날씨 요인에 따른 농산물 가격 예측

2조
최현묵,임병남

* 페이지 내 인물 사진은 샘플이미지 입니다.

TABLE OF CONTENTS

* 페이지 내 인물 사진은 샘플이미지 입니다.

프로젝트 개요

프로젝트 주제 선정 배경 및 기대효과

1. 농산물 가격 변화에 따른 문제 발생

- 소비자 부담 증가 : 가격 급등으로 인해 소비자의 경제적인 부담감이 커집니다
- 농가의 수익성 불안정 : 가격 급락은 농가 수익성을 악화시키며, 농업의 지속 가능성을 위협합니다
- 국가 경제에 영향 : 농산물 수출입 및 국내 소비 경제의 균형을 해칠 수 있습니다.

2. 농산물 가격 예측 모델의 기대 효과

- 소비자 보호 : 가격 변동성을 예측하여 소비자들이 더 나은 구매 계획을 세울 수 있도록 도움
- 농업인 지원 : 농산물 생산 및 판매 전략을 세울때 도움
- 정책적 활용 : 정부가 농산물 수급 조절 정책을 수립하거나 긴급 구호 계획을 세울 수 있도록 도움

프로젝트 개요

공공데이터 출처

농넷 (농산물유통 종합정보시스템)

- **데이터 내용:** 농산물 거래량, 가격, 품종, 생산지, 수입 데이터 등

농사로

- **데이터 내용:** 농산물 재배방식 정보

기상청

- **데이터 내용:** 농업 기상 데이터(강수량, 온도, 풍속 등)

프로젝트 개요

일정 계획

1. **프로젝트 기획** (2024.11.20 ~ 2024.11.20)
    1. 비즈니스 목표 설정
    2. 데이터 소스 조사

1. **분석 환경 구축** (2024.11.21 ~ 2024.11.21)
    2. 데이터 수집 환경
    3. 협업 환경

1. **데이터 수집 및 가공** (2024.11.21 ~ 2024.11.25)
    2. 데이터 수집
    3. 기초 전처리 및 저장

1. **탐색적 데이터 분석 및 전처리** (2024.11.26 ~ 2024.11.29)
    2. 기초 통계량 분석 및 분석 시각화
    3. 데이터 전처리

1. **예측 모델링** (2024.11.29 ~ 2024.12.03)
    2. 예측 모델 개발 및 모델 적용

1. **발표 및 시연 준비** (2024.12.03 ~2024.12.04)

데이터 수집 및 전처리

수집 데이터

양배추 데이터

	DATE	거래단위	평균가격	총거래물량	총거래금액	도매시장	도매법인	품목	품종	산지-광역시도	산지-시군구	등급
0	2024-11	8kg그물망	7570.347368	2263192.0	2141643700	서울가락도매	대아청과	양배추	양배추(일반)	충청남도	서산시	특
1	2024-11	8kg그물망	8394.011827	1803288.0	1892102600	서울가락도매	대아청과	양배추	양배추(일반)	전라남도	무안군	특
2	2024-11	8kg그물망	9639.916816	921336.0	1110200300	서울가락도매	대아청과	양배추	양배추(일반)	강원도	정선군	특
3	2024-11	8kg그물망	7854.459906	420816.0	413160300	서울가락도매	대아청과	양배추	양배추(일반)	충청남도	태안군	특
4	2024-11	8kg그물망	6939.585367	417912.0	362517000	서울가락도매	대아청과	양배추	양배추(일반)	충청남도	당진시	특

데이터 수집 및 전처리

양배추 데이터 전처리

- 거래단위가 제각각이라 평균가격을 예측하기 위해 1kg 단위로 총거래물량과 평균가격을 변환
- 거래단위에 kg 앞에 있는 숫자 추출하여 평균가격과 총거래물량을 1kg단위로 변환

	DATE	거래단위	평균가격	총거래물량	총거래금액	도매시장	도매법인	품목	품종	산지-광역시도	산지-시군구	등급	1kg_평균가격	1kg_물량
0	2024-11	8kg그물망	7570.347368	2263192.0	2141643700	서울가락도매	대아청과	양배추	양배추(일반)	충청남도	서산시	특	946.293421	18105536.0
1	2024-11	8kg그물망	8394.011827	1803288.0	1892102600	서울가락도매	대아청과	양배추	양배추(일반)	전라남도	무안군	특	1049.251478	14426304.0

데이터 수집 및 전처리

양배추 데이터 전처리

- 양배추의 피처들 확인해서 전처리해야 할 것들 처리
- 피처들의 타입 확인
- DATE 피처 -> datetime 타입으로 변경

```
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   DATE       21414 non-null  object
 1   거래단위       21414 non-null  object
 2   평균가격       21414 non-null  float64
 3   총거래물량      21414 non-null  float64
 4   총거래금액      21414 non-null  int64
 5   도매시장       21414 non-null  object
 6   도매법인       21414 non-null  object
 7   품목         21414 non-null  object
 8   품종         21414 non-null  object
 9   산지-광역시도    21414 non-null  object
 10  산지-시군구     21414 non-null  object
 11  등급         21414 non-null  object
 12  1kg_평균가격  21414 non-null  float64
 13  1kg_물량     21414 non-null  int64
```

```
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   DATE       21414 non-null  datetime64[ns]
 1   거래단위       21414 non-null  object
 2   평균가격       21414 non-null  float64
 3   총거래물량      21414 non-null  float64
 4   총거래금액      21414 non-null  int64
 5   도매시장       21414 non-null  object
 6   도매법인       21414 non-null  object
 7   품목         21414 non-null  object
 8   품종         21414 non-null  object
 9   산지-광역시도    21414 non-null  object
 10  산지-시군구     21414 non-null  object
 11  등급         21414 non-null  object
 12  1kg_평균가격  21414 non-null  float64
 13  1kg_물량     21414 non-null  int64
```

데이터 수집 및 전처리

양배추 데이터 전처리

- 산지-광역시도 전처리
- 피처 확인하면 똑같은 지역이지만 명칭이 다른 지역들이 있기에 기준 디렉토리 만들어서 통일 시켜주었다
- 또한 우리나라가아닌 뉴질랜드, 수입산과 같은 데이터는 제거

```
array(['충청남도', '전라남도', '강원도', '경상북도', '충남', '전남', '서울', '전라북도', '중국',
       '경기도', '대구광역시', '제주도', '부산', '강원', '부산광역시', '-', '서울특별시', '충청북도',
       '경상남도', '인천', '충북', '경북', '전북', '경남', '제주', '경기', '인천광역시', '울산광역시',
       '대구', '광주광역시', '광주', '울산', '대전광역시', '대전', '뉴질랜드', '수입산'],
      dtype=object)
```

```
array(['충청남도', '전라남도', '강원도', '경상북도', '서울특별시', '전라북도', '경기도', '대구광역시',
       '제주도', '부산광역시', '충청북도', '경상남도', '인천광역시', '울산광역시', '광주광역시', '대전광역시'],
      dtype=object)
```

```
region_sum = {
    '충남': '충청남도',
    '강원': '강원도',
    '제주':'제주도',
    '전남' : '전라남도',
    '전북' : '전라북도',
    '강원' : '강원도',
    '충북' : '충청북도',
    '부산' : '부산광역시',
    '경남' : '경상남도',
    '서울' : '서울특별시',
    '광주' : '광주광역시',
    '경북' : '경상북도',
    '경기' : '경기도',
    '인천' : '인천광역시',
    '대구' : '대구광역시',
    '울산' : '울산광역시',
    '대전' : '대전광역시'
```

데이터 수집 및 전처리

양배추 데이터 전처리

- 도매시장 전처리
- 도매시장이 피처로 들어가기 때문에 수집한 기간중에 데이터가 부족한 도매시장은 제거해주는 처리 진행
- 기준은 연도별 최소 6개월 이상의 데이터를 가지고 있는 도매시장
- 32개 도매시장 -> 29개 도매시장

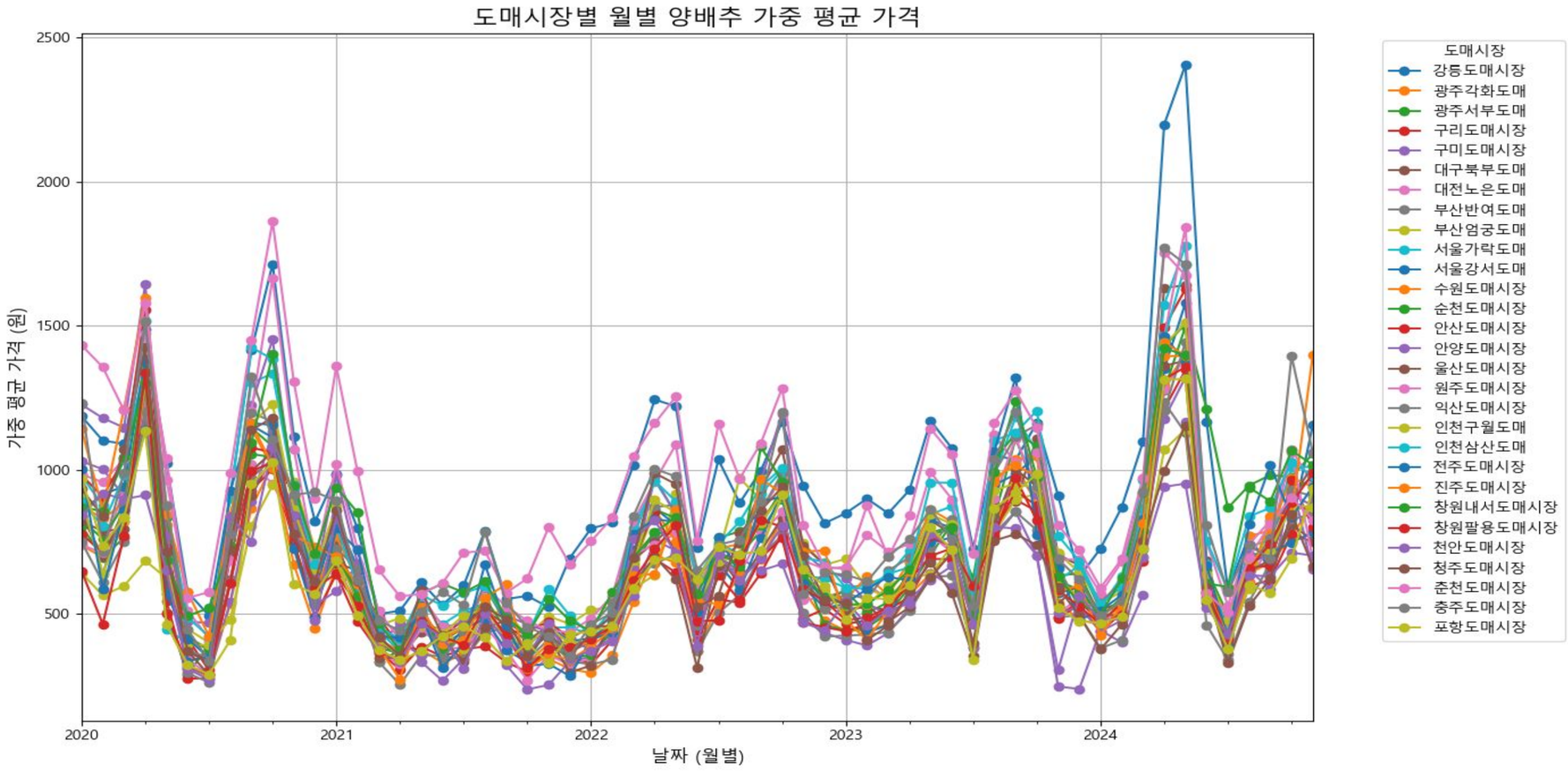
```
array(['서울가락도매', '인천삼산도매', '안산도매시장', '춘천도매시장', '대구북부도매', '부산반여도매',
       '강릉도매시장', '충주도매시장', '포항도매시장', '광주서부도매', '안양도매시장', '순천도매시장',
       '청주도매시장', '창원내서도매시장', '인천구월도매', '울산도매시장', '원주도매시장', '광주각화도매',
       '창원팔용도매시장', '전주도매시장', '구리도매시장', '수원도매시장', '부산엄궁도매', '구미도매시장',
       '서울강서도매', '진주도매시장', '천안도매시장', '대전오정도매', '정읍도매시장', '대전노은도매',
       '익산도매시장', '안동도매시장'], dtype=object)
```

```
array(['서울가락도매', '인천삼산도매', '안산도매시장', '춘천도매시장', '대구북부도매', '부산반여도매',
       '강릉도매시장', '충주도매시장', '광주서부도매', '안양도매시장', '순천도매시장', '청주도매시장',
       '창원내서도매시장', '인천구월도매', '울산도매시장', '원주도매시장', '포항도매시장', '광주각화도매',
       '창원팔용도매시장', '전주도매시장', '구리도매시장', '수원도매시장', '부산엄궁도매', '구미도매시장',
       '서울강서도매', '진주도매시장', '천안도매시장', '대전노은도매', '익산도매시장'], dtype=object)
```

데이터 수집 및 전처리

양배추 데이터 전처리

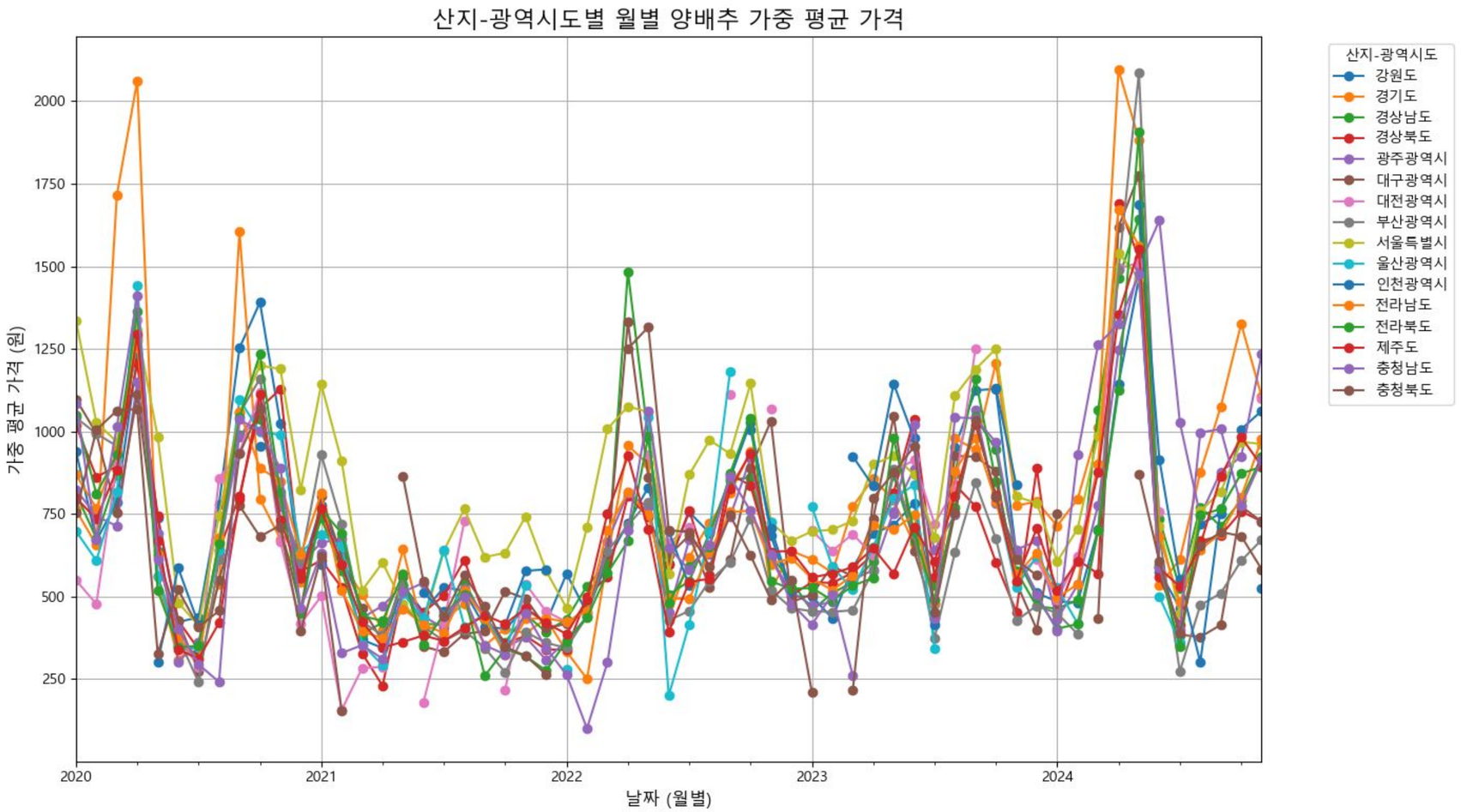
- 도매시장별 가격 시각화
- 가격변화의 추세를 보기 위해 시각화하여 확인
- 물량을 기준으로 가중치를 주어 평균가격을 계산
- 특정 월마다 가격차이가 나고 도매시장별로도 가격차이가 있는 것을 확인



데이터 수집 및 전처리

양배추 데이터 전처리

- 산지-광역시도별 가격 시각화
- 방식은 도매시장과 똑같은 방식으로 진행
- 전체적인 추세는 비슷하게 보이는 것으로 확인



데이터 수집 및 전처리

기후 데이터 전처리

- 기후 데이터 확인
- 양배추의 DATE, 산지-광역시도에 맞춰 기후데이터를 병합
- 지점명을 광역시도로 바꾸는 작업 진행

	지점	지점명	일시	평균기온(℃)	평균풍속(m/s)	월강수량합(mm)
0	12	안면도(감)	2019-01	1.3	2.6	0.5
1	12	안면도(감)	2019-02	1.7	2.6	15.0
2	12	안면도(감)	2019-03	6.1	2.6	28.0
3	12	안면도(감)	2019-04	10.4	2.1	49.5
4	12	안면도(감)	2019-05	15.8	2.7	19.0
...	...	...	...	...	...	...
36805	996	화동	2024-06	21.4	2.5	122.0
36806	996	화동	2024-07	24.3	2.5	584.0
36807	996	화동	2024-08	25.8	2.3	34.5

데이터 수집 및 전처리

기후 데이터 전처리

- 기후 관측지 정보데이터를 수집하여 지점주소에 첫번째 단어를 가져와서 광역시도 컬럼으로 저장
- 기후데이터와 관측지 정보데이터에서 지점이 일치하면 기후데이터의 광역시도 컬럼이 추가되게 진행

	지점	시작일	종료일	지점명	지점주소	관리관서	위도	경도	노장해발고도(m)	기압계(관측장비지상높이(m))	기온계(관측장비지상높이(m))	풍속계(관측장비지상높이(m))	강우계(관측장비지상높이(m))
0	12	2007-11-30	NaN	안면도(감)	NaN	NaN	36.5333	126.3167	60.00	NaN	NaN	NaN	NaN
1	96	2020-10-23	NaN	독도	경상북도 울릉군울릉읍 독도이사부길63	울릉도기상대(115)	37.2395	131.8698	99.08	NaN	NaN	NaN	NaN
2	96	2009-10-27	2020-10-23	독도	경상북도 울릉군울릉읍 독도이사부길63	울릉도기상대(115)	37.2395	131.8698	96.15	NaN	NaN	NaN	NaN
3	116	2023-10-21	NaN	관악(레)	경기도 과천시 중앙로 관악산길(관악산기상레이더관측소)	NaN	37.4453	126.9640	624.82	NaN	NaN	NaN	NaN

데이터 수집 및 전처리

기후 데이터 전처리

- 기후 데이터의 추가한 광역시도 컬럼 확인
- 이름이 다른 광역시도 양배추 데이터 산지-광역시도 전처리처럼 똑같이 진행
- 최종 기후 데이터 확인

```
array(['강원특별자치도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시',
       '부산광역시', '서귀포시', '서울특별시', '세종특별자치시', '울산광역시', '인천광역시', '전라남도',
       '전북특별자치도', '제주특별자치도', '충청남도', '충청북도'], dtype=object)
```

```
array(['강원도', '경기도', '경상남도', '경상북도', '광주광역시', '대구광역시', '대전광역시', '부산광역시',
       '제주도', '서울특별시', '세종시', '울산광역시', '인천광역시', '전라남도', '전라북도', '충청남도',
       '충청북도'], dtype=object)
```

최종 기후 데이터

	광역시도	연도	월	평균기온	평균풍속	평균강수량
0	강원도	2019	1	-3.48	2.06	2.07
1	강원도	2019	2	-1.12	1.82	22.44
2	강원도	2019	3	4.30	2.18	41.57
3	강원도	2019	4	9.15	2.13	55.95
4	강원도	2019	5	16.84	2.24	18.88
...	...	...	...	...	...	...

데이터 수집 및 전처리

양배추 + 기후 데이터 전처리

- 양배추와 기후 데이터 DATE와 광역시도를 기준으로 병합
- 필요없는(사용하지 않는) 피처 제거 후 최종적인 데이터 확인

	DATE	거래 단위	평균가격	총거래물량	도매시장	산지-광역시도	1kg_평균가격	1kg_물량	Year	Month	광역시도	평균기온	평균풍속	평균강수량
87	2024-10-01	8kg 그물망	9056.759985	2853024.0	서울가락도매	강원도	1132.094998	22824192	2024	10	강원도	12.93	1.56	130.36
88	2024-10-01	8kg 그물망	5948.040999	1297192.0	서울가락도매	충청남도	743.505125	10377536	2024	10	충청남도	16.32	2.29	92.38
89	2024-10-01	8kg 그물망	9985.847533	1292552.0	서울가락도	강원도	1248.230942	10340416	2024	10	강원도	12.93	1.56	130.36

데이터 수집 및 전처리

양배추 + 기후 데이터 전처리

- 결측치 및 이상치 확인
- 결측치는 존재하지 않는걸로 확인되고
- 기술통계로 데이터 확인

```
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   DATE       20538 non-null  datetime64[ns]
 1   도매시장       20538 non-null  object
 2   산지-광역시도    20538 non-null  object
 3   1kg_물량     20538 non-null  int64
 4   1kg_평균가격   20538 non-null  float64
 5   Year       20538 non-null  int64
 6   Month      20538 non-null  int64
 7   평균기온       20538 non-null  float64
 8   평균풍속       20538 non-null  float64
 9   평균강수량      20538 non-null  float64
dtypes: datetime64[ns](1), float64(4), int64(3), object(2)
```

	단위_무게(kg)	1kg_물량	1kg_평균가격	Year	Month	평균기온	평균풍속	평균강수량
count	20538.00	20538.00	20538.00	20538.00	20538.00	20538.00	20538.00	20538.00
mean	9.61	263517.83	689.21	2022.02	6.88	15.02	2.21	156.64
std	1.52	1630234.87	389.35	1.38	3.19	8.08	0.79	135.80
min	8.00	64.00	30.00	2020.00	1.00	-5.45	0.75	0.00
25%	8.00	9728.00	423.62	2021.00	5.00	8.28	1.60	53.20
50%	10.00	35775.00	601.05	2022.00	7.00	16.32	2.03	119.75
75%	10.00	121649.25	857.38	2023.00	10.00	22.11	2.62	225.64
max	15.00	51842752.00	7120.00	2024.00	12.00	28.99	4.34	706.06

데이터 수집 및 전처리

양배추 + 기후 데이터 전처리

- 결측치 및 이상치 확인
- 결측치는 존재하지 않는걸로 확인되고
- 기술통계로 데이터 확인

```
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   DATE       20538 non-null  datetime64[ns]
 1   도매시장       20538 non-null  object
 2   산지-광역시도    20538 non-null  object
 3   1kg_물량     20538 non-null  int64
 4   1kg_평균가격   20538 non-null  float64
 5   Year       20538 non-null  int64
 6   Month      20538 non-null  int64
 7   평균기온       20538 non-null  float64
 8   평균풍속       20538 non-null  float64
 9   평균강수량      20538 non-null  float64
dtypes: datetime64[ns](1), float64(4), int64(3), object(2)
```

	단위_무게(kg)	1kg_물량	1kg_평균가격	Year	Month	평균기온	평균풍속	평균강수량
count	20538.00	20538.00	20538.00	20538.00	20538.00	20538.00	20538.00	20538.00
mean	9.61	263517.83	689.21	2022.02	6.88	15.02	2.21	156.64
std	1.52	1630234.87	389.35	1.38	3.19	8.08	0.79	135.80
min	8.00	64.00	30.00	2020.00	1.00	-5.45	0.75	0.00
25%	8.00	9728.00	423.62	2021.00	5.00	8.28	1.60	53.20
50%	10.00	35775.00	601.05	2022.00	7.00	16.32	2.03	119.75
75%	10.00	121649.25	857.38	2023.00	10.00	22.11	2.62	225.64
max	15.00	51842752.00	7120.00	2024.00	12.00	28.99	4.34	706.06

데이터 수집 및 전처리

양배추 + 기후 데이터 전처리

- 기술 통계 확인 결과 타겟변수로 설정할 1kg_평균가격의 극단값들이 존재하는 것을 확인
- 최대값과 최소값을 확인하면 결측치로 판단해도 되는정도의 극단값들이 존재
- IQR 방식으로 이상치 제거

	단위_무게(kg)	1kg_물량	1kg_평균가격	Year	Month	평균기온	평균풍속	평균강수량
count	20538.00	20538.00	20538.00	20538.00	20538.00	20538.00	20538.00	20538.00
mean	9.61	263517.83	689.21	2022.02	6.88	15.02	2.21	156.64
std	1.52	1630234.87	389.35	1.38	3.19	8.08	0.79	135.80
min	8.00	64.00	30.00	2020.00	1.00	-5.45	0.75	0.00
25%	8.00	9728.00	423.62	2021.00	5.00	8.28	1.60	53.20
50%	10.00	35775.00	601.05	2022.00	7.00	16.32	2.03	119.75
75%	10.00	121649.25	857.38	2023.00	10.00	22.11	2.62	225.64
max	15.00	51842752.00	7120.00	2024.00	12.00	28.99	4.34	706.06

데이터 수집 및 전처리

양배추 + 기후 데이터 전처리

- IQR 방식으로 제거하였지만 극단값들이 제거되지 않은것으로 보임
- 가격의 분포가 너무 커서 제거가 정상적으로 안되는 것으로 판단
- 최종적으로 신뢰구간으로 극단값 제거하기로 결정
- 하위 10%, 상위 10% 경계를 계산 후 필터링

	단위_무게(kg)	1kg_물량	1kg_평균가격	Year	Month	평균기온	평균풍속	평균강수량
count	19846.000000	1.984600e+04	19846.000000	19846.000000	19846.000000	19846.000000	19846.000000	19846.000000
mean	9.625617	2.660831e+05	656.216115	2022.002872	6.913282	15.028425	2.205768	157.799859
std	1.526425	1.646686e+06	336.263150	1.366644	3.198669	8.170307	0.796235	136.801094
min	8.000000	6.400000e+01	30.000000	2020.000000	1.000000	-5.450000	0.750000	0.000000
25%	8.000000	9.900000e+03	417.910593	2021.000000	5.000000	8.250000	1.600000	53.140000
50%	10.000000	3.600000e+04	591.891034	2022.000000	7.000000	16.460000	2.030000	120.230000
75%	10.000000	1.229680e+05	824.972160	2023.000000	10.000000	22.160000	2.620000	226.760000
max	15.000000	5.184275e+07	5775.000000	2024.000000	12.000000	28.990000	4.340000	706.060000

데이터 수집 및 전처리

양배추 + 기후 데이터 전처리

- 다시 기술통계 확인결과 1kg_평균가격의 극단값들은 많이 줄어든것으로 확인

	단위_무게(kg)	1kg_물량	1kg_평균가격	Year	Month	평균기온	평균풍속	평균강수량
count	16437.000000	1.643700e+04	16437.000000	16437.000000	16437.000000	16437.000000	16437.000000	16437.000000
mean	9.650119	2.921856e+05	639.853430	2022.051408	6.847052	14.858885	2.235477	155.350192
std	1.544561	1.763690e+06	217.823745	1.352408	3.250564	8.240877	0.816514	133.810646
min	8.000000	6.400000e+01	312.500000	2020.000000	1.000000	-5.450000	0.750000	0.000000
25%	8.000000	1.081600e+04	458.333333	2021.000000	4.000000	7.820000	1.600000	54.540000
50%	10.000000	4.000000e+04	600.939035	2022.000000	7.000000	16.280000	2.040000	118.630000
75%	10.000000	1.430000e+05	798.986486	2023.000000	10.000000	22.110000	2.640000	223.710000
max	15.000000	5.184275e+07	1160.000000	2024.000000	12.000000	28.990000	4.340000	706.060000

탐색적 분석 (EDA)

탐색적 분석

- 상관관계 확인
- 농산물(양배추)의 기후데이터가 고려될 수 있으려면 양배추가 자라는 기간의 기후데이터가 수확기의 물량과 매칭 시켜야함
- 각 재배 방식에 대한 정의 및 물량과 기후와의 상관관계분석 진행

```
# 각 재배 방식에 대한 파종 시기와 수확 시기 정의
cultivation_periods = {
    "봄재배": {"planting": [3, 4, 5], "harvesting": [6, 7]},
    "가을재배": {"planting": [7, 8], "harvesting": [10, 11]},
    "월동재배": {"planting": [9, 10], "harvesting": [3, 4, 5]},
    "여름재배": {"planting": [5, 6, 7], "harvesting": [9, 10]}
}
```

상관계수

	평균기온	평균풍속	평균강수량
봄재배	-0.381167	0.608670	0.430355
가을재배	-0.124116	-0.681694	-0.114611
월동재배	-0.536632	0.817298	0.901628
여름재배	0.978984	0.565597	-0.246479

탐색적 분석 (EDA)

탐색적 분석

- 상관관계 확인
- 특정 재배유형에서 물량에 따라 가격이 어떻게 변하는지 상관관계분석 진행
- 재배유형은 아까전과 같이 진행

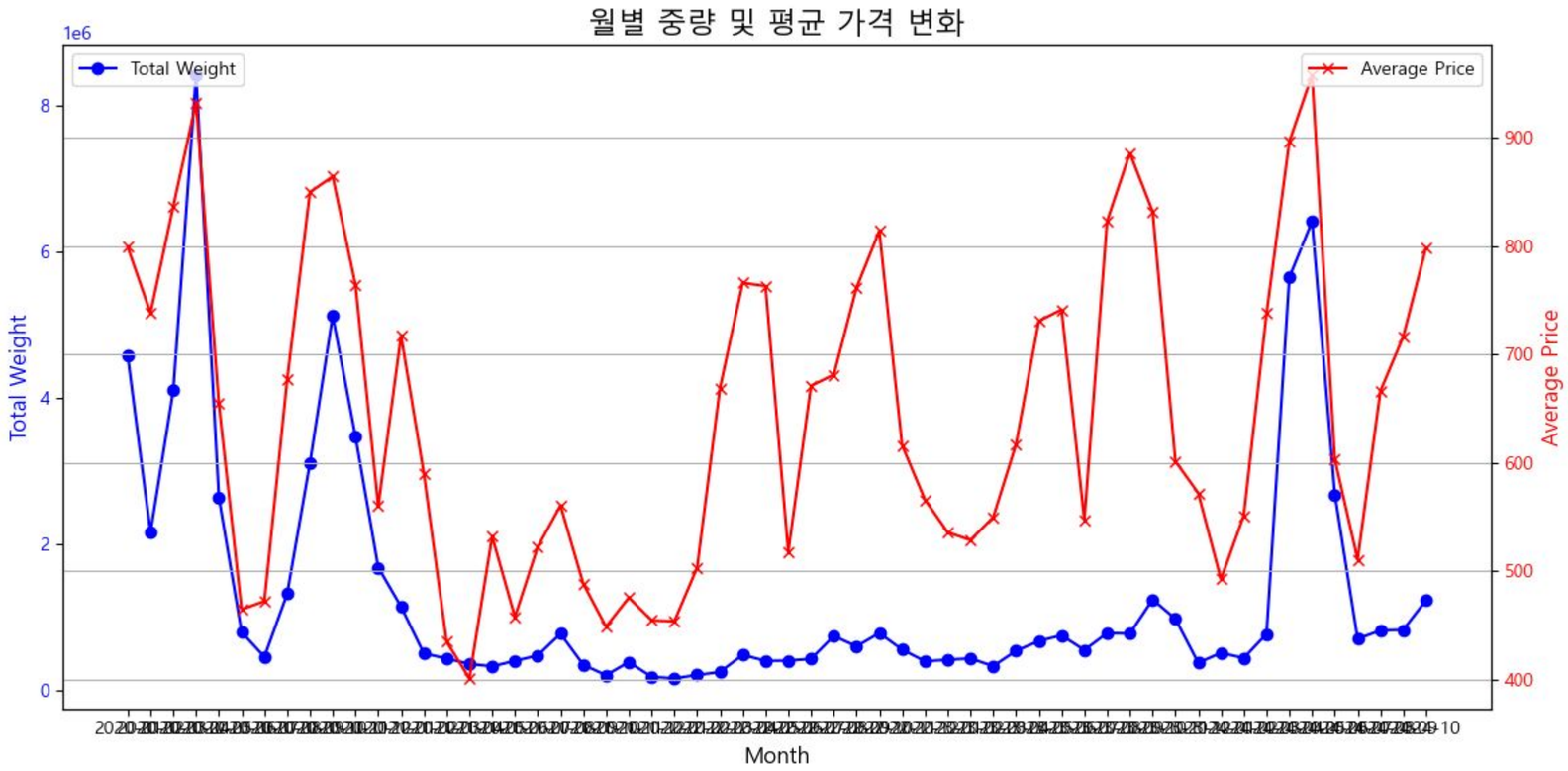
	1kg_평균가격
봄재배	0.530808
가을재배	-0.389053
월동재배	-0.614130
여름재배	-0.520921

탐색적 분석 (EDA)

탐색적 분석

- 양배추 수입 데이터 확인 및 피처로 사용할 수 있는지 고려 진행
- 양배추 중량 데이터 수집 후 월별 중량으로 그룹화 후 시각화 진행
- 확인결과 양배추 수입은 사후요인으로 보이기에 피처로는 적절하지 않다고 판단

(일반적으로 물량과 가격은 음의 상관관계를 보이는데 양의 상관관계를 보이는 것으로 보아 가격이 올라갔을 때 물량을 수입해오는 사후 요인으로 판단)



모델 적용

회귀 모델

- 회귀 모델 적용
- 데이터가 비선형 데이터이기에 Random Forest, Gradient Boosting 적용
- 재배 유형(파종기의 기후데이터를 입력피처로 수확기의 평균가격을 타겟변수로)이 중요한 피처가 될 것이라고 판단해 정의 후 피처로 추가

```
# 재배 유형 정의
cultivation_periods = {
    "봄재배": {"planting": [3, 4, 5, 6], "harvesting": [6, 7]},
    "가을재배": {"planting": [8, 9, 10], "harvesting": [10, 11]},
    "월동재배": {"planting": [10, 11], "harvesting": [3, 4, 5]},
    "여름재배": {"planting": [6, 7, 8], "harvesting": [9, 10]}
}
```

- **피처: 평균기온, 평균풍속, 평균강수량, 도매시장, 산지-광역시도, 재배유형**
- **타겟변수: 1kg_평균가격**

- **물량데이터**를 피처로 넣지 않았는데 수확기의 평균가격을 예측하는데 수확기의 물량데이터를 피처로 집어 넣는것을 실제 예측환경에서 적절하지 않은 방법이라고 생각해 제외하고 진행

모델 적용

회귀 모델

Random Forest 평가지표

```
mae : 135.57284800713052
mse : 31374.526204213493
rmse : 177.12855840946003
r2 : 0.34224769651679565

mape : 0.21188109898758659
```

Gradient Boosting 평가지표

```
mae : 134.71079403494997
mse : 29620.55361554012
rmse : 172.10622770701855
r2 : 0.3790189134887152

mape : 0.21053383111133292
```

모델 적용 결과 두 모델다 성능이 안나와서 왜그런지 원인 분석 진행

모델 적용

모델 적용

- 원인 분석
- 재배방식정의하여 피쳐로 집어넣었지만 타겟변수에 고려하지 못했었다

  -> 타겟변수를 해당 재배방식 **수확기의 1kg_평균가격**으로 설정

- 재배여부 피처로 집어 넣었을때 각각의 재배방식의 planting 월에 기후 데이터가 해당하면 True 아니면 False로 원-핫 인코딩 방식처럼 처리되게 설정

  ex) 봄 재배방식일 경우 3, 4, 5월은 True 나머지는 False

- 수정 후 다시 모델 적용

Random Forest 평가지표

```
{'MAE': np.float64(24.524279806505763),
 'RMSE': np.float64(62.72724846273508),
 'R-squared': 0.8731850042230339,
 'mape': np.float64(0.03529986111567536)}
```

Gradient Boosting 평가지표

```
{'MAE': np.float64(48.03863850544174),
 'RMSE': np.float64(74.61039505462037),
 'R-squared': 0.8205857989460483,
 'mape': np.float64(0.03529986111567536)}
```

모델 적용

회귀 모델

- 성능이 매우 뛰어나게 모델이 만들어져서 혹시 과정 중 문제가 있었는지 확인 진행

  ex) 데이터 유출, 데이터 중복

- 확인 결과 데이터 중복이 발생되서 회귀모델이 마치 분류처럼 진행되었을 가능성이 높다
- 중복 제거한 뒤 평가지표
- 확연하게 떨어지는 모습

```
{'MAE': np.float64(122.13201625779567),
 'RMSE': np.float64(162.86847006450952),
 'R-squared': 0.27681222313482157}
```

- 하지만 중복을 제거하는건 올바르지 않은 것 같다
- 예를들어 3월달에 특정도매시장에 특정생산지 물품이 여러번 들어오는 것은 당연한 결과
- 데이터를 월별로 수집하다보니 이런 문제가 발생한 것 같다

모델 적용

시계열 모델

- 다음은 시계열 모델을 진행
- 모델은 SARIMAX와 ETS 적용하였고 SARIMAX는 외생변수를 추가할 수 있어 기본모델과 외생변수로 도매시장, 산지-광역시도, 기후데이터, 재배방식을 추가하여 진행
- 계절주기는 12로 진행하였고
- SARIMAX(외생변수 포함)모델은 해당 타겟변수인 수확기에서 3개월 전까지의 기후데이터를 고려할 수 있게 매핑해줬다

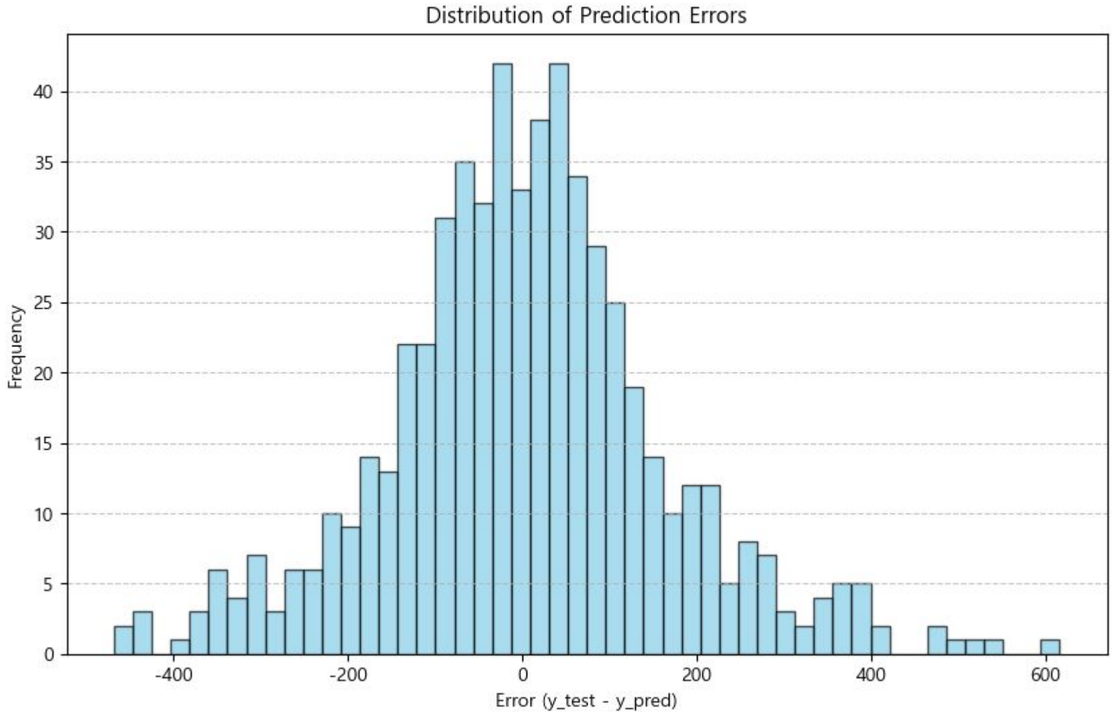
	SARIMAX	SARIMAX(외생변수 포함	ETS
MAE	129.29	138.6	139.2
MSE	174.19	31065	23233
MAPE	21.15	23.7	22.60

모델 적용

시계열 모델

- 시계열 모델을 진행였지만 성능이 그렇게 뛰어나지 않았다
- 또한 외생변수를 추가한 시계열모델이 오히려 성능이 떨어지는 모습을 보임
- 이상치 제거가 제대로 안되서 오차의 분포가 커서 모델이 제대로 적용이 안되는 것 같다

오차 분포 시각화



모델 적용

분류 모델

- 마지막으로 분류모델을 진행
- 이상치 제거가 완벽하게 되지 않고 회귀모델과 시계열모델의 성능이 좋게 나오지 않기 때문에 분류모델까지 진행
- 분류모델은 **Logistic Regression, Decision Tree, Random Forest** 총 3가지 모델 진행

- 피처는 기존과 동일하게 진행하였고 타겟변수는 도매시장, 산지-광역시도를 고려하여 평균금액을 비싸다, 싸다로 정의하였다

평가지표

Model	Accuracy	Precision (비싸다)	Recall (비싸다)	\
Logistic Regression	0.638548	0.620340	0.565534	
Decision Tree	0.707881	0.675513	0.710384	
Random Forest	0.707484	0.682072	0.689838	

F1-Score (비싸다)	Precision (싸다)	Recall (싸다)	F1-Score (싸다)
0.591670	0.651849	0.701513	0.675770
0.692510	0.738603	0.705722	0.721788
0.685933	0.729869	0.722702	0.726268

모델 적용

분류 모델

- 타겟변수에 도매시장, 산지-광역시도를 고려하였을때 이상치에 더 민감하게 반응을 해서 평가지표의 성능이 떨어지는 것으로 예상됨
- 타겟변수를 전체 데이터의 가격의 평균으로 정의하고 다시 모델 진행

평가지표

	Model	Accuracy	Precision (비싸다)	Recall (비싸다)	\
0	Logistic Regression	0.564477	0.550336	0.112175	
1	Decision Tree	0.752129	0.712410	0.742134	
2	Random Forest	0.754866	0.722524	0.728454	

	F1-Score (비싸다)	Precision (싸다)	Recall (싸다)	F1-Score (싸다)
0	0.186364	0.565886	0.926616	0.702658
1	0.726968	0.786402	0.760131	0.773044
2	0.725477	0.781147	0.776013	0.778571

모델 적용

분류 모델

- 최종적으로 성능이 가장 좋게나온 Random Forest를 최종모델로 설정하고 그리드 서치를 진행하여 최적의 하이퍼 파라미터 찾는 과정 진행

자체 평가 의견

자체 평가 의견

- 이번 프로젝트에서 랜덤포레스트(Random Forest)를 활용하여 머신러닝 모델을 구축했으나, 모델 성능 향상에 몇 가지 한계가 있었습니다. 특히, 데이터셋 내 이상치가 많아 성능 저하로 이어진 점이 아쉬웠습니다. 이를 해결하기 위해 보다 정교한 이상치 탐지 및 제거 기법을 적용하거나, 이상치의 영향을 최소화할 수 있는 데이터 전처리 방식을 고려할 필요가 있음을 느꼈습니다

- 하이퍼파라미터 최적화를 위해 그리드서치(Grid Search)를 시도했지만, 시간 제약으로 인해 최적의 설정을 찾는 과정을 충분히 완료하지 못했습니다. 추후 여유를 가지고 작업을 진행한다면, 하이퍼파라미터 튜닝 과정을 마무리하여 최적화된 모델을 적용하고 그 성능을 검증해 보고 싶습니다. 이를 통해 모델의 예측 성능을 더욱 개선할 수 있을 것으로 기대합니다

프로젝트 개요

프로젝트 요구 사항

- **데이터 처리**
  - 데이터 전처리: 결측값 처리, 이상치 탐지 및 제거, 데이터 정규화.
  - 데이터 통합: 날씨 데이터와 농산물 가격 데이터를 병합하여 분석 가능 형태로 변환.
- **머신러닝모델 개발**
  - 머신러닝: 랜덤 포레스트, XGBoost등 회귀 모델 활용.
- **농산물 가격 데이터**
  - 도매 및 소매 가격(일별, 월별 데이터).
  - 데이터 출처: KAMIS(농산물유통정보), 통계청, 농넷 등.

데이터 전처리 요구사항시간 범위:

  - 최근 4년(예: 2020년~현재) 데이터를 수집하여 시계열 분석 가능하도록 설정.
- **결측값 처리:**
  - 평균 대체, 선형 보간법 등 적합한 방법으로 결측값 처리.
- **범주형 데이터 인코딩:**
  - 지역이나 품종 등의 범주형 변수를 One-Hot Encoding 또는 Label Encoding으로 변환.
- **데이터 스케일링:**
  - 기상 데이터와 가격 데이터를 Min-Max Scaling 또는 Standard Scaling으로 정규화.

목표 성과 요구사항

- **변수 중요도 분석:**
  - 날씨 요인이 농산물 가격에 미치는 영향을 가시적으로 설명.
- **시계열 예측:**
  - 최소 1~3개월 단위로 가격 예측 성능 확보

프로젝트 개요

공공데이터 출처

농림축산식품부 (MAFRA)

- **데이터 내용:** 농산물 생산량, 유통량, 소비량, 수출입 동향, 가격 동향 등.
- **활용 방법:** 농림축산식품부의 공식 홈페이지에서 정기 보고서 및 통계 자료를 다운로드 가능
- **활용 방법:**
  - **KAMIS (농산물유통정보):** 도매가격, 소매가격, 산지정보 등을 제공.
  - **ATIS:** 농수산물 수출입 데이터를 상세히 조회 가능.
- **웹사이트:**
  - 농산물유통정보(KAMIS)
  - 한국농수산식품유통공사(ATIS)

농촌진흥청 (RDA)

- **데이터 내용:** 농작물 재배기술, 품종 정보, 농업 생산 통계, 기후와 작물 생산성 연구 자료.
- **활용 방법:** 농업과학도서관 및 데이터베이스를 통해 연구 보고서와 데이터 다운로드 가능.
- **웹사이트:** 농촌진흥청

통계청 (KOSTAT)

- **데이터 내용:** 농업 생산량, 재배 면적, 농업 인구, 지역별 농업 생산 통계.
- **활용 방법:** 통계청 국가통계포털(KOSIS)에서 데이터 조회 및 다운로드.

1.5 기상청

- **데이터 내용:** 농업 기상 데이터(강수량, 온도, 일조량, 풍속 등).
- **활용 방법:** 기상청 기상자료개방포털에서 농업과 관련된 날씨 데이터를 다운로드 가능.
- **웹사이트:** 기상자료개방포털

프로젝트 요약

프로젝트를 전체적으로 요약한 페이지입니다.

1. 생산성 향상 80% 예상

중요한 내용이나 길게 의견을 보여주어야 할 경우 해당 페이지를 활용해보세요. 프레젠테이션은 가독성이 생명입니다. 가독성의 핵심은 얼마나 내용을 간결하게 전달하는가 입니다. 쉽게 읽을 수 있고, 눈에 잘 띄는 것이 중요해요.

---

2. 연구 성과 향상

중요한 내용이나 길게 의견을 보여주어야 할 경우 해당 페이지를 활용해보세요. 프레젠테이션은 가독성이 생명입니다. 가독성의 핵심은 얼마나 내용을 간결하게 전달하는가 입니다. 쉽게 읽을 수 있고, 눈에 잘 띄는 것이 중요해요.
프레젠테이션의 전체적인 톤앤매너와맞는 폰트와 색상 등을 활용해주세요.

* 페이지 내 인물 사진은 샘플이미지 입니다.