시계열마법사

1팀
김태현
박상훈
곽동길
권능주
박윤아
이경미

INDEX

1

데이터 EDA

1 데이터 EDA

데이터 구조 파악

Train data: **13,000 X 505**

Test data: **2,000 X 504**

13,000 X 505

Id	시계열(0~499)			Year	Country	S/N	Label
0	-27.4	…	-33.4	1998	중국	PSCG-68053	0
1	-8.8	…	-25.6	2009	美国	PSCG-79993	1
…							
12999	-61.3	…	-46.5	1992	america	PSCG-74202	0

train.csv

1 데이터 EDA

데이터 구조 파악

Train data: **13,000 X 505**

Test data: **2,000 X 504**

2,000 X 504

Id	시계열(0~499)			Year	Country	S/N
0	-35.5	…	-31.2	1995	중국	PSCG-64661
1	-7.7	…	5.5	1994	South Korea	PSCG-28815
			…			
1999	-0.125	…	-0.135	1994	중국	PSFT-05896

test.csv

데이터 구조 파악 | Country

Train data: **13,000 X 505**

Test data: **2,000 X 504**

13,000 X 505

Id	시계열(0~499)			Year	Country	S/N	Label
0	-27.4	…	-33.4	1998	중국	PSCG-68053	0
1	-8.8	…	-25.6	2009	美国	PSCG-79993	1
…							
12999	-61.3	…	-46.5	1992	america	PSCG-74202	0

train.csv

데이터 구조 파악 | Country

중국, 미국, 한국 세 국가만 있지만 표현 방법이 다양함

⋮

단일 표현으로 통일!

중국 · china · 中国	CHN
미국 · America · U.S · 美国	USA
한국 · 대한민국 · Korea · South Korea · 韩国	KOR

1 데이터 EDA

데이터 구조 파악 | S/N

Train data: **13,000 X 505**

Test data: **2,000 X 504**

13,000 X 505

Id	시계열(0~499)			Year	Country	S/N	Label
0	-27.4	…	-33.4	1998	중국	PSCG-68053	0
1	-8.8	…	-25.6	2009	美国	PSCG-79993	1
…							
12999	-61.3	…	-46.5	1992	america	PSCG-74202	0

train.csv

1 데이터 EDA

데이터 구조 파악 | S/N

S/N는 제품과 관련된 번호로 추측

S/N 열의 데이터 모두 PSCG, PSFT + 5자리 숫자로 이루어짐

```python
train['S/N_prefix'] = train['S/N'].str[:4]

# 'S/N_prefix' 열의 cardinality(고유값 개수)를 계산합니다.
cardinality = train['S/N_prefix'].nunique()

print(f"S/N 열의 앞 4글자 패턴의 cardinality: {cardinality}")

# 'S/N_prefix' 열의 고유한 값들을 출력합니다.
unique_prefixes = train['S/N_prefix'].unique()
print("고유한 앞 4글자 패턴:")
print(unique_prefixes)
```

```
S/N 열의 앞 4글자 패턴의 cardinality: 2
고유한 앞 4글자 패턴:
['PSCG' 'PSFT']
```


```
S/N 열의 앞 4글자 패턴의 cardinality: 2
고유한 앞 4글자 패턴:
['PSCG' 'PSFT']
```

데이터 구조 파악 | Year

Train data: **13,000 X 505**

Test data: **2,000 X 504**

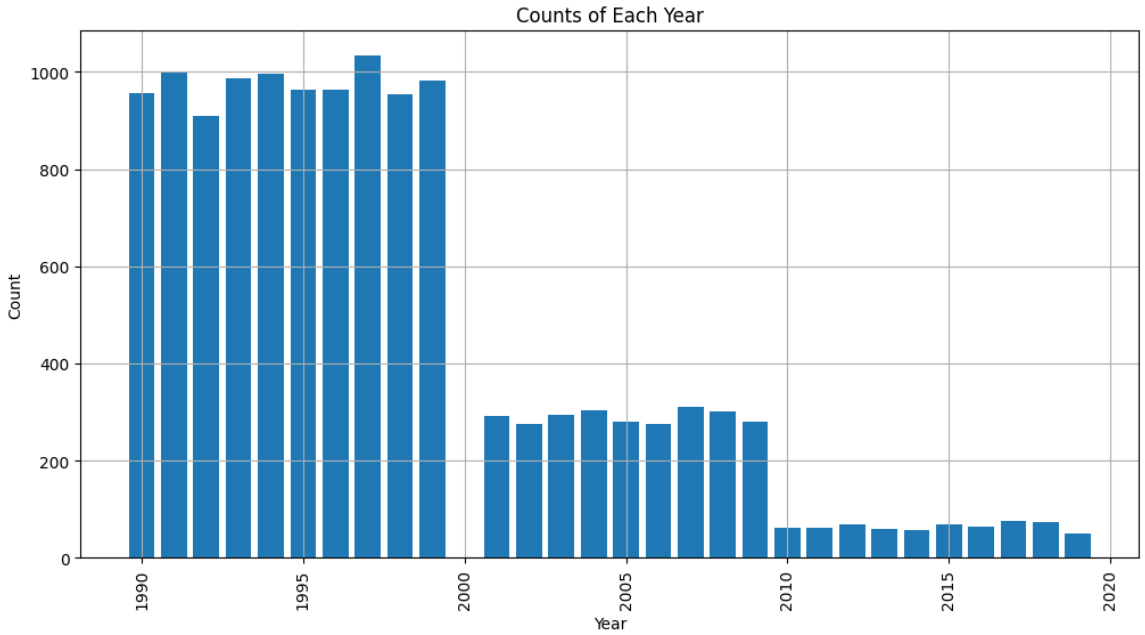
13,000 X 505

Id	시계열(0~499)			Year	Country	S/N	Label
0	-27.4	…	-33.4	1998	중국	PSCG-68053	0
1	-8.8	…	-25.6	2009	美国	PSCG-79993	1
…							
12999	-61.3	…	-46.5	1992	america	PSCG-74202	0

train.csv

1 데이터 EDA

데이터 구조 파악 | Year



10년 단위로 데이터의 수가 비슷하게 유지됨

1990년대 데이터 수가 가장 많고, 2010년대 데이터 수가 가장 적음

데이터 구조 파악 | Label

Train data: **13,000 X 505**

Test data: **2,000 X 504**

13,000 X 505

Id	시계열(0~499)			Year	Country	S/N	Label
0	-27.4	…	-33.4	1998	중국	PSCG-68053	0
1	-8.8	…	-25.6	2009	美国	PSCG-79993	1
…							
12999	-61.3	…	-46.5	1992	america	PSCG-74202	0

train.csv

1 데이터 EDA

데이터 구조 파악 | Label

Label은 0, 1, 2가 있음

Label 1은 10000개, Label 1과 2는 각각 1500개로 이루어짐

Label1과 Label2의 개수가 적기 때문에

데이터 불균형 문제를 해결해야 함

뒤에서 자세히 다룰 예정!

1 데이터 EDA

데이터 구조 파악 | 시계열 데이터

Train data: **13,000 X 505**

Test data: **2,000 X 504**

13,000 X 505

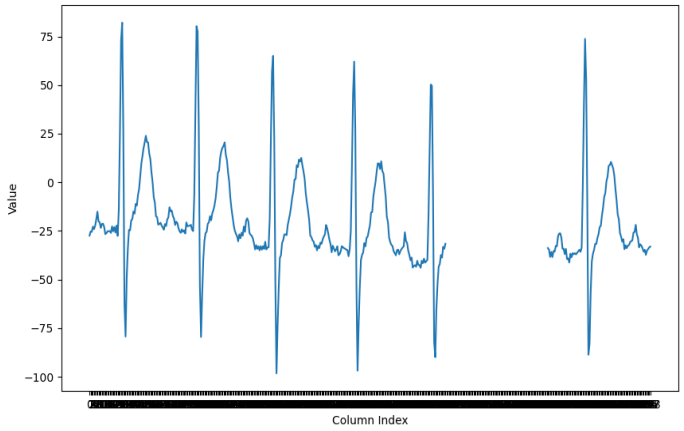
Id	시계열(0~499)			Year	Country	S/N	Label
0	-27.4	…	-33.4	1998	중국	PSCG-68053	0
1	-8.8	…	-25.6	2009	美国	PSCG-79993	1
…							
12999	-61.3	…	-46.5	1992	america	PSCG-74202	0

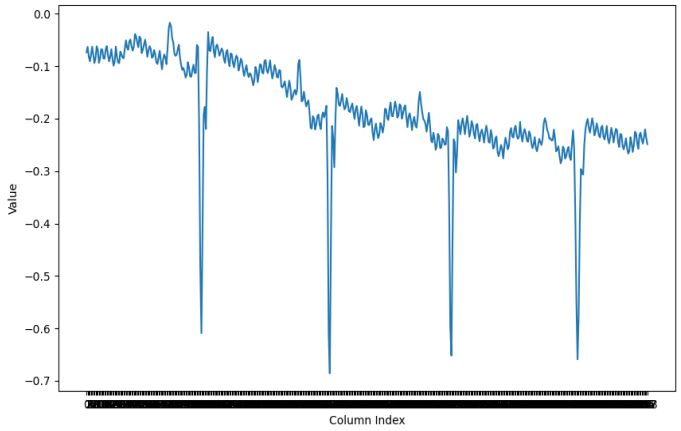
train.csv

데이터 구조 파악 | 시계열 데이터

행 방향으로 이루어진 시계열 데이터

데이터 형태 파악을 위해 시각화 진행



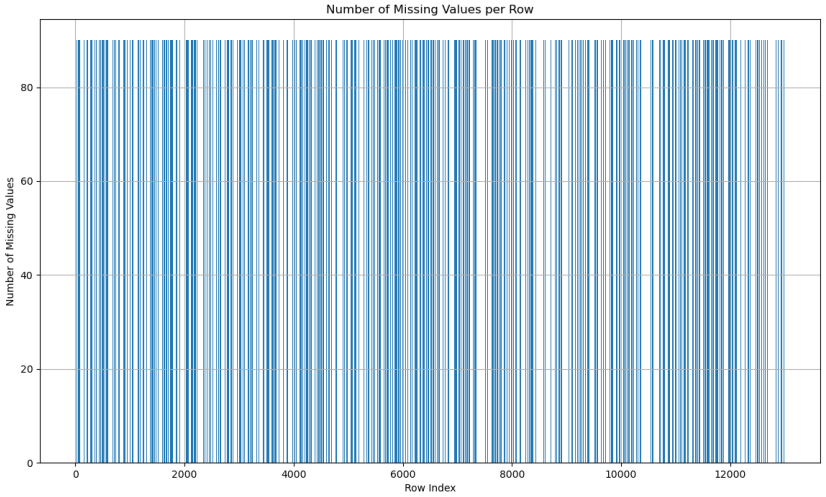


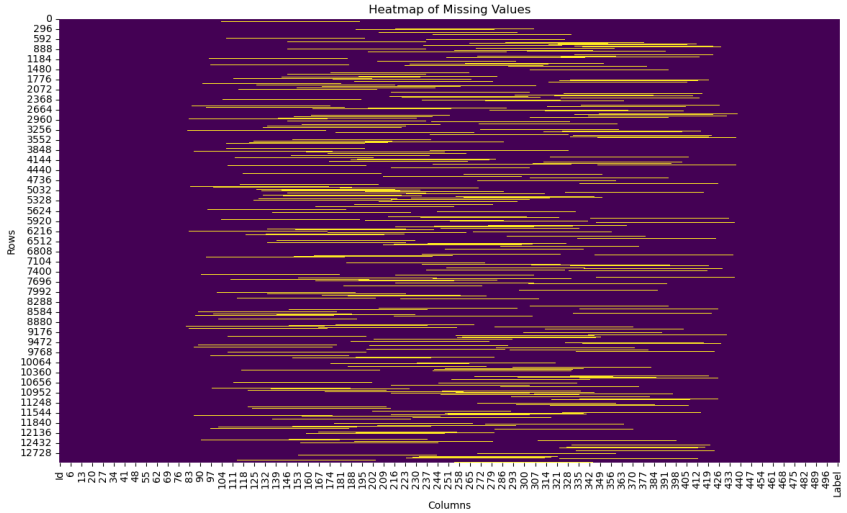
주기성이 있는 시계열 데이터가 있음을 확인

1 데이터 EDA

결측치 확인

결측치 파악을 위해 다양한 방법으로 결측치 그래프 시각화

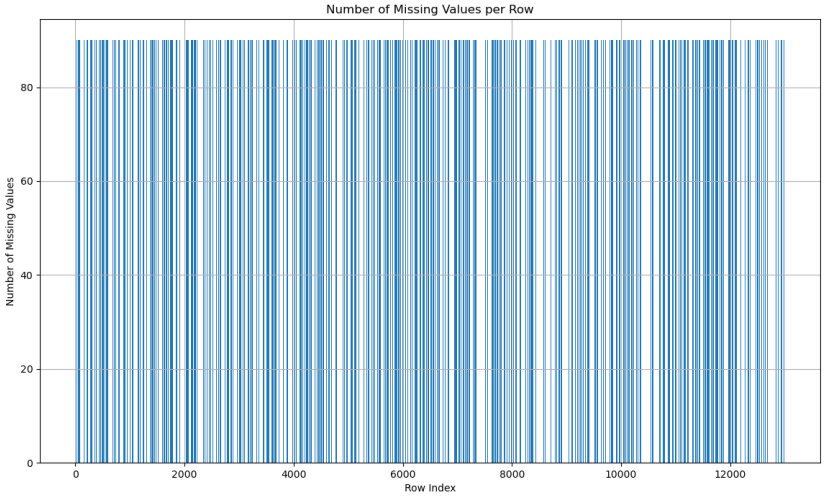


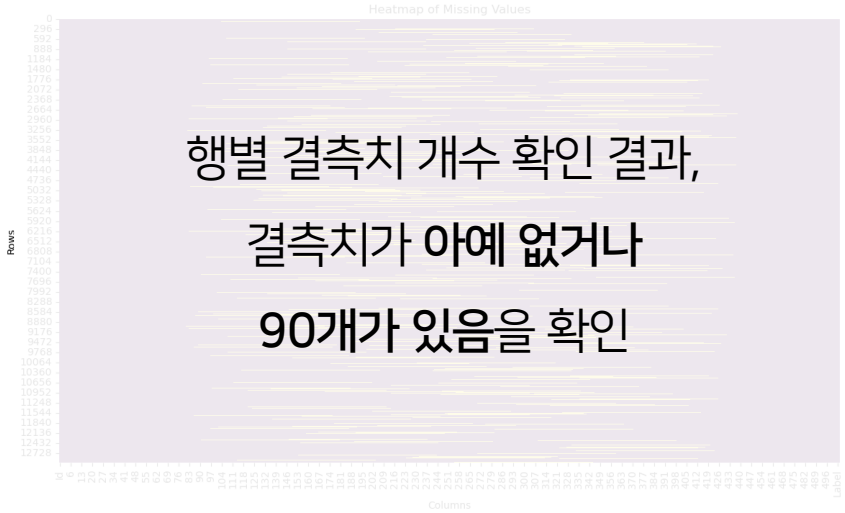


1 데이터 EDA

결측치 확인

결측치 파악을 위해 다양한 방법으로 결측치 그래프 시각화





행별 결측치 개수 확인 결과,

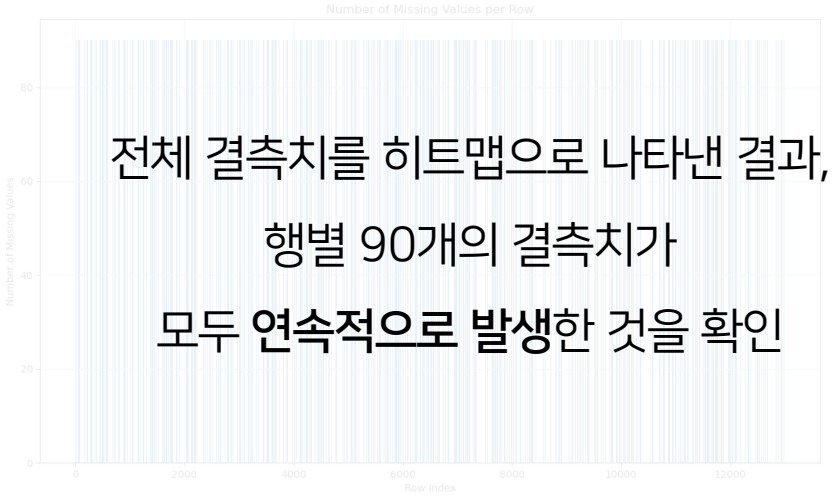
결측치가 **아예 없거나**

90개가 있음을 확인

1 데이터 EDA

결측치 확인

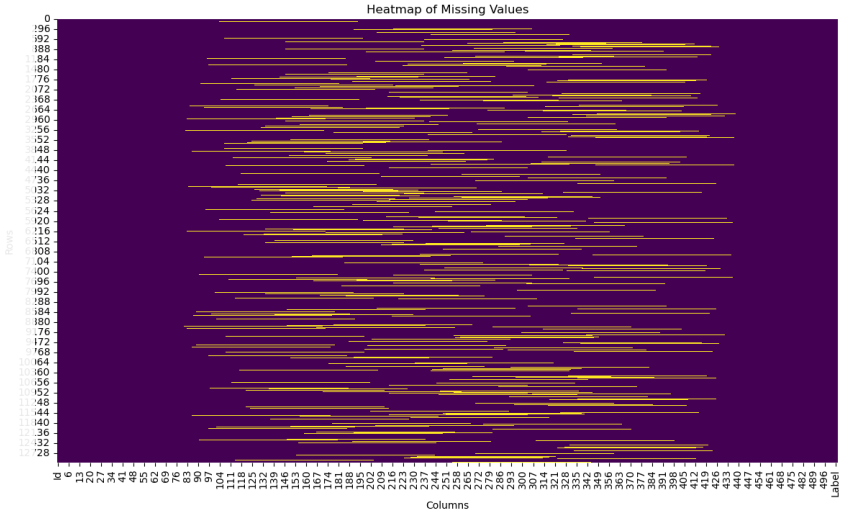
결측치 파악을 위해 다양한 방법으로 결측치 그래프 시각화



전체 결측치를 히트맵으로 나타낸 결과,

행별 90개의 결측치가

모두 **연속적으로 발생**한 것을 확인

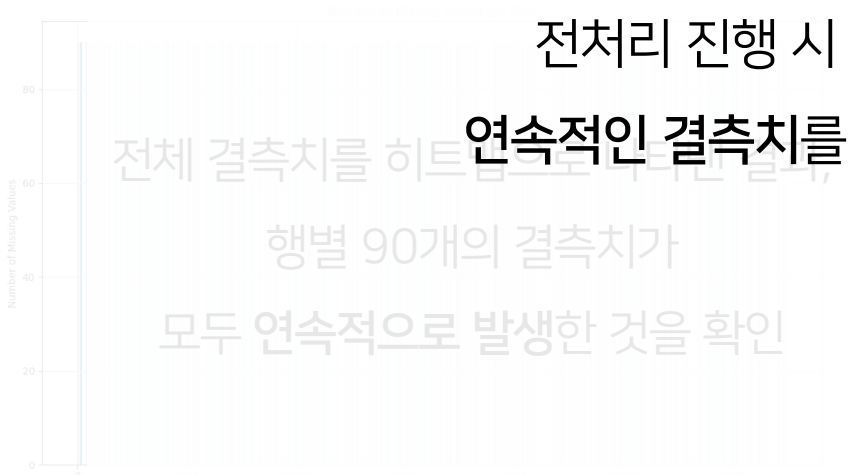


결측치 확인

결측치 파악을 위해 다양한 방법으로 결측치 그래프 시각화

전처리 진행 시 시계열 데이터 속

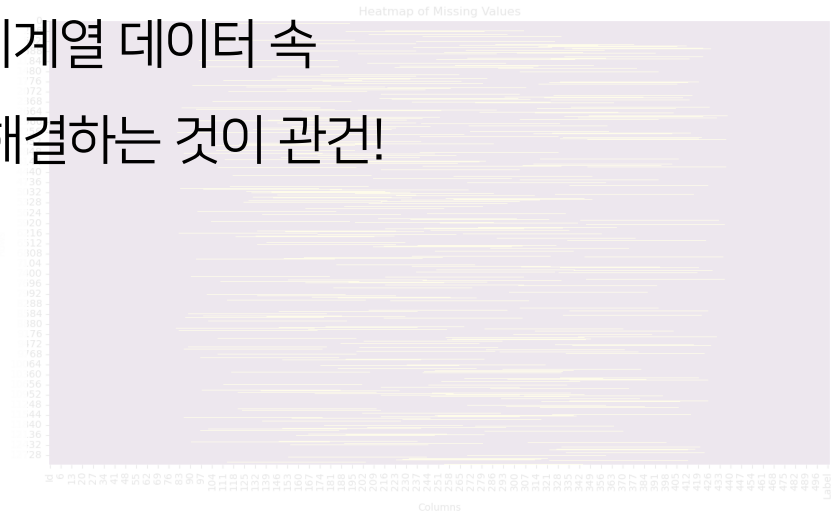
연속적인 결측치를 해결하는 것이 관건!



전체 결측치를 히트맵으로 나타낸 결과,

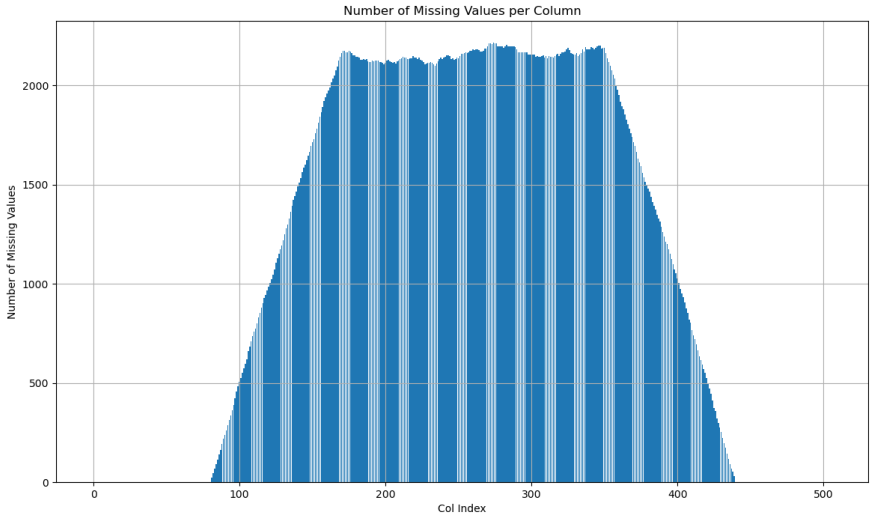
행별 90개의 결측치가

모두 **연속적으로 발생**한 것을 확인



결측치 확인

결측치 파악을 위해 다양한 방법으로 결측치 그래프 시각화



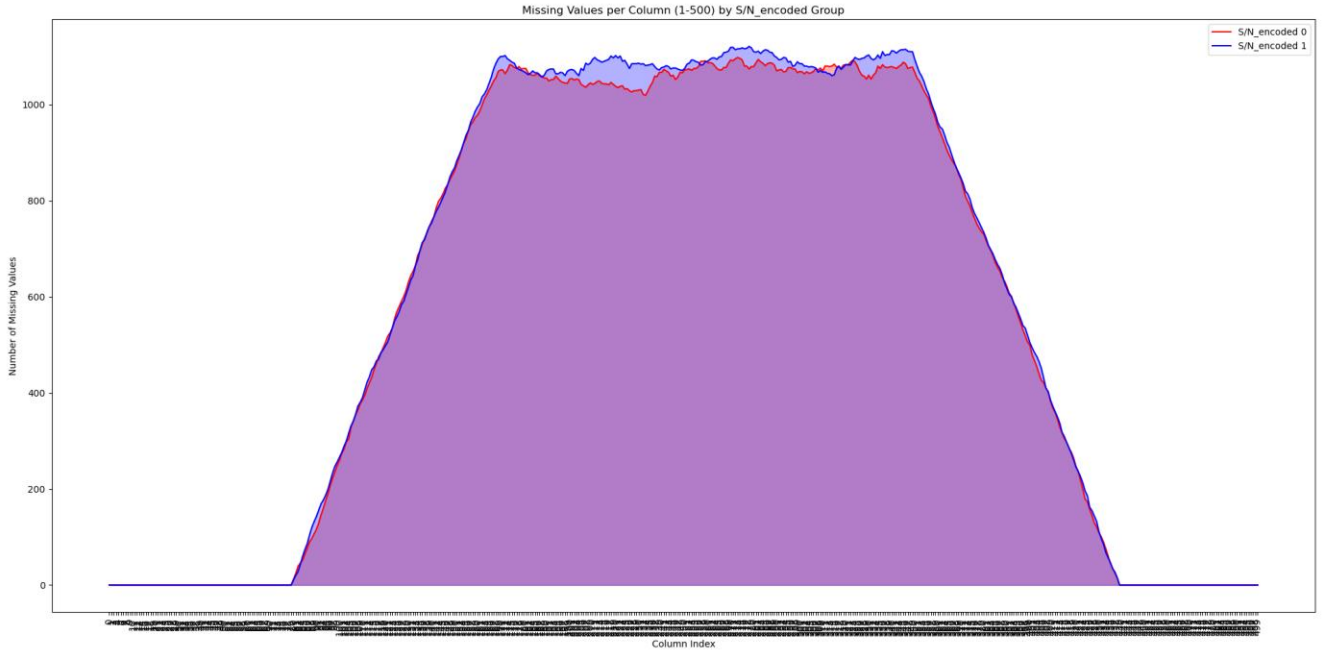
열별 결측치 개수 확인 결과, 결측치가 중간 시점에 많이 발생함을 확인

1 데이터 EDA

결측치 확인

제품그룹별, 국가별, 연대별, Label별 결측치의 형태도 확인해보았지만

그룹 크기에 따라 결측치 수가 다른 것 외에는 열별 결측치 수가 비슷한 것을 확인

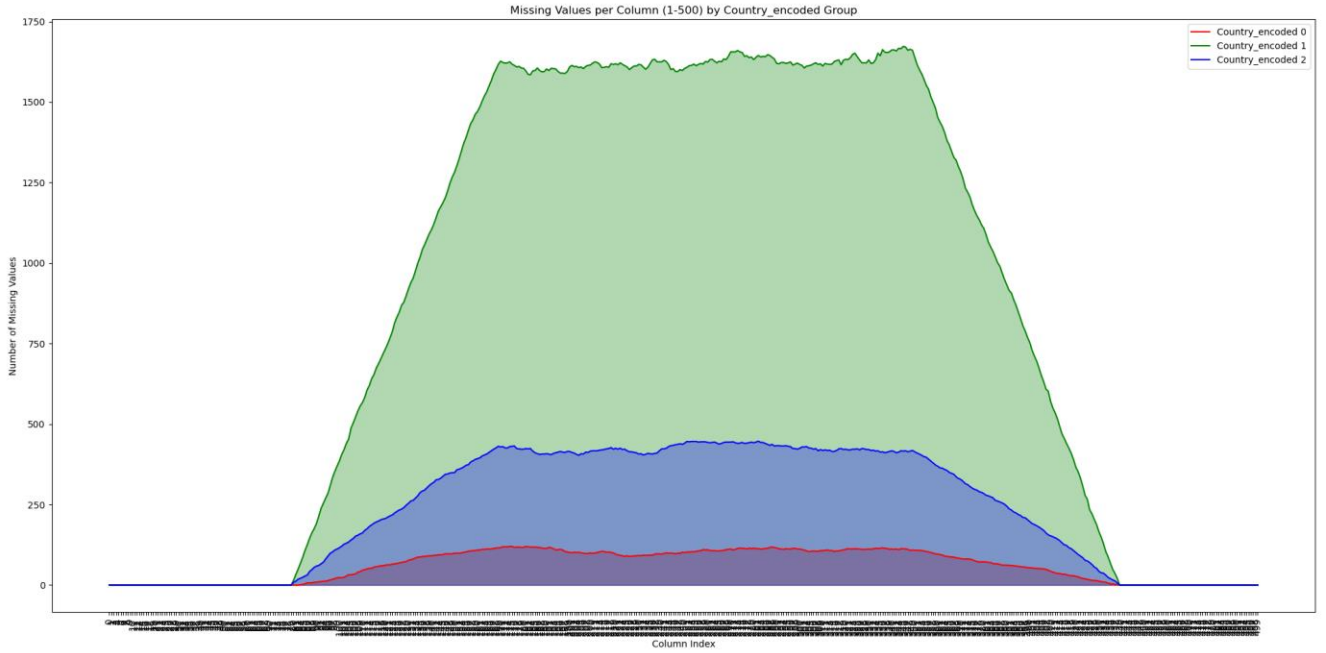


S/N 그룹에 따른 결측치 시각화

결측치 확인

제품그룹별, 국가별, 연대별, Label별 결측치의 형태도 확인해보았지만

그룹 크기에 따라 결측치 수가 다른 것 외에는 열별 결측치 수가 비슷한 것을 확인

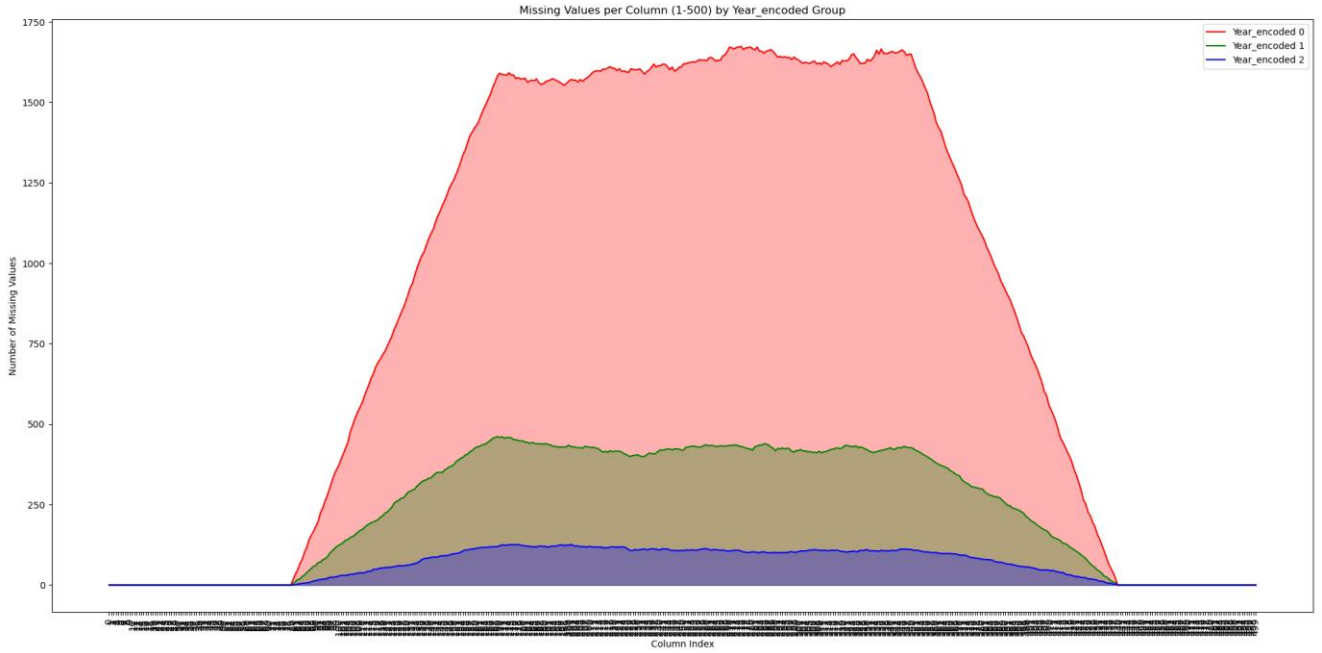


Country 그룹에 따른 결측치 시각화

결측치 확인

제품그룹별, 국가별, 연대별, Label별 결측치의 형태도 확인해보았지만

그룹 크기에 따라 결측치 수가 다른 것 외에는 열별 결측치 수가 비슷한 것을 확인

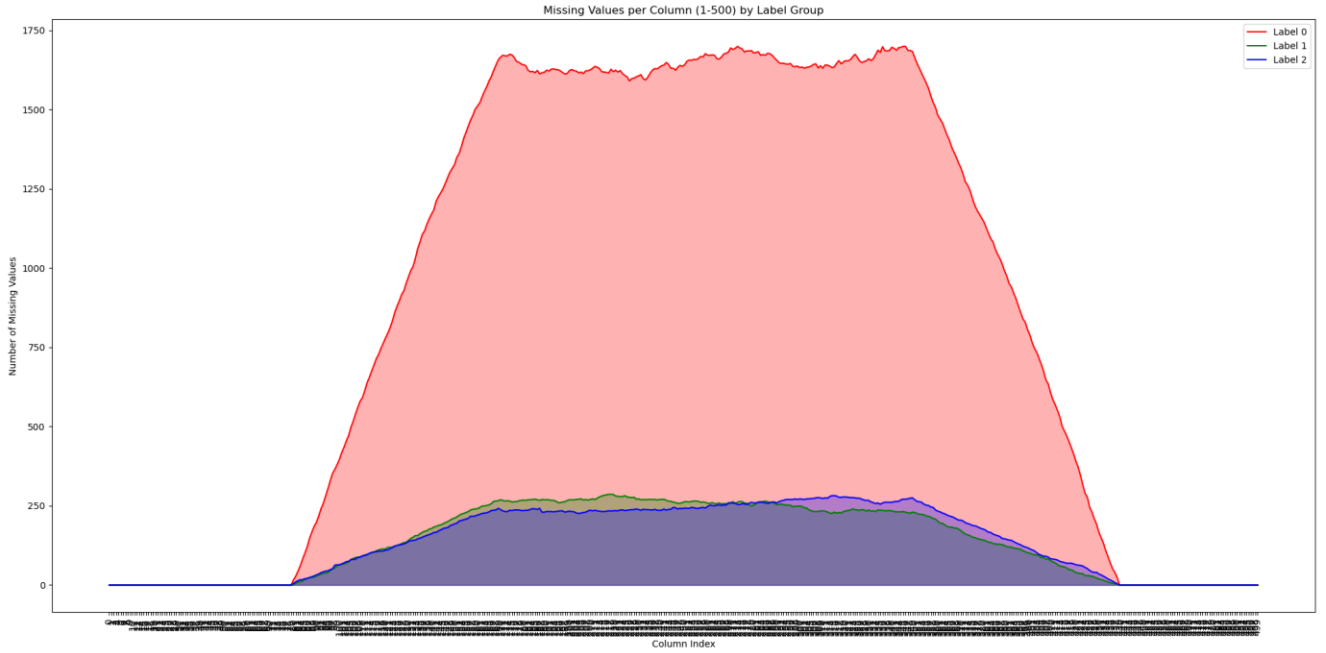


Year 그룹에 따른 결측치 시각화

결측치 확인

제품그룹별, 국가별, 연대별, Label별 결측치의 형태도 확인해보았지만

그룹 크기에 따라 결측치 수가 다른 것 외에는 열별 결측치 수가 비슷한 것을 확인



Label 그룹에 따른 결측치 시각화

1 데이터 EDA

데이터 형태 확인

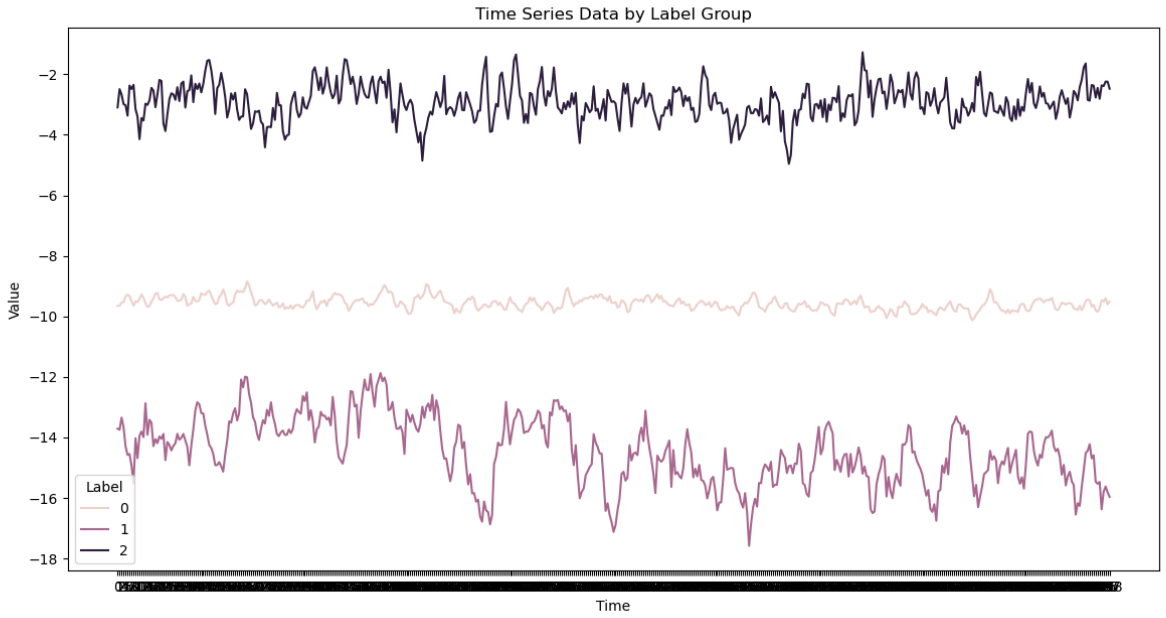
13,000개 행의 시계열 데이터의 특징을 파악하기 위해
그룹을 나누어 그래프 확인

1 데이터 EDA

데이터 형태 확인

시계열 데이터와 다른 열들의 연관성을 찾기 위해

그룹별로 시계열 데이터의 타임 스탬프별 평균을 구해 시각화



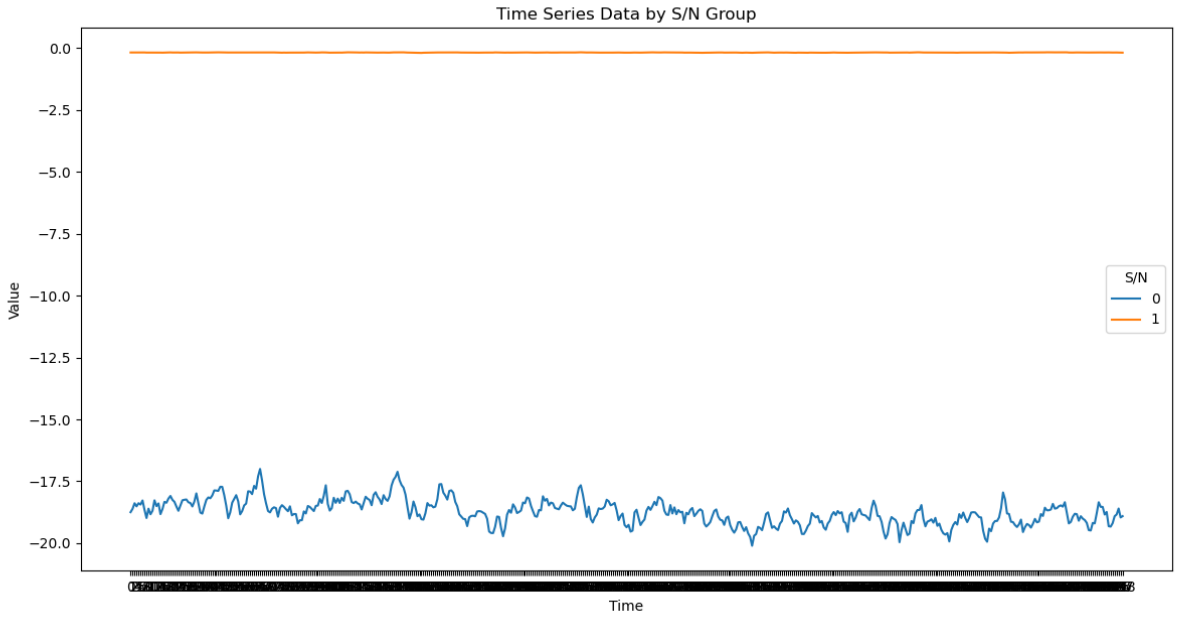
Label별 시계열 데이터의 **분포**와 **Data generating process**가 다름을 확인

1 데이터 EDA

데이터 형태 확인

시계열 데이터와 다른 열들의 연관성을 찾기 위해

그룹별로 시계열 데이터의 타임 스탬프별 평균을 구해 시각화



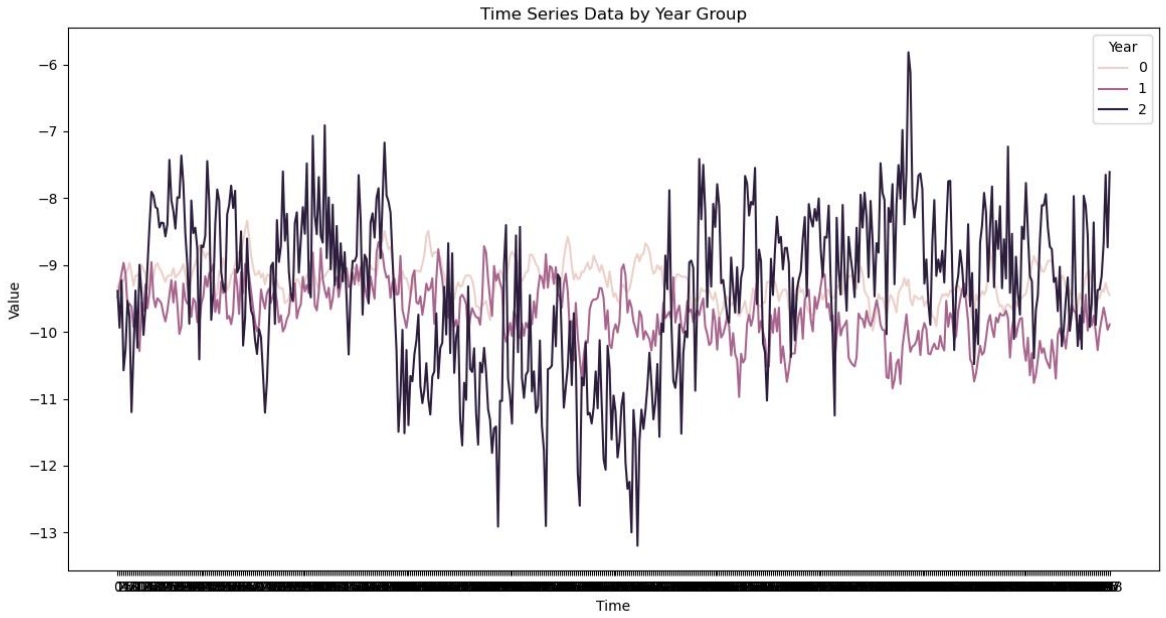
S/N 그룹별로도 데이터의 분포가 나뉨

1 데이터 EDA

데이터 형태 확인

시계열 데이터와 다른 열들의 연관성을 찾기 위해

그룹별로 시계열 데이터의 타임 스탬프별 평균을 구해 시각화



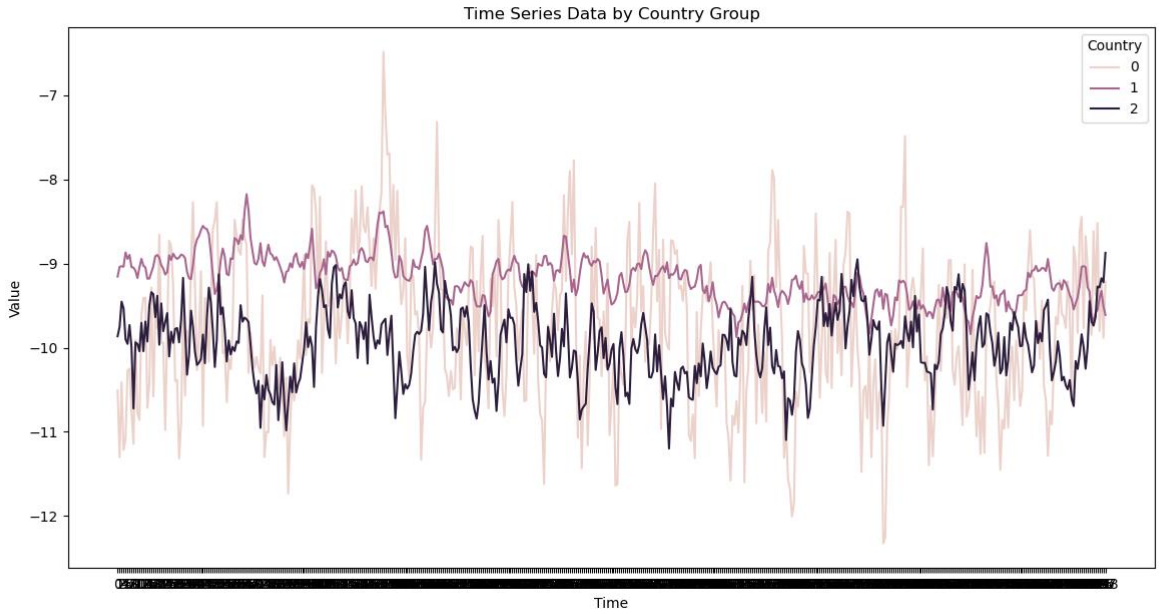
연대별, 국가별로 보았을 때, 평균은 비슷하지만 분산이 조금씩 다름

1 데이터 EDA

데이터 형태 확인

시계열 데이터와 다른 열들의 연관성을 찾기 위해

그룹별로 시계열 데이터의 타임 스탬프별 평균을 구해 시각화



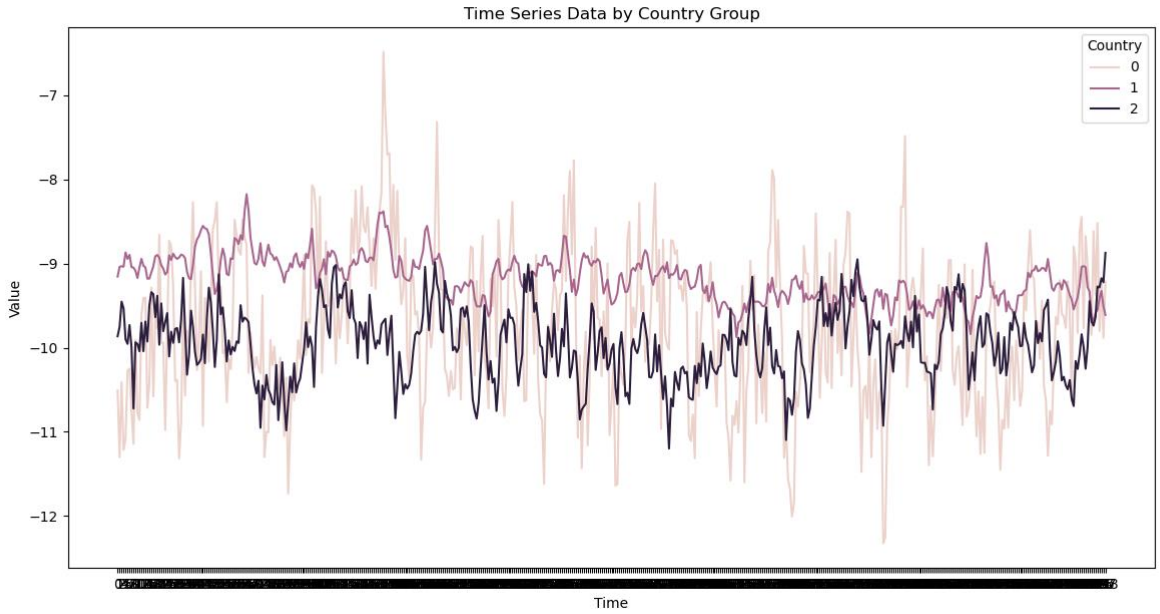
연대별, 국가별로 보았을 때, 평균은 비슷하지만 분산이 조금씩 다름

1 데이터 EDA

데이터 형태 확인

시계열 데이터와 다른 열들의 연관성을 찾기 위해

그룹별로 시계열 데이터의 타임 스탬프별 평균을 구해 시각화



연대별, 국가별로 보았을 때, 평균은 비슷하지만 분산이 조금씩 다름

2

데이터 전처리

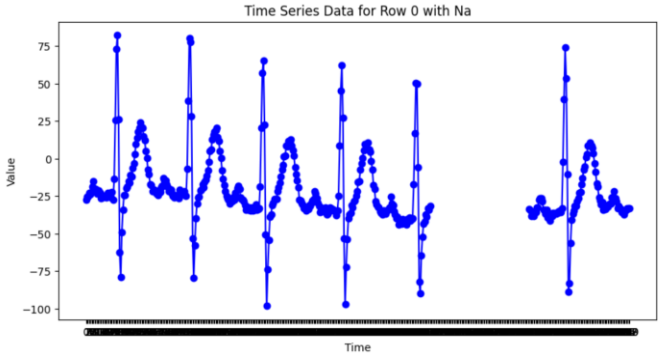
2 데이터 전처리

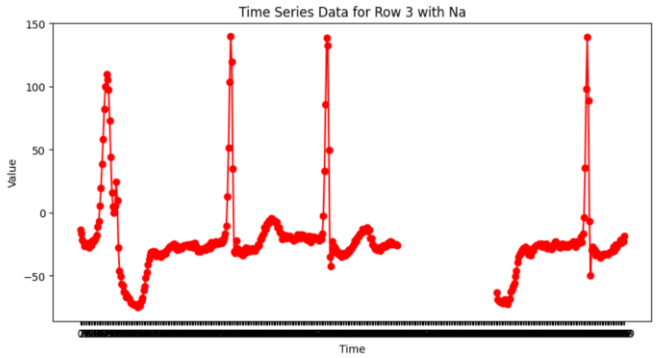
변수 추가 | 인코딩

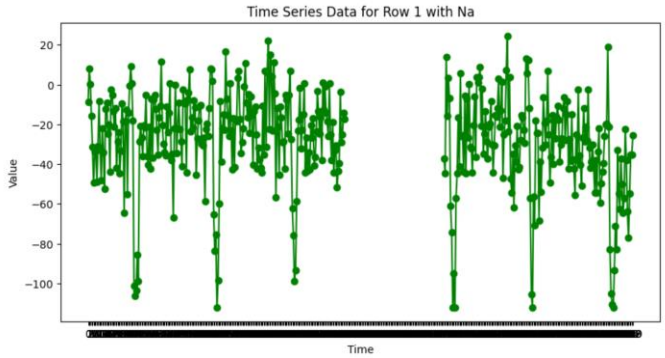
Country	Country가 **한국, 미국, 중국**으로 나뉨에 따라 0,1,2로 인코딩
Year	10년 단위로 데이터의 개수가 비슷하므로 1990s, 2000s, 2010s로 묶은 뒤 0, 1, 2로 인코딩
S/N	시작하는 코드 (PSCG, PSFT) 에 따라 0, 1로 인코딩

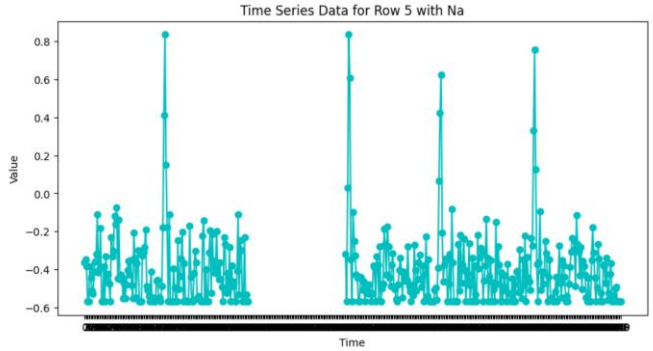
2 데이터 전처리

결측치 처리





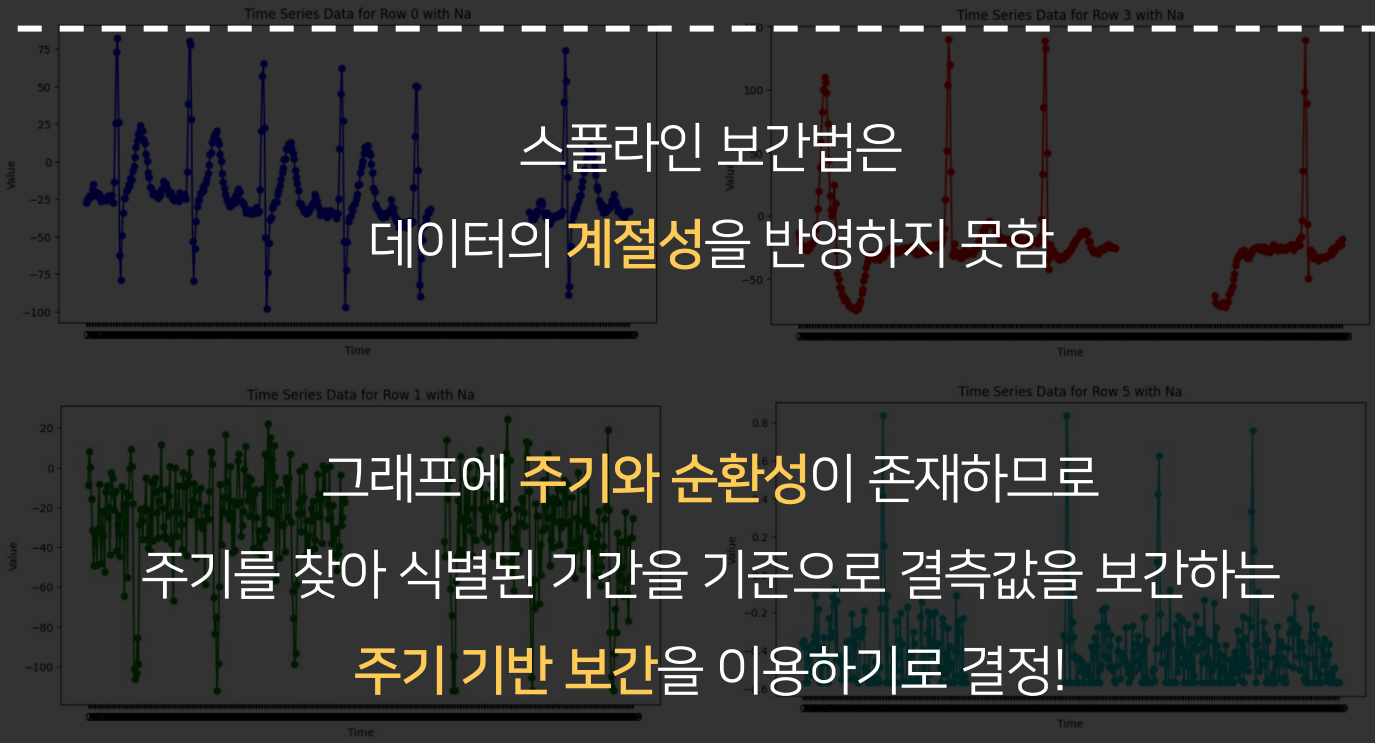




그래프가 뚜렷한 추세를 보이지 않아 선형 보간법 대신 **스플라인 보간법** 시도

2 데이터 전처리

결측치 처리



스플라인 보간법은

데이터의 **계절성**을 반영하지 못함

그래프에 **주기와 순환성**이 존재하므로

주기를 찾아 식별된 기간을 기준으로 결측값을 보간하는

주기 기반 보간을 이용하기로 결정!

그래프가 뚜렷한 추세를 보이지 않아 선형 보간법 대신 **스플라인 보간법** 시도

결측치 처리 | 주기 기반 보간

주기 기반 보간

시계열 데이터에서 결측치를 주기 기반으로 보간하는 기법

Seasonal smoothing과 비슷함!

주기 기반 보간 기법의 원리

입력 시계열 데이터를 복사하여 새로운 Series를 만듦

시계열 데이터의 각 요소를 순회하며 결측치인 경우

주기 기반으로 이전 값들을 찾아 **평균**을 구함

구한 평균값을 결측치 위치에 삽입

결측치 처리 | 주기 기반 보간 : 주기 추정

주기 기반 보간

시계열 데이터에서 결측치를 주기 기반으로 보간하는 기법

Seasonal smoothing과 비슷함!

주기 기반 보간 기법의 종류

주기는 어떻게 추정할 수 있을까?

1. 입력 시계열 데이터를 복사하여 새로운 Series를 만듦
2. 시계열 데이터의 각 요소를 순회하며 결측치인 경우
   주기 기반으로 이전 값들을 찾아 **평균**을 구함
3. 구한 평균값을 결측치 위치에 삽입

결측치 처리 | 주기 기반 보간 : 주기 추정

find_peaks()

주변 값들보다 큰 값을 peak로 간주하여 주기를 추정할 때 사용

parameter

distance = 100 으로 설정하여

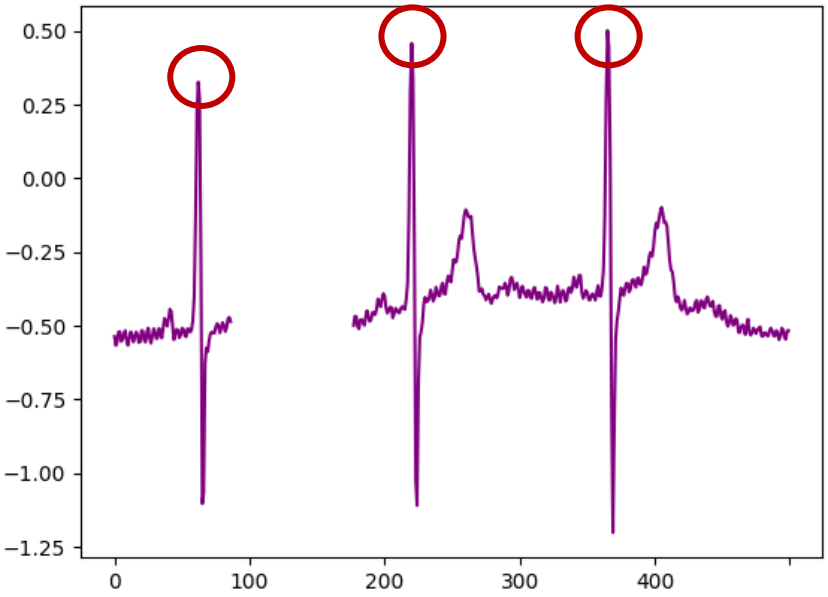
지나치게 많은 **local minimum**을 peak로 검출하는 것을 방지

distance = peak 간 최소거리

결측치 처리 | 주기 기반 보간 : 주기 추정

find_peaks()

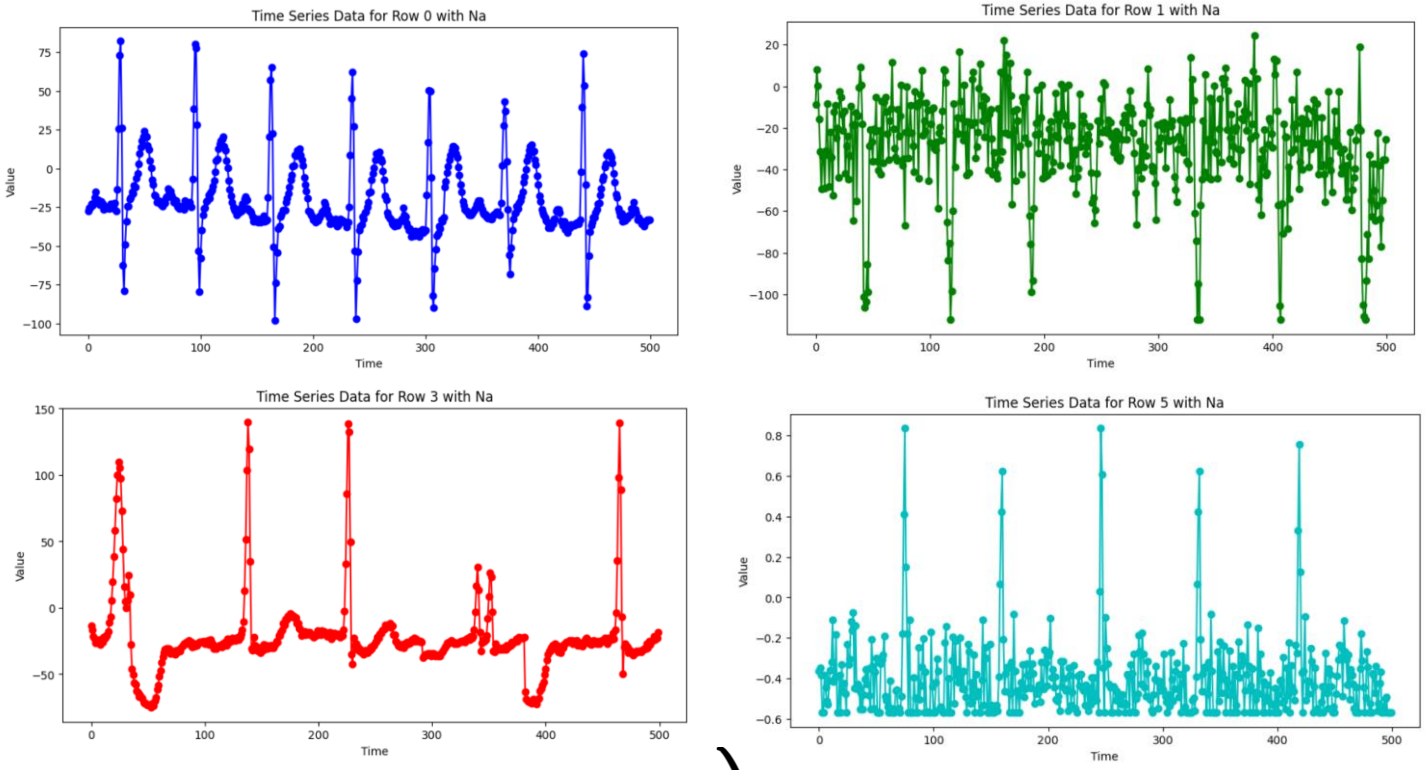
주변 값들보다 큰 값을 peak로 간주하여 주기를 추정할 때 사용



피크 간 간격들의 **median**을
주기(period)로 설정!

2 데이터 전처리

결측치 처리 | 주기 기반 보간



보간된 패턴이 직관과 부합!

이상치 처리 | 처리 과정

이상치가 포함된 데이터셋에서 **1차적으로 결측치 보간** (주기 기반)

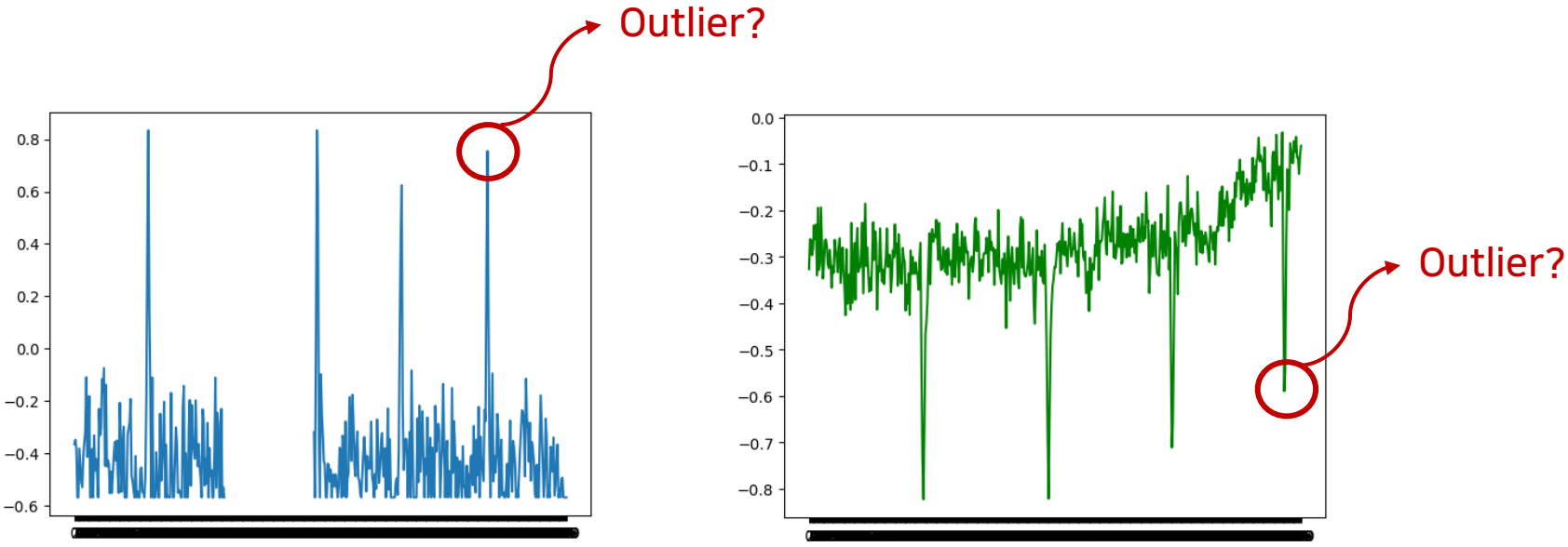
↓

결측치가 처리된 시계열 데이터(행) 별 **이상치 탐지**

↓

이상치를 Na 처리 후 처음과 같은 방법으로 **Na값 보간**

이상치 처리



시계열 데이터의 특성상 **추세**와 **계절성**이 있을 수 있으므로

단순히 크거나 낮은 수치를 **이상치로 간주해서는 안될 것**이라고 판단!

2 데이터 전처리

이상치 처리

시계열 분해를 통해 추세와 계절성을 제거해보자!

2 데이터 전처리

이상치 처리 | 시계열 분해

Classical decomposition

전체 기간에 걸친 데이터의 평균적 패턴을 기반으로

추세와 계절성을 추정하여 이상치에 민감

STL decomposition

구간을 나누어 반복적으로 추세와 계절성을 추정하므로

상대적으로 이상치에 robust함

자세한 내용은 CV팀 주제분석 3주차 PPT참고!

2 데이터 전처리

이상치 처리 | 시계열 분해

주어진 데이터는 주기가 다양하므로

STL decomposition을 사용하기로 결정!

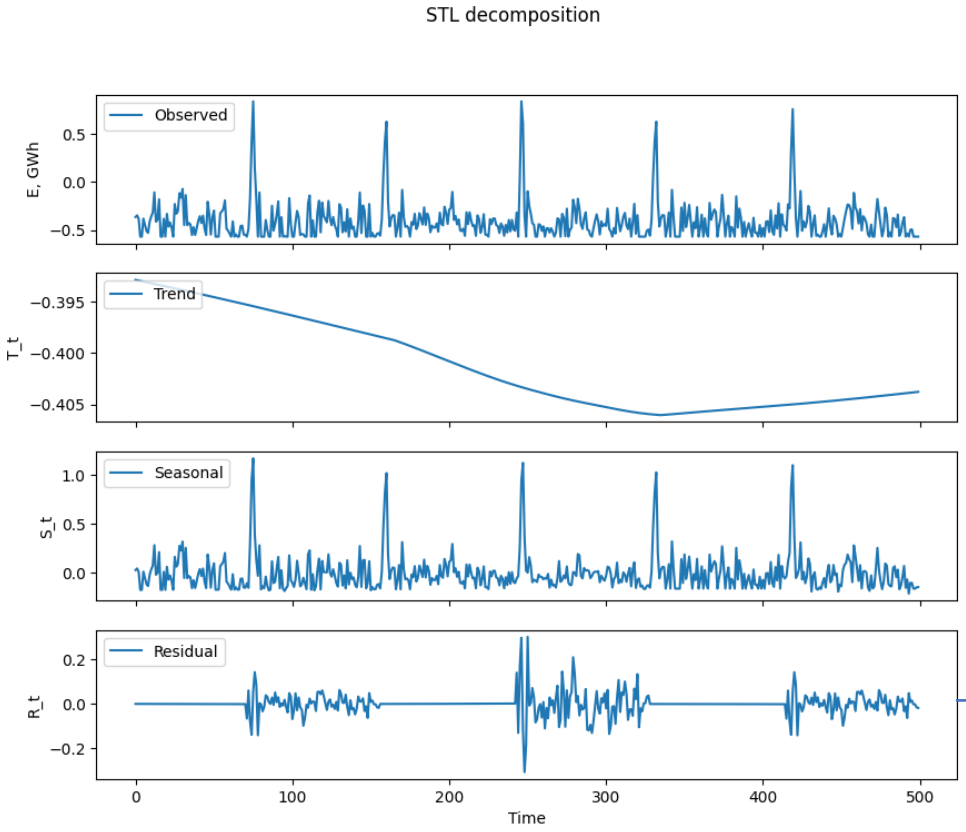
STL decomposition

구간을 나누어 반복적으로 추세와 계절성을 추정하므로

상대적으로 이상치에 robust함

자세한 내용은 CV팀 주제분석 3주차 PPT참고!

이상치 처리 | STL decomposition



$$Z = \frac{(x - \mu)}{\sigma}$$

Residual의 Z값을 구한 후

임계값보다 큰 것을 이상치로 간주

2 데이터 전처리

이상치 처리 | STL decomposition

일반적으로 사용되는 임계값인 2~3을 사용한 결과

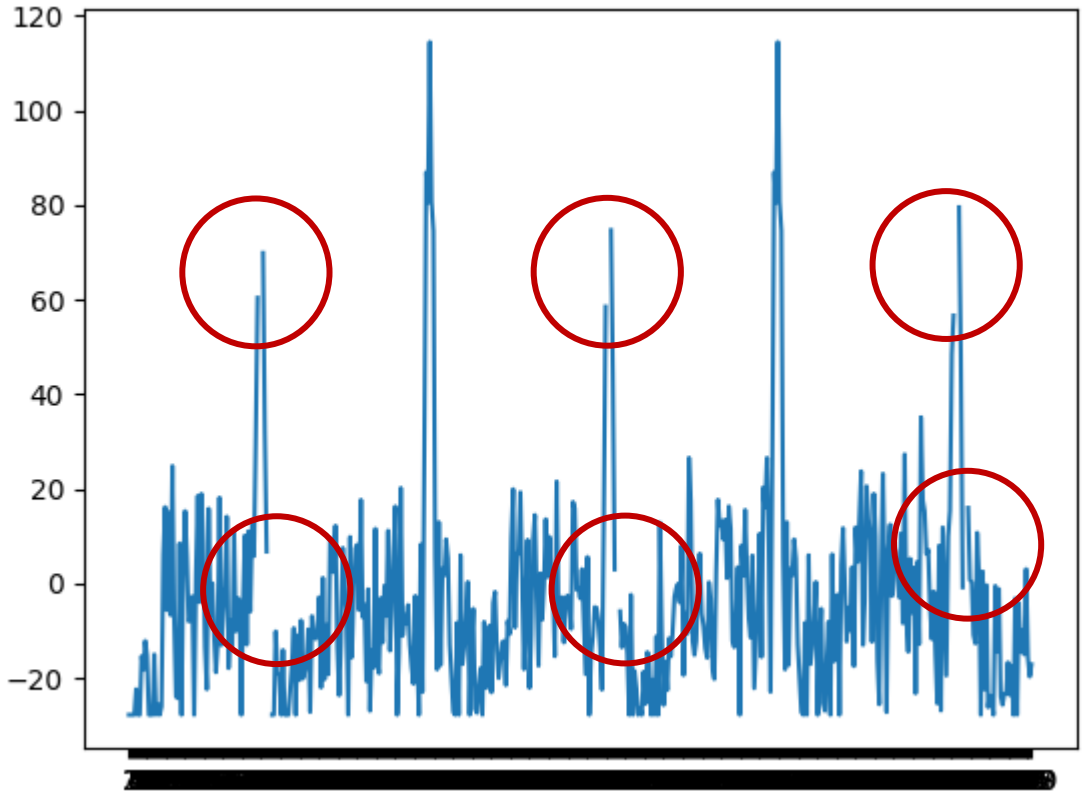
지나치게 많은 값들이 이상치로 분류되어 주기 보간을 할 수 없고,

큰 값을 사용하게 되면 기존 데이터와 차이가 거의 없음

따라서 기존 데이터를 그대로 사용하기로 결정

2 데이터 전처리

이상치 처리 | STL decomposition



Index(
['73', '74',
'78', '79',
'265', '266',
'270', '271',
'457', '458',
'462', '463'])

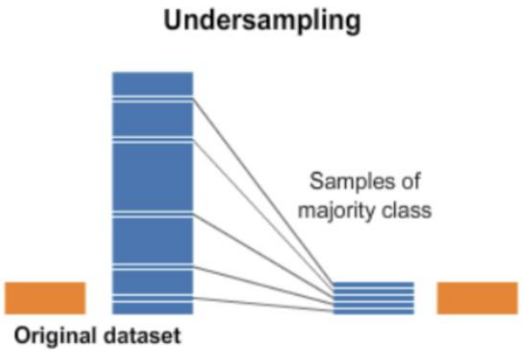
일정한 간격으로 이상치가 탐지되어 보간 실패..

2 데이터 전처리

데이터 불균형 처리

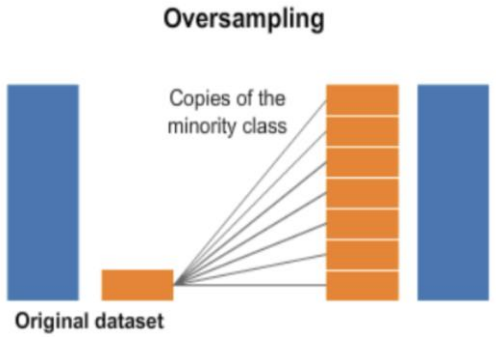
데이터 불균형 처리 방법

Undersampling



데이터의 소실이 매우 크고,
중요한 데이터를 잃을 수 있다는 위험이 있음

Oversampling



소수 데이터를 복제하여 데이터 불균형을
해결하지만, 오버 피팅의 위험이 있음

2 데이터 전처리

데이터 불균형 처리

추후 복잡한 모델을 사용하기 위해

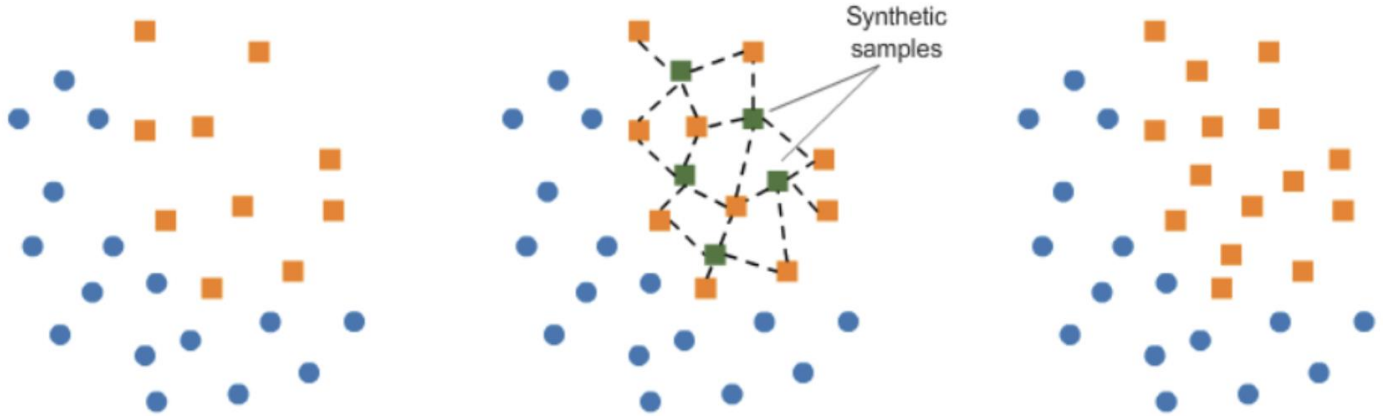
많은 데이터 확보가 데이터 분석에 더 효과적이라고 판단

SMOTE 기법을 활용한 Oversampling을 하기로 결정 !

데이터 불균형 처리

SMOTE 기법

낮은 비율 클래스 데이터들의 최근접 이웃을 이용하여 새로운 데이터 생성



완전히 똑같은 데이터를 복제하는 것은 학습에 의미가 없기 때문에

근접한 데이터로 데이터 생성

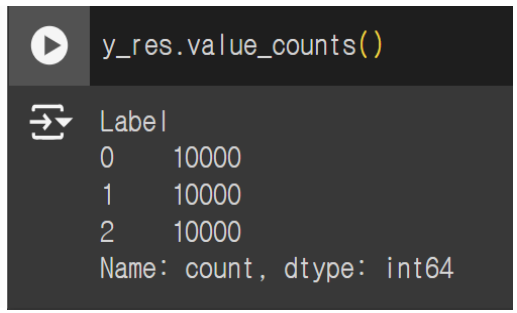
데이터 불균형 처리

SMOTE 기법

낮은 비율 클래스 데이터들의 최근접 이웃을 이용하여 새로운 데이터 생성

```
[26] # Upsampling by SMOTE
     from imblearn.over_sampling import SMOTE
     SEED = 1234

     smote = SMOTE(random_state=SEED)

     X_res, y_res = smote.fit_resample(X, y)

[27] y.value_counts()

     Label
     0    10000
     1     1500
     2     1500
     Name: count, dtype: int64

     y_res.value_counts()

     Label
     0    10000
     1    10000
     2    10000
     Name: count, dtype: int64
```

Label 0, Label 1, Label 2의 클래스 불균형이 해소되었음을 알 수 있음



2 데이터 전처리

데이터 불균형 처리

SMOTE 기법

낮은 비율 클래스 데이터들의 최근접 이웃을 이용하여 새로운 데이터 생성

그러나, 우리가 다루는 특징이 매우 고차원적이고 변수간의 거리가 멀어

SMOTE 기법 이용 시 **데이터의 왜곡**이 클 것이라 판단

Label 0, Label 1, Label 2의 클래스

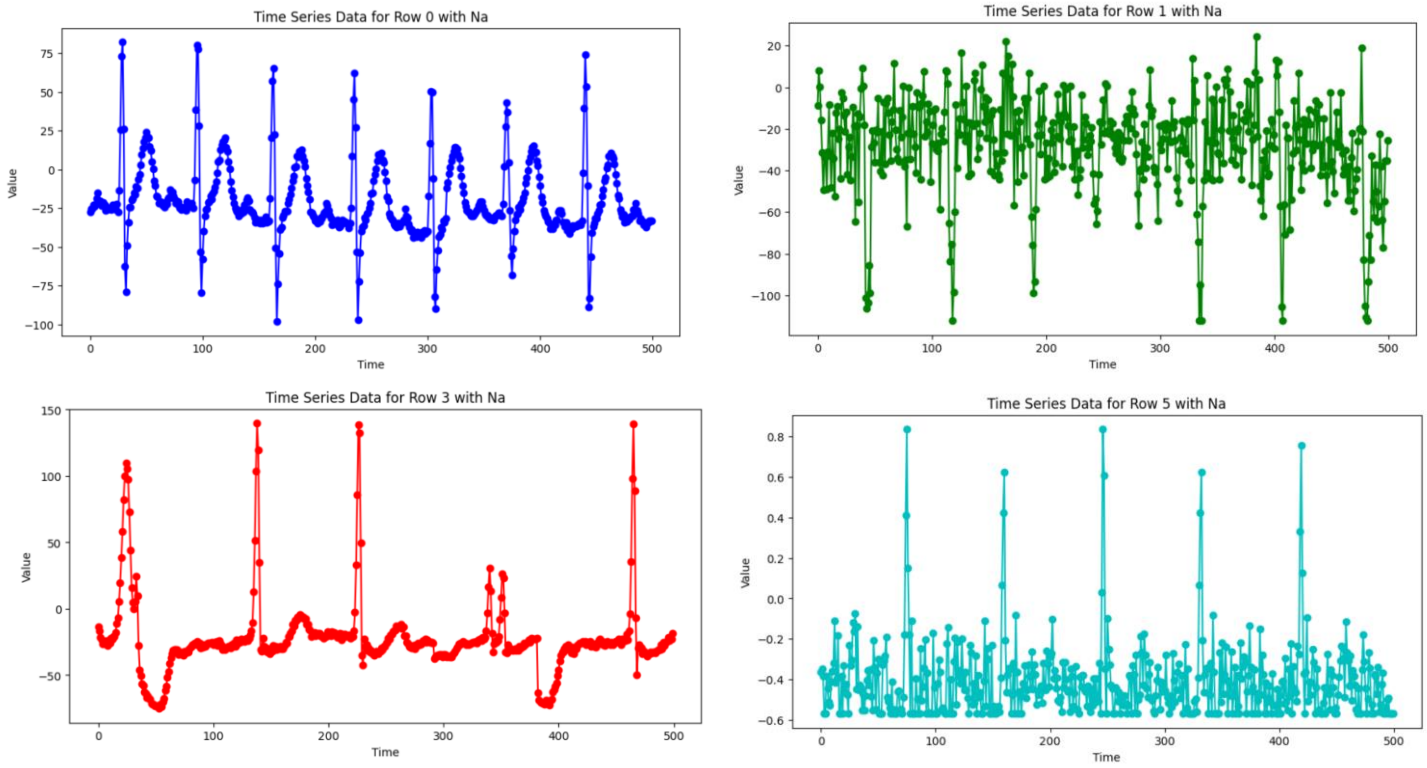
따라서 불균형한 클래스의 샘플 비율에 따라서 [illegible]었음을 알 수 있음

자동으로 **가중치 조정**을 하는 방법과 비교 후 결정하기로 함

```
[26] # Upsampling by SMOTE
     from imblearn.over_sampling import SMOTE
     SEED = 1234

     smote = SMOTE(random_state=SEED)

     X_res, y_res = smote.fit_resample(X, y)

[27] y.value_counts()

     Label
     0    10000
     1    15000
     2    15000
     Name: count, dtype: int64

     y_res.value_counts()

     Label
     0    10000
     1    10000
     2    10000
     Name: count, dtype: int64
```

```
y_res.value_counts()

Label
0    10000
1    10000
2    10000
Name: count, dtype: int64
```

변수 변환 | 특징 추출



모든 시계열 데이터는 고유한 Data generating process를 가짐

2 데이터 전처리

변수 변환 | 특징 추출

EDA 파트 참조!

클래스별로 시계열의 Data generating process에

유의미한 차이가 있다고 판단됨

최댓값, 최솟값, 계절성, 주기, 추세 등

Process가 갖는 **고유한 특징들**을

클래스 예측을 위한 유용한 **설명변수로 사용**할 수 있겠다고 판단!

변수 변환 | 특징 추출

Tsfresh

시계열 데이터를 분석하고 다양한 통계적 특징을 자동으로 추출하는 라이브러리

feature_extract()

기초 통계량, 자기 상관, 변동성,
주파수 특성 등 **다양한 특징을 추출**

추출되는 특징의 종류는 고정됨

select_features()

반응변수와 통계적으로 유의미한
관계가 있는 특징들을 선별

설명변수의 수를 줄이는 데 사용

변수 변환 | 특징 추출

Tsfresh

시계열 데이터를 분석하고 다양한 통계적 특징을 자동으로 추출하는 라이브러리

feature_extract()

기초 통계량, 자기 상관, 변동성,
주파수 특성 등 **다양한 특징을 추출**

추출되는 특징의 종류는 고정됨

select_features()

반응변수와 통계적으로 유의미한
관계가 있는 특징들을 선별

설명변수의 수를 줄이는 데 사용

변수 변환 | 특징 추출

Tsfresh

시계열 데이터를 분석하고 다양한 통계적 특징을 자동으로 추출하는 라이브러리

feature_extract()

기초 통계량, 자기 상관, 변동성,
주파수 특성 등 **다양한 특징을 추출**

추출되는 특징의 종류는 고정됨

select_features()

반응변수와 통계적으로 유의미한
관계가 있는 특징들을 선별

설명변수의 수를 줄이는 데 사용

2 데이터 전처리

변수 변환 | 특징 추출

13,000 X 500

Index	0	1	2	3	4	…	498	499
0	-27.419	-25.272	-25.474	-22.805	-24.078	…	-33.125	-33.406
1	-8.827	8.234	0.381	-15.821	-31.289	…	-35.418	-25.590
2	-0.073	-0.063	-0.082	-0.091	-0.078	…	-0.248	-0.239

Train 시계열 데이터

feature_extract()

13,000 X 777

	value_variance_larger_than_standard_deviation	value_mean_second_derivative_central	value_median	…
0	1.0	-0.028732	-25.922000	…
1	1.0	-0.051387	-25.019465	…
2	0.0	-0.000014	-0.197420	…

추출된 feature

13,000개의 시계열 sample path에 대하여 특징 추출!

변수 변환 | 특징 추출

추출된 feature의 예

“value__linear_trend__attr_stderr”	시계열 데이터의 선형 추세와 그 표준 오차
“value__mean_abs_change”	시계열 데이터의 연속된 값들 사이의 평균 절대 변화량
“value__variance”	시계열 데이터의 분산

변수 변환 | 특징 추출

추출된 feature의 예

“value__autocorrelation__lag_5”	Lag 5에 대한 자기상관
“value__linear_trend__attr_pvalue”	시계열 데이터의 선형 추세의 p-value

3

모델링

3 모델링

모델 후보군

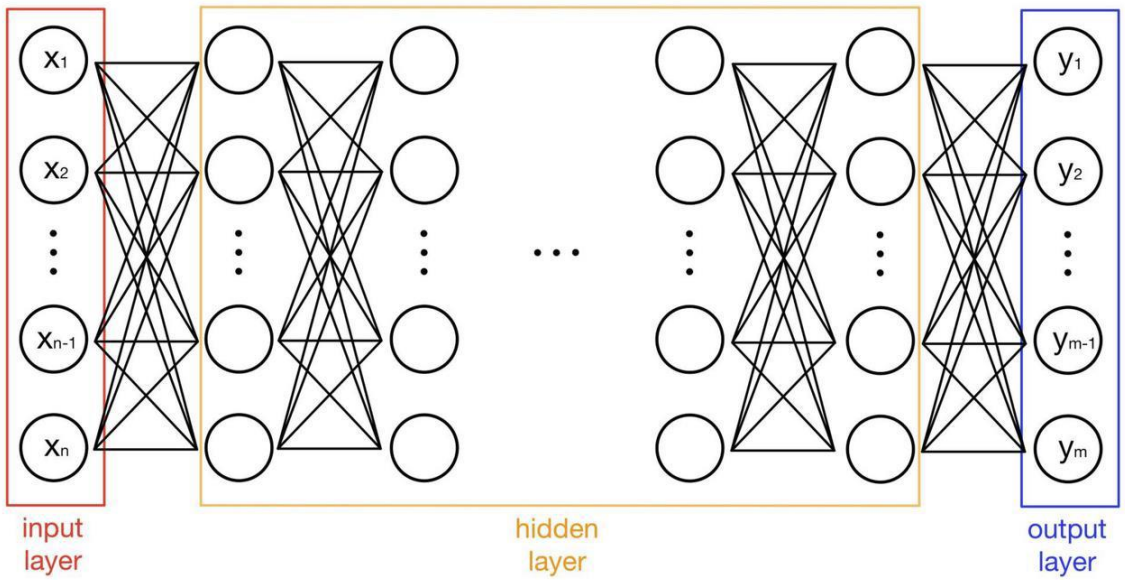
Random Forest	XGBoost	MLP
KNN Classifier	LightGBM	LSTM
Logistic Regression	Ensemble Models	

고차원 & 비선형 관계를 반영할 수 있는 모델들을 추려 시도함

모델 후보군 | NNs

MLP(Multi-Layer Perceptron)

퍼셉트론을 여러 층으로 쌓은 순방향의 신경망

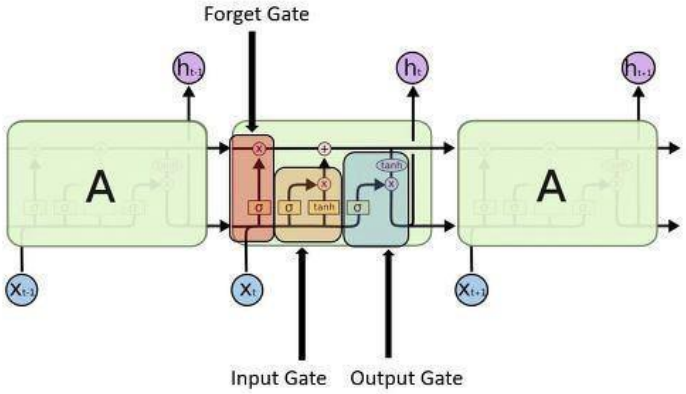


모델 후보군 | NNs

LSTM

단기 기억, 장기 기억 두 개의 hidden vector를 사용해

은닉층의 과거 정보가 마지막까지 전달될 수 있도록 하는 RNN 기반 모델



시계열 데이터의 패턴 학습 및 장기 의존성 학습에 용이

3 모델링

모델 후보군 | NNs

RNN

정형 데이터에 대하여

트리 모델이 딥러닝 모델보다 우수한 성능을 보임

Score: 4---

Score: 4

떠오르는 논문 스터디의 기억...

여러 시도를 해 보았으나 단순 MLP의 성능이 훨씬 좋았음

특징 추출한 데이터셋은 순수 정형 데이터

뭔가 더 나은 방법을
고민하는 짤...

XGBoost 시도!

3 모델링

모델 후보군 | Gradient Boosting Trees

XGBoost

병렬 처리와 정규화 기법을 사용하여 성능을 극대화한 모델

2015 Kaggle Winning solution

17/29 Used XGBoost

11/29 Used DeepNeural Nets

2015 KDDcup

Top 10 all used XGBoost

3 모델링

모델 선정 과정

<table>
<tr><th rowspan="2">사용 모델</th><th colspan="3">첨가한 기법</th><th rowspan="2">성능</th></tr>
<tr><th>특징 추출</th><th>불균형 처리</th><th>스케일링</th></tr>
<tr><td>LSTM</td><td rowspan="2"></td><td rowspan="3"></td><td rowspan="6">Standard</td><td>1201</td></tr>
<tr><td rowspan="3">MLP</td><td>1764</td></tr>
<tr><td rowspan="2">특징 추출 (777)</td><td>4399</td></tr>
<tr><td rowspan="3">SMOTE</td><td>4411</td></tr>
<tr><td rowspan="2">특징 선별 (504)</td><td>4573</td></tr>
<tr><td rowspan="2">XGBoost</td><td>4742</td></tr>
<tr><td rowspan="2">Weight</td><td rowspan="2">Robust</td><td>4791</td></tr>
<tr><td>특징 선별 (254)</td><td>4790</td></tr>
</table>

사용한 모델과 적용한 기법을 시도해본 순서대로 정리해보자!

3 모델링

모델 선정 과정

사용 모델	첨가한 기법			성능
	특징 추출	불균형 처리	스케일링	
LSTM	X	X	Standard	1201
MLP	X	X	Standard	1764
MLP	특징 추출 (777)	X	Standard	4399
MLP	특징 추출 (777)	SMOTE	Standard	4411
MLP	특징 선별 (504)	SMOTE	Standard	4573
XGBoost	특징 선별 (504)	SMOTE	Standard	4742
XGBoost	특징 선별 (504)	Weight	Robust	4791
XGBoost	특징 선별 (254)	Weight	Robust	4790

시계열 데이터이므로 LSTM을 사용했으나 MLP에 비해 저조한 성적을 기록함

3 모델링

모델 선정 과정

사용 모델	첨가한 기법			성능
	특징 추출	불균형 처리	스케일링	
LSTM	X	X	Standard	1201
MLP	X	X	Standard	1764
MLP	특징 추출 (777)	X	Standard	4399
MLP	특징 추출 (777)	SMOTE	Standard	4411
MLP	특징 선별 (504)	SMOTE	Standard	4573
XGBoost	특징 선별 (504)	SMOTE	Standard	4742
XGBoost	특징 선별 (504)	Weight	Robust	4791
XGBoost	특징 선별 (254)	Weight	Robust	4790

특징 추출을 하여 시계열 데이터가 아닌 평범한 정형 데이터셋으로 바꾼 후
MLP를 적용했더니 성능이 대폭 상승!

3 모델링

모델 선정 과정

사용 모델	첨가한 기법			성능
	특징 추출	불균형 처리	스케일링	
LSTM	X	X	Standard	1201
MLP	X	X	Standard	1764
MLP	특징 추출 (777)	X	Standard	4399
MLP	특징 추출 (777)	SMOTE	Standard	4411
MLP	특징 선별 (504)	SMOTE	Standard	4573
XGBoost	특징 선별 (504)	SMOTE	Standard	4742
XGBoost	특징 선별 (504)	Weight	Robust	4791
XGBoost	특징 선별 (254)	Weight	Robust	4790

SMOTE로 불균형 문제를 추가로 처리했을 때 성능이 소폭 상승함

3 모델링

모델 선정 과정

사용 모델	첨가한 기법			성능
	특징 추출	불균형 처리	스케일링	
LSTM	X	X	Standard	1201
MLP	X	X	Standard	1764
MLP	특징 추출 (777)	X	Standard	4399
MLP	특징 추출 (777)	SMOTE	Standard	4411
MLP	특징 선별 (504)	SMOTE	Standard	4573
XGBoost	특징 선별 (504)	SMOTE	Standard	4742
XGBoost	특징 선별 (504)	Weight	Robust	4791
XGBoost	특징 선별 (254)	Weight	Robust	4790

선별 함수를 적용하여 특징의 수를 줄이고 데이터셋의 차원을
700여개에서 500여개로 줄이는 것 또한 성능 개선에 기여함

3 모델링

모델 선정 과정

사용 모델	첨가한 기법			성능
	특징 추출	불균형 처리	스케일링	
LSTM	X	X	Standard	1201
MLP	X	X	Standard	1764
MLP	특징 추출 (777)	X	Standard	4399
MLP	특징 추출 (777)	SMOTE	Standard	4411
MLP	특징 선별 (504)	SMOTE	Standard	4573
XGBoost	특징 선별 (504)	SMOTE	Standard	4742
XGBoost	특징 선별 (504)	Weight	Robust	4791
XGBoost	특징 선별 (254)	Weight	Robust	4790

MLP보다 정형 데이터에 더 적합한 XGBoost 모델 사용시 더 좋은 결과를 냄

3 모델링

모델 선정 과정

사용 모델	첨가한 기법			성능
	특징 추출	불균형 처리	스케일링	
LSTM	X	X	Standard	1201
MLP	X	X	Standard	1764
MLP	특징 추출 (777)	X	Standard	4399
MLP	특징 추출 (777)	SMOTE	Standard	4411
MLP	특징 선별 (504)	SMOTE	Standard	4573
XGBoost	특징 선별 (504)	SMOTE	Standard	4742
XGBoost	특징 선별 (504)	Weight	Robust	4791
XGBoost	특징 선별 (254)	Weight	Robust	4790

데이터셋 내 클래스의 비율을 고려해 각 샘플에 가중치를 고려하여
학습을 진행하는 방식으로 불균형 처리 전략을 변경

3 모델링

모델 선정 과정

사용 모델	첨가한 기법			성능
	특징 추출	불균형 처리	스케일링	
LSTM	X	X	Standard	1201
MLP	X	X	Standard	1764
MLP	특징 추출 (777)	X	Standard	4399
MLP	특징 추출 (777)	SMOTE	Standard	4411
MLP	특징 선별 (504)	SMOTE	Standard	4573
XGBoost	특징 선별 (504)	SMOTE	Standard	4742
XGBoost	특징 선별 (504)	Weight	Robust	4791
XGBoost	특징 선별 (254)	Weight	Robust	4790

이후 이상치에 Robust한 Robust 스케일러를 적용 후 성능 개선을 확인

3 모델링

모델 선정 과정

사용 모델	첨가한 기법			성능
	특징 추출	불균형 처리	스케일링	
LSTM	X	X	Standard	1201
MLP	X	X	Standard	1764
MLP	특징 추출 (777)	X	Standard	4399
MLP	특징 추출 (777)	SMOTE	Standard	4411
MLP	특징 선별 (504)	SMOTE	Standard	4573
XGBoost	특징 선별 (504)	SMOTE	Standard	4742
XGBoost	특징 선별 (504)	Weight	Robust	4791
XGBoost	특징 선별 (254)	Weight	Robust	4790

특징 선별 함수의 파라미터를 다중 분류 목적에 맞게 재조정 함

더 적은 수의 특징들을 선별해내어 데이터셋의 차원을 줄임

4

최종 모델

4 최종 모델

최종 데이터셋

전처리 이후,
Feature extract를
한 데이터셋에서
주요 변수만 선별한 데이터셋

+

Country, S/N, Year
인코딩 변수 데이터셋

4 최종 모델

최종 데이터셋

수치형 변수에 대해서만 Robust Scaling 적용

```
scaler = RobustScaler()
x_train_scale = scaler.fit_transform(X[continuous_features])
x_test_scale = scaler.transform(t[continuous_features])

X_continuous_df = pd.DataFrame(x_train_scale, index=X.index, columns=continuous_features)
x_test_continuous_df = pd.DataFrame(x_test_scale, index=t.index, columns=continuous_features)

train_set = pd.concat([X[categorical_features], X_continuous_df], axis=1)
test_set = pd.concat([t[categorical_features], x_test_continuous_df], axis=1)
```

⋮

Data Leakage를 방지하기 위해

train set에 scaler를 fit하여 test set에 적용한 이후,

범주형 변수와 수치형 변수를 합쳐 최종 데이터셋을 생성

4 최종 모델

최종 모델 | XGBoostClassifier with Sample Weight

모델링 과정에서 선별된 XGBoostClassifier 모델에

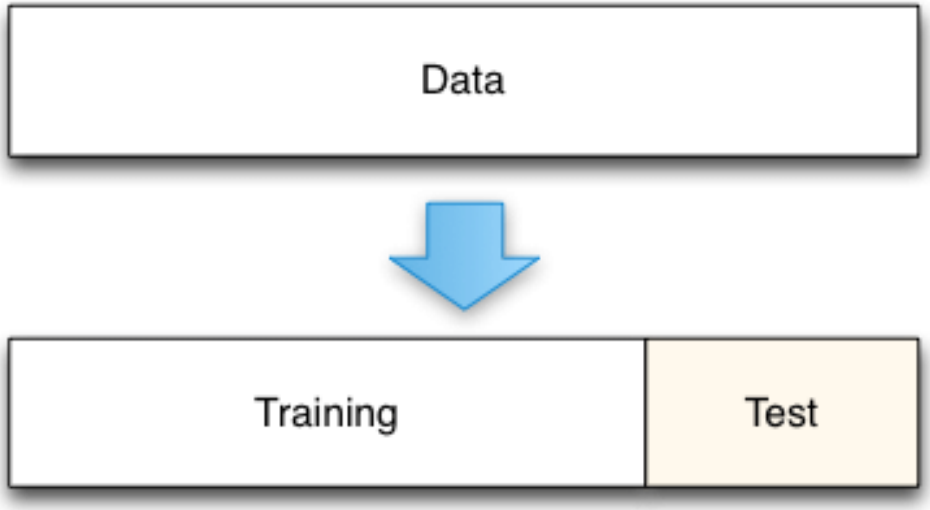
target 값의 비율을 계산한 **Sample Weight**를 설정한 모델을 완성

4 최종 모델

최종 모델 | 하이퍼파라미터 최적화(Optuna)

모델 간의 성능 점수 차이를 비교한 결과,

XGBClassifier 모델을 활용하기로 결정



Training set(80%), Test set(20%)로 구분한 이후

Test set의 결과 예측 정확도가 가장 높았던 하이퍼파라미터를 베스트 파라미터로 지정

4 최종 모델

최종 모델 | 하이퍼파라미터 최적화(Optuna)

모델 간의 성능 점수 차이를 비교한 결과,

XGBClassifier 모델을 활용하기로 결정

Train 전체 데이터셋에서 스케일링을 한 이후

train_test_split을 할 경우 valid set의 data leakage로 인해 정확하지 않음

따라서 train_test_split을 한 이후 스케일링을 적용하고,

실제 test 예측 시 다시 전체 train set에 스케일링을 적용하였음

Training set(80%), Test set(20%)로 구분한 이후

Test set의 결과 예측 정확도가 가장 높았던 하이퍼파라미터를 베스트 파라미터로 지정

최종 모델 | 하이퍼파라미터 최적화(Optuna)

예측 클래스와 실제 클래스의 일치도를 비교한 값들 중
가장 높은 값을 베스트 하이퍼파라미터로 설정하여 test set 결과 예측

Parameter	Tuning Result
n_estimators	433
learning_rate	0.17544298336548947
max_depth	3
colsample_bytree	0.7979479822008998
subsample	0.6814118250794118
alpha	0.07158768442748387
lambda	2.8854317762003943

4 최종 모델

최종 모델 파이프라인

시계열 결측치 처리

데이터 범주형 인코딩

특징 추출 및 선택

⋮

최종 Dataset

Robust Scaling

XGBoost + Sample Weight

Optuna

5

세미나 후기

5 세미나 후기

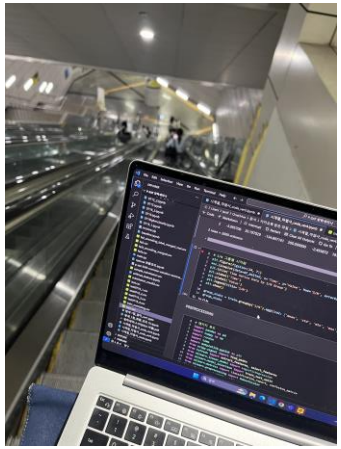
마법도 밥심이다.zip

바람직한 출석률

단체 회식도 빠지지 않긔

5 세미나 후기

너무 화목한 1팀.zip



생일날 밤 12시까지
불태웠던 태현언니의 열정 ...

또 한 번 푸드파이팅

디저트까지 더블로 가
(회귀팀장님 고마워요)

#	Team	Members	Score
1	[시계열마법사✨🌟] Dobby		4411
	Your Best Entry! Your most recent submission scored 4411, which is an improvement of your previous score of 4399. Great job!		
2	[시계열마법사✨🌟] Voldemort		4406
3	[시계열마법사✨🌟] Draco Malfoy		4376
4	[시계열마법사✨🌟] Severus Snape 👀		4364
5	[시계열마법사✨🌟] Hedwig		4309
6	[시계열마법사✨🌟✨] Ron Weasley		3910

한때의 영광...

5 세미나 후기

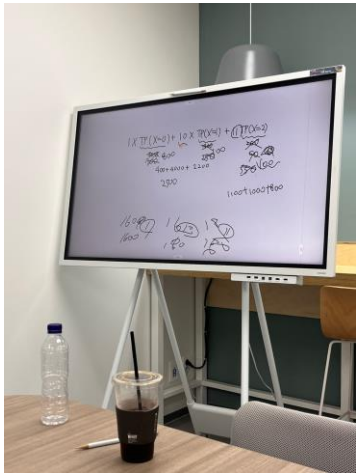
너무 화목한 1팀 2.zip

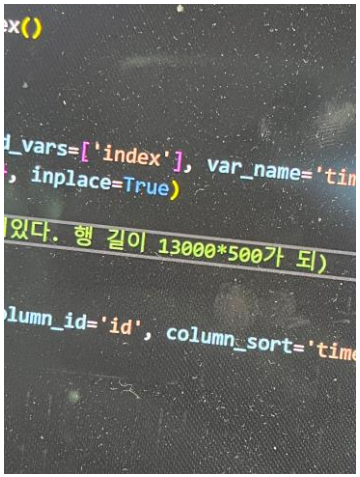
스네이프 교수님: 우리 1팀이니까...

론 위즐리: 볼에다가 해요 예쁜짓

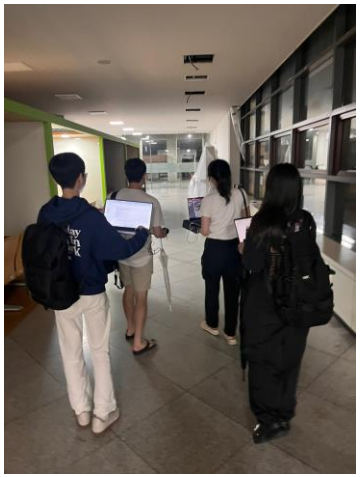
만점 추정하기

(계산하자마자 갱신...)



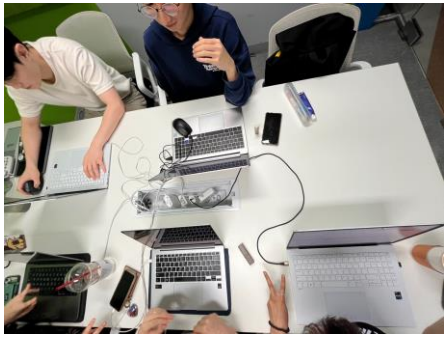


낙서한 사람: 박 상 훈



노트북을 가방에 넣을 줄 모르는 1팀 ... ㅎㅎ

아침형 마법사들의 첫 야근...



5 세미나 후기

세미나 후기

이 장면이 해리포터의 핵심 명장면이라고 합니다...

세베루스 스네이프 (동길)

처음에는 5일이라는 짧은 시간 동안에 성과를 낼 수 있을까 의문이 들었는데 EDA부터 모델링까지 하나하나 같이 살펴보면서 얘기하다보니 한 걸음 한 걸음 나아갈 수 있었던 것 같습니다. 열정적인 팀원들을 만나 방학세미나를 하는 5일이 너무 행복했고 내일이 기다려지는 기간이었습니다. 이렇게 6명이 한 팀을 해도 좋을 것 같다는 생각이 들 정도로 케미가 잘 맞았고 (제 개인적인 생각 ㅎ) 1팀 분들과 같은 팀을 하시게 될 분들이 정말 부럽습니다 ㅎㅎ

드레이코 말포이 (윤아)

이렇게 방학세미나도 끝이 나는군요.. 다들 방학 세미나 때 얻어가는 게 많다고 해서 기대도 되고 두려움도 있었는데 확실히 밀도 있게 많이 배워 간 시간이었던 것 같습니다. 사실 저보다는 팀원들이 고생을 많이 해서 저는 묻어간 감이 없잖아 있습니다ㅎㅎ 벌써 어엿한 팀장 같은 상훈이와 태현 언니, 보간법을 찾아낸 동길 오빠, 모델링 잘 하는 경미, 막내이지만 뭐든 잘 하는 능주까지, 너무나 좋은 팀원들을 만나서 좋았습니다. 팀원들이 있었기 때문에 이 일주일을 버텨낼 수 있었던 것 같습니다. 방학세미나를 진행하며 신입기수를 맞이할 마음의 준비를 했는데 기존부원으로 한 학기를 보낼 생각을 하니 벌써 떨리네요. 그래도 지난 학기에 배운 것들과 방학 세미나의 경험이 저에게 힘이 되어주면 좋겠습니다. 한 주간 모두 수고하셨고, 남은 한 학기도 힘내시길 바랍니다. 파이팅!

5 세미나 후기

세미나 후기

헤르미온느 그레인저 (상훈)

팀원들의 좋은 아이디어들을 하나 둘씩 조합하며 문제가 해결되는 과정이 너무 신기했습니다.
분위기 메이커, 아이디어 뱅크, 컴퓨팅 도사, 피피티 도사가 다 모인 정말 완벽한 팀이었습니다.
이번 방세를 통해 저도 많이 배울 수 있었고, 팀원들도 그런 기회였으면 해요!
웃으면서 즐거운 방세할 수 있었던 건 전부 1팀 덕분이라고 생각합니다.
남은 학회 일정 파이팅이고, 우리 1팀 모두 원하는 팀장/팀원 만날 수 있었으면 좋겠습니다!

도비 (태현)

좋은 팀원들과 함께해서 너무 즐겁고 알찼던 방학세미나가 벌써 이렇게 끝이 나네요. 일주일이라는 기간이 길다면 길고 짧다면 짧은 기간이지만 정말 어떻게 흘렀는지 모를 만큼 바쁘게 지나갔던 것 같습니다.
한 학기 동안 피셋 활동을 했었지만, 다른 팀원 분들이랑 많이 친해질 기회는 없었던 것 같은데, 이 기회로 또 너무 좋은 사람들을 만날 수 있게 된 것 같습니다 TT 정말 항상 열정적으로 우리를 이끌어준 상훈이, 아이디어 뱅크 동길이, 항상 꼼꼼하게 피드백해주는 경미, 항상 멋진 실행력을 보여준 윤아, 그리고 다방면으로 갖기인 똑순이 능주까지 모두 일주일 동안 여러분과 같은 팀이어서 좋았어요. 다음 학기에 다른 팀이 되더라도 모두 기존으로서 정말정말정말 너무 잘하실 것 같다는 생각이 들고 일주일동안 저도 많이 배울 수 있는 시간이었습니다. 너무너무 고생많으셨습니다 ♥

5 세미나 후기

세미나 후기

헤드위그 (능주)

오티 때까지만 해도 정말 막막했는데, 5일 간 팀원들과 머리를 맞대고 고민하다 보니 조금씩 성능이 오르는 게 신기했어요. (도파민 MAX) 짧으면서도 길었던 여정이 끝나간다니 후련하면서도 시원섭섭하지만, 한편으로는 합법적으로 다른 일을 하지 않아도 되어서 좋았어요 ㅎㅎ 정말 많은 것을 배웠고, 마법사들과도 잊지 못 할 추억을 많이 쌓은 것 같아요! 어떤 일을 하든 재미와 유익함을 동시에 잡기 쉽지 않은데, 피셋에서 함께한 두 팀과 너무나도 재밌고 알찬 시간을 보낸 것 같습니다. 팀장님들 저희 팀원들 마구마구 데려가주세요 ㅎㅎ 다음 학기에도 잘 부탁드립니다:)

론 위즐리 (경미)

기존 기수분들이 다들 입모아 강추!했던 방세여서 많이 기대됐는데 기대 이상으로 유익하고 즐거웠던 시간이였습니다. 꼼꼼하게 EDA 그래프들 뽑아주고 데이터 파악을 너무 잘해줬던 윤아언니, 결측치 보간이라는 까다로운 부분을 금새 해결해준 동길오빠.. 덕분에 저희 팀 출발이 너무 순조로웠던 것 같습니다 ㅎㅎ 뭐든 야무지고 피피티까지 뚝딱뚝딱 만들어내는 우리팀 갓기 막내 능주와 항상 노션에 잘 정리해주고 어떤 분야든 열정 불태워준 태현언니 „ 마지막으로 막힐때마다 기발한 아이디어와 방향성 제시해준 문제해결(마법)사 상훈오빠까지 !!! 모두 데마 NLP 가십셔 ... 너무너무 매력 넘치는 사람들과 하루종일 함께 있으면서 지루할 틈 없이 많이 배우고 또 행복했습니다. 이제는 어엿한 기존이 될 생각에 걱정도 되고 떨리기도 하지만, 또 얼마나 좋은 사람들을 만나게 될지, 또 어떤 유익한 시간을 보내게 될지 두근두근하네요 ㅎㅎ 모두 남은 한 학기도 건강하고 행복하게 피셋해요 ! ☺

THANK YOU


베스트 후렌드
우린 칭구칭귀
텐션 올려
킹킹
데박
아무도 우릴 막을수 없쎔
빠샤
keep yourself alive at photoism

6

Appendix

6 Appendix

select_features() 관련 추가 자료