IOP Conf. Series: Journal of Physics: Conf. Series **1487** (2020) 012005 doi:10.1088/1742-6596/1487/1/012005

Quadratic Mutual Information based Regression for Prediction of Quality Variable in Batch Process

Zheng Li[1,2,3,4, a], Pu Wang[1,2,3,4], Xuejin Gao[1,2,3,4, b], Yongsheng Qi[5] and Huihui Gao[1,2,3,4]

[1]Faculty of Information Technology, Beijing University of Technology, Beijing, China
[2]Engineering Research Centre of Digital Community, Ministry of Education, Beijing, China
[3]Beijing Laboratory for Urban Mass Transit, Beijing, China
[4]Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing, China
[5]School of Electric Power, Inner Mongolia University of Technology, Hohhot, China

Email: [a]lizeebm78@163.com; [b]gaoxuejin@bjut.edu.cn

Abstract. Quality prediction is of great importance for batch processes. Predicting quality variable is a challenging task because of various factors such as strong nonlinearity and non-Gaussian exist in batch data. A quadratic mutual information based regression method is proposed to handle the problem. The proposed method takes into account higher order statistics that reveal the non-linear dependencies between the process variables and important quality variables. Furthermore, the proposed method is implemented without the hypothesis of Gaussian distribution of the dataset as in MPLS. The effectiveness of the QMIR method is illustrated by a dataset of industrial Escherichia coli fermentation process, compared with MPLS.

1. Introduction

Batch processes have gain great importance in modern manufacturing industries, in which the product quality is usually available by offline assay after a single batch run. Biomedical industry, fine chemistry, pharmaceutical, and semiconductor industry are all typical batch processes. In batch processes, some key variables always exist and can be treated as indicators of production status or the final product quality. For example, biomass concentration is usually a key index to reflect the final product quality in a fermentation process, which is a typical batch process. However, these key variables, also known as quality variables, are hardly measured online at present. An online sampling but "offline measuring in lab" is always adopted, which is always accompanied by measurement delay. The measured delay is impeding the direct use of quality variables for closed loop control. Online prediction of quality variables has become a challenging task for modelling and quality control of batch processes.

The regression models, using the easy-measured process variables (input data) to estimate the hard-measured quality variables (output data), provide an effective way to implement the prediction of quality variables. Inherent nonlinearity, non-Gaussian, and batch-to-batch variations are unique characteristics of batch processes. Multiway partial least squares (MPLS) is one of the most popular prediction methods for batch processes. However, MPLS is a second-order method, which extracts features by taking into account the second-order statistics such as variance or covariance [1]. Compared with variance or

covariance, mutual information (MI) measures nonlinear dependencies between features and paying attention to high order statistics existing in the original data, which is derived from information theory [2]. Quadratic mutual information (QMI) is first proposed by Principe J.C as a new distance measure, which also reflects the nonlinear dependencies between variables or features [3, 4]. The advantage of QMI is that it combines Renyi's entropy with Parzen window, providing a more simple methodology for estimating the necessary probability density functions than MI. Vera et al. proposed an unsupervised feature extraction method by maximizing the QMI among the original input and the feature space data [5]. Torkkola used QMI as a criterion to find a linear projection of features in classification problems [6].

In this paper, a QMI based regression (QMIR) method is proposed to predict quality variables for batch processes. A linear projection is performed on the process variable matrix to get the new features. The feature extraction method is based on QMI metric which has its roots in Cauchy–Schwarz inequality and Renyi's quadratic entropy. The optimal projection is obtained not only maximizing the QMI between the transformed input process variables (features) and the key quality variables, but also maximizing the Renyi's entropy of process variables. QMI aims at finding the most relevant features of quality variables as well as preserving the data distribution of process variables.

2. Theoretical background

2.1. Renyi's entropy

Entropy was defined in information domain to demonstrate the average information conveyed by a certain event X. Shannon's entropy plays an important role in information theoretic domain. Renyi's entropies were introduced as a generalization of Shannon's entropy [7]. The differential Renyi's entropy with order α of a certain event X is defined as:

$$H_{R\alpha}(X) = \frac{1}{1-\alpha}\log\int f^{\alpha}(x)dx \tag{1}$$

Here, $f(x)$ is the probability density function (pdf). When α=2, it becomes

$$H_{R2}(X) = -\log\int f^{2}(x)dx \tag{2}$$

$H_{R2}(X)$ is also called the Renyi's quadratic entropy since $f(x)$ is in its square form. Shannon's entropy can be considered as a limiting situation of Renyi's entropy when $\alpha \to 1$. If the aim is to manipulate entropy, then Renyi's entropy and Shannon's entropy are almost equal [8]. For a computation scope, Renyi's entropy outperforms Shannon's entropy because it is much easier to estimate by using Parzen window method.

The Parzen window, also named kernel function, is used for the estimation of the pdf directly from the samples. For a set of samples $\{x_i\}_{i=1}^{N}$, the pdf is estimated as:

$$\hat{f}(x) = \frac{1}{N}\sum_{i=1}^{N} k(x, x_i) \tag{3}$$

Here, $k(x, x_i)$ represents the Parzen window, or the kernel function, centred at x_i .It must satisfy the properties of a pdf. Among all kinds of kernel functions, the most widely used is the Gaussian kernel:

$$k(x, x_i) = G(x - x_i, \sigma) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2\sigma^2}(x - x_i)^2\right) \tag{4}$$

The width of the window, or the kernel size, is decided by parameter σ. In this way, the estimation of $H_{R2}(X)$ of a sample set $\{x_i\}_{i=1}^{N}$ by using Parzen window is:

$$H_{R2}(X) = -\log\int \hat{f}^{2}(x)dx = -\log V(X) \tag{5}$$

$$V(X) = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G(x_i - x_j, 2\sigma) \tag{6}$$

It can be seen from equation (6) that Parzen window is an entropy estimator based on a sum of kernels in a pairwise form of the samples.

2.2. Quadratic mutual information

MI was first introduced in the domain of communication by Shannon and it measures the removal of uncertainty of event Y when event X has happened. MI is presented as [9]:

$$I(X,Y) = H_S(Y) - H_S(Y \mid X) = \iint f_{XY}(xy) \log \frac{f_{XY}(x,y)}{f_X(x) f_Y(y)} dxdy = KL(f_{XY}(x,y),\, f_X(x) f_Y(y)) \tag{7}$$

Shannon's entropy $H_S(Y) = -\int f_Y(y) \log f_Y(y) dy$ is revisited for the definition of MI. $f_X(x)$ and $f_Y(y)$ are the individual marginal pdfs, and $f_{XY}(x, y)$ is the joint pdf. We can see from equation (7) that mutual information is also the Kullback-Leibler divergence (KL divergence) between the joint pdf, i.e., $f_{XY}(x, y)$ and the factorized marginal pdfs, i.e., $f_X(x)f_Y(y)$. If the mutual information is zero, it indicates that the two events are independent statistically from each other [10]. It should be noted that the Shannon's definition of MI is not easily estimated from samples since the logarithm is just inside the integral. From equation (1) and equation (2), we can see that Renyi's entropies are logarithm of sum of a certain power of pdf, so it is much easier to be estimated directly from data samples. Quadratic mutual information (QMI) is proposed to estimate the distance between pdfs which is integrated with the quadratic of pdfs. The QMI with Euclidean distance (ED) is:

$$QMI_{ED}(X,Y) = \iint f_{XY}{}^2(x,y)dxdy + \iint f_X{}^2(x) f_Y{}^2(y)dxdy - 2\iint f_{XY}(x,y) f_X(x) f_Y(y)dxdy \tag{8}$$

According to Cauchy-Schwarz inequality, $QMI_{ED}(X,Y) \geq 0$. The equality will hold only if X and Y are statistically independent. Let

$$QMI_{ED}(X,Y) = V_E = V_J + V_M - 2V_C \tag{9}$$

Where

$$V_J = \iint f_{XY}{}^2(x,y)dxdy, \quad V_M = \iint f_X{}^2(x) f_Y{}^2(y)dxdy, \quad V_C = \iint f_{XY}(x,y) f_X(x) f_Y(y)dxdy \tag{10}$$

Here, V_J represents the pairwise interaction of samples in the joint space. V_M is the interaction in factorized marginal space and V_C indicates the cross information between different spaces.

2.3. Batch data reprocessing

Unlike continuous processes data only consisting of two-dimensional arrays, i.e., variable direction and time direction, original data of batch processes is integrated with an additional attribute, i.e., batch direction. Variable-wise unfolding is adopted here to transform the cubic data matrix into an ordinary data matrix [11]. As shown in figure 1, a collection of historical batch data is denoted as a three-dimensional matrix $\boldsymbol{X}_o(I \times M \times K)$. M input process variables are sampled during the K sample times with a batch run and I is the batch number. Firstly, $\boldsymbol{X}_o(I \times M \times K)$ is unfolded variable-wise into a two-dimensional matrix $\boldsymbol{X}_v(KI \times M)$. Suppose L quality variables are also measured over K sample time. Quality variables matrix $Y_o(I \times L \times K)$ can be unfolded in the same way as process variables to get $Y_v(KI \times L)$. Let $N=KI$. After the unfolding step, the historical batch data set is now reorganized into an input data set $X_v \in R^M$ with N samples and an output data set $Y_v \in R^L$ with N samples, respectively. For a more simplistic expression in the follow-up modelling process, X_v and Y_v are both rearranged in their corresponding transpose, i.e., $X=X_v^{\mathrm{T}}=[x_1,\ldots, x_N]$ and $Y=Y_v^{\mathrm{T}}=[y_1,\ldots, y_N]$.

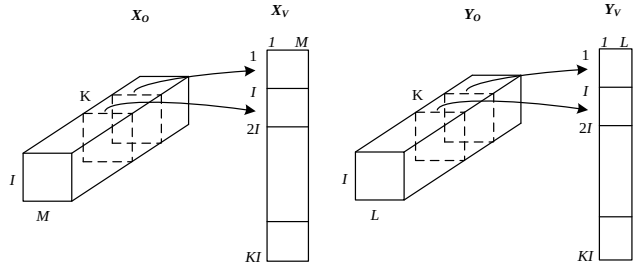

Figure 1. Variable-wise unfolding of three-dimensional batch data.

3. QMI based regression (QMIR)

3.1. Object function

Our goal is to extract optimal features from high-dimensional input space X and built the regression model between the features and the output space Y. The features should best satisfy the followings: (1) the features can best describe the QMI between original input and output variables, in other words, QMI between input data and output data is maximized. (2) The features should also reveal the underlying structure of input data in terms of Renyi's entropy.

As shown in figure 2, high-dimensional input X (R^M) is first projected to a lower dimensional vector (R^r) through a parametric mapping $T=g(X, W)$. W is a set of parameters. To lower the computational complexity, a linear projection is adopted to get the form $T=W^TX$ [7]. In this way, W becomes the $M\times r$ projection matrix whose columns are consist of projection axes. r ($r<M$) represents the dimension of each new feature. T is the score vector in the new feature space. A new objective function $J(W)$ is proposed in our paper not only maximize the QMI between input space and output space but also maximize the Renyi's quadratic entropy of input space X. $J(W)$ is proposed as:

$$J(W) = -H_{R2}(T) - \alpha \cdot QMI_{ED}(T,Y) \tag{11}$$

α is the weight parameter which is used to balance the importance of $H_{R2}(T)$ and $QMI_{ED}(T, Y)$. In this way, our goal is to minimize J.

As mentioned earlier, the quadratic mutual information between input feature space T and Y is estimated by Parzen window estimators, yields:

$$\begin{aligned} &QMI_{ED}(T,Y) = V_E = V_J + V_M - 2V_C \quad V_J = \iint \hat{f}_{TY}{}^2(t,y)dtdy = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G(t_i - t_j, 2\sigma_t^2)G(y_i - y_j, 2\sigma_y^2) \\ &V_M = \iint \hat{f}_T{}^2(t)\hat{f}_Y{}^2(y)dtdy = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G(t_i - t_j, 2\sigma_t^2)\cdot\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G(y_i - y_j, 2\sigma_y^2) \\ &V_C = \iint \hat{f}_{TY}(t,y)\hat{f}_T(t)\hat{f}_Y(y)dtdy = \frac{1}{N}\sum_{i=1}^{N}\left\{\frac{1}{N}\sum_{j=1}^{N} G(t_i - t_j, 2\sigma_t^2)\cdot\frac{1}{N}\sum_{j=1}^{N} G(y_i - y_j, 2\sigma_y^2)\right\} \end{aligned} \tag{12}$$

For the entropy of input feature space, Renyi's quadratic entropy is calculated according to equation (5) and equation (6), we get

$$H_{R2}(T) = -\log\int \hat{f}_T{}^2(t)dt = -\log V(T) = -\log\left\{\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G(t_i - t_j, 2\sigma_t^2)\right\} \tag{13}$$

3.2. Learning algorithm

The optimal W^* is to be found which minimize $J(W)$, yields

$$W^* = \arg\min_W J(W) \tag{14}$$

Gradient descent method is applied to search for the optimize solutions of W. An iterative process is started with a random vector. W is updated in every iteration step as follows until J converges:

$$W(t+1) = W(t) - \lambda\frac{\partial J}{\partial W} \qquad \frac{\partial J}{\partial W} = -\frac{\partial H_{R2}(T)}{\partial W} - \alpha\frac{\partial V_E}{\partial W} = \frac{1}{V(T)}\cdot\frac{\partial V(T)}{\partial W} - \alpha\frac{\partial V_E}{\partial W} \tag{15}$$

λ represents the learning rate. For the linear projection $T=W^TX$, we can write:

$$\frac{\partial(t_i - t_j)}{\partial W} = (x_i - x_j)^T \tag{16}$$

We obtain

$$\frac{\partial V(T)}{\partial W}=\frac{\partial V(T)}{\partial(t_i-t_j)}\frac{\partial(t_i-t_j)}{\partial W}=-\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}V_{ij}^{t}(x_i-x_j)\frac{(t_i-t_j)^{\mathrm{T}}}{2\sigma_t^2} \tag{17}$$

$$\frac{\partial V_E}{\partial W}=\frac{\partial V_J}{\partial W}+\frac{\partial V_M}{\partial W}-2\frac{\partial V_C}{\partial W}\quad \frac{\partial V_J}{\partial W}=\frac{\partial V_J}{\partial(t_i-t_j)}\frac{\partial(t_i-t_j)}{\partial W}=-\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}V_{ij}^{t}V_{ij}^{y}(x_i-x_j)\frac{(t_i-t_j)^{\mathrm{T}}}{2\sigma_t^2}$$

$$\frac{\partial V_M}{\partial W}=\frac{\partial V_M}{\partial(t_i-t_j)}\frac{\partial(t_i-t_j)}{\partial W}=-\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}V_{ij}^{y}\cdot\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}V_{ij}^{t}(x_i-x_j)\frac{(t_i-t_j)^{\mathrm{T}}}{2\sigma_t^2} \tag{18}$$

$$\frac{\partial V_C}{\partial W}=\frac{\partial V_C}{\partial(t_i-t_j)}\frac{\partial(t_i-t_j)}{\partial W}=-\frac{1}{N}\sum_{i=1}^{N}\left\{\frac{1}{N}\sum_{j=1}^{N}V_{ij}^{y}\cdot\frac{1}{N}V_{ij}^{t}(x_i-x_j)\frac{(t_i-t_j)^{T}}{2\sigma_t^2}\right\}$$

Here, we use $V_{ij}^{t}=G(t_i-t_j,2\sigma_t^2),V_{ij}^{u}=G(u_i-u_j,2\sigma_u^2)$ for more simple expressions. A regression model is built between T and Y in which the regression coefficient matrix θ is:

$$\theta=\left(TT^{T}\right)^{-1}TY^{T}\qquad T=W^{*T}X \tag{19}$$

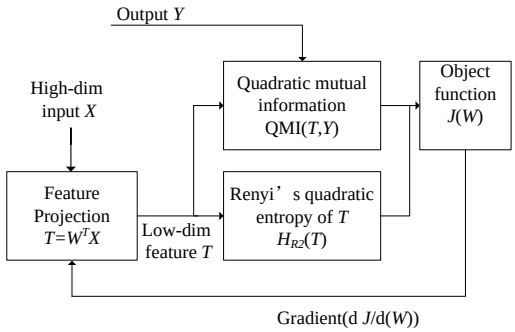

Figure 2. Framework of determining optimal features by using quadratic mutual information.

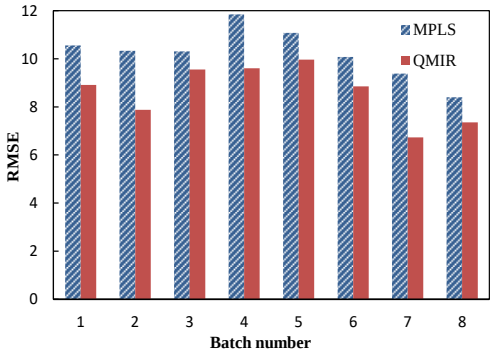

Figure 3. RMSEs of all test batches.

3.3. Prediction of quality variable

Offline modelling

- The training data of batch process is reprocessed as in section 2.3. Process variables are arranged as input X and quality variables are arranged as output Y.
- Normalize X and Y, respectively. Determine the kernel size σ_x, σ_t and σ_y according to Silverman's rule of thumb [12].
- Initialize W randomly and set the learning rate λ. Using gradient descent method to find the optimal W that minimize J. If a threshold of minimum difference of J is satisfied or the maximize number of iteration is reached, stop the iteration and get the optimal W^*. Else, update W by equation (15) to equation (18), and repeat the iteration.

Online prediction

- A new online data x_{new} is sampled. Normalize x_{new} with the means and variances of the training data.
- Calculate y_{new} through the regression model: $y_{new}=\theta^{T}W^{*\mathrm{T}}x_{new}$.
- Add back the means and the variance of y_{new} to get the final prediction value y_p.

We used root mean square error (RMSE) to evaluate the performance of QMIR quantitatively. Batch level RMSE is first defined as follows:

$$RMSE=\left(\frac{1}{K}\sum_{k=1}^{K}\left\|y_k-y_{kp}\right\|^2\right)^{\frac{1}{2}} \tag{20}$$

K is the whole sample time, y_k is the offline measured value of a certain quality variable at sample time k. y_{kp} is the prediction result. To demonstrate the prediction performances for every sample point, RMSE(k) is adopted:

$$RMSE(k) = \left(\frac{1}{I_{test}} \sum_{j=1}^{I_{test}} \left\| y_j^k - y_j^{kp} \right\|^2 \right)^{\frac{1}{2}} \tag{21}$$

I_{test} represents the amount of test batches. y_j^k stores the offline measurement for sample point k. y_j^{kp} is the corresponding prediction result of y_j^k by using the prediction method.

4. Experiments

The prediction performance of QMIR is explored in comparison with MPLS. Variable-wise unfolding is also performed on the cubic historical batch data for MPLS.

4.1. Datasets

The prediction methods are tested by production data of industrial E. coli fermentation process. The experiment data were collected from a plant located in Beijing. OD (optical density) is the hard-to-measure quality variable which indicates the E. coli concentration. In actual production, OD is usually measured offline. In our work, 8 process variables (PH, DO saturation, pressure, temperature, aeration power, added glucose, added nitrogen, and aeration rate) and a quality variable (OD) are selected. A collection of 28 normal batch data are aligned with the same sample time. For the training process, 20 batches are chosen randomly. The other 8 normal batches are ready for testing. The fermentation duration is 6.5 h and the sample interval is 10 minutes.

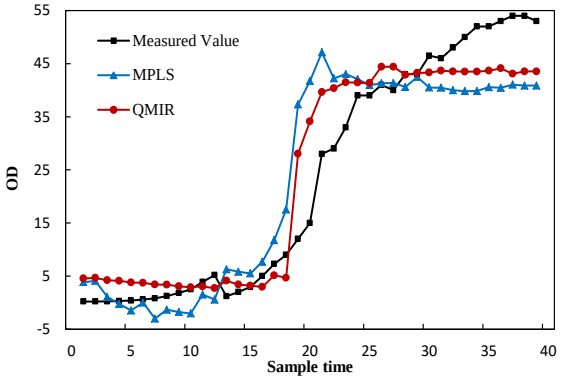

Figure 4. Prediction results of OD for a test batch.

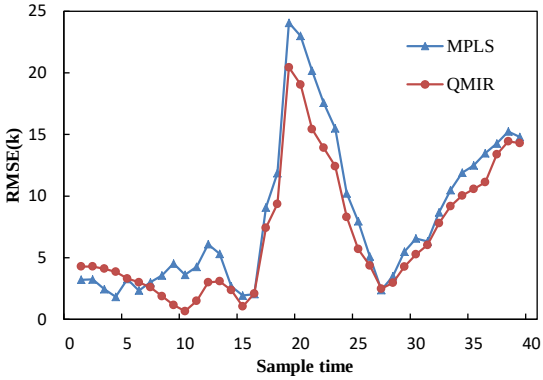

Figure 5. RMSE(k)s of all test batches.

4.2. Prediction results

The batch level RMSE values are shown in figure 3. As a result, the average RMSE values are 10.25 and 8.61 for the MPLS and QMIR, respectively. It can be seen that QMIR has smaller RMSE values of each test batch. It indicates that QMIR can give higher prediction accuracy than MPLS. It is because QMIR can extract features based on higher order statistics QMI, not just the second order statistics as in MPLS. In MPLS, the number of latent variable is set to 2 according to cross validation. In QMIR, r is set to 1, weight parameter α is set to 80 and learning rate λ is set to 0.5. The prediction results for OD of a single batch are shown in figure 4.We can see that both MPLS and QMIR follow the measured value of OD well before sample time 19. However, the prediction result of MPLS is accompanied with more fluctuations. An obvious deviation is appeared in MPLS Around sample time 22. Both of the two methods emerge performance decrease after sample time 33. But QMIR still performs better until the end of the whole batch run. We can see that QMIR has higher prediction accuracy than MPLS. RMSE(k) values are shown in figure 5. Obviously, QMIR has smaller RMSE(k) values and has better prediction accuracy than MPLS during most of the sample times, except at the beginning of the batch.

5. Conclusions

A new regression method based on the quadratic mutual information has been proposed. The results show that QMIR performs better than MPLS in terms of quality prediction on industrial E. coli fermentation dataset, the typical batch process data. Compared with conventional methods, the proposed QMIR method has two main properties. First, the QMI metric takes into account higher order statistics which can reveal the non-linear dependencies between the process variables and the quality variables. Second, the estimation of QMI is non-parametric, without Gaussian assumption of data distribution as MPLS.

Acknowledgments

This work was funded by the National Natural Science Foundation of China under grant 61640312, 61763037, and 61803005, the Natural Science Foundation of Beijing Municipality under grant 4172007 and 4192011, and the Beijing Municipal Commission of Education.

References

[1] Wang Y, Si Y, Huang B and Lou Z 2018 Survey on the theoretical research and engineering applications of multivariate statistics process monitoring algorithms: 2008-2017 *The Canadian Journal of Chemical Engineering* **96** pp.2073-85

[2] Han M, Ren W and Liu X 2015 Joint mutual information-based input variable selection for multivariate time series modeling *Engineering Applications of Artificial Intelligence* **37** pp.250-57

[3] Principe J.C., Fisher III J.W. and Xu D 2000 Information theoretic learning. In: Haykin, S. (Ed.) *Unsupervised Adaptive Filtering* (New York: John Wiley)

[4] Principe J.C. 2010 *Information Theoretic Learning. Renyi's Entropy and Kernel Perspectives* (Springer)

[5] Vera P.A., Estevez P.A. and Principe J.C. 2010 Linear projection method based on information theoretic learning *Proc. Int. Conf. on Artificial Neural Networks Part III* pp.178-87 (Berlin: Springer)

[6] Torkkola K 2003 Feature extraction by non-parametric mutual information maximization *Journal of Machine Learning Research* **3**(3) pp.1415-38

[7] Renyi A 1961 On measures of entropy and information *Proc. of the 4th Berkeley Symp.Math.Statist.Prob* **1** pp.547-61 (Berkeley University Press)

[8] Xu D and Principe J.C. 2001 Feature evaluation using quadratic mutual information *Proc. Int. Joint Conf. on Neural Networks* pp.459-63 (IEEE)

[9] Jiang B, Luo Y and Lu Q 2018 Maximized mutual information analysis based on stochastic representation for process monitoring *IEEE Transactions on Industrial Informatics* **15**(3) pp.1579-87

[10] He Y, Zhou L, Ge Z and Song Z 2018 Dynamic mutual information similarity based transient process identification and fault detection *The Canadian Journal of Chemical Engineering* **96** pp.1541-58

[11] Stubbs S, Zhang J and Morris J 2013 Multiway interval partial least squares for batch process performance monitoring *Industrial & Engineering Chemistry Research* **52**(35) pp.12399-407

[12] Silverman B.W. 1986 *Density Estimation for Statistics and Data Analysis* (London: Chapman and Hall)