df = pd.read\_csv('Test Days/Dummy\_data\_1week\_7Jan.csv')

df = df.set\_index(['lat\_grid','long\_grid'])

df

from scipy.spatial.distance import pdist, euclidean, squareform

lat\_range = {'min': 28.486, 'max': 28.72}

long\_range = {'min': 77.1, 'max': 77.32}

n\_lat\_grid = 25

n\_long\_grid = 25

#adjacency matrix

grid\_points = pd.read\_csv('grid\_points\_25.csv')

n\_grid\_points = grid\_points.shape[0]

X = grid\_points[["lat\_grid", "long\_grid"]].to\_numpy()

grid\_points = grid\_points.pivot\_table(index=['lat\_grid', 'long\_grid'])

grid\_points\_indices = grid\_points.index

# adjacency matrix will be 1 for the first(acc to eucledian distance) nerighbouring 8 grids that have pollution values and the self grid, 0 otherwise

pairwise\_distances = {}

for u in X:

distance\_u = [euclidean(u, vertex) for vertex in X]

pairwise\_distances[(u[0], u[1])] = distance\_u

grid\_indices = df.index.tolist()

for i in range(1, 15):

total\_adj\_nodes = 0

granger\_adj\_nodes = 0

for u in grid\_indices:

print(u)

first\_i\_neighbours = X[np.argsort(pairwise\_distances[u])[1:i+1]] #excluding self

series1 = df.loc[u]

for v in first\_i\_neighbours:

v = (v[0], v[1])

print(v)

if v in grid\_indices:

total\_adj\_nodes += 1

series2 = df.loc[(v[0], v[1])]

gragres = grangercausalitytests(pd.DataFrame({'1':series1, '2':series2}), maxlag=[6])

ftest\_pvalue = gragres[6][0]['params\_ftest'][1]

if ftest\_pvalue < 0.05:

granger\_adj\_nodes += 1

if total\_adj\_nodes == 0:

frac\_granger\_nodes = 0

else:

frac\_granger\_nodes = granger\_adj\_nodes/total\_adj\_nodes

print(frac\_granger\_nodes)

grangercausalitytests(pd.DataFrame({'1':df.loc[(2, 20)], '2':df.loc[(1, 20)]}), maxlag=[6])