

# UPTOWN



AI를 이용한 연삭 휠 수명 관리 및 교체시기 예측 솔루션



# UpTown

업타운은 빅데이터 솔루션 유통 및 개발의 선두주자로서  
다수의 운영 및 보안 관제 구축을 통해 특화된 노하우를 보유한  
빅데이터 기반 인공지능 서비스 전문 기업입니다.



## 고객 맞춤형

빅데이터 분석 및 보안 분야에서  
고객의 환경에 맞는  
서비스를 제공하며  
다양한 구축 경험을 가지고 있습니다.



## 자체 개발 플랫폼

자체 개발한 빅데이터 인프라와 인공지능 분석  
플랫폼을 기반으로한  
데이터분석정보 서비스를 통해 고객사에 맞는  
빅데이터 가치를 제공합니다.



# CONTENTS

프로젝트 개요 01

홈페이지 설명 03

기대효과 05

데이터 설명 02

프로젝트 세부내용 보고 04

프로젝트 소감 06



A black and white photograph of a person's hand holding a pen, poised to write on a document. The hand is positioned over a laptop keyboard, which is partially visible. The background is dark and out of focus, showing some papers and the person's arm. A semi-transparent white rectangular box is overlaid on the right side of the image, containing the text.

**01**

**프로젝트 개요**

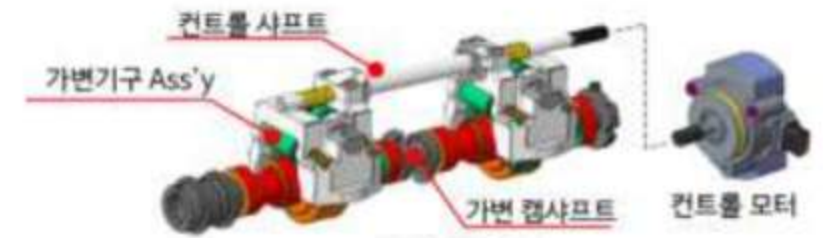


# 의뢰기업 : 서진캠

자동차 부품 생산 업체로 단순한 공정 개선이나 품질 향상을 넘어  
신기술, 신공법, 신제품 연구개발에 많은 투자와 회사의 역량을 집중하고 있으며,  
이를 토대로 시장을 선도하는 혁신제품개발과 최고수준의 품질 확보 실현중인 기업  
<주요고객> 현대자동차, 기아, 현대글로비스, 현대위아, GM KOREA, 현대모비스, 쌍용자동차



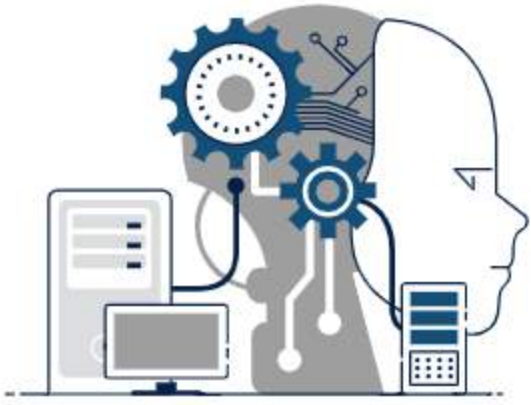
## <생산 품목>



	엔진명	구분	비고
V6		3기통	
V8		4기통	
V10		5기통	



## 의뢰 기업이 겪고 있던 상황



기본적인  
스마트 공장  
구축 준비 완료



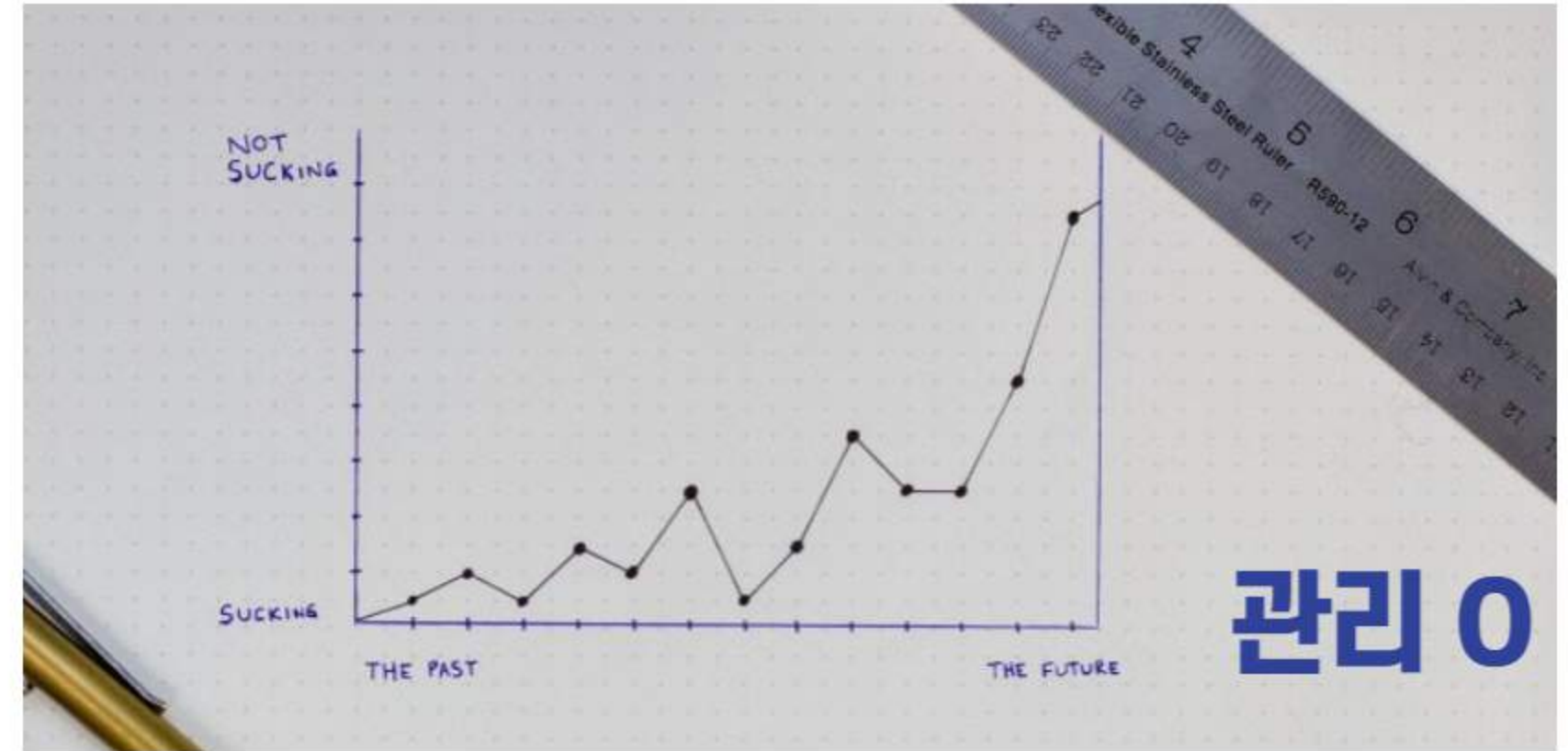
수집되는  
빅데이터  
활용 X



품질 향상에  
대한 요구사항  
증가

“ 설비에서 발생하는 데이터를 SI와 융합하여 ”  
설비의 건전성을 파악하고자 함

# 정기적인 연삭휠 드레싱과 컨디셔닝 작업의 필요성



① 불필요한 관리 지출(고비용)

② 장비 스펀들 같은

중요 부품에 필요 이상의 스트레스

① 연삭 버닝 현상 감소

② 연삭휠의 생산성과 효율성 극대화

③ 연삭 품질 향상



1  
연삭 휠  
수명 관리

2  
교체 시기  
예측

3  
설비 운영  
효율화

4  
제품 고도화

연삭기 센서 데이터 및 MES 분석을 통한  
연삭 휠 수명관리 및 교체시기 예측 솔루션



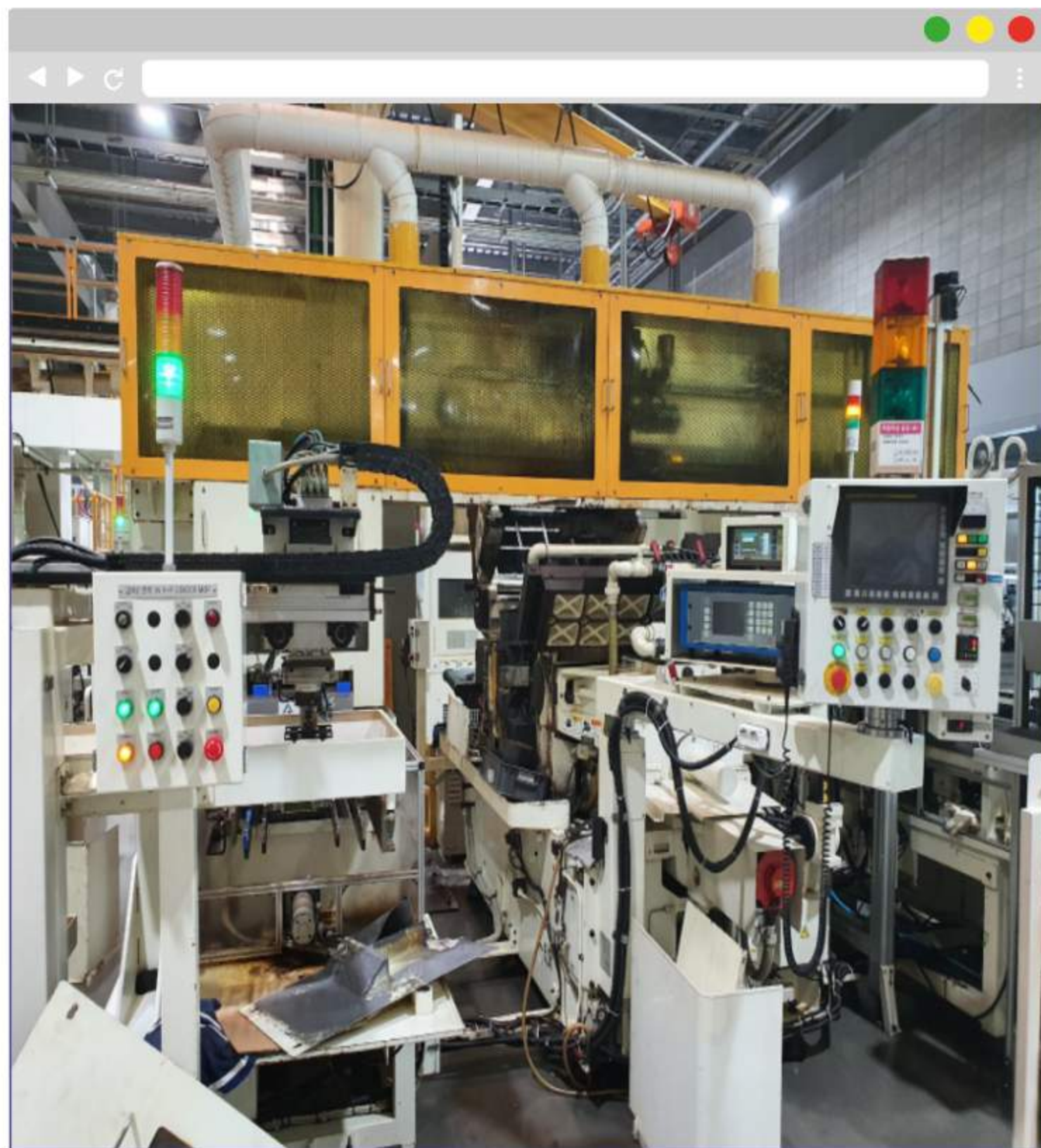


02

데이터 설명



## 데이터 수집 대상 장비



▶ 센서가 부착 될 연삭기 장비



▶ 센서 데이터가 수집 되는 Edge PC



# 센서 데이터 수집 구성도

A/D Converter  
센서 신호 변환 장치

Edge Computing  
센서 데이터 저장 및 필터링

전류센서

진동센서



Database  
데이터 저장



# 수집 데이터 리스트

## Edge PC 수집 데이터



- ▶ 전류 센서 1개
- ▶ 진동 센서 1개
- ▶ EdgePC 상태

## 연삭휠 데이터



- ▶ 휠 잔량
- ▶ 드레싱 주기
- ▶ 동작 상태

## MES 데이터



- ▶ 생산 실적
- ▶ 재고 현황
- ▶ 불량 판정





03

홈페이지 설명

# 제작 홈페이지 상세설명

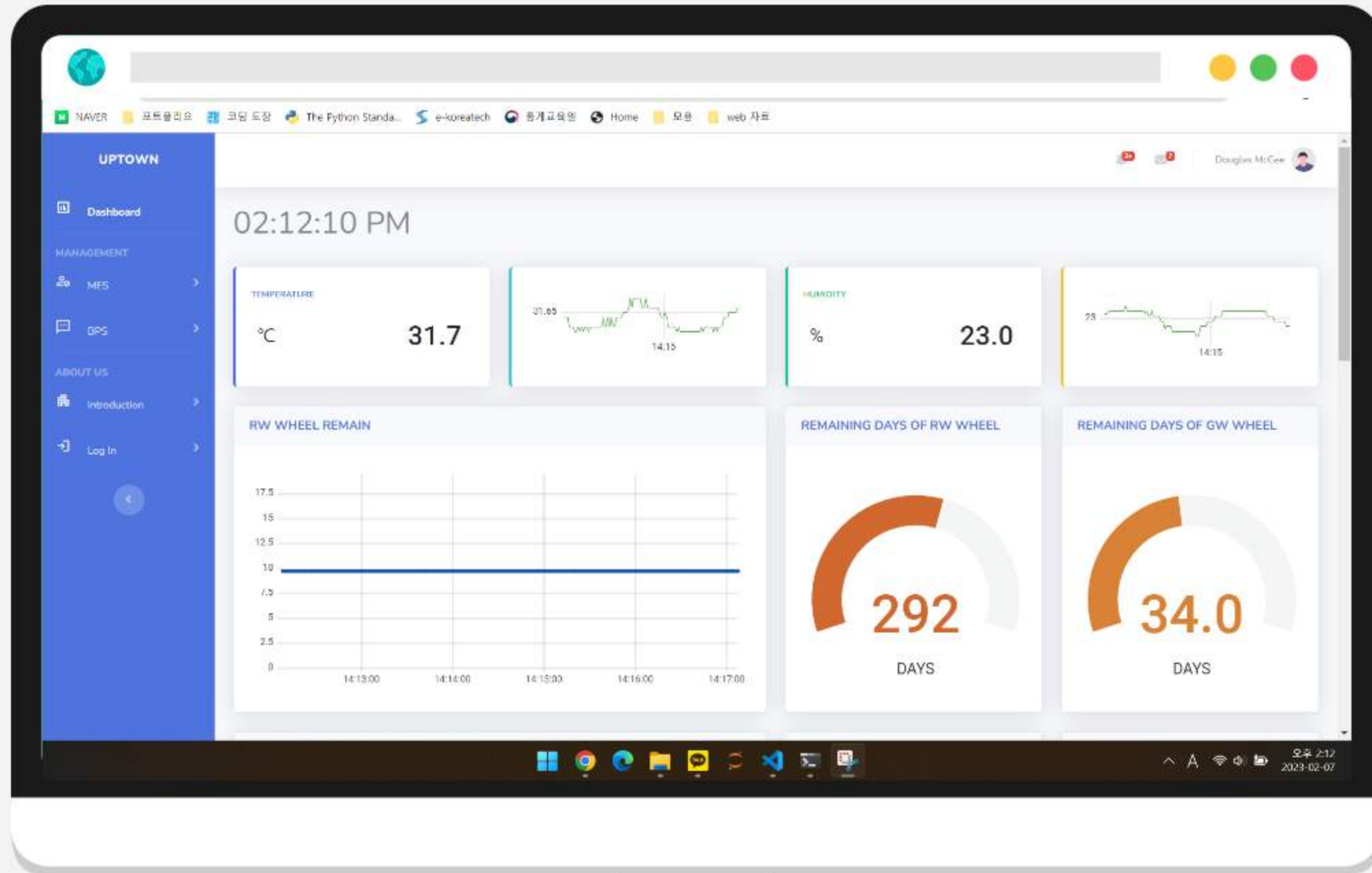
: UI 및 기능 설명

## 01. Dashboard

- 전반적인 설비 기능 시각화
- 온도/습도 시스템 상태, RW 휠 잔량 상태, RW/GW 휠 잔량 예측 잔량에 따른 교체 시기 예측 표시

## 02. MES

- MES 페이지로서, 생산현황, 재고 개수, 불량품 개수 관련 데이터 표 표시



## 03. BPS

- 누적 출입고량, 불량품 개수, 기업의 비전과 미션, 장기 목표 리스트, 부서별 요구 사항 입력 폼

## 04. Introduction

- 기업 소개, 팀원 소개, 찾아오는 길, 문의 사항

## 05. Log in

- 로그인, 회원가입, 마이페이지 창

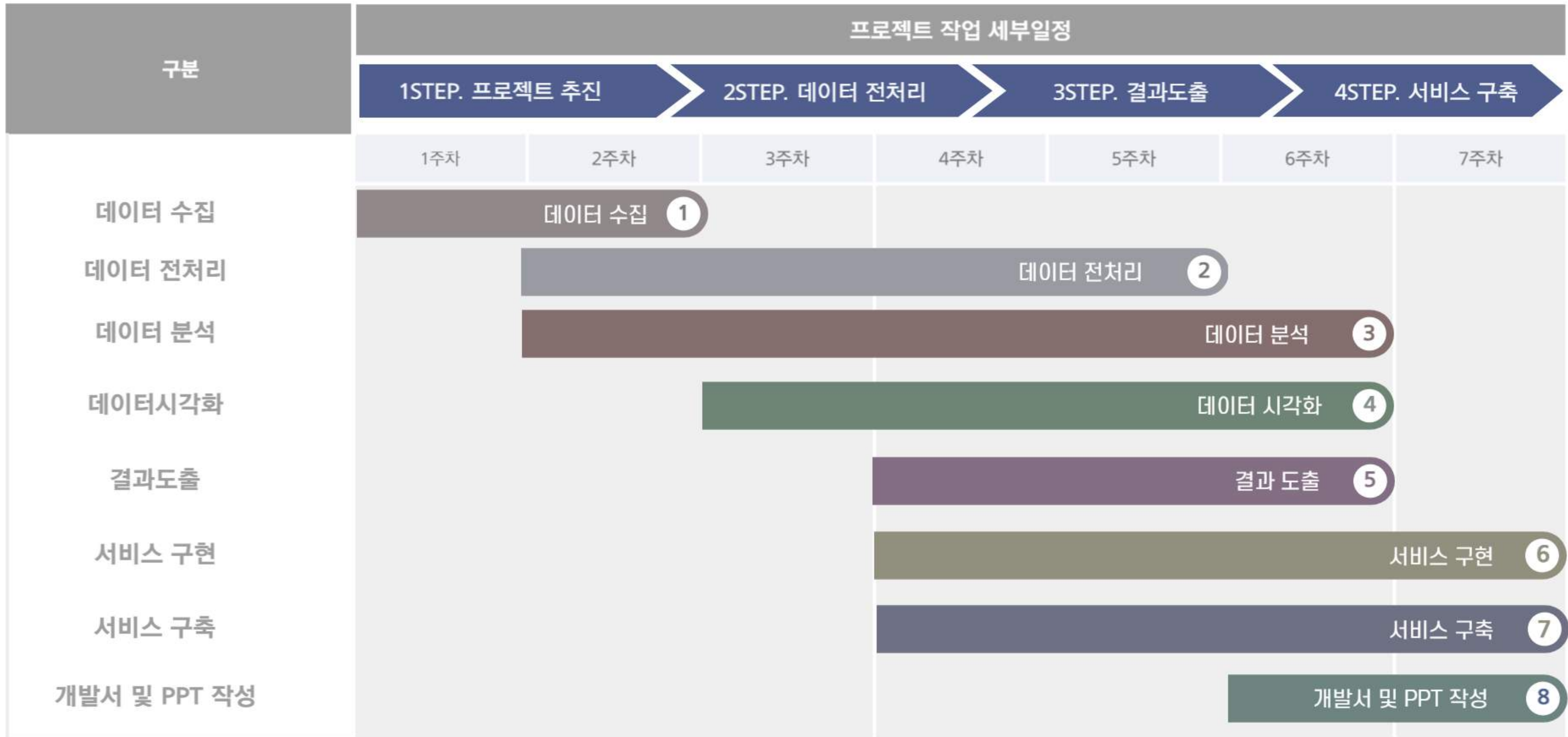




04

프로젝트 세부내용 보고

# 프로젝트 스케줄





# 개발환경

01

OS (운영체제) 버전

Windows 10  
Windows 11

03

데이터 베이스

influxdb 1.8.10

05

데이터 분석 버전

Python 3.9.7

- ▷ 전처리 : Numpy 1.23.0, Pandas 1.4.3, FFT, PCA, t-SNE 알고리즘
- ▷ 시각화 : Grafana 9.3.2
- ▷ 머신러닝 : Keras 2.9.0, Tensorflow 2.9.2, Scikit-Learn 1.0.2
- ▷ 딥러닝 : ARIMA, LSTM

02

개발툴

Colab pro+

04

형상 관리

Git, GitHub

06

웹 개발 프레임워크

Django 4.1.5

- ▷ 프론트엔드 : HTML5, JS, CSS

# 무엇을 어떻게 예측해야하나?

무엇을?

AI 알고리즘을 사용하여  
월 잔량을 예측해보자!

어떻게?

7일을 기준으로  
모델을 학습시키고  
다음 1일을 예측!

무엇을?

잔량을 실시간으로 탐지 후  
교체시기를 예측해보자!

어떻게?

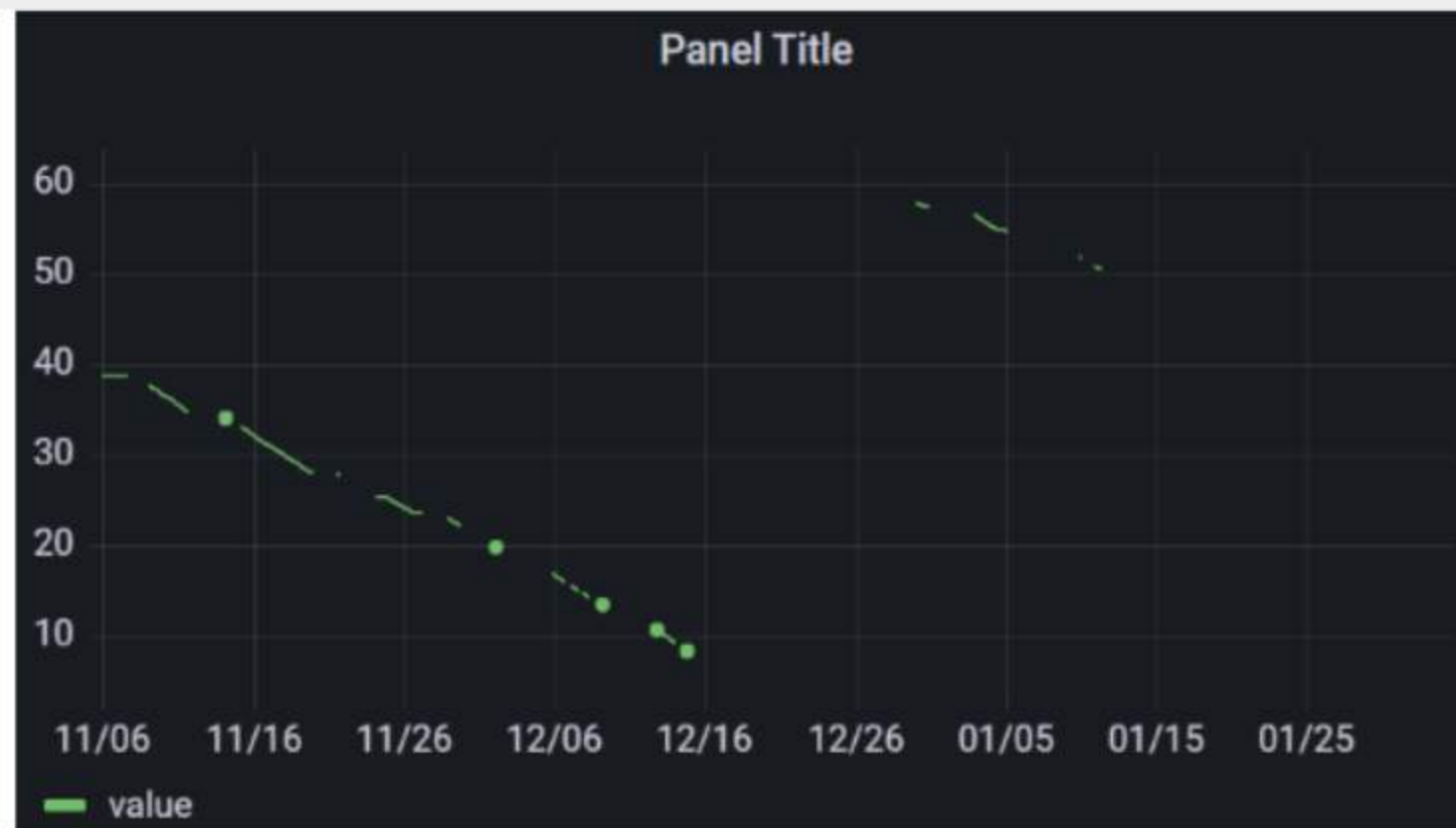
7일을 기준으로  
월 감소량을 계산하고  
교체 시기 예측!





## 이상치 및 결측치 처리

### GW\_WHEEL\_REMAIN



### RW\_WHEEL\_REMAIN



## 이상치 및 결측치 처리

Panel Title



Panel Title





## 이상치 처리

- ✓ 7일보다 짧은 주기로 월 잔량의 감소율을 계산하니  
예상 교체 날짜가 크게 변경되어서 감소율을 계산하는 기간을 7일로 결정
- ✓ 감소율이 평균보다 크게 나타나는 부분은 이상치로 판단 후 이전 값으로 대체



## 결측치 처리

- ✓ 'time'열에 결측치 존재함.
- ✓ 통신 장애로 인해 데이터가 수집되지 않은 날이 많았음.
- ✓ date\_range를 통해 'time'열의 결측값을 채워줌.
- ✓ 'time'열의 새로운 행이 생기면서 다른 열의 결측치가 새로 생기게 됨.
- ✓ 교체된 날이 기록되지 않아서 이전, 이후 값을 토대로 전체 기울기를 계산하여 교체 지점을 예측하여 지정
- ✓ 'RW\_WHEEL\_REMAIN' 과 'GW\_WHEEL\_REMAIN'열의 결측치를 선형 보간법을 통해 보간

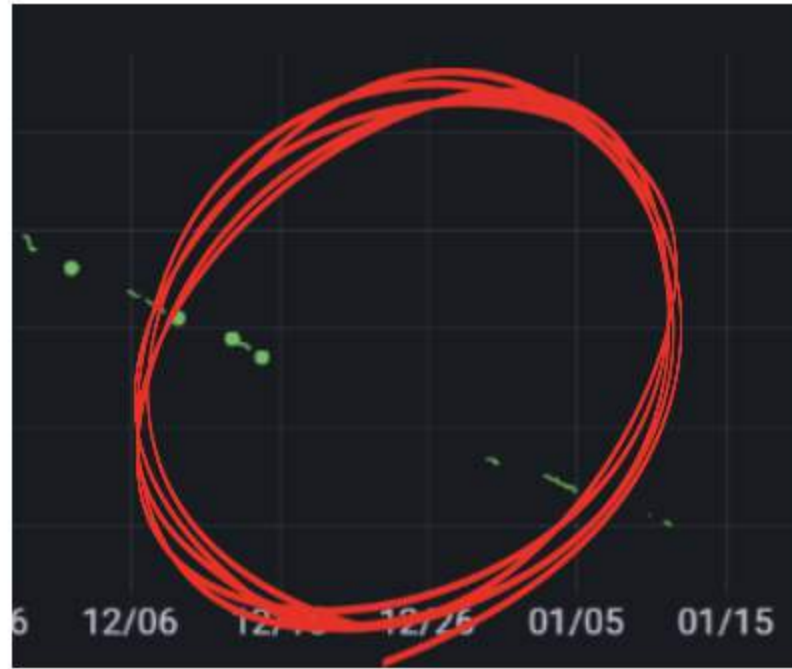




## 시계열 자료 기울기 문제 해결과정 (1)



월 교체 데이터 삭제



사라진 구간 :

2022년 12월 14일 오후 ~ 12월 30일 오전

$$\frac{y_2 - y_1}{x_2 - x_1} = \text{기울기}$$

$$\frac{8.44 - 39.62}{40} = -0.78$$

## 계산을 통해 보간한 그래프



- 노란 그래프 : 전처리 전
- 녹색 그래프 : 전처리 후



## 계산을 통해 보간한 그래프

GW 전처리 전후



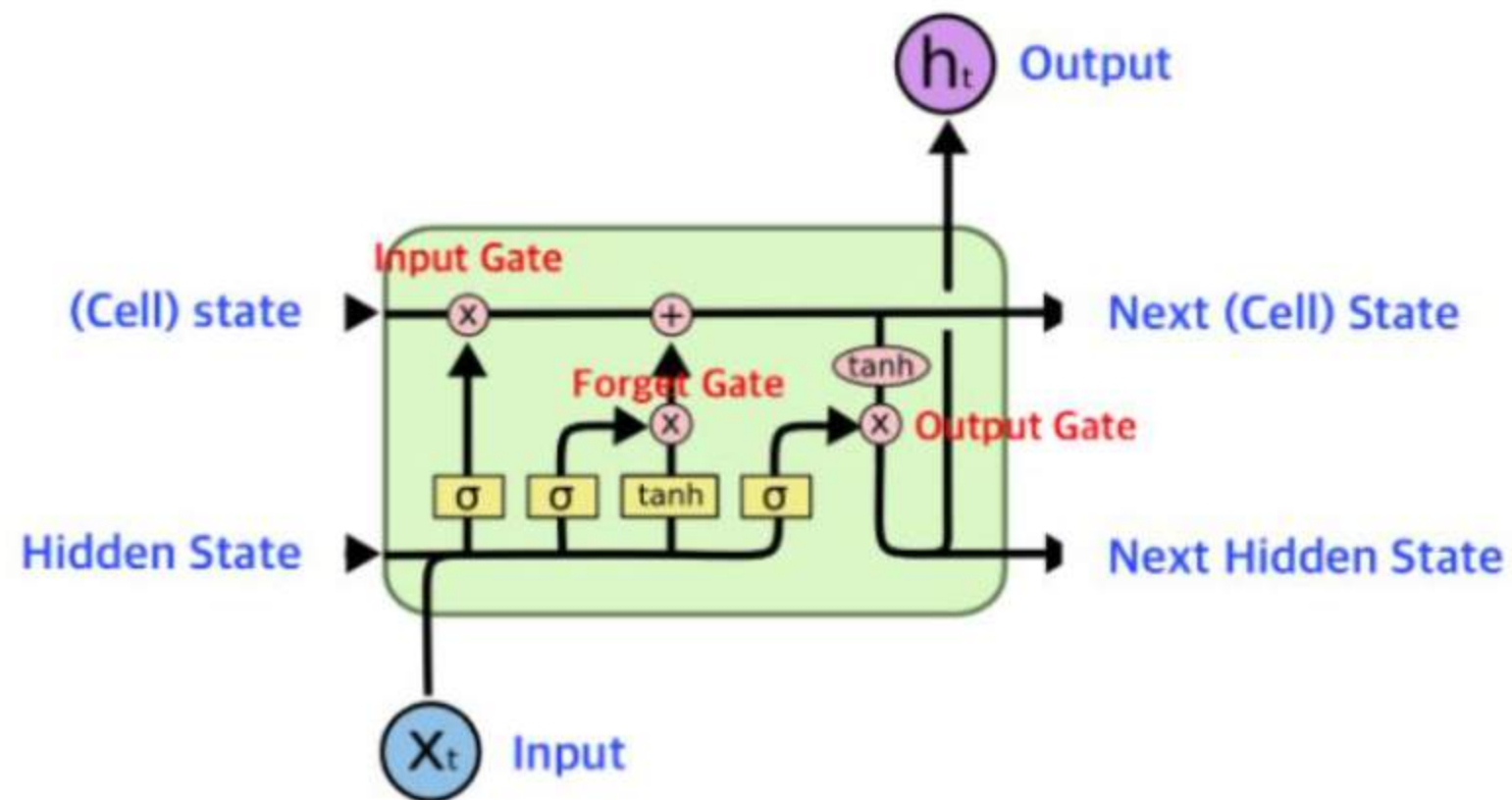
RW 전처리 전후



- 노란 그래프 : 전처리 전
- 녹색 그래프 : 전처리 후

# 사용 알고리즘

## LSTM



순환 신경망 기법 중 하나로 셀, 입력 게이트,  
출력 게이트, 망각 게이트를 이용해  
기존 순환 신경망의 문제점을 방지한 알고리즘

## ARIMA

$$ARIMA(p, d, q)$$

AR 모형 차수

차분

MA 모형 차수

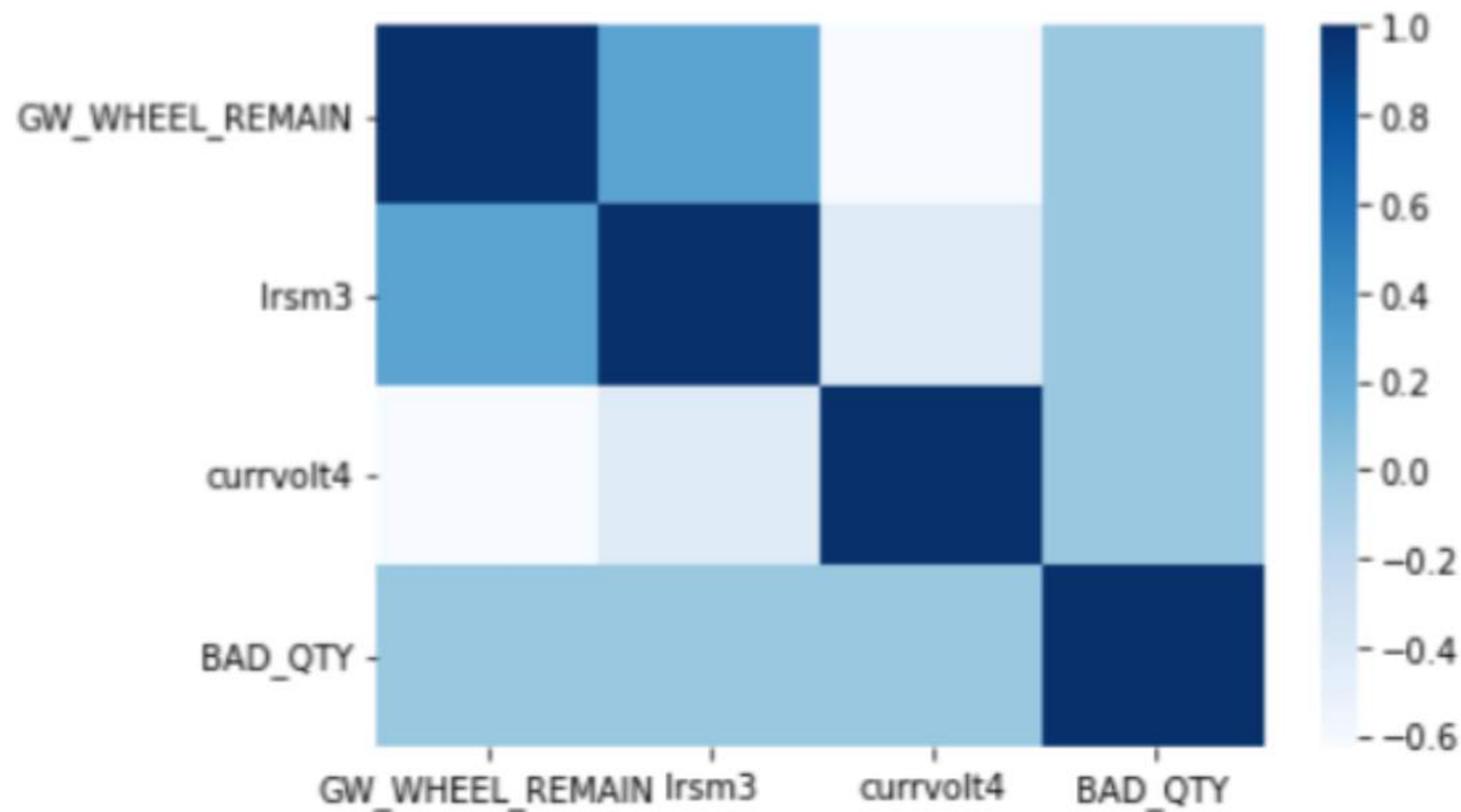
ARIMA는 차분, 변환을 통해  
AR, MA, ARMA로 정상화

- $p=0$ 이면 IMA( $d, q$ ) ->  $d$ 번 차분하면 MA( $q$ )
- $d=0$ 이면 ARMA( $p, q$ ) -> 정상성 만족
- $q=0$ 이면 ARI( $p, d$ ) ->  $d$ 번 차분하면 AR( $p$ )

자기회귀와 이동평균을 모두 고려하여  
시계열 데이터 기반 분석 기법으로  
현재 값을 과거 값과 과거 예측값의 오차를  
통해 설명하며 예측하는 알고리즘

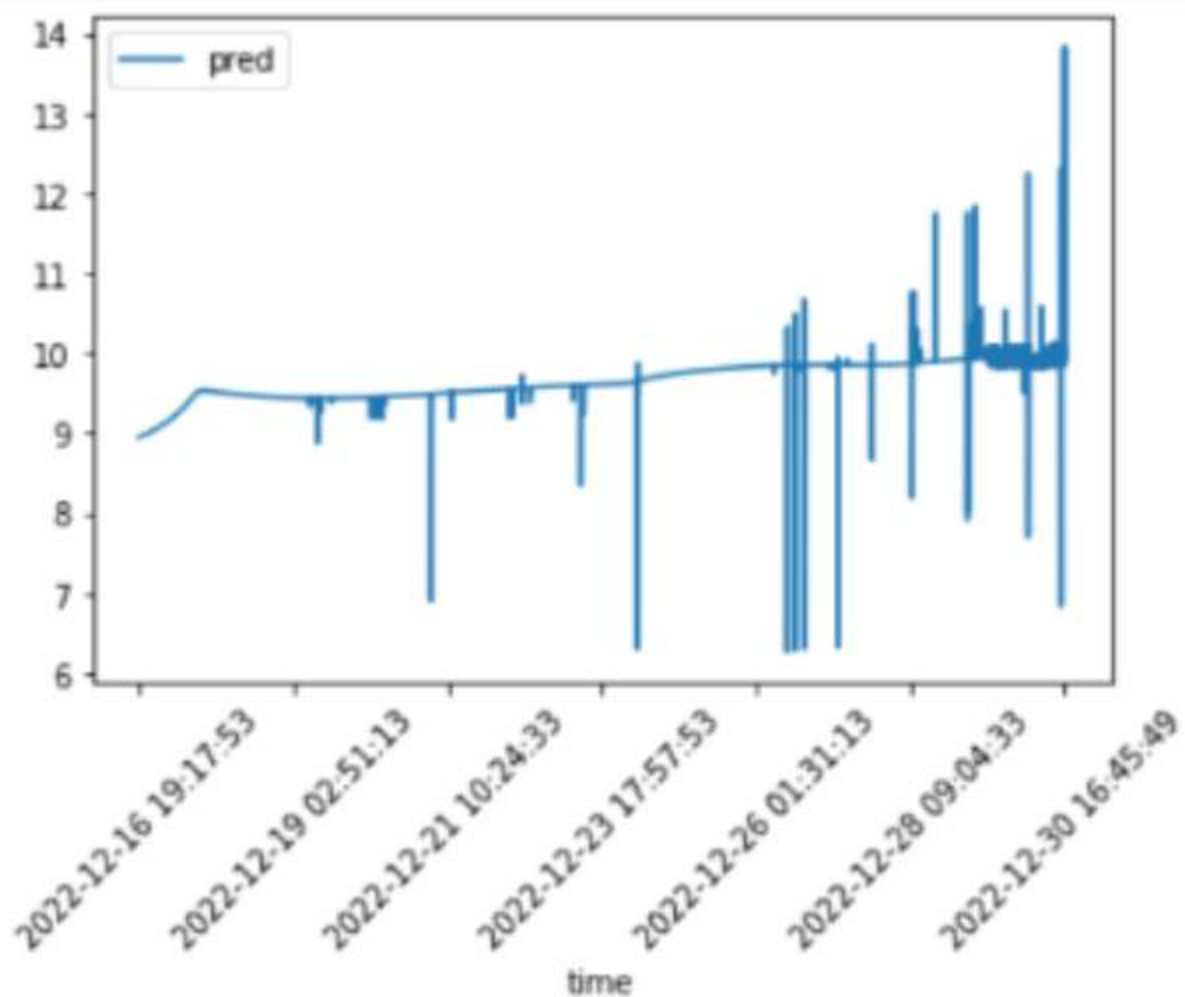


## 데이터 전처리 Feature 탐색

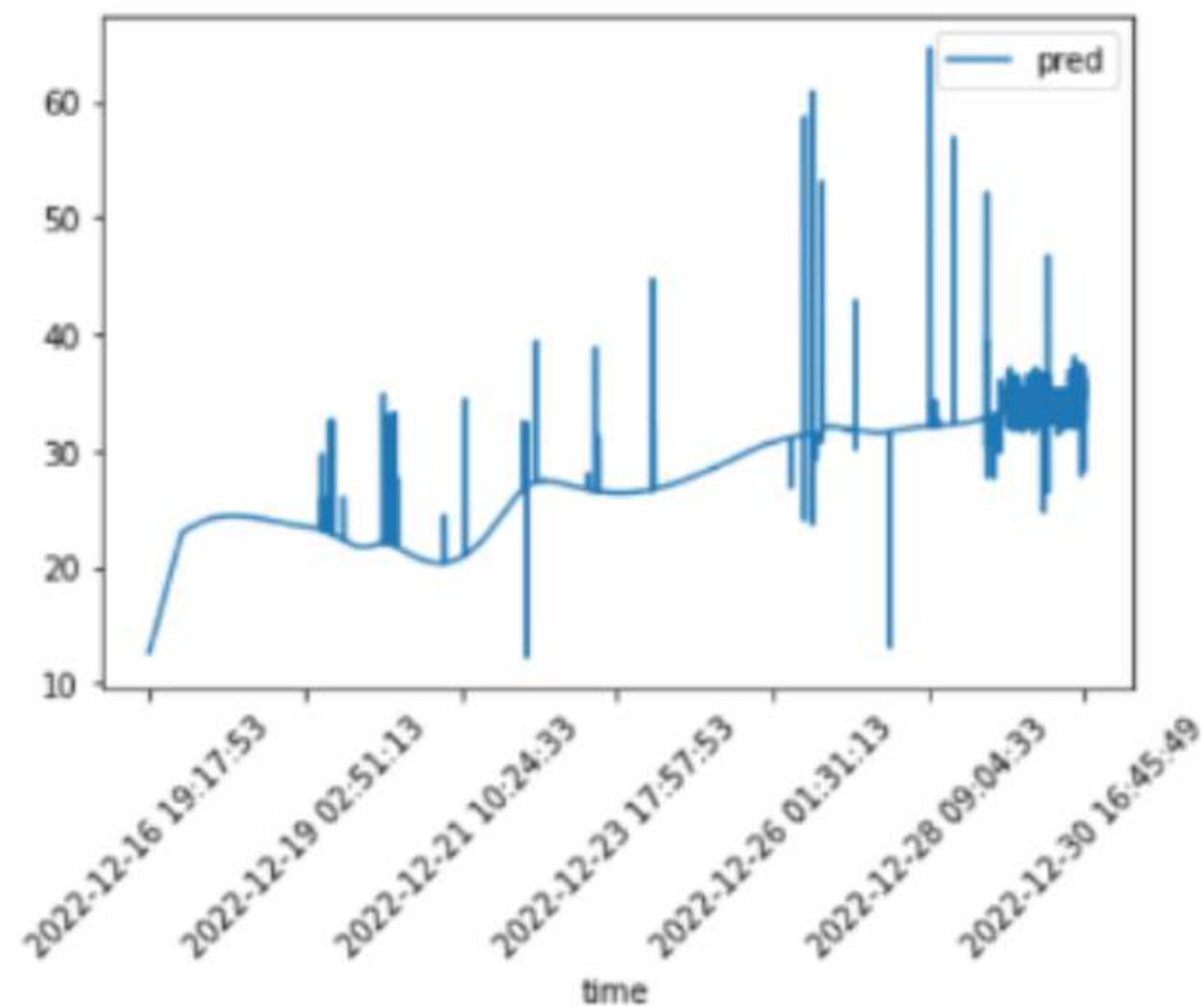


예측해야할 열인  
GW\_WHEEL\_REMAIN과  
RW\_WHEEL\_REMAIN과  
상관관계를 보이는 열이  
없음을 알 수 있다.

# LSTM



RW\_WHEEL\_REMAIN 예측 그래프



GW\_WHEEL\_REMAIN 예측 그래프

LSTM을 사용하고자 하였으나 독립변수들간의 상관성이 낮았기 때문에 예측이 제대로 되지 않음



# ARIMA

Performing stepwise search to minimize aic

ARIMA(2,1,2)(0,0,0)[0] intercept : AIC=-10250185.215, Time=212.11 sec

ARIMA(0,1,0)(0,0,0)[0] intercept : AIC=-10223672.280, Time=82.45 sec

ARIMA(1,1,0)(0,0,0)[0] intercept : AIC=-10245366.942, Time=104.53 sec

ARIMA(0,1,1)(0,0,0)[0] intercept : AIC=-10241692.866, Time=324.58 sec

ARIMA(0,1,0)(0,0,0)[0] : AIC=-10223143.811, Time=25.26 sec

ARIMA(1,1,2)(0,0,0)[0] intercept : AIC=-10250114.441, Time=190.16 sec

ARIMA(2,1,1)(0,0,0)[0] intercept : AIC=-10250054.021, Time=176.38 sec

ARIMA(3,1,2)(0,0,0)[0] intercept : AIC=-10250204.681, Time=237.42 sec

ARIMA(3,1,1)(0,0,0)[0] intercept : AIC=-10250110.039, Time=181.26 sec

ARIMA(4,1,2)(0,0,0)[0] intercept : AIC=-10250193.457, Time=302.96 sec

ARIMA(3,1,3)(0,0,0)[0] intercept : AIC=-10250202.716, Time=301.47 sec

ARIMA(2,1,3)(0,0,0)[0] intercept : AIC=-10250204.575, Time=273.29 sec

ARIMA(4,1,1)(0,0,0)[0] intercept : AIC=-10250173.053, Time=266.81 sec

ARIMA(4,1,3)(0,0,0)[0] intercept : AIC=-10250200.567, Time=332.71 sec

ARIMA(3,1,2)(0,0,0)[0] : AIC=-10250078.442, Time=314.53 sec

Best model: ARIMA(3,1,2)(0,0,0)[0] intercept

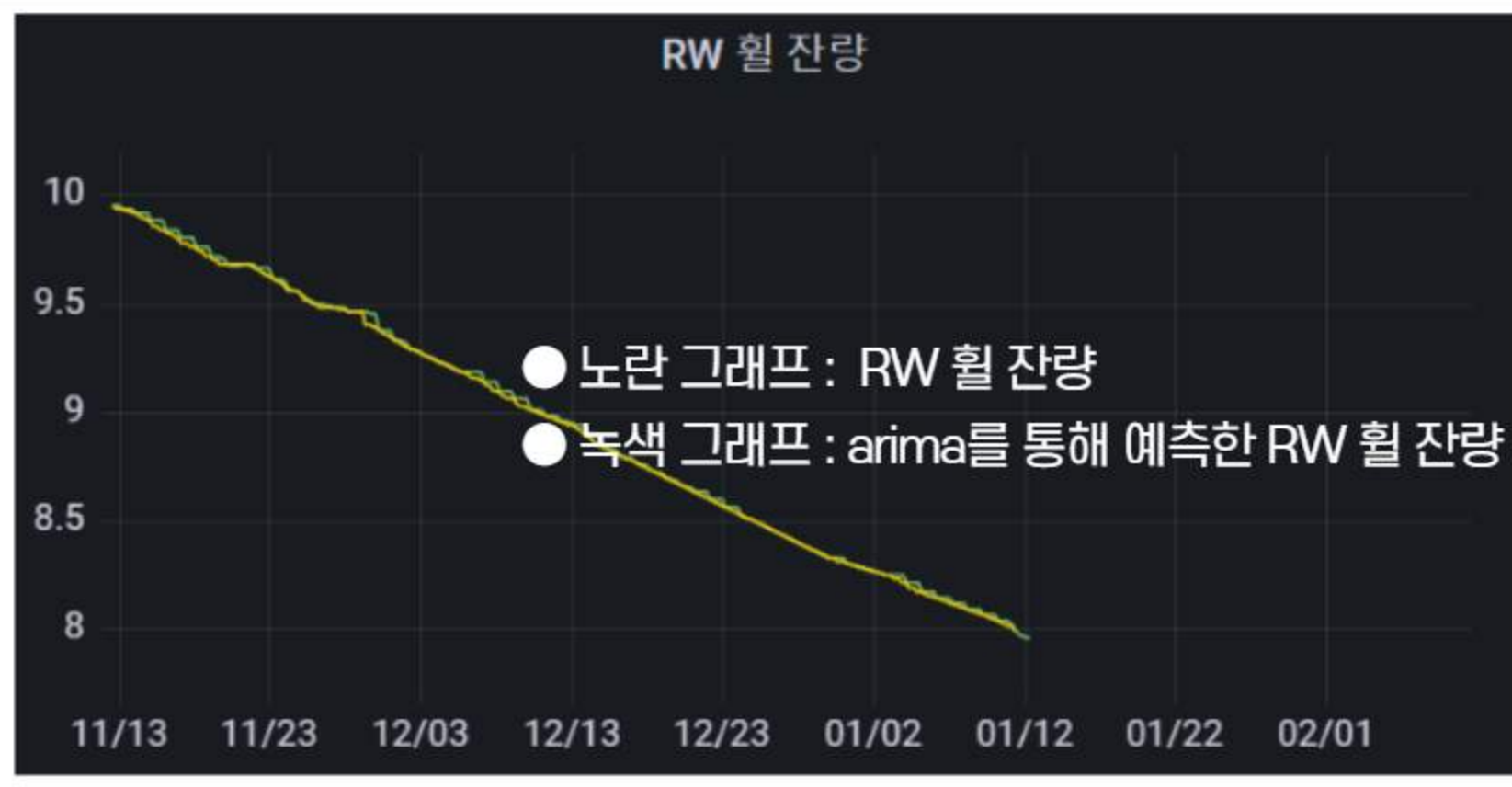
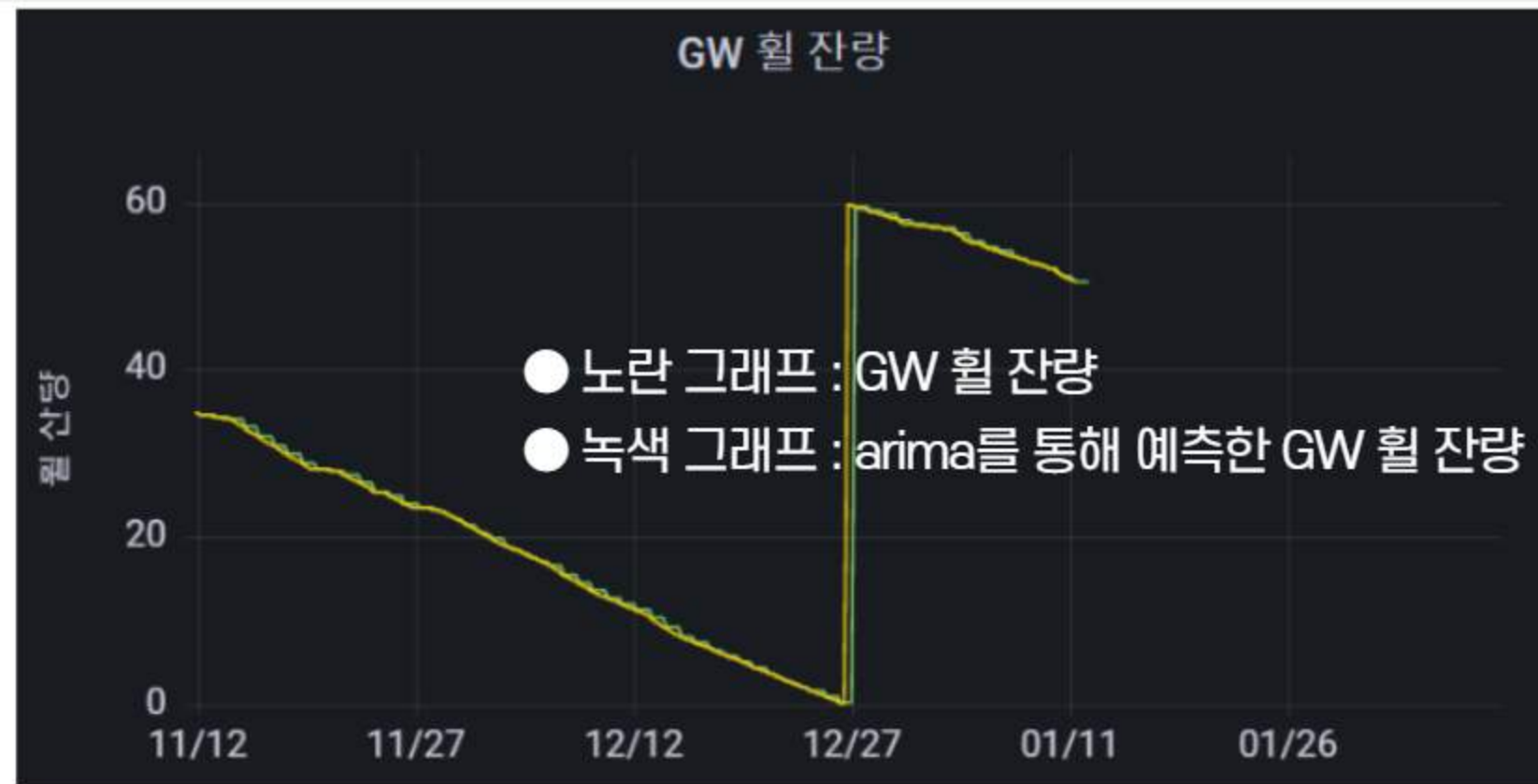
Total fit time: 3325.943 seconds

ARIMA의 차수 p,d,q 값을 얻기 위해  
auto arima를 실행해서 best model을 구했다.

최적 모델인 ARIMA(3,1,2)를 이용해서  
모델링을 진행했다.

정확도를 높이기 위해  
처음 7일을 이용해 모델을 fit 한 후,  
다음 1일을 forecast하는 방식을 이용했다.

# ARIMA



## 모형평가

구분	RW 월	GW 월
MAE	0.012	0.297
RMSE	0.0177	0.4060
MAPE	0.129	3.792
R <sup>2</sup>	0.999	0.998

제공받은 RW월 잔량 데이터에 교체된 구간에 대한 데이터가 누락되었기에 그 부분에 대한 데이터가 보완된다면 조금 더 나은 결과를 도출할 수 있을 것으로 기대된다.



# 무엇이 문제인가?

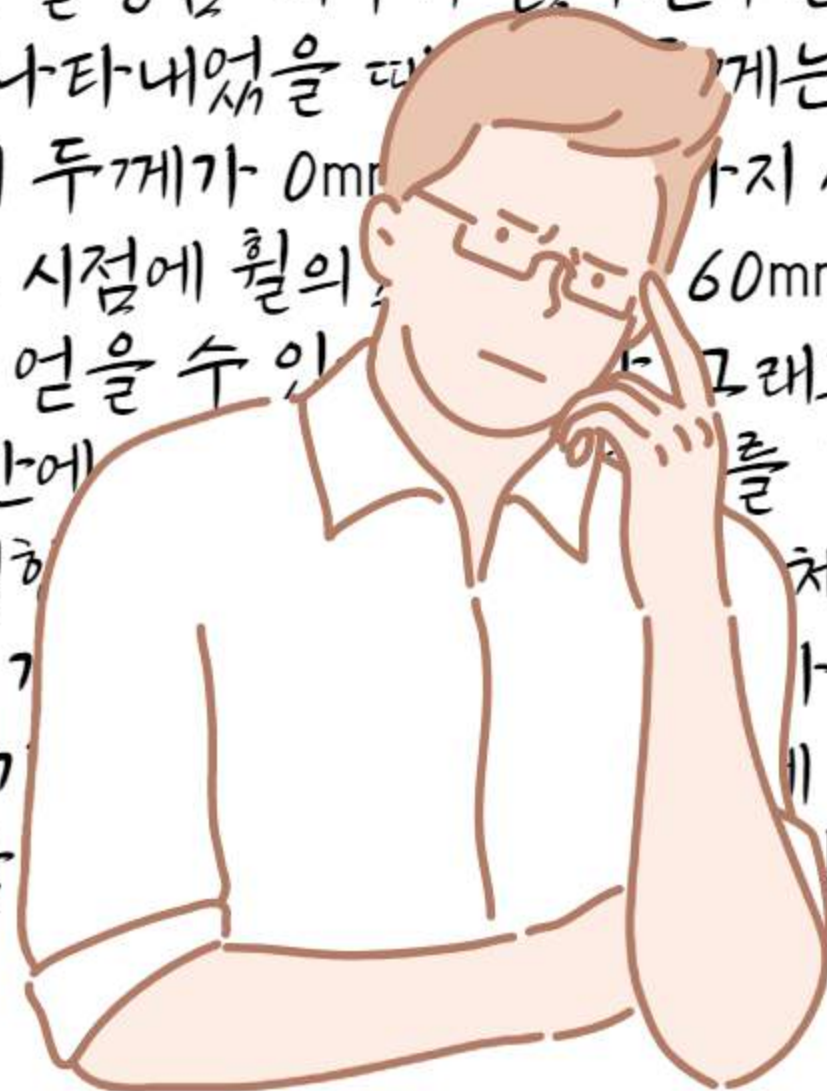
데이터  
부족

모델 사용  
불가능

상관성  
부족

기울기  
계산

보간법

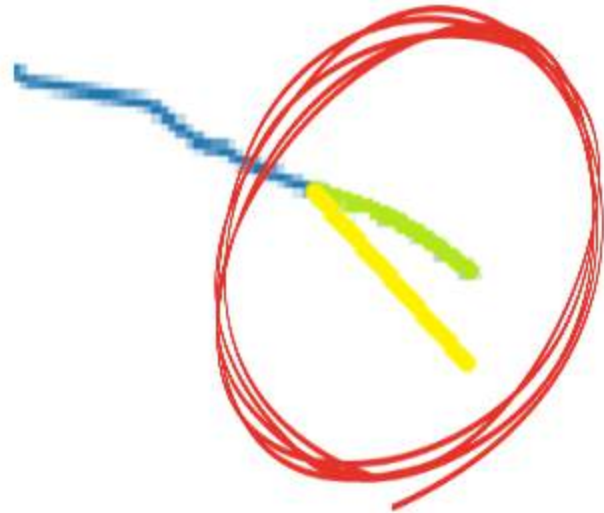


(실제 생각)

시계열 데이터를 받았을 때 빈 시간대가 너무 많아서 데이터를 사용할 수 없는 상황이었다. (빈 시간을 다 채울 경우 빈 행이 70%이상) 심지어 주기를 학습해야하는데 교체하는 시점의 데이터가 오류로 인해서 전부 사라진 상태였다. (사라진 구간: 2022-11-01 오후 ~ 12월 30일 오전) 월을 교체했을 때의 초기 두께를 알아낸 후 전체 그래프에서 데이터가 사라지기 전 기울기를 얻어냈다. 2022-11-04 (39.62mm) ~ 2022-12-14 (8.44mm) (총 40일)  $y2 - y1 / x2 - x1 = 기울기$   $8.44 - 39.62 / 40 = -0.78$ , 기울기는 -0.78이다. 따라서, 하루 당 기울기 감소량이 0.78이라는 결론을 도출했다. 구한 기울기를 빈 구간을 채우려고 계산하니 월의 두께가 0이 될 때까지 사용해야 빈 구간을 채울 수 있다는 결론이 나오는데 그 결론에서 얻을 수 있었던 것은 월 두께가 0이 될 때까지 사용할 시 불량품 개수가 증가할 수 있다는 것이었다. 월의 두께가 얇아질 때 불량품 개수가 많아진다면, 월의 교체 시점을 예측 가능할 것이다. 하지만 불량품 개수를 시계열 그래프로 나타내었을 때, 불량품 개수는 불량품 개수의 양의 증감을 나타내는데 불량품의 개수에 상관없이 월의 두께가 0mm가 될 때까지 사용된 월의 두께를 나타내지 않는다. (위 그래프)를 참고하여 시점에 월의 두께가 60mm를 초과하면 불량품 개수가 급격히 증가하는 것을 확인했다. 이를 바탕으로 기울기를 계산하고, 기울기를 현재 시점에서 일주일 뒤로 extrapolate 해서 계산하고, 그 결과로 일주일 뒤의 불량품 개수가 나타남 일주일 뒤까지 extrapolate 하자 월 교체시점에 급격한 기울기 변화가 발생하며 불량품 개수가 양수가 되면서 교체일수 예측에 문제가 발생한다. (남은 일수: -777) 기울기가 양수가 될 시점의 불량품 개수를 예측할 수 있다고 파악됨.



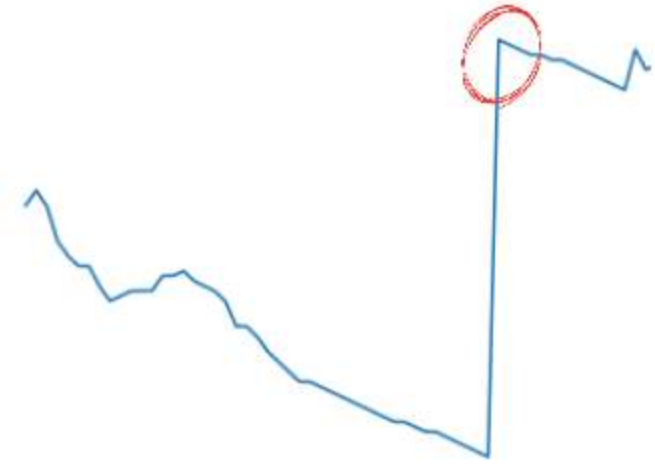
## 시계열 자료 기울기 문제 해결과정 (2)



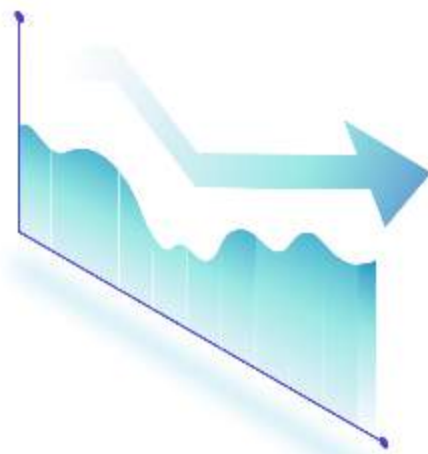
전체 기울기를 적용하니(-0.78)  
기울기가 맞지 않는 것을 확인



현재 시점에서  
일주일 뒤까지만  
들고와서 계산하도록 변경



월 교체시점에  
급격한 기울기 증가 발생



기울기가 양수가 되면서  
교체일수 예측에 문제가 발생  
(남은 일수 : - 777)

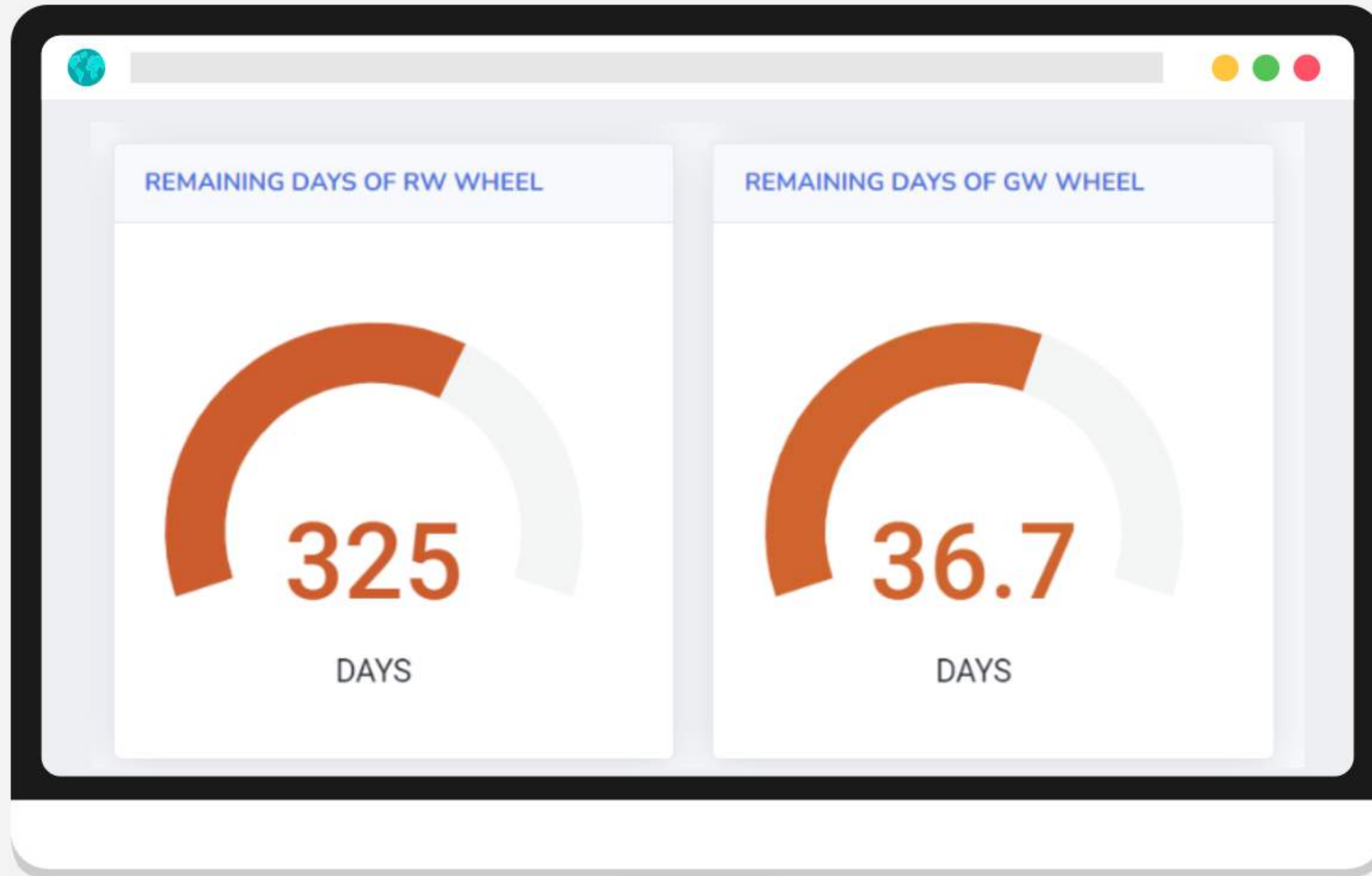


기울기가 양수가 될 시  
전의 기울기를 가져와서  
덮어쓰게 변경



# 시계열 예측 데이터 활용안

: GW/RW 혈잔량 및 교체시기 예측





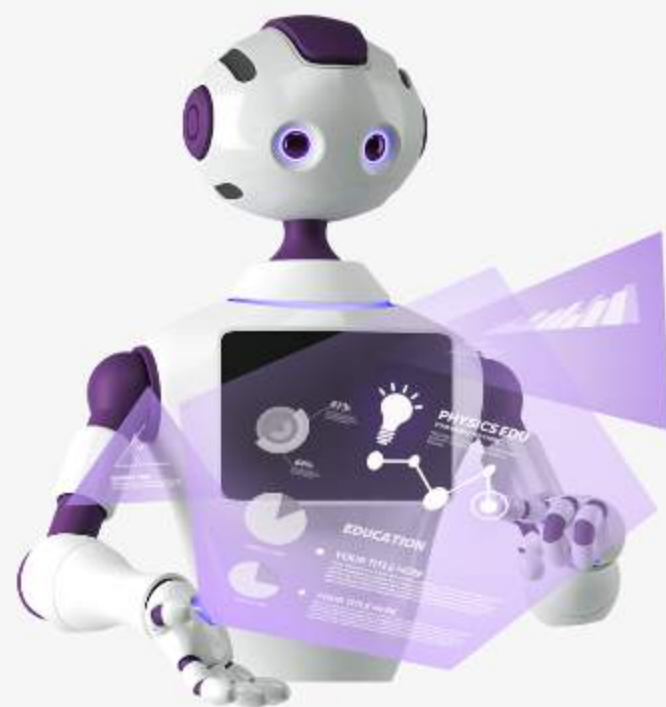
05

기대효과



# 프로젝트 기대효과

## 정량적 효과



- AI 데이터 분석을 통해 기존의 휠 교체주기 대비 생산량이 10% 이상 상승할 것으로 기대
- 휠 잔량 및 교체 주기를 예측하는 AI 분석 모델을 이용하여 이상감지 정확도가 80% 이상 될 것으로 기대

## 정성적 기대효과



- 설비 상태 및 휠 잔량 데이터를 분석하여 설비 예지보전 시스템 구축
- 실시간 모니터링 시스템을 통해 제조 라인에서 작업자들이 실시간 대응이 가능하도록 구현

A close-up, slightly blurred photograph of a person's hands typing on a white Apple keyboard. The keyboard is connected to a white iMac monitor, which is visible in the upper right corner. The background is a light-colored desk surface. A semi-transparent white rectangular box is overlaid on the right side of the image, containing the text '06 프로젝트 소감'.

**06**

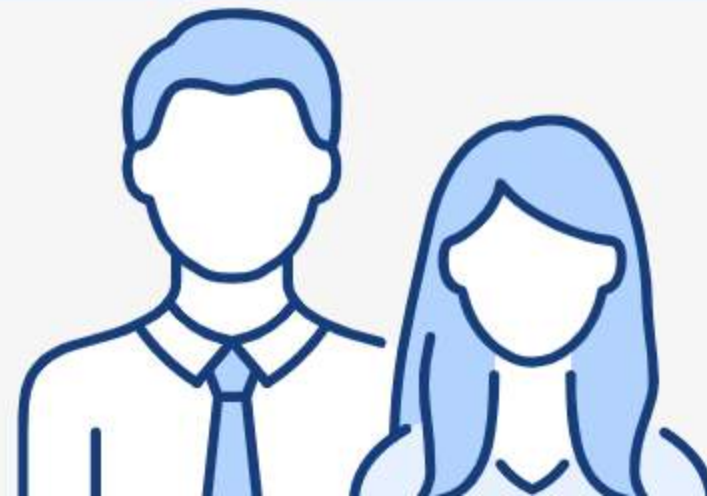
**프로젝트 소감**



## 한계점 및 개선점

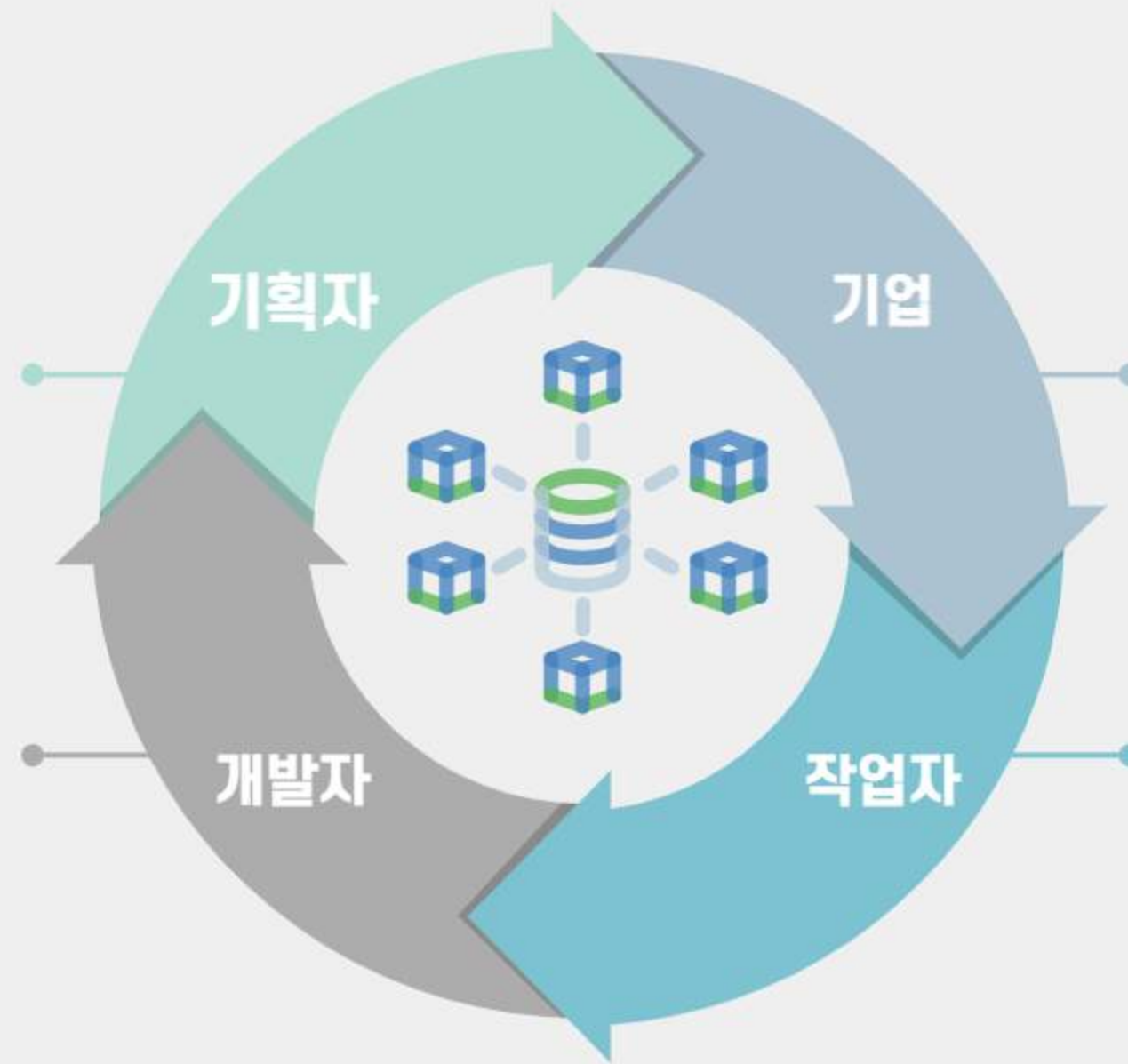
데이터의 대부분이 월의 두께와 상관없는 열밖에 없었다.  
따라서 **연관성이 있는 데이터가 있었다면**  
**다양한 알고리즘을 사용해서 더 좋은 결과가 나왔을 것.**

기업 특성상 실시간 데이터를 제공해 줄 수 없다고 해서  
이전 시점의 데이터만 사용해서 예측을 하였다.  
하지만 **데이터를 실시간으로 받을 수 있다면**  
**DB에 실시간으로 저장하고 바로 전처리 후 모델에까지 적용** 할 수 있을 것.



스마트 팩토리를  
이상적으로 구축하려는 기획자

열악한 데이터를 통해  
시스템을 만들어야하는 개발자



보안상의 이유로  
한정된 데이터만 전달하는 기업

실제 현장에서 일하는 작업자

이 모두가 능동적으로 서로의 간격을 좁히는데  
스마트팩토리의 악순환을 푸는 열쇠



---

**Q & A**