НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра прикладної математики

Курсовий проект

із дисципліни «Алгоритми і системи комп’ютерної математики»

На тему

«Прогнозування кількості хворих на COVID-19»

Етап №5

Виконав: Керівник:

студент групи КМ-93 доцент

Костенко О. А. Олефір О. С.

Київ — 2022

Код програми

import pandas as pd

import plotly.express as px

from statsmodels.tsa.statespace.sarimax import SARIMAX

from sklearn.metrics import mean\_squared\_log\_error

import plotly.graph\_objects as go

TRAINING\_SPLIT=0.7

def split(df):

    row\_number=df.shape[0]

    global TRAIN\_NUM

    TRAIN\_NUM=int(row\_number\*TRAINING\_SPLIT)

    df\_train=df.iloc[:TRAIN\_NUM, :]

    df\_test=df.iloc[TRAIN\_NUM:, :]

    return df\_train, df\_test

def losses(confirmed, df):

    actual=list(confirmed['Confirmed'])

    predcited=list(df['Forecast'])

    return mean\_squared\_log\_error(actual, predcited)

def plot\_results(df\_test, df):

    fig=go.Figure()

    fig.add\_trace(go.Scatter(

        x=df\_test['Date'],

        y=df\_test['Confirmed'],

        mode='lines',

        name='Actual values'

    ))

    fig.add\_trace(go.Scatter(

        x=df.iloc[TRAIN\_NUM:,0],

        y=df.iloc[TRAIN\_NUM:,2],

        mode='lines',

        name='Predicted values'

    ))

    fig.show()

def build\_model(df\_train):

    sarimax\_model=SARIMAX(df\_train['Confirmed'], order=(4, 2, 0), seasonal\_order=(0, 1, 1, 7))

    sarimax\_model\_fit=sarimax\_model.fit()

    return sarimax\_model\_fit

def sarimax\_predict(model, df\_test):

    predicted=pd.DataFrame()

    forecast\_test=model.forecast(len(df\_test))

    predicted['Date']=df\_test['Date']

    predicted['Forecast']=list(forecast\_test)

    return predicted

def sarimax(df):

    print(f'Raw Data:\n{df.head(10)}')

    fig=px.line(df, x='Date', y='Confirmed')

    fig.show()

    df\_train, df\_test=split(df)

    model=build\_model(df\_train)

    predicted=sarimax\_predict(model, df\_test)

    df['Forecast']=[None]\*TRAIN\_NUM+list(predicted['Forecast'])

    df.plot()

    print(f'Predicted Values: \n{predicted.head(10)}')

    plot\_results(df\_test, df)

    plot\_results(df, df)

    loss=losses(df\_test, predicted)

    print(f'Losses: Mean Squared Log Error: {losses}')

if \_\_name\_\_=='\_\_main\_\_':

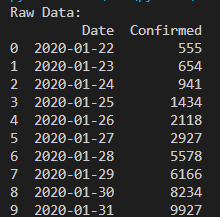
    df=pd.read\_csv('day\_wise.csv')

    covid\_df=df[['Date', 'Confirmed']].dropna()

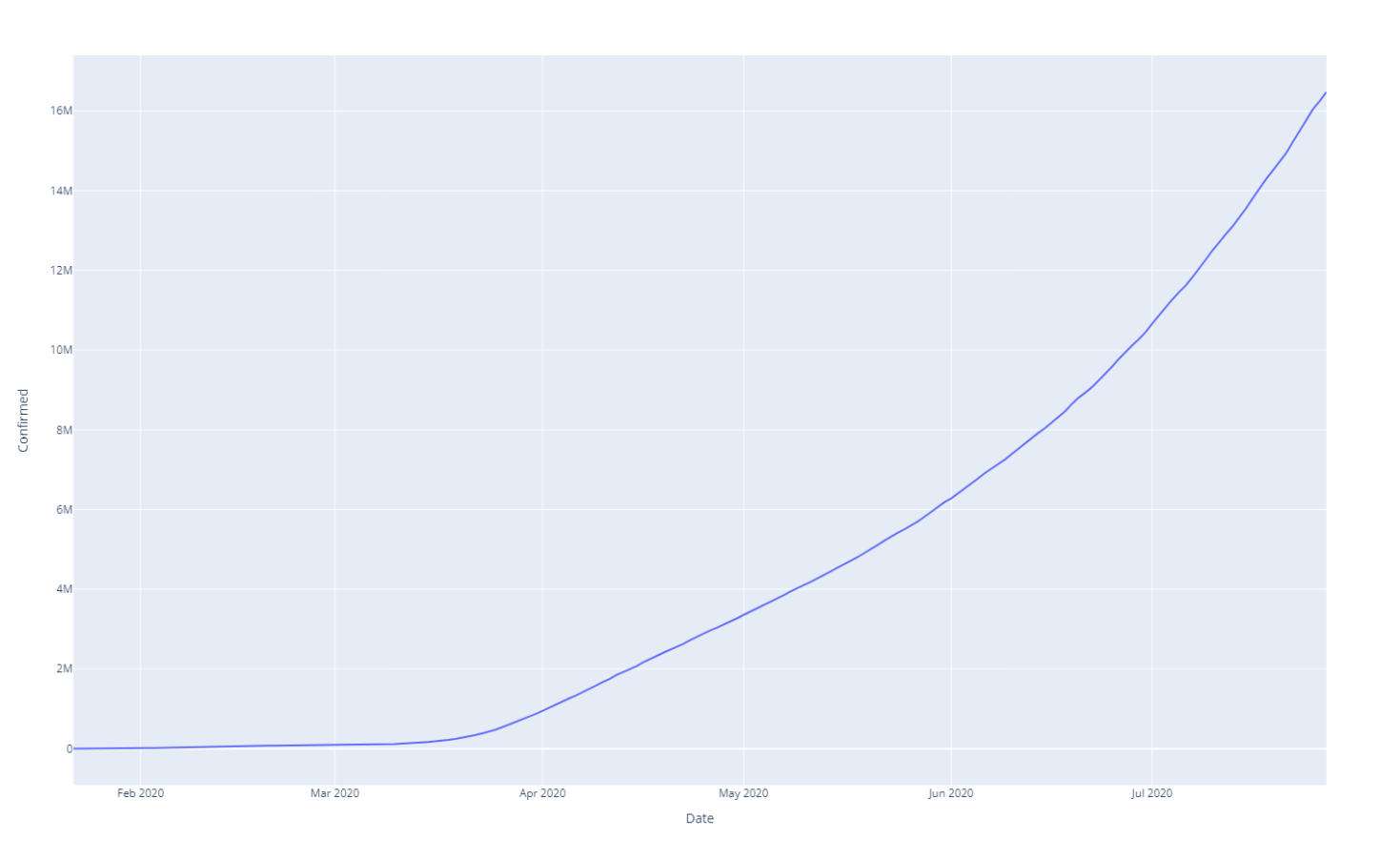
    sarimax(covid\_df)

Опис результатів

Вхідні дані є .csv файл, що містить інформацію про дату та кількість підтверджених випадків COVID-19, що відповідають цій даті.

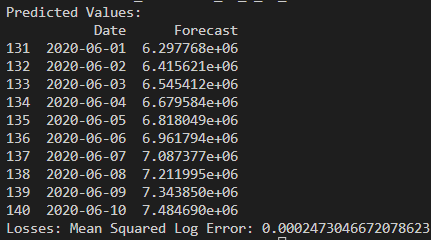


Візуалізація вхідних даних:



Розділимо вхідні дані на навчальну та тестову вибірки, де навчальна вибірка складає 70% від вхідного набору даних та використовується для навчання моделі SARIMAX, а тестова для перевірки результатів прогнозування.

В результаті прогнозування було отримано такі результати:



Похибка прогнозування складає 0,0002473 на тестових даних, що означає, що модель дає досить точні результати.

Візуалізація результатів прогнозування: