Prediction of the Number of COVID-19 Infected Using XGBoost

XGBoost를 활용한 코로나19 예측

도 시 계 획 실 습 (4)
이 훈 I 김하연 I 이산하

01 선정 배경

코로나19 (COVID-19)

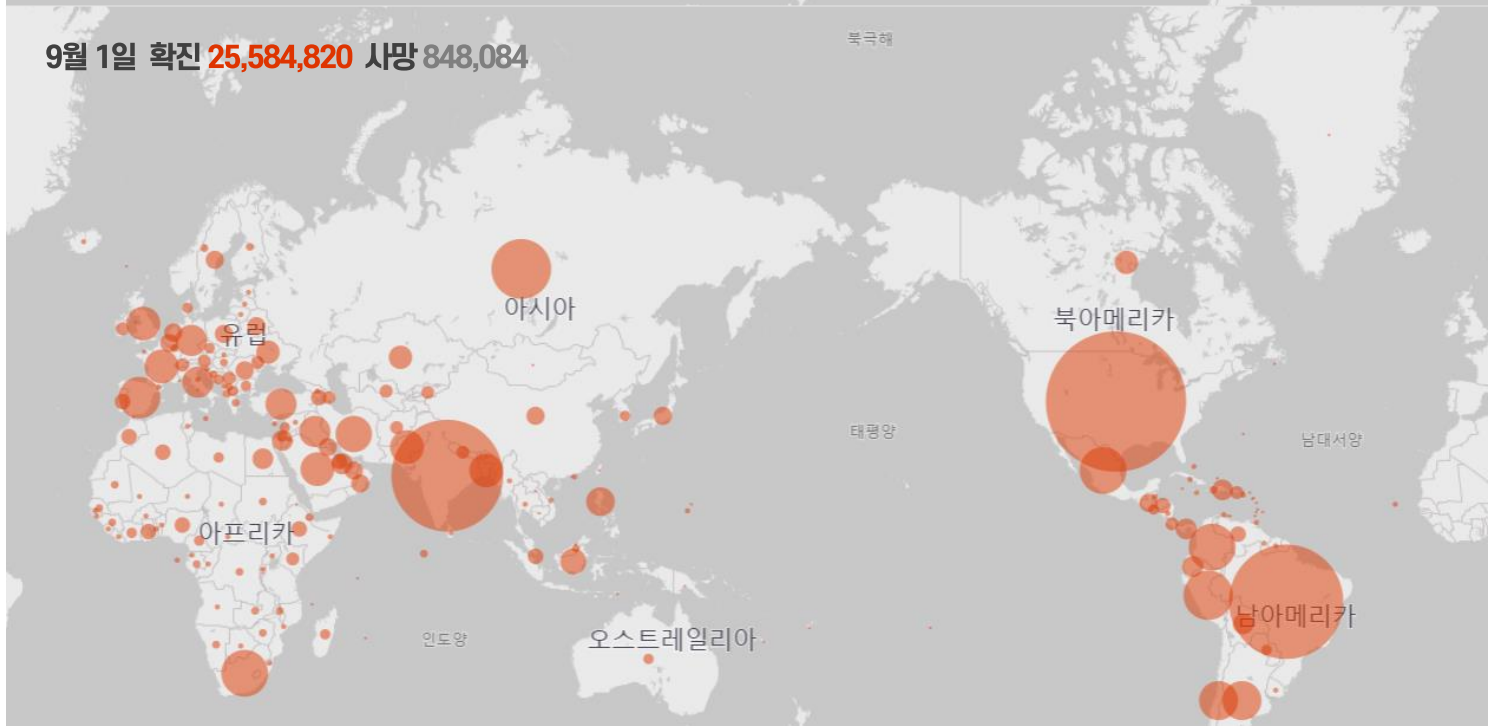
2019년 12월 중국 우한에서 처음 발생한 이후
전 세계로 확산된, 새로운 유형의 코로나바이러스

팬데믹 (PANDAMIC)

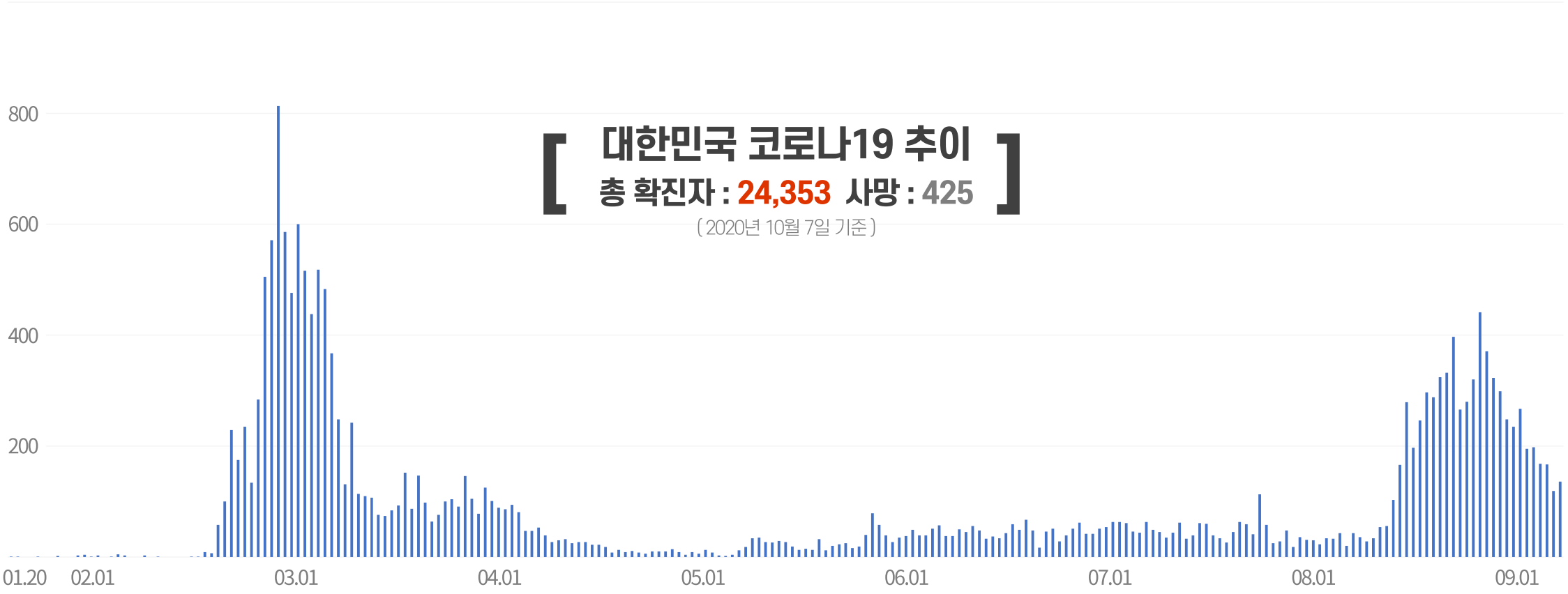
세계보건기구(WHO)는 홍콩독감, 신종플루에 이어
사상 세 번째로 감염병 최고 경고 등급인 팬데믹 선언

짧아지는 전염병 주기

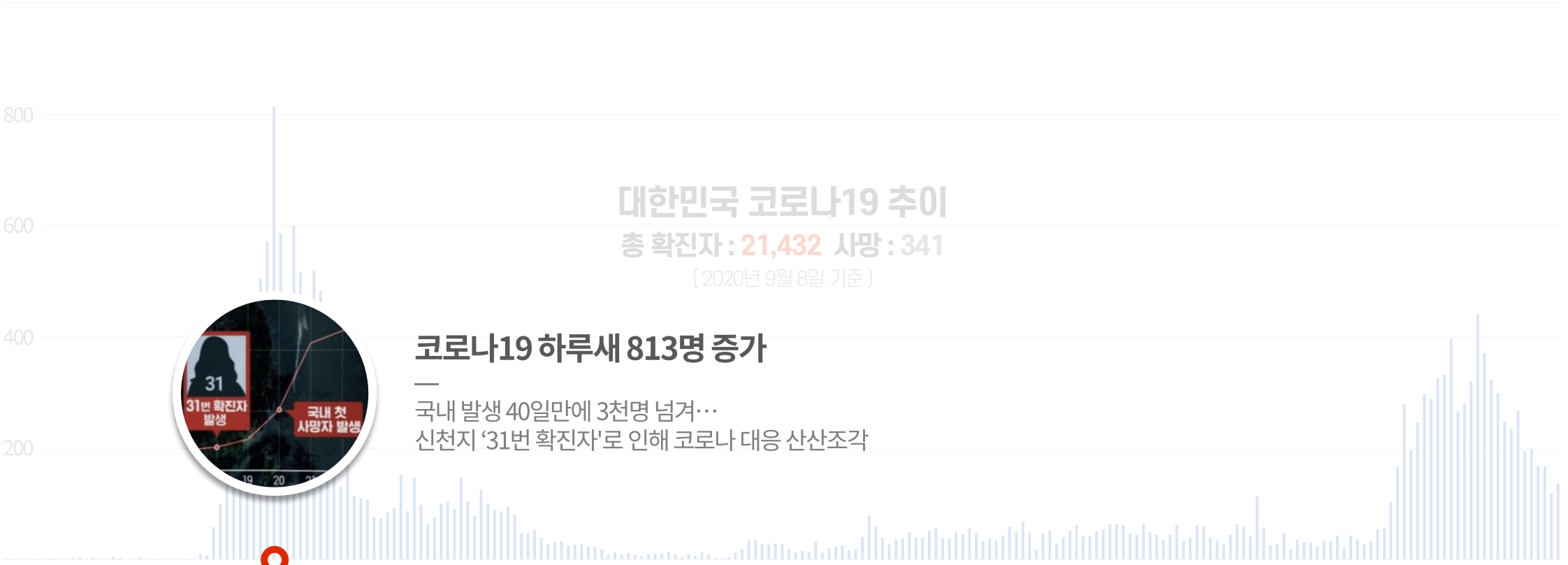
사스와 메르스, 코로나19까지
점점 짧아지는 전염병 발생 주기



01 선정 배경


대한민국 코로나19 추이
총 확진자 : 24,353 사망 : 425
(2020년 10월 7일 기준)
800
600
400
200
01.20
02.01
03.01
04.01
05.01
06.01
07.01
08.01
09.01

01 선정 배경

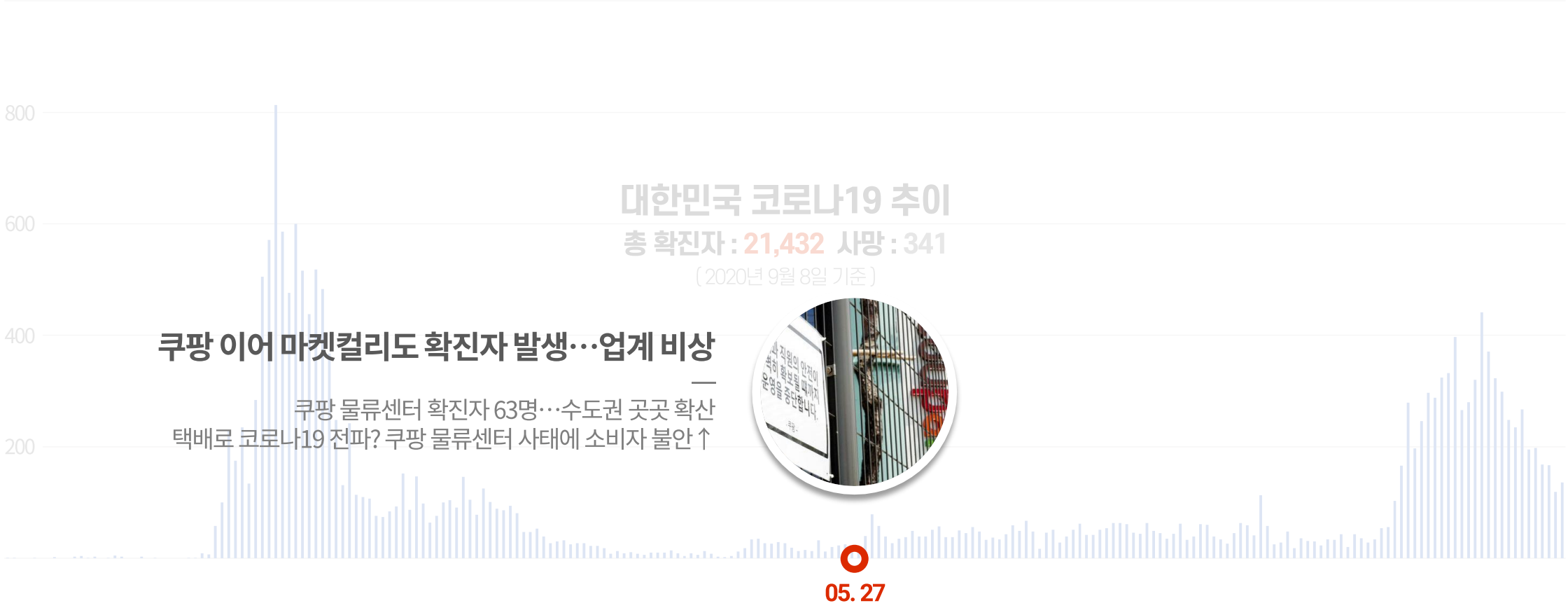


코로나19 하루새 813명 증가

국내 발생 40일만에 3천명 넘겨…
신천지 '31번 확진자'로 인해 코로나 대응 산산조각

01

선정 배경

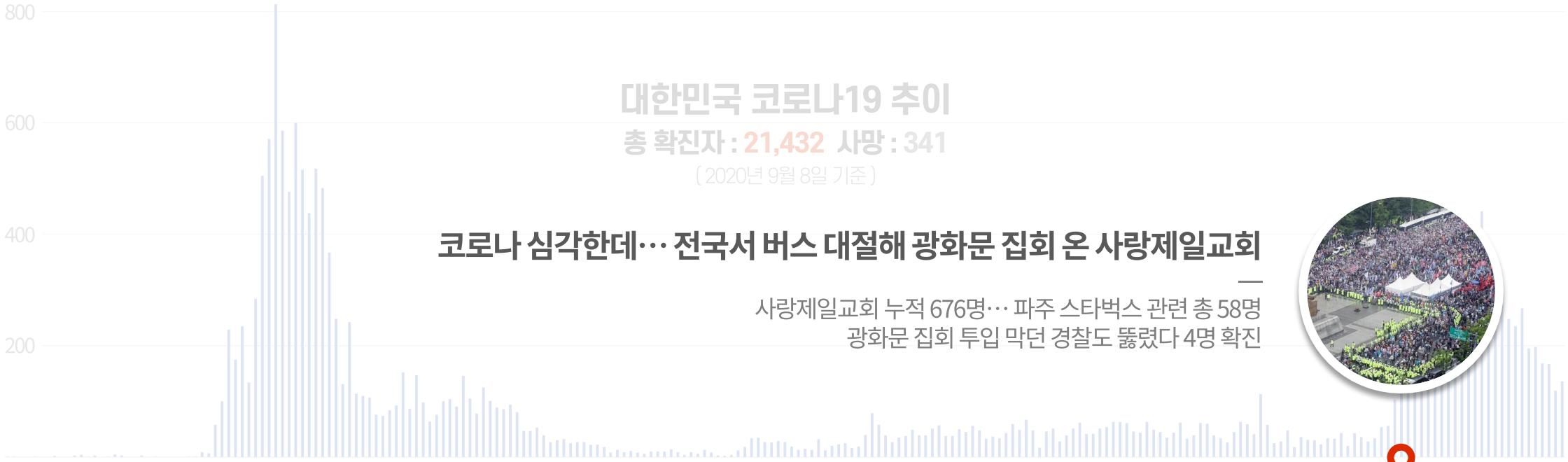


쿠팡 이어 마켓컬리도 확진자 발생…업계 비상

—

쿠팡 물류센터 확진자 63명…수도권 곳곳 확산
택배로 코로나19 전파? 쿠팡 물류센터 사태에 소비자 불안↑

01 선정 배경



코로나 심각한데… 전국서 버스 대절해 광화문 집회 온 사랑제일교회

사랑제일교회 누적 676명… 파주 스타벅스 관련 총 58명
광화문 집회 투입 막던 경찰도 뚫렸다 4명 확진

08. 15

02 연구 주제

주제

인공지능 기법을 통한 **국내 코로나19 확진자 수 예측**

사용 기법

인공지능 기법 중 머신러닝을 사용하여 예측 분석
머신러닝은 XGBoost 모델 채택

보완

Regression Analysis를 통해 통계적으로 유의한지 확인

03 방법론

머신 러닝 (Machine Learning)

- 인공지능(AI)의 한 분야로, 컴퓨터에 명시적인 프로그램 없이 배울 수 있는 능력을 부여하는 연구 분야
- 사람이 학습하듯이 컴퓨터에도 데이터들을 줘서 학습하게 함으로써 새로운 지식을 얻어내게 하는 분야

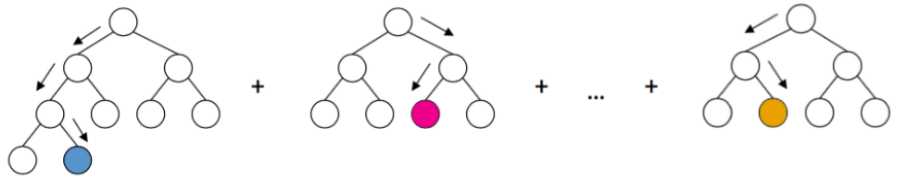
회귀분석 (Regression Analysis)

- 매개변수를 이용하여 통계적으로 변수들 사이의 관계를 추정하는 분석방법
- 독립변수가 종속변수에 미치는 영향을 확인하고자 사용하는 분석방법
- 종속변수와 관련이 있는 독립변수를 찾을 때, 또 독립변수들 간의 관계를 이해하고자 할 때 사용

03 방법론 - 머신러닝

XGBoost (eXtreme Gradient Boosting)

- Boosting 기법 중 Tree Boosting 기법을 활용한 모델
- Tree Boosting 방식에 경사하강법을 사용하여 optimization을 하는 모델
- 여러 개의 Decision Tree를 사용하지만 단순히 결과의 평균을 내는 것이 아니라 결과를 보고 오답에 대한 가중치 부여
- 가중치가 적용된 오답에 대해서는 정답이 될 수 있도록 결과를 만들고 해당 결과에 대한 다른 오답을 찾아 다시 똑같은 작업을 반복적으로 진행
- GBMd에 기반하고 있지만, GBM의 단점인 느린 수행시간과 과적합 규제(Regularization) 부재 등의 문제 보완
- 연산량을 줄이기 위해 Decision Tree를 구성할 때 병렬 처리를 사용해 빠른 시간에 학습 가능
- 분류와 회귀영역에서 뛰어난 예측 성능 발휘 및 결손값 자체 처리
- 평가 함수를 포함하여 다양한 커스텀 최적화 옵션을 제공하는 등 높은 유연성 보유
- Early Stopping 기능 보유



앙상블(Ensemble)

: 여러 개의 학습 알고리즘을 사용해 더 좋은 성능을 얻는 방법

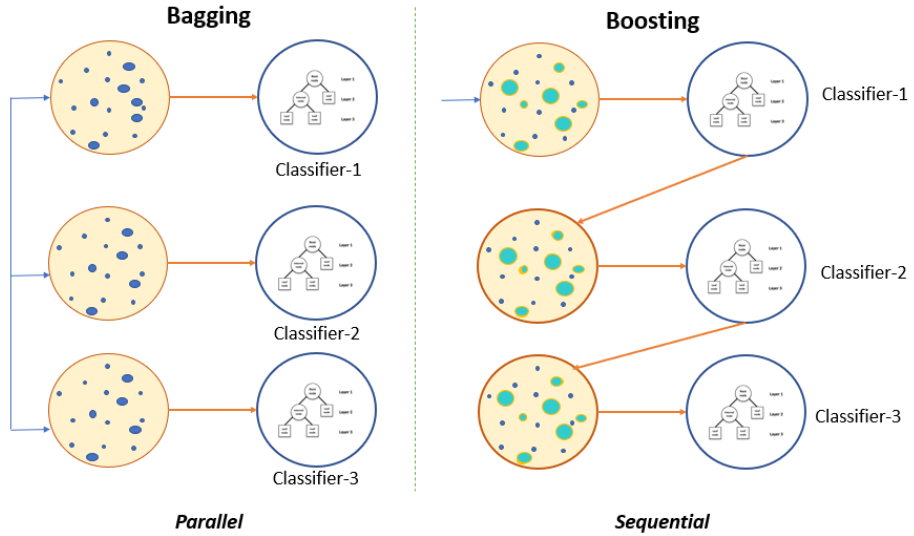
: 방식에 따라 Bagging과 Boosting으로 분류

Bagging

: 여러 개의 학습 알고리즘, 모델을 통해 각각 결과를 예측하고 모든 결과를 동등하게 보고 취합해서 결과를 얻는 방식

Boosting

: 여러 알고리즘, 모델의 결과를 순차적으로 취합하는데, 단순히 하나씩 취하는 방법이 아니라 이전 알고리즘, 모델이 학습 후 잘못 예측한 부분에 가중치를 줘서 다시 모델로 가서 학습하는 방식



04 연구 방법 - 데이터 및 매개 변수

데이터 수집

O28

	A	B	C	D	E	F	G	H	I	J	K	L
1	Confirmed	Deaths	Recovered	PTmask	Mask	Social	Support	event	dayname	day	week	month
2	1	0	0	0	0	0	0	0	1	20	4	1
3	1	0	0	0	0	0	0	0	5	21	4	1
4	0	0	0	0	0	0	0	0	6	22	4	1
5	0	0	0	0	0	0	0	0	4	23	4	1
6	1	0	0	0	0	0	0	0	0	24	4	1
7	0	0	0	0	0	0	0	0	2	25	4	1
8	0	0	0	0	0	0	0	0	3	26	4	1
9	2	0	0	0	0	0	0	0	1	27	5	1
10	0	0	0	0	0	0	0	0	5	28	5	1
11	0	0	0	0	0	0	0	0	6	29	5	1
12	3	0	0	0	0	0	0	0	4	30	5	1
13	4	0	0	0	0	0	0	0	0	31	5	1
14	1	0	0	0	0	0	0	0	2	1	5	2
15	3	0	0	0	0	0	0	0	3	2	5	2
16	0	0	0	0	0	0	0	0	1	3	6	2
17	1	0	0	0	0	0	0	0	5	4	6	2
18	5	0	1	0	0	0	0	0	6	5	6	2
19	3	0	1	0	0	0	0	0	4	6	6	2
20	0	0	0	0	0	0	0	0	0	7	6	2
21	0	0	0	0	0	0	0	0	2	8	6	2
22	3	0	1	0	0	0	0	0	3	9	6	2
23	0	0	1	0	0	0	0	0	1	10	7	2

변수 설정

Ptmask
대중교통 마스크 착용 의무화

Mask
마스크 착용 의무화

Social
사회적 거리두기 단계

Support
정부재난지원금 사용기간

Event
대규모 집단감염사태

dayname, day, week, month
해당 요일, 일자, 주차, 월

04

연구 방법 - 환경설정

In [30]:
```
data.head()
```

Out[30]:

	Confirmed	Deaths	Recovered	PTmask	Mask	Social	Support	Event	dayname	day	week	month
0	1	0	0	0	0	0.0	0	0	1	20	4	1
1	1	0	0	0	0	0.0	0	0	5	21	4	1
2	0	0	0	0	0	0.0	0	0	6	22	4	1
3	0	0	0	0	0	0.0	0	0	4	23	4	1
4	1	0	0	0	0	0.0	0	0	0	24	4	1

모델링

In [31]:
```
test=data.iloc[-30:,:]
train=data.iloc[:-30,:]

X_train=train.drop('Confirmed',axis=1)
Y_train=train['Confirmed']

X_test=test.drop('Confirmed',axis=1)
Y_test=test['Confirmed']
```

In [32]:
```
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(210, 11) (210,)
(30, 11) (30,)
```

Modeling

① 그래프 시각화, 한글화 설정

② csv 데이터 추출, 파생변수 생성

③ LabelEncoder 모듈을 활용해 문자를 숫자로 매핑

④ 7:1 비율로 학습 셋, 테스트 셋 분할

In [33]:
```
from xgboost import XGBRegressor
from sklearn.metrics import mean_squared_error,mean_ab
```

In [34]:
```
xgb=XGBRegressor()
xgb.fit(X_train,Y_train)
Y_pred=xgb.predict(X_test)
print('RMSE:',np.sqrt(mean_squared_error(Y_test,Y_pred)))
print('MAE:',mean_absolute_error(Y_test,Y_pred))
```

```
RMSE: 144.88148927492082
MAE: 120.05676523844402
```

In [35]:
```
plt.figure(figsize=[15,8])
sns.lineplot(x=dt[210:],y=Y_test.to_list())
sns.lineplot(x=dt[210:],y=Y_pred)
plt.ylim(0,600)
plt.title('실제 확진자 수와 예측 확진자 수')
```

Out[35]: Text(0.5, 1.0, '실제 확진자 수와 예측 확진자 수')

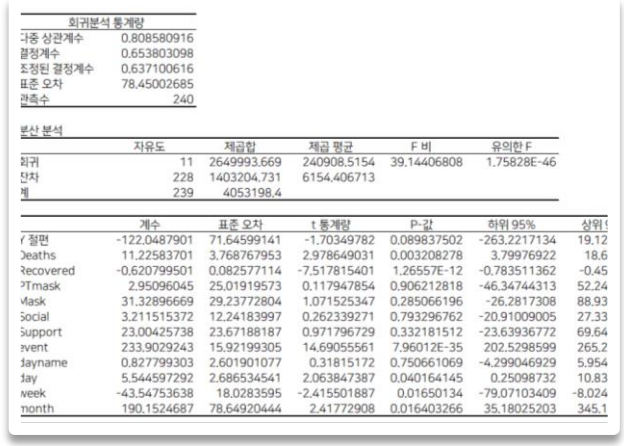
Machine Learning

① XGBoost 모델 구축

② 학습셋으로 모델 규칙 생성

③ RMSE, MAE 추출

④ 그래프 구현 및 가시성 조정



회귀분석 통계량

다중 상관계수	0.808580916
결정계수	0.653803098
조정된 결정계수	0.637100616
표준 오차	78.45002685
관측수	240

분산 분석

	자유도	제곱합	제곱 평균	F 비	유의한 F
회귀	11	2649993.669	240908.5154	39.14406808	1.75828E-46
단차	228	1403204.731	6154.406713		
계	239	4053198.4			

	계수	표준 오차	t 통계량	P-값	하위 95%	상위
Y 절편	-122.0487901	71.64599141	-1.70349782	0.089837502	-263.2217134	19.12
Deaths	11.22583701	3.768767953	2.978649031	0.003208278	3.79976922	18.6
Recovered	-0.620799501	0.082577114	-7.517815401	1.26557E-12	-0.783511362	-0.45
PTmask	2.95096045	25.01919573	0.117947854	0.906212818	-46.34744313	52.24
Mask	31.32896669	29.23772804	1.071525347	0.285066196	-26.2817308	88.93
Social	3.211515372	12.24183997	0.262339271	0.793296762	-20.91009005	27.33
Support	23.00425738	23.67188187	0.971796729	0.332181512	-23.63936772	69.64
Event	233.9029243	15.92199305	14.69055561	7.96012E-35	202.5298599	265.2
dayname	0.827799303	2.601901077	0.31815172	0.750661069	-4.299046929	5.954
day	5.544597292	2.686534541	2.063847387	0.040164145	0.25098732	10.83
week	-43.54753638	18.0283595	-2.415501887	0.01650134	-79.07103409	-8.024
month	190.1524687	78.64920444	2.41772908	0.016403266	35.18025203	345.1

Regression Analysis

① 독립변수와 종속변수의 통계적 상호관계 확인

② 결정계수, 유의한 F, P-값 확인

③ 유의하다고 판단되는 6개의 변수만으로 다시 분석

④ 계수, Y절편으로 식 성립 후 오차 계산

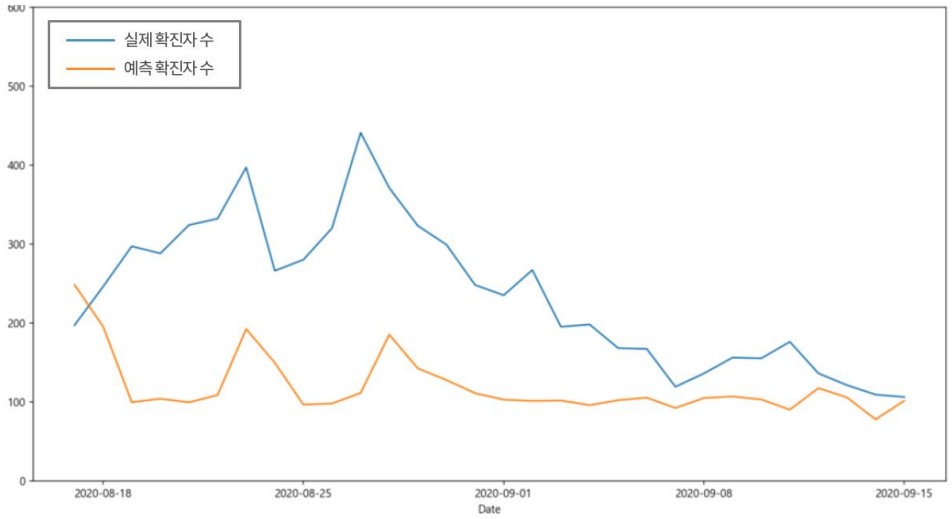
⑤ MSE, RMSE 추출

평균 제곱근 오차(RMSE : Root Mean Squared Error) : 예측 오차는 양수와 음수로 나타나기 때문에 오차를 제곱하여 n으로 나눈 값인 MSE를 다시 제곱근 시킨 값

평균 절대 오차(MAE : Mean Absolute Error) : 오차의 크기만 고려하기 위해 오차의 절댓값을 씌우고 데이터 수로 나눈 값

05 연구 결과

머신러닝 [XGBoost]

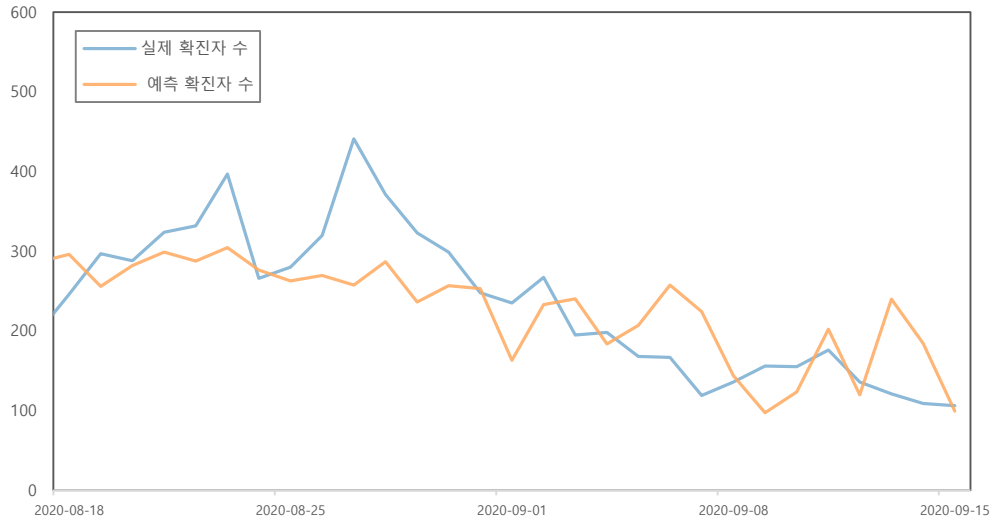


MAE : 120.06

MSE : 20,990.64

RMSE : 144.88

회귀 분석 [Regression Analysis]



MSE : 5892.36

RMSE : 76.76

RMSE : 표준편차를 일반화시킨 척도로써 실제 값과 예측값의 차이가 얼마인지를 나타내는 수치. 값이 작을수록 정밀도가 높음.

05 연구 결론

결과

- 머신러닝(XGBoost)은 144.88의 RMSE를 출력
- 통계적으로 유의한지에 대한 회귀분석은 76.46의 RMSE를 출력
- 유의하다고 판단된 6개의 변수(사망자 수, 회복자 수, 대규모 집단 전염 사태, 일자, 주차, 월)만으로 다시 진행한 회귀분석은 76.76의 RMSE를 출력

한계

- 인공지능은 보통 수만 개 이상의 타임 시리즈가 전제되어야 하지만, 연구에 사용된 데이터는 코로나 발생이 시작된 이후의 데이터로 240일 정도에 불과
- 전염 예측에 중요한 의학 데이터나 위치 데이터 등 개인정보 관련 데이터들은 개인이 수집하는 데 제한적이라 테스트 셋을 설정 및 미래 예측에 한계가 존재

결론

- 회귀분석 결과, 6개의 독립변수 이외의 변수들은 통계적으로 다소 유의하지 않은 것으로 판단
- 따라서, 향후 보다 전염예측에 정확하고, 효과적인 변수를 반영하여 예측 오차를 줄이는 것에 대한 추가적인 연구가 필요할 것으로 판단


Thank You