Rare Event Classification in Multivariate Time Series

Wonjae Lee ([wlee@mail.missouri.edu](mailto:wlee@mail.missouri.edu))

Kangwon Seo (seoka@missouri.edu)

University of Missouri

# How to execute the code

This section describes how to execute the code, R project file. The instruction is made based on the Windows.

1. Download and install R software (https://cran.r-project.org/bin/windows/base/old/3.4.4/).
2. Download and install R Studio (<https://www.rstudio.com/products/rstudio/download/>).
3. Unzip file (TS.zip).
4. Run R Studio, and open TS R project file (TS.Rproj).
5. Open "TS\_execute.R" file in R Studio.
6. Install packages if packages belows are not installed on your computer.

* install.packages("caret")
* install.packages("dtw")
* install.packages("C50")
* install.packages("gmodels")
* install.packages("kernlab")
* install.packages("e1071")

1. Use "setwd" function to set working directory where your test file (\*.csv) is located.

* setwd("path/TS")

1. Implement the "my\_test" function with your file name. The process time will take 3 – 5 minutes depending on the test data and your computer performance.

* my\_test("test.csv")

1. Check the result (precision, recall, FPR, F-measure).

# Data Challenge Report

**Problem description**

A multivariate time series (MTS) is produced when multiple interconnected streams of data are recorded over time. They are commonly found in manufacturing processes that have several interconnected sensors collecting the data in over time. In this problem, we have a similar multivariate time series data from a pulp-and-paper industry with a rare event associated with them. It is an unwanted event in the process—a paper break, in our case—that should be prevented. The objective of the problem is

1. predict the event before it occurs, and
2. identify the variables that are expected to cause the event (in order to be able to prevent it).

**Challenges in the problem**

The data that we received is the Multivariate time series data which contains the information on the breaks that are rarely occurred. The number of breaks in the data is only 124 which is not enough to find the features to predict the occurrences in other test data. Most radical changes are occurred right at the time when it breaks. It is hard to find the distinct feature for early detection. Also, the data has a number of variables which we didn’t have any information about.

**Approach to address the challenges**

### Feature extraction (1-NN)

To extract the feature of the time series data, we measure the similarity between two temporal sequences. Even though we tried both DTW (Dynamic Time Warping) and Euclidean distance, we chose to use Euclidean distance considering model performance, time to analyze and efficiency based on the multivariate time series.

### Early detection (Moving the y column)

The model that we submit is to detect one-step (2 min) earlier by moving the y column (class) up by 1 row. Also, we took the second derivate of variables to capture the abrupt change in time series.

### Classification (Decision Tree)

We gained train data set and test data set as a result of 1-NN Euclidean distance. For the purpose of classification, the decision tree model was applied to these data set. To improve the model performance, adaptive boosting and cost matrix are used.

**Classification Result**

The model is designed to detect the break one-step earlier (2 minutes). Because of the lack of data which has only 124 breaks and the fact that events are rarely occurred during operation, it’s hard to truly evaluate the model performance. We divided data into train and test (0.9 train, 0.1 test) as instruction asked and estimated the model performance (Precision, Recall, False Positive Rate and F-1 measure).

|  |  |
| --- | --- |
| TN (True Negative) | FN (False Negative) |
| FP (False Positive) | TP (True Positive) |

* Precision = TP / (TP + FP)
* Recall = TP / (TP + FN)
* FPR = FP / (FP + TN)
* F-measure = (2 × TP) / (2 × TP + FP + FN)

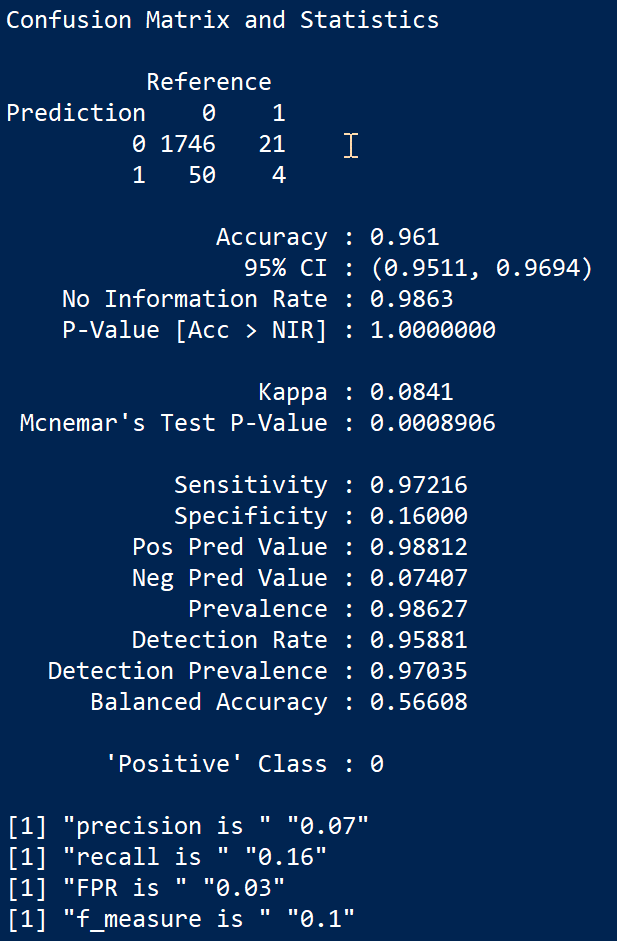


Figure 1. The result of model Evaluation

**Root Cause Analysis**

As we mentioned earlier, we used the decision tree model (C5.0) with the result of 1-NN Euclidean distance for the time series classification. We measured the variable importance for C5.0 model using C5imp function. According to the decision tree model, we figure out that the columns x2, x3, x18, x32, x40, x43 affect the classification model the most. See Table 1, Figure 2 for more details.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Rank | Variable | Attribute usage | Rank | Variable | Attribute usage |
| 1 | x2 | 100% | 32 | x7 | 40% |
| 2 | x3 | 100% | 33 | x22 | 37% |
| 3 | x18 | 100% | 34 | x14 | 37% |
| 4 | x32 | 100% | 35 | x24 | 35% |
| 5 | x40 | 100% | 36 | x10 | 34% |
| 6 | x43 | 100% | 37 | x48 | 34% |
| 7 | x37 | 99% | 38 | x11 | 32% |
| 8 | x5 | 97% | 39 | x55 | 31% |
| 9 | x34 | 92% | 40 | x25 | 30% |
| 10 | x54 | 92% | 41 | x19 | 28% |
| 11 | x59 | 91% | 42 | x15 | 26% |
| 12 | x12 | 88% | 43 | x17 | 25% |
| 13 | x31 | 84% | 44 | x45 | 24% |
| 14 | x33 | 82% | 45 | x23 | 23% |
| 15 | x51 | 79% | 46 | x57 | 23% |
| 16 | x20 | 78% | 47 | x30 | 23% |
| 17 | x39 | 68% | 48 | x21 | 23% |
| 18 | x38 | 66% | 49 | x42 | 20% |
| 19 | x49 | 62% | 50 | x58 | 20% |
| 20 | x44 | 58% | 51 | x60 | 14% |
| 21 | x41 | 57% | 52 | x35 | 14% |
| 22 | x1 | 57% | 53 | x8 | 11% |
| 23 | x47 | 56% | 54 | x53 | 10% |
| 24 | x9 | 52% | 55 | x29 | 9% |
| 25 | x26 | 45% | 56 | x52 | 9% |
| 26 | x36 | 44% | 57 | x27 | 8% |
| 27 | x13 | 44% | 58 | x56 | 5% |
| 28 | x46 | 43% | 59 | x28 | 0% |
| 29 | x6 | 42% | 60 | x50 | 0% |
| 30 | x16 | 41% | 61 | x61 | 0% |
| 31 | x4 | 41% |  |  |  |

Table . Variable Importance Measures for C5.0 Models

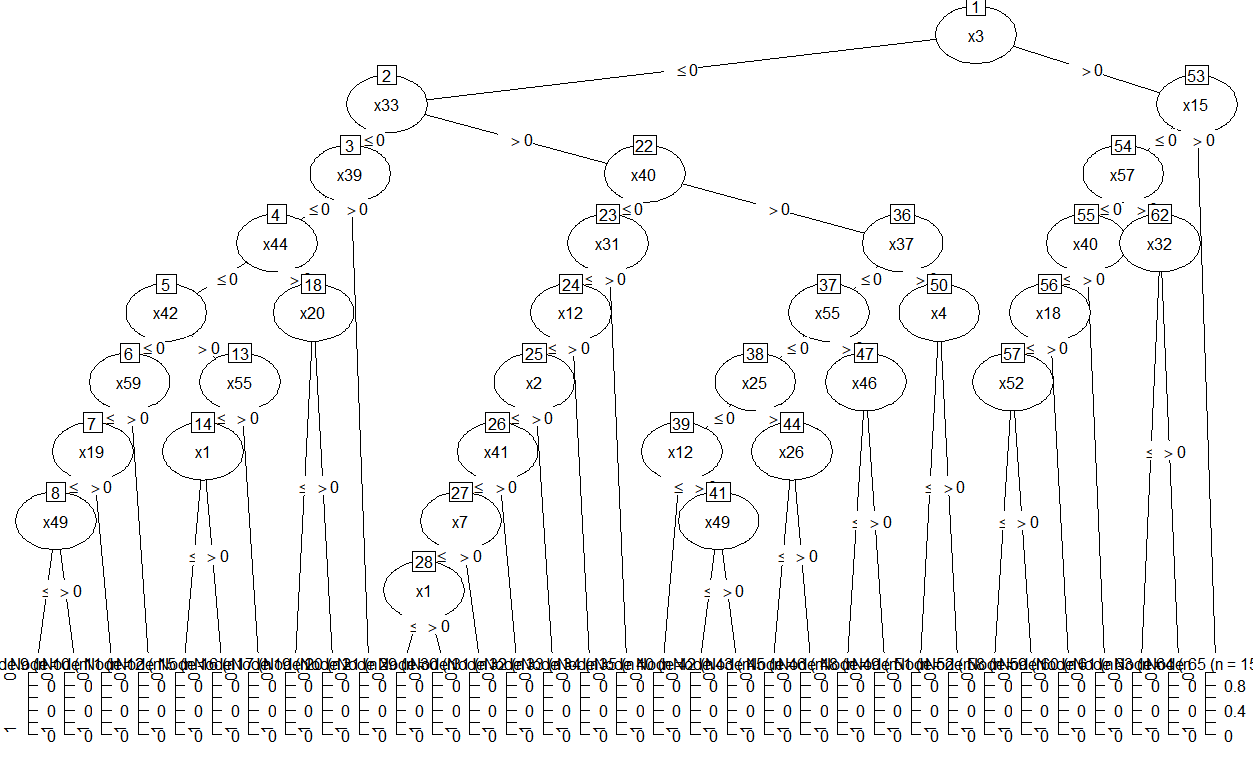


Figure . C5.0 Decision Tree Plot