Journal of the Korean Institute of Industrial Engineers
Vol. 51, No. 6, pp. 465-474, December 2025.
ISSN 1225-0988 | EISSN 2234-6457
https://doi.org/10.7232/JKIIE.2025.51.6.465

데이터파밍을 활용한 빅데이터 기반 품질개선 방법

주혜진 · 송유진 · 변재현*†

경상국립대학교 산업시스템공학부

Quality Improvement Method Using Data Farming on Quality Big Data

Hyejin Ju · Yujin Song · Jai-Hyun Byun

Department of Industrial and Systems Engineering, Gyeongsang National University

Machine learning has been utilized across various industries to optimize quality characteristics for quality improvement. However, existing methods such as Bayesian optimization and genetic algorithm suffer from drawbacks including theoretical complexity and a lack of knowledge about quality characteristics near the optimal conditions. This study proposes a data farming method to systematically find the optimal region of features by applying the Nearly Orthogonal Latin Hypercube(NOLH) design to quality big data. The proposed method employs easy-to-implement region-reduction technique by considering significant features for the labels. Moreover, unlike existing methods, it can present optimal features' region ensuring a desired level of quality characteristics even if slight fluctuations occur in the process features. A case study shows that the proposed method performs better than other methods. The data farming method is expected to help practitioners to improve process performance using quality big data.

Keywords: Quality Improvement, Data Farming, Nearly Orthogonal Latin Hypercube Design, Machine Learning, Optimal Region Exploration

1. 서 론

빅데이터를 수집할 수 있게 되면서 머신러닝을 활용한 데이터 분석과 그에 관련된 연구가 다양한 분야에서 활발히 진행되고 있다. 그중 품질관리 측면에서 머신러닝은 품질특성을 예측하거나 최적화하는 용도로 주로 사용된다. 머신러닝을 기반으로 한 품질특성 최적화는 제품이나 공정의 품질을 개선하기 위한 목적으로 진행되며, 품질특성이 레이블(반응변수)인 머신러닝 모델을 구축한 후, 최적화 방법을 활용하여 품질특성을 개선하기 위한 피처(설명변수) 조합을 도출한다. 이와 관련된 사례 연구가 1990년대부터 꾸준히 진행되었는데, 대부분 기존 최적화 방법을 그대로 이용하거나, 이를 발전시킨 연구이다(Koksal *et al.*, 2011; Weichert *et al.*, 2019; Ullrich *et al.*, 2024).

품질특성을 최적화하는 기존 연구에서 주로 활용하는 방법은 수치 최적화, 베이지안 최적화(Baysian Optimization), 유전알고리즘(Genetic Algorithm) 등이 있다. 수치 최적화 방법은 경사하강법(Gradient Descent)과 같이 미분을 활용하여 모델의 최적점을 찾는 가장 기본적인 방법이지만, 모델이 미분 불가능한 경우에 적용하기 어렵다. 베이지안 최적화는 사전분포와 관측데이터를 기반으로 사후분포를 추정하여 최적화 문제를 해결하는 방법이다. 이는 적은 수의 데이터 포인트로 최적점을 찾을 수 있지만, 이론적으로 복잡하여 계산 비용이 많이 발생하는 단점이 있다. 유전알고리즘은 다윈의 적자생존 원리, 멘델의 유전법칙과 같은 자연의 진화과정을 응용한 메타 휴리스틱(Meta-Heuristic) 방법으로, 데이터 차원의 크기와 상관없이 사용할 수 있다. 그러나, 제시한 해가 최적이라고 보장할 수 없고, 피처와 레이블 사이 관계를 명시적으로 고려하지 않는다. 이러한 최적화 방법들의 공통적인 특징은 분석 결과를 한 개 혹은 여러 개의 피처 값 조합으로 제시한다는 것이다. 이 경우, 제시된 최적 조건 주변에서 품질특성 값이 어떻게 나타날

† 연락저자 : 변재현 교수, 52828, 경남 진주시 진주대로 501 경상국립대학교, Tel: 055-772-1692, Fax: 055-772-1699, e-mail: jbyun@gnu.ac.kr
2025년 7월 31일 접수; 2025년 11월 4일 수정본 접수; 2025년 12월 8일 게재 확정.

지 알 수 없으므로 피처 값이 조금 변경된다면, 품질특성 값이 급격히 내려갈 가능성이 있다. 따라서 원하는 품질특성을 안정적으로 얻기 위해서 최적점뿐만 아니라 그 주변에서도 특성이 우수한 최적의 피처 영역을 찾는 방법이 필요하다.

본 연구는 데이터파밍(Data Farming)을 활용하여 최적의 피처 영역을 모색하는 빅데이터 기반 품질특성 최적화 방법을 제안하고자 한다. 데이터파밍은 시스템에서 발생할 수 있는 다양한 결과를 조사하여 시스템의 행태를 파악할 수 있도록 도와주는 방법이다(Horne and Sechter, 2013; Horne and Schwierz, 2016). 이 방법은 전체 데이터 공간을 효과적으로 탐색하는 것에 유용하기 때문에, 적은 수의 데이터를 이용하여 품질특성이 최적인 영역을 찾는 데에 적합하다. 본 논문에서 제안하는 방법을 이용하여 최적 영역을 찾게 되면, 실제 운영 조건이 최적 조건에서 조금 벗어나더라도 품질특성이 안정적으로 유지될 수 있는 영역을 알 수 있어서 사용자가 공정을 유연하게 운영할 수 있다. 따라서 최적 조건에 변동이 있을 때, 품질특성을 보장하기 어렵다는 기존 연구의 단점을 보완할 수 있다. 또한, 본 논문에서 제안한 방법은, 데이터파밍을 이용하여 레이블에 유의한 피처를 고려하면서 영역을 축소해 나가는 과정을 활용하므로, 이해하고 적용하기가 쉽다.

본 논문의 구성은 다음과 같다. 제2장에서는 데이터파밍의 기본 개념과 본 연구에서 활용된 탐색 기법을 중점적으로 설명하고, 제3장에서는 데이터파밍을 이용하여 품질특성을 최적화하는 새로운 방법을 제안한다. 제4장에서는 사례데이터를 제안한 방법으로 분석하여 그 결과를 제시하며, 제5장에서는 결론을 기술한다.

2. 데이터파밍

2.1 데이터파밍의 개념과 활용

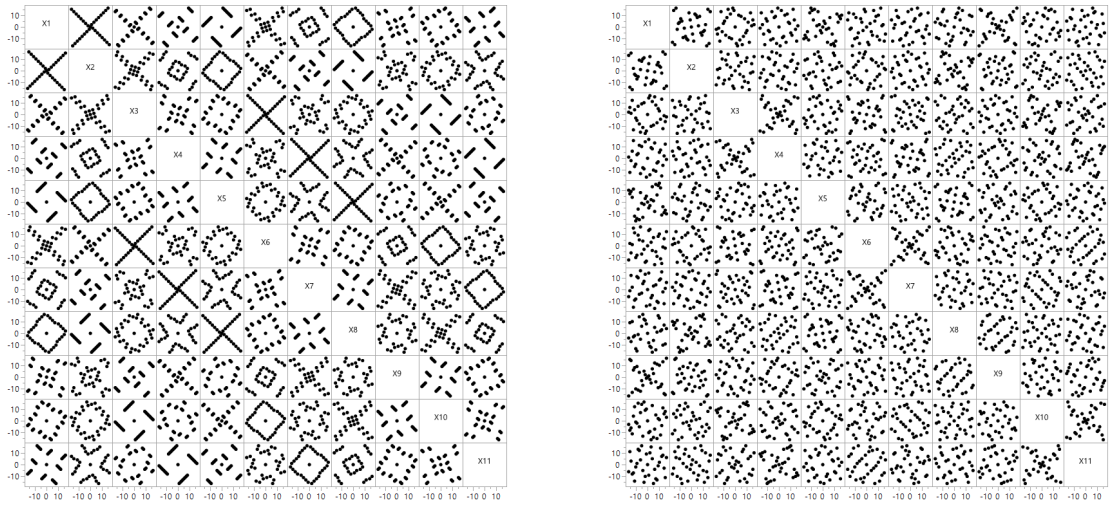
데이터파밍(Data Farming)은 1997년 미해병대(USMC)에서 처음 활용한, 국방 분야에서 맨 먼저 사용된 방법으로서, 시뮬레이션, 계산 모델 등을 이용하여 시스템 행태에 관한 유용한 정보를 쉽게 파악할 수 있도록 다양한 결과를 조사하고 그 과정에서 인사이트를 얻어 의사결정을 지원하기 위해 개발되었다(Horne and Sechter, 2013; Horne and Schwierz, 2016; Sanchez, 2020). 이 방법은 대규모 데이터 공간을 빠르게 탐색하여 시스템의 행태 정보를 효과적으로 파악하고자 할 때 주로 사용한다(Horne and Schwierz, 2016; Sanchez, 2020).

데이터파밍에서는 공간채움설계(Space-Filling Design)를 주로 활용한다. 공간채움설계는 전체 데이터 공간을 균형 있게 탐색할 수 있도록 설계점을 배치하는 방법으로, 최소최대거리설계(Minimax Distance Design), 최대최소거리설계(Maximin Distance Design), 라틴하이퍼큐브설계(Latin Hypercube Design, LHD) 등이 있다. 이러한 설계에 필요한 주요 특성은 '공간채움(Space-filling) 특성'과 '투시성(Projectivity)'이다. 공간채움 특성은 설계점이 전체 데이터 공간에 잘 흩어진 정도를 나타내는 것이고, 투시성은 고차원 상의 설계점을 저차원으로 표현했을 때, 전체 설계점 수가 유지되는지를 나타내는 것이다. 투시성이 좋다면, 각 인자(factor)는 전체 설계점 수만큼 수준을 가지게 되어, 각 인자의 다양한 입력값에 따른 시스템 결과를 확인할 수 있다. 여기서 '인자'는 공간채움설계를 구성할 때 설명변수를 지칭하는 용어로, 머신러닝 모델에서 '피처(feature)'와 같은 의미이다. 최소최대거리설계와 최대최소거리설계는 설계점 간 거리를 기반으로 각 점이 넓게 퍼지도록 배치하는 설계이다. 이 설계는 공간채움 특성이 뛰어나지만, 투시성은 다소 떨어진다. 라틴하이퍼큐브설계는 각 인자의 범위를 동일한 간격으로 나눠 수준을 결정한 뒤, 수준별로 설계점이 한 개만 있도록 배치하는 설계이다. 이는 투시성이 뛰어나지만, 공간채움 특성이 좋지 않을 수 있다.

라틴하이퍼큐브설계를 좀 더 살펴보고자 한다. LHD는 수준별로 하나의 설계점만 배치하기 때문에 투시성이 항상 좋고, 다양한 설계행렬을 구성할 수 있다. Morris and Mitchell(1995)은 설계에 따라서 LHD의 공간채움 특성이 좋지 않을 수도 있는 특징을 보완하기 위해 최대최소 라틴하이퍼큐브설계(Maximin Latin Hypercube Design)를 고안하였다. Ye(1998)는 다른 인자와 상관없이 특정인자의 효과를 파악하기 위해 설계행렬이 직교성(Orthogonality)을 가지는 직교 라틴하이퍼큐브설계(Orthogonal Latin Hypercube Design, OLH Design)를 개발하였다. <Figure 1(a)>는 11개의 인자를 대상으로 33개의 설계점을 가지는 OLH 설계(O_{11}^{33})의 각 인자를 2개씩 묶어 행렬 산점도(Matrix Scatter Plot)를 작성한 것이다. 이 그림을 보면, 각 산점도에서 두 인자의 관계가 특정한 패턴으로 나타나면서 설계점이 공간에 골고루 퍼져 있지 않아 공간채움 특성이 좋지 않음을 알 수 있다. Cioppa(2002)는 이를 보완하기 위해 OLH 설계의 공간채움 특성을 강화하면서 설계의 직교성은 최대한 유지하는 '거의 직교하는 라틴하이퍼큐브설계(Nearly Orthogonal Latin Hypercube Design, NOLH Design)'를 개발하였다. <Figure 1(b)>는 11개 인자를 대상으로 33개의 설계점을 가지는 NOLH(이하 놀) 설계(N_{11}^{33})의 각 인자를 2개씩 묶어 행렬 산점도를 작성한 것이다. <Figure 1>의 두 그림을 비교하면, 놀설계는 각 설계점이 전체 데이터 공간에 골고루 퍼져 있어 OLH 설계보다 공간채움 특성이 우수하다는 것을 알 수 있다.

2.2 놀설계(NOLH Design)

본 논문에서는 놀설계를 이용한 빅데이터 기반 품질특성 최적화 과정을 제시한다. 놀설계는 라틴하이퍼큐브설계에서 파생되어 투시성이 우수하고, 공간채움 특성도 좋아 데이터 공간을 효율적으로 탐색할 수 있으며, 직교성을 거의 유지하기 때문에 각 인자의 효과를 다른 인자와 거의 무관하게 구할 수



(a) O_{11}^{33} Plot (b) N_{11}^{33} Plot

Figure 1. Matrix Scatter Plot for O_{11}^{33} and N_{11}^{33} Designs

있다(Cioppa, 2002; Cioppa and Lucas, 2007). 따라서 놀설계는 데이터 공간을 균형 있게 탐색하고 레이블에 유의한 피처를 파악하여 최적의 피처 영역을 체계적으로 찾는 데에 유용하다.

놀설계는 N_k^n로 표기하는데, 이는 k개의 인자를 대상으로 n번 실험을 진행하는 놀설계를 의미한다. 라틴하이퍼큐브설계는 인자의 각 수준에서 실험을 1회 진행하도록 설계되므로, 놀설계의 인자별 수준 수는 n개이다. N_k^n에서 설계점 수(n)와 인자 수(k)는 임의의 정수 m에 의해 각각 2^m+1과 $m+\binom{m-1}{2}$로 결정되기 때문에, 놀설계는 그 종류가 정해져 있다. m에 따른 놀설계는 N_7^{17}, N_{11}^{33}, N_{16}^{65}, N_{22}^{129}, N_{29}^{257}가 있고, 각 설계가 수용할 수 있는 인자 수는 순서대로 2~7개, 8~11개, 12~16개, 17~22개, 23~29개이다. 2.1절에 제시된 놀설계행렬 산점도(<Figure 1(b)>)에 대응하는 N_{11}^{33}의 설계행렬을 <Table 1>에 나타내었다. 이 행렬을 이용하면 11개의 인자를 대상으로 33개의 수준을 반영할 수 있다. 수준은 0을 중심으로 좌우에 $(33-1)/2=16$개를 만들어서, 총 33개의 수준을 -16, -15, …, -1, 0, 1, …, 15, 16으로 구성한다. 놀설계행렬의 구성은 Cioppa(2002)와 데이터파밍 종자 센터(Seed Center for Data Farming)의 홈페이지인 'https://nps.edu/web/seed'에서 제공하는 파일로 확인할 수 있다.

Table 1. N_{11}^{33} Design Matrix

No	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}
1	16	-13	-2	-1	-10	12	6	4	16	3	6
2	13	16	-12	-6	-4	-1	-2	-10	13	11	8
3	12	-2	13	-15	-11	-15	15	3	-6	-4	7
4	2	12	16	-14	-3	14	-13	-11	-2	-9	10
5	14	-15	-1	2	-9	6	-12	7	-14	6	-3
6	15	14	-6	12	-7	-2	1	-9	-15	8	-11
7	6	-1	15	13	-8	-16	-16	5	12	-7	-4
8	1	6	14	16	-5	13	14	-8	1	-10	-9
9	5	-8	-9	-10	1	7	3	-6	3	-14	-16
10	8	5	-7	-4	6	-9	-10	1	11	-15	-13
11	7	-9	8	-11	15	-5	4	-14	-4	5	-12
12	9	7	5	-3	14	8	-11	15	-9	2	-1
13	3	-11	-10	9	2	3	-7	-12	-5	-16	15
14	11	3	-4	7	12	-11	9	2	-8	-12	14
15	4	-10	11	8	13	-4	-5	-16	7	13	5
16	10	4	3	5	16	10	8	13	10	1	2

Table 1. N_{11}^{33} Design Matrix(Continued)

No	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}
17	0	0	0	0	0	0	0	0	0	0	0
18	-16	13	2	1	10	-12	-6	-4	-16	-3	-6
19	-13	-16	12	6	4	1	2	10	-13	-11	-8
20	-12	2	-13	15	11	15	-15	-3	6	4	-7
21	-2	-12	-16	14	3	-14	13	11	2	9	-10
22	-14	15	1	-2	9	-6	12	-7	14	-6	3
23	-15	-14	6	-12	7	2	-1	9	15	-8	11
24	-6	1	-15	-13	8	16	16	-5	-12	7	4
25	-1	-6	-14	-16	5	-13	-14	8	-1	10	9
26	-5	8	9	10	-1	-7	-3	6	-3	14	16
27	-8	-5	7	4	-6	9	10	-1	-11	15	13
28	-7	9	-8	11	-15	5	-4	14	4	-5	12
29	-9	-7	-5	3	-14	-8	11	-15	9	-2	1
30	-3	11	10	-9	-2	-3	7	12	5	16	-15
31	-11	-3	4	-7	-12	11	-9	-2	8	12	-14
32	-4	10	-11	-8	-13	4	5	16	-7	-13	-5
33	-10	-4	-3	-5	-16	-10	-8	-13	-10	-1	-2

3. 품질빅데이터를 이용한 데이터파밍 적용 방법

본 논문에서는 데이터파밍을 통해 경제적이고 체계적으로 머신러닝 모델의 예측값을 확인할 위치를 선택하고, 품질특성 예측 결과가 좋은 위치를 기준으로 탐색 공간을 점차 줄이면서 최적의 피처 영역을 모색하는 방법을 제안한다.

머신러닝 모델을 이용한 수치 최적화, 베이지안 최적화, 유전알고리즘 등의 품질특성 최적화 연구는 미분 불가능한 모델에는 적용하기 어렵거나, 이론적으로 복잡하여 계산 비용이 많이 들거나, 피처와 레이블 관계를 고려하지 않는다는 한계점이 있다. 또한, 이들을 이용한 연구는 대부분 특정한 최적 조건(최적점)을 제시하므로, 그 주변에서 품질특성 값이 어떻게 나타날지 알 수 없어서 공정의 미세한 변동에 품질특성이 유지될 것이라 확신할 수 없다. 따라서 이러한 단점을 개선하기 위하여, 원하는 품질특성을 안정적으로 얻을 수 있도록 최적의 피처 영역을 구하기 위하여 본 논문에서 제안하는 방법의 절차를 순서도로 <Figure 2>에 나타낸다.

<Figure 2> 순서도에 있는 각 단계의 자세한 수행 절차는 다음과 같다.

(1) 머신러닝 모델 구축

보유한 데이터에 맞게 데이터 전처리 및 모델 개발 과정을 거쳐 머신러닝 모델을 구축하는 단계이다. 여러 머신러닝 모델 중, 예측 성능이 가장 좋은 모델을 선택한다. 머신러닝 모델은 데이터 영역 탐색을 위해 선정된 지점의 예측값을 구할 때 사용된다.

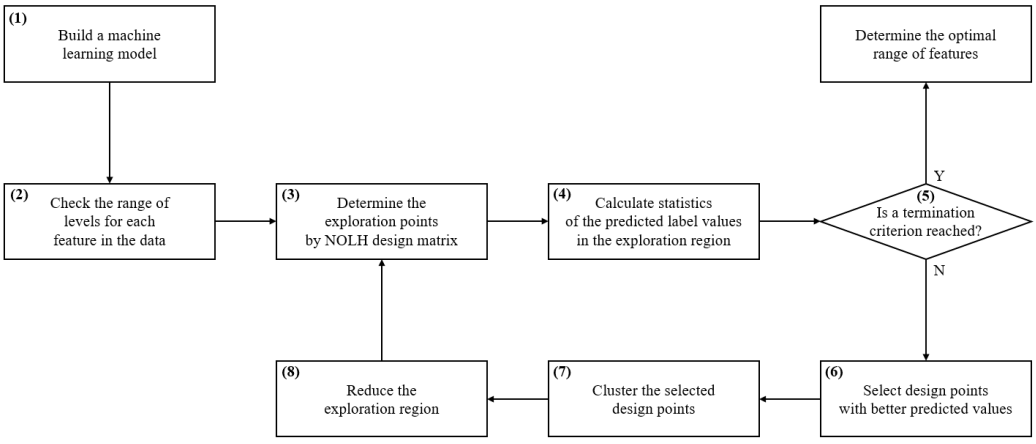


Figure 2. Flowchart of Data Farming Application to Quality Big Data

(2) 피처별 수준 범위 확인

놀설계행렬을 구축하기 위해 피처별 수준 값의 범위를 설정하는 단계이다. 보유한 데이터의 피처별 최솟값, 최댓값을 확인하고, 이를 수준 범위로 정한다.

(3) 데이터 영역 탐색

놀설계로 탐색할 위치를 정하고 데이터 영역을 탐색하는 단계이다. 피처의 수에 따라 적합한 놀설계를 선정하고, 데이터 공간의 피처별 수준 범위에 맞춰 설계행렬을 구축하여 탐색할 위치를 정한다. 이를 머신러닝 모델에 입력하여 설계점별로 레이블 예측값($\hat{y}_i$)을 얻는다.

(4) 설계점의 레이블 예측값 통계량 확인

(3) 단계에서 구한 레이블 예측값의 통계량을 확인하는 단계이다. 설계점별로 얻은 레이블 예측값들의 통계량인 평균, 표준편차, 최솟값, 최댓값을 계산한다. 이들은 다음의 (5) 단계에서 분석 종료 여부를 판단할 때 사용한다.

(5) 분석 종료 여부 결정

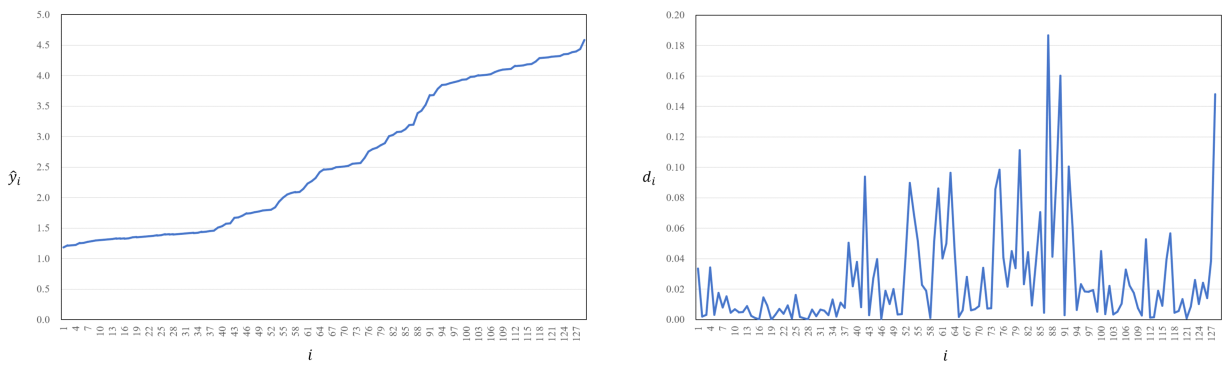
레이블 예측값의 통계량을 바탕으로 분석을 계속 이어갈지, 종료할지 결정하는 단계이다. 분석 종료 요건은 2가지이다. 첫 번째, 모든 레이블 예측값이 사전에 정한 기준값을 달성하면 분석을 종료한다. 예를 들어, 품질특성 값을 최소화할 때, 레이블 예측값의 최댓값이 목푯값 이하이면 분석을 종료한다. 두 번째, 기준값을 달성하지 못했더라도 현재 탐색 영역과 직전 영역에서 확인한 레이블 예측값의 통계량이 모두 비슷하면서 현재 탐색 영역의 표준편차가 작다면 분석을 종료한다. 여기서 통계량은 두 영역 내 예측값의 분포를 비교하기 위한 것으로, 평균값은 중심위치를, 표준편차는 산포를, 최솟값과 최댓값은 예측값의 범위를 파악하는 데 사용된다. 두 영역 내 예측값의 통계량에 서로 차이가 없다면 예측값의 분포가 유사하다는 의미로, 이후 영역을 축소하여도 축소한 영역의 품질특성 값이 크게 달라지지 않기 때문에 더 이상 분석을 진행하지 않는 것이다. 사용자가 미리 정한 기준값이 있으면 첫 번째 조건을, 기준값이 정해져 있지 않고 최소화, 최대화 등 최적화 방향만 정해져 있다면 두 번째 조건을 분석 종료 조건으로 결정하면 된다.

분석 종료 조건을 충족했다면, 최적의 피처 영역은 가장 마지막에 구축한 놀설계의 피처별 수준 범위로 결정된다. 만일 위 조건을 만족하지 못했다면, (6) 단계로 분석을 이어간다.

(6) 예측 결과가 좋은 데이터 선택

레이블 예측값 중 기준값에 가깝게 나온 놀설계 데이터 일부를 선택하는 단계이다. 데이터를 선택하는 과정은 다음과 같다. 레이블 예측값을 크기순으로 나열한 후, 예측값이 서서히 변화하다가 어느 순간 급변하는 지점이 발생하면, 이를 기준으로 첫 번째부터 그 지점까지 데이터를 선택한다. 이는 엘보우 방법(Elbow method)의 기본 개념에서 착안하여 원하는 품질특성에 가까운 설계점을 선택하기 위함이다. 레이블 예측값을 나열하는 방향은 품질특성을 최소화하고 싶으면 오름차순으로 나열하고, 최대화를 원하면 내림차순으로 나열한다. 이때, 최소한 '놀설계 데이터 수'의 10%만큼 데이터를 선택한다. 이렇게 정한 이유는 너무 적은 수의 데이터를 선택하면 국소 최적점(local optimum)에 도달할 수 있기 때문이다. 이렇게 정하면, 놀설계 데이터 수가 최소가 되는 N_7^{17}을 이용할 때도 데이터가 최소한 2개는 선택된다.

값이 급변하는 지점은 꺾은선 그래프를 사용하여 파악한다. 꺾은선 그래프를 그리는 방법은 2가지인데, 데이터를 그대로 이용하거나 i번째 데이터와 $(i+1)$번째 데이터의 차이 값을 이용하는 것이다. <Figure 3(a)>는 레이블 예측값인 $\hat{y}_i$으로 그린 것인데, 이 방법은 별도의 계산 없이 그래프를 간편하게 작성할 수 있다. 반면 <Figure 3(b)>는 $\hat{y}_i$와 $\hat{y}_{i+1}$의 차이 값인 $d_i = \hat{y}_{i+1} - \hat{y}_i$를 이용하여 그린 것이다. 이 방법은 추가 계산이 필요하지만, 값의 변동을 좀 더 명확하게 파악할 수 있다. <Figure 3>의 두 그래프는 같은 데이터를 이용하여 작성한 것이다. 데이터 수가 많지 않거나 데이터값 차이가 명확하게 나타나면 원래 데이터로 그래프를 그려도 값이 크게 변하는 지점을 파악하기 쉬우므로 전자의 방법을, 데이터 수가 많거나



(a) Line Graph of $\hat{y}_i$ (b) Line Graph of $\hat{y}_{i+1} - \hat{y}_i$

Figure 3. Line Graphs of $\hat{y}_i$ and $\hat{y}_{i+1} - \hat{y}_i$

데이터의 값들 차이가 크게 바뀌지 않을 때는 후자의 방법을 사용하는 것이 좋다.

따라서 데이터를 선택하는 기준점은 각 영역 탐색에서 사용한 설계점 수가 n개일 때, 그래프에서 $0.1n$ 위치를 지난 후 값이 크게 달라지는 '첫 번째 지점'이다.

(7) 선택한 데이터로 군집화 수행

전 단계에서 선택한 데이터를 이용하여 다음에 탐색할 영역을 찾기 위해 군집화를 수행하는 단계이다. 선택한 데이터들이 한곳에 모여 있을 수도 있지만 여러 군데 나눠 분포할 수 있으므로, 결과가 좋은 데이터들이 밀집된 영역을 확인한다. 이때 사용할 수 있는 방법은 'K-평균 군집화(K-Means Clustering)', '평균이동 군집화(Mean Shift Clustering)', 'DBSCAN (Density Based Spatial Clustering of Applications with Noise)' 등이 있다. 본 연구에서는 데이터가 밀집된 지역을 중심으로 군집을 형성하고, 군집의 개수를 별도로 정할 필요가 없는 평균이동 군집화를 사용한다.

(8) 탐색 영역 축소

여러 군집 중 하나의 군집을 선택하고, 군집에 포함된 데이터를 바탕으로 탐색할 영역을 축소하는 단계이다. 먼저, 군집을 하나 선택해야 한다. (7) 단계에서 구한 군집 내 데이터 수가 많다는 것은, 그 군집 영역에 품질특성이 좋은 데이터가 밀집되어 있음을 의미한다. 따라서 여러 군집 중 포함된 데이터 수가 가장 큰 군집을 선택한다. 군집 내 품질특성이 좋은 데이터 수가 비슷한 것이 2개 이상이면, 포함된 데이터의 레이블 예측값이 사전에 정한 기준값에 더 가까운 군집을 선택한다. 만일 2개 이상의 군집 내 데이터의 수와 레이블 예측값에 차이가 아주 작으면, 그들을 모두 탐색할 수도 있다.

군집 내 데이터를 포함하는 영역을 축소하기 위해, <Figure 4>처럼 품질특성 값에 영향을 미치는 피처는 그 범위를 축소하고, 다른 피처들의 범위는 그대로 유지한다. 이를 위하여 영역 탐색에 사용한 놀설계행렬과 각 설계점의 레이블 예측값으로 회귀분석을 실시하여 레이블에 유의한 피처를 파악한다. 유의하다고 판단된 피처의 범위는 군집 내 데이터를 대상으로 각 피처의 최댓값, 최솟값으로 재설정하여 축소하고, 유의하지 않은 피처의 수준 범위는 그대로 유지한다. 이때, 탐색 영역이 달라질 때마다 유의한 인자가 달라질 수 있으므로, 영역을 축소할 때마다 회귀분석을 다시 진행해야 한다.

데이터파밍을 활용한 빅데이터 기반 품질개선 방법은 넓은 데이터 공간을 체계적으로 탐색하고, 레이블의 예측값이 좋은 위치로 탐색 공간을 축소하면서 최적 영역을 찾아가는 것인데, 다음과 같은 특징이 있다. 첫째, 적은 수의 데이터로 넓은 공간을 체계적으로 탐색할 수 있고, 최적 영역을 찾아가는 과정이 복잡하지 않다. 둘째, 분석 과정에서 사용한 놀설계행렬은 거의 직교하여 특정 피처가 레이블에 미치는 영향을 다른 피처와 상관없이 파악할 수 있다. 따라서 피처와 레이블 간의 관계를 고려하여, 레이블에 유의한 피처의 범위만 축소하면서

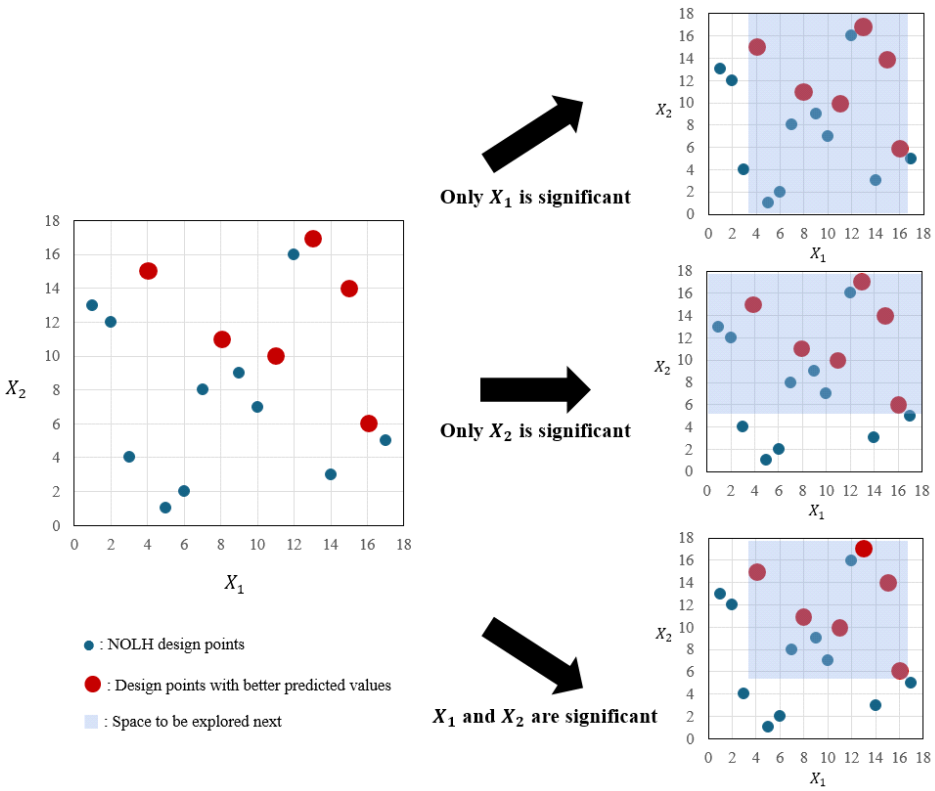


Figure 4. Reducing the Exploration Space

최적 영역을 찾아갈 수 있다. 셋째, 이 방법을 이용하여 선정한 최적 영역은 공정 변동에 강건하다. 특정 지점의 피처 수준 조합만 제시하는 기존 연구들은 공정 운영 중 피처 값이 제시된 최적 조건에서 벗어났을 때, 품질특성 결과의 안정성을 보장하기 어렵다. 따라서 최적 조건에서 공정을 운영하지 못하게 되면 분석을 다시 해야 한다. 반면, 본 연구에서 제안한 방법은 최적 영역을 제시함으로써, 피처에 약간의 변동이 있거나, 다른 이유로 최적 조건에서 조금 벗어난 조건에서 공정을 운영해야 할지라도 추가 분석 없이 최적 영역 내 다른 조건을 차선책으로 구할 수 있다.

4. 사례연구

제3장에서 제안한 방법을 실제 품질빅데이터에 적용하여 분석해 보고자 한다. 레이블이 수치형인 데이터를 선택하여 회귀모델을 구축하고, 품질특성 최적화 과정을 수행하였다.

4.1 분석 데이터

분석에 사용한 데이터는 채광공정(mining process)에서 광석에 불순물인 실리카(silica)가 얼마나 들어있는지 파악하기 위해 수집된 것으로서, Oliveira(2017)가 캐글(Kaggle)에 게시하였다. 레이블은 실리카 함량이고, 분석의 목적은 실리카 함량을 최소화하는 영역을 찾는 것이다. 이 사례데이터는 목푯값이 제시되어 있지 않으므로, 분석 종료 여부는 현재 탐색 영역과 직전 탐색 영역의 레이블 예측값 결과를 비교하여 결정하기로 한다.

4.2 분석 과정

(1) 머신러닝 모델 구축 및 데이터의 피처별 수준 범위 확인

머신러닝 모델 구축에 사용한 채광공정 데이터의 수는 737,453개이고, 피처는 22개이다. 머신러닝 모델 중 레이블이 수치형인 데이터를 다룰 수 있는 모델인 선형 회귀(Linear Regression), 랜덤 포레스트 회귀(Random Forest Regressor), 익스트림 랜덤 트리 회귀(Extremely Randomized Trees Regressor), 익스트림 그래디언트 부스팅 회귀(Extreme Gradient Boosting Regressor) 등으로 모델을 구축하였고, 그중 결정계수(R^2)가 0.9994로 가장 높은 익스트림 랜덤 트리 회귀모델을 최종 선택하였다. 익스트림 랜덤 트리 모델(Geurts *et al.*, 2006)은 여러 개의 의사결정나무를 앙상블한 방법의 하나로, 랜덤포레스트와 달리 부트스트랩 샘플링을 수행하지 않고 전체 학습 데이터를 사용하며, 노드 분할에 사용할 피처와 분할 기준을 무작위로 선택한다. 분석에 사용한 데이터는 결측치가 없고, 이상치 처리, 피처 엔지니어링 등의 데이터 전처리를 했을 때 머신러닝 모델 성능이 저하되어 별도의 전처리 과정은 수행하지 않았다.

데이터 공간을 탐색하기 위해 모델 구축 시 사용한 데이터의 피처별 최솟값, 최댓값을 확인하였고, 그 결과를 바탕으로 놀설계에 이용할 피처별 수준 범위를 결정하였다.

(2) 첫 번째 영역 탐색

22개의 피처로 구성된 전체 데이터 공간을 탐색하기 위해 N_{22}^{129} 놀설계행렬을 이용한다. 이때, 수준 범위는 앞에서 확인한 피처별 최솟값과 최댓값을 이용한다. N_{22}^{129} 설계행렬을 머신러닝 모델에 입력하여 설계점별로 레이블 예측값을 구하고, 레이블 예측 결과가 좋은 데이터를 선택했다. 데이터 선택 기준을 정하기 위해 레이블 예측값을 오름차순으로 나열하고, 값들의 차이가 비슷하므로 예측치의 차이값인 d_i로 작성한 <Figure 5>를 이용하여 값이 급변하는 지점을 파악하였다. 제3장의 (6) 단계에 따라, 데이터는 각 영역 탐색에 사용한 데이터 수인 129개의 10%보다 많이 선택해야 하므로, 13개 이상 골라야 한다. 따라서 <Figure 5>에서 13번째 지점 이후, 처음으로 그래프가 급증하는 지점인 38번째를 데이터 선택 기준으로 정했다. $\hat{y}_{38}$에서 $\hat{y}_{39}$ 사이 차이가 급격히 증가했으므로, 레이블

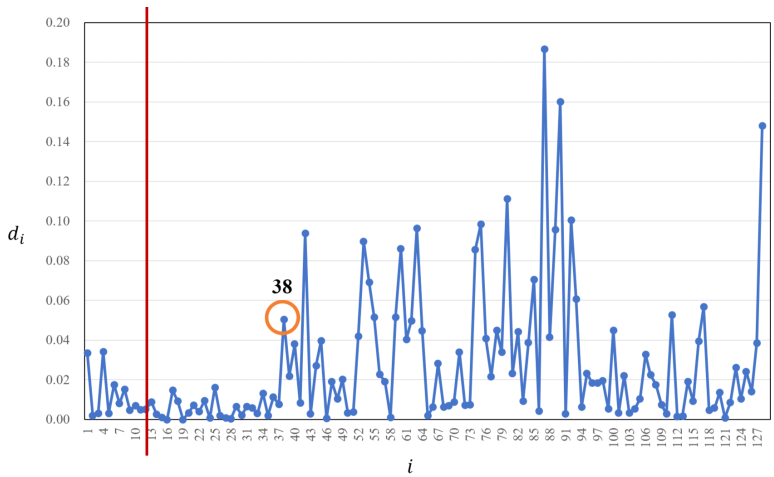


Figure 5. Line Graph Using d_i for the First Region Exploration

예측값을 오름차순으로 나열했을 때 1번째 데이터부터 38번째 데이터까지 선택했다.

선택한 데이터 38개로 평균이동 군집화를 수행하면, 2개의 군집이 형성된다. 1번 군집에는 26개, 2번 군집에는 12개의 데이터가 포함되어, 1번 군집에 속한 데이터를 이용하여 영역을 축소한다. 다음에 탐색할 영역을 피처와 레이블 사이 관계를 고려하여 축소하기 위해 N_{22}^{129} 설계행렬과 설계점별 레이블 예측값을 이용하여 회귀분석을 실시하였다. 수정결정계수인 R_{adj}^2 을 기준으로 후진제거법을 진행하였고, 8개의 피처가 유의하지 않음을 확인했다. 유의한 피처는 선택된 군집 내 데이터를 대상으로 각 피처의 최솟값과 최댓값으로 수준 범위를 변경하여 영역을 축소하였고, 나머지 유의하지 않은 피처는 첫 번째 탐색 영역의 범위를 그대로 유지했다.

4.3 분석 결과

4.1절에서 제시한 분석 종료 조건을 충족할 때까지 위 과정을 반복한다. 14번에 걸쳐 영역을 탐색한 결과, 통계량을 <Table 2>에 정리하였다. 14번째 영역 탐색을 시행한 레이블 예측값 결과가 13번째 영역 탐색 결과와 유사하여 분석을 종료하였다. 따라서 14번째 탐색 영역 결과, 피처의 범위가 품질 개선을 위한 최적의 피처 영역으로 결정되었고, 그 결과를 전체 데이터 범위와 함께 <Table 3>에 제시하였다.

Table 2. Summary of Exploration Results for Minimizing Silica Content

Region exploration stage	Statistics of $\hat{y}$ for NOLH design points for each region exploration stage			
	Average	Standard deviation	Min	Max
1	2.5947	1.1506	1.1848	4.5841
2	1.3585	0.1131	1.1464	1.6824
3	1.2298	0.0848	1.0467	1.4320
4	1.1559	0.0632	0.9940	1.3184
5	1.0972	0.0509	0.9514	1.2265
6	1.0513	0.0439	0.9664	1.1895
7	0.9906	0.0261	0.9260	1.0510
8	0.9632	0.0189	0.9147	1.0082
9	0.9492	0.0147	0.9137	0.9859
10	0.9431	0.0142	0.9094	0.9893
11	0.9337	0.0111	0.9103	0.9635
12	0.9266	0.0087	0.9109	0.9486
13	0.9197	0.0063	0.9078	0.9353
14	**0.9171**	**0.0056**	**0.9058**	**0.9316**

Table 3. Initial and Optimal Ranges of Features to Minimize Silica Content

No	Features	Initial		Optimal	
		Min	Max	Min	Max
1	% Iron Feed	42.740	65.780	58.330	59.010
2	% Silica Feed	1.310	33.400	10.700	11.830
3	Starch Flow	0.002	6,300.230	2,212.420	2,238.190
4	Amina Flow	241.669	739.538	345.029	379.575
5	Ore Pulp Flow	376.249	418.641	391.644	398.298
6	Ore Pulp pH	8.753	10.808	8.887	8.898
7	Ore Pulp Density	1.520	1.853	1.687	1.733
8	Flotation Column 01 Air Flow	175.510	373.871	296.990	300.008
9	Flotation Column 02 Air Flow	175.156	375.992	300.255	320.725
10	Flotation Column 03 Air Flow	176.469	364.346	296.774	333.813
11	Flotation Column 04 Air Flow	292.195	305.871	297.224	298.940
12	Flotation Column 05 Air Flow	286.295	310.270	298.125	302.068
13	Flotation Column 06 Air Flow	189.928	370.910	262.727	282.686
14	Flotation Column 07 Air Flow	185.962	371.593	283.092	297.982
15	Flotation Column 01 Level	149.218	862.274	419.309	497.124
16	Flotation Column 02 Level	210.752	828.919	340.197	349.750
17	Flotation Column 03 Level	126.255	886.822	473.990	550.079
18	Flotation Column 04 Level	162.201	680.359	371.581	420.070
19	Flotation Column 05 Level	166.991	675.644	386.941	442.601
20	Flotation Column 06 Level	155.841	698.861	287.884	371.435
21	Flotation Column 07 Level	175.349	659.902	323.027	414.399
22	% Iron Concentrate	62.050	68.010	67.600	67.640

4.4 기존 방법과 비교

제안한 방법의 우수성을 판단하기 위해 기존 방법인 베이지안 최적화, 유전알고리즘의 분석 결과와 비교해 보고자 한다. 제안한 방법에서 최적 영역을 제시하기까지 14번의 영역 탐색을 진행하였고, 각 영역 탐색에서 129개의 데이터를 고려했으므로 전 과정에서 사용한 데이터 수는 총 1,806개이다. 기존 방법으로 분석할 때도 고려되는 데이터 수를 가능한 한 1,806개에 맞추고자 하였다. 베이지안 최적화는 scikit-optimize 라이브러리의 gp_minimize 함수를 사용했는데, 분석 중 고려하는 데이터 수를 맞추기 위해 사이클 반복 횟수만 조정하며, 그 외 매개변수는 기본값을 사용하였다. gp_minimize 함수의 기본 설정은 사이클당 20개의 데이터 중 최적값을 찾기에 적합한 데이터 1개를 골라 레이블을 확인하기 때문에, 사이클을 총 91번 진행하여 전체 1,820개 데이터를 고려하도록 설정하였다. 유전알고리즘은 DEAP(Distributed Evolutionary Algorithms in Python) 라이브러리를 이용하여 제안한 방법과 같이 사이클을 14번 진행하고, 사이클당 129개의 데이터를 고려하도록 정하였다. 그 외 설정은 DEAP 라이브러리의 프레임워크를 제시한 Rainville *et al.*(2012)의 '예제 1(Example 1)'을 참고하였다. 두 방법 모두 예측모델은 4.2.1절의 익스트림 랜덤 트리 회귀모델을 그대로 사용하였다. 각 방법으로 품질특성 최적화를 진행한 결과는 <Table 4>와 같다. 여기서 본 연구에서 제안한 방법의 예측 결과(predicted values)에는 도출한 최적 영역의 N_{22}^{129} 놀설계점 129개 레이블 예측값의 평균, 최솟값, 최댓값을 나타내었다. 이때, 영역 내 최댓값이 기존 방법의 분석 결과에 따른 품질특성 예측값보다 작아 제안한 방법의 성능이 더 낫다는 것을 확인하였다. 또한, 제시된 분석 결과는 다른 방법들과 달리 개별 최적 조건만이 아닌 영역을 제시함으로써, 피처가 최적 조건에서 약간 벗어나더라도 품질특성이 안정적으로 나타나는 영역을 알 수 있게 하여 변동에 강건하게 공정을 운영할 수 있다.

Table 4. Comparison of Optimization Methods for Minimizing Silica Content

No	Method	Predicted values		Number of points
1	Bayesian Optimization	0.9483		1,820
2	Genetic Algorithm	0.9321		1,806
3	Proposed Region Reduction Method	Average	0.9171	1,806
		Min	0.9058	
		Max	0.9316	

5. 결 론

본 논문에서는 빅데이터 기반 품질개선 과정에서 데이터파밍을 활용하여 최적의 피처 영역을 모색하는 방법을 제안하였다. 데이터파밍은 적은 수의 데이터로 공간 탐색을 하는 데 유용하므로, 데이터 공간 내 품질특성이 우수한 피처 영역을 효율적으로 찾기에 적합한 방법이다. 본 연구는 데이터파밍으로 선정한 탐색 위치에서 머신러닝 모델을 이용하여 품질특성 값을 예측한 다음, 예측값이 좋게 나오는 영역으로 데이터 탐색 공간을 점차 축소하면서 품질특성이 최적인 영역을 찾아가는 방법을 제시했다. 사례연구를 통해 본 연구에서 제안한 데이터파밍을 활용한 품질특성 최적화 방법이 기존 방법보다 품질특성이 더 좋은 공정 운영 조건을 찾을 수 있음을 확인했다.

품질개선을 위한 빅데이터 기반 품질특성 최적화 연구는 종종 진행됐다. 그러나 기존 연구들은 대부분 원하는 품질특성을 나타내는 최적 조건만 제시하므로, 그 최적 조건의 주변 결과는 알 수 없다. 이 경우, 최적 조건에서 변동이 발생할 때 원하는 품질특성을 얻을 수 있을지 확신하기 어렵다. 반면 제안한 방법으로 분석한다면, 품질특성이 좋은 최적 영역을 제시하기 때문에 최적 조건에 따라 운영되던 공정에서 약간의 변동이 발생하여도 안정적으로 일정 수준 이상의 특성을 얻을 수 있다. 또한, 불가피하게 최적 조건에서 공정을 운영할 수 없게 될 때, 추가로 분석하지 않아도 최적 영역에서 차선책을 구할 수 있으므로 문제 해결 시간을 절약할 수 있다. 아울러 제안한 방법의 탐색 과정이 단순하고, 피처와 레이블 간의 관계를 고려하여 유의한 피처를 대상으로 영역을 축소하므로, 쉽고 체계적으로 분석을 수행할 수 있다.

본 연구에서 사용된 놀설계는 2.2절에 제시된 5가지만 있으며 사용가능한 피처의 수는 최대 29개로 한정되어 있어서 다양한 설계점을 구성하기 어렵다는 한계가 있다. 본 연구 결과를 다양한 상황에 적용할 수 있으려면 직교하면서 공간을 잘 채우는 특성을 가진 설계 방법이 추가로 연구되어야 한다. 또한, 유전알고리즘과 마찬가지로 제안하는 방법도 도출한 최적 영역이 항상 최적이라 보장할 수는 없다. 하지만 본 논문에서 제안하는 방법은 품질특성을 최적화하는 영역을 제시하므로 실무자가 품질빅데이터를 활용하여 안정적으로 품질을 개선하는 데 도움이 될 것이라고 기대한다.

참고문헌

Cioppa, T. M. (2002), *Efficient Nearly Orthogonal and Space-Filling Designs for High-Dimensional Complex Models*, Doctoral Dissertation, Naval Postgraduate School, Monterey, CA.

Cioppa, T. M. and Lucas, T. W. (2007), Efficient Nearly Orthogonal and Space-Filling Latin Hypercubes, *Technometrics*, **49**(1), 45-55.

Geurts, P., Ernst, D., and Wehenkel, L. (2006), Extremely Randomized Trees, *Machine Learning*, **63**, 3-42.

Horne, G. and Schwierz, K. P. (2016), Summary of Data Farming, *Axioms*, **5**(1), Article 8.

Horne, G. and Seichter, S. (2013), *Data Farming Support to NATO: A Summary of MSG-088 Work*(STO-MP-MSG-111-14), North Atlantic

Treaty Organization(NATO) Science and Technology Organization.
Koksal, G., Batmaz, İ., and Testik, M. C. (2011), A Review of Data Mining Applications for Quality Improvement in Manufacturing Industry, *Expert Systems with Applications*, **38**, 13448-13467.
Morris, M. D. and Mitchell, T. J. (1995), Exploratory Designs for Computational Experiments, *Journal of Statistical Planning and Inference*, **43**, 381-402.
Oliveira, E. M. (2017), Quality Prediction in a Mining Process, *Kaggle*, Retrieved on September 7, 2019, https://www.kaggle.com/.
Rainville, F. -M. D., Fortin, F. -A., Gardner, M. -A., Parizeau, M., and Gagné, C. (2012), DEAP: A Python Framework for Evolutionary Algorithms, *GECCO '12: Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation*, 85-92.
Sanchez, S. M. (2020), Data Farming : Methods for the Present, Opportunities for the Future, *ACM Transactions on Modeling and Computer Simulation*, **30**(4), Article 22.
Seed Center for Data Farming (2025), https://nps.edu/web/seed/software-downloads, downloaded on April 9.
Ullrich, K., von Elling, M., Gutzeit, K., Dix, M., Weigold, M., Aurich, J. C., Wertheim, R., Jawahir, I. S., and Ghadbeigi, H. (2024), AI-Based Optimisation of Total Machining Performance: A Review, *CIRP Journal of Manufacturing Science and Technology*, **50**, 40-54.
Weichert, D., Link, P., Stoll, A., Ruping, S., Ihlenfeldt, S., and Wrobel, S. (2019), A Review of Machine Learning for the Optimization of Production Process, *International Journal of Advanced Manufacturing Technology*, **104**, 1889-1902.
Ye, K. Q. (1998), Orthogonal Column Latin Hypercubes and Their Application in Computer Experiments, *Journal of the American Statistical Association*, **93**(444), 1430-1439.

저자소개

주혜진 : 경상국립대학교에서 산업공학 학사 및 석사학위를 취득하였다. 관심 분야는 실험계획법, 품질빅데이터 분석, 품질공학이다.

송유진 : 경상국립대학교에서 산업공학 학사학위를 받았고, 산업시스템공학과 석사과정 학생이다. 관심 분야는 품질공학, 실험계획법, 품질빅데이터 분석이다.

변재현 : 서울대학교에서 산업공학 학사, KAIST에서 산업공학 석사 및 박사 학위를 취득하였고, 현재 경상국립대학교 산업시스템공학부에서 교수로 근무하고 있다. 관심 분야는 실험계획법, 품질경영, 데이터 분석공학이다.