

농산물유통종합정보시스템

가격예측 AI 모델

DACON 2021 농산물 가격예측 AI 경진대회



데이터 전처리, EDA, LSTM 모델

조동 *

LSTM 모델 결과 csv 출력, PPT 제작

최휘 *

인사이트 도출, GRU 모델, Django 제작

신은 *

강영 *

팀장

데이터 전처리, 프로젝트 총괄

김성 *

데이터 정제, EDA, RNN 연구

나혜 *

GRU 모델, Django 틀 구현, PPT 제작



TABLE OF CONTENTS

#1 / About Project

2021 농산물 가격 예측 AI 경진대회

#2 / EDA

데이터 전처리
탐색적 데이터 분석

#3 / Machine Learnig

모델 생성 및 예측
예측 방법 개선

#4 / Conclusion

프로젝트 마무리, 결론



ABOUT PROJECT

2021

농산물 가격예측

AI 경진대회

#목적

기존 농산물유통 종합정보시스템의 농산물 가격
예측 모형을 개선할 수 있는 새로운 아이디어와
알고리즘을 탐색

#문제접근

2016.01.01 ~ 2020.09.28 기간 내 21가지 농산물
가격, 거래량 데이터로 가격 예측 모델을 생성
2020.09.29 ~ 2020.11.05 의 데이터로
2020년 10월 6일(화) ~ 2020년 11월 12일(목)
기간 내 품목별 1주, 2주, 4주 후 가격을 예측

#평가요소

2021년 10월 5일(화) ~ 2021년 11월 11일(목)
기간 내 품목별 1주, 2주, 4주 후 가격을 예측





EDA

탐색적 데이터 분석

☑ 데이터 전처리

☑ 상관관계 분석

☑ 다양한 기준별 데이터 분석

데이터 설명

products.csv

products.csv

2016.01.01 ~ 2020.11.05 농산물 거래량/가격 데이터

EDA는 train.csv와 test_files를 합쳐 products.csv로 진행,
모델 생성은 train.csv만 사용하고 test_files는 leaderboard의
성능 평가 용으로 사용

배추, 무, 양파, 건고추, 마늘, 대파, 얼갈이배추, 양배추, 깻잎, 시
금치, 미나리, 당근, 파프리카, 새송이, 팽이버섯, 토마토, 청상추,
백다다기, 애호박, 캠벨얼리, 샤인마스캇 - 총 21개 농산물의
거래량과 가격 데이터

#	칼럼명	dtype
0	date	object
1	요일	object
2	배추_거래량(kg)	float64
3	배추_가격(원/kg)	float64
4	무_거래량(kg)	float64
5	무_가격(원/kg)	float64
6	양파_거래량(kg)	float64
...		
40	캠벨얼리_거래량(kg)	float64
41	캠벨얼리_가격(원/kg)	float64
42	샤인마스캇_거래량(kg)	float64
43	샤인마스캇_가격(원/kg)	float64

데이터 전처리



중복데이터 및
결측값 처리



이상치
탐색 / 제거



3차원 시계열
배열 생성

☑ RNN 모델은 3차원 형식
input type 필요

RNN 모델 사용을 위한 3차원 시계열 배열 생성

```
[[[0. 0. 0. ... 0. 0. 0. ]  
 [0.16666667 0.01551951 0.21788079 ... 0.26239333 0. 0. ]  
 [0.33333333 0. 0. ... 0. 0. 0. ]  
 ...  
 [0.66666667 0.14522328 0.41258278 ... 0.44492215 0. 0. ]  
 [0.83333333 0.18698317 0.42450331 ... 0.34838121 0. 0. ]  
 [1. 0.19904495 0.43112583 ... 0.10982998 0. 0. ]]  
 ...  
 [[0.16666667 0.01551951 0.21788079 ... 0.26239333 0. 0. ]  
 [0.33333333 0. 0. ... 0. 0. 0. ]  
 [0.5 0.27306791 0.31655629 ... 0.50615595 0. 0. ]  
 ...  
 [0.83333333 0.18698317 0.42450331 ... 0.34838121 0. 0. ]  
 [1. 0.19904495 0.43112583 ... 0.10982998 0. 0. ]  
 [0. 0.22318876 0.41390728 ... 0.43892906 0. 0. ]]  
 ...  
 [[0.33333333 0. 0. ... 0. 0. 0. ]  
 [0.5 0.27306791 0.31655629 ... 0.50615595 0. 0. ]  
 [0.66666667 0.22402933 0.29271523 ... 0.37170217 0. 0. ]  
 ...  
 [1. 0.19904495 0.43112583 ... 0.10982998 0. 0. ]  
 [0. 0.22318876 0.41390728 ... 0.43892906 0. 0. ]  
 [0.16666667 0.14334917 0.48543046 ... 0.43202397 0. 0. ]]
```

28일 (window size)

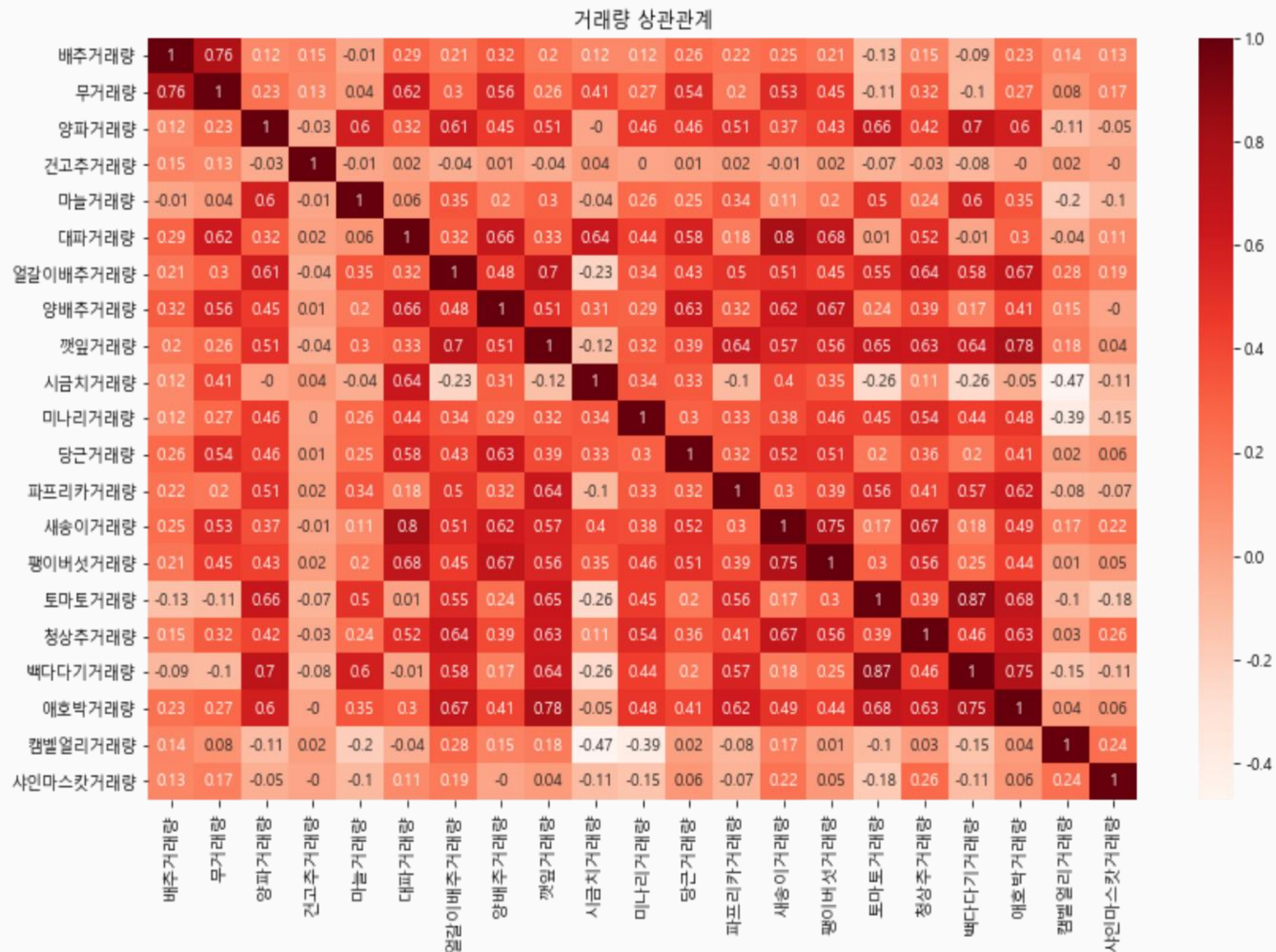
(1677, 28, 43)

총 데이터 개수

열 개수 (input size)
(Lable은 21)

상관관계 분석

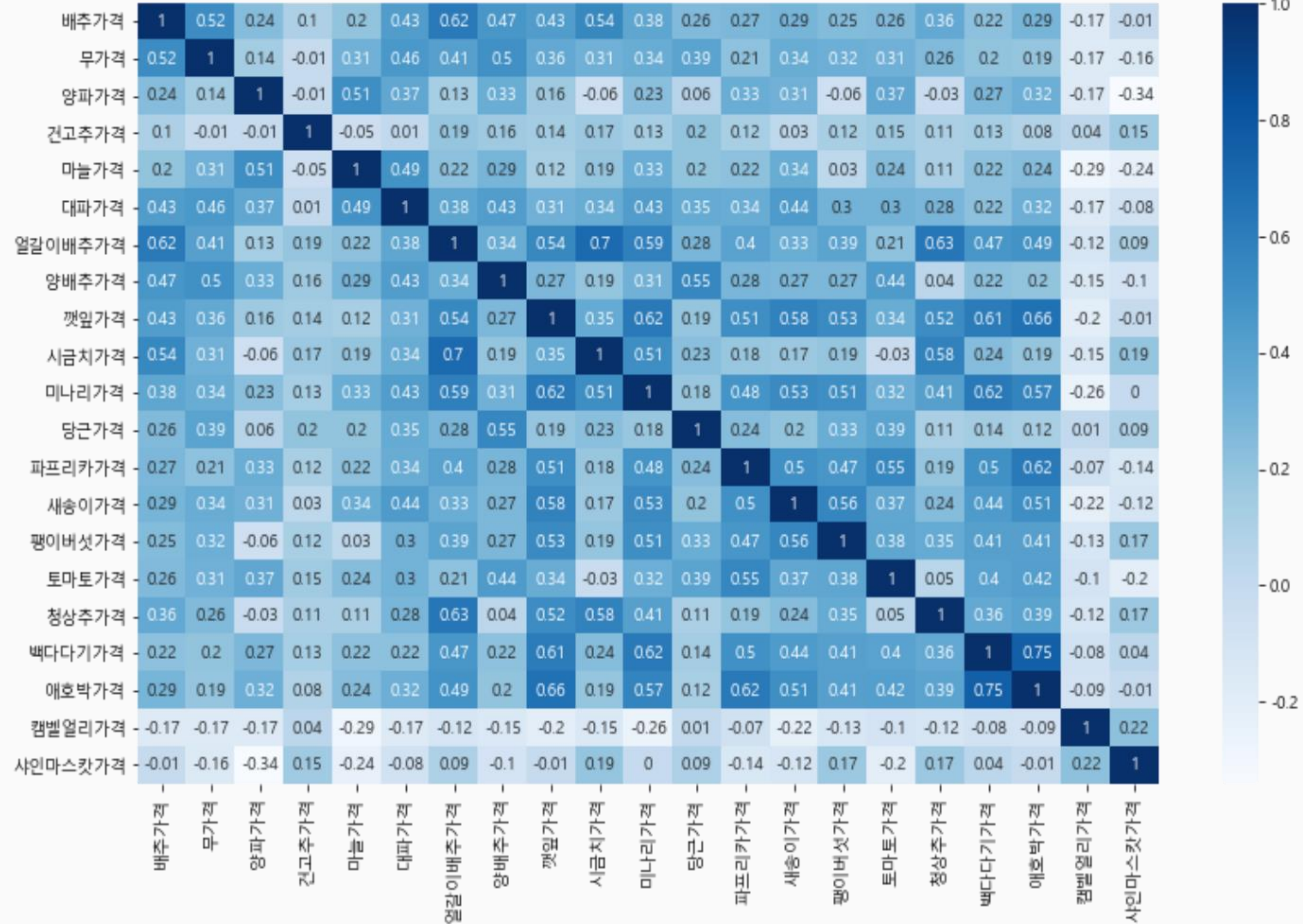
거래량 상관관계



상관관계 분석

가격 상관관계

가격 상관관계



거래량 상관관계

무와 배추거래량의 상관관계는 0.76로 강한 양의 상관관계를 보인다.
애호박은 깻잎과의 상관관계가 0.78로 가장 강한 양의 상관관계를 보인다.
토마토와 백다다기의 상관관계는 0.87로 가장 강한 양의 상관관계를 보인다.
반면에 시금치와 캠벨얼리는 -0.47로 음의 상관관계를 나타내고 있다.
전체적으로 상품 간 상관관계가 높은 경우가 많고,
특히 특정 상품과 양의 상관관계를 보이는 상품이 다수 있다.

가격 상관관계

애호박은 백다다기와 깻잎과 각각 0.75, 0.66의 강한 양의 상관관계를 나타내고 있다.
얼갈이배추는 시금치, 배추와 각각 0.7, 0.62로 강한 양의 상관관계를 나타내고 있다.
양파와 샤인마스캇의 상관관계는 -0.34로 음의 상관관계를 보이고 있고 다른 농산물들도
캠벨얼리와 샤인마스캇과는 높지 않은 상관관계를 보인다.
비교적 거래량의 상관관계보다 낮은 수치를 나타내고 있으며,
특정 상품과의 상관관계가 높은 경우가 많다.



샤인마스캇의 거래량과 가격은 독립적인 성향을 보인다.

데이터를 통해 샤인마스캇은 특정 시즌에만 거래된다는 것을 확인 할 수 있었다.

상관관계분석 REPORT



[월 기준 평균 거래량과 평균 가격 분석]

다양한 인사이트를 얻기 위해
여러 기준을 세워 분석해보기

다양한 기준별 데이터 분석

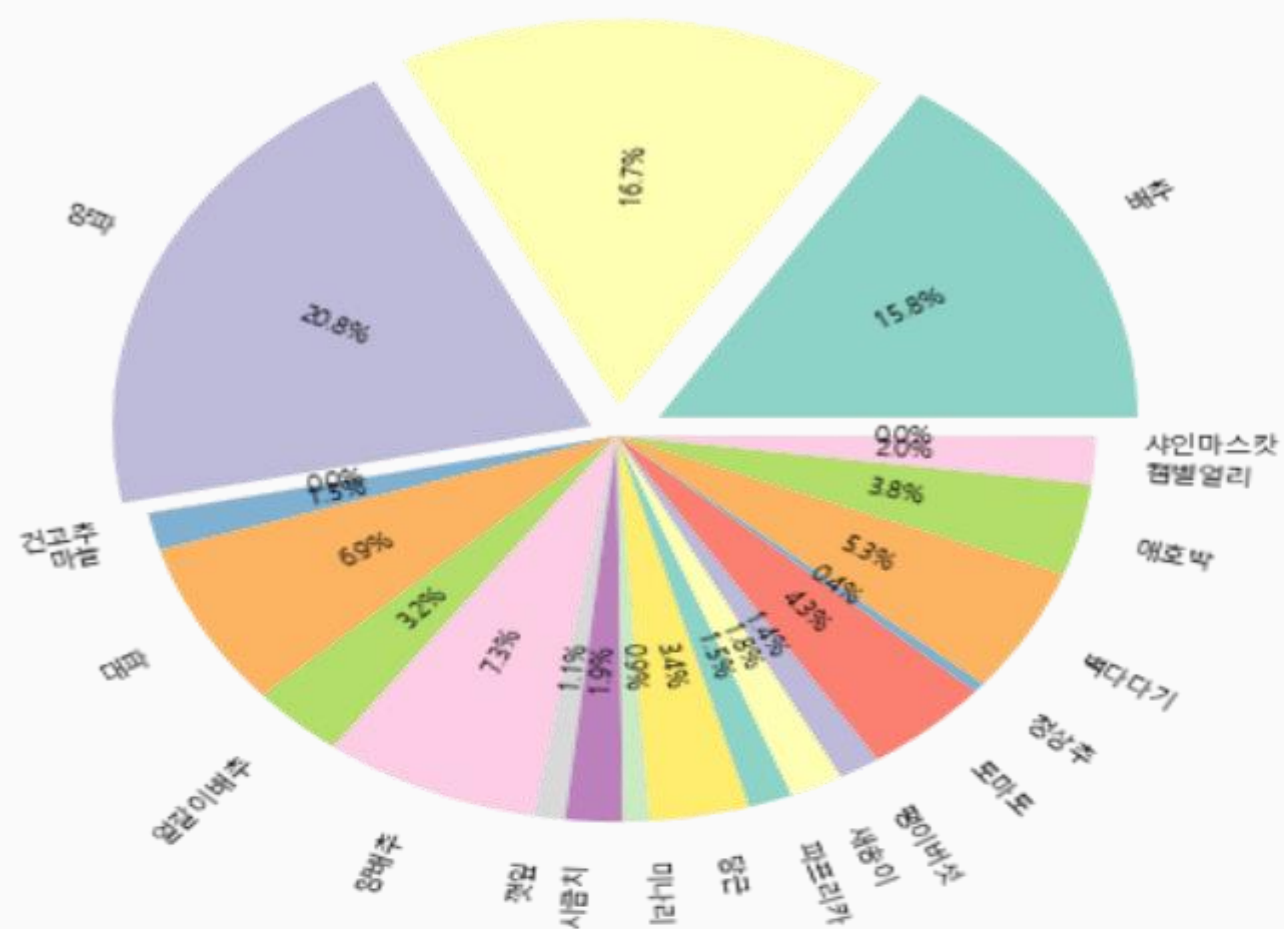
[연별 농산물 총거래량 분석]

[거래량이 높은 배추, 무,
양파의 요일별 거래량]

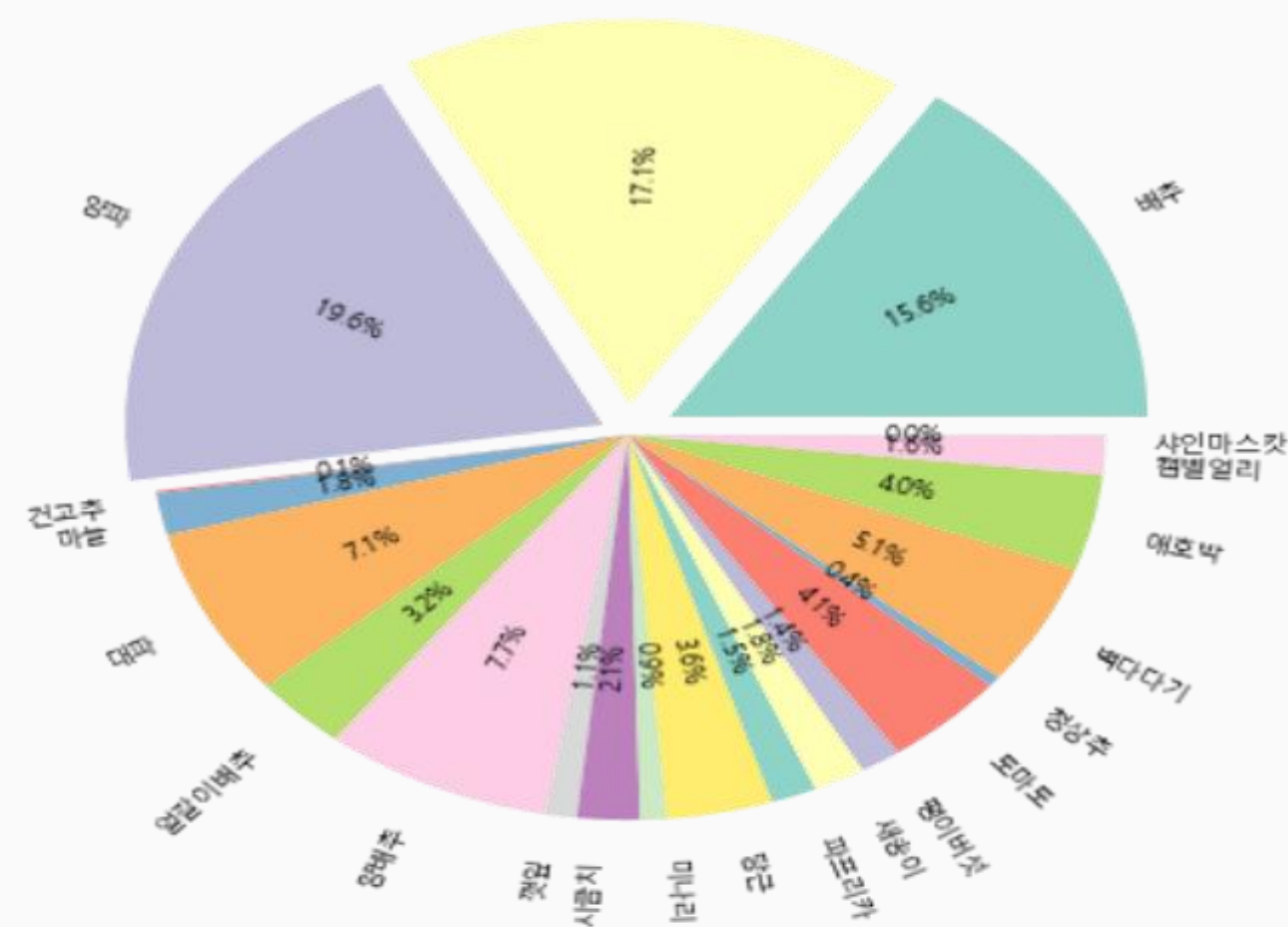
기준별 데이터 분석

연별 농산물
총거래량 분석

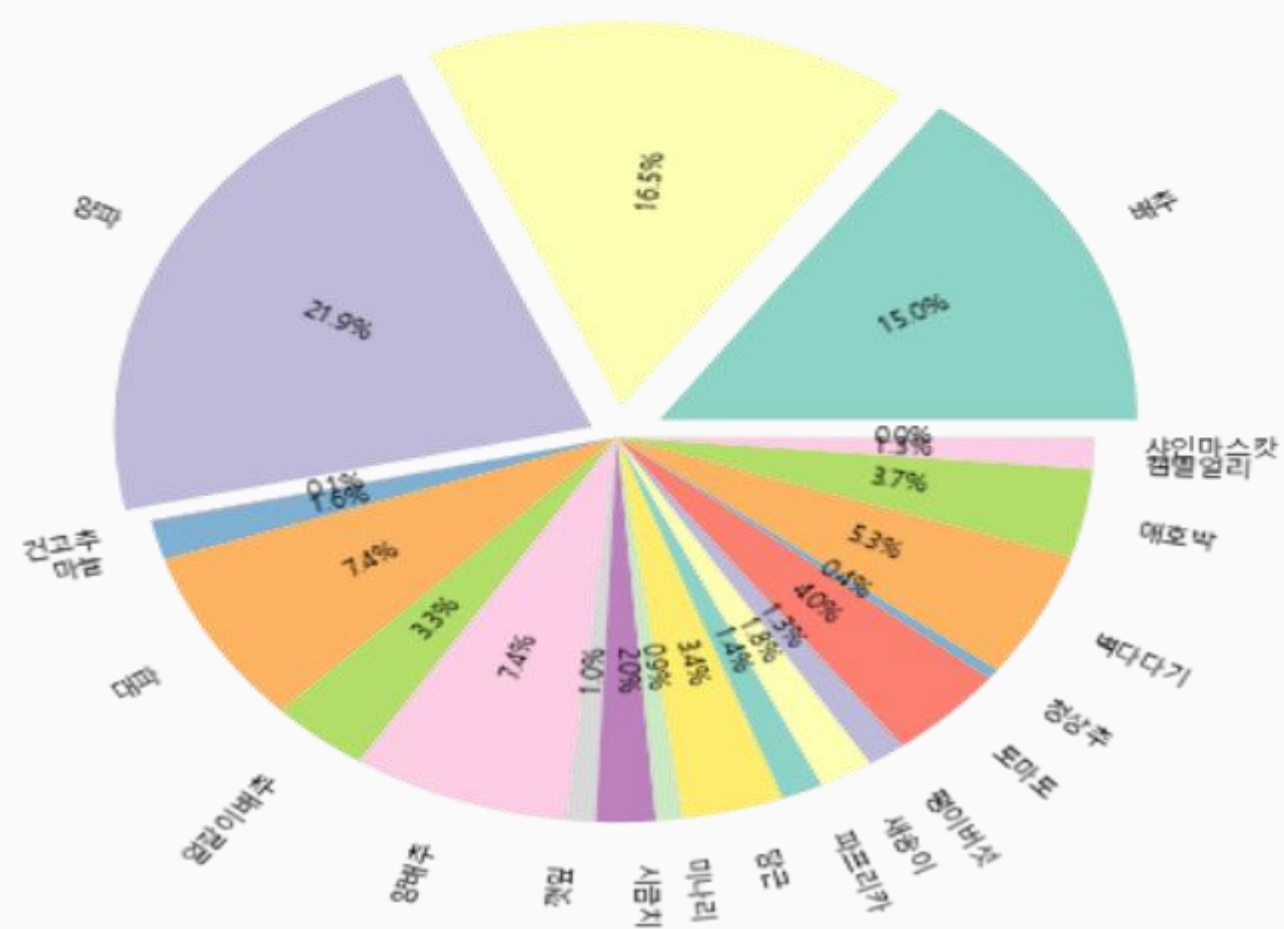
2016년 농산물 총거래량
마



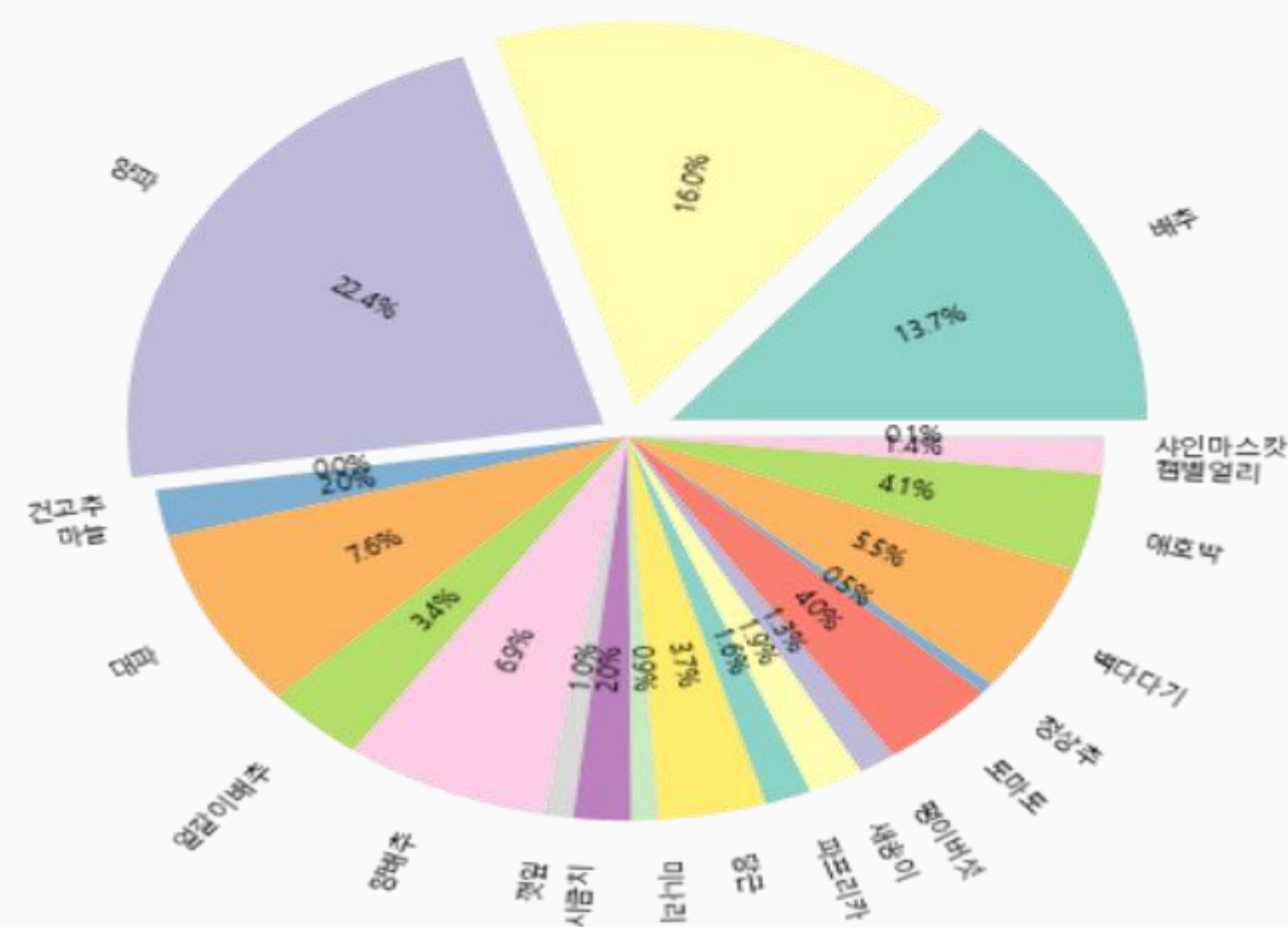
2017년 농산물 총거래량
마



2018년 농산물 총거래량
마

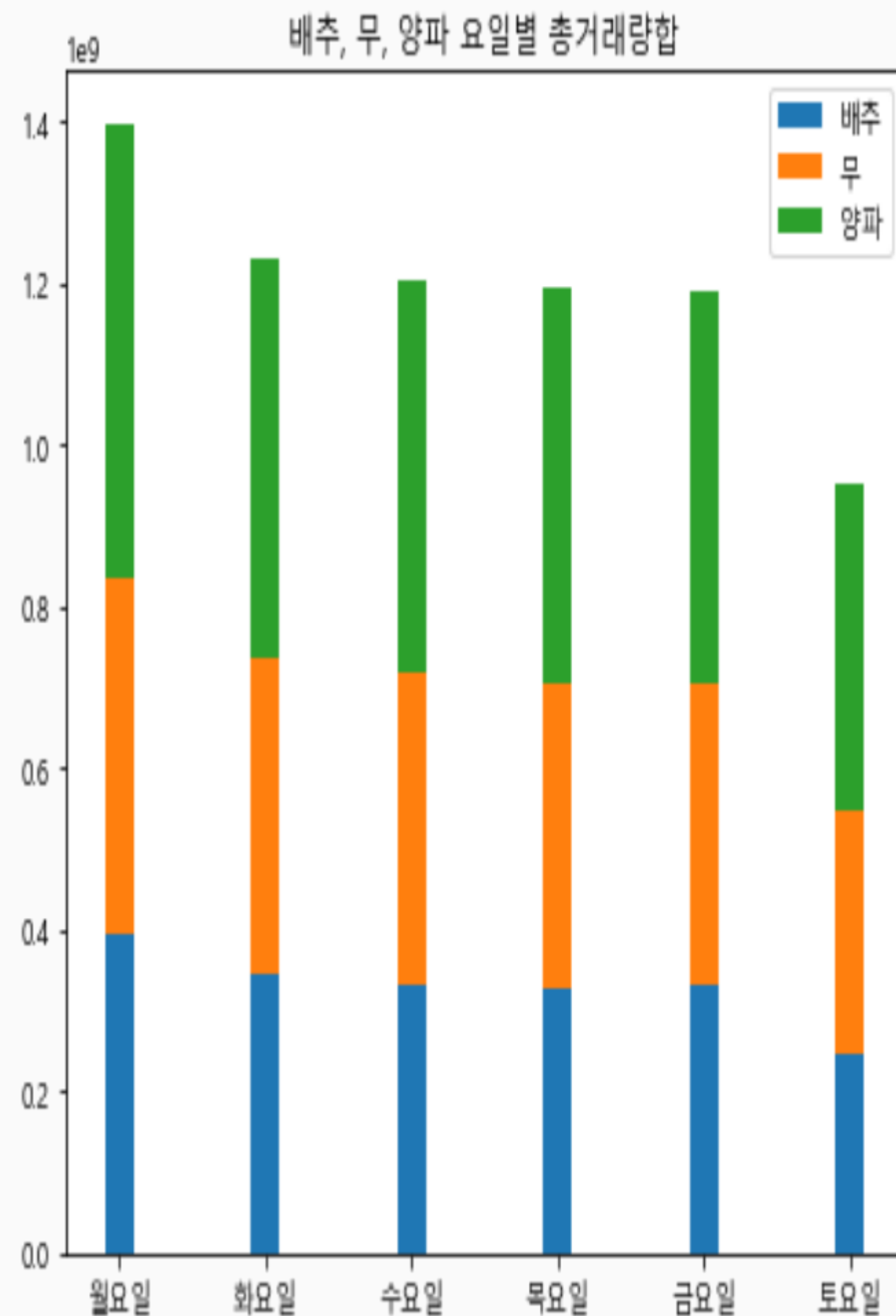
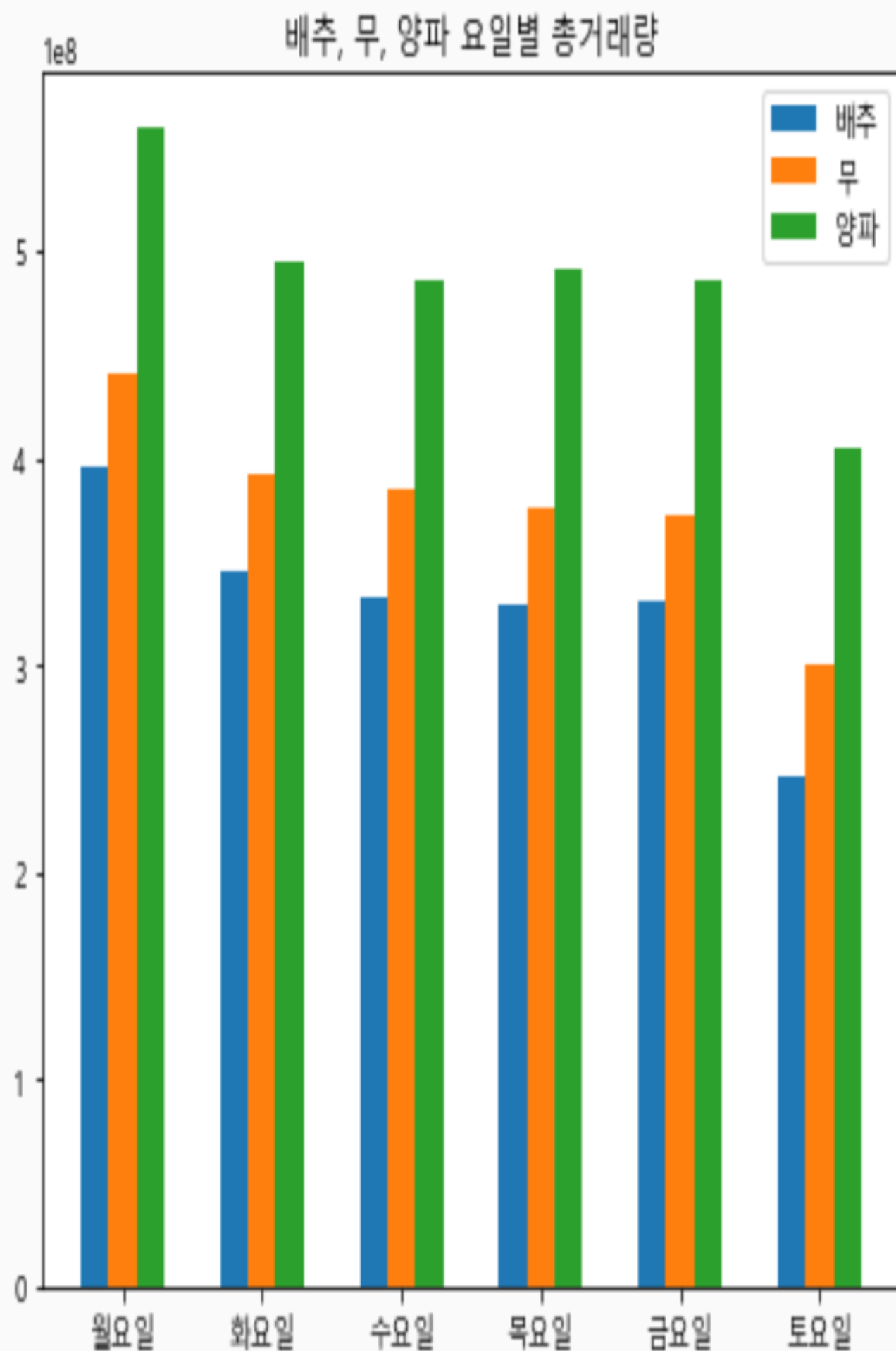


2019년 농산물 총거래량
마



기준별 데이터 분석

거래량이 높은
배추,무,양파의
요일별 거래량



월 기준
평균 거래량과
평균 가격 분석

[illegible]

The chart displays the monthly price trends for 15 types of Jeogori. The Y-axis represents the price in Korean Won (₩), ranging from 0 to 15,000. The X-axis represents the months from 1 to 12. The legend identifies the following items:

- 배추가격 (Cabbage Jeogori)
- 무가격 (Radish Jeogori)
- 양파가격 (Onion Jeogori)
- 건고추가격 (Dried Chili Jeogori)
- 마늘가격 (Garlic Jeogori)
- 대파가격 (Green Onion Jeogori)
- 얼갈이배추가격 (Pickled Cabbage Jeogori)
- 양배추가격 (Broccoli Jeogori)
- 갯잎가격 (Seaweed Jeogori)
- 시금치가격 (Spinach Jeogori)
- 미나리가격 (Watercress Jeogori)
- 당근가격 (Carrot Jeogori)
- 파프리카가격 (Bell Pepper Jeogori)
- 새송이가격 (Mushroom Jeogori)
- 팽이버섯가격 (Enoki Mushroom Jeogori)
- 토마토가격 (Tomato Jeogori)
- 청상추가격 (Green Chili Jeogori)
- 백다다기가격 (White Radish Jeogori)

Key observations from the chart include:

- 배추가격 (Cabbage Jeogori):** Shows a significant peak in June, reaching approximately 18,000 ₩.
- 건고추가격 (Dried Chili Jeogori):** Shows a peak in August, reaching approximately 15,000 ₩.
- 얼갈이배추가격 (Pickled Cabbage Jeogori):** Shows a steady increase from January to June, peaking at approximately 10,000 ₩.
- 시금치가격 (Spinach Jeogori):** Shows a peak in August, reaching approximately 7,000 ₩.
- 갯잎가격 (Seaweed Jeogori):** Shows a peak in September, reaching approximately 7,500 ₩.

연별 농산물 총거래량 분석

대체로 매년 일정한 거래량을 유지하고 있지만 높은 비중을 차지하는 배추는 매년 조금씩 감소하고 있음을 알 수 있다. 또한 일정한 수준으로 양파, 무, 배추의 거래량이 가장 많고, 건고추와 샤인마스캇의 거래량은 매우 낮은 것을 알 수 있다.

거래량이 높은 배추, 무, 양파의 요일별 거래량

양파, 무, 배추 순으로 거래량이 많다.

월요일에 거래시장이 제일 활발하고 날이 지날수록 점점 떨어져서 토요일에는 거래량이 제일 작다.

기준별 분석 REPORT



월 기준 평균 거래량과 평균 가격 분석

샤인마스캇 : 샤인마스캇 가격이 6월에 폭등하는 이유는 특정 시즌에 한정적으로 거래되어 수요에 비해 상품양이 적기 때문에 사전예약 등으로 가격이 많이 오르는 것으로 판단된다. 거래량은 다른 농산물에 비해 알려진지 오래 되지 않았고 가격도 높아 아직 많이 거래되고 있지 않는 것으로 보인다.

건고추 : 수확시기인 7-10월에 거래량이 증가하여 가격이 대체로 높으며, 수요가 많은 김장철(10-11월)에 거래량이 가장 많아지면서 가격이 낮아졌다가 이후 다시 높아지는 형태를 보인다.

캠벨얼리 : 8-9월이 제철인 과일로 거래량이 급격하게 증가하고 가격은 가장 낮은 시기이다.

양파 : 1-4월에 가격이 가장 높고 거래량도 3-4월에 가장 크게 증가한다. 제철인 6-9월에 저렴하게 구할 수 있고 3월부터 출하되기 때문에 거래량이 급증한다.

무, 배추 : 8월부터 점차 가격이 증가하여 김장철부터 제철인 겨울내내 거래량이 가장 많고, 이후 낮아져 평탄한 형태를 유지한다.

기준별 분석 REPORT





Machine Learning

신경망을 사용한 예측

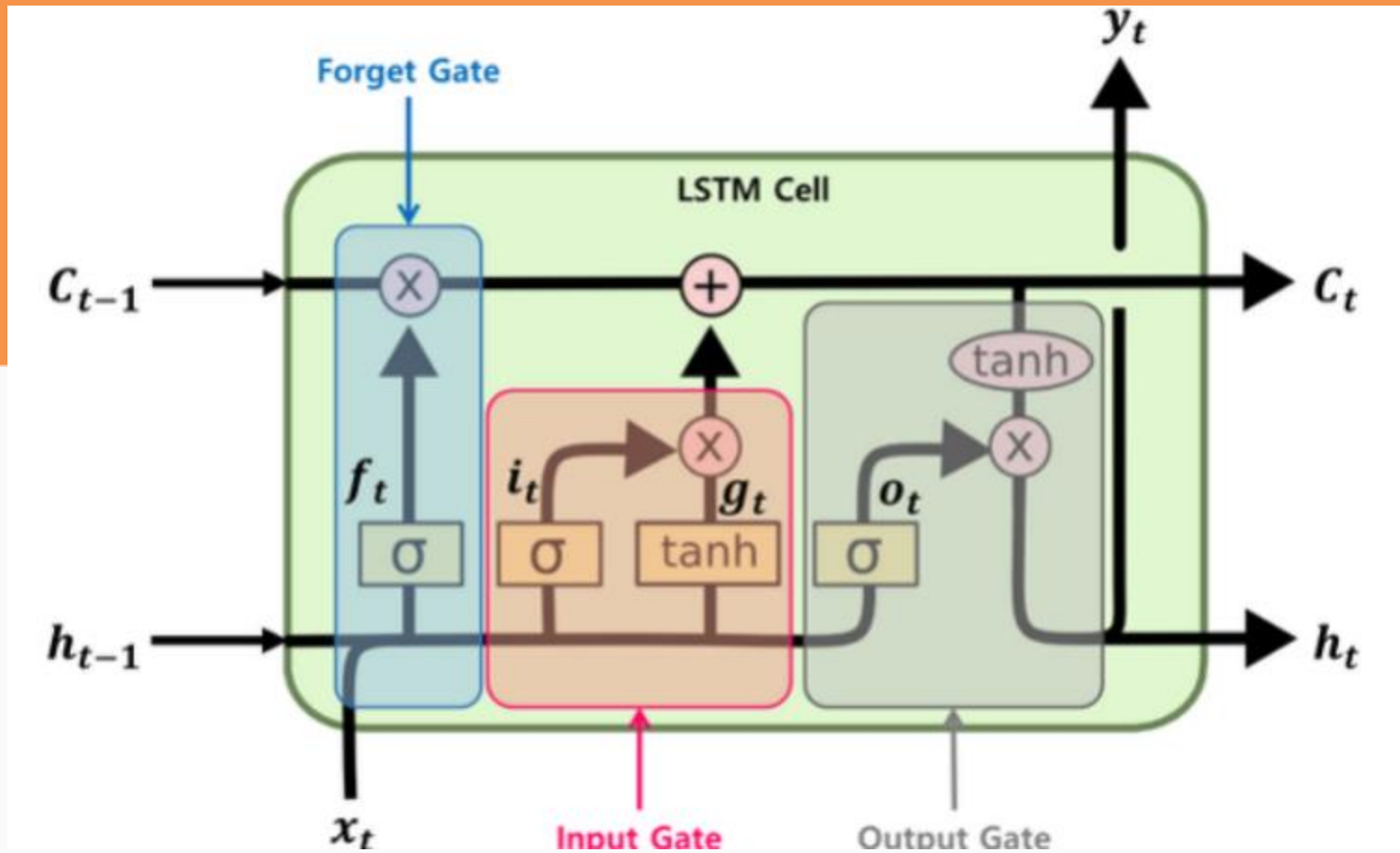
☑ 모델 생성

☑ 예측

☑ 예측 방법 개선

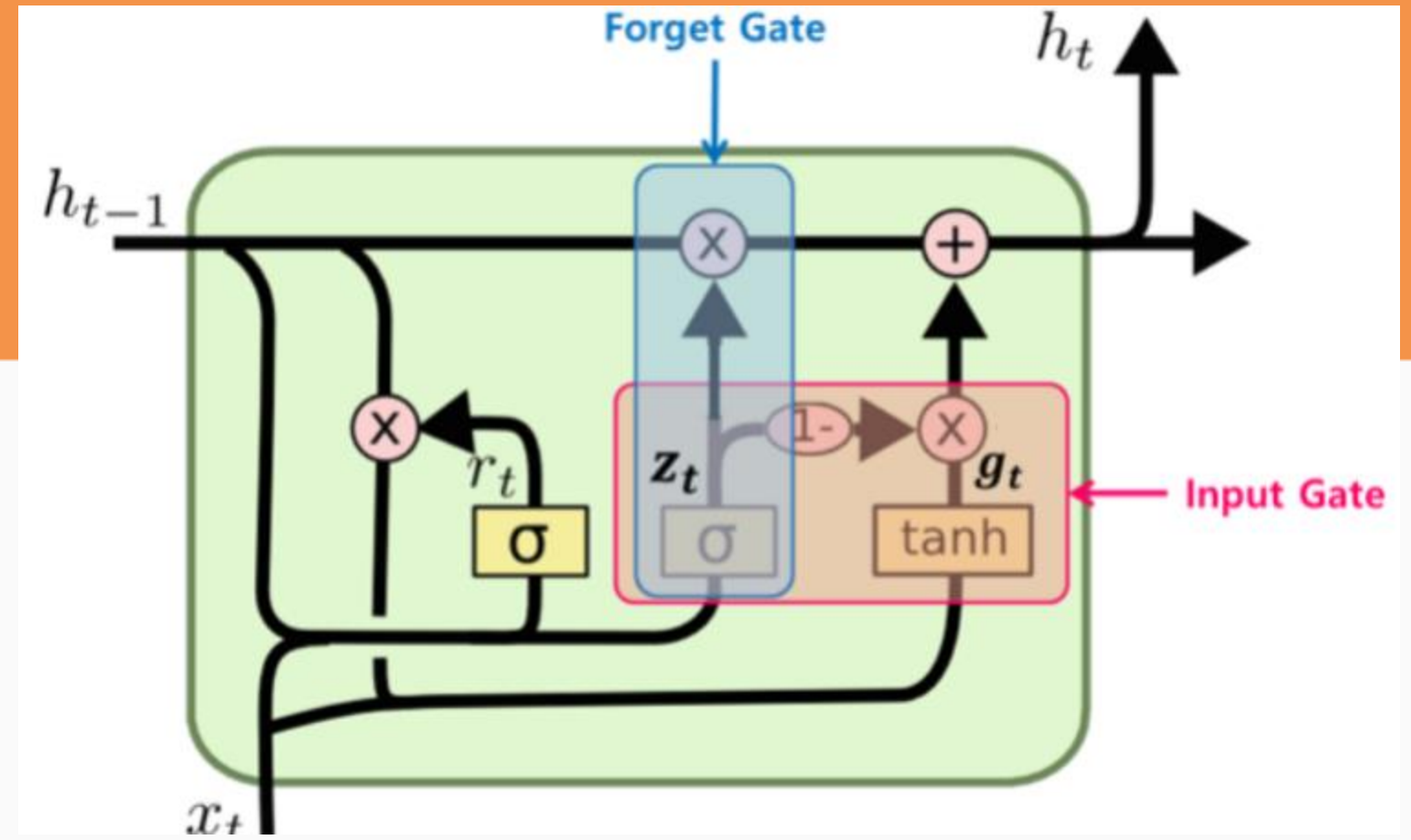
모델 설명

LSTM



LSTM 셀에서는 상태(state)가 두 개의 벡터 h_t (단기 상태)와 c_t (장기 상태)로 나누어진다. LSTM의 핵심은 네트워크가 장기 상태에서 기억할 부분, 삭제할 부분, 읽어 들일 부분을 학습하는 것으로 forget gate를 지나면서 일부는 기억(정보)을 잃고, 그 다음 input gate로 부터 새로운 기억 일부를 추가한다. 이렇게 만들어진 c_t 는 타임 스텝마다 일부의 기억을 삭제하고 추가하는 과정을 거치며, 덧셈 연산 후 복사되어 output gate의 tanh함수로 전달되어 h_t 와 셀의 출력인 y_t 를 만든다.

GRU



GRU는 LSTM 셀의 간소화된 버전으로, LSTM Cell에서의 두 상태 벡터 c_t 와 h_t 가 하나의 벡터 h_t 로 합쳐진다. 하나의 gate controller인 z_t 가 forget과 input 게이트(gate)를 모두 제어하며, 이전($t-1$)의 기억이 저장 될때 마다 타임 스텝 t 의 입력은 삭제된다. GRU 셀은 output 게이트가 없어 전체 상태 벡터 h_t 가 타임 스텝마다 출력되며, 이전 상태 h_{t-1} 의 어느 부분이 출력될지 제어하는 새로운 gate controller인 r_t 가 있다.

☑ 데이터가 적으면 LSTM보다 GRU가 더 적합

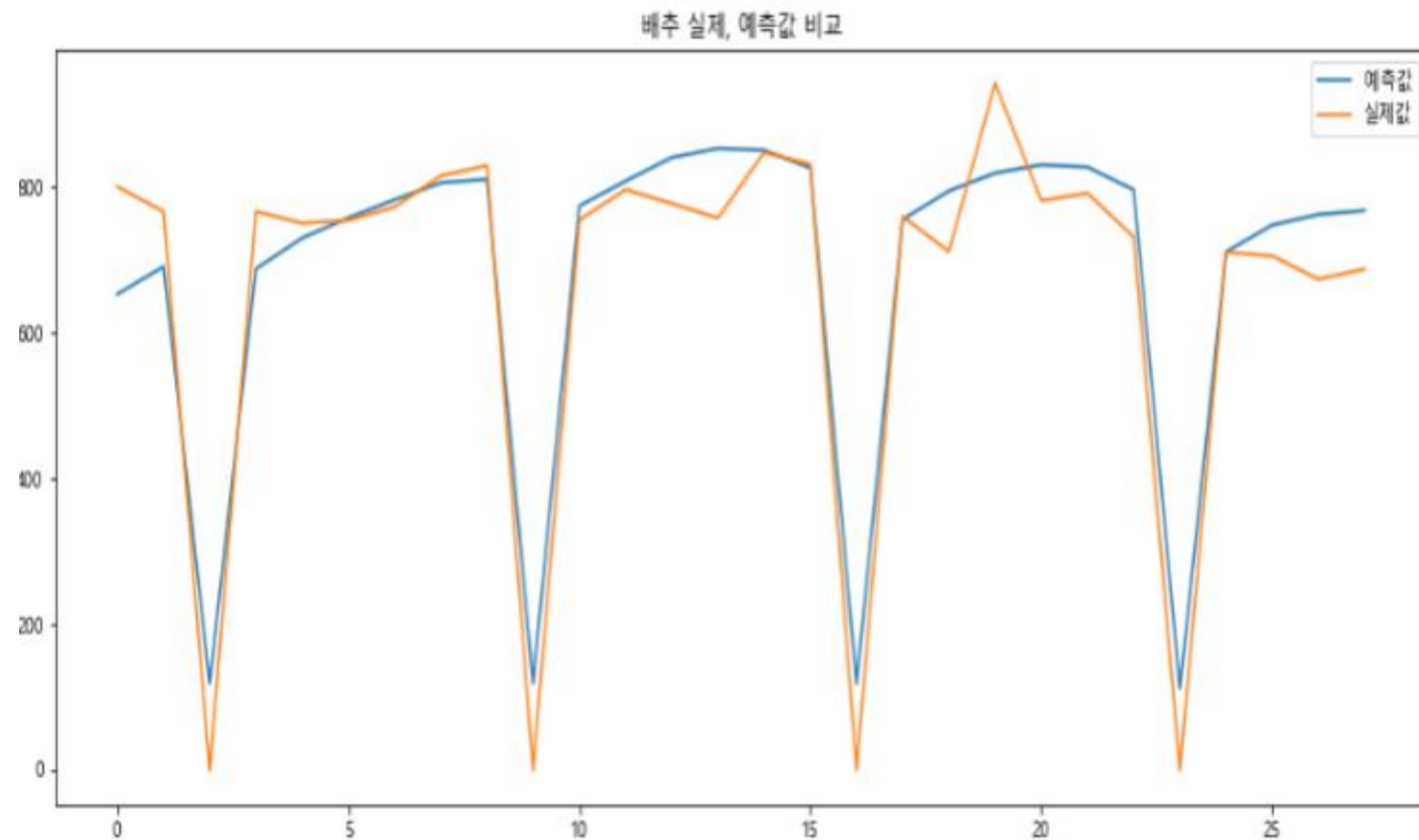
예측

LSTM

```
model = Sequential()
model.add(LSTM(400, activation = 'leaky_relu', input_shape = (28, 43), return_sequences=True))
model.add(Dropout(0.3))
model.add(LSTM(300, activation = 'leaky_relu', return_sequences=True))
model.add(Dropout(0.3))
model.add(LSTM(200, activation = 'leaky_relu', return_sequences=True))
model.add(Dropout(0.3))
model.add(Dense(100, activation = 'leaky_relu'))
model.add(Dense(50, activation = 'leaky_relu'))
model.add(Dropout(0.3))
model.add(Dense(21))

es = EarlyStopping(monitor='val_loss', patience=5)
optimizer = Adam(learning_rate=0.01)
model.compile(loss='mse', optimizer='adam', metrics=['mse'])
history = model.fit(x_data_train, y_data_train, epochs=100, batch_size=BATCH_SIZE,
                    validation_split=0.3, verbose=1, callbacks=[es])
```

평균 r2 0.06828907087272724
평균 mae 0.09700262013852856
평균 mse 0.020894660312816563

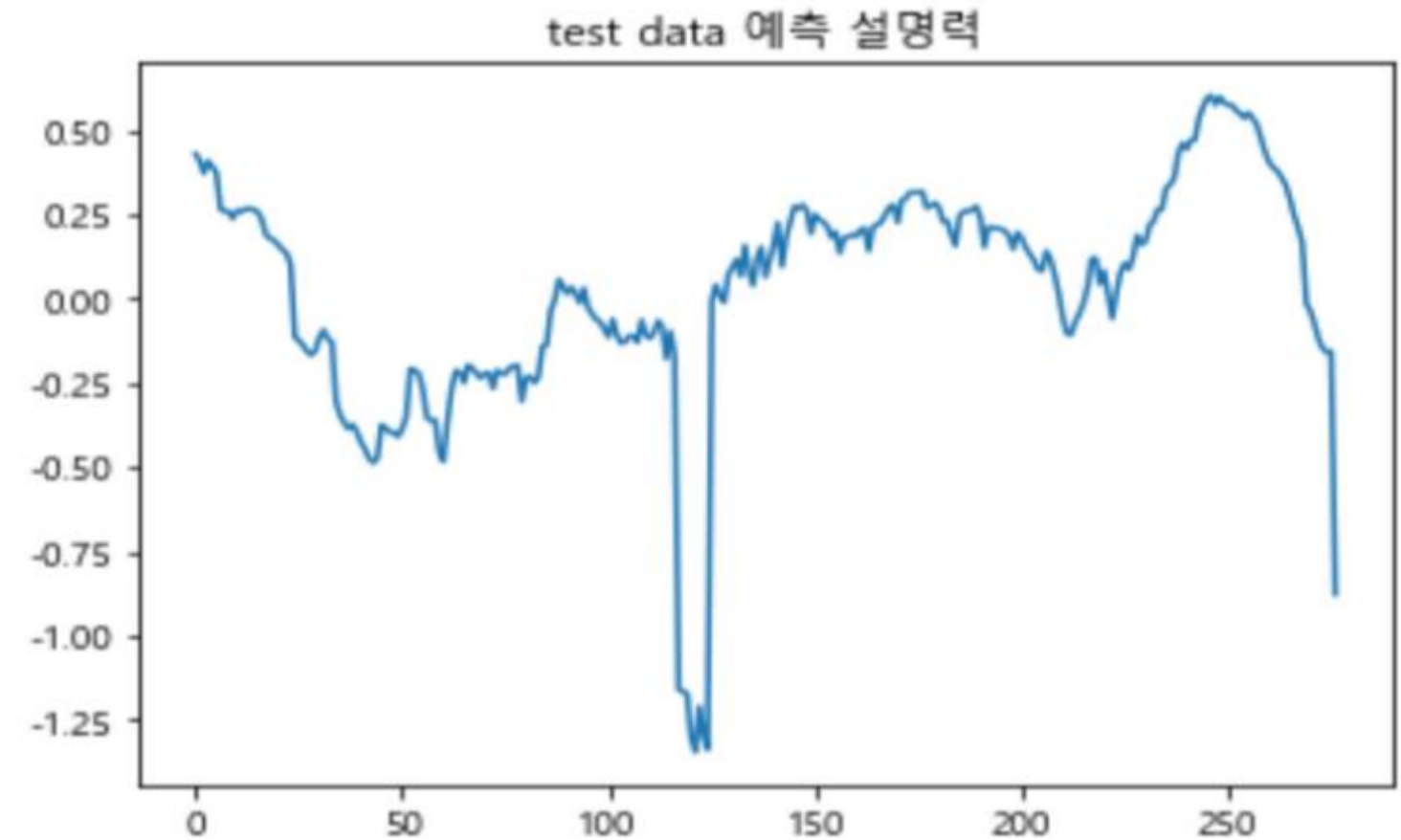


GRU

```
model = Sequential()
model.add(GRU(400, activation = 'relu', input_shape = (28, 43), return_sequences=True))
model.add(GRU(300, activation = 'relu', return_sequences=True))
model.add(Dropout(0.2))
model.add(Dense(200, activation = 'leaky_relu'))
model.add(Dense(100, activation = 'leaky_relu'))
model.add(Dropout(0.2))
model.add(Dense(21))

optimizer = RMSprop(learning_rate=0.001)
es = EarlyStopping(monitor='val_loss', patience=5)
model.compile(loss='mse', optimizer=optimizer, metrics=['mse'])
history = model.fit(x_data_train, y_data_train, epochs=100, batch_size=BATCH_SIZE,
```

평균 r2 0.07909671574752568
평균 mae 0.10454512562321266
평균 mse 0.022832093671432543

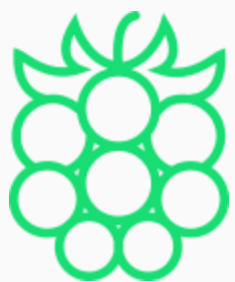


예측 방법 개선안



파라미터 값 조정

활성화 함수, 학습률 등 여러 파라미터 값을 조정해
모델 최적화



샤인마스캇 분리

샤인마스캇의 가격과 거래량이 독립적 성향을 띠므로 결과
예측에 좋지 않은 영향을 미친다고 예상
이에 따라 농산물을 20:1로 나눠서 따로 예측



개별 예측

모든 농산물을 21개의 모델로 개별 예측



Conclusion

결론

—



결론

LSTM과 GRU를 사용해 21개의 농산물의 한달 후 가격을 예측하는 모델을 만들어 보았지만 전체적으로 예측력은 좋지 않았다. 약간의 패턴을 찾기는 했지만 과소적합인 모습을 보였다.

21개의 농산물의 가격과 거래량은 특수한 조건을 가진 농산물을 제외하고는 어느 정도 상관성을 띠었다.

두 개의 모델을 사용하여 이상치 존재 유무에 따른 예측, 개별 농산물 예측 등 다양한 개선을 시도 해보았고 설명력이 규칙적으로 떨어지는 구간이 있다는 것을 발견했다.

따라서 모델의 세부적인 조정을 통해 성능 개선이 필요하다.

THANK YOU

TEAM DATA ARTISTS

