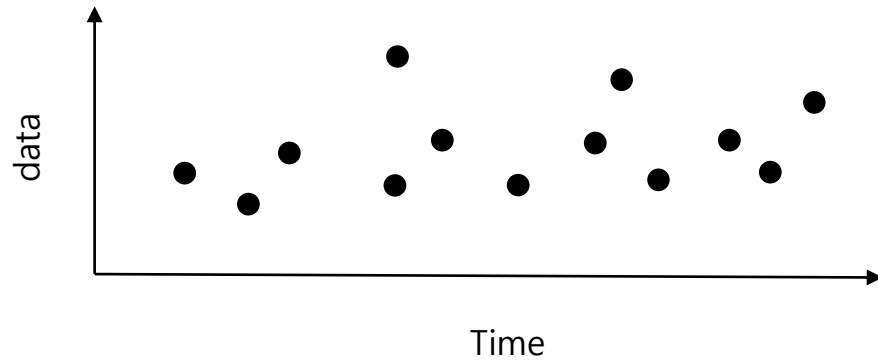


# **Anomaly Detection**

# 데이터의 이상 판단

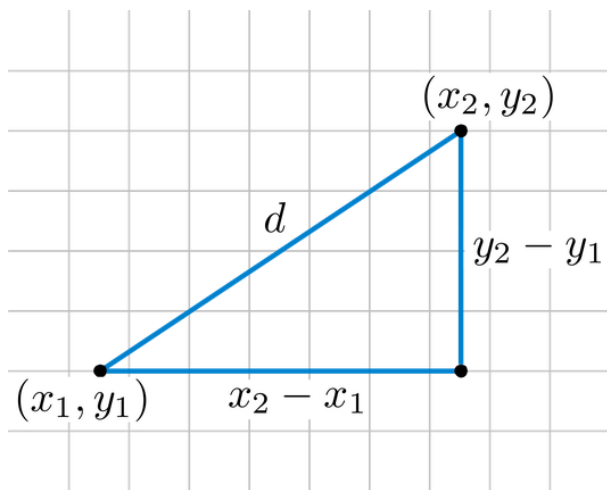


데이터의 이상을 어떻게 판단할 것인가?

**Distance**

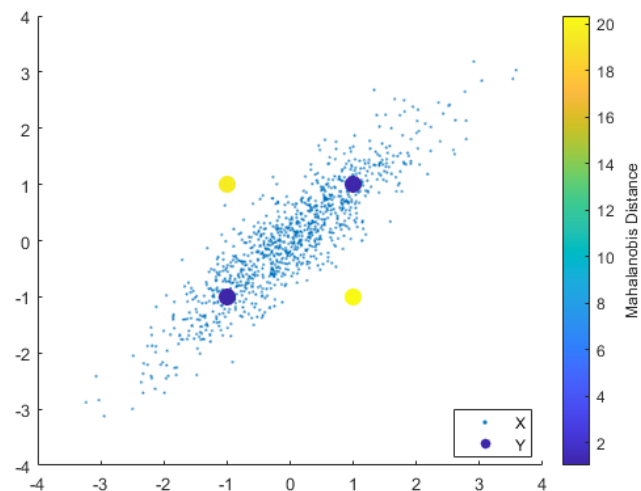
# Distance

[유클리디안 거리]



$$\sqrt{(X - \bar{X})(X - \bar{X})^T}$$

[마할라노비스 거리]



$$\sqrt{(X - \bar{X})\Sigma^{-1}(X - \bar{X})^T}$$

대표적으로 거리가 짧으면 정상, 거리가 멀면 이상인 rule을 이용해서 판단이 가능하다.

# 이상 판단의 척도

이상을 판단하는 기준이 거리만 있는가?

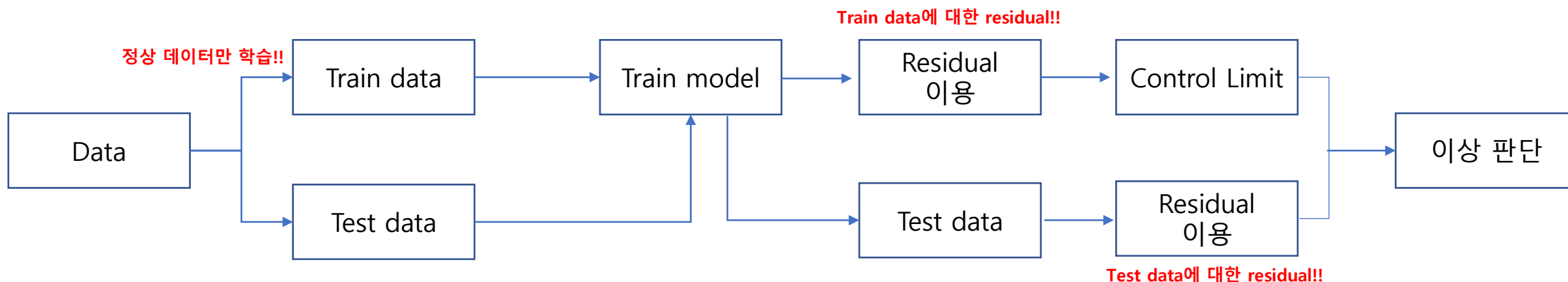
# **MSET**

## **Multivariate State Estimation Technique**

# MSET

Distance처럼 이상감지를 판단하는 하나의 사상 **모델이 아님!!**

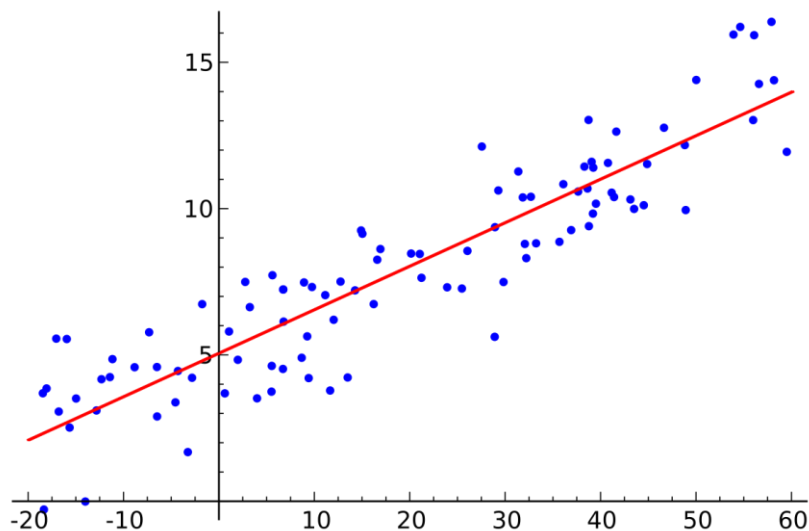
MSET 프로세스



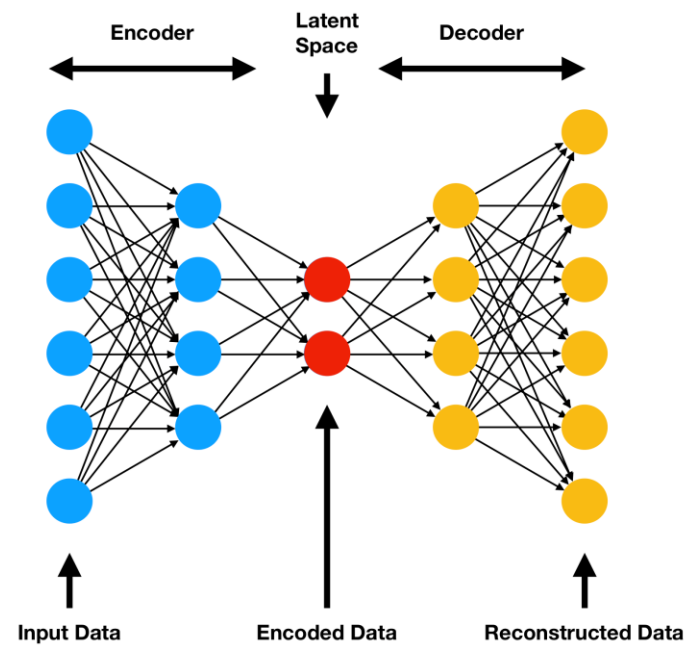
정상 데이터만을 학습 시켜서 정상데이터에 대한 잔차를 구하고 이 정상 데이터에 대한 잔차의 기준선을 통계적 기법을 이용해서 정한 후 이 기준을 넘어가면 이상으로 판단하겠다!!

# MSET 모델

[Linear Regression]



[AutoEncoder]

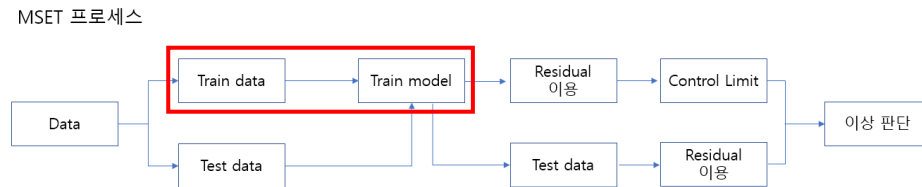


잔차를 이용한다는건 **머신러닝 모델**을 사용한다는 것  
모델은 사용자가 쓰고 싶은 모델 사용!!



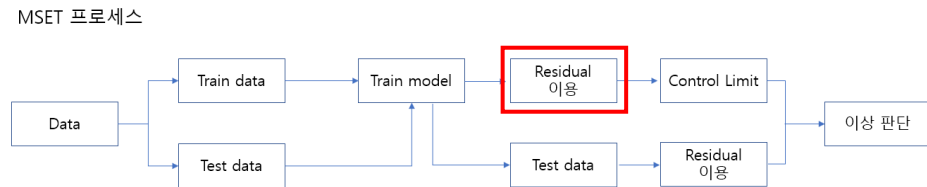
# MSET 프로세스

Step1: 정상 데이터만을 이용해서 모델을 학습시킵니다.



## 정상 데이터만으로 모델 학습!!

# MSET 프로세스



Step2: train 데이터(정상 데이터)를 모델에 넣어서 잔차 값을 구함.

## MSET에서 잔차를 사용하는 이유

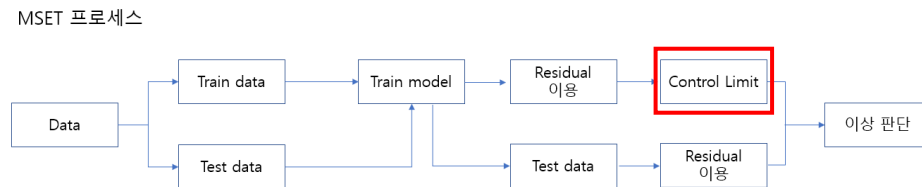
정상데이터로만 모델을 학습시켰기 실제 데이터가 정상일 경우 잔차 값이 0에 근접하게 됨.

정상데이터로 학습 시킨 모델에서 정상 데이터를 넣었으니 당연히 그 차이는 0에 근접하다!!

정상 데이터로 학습한 모델에 이상 데이터가 들어갔을 경우 잔차 값은 양의 방향 혹은 음의 방향으로 커지게 됨.

즉 잔차를 이용해서 0을 기준으로 값이 튀는 부분을 잡아내겠다는 의미가 됨.

# MSET 프로세스



Step3: 정상 데이터의 잔차에 대한 Control Limit값을 구함.

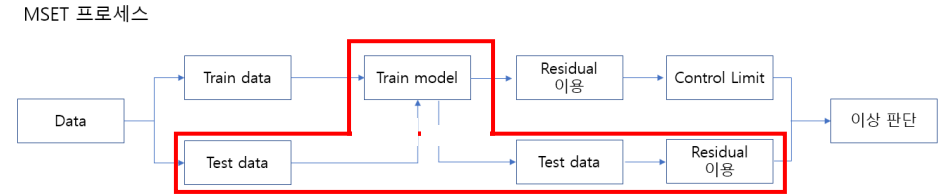
이상을 판단하기 위해서는 정상까지의 범위가 어디인지 정해 줘야한다.

주로 통계적 기법을 이용해서 구함.

ex) Bootstrap, F-distribution

빅데이터분석실무 때 했던 수업 내용을 다시 기억해보자!!

# MSET 프로세스

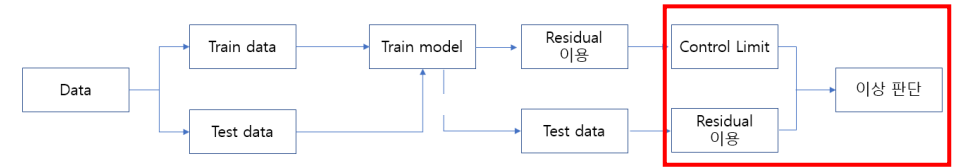


Step4: 정상을 판단하는 모델에 실제 test 데이터를 넣고 잔차를 구함.

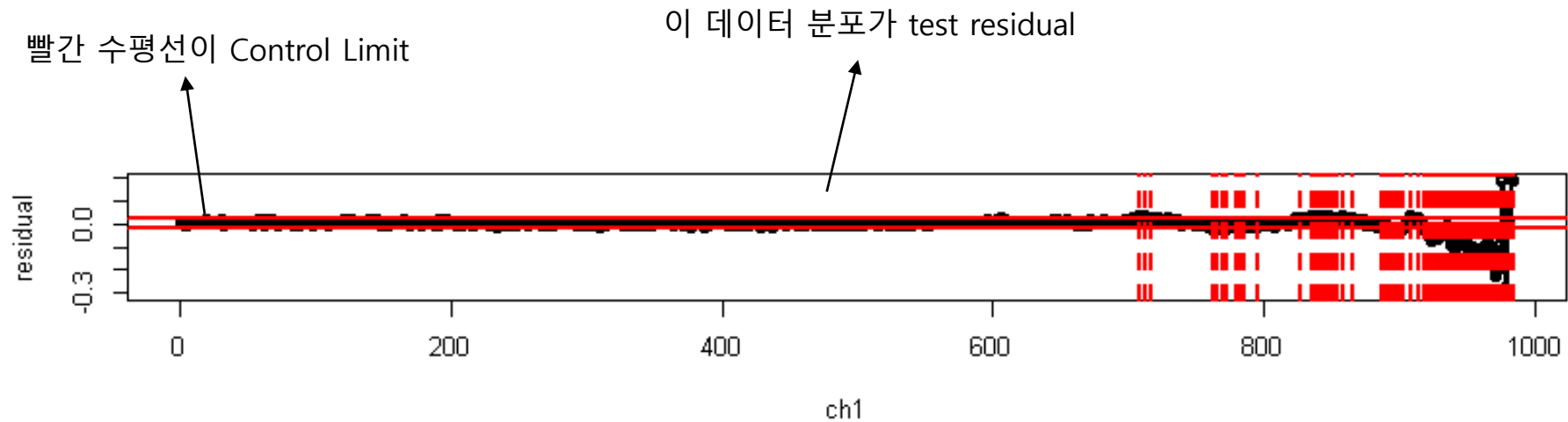
**정상 모델에 test 데이터를 넣어서 이상 판단을 하기 위한 준비를 하자!!**

# MSET 프로세스

MSET 프로세스



Step5: 정상 데이터 잔차의 Control Limit값 비교 후 이상 판단



# MSET 요약정리

## MSET Step

정상 데이터만을 학습 시켜서 모델을 만들고 정상데이터를 넣은 모델에 대한 잔차를 구함.

정상 데이터 잔차에 대한 기준선을 구함.

Test 데이터를 정상 모델에 넣어서 Test 데이터에 대한 잔차 값을 구함.

정상데이터를 학습시킨 모델에서 구한 기준값을 넘으면 이상 아니면 정상으로 판단한다.

## MSET 특징

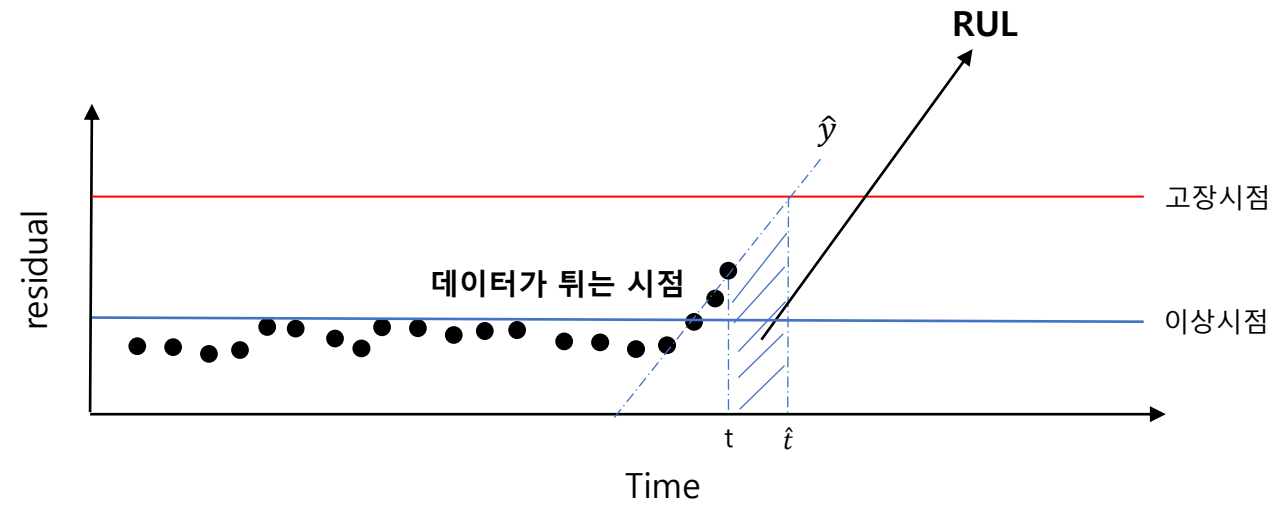
0을 기준으로 값이 튀는 지점을 보기 위해서 잔차를 사용한다.

모델 같은 경우는 사용자가 쓰고 싶은 모델을 사용.

**RUL**

**Remain Useful Life**

# RUL





# RUL

## RUL이 왜 중요한가?

- 기계의 잔여수명을 미리 예측함으로써 유지보수 시기를 미리 정해 비용을 아낄 수 있다.
- 발화 위험이 있는 제품에 대한 잔여수명을 미리 예측함으로써 안전성 확보를 할 수 있음.

# RUL 모델

- Similarity Model

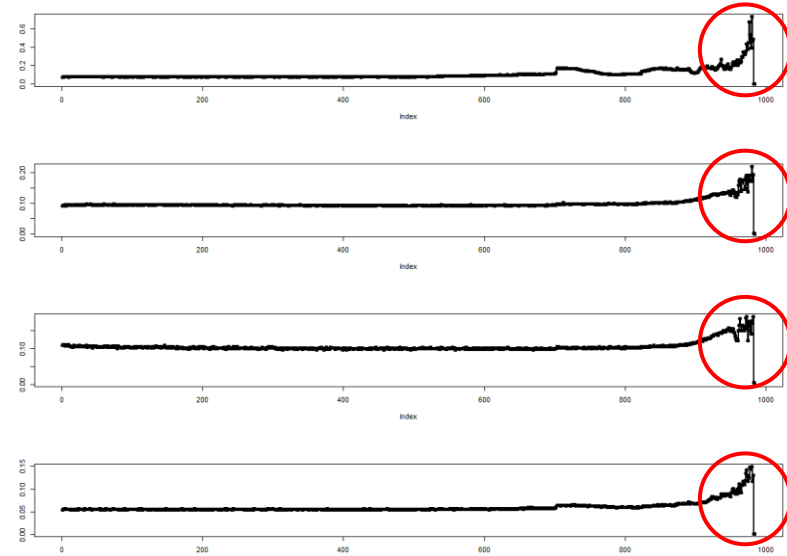
유사한 고장을 가진 데이터를 가지고 있는 경우

- Survival Model

비슷한 수명 데이터를 가지고 있는 경우

- Degradation Model

고장 시점에 대한 임계값을 가지고 있는 경우



[베어링 데이터: 고장 난 시점에서 데이터 수집 종료]

# Degradation Model

$R(t_i) = [r_{1i} + r_{2i} + r_{3i} \cdots + r_{ni}]^T$  residual을 의미

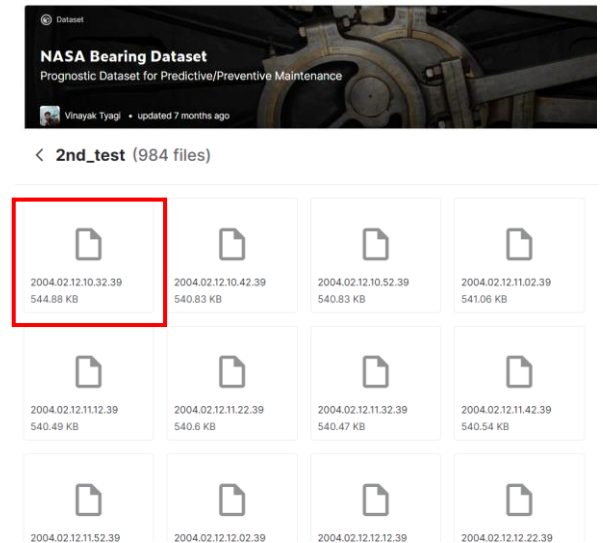
센서를 통한 진동 데이터 같은 경우 1초에 20,000개 이상의 데이터 값이 측정됨.

$R(t_1) = [r_{11} + r_{21} + r_{31} \cdots + r_{n1}]^T$

$R(t_1)$ 은 첫번째 시점에서 발생하는 모든 잔차를 의미

$\|R(t_i)\| = \sqrt{r_{1i}^2 + r_{2i}^2 + r_{3i}^2 \cdots r_{ni}^2}$  Residual의 Euclidean Norm

첫번째 시점에서 발생하는 20,000개의 잔차를 하나의 크기로 표현 한 것 RMS변환 했다고 생각하면 됨



# Degradation Model

Accumulated Degradation

$$Ac\_De(t_k) = \sum_{i=1}^k ||R(t_i)||$$

잔차들을 **누적합**으로 표현한 모델

스트레스가 많이 쌓일수록 기계는 고장이 남.

여기서 잔차를 기계가 받는 스트레스로 보는 것.

**누적된 스트레스를 이용해서 고장을 판단하겠다!!** 라는 의미

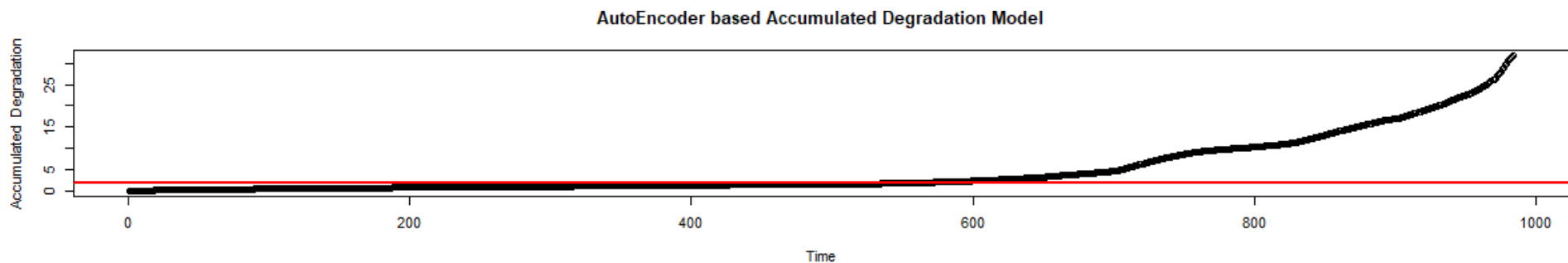
# Degradation Model

Accumulated Degradation이 갖는 문제점은 단조증가 하는 특성을 띠

단조증가: 지속적으로 값이 커지는 성질

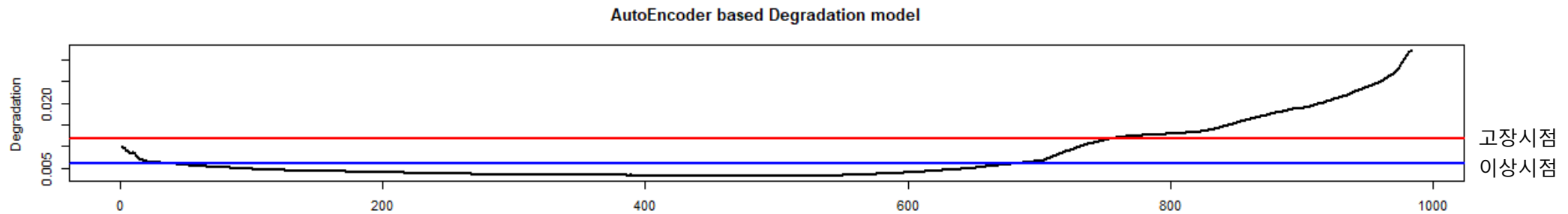
실제로 기계에 이상이 없어도 단조증가 하는 성질 때문에 이상이 있는지 없는지 구분하기 어려움이 있음.

-> 데이터의 개수 k만큼 나누어서 단조 증가하는 문제점을 해결



# Degradation Model

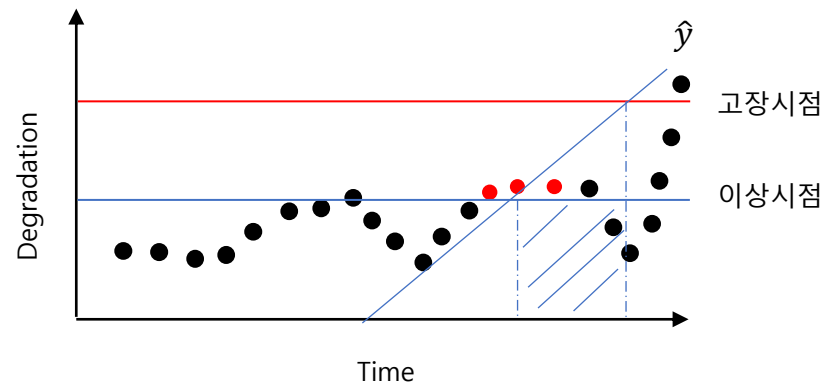
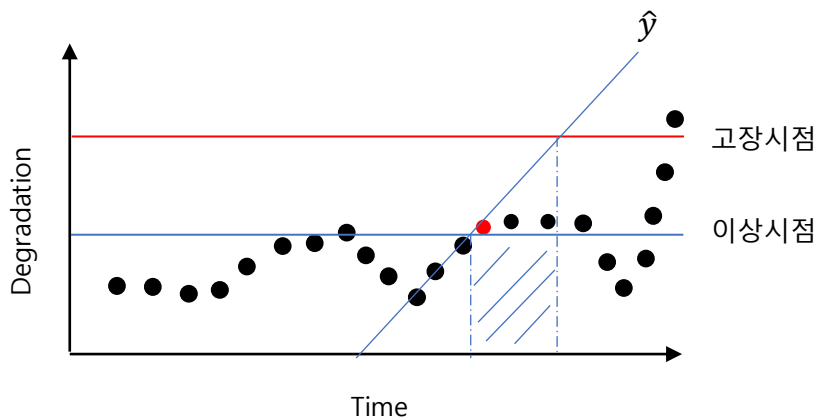
$$\text{Degradation} = \text{De}(t_k) = \frac{1}{k} \sum_{i=1}^k ||R(t_i)||$$



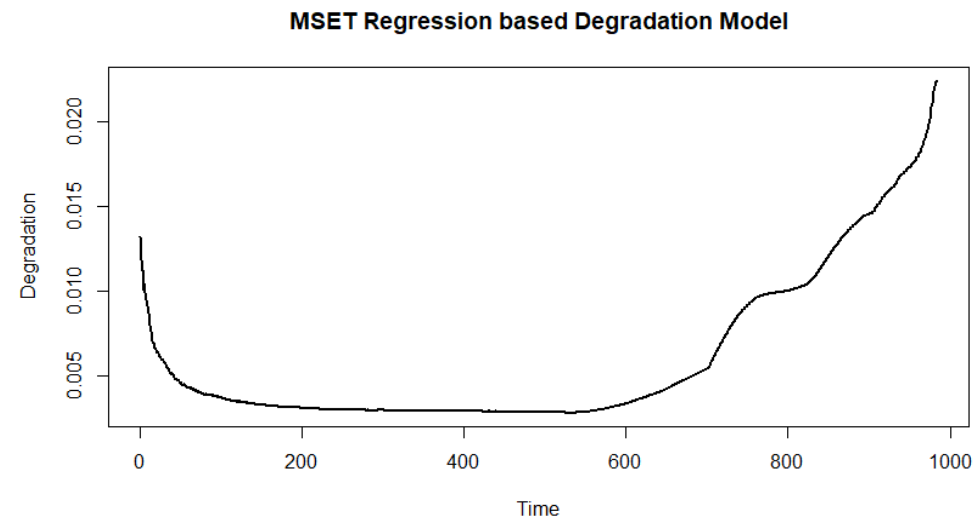
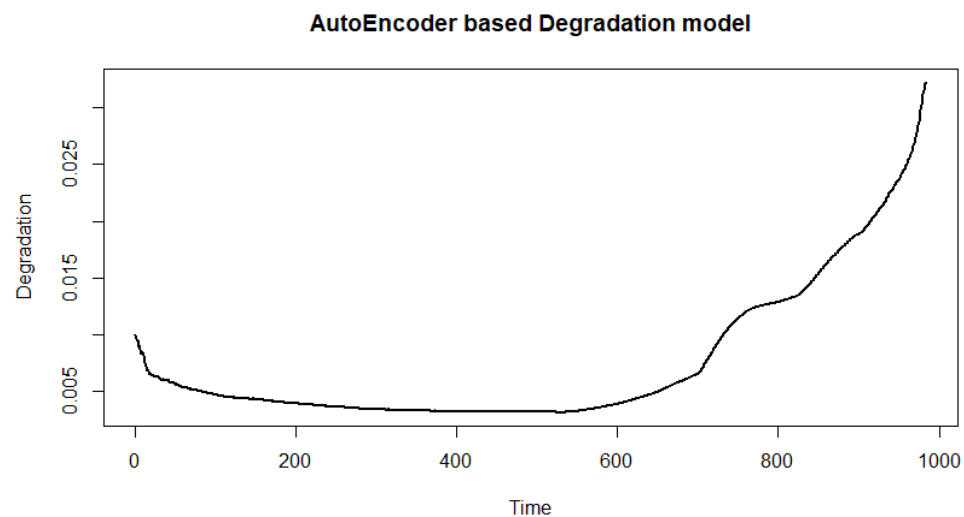
# RUL 추정을 위한 회귀모델

## 연구할 만한 내용

- 데이터가 튀는 시점 이후 연속적으로 이상치가 계속 발생했을 경우  
이상 데이터가 많이 발생 했다는 건 기계가 스트레스를 많이 받았다는 의미이고,  
기계가 스트레스를 많이 받을수록 RUL이 줄어드는 현상을 반영하고 싶다는 의미  
(모니터링에서 모든 이상 데이터는 기계의 수명에 영향을 준다는 가정이 필요)



# Base Model 별 Degradation Model



MSET Regression base Degradation 모델과 MSET AutoEncoder base Degradation 모델 비교 결과  
큰 차이가 나지는 않음.



# RUL 요약정리

## RUL 개념

RUL은 데이터가 튀는 시점이 발생했을 때 회귀선을 그어 고장시점과 맞닿는 부분에 대한 X값, 즉 고장시점을 예측한다.

RUL은 데이터가 튀는 시점과 예측한 고장시점의 차이를 의미한다.

## Degradation Model

고장 시점에 대한 임계값을 가지고 있는 경우

$$De(t_k) = \frac{1}{k} \sum_{i=1}^k ||R(t_i)||$$

잔차들의 누적합으로 표현한 모델

잔차를 기계가 받는 스트레스로 보며, 누적된 스트레스를 이용해서 고장을 판단하겠다는 의미를 가짐.

누적합은 단조증가 하는 경향을 띠고 있기 때문에 데이터의 개수만큼 나누어 준 모델.

# 참고문헌

Cheng, S., & Pecht, M. (2007). Multivariate State Estimation Technique for Remaining Useful Life Prediction of Electronic Products. AAAI Fall Symposium: Artificial Intelligence for Prognostics.

[http://www.phm.or.kr/industry/KSPHM\\_industry.pdf](http://www.phm.or.kr/industry/KSPHM_industry.pdf)

**Thank You**

---