

개선된 Support Vector Machine을 이용한 중합 공정 모델링

이동언, 송상옥, 이창준, 윤인섭
서울대학교 응용화학부

Modified Support Vector Machine in modeling of polymerization process

Dong Eon Lee, Sang-Oak Song, Chang Jun Lee, En Sup Yoon
School of Chemical Engineering, Seoul National University

서론

Support Vector Machine(SVM)은 최근 주목받고 있는 통계학습방법으로써 일반적으로 신경망 학습 방법이 채용하고 있는 임피리컬 리스크 최소화 방법(Empirical Risk Minimization)보다 구조적 에러 최소화 방법(Structural Risk Minimization principle)을 채용하고 있어 통계적 학습방법의 기본 목적인 일반화 성능이 뛰어나다. 또한 학습데이터의 차원의 제약이 없어 변수가 많은 화학공정의 모델링과 같은 분야에 적용하여 신뢰성 있는 결과를 얻을 수 있다. 본 연구에서는 이런 SVM방법을 변수가 많고 그 관계가 매우 비선형적인 중합 공정에 적용하기 위해서 학습시 사용되는 학습데이터들에 가중치를 부여하는 Locally Weighted Regression(LWR)방법의 기본 개념을 차용하여 SVM을 개선하고 그 성능을 사례 연구를 통해 확인하였다.

주요어 : Support Vector Machine, Empirical Risk Minimization, Locally Weighted Regression, 중합공정, 모델링

본론

1. Support Vector Machine

주어진 데이터 세트, $G = \{(x_i, d_i)\}_i^n$ (x : 입력 벡터, d : 목표값, n : 데이터 개수)가 주어질때, SVM은 다음의 특성에 의해 내재 함수를 근사한다. (i) SVM은 선형 회기법이다. (ii) SVM은 ϵ -인센서티브 에러 함수(ϵ -insensitive loss function)이라는 개념의 관점에서 에러 최소화(risk minimization) 문제로 선형회기법은 수행한다. (iii) SVM은 부등식 $\|w\| \leq$ 상수로 정의되는 요소를 포함하는 SRM(Structural Risk Minimization) 방법에 기반하여 에러를 최소화 시키는 방법으로 학습한다. 선형함수는 입력 벡터들이 더 높은 차원으로 맵핑된 feature 공간에서 다음의 식 (1)과 같이 만들어진다.

$$y = f(x) = w\phi(x) + b \quad (1)$$

이때, $\phi(x)$ 는 입력 공간의 벡터 x 가 높은 차원의 feature공간으로 비선형 맵핑된 공간이다. 앞서 언급된 SVM의 특성 (ii), (iii)에 의해서 계수 w 와 b 를 결정한다. 주어진 학습 데이터들은 표현하는 함수를 나타내기 위해 다음의 에러함수를 사용한다.

$$R(C) = C \frac{1}{n} \sum_{i=1}^n L_{\epsilon}(d_i, y_i) + \frac{1}{2} \|w\|^2 \quad (2)$$

$$L_{\epsilon}(d, y) = \begin{cases} |d - y| - \epsilon, & |d - y| \geq \epsilon \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

첫 항목, 임피리컬(empirical) 에러 $C(1/n) \sum_{i=1}^n L_{\epsilon}(d_i, y_i)$ 은 ϵ -인센서티브 에러 함수(ϵ -insensitive loss function) (3)에 의해 구해진다. 두 번째 항목 $(1/2)\|w\|^2$ 는 조정항(regularized term) 이다. ϵ 는 SVM의 튜브 크기이고, C 는 임피리컬 에러와 조정항의 기여도를 결정하는 상수이다.(사용자에 의해 미리 결정되는 상수) 위의 식은 슬랙 변수(slack variable) ζ, ζ^* 을 도입하여 다음과 같은 식으로 나타낼 수 있다.

$$\text{Minimize } R(w, \zeta, \zeta^*) = \frac{1}{2}\|w\|^2 + C \frac{1}{n} \sum_{i=1}^n (\zeta_i + \zeta_i^*)$$

Subject to

$$\begin{aligned} d_i - w\phi(x_i) - b_i &\leq \epsilon + \zeta_i \\ w\phi(x_i) + b_i - d_i &\leq \epsilon + \zeta_i^* \\ \zeta_i, \zeta_i^* &\geq 0 \end{aligned} \quad (4)$$

위의 식에서 라그랑지 승수를 적용하여 다음의 식을 최대화하는 최적화 문제로 귀결시킨다.

$$R(a_i, a_i^*) = \sum_{i=1}^n d_i(a_i - a_i^*) - \epsilon \sum_{i=1}^n (a_i + a_i^*) - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (a_i - a_i^*)(a_j - a_j^*) K(x_i, x_j) \quad (5)$$

제약조건은 다음과 같다.

$$\sum_{i=1}^n (a_i - a_i^*) = 0$$

$$0 \leq a_i \leq C, \quad i = 1, 2, \dots, n$$

$$0 \leq a_i^* \leq C, \quad i = 1, 2, \dots, n$$

식 (5)에서 $K(x_i, x_j)$ 은 두개의 입력벡터 x_i, x_j 가 feature 공간으로 맵핑된 점 $\phi(x_i), \phi(x_j)$ 들간의 커널 함수로 표현 되는 내적이다.

2. Locally Weighted Regression

Locally Weighted Regression 방법은 학습데이터를 새로운 데이터를 이용한 예측을 수행할 때 마다 다시 이용하는 non-parametric 접근 방식인 메모리 기반(memory-based) 학습 방법 중 하나다. LWR는 예측할 입력 데이터 x_0 에 가까운 학습 데이터들에 더 많은 가중치를 줌으로써 근사를 수행하는 학습 방법이다. 보통 많이 쓰이는 가중치 방법은 트리큐빅(tricubic) 함수(6)나 가우시안(Gaussian)함수(7)를 많이 쓴다.

$$w_j(x_0) = W(\rho(x_j, x_0)/d(x_0)) \quad (6)$$

$$w_j(x_0) = \exp(-k\rho(x_j, x_0)) \quad (7)$$

이 때, $W(u) = (1 - u^3)^3$

$\rho(x_j, x_0)$: x_j 과 x_0 간의 거리

$d(x_0)$: x_0 과 학습 데이터 포인트간 거리중 가장 큰 값

3. Modified Support Vector Regression

식(4)에서처럼 임피리컬 에러 함수는 실제값과 학습을 통해 예측된 값 사이의 모든 ϵ -인 센서티브 에러에 대해서 항상 같은 가중치를 가진다. 이 C값이 커질수록 임피리컬 에러의 최적화의 기여도가 더 커진다. 따라서 새롭게 예측하고자 하는 입력값에 가까운 학습 데이터에 상대적으로 더 멀리 위치한 것보다 더 큰 C값을 부여함으로써 예측 성능을 더 높인다. 본 연구에서는 C 값의 가중치를 트리큐빅 함수를 사용한다. 따라서 최소화하는 조정 위험 함수(regularized risk function)(4)는 다음과 같이 표현될 수 있다.

$$R_{mSVM}(w, \zeta_i, \zeta_i^*) = \frac{1}{2} \|w\|^2 + \sum_{i=1}^n C_i (\zeta_i + \zeta_i^*) \quad (8)$$

$$C_i = C w_i(x_0) \quad (9)$$

이 식을 라그랑지 승수를 사용하여 실제 최적화 문제의 해를 찾는 식의 형태는 다음과 같이 나타난다.

$$R(a_i, a_i^*) = \sum_{i=1}^n d_i (a_i - a_i^*) - \epsilon \sum_{i=1}^n (a_i + a_i^*) - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (a_i - a_i^*) (a_j - a_j^*) K(x_i, x_j) \quad (10)$$

제약조건

$$\sum_{i=1}^n (a_i - a_i^*) = 0$$

$$0 \leq a_i \leq C_i \quad i = 1, 2, \dots, n$$

$$0 \leq a_i^* \leq C_i \quad i = 1, 2, \dots, n$$

4. 사례연구

본 연구에서 제안된 방법의 성능 개선을 확인하기 위해서 임의의 함수의 데이터를 이용하여 회귀를 수행해 보았다.

x	1.0	3.0	4.0	5.6	10.2	11.0	11.5	12.7
y	-1.6	-1.8	-1.0	1.2	6.8	10.0	10.0	10.0

표1. Regression Data

	standard SVM	Modified SVM
root mean square error	0.3526	0.3216

표2. 결과 비교(1)

위의 결과로 본 연구에서 제안된 방법의 일차적인 성능 개선을 확인하고 이 방법을 폴리머 파일롯 플랜트 데이터(Lyle H. Unger, 1994)에 적용하였다. 이 데이터는 폴리머 공정에서 온도, 유량 등과 같은 10개의 차원을 가지는 입력 변수와 4개의 출력 변수를 가진다. 학습 후 실제 출력값과 예측값의 비교는 아래 표3, 그림 1과 같다.

	standard SVM				modified SVM			
	y_1	y_2	y_3	y_4	y_1	y_2	y_3	y_4
RMSe	0.0451	0.0194	0.0293	0.0260	0.0226	0.0188	0.0269	0.0224

표3. 결과 비교(2)

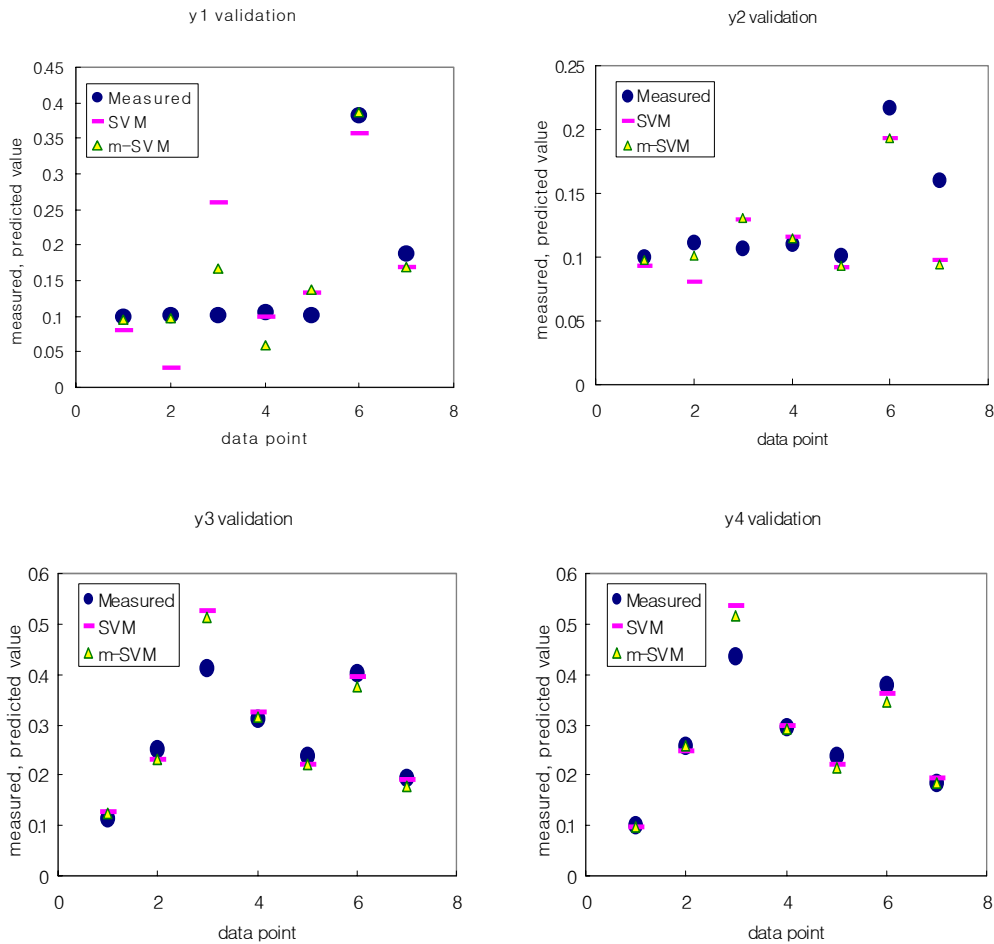


그림 1. Measured vs Predicted Value comparison

결론

본 연구에서는 학습데이터의 차원이 매우 높고 비선형성이 매우 큰 중합공정의 모델링을 위해서 개선된 SVM 방법을 제안하였다. 학습을 통해 예측된 출력값과 실제값의 차이를 각각의 데이터를 동등한 가중치로 고려하여 최소화하는 기본적인 SVM 방법을 예측하고자 하는 데이터와 비슷한(입력데이터 공간상에서 데이터가 거리가 가까운) 데이터에 더 높은 가중치를 부여하여 학습하는 LWR방법의 개념을 차용하여 개선하였다. 이는 비선형성이 강하고 입력 변수의 변화에 따른 출력변수의 변화가 심한 화학공정, 특히 중합공정 모델링에 적용하여 뛰어난 성능을 나타낼 수 있다. 이렇게 개선된 SVM을 임의의 함수에서 만든 예제데이터의 학습에 적용하여 성능개선을 확인하였고 이를 Dupont 사의 고분자 파이프 플랜트 데이터에 적용하였다.

참고문헌

1. W. Cherkassky and F. Mulier, "Learning from data", John Wiley & Sons, 1998
2. N. Cristianini and J. Shawe-Taylor, " An Introduction to Support Vector Machine", Cambridge univ. press, 2001
3. V. Kecman, " Learning and Soft Computing", MIT Press, 2001
4. Francis E.H. Tay and L.J. Cao, *Neurocomputing*, in press
5. R. J. Cleveland and J. M. McArthur, *J. Ameriacn Statistics Asso.* **83**, 596 (1988)