduction and Feature Extraction

### Feature Selection

<a href="#">fscchi2</a>	Univariate feature ranking for classification using chi-square tests
<a href="#">fscmrmmr</a>	Rank features for classification using minimum redundancy maximum relevance (MRMR) algorithm
<a href="#">fscnca</a>	Feature selection using neighborhood component analysis for classification
<a href="#">fsrfstest</a>	Univariate feature ranking for regression using $F$ -tests
<a href="#">fsrnca</a>	Feature selection using neighborhood component analysis for regression
<a href="#">fsulaplacian</a>	Rank features for unsupervised learning using Laplacian scores
<a href="#">plotPartialDependence</a>	Create partial dependence plot (PDP) and individual conditional expectation (ICE) plots
<a href="#">oobPermutedPredictorImportance</a>	Predictor importance estimates by permutation of out-of-bag predictor observations for random forest of classification trees
<a href="#">oobPermutedPredictorImportance</a>	Predictor importance estimates by permutation of out-of-bag predictor observations for random forest of regression trees
<a href="#">predictorImportance</a>	Estimates of predictor importance for classification tree
<a href="#">predictorImportance</a>	Estimates of predictor importance for classification ensemble of decision trees
<a href="#">predictorImportance</a>	Estimates of predictor importance for regression tree
<a href="#">predictorImportance</a>	Estimates of predictor importance for regression ensemble
<a href="#">relieff</a>	Rank importance of predictors using ReliefF or RReliefF algorithm
<a href="#">sequentialfs</a>	Sequential feature selection using custom criterion
<a href="#">stepwiselm</a>	Perform stepwise regression
<a href="#">stepwiseglm</a>	Create generalized linear regression model by stepwise regression

### Feature Extraction

<a href="#">rica</a>	Feature extraction by using reconstruction ICA
<a href="#">sparsefilt</a>	Feature extraction by using sparse filtering
<a href="#">transform</a>	Transform predictors into extracted features

- **PCA and Canonical Correlation**
- **Factor Analysis**
- **Multidimensional Scaling**
- ...

[link to help](#)

# 总结

- 不同场景下可以使用对应的特征提取目标数据
  - 信号处理函数
  - 统计和数值计算函数
  - 矩阵索引与数组函数
  - 参数辨识计算模型特征
- 数据量大的情况下使用大数据框架的编程方法
  - Map Reduce
  - Tall
  - Spark API

谢谢