

# 교반구동장치 AI 데이터셋

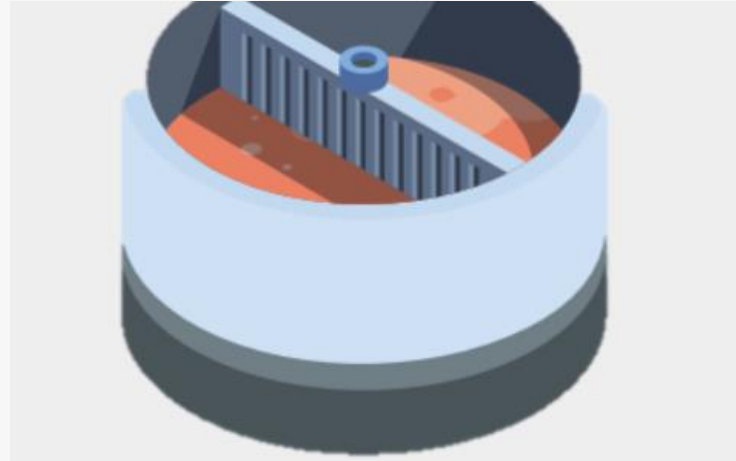
| 스마트팩토리 동아리 C조 |

# 목차

1. 분석 배경
2. 분석 목표
3. 제조 데이터 소개
4. 분석 모델 소개
5. 결과 분석

# 1. 분석 배경

용해 공정 : 분말 원재료를 정제수 등에 녹이는 작업. 원재료의 전처리 공정의 첫 번째 단계이므로 매우 중요하다.



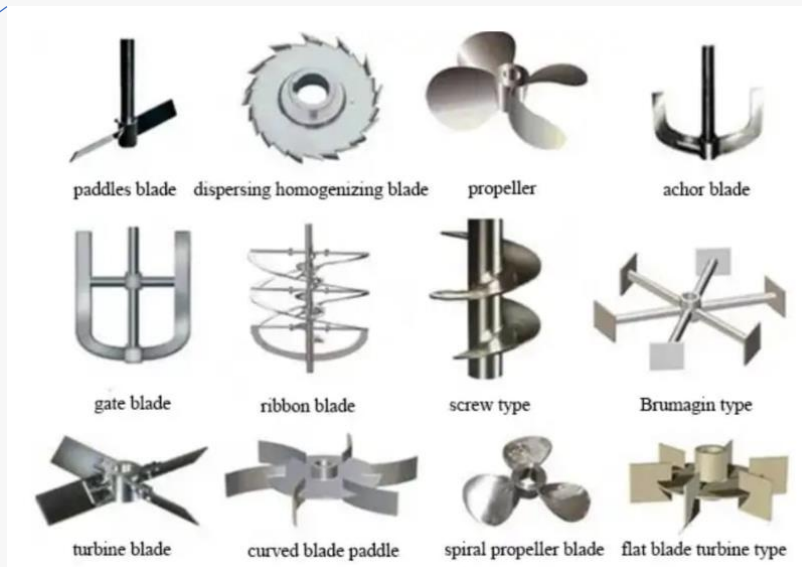
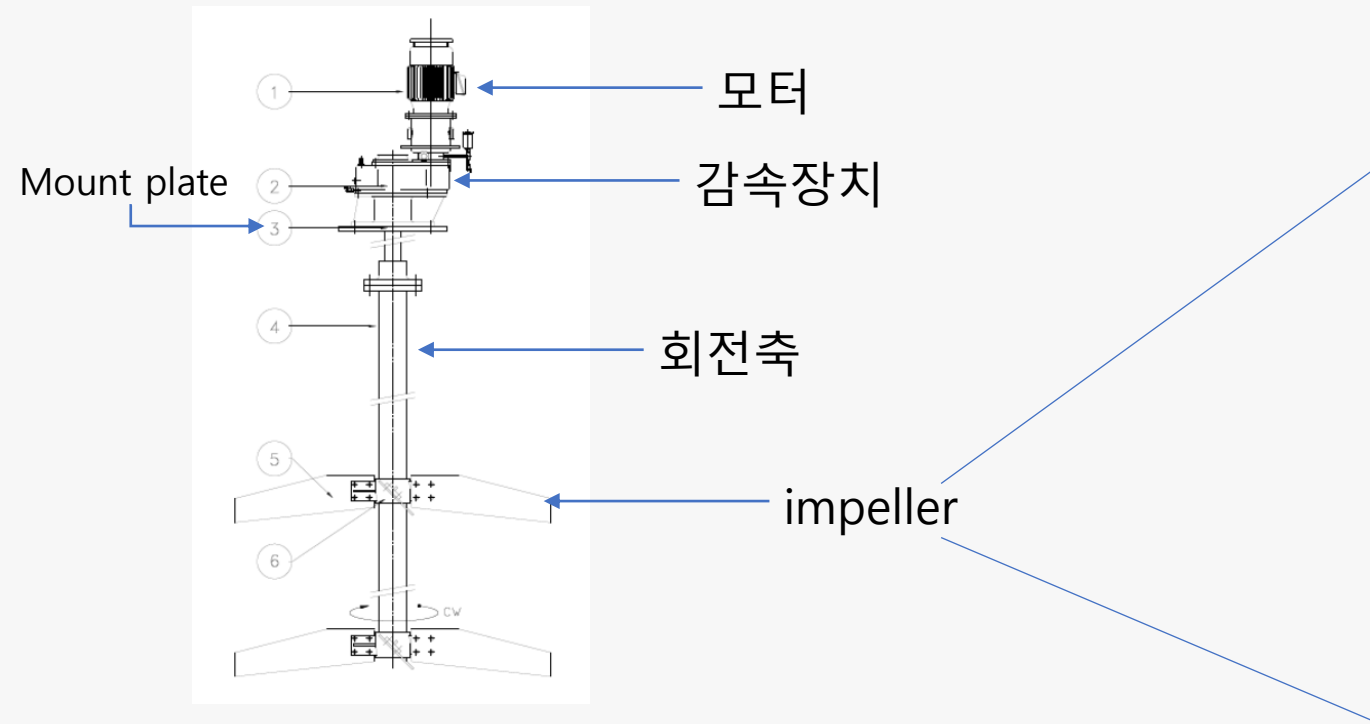
→ 용해 탱크(용해 공정)의 교반구동장치는 분말 원재료를 액상 원재료에 녹이는 과정에서 원료를 교반하여 더 잘 용해되도록 한다.

# 1. 분석 배경

교반기란?

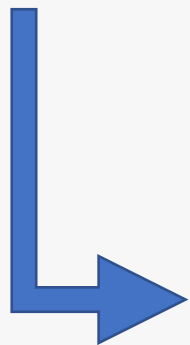
1. 두 가지 이상의 물질을 단순히 섞거나
2. 섞는 동시에 물질의 반응을 촉진시키거나
3. 침전되는 것을 방지하기 위해 사용하는 동력기계장치이다.

교반구동장치 구조 : 모터(motor), 감속장치(reducer), 회전축(mixing shaft), 교반용 날개(impeller)등.



## 2. 분석 목표

1. 전처리 단계에서 모든 원료가 균일하게 혼합되는 것이 매우 중요.
2. 용해탱크는 사용빈도가 높아 고장에 취약함
3. 특히 모터 베어링에 주로 고장 발생
4. 첫 이상징후를 보일 때 대처하는 것이 좋으나 이를 알기 어려움.



교반기동장치 모터의 결함여부에 따른 정상작동과 이상작동 시의 진동데이터 패턴 파악

판별 및 예측할 수 있는 AI 기법을 적용

설비 가동 중 모터의 진동데이터를 이용하여 결함 및 고장 징후를 검출할 수 있는 모델 도출

## 3. 제조 데이터 소개

### 3.1 데이터 수집 방법

- 제조 분야 : 분무건조공법을 이용한 분말유크림 제조
- 제조 공정명 : 용해혼합(교반구동장치)
- 수집장비 : 무선 진동센서
- 수집기간 : 2020년 11월 18일 ~ 2020년 11월 20일 (약 3일)
- 수집 주기 : 사이클 타임 약 10분

### 3. 제조 데이터 소개

#### 3.2 데이터 유형/구조

자동 생성  
인덱스

날짜 / 시간

Z축 진동데이터 결함여부

	A	B	C	D	E
1		Date	Sensor	Quality	
2	0	2021-11-18T01:39:59.605000+00:00	0.064453	PASSED	정상
3	1	2021-11-18T01:39:59.605000+00:00	0.240234	PASSED	
4	2	2021-11-18T01:39:59.605000+00:00	-0.12891	PASSED	
5	3	2021-11-18T01:39:59.605000+00:00	-0.92578	PASSED	
6	4	2021-11-18T01:39:59.605000+00:00	-1.03711	PASSED	
7	5	2021-11-18T01:39:59.605000+00:00	-1.13672	PASSED	
8	6	2021-11-18T01:39:59.605000+00:00	-1.41797	PASSED	
9	7	2021-11-18T01:39:59.605000+00:00	-2.70703	FAILED	결함

### 3. 제조 데이터 소개

#### 3.2 데이터 유형/구조

• 기술통계 :

구분	Sensor
개 수	351,232
평 균	-0.108910
표준편차	2.563446
최 소 값	-9.023438
중 간 값	-0.140625
최 대 값	9.203906
최 빈 값	-0.1484375

• 독립변수/종속변수 정의 :

구분	명 칭	비 고
독립변수	하나의 패턴 내 시간	분석 코드에서 직접 생성
종속변수	z-축 진동데이터	



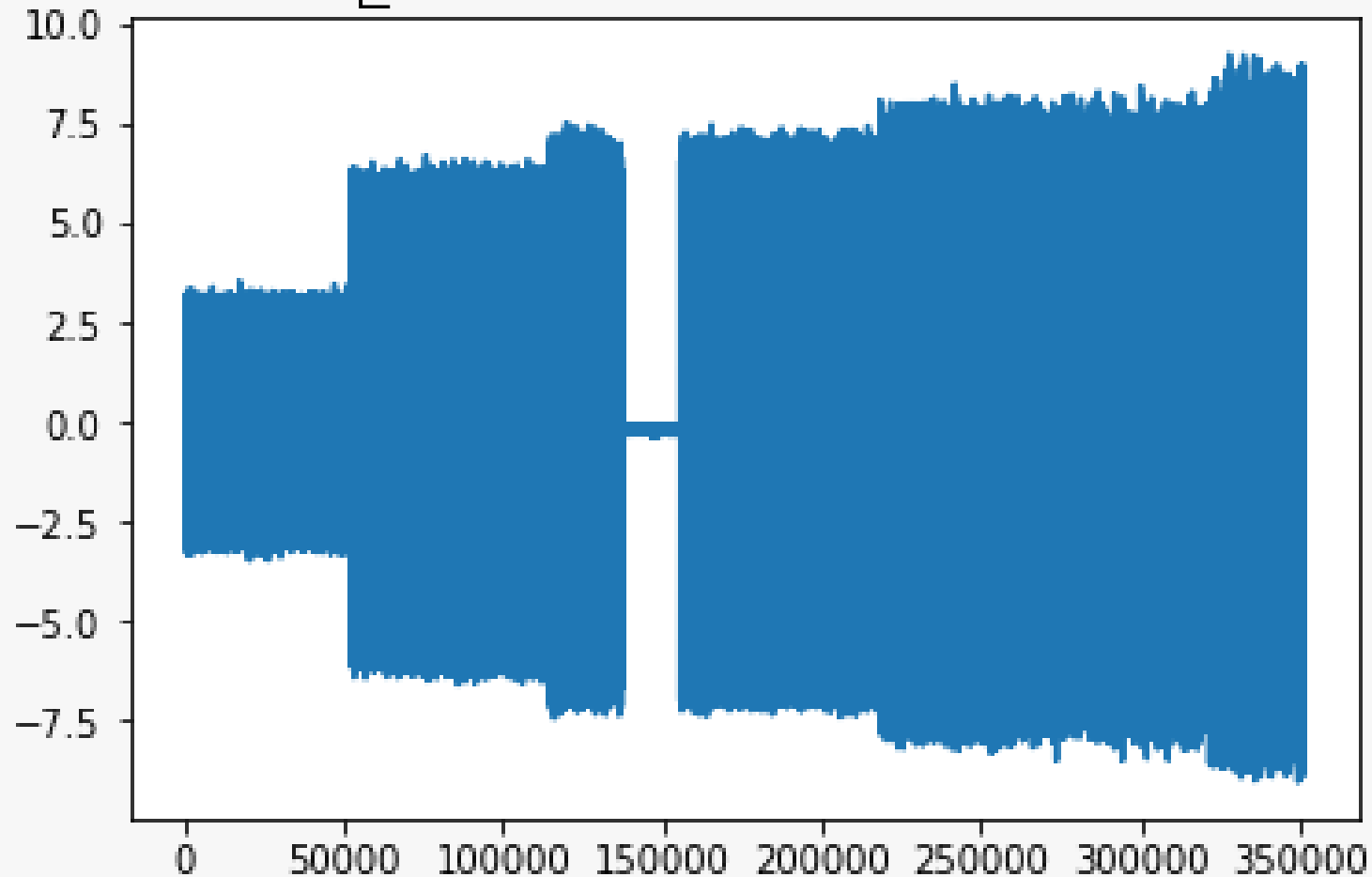
### 3. 제조 데이터 소개

#### 3.3 데이터 통계

데이터 분포 확

인

Z축 진동데이터  
[ $m/s^2$ ]



시간, 시간의 흐름

# 3. 제조 데이터 소개

## 3.3 데이터 통계 데이터 통계량 확인

```
[14] 1 data['Sensor'].describe()
```

```
count    351232.000000
mean      -0.108910
std        2.563446
min       -9.023438
25%       -1.992188
50%       -0.140625
75%        1.843750
max        9.203906
Name: Sensor, dtype: float64
```

연속형인 센서의 변수 기술  
통계

```
▶ 1 data['Quality'].value_counts()
```

```
PASSED    191008
FAILED     160224
Name: Quality, dtype: int64
```

이산형인 Quality 변수의 기술  
통계 (PASSED 기준 : -2이상 2이하)

### 3. 제조 데이터 소개

#### 3.4 데이터 정제

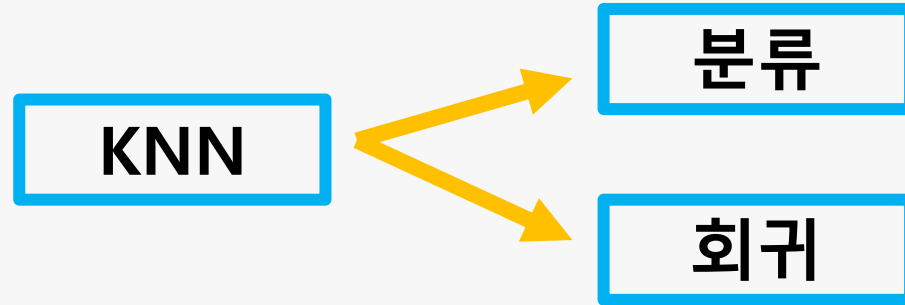
불필요한 변수 제거

```
data = data.drop(['Unnamed: 0'], axis=1)
```

자동 생성 인덱스의 열 값 제거

## 4. 분석 모델 소개

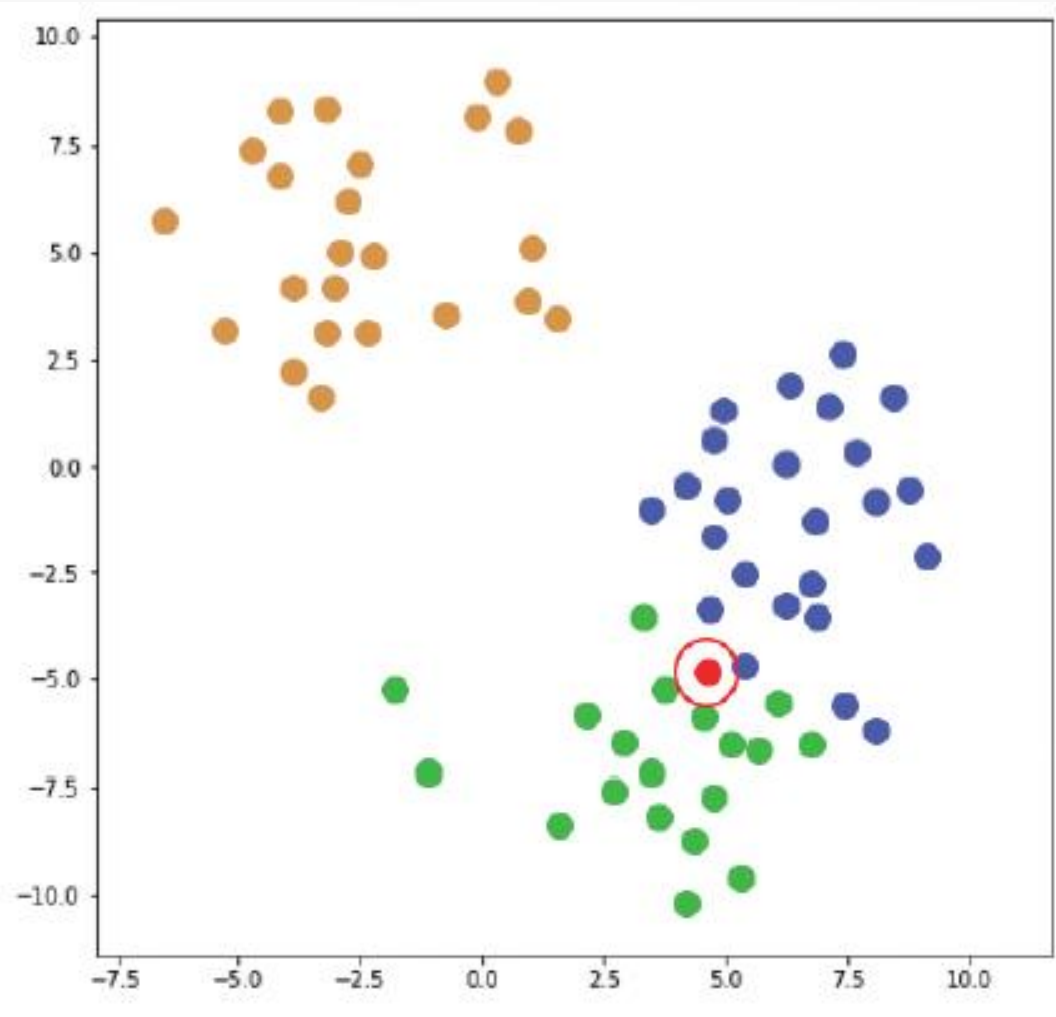
AI 분석 모델 : K-최근접 이웃 회귀 (K-Nearest Neighbor)



- 지도학습 알고리즘으로서 가장 간단한 분류/예측 알고리즘 중 하나
- 이웃과의 유사성을 판단하기 위해 거리를 계산하기 때문에 모든 예측변수들은 수치형
- 비선형 데이터에도 적용이 가능함
- K 최근접 이웃 분류 개념 + 선형회귀분석 개념
- 하이퍼 파라미터 : K (최근접 이웃 수), 거리 함수(데이터 간의 거리 계산식)

## 4. 분석 모델 소개

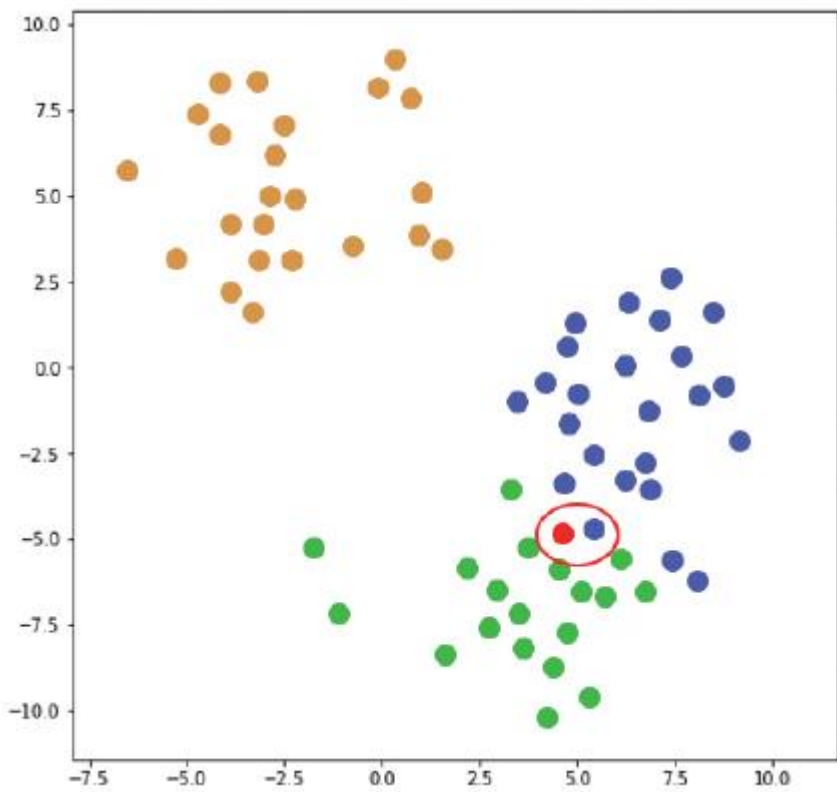
### 4.1 KNN 분류 : 새로운 데이터의 클래스를 분류하기 위한 방법



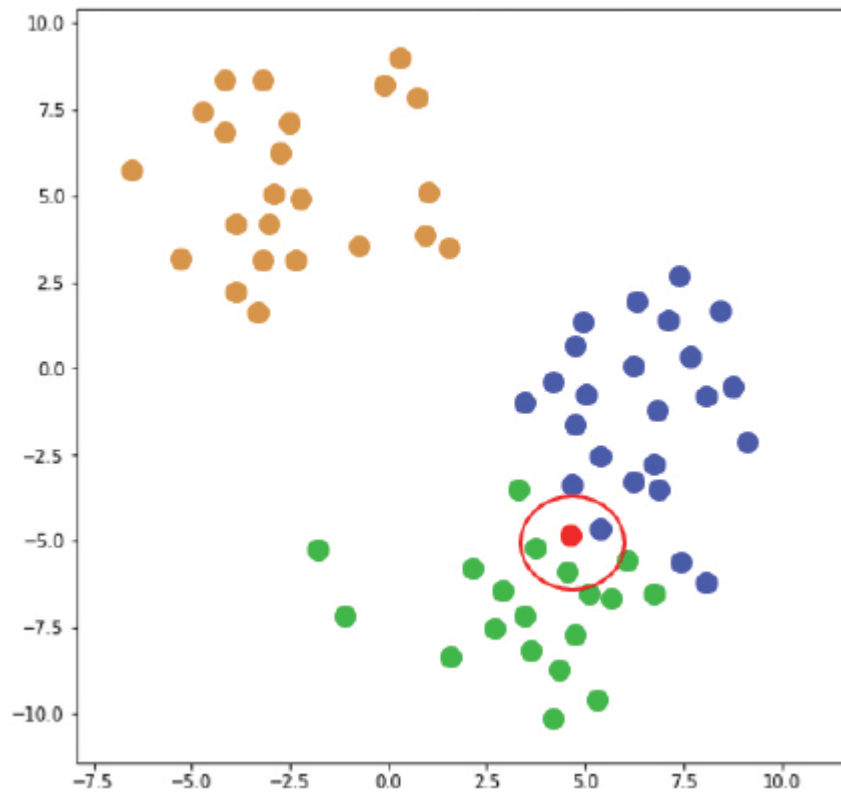
- 하이퍼 파라미터 (Hyper Parameter)

1. K(최근접 이웃 수) : 몇 개로 할 것인지
2. 거리 함수(데이터 간의 거리 계산 식)

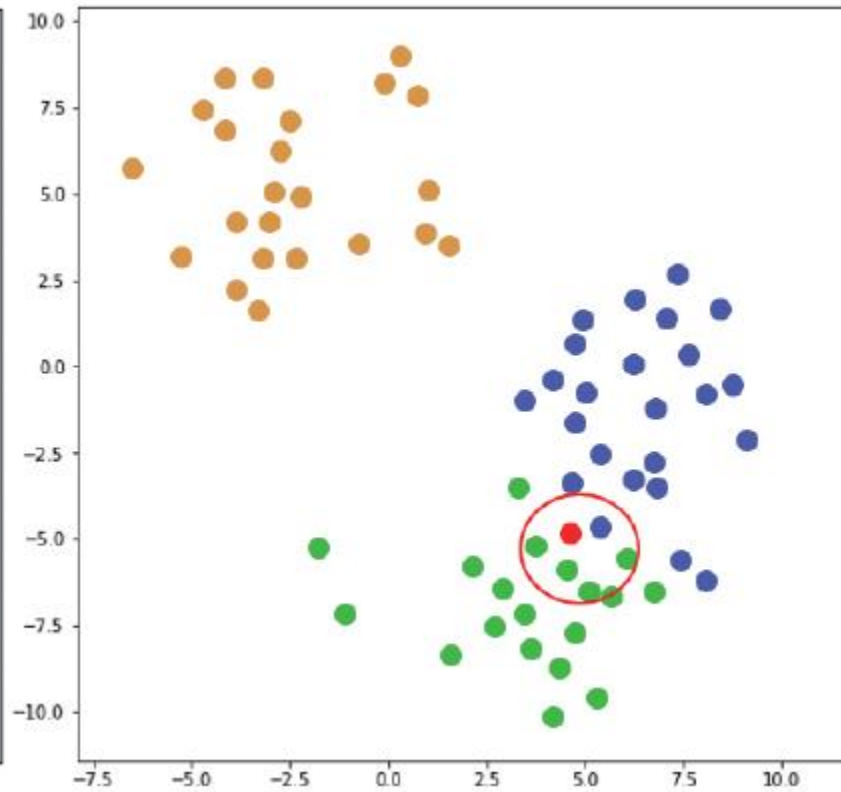
## 4. 분석 모델 소개



**K=1**



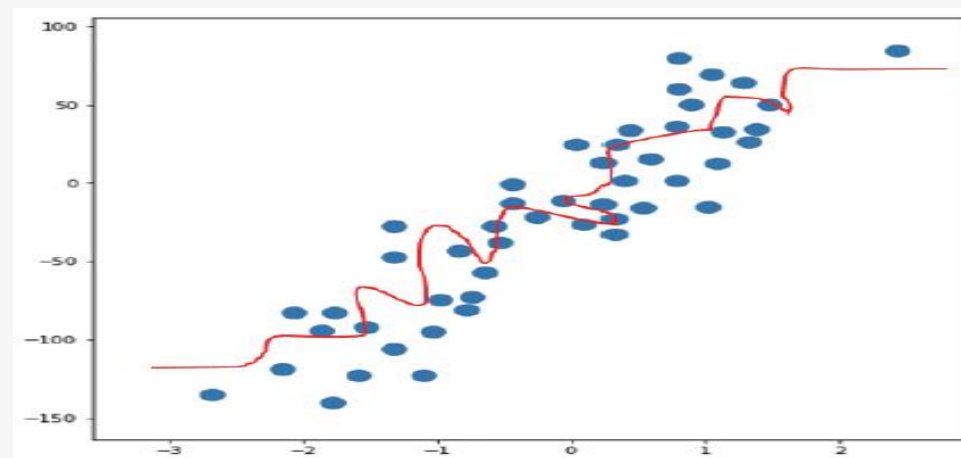
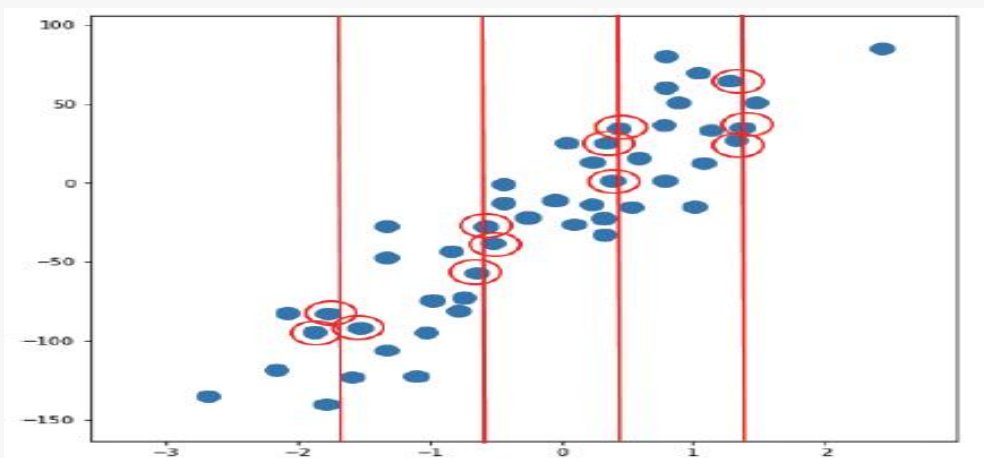
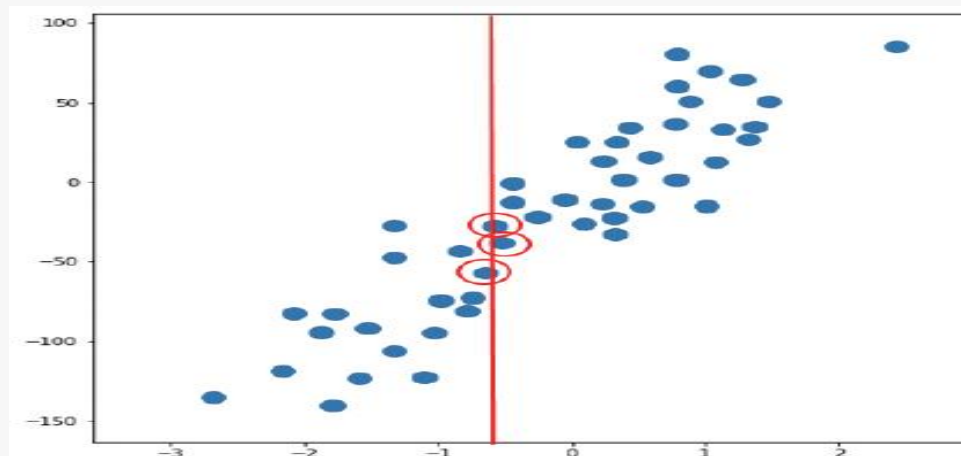
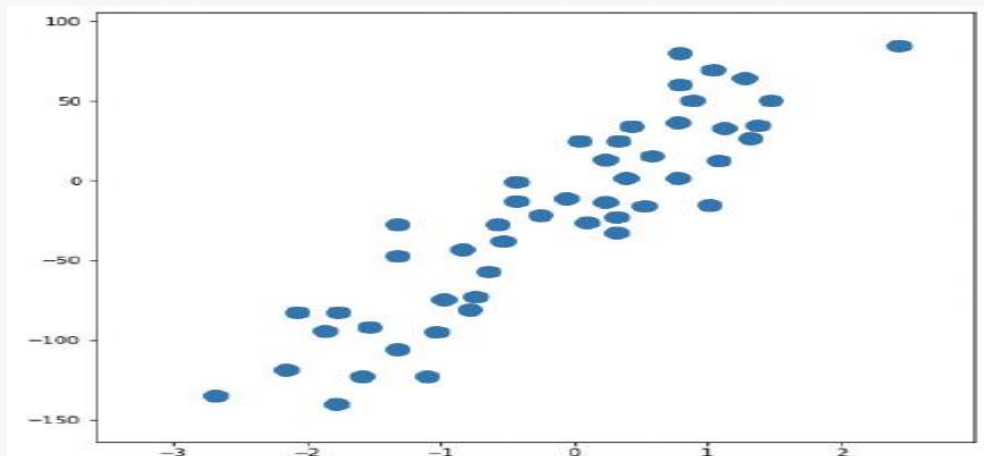
**K=3**



**K=5**

## 4. 분석 모델 소개

### 4.2 KNN 회귀 : 가까운 이웃들의 평균을 예측값으로 사용



## 4. 분석 모델 소개

### 4.3 K-최근접 이웃 회귀 선정 이유

- 센서데이터의 시계열적 패턴에 적합한 표준데이터를 추출하기 위함  
\*표준데이터 = 임계치로서 불량을 검출할 때 사용하는 수치로 된 기준
- 모든 시점에 동일한 임계치를 적용할 경우 시간에 따라 진폭이 달라지는 경우를 검출하기 어려움
- 표준데이터를 추출 해놓는 원리이므로 문제점을 예방할 수 있다.
- KNN 회귀 알고리즘 -> 예측선을 추출할 수 있다.



## 4. 분석 모델 소개

### 4.4 분류기의 성능평가

정확도 : 참과 거짓이 모두 올바르게 분류된 확률 (정답을 맞춘 확률) =  $\frac{TP + TN}{TP + FP + FN + TN}$

정밀도 : 참으로 예측한 것 중 실제 참이 맞는 경우의 비율 =  $\frac{TP}{TP+FP}$

민감도 : 실제 참인 경우 중 참으로 예측된 경우의 비율 =  $\frac{TP}{TP+FN}$

재현율 : 민감도와 동일한 의미의 지표 =  $\frac{TP}{TP+FN}$

특이도 : 실제 거짓인 경우 중 거짓으로 예측된 것의 비율 =  $\frac{TN}{FP+TN}$

		Condition(실제)	
		Positive	Negative
Prediction (예측)	Positive	TP(True Positive)	FP(False Positive)
	Negative	FN(False Negative)	TN(True Negative)

[그림 18] 컨퓨전 매트릭스(Confusion Matrix)

## 4. 분석 모델 소개

### 4.4 분류기의 성능평가

- 정밀도와 재현율의 Trade - Off 관계

		Condition(실제)	
		Positive	Negative
Prediction (예측)	Positive	TP(True Positive)	FP(False Positive)
	Negative	FN(False Negative)	TN(True Negative)

[그림 18] 컨퓨전 매트릭스(Confusion Matrix)

#### 1. Threshold를 낮출 경우

분류기 → True 예측 응답 많아진다. (TP, FP 증가)

False 예측 응답 줄어든다. (FN 감소)

$$\text{재현율}(\text{Recall}) = \frac{TP}{TP + FN}$$

#### 2. Threshold를 높일 경우

분류기 → False 예측 응답 많아진다. (TN, FN 증가)

$$\text{정밀도}(\text{Precision}) = \frac{TP}{TP + FP}$$

높은 신뢰 점수를 갖는 입력에만 True라고 응답 -> FP감소, TP증가

→ 정밀도와 재현율 동시에 높이는 것은 어려움

## 4. 분석 모델 소개

### 4.4 분류기의 성능평가

**F1 Score** : 정밀도와 재현율의 조화 평균을 계산한 값으로 정밀도와 재현율을 하나의 값으로 보기 위한 값

$$F_1 = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$

## 4. 분석 모델 소개

### 4.4 분류기의 성능평가

- Trade off 관계에 따라 어떤 임계값을 설정해야 되는지?

Ex1) 방법 시스템

임계값을 낮춰 시스템이 자주 작동되어 도둑을 확실히 잡는 것이 좋음 -> 높은 재현율, 낮은 정밀도

Ex2) 인터넷 광고 차단 시스템

유용한 광고를 몇 개 못보더라도 불법 광고는 확실히 차단하는 것이 좋음 -> 낮은 재현율, 높은 정밀도

Ex3) 환자의 질병 검사기

높은 재현율, 높은 정밀도 But, 만들기 매우 어려움

## 5. 결과 분석

실제 불량 횟수: 28969  
표준데이터를 이용한 불량 검출 횟수: 26707  
실제 불량 중 표준데이터를 이용하여 불량을 정확히 검출한 비율: 92.0 %

[그림 85] 실제 불량인 경우를 표준데이터를 이용하여 예측한 결과에 대한 정리

실제 불량 횟수: 28969  
평균 임계치보다 클 경우 불량을 검출한 횟수: 13017  
실제 불량 중 평균 임계치를 이용하여 불량을 정확히 검출한 비율: 44.93 %

[그림 90] 실제 y값이 평균 데이터보다 클 경우를 불량으로 하여 예측한 결과에 대한 정리

실제 불량 횟수: 28969  
평균 임계치보다 작을 경우 불량을 검출한 횟수: 15952  
실제 불량 중 평균 임계치를 이용하여 불량을 정확히 검출한 비율: 55.07 %

[그림 93] 실제 y값이 평균 데이터보다 작을 경우를 불량으로 하여 예측한 결과에 대한 정리

- ➡ K-최근접 이웃 회귀를 이용하여 표준데이터와 상한, 하한 임계치를 추출
- ➡ 시간의 흐름과 센서 데이터의 패턴에 따라 유연한 적용이 가능하다.