## 탈 실험 단원자 증착 공정 설계

### [공정 설명] 단원자 증착법

증착 공정 (Deposition process)은 반도체 소자 재조를 위한 단위공정들 중 하나로 반도체 산업에 핵심적인 기술이다. 보통 증착 공정은 단결정의 규소기판에 반도체, 절연체, 도체등의 재료를 박막의 형태로 증착시켜 집적회로의 미세구조 형성하는데 기여한다. 일반적으로 증착 공정은 기판위 회로 규모에 가장 큰 영향을 주므로, 소자의 집적도에 가장 큰 기여를 한다. 따라서 고집적 박막 형성을 위한 효율적이고 섬세한 증착 공정 개발이 차세대 반도체 개발을 위해 핵심적이다.

기존의 반도체 공정에서의 증착법은 전통적으로 물리기상증착법 (Pysical vapor deposition, PVD)과 화학기상증착법 (Chemical vapor deposition, CVD)에 의존한다. PVD는 도포에 사용할 물질들을 높은 에너지나 열에 노출시켜 입자를 기판 표면에 방사시켜 증착하는 방식이다. 이 기술은 높은 생산성을 갖고 준수한 박막 두께 조절력을 갖는다. CVD는 기질 표면에서 반응기체를 증착시키는 방법으로, 박막 두께 조절에 우수하고 적절한 생산성을 갖는다. 하지만 CVD와 PVD는 모두 200 Å 이하의 박막 증착에는 사용할 수 없는데, 반도체 박막 증착 두께가 높은 집적도의 소자 개발에 필수적임으로 새로운 기술이 요구되는 실정이다. 단원자 증착법 (Atomic layer deposition, ALD)은 소자 미세화와 집적도 향상을위한 가장 이상적인 무기 및 금속 박막 증착 기술이다. ALD은 원자 단위에서 단원자층을 증착하는 기술로써 200 Å 이하의 박막 형성이 가능하여 고용량 반도체 소자 개발에 핵심적이다. 또한, 3차원 구조의 반도체 제조 공정에 응용할 수 있어 반도체 생산에 큰 각광을 받고있다.

ALD은 기판과의 전구체 물질 (Precursor) 사이의 선택적 표면 화학반응을 통한 반응물 분자의 증착을 유도한다. 이를 위해 밀도범함수 이론 (Density functional theory, DFT), kinetic Monte Carlo 방법 등이 기판-전구체 사이의 반응 예측과 해석에 이용된다. 하지만, 수많은 전구체 물질, 반응물, 그리고 기판 사이의 현상 실험은 비용과 시간의 한계점이 명확히 존재한다. 따라서 비효율적인 단원자 증착법을 탈피해 실용화를 위한 고성능 물질 탐색을 위해 혁신이 요구된다.

서포트 벡터 회귀 (Support vector regression)은 고차원 데이터 식별을 위한 서포트 벡터 머신 (Support vector machine) 으로부터 발달한 인공지능 알고리즘이다. 서포트 벡터 회귀는 단순한 하이퍼파라미터 구조를 통해 신속한 모델 생성이 가능한 특징을 지녔다. 오늘날 서포트 벡터 회귀는 다양한 모델과 결합된 hybrid 형태로 화학공정의 많은 복잡계 문제를 해결하고 있다.

본 실습에서는 탈실험화된 단원자 증착 공정 설계를 위해 서포트벡터머신 기반의 전구체 성능 예측 모델을 개발한다.

### [문제] ## 추가 필요

### [방법] SVM 예제

#### Q1. SVM을 활용하기 위한 라이브러리를 조사하라. SVM을 regression 형태로 사용하기위한 명령어는 무엇인가?

**A1.** SVM 활용을 위한 라이브러리로 ‘e1071’이 있다. SVM은 input데이터 형식에 따라 자동으로 classification/regression을 설정한다. 따라서 ‘svm()’함수를 불러오고, 데이터를 독립/종속변수를 설정한다.

|  |
| --- |
| Library(‘e1071’)  Model\_SVM <- e1071::svm(Data[,:1:x], Data[,x], gamma = , cost = ) |

#### Q2. SVM의 학습은 하이퍼파라미터에 따라 다른 성능의 예측모델을 생성한다. SVM의 하이퍼파라미터는 무엇인가? 하이퍼파라미터가 의미하는 것은 무엇인가?

**A2.** SVM의 하이퍼파라미터에는 ‘gmma’와 ‘cost’가 있다. ‘gamma’는 데이터 샘플의 영향력 행사 거리 (분산)을 가정한다. ‘gamma’ 가 클수록 데이터가 다른 데이터에 더 큰 영향을 줄 수 있다고 판단한다. ‘cost’는 데이터 샘플의 최대/최소 허용 클래스 수를 가정한다. ‘cost’가 크면 이상치의 가능성을 매우 작게 판단한다.

### [응용] SVM 기반 고성능 전구체 물질 예측 모델 개발

예제는 R 4. 0. 2 프로그래밍 언어를 기준으로 Rstudio 개발환경에서 작성되었다. 예제 실습을 위해 Anaconda 프로그램을 아래 URL로부터 다운로드할 수 있다.

https://www.anaconda.com/

#### Q3. 데이터를 Jupyter환경으로부터 불러오고 데이터를 확인하라.

**A3.** 다음과 같은 code를 사용하여 불러올 수 있다.

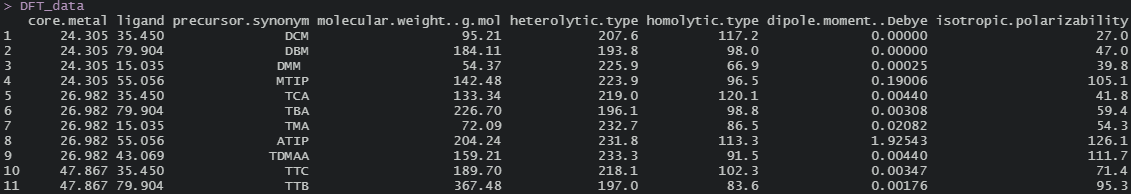
|  |
| --- |
| DFT\_data <- read.csv(file.choose(), header = T) |

‘read.csv’ 함수를 사용하여 데이터 파일이 저장된 장소를 직접 찾아 ‘DFT\_data’ 이름으로 데이터를 불러온다. 해당 데이터는 column 이름이 이미 존재하는 데이터로 ‘header = T’ arg를 통해 이를 밝힌다.

|  |
| --- |
| str(DFT\_data) |

‘str()’ 함수를 사용하여 데이터의 형태를 확인할 수 있다.

|  |
| --- |
| DFT\_data |



불러온 데이터를 확인해본다.

#### Q4. 인공지능 학습을 위해서 데이터의 단위를 무시한 상대적 영향력을 파악할 필요가 있다. DFT데이터를 정규화하라.

**A4.** 다음과 같은 절차를 통해 정규화한다.

|  |
| --- |
| DFT\_wo\_NA <- data.frame(DFT\_data[,1:2], DFT\_data[,4:28]) |

데이터의 3번째 열은 전구체 물질의 이름이므로 이를 제거한다. 추출된 데이터는 ‘DFT\_wo\_NA’로 명명한다.

|  |
| --- |
| DFT\_wo\_NA\_symbol <- data.frame(DFT\_wo\_NA)  DFT\_wo\_NA\_symbol\_scale <- data.frame(scale(DFT\_wo\_NA\_symbol)) |

데이터를 ‘DFT\_wo\_NA’로 명명하고 ‘scale()’ 함수를 사용해서 정규화한다. 정규화된 데이터는 ‘DFT\_wo\_NA\_symbol\_scale’로 명명한다

#### Q5. 9개의 종속변수 각각 예측하는 SVM 모델을 생성할 것이다. 데이터를 이에 맞춰 나누어라.

**A5.** 다음과 같은 절차를 통해 데이터를 나눌 수 있다.

|  |
| --- |
| DFT\_EadH <- data.frame(DFT\_wo\_NA\_symbol\_scale[,1:9], DFT\_wo\_NA\_symbol\_scale[,10])  DFT\_EadOH <- data.frame(DFT\_wo\_NA\_symbol\_scale[,1:9], DFT\_wo\_NA\_symbol\_scale[,11])  DFT\_EadNH2 <- data.frame(DFT\_wo\_NA\_symbol\_scale[,1:9], DFT\_wo\_NA\_symbol\_scale[,12]) |

데이터의 종속변수인 3가지 표면에서의 adsorption 에너지 (Ead\_H, Ead\_OH, Ead\_NH2)에 대해 각각 독립변수-종속변수 관계를 규명한 데이터를 생성한다

|  |
| --- |
| DFT\_EaH <- data.frame(DFT\_wo\_NA\_symbol\_scale[,1:9], DFT\_wo\_NA\_symbol\_scale[,13])  DFT\_EaOH <- data.frame(DFT\_wo\_NA\_symbol\_scale[,1:9], DFT\_wo\_NA\_symbol\_scale[,14])  DFT\_EaNH2 <- data.frame(DFT\_wo\_NA\_symbol\_scale[,1:9], DFT\_wo\_NA\_symbol\_scale[,15]) |

데이터의 종속변수인 3가지 표면에서의 active barrier 에너지 (Ea\_H, Ea\_OH, Ea\_NH2)에 대해 각각 독립변수-종속변수 관계를 규명한 데이터를 생성한다

|  |
| --- |
| DFT\_dEH <- data.frame(DFT\_wo\_NA\_symbol\_scale[,1:9], DFT\_wo\_NA\_symbol\_scale[,16])  DFT\_dEOH <- data.frame(DFT\_wo\_NA\_symbol\_scale[,1:9], DFT\_wo\_NA\_symbol\_scale[,17])  DFT\_dENH2 <- data.frame(DFT\_wo\_NA\_symbol\_scale[,1:9], DFT\_wo\_NA\_symbol\_scale[,18]) |

데이터의 종속변수인 3가지 표면에서의 dE (dE\_H, dE\_OH, dE\_NH2)에 대해 각각 독립변수-종속변수 관계를 규명한 데이터를 생성한다

#### Q6. SVM 학습을 위해 데이터를 학습/검증 데이터로 나누어야 한다. 학습/검증 데이터를 각각 70%와 30%로 나누어라.

**A6.** 다음과 같은 절차를 통해 데이터를 나눌 수 있다.

|  |
| --- |
| smp\_size <- floor(0.7 \* nrow(DFT\_EadH))  set.seed(1)  train\_ind <- sample(seq\_len(nrow(DFT\_wo\_NA)), size = smp\_size) |

‘floor()’ 함수를 사용하여 ‘WGSR\_scale’ 데이터의 행 개수의 70%에 해당되는 값을 생성한다. 이 값은 ‘smp\_size’로 명명하여, 후에 무작위 데이터 추출에 사용한다.

‘sample()’ 함수를 사용해서 ‘WGSR\_scale’ 데이터로부터 행 번호를 무작위 추출을 진행한다. ‘size’ arg를 ‘smp\_size’로 설정하여 총 70% 데이터를 추출한다. ‘seq\_len()’ 함수를 통해 1부터 임의의 지정된 숫자까지 순차 데이터를 생성할 수 있다.

|  |
| --- |
| SVM\_train <- DFT\_EadH[train\_ind,]  SVM\_test <- DFT\_EadH[-train\_ind,]  SVM\_train <- DFT\_EadOH[train\_ind,]  SVM\_test <- DFT\_EadOH[-train\_ind,]  SVM\_train <- DFT\_EadNH2[train\_ind,]  SVM\_test <- DFT\_EadNH2[-train\_ind,] |

SVR의 학습과 검증을 위해 세가지 표면에서의 adsorption 에너지 대한 데이터를 생성한다.

|  |
| --- |
| SVM\_train <- DFT\_EaH[train\_ind,]  SVM\_test <- DFT\_EaH[-train\_ind,]  SVM\_train <- DFT\_EaOH[train\_ind,]  SVM\_test <- DFT\_EaOH[-train\_ind,]  SVM\_train <- DFT\_EaNH2[train\_ind,]  SVM\_test <- DFT\_EaNH2[-train\_ind,] |

SVR의 학습과 검증을 위해 세가지 표면에서의 active barrier 에너지 대한 데이터를 생성한다.

|  |
| --- |
| SVM\_train <- DFT\_dEH[train\_ind,]  SVM\_test <- DFT\_dEH[-train\_ind,]  SVM\_train <- DFT\_dEOH[train\_ind,]  SVM\_test <- DFT\_dEOH[-train\_ind,]  SVM\_train <- DFT\_dENH2[train\_ind,]  SVM\_test <- DFT\_dENH2[-train\_ind,] |

SVR의 학습과 검증을 위해 세가지 표면에서의 dE 대한 데이터를 생성한다.

#### Q7. 위에서 만든 데이터를 사용하여 SVM 모델을 생성할 것이다. Gamma = 0.1, cost = 10 환경에서 SVM 모델을 생성하라.

**A7.** 다음과 같은 절차를 통해 SVM을 생성할 수 있다.

|  |
| --- |
| library(e1071)  library(Metrics) |

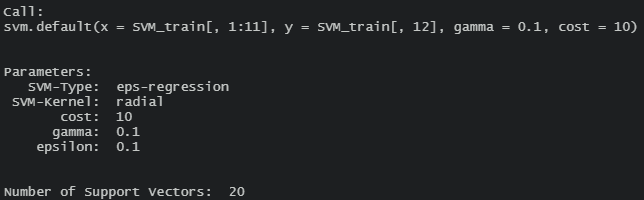
SVR 모델을 내장하고있는 ‘e1071’ package를 import한다.

SVR 모델의 검증을 예측 정확도 평가 지표를 내장하고 있는 ‘Metrics’ package를 import 한다.

|  |
| --- |
| model\_SVM <- e1071::svm(SVM\_train[,1:9], SVM\_train[,10],  gamma = 0.001, cost = 60) |

SVR의 학습을 진행한다. Gamma와 cost는 ‘svm()’ 함수의 ‘arg’로, 하이퍼파라미터에 해당한다. 이는 여러 번의 시행착오를 통해 얻도록 한다.

|  |
| --- |
| summary(model\_SVM) |



‘summary()’ 함수를 통해 모델의 정보를 얻을 수 있다..

#### Q8. 모델의 예측 정확도를 파악하라. 예측 정확도는 R2, MSE, RMSE, 그리고 MAE 지표를 사용해 판단하라.

**A8.** 다음과 같은 절차를 통해 정확도를 평가할 수 있다.

|  |
| --- |
| pred\_train <- predict(model\_SVM, SVM\_train[,1:9])  pred\_test <- predict(model\_SVM, SVM\_test[,1:9]) |

SVR의 모델을 통해 학습 데이터와 검증 데이터를 예측한다.

|  |
| --- |
| rsq <- function (x, y) cor(x, y) ^ 2 |

예측 정확도 평가를 위한 R2 계산식을 만든다. 해당 함수는 ‘rsq()’로 명명한다.

|  |
| --- |
| rsq(pred\_train,SVM\_train[,10])  Metrics::mse(pred\_train,SVM\_train[,10])  Metrics::rmse(pred\_train,SVM\_train[,10])  Metrics::mae(pred\_train,SVM\_train[,10])  rsq(pred\_test,SVM\_test[,10])  Metrics::mse(pred\_test,SVM\_test[,10])  Metrics::rmse(pred\_test,SVM\_test[,10])  Metrics::mae(pred\_test,SVM\_test[,10]) |

학습 데이터 예측에 대한 정확도와 검증 데이터 예측에 대한 정확도를 계산한다.

|  |
| --- |
| plot(pred\_train, SVM\_train[,10])  plot(pred\_test, SVM\_test[,10]) |

예측 값과 실제 값을 시각화 한다.

### [결론] ## 추가 필요

### [학습 결과]

* 학습 내용

반도체 소재 개발을 위한 SVM 방법론 기반 단원자 증착 설계 익히기.

* 학습 결과 확인하기

SVM 알고리즘의 활용 방법 및 예측 결과 해석 방법 익히기.

* 학습 결과 응용하기

본 장의 학습내용에 기반해 단원자 증착을 위한 전구체 물질 탐색 및 효과적 표면 화학반응 설계를 위한 전구체 물질 성능 예측하기.