MATLAB Examples MathWorks®

Examples of Data Analytics for Predictive Maintenance

By Akira Agata

Explore:
Demo Files for Predictive Maintenance

This example also uses:
Statistics and Machine Learning Toolbox

View in: File Exchange

Contents

Summary

Rare events prediction in complex technical systems has been very interesting and critical issue for many industrial and commercial fields due to huge increase of sensors and rapid growth of Internet of Things (IoT). To detect anomalies and foresee machine failure during normal operation, various types of Predictive Maintenance (PdM) techniques have been studied, as shown in the following table.

<table>
<tr><th>Anomaly detection methods</th><th>Data type</th><th>Learning method</th><th>Assumed distribution</th></tr>
<tr><td>Hotelling's T^2 method</td><td rowspan="7">Multi-dimensional</td><td>Unsupervised</td><td rowspan="3">Unimodal</td></tr>
<tr><td>Mahalanobis Taguchi (MT) method</td><td rowspan="3">Supervised</td></tr>
<tr><td>Naive Bayes classifier</td></tr>
<tr><td>k-nearest neighbor (KNN)</td><td>Nonlinear</td></tr>
<tr><td>Gaussian mixture model</td><td rowspan="6">Unsupervised</td><td>Multimodal</td></tr>
<tr><td>One-class SVM</td><td>Nonlinear</td></tr>
<tr><td>vMF* distribution-based method</td><td>Unimodal</td></tr>
<tr><td>Gaussian Process Regression (GPR)</td><td>Multi-dimensional / Time-series</td><td>Nonlinear</td></tr>
<tr><td>Subspace Method</td><td rowspan="2">Time-series</td><td rowspan="2">-</td></tr>
<tr><td>Control chart (Shewhart charts)</td></tr>
</table>

* vMF: von Mises-Fisher

Among these techniques, unsupervised anomaly detection methods for multi-dimensional dataset would be of more interest in many practical cases. So, in this demo, I have selected following three typical methods.

1. Htelling's T-square method (Demo)
2. Gaussian mixture model (Demo)

Related Examples

Examples of Data Ana Predictive Maintenanc

Among many statistical an detection techniques, Hote T-square method, a multiv statistical analysis techniq

Examples of Data Ana Predictive Maintenanc

3. One-class SVM (Demo)

MATLAB Examples

The previous methods, Hc T-square method and Gau mixture model, use Gauss distribution-based parame

To emulate a realistic situation, in this demo, I will use the dataset provided by C-MAPSST (Commercial Modular Aero-Propulsion SystemSimulation) [1, 2]. This dataset includes various sensor data from aircraft engines throughout their usage cycle. The data is divided into traininng and test set. The training set has trajectories that ends at the cycle in which the failure occurs for each engine. The test set also has trajectories but ends at the cycle prior to the failure. The number of additional cycles till failure occurs for each engine in the test set is given by a separate file. Details of the data set used in this example is as follows:

Data Set: FD001

- Train trjectories: 100
- Test trajectories: 100
- Conditions: ONE (Sea Level)
- Fault Modes: ONE (HPC Degradation)

[1] A. Saxena, K. Goebel, D. Simon and N. Eklund, "Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation," International Conference on Prognostics and Health Management, (2008).

[2] Turbofan Engine Degradation Simulation Data Set

Load the Dataset

Load the training and test set of FD001.

```
dataTrain = importFDdata('train_FD001.txt');
dataTest = importFDdata('test_FD001.txt');
RULTest = csvread('RUL_FD001.txt');
```

Data Preprocessing (Training set)

As a data preprocessing, the following process is applied to the training data:

- Setting the valiable names
- Extracting effective sensors
- Labeling the condition into 4 categories based on the remaining cycles till failure occurs

The relationship between each category and remaining cycles is as follws:

1. 0~50 cycles : urgent
2. 51~125 cycles : short
3. 126~200 cycles: medium
4. 201~ : long

```
dataTrain = dataCleaning(dataTrain);
```

Data Preprocessing (Test set)

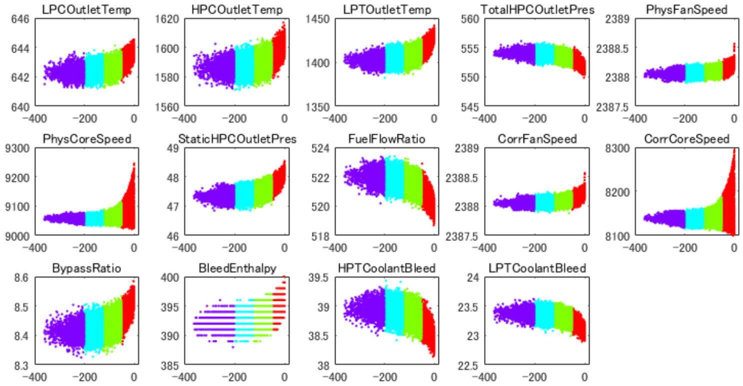
```
% Apply the same preprocessing to the test set
dataTest = dataCleaning(dataTest);

% Extract the last record of sensors for each engine
dataTest = dataTest(dataTest.Time == 0, 1:end-1);
dataTest.Time = -RULTest;

% Labeling the condition
catname = {'urgent','short','medium','long'};
dataTest.Label = discretize(-dataTest.Time, [0 51 126 201 inf],'categor
```

Visualize The Data

```
% Visualize the training data (each color represents long/medium/short/u
figure('Position',[50 500 900 420])
for kk = 1:14
    subplot(3,5,kk);
    gscatter(dataTrain.Time, dataTrain{:,2+kk}, dataTrain.Label);
    title(dataTrain.Properties.VariableNames{2+kk})
    legend('off');
end
```



Standardize Sensor Data

```
dataTrainZ = dataTrain;
[dataTrainZ{:,3:end-1}, mu, sigma] = zscore(dataTrainZ{:,3:end-1});
```

Dimensionality Reduction using PCA

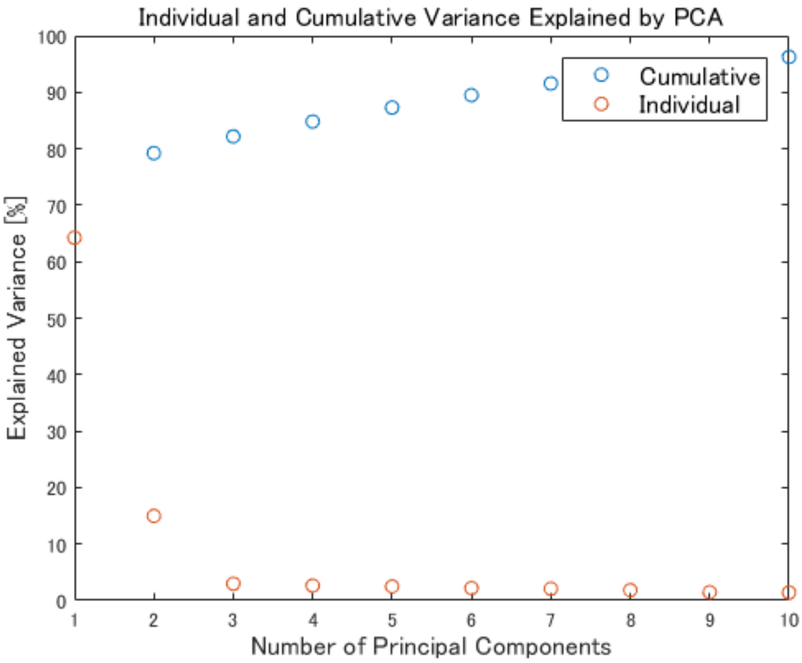
Applying PCA (Principal Component Analysis) to the standardized training set. In this process, you can see the only the first two (instead of the total 14) components explain 79% of the total variance.

In addition, by creating a plot of the first two components, you can conrirm that (1) 'long' class data forms three clusters, and (2) value of the first component increases as the the number of cycles.

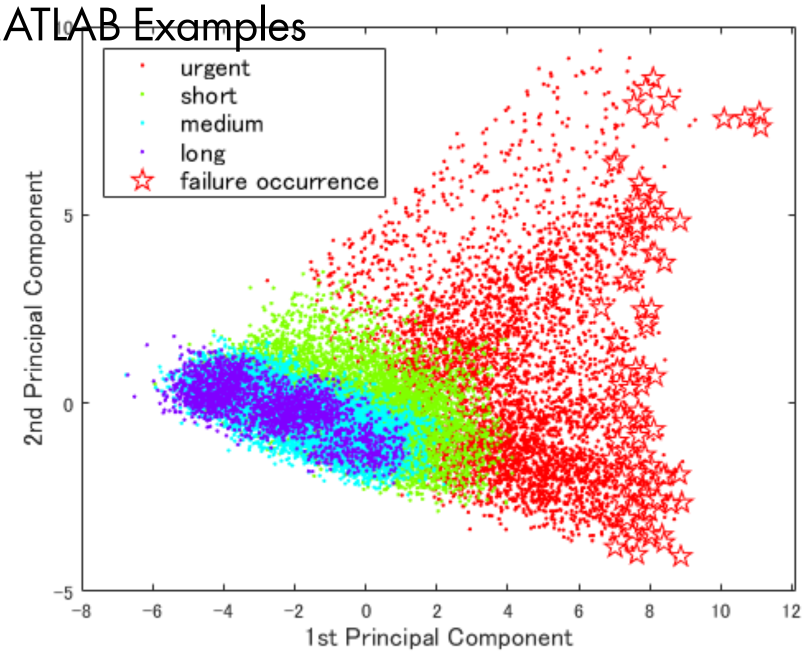
```
varName = dataTrainZ.Properties.VariableNames(3:end-1);
[wcoeff,score,latent] = pca(dataTrainZ{:,varName});

% Plot the individual and cumulative variance explained by PCA componen
figure
plot([cumsum(latent(1:10))/sum(latent) latent(1:10)/sum(latent)]*100,'o
xlabel('Number of Principal Components','FontSize', 12);
ylabel('Explained Variance [%]','FontSize', 12);
legend({'Cumulative','Individual'},'FontSize', 12);
title('Individual and Cumulative Variance Explained by PCA','FontSize',

% Creating a plot of the first two components
figure
s1 = gscatter(score(:,1), score(:,2), dataTrainZ.Label);
hold on;
idx = dataTrainZ.Time == 0;
s2 = plot(score(idx,1), score(idx,2),'rp','MarkerSize',10,'MarkerFaceCol
legend([s1; s2],{'urgent','short','medium','long','failure occurrence'},
    'Color',    [1 1 1],...
    'Location', 'northwest',...
    'FontSize', 12);
xlabel('1st Principal Component','FontSize',12);
ylabel('2nd Principal Component','FontSize',12);
```



MATLAB Examples

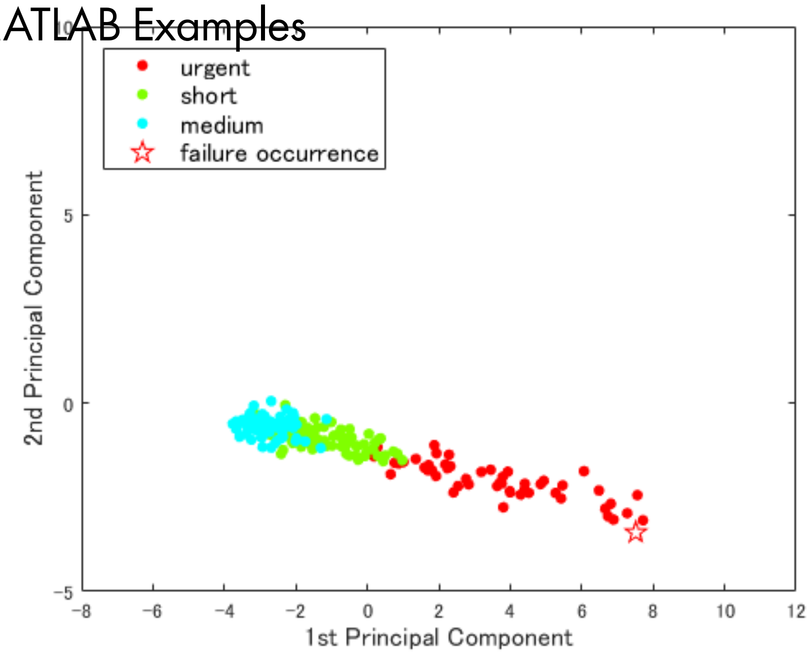


Visualize a Trajectory of Unit-1 in 1st and 2nd Principal Components Plane

```
idxUnit = dataTrain.Unit == 1;
idxFailure = idxUnit & dataTrainZ.Time == 0;

figure
s1 = gscatter(score(idxUnit,1), score(idxUnit,2), dataTrainZ.Label(idxUi
hold on;
idx = dataTrainZ.Time == 0;
s2 = plot(score(idxFailure,1), score(idxFailure,2),'rp','MarkerSize',10,
legend([s1; s2],{'urgent','short','medium','failure occurrence'},...
    'Color',    [1 1 1],...
    'Location', 'northwest',...
    'FontSize', 12);
xlabel('1st Principal Component','FontSize',12);
ylabel('2nd Principal Component','FontSize',12);
ax = gca;
ax.XLim = [-8 12];
ax.YLim = [-5 10];
```

MATLAB Examples



Save the Pre-processed Data

These data will be used in the following demos:

- Method1_HotellingsT2.m
- Method2_GaussianMixture.m
- Method3_OneClassSVM.m

```
save('Preprocessed_FD001.mat','dataTrainZ','dataTest','score','wcoeff',
```

mathworks.com