



Busan science high school

2023 Ocean ICT Festival

2023 BOIF

B
30

QR 코드 영역
QR 삽입 후
테두리 삭제

Youtube 영상 QR

엘리뇨 시기의 어획량 계산 프로그램 제작

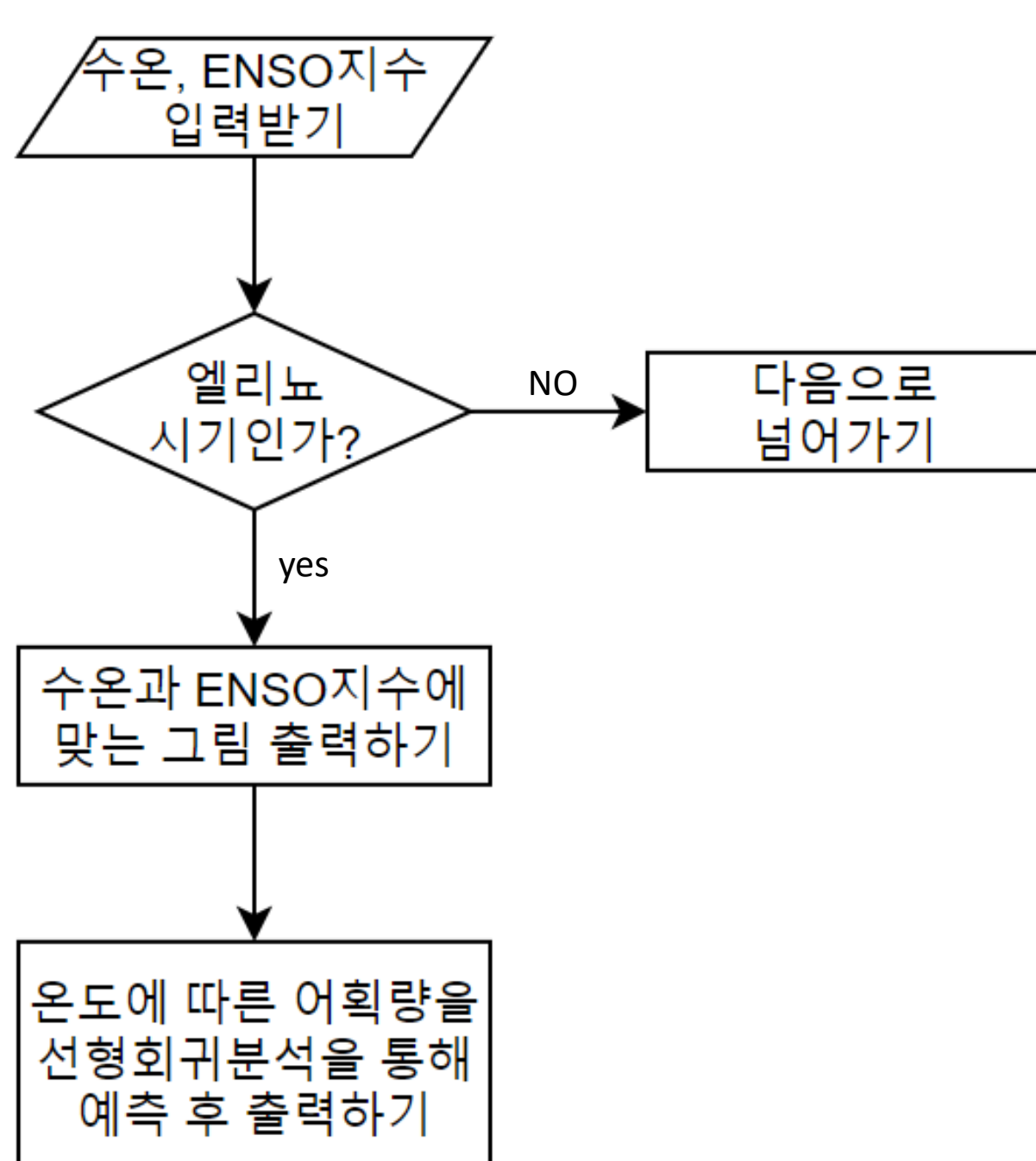
어어어송 : 2403 박세영 2404 설예린

작품개요

여러 전문가들이 2023년 여름은 사상 최강의 엘리뇨가 발생할 것이라 말하고 있다. 엘리뇨가 발생하면, 페루 앞바다의 해수 온도가 높아진다. 이러한 엘리뇨 현상이 발생할 때, 한국에서는 강수량 부족, 겨울철 기온 상승 등의 문제가 발생하고, 해양심층수의 용승이 억제되어 어업량이 감소하게 된다는 문제 또한 발생한다. 이러한 상황을 배경으로 온도 혹은 ENSO 지수를 입력하고, 이에 따라 엘리뇨 현상이 일어났을 때 전세계의 온도를 지도 나타내게 한다. 이렇게 입력 받은 온도로 각 어종이 얼마나 잡히는지를 예측할 수 있는 프로그램을 만들고자 한다!

코드 설명

알고리즘 구성



엘리뇨/라니냐 판정 코드

```
import csv
import numpy as np
import pandas as pd
import matplotlib.pyplot
from sklearn.metrics import mean_squared_error
from PIL import image
from sklearn.linear_model import LinearRegression

#가장 최적의 년도 구하기

file = 'c:/Users/user/Documents/오션 아이시티.csv'
df1 = pd.read_csv(file)

N12 = float(input('NINO1+2 지역의 온도를 입력하세요 : '))
N3 = float(input('NINO3 지역의 온도를 입력하세요 : '))
N4 = float(input('NINO4 지역의 온도를 입력하세요 : '))
N34 = float(input('NINO3.4 지역의 온도를 입력하세요 : '))

input_x = np.array([N12, N3, N4, N34])
column_indices = [2, 4, 6, 8]
data_values = df1.iloc[:, column_indices].values.tolist()
data_x = np.array(data_values)

s = []
for i in range(len(data_x)):
    s.append(mean_squared_error(data_x[i], input_x))

mse_x = np.min(s)
min_row_index = np.argmin(s)
print(mse_x)
print(min_row_index)
print(df1.iloc[min_row_index, [0, 1]].values.tolist())

# 엘리뇨 판정

file = 'c:/Users/user/Documents/엘리뇨 판정 파일.csv'
df2 = pd.read_csv(file)
print(df2)

def month(num):
    month = {
        0 : 'D',
        1 : 'J',
        2 : 'F',
        3 : 'M',
        4 : 'A',
        5 : 'M',
        6 : 'J',
        7 : 'J',
        8 : 'A',
        9 : 'S',
        10 : 'O',
        11 : 'N',
        12 : 'D',
        13 : 'J'
    }

    return month.get(num)
def mod_12(num):
    return num % 12

def read():
    return int(df1.iloc[min_row_index][1])

def find_matching_rows(df, column_names, input_values):
    matching_rows = df[(df[column_names[0]] == input_values[0])]
    return matching_rows

result1 = month(mod_12(read() - 1))
result2 = month(read())
result3 = month(mod_12(read() + 1))

year = df1.iloc[min_row_index][0]
month = result1 + result2 + result3
print(year, month)

matching_rows = find_matching_rows(df2, ['YR', 'SEAS'], [year,
print(matching_rows)
print('엘리뇨 시기가 아닙니다..')

#사진 불러오기

df1.iloc[min_row_index, [0, 1]].values.tolist()
year_ = int(df1.iloc[min_row_index, [0, 1]].values.tolist()[0])
month_ = str(int(df1.iloc[min_row_index, [0, 1]].values.tolist()[1]))
image_path1 = 'C:/Users/user/Downloads/Ocean/SSTA.monthly.%d%s'
image1 = Image.open(image_path1)
display(image1)

image_path2 = 'C:/Users/user/Downloads/Ocean/SSTA.colorbar.png'
image2 = Image.open(image_path2)
display(image2)

image_path3 = 'C:/Users/user/Downloads/Ocean/NINO 지역.gif'
image3 = Image.open(image_path3)
display(image3)
```

수온에 따른 어획량 예측 코드

```
#어획량 출력 코드

# csv 파일 로드
data = pd.read_csv('c:/Users/user/Documents/ocean.csv') |
# 특성과 타겟 분리
X = data[['temper']].values
y = data['amount'].values

# 선형 회귀 모델 생성 및 훈련
model = LinearRegression()
model.fit(X, y)

#어획량 확인할 지역 입력 받기

def tem(region):
    region_data = {
        'Nino1+2': N12,
        'Nino3': N3,
        'Nino4': N4,
        'Nino3.4': N34
    }
    return region_data.get(region)

# 사용자 입력을 받아 어획량 예측
while True:
    input_region = input('어획량을 확인할 지역을 입력하세요 : ')

    try:
        input_temp = tem(input_region)
        predicted_amount = model.predict([[input_temp]])
        print(f'예상 어획량(kt): {predicted_amount[0]}')
    except ValueError:
        print("유효한 숫자를 입력하세요.")

continue_prompt = input("더 예측하시겠습니까? (y/n): ")
if continue_prompt.lower() != 'y':
    break
```

csv 파일

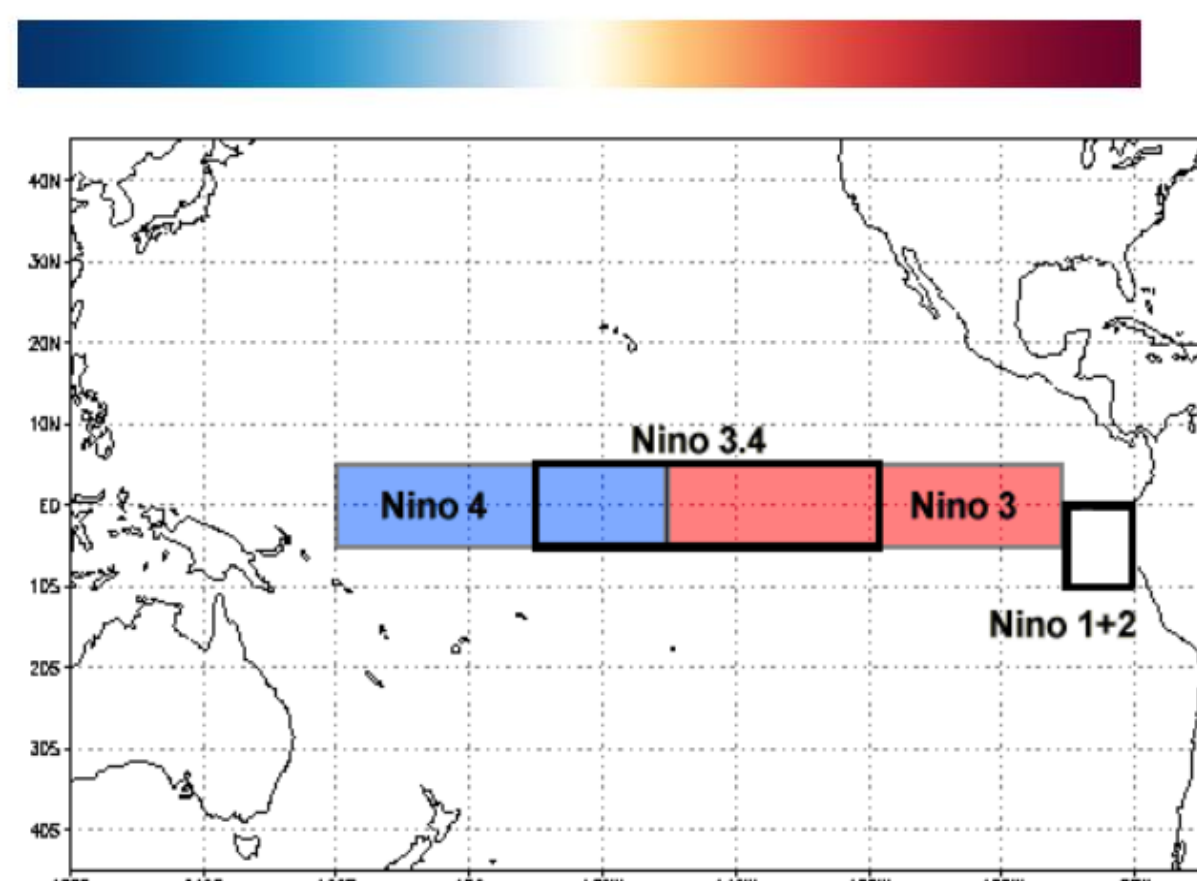
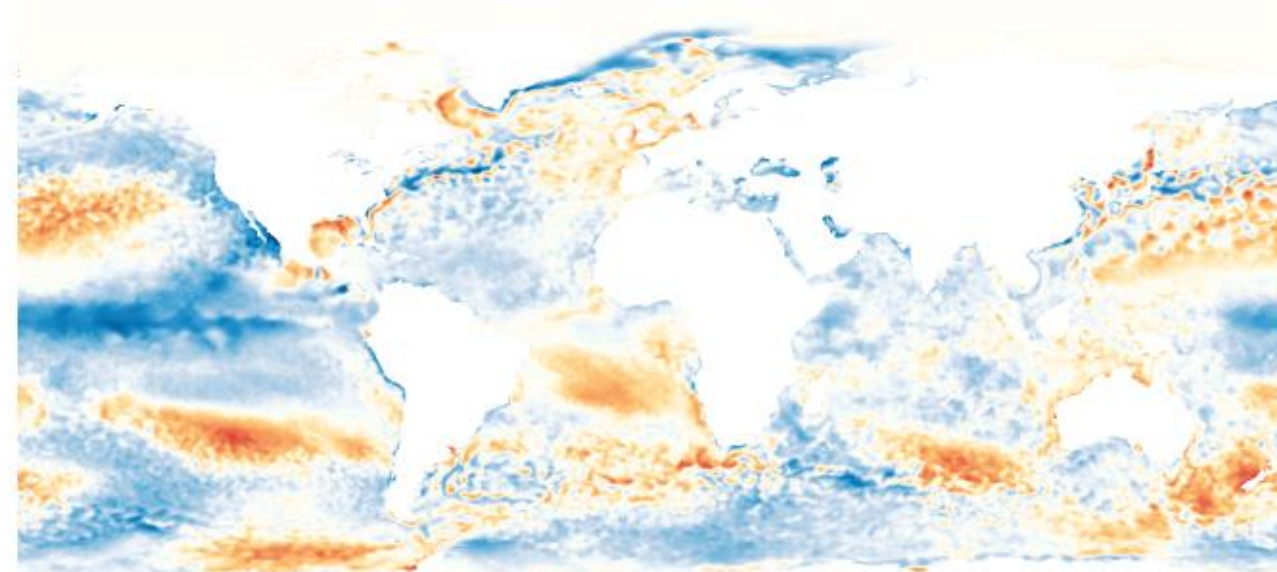
	A	B	C	D	E	F	G	H	I	J
1	YR	MON	NINO1+2	ANOM	NINO3	ANOM	NINO4	ANOM	NINO3.4	ANOM
2	1982	1	24.28	-0.24	25.84	0.17	28.01	-0.21	26.65	0.08
3	1982	2	25.38	-0.72	26.26	-0.11	27.99	-0.11	26.54	-0.2
4	1982	3	25.22	-1.38	26.92	-0.25	28.18	-0.05	27.09	-0.14
5	1982	4	24.57	-1.16	27.52	-0.05	28.61	0.1	27.83	0.02
6	1982	5	24	-0.62	27.7	0.49	29.19	0.4	28.37	0.49
7	1982	6	22.88	-0.29	27.1	0.46	29.46	0.62	28.35	0.65
8	1982	7	22.07	0.24	26.14	0.27	28.76	-0.02	27.57	0.27
9	1982	8	21.71	0.85	26.05	0.83	28.7	0.01	27.57	0.86
10	1982	9	21.91	1.33	26.5	1.49	28.91	0.24	28.01	1.24
11	1982	10	22.67	1.8	27.04	1.96	29.11	0.43	28.5	1.73
12	1982	11	24.4	2.78	27.39	2.19	28.85	0.18	28.5	1.68

	A	B
1	temper	amount
2	2.8	215
3	2.9	215
4	3	215
5	3.1	215
6	3.2	215
7	3.3	215
8	3.4	215
9	3.5	215

결과화면

```
... NINO1+2 지역의 온도를 입력하세요 : 32
NINO3 지역의 온도를 입력하세요 : 3
NINO4 지역의 온도를 입력하세요 : 4
NINO3.4 지역의 온도를 입력하세요 : 32
267.349425
84
[1989.0, 1.0]
SEAS YR TOTAL ANOM
0 DJF 1950 24.72 -1.53
1 JFM 1950 25.17 -1.34
2 FMA 1950 25.75 -1.16
3 MAM 1950 26.12 -1.18
4 AMJ 1950 26.32 -1.07
.. ... ..
876 DJF 2023 25.96 -0.67
877 JFM 2023 26.44 -0.43
878 FMA 2023 27.15 -0.14
879 MAM 2023 27.85 0.16
880 AMJ 2023 28.30 0.47

[881 rows x 4 columns]
1989.0 DJF
SEAS YR TOTAL ANOM
468 DJF 1989 24.84 -1.69
엘리뇨 시기가 아닙니다.
```



```
어획량을 확인할 지역을 입력하세요: Nino4
예상 어획량(kt): 220.07466915606602
더 예측하시겠습니까? (y/n): y
어획량을 확인할 지역을 입력하세요: Nino3
예상 어획량(kt): 218.33898062648424
더 예측하시겠습니까? (y/n): y
어획량을 확인할 지역을 입력하세요: Nino3.4
예상 어획량(kt): 268.673947984356
더 예측하시겠습니까? (y/n): y
어획량을 확인할 지역을 입력하세요: Nino1+2
예상 어획량(kt): 268.673947984356
더 예측하시겠습니까? (y/n): n
```

기대 효과

1. ENSO지수를 활용하여 전 세계의 온도를 시각화함으로써 엘리뇨로 인해 발생하는 기후 변화를 직관적으로 알 수 있다.
2. 엘리뇨 시기의 각 해역의 온도에 따른 어획량을 예측할 수 있다.