

# 기상변화에 따른 배추 가격 예측

-2조-

빅데이터융합전공 202213212 박소현

빅데이터융합전공 202213295 안현정

빅데이터융합전공 202213436 장원혁

통계데이터사이언스전공 202213471 정지원

# 목차

01 주제 선정 배경

02 데이터소개

03 EDA 및 전처리

04 공급량 예측 모델

05 가격 예측 모델

06 결과 및 기대효과

# 주제 선정 배경

# 주제 선정 배경

경제

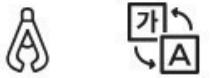
## 金사과·金상추 이어 배추값 쇼크 … 기껏 '양배추 김치' 꺼낸 정부

경제 > 일반경제

이희조 기자 love@mk.co.kr

박홍주 기자 hongju@mk.co.kr

입력 : 2024-09-24 18:11:47 수정 : 2024-09-24 20:09:07



# 연구 대상 및 목적



재배기술이 전국적으로 평준화되어있음

→ 기온이나 강수량 등 자연적인 기상요인에 의한  
생산량의 변화가 큰 것으로 보고됨



국내 채소류 5대 수급 민감 품목

→ 가격 변동에 국민 체감이 높은 식품임



연구 결과를 농민, 정부, 소비자 모두 활용할 수 있음

(수익안정화, 정책효율화, 가계부담완화)

# 분석 방향성

## 가격 메커니즘

상품가격은 살 사람과 팔 사람 즉 수요와 공급에 의해 결정된다.

공급 > 수요 : 가격이 내림

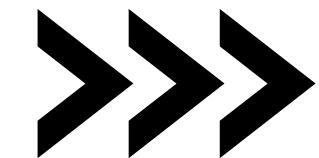
공급 < 수요: 가격은 오름

가격이 올라가면 수요가 줄고 공급은 늘어나 새로운 수급균형이 이뤄진다.

이 같은 조정기능이 가격메커니즘이다.

[네이버 지식백과] [가격메커니즘](#) [메커니즘, price mechanism] (매일경제, 매경닷컴)

공급량 예측



가격 예측

# 데이터 소개

# 데이터 소개



## 기상 데이터

기상청 – 기상자료개방포털

평균/최저/최고기온(°C), 일강수량(mm),  
강수 계속시간(hr), 최대/평균풍속(m/s),  
평균 상대습도(%), 일사량(MJ/m<sup>2</sup>),  
일조시간(hr), 일 최심신적설(cm)



## 배추 공급량 데이터

농넷 – 농산물유통 종합정보시스템

전국 도매시장 경락데이터  
총거래물량, 도매시장, 품종, 산지–광역시도  
봄, 여름, 고랭지, 가을, 월동, 기타배추, 배양채 등

→ 2020년~2024.11.18 까지의 데이터를 일별로 사용

# 변수 선택

평균기온(°C), 최저기온(°C), 최고기온(°C), 강수 계속시간(hr), 일강수량(mm), 최대 풍속(m/s), 평균 풍속(m/s), 평균 상대습도(%),  
합계 일조시간(hr), **가조시간(hr)**, 합계 일사량(MJ/m<sup>2</sup>), 일 최심신적설(cm), **평균 지면온도(°C)**

**온도:** 배추 성장과 발육에 중요 (농촌 진흥청 기준 최적 생육 온도: 15~20°C 정도)

- (1) 평균 기온: 전반적인 생육 환경을 나타냄
- (2) 최저 기온: 배추가 저온 피해(서리, 냉해 등)를 입을 가능성을 판단하는 데 유용
- (3) 최고 기온: 고온 스트레스, 병충해 발생 가능성, 증발산 증가로 인한 생육 저해를 예측하는 데 중요
- (4) 일 최심신적설 (cm): 저온 환경과 물리적 피해(적설량이 많을 경우, 수확이 지연되거나 유통 과정에서 품질이 저하될 수 있음)

**강수:** 수분 공급과 생육 환경, 병해 관리에 중요

- (1) 일강수량 (mm): 생장기에 충분한 수분이 필요→배추 뿌리 흡수력 높임
- (2) 강수 계속시간 (hr): 토양 과습과 병해 관리에 있어 중요

# 변수 선택

평균기온(°C), 최저기온(°C), 최고기온(°C), 강수 계속시간(hr), 일강수량(mm), 최대 풍속(m/s), 평균 풍속(m/s), 평균 상대습도(%),  
합계 일조시간(hr), **가조시간(hr)**, 합계 일사량(MJ/m<sup>2</sup>), 일 최심신적설(cm), **평균 지면온도(°C)**

## 풍속: 배추 작물에 바람의 물리적, 환경적 영향

- (1) 평균 풍속(m/s): 배추 밭의 통풍 상태를 나타냄
- (2) 최대 풍속(m/s): 강한 바람으로 인한 물리적 피해(잎의 찢김, 작물 전도 등)를 예측하는 데 중요

## 습도: 병해 발생에 영향

평균 상대습도 (%): 습도가 높으면 병해충 발생률 증가, 습도가 낮으면 증발이 높아져 생육 저하됨

## 일사: 광합성과 에너지 흡수

일사량 (MJ/m<sup>2</sup>): 광합성 가능 시간을 나타냄

## 일조: 생육과 생장 시간의 조절

일조시간 (hr): 배추의 생육 단계와 관련이 깊음고 특히 결구(배추의 머리 형성) 과정에서 중요

# 변수 선택

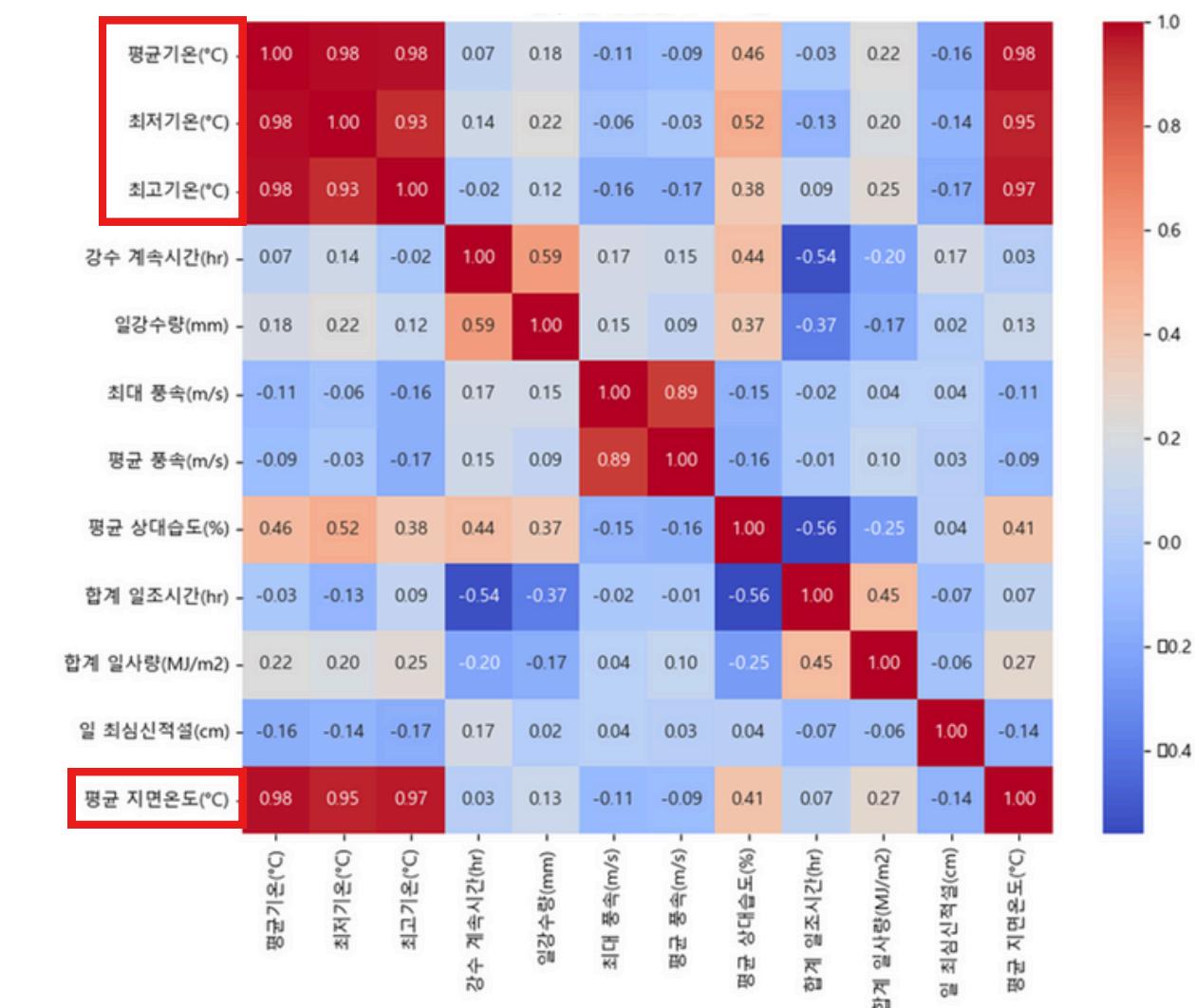
평균기온( $^{\circ}\text{C}$ ), 최저기온( $^{\circ}\text{C}$ ), 최고기온( $^{\circ}\text{C}$ ), 강수 계속시간(hr), 일강수량(mm), 최대 풍속(m/s), 평균 풍속(m/s), 평균 상대습도(%),  
합계 일조시간(hr), **가조시간(hr)**, 합계 일사량(MJ/m<sup>2</sup>), 일 최심신적설(cm), **평균 지면온도( $^{\circ}\text{C}$ )**

**가조시간**: 일출부터 일몰까지의 시간

→ 일조시간만 사용

**평균 지면온도**: 맨땅 또는 짧은 잔디 아래의 온도

→ 상관관계 히트맵 결과 기온과 상관계수 높음  
다중공선성 문제



# 데이터 소개



## 배추/무 가격데이터

한국농수산식품유통공사\_채소류

등급 : 상품,

배추 단위 : 10kg(그물망 3포기)

무 단위 : 20kg

서울, 부산, 대구, 광주, 대전 주요 5개 도시의  
중도매가를 일별로 나타냄

→ 2020년~2024.10까지의 데이터를 일별로 사용



## 배추 월별수입량

농넷 – 농식품수출정보

→ 2020년~2024년 데이터를 월별로 사용

## ... 소비자물가지수

KOSIS

→ 2020년~2024년 데이터를 월별로 사용



## 무 가격데이터

- \* 대체재 관계 고려: 무는 배추와 함께 김장철 주요 채소로 사용되며, 상호 대체재로 작용
- \* 시장 경쟁 요인 반영: 무 가격은 배추와의 시장 경쟁 관계를 나타내므로, 이를 통해 배추의 상대적 시장 위치와 가격 탄력을 분석할 수 있음



## 배추 월별수입량

- \* 국내 공급 보완 효과: 국내 수급 상황의 보완적 지표로, 공급 부족 시 가격 급등을 완화하거나 수입 감소 시 가격 상승을 줄시키는 요인을 설명
- \* 글로벌 시장 영향 반영: 배추 수입량은 국제 농산물 시장의 영향을 받으며, 수입량 증감이 국내 가격에 직접적인 영향을 미칠 수 있음



## 소비자물가지수

- \* 단경제 전반의 물가 수준 반영: 소비자물가지수(CPI)는 전반적인 물가 상승률을 나타내며, 배추 가격 또한 이러한 경제 상황의 영향을 받음
- \* 시장 전반의 동조화된 변동성 포함: 배추는 필수 소비재로, CPI와의 상관관계를 통해 가격 변동의 거시적 요인을 분석할 수 있음

# EDA 및 전처리

# 기상 데이터

2018~2024까지 7개년에 모두 존재하는  
지점을 사용하여 16개 도별 평균으로 구성

세종 → 충남에 포함

```
# 강원도
gangwon = [90, 95, 100, 105, 106, 101, 212, 216, 121, 211]
# 경기도
sudogwon = [98, 119, 202, 203]
# 충청북도
chungcheongbukdo = [221, 226, 127, 131, 135]
# 충청남도
chungcheongnamdo = [232, 235, 236, 238, 129]
# 전라북도
jeollabukdo = [243, 244, 245, 247, 248, 140, 146]
# 전라남도
jeollanamdo = [260, 261, 262, 165, 168, 169, 170]
# 경상북도
gyeongsangbukdo = [115, 130, 136, 138, 271, 272, 273, 277, 278, 279, 281]
# 경상남도
gyeongsangnamdo = [155, 284, 285, 288, 289, 294, 295, 192, 162]
# 제주도
jeju = [184, 185, 188, 189]
#
# 광역시도
code=[108, 159, 143, 112, 156, 133, 152]
name=['서울','부산','대구','인천','광주','대전','울산']
```

# 기상 데이터

지역별 2018~2024 일별

지역	일시	평균기온 (°C)	최저기온 (°C)	최고기온 (°C)	강수 계속시간 (hr)	일강수량 (mm)	최대 풍속 (m/s)	평균 풍속 (m/s)	평균 상대습도 (%)	합계 일조시간 (hr)	합계 일사량 (MJ/m2)	일 최심신적설 (cm)
0	서울 2018-01-01	-1.300	-5.100	3.800	0.0000	0.000	3.800	1.400	39.100	8.300	6.1400	0.0
1	서울 2018-01-02	-1.800	-4.300	1.800	0.0000	0.000	4.900	1.800	42.000	7.900	5.3600	0.0
2	서울 2018-01-03	-4.700	-7.100	-0.400	0.0000	0.000	3.500	2.200	42.300	8.600	6.5600	0.0
3	서울 2018-01-04	-4.700	-8.700	-0.700	0.0000	0.000	3.500	1.400	43.000	6.200	4.7300	0.0
4	서울 2018-01-05	-3.000	-5.600	1.600	0.0000	0.000	3.600	1.700	48.400	8.200	5.9600	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...
40219	제주 2024-11-14	18.725	15.825	22.875	0.0825	0.000	5.050	2.575	69.700	5.200	8.8300	0.0
40220	제주 2024-11-15	18.600	16.925	20.725	6.6200	16.475	5.325	2.350	89.825	0.175	2.9150	0.0
40221	제주 2024-11-16	20.250	18.225	23.475	1.7700	6.250	5.925	2.450	84.250	4.150	7.6550	0.0
40222	제주 2024-11-17	16.025	12.350	19.850	0.0000	0.000	8.150	5.325	62.500	6.325	9.2325	0.0
40223	제주 2024-11-18	10.450	8.200	13.350	0.0000	0.000	10.075	5.225	49.775	8.525	11.2050	0.0

40224 rows × 13 columns

# 기상 데이터

16개 지역 2018~2024 일별평균

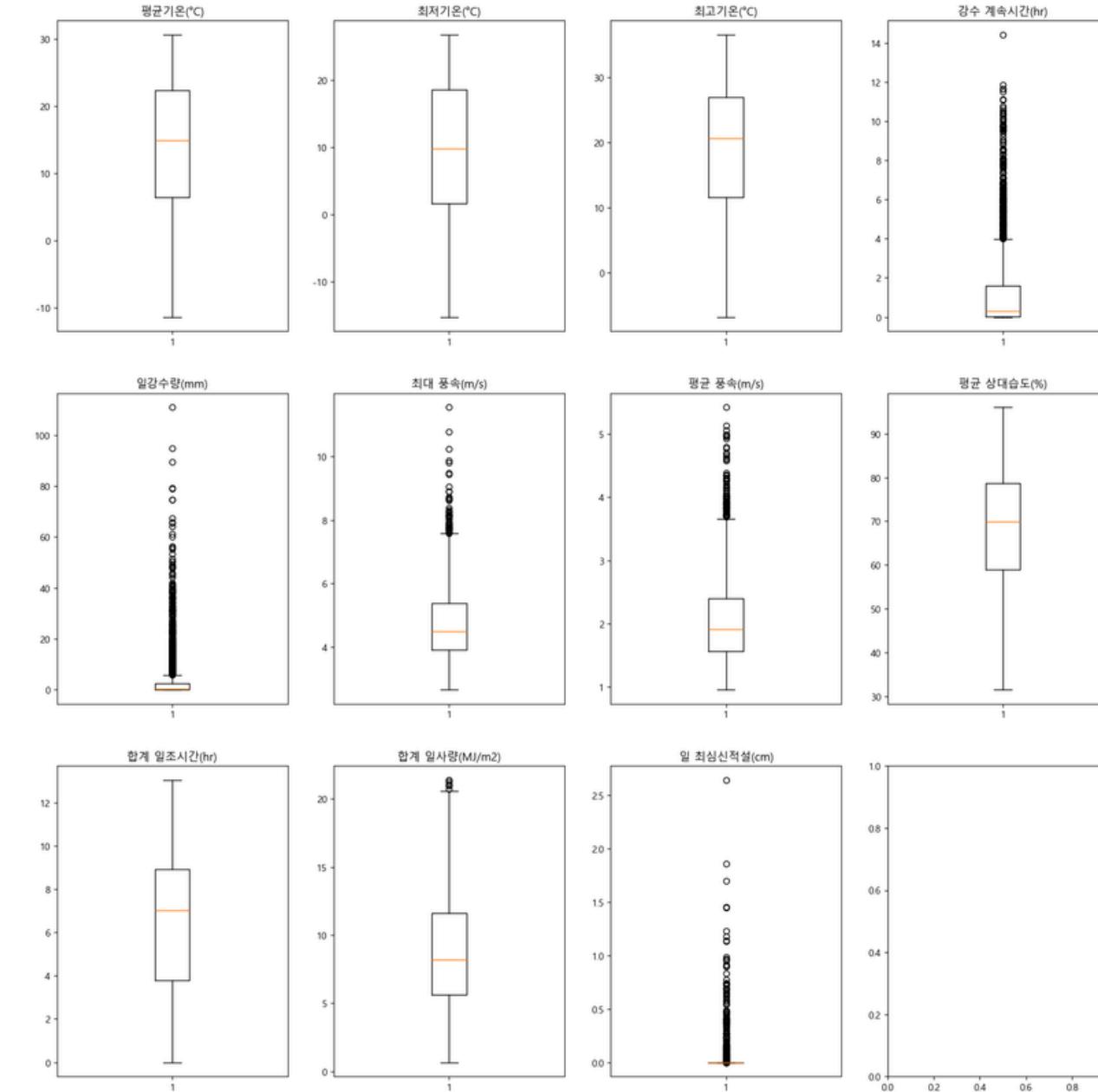
	날짜	평균기온 (°C)	최저기온 (°C)	최고기온 (°C)	강수 계속시간 (hr)	일강수량 (mm)	최대 풍속 (m/s)	평균 풍속 (m/s)	평균 상대습도 (%)	합계 일조시간 (hr)	합계 일사량 (MJ/m2)	일 최심신적설 (cm)
0	2018-01-01	0.267361	-3.760766	5.302104	0.011364	0.000000	5.009205	2.025397	48.868839	8.394137	4.727196	0.000000
1	2018-01-02	0.671816	-4.146029	5.783363	0.015122	0.000000	4.739608	1.843112	54.916389	5.901650	3.671890	0.000000
2	2018-01-03	-1.148651	-4.342379	3.032239	0.095852	0.021591	4.984683	2.529236	44.778602	8.165281	4.816750	0.020455
3	2018-01-04	-1.397601	-5.103602	2.269814	0.075710	0.129972	4.171818	1.829650	51.425960	4.775826	3.394089	0.030114
4	2018-01-05	0.199593	-2.937990	4.781875	0.608918	0.271559	5.062048	2.133430	55.193296	6.340706	3.954609	0.002841
...	...	...	...	...	...	...	...	...	...	...	...	...
2509	2024-11-14	14.350618	10.302918	19.051204	0.880449	0.097050	3.169255	1.349979	75.634007	2.195021	5.137481	0.000000
2510	2024-11-15	15.566232	12.773051	19.972371	1.752232	2.338785	3.447706	1.403988	78.383544	2.414799	4.945873	0.000000
2511	2024-11-16	17.044227	12.879399	22.797258	1.063522	1.573466	4.033304	1.609690	76.852073	5.841150	6.755576	0.000000
2512	2024-11-17	10.931144	5.635602	16.466811	0.186948	0.200511	5.856096	3.263261	54.913289	1.749926	4.247110	0.000000
2513	2024-11-18	4.577782	0.846177	9.378380	0.073864	0.157955	6.192763	2.990282	40.788777	8.814702	9.771494	0.000000

2514 rows × 12 columns

# 기상 데이터

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2514 entries, 0 to 2513
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   날짜             2514 non-null    object  
 1   평균기온(°C)     2514 non-null    float64
 2   최저기온(°C)     2514 non-null    float64
 3   최고기온(°C)     2514 non-null    float64
 4   강수 계속시간(hr) 2514 non-null    float64
 5   일강수량(mm)    2514 non-null    float64
 6   최대 풍속(m/s)   2514 non-null    float64
 7   평균 풍속(m/s)   2514 non-null    float64
 8   평균 상대습도(%) 2514 non-null    float64
 9   합계 일조시간(hr) 2514 non-null    float64
 10  합계 일사량(MJ/m2) 2514 non-null    float64
 11  일 최심신적설(cm) 2514 non-null    float64
dtypes: float64(11), object(1)
memory usage: 235.8+ KB
```

-> 결측값은 없음



# 공급량 데이터

	DATE	거래단위	평균가격	거래단위별 총거래물량	총거래금액	도매시장	도매법인	품목	품종	산지-광역시도	등급	
0	2024-11-18	10kg그물망	5986.161987		487570.0	291867300	서울가락도매	대아청과	배추	김장(가을)배추	전라남도	특
1	2024-11-18	10kg그물망	5916.771954		319760.0	189194700	서울가락도매	대아청과	배추	김장(가을)배추	강원도	특
2	2024-11-18	12kg그물망	6427.236035		74328.0	39810300	서울강서도매	서부청과	배추	기타배추	전라남도	특
3	2024-11-18	10kg그물망	5222.22061		68900.0	35981100	서울가락도매	대아청과	배추	김장(가을)배추	전라북도	특
4	2024-11-18	10kg그물망	5006.465328		68210.0	34149100	서울가락도매	대아청과	배추	김장(가을)배추	경상북도	특
...	...	...	...	...	...	...	...	...	...	...	...	
544387	2020-01-02	2kg상자	4000		2.0	4000	전주도매시장	전주원협(공)	배추	기타배추	-	특
544388	2020-01-02	0.3kg비닐봉지	6000		1.5	30000	춘천도매시장	춘천원협(공)	배추	기타배추	서울	특
544389	2020-01-02	0.2kg비닐봉지	4000		1.0	20000	춘천도매시장	춘천원협(공)	배추	기타배추	서울	특
544390	2020-01-02	0.1kg비닐봉지	1500		1.0	15000	춘천도매시장	춘천원협(공)	배추	기타배추	서울	특
544391	2020-01-02	0.5kg단	10000		0.5	10000	춘천도매시장	춘천원협(공)	배추	기타배추	서울	특

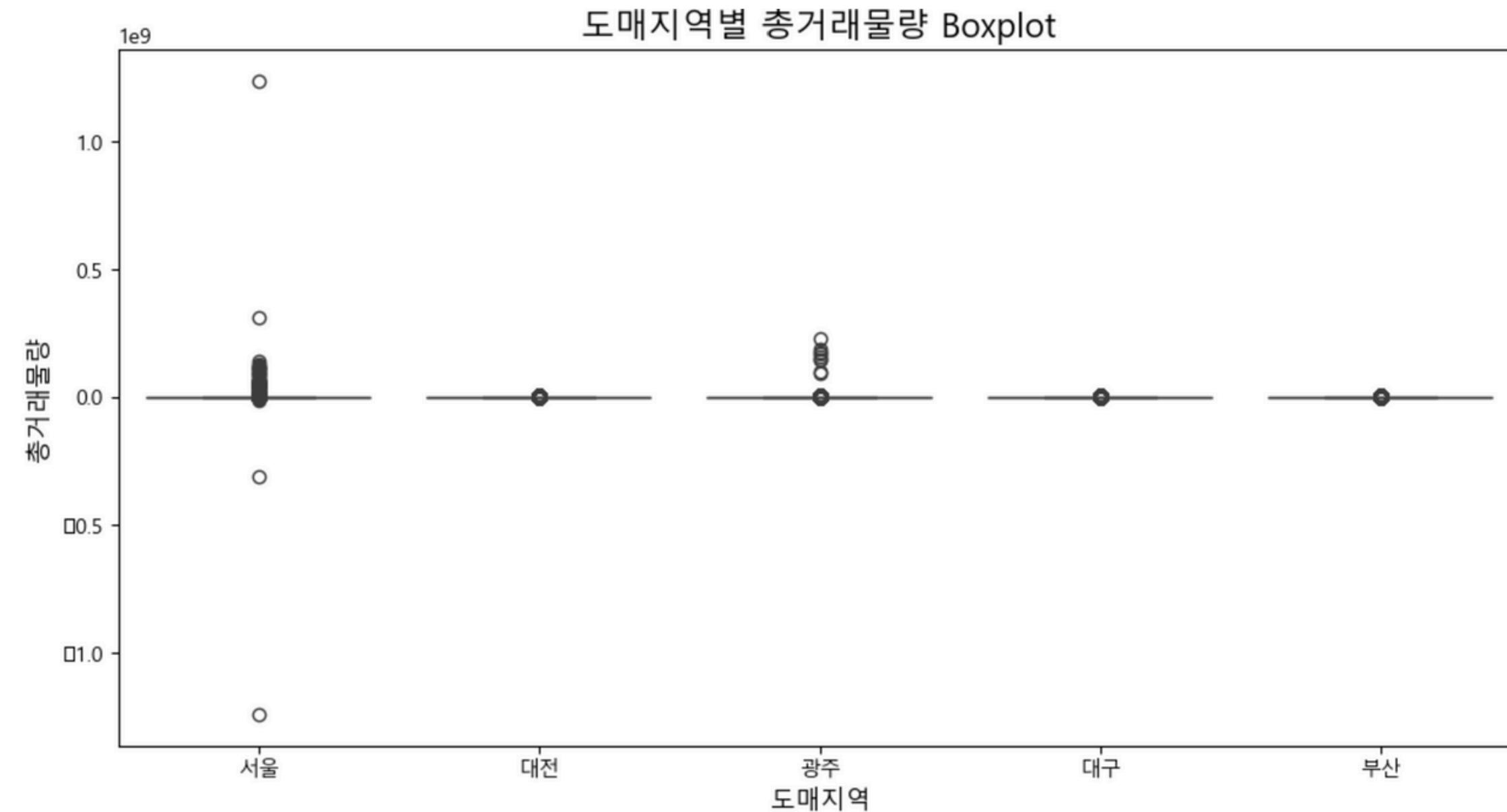
544392 rows × 11 columns

# 공급량 데이터

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 544392 entries, 0 to 544391
Data columns (total 11 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   DATE        544392 non-null   datetime64[ns]
 1   거래단위     544392 non-null   object 
 2   평균가격     544392 non-null   object 
 3   거래단위별 총거래물량  544392 non-null   float64
 4   총거래금액    544392 non-null   int64  
 5   도매시장     544392 non-null   object 
 6   도매법인     544392 non-null   object 
 7   품목         544392 non-null   object 
 8   품종         544392 non-null   object 
 9   산지-광역시도 544392 non-null   object 
 10  등급         544392 non-null   object 
dtypes: datetime64[ns](1), float64(1), int64(1), object(8)
memory usage: 45.7+ MB
```

→ 결측값은 없음

# 공급량 데이터



# 공급량 데이터

날짜|거래단위(kg)|거래단위별 총거래물량(갯수)|도매지역|품종|산지|총거래물량

89397	2024-01-24	10.0	-10020.0	대전	월동배추	전북	-100200.0
338675	2021-11-15	20.0	-120.0	대전	절임배추	충북	-2400.0
486934	2020-07-29	0.7	-2.1	부산	여름배추	부산	-1.47
263362	2022-08-22	8.0	-56.0	대구	쌈배추	강원	-448.0
270944	2022-07-26	8.0	-480.0	대구	쌈배추	강원	-3840.0
438863	2020-11-30	20.0	-11080.0	대구	절임배추	경북	-221600.0
14904	2024-10-18	4.0	-60.0	서울	쌈배추	강원	-240.0
14905	2024-10-18	4.0	-120.0	서울	쌈배추	강원	-480.0
20527	2024-10-03	8.0	-8.0	서울	쌈배추	서울	-64.0
29262	2024-09-05	20.0	-11940.0	서울	포기김치	강원	-238800.0
38464	2024-08-05	10.0	-800.0	서울	배추(주입)	인천	-8000.0
...	...	...	...	...	...	...	...
531416	2020-02-15	12.0	-96.0	서울	기타배추	전남	-1152.0
540793	2020-01-14	10.0	-1000.0	서울	월동배추	강원	-10000.0
542729	2020-01-08	10.0	-800.0	서울	월동배추	경북	-8000.0
543909	2020-01-04	10.0	-500.0	서울	월동배추	서울	-5000.0
543910	2020-01-04	10.0	-500.0	서울	월동배추	강원	-5000.0

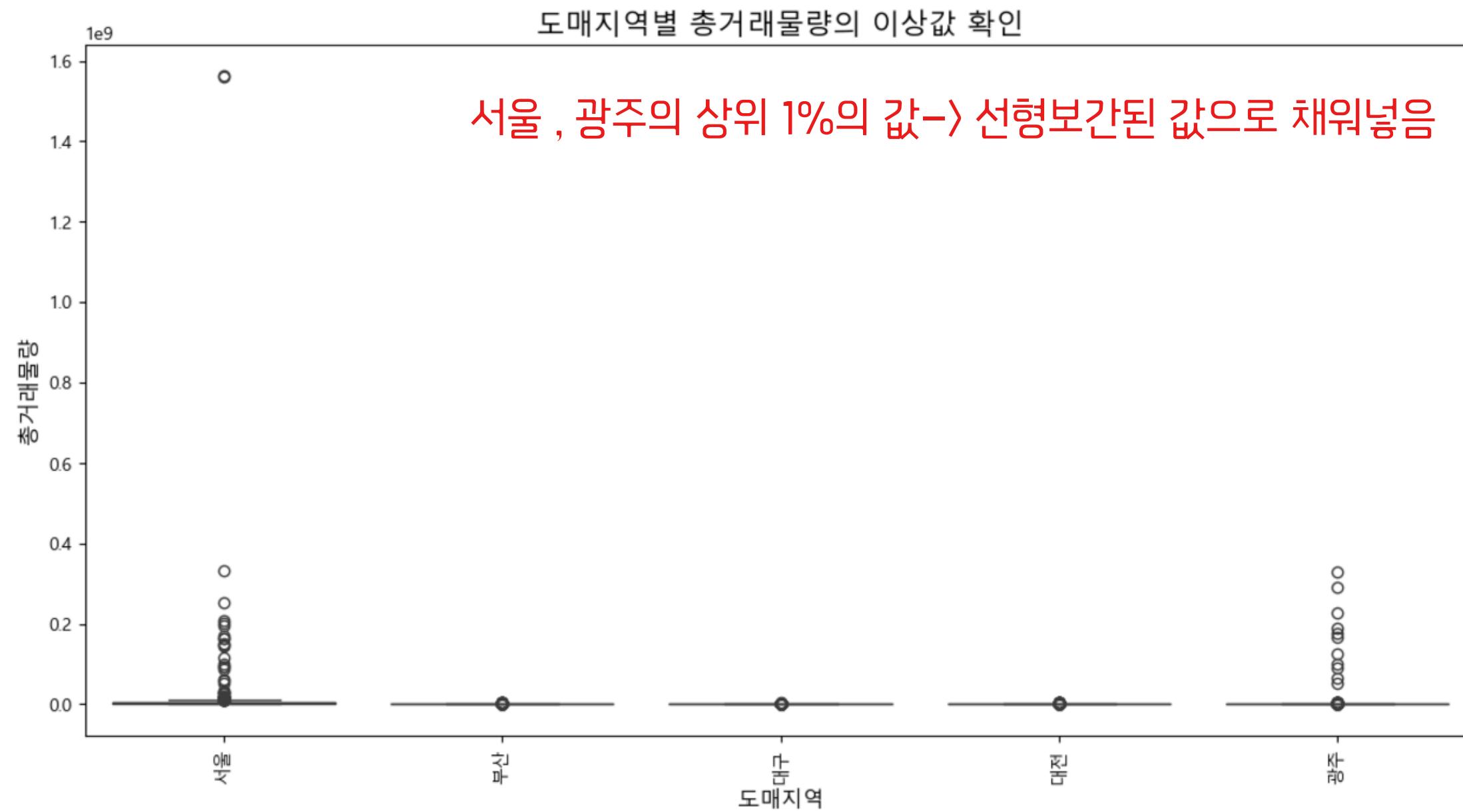
[ 99 rows x 7 columns ]

# 공급량 데이터

날짜	거래단위	거래단위별 총거래물량	도매지역	품종	산지
0 2024-11-18	10kg그물망	487570.0	서울	김장(가을)배추	전남
1 2024-11-18	10kg그물망	319760.0	서울	김장(가을)배추	강원
2 2024-11-18	12kg그물망	74328.0	서울	기타배추	전남
3 2024-11-18	10kg그물망	68900.0	서울	김장(가을)배추	전북
4 2024-11-18	10kg그물망	68210.0	서울	김장(가을)배추	경북
...	...	...	...	...	...

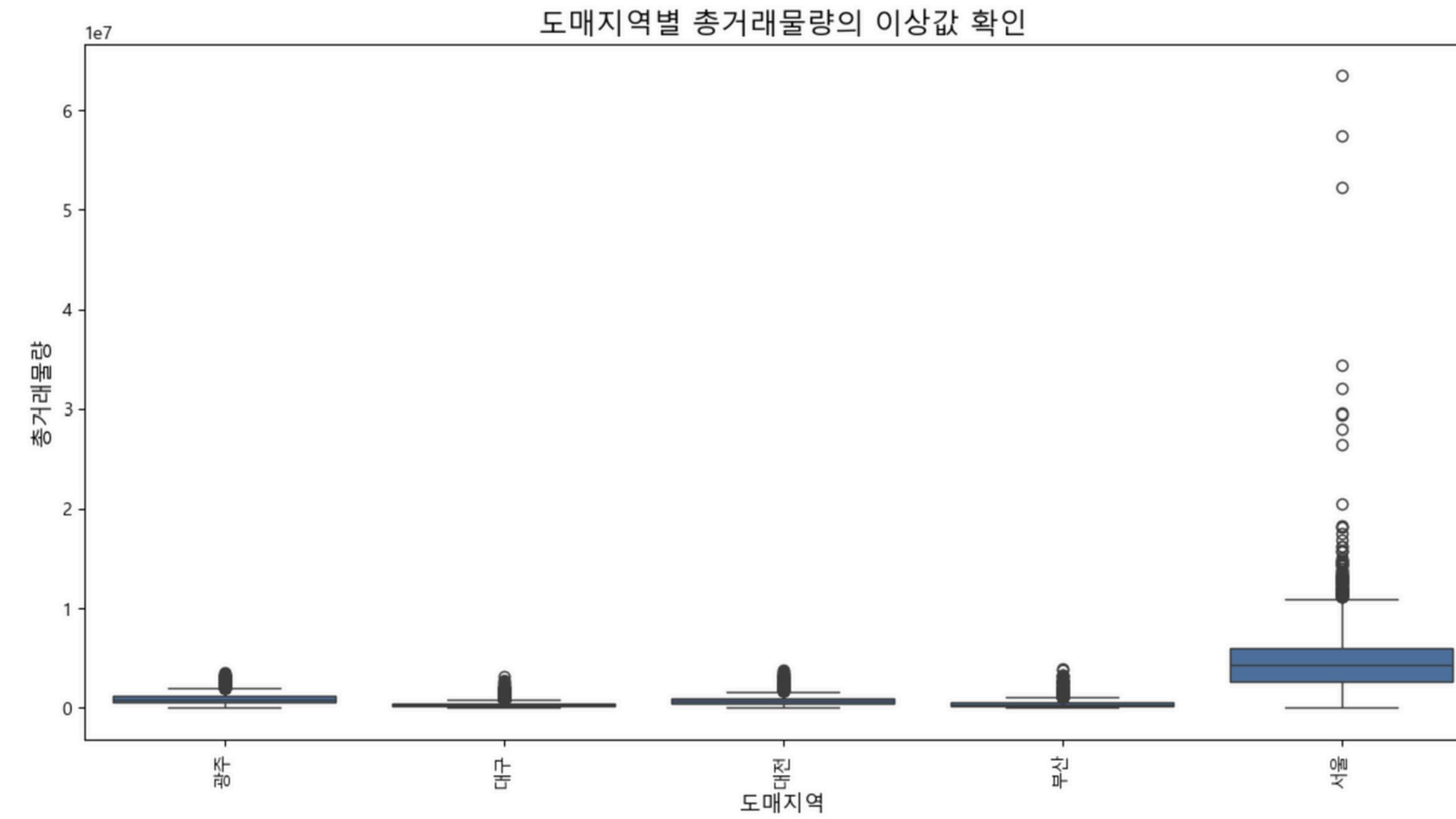
- 날짜, 거래단위, 거래단위별 총거래물량, 도매시장, 품종, 산지를 제외한 열 제거
- 도매 시장 명의 앞 두글자만 남겨두고 도매지역으로 열 이름을 바꿈  
ex) 서울가락도매(열 이름: 도매시장) → 서울(열이름: 도매지역)
- 거래단위별 총거래물량에서 kg앞의 글자만 남기기
- 거래단위와 거래단위별 총거래물량 열을 숫자형으로 변환 후 곱해 ‘총거래물량’ 열을 추가
- 총거래물량이 음수인 값은 절대값 처리

# 공급량 데이터



날짜	도매지역	총거래물량
8551	2023-11-25	광주 330,233,540
8549	2023-11-23	광주 290,879,988
8557	2023-12-01	광주 228,393,684
8589	2024-01-02	광주 188,995,333
8542	2023-11-16	광주 175,858,320
8552	2023-11-26	광주 166,892,804
8588	2024-01-01	광주 126,379,374
8056	2022-07-18	광주 101,168,352
8842	2024-09-11	광주 92,330,824
8587	2023-12-31	광주 63,763,415
8055	2022-07-17	광주 50,959,916
8490	2023-09-25	광주 5,294,168
8489	2023-09-24	광주 4,178,156
7456	2020-11-25	광주 3,972,160

# 공급량 데이터



# 공급량 데이터

- 절임배추, 우거지, 배추(수입), 우거지(수입), 배추뿌리, 저장배추, 배양채 행제거
- **남은 모든 품종을 하나로 합침**
- 도매지역 대전, 광주, 서울, 부산, 대구를 제외한 행 제거
- 날짜, 도매지역 값이 같은 '총거래물량' 값 합침

전처리 후

도매지역 => '서울', '부산', '대구', '대전', '광주'

	날짜	총거래물량	도매지역
0	2020-01-03	8322604.0	서울
1	2020-01-04	5243075.0	서울
2	2020-01-05	5464235.0	서울
3	2020-01-06	5685395.0	서울
4	2020-01-07	3762942.0	서울
...	...	...	...
1778	2024-11-14	1775492.0	광주
1779	2024-11-15	1348792.0	광주
1780	2024-11-16	1273132.0	광주
1781	2024-11-17	1070848.0	광주
1782	2024-11-18	868564.0	광주

8911 rows × 3 columns

# 공급량 예측 데이터프레임

## 도매지역별

날짜	총거래물량	도매지역	평균기온(°C)	최저기온(°C)	최고기온(°C)	강수 계속시간(hr)	일강수량(mm)	최대 풍속(m/s)	평균 풍속(m/s)	평균 상대습도(%)	합계 일조시간(hr)	합계 일사량(MJ/m2)	일 최심신적설(cm)	
0	2020-01-03	8322604.0	서울	2.652054	-1.152863	7.798667	0.000000	0.000000	4.514822	1.909097	64.363028	7.478393	5.438115	0.0
1	2020-01-04	5243075.0	서울	2.213061	-2.263931	7.995609	0.012670	0.008523	4.730504	1.859670	63.722875	6.934945	5.269988	0.0
2	2020-01-05	5464235.0	서울	2.461428	-3.098501	8.978610	0.034010	0.026705	2.998543	1.154877	59.174106	7.379017	5.966763	0.0
3	2020-01-06	5685395.0	서울	4.486334	0.855391	6.658482	8.310125	8.954236	3.854252	1.646484	82.079009	0.007670	1.010279	0.0
4	2020-01-07	3762942.0	서울	9.650072	5.806695	14.119640	11.625856	38.428220	7.498910	2.766200	95.378831	0.054688	0.679592	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
8906	2024-11-14	1775492.0	광주	14.350618	10.302918	19.051204	0.880449	0.097050	3.169255	1.349979	75.634007	2.195021	5.137481	0.0
8907	2024-11-15	1348792.0	광주	15.566232	12.773051	19.972371	1.752232	2.338785	3.447706	1.403988	78.383544	2.414799	4.945873	0.0
8908	2024-11-16	1273132.0	광주	17.044227	12.879399	22.797258	1.063522	1.573466	4.033304	1.609690	76.852073	5.841150	6.755576	0.0
8909	2024-11-17	1070848.0	광주	10.931144	5.635602	16.466811	0.186948	0.200511	5.856096	3.263261	54.913289	1.749926	4.247110	0.0
8910	2024-11-18	868564.0	광주	4.577782	0.846177	9.378380	0.073864	0.157955	6.192763	2.990282	40.788777	8.814702	9.771494	0.0

8911 rows × 14 columns

# 수입량 데이터

농넷–농산물 수출입 데이터 수입 배추  
단위 : t(톤)

2020~2024년 월별 데이터를 일별로 중복 처리

	날짜	배추 수입량
0	2020-01-01	0.0
1	2020-01-02	0.0
2	2020-01-03	0.0
3	2020-01-04	0.0
4	2020-01-05	0.0
	...	...
1761	2024-10-27	1870.6
1762	2024-10-28	1870.6
1763	2024-10-29	1870.6
1764	2024-10-30	1870.6
1765	2024-10-31	1870.6

1766 rows × 2 columns

# 소비자물가지수

KOSIS-소비자물가지수

2020년도 100을 기준으로 한 소비자 물가지수

2020~2024년 월별 데이터를 일별로 중복 처리

	날짜	물가지수
0	2020-01-01	100.09
1	2020-01-02	100.09
2	2020-01-03	100.09
3	2020-01-04	100.09
4	2020-01-05	100.09
...	...	...
1761	2024-10-27	114.69
1762	2024-10-28	114.69
1763	2024-10-29	114.69
1764	2024-10-30	114.69
1765	2024-10-31	114.69

1766 rows × 2 columns

# 무 가격 데이터

한국농수산식품유통공사\_채소류/무/전체

무 상품 기준 서울, 부산, 대구, 대전, 광주

일간 무가격 데이터 (단위 : 20kg(그물망 3포기))

NaN 값은 선형 보간으로 대체

	날짜	지역	무가격
0	2020-01-02	광주	12000
1	2020-01-03	광주	10500
2	2020-01-04	광주	11136
3	2020-01-05	광주	10681
4	2020-01-06	광주	10500
...	...	...	...
8965	2024-11-25	서울	10400
8966	2024-11-26	서울	11400
8967	2024-11-27	서울	11400
8968	2024-11-28	서울	13400
8969	2024-11-29	서울	12600

8970 rows × 3 columns

# 배추 가격 데이터

한국농수산식품유통공사\_채소류/배추/전체

배추 상품 기준 서울, 부산, 대구, 대전, 광주

일간 배추가격 데이터 (단위 : 10kg(그물망 3포기))

NaN 값은 선형 보간으로 대체

	날짜	지역	배추가격
0	2020-01-02	광주	12000
1	2020-01-03	광주	10500
2	2020-01-04	광주	11136
3	2020-01-05	광주	10681
4	2020-01-06	광주	10500
...	...	...	...
8965	2024-11-25	서울	10400
8966	2024-11-26	서울	11400
8967	2024-11-27	서울	11400
8968	2024-11-28	서울	13400
8969	2024-11-29	서울	12600

8970 rows × 3 columns

# 공급량 예측 모델

# 공급량 예측 모델

[과정] 목적시계열 : 배추 공급량  
공변량 : 평균/최저/최고기온, 일강수량, 강수 계속시간,  
최대/평균풍속, 평균 상대습도, 일사량, 일조시간, 일 최심신적설  
train:test=7:3비율로 데이터셋 분할  
Min-Max 스케일링 시행  
RNN, CNN-RNN, LSTM, CNN-LSTM 학습  
결과를 원래 스케일로 역변환(inverse\_transform)  
MAE, MSE, RMSE, R2로 성능평가

# 공급량 예측 모델

```
model = Sequential()
model.add(SimpleRNN(50, activation="relu", input_shape=(X_train.shape[1], X_train.shape[2])))
model.add(Dense(1))
model.compile(optimizer="adam", loss="mse")
early_stopping = EarlyStopping(monitor="val_loss", patience=10, restore_best_weights=True)
history = model.fit(X_train, y_train, epochs=100, batch_size=16, validation_data=(X_test, y_test), callbacks=[early_stopping])
```

RNN

```
model = Sequential()
model.add(LSTM(50, activation="relu", input_shape=(X_train.shape[1], X_train.shape[2])))
model.add(Dense(1))
model.compile(optimizer="adam", loss="mse")
early_stopping = EarlyStopping(monitor="val_loss", patience=10, restore_best_weights=True)
history = model.fit(X_train, y_train, epochs=100, batch_size=16, validation_data=(X_test, y_test), callbacks=[early_stopping])
```

LSTM

# 공급량 예측 모델

```
model = Sequential()
model.add(Conv1D(filters=64, kernel_size=3, activation='relu', input_shape=(X_train.shape[1], X_train.shape[2])))
model.add(MaxPooling1D(pool_size=2))
model.add(SimpleRNN(50, activation="relu"))
model.add(Dense(1))
model.compile(optimizer="adam", loss="mse")
early_stopping = EarlyStopping(monitor="val_loss", patience=10, restore_best_weights=True)
history = model.fit(X_train, y_train, epochs=100, batch_size=16, validation_data=(X_test, y_test), callbacks=[early_stopping])
```

CNN-RNN

```
model = Sequential()
model.add(Conv1D(filters=64, kernel_size=3, activation='relu', input_shape=(X_train.shape[1], X_train.shape[2])))
model.add(MaxPooling1D(pool_size=2))
model.add(LSTM(50, activation="relu"))
model.add(Dense(1))
model.compile(optimizer="adam", loss="mse")
early_stopping = EarlyStopping(monitor="val_loss", patience=10, restore_best_weights=True)
history = model.fit(X_train, y_train, epochs=100, batch_size=16, validation_data=(X_test, y_test), callbacks=[early_stopping])
```

CNN-LSTM

# 공급량 예측 모델

RNN

timestep=60

	mae	mse	rmse	r2
서울	1.754743e+06	5.454388e+12	2.335463e+06	0.251885
광주	3.450766e+05	2.409839e+11	4.909011e+05	0.197707
대전	2.633741e+05	1.302479e+11	3.608988e+05	0.235421
부산	1.233322e+05	2.595113e+10	1.610935e+05	0.218780
대구	1.438539e+05	4.439104e+10	2.106918e+05	0.347396

timestep=80

	mae	mse	rmse	r2
서울	1.974253e+06	7.136585e+12	2.671439e+06	0.158302
광주	3.886249e+05	2.921797e+11	5.405365e+05	0.252726
대전	2.928724e+05	1.607457e+11	4.009310e+05	0.207947
부산	1.463545e+05	3.592983e+10	1.895517e+05	-0.072638
대구	1.576398e+05	5.174768e+10	2.274812e+05	0.470112

LSTM

timestep=60

	mae	mse	rmse	r2
서울	1.489535e+06	3.806982e+12	1.951149e+06	0.477841
광주	3.607250e+05	2.629138e+11	5.127512e+05	0.124698
대전	2.369573e+05	1.097508e+11	3.312866e+05	0.355743
부산	1.109956e+05	2.189851e+10	1.479814e+05	0.340778
대구	1.428500e+05	4.505950e+10	2.122722e+05	0.337569

timestep=80

	mae	mse	rmse	r2
서울	1.766256e+06	5.915906e+12	2.432264e+06	0.302271
광주	3.727489e+05	3.094487e+11	5.562812e+05	0.208559
대전	2.327024e+05	1.014989e+11	3.185890e+05	0.499878
부산	1.238626e+05	2.489342e+10	1.577765e+05	0.256840
대구	1.388962e+05	4.236560e+10	2.058290e+05	0.566183

# 공급량 예측 모델

timestep=60

	mae	mse	rmse	r2
서울	1.515122e+06	4.470429e+12	2.114339e+06	0.386843
광주	3.626048e+05	2.558430e+11	5.058093e+05	0.148238
대전	2.535307e+05	1.154482e+11	3.397767e+05	0.322298
부산	1.211466e+05	2.503357e+10	1.582200e+05	0.246401
대구	1.468605e+05	4.957757e+10	2.226602e+05	0.271148

CNN-RNN

timestep=80

	mae	mse	rmse	r2
서울	2.222100e+06	8.134733e+12	2.852145e+06	0.040580
광주	3.763291e+05	2.985865e+11	5.464307e+05	0.236340
대전	2.617434e+05	1.391138e+11	3.729796e+05	0.314535
부산	1.343740e+05	3.192394e+10	1.786727e+05	0.046953
대구	1.851413e+05	7.165641e+10	2.676871e+05	0.266249

timestep=60

	mae	mse	rmse	r2
서울	1.355355e+06	3.325128e+12	1.823493e+06	0.543931
광주	3.470366e+05	2.424638e+11	4.924062e+05	0.192781
대전	2.354966e+05	1.008835e+11	3.176217e+05	0.407796
부산	1.143358e+05	2.274108e+10	1.508015e+05	0.315413
대구	1.381123e+05	4.098701e+10	2.024525e+05	0.397440

CNN-LSTM

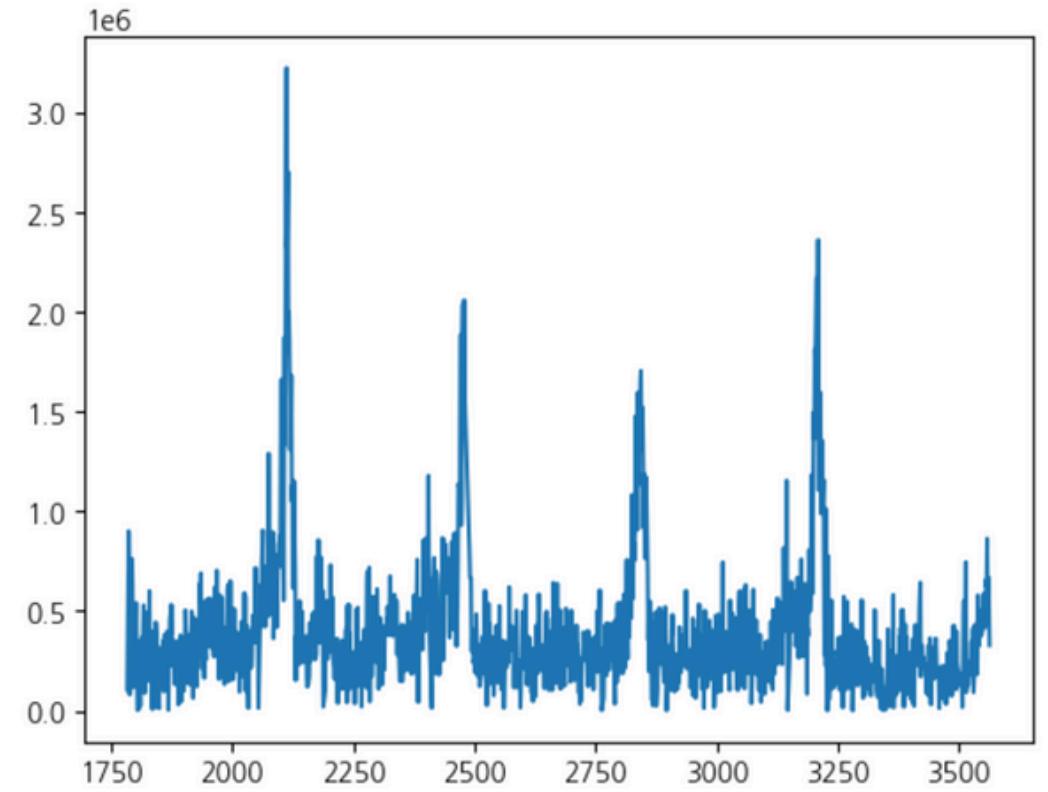
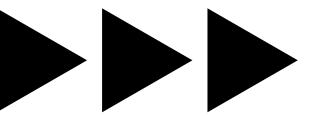
timestep=80

	mae	mse	rmse	r2
서울	1.582590e+06	4.554250e+12	2.134069e+06	0.462866
광주	3.541332e+05	2.642937e+11	5.140951e+05	0.324047
대전	2.226062e+05	9.536536e+10	3.088128e+05	0.530100
부산	1.111262e+05	2.363611e+10	1.537404e+05	0.294375
대구	1.680253e+05	5.424498e+10	2.329055e+05	0.444540

# 모델 개선

베이스

CNN-LSTM  
timestep=80



개선방안

버터워스 필터

사비츠키-골레이 필터

EMD 분해

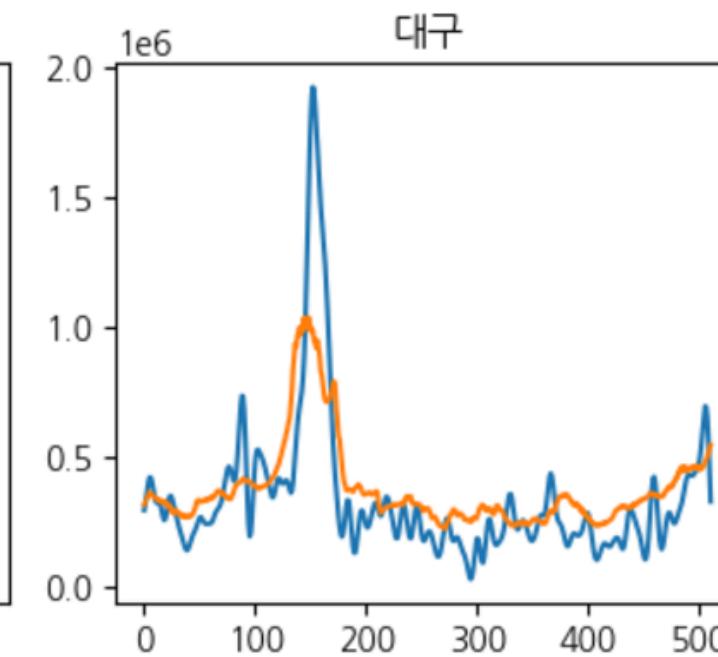
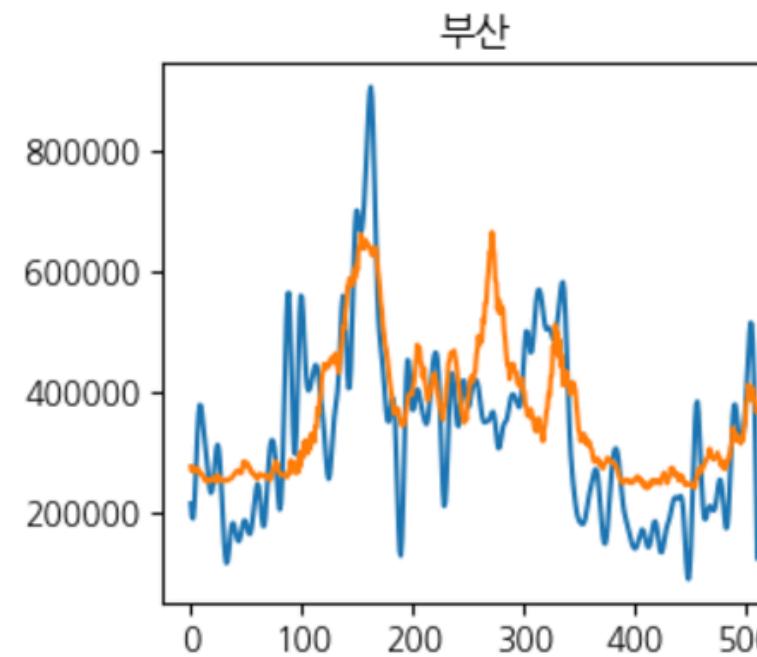
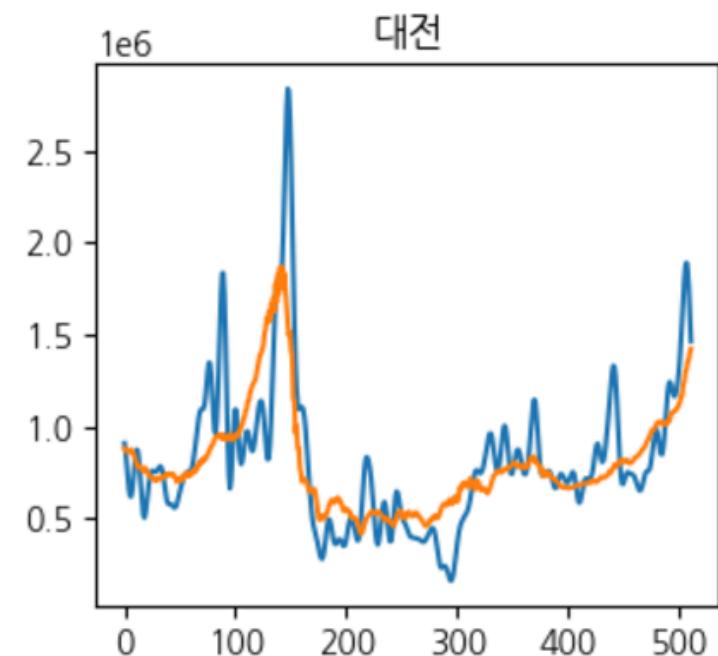
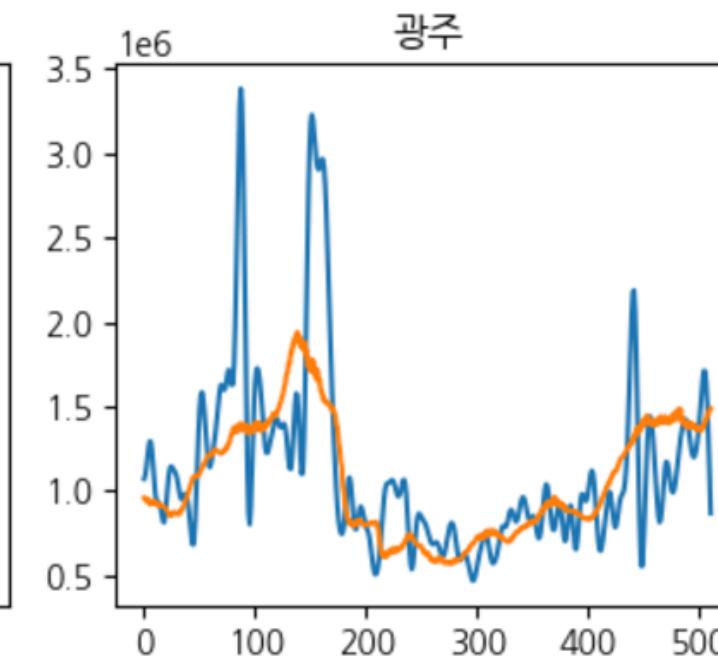
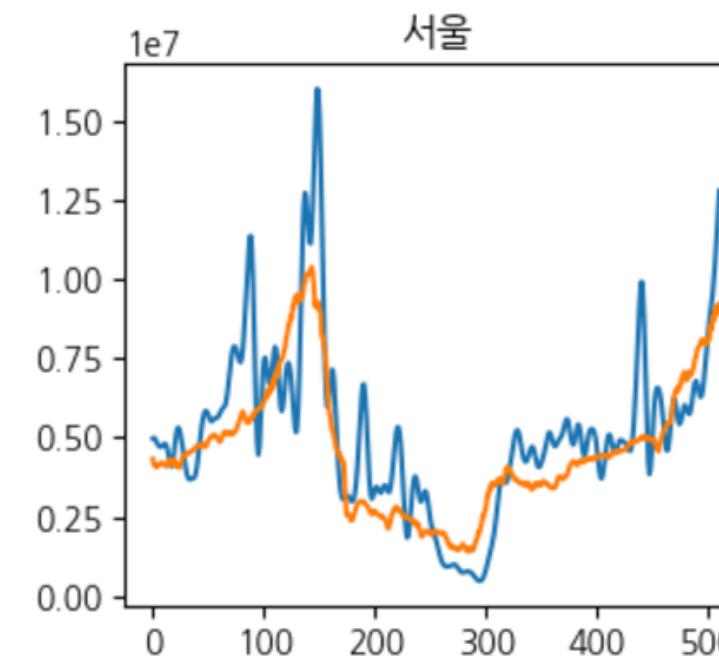
BILSTM 적용

모델 쌓기

진동제어

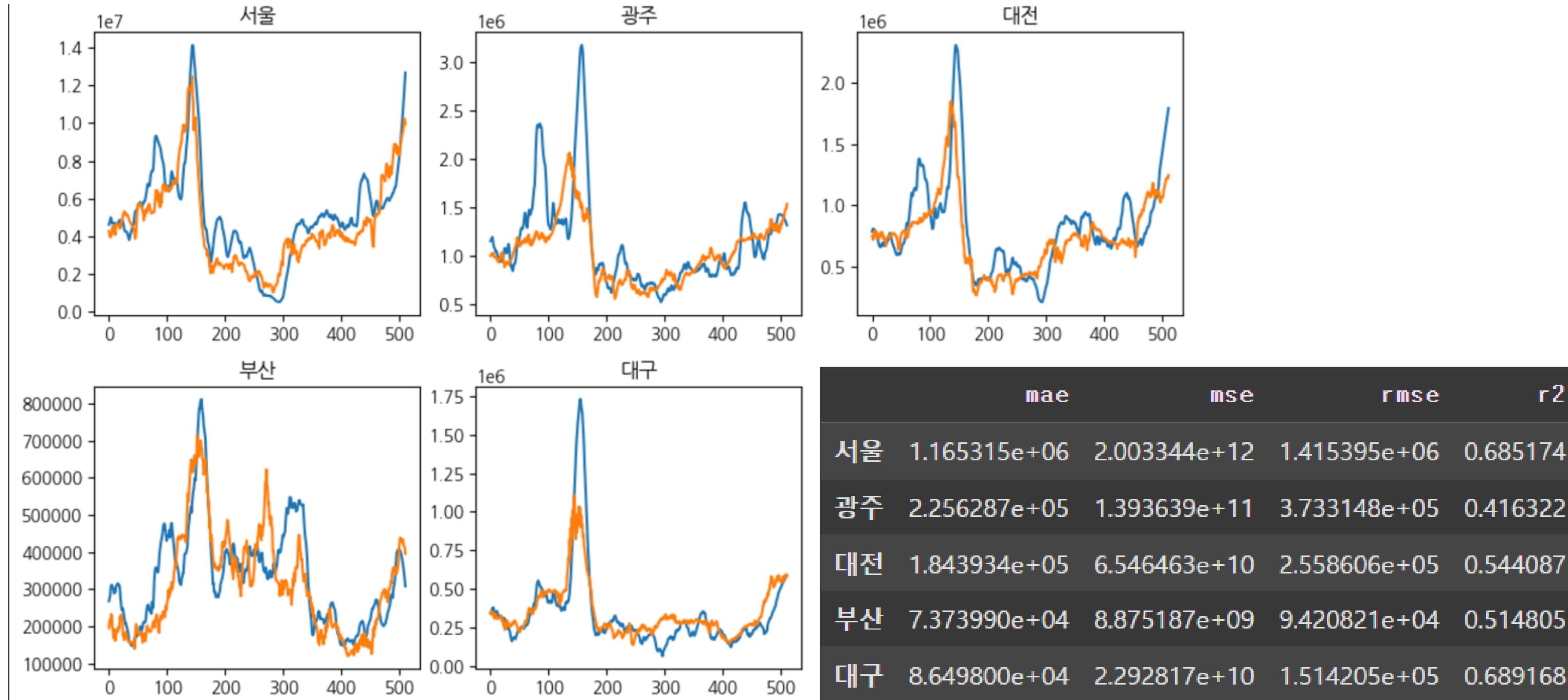
모델 정밀도 개선

# 모델 개선 – 버터워스



	mae	mse	rmse	r2
서울	1.202287e+06	2.640703e+12	1.625024e+06	0.628677
광주	2.653667e+05	1.691579e+11	4.112881e+05	0.423886
대전	1.763899e+05	6.488045e+10	2.547164e+05	0.603731
부산	8.752649e+04	1.165649e+10	1.079652e+05	0.457961
대구	1.230175e+05	3.336073e+10	1.826492e+05	0.584122

# 모델 개선 – 사비츠키 – 골레이



# 모델 개선 - EMD분해

서울

	mae	mse	rmse	r2
0	2.151581e+06	8.728550e+12	2.954412e+06	-0.029456
1	1.909170e+06	6.580855e+12	2.565318e+06	0.223846
2	1.608468e+06	4.863925e+12	2.205431e+06	0.426343
3	1.573496e+06	4.455275e+12	2.110752e+06	0.474539
4	1.550873e+06	4.377196e+12	2.092175e+06	0.483748
5	1.644765e+06	5.162110e+12	2.272028e+06	0.391174
6	1.713632e+06	5.046711e+12	2.246488e+06	0.404785
7	1.998021e+06	7.782124e+12	2.789646e+06	0.082167

광주

	mae	mse	rmse	r2
0	448661.859050	3.974182e+11	630411.123259	-0.016430
1	486477.876744	4.259110e+11	652618.596756	-0.089303
2	456210.018848	3.691130e+11	607546.741260	0.055962
3	380352.135766	3.020759e+11	549614.360196	0.227415
4	364597.015100	2.798828e+11	529039.531442	0.284176
5	359795.520971	2.670296e+11	516749.053606	0.317049
6	368795.834327	2.937431e+11	541980.742433	0.248727
7	339131.439770	2.446721e+11	494643.442442	0.374230

대전

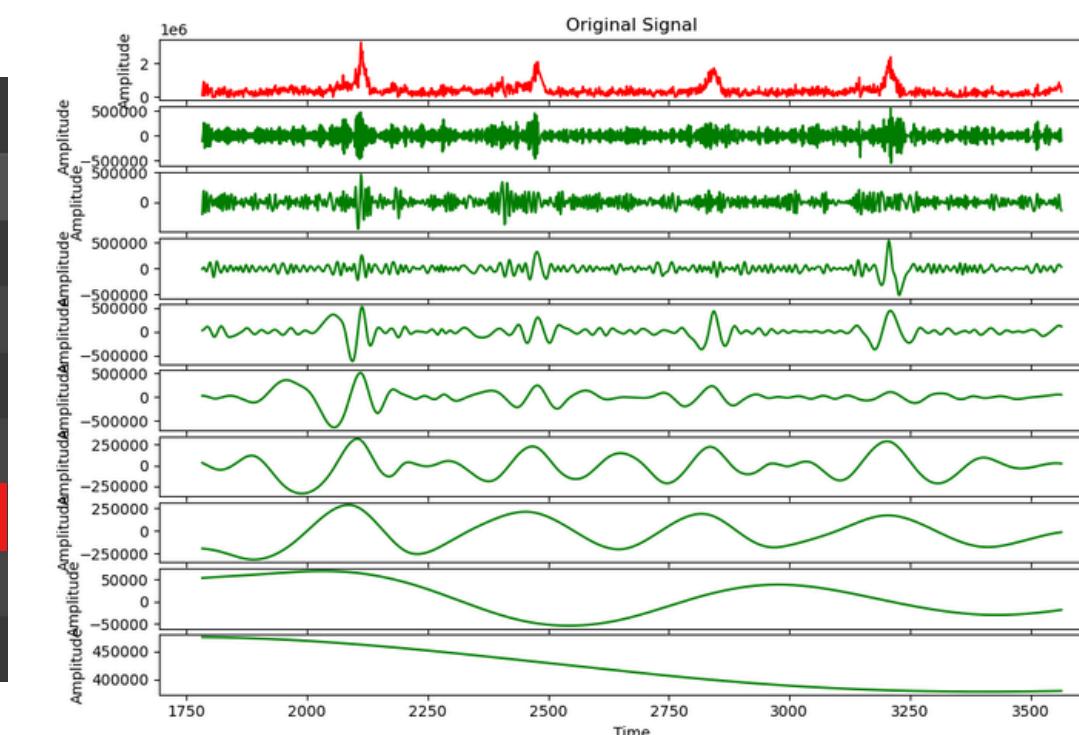
	mae	mse	rmse	r2
0	332258.635090	2.037487e+11	451385.321230	-0.003945
1	292715.775077	1.529792e+11	391125.588018	0.246215
2	262016.885555	1.238669e+11	351947.319416	0.389662
3	282262.921545	1.425016e+11	377493.820911	0.297842
4	225225.298279	9.992258e+10	316105.329396	0.507645
5	227248.777313	1.098687e+11	331464.538555	0.458636
6	230297.305658	1.127688e+11	335810.716186	0.444347
7	270649.780358	1.421452e+11	377021.502923	0.299598

부산

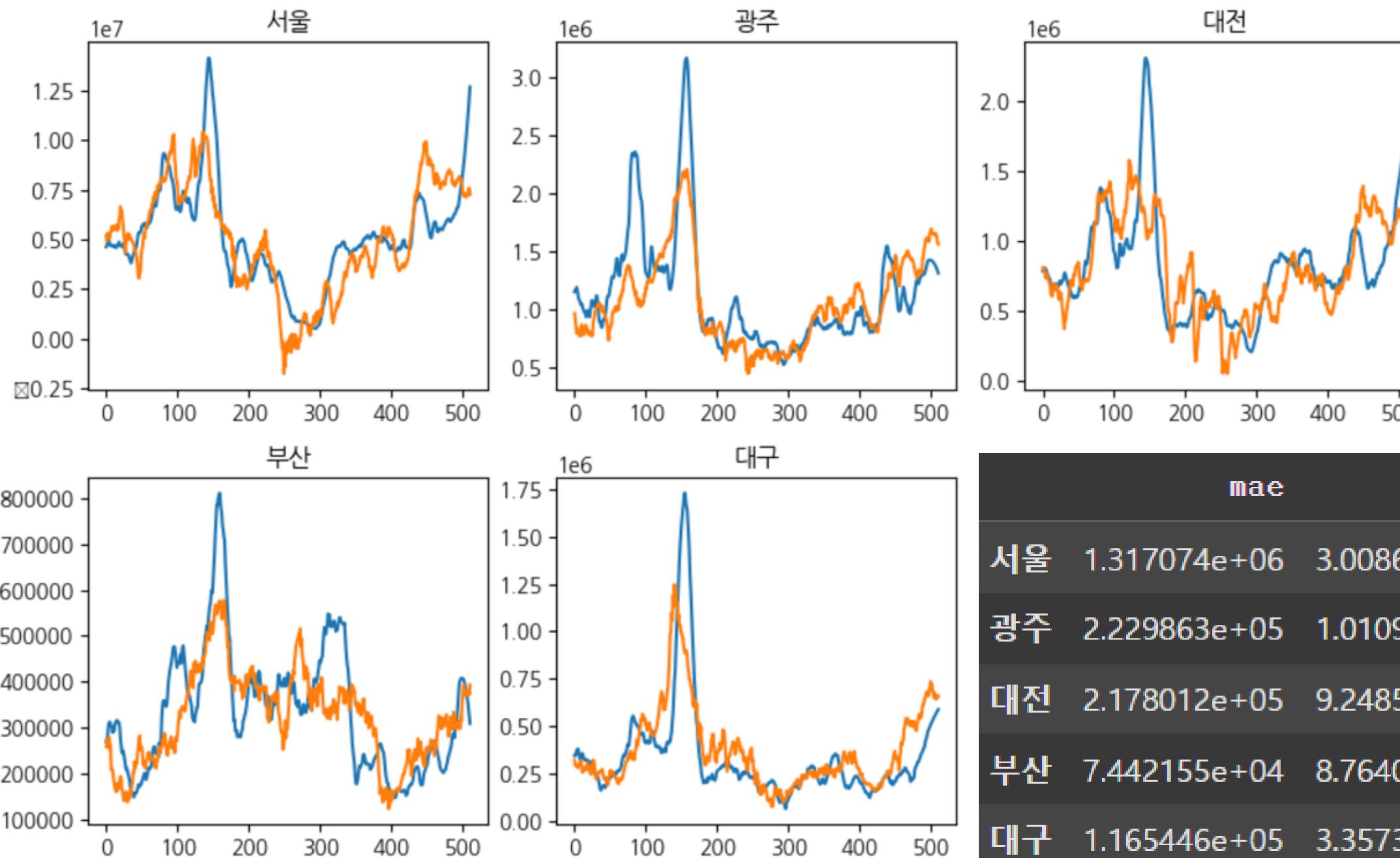
	mae	mse	rmse	r2
0	147191.941900	3.345812e+10	182915.602784	0.001152
1	141373.761311	3.153062e+10	177568.644947	0.058695
2	129156.247462	2.821677e+10	167978.495449	0.157625
3	116614.638710	2.485430e+10	157652.474903	0.258008
4	133219.635669	3.020220e+10	173787.804156	0.098353
5	129047.561163	2.662499e+10	163171.657161	0.205146
6	123757.218644	2.781596e+10	166781.176055	0.169591
7	123479.337448	2.446461e+10	156411.675811	0.269641

대구

	mae	mse	rmse	r2
0	242747.418719	1.062482e+11	325957.373960	-0.087966
1	229029.412971	1.015138e+11	318612.246001	-0.039485
2	183844.452519	7.957441e+10	282089.364434	0.185170
3	156484.757190	5.986345e+10	244670.079351	0.387007
4	156414.018393	5.978814e+10	244516.136807	0.387779
5	152481.363262	5.070063e+10	225168.005204	0.480833
6	156217.925217	5.864710e+10	242171.627603	0.399463
7	167170.348894	5.225309e+10	228589.358530	0.464936

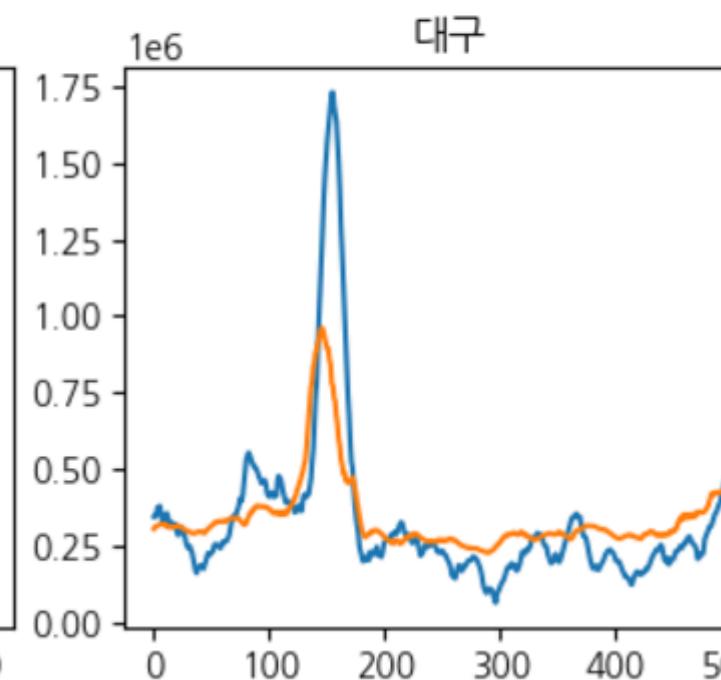
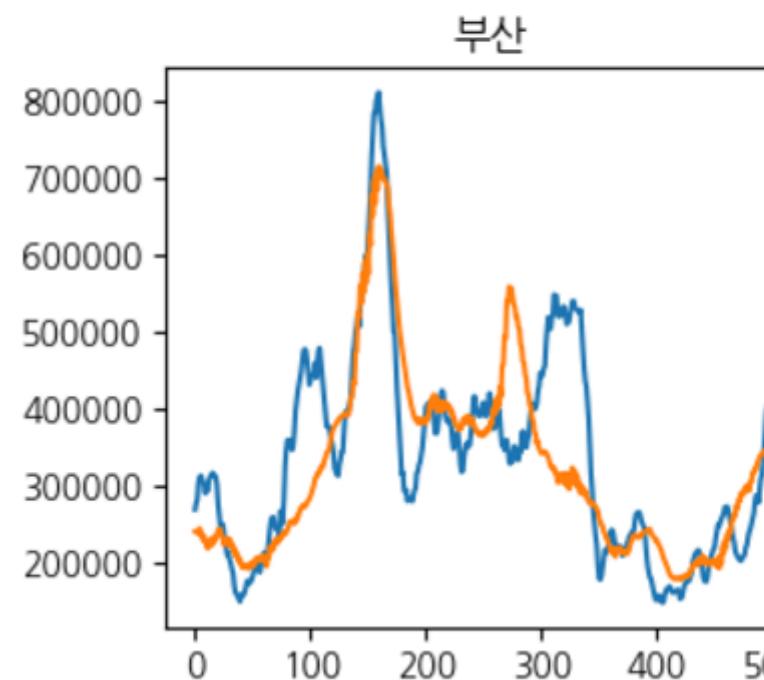
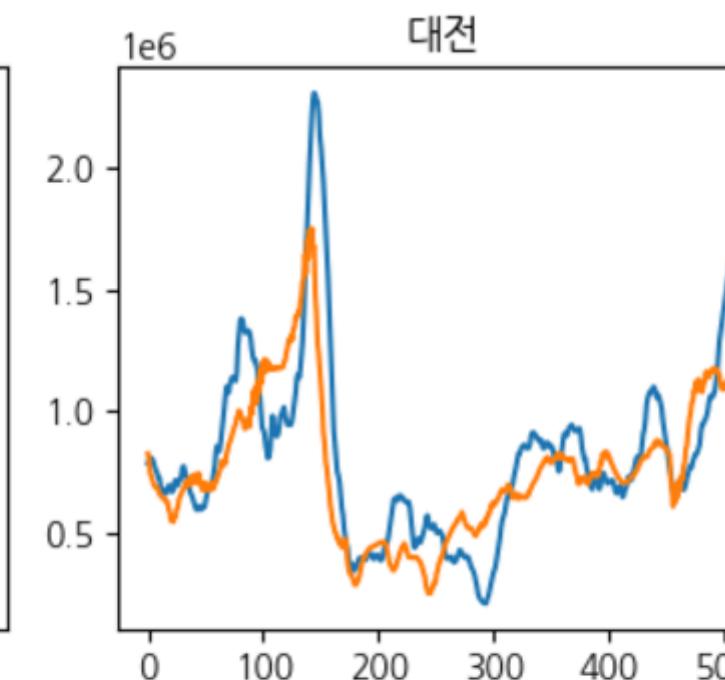
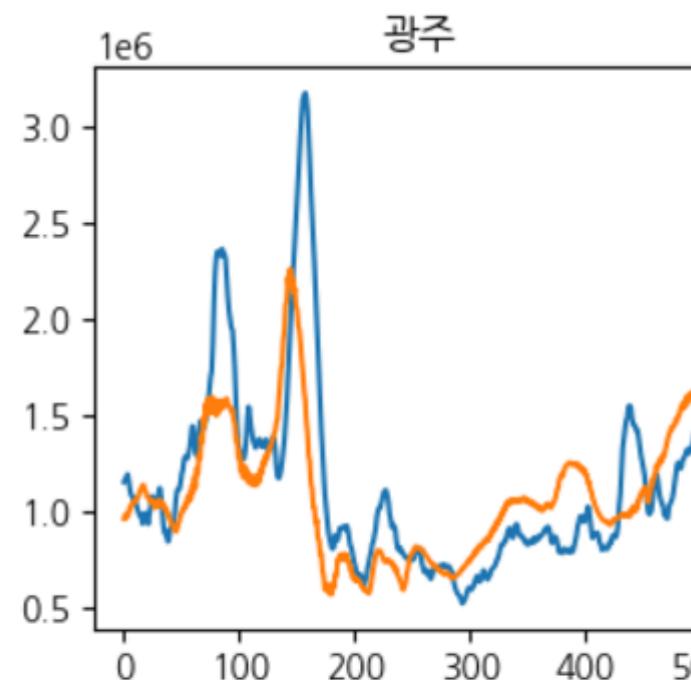
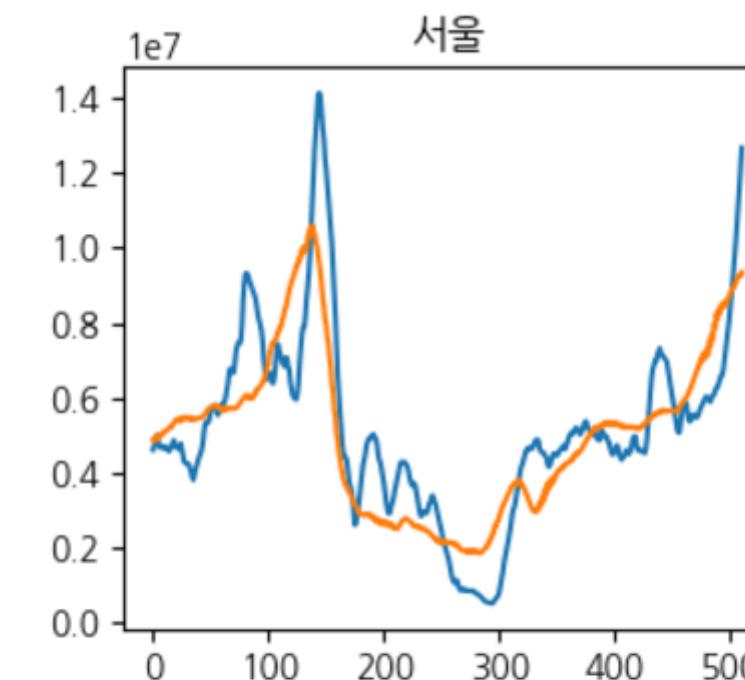


# 모델 개선 – BiLSTM



	mae	mse	rmse	r2
서울	1.317074e+06	3.008658e+12	1.734548e+06	0.527189
광주	2.229863e+05	1.010961e+11	3.179562e+05	0.576593
대전	2.178012e+05	9.248520e+10	3.041138e+05	0.355908
부산	7.442155e+04	8.764095e+09	9.361675e+04	0.520879
대구	1.165446e+05	3.357372e+10	1.832313e+05	0.544849

# 모델 개선 – 레이어 쌓기(2)



	mae	mse	rmse	r2
서울	1.122320e+06	2.045285e+12	1.430135e+06	0.678583
광주	2.382842e+05	1.289276e+11	3.590650e+05	0.460030
대전	1.745652e+05	6.027418e+10	2.455080e+05	0.580234
부산	6.324576e+04	7.764306e+09	8.811530e+04	0.575536
대구	1.035659e+05	3.170483e+10	1.780585e+05	0.570185

# 가격 예측 모델

# 배추가격 예측 데이터프레임

	날짜	지역	배추가격	물가지수	배추 수입량	무가격	예측공급량
0	2020-03-22	광주	9181	99.94	0.0	9181	697378.00
1	2020-03-23	광주	9000	99.94	0.0	9000	706823.70
2	2020-03-24	광주	9000	99.94	0.0	9000	714141.94
3	2020-03-25	광주	9000	99.94	0.0	9000	718902.25
4	2020-03-26	광주	9000	99.94	0.0	9000	725977.30
...	...	...	...	...	...	...	...
8416	2024-10-27	서울	13109	114.69	1870.6	13109	7169453.50
8417	2024-10-28	서울	13200	114.69	1870.6	13200	7028396.00
8418	2024-10-29	서울	14350	114.69	1870.6	14350	7187834.50
8419	2024-10-30	서울	13700	114.69	1870.6	13700	7155807.00
8420	2024-10-31	서울	13200	114.69	1870.6	13200	7383922.50

8421 rows × 7 columns

# 가격 예측 모델

[과정] 목적시계열: 배추 가격

공변량: 물가지수, 배추수입량, 공급량, 무가격

train:test=7:3비율로 데이터셋 분할

Min-Max 스케일링 시행

RNN, CNN-RNN, LSTM, CNN-LSTM 학습

결과를 원래 스케일로 역변환(inverse\_transform)

MAE, MSE, RMSE, R2로 성능평가

# 가격 예측 모델

```
# ----- RNN 모델 -----
rnn_model = Sequential([
    SimpleRNN(50, return_sequences=True, input_shape=(X_train.shape[1], X_train.shape[2])),
    SimpleRNN(50, return_sequences=False),
    Dense(1)
])
rnn_model.compile(optimizer='adam', loss='mse')
rnn_model.fit(X_train, y_train, epochs=100, batch_size=32, verbose=0)
rnn_pred = rnn_model.predict(X_test)
```

RNN

```
# ----- LSTM 모델 -----
lstm_model = Sequential([
    LSTM(50, return_sequences=True, input_shape=(X_train.shape[1], X_train.shape[2])),
    LSTM(50, return_sequences=False),
    Dense(1)
])
lstm_model.compile(optimizer='adam', loss='mse')
lstm_model.fit(X_train, y_train, epochs=100, batch_size=32, verbose=0)
lstm_pred = lstm_model.predict(X_test)
```

LSTM

# 가격 예측 모델

```
# ----- CNN-RNN 모델 -----
cnn_rnn_model = Sequential([
    Conv1D(64, kernel_size=3, activation='relu', input_shape=(X_train.shape[1], X_train.shape[2])),
    MaxPooling1D(pool_size=2),
    SimpleRNN(50, return_sequences=False),
    Dense(1)
])
cnn_rnn_model.compile(optimizer='adam', loss='mse')
cnn_rnn_model.fit(X_train, y_train, epochs=100, batch_size=32, verbose=0)
cnn_rnn_pred = cnn_rnn_model.predict(X_test)
```

CNN-RNN

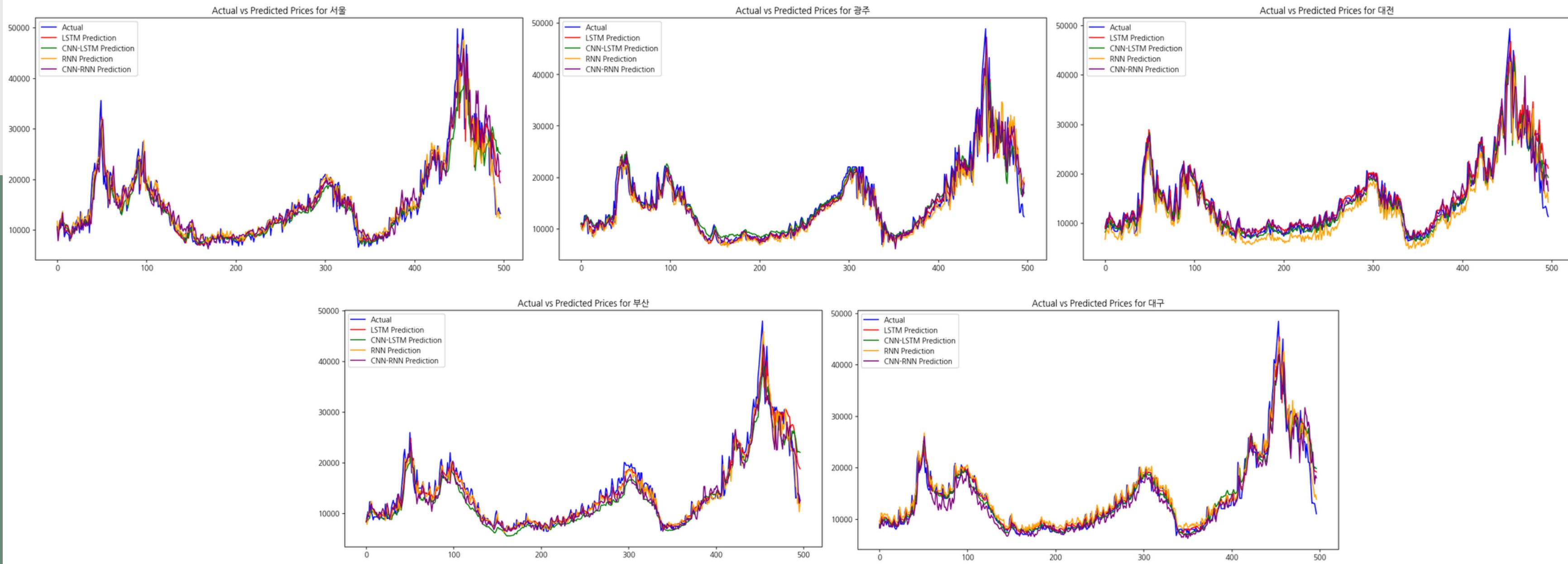
```
# ----- CNN-LSTM 모델 -----
cnn_lstm_model = Sequential([
    Conv1D(64, kernel_size=3, activation='relu', input_shape=(X_train.shape[1], X_train.shape[2])),
    MaxPooling1D(pool_size=2),
    LSTM(50, return_sequences=False),
    Dense(1)
])
cnn_lstm_model.compile(optimizer='adam', loss='mse')
cnn_lstm_model.fit(X_train, y_train, epochs=100, batch_size=32, verbose=0)
cnn_lstm_pred = cnn_lstm_model.predict(X_test)
```

CNN-LSTM

# 가격 예측 모델

지역	광주	대구	대전	부산	서울
LSTM_MAE	1394.235	1192.618	1589.840	1200.902	1517.837
LSTM_MSE	5131704.390	3575544.252	6219832.088	3721988.296	5968965.505
LSTM_RMSE	2265.327	1890.911	2493.959	1929.246	2443.147
LSTM_R^2	0.894	0.935	0.885	0.932	0.909
CNN-LSTM_MAE	1622.599	1311.257	1604.352	1866.673	1922.290
CNN-LSTM_MSE	6249734.008	4344844.547	5794013.460	7081093.403	9325850.709
CNN-LSTM_RMSE	2499.947	2084.429	2407.076	2661.032	3053.826
CNN-LSTM_R^2	0.871	0.921	0.893	0.871	0.858
RNN_MAE	1586.358	1271.674	2159.510	1100.803	1434.831
RNN_MSE	6180310.008	2912364.365	7208535.335	2903787.378	4764723.087
RNN_RMSE	2486.023	1706.565	2684.872	1704.050	2182.825
RNN_R^2	0.873	0.947	0.867	0.947	0.927
CNN-RNN_MAE	1599.811	1567.633	1780.416	1569.516	1798.238
CNN-RNN_MSE	6416568.799	5406699.429	7070325.456	5290041.610	7049401.058
CNN-RNN_RMSE	2533.095	2325.231	2659.008	2300.009	2655.071
CNN-RNN_R^2	0.868	0.902	0.869	0.904	0.893

# 가격 예측 모델



# 결과 및 기대효과

# 분석 결과



## 설명력 ( $R^2$ )

서울: 0.685  
대구: 0.689  
대전: 0.544  
부산: 0.515  
광주: 0.416



## $R^2$ 가 가장 높은 모델

서울: RNN,  $R^2 = 0.947$   
대구: CNN-LSTM,  $R^2 = 0.902$   
대전: RNN,  $R^2 = 0.947$   
부산: RNN,  $R^2 = 0.930$   
광주: LSTM,  $R^2 = 0.894$

# 기대효과

## 농민, 유통업계

### (1) 수익안정화

- 예측된 가격 데이터를 기반으로 한 재배 및 출하 계획 수립
- 과잉 생산 방지

### (2) 생산비 절감

- 효율적인 농업 자원 관리

### (3) 물류비, 재고 최적화

- 자원 낭비 줄이고 환경적 부담 완화

## 정부

### (1) 가격 안정화 정책 효율성 강화

- 공급량 조절 및 시장 개입 시점 결정
- 효율적인 수입 정책 수립

### (2) 소비자 물가 관리

- 소비자 물가 안정

### (3) 위기 대응 체계 개선

- 기상 변화로 인한 피해 관리

## 소비자

### (1) 가계 부담 경감

- 짐장철 등 특정 시기 가격 안정
- 구매 시기 및 방법 결정 지원

### (2) 가격 정보 제공으로 소비자 만족도 향상

- 투명한 정보 제공

### (3) 다양한 대체 소비 전략 개발

- 대체 품목으로 전환 가능

# 마무리 및 소감

## 참고 문헌

- 이창민, 송성광, 정성욱. (2022). 앙상블 Voting 기법을 활용한 배추 가격 예측에 관한 연구. 융합정보논문지, 12(3), 1–10, <https://doi.org/10.22156/CS4SMB.2022.12.03.001>
- 김용석, 심교문, 정명표, 최인태. (2015). 배추 생산량 추정에 영향을 미치는 기상요소 연구. 한국기후변화학회지, 6(4), 277–281, [10.15531/ksccr.2015.6.4.277](https://doi.org/10.15531/ksccr.2015.6.4.277) [10.15531/ksccr.2015.6.4.277](https://doi.org/10.15531/ksccr.2015.6.4.277)
- 김성우, 노호영, 윤선희. (2016-04-28). 배추, 무 월별 소비량 추정 및 시사점. 한국농촌경제연구원
- KiJeon Nam, Soonho Hwangbo, ChangKyoo Yoo,
- A deep learning-based forecasting model for renewable energy scenarios to guide sustainable energy policy: A case study of Korea, Renewable and Sustainable Energy Reviews, Volume 122, 2020, 109725, ISSN 1364–0321

감사합니다