

Block-wise RPLS와 상관관계를 이용한 적응 모델링

윤경우, 이영학*, 한종훈*
포항공과대학교 환경공학부, 포항공과대학교 화학공학과 및 아이시스텍(주)*

Adaptive modeling using block-wise RPLS and correlation

Kyong-U Yun, Young-Hak Lee*, Chonghun Han*
School of Environmental Science and Engineering, POSTECH.
Department of Chemical Engineering, POSTECH and ISYSTECH, Inc.*

서론

부분 최소 자승법 (Partial Least Squares, PLS) 은 서로 상관성이 큰 변수들이 포함된 경우에 공정을 모델링하고 모니터링 하는데 효과적으로 사용된다[1]. 하지만 공정 변화에 적응하는 온라인 모델을 만들기 위해 PLS를 이용하는 것은 많은 제약이 따른다. 그래서 많이 이용되고 있는 방법이 recursive PLS (RPLS) 이다[2]. 이 논문에서는 공정 변화에 잘 적응하는 모델을 만들기 위해 block-wise RPLS의 온라인 적응 방법인 moving window 방법을 이용했고[3], window내에서 사용된 모델 블록들 사이의 상관관계를 나타내는 지표를 정의한 다음, 그 지표를 통해 forgetting factors를 정하는 방법을 제시하였다.

이론

Partial Least Squares (PLS)

\mathbf{X} 와 \mathbf{Y} , 두 자료 행렬 사이의 선형 관계를 분석하는데 있어서 주로 사용되어 온 방법은 다중 선형 회귀법이다. 그러나 각 행렬 내에 서로 상관성이 큰 변수들이 포함된 경우, 그 예측력은 상당히 떨어지게 되는데 PLS는 이 문제를 효과적으로 처리하여 예측력이 뛰어나고 공정 잡음과 어느 정도의 센서 고장 시에도 강건한 모델을 제공할 수 있는 방법이다. PLS 모델을 만들기 위해 우선 모니터 되어야 할 응답 변수들, 즉 품질변수들의 데이터들로 \mathbf{Y} 를 구성하고, 예측자 변수들, 즉 온라인으로 측정되는 공정 변수들로 \mathbf{X} 를 구성한다. 그렇게 구성된 데이터들로부터 \mathbf{X} 와 \mathbf{Y} 의 상관관계를 이용하여 모델링을 한다.

$$\mathbf{X} = \mathbf{T}\mathbf{P}^T + \mathbf{E} \quad (1)$$

$$\mathbf{Y} = \mathbf{U}\mathbf{Q}^T + \mathbf{F} \quad (2)$$

$$\mathbf{U} = \mathbf{T}\mathbf{B} + \mathbf{G} \quad (3)$$

$$\mathbf{B} = (\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{U} \quad (4)$$

여기서 \mathbf{E} , \mathbf{F} , \mathbf{G} 는 잔차 행렬 (residual matrix) 들이고 \mathbf{T} 는 \mathbf{Y} 를 고려한 \mathbf{X} 축소 공간에서의 이력을 나타내는 score matrix, \mathbf{U} 는 \mathbf{X} 를 고려한 \mathbf{Y} 축소 공간에서의 이력을 나타내는 score matrix이다. 그리고 \mathbf{P} 와 \mathbf{Q} 는 \mathbf{X} 와 \mathbf{Y} 에 대한 loading matrix이고, \mathbf{B} 는 \mathbf{X} 와 \mathbf{Y} 사이를 내적 관계로 표현하는 내적 모델 계수들의 diagonal matrix이다.

Recursive Partial Least Squares (RPLS)

PLS 회귀 계수 행렬 (regression coefficient matrix) 은 다음과 같다.

$$\mathbf{C}^{PLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} = \mathbf{W}(\mathbf{P}^T\mathbf{W})^{-1}\mathbf{B}\mathbf{Q}^T \quad (5)$$

여기서 \mathbf{W} 는 \mathbf{X} 와 \mathbf{Y} 사이의 관계를 나타내는 weighting matrix이다. 새로운 데이터 $\{\mathbf{X}_1, \mathbf{Y}_1\}$ 을 이용해서 기존에 있던 PLS 모델을 갱신할 때 사용되는 자료 행렬은 다음과 같다.

$$\mathbf{X}_{new} = \begin{bmatrix} \mathbf{X} \\ \mathbf{X}_1 \end{bmatrix} \quad \text{and} \quad \mathbf{Y}_{new} = \begin{bmatrix} \mathbf{Y} \\ \mathbf{Y}_1 \end{bmatrix} \quad (6)$$

(6)식의 행렬을 가지고 PLS 회귀 계수 행렬을 구하면 다음과 같다.

$$C_{new}^{PLS} = \left(\begin{bmatrix} X \\ X_1 \end{bmatrix}^T \begin{bmatrix} X \\ X_1 \end{bmatrix} \right)^+ \begin{bmatrix} X \\ X_1 \end{bmatrix}^T \begin{bmatrix} Y \\ Y_1 \end{bmatrix} \quad (7)$$

그리고

$$X^T X = P^T T P^T = P P^T \quad (8)$$

$$X^T Y = P^T T B Q^T + P^T T F = P B Q^T \quad (9)$$

그러므로 위의 (7)식은 다음과 같이 된다.

$$C_{new}^{PLS} = \left(\begin{bmatrix} P^T \\ X_1 \end{bmatrix}^T \begin{bmatrix} P^T \\ X_1 \end{bmatrix} \right)^+ \begin{bmatrix} P^T \\ X_1 \end{bmatrix}^T \begin{bmatrix} B Q^T \\ Y_1 \end{bmatrix} \quad (10)$$

(7)식과 (10)식의 비교를 통해, 새로운 데이터를 이용한 RPLS의 모델 갱신은 다음 자료 행렬에 의해 수행된다.

$$X_{new} = \begin{bmatrix} P^T \\ X_1 \end{bmatrix} \quad \text{and} \quad Y_{new} = \begin{bmatrix} B Q^T \\ Y_1 \end{bmatrix} \quad (11)$$

즉, PLS로 모델을 갱신할 때는 지난 데이터와 새로운 데이터를 이용하지만, RPLS는 지난 PLS 모델과 새로운 데이터를 이용한다. 그래서 새로 구성된 자료 행렬의 크기가 작기 때문에 PLS 모델을 만드는데 부하가 적고 계산 시간이 단축되는 이점이 있다.

Block-wise RPLS

새로운 데이터 블록이 들어 왔을 때, 기존의 PLS 모델과 새로운 블록을 이용해 만든 PLS 모델을 이용해 모델을 갱신하는 방법이다. 즉, sub-model 두 개를 가지고 자료 행렬을 구성하고 그것을 이용해 하나의 모델을 만든다. 새로운 자료 행렬이 블록을 압축한 축소 모델이고, 블록 자체는 많은 샘플들로 구성되어 있기 때문에 RPLS보다 모델 갱신을 자주 하지 않아도 된다.

$$X_{new} = \begin{bmatrix} P^T \\ P_1^T \end{bmatrix} \quad \text{and} \quad Y_{new} = \begin{bmatrix} B Q^T \\ B_1 Q_1^T \end{bmatrix} \quad (12)$$

Block-wise RPLS with a moving window and forgetting factors

Window 크기를 일정하게 유지하면서 새로운 블록과 지난 블록들의 모델에 forgetting factors를 지수함수 꼴로 감소하면서 곱해 준다. Forgetting factors가 작을수록 지난 데이터를 빨리 잊는다. 다음은 새로운 블록이 들어 왔을 때의 자료 행렬이다.

$$\begin{bmatrix} P_S^T \\ \lambda P_{S-1}^T \\ \cdot \\ \cdot \\ \lambda^{w-1} P_{S-w+1}^T \end{bmatrix}, \begin{bmatrix} B_S Q_S^T \\ \lambda B_{S-1} Q_{S-1}^T \\ \cdot \\ \cdot \\ \lambda^{w-1} B_{S-w+1} Q_{S-w+1}^T \end{bmatrix} \quad (13)$$

Input matrix

Output matrix

여기서, λ : forgetting factor
 S : 데이터의 현재의 블록

W : window에서의 블록들의 수

이 방법은 온라인 모델 갱신을 위한 방법이고 일정 기간의 데이터만을 이용하므로 부하나 계산 속도 측면에서 위의 방법 이상의 장점을 가지고 있다.

방법론 및 사례연구

대상공정

대상공정은 정유 공장의 원유를 가열하는 공정이다. 주요 unit는 fired heater와 air pre heater이고, 25개의 공정변수와 1개의 품질 변수 (NO_x) 로 구성되어 있다. 이 공정은 크게 두 개의 조업모드로 운전되고 있고, 각 조업 모드에 해당하는 데이터인 2001년 11월 27일부터 2002년 1월 8일까지와 2002년 2월 5일부터 2002년 4월 10일까지의 2분 샘플 데이터 중 정상적인 조업시의 데이터를 사용해서 모델을 만들고 검증했고, 한 블록은 1주일 분량의 샘플수를 이용했다.

상관관계를 통한 forgetting factor의 결정 방법

1. 모델링에 필요한 window 크기를 두 블록으로 유지하면서 새로운 블록을 예측한다. 모델링에 사용된 두 블록 각각의 sub-model을 PLS로 모델링 한다.
2. 두 PLS 모델에 의해 구해진 loading matrices와 결정계수를 이용해 다음과 같이 구한다. 이 식에서 사용된 곱은 element-by-element multiplication이다.

$$\mathbf{P}_1^T (r_1 \text{ by } n) \times \text{expanded R2Y}_{(r_1 \text{ by } n)} = \mathbf{A}_{(r_1 \text{ by } n)} = [a_{jk}] \quad (14)$$

$$\mathbf{P}_2^T (r_2 \text{ by } n) \times \text{expanded R2Y}_{(r_2 \text{ by } n)} = \mathbf{B}_{(r_2 \text{ by } n)} = [b_{jk}] \quad (15)$$

여기서,

1, 2 : 모델에 사용된 두 블록

r₁, r₂ : 각 블록의 모델에 대한 latent variables의 수

n : 변수의 수

expanded R2Y : R2Y column vector를 변수 수만큼 확장한 행렬

3. (14)와 (15)식은 각 변수별로 압축된 정보인 다음과 같은 벡터로 변형된다.

$$\left[\sum_{j=1}^{r_1} a_{j1} \quad \sum_{j=1}^{r_1} a_{j2} \quad \sum_{j=1}^{r_1} a_{j3} \quad \dots \quad \sum_{j=1}^{r_1} a_{jn} \right] = \mathbf{AA}_{(1 \text{ by } n)} \quad (16)$$

$$\left[\sum_{j=1}^{r_2} b_{j1} \quad \sum_{j=1}^{r_2} b_{j2} \quad \sum_{j=1}^{r_2} b_{j3} \quad \dots \quad \sum_{j=1}^{r_2} b_{jn} \right] = \mathbf{BB}_{(1 \text{ by } n)} \quad (17)$$

4. (16)과 (17)식의 elements은 각 블록의 변수 정보를 나타내는 값이다. 이 블록들에 대한 상관성 정보를 얻기 위해 두 벡터의 상관계수 (r) 를 구한다. 이 상관계수는 모델에 사용된 두 블록의 상관관계를 나타내므로 두 블록으로 모델링할 때, 이 값을 forgetting factor로 사용하면 된다. 즉, 두 블록 중 과거 블록에 상관계수를 곱하고 현재에 가까운 블록에 1을 곱해서 자료 행렬을 만든 다음, 그 자료 행렬을 이용해서 PLS 모델을 수행한다. 상관계수가 작으면 그만큼 두 블록의 상관성이 작다는 것이므로, 예측하는 블록도 상관성이 떨어질 것이다. 그래서 모델 블록 중 최근 블록에 weight를 더 크게 주어야 하므로 상대적으로 과거 블록에 weight를 작게 준다. 그리고 두 블록이 상관성이 크다는 것은 그만큼 다음 블록도 상관성이 클 가능성이 높으므로 두 블록사이의 weight 차이를 작게 해서 최대한 정보를 많이 가지고 예측한다. 그러므로 상관계수를 forgetting factor로 사용해서 모델을 만든다.

결과 및 토론

1. 2001년 11월 27일 ~ 2002년 1월 8일 : 총 5개의 블록을 통한 검증

표 1은 모델 블록의 forgetting factors에 따른 예측 블록의 RMSE (root mean square error)

값을 나타낸 것이다. Min.은 각 블록 예측 후, 에러가 최소가 되는 λ 값에서의 RMSE 값을 나타내었고, Accuracy는 $100 \times \text{RMSE of Min.} / \text{RMSE of } (\lambda = r)$ 을 나타내므로 상관계수 값을 forgetting factor로 사용했을 때 얼마나 좋은지를 판단할 수 있는 근거가 된다.

Number of Prediction Blocks	r (model blocks)	$\lambda=1$	$\lambda= r$	Min.	Accuracy(%)
3(model blocks:1,2)	0.0844	17.5654	11.1224	11.1224	100
4(model blocks:2,3)	0.5158	12.2814	12.9181	11.7056	91
5(model blocks:3,4)	0.1382	12.0109	11.7261	11.4144	97
Total blocks		14.1849	11.9455	11.4166	96

<표 1. 예측 블록에 대한 RMSE>

2. 2002년 2월 5일 ~ 2002년 4월 10일 : 총 8개의 블록을 통한 검증

표 2도 역시 표1과 같은 RMSE 값을 나타낸다.

Number of Prediction Blocks	r (model blocks)	$\lambda=1$	$\lambda= r$	Min.	Accuracy(%)
3(model blocks:1,2)	0.5072	8.0975	8.0230	8.0230	100
4(model blocks:2,3)	0.7920	6.8223	6.5220	6.4233	98
5(model blocks:3,4)	0.6914	6.3937	6.5774	6.1595	94
6(model blocks:4,5)	0.6696	8.5597	8.4139	8.0352	95
7(model blocks:5,6)	0.1220	12.3775	8.7908	8.4313	96
8(model blocks:6,7)	0.4189	12.0530	12.5536	12.0530	96
Total blocks		9.3518	8.7162	8.4112	97

<표 2. 예측 블록에 대한 RMSE>

위의 표1과 2에서 보면, 모델에 사용된 두 블록의 정보를 그대로 이용한 $\lambda=1$ 은 본 저자가 제안한 $\lambda= r$ 일 때보다 예측력이 떨어짐을 알 수 있었다. 그리고 r값을 forgetting factor로 사용했을 때 실제로 얻을 수 있는 최소 에러에 상당히 근접한 값을 가짐을 확인할 수 있었다. 그러므로 모델 블록의 상관성을 나타내는 지표가 중요하고 그 지표를 forgetting factor로 사용해 모델을 만드는 것이 효과적이라고 할 수 있다. 실제로 이 지표는 미래의 예측에 대한 최선의 모델을 제시할 수 있다는데 그 의의가 있다고 할 수 있다. 그리고 별다른 공정 지식 없이 동일한 샘플수의 블록의 상관성만으로도 좋은 모델을 만들 수 있고, RPLS 방법의 장점을 최대한 이용할 수 있다는 점에서 온라인 모델링을 위한 좋은 방법이라고 할 수 있다.

감사의 글

본 연구는 두뇌한국 21 사업에 의해 지원되었으며 이에 감사를 드립니다.

참고문헌

1. Paul Geladi and Bruce R. Kowalski. : "Partial Least-Squares Regression : A tutorial", *Analytica Chimica Acta.*, 185, 1-17(1986).
2. Helland, K., Berntsen, H. E., Borgen, O. S. and Martens, H. : "Recursive algorithm for partial least squares regression", *Chemometrics Intell. Lab. Syst.*, 14, 129-137(1992).
3. Qin, S. J. : "Recursive PLS algorithms for adaptive data modeling", *Computers chem. Engng.*, 22, 503-514(1998).