s c c h
software competence center
hagenberg

FUZZY.LOGIC.LAB.LINZ

Advances in Knowledge-Based Technologies

Proceedings of the Master and PhD Seminar Summer term 2007, part 2

Softwarepark Hagenberg
SCCH, Room 2/8
July 5, 2007

Software Competence Center Hagenberg
Softwarepark 21
A-4232 Hagenberg
Tel. +43 7236 3343 800
Fax +43 7236 3343 888
www.scch.at

Fuzzy Logic Laboratorium Linz
Softwarepark 21
A-4232 Hagenberg
Tel. +43 7236 3343 431
Fax +43 7236 3343 434
www.flll.jku.at

Program

16:00–17:00 Session 1 ***(Chair: Bernhard Moser)***

16:00 Cheng He:
Learning of Decision Trees with C4.5 and CART

16:30 Holger Schöner, Edwin Lughofer:
Challenges and Solutions for Process Monitoring

Learning of Decision Trees with C4.5 and CART

Zheng He 0656422
e-mail: hz_23@hotmail.com

July 2, 2007

Abstract

The well-known learning algorithm of decision tree **C4.5** is a successor and extension of **ID3**. It has the same procedure to construct decision tree from training datasets, deal with noise and treat attributes with unknown values as **ID3**, which we have described in the previous introduction. Besides, **C4.5** has some special tricks to refine the initial decision tree just like pruning, generalizing production rules, windowing method, grouping attribute values and interacting with classification models, which we would like to introduce in the following contents.

Another widely used decision tree learning algorithm is **CART**, which is short for $Classification and Regression Trees$. Nowadays, the construction of **CART** has become a common basic method for building statistical models from simple feature data. **CART** is powerful because it can deal with incomplete data, multiple types of features (floats, unnumbered sets) both in input features and predicted features, and the trees it produces often contain rules which are humanly readable. Also, **CART** provides a general framework that can be instantiated in various ways to produce different decision trees.

After introducing these two famous learning algorithm, we do some experimental application to compare the performance of them in the following section.

1 Challenges and Solutions for Process Monitoring

Holger Schöner, Edwin Lughofer
Software Competence Center Hagenberg GmbH (SCCH),
Fuzzy Logic Laboratorium Linz-Hagenberg (FLLL)

1.1 Abstract

Automating industrial production processes can greatly benefit from the ability of on-line detection of instabilities in measured machine and product parameters. In this paper we describe a framework and its application to on-line instability detection for a discrete manufacturing process, injection moulding.

1.2 Introduction

One application for data driven methods, a focus of the IDM group of the SCCH and of the FLLL, in the area of process control is the stability detection prototype for injection moulding machines, which was developed for Engel Austria GmbH. Here, as in other industrial applications, the production process can greatly benefit, in terms of quality and reliability, from a classification of the production state while producing each part. During stable production phases, the quality of produced parts is usually stable as well. When an instability is detected, it is also interesting to identify the reason for it. Such stability recognition allows a further automation of machine monitoring, and can allow improvments in the quality of produced parts.

This task setting leads to certain requirements concerning the stability detection. Because of the wide variety of machines, the applied method has to be flexible, adaptive, and very robust. The system has to be easily configurable for the machine manufacturer, and even more so for the machine buyer. Additionally, because of the planned integration into machine control, the resource consumption of the used methods has to be very limited. Finally, another important requirement is the possibility to obtain diagnostic information about the reasons for detected instabilities.

To satisfy these requirements we developed a framework based on very simple and specialized detection methods, whose detections on several sensor channels are integrated into one overall stability prediction. The available channels, between 10 and 50 depending on the machine and configuration, produce one value for every produced part, among which are temperatures, pressures, timing and geometrical information. Each method can give details about its detection reasons, which are summarized by the integration framework.

Unfortunately, the problem setting of detecting instabilities can be ill-posed, as there is (often) no general definition of stability of a production process. Often it is empirically understood as time intervals, in which the quality of produced parts is acceptable and there are no evident delays or disturbances in the production process. In the literature, there are approaches for outlier detection, anomaly detection, detection of unusual patterns, or of abnormal regimes. Even definitions of these widely used terms can vary; and there are definite differences in the applicability of methods developed for certain of these problems.

Well known methods to deal with outlier data are limit methods (defining upper and lower acceptance limits) and discrepancy methods (scoring deviation from expected values or learned predictions), as mentioned in [03]. They are usable very efficiently, but are limited to certain kinds of anomalies, mainly jumps and deviations from learnable or well known dependencies. Nevertheless, they are very reliable for detecting outliers, easily configurable or adaptable, and are incorporated as components in our

framework. Multivariate methods, e.g. clustering or density estimation, often are not sensible for high dimensional data, although there are approaches for circumventing these (e.g. dimensionality reduction or projection methods in [04]). There is also a host of interesting, more sophisticated methods (local model parameter estimation and comparison [03], pattern frequency estimation [05], compressibility analysis [06], discord detection [07]). For our problem setting, these methods are not applicable, because they are either too complex for online application, are suitable only for post-hoc analysis, they require labelled training data, or dynamic channel addition or removal cannot easily be handled.

None of the existing approaches could fulfil the requirements, so we decided to take different simple and robust approaches, and combine them in a framework integrating the decisions of the submodels into an overall decision, and providing diagnostic information about the reasons for its decision.

1.3 Integration Framework

1.3.1 Detection Methods

For its decisions, the framework relies on sub-models. Among these are limit methods and discrepancy methods for detection of single outliers (e.g. sticking parts) or jumps (e.g. change of production parameters). We did not find methods able to detect transients, and especially their duration. These occur especially during initial production, when the machine has to get warm, or after production parameter changes. Instead, we developed a new method for detection of these, based on a sliding linear regression of the last few samples. The slope of this line is compared to those slopes occurring during normal operation of the machine. Before, the slope is normalized using a variety of methods to make this method applicable to a wide variety of machines and channel types. Furthermore, because no labels for the data exist, which would tell about normal operation phases, the recent history of each channel is taken into account, after filtering with robust outlier detection.

1.3.2 Normalization and Diagnostics

Because the individual decesions of the sub-models on the different channels have to be compared for reaching an overall decision, we normalized the output of each model into the interval from 0 (stable) to 1 (surely instable). For instabilities above 0.3, each method generated a short description about where (channel, time) and what kind (how much above limit, etc.) of instability was detected. For normalization, we used a transformation function; a kind of sigmoid was appropriate for transformation from the interval [0; ∞], which most methods produced as output, to [0; 1].

1.3.3 Framework

Often, an instability detection problem involves not only one channel, but several ones, with varying characteristics. Furthermore, sometimes any single of the above mentioned simple methods for instability detection will not suffice. Consequently, our instability detection framework consists of a collection of several instances of the detection methods described so far. It can be configured, which method with which parameter settings should be applied to any given channel, although the normalizations performed for each method ensure that, for many channels, some generic default parameters suffice, thus reducing problem dependent configuration efforts.

Each model, after application of the normalization, returns a value between 0 and 1. All the values have to be integrated into a single instability prediction. A very simple method, which we are currently using is to simply return the maximum of the predictions any of the methods returns. This seems reasonable, as the final prediction honors the largest instability detected by any of the methods in any

of the channels. In addition, the diagnostic information of each model is collected, structured, and summarized if necessary, thus allowing an operator to determine the reason and significance of a detected instability.

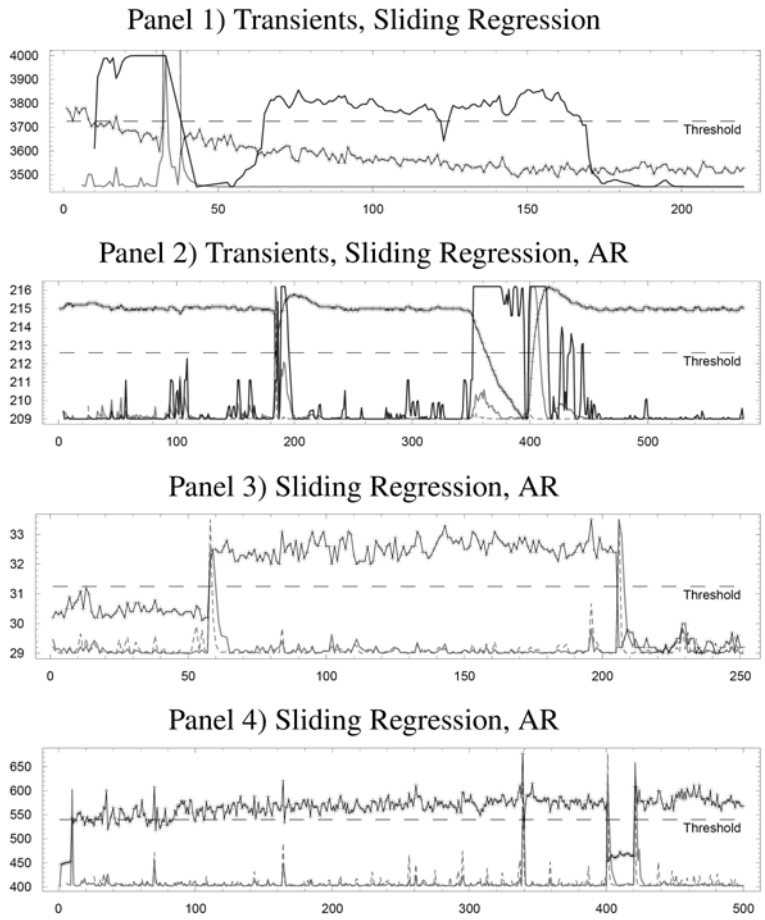

Figure 1: ***Examples of method performances on injection moulding data. Samples are placed along horizontal axis, vertical axis shows channel values. Instability predictions by different methods are overlaid, with 0 (no instability) at the bottom and 1 (almost certain instability) at the top of the plot. Black solid lines with + marker: channel data, dark solid lines, light grey lines, light grey dashed lines: individual detection methods.***

More sophisticated integration methods are possible. It could be useful to take into account dependencies between the model responses, introduced because they are working on the same or related channels or because they are using the same detection method. This could help to avoid overdetections (because more than one model must detect instabilities), or to detect weak instabilities by allowing predictions of different models to amplify each other.

1.4 Application

The system described in the previous Sections was developed for instability detection in a variety of injection moulding machines. Data is sampled from the available channels once for each part produced, and contains information about configuration settings, timing, forces and pressures, temperatures, speed, and dimensions, among others. Figure 1 shows examples for data from some of the more important channels, and detections by different instances of the presented methods (online, i.e. using only data points up to the respective point in time; not all methods used on all channels).

Several channels contain mainly instabilities manifesting themselves as more or less obvious jumps (panels 3 and 4); these can be handled quite well using the limit and discrepancy methods. Depending on the detection sensitivity desired, either the threshold (here set to 0.5) or the configuration of the method can be adapted to allow detection of more or less instabilities, if desired.

Some other channels, containing mostly smooth curves with transients (panel 2) or even mixtures of transients and single outliers or jumps (panel 1), are harder to analyze, also for human experts, one of the reasons, why the dataset is not labeled. Unfortunately, because of this it is not possible to tell, where the detections are right or wrong (and thus to give values for performance measures like sensitivity or selectivity). Furthermore, the really interesting question would be, whether the produced part was acceptable, or not. As it is often not possible to tell for sure, the consequence is, that these channels and the detections should be interpreted as hints, when it would make sense to check for the quality of the produced parts. Which can be an important improvement anyway.

1.5 Bibliography

[01] J. Korbicz, J. Koscielny, Z. Kowalczuk, and W. Cholewa, *Fault Diagnosis - Models, Artificial Intelligence and Applications*. Berlin Heidelberg: Springer Verlag, 2004.

[02] L. Chiang, E. Russell, and R. Braatz, *Fault Detection and Diagnosis in Industrial Systems*. London, Great Britain: Springer Verlag London Berlin Heidelberg, 2001.

[03] S. Bay, K. Saito, N. Ueda, and P. Langley, *A framework for discovering anomalous regimes in multivariate time-series data with local models*. In Symposium on Machine Learning for Anomaly Detection, Stanford, U.S.A., 2004.

[04] C.C. Aggarwal and P.S. Yu, *Outlier Detection for High Dimensional Data*. SIGMOD Conference, 2001.

[05] E. Keogh and S. Lonardi and W. Chiu, *Finding Surprising Patterns in a Time Series Database in Linear Time and Space*. Proc. of the Eighth ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, pages 550-556, 2002.

[06] E. Keogh and S. Lonardi and C.A. Ratanamahatana, *Towards Parameter-Free Data Mining*. Proc. of the Tenth ACM SIGKDD Int. Conf. on Knowlede Discovery and Data Mining, 2004.

[07] E. Keogh, J. Lin, and A. Fu, *Hot sax: Efficiently finding the most unusual time series subsequence*. In Proceedings of the 5th IEEE International Conference on Data Mining (ICDM 2005), Houston, Texas, 2005, pp. 226-233.

1.6 Acknowledgements

We thank Engel Austria GmbH for the permission to publish results from our cooperation.