

효율적인 온도 관리를 위한

과열 지점 예측

AI 모델 성능 평가

00

목차

01 배경 및 목표

04 AI 모델 비교

02 진행 과정

05 결과

03 데이터 분석

06 결론 및 기대효과

01

배경 및 목표

배경 및 목표

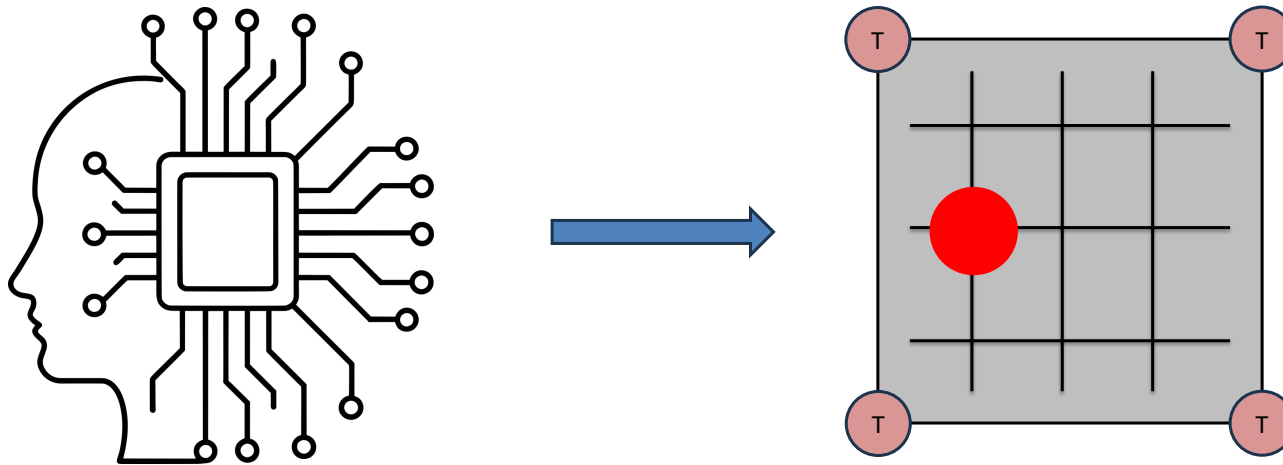
배경

- 온도 관리는 여러 산업과 일상생활에서 다양하게 활용
- 적절하지 못한 온도 관리는 사고와 밀접 한 관계
- 효율적인 온도 관리를 통한 적절한 온도 유지 필요



목표

- AI를 활용하여 외부 온도 센서를 통해 내부의 과열 지점을 예측



효율적인 온도 관리를 위한
과열 지점 예측
AI 모델 성능 평가

02

진행 과정

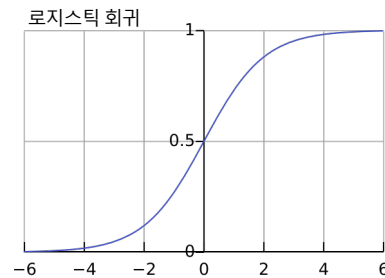
진행 과정

1. 금속판을 이용하여 온도 관리 시스템 조성
2. 금속판 각 모서리에 온도센서 부착, 각각 index 부여 (a~d)
3. 금속판 내부를 3X3의 좌표로 구분, 각각 class 부여 (1~9)
4. 구분된 좌표를 각각 가열과 냉각을 반복하여 온도 측정, 온도 데이터 수집 (약 4,000개)
5. 수집된 데이터를 다양한 AI 분류 모델을 통해 학습 후 성능 평가

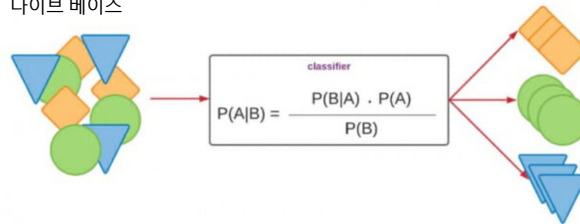


진행 과정

- 사용 AI 모델
 - 로지스틱 회귀, 서포트 벡터 머신, 랜덤 포레스트, k-최근접 이웃, 나이브 베이즈
- Train, Test 8:2 분리
- GridSearchCV를 통해 최적의 파라미터 탐색
- 교차 검증 5회



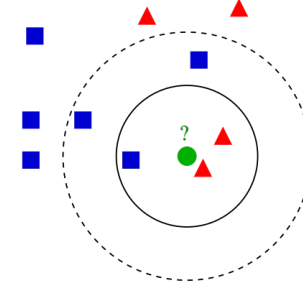
나이브 베이즈



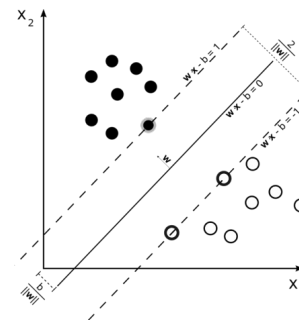
랜덤 포레스트



k-최근접 이웃



서포트 벡터 머신

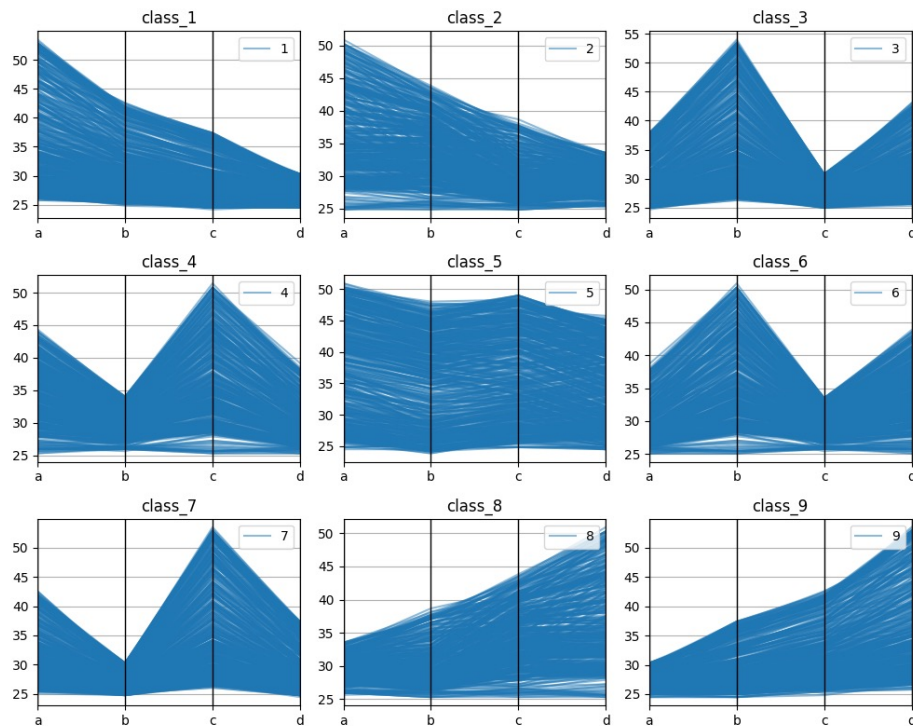


효율적인 온도 관리를 위한
과열 지점 예측
AI 모델 성능 평가

03

데이터 분석

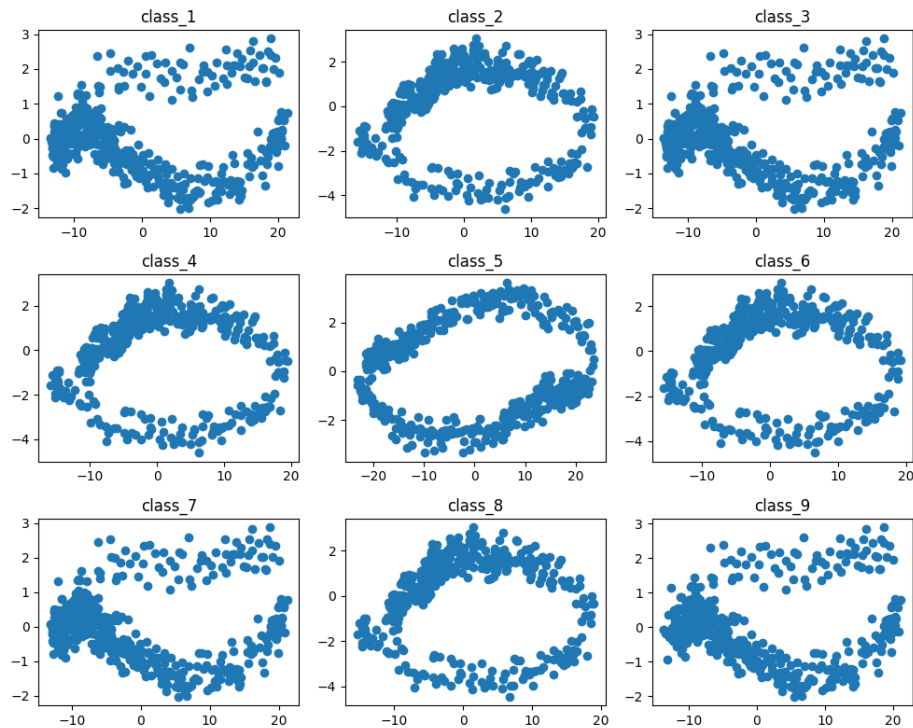
데이터 분석: 평행 좌표 그래프



- 4개의 특징(온도계) 별 각 클래스의 온도 분포
- 클래스 {1, 2}, {3, 6}, {4, 7}, {8, 9}의 온도 분포가 유사

Q: 온도 분포가 유사한 클래스를 구분 가능한가?

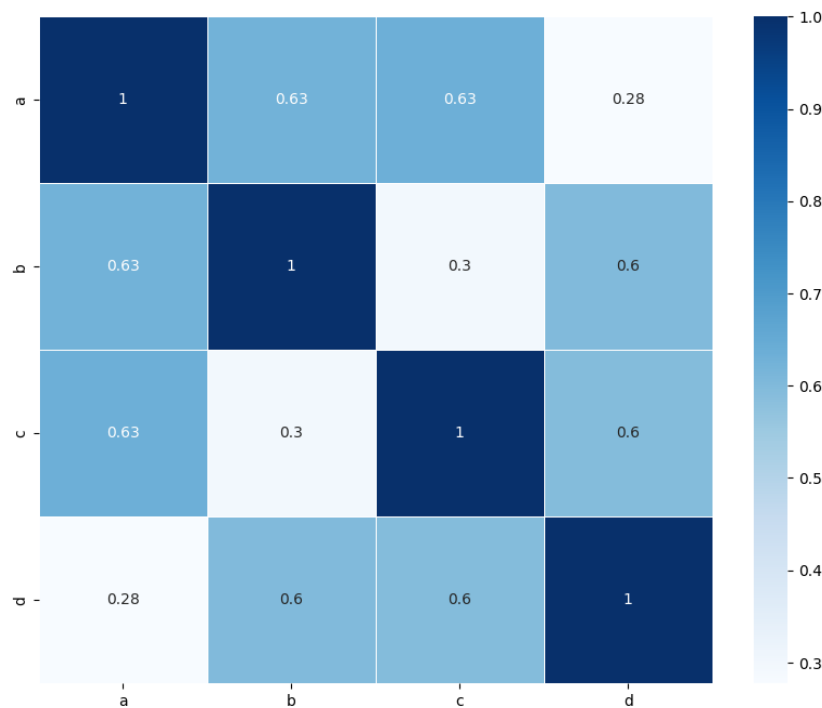
데이터 분석: 주성분 분석



- 4차원 데이터를 2차원으로 축소한 온도 분포
- 클래스 {1, 3, 7, 9}, {2, 4, 6, 8}의 온도 분포가 유사

A: 클래스 {1, 2}, {3, 6}, {4, 7}, {8, 9}의 분포가 각각 다른 양상 즉, 구분 가능

데이터 분석: 상관관계



- 네 가지 특징의 상관 관계 분석
- 서로 대각선에 위치한 온도 센서끼리 상관 관계가 낮은 편
 - $\{a, d\} : 0.29$
 - $\{b, c\} : 0.3$
- 인접한 온도 센서끼리 상관 관계가 높은 편
 - $\{a \mid b, c\} : 0.64$
 - $\{b \mid a, d\} : 0.62$
 - $\{c \mid a, d\} : 0.62$
 - $\{d \mid b, c\} : 0.6$

효율적인 온도 관리를 위한
과열 지점 예측
AI 모델 성능 평가

04

AI 모델 비교

AI 모델 비교: 로지스틱 회귀

로지스틱 회귀 최적 하이퍼파라미터: {'C': 10, 'penalty': 'l1', 'solver': 'liblinear'}

교차 검증 점수: [0.89759036 0.88687783 0.85520362 0.87330317 0.85520362]
 평균 교차 검증 점수: 0.8736357193479801

	precision	recall	f1-score	support
1	0.91	1.00	0.95	117
2	0.99	0.75	0.85	88
3	0.90	0.96	0.93	95
4	0.83	0.88	0.86	94
5	0.85	0.58	0.69	98
6	0.96	0.89	0.92	95
7	0.85	0.94	0.89	83
8	0.80	0.94	0.87	83
9	0.84	0.96	0.90	76
accuracy			0.88	829
macro avg	0.88	0.88	0.87	829
weighted avg	0.88	0.88	0.87	829

예측시간: 0.001996755599975586

선형 회귀를 기반으로 확률을 계산하여
 이진 분류 및 다중 클래스 분류를
 수행하는 모델

- 교차 검증 평균: 0.87
- 테스트 성능 평균: 0.87
- 예측 시간: 0.00

AI 모델 비교: 서포트 벡터 머신

서포트 벡터 머신 최적 하이퍼파라미터: {'C': 10, 'gamma': 'auto', 'kernel': 'rbf'}

교차 검증 점수: [0.97289157 0.96832579 0.9638009 0.95625943 0.95776772]
평균 교차 검증 점수: 0.9638090824837813

	precision	recall	f1-score	support
1	0.96	0.99	0.97	117
2	0.99	0.94	0.97	88
3	0.96	0.97	0.96	95
4	0.92	0.96	0.94	94
5	1.00	0.93	0.96	98
6	0.97	0.92	0.94	95
7	0.95	0.95	0.95	83
8	0.94	0.99	0.96	83
9	0.96	1.00	0.98	76
accuracy			0.96	829
macro avg	0.96	0.96	0.96	829
weighted avg	0.96	0.96	0.96	829

예측시간: 0.10692739486694336

데이터를 고차원 공간으로 매핑하여 최대
마진 초평면을 찾는 모델

- 교차 검증 평균: 0.96
- 테스트 성능 평균: 0.96
- 예측 시간: 0.11

AI 모델 비교: 랜덤 포레스트

랜덤 포레스트 최적 하이퍼파라미터: {'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 200}

교차 검증 점수: [0.96987952 0.97285068 0.9638009 0.95324284 0.96229261]
 평균 교차 검증 점수: 0.9644133093459812

	precision	recall	f1-score	support
1	0.95	1.00	0.97	117
2	1.00	0.92	0.96	88
3	0.97	0.99	0.98	95
4	0.95	0.95	0.95	94
5	0.98	0.95	0.96	98
6	1.00	0.98	0.99	95
7	0.96	0.99	0.98	83
8	0.98	0.99	0.98	83
9	0.99	1.00	0.99	76
accuracy			0.97	829
macro avg	0.97	0.97	0.97	829
weighted avg	0.97	0.97	0.97	829

예측시간: 0.0393068790435791

여러 개의 의사결정 트리를 앙상블로
 결합한 모델

- 교차 검증 평균: 0.96
- 테스트 성능 평균: 0.97
- 예측 시간: 0.04

AI 모델 비교: k-최근접 이웃

k-최근접 이웃 최적 하이퍼파라미터: {'metric': 'manhattan', 'n_neighbors': 5, 'weights': 'distance'}

교차 검증 점수: [0.96536145 0.96983409 0.94871795 0.94570136 0.95475113]
 평균 교차 검증 점수: 0.9568731941340021

	precision	recall	f1-score	support
1	0.94	1.00	0.97	117
2	1.00	0.92	0.96	88
3	0.97	0.99	0.98	95
4	0.93	0.95	0.94	94
5	0.99	0.93	0.96	98
6	0.99	0.96	0.97	95
7	0.95	0.96	0.96	83
8	0.94	0.96	0.95	83
9	0.97	1.00	0.99	76
accuracy			0.96	829
macro avg	0.97	0.96	0.96	829
weighted avg	0.96	0.96	0.96	829

예측시간: 0.007903575897216797

새로운 데이터 포인트의 클래스를 주변 데이터 포인트(k개)의 다수결로 결정하는 모델

- 교차 검증 평균: 0.96
- 테스트 성능 평균: 0.96
- 예측 시간: 0.00

AI 모델 비교: 나이브 베이즈

나이브 베이즈 최적 하이퍼파라미터: {'var_smoothing': 1e-09}

교차 검증 점수: [0.54066265 0.50678733 0.53092006 0.53393665 0.52187029]
 평균 교차 검증 점수: 0.5268353958821711

	precision	recall	f1-score	support
1	0.37	0.42	0.39	117
2	0.58	0.43	0.49	88
3	0.49	0.40	0.44	95
4	0.73	0.57	0.64	94
5	0.41	0.52	0.46	98
6	0.72	0.46	0.56	95
7	0.38	0.48	0.43	83
8	0.77	0.60	0.68	83
9	0.36	0.58	0.44	76
accuracy			0.49	829
macro avg	0.53	0.50	0.50	829
weighted avg	0.53	0.49	0.50	829

예측시간: 0.002730846405029297

베이즈 정리를 기반으로 독립 가정 하에
조건부 확률을 계산하는 모델

- 교차 검증 평균: 0.53
- 테스트 성능 평균: 0.50
- 예측 시간: 0.00

05

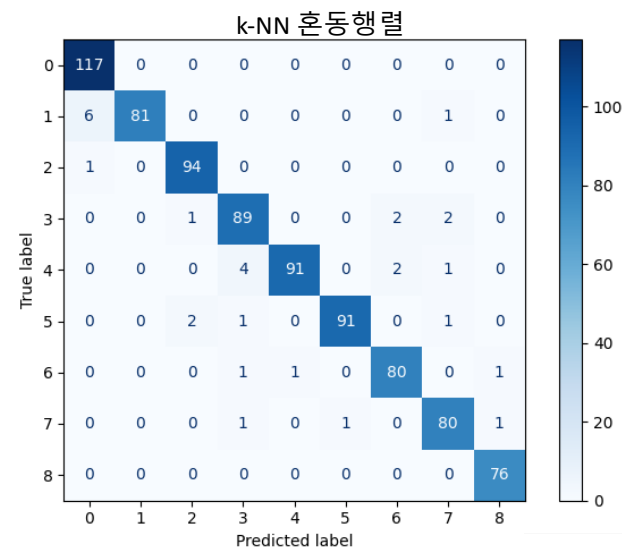
결과

최종 성능 점수 = 100 x (평가 점수 평균 - 0.5 x 예측 시간)

	Logistic Regression	Support Vector Machine	Random Forest	k-Nearest Neighbor	Naive Bayes
교차 검증 평가	0.87	0.96	0.96	0.96	0.53
테스트 성능 평가	0.87	0.96	0.97	0.96	0.50
평가 점수 평균	0.87	0.96	0.97	0.96	0.52
예측 시간	0.00	0.11	0.04	0.00	0.00
최종 성능 점수	87	91	95	96	52

성능 순위

k-최근접 이웃 > 랜덤 포레스트 > 서포트 벡터 머신 > 로지스틱 회귀 > 나이브 베이즈



랜덤 값 테스트

```

0 test:  [[47, 30, 36, 22]]
  result: [1]
1 test:  [[23, 51, 32, 52]]
  result: [6]
2 test:  [[30, 21, 56, 37]]
  result: [7]
3 test:  [[49, 54, 48, 53]]
  result: [5]
4 test:  [[43, 37, 52, 29]]
  result: [4]

```

06

결론 및 기대효과

결론 및 기대효과

결론

- K-Nearest Neighbor (k-최근접 이웃) 모델이 가장 적합
 - 성능 평가 0.96로 정확한 모델
 - 예측 시간 0.00로 빠른 모델
- 외부의 온도 센서를 통해 내부의 과열 지점 예측 가능

기대효과

- 내부의 과열 지점 예측으로 인해 효율적인 온도 관리 가능
 - 불필요한 냉각x
- 2차원 평면을 3차원 공간으로 확장 시 다양한 분야에 활용 가능
ex) 배터리, 데이터 센터, 스마트 팜 등...

END

010-8793-8165
sehyun712@naver.com