

팔당댐 홍수 안전운영에 따른 한강 수위예측 AI 경진대회

[생각하는 감자들]

조규원 성형훈 이남선 이동호 김성주

상어 tony_hsung solsol123 as9786 말감

팔당댐 - 생각하는 감자들

목차

1 데이터 설명

2 전처리

- 1) 결측치/이상치 처리
- 2) Log 변환

3 VAR 모델

- 1) VAR 모델 설명
- 2) VAR 모델 적합성 확인
- 3) 시차길이(p) 결정

4 VAR 모델 튜닝

- 1) 학습기간
- 2) 스케일
- 3) 변수 선택
- 4) 앙상블

5 결론

팔당댐 - 생각하는 감자들

1. 데이터 설명

1. 데이터 설명 - 제공 데이터

Index	설명
ymdhm	년월일시분
독립변수	설명
swl	팔당댐 현재수위 (단위: El.m)
inf	팔당댐 유입량 (단위: m ³ /s)
sfw	팔당댐 저수량 (단위: 만m ³)
ecpc	팔당댐 공용량 (단위: 백만m ³)
tototf	총 방류량 (단위: m ³ /s)
tide_level	강화대교 조위 (단위: cm)
fw_1018662	청담대교 유량 (단위: m ³ /s)
fw_1018680	잠수교 유량 (단위: m ³ /s)
fw_1018683	한강대교 유량 (단위: m ³ /s)
fw_1019630	행주대교 유량 (단위: m ³ /s)
rf_10184100	대곡교 강수량
rf_10184110	진관교 강수량
rf_10184140	송정동 강수량
종속변수	설명
wl_1018662	청담대교 수위 (단위: cm)
wl_1018680	잠수교 수위 (단위: cm)
wl_1018683	한강대교 수위 (단위: cm)
wl_1019630	행주대교 수위 (단위: cm)

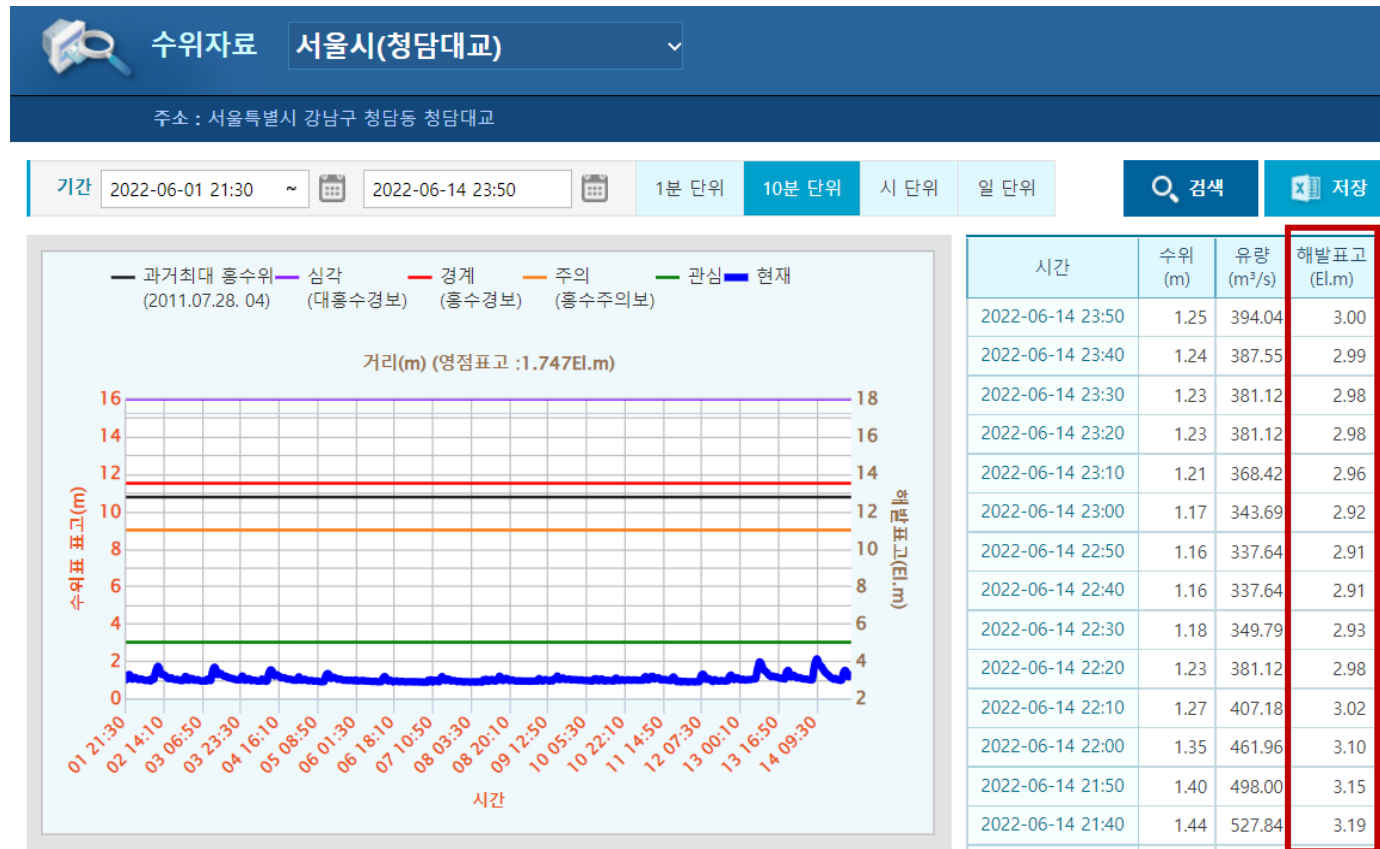
변수 간 연관성 및 특이사항

- inf : 측정 시 저수량과 전 시간의 저수량 차이와 단위시간동안 방류량을 더한 양을 단위환산한 값 (즉, 저수량차보다 방류량이 더 크다면 음수 발생 가능)
- sfw : swl를 통해 계산되는 파생변수
- 저수량(sfw)+공용량(ecpc)=총저수량(244.0)
- fw_1018680 : 본래 측정이 되지 않는 데이터
- fw_1018683 : 강물 역류 시 마이너스 값으로 측정
- 강수량 데이터 : 값이 0.0인 관측치가 매우 많음 (참고: 최근 10년 서울 연간 강수일수 약 100일로 전체일수 중 약 27%)

1. 데이터 설명 - 외부 데이터

- 외부데이터


- '한강 홍수 통제소 - 실시간 수문자료'에서 target 변수들의 test 기간 데이터 수집
- 해발표고(EI.m) 데이터 수집하여 100을 곱한 값을 사용



출처 : <http://hrfco.go.kr/sumun/waterlevelList.do#>

2. 데이터 전처리

2. 데이터 전처리 - 결측치 처리



<code>df1.isnull().sum()</code>	
swl	743
inf	743
sflw	743
ecpc	743
tototf	743
tide_level	4927
wl_1018662	59
fw_1018662	16380
wl_1018680	59
fw_1018680	196849
wl_1018683	59
fw_1018683	1279
wl_1019630	59
fw_1019630	59
rf_10184100	0
rf_10184110	0
rf_10184140	0
dtype: int64	

<code>df1.interpolate(method='linear', inplace=True)</code>	
<code>df1.isnull().sum()</code>	
swl	0
inf	0
sflw	0
ecpc	0
tototf	0
tide_level	0
wl_1018662	0
fw_1018662	0
wl_1018680	0
fw_1018680	0
wl_1018683	0
fw_1018683	0
wl_1019630	0
fw_1019630	0
rf_10184100	0
rf_10184110	0
rf_10184140	0
dtype: int64	

- 시계열 데이터, 특히 짧은 간격을 두고 측정된 시계열 데이터의 경우, **시간의 흐름에 따라 연속적인 값을 갖는 특징이 있음**

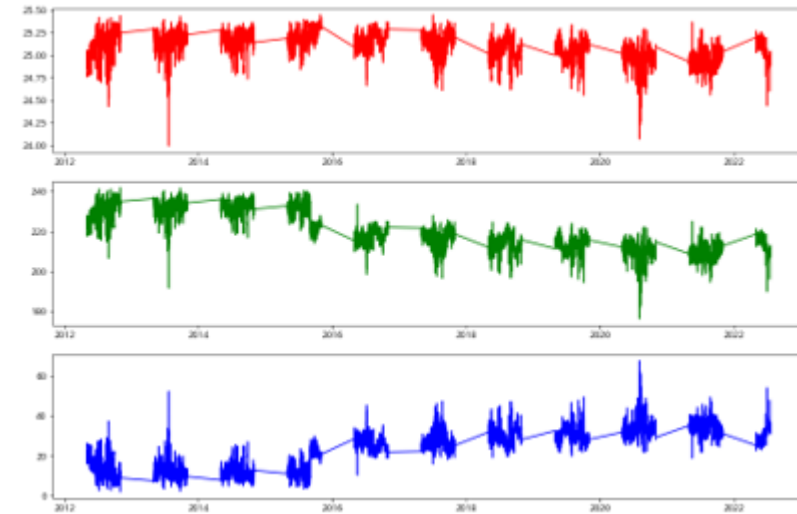
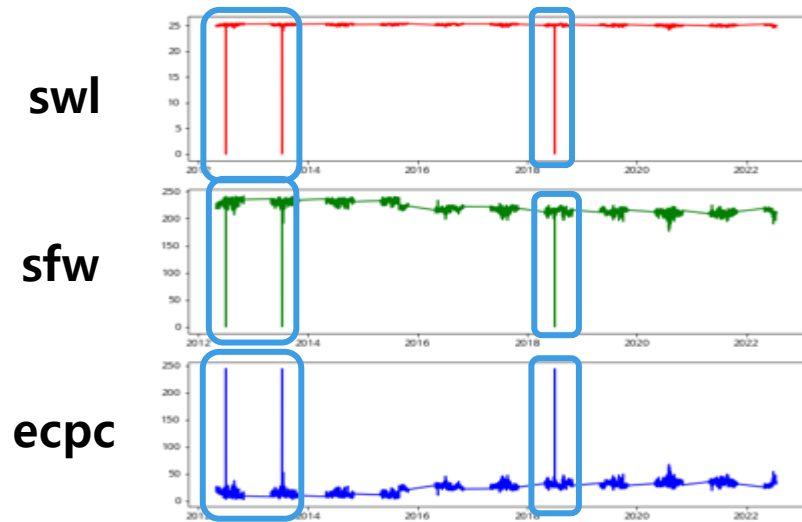
→ 따라서, 결측치를 단순히 평균값으로 대체하면 오히려 큰 왜곡이 발생할 가능성이 존재하고 특히, skewedness가 큰 데이터의 경우, 평균 대체 시 더 큰 왜곡이 발생할 수 있음

예를 들어, 강수량과 같이 0값이 대부분인 경우 평균은 0에 가깝게 나타나게 되는데, 폭우가 온 시점의 결측치를 평균으로 대체할 경우 큰 왜곡이 발생

➤ 따라서, 결측치 이전과 이후 관측치 데이터를 스무스하게 연결할 수 있는 사이값을 찾아 결측치를 대체하는 **보간법** 채택

- `pandas.DataFrame.interpolate()` 함수 사용

2. 데이터 전처리 - 이상치 처리

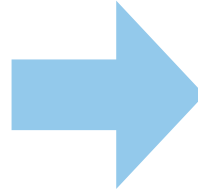


요약 통계량 확인 결과, 다음의 이상치 확인

- swl 현재수위 최소값 0.0
- sfw 저수량 최소값 -0.01
- ecpc 공용량 최대값 -990
- swl, sfw, ecpc는 서로 연관이 있는 데이터인 것을 파악했기 때문에 세 변수 동시에 확인 결과, 이상치 관측 시기가 동일한 것을 확인
=> 보간법으로 이상치 대체
- 이 때, ecpc는 sfw를 먼저 보간법으로 대체한 뒤 '244-sfw'값으로 변환

2. 데이터 전처리 - 이상치 처리

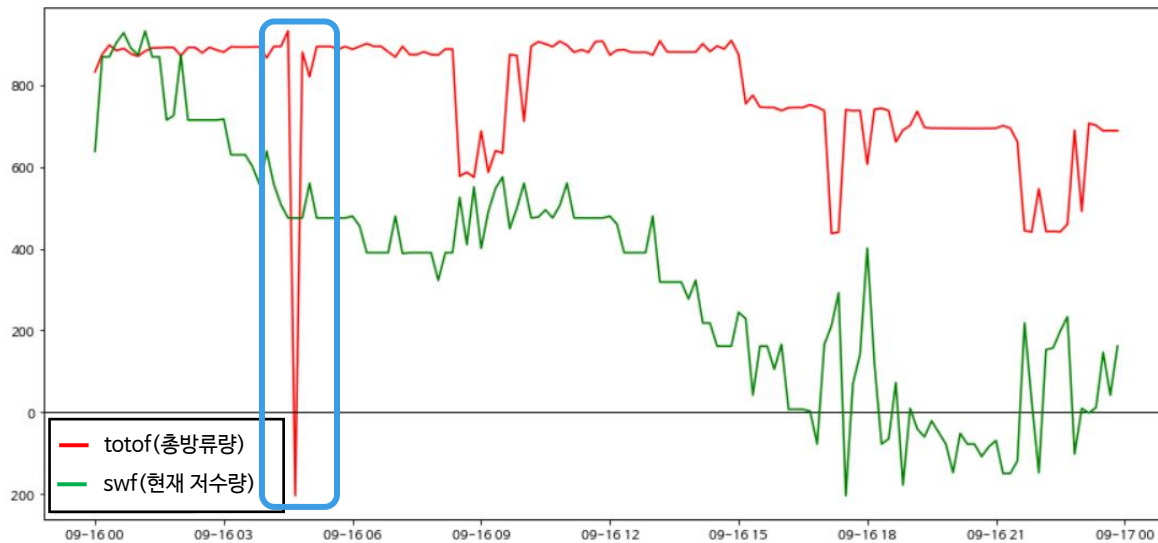
tototf 총 방류량 최소값이 -990인 이상치를 처리하기 위해 연관 변수들인 sfw, inf를 함께 확인한 결과



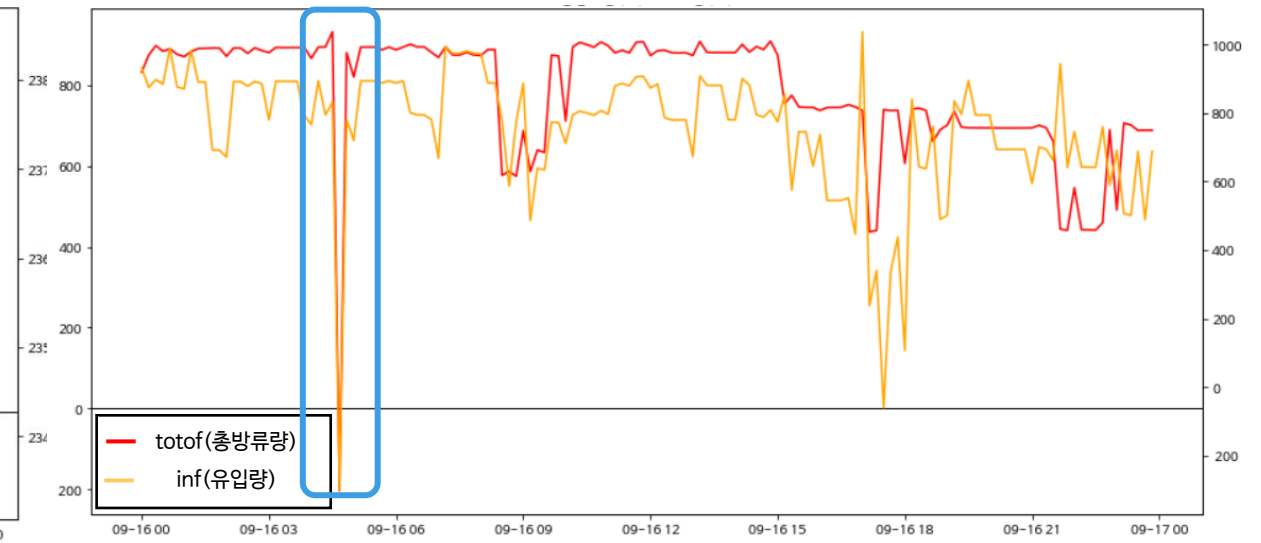
총 방류량과 유입량이 앞뒤 시점과 달리 크게 변동하는 것을 확인

⇒ 결측치 처리와 마찬가지로 tototf와 inf 모두 보간법으로 대체

총 방류량(좌)과 현재 저수량(우)

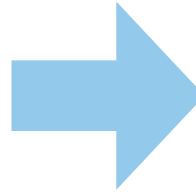


총 방류량(좌)과 유입량(우)



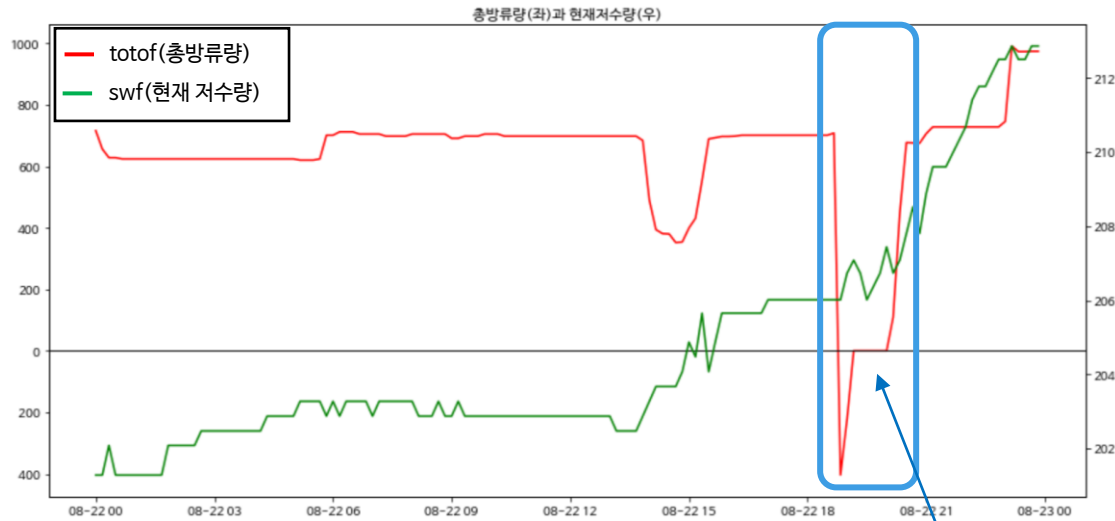
2. 데이터 전처리 - 이상치 처리

2020-08-22 18:50, 19:00 이상치의 경우
다음 시점의 방류량도 0이기 때문에 보간법을
사용하지 않고 inf 산출 공식*에 대입하여 생각



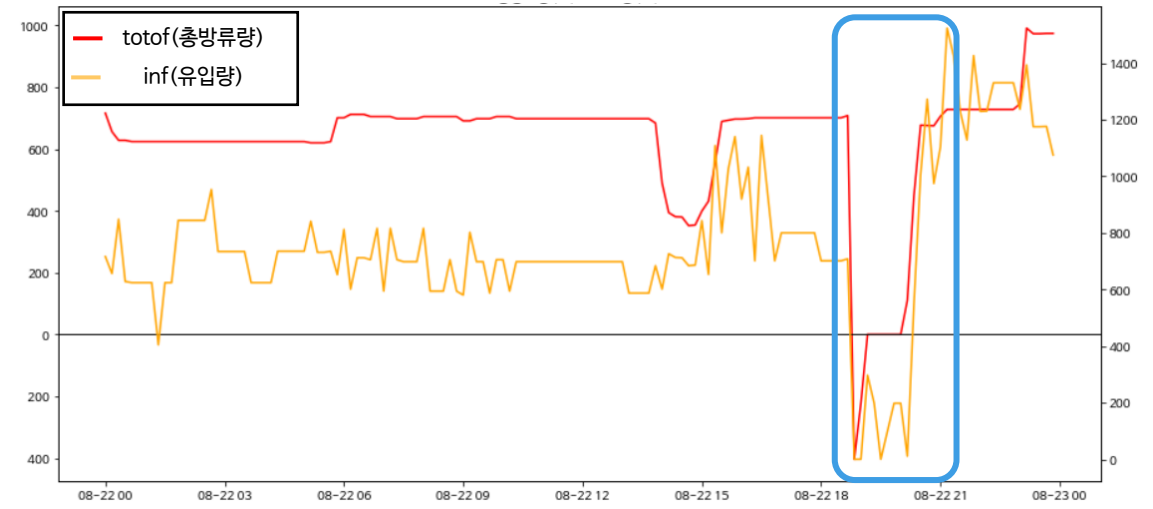
18:50 시점 저수량 변화량은 0,
유입량도 0이므로 방류량도 0으로 변환하고,
19:00 시점 이후에도 총 방류량 0이므로 연속성,
보간법에 의해 19:00 시점 방류량을 0으로 변환.
이 경우, 공식*에 의해 유입량은 양수가 되어야 하
므로 유입량은 bfill로 대체

총 방류량(좌)과 현재 저수량(우)



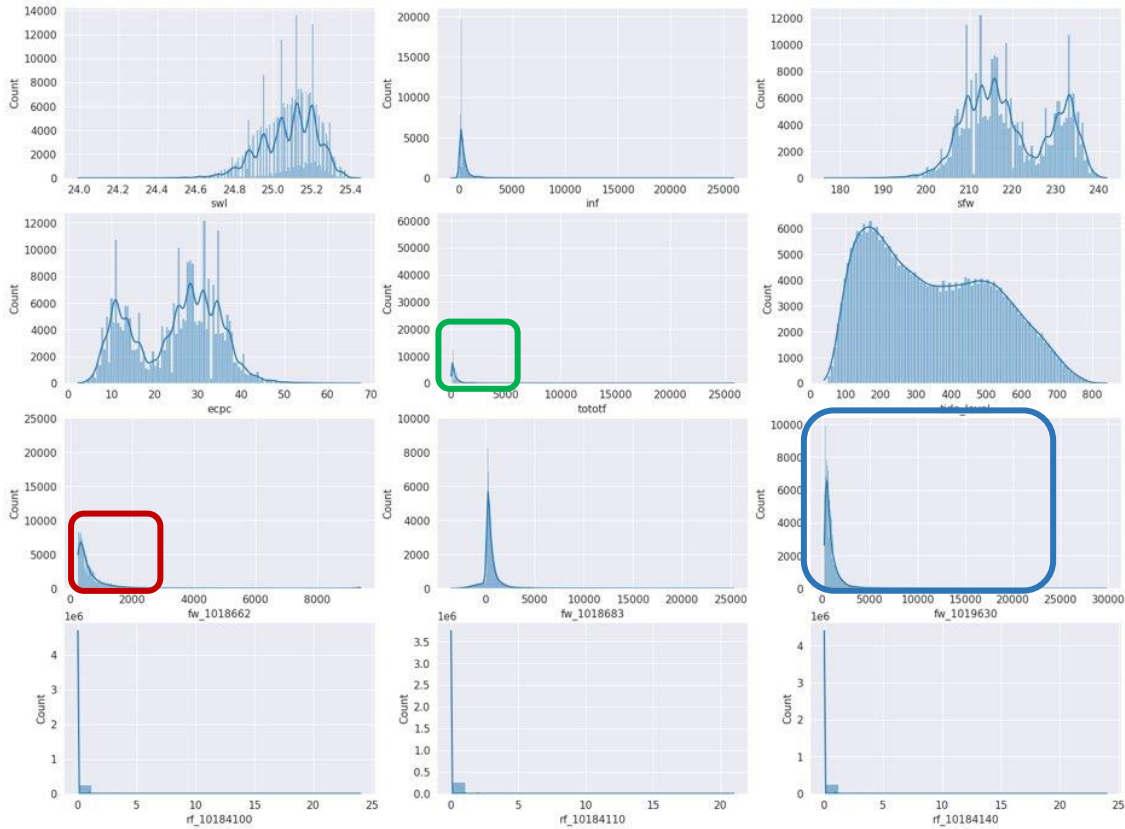
이상치 이후 시점도
방류량 0 유지

총 방류량(좌)과 유입량(우)

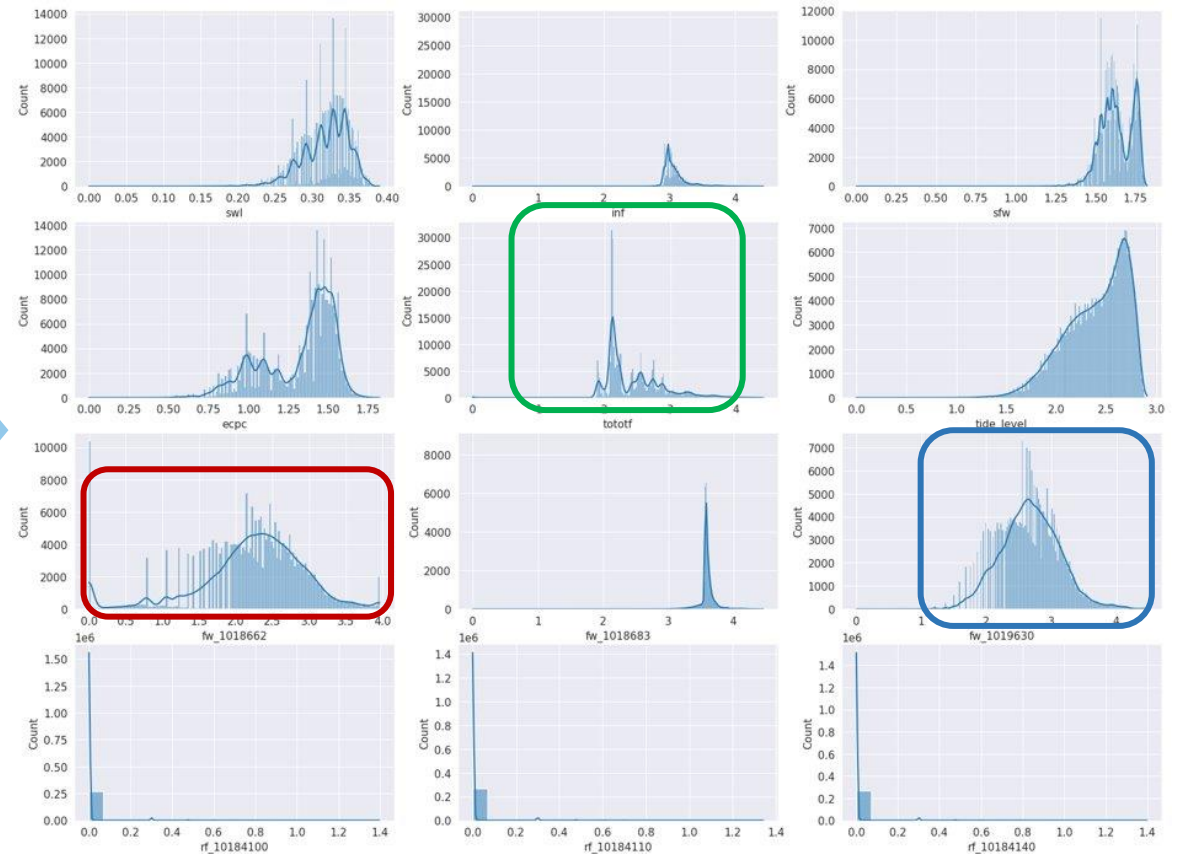


*저수량 변화량을 단위환산한 값 - 총방류량 = 유입량

2. 데이터 전처리 - log 변환



- 홍수발생 케이스가 많지 않기 때문에 데이터들이 대체적으로 왜도가 높은 분포를 보임
- 왜도를 낮추고 정규분포에 가깝게 만들기 위해 log transformation



- Log 함수는 $x > 0$ 에서만 정의되기 때문에 전체 데이터 분포를 $x > 0$ 으로 shift시킨 뒤 상용로그 변환

$$x_{new} = \log_{10}(x_{old} - \min(x_{old}) + 1)$$

2. 데이터 전처리 - log 변환

```
y_cols=['wl_1018662', 'wl_1018680', 'wl_1018683', 'wl_1019630']
x_cols=df.drop(y_cols,axis=1).columns

df_log=pd.DataFrame.copy(df)
df_log=df_log.drop(y_cols, axis=1)

# 전체 상용로그변환 (x-min(x)+1) 변환
for i in range(len(x_cols)):
    df_log[x_cols[i]]=np.log10(df_log[x_cols[i]]-min(df_log[x_cols[i]])+1)
```

팔당댐 - 생각하는 감자들

3. VAR 모델

3. VAR 모델 - 모델 설명

☑ Vector Autoregressive Model (벡터자기회귀모형)

- VAR 모형은 통계 기반의 시계열분석 모형으로 많이 사용되는 ARIMA 모형 등이 일변량 분석이라는 한계점을 보완한 다변량 분석 모형

- 모형 내의 모든 변수들의 과거 데이터를 동시에 독립변수로 사용하여 현재 및 미래 시점을 분석/예측
- 단기간 예측에 뛰어난 성능을 보이는 것으로 알려짐

⇒ 외부데이터 포함 p시점 전까지의 모델 내 모든 데이터를 활용해 10분 후 1개 시점만 예측하는 초단기간 예측 진행

$$\begin{aligned} X_t &= C + \Theta_1 X_{t-1} + \dots + \Theta_p X_{t-p} + \varepsilon_t \\ &= C + \sum_{i=1}^p \Theta_i X_{t-i} + \varepsilon_t \end{aligned}$$

$$\begin{pmatrix} x_{1,t} \\ x_{2,t} \\ \vdots \\ x_{N,t} \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_N \end{pmatrix} + \sum_{i=1}^p \begin{pmatrix} \Theta_{11} & \Theta_{12} & \dots & \Theta_{1N} \\ \Theta_{21} & \Theta_{22} & \dots & \Theta_{2N} \\ \dots & \dots & \dots & \dots \\ \Theta_{N1} & \Theta_{N2} & \dots & \Theta_{NN} \end{pmatrix}^i \begin{pmatrix} x_{1,t-i} \\ x_{2,t-i} \\ \vdots \\ x_{N,t-i} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1,t} \\ \varepsilon_{2,t} \\ \vdots \\ \varepsilon_{N,t} \end{pmatrix}$$

3. VAR 모델 - 모델 적합성 확인

1) 그레인저 인과관계 (Granger causality)

: 인과관계를 규명하는 것은 어렵기 때문에 두 요인 중 상대적으로 먼저 영향을 미치는 (즉, 독립변수로 작용하는) 변수를 파악하고자 할 때 사용

- 귀무 가설 : 변수1이 변수2를 예측하는 데 도움이 되지 않는다

- 대립 가설 : 변수1이 변수2를 예측하는 데 도움이 된다

=> p-value가 유의수준보다 작을 경우 귀무 가설 기각.

즉, 변수1이 변수2에 선행하고 변수2를 예측하는 데 도움이 됨(→ 독립변수로 채택)

※ 확인 결과, 모든 변수들이 한강다리 수위를 예측하는 데 도움이 되는 변수

	swl_x	inf_x	sfw_x	ecpc_x	tototf_x	tide_level_x	fw_1018662_x	fw_1018683_x	fw_1019630_x	rf_10184100_x	rf_10184110_x	rf_10184140_x
wl_1018662_y	0.0	0.0	0.0000	0.0000	0.0	0.0000	0.0	0.0000	0.0000	0.0000	0.000	0.0000
wl_1018680_y	0.0	0.0	0.0000	0.0000	0.0	0.0000	0.0	0.0000	0.0000	0.0000	0.000	0.0000
wl_1018683_y	0.0	0.0	0.0000	0.0000	0.0	0.0000	0.0	0.0000	0.0000	0.0000	0.000	0.0000
wl_1019630_y	0.0	0.0	0.0000	0.0000	0.0	0.0000	0.0	0.0000	0.0000	0.0000	0.000	0.0000

3. VAR 모델 - 모델 적합성 확인

2) 다중 공선성

: 특정 변수가 다른 변수로부터 파생된 경우 등과 같이 변수 간에 강한 상관관계가 있는 경우 독립변수들이 다중 공선성을 가짐.

: 다중 공선성이 있을 경우 종속변수와의 관계를 나타내는 가중치에 왜곡 발생 및 모형 성능 저하를 초래할 수 있음
(단, 종속변수와 독립변수의 관계를 설명하는 목적이 아니라 단지 예측을 위한 경우에는 크게 고려되지 않을 수 있음)

- VIF 지표를 통해 파악하며 그 값이 10 이상인 경우 다중공선성이 있다고 판단

※ 확인 결과, swl, inf, sfw, ecpc, tototf에서 아주 강한 다중 공선성이 나타남.

댐 자료 변수들은 다른 변수로부터 파생된 변수들이 많아 그런 것으로 파악하여 추후 일부 독립변수를 제거한 모델도 돌려보기로 결정.

〈전체〉

	VIF_Factor	Feature
0	312.642865	swl
1	1153.559528	sfw
2	93.192692	ecpc
3	74.878437	tototf
4	64.588002	tide_level
5	45.211044	fw_1018662
6	1181.224681	fw_1018683
7	180.631163	fw_1019630
8	1.441315	rf_10184100
9	1.466396	rf_10184110
10	1.597194	rf_10184140

〈inf 제거〉

	VIF_Factor	Feature
0	155.288922	swl
1	52.170467	ecpc
2	71.402666	tototf
3	60.048087	tide_level
4	45.106617	fw_1018662
5	462.069564	fw_1018683
6	179.735972	fw_1019630
7	1.441314	rf_10184100
8	1.466273	rf_10184110
9	1.597139	rf_10184140

〈inf, sfw 제거〉

	VIF_Factor	Feature
0	47.739621	swl
1	29.778153	ecpc
2	52.613550	tototf
3	58.499421	tide_level
4	44.985571	fw_1018662
5	178.029813	fw_1019630
6	1.440984	rf_10184100
7	1.465443	rf_10184110
8	1.596778	rf_10184140

3. VAR 모델 - 모델 적합성 확인

3) 정상성(stationarity)* 확인

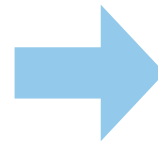
: 다른 시계열분석 모델과 마찬가지로 VAR 모델 역시 정상 시계열 데이터로 분석을 수행해야 함

- 데이터의 정상성 확인을 위해 주로 사용하는 ADF(Augmented Dickey and Fuller) 검정 수행
- p-value가 유의수준 보다 작을 경우, 데이터가 non-stationary하다는 귀무 가설을 기각. 즉, stationary.

```
# df_train 전체 각 열에 ADF Test
for name, column in df_train.iteritems():
    adfuller_test(column, name=column.name)
    print('\n')
```



```
Augmented Dickey-Fuller Test on "swl"
-----
Null Hypothesis: Data has unit root. Non-Stationary.
Significance Level      = 0.05
Test Statistic          = -15.1588
No. Lags Chosen         = 87
Critical value 1%       = -3.43
Critical value 5%       = -2.862
Critical value 10%      = -2.567
=> P-Value = 0.0. Rejecting Null Hypothesis.
=> Series is Stationary.
```



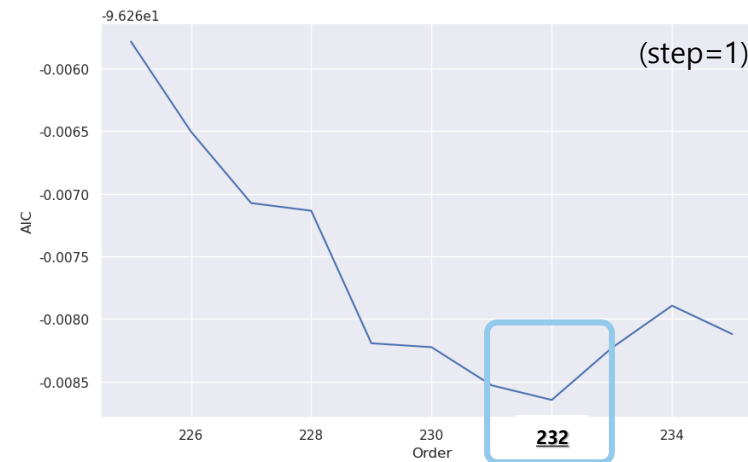
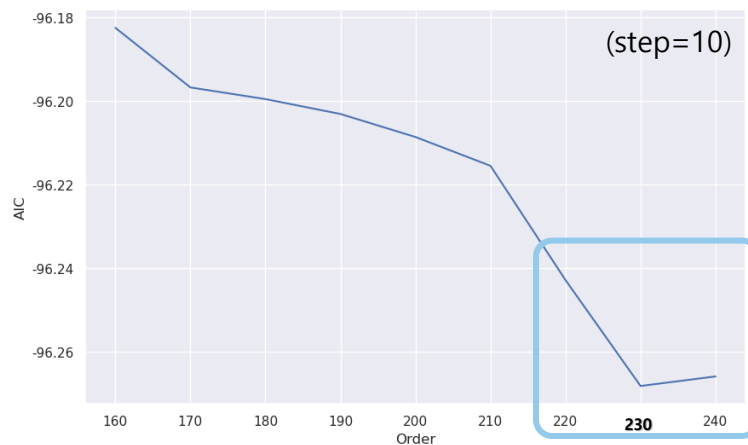
모든 변수가 정상 시계열 데이터임 확인

3. VAR 모델 - 시차길이(p) 결정

$$X_t = C + \Theta_1 X_{t-1} + \dots + \Theta_p X_{t-p} + \varepsilon_t$$

- VAR 모델에서 몇 시차(p) 전까지의 데이터를 모델에 포함시킬 것인지 결정하기 위해 Akaike Information Criterion(AIC) 지표 확인
 - AIC는 예측 오차에 대한 추정치로 최소의 정보 손실을 갖는 모델을 선택하는 방법
 - $AIC = -2 \ln(L) + 2k$ (L =Likelihood function, k = 파라미터의 수)
 - 따라서, AIC는 모델 예측력이 높을수록 감소하고 파라미터 수가 많을수록 증가 → AIC를 최소로 만드는 시차(p)를 선택

※ 확인 결과, p=232에서 AIC 최소값



4. VAR 모델 튜닝

4. VAR 모델 튜닝 - 학습기간 / Scaling

학습기간

- 제공데이터의 10월부터 그 다음해 4월까지의 데이터에 공백이 있어,
먼저 2022-05-01 00:00부터 2022-05-31 23:50까지의 데이터만 학습데이터로 사용
- 이후, 많은 양의 학습데이터로 학습시킬 경우 성능이 향상될 것을 기대하며 2012년도부터 2022년 5월까지
의 데이터를 학습데이터로 사용 ⇒ 성능 향상

Scaling 3가지 방식

1. Standard Scaling
2. MinMax Scaling
3. Robust Scaling

4. VAR 모델 튜닝 - 변수 선택

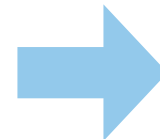
PCA 차원 축소

☑ PCA(주성분분석)

- feature extraction의 대표적인 방법으로 기존 데이터들을 적은 수의 새로운 축으로 영사시켜 정보의 손실을 최소화하면서 feature의 개수를 줄이는 방법
 - 변수 간 다중공선성을 제거하고 feature 개수를 줄였을 때 모델 성능 개선을 기대하며 PCA 진행
 - EDA를 통해 'swl', 'sfw', 'ecpc'와 'sfw', 'inf', 'tototf' 변수가 서로 연관됨을 알았기 때문에 해당 2개와 한강다리 수위 데이터, 한강다리 유량 데이터, 강수량 데이터까지 총 5개의 차원으로 축소할 수 있을 것으로 예상(n_components=5)
- ☞ 결과가 현저히 좋지 않아 최종 모델에서 **제외**

다중공선성 기반 변수 선택

모든 변수를 feature로 사용한 모델과
다중공산성 방지를 위해 VIF 값이 높은
inf와 sfw 변수를 제거한 모델 성능 비교



전체적으로
inf와 sfw 변수를 제거한
model이 성능이 좋음

※ inf와 sfw 변수를 제외한 변수들만 모델에 반영하기로 결정

4. VAR 모델 튜닝 - Ensemble

```
df1 = pd.read_csv('minmax.csv', index_col=0)
df2 = pd.read_csv('robust.csv', index_col=0)
df3 = pd.read_csv('standard.csv', index_col=0)
df4 = pd.read_csv('VAR(232)_logscale.csv', index_col=0)
```

```
result = (df1.values+df2.values+df3.values+df4.values)/4
result
```

- inf, swl 변수들을 제거한 후 학습한 model이 성능이 좋으므로 4가지 방법으로 scaling을 진행한 이 model들을 바탕으로 ensemble을 진행
- Ensemble 방법은 4가지 model들이 예측한 예측값들의 평균을 계산

5. 결론

5. 결론

- Log 변환하여 왜도가 높은 데이터 분포 변환
- VAR 모델 적용하기 위해 Granger causality, 다중공선성, 정상성 확인
 - Granger causality : 모든 변수들이 한강다리 수위를 예측하는 데 유의미한 변수임
 - 다중공선성 : inf, sfw 등 변수의 다중공선성이 높은 것으로 파악됨
 - 정상성 : 모든 데이터가 정상 시계열 데이터로 VAR 모델에 적용하기 적합
- VAR(p) 모델의 최적 시차 p 도출을 위해 AIC 확인
 - $p=232$ 에서 AIC가 최소인 것으로 확인되어 VAR(232) 모델 적용하기로 결정
- 모델 튜닝을 위해 학습데이터 기간, 스케일러, 변수 선택 방법을 다양하게 시도
 - 2012년~2021년 기간을 학습데이터로 사용하는 것이 가장 효과적
 - 스케일러를 적용하지 않은 모델과 Standard, MinMax, Robust 스케일러를 각각 적용한 모델을 앙상블 하는 것이 가장 효과적
 - 다중공선성이 가장 높은 inf, sfw 변수를 제외한 모델이 가장 효과적
- 따라서, 'Log 변환 + VAR(232) 모델 + scaler 앙상블' 모델이 최종적으로 가장 성능이 좋은 것으로 나타남

팔당댐 - 생각하는 감자들

Requirements

Requirements

- `absl-py==0.12.0`
- `alembic @ file:///tmp/wheelhouse/alembic-1.4.3-py2.py3-none-any.whl`
- `anyio==2.2.0`
- `argon2-cffi==20.1.0`
- `astunparse==1.6.3`
- `async-generator @ file:///tmp/wheelhouse/async_generator-1.10-py3-none-any.whl`
- `attrs @ file:///tmp/wheelhouse/attrs-20.3.0-py2.py3-none-any.whl`
- `Babel==2.9.0`
- `backcall==0.2.0`
- `backports.entry-points-selectable==1.1.1`
- `beautifulsoup4==4.6.0`
- `benepar==0.2.0`
- `bleach==3.3.0`
- `blis==0.7.4`
- `boto3==1.17.53`
- `botocore==1.20.53`
- `bs4==0.0.1`
- `cachetools==4.2.1`
- `catalogue==2.0.3`
- `catboost==0.25.1`
- `certifi @ file:///tmp/wheelhouse/certifi-2020.12.5-py2.py3-none-any.whl`
- `certipy @ file:///tmp/wheelhouse/certipy-0.1.3-py3-none-any.whl`
- `cffi @ file:///tmp/wheelhouse/cffi-1.14.4-cp38-cp38-manylinux1_x86_64.whl`
- `chardet @ file:///tmp/wheelhouse/chardet-3.0.4-py2.py3-none-any.whl`
- `click==7.1.2`
- `colorama==0.4.3`

Requirements

- cryptography @ file:///tmp/wheelhouse/cryptography-3.3.1-cp36-abi3-manylinux2010_x86_64.whl
- cssselect==1.1.0
- customized-konlpy==0.0.64
- cycler==0.10.0
- cymem==2.0.5
- Cython==0.29.23
- dbus-python==1.2.16
- decorator==4.4.2
- defusedxml==0.7.1
- deprecation==2.1.0
- distlib==0.3.3
- entrypoints @ file:///tmp/wheelhouse/entrypoints-0.3-py2.py3-none-any.whl
- feedfinder2==0.0.4
- feedparser==6.0.2
- filelock==3.4.0
- flatbuffers==1.12
- funcy==1.15
- future==0.18.2
- gast==0.3.3
- gensim==4.0.1
- gluonnlp==0.10.0
- google-auth==1.29.0
- google-auth-oauthlib==0.4.4
- google-pasta==0.2.0
- graphviz==0.8.4
- grpcio==1.32.0

Requirements

- h5py==2.10.0
- idna @ file:///tmp/wheelhouse/idna-2.10-py2.py3-none-any.whl
- ipykernel==5.5.3
- ipython==7.22.0
- ipython-genutils @ file:///tmp/wheelhouse/ipython_genutils-0.2.0-py2.py3-none-any.whl
- jedi==0.18.0
- jieba3k==0.35.1
- Jinja2 @ file:///tmp/wheelhouse/Jinja2-2.11.2-py2.py3-none-any.whl
- jmespath==0.10.0
- joblib==1.0.1
- JPype1==1.2.1
- json5==0.9.5
- jsonschema @ file:///tmp/wheelhouse/jsonschema-3.2.0-py2.py3-none-any.whl
- konlpy==0.5.2
- krwordrank==1.0.3
- kss==2.5.0
- lightgbm==3.2.1
- lxml==4.6.3
- Mako @ file:///tmp/wheelhouse/Mako-1.1.3-py2.py3-none-any.whl
- Markdown==3.3.4
- MarkupSafe @ file:///tmp/wheelhouse/MarkupSafe-1.1.1-cp38-cp38-manylinux1_x86_64.whl
- matplotlib==3.4.1
- mecab-python==0.996-ko-0.9.2
- mistune==0.8.4
- murmurhash==1.0.5
- mxnet==1.8.0.post0

Requirements

- jupyter-client==6.1.12
- jupyter-core==4.7.1
- jupyter-packaging==0.9.1
- jupyter-server==1.6.1
- jupyter-telemetry @ file:///tmp/wheelhouse/jupyter-telemetry-0.1.0-py3-none-any.whl
- jupyterhub @ file:///tmp/wheelhouse/jupyterhub-1.3.0-py3-none-any.whl
- jupyterlab==3.0.14
- jupyterlab-pygments==0.1.2
- jupyterlab-server==2.4.0
- Keras==2.4.3
- Keras-Preprocessing==1.1.2
- kiwisolver==1.3.1
- kobert-transformers==0.4.1
- nbclassic==0.2.7
- nbclient==0.5.3
- nbconvert==6.0.7
- nbformat==5.1.3
- nest-asyncio==1.5.1
- networkx==2.5.1
- newspaper3k==0.2.8
- nltk==3.6.1
- node2vec==0.4.3
- notebook==6.3.0
- numexpr==2.7.3
- numpy==1.19.5
- oauthlib @ file:///tmp/wheelhouse/oauthlib-3.1.0-py2.py3-none-any.whl

Requirements

- opencv-python==4.5.1.48
- opt-einsum==3.3.0
- packaging==21.3
- pamela @ file:///tmp/wheelhouse/pamela-1.0.0-py2.py3-none-any.whl
- pandas==1.2.4
- pandocfilters==1.4.3
- parso==0.8.2
- pathy==0.4.0
- patsy==0.5.2
- pexpect==4.8.0
- pickle5==0.0.11
- pickleshare==0.7.5
- Pillow==8.2.0
- platformdirs==2.4.0
- plotly==4.14.3
- preshed==3.0.5
- prometheus-client @ file:///tmp/wheelhouse/prometheus_client-0.9.0-py2.py3-none-any.whl
- prompt-toolkit==3.0.18
- protobuf==3.15.8
- psutil==5.8.0
- ptyprocess==0.7.0
- pyasn1==0.4.8
- pyasn1-modules==0.2.8
- pybind11==2.6.2
- pycparser @ file:///tmp/wheelhouse/pycparser-2.20-py2.py3-none-any.whl
- pycrfsuite-spacing==1.0.2

Requirements

- pycurl==7.43.0.2
- pydantic==1.7.3
- Pygments==2.8.1
- PyGObject==3.36.0
- pyLDAvis==3.3.1
- pyOpenSSL @ file:///tmp/wheelhouse/pyOpenSSL-20.0.0-py2.py3-none-any.whl
- pyparsing==2.4.7
- pypersistent @ file:///tmp/wheelhouse/pypersistent-0.17.3-cp38-cp38-linux_x86_64.whl
- PySocks==1.7.1
- python-apt==2.0.0+ubuntu0.20.4.4
- python-crfsuite==0.9.7
- python-dateutil @ file:///tmp/wheelhouse/python_dateutil-2.8.1-py2.py3-none-any.whl
- python-editor @ file:///tmp/wheelhouse/python_editor-1.0.4-py3-none-any.whl
- python-json-logger @ file:///tmp/wheelhouse/python_json_logger-2.0.1-py34-none-any.whl
- pytorch-crf==0.7.2
- pytorch-pretrained-bert==0.6.2
- pytorch-transformers==1.2.0
- pytz==2021.1
- pyxattr==0.6.1
- PyYAML==5.4.1
- pyzmq==22.0.3
- regex==2021.4.4
- requests @ file:///tmp/wheelhouse/requests-2.25.0-py2.py3-none-any.whl
- requests-file==1.5.1
- requests-oauthlib==1.3.0
- retrying==1.3.3

Requirements

- rsa==4.7.2
- ruamel.yaml @ file:///tmp/wheelhouse/ruamel.yaml-0.16.12-py2.py3-none-any.whl
- ruamel.yaml.clib @ file:///tmp/wheelhouse/ruamel.yaml.clib-0.2.2-cp38-cp38-manylinux1_x86_64.whl
- s3transfer==0.3.7
- sacremoses==0.0.44
- scikit-learn==0.24.1
- scipy==1.4.1
- seaborn==0.11.1
- selenium==3.141.0
- Send2Trash==1.5.0
- sentencepiece==0.1.95
- seqeval==1.2.2
- sgmlib3k==1.0.0
- six @ file:///tmp/wheelhouse/six-1.15.0-py2.py3-none-any.whl
- sklearn==0.0
- sklearn-crfsuite==0.3.6
- smart-open==3.0.0
- sniffio==1.2.0
- soylemma==0.2.0
- soynlp==0.0.493
- spacy==3.0.5
- spacy-legacy==3.0.2
- SQLAlchemy @ file:///tmp/wheelhouse/SQLAlchemy-1.3.20-cp38-cp38-manylinux2010_x86_64.whl
- srsly==2.4.1
- statsmodels==0.13.2

Requirements

- tabulate==0.8.9
- tensorboard==2.4.1
- tensorboard-plugin-wit==1.8.0
- tensorflow==2.4.1
- tensorflow-estimator==2.4.0
- termcolor==1.1.0
- terminado==0.9.4
- testpath==0.4.4
- thinc==8.0.2
- threadpoolctl==2.1.0
- tinysegmenter==0.3
- tldextract==3.1.0
- tokenizers==0.10.2
- tomlkit==0.7.0
- tomotopy==0.11.1
- torch==1.8.1+cu111
- torch-struct==0.5
- torchaudio==0.8.1
- torchvision==0.9.1+cu111
- tornado @ file:///tmp/wheelhouse/tornado-6.1-cp38-cp38-manylinux2010_x86_64.whl
- tqdm==4.60.0
- traitlets @ file:///tmp/wheelhouse/traitlets-5.0.5-py3-none-any.whl
- transformers==4.5.1
- treform==1.1.5
- tweepy==3.10.0
- typer==0.3.2

Requirements

- typing-extensions==3.7.4.3
- ubuntu-drivers-common==0.0.0
- urllib3 @ file:///tmp/wheelhouse/urllib3-1.26.2-py2.py3-none-any.whl
- virtualenv==20.10.0
- wasabi==0.8.2
- wcwidth==0.2.5
- webencodings==0.5.1
- Werkzeug==1.0.1
- wordcloud==1.8.1
- wrapt==1.12.1
- xgboost==1.4.0
- xkit==0.0.0
- youtube-dl==2020.3.24
- sklearn 0.24.1
- python 3.8.5
- os 환경 : Window 10 Pro x64