Pattern-based feature extraction for fault detection in quality relevant process control

Serena Peruzzo, Mike Holenderski, and Johan J. Lukkien
Department of Mathematics and Computer Science, Technische Universiteit Eindhoven
Den Dolech 2, 5600 AZ Eindhoven, The Netherlands

***Abstract*—Statistical quality control (SQC) applies multivariate statistics to monitor production processes over time and detect changes in their performance in terms of meeting specification limits on key product quality metrics. These limits are imposed by customers and typically assumed to be a single target value, however, for some products, it is more reasonable to target a range of values. Under this assumption we propose a multi-stage approach for mapping operating conditions to product quality classes. We use principal component analysis (PCA) and a pattern mining algorithm to reduce dimensionality and identify predictive patterns in time series of operating conditions in order to improve the performance of the classifier. We apply this approach to an industrial machining process and obtain significant improvements over models trained using features based on the last value of each process variable.**

I. Introduction

Modern industrial processes are increasingly complex and produce a large amount of process and quality data that can be used to optimize their performance. Developments in data collection and computing capabilities in the last two decades have made it possible to use large amounts of data for this purpose. Methods can be classified in pure data-based and integrated model-data based methodologies. Among the pure data-based techniques, multivariate statistical analysis has had huge success because of the ability of handling large amounts of data and highly correlated datasets.

Quality relevant process control aims to control the operating conditions such that the output quality is maintained at a desired level. Over the last two decades, research efforts in this area have focused on multivariate statistical approaches such as Principal Component Analysis (PCA) and Partial Least Squares (PLS) [4]. Both techniques extract variable correlations and project the operating condition data into a lower dimensional space. The subspace is then monitored in order to detect major deviations of the quality metrics from their desired level.

Multivariate statistical approaches are generally easily implemented but have limited capabilities in dealing with dynamic processes and non linearity in complicated plants. Moreover they are only able to deal with real valued targets. In quality relevant control, output quality specification limits are generally imposed, directly or indirectly, by customers and often refer to a *range of values*, rather than a single target. In these situations, it makes sense to rephrase the problem of quality control as a classification task, aiming to control the operating conditions such that the output quality falls within the range. Doing so, the specification limits partition the space of operating conditions in two subspaces: *in control* and *out of control*.

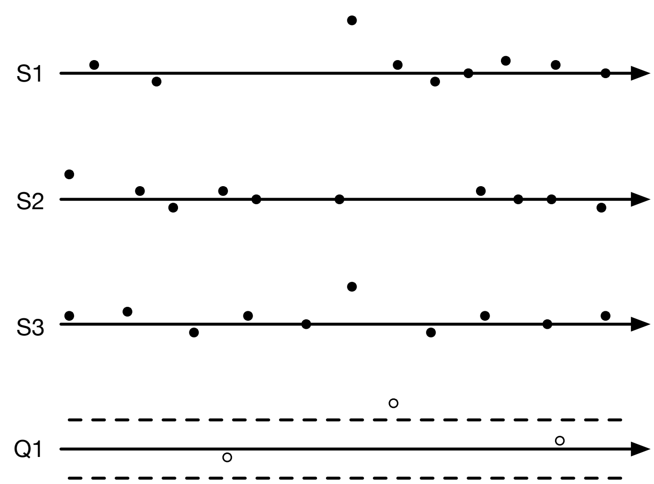

Fig. 1. An example of three operating conditions and one quality metric data over time. For each parameter, the intervals between measurements are irregular, and not necessarily aligned with the other parameters. The dashed lines illustrate the specification limits for the quality metric.

For solving the classification problem it is important to define features that summarize the history of the process and identify patterns in the operating conditions leading to a situation of *out of control*. Most classification algorithms are designed for simple data and are not easily adapted to time series analysis. A common approach to feature extraction for this purpose is computing one or more summary statistics for each time series over a window of fixed length, but this method doesn't allow to exploit more complex relationships between the variables. Mining temporal data and analyzing these relationships is made difficult by variability in the sampling intervals and temporal granularity, and missing data. Over the last decade, pattern mining algorithms have been used to circumvent this problem and create features that represent not only the historic behavior of a univariate time series, but also its relationship to the other time series in the system.

In this paper we propose a method for data driven quality control that can deal with quality target ranges, exploits the history of operating condition data (rather than just the last value), and is resilient to irregular time intervals and missing data. The proposed approach is based on a combination of traditional statistical quality control with a pattern mining algorithm, exploiting the large amount of data available without losing the temporal relationships between the operating conditions and output quality. We evaluate the approach on data from an industrial machining process used for cutting complex shapes into metal pieces.

The paper is structured as follows. Section II defines the problem, followed by a discussion of the related work in Section III. Our proposed method is described in Section IV and evaluated in Section V.

II. Problem Definition

An industrial process is represented by quality data sampled from $Y = \mathbb{R}^{n \times q}$ and operating conditions sampled from $X = \mathbb{R}^{n \times m \times p}$, where n, q, m and p are respectively the number of samples, the number of quality metrics, the number of operating variables and the size of the operating window corresponding to a single quality measurement (see Figure 2). Each quality metric $j \in \{1, \dots, q\}$ is associated with a set of specification limits (T_L^j, T_U^j) such that the process is defined out of control iff *any* of the quality metrics is found outside of specification limits, i.e. $y_i \in \{0, 1\}$, with

$$y_i = \begin{cases} 0 & \text{if } \forall j \in \{1, \dots, q\} : T_L^j \leq y_{i,j} \leq T_U^j \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

where $y_i = 1$ means that the process is out of control at the i^{th} quality measurement. Let $f : X \rightarrow \{0, 1\}$ be a model mapping the operating conditions to in or out of control. Using the cross entropy cost function, the objective is to learn a model $\hat{f}$, such that

$$\hat{f} = \arg\min_f - \sum_i \left(y_i \log(f(x_i) + (1 - y_i) \log(1 - f(x_i)) \right) \tag{2}$$

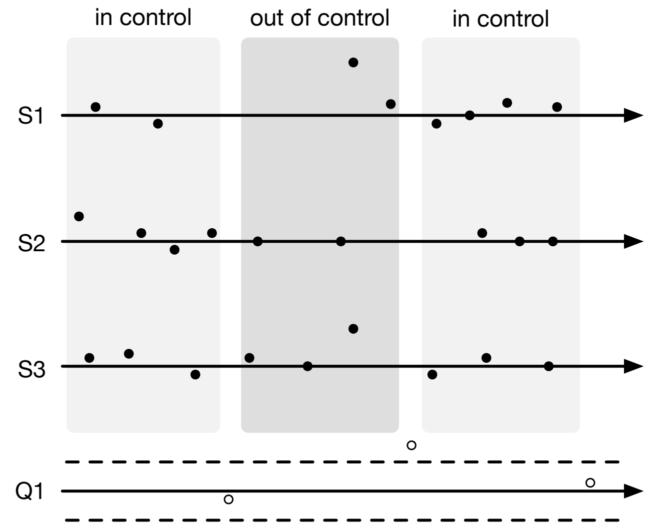

Fig. 2. An example of samples for the three output quality measurements. The quality metric determines the label (in control or out of control).

III. Related Work

Multivariate statistics has been extensively used in monitoring of quality variables and detection of quality problems. Research efforts in this area focus on variation of the PLS algorithm that can provide a complete monitoring of the output variations and a concise decomposition of the input data space into output-relevant and input-relevant subspaces [4]. The multivariate statistical approach has two main limitations: it can only be applied to numeric quality measures, and it does not take into account the past history of the process (only the last value). An attempt to design a procedure that can cope with discretized quality is done in [5], [7]. The approach proposed combines PCA with Linear Discriminant Analysis (LDA) successfully applied in relating operating conditions and product quality in the steel industry. The problem of including history in the data can be solved using pattern-based feature selection techniques to identify local properties or patterns in the time series.

In [1], [6] patterns are extracted by discretizing the time series and mining the symbolic intervals to identify frequent patterns in the data. Each sequence is then transformed into a binary vector indicating the presence or absence of the frequent patterns, which is input to a conventional classifier. They focus on the challenge of efficiently searching for frequent, relevant and non redundant features, which can be time consuming and a serious limitation. The technique we propose allows to deal with categorical quality metrics, taking into account the history of the process without sacrificing efficiency in the searching algorithm.

IV. Methodology

In this section we summarize our approach for learning a classifier for out of control conditions in industrial processes. The basic assumption is that recent history of operating conditions is more suited to explain the variability in the labels, then just their last value. However, learning a classifier from the raw operating conditions data is difficult because of the large number of variables and the fact that they are each measured at different and irregular intervals. We use principal component analysis (PCA) and a pattern mining algorithm [1] to represent each instance x_i as a fixed-length feature vector x'_i, preserving as much information on the history of the variables as possible. To do this we apply the following steps:

1) Project the raw operating condition data into a lower dimensional space using PCA.
2) Transform each projected instance into multivariate state sequences.
3) Apply a mining algorithm to extract predictive temporal patterns from the operating conditions.
4) Represent each instance x_i as a binary vector using the patterns obtained at step 2.

Applying this transformation we obtain a training set that allows the use of standard machine learning classifiers to learn f from the training set $D = \{(x'_i, y_i)\}_{i=1}^{n}$. The following sections explain the steps of this process.

A. Principal Component Analysis

We use PCA to reduce the number of variables and extract the latent structure of the operating condition data. PCA is a non-parametric technique used to project high dimensional, intercorrelated data into a lower dimensional space while retaining as much information as possible. The new set of variables (principal components) are obtained as a linear combination of the original ones, and are uncorrelated and ordered so that the *top* ones explain most of the variance in the original dataset. Let X be a $n \times m$ matrix where n and m are respectively the number of samples and the number of variables measured. Then $\hat{X} = XP$ where $\hat{X}$ is an other $n \times m$ matrix, related to X by a linear transformation P. The columns of P are the principal components of X and are the eigenvectors of the covariance matrix $C_X = \frac{1}{n} X X^T$. The rows of $\hat{X}$ are the representation of the original data in the principal components space. For more details of the PCA algorithm we refer readers to [8].

B. Temporal Abstraction and Multivariate State Sequences representation

Temporal abstraction (TA) is the process of transforming a point time series into a series of state intervals and is widely used in data mining to aggregate multivariate time series into a representation that is easier to analyze. This higher level representation is equivalent to a smoothing of the data and solves a number of common problems of time series analysis, such as irregular intervals in the measurements, different granularity across time series, and missing data.

Instead of a series of points, a variable is represented by a series of intervals during which the value is constant; the original points are begin and endpoints of these intervals. Moreover, the values are discretized using an abstraction alphabet Σ that represents the set of possible *value ranges* a variable can assume (Figure 3). Let S be a state represented by the tuple (F, V) where F is a variable and $V \in \Sigma$. An *interval state* is a state that holds over a time interval and is represented by a 4-tuple $E = (F, V, s, e)$ where F is a variable, $V \in \Sigma$, and s and e are respectively the start and end time of the state interval.

After creating interval states for all the time series in $\hat{X}$, every sequence $\hat{x}_i \in \hat{X}$ is represented as a multivariate state sequence (MSS)

$$Z_i = \langle (E_1, ..., E_l) : E_{j.s} \leq E_{j+1.s} \wedge 1 \leq j < l \rangle \quad (3)$$

with $Z_{i.e}$ representing the end time of the sequence.

C. Mining of frequent temporal patterns

We use the definition of a Recent Temporal Pattern (RTP) and the mining algorithm in [1] to find predictive patterns in our dataset. For the purpose of the RTP mining algorithm, two types of temporal relationships are defined:

- E_i is **before** E_j, denoted as $b(E_i, E_j)$, if E_i ends before E_j starts, i.e. if $E_{i.e} < E_{j.s}$
- E_i **co-occurs** with E_j, denoted as $c(E_i, E_j)$, if E_i starts before E_j and there is a non empty period of time where both E_i and E_j occur, i.e. if $E_{i.s} \leq E_{j.s} \leq E_{i.e}$

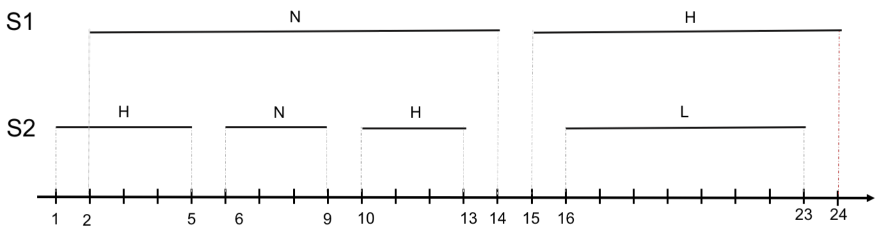

Fig. 3. Example of MSS with two variables measured over 24 time units. The values of S1 and S2 are discretized as low (L0, normal (N) or high (H) and the sequence is represented as a series of state intervals $Z = (E_1 = (S2, H, 1, 5), E_2 = (S2, N, 2, 14), E_3 = (S2, N, 6, 9), E_4 = (S2, H, 10, 13), E_5 = (S1, H, 15, 24), E_6 = (S2, L, 16, 23))$

A temporal pattern $TP = ((S_1, ..., S_k), R)$ is a k-pattern, i.e. an abstract representation of a series of k states and their pairwise relationships, represented in the upper triangular matrix R, with $R_{i,j} \in \{b, c\}$ for $i \in \{1, ..., k-1\} \wedge j \in \{i+1, ..., k\}$. Given an MSS Z, a pattern TP and a maximum gap parameter g, TP is a Recent Temporal Pattern in Z, denoted $RTP_g(TP, Z)$ if there exists a mapping from the states of TP to the state intervals of Z (Figure 4). Additionally, the last state of TP should map to a recent state interval of Z, i.e. S_1 maps to $E_j \in Z_i$ with $Z_{i.e} - E_{j.e} \leq g$, and any pair of consecutive states in TP should map to state intervals of Z no more than g away from each other.
Given a dataset D and a gap parameter g, the support for an RTP TP is defined as $sup_g(TP, D) = |\{Z_i : Z_i \in D \wedge RTP_g(TP, Z_i)\}|$. Then TP is a frequent RTP in D if $sup_g(TP, D) \geq \sigma$, for some threshold σ.

Given the inputs σ, g and D, the mining algorithm finds all frequent single states, then alternates between two phases until no more RTPs are found:

1) **Candidate generation**: Generate candidate (k+1)-patterns by extending k-RTPs backward in time
2) **Counting**: Obtain the frequent (k+1)-RTPs by removing all RTPs with $sup_g \leq \sigma$

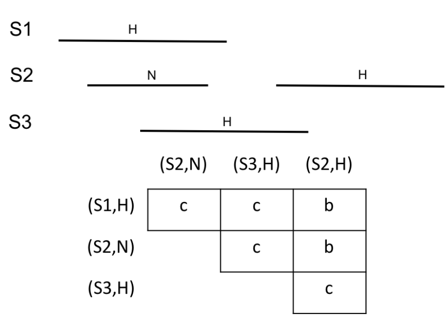

Fig. 4. a temporal pattern with states $((S1, H), (S2, H), (S3, H), (S2, H))$ and temporal relations $R_{1,2} = c$, $R_{1,3} = c$, $R_{1,4} = b$, $R_{2,3} = c$, $R_{2,4} = b$ and $R_{3,4} = c$

The mining task is performed separately on the two classes, using a separate σ for each class. This makes particular sense when the classes are inbalanced: searching for frequent patterns using a global threshold for sup_g may result in discarding patterns that are frequent in the smaller classes but not across the entire dataset.
Once the frequent RTPs are found in each class, they are combined in a single set Ω and used to transform the multivariate sequences into binary vectors. For each MSS Z_i we obtain a vector x'_i where each $x'_{i,j}$ corresponds to a pattern $TP_j \in \Omega$ and is equal to 1 iff TP_j is an RTP in Z_i and 0 otherwise.

D. Classification

Given a set of multivariate labelled time series we learn a classifier applying the following steps:

1) Transform all quality metrics into binary variables according to the specification limits
2) Transform all sequences in X into multivariate state sequences (MSS)
3) Tranform every MSS in a binary vector where $x_i = 1$ if pattern i is in the sequence, 0 otherwise
4) Use the binary vectors in the training data to learn a classifier

V. Evaluation

We apply the methodology described in the previous sections to a machining process and compare the results with a model trained on features obtained from the last value of the process variables, and traditional approaches to statistical quality control.

A. Experimental Set Up

We consider the case of a production line that uses a machining process for cutting complex shapes into metal pieces. Data is collected from several sources:

- Material: every time a new batch of raw materials arrive at the machining process, several sample measurements are taken from the full batch of raw products.

- Tooling: various metrics (including product counters) regarding the state of the tooling.
- Process: sensor output measured at every machine, aggregated after producing one item.
- Quality: geometrical quality metrics for pieces, sampled and measured every few hours per machine.

For this experiment we only employ 19 process variables as input of the model, and 4 quality metrics as target variables. The dataset includes a total of 669 sequences, 117 of which are labelled out of control. The sets of thresholds for the discretization of the quality variables to a single binary value are defined by experts. The segments of operating condition time series between one quality measurement and the following one, represent the multivariate sequences to be classified.

B. Results

We first run PCA and transform the original input matrix into a lower dimensional space keeping the top 3 principal components, then use temporal abstraction as described in section IV-B. The abstraction alphabet is defined as $\Sigma = \{VL, L, N, H, VH\}$ and we use the 10th, 25th, 50th, 75th and 90th quantile to discretize the time series with VL (Very Low) for values below the 10th percentile, L (Low) for values between the 10th and 25th percentile and so on.
Finally we set the minimum support σ to 0.15 and the gap parameter g to 10 minutes[1] and run the mining algorithm separately on each class to extract predictive patterns to use as features to build a classifier using linear discriminant analysis.

We compare the classification performance of three classifiers:

1) **Naive**: the features are formed based on the most recent measurement of the process variables.
2) **PLS**: the target variable is predicted with a two step process combining PLS regression on the quality metrics and discretization of the results. This is currently the approach used by our industrial partner for quality relevant control of their machining process [2].
3) **RTP**: the features correspond to the frequent recent temporal patterns found in the dataset.

We compare the results of the three methods using three different metrics: precision and recall calculated with respect of the *out of control* class, and overall accuracy. In terms of business case, we are particularly interested in maximizing the value of recall, i.e. the proportion of sequences correctly predicted as *out of control*. Figure5 summarizes the results of a cross-validation process adapted to time series: the model is trained on 3 months worth of data and used to predict the quality for the following week, computing the evaluation metrics on these predictions. This step is repeated moving the training window forward by 1 week until all the data is exhausted.

The results show that classification methods based on temporal pattern mining obtain significantly better results when it comes to predicting *out of control* sequences. The recall is improved with increasing the history length, up to a point. The decrease in recall for very long histories can be explained by longer histories containing patterns which are not discriminating and instead add to the noise, countering the benefit of taking history into account. The higher accuracy obtained with *naive* model is due to the highly unbalanced dataset: the total number of sequences *out of control* is relatively small compared to the *in control* so that even misclassifying the large majority of them, the model achieves relatively high accuracy overall. Even obtaining a lower overall accuracy the method based on pattern mining outperforms the other methods on recall, suggesting that including historical behavior can improve the classification of sequences of operating condition data.

In the current approach, the frequent patterns from all classes are combined in a single set of patterns, which are subsequently used as input features for training the classifier. The resulting set of patterns is prone to contain short patterns which are common to all classes, similar to stop words in Natural Language Processing. It remains to be investigated whether the set of features can be reduced by focusing on the representative patterns, i.e. those which are unique for each class or not commonly shared between classes, in combination with feature selection.

VI. Conclusion

Data-driven process control and monitoring has been a popular field of research over the last two decades. Research efforts in this area have focused on multivariate statistical approaches that project the operating condition data into a lower dimensional space and monitor it to identify *out of control* situations. PCA and PLS can be easily implemented and effective when the relationship between process and quality variables is linear and the quality target is clearly specified as single value, but

[1] The values for σ and g were chosen after performing a grid search.

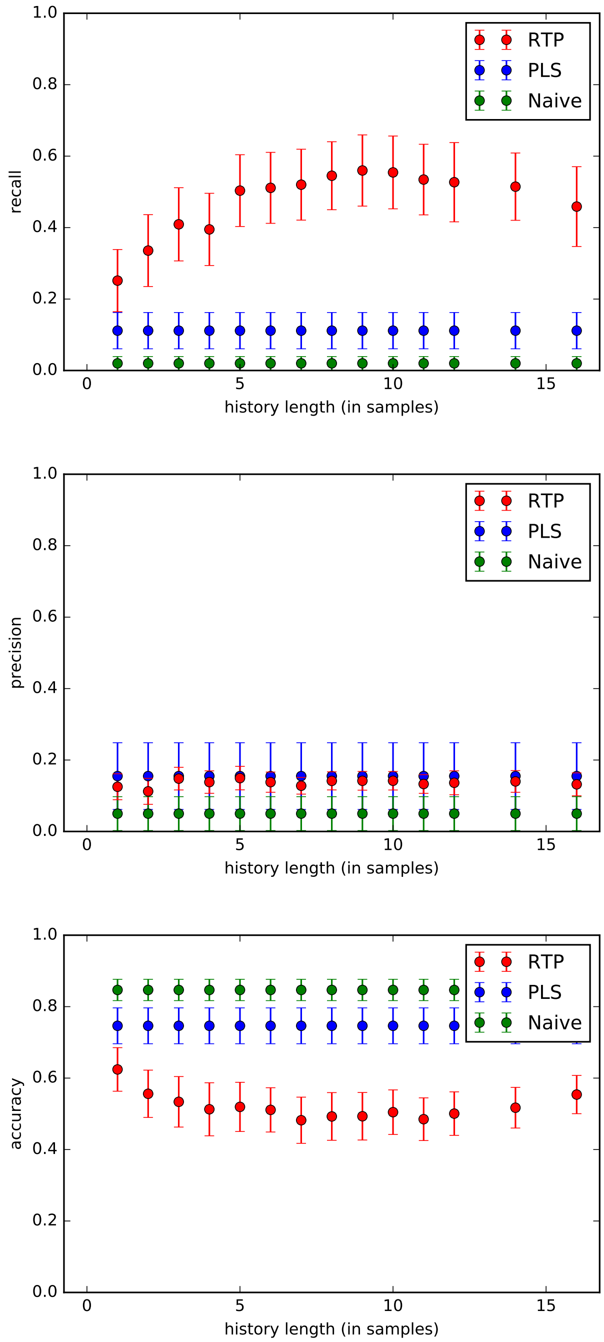

Fig. 5. Recall and precision for different history lengths. The error bards show the standard error of the mean recall or precision.

they have limited capabilities in dealing with dynamic, non-linear systems and qualitative quality metrics. We rephrase the issue of quality control into a classification problem and propose an approach for extracting relevant features and partitioning the operating conditions in *in control* and *out of control*. Applying temporal abstraction we were able to overcome common difficulties in the analysis of multivariate time series, such as irregular intervals, different granularity and missing data, whilst the use of pattern mining to search for predictive temporal patterns in both subspaces allows to take into account the history without the aid of any domain knowledge. We tested this approach on an industrial machining process and obtain a significant increase both in precision and recall of the classification model.

ACKNOWLEDGMENTS

We would like to thank Paulien Dam and Bas Tijsma for their contributions and feedback. This work has been partly supported by the ECSEL project MANTIS [3].

REFERENCES

[1] Iyad Batal, Hamed Valizadegan, Gregory F. Cooper, and Milos Hauskrecht. A temporal pattern mining approach for classifying electronic health record data. *ACM Trans. Intell. Syst. Technol.*, 4(4):63:1–63:22, October 2013.

[2] Paulien Dam. Multivariate process monitoring and control applied to an electrochemical process. Master's thesis, University of Groningen, 2014.

[3] ECSEL-JU 662189. MANTIS: Cyber physical system based proactive collaborative maintenance. http://www.mantis-project.eu/, 2015.

[4] S. Joe Qin. Statistical process monitoring: basics and beyond. *Journal of Chemometrics*, 17(8-9):480–502, 2003.

[5] Manabu Kano, Koichi Fujiwara, Shinji Hasebe, and Hiromu Ohno. Product quality improvement using multivariate data analysis. In *Preprints of the 16th IFAC World Congress*, 2005.

[6] Robert Moskovitch and Yuval Shahar. Classification of multivariate time series via temporal abstraction and time intervals mining. *Knowledge and Information Systems*, 45(1):35–74, 2015.

[7] Yoshiaki Nakagawa, Shigemasa Nakagawa, Manabu Kano, and Takashi Tanizaki. Quality improvement of steel products by using multivariate data analysis. In *SICE Annual Conference 2007*, pages 2428–2432, Sept 2007.

[8] Jonathon Shlens. *A Tutorial on Principal Component Analysis*, 2014.