CHEMICAL ENGINEERING TRANSACTIONS, Volume 11, 2007
Edited by Sauro Pierucci
Copyright © 2007, AIDIC Servizi S.r.l., ISBN 978-88-95608-00-6

Cluster Analysis in Process and Environmental Monitoring

S. Beaver[1], A. Palazoglu[1], and J.A. Romagnoli[2]
[1]University of California, Davis, CA USA
[2]Louisiana State University, Baton Rouge, LA 70803 USA

A clustering algorithm is proposed for multivariate time series data sampled from cyclic systems. It applies Principal Component Analysis (PCA) in an unsupervised fashion to identify a number of recurring system states from historical data sets; times of occurrence for each identified state are labelled. A cycle of period *T* samples is known to exist for at least some system variables. This "confounding cycle" represents unimportant variability that can be statistically separated from variability at other, important time scales having events of interest. A moving window is used to generate *N* subsets (or batches) of time series, which are then subjected to cluster analysis. The window length *L*, the number of samples in each batch, determines the time scale at which patterns are isolated. One case study focuses on regimes of normal and faulty operation for a pilot scale chemical process. Another case study is aimed at determining meteorological patterns affecting air quality in northern California.

1. Introduction

Clustering algorithms (Everitt, 1993) belong to a class of unsupervised, multivariate statistical analysis techniques used to determine homogeneous groups of observations existing in a heterogeneous data set. Ultimately, a clustering algorithm seeks to partition a set of *N* observations into *k* groups (called "clusters") based on measurements for *p* variables sampled for each observation. When applied to time series data sets, in which the *N* observations are recorded serially in time at a uniform sample rate for some physical system, the identified clusters can correspond to the system states or regimes. The sequence of such cluster labels describes the temporal evolution of the system through a discrete set of states. Continuous periods in time bearing the same cluster label are associated with persistent realizations of some system state, while points in time at which the cluster labels change indicate transitions between system states.

The proposed clustering method is intended for data sets exhibiting two important features. First is the inherently autocorrelated nature of the measurements, which may represent events occurring at multiple time scales. Time scales can range from brief disturbances affecting only a single sample to steady states that may persist for long periods of time. The second is a cyclic component superimposed on at least one of the process variables. Such a "confounding cycle" can be caused by the presence of a periodic disturbance or other input affecting the system. The cycle is known to affect the process, and represents undesired variability that should not be reflected in the ultimate labelling of the *N* observations among a set of *k* identified process states.

Traditional clustering algorithms, such as the *k*-means algorithm, are intended for independent observations; this assumption is violated for time series data sets. Such algorithms group observations based on mean squared deviations in the *p*-dimensional measurement space, yielding clusters that are distinguished by the levels of their means. Dynamic events do not have constant mean, and thus cannot be properly detected by traditional clustering algorithms. Additionally, cyclic variables do not have a truly constant mean, but rather oscillate through peaks and valleys. Thus, traditional clustering algorithms tend to produce “periodically biased” results when applied to cyclic time series measurements. Such periodically biased cluster labels capture the phase of the confounding cycle, tracking the oscillations for the cyclic variables. This problem becomes severe for identifying any steady states existing in the data set, as the cyclic component represents the dominant source of variability for these system states. Because of the inappropriate nature of their statistical models, traditional clustering algorithms are not useful for clustering autocorrelated and/or cyclic data.

Here, a clustering algorithm for autocorrelated and cyclic data sets is presented in which the data are modelled using Principal Component Analysis (PCA, Jackson, 1991). Though PCA is technically not appropriate for time series data, it is frequently applied to time series measurements with success. PCA is a linear model based on the correlation structure of the observed data and is also capable of modelling linear trajectories of the variables in time. In the event that the system dynamics are too complicated to be adequately modelled by PCA, an extension known as Dynamic PCA (Ku et al., 1995) can be implemented by simply concatenating temporally lagged variables to the observed data matrix.

To implement the cluster analysis, a moving window is first applied to divide the time series data set into N batches of equal length L samples. The clustering algorithm then partitions these N batches into k clusters. The moving window length L determines the time scale for any detected patterns. By setting L equal to the period of the confounding cycle T, any periodic biases can be averaged out from the cluster labels. Unfortunately, this window length will also average out high frequency events of shorter duration than the confounding cycle. Thus, setting $L = T$ reveals any low frequency states existing in the time series. A separate analysis using smaller L can then reveal any high frequency events having time scales of less than T samples.

The proposed clustering method is applied in two case studies. The first considers various states for a pilot scale chemical plant, and the second investigates meteorological patterns affecting air quality in northern California.

2. Theory

2.1 (Dynamic) Principal Component Analysis

PCA is performed for a rank p data matrix $\boldsymbol{X}$ by Singular Value Decomposition (SVD) to yield a diagonal matrix $\boldsymbol{S}$ of ordered singular values s_i and corresponding right singular vectors $\mathbf{v}_i$ appearing in the columns of $\boldsymbol{V}$.

$$\boldsymbol{X} = \boldsymbol{USV}^T \tag{1}$$

The $q < p$ singular vectors corresponding to the largest singular values are stacked into the PCA loading matrix $\boldsymbol{P}$. The PCA model order q is taken as the smallest integer capturing some desired threshold level of variability in data set $\boldsymbol{X}$ (e.g. 95%).

$$\text{\% Variance Captured} = \frac{\sum_{i=1}^{q} s_i}{\sum_{i=1}^{p} s_i} \times 100 \tag{2}$$

The scalar error metric Q quantifies the degree of model fit upon projection of data matrix $\boldsymbol{X}$ onto PCA model $\boldsymbol{P}$, where $\boldsymbol{I}$ is an identity matrix.

$$Q = \left\| X(I - P^T P) \right\|_2 \tag{3}$$

In the event that the autocorrelated data are not adequately modelled using PCA, an extension known as Dynamic PCA (Ku et al., 1995) can be applied. Data matrices $\boldsymbol{X(t\text{-}m)}$ having each element lagged by m sampling intervals are concatenated to form a single matrix $\boldsymbol{X}$ of rank $(M{+}1)p$. M is the number of lags and is estimated using the Partial Autocorrelation Function (PACF) as described by Shumway and Stoffer (2000).

$$\boldsymbol{X} = [\boldsymbol{X(t), X(t\text{-}1), \ldots, X(t\text{-}M)}] \tag{4}$$

2.2 Clustering Algorithm

Prior to executing the clustering algorithm, a moving window is used to divide the continuous, time series data set into N batches $\boldsymbol{X}_i$ labelled serially for index i ranging 1 to N. The moving window is defined by the window length of L samples and spacing between adjacent windows of R samples. These N batches $\boldsymbol{X}_i$ are then input to the clustering algorithm to generate N homogenous groups of time series data.

The non-hierarchical clustering algorithm begins by specifying k, the number of clusters. A single batch $\boldsymbol{X}_i$ is randomly selected to seed each cluster. The clustering algorithm is iterative in nature and seeks to optimize the configuration of the N batches among k clusters. For each iteration, all batches assigned to each cluster r are collected to estimate the PCA models $\boldsymbol{P}_r$ as shown using Equation (1). Then, each individual batch $\boldsymbol{X}_i$ is projected into each PCA model $\boldsymbol{P}_r$ to determine loss rates $Q(i,r)$. The iteration is completed when each batch $\boldsymbol{X}_i$ is reassigned to the cluster r producing the smallest modelling error $Q(i,r)$. The iterations are continued until no reassignments are indicated— each batch is assigned to the cluster whose PCA model produces the least error.

The method of Beaver and Palazoglu (2006) is applied to compute an aggregated cluster solution based on a randomly initialised ensemble of solutions generated by the above clustering algorithm. This method avoids having the user specify the parameter k in advance, and generates a reproducible and robust partitioning of the observations. Ultimately, each batch of L observations is assigned to a single cluster. As the batches may be overlapping in time, however, multiple cluster assignments may exist for a given sampling time. Final, fractional cluster assignments for each sample are calculated as the fractional representation of sample i in cluster r.

2.3 Parameterization of the Moving Window to Identify Events of Interest

The window length L is specified in advance by the user to determine the temporal properties of the cluster solution. Different choices for L will identify events occurring at different time scales. The spacing between temporally adjacent batches R is a second parameter that defines the moving window. Parameter R is used to define the window density LR^{-1}, or number of batches in which each sample is contained. The window density determines the temporal resolution of the cluster solution— larger window densities allow increased temporal resolution at which the transitions points between system states occur. The cluster analysis is performed for two independent and complimentary parameterizations of the moving window to reveal both the high and low frequency content of the time series data.

First, the low frequency content of the historical data is isolated by setting $L = nT$, where T is the confounding cycle period and n is a positive integer. This window length ensures each batch contains exactly n observations from each phase of the confounding cycle, thereby averaging out any periodic biases from appearing in the cluster labels. Due to this relatively long window length, high frequency events of short duration also tend to be averaged out of the cluster labels. Thus, this choice of window parameterisation indicates low frequency events of at least nT samples in duration. Because of the inherent loss of temporal resolution for the cluster transition points occurring with such large window lengths, the window spacing R is chosen to produce a small integer window density to maximize computational efficiency.

If desired, a second, independent cluster analysis can be performed to isolate any high frequency events existing in the historical data. A window spacing $L < T/2$ is selected to capture events of shorter duration than the cycle period T. Using such a window length causes the previously identified low frequency events to appear with periodic biases in the sequence of cluster labels. The true high frequency states can readily be identified upon corroboration between the low and high frequency cluster analyses. The window spacing R should be set to 1 to maximize the temporal resolution for the high frequency analysis.

3. Case Study: Pilot Plant Monitoring

The clustering algorithm is applied to pilot plant data (Joe et al., 2004). Feed is heated with steam before entering a jacketed, water cooled, reactor. Cyclic variability for several process variables exists due to periodic disturbances in the steam utility. Nine variables are monitored for 400 observations. The cycle period is 21 samples based on visual inspection of the data. The plant is operated through 4 sequential regimes of 100 observations each: normal operation (samples 1-100), feed flow rate increase (101-200), and 2 different faults (201-300 and 301-400). Additionally, several “spike faults” exist in which the coolant temperature is reduced for a single sample in duration (92, 100, 292, 300, 392, and 400). The pilot plant data are shown in Figure 1.

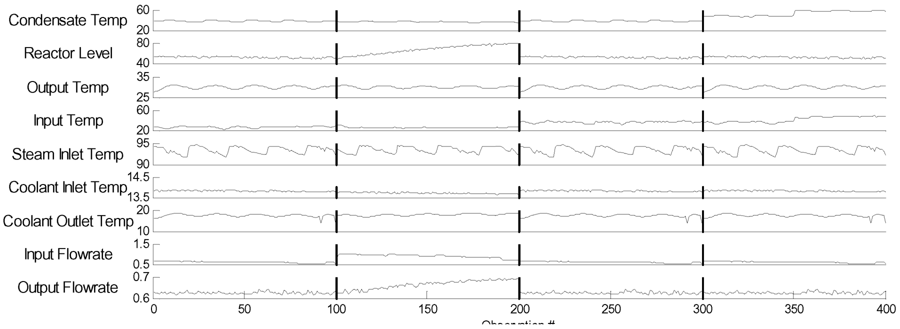
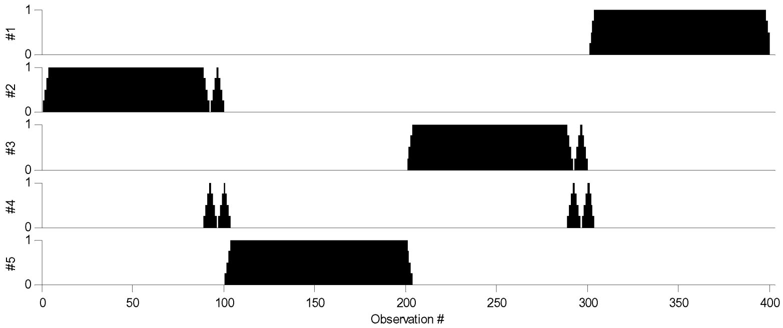


Figure 1. Time series for 9 pilot plant variables. Dashed vertical lines denote regimes.



.*Figure 2. Fractional cluster assignments for 400 observations.*

The large mean shifts in the data can be detected using PCA, and Dynamic PCA is not necessary for this application. Low frequency analysis is performed using L = 21 samples and R = 7 samples. Each of the 4 operating regimes are identified. Next, the high frequency analysis is performed using widow length of 4 samples and window spacing of 1 sample. The 4 previously identified operating regimes appear in the solution with periodic biases. True high frequency events are identified as the transitions between the low frequency states and the spike faults. The cluster labels are shown in Figure 2. Clusters #2, #5, #3, and #1 capture the four sequential process regimes, while #4 isolates the spike faults.

4. Case Study: Meteorological Patterns Affecting Air Quality

The clustering algorithm is applied to hourly wind field measurements obtained from a network of 12 monitoring stations positioned throughout the San Francisco Bay Area of California. Wind speed and direction from each meteorological station are transformed into northerly and easterly vector components and treated as individual variables, for a total of 24 variables. A confounding cycle of period 24 hours exists in the data set due to the inherent diurnal (daily) cycle inherent in most environmental systems.

A window length of $L = 48$ hours is used to both suppress diurnal biases in the cluster labels and identify events occurring at the synoptic time scale— meteorological states persisting for multiple, consecutive days on each realization. Due to the strong autocorrelation present in the hourly wind observations, the clustering algorithm is implemented using Dynamic PCA with $M = 2$ lags. High frequency events (those persisting < 24 hrs) are not investigated for this data set because ground-level ozone (photochemical smog) levels are known to be strongly influenced by synoptic meteorological variability for the Bay Area. Each day from the study period of 1 June through 30 September of the years 1996-2004 is labelled using 4 identified cluster patterns. Days are assigned to clusters based on majority membership for their 24 hourly observations (Figure 3). The clusters are distinguished in terms of dispersion patterns and ozone levels. Clusters #1 and #4 capture meteorological conditions favoring significantly poorer air quality than #2 or #3.

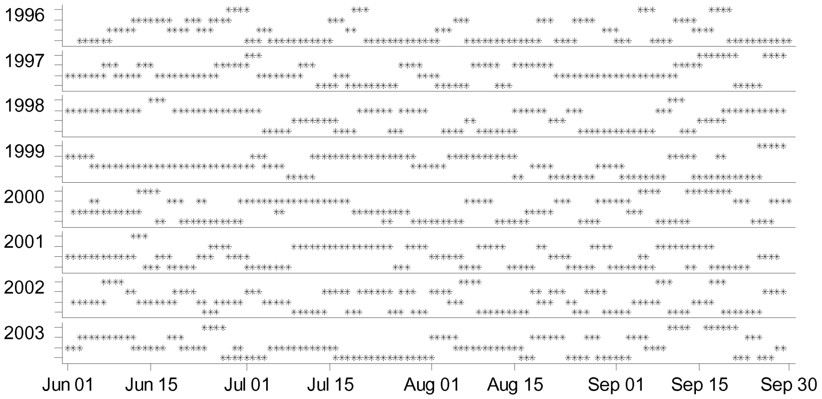


Figure 3. Y-position of asterisk indicates cluster label for each day from 8 summers.

3. References

Beaver, S. and A. Palazoglu, 2006, A Cluster Aggregation Scheme for Ozone Episode Selection in the San Francisco, CA Bay Area. Atmos. Environ., 713, 40.

Everitt, B. S., 1993, Cluster Analysis. Heinemann Education, London.

Jackson, J. E., 1991, A User's Guide to Principal Components. Wiley, New York.

Joe, Y.Y., D. Wang, J.A. Romagnoli, and A. Tay, 2004. Robust and Efficient Joint Data Reconciliation-Parameter Estimation Using a Generalized Objective Function. *7 Int. Symp. On Dynamics and Control of Process Systems*, Cambridge, MA, 2004.

Ku, W., R.H. Storer, and C. Georgakis, 1995, Disturbance Detection and Isolation by Dynamic Principal Component Analysis. Chemom. Intell. Lab. Syst., 179, 30.

Shumway, R.H. and D.S. Stoffer, 2000, Time Series Analysis and Its Applications, Springer, New York.