# 회귀분석을 이용한 토마토 가격 예측

B조 안수정, 이종승

# Table of Contents





# #Part 1,

# Introduction

# 기획 의도 및 목표

## 기획 의도

- 토마토는 가격 변동이 큰 작물로, 지역별 날씨 데이터 & 다양한 변수를 활용하여  예측한다면 가격 안정화를 시키는 의미있는 연구가 될 것으로 예상
- 타국 토마토 수입량 Top 5를 선정해서 추가적인 변수를 고려하면 가격을 예측하는데 좀 더 정확한 결과를 얻을 것으로 예상하고 해당 프로젝트를 진행

목표 : 토마토 예측 가격과 실제 가격의 오차 분석

# 선행자료 조사

- 농촌진흥청 자료

> 온도 35도 넘으면, 토마토 열매량 4분의 1가량 줄어
>
> - 빛가림 막이나 도포제 활용…습도 60~80% 맞춰야 병 예방 -

- 한국농어민신문

**“토마토 수도권 출하 몰려 가격편차 심화...계획 생산·출하 필요”**

# 선행자료 조사

- 한경 경제

  일조량 부족에 출하량 줄어...토마토 가격 1년새 31% 올라

- 파이낸셜 뉴스

  농산물값 낮추고 유류 저가공급... 농협, 고물가 '고통분담'

# #Part 2,

# Main

# 데이터 수집 사이트




# 데이터 컬럼 설명

**A**

- date : 날짜(2016.01.01~2020.12.31)
- weekdays : 요일
- price : 토마토 상품 가격(上등급)(원/5kg)
- production : 토마토 생산량(톤)
- area : 토마토 생산면적(ha)
- CPI : 소비자물가지수
- P.Gas : 고급휘발유(원)
- R.Gas : 보통휘발유(원)
- diesel : 자동차용경유(원)
- avg.temp : 평균온도(℃)
- precipitation : 강수량(mm)
- wind.speed : 풍속(m/s)
- avg.humid : 평균 상대습도(%)
- t.sunhour : 합계 일조시간(hr)
- t.insolation : 합계 일사량(MJ/㎡)
- snow.depth : 일 최심적설(cm)
- avg.gtemp : 평균 지면온도(℃)
- CHN.volume : 중국 물량(톤)
- USA.volume : 미국 물량(톤)
- ITA.volume : 이탈리아 물량(톤)
- CHL.volume : 칠레 물량(톤)
- ESP.volume : 스페인 물량(톤)

**B**

| | |
|---|---|
| 일별 날씨 정보(8) | - 평균온도<br>- 강수량<br>- 풍속<br>- 평균 상대습도<br>- 합계 일조시간<br>- 합계 일사량<br>- 일 최심적설<br>- 평균 지면온도 |
| 일별 유류 가격(3) | - 고급휘발유<br>- 보통 휘발유<br>- 자동차용경유 |
| 월별 수입량(5) | - 중국 물량(톤)<br>- 미국 물량(톤)<br>- 이탈리아 물량(톤)<br>- 칠레 물량(톤)<br>- 스페인 물량(톤) |
| 연도별 기타 정보(3) | - 토마토 생산량(톤)<br>- 토마토 생산면적<br>- 소비자 물가지수 |

# 데이터 전처리 - 결측치에 대한 고민

1. 주말 및 공휴일

- 거래가 발생하지 않는 주말 및  공휴일의 데이터를 모두 제거하는 것이 모델의 성능 저하를 유발할까?
- 거래가 발생하지 않는 날 중 주말 데이터만 제거하고 공휴일의 가격 데이터는 0원으로 처리했을 때 모델 성능의 변화를 알 수 있을까?

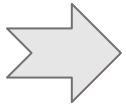
2. 평균온도가 없는 날 존재

- 하루 전/후 평균값으로 대체

# 데이터 전처리

- One-Hot Encoding을 이용하여 요일정보 숫자로 변환

1 weekdays 1227 non-null object



```
df = pd.get_dummies(df)
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1227 entries, 0 to 1226
Data columns (total 29 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   date            1227 non-null   datetime64[ns]
 1   price           1227 non-null   int64
 2   production      1227 non-null   float64
 3   area            1227 non-null   float64
 4   CPI             1227 non-null   float64
 5   P.Gas           1227 non-null   float64
 6   R.Gas           1227 non-null   float64
 7   diesel          1227 non-null   float64
 8   avg.temp        1227 non-null   float64
 9   precipitation   1227 non-null   float64
 10  wind.speed      1227 non-null   float64
 11  avg.humid       1227 non-null   float64
 12  t.sunhour       1227 non-null   float64
 13  t.insolation    1227 non-null   float64
 14  snow.depth      1227 non-null   float64
 15  avg.gtemp       1227 non-null   float64
 16  CHN.volume      1227 non-null   float64
 17  USA.volume      1227 non-null   float64
 18  ITA.volume      1227 non-null   float64
 19  CHL.volume      1227 non-null   float64
 20  ESP.volume      1227 non-null   float64
 21  year            1227 non-null   int64
 22  month           1227 non-null   int64
 23  day             1227 non-null   int64
 24  weekdays_금      1227 non-null   uint8
 25  weekdays_목      1227 non-null   uint8
 26  weekdays_수      1227 non-null   uint8
 27  weekdays_월      1227 non-null   uint8
 28  weekdays_화      1227 non-null   uint8
```

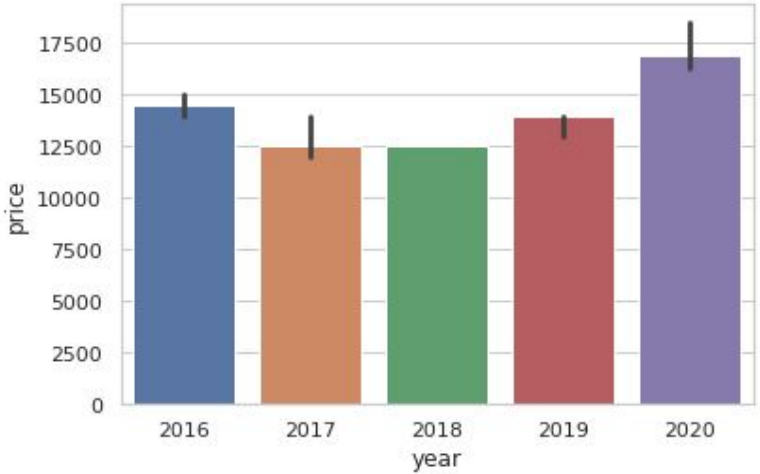
# 데이터 전처리

- 날짜를 year / month / day로 구분하여 데이터 처리

```
df['year'] = df['date'].dt.year
df['month'] = df['date'].dt.month
df['day'] = df['date'].dt.day
df.head()
```

| | date | weekdays | price | production | area | CPI | P.Gas | R.Gas | diesel | avg.temp | ... | snow.depth | avg.gtemp | CHN.volume | USA.volume | ITA.volume | CHL.volume | ESP.volume | year | month | day |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 2016-01-04 | 월 | 13000 | 8604.294118 | 375.941176 | 95.232 | 1884.70 | 1489.22 | 1274.66 | 2.0 | ... | 0.0 | 3.0 | 1613.0 | 1511.07 | 673.0 | 77.0 | 130.0 | 2016 | 1 | 4 |
| 1 | 2016-01-05 | 화 | 13000 | 8604.294118 | 375.941176 | 95.232 | 1885.33 | 1488.01 | 1273.53 | -2.7 | ... | 0.0 | 0.1 | 1613.0 | 1511.07 | 673.0 | 77.0 | 130.0 | 2016 | 1 | 5 |
| 2 | 2016-01-06 | 수 | 14000 | 8604.294118 | 375.941176 | 95.232 | 1883.38 | 1486.25 | 1271.49 | -1.7 | ... | 0.0 | -0.5 | 1613.0 | 1511.07 | 673.0 | 77.0 | 130.0 | 2016 | 1 | 6 |
| 3 | 2016-01-07 | 목 | 12500 | 8604.294118 | 375.941176 | 95.232 | 1880.12 | 1485.34 | 1269.80 | -3.4 | ... | 0.0 | -1.0 | 1613.0 | 1511.07 | 673.0 | 77.0 | 130.0 | 2016 | 1 | 7 |
| 4 | 2016-01-08 | 금 | 12000 | 8604.294118 | 375.941176 | 95.232 | 1878.63 | 1484.19 | 1268.43 | -3.3 | ... | 0.0 | -1.5 | 1613.0 | 1511.07 | 673.0 | 77.0 | 130.0 | 2016 | 1 | 8 |

# EDA - 연도별 토마토 가격 추이 (단위: 원/상품/5kg)

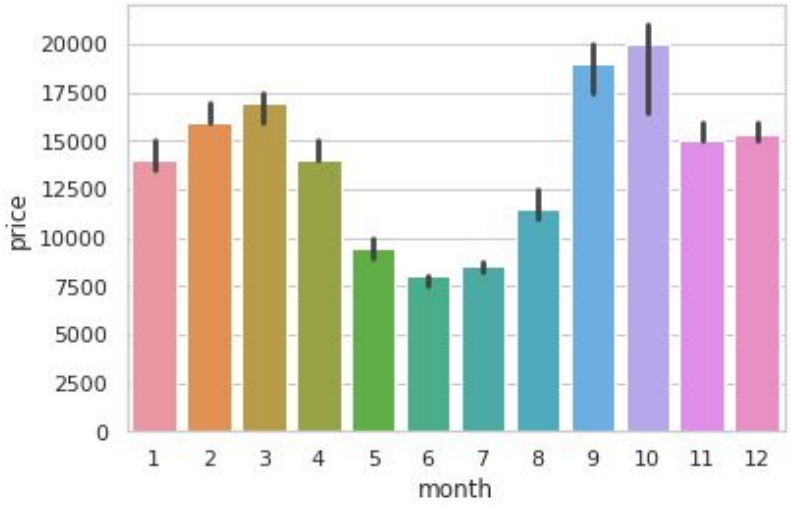



**2020년도 여름은 사상 최장 장마와 강력한 태풍으로 토마토 가격 급등**

역대 최장 장마 기록한 2020...평년보다 20일 더 내린 장맛비

밥상 덮친 이상기후... 햄버거에 '토마토'도 사라졌다

# EDA - 월별 토마토 가격 추이 (단위: 원/상품/5kg)




**가격 상승 원인 :**
**토마토는 8월 초순이면 수확이 모두 끝나서 8월부터는 토마토의 품귀상태가 나타남**

# EDA - 컬럼 제거 전 / 후 모델 설명

Model 1(원본 데이터) - 컬럼을 따로 제거 하지 않고 진행

Model 2(평균온도 제거 Group) - 평균온도, 합계 일조시간, 합계 일사량, 평균 지면온도

Model 3(유류가격 제거 Group) - 고급휘발유, 보통휘발유, 자동차용경유

Model 4(일조량 제거 Group) - 강수량, 풍속, 평균 상대습도, 일 최심적설

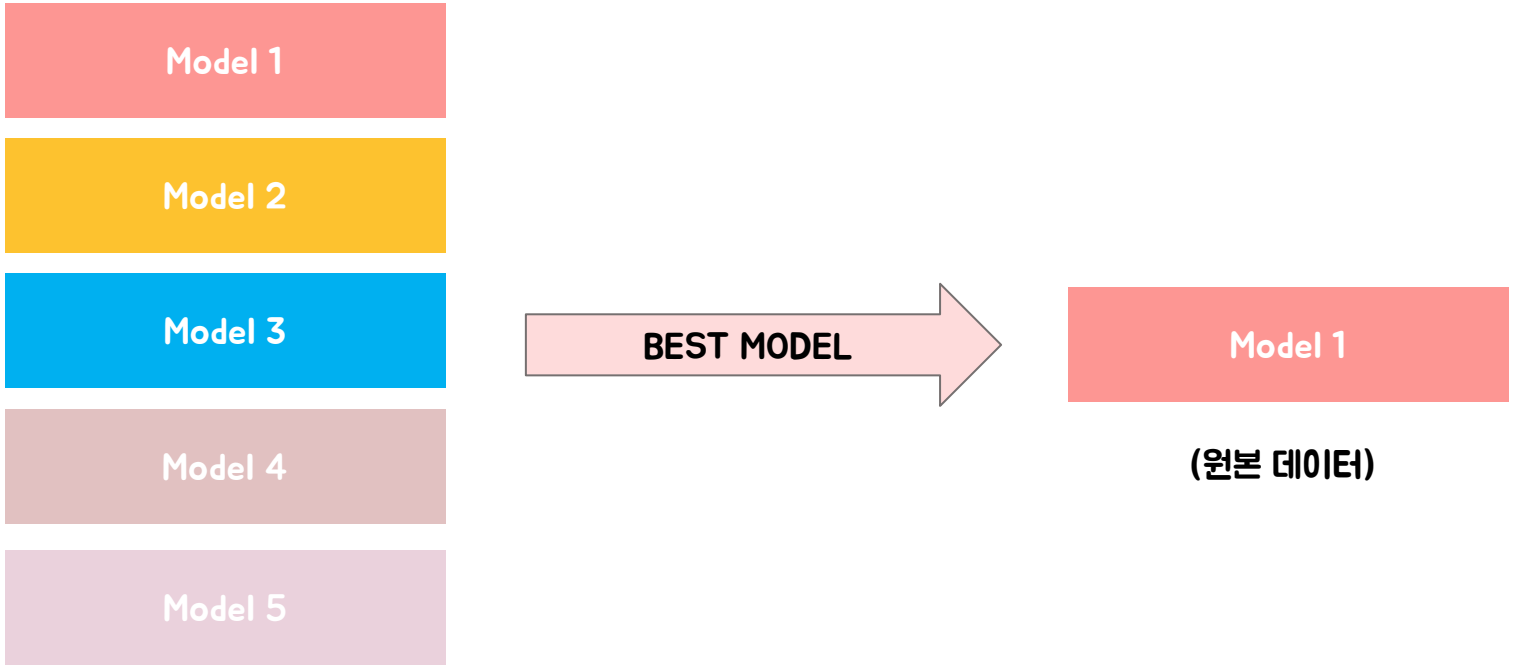
Model 5(타국 수입량 제거 Group) - 중국, 미국, 이탈리아, 칠레, 스페인 물량

# EDA - 컬럼 제거 전 / 후 모델 성능비교

| | Model 1 | Model 2 | Model 3 | Model 4 | Model 5 |
|---|---|---|---|---|---|
| Train_set | 0.516 | 0.374 | 0.454 | 0.508 | 0.442 |
| Test_set | 0.526 | 0.411 | 0.458 | 0.518 | 0.458 |

(평균온도 제거 Group)

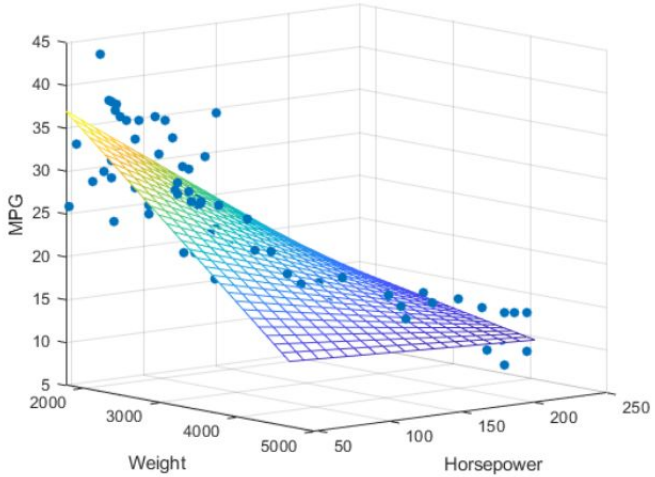
# MODEL 선정 과정



# 분석기법

## 다중회귀분석(Multiple Regression Analysis)

- 두 개 이상의 독립변수가 하나의 종속변수에 미치는 영향을 검증하는 분석 방법
- 단순회귀분석의 개념과 분석방법이 동일, 독립변수의 수에서 차이



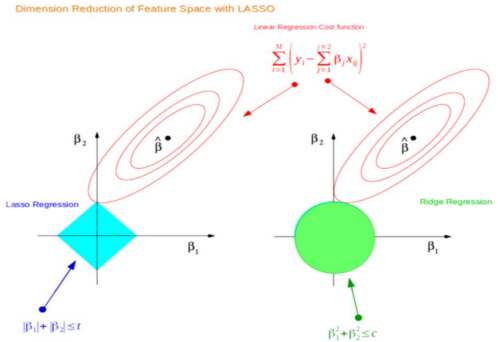
# 분석기법

## 라쏘회귀분석(Lasso Regression Analysis)

- 모델의 설명력에 기여하지 못하는 독립변수의 회귀계수를 0으로 만든다
- L1 페널티항으로 회귀모델에 페널티를 부과함으로써 회귀계수를 축소한다.

## 릿지회귀분석(Ridge Regression Analysis)

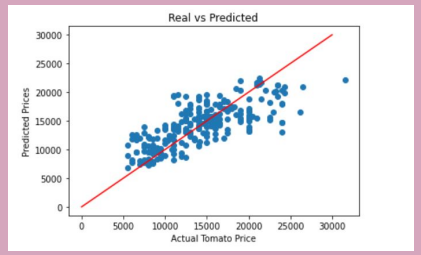
- 모델의 설명력에 기여하지 못하는 독립변수의 회귀계수 크기를 0에 근접하도록 축소한다.
- L2 페널티항으로 회귀모델에 페널티를 부과함으로써 회귀계수를 축소한다.



#Part 3,

# Conclusion

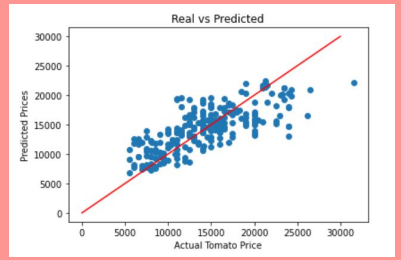
# 결과 시각화 및 해석 - Model 1 실제 / 예측 값




다중회귀

```
pred_test.mean(), y_test.mean()
```

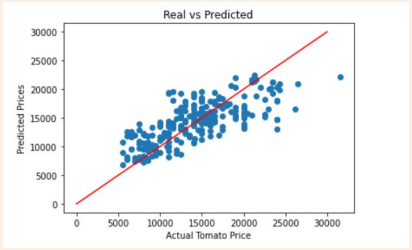
```
(14385.162624555762, 14408.739837398374)
```




Ridge

```
pred_test.mean(), y_test.mean()
```

```
(14385.162624555762, 14336.970684039088)
```




Lasso

```
pred_test.mean(), y_test.mean()
```

```
(14384.475547858548, 14177.035830618892)
```

# 결과 시각화 및 해석 - 모델 성능 비교(주말, 공휴일 제거)

| | | Linear | Ridge | Lasso |
|---|---|---|---|---|
| Score | train | 0.51647 | 0.51641 | 0.54268 |
| | test | 0.52582 | 0.52581 | 0.44046 |
| RMSE | train | 3436.18069 | 3436.35665 | 3436.19509 |
| | test | 3428.80944 | 3429.53345 | 3428.01270 |
| MAE | train | 2698.85325 | 2699.58882 | 2699.01868 |
| | test | 2676.26547 | 2675.24144 | 2675.54444 |

# 분석결과 요약

- 토마토 가격은 온도와 관련된 날씨의 영향을 가장 많이 받음
- 주말 및 공휴일과 같은 결측치 데이터를 포함시키지 않을때 성능이 더 좋음
- 총 3가지 모델을 진행해 본 결과, 모델 간의 성능은 큰 차이가 없었고 각 모델별 예측치와 실제값의 차이는 아래와 같이 확인할 수 있음

| 모델 | 가격 차이 |
|---|---|
| 다중회귀 | 23.6원 |
| Ridge | 48.2원 |
| Lasso | 207.4원 |

# 추가 진행 예정 사항 및 한계점 & 보완 방법

- 현재 진행상황 이후 추가적으로 딥러닝(Ex. RNN, LSTM)을 사용하여 가격 정밀 분석 예정
- 토마토 주산지 데이터를 포함시키지 않아 데이터를 예측하는데 어려움이 있었음
- 데이터 개수의 한계로 각 분석기법마다의 차이가 크지 않아 정확한 모델 선정이 어려움
- 결과에만 압도되지 않도록 목적에 맞는 다양한 가설을 설정하는 연습을 계속 해야됨을 느꼈음

## 참고자료


- 이도영, 양예원, 이주형, 박지홍, 강민구. (2020). Lasso 회귀분석을 활용한 농산물 가격예측 모델 변수 선정 연구
- 신성호, 이미경, 송사광. (2018). LTSM 네트워크를 활용한 농산물 가격 예측 모델. 한국콘텐츠학회 논문지, 18(11), 416-429.
- 김주현, '역대 최장 장마 기록한 2020… 평년보다 20일 더 내린 장맛비', 머니투데이, 2020.08.21, https://news.mt.co.kr/mtview.php?no=2020082110461160385
- 고성진, '토마토 수도권 출하 몰려 가격편차 심화…계획 생산·출하 필요', 한국농어민신문, 2021.11.12, http://www.agrinet.co.kr/news/articleView.html?idxno=304998
- 김현철, '농산물값 낮추고 유류 저가공급… 농협, 고물가 '고통분담'', 파이낸셜뉴스, 2022.06.07, https://www.fnnews.com/news/202206071809264401
- 김유연, '온도 35도 넘으면 토마토 열매량 4분의 1가량 줄어', 월간환경, 2022.06.14, http://www.ecocody.co.kr/news/articleView.html?idxno=2856

# 감사합니다