한글 설치

- 한번 실행하고 런타임 다시 시작해서 다시 실행해야 적용됨.

```
import matplotlib.pyplot as plt


#한글 폰트 살리기 위함 -> 타이틀에만 반영되더라 ㅠ
from matplotlib import rc
from matplotlib import font_manager as fm


from matplotlib import rcParams
import matplotlib as mpl


#nanum 폰트 설치
!apt-get update -qq
!apt-get install fonts-nanum* -qq


sys_font=fm.findSystemFonts()
print(f"sys_font number: {len(sys_font)}")

nanum_font = [f for f in sys_font if 'Nanum' in f]
print(f"nanum_font number: {len(nanum_font)}")
```

```
sys_font number: 48
nanum_font number: 31
```

```
nanum_font
```

```
['/usr/share/fonts/truetype/nanum/NanumBarunGothicBold.ttf',
 '/usr/share/fonts/truetype/nanum/NanumMyeongjoEcoExtraBold.ttf',
 '/usr/share/fonts/truetype/nanum/NanumGothicCoding.ttf',
 '/usr/share/fonts/truetype/nanum/NanumBarunGothicUltraLight.ttf',
 '/usr/share/fonts/truetype/nanum/NanumGothic.ttf',
 '/usr/share/fonts/truetype/nanum/NanumGothicCoding-Bold.ttf',
 '/usr/share/fonts/truetype/nanum/NanumBarunGothicLight.ttf',
 '/usr/share/fonts/truetype/nanum/NanumSquareR.ttf',
 '/usr/share/fonts/truetype/nanum/NanumBrush.ttf',
 '/usr/share/fonts/truetype/nanum/NanumGothicBold.ttf',
```

```
!python --version
def current_font():
  print(f"설정 폰트 글꼴: {plt.rcParams['font.family']}, 설정 폰트 사이즈: {plt.rcParams['font.size

current_font()
```

```
Python 3.6.9
설정 폰트 글꼴: ['sans-serif'], 설정 폰트 사이즈: 10.0
```

```
path = '/usr/share/fonts/truetype/nanum/NanumBarunGothic.ttf'  # 설치된 나눔글꼴중 원하는 녀석의 전
#여기가 폰트 사이즈를 정하는 곳.
font_name = fm.FontProperties(fname=path, size=16).get_name()
print(font_name)
plt.rc('font', family=font_name)
```

```
NanumBarunGothic
```

```
fm._rebuild()
```

```
!pip install statsmodels
```

```
Requirement already satisfied: statsmodels in /usr/local/lib/python3.6/dist-packages (0.11.1)
Requirement already satisfied: numpy>=1.14 in /usr/local/lib/python3.6/dist-packages (from st
Requirement already satisfied: scipy>=1.0 in /usr/local/lib/python3.6/dist-packages (from sta
Requirement already satisfied: patsy>=0.5 in /usr/local/lib/python3.6/dist-packages (from sta
Requirement already satisfied: pandas>=0.21 in /usr/local/lib/python3.6/dist-packages (from s
Requirement already satisfied: six in /usr/local/lib/python3.6/dist-packages (from patsy>=0.5
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.6/dist-packages (from p
Requirement already satisfied: python-dateutil>=2.6.1 in /usr/local/lib/python3.6/dist-packag
```

```
!pip install pmdarima
```

```
Requirement already satisfied: pmdarima in /usr/local/lib/python3.6/dist-packages (1.7.1)
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.6/dist-packages (from p
Requirement already satisfied: statsmodels<0.12,>=0.11 in /usr/local/lib/python3.6/dist-packa
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.6/dist-packages (from p
```

```
!pip install tqdm
```

```
Requirement already satisfied: tqdm in /usr/local/lib/python3.6/dist-packages (4.41.1)
Requirement already satisfied: setuptools<50.0.0 in /usr/local/lib/python3.6/dist-packages (f
```

```
import numpy as np
import pandas as pd
import plotly.offline as pyo
import plotly.graph_objs as go
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import plotly.express as px
import statsmodels.api as sm
import re
import math
from sklearn.metrics import r2_score


from pmdarima.arima import auto_arima


import datetime
from dateutil.relativedelta import relativedelta

import statsmodels
import statsmodels.api as sm
from statsmodels.tsa.stattools import acf
from statsmodels.tsa.stattools import pacf
from statsmodels.tsa.seasonal import seasonal_decompose


import warnings
warnings.filterwarnings(action='ignore')


from tqdm.notebook import tqdm


loc_inf=pd.read_csv('/content/drive/My Drive/data/whole.csv',index_col=0)


loc_inf=loc_inf[['날짜','시구분', '구구분', '에볼라바이러스병', '마버그열', '라싸열', '크리미안콩고
       '리프트밸리열', '두창', '페스트', '탄저', '보툴리눔독소증', '야토병', '신종감염병증후군',
       '중증급성호흡기증후군(SARS)', '중동호흡기증후군(MERS)', '동물인플루엔자 인체감염증', '신종인
       '디프테리아', '수두', '홍역', '콜레라', '장티푸스', '파라티푸스', '세균성이질', '장출혈성대장
       'A형간염', '백일해', '유행성이하선염', '풍진(2018년이전)', '풍진(선천성)', '풍진(후천성)', '
       '수막구균 감염증', 'b형헤모필루스인플루엔자', '폐렴구균 감염증', '한센병', '성홍열',
       '반코마이신내성황색포도알균(VRSA) 감염증', '카바페넴내성장내세균속균종(CRE) 감염증', 'E형간염
       'B형간염', '일본뇌염', 'C형간염', '말라리아', '레지오넬라증', '비브리오패혈증', '발진티푸스'
       '쯔쯔가무시증', '렙토스피라증', '브루셀라증', '공수병', '신증후군출혈열',
       '크로이츠펠트-야콥병(CJD) 및 변종크로이츠펠트-야콥병(vCJD)', '황열', '뎅기열', '큐열', '웨스
       '라임병', '진드기매개뇌염', '유비저', '치쿤구니야열', '중증열성혈소판감소증후군(SFTS)', '지카
```

```
        prepro_pd['과거 5년 중앙값']=0
        prepro_pd['중앙값 초과 여부']=0
        prepro_pd['전처리 후 발생건수']=prepro_pd[virus]
        for i in prepro_pd.index:
          try:
            virus_5=[]
            date_5=[]
            for r in range(5,0,-1):
              for m in range(1,-2,-1):
                if prepro_pd.iloc[i]['월']==12:
                  if m != -1:
                    virus_5.append(prepro_pd[(prepro_pd['년']==(prepro_pd.iloc[i]['년']-r))&(prepro_pd[
                  else:
                    virus_5.append(prepro_pd[(prepro_pd['년']==(prepro_pd.iloc[i]['년']-r+1))&(prepro_p
                elif prepro_pd.iloc[i]['월']==1:
                  if m != 1:
                    virus_5.append(prepro_pd[(prepro_pd['년']==(prepro_pd.iloc[i]['년']-r))&(prepro_pd[
                  else:
                    virus_5.append(prepro_pd[(prepro_pd['년']==(prepro_pd.iloc[i]['년']-r-1))&(prepro_p
                else:
                  virus_5.append(prepro_pd[(prepro_pd['년']==(prepro_pd.iloc[i]['년']-r))&(prepro_pd['월
            prepro_pd['과거 5년 중앙값'][i]=np.median(virus_5)
            if (prepro_pd[virus][i]>np.median(virus_5))&(np.median(virus_5)!=0):
              prepro_pd['중앙값 초과 여부'][i]=1
              prepro_pd['전처리 후 발생건수'][i]=new_lin.predict(prepro_pd['날짜'][i])
          except:
            pass
        for i in prepro_pd.index:
          if prepro_pd['중앙값 초과 여부'][i]==1:
            try:
              month_5=[]
              month_date5=[]
              for r in range(5,0,-1):
                  month_5.append(prepro_pd[(prepro_pd['년']==(prepro_pd.iloc[i]['년']-r))&(prepro_pd['월
                  month_date5.append((prepro_pd.iloc[i]['년']-r)*100+(prepro_pd.iloc[i]['월']))
              new_lin = sm.OLS(month_5,month_date5).fit()
              if new_lin.predict(prepro_pd['날짜'][i])<prepro_pd[virus][i]:
                prepro_pd['전처리 후 발생건수'][i]=new_lin.predict(prepro_pd['날짜'][i])
              else:
                prepro_pd['전처리 후 발생건수'][i]=prepro_pd['과거 5년 중앙값'][i]
            except:
              pass
        return prepro_pd

      def arima_predict(self,data,virus,scope):
        y_forec_list=[]
        conf_list=[]
        ari_data=self.preprocessing_pd(data[['날짜',virus]],virus)
        print(ari_data)
        if scope=='all':
          for i in ari_data.index:
            try:
              train=ari_data.iloc[i-60:i][['날짜','전처리 후 발생건수']]
              train['날짜']= train['날짜'].apply(lambda x : str(x)[:4]+'-'+str(x)[4:])
              train.index=pd.to_datetime(train['날짜'])
```

```
            train=train[['전처리 후 발생건수']]
            arima_model=auto_arima(train, start_p=0, d=0, start_q=0,
                                   max_p=3, max_d=3, max_q=3, start_P=0, D=1,
                                   start_Q=0, max_P=3, max_D=1,
                                   max_Q=3,m=12,seasonal=True,
                                   error_action='warn',trace=False,
                                   supress_warnings=True, stepwise=True,
                                   random_state=20, n_fits=50)
            y_forec, conf_int = arima_model.predict(n_periods=1,return_conf_int=True,alpha=0.05)
            y_forec_list.append([y_forec[0],ari_data.iloc[i]['날짜']])
            conf_list.append([conf_int[0][0],conf_int[0][1],ari_data.iloc[i]['날짜']])
            if i == ari_data.index[-1]:
              print(arima_model.summary())
          except:
            pass
        elif scope == 'recent':
            i=ari_data.index[-1]
            train=ari_data.iloc[i-60:i][['날짜','전처리 후 발생건수']]
            train['날짜']= train['날짜'].apply(lambda x : str(x)[:4]+'-'+str(x)[4:])
            train.index=pd.to_datetime(train['날짜'])
            train=train[['전처리 후 발생건수']]
            arima_model=auto_arima(train, start_p=0, d=0, start_q=0,
                                   max_p=3, max_d=3, max_q=3, start_P=0, D=1,
                                   start_Q=0, max_P=3, max_D=1,
                                   max_Q=3,m=12,seasonal=True,
                                   error_action='warn',trace=False,
                                   supress_warnings=True, stepwise=True,
                                   random_state=20, n_fits=50)
            y_forec, conf_int = arima_model.predict(n_periods=1,return_conf_int=True,alpha=0.05)
            y_forec_list.append([y_forec[0],ari_data.iloc[i]['날짜']])
            conf_list.append([conf_int[0][0],conf_int[0][1],ari_data.iloc[i]['날짜']])
        prediction=pd.DataFrame(y_forec_list,columns=['predict_virus','날짜'])
        conf_lv=pd.DataFrame(conf_list,columns=['Confidence Lower','Confidence Upper','날짜'])
        conf_lv['날짜']= conf_lv['날짜'].apply(lambda x : str(int(x))[:4]+'-'+str(int(x))[4:])
        prediction['날짜']= prediction['날짜'].apply(lambda x : str(int(x))[:4]+'-'+str(int(x))[4:])
        conf_lv.index=pd.to_datetime(conf_lv['날짜'])
        prediction.index=pd.to_datetime(prediction['날짜'])
        conf_lv=conf_lv[['Confidence Lower','Confidence Upper']]
        prediction=prediction[['predict_virus']]
        return prediction, conf_lv

    def regression_predict(self,data,virus,scope):
        reg_pd=self.preprocessing_pd(data[['날짜',virus]],virus)
        reg_pd['c1']=np.cos(2*math.pi*reg_pd.index/12)
        reg_pd['d1']=np.sin(2*math.pi*reg_pd.index/12)
        reg_pd['c2']=np.cos(4*math.pi*reg_pd.index/12)
        reg_pd['d2']=np.sin(4*math.pi*reg_pd.index/12)
        reg_pd_train=sm.add_constant(reg_pd.reset_index()[['index','c1','d1','c2','d2']])
        reg_pd['predict_virus']=0
        reg_pd['obs_ci_lower']=0
        reg_pd['obs_ci_upper']=0
        if scope=='all':
          for i in reg_pd.index:
            try:
              pre_lin = sm.OLS(reg_pd[i-60:i]['전처리 후 발생건수'],reg_pd_train[i-60:i]).fit()
```

```
            pre_pd=pre_lin.get_prediction(reg_pd_train.iloc[i:i+1,:]).summary_frame(alpha = 0.05)
            reg_pd['predict_virus'][i]=pre_pd['mean'].values[0]
            reg_pd['obs_ci_lower'][i]=pre_pd['obs_ci_lower'].values[0]
            reg_pd['obs_ci_upper'][i]=pre_pd['obs_ci_upper'].values[0]
            if i == reg_pd.index[-1]:
              print(pre_lin.summary())
          except:
            pass
      elif scope=='recent':
        i=reg_pd.index[-1]
        pre_lin = sm.OLS(reg_pd[i-60:i]['전처리 후 발생건수'],reg_pd_train[i-60:i]).fit()
        pre_pd=pre_lin.get_prediction(reg_pd_train.iloc[i:i+1,:]).summary_frame(alpha = 0.05)
        reg_pd['predict_virus'][i]=pre_pd['mean'].values[0]
        reg_pd['obs_ci_lower'][i]=pre_pd['obs_ci_lower'].values[0]
        reg_pd['obs_ci_upper'][i]=pre_pd['obs_ci_upper'].values[0]
      reg_pd['날짜']=reg_pd['날짜'].apply(lambda x : str(x)[:4]+'-'+str(x)[4:])
      reg_pd.index=pd.to_datetime(reg_pd['날짜'])
      return reg_pd

    def personal_caution(self):
      for virus in self.virus_list:
        personal_virus=self.virus_pd(self.residence_si,self.residence_gu,virus)
        print('{}_{}_{}'.format(self.residence_gu,virus,personal_virus[virus].iloc[-1]))
        if personal_virus[virus].iloc[-1]>1:
          arima_pred, arima_conf=self.arima_predict(personal_virus,virus,'all')
          reg_pred=self.regression_predict(personal_virus,virus,'all')
          personal_virus['날짜']=personal_virus['날짜'].apply(lambda x : str(x)[:4]+'-'+str(x)[4:])
          personal_virus.index=pd.to_datetime(personal_virus['날짜'])
          print(personal_virus)
          print('--------------------')
          print(arima_pred)
          print('--------------------')
          print(arima_conf)
          print('--------------------')
          print(reg_pred)
          print('====================')
          plt.figure(figsize=(14,6))
          plt.plot(personal_virus.index,personal_virus[[virus]],linestyle = "None", marker = "o",mark
          plt.plot(arima_conf.index,arima_pred, color = "red")
          plt.plot(arima_conf.index,arima_conf['Confidence Upper'], color = "blue", linestyle = "--")
          plt.plot(arima_conf.index,arima_conf['Confidence Lower'], color = "blue", linestyle = "--")
          plt.legend(['발생건수','arima_predict','arima_conf_upper','arima_conf_lower'],prop={'size':
          if self.residence_si=='전국':
            plt.title('{}_{} arima_time_series'.format(self.residence_si,virus),fontsize=20)
          else:
            plt.title('{} {}_{} arima_time_series'.format(self.residence_si,self.residence_gu,virus),
          plt.show()
          if arima_conf['Confidence Upper'].iloc[-1]<personal_virus[virus].iloc[-1]:
            print('{} {}_{} 유행'.format(self.residence_si,self.residence_gu,virus))
          plt.figure(figsize=(14, 6))
          plt.plot(reg_pred.index, reg_pred[virus], linestyle = "None", marker = "o",markerfacecolor
          plt.plot(reg_pred.index, reg_pred['predict_virus'], color = "red")
          plt.plot(reg_pred.index, reg_pred['obs_ci_upper'], color = "blue", linestyle = "--")
          plt.plot(reg_pred.index, reg_pred['obs_ci_lower'], color = "blue", linestyle = "--")
          plt.legend(['발생건수','reg_predict','reg_conf_upper','reg_conf_lower'],prop={'size':10})
          if self.residence_si=='전국':
```

```
loc_inf
```

	날짜	시구분	구구분	에볼라바이러스병	마버그열	라싸열	크리미안콩고출혈열	남아메리카출혈열	리프트밸리열	두창	페스트	탄저	보툴리눔독소증	야토병	신종감염병증후군	급성호흡기증후군(SARS)	중동호흡기증후군(MERS)	동물인플루엔자인체감염증
0	201001	전국	전국	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	201001	서울	서울	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	201001	서울	강남구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	201001	서울	강동구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	201001	서울	강북구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
285	202007	제주	제주시	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
286	202007	제주	남제주군	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
287	202007	제주	북제주군	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
288	202007	세종	세종	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
289	202007	세종	세종시	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

36830 rows × 67 columns

```
nowon_su=loc_inf[(loc_inf['시구분']=='서울')&(loc_inf['구구분']=='노원구')]
```

```
        if self.residence_si=='전국':
          plt.title('{}_{} reg_time_series'.format(self.residence_si,virus),fontsize=20)
        else:
          plt.title('{} {}_{} reg_time_series'.format(self.residence_si,self.residence_gu,virus),fo
        plt.show()
        if reg_pred['obs_ci_upper'].iloc[-1]<personal_virus[virus].iloc[-1]:
          print('{} {}_{} 유행'.format(self.residence_si,self.residence_gu,virus))

    def destination_caution(self,destination_si,destination_gu):
      for virus in self.virus_list:
        destination_virus=self.virus_pd(destination_si,destination_gu,virus)
        print('{}_{}_{}'.format(destination_gu,virus,destination_virus[virus].iloc[-1]))
        if destination_virus[virus].iloc[-1]>1:
          arima_pred, arima_conf=self.arima_predict(destination_virus,virus,'all')
          reg_pred=self.regression_predict(destination_virus,virus,'all')
          destination_virus['날짜']=destination_virus['날짜'].apply(lambda x : str(x)[:4]+'-'+str(x)[
          destination_virus.index=pd.to_datetime(destination_virus['날짜'])
          print(destination_virus)
          print('--------------------')
          print(arima_pred)
          print('--------------------')
          print(arima_conf)
          print('--------------------')
          print(reg_pred)
          print('====================')
          plt.figure(figsize=(14,6))
          plt.plot(destination_virus.index,destination_virus[[virus]],linestyle = "None", marker = "o
          plt.plot(arima_conf.index,arima_pred, color = "red")
          plt.plot(arima_conf.index,arima_conf['Confidence Upper'], color = "blue", linestyle = "--")
          plt.plot(arima_conf.index,arima_conf['Confidence Lower'], color = "blue", linestyle = "--")
          plt.legend(['발생건수','arima_predict','arima_conf_upper','arima_conf_lower'],prop={'size':
          if destination_si=='전국':
            plt.title('{}_{} arima_time_series'.format(destination_si,virus),fontsize=20)
          else:
            plt.title('{} {}_{} arima_time_series'.format(destination_si,destination_gu,virus),fontsi
          plt.show()
          if arima_conf['Confidence Upper'].iloc[-1]<destination_virus[virus].iloc[-1]:
            print('{} {}_{} 유행'.format(destination_si,destination_gu,virus))
          plt.figure(figsize=(14, 6))
          plt.plot(reg_pred.index, reg_pred[virus], linestyle = "None", marker = "o",markerfacecolor
          plt.plot(reg_pred.index, reg_pred['predict_virus'], color = "red")
          plt.plot(reg_pred.index, reg_pred['obs_ci_upper'], color = "blue", linestyle = "--")
          plt.plot(reg_pred.index, reg_pred['obs_ci_lower'], color = "blue", linestyle = "--")
          plt.legend(['발생건수','reg_predict','reg_conf_upper','reg_conf_lower'],prop={'size':10})
          if destination_si=='전국':
            plt.title('{}_{} reg_time_series'.format(destination_si, virus),fontsize=20)
          else:
            plt.title('{} {}_{} reg_time_series'.format(destination_si,destination_gu, virus),fontsiz
          plt.show()
          if reg_pred['obs_ci_upper'].iloc[-1]<destination_virus[virus].iloc[-1]:
            print('{} {}_{} 유행'.format(destination_si,destination_gu,virus))

    def epidemic_now(self):
      epidemic_list=[]
      for virus_si in tqdm(self.si_list):
        for virus_gu in tqdm(self.loc_inf[self.loc_inf['시구분'] == virus_si]['구구분'].unique()):
```

```
        for virus_gu in tqdm(self.loc_inf[self.loc_inf['시구분']==virus_si]['구구분'].unique()):
          for virus in self.virus_list:
            try:
              virus_sigu=self.virus_pd(virus_si,virus_gu,virus)
              if virus_sigu[virus].iloc[-1]>1:
                arima_pred, arima_conf=self.arima_predict(virus_sigu,virus,'recent')
                reg_pred=self.regression_predict(virus_sigu,virus,'recent')
                if (arima_conf['Confidence Upper'].iloc[-1]<virus_sigu[virus].iloc[-1])|(reg_pred['ob
                  epidemic_list.append([virus_si,virus_gu,virus_sigu['날짜'].iloc[-1],virus,virus_sig
                                        arima_pred['predict_virus'].iloc[-1],arima_conf['Confidence U
                                        reg_pred['predict_virus'].iloc[-1],reg_pred['obs_ci_upper'].i
                else:
                  epidemic_list.append([virus_si,virus_gu,virus_sigu['날짜'].iloc[-1],virus,virus_sig
                                        arima_pred['predict_virus'].iloc[-1],arima_conf['Confidence U
                                        reg_pred['predict_virus'].iloc[-1],reg_pred['obs_ci_upper'].i
              else:
                epidemic_list.append([virus_si,virus_gu,virus_sigu['날짜'].iloc[-1],virus,virus_sigu[
                                      0,0,0,0,0])
            except:
              pass
    epidemic_pd=pd.DataFrame(epidemic_list,columns=['시구분','구구분','날짜','감염병명','발생건수',
    return epidemic_pd


personal=pd.DataFrame([['서울','노원구',20]],columns=['residence_si','residence_gu','age'])


test_caution=pre_caution(loc_inf)


no_test=test_caution.virus_pd('서울','노원구','A형간염')


no_test2=test_caution.preprocessing_pd(no_test,'A형간염')


no_test2
```

nowon_su

	날짜	시구분	구구분	에볼라바이러스병	마버그열	라싸열	크리미안콩고출혈열	남아메리카출혈열	리프트밸리열	두창	페스트	탄저	보툴리눔독소증	야토병	신종감염병증후군	급성호흡기증후군(SARS)	동물인플루엔자인체감염증 중동호흡기증후군(MERS)	호흡기증후군
10	201001	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	201002	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	201003	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	201004	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	201005	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
10	202003	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	202004	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	202005	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	202006	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	202007	서울	노원구	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

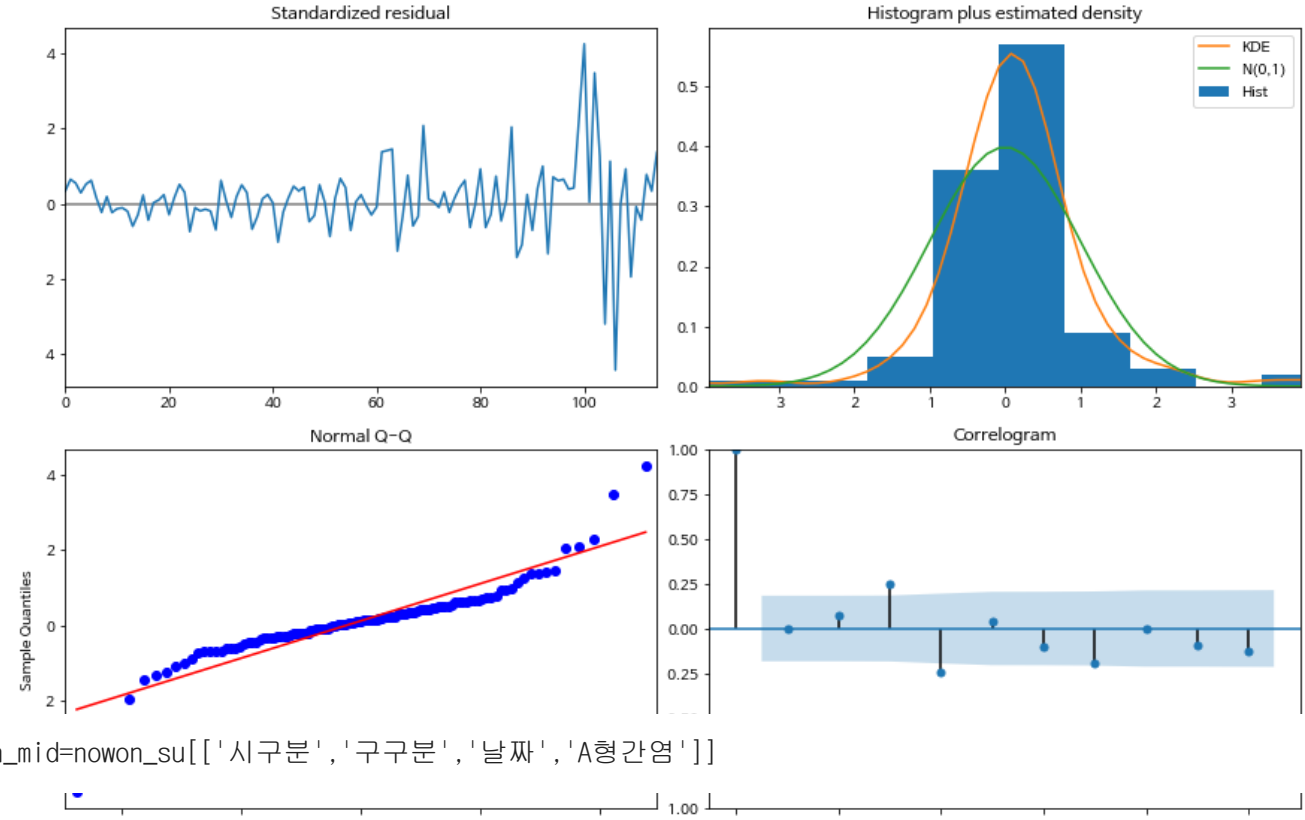
127 rows × 67 columns

```
arima_model=auto_arima(nowon_su['A형간염'], start_p=0, d=0, start_q=0,
                           max_p=3, max_d=3, max_q=3, start_P=0, D=1,
                           start_Q=0, max_P=3, max_D=1,
                           max_Q=3,m=12,seasonal=True,
                           error_action='warn',trace=False,
                           supress_warnings=True, stepwise=True,
                           random_state=20, n_fits=50)
```

```
mod = sm.tsa.SARIMAX(nowon_su['A형간염'],order=(2,0,0), seasonal_order=(0,1,[1,2],12))
results = mod.fit()
print (results.summary())
```

```
                                     SARIMAX Results
==========================================================================================
Dep. Variable:                                 A형간염   No. Observations:
Model:             SARIMAX(2, 0, 0)x(0, 1, [1, 2], 12)   Log Likelihood                -279.9
Date:                                 Mon, 28 Sep 2020   AIC                            569.9
Time:                                         05:39:39   BIC                            583.6
Sample:                                             0   HQIC                           575.5
                                                  - 127
Covariance Type:                                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.6807      0.073      9.369      0.000       0.538       0.823
ar.L2          0.1671      0.078      2.149      0.032       0.015       0.319
ma.S.L12      -1.0388      0.133     -7.821      0.000      -1.299      -0.778
ma.S.L24       0.2694      0.152      1.778      0.075      -0.028       0.566
sigma2         6.6431      0.614     10.821      0.000       5.440       7.846
===================================================================================
Ljung-Box (Q):                       48.69   Jarque-Bera (JB):               216.80
Prob(Q):                              0.16   Prob(JB):                         0.00
Heteroskedasticity (H):              16.11   Skew:                            -0.09
Prob(H) (two-sided):                  0.00   Kurtosis:                         9.72
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

```
results.plot_diagnostics(figsize=(12,8))
plt.tight_layout(pad=0.4, w_pad=0.5, h_pad=1.0)
```




```
nowon_mid=nowon_su[['시구분','구구분','날짜','A형간염']]
```

```
nowon_mid
```

	시구분	구구분	날짜	A형간염
10	서울	노원구	201001	0
10	서울	노원구	201002	0
10	서울	노원구	201003	0
10	서울	노원구	201004	0
10	서울	노원구	201005	0
...	...	...	...	...
10	서울	노원구	202003	3
10	서울	노원구	202004	1
10	서울	노원구	202005	2
10	서울	노원구	202006	3
10	서울	노원구	202007	6

127 rows × 4 columns

```
nowon_mid[((nowon_mid['날짜']%100==7)|(nowon_mid['날짜']%100==6)|(nowon_mid['날짜']%100==8))&(nowon
```

	시구분	구구분	날짜	A형간염
0	서울	노원구	201506	3
1	서울	노원구	201507	4
2	서울	노원구	201508	1
3	서울	노원구	201606	4
4	서울	노원구	201607	6
5	서울	노원구	201608	3
6	서울	노원구	201706	6
7	서울	노원구	201707	4
8	서울	노원구	201708	3
9	서울	노원구	201806	2
10	서울	노원구	201807	1
11	서울	노원구	201808	1
12	서울	노원구	201906	17
13	서울	노원구	201907	25

```
no_07mid=np.median(nowon_mid[((nowon_mid['날짜']%100==7)|(nowon_mid['날짜']%100==6)|(nowon_mid['날짜
```

15 서울 노원구 202006 3

```
pd.DataFrame([['서울','노원구',202007,6,no_07mid]],columns=['시구분','구구분','날짜','A형간염','과거
```

	시구분	구구분	날짜	A형간염	과거 5년 중앙값
0	서울	노원구	202007	6	4.0

```
class pre_caution():

  def __init__(self,loc_inf):
    self.loc_inf=loc_inf
    self.virus_list=loc_inf.columns[3:].tolist()
    self.si_list=loc_inf['시구분'].unique().tolist()
    self.residence_si=personal.residence_si[0]
    self.residence_gu=personal.residence_gu[0]
    self.age=personal.age[0]

  def virus_pd(self,loc_si,loc_gu,virus):
    loc_virus=self.loc_inf[['구구분','날짜',virus]][(self.loc_inf['구구분']==loc_gu)&(self.loc_inf[
    loc_virus[virus] = loc_virus[virus].apply(lambda x : float(x))
    return loc_virus

  def preprocessing_pd(self,raw_data,virus):
    prepro_pd=raw_data[['날짜',virus]]
    prepro_pd['년']=prepro_pd['날짜'].apply(lambda x : x//100)
    prepro_pd['월']=prepro_pd['날짜'].apply(lambda x : x%100)
    prepro_pd=prepro_pd.reset_index(drop=True)
```

날짜	A형간염	년	월	과거 5년 중앙값	중앙값 초과 여부	전처리 후 발생건수

```
no_test2[no_test2['중앙값 초과 여부']==1]
```

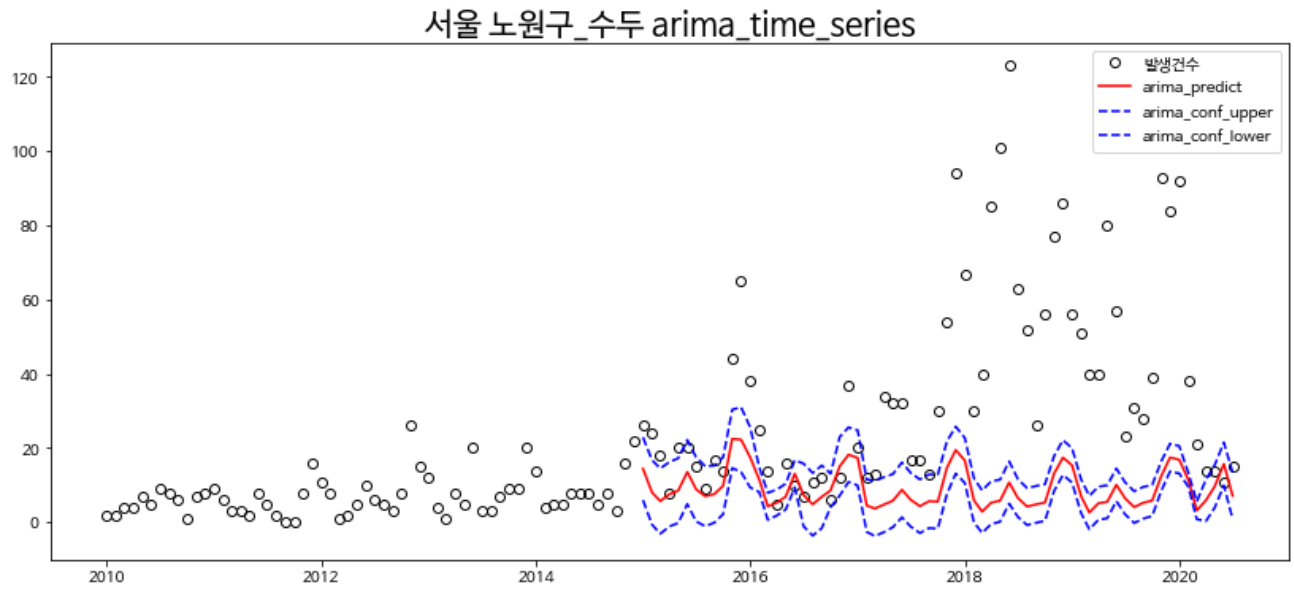
	날짜	A형간염	년	월	과거 5년 중앙값	중앙값 초과 여부	전처리 후 발생건수
62	201503	2.0	2015	3	1	1	1.401888
65	201506	3.0	2015	6	1	1	1.601688
66	201507	4.0	2015	7	1	1	1.802285
73	201602	4.0	2016	2	1	1	1.401688
74	201603	7.0	2016	3	1	1	1.682264
75	201604	10.0	2016	4	1	1	1.801786
76	201605	5.0	2016	5	1	1	2.002184
77	201606	4.0	2016	6	1	1	1.922025
78	201607	6.0	2016	7	1	1	2.162741
79	201608	3.0	2016	8	1	1	1.401389
80	201609	2.0	2016	9	1	1	1.201290
81	201610	7.0	2016	10	1	1	0.801092
82	201611	6.0	2016	11	1	1	1.201887
83	201612	5.0	2016	12	1	1	0.400496

```
test_caution.personal_caution()
```

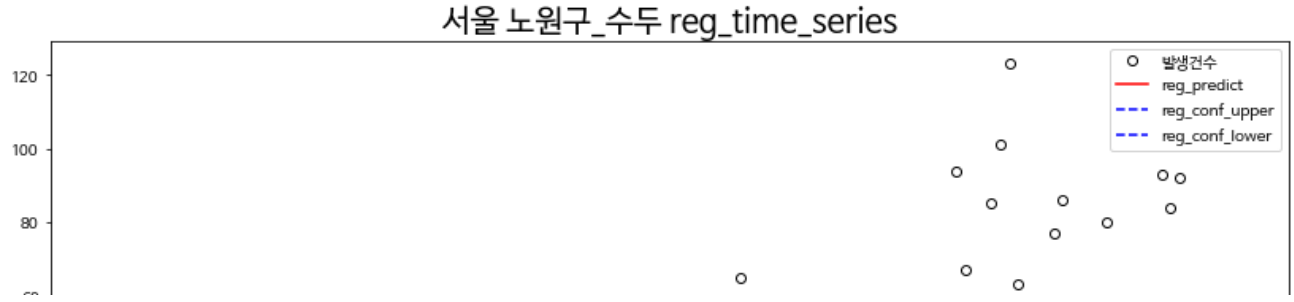
```
노원구_에볼라바이러스병_0.0
노원구_마버그열_0.0
노원구_라싸열_0.0
노원구_크리미안콩고출혈열_0.0
노원구_남아메리카출혈열_0.0
노원구_리프트밸리열_0.0
노원구_두창_0.0
노원구_페스트_0.0
노원구_탄저_0.0
노원구_보툴리눔독소증_0.0
노원구_야토병_0.0
노원구_신종감염병증후군_0.0
노원구_중증급성호흡기증후군(SARS)_0.0
노원구_중동호흡기증후군(MERS)_0.0
노원구_동물인플루엔자 인체감염증_0.0
노원구_신종인플루엔자_0.0
노원구_디프테리아_0.0
노원구_수두_15.0
        날짜    수두     년  월  과거 5년 중앙값  중앙값 초과 여부  전처리 후 발생건수
0    201001   2.0  2010  1          0          0         2.0
1    201002   2.0  2010  2          0          0         2.0
2    201003   4.0  2010  3          0          0         4.0
3    201004   4.0  2010  4          0          0         4.0
4    201005   7.0  2010  5          0          0         7.0
..      ...   ...   ... ..        ...        ...         ...
122  202003  21.0  2020  3         25          0        21.0
123  202004  14.0  2020  4         32          0        14.0
124  202005  14.0  2020  5         32          0        14.0
125  202006  11.0  2020  6         23          0        11.0
126  202007  15.0  2020  7         20          0        15.0

[127 rows x 7 columns]
                                SARIMAX Results
==========================================================================================
Dep. Variable:                                 y   No. Observations:                   60
Model:             SARIMAX(1, 0, 0)x(0, 1, 0, 12)   Log Likelihood                -123.895
Date:                            Mon, 28 Sep 2020   AIC                            251.790
Time:                                    02:24:27   BIC                            255.532
Sample:                                         0   HQIC                           253.204
                                              - 60
Covariance Type:                              opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.4107      0.172      2.382      0.017       0.073       0.749
sigma2        10.1813      0.677     15.028      0.000       8.854      11.509
===================================================================================
Ljung-Box (Q):                       43.43   Jarque-Bera (JB):                608.27
Prob(Q):                              0.33   Prob(JB):                         0.00
Heteroskedasticity (H):               4.56   Skew:                             3.22
Prob(H) (two-sided):                  0.00   Kurtosis:                        19.21
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
                            OLS Regression Results
==============================================================================
Dep. Variable:              전처리 후 발생건수   R-squared:                       0.706
```

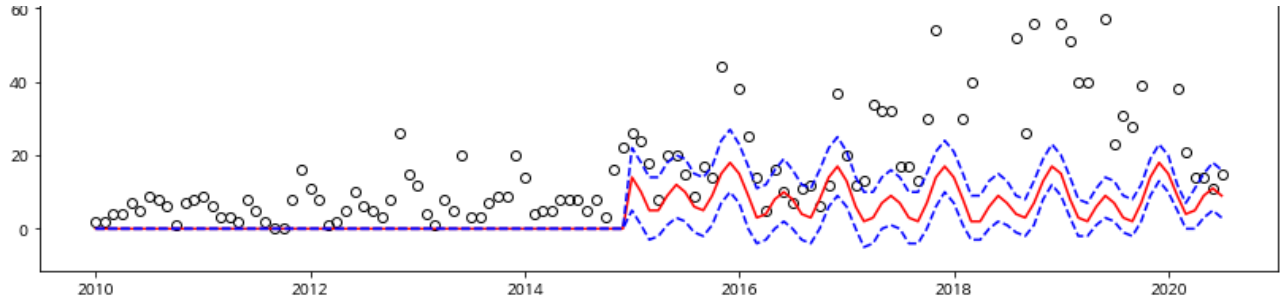
```
Dep. Variable:                      y   R-squared:                       0.705
Model:                            OLS   Adj. R-squared:                  0.679
Method:                 Least Squares   F-statistic:                     25.94
Date:                Mon, 28 Sep 2020   Prob (F-statistic):           3.02e-13
Time:                        02:24:30   Log-Likelihood:                -146.73
No. Observations:                  60   AIC:                             305.5
Df Residuals:                      54   BIC:                             318.0
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          2.3899      2.162      1.105      0.274      -1.945       6.725
index          0.0688      0.022      3.085      0.003       0.024       0.113
c1             3.7387      0.538      6.954      0.000       2.661       4.817
d1            -1.1299      0.544     -2.078      0.042      -2.220      -0.040
c2             2.4306      0.538      4.521      0.000       1.353       3.509
d2            -3.6020      0.539     -6.688      0.000      -4.682      -2.522
==============================================================================
Omnibus:                       43.397   Durbin-Watson:                   1.710
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              188.927
Skew:                           1.942   Prob(JB):                     9.44e-42
Kurtosis:                      10.777   Cond. No.                         553.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
           구구분       날짜    수두
날짜
2010-01-01  노원구  2010-01   2.0
2010-02-01  노원구  2010-02   2.0
2010-03-01  노원구  2010-03   4.0
2010-04-01  노원구  2010-04   4.0
2010-05-01  노원구  2010-05   7.0
...         ...      ...   ...
2020-03-01  노원구  2020-03  21.0
2020-04-01  노원구  2020-04  14.0
2020-05-01  노원구  2020-05  14.0
2020-06-01  노원구  2020-06  11.0
2020-07-01  노원구  2020-07  15.0

[127 rows x 3 columns]
------------------
           predict_virus
날짜
2015-01-01      14.465359
2015-02-01       8.058310
2015-03-01       5.726973
2015-04-01       7.582990
2015-05-01       8.553728
...                   ...
2020-03-01       3.204339
2020-04-01       5.935513
2020-05-01       9.696193
2020-06-01      15.746907
2020-07-01       7.161469

[67 rows x 1 columns]
--------------------
```

```
            Confidence Lower  Confidence Upper
날짜
2015-01-01          6.000069         22.930649
2015-02-01         -0.628063         16.744683
2015-03-01         -3.079086         14.533031
2015-04-01         -1.107697         16.273676
2015-05-01         -0.163276         17.270733
...                      ...               ...
2020-03-01          0.840739          5.567940
2020-04-01          0.309666         11.561361
2020-05-01          3.957707         15.434679
2020-06-01          9.939986         21.553828
2020-07-01          0.907573         13.415365

[67 rows x 2 columns]
--------------------
                날짜    수두     년  ...  predict_virus  obs_ci_lower  obs_ci_upper
날짜                                  ...
2010-01-01  2010-01   2.0  2010  ...              0             0             0
2010-02-01  2010-02   2.0  2010  ...              0             0             0
2010-03-01  2010-03   4.0  2010  ...              0             0             0
2010-04-01  2010-04   4.0  2010  ...              0             0             0
2010-05-01  2010-05   7.0  2010  ...              0             0             0
...             ...   ...   ...  ...            ...           ...           ...
2020-03-01  2020-03  21.0  2020  ...              4             0             7
2020-04-01  2020-04  14.0  2020  ...              5             0            11
2020-05-01  2020-05  14.0  2020  ...              9             3            15
2020-06-01  2020-06  11.0  2020  ...             11             5            18
2020-07-01  2020-07  15.0  2020  ...              9             3            16

[127 rows x 14 columns]
====================
```



서울 노원구_수두 유행





노원구_홍역_0.0
노원구_콜레라_0.0
노원구_장티푸스_0.0
노원구_파라티푸스_0.0
노원구_세균성이질_0.0
노원구_장출혈성대장균감염증_2.0

	날짜	장출혈성대장균감염증	년	월	과거 5년 중앙값	중앙값 초과 여부	전처리 후
0	201001	0.0	2010	1	0	0	0.0
1	201002	0.0	2010	2	0	0	0.0
2	201003	0.0	2010	3	0	0	0.0
3	201004	0.0	2010	4	0	0	0.0
4	201005	0.0	2010	5	0	0	0.0
..	...	...	...	..	...	...	...
122	202003	0.0	2020	3	0	0	0.0
123	202004	0.0	2020	4	0	0	0.0
124	202005	0.0	2020	5	0	0	0.0
125	202006	1.0	2020	6	0	0	1.0
126	202007	2.0	2020	7	0	0	2.0

[127 rows x 7 columns]

SARIMAX Results

Dep. Variable:	y	No. Observations:	60
Model:	SARIMAX(0, 0, 1)x(0, 1, [], 12)	Log Likelihood	-34.797
Date:	Mon, 28 Sep 2020	AIC	75.594
Time:	02:38:40	BIC	81.208
Sample:	0	HQIC	77.715
	- 60		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
intercept	0.1205	0.055	2.184	0.029	0.012	0.229
ma.L1	-0.3412	0.151	-2.261	0.024	-0.637	-0.045
sigma2	0.2489	0.036	6.943	0.000	0.179	0.319

Ljung-Box (Q):	29.86	Jarque-Bera (JB):	33.66
Prob(Q):	0.88	Prob(JB):	0.00
Heteroskedasticity (H):	2.49	Skew:	1.12
Prob(H) (two-sided):	0.08	Kurtosis:	6.44

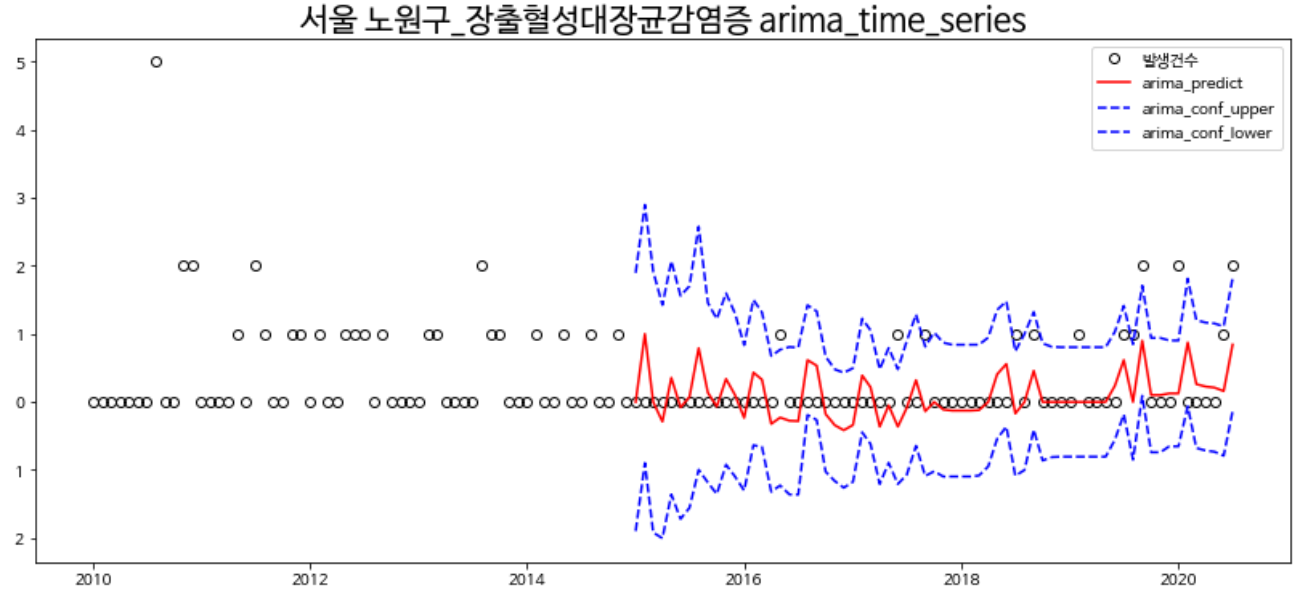
Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).

OLS Regression Results

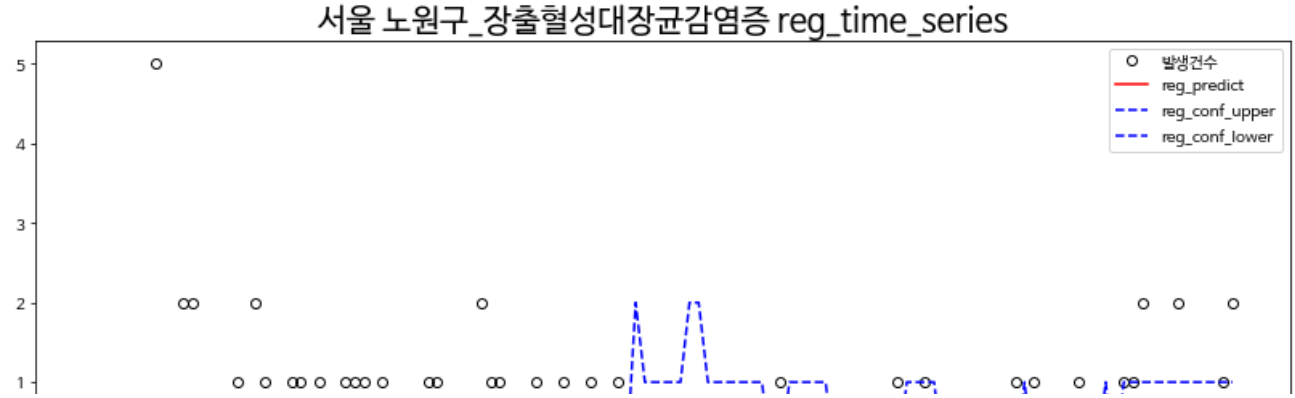
Dep. Variable:	전처리 후 발생건수	R-squared:	0.216
Model:	OLS	Adj. R-squared:	0.144
Method:	Least Squares	F-statistic:	2.979
Date:	Mon, 28 Sep 2020	Prob (F-statistic):	0.0190

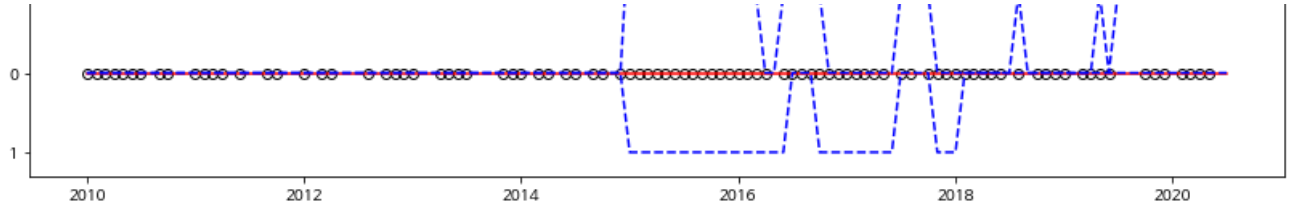
```
Time:                        02:38:43   Log-Likelihood:                -34.559
No. Observations:                60   AIC:                             81.12
Df Residuals:                    54   BIC:                             93.68
Df Model:                         5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         -0.7407      0.333     -2.221      0.031      -1.409      -0.072
index          0.0100      0.003      2.917      0.005       0.003       0.017
c1            -0.1511      0.083     -1.822      0.074      -0.317       0.015
d1            -0.0907      0.084     -1.082      0.284      -0.259       0.077
c2             0.1267      0.083      1.528      0.132      -0.040       0.293
d2             0.1040      0.083      1.252      0.216      -0.063       0.270
==============================================================================
Omnibus:                       30.362   Durbin-Watson:                   2.312
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               56.050
Skew:                           1.729   Prob(JB):                     6.74e-13
Kurtosis:                       6.234   Cond. No.                         553.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
            구구분        날짜  장출혈성대장균감염증
날짜
2010-01-01  노원구  2010-01         0.0
2010-02-01  노원구  2010-02         0.0
2010-03-01  노원구  2010-03         0.0
2010-04-01  노원구  2010-04         0.0
2010-05-01  노원구  2010-05         0.0
...         ...      ...         ...
2020-03-01  노원구  2020-03         0.0
2020-04-01  노원구  2020-04         0.0
2020-05-01  노원구  2020-05         0.0
2020-06-01  노원구  2020-06         1.0
2020-07-01  노원구  2020-07         2.0

[127 rows x 3 columns]
--------------------
            predict_virus
날짜
2015-01-01       0.000000
2015-02-01       1.000000
2015-03-01       0.000000
2015-04-01      -0.289008
2015-05-01       0.355599
...                   ...
2020-03-01       0.262396
2020-04-01       0.226573
2020-05-01       0.210888
2020-06-01       0.160320
2020-07-01       0.843203

[67 rows x 1 columns]
--------------------
            Confidence Lower  Confidence Upper
날짜
2015-01-01         -1.897718          1.897718
```

```
2015-02-01        -0.897718        2.897718
2015-03-01        -1.918688        1.918688
2015-04-01        -2.004078        1.426061
2015-05-01        -1.359516        2.070715
...                     ...             ...
2020-03-01        -0.676986        1.201777
2020-04-01        -0.714085        1.167231
2020-05-01        -0.731976        1.153751
2020-06-01        -0.790570        1.111211
2020-07-01        -0.134672        1.821077

[67 rows x 2 columns]
--------------------
                날짜  장출혈성대장균감염증  ...  obs_ci_lower  obs_ci_upper
날짜                                  ...
2010-01-01  2010-01         0.0  ...             0             0
2010-02-01  2010-02         0.0  ...             0             0
2010-03-01  2010-03         0.0  ...             0             0
2010-04-01  2010-04         0.0  ...             0             0
2010-05-01  2010-05         0.0  ...             0             0
...            ...         ...  ...           ...           ...
2020-03-01  2020-03         0.0  ...             0             1
2020-04-01  2020-04         0.0  ...             0             1
2020-05-01  2020-05         0.0  ...             0             1
2020-06-01  2020-06         1.0  ...             0             1
2020-07-01  2020-07         2.0  ...             0             1

[127 rows x 14 columns]
==================
```



서울 노원구_장출혈성대장균감염증 유행





서울 노원구_장출혈성대장균감염증 유행
노원구_A형간염_6.0

	날짜	A형간염	년	월	과거 5년 중앙값	중앙값 초과 여부	전처리 후 발생건수
0	201001	0.0	2010	1	0	0	0.000000
1	201002	0.0	2010	2	0	0	0.000000
2	201003	0.0	2010	3	0	0	0.000000
3	201004	0.0	2010	4	0	0	0.000000
4	201005	0.0	2010	5	0	0	0.000000
..	...	...	...	..	...	...	...
122	202003	3.0	2020	3	4	0	3.000000
123	202004	1.0	2020	4	5	0	1.000000
124	202005	2.0	2020	5	4	0	2.000000
125	202006	3.0	2020	6	4	0	3.000000
126	202007	6.0	2020	7	4	1	1.528936

[127 rows x 7 columns]

SARIMAX Results

Dep. Variable:	y	No. Observations:	60
Model:	SARIMAX(3, 1, 0, 12)	Log Likelihood	-53.966
Date:	Mon, 28 Sep 2020	AIC	115.932
Time:	03:03:56	BIC	123.417
Sample:	0	HQIC	118.761
	- 60		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
ar.S.L12	-0.9284	0.215	-4.325	0.000	-1.349	-0.508
ar.S.L24	-0.6359	0.302	-2.107	0.035	-1.227	-0.044
ar.S.L36	-0.4232	0.361	-1.173	0.241	-1.130	0.284
sigma2	0.4048	0.136	2.982	0.003	0.139	0.671

Ljung-Box (Q):	36.20	Jarque-Bera (JB):	127.42
Prob(Q):	0.64	Prob(JB):	0.00
Heteroskedasticity (H):	0.68	Skew:	2.08
Prob(H) (two-sided):	0.45	Kurtosis:	9.81

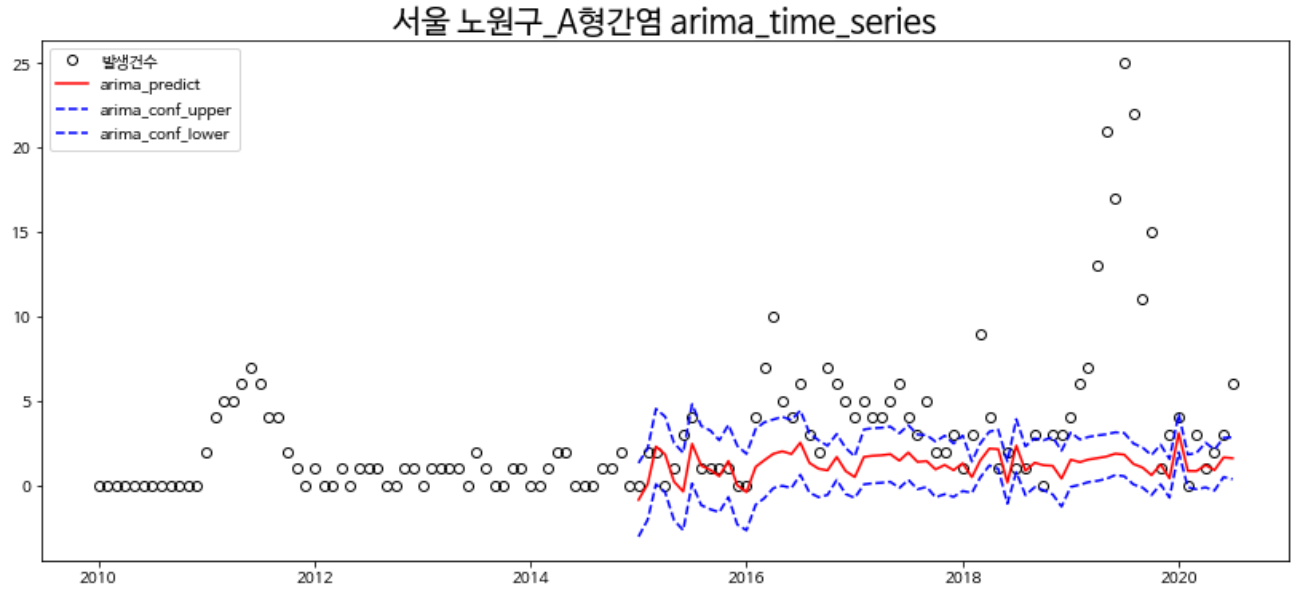
Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).

OLS Regression Results

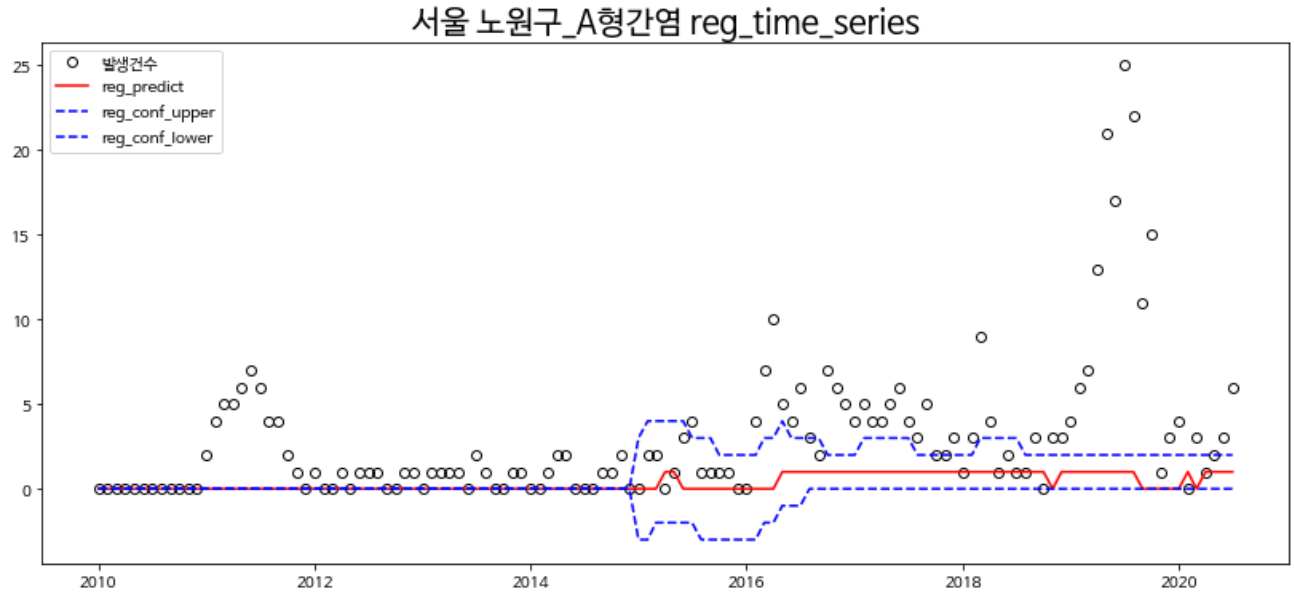
Dep. Variable:	전처리 후 발생건수	R-squared:	0.215
Model:	OLS	Adj. R-squared:	0.143
Method:	Least Squares	F-statistic:	2.966
Date:	Mon, 28 Sep 2020	Prob (F-statistic):	0.0194
Time:	03:04:00	Log-Likelihood:	-56.708
No. Observations:	60	AIC:	125.4
Df Residuals:	54	BIC:	138.0
Df Model:	5		
Covariance Type:	nonrobust		

```
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          1.3870      0.482      2.876      0.006       0.420       2.354
index         -0.0022      0.005     -0.445      0.658      -0.012       0.008
c1            -0.2237      0.120     -1.866      0.068      -0.464       0.017
d1             0.3955      0.121      3.261      0.002       0.152       0.639
c2             0.0793      0.120      0.662      0.511      -0.161       0.320
d2            -0.0537      0.120     -0.447      0.656      -0.295       0.187
==============================================================================
Omnibus:                       43.736   Durbin-Watson:                   2.259
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              175.953
Skew:                           2.012   Prob(JB):                     6.20e-39
Kurtosis:                      10.362   Cond. No.                         553.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
           구구분       날짜  A형간염
날짜
2010-01-01  노원구  2010-01   0.0
2010-02-01  노원구  2010-02   0.0
2010-03-01  노원구  2010-03   0.0
2010-04-01  노원구  2010-04   0.0
2010-05-01  노원구  2010-05   0.0
...         ...      ...   ...
2020-03-01  노원구  2020-03   3.0
2020-04-01  노원구  2020-04   1.0
2020-05-01  노원구  2020-05   2.0
2020-06-01  노원구  2020-06   3.0
2020-07-01  노원구  2020-07   6.0

[127 rows x 3 columns]
--------------------
           predict_virus
날짜
2015-01-01      -0.871107
2015-02-01       0.063453
2015-03-01       2.291450
2015-04-01       1.819288
2015-05-01       0.211041
...                   ...
2020-03-01       0.836294
2020-04-01       1.188682
2020-05-01       0.893492
2020-06-01       1.646456
2020-07-01       1.595808

[67 rows x 1 columns]
--------------------
           Confidence Lower  Confidence Upper
날짜
2015-01-01         -3.055051          1.312836
2015-02-01         -2.047341          2.174248
2015-03-01          0.057104          4.525797
2015-04-01         -0.424002          4.062578
2015-05-01         -2.075605          2.497687
...                      ...               ...
2020-03-01          0.238076          1.210665
```

```
2020-03-01        -0.238076        1.910665
2020-04-01        -0.134562        2.511925
2020-05-01        -0.320446        2.107430
2020-06-01         0.485442        2.807471
2020-07-01         0.348772        2.842843

[67 rows x 2 columns]
--------------------
                날짜  A형간염      년  ...  predict_virus  obs_ci_lower  obs_ci_upper
날짜                                  ...
2010-01-01  2010-01   0.0  2010  ...              0             0             0
2010-02-01  2010-02   0.0  2010  ...              0             0             0
2010-03-01  2010-03   0.0  2010  ...              0             0             0
2010-04-01  2010-04   0.0  2010  ...              0             0             0
2010-05-01  2010-05   0.0  2010  ...              0             0             0
...             ...   ...   ...  ...            ...           ...           ...
2020-03-01  2020-03   3.0  2020  ...              0             0             2
2020-04-01  2020-04   1.0  2020  ...              1             0             2
2020-05-01  2020-05   2.0  2020  ...              1             0             2
2020-06-01  2020-06   3.0  2020  ...              1             0             2
2020-07-01  2020-07   6.0  2020  ...              1             0             2

[127 rows x 14 columns]
====================
```



서울 노원구_A형간염 유행



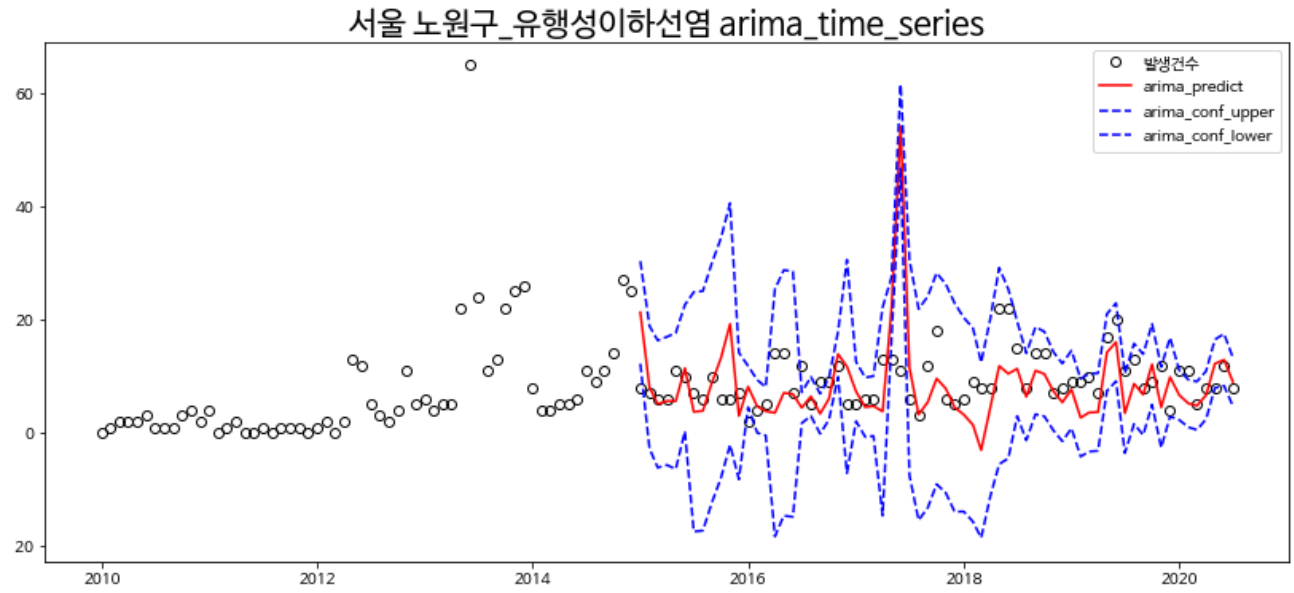
```
서울 노원구_A형간염 유행
노원구_백일해_0.0
노원구_유행성이하선염_8.0
        날짜  유행성이하선염     년  월  과거 5년 중앙값  중앙값 초과 여부  전처리 후 발생건
0     201001       0.0  2010  1          0          0        0.0
1     201002       1.0  2010  2          0          0        1.0
2     201003       2.0  2010  3          0          0        2.0
3     201004       2.0  2010  4          0          0        2.0
4     201005       2.0  2010  5          0          0        2.0
..       ...       ...   ... ..        ...        ...        ...
122   202003       5.0  2020  3          7          0        5.0
123   202004       8.0  2020  4         10          0        8.0
124   202005       8.0  2020  5         13          0        8.0
125   202006      12.0  2020  6         12          0       12.0
126   202007       8.0  2020  7         10          0        8.0

[127 rows x 7 columns]
                                SARIMAX Results
==========================================================================================
Dep. Variable:                          y   No. Observations:                   60
Model:             SARIMAX(2, 1, 0, 12)   Log Likelihood                -110.001
Date:                  Mon, 28 Sep 2020   AIC                            228.002
Time:                          03:33:28   BIC                            235.487
Sample:                               0   HQIC                           230.831
                                   - 60
Covariance Type:                    opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
intercept      1.2261      0.479      2.558      0.011       0.287       2.166
ar.S.L12      -0.6555      0.126     -5.183      0.000      -0.903      -0.408
ar.S.L24      -0.4723      0.142     -3.334      0.001      -0.750      -0.195
sigma2         4.7783      1.118      4.273      0.000       2.586       6.970
===================================================================================
Ljung-Box (Q):                       31.89   Jarque-Bera (JB):                 8.43
Prob(Q):                              0.82   Prob(JB):                         0.01
Heteroskedasticity (H):               1.39   Skew:                             0.78
Prob(H) (two-sided):                  0.51   Kurtosis:                         4.32
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
                            OLS Regression Results
==============================================================================
Dep. Variable:            전처리 후 발생건수   R-squared:                       0.586
Model:                            OLS   Adj. R-squared:                  0.548
Method:                 Least Squares   F-statistic:                     15.29
Date:                Mon, 28 Sep 2020   Prob (F-statistic):           2.39e-09
Time:                        03:33:31   Log-Likelihood:                -127.15
No. Observations:                  60   AIC:                             266.3
Df Residuals:                      54   BIC:                             278.9
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          1.7926      1.560      1.149      0.256      -1.335       4.921
index          0.0515      0.016      3.204      0.002       0.019       0.084
```

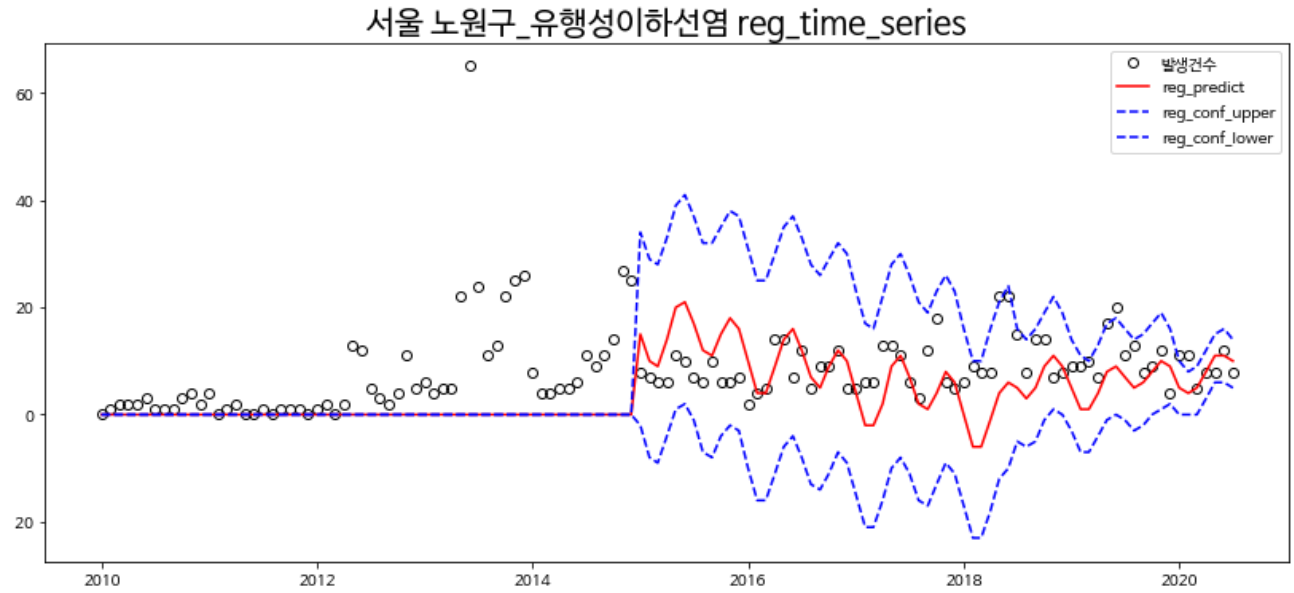
```
c1            -2.3217      0.388     -5.984      0.000      -3.100      -1.544
d1            -0.5910      0.392     -1.507      0.138      -1.377       0.195
c2            -0.3275      0.388     -0.844      0.402      -1.105       0.450
d2            -2.0295      0.389     -5.222      0.000      -2.809      -1.250
==============================================================================
Omnibus:                       21.699   Durbin-Watson:                   1.817
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               44.177
Skew:                           1.105   Prob(JB):                     2.55e-10
Kurtosis:                       6.576   Cond. No.                         553.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
            구구분       날짜  유행성이하선염
날짜
2010-01-01  노원구  2010-01      0.0
2010-02-01  노원구  2010-02      1.0
2010-03-01  노원구  2010-03      2.0
2010-04-01  노원구  2010-04      2.0
2010-05-01  노원구  2010-05      2.0
...         ...      ...      ...
2020-03-01  노원구  2020-03      5.0
2020-04-01  노원구  2020-04      8.0
2020-05-01  노원구  2020-05      8.0
2020-06-01  노원구  2020-06     12.0
2020-07-01  노원구  2020-07      8.0

[127 rows x 3 columns]
--------------------
            predict_virus
날짜
2015-01-01      21.300744
2015-02-01       8.062536
2015-03-01       5.085467
2015-04-01       5.647927
2015-05-01       5.613626
...                   ...
2020-03-01       4.797113
2020-04-01       6.652445
2020-05-01      12.224860
2020-06-01      12.917887
2020-07-01       9.002321

[67 rows x 1 columns]
--------------------
            Confidence Lower  Confidence Upper
날짜
2015-01-01         12.230274         30.371215
2015-02-01         -2.767583         18.892656
2015-03-01         -6.160482         16.331415
2015-04-01         -5.684919         16.980773
2015-05-01         -6.429361         17.656612
...                      ...               ...
2020-03-01          0.524345          9.069882
2020-04-01          2.400897         10.903994
2020-05-01          7.944723         16.504996
2020-06-01          8.322535         17.513240
2020-07-01          4.717968         13.286675
```

```
[67 rows x 2 columns]
--------------------
```

날짜	날짜	유행성이하선염	년	...	predict_virus	obs_ci_lower	obs_ci_upper
날짜				...			
2010-01-01	2010-01	0.0	2010	...	0	0	0
2010-02-01	2010-02	1.0	2010	...	0	0	0
2010-03-01	2010-03	2.0	2010	...	0	0	0
2010-04-01	2010-04	2.0	2010	...	0	0	0
2010-05-01	2010-05	2.0	2010	...	0	0	0
...	...	...	...	...	...	...	...
2020-03-01	2020-03	5.0	2020	...	5	0	9
2020-04-01	2020-04	8.0	2020	...	8	3	12
2020-05-01	2020-05	8.0	2020	...	11	6	15
2020-06-01	2020-06	12.0	2020	...	11	6	16
2020-07-01	2020-07	8.0	2020	...	10	5	14

```
[127 rows x 14 columns]
==================
```





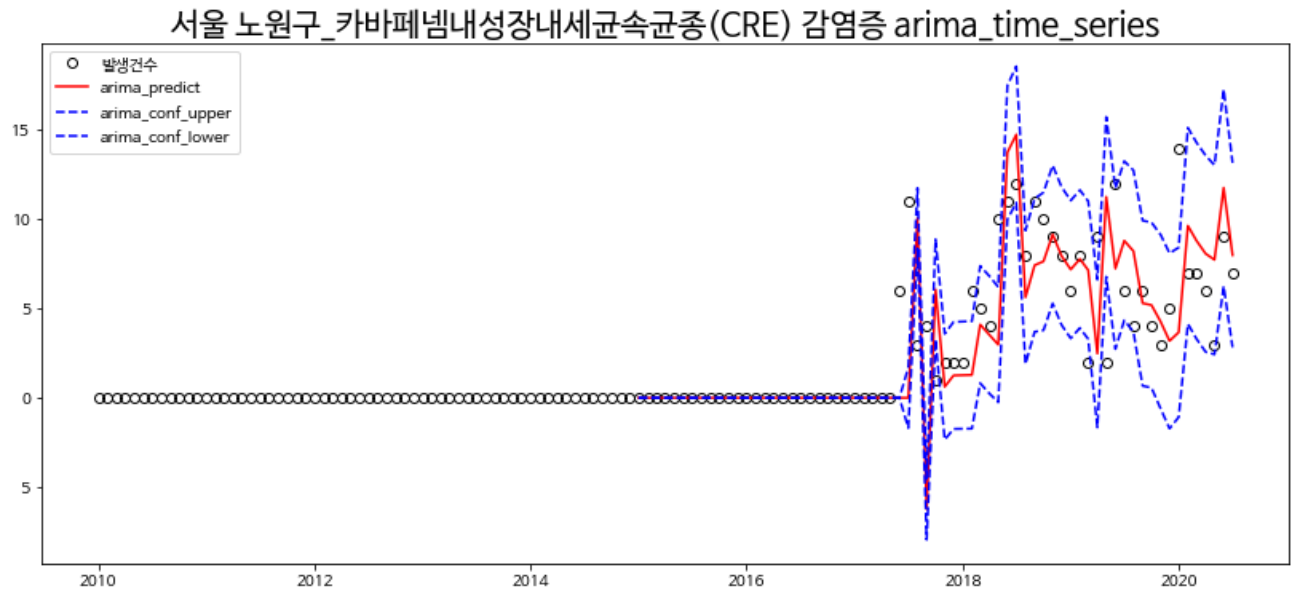
노원구_풍진(2018년이전)_0.0
노원구_풍진(선천성)_0.0
노원구_풍진(후천성)_0.0
노원구_폴리오_0.0
노원구_수막구균 감염증_0.0
노원구_b형헤모필루스인플루엔자_0.0

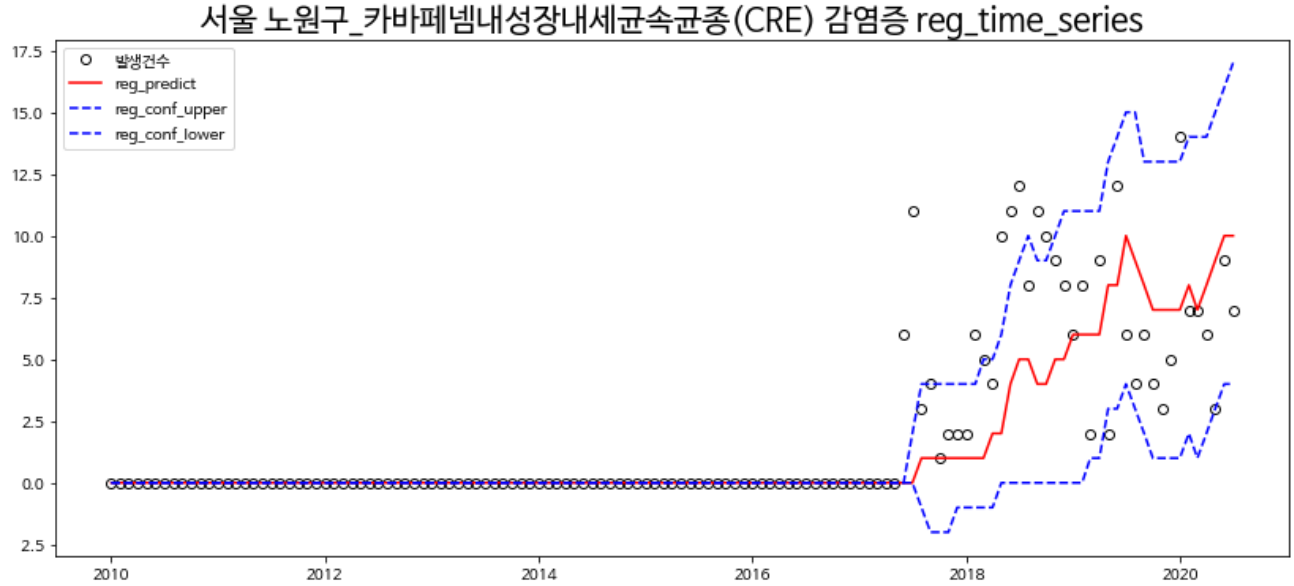
```
노원구_폐렴구균 감염증_0.0
노원구_한센병_0.0
노원구_성홍열_1.0
노원구_반코마이신내성황색포도알균(VRSA) 감염증_0.0
노원구_카바페넴내성장내세균속균종(CRE) 감염증_7.0
        날짜  카바페넴내성장내세균속균종(CRE) 감염증     년  월  과거 5년 중앙값  중앙값 초과
0    201001                    0.0  2010  1          0          0    0.000000
1    201002                    0.0  2010  2          0          0    0.000000
2    201003                    0.0  2010  3          0          0    0.000000
3    201004                    0.0  2010  4          0          0    0.000000
4    201005                    0.0  2010  5          0          0    0.000000
..      ...                    ...   ... ..        ...        ...         ...
122  202003                    7.0  2020  3          0          0    7.000000
123  202004                    6.0  2020  4          0          0    6.000000
124  202005                    3.0  2020  5          0          0    3.000000
125  202006                    9.0  2020  6          2          1    5.812099
126  202007                    7.0  2020  7          4          1    5.811007

[127 rows x 7 columns]
                                     SARIMAX Results
==========================================================================================
Dep. Variable:                                  y   No. Observations:                   60
Model:             SARIMAX(2, 0, 0)x(3, 1, 0, 12)   Log Likelihood                -122.802
Date:                            Mon, 28 Sep 2020   AIC                            259.605
Time:                                    03:44:44   BIC                            272.703
Sample:                                         0   HQIC                           264.554
                                              - 60
Covariance Type:                              opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
intercept      1.9831      2.122      0.934      0.350      -2.176       6.143
ar.L1          0.2616      0.153      1.705      0.088      -0.039       0.562
ar.L2          0.3529      0.204      1.727      0.084      -0.048       0.753
ar.S.L12      -0.8376      0.256     -3.277      0.001      -1.339      -0.337
ar.S.L24      -0.7121      0.316     -2.252      0.024      -1.332      -0.092
ar.S.L36      -0.4594      0.366     -1.254      0.210      -1.177       0.259
sigma2         6.9564      2.637      2.638      0.008       1.787      12.126
===================================================================================
Ljung-Box (Q):                       30.65   Jarque-Bera (JB):                 20.24
Prob(Q):                              0.86   Prob(JB):                         0.00
Heteroskedasticity (H):               2.77   Skew:                             1.23
Prob(H) (two-sided):                  0.05   Kurtosis:                         5.03
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
                            OLS Regression Results
==============================================================================
Dep. Variable:            전처리 후 발생건수   R-squared:                       0.511
Model:                            OLS   Adj. R-squared:                  0.466
Method:                 Least Squares   F-statistic:                     11.30
Date:                Mon, 28 Sep 2020   Prob (F-statistic):           1.75e-07
Time:                        03:44:47   Log-Likelihood:                -147.82
No. Observations:                  60   AIC:                             307.6
Df Residuals:                      54   BIC:                             320.2
Df Model:                           5
Covariance Type:            nonrobust
```

```
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const        -11.5586      2.202     -5.250      0.000     -15.973      -7.144
index          0.1620      0.023      7.138      0.000       0.117       0.208
c1            -0.9274      0.548     -1.694      0.096      -2.025       0.170
d1            -0.2143      0.554     -0.387      0.700      -1.324       0.896
c2             1.0922      0.548      1.995      0.051      -0.005       2.190
d2             0.0551      0.548      0.100      0.920      -1.044       1.155
==============================================================================
Omnibus:                        3.831   Durbin-Watson:                   1.289
Prob(Omnibus):                  0.147   Jarque-Bera (JB):                3.733
Skew:                           0.574   Prob(JB):                        0.155
Kurtosis:                       2.579   Cond. No.                         553.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
           구구분       날짜  카바페넴내성장내세균속균종(CRE) 감염증
날짜
2010-01-01  노원구  2010-01                    0.0
2010-02-01  노원구  2010-02                    0.0
2010-03-01  노원구  2010-03                    0.0
2010-04-01  노원구  2010-04                    0.0
2010-05-01  노원구  2010-05                    0.0
...         ...      ...                    ...
2020-03-01  노원구  2020-03                    7.0
2020-04-01  노원구  2020-04                    6.0
2020-05-01  노원구  2020-05                    3.0
2020-06-01  노원구  2020-06                    9.0
2020-07-01  노원구  2020-07                    7.0

[127 rows x 3 columns]
------------------
            predict_virus
날짜
2015-01-01       0.000000
2015-02-01       0.000000
2015-03-01       0.000000
2015-04-01       0.000000
2015-05-01       0.000000
...                   ...
2020-03-01       8.809450
2020-04-01       8.076849
2020-05-01       7.734528
2020-06-01      11.771127
2020-07-01       7.996913

[67 rows x 1 columns]
------------------
            Confidence Lower  Confidence Upper
날짜
2015-01-01         -0.000002          0.000002
2015-02-01         -0.000002          0.000002
2015-03-01         -0.000002          0.000002
2015-04-01         -0.000002          0.000002
2015-05-01         -0.000002          0.000002
...                      ...               ...
```

```
2020-03-01        3.306549        14.312351
2020-04-01        2.573884        13.579814
2020-05-01        2.432545        13.036511
2020-06-01        6.258830        17.283424
2020-07-01        2.827492        13.166334

[67 rows x 2 columns]
--------------------
```

```
                날짜  카바페넴내성장내세균속균종(CRE) 감염증  ...  obs_ci_lower  obs_ci_uppe
날짜                                          ...
2010-01-01  2010-01                  0.0  ...             0            0
2010-02-01  2010-02                  0.0  ...             0            0
2010-03-01  2010-03                  0.0  ...             0            0
2010-04-01  2010-04                  0.0  ...             0            0
2010-05-01  2010-05                  0.0  ...             0            0
...            ...                  ...  ...           ...          ...
2020-03-01  2020-03                  7.0  ...             1           14
2020-04-01  2020-04                  6.0  ...             2           14
2020-05-01  2020-05                  3.0  ...             3           15
2020-06-01  2020-06                  9.0  ...             4           16
2020-07-01  2020-07                  7.0  ...             4           17

[127 rows x 14 columns]
====================
```





노원구 E형간염 0.0

```
노원구_파상풍_0.0
노원구_B형간염_0.0
노원구_일본뇌염_0.0
노원구_C형간염_0.0
노원구_말라리아_0.0
노원구_레지오넬라증_0.0
노원구_비브리오패혈증_0.0
노원구_발진티푸스_0.0
노원구_발진열_0.0
노원구_쯔쯔가무시증_0.0
노원구_렙토스피라증_0.0
노원구_브루셀라증_0.0
노원구_공수병_0.0
노원구_신증후군출혈열_0.0
노원구_크로이츠펠트-야콥병(CJD) 및 변종크로이츠펠트-야콥병(vCJD)_0.0
노원구_황열_0.0
노원구_뎅기열_0.0
노원구_큐열_0.0
노원구_웨스트나일열_0.0
노원구_라임병_0.0
노원구_진드기매개뇌염_0.0
노원구_유비저_0.0
노원구_치쿤구니야열_0.0
노원구_중증열성혈소판감소증후군(SFTS)_0.0
노원구_지카바이러스감염증_0.0
```

```
epi_all=test_caution.epidemic_now()
```

```
0%                                                  0/18 [00:00<?, ?it/s]
0%                                                  0/1 [00:00<?, ?it/s]
```

```
epi_all.to_csv('/content/drive/My Drive/BigCon_BiKong/최종자료/epidemic_now_final.csv')
```

```
r2_score(epi_all[epi_all['발생건수']!=0]['발생건수'],epi_all[epi_all['발생건수']!=0]['ARIMA_pred'])
```

```
0.926295949469186
```

```
r2_score(epi_all[epi_all['발생건수']!=0]['발생건수'],epi_all[epi_all['발생건수']!=0]['REG_pred'])
```

```
0.879572145830851
```

```
test_caution.destination_caution('전국','전국')
```

```
test_caution.destination_caution('서울','강남구')
```