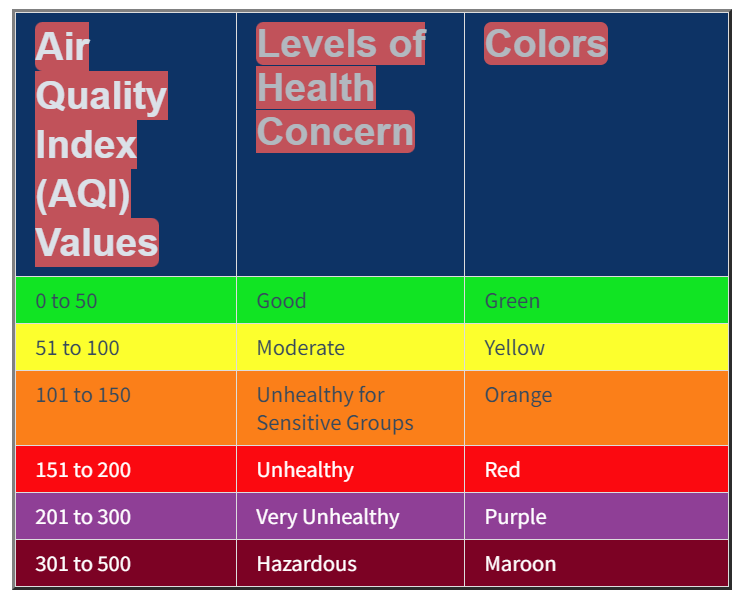
## 2.7.5 空间动力学——空间Kendall’s Tau

### 2.7.5.1 空气质量指数数据集

#### 1）空气质量指数（Air quality index，AQI）

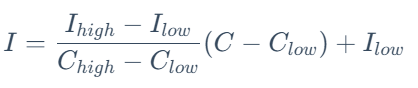
AQI一般通过空气质量传感器均值读数获得，通常因为车辆交通、森林火灾及任何可能增加空气污染的因素使得AQI值增加[1]。测试的污染物一般有臭氧、二氧化氮、二氧化硫等。AQI常被政府机构[2]用来向公众传达当前空气污染程度或者预测的空气污染程度[3][4]。公共健康风险伴随AQI值的增加而增加，特别影响到儿童、老人和有呼吸道或心血管问题的人，因此对AQI的变化监测，对其分布特征和成因分析，有助于平衡人类生产活动，减轻公共健康的影响程度。不同国家AQI计算方法各异，对应各自的空气质量标准，因为分析的AQI数据集符合美国环境保护局（ United States Environmental Protection Agency，EPA） 制定的标准，因此以此为例。此类AQI值被划分为6类，如下图[1]：



AQI测量的污染物基于*Clean Air Act*①中规定的5种指示污染物：地面臭氧、颗粒物、一氧化碳、二氧化硫和二氧化氮。同时为了保护公众健康，EPA为每一种污染物制定了国家环境空气质量标准（National Ambient Air Quality Standards，NAAQS），且要求每五年审查一次国家环境空气质量标准，以反映不断变化的健康影响信息，空气质量指数定期调整以反映这些变化。

* AQI计算方法

AQI是污染物浓度的一个分段线性函数（piecewise linear function）。在AQI分段之间的边界，会出现一个AQI单位的不连续跳跃，将空气污染浓度转换为AQI，公式如下[5]：



如果测量了多种污染物，计算出的AQI是对每种污染物应用上述公式计算出的最高值。

式中，  
I为AQI； C为污染物浓度； Clow​为对应分段小于C的浓度断点； Chigh​为对应分段大于C的浓度断点； Ilow​为对应Clow​的索引断点； Ihigh​为对应Chigh​的索引断点。EPA给定的断点表如下[1][6][7][8] ：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**O3**](https://richiebao.github.io/wiki/Ozone)**(ppb)** | [**O3**](https://richiebao.github.io/wiki/Ozone)**(ppb)** | [**PM2.5**](https://richiebao.github.io/wiki/Particulates)**(μg/m3)** | [**PM10**](https://richiebao.github.io/wiki/Particulates)**(μg/m3)** | [**CO**](https://richiebao.github.io/wiki/Carbon_monoxide)**(ppm)** | [**SO2**](https://richiebao.github.io/wiki/Sulfur_dioxide)**(ppb)** | [**NO2**](https://richiebao.github.io/wiki/Nitrogen_dioxide)**(ppb)** | **AQI** | **AQI** |
| *Clow* – *Chigh (avg)* | *Clow* – *Chigh (avg)* | *Clow* – *Chigh (avg)* | *Clow* – *Chigh (avg)* | *Clow* – *Chigh (avg)* | *Clow* – *Chigh (avg)* | *Clow* – *Chigh (avg)* | *Ilow* – *Ihigh* | **Category** |
| 0–54 (8-hr) | — | 0.0–12.0 (24-hr) | 0–54 (24-hr) | 0.0–4.4 (8-hr) | 0–35 (1-hr) | 0–53 (1-hr) | 0–50 | Good |
| 55–70 (8-hr) | — | 12.1–35.4 (24-hr) | 55–154 (24-hr) | 4.5–9.4 (8-hr) | 36–75 (1-hr) | 54–100 (1-hr) | 51–100 | Moderate |
| 71–85 (8-hr) | 125–164 (1-hr) | 35.5–55.4 (24-hr) | 155–254 (24-hr) | 9.5–12.4 (8-hr) | 76–185 (1-hr) | 101–360 (1-hr) | 101–150 | Unhealthy for Sensitive Groups |
| 86–105 (8-hr) | 165–204 (1-hr) | 55.5–150.4 (24-hr) | 255–354 (24-hr) | 12.5–15.4 (8-hr) | 186–304 (1-hr) | 361–649 (1-hr) | 151–200 | Unhealthy |
| 106–200 (8-hr) | 205–404 (1-hr) | 150.5–250.4 (24-hr) | 355–424 (24-hr) | 15.5–30.4 (8-hr) | 305–604 (24-hr) | 650–1249 (1-hr) | 201–300 | Very Unhealthy |
| — | 405–504 (1-hr) | 250.5–350.4 (24-hr) | 425–504 (24-hr) | 30.5–40.4 (8-hr) | 605–804 (24-hr) | 1250–1649 (1-hr) | 301–400 | Hazardous |
| — | 505–604 (1-hr) | 350.5–500.4 (24-hr) | 505–604 (24-hr) | 40.5–50.4 (8-hr) | 805–1004 (24-hr) | 1650–2049 (1-hr) | 401–500 |

假设传感器监测到PM2.5的24小时测量均值为26.4微克/立方米，根据上述公式并查表得出：，根据表查到81.073对应到的AQI分类为“Moderate”。

#### 2）AQI数据集

从[Kaggle](https://www.kaggle.com/)②中搜索AQI相关数据集，选择[Air Quality Index Dataset](https://www.kaggle.com/datasets/pratikshapandapkp/air-quality-index-dataset)④的AQI数据集，并参考该作者数据处理可视化的方法进行数据预处理。该数据列出了世界大部分国家从2022年7月21日到2022年9月22日约63天，以日为周期的AQI数值。

Kaggle是Google的子公司，是一个数据科学家和机器学习从业者的在线社区。Kaggle允许用户寻找和发布数据集，在基于网络的数据科学环境中探索和建立模型，与其他数据科学家和机器学习工程师合作，并通过参加比赛的形式解决数据科学挑战问题。Kaggle于2010年首次推出，提供机器学习比赛，现在还提供一个公共数据平台，及人工智能教育。③

参数管理使用AttrDict()方法（具体查看“Cityscapes数据集——参数管理”一节）。其中数据读写位置路径置于data子属性下。

from util\_misc import AttrDict  
\_\_C=AttrDict()   
args=\_\_C  
  
\_\_C.data=AttrDict()  
\_\_C.data.AQI='./data/Air\_Quality\_Index\_Data.csv' # AQI数据集文件路径  
\_\_C.data.aqi\_world='./data/AQI\_world.geojson' # 将AQI数据集合并世界国家边界地图后，文件存储路径

通过pandas库提供的read\_csv方法读取CSV格式数据为DataFrame格式，并将时间字符列通过pd.to\_datetime方法转换为datetime64[ns]类型时间格式且增加新列为ts。为了便于查看DataFrame格式数据的主要信息，定义df\_info\_print()函数，可以统一打印数据行列数、列唯一值数据、基础统计量、各列空值数据量和数值类型等。

import pandas as pd  
  
df\_info\_print=lambda df:print(f"# df shape: {df.shape};\n# nunique:\n {df.nunique()};\n# describe: \n{df.describe()}\n# isna sum:\n {df.isna().sum()}\n#type:\n {df.dtypes}")  
   
aqi\_df=pd.read\_csv(args.data.AQI)  
aqi\_df['ts']=pd.to\_datetime(aqi\_df.Date,format='%d-%m-%Y')  
df\_info\_print(aqi\_df)  
aqi\_df.head()

# df shape: (9994, 5);  
# nunique:  
 Date 64  
Country 142  
Status 6  
AQI Value 256  
ts 64  
dtype: int64;  
# describe:   
 AQI Value  
count 9994.000000  
mean 60.795477  
std 45.763604  
min 1.000000  
25% 29.000000  
50% 52.000000  
75% 82.000000  
max 868.000000  
# isna sum:  
 Date 0  
Country 0  
Status 0  
AQI Value 0  
ts 0  
dtype: int64  
#type:  
 Date object  
Country object  
Status object  
AQI Value int64  
ts datetime64[ns]  
dtype: object

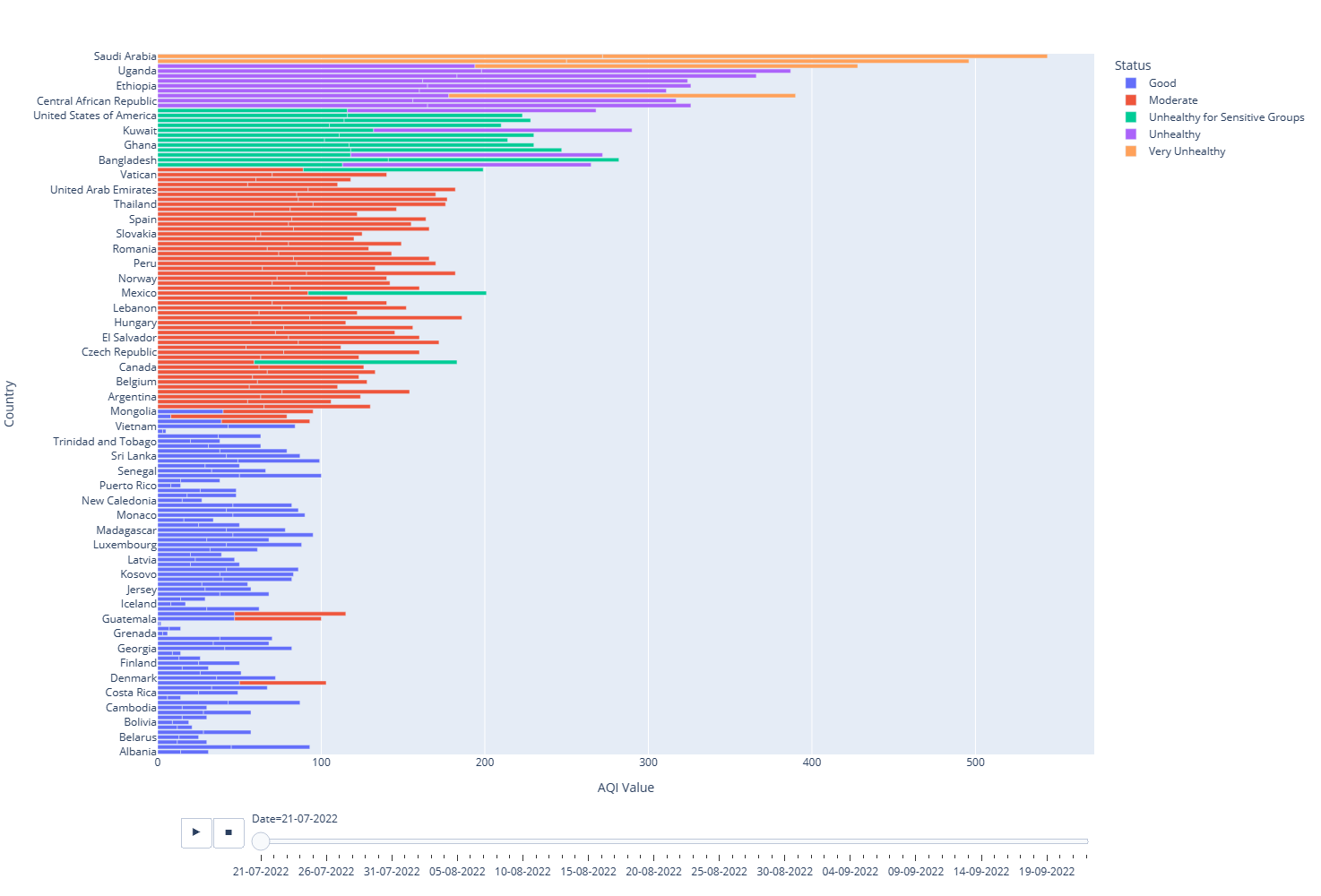
|  | **Date** | **Country** | **Status** | **AQI Value** | **ts** |
| --- | --- | --- | --- | --- | --- |
| **0** | 21-07-2022 | Albania | Good | 14 | 2022-07-21 |
| **1** | 21-07-2022 | Algeria | Moderate | 65 | 2022-07-21 |
| **2** | 21-07-2022 | Andorra | Moderate | 55 | 2022-07-21 |
| **3** | 21-07-2022 | Angola | Unhealthy for Sensitive Groups | 113 | 2022-07-21 |
| **4** | 21-07-2022 | Argentina | Moderate | 63 | 2022-07-21 |

需要对国家名进行修正，一方面是与下文中世界国家边界数据的国家名未对应部分的修正；另一方面是分类不正确的区域命名的修正。

aqi\_df["Country"]=aqi\_df["Country"].replace({'United Kingdom of Great Britain and Northern Ireland':"United Kingdom",  
 'Hong Kong':'Hong Kong,China',  
 'Taiwan': 'Taiwan, China'})

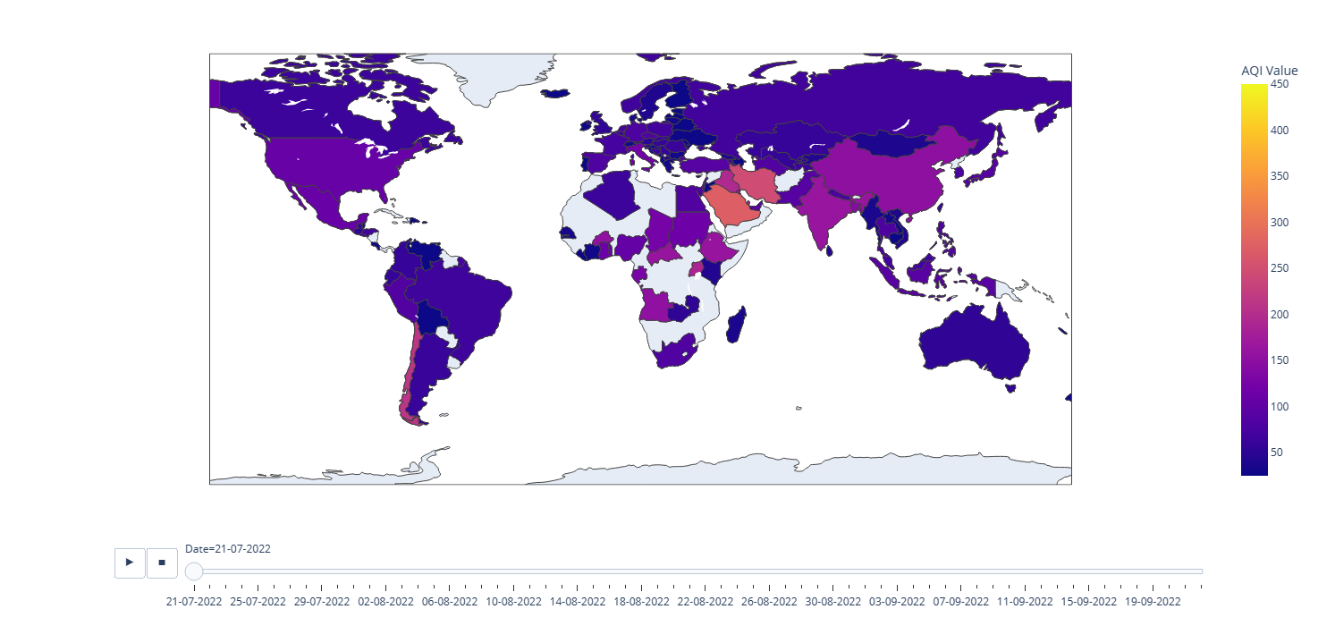
[Plotly](https://plotly.com/python/#controls)⑤库提供了交互动态观察时间序列数据的图表方法，使用px.bar方法，指定值参数为AQI数值列，按照国家地区，给定动画帧参数animation\_frame为时间列，打印条形图。可以通过拖动显示的滑条或者运行停止按钮动态观察不同时间AQI值的变化情况。

import plotly.express as px  
import matplotlib.pyplot as plt  
  
plt.figure(figsize=(25,20))  
px.bar(aqi\_df , x="AQI Value", y="Country", color="Status", animation\_frame="Date", width=1400,height=1000)



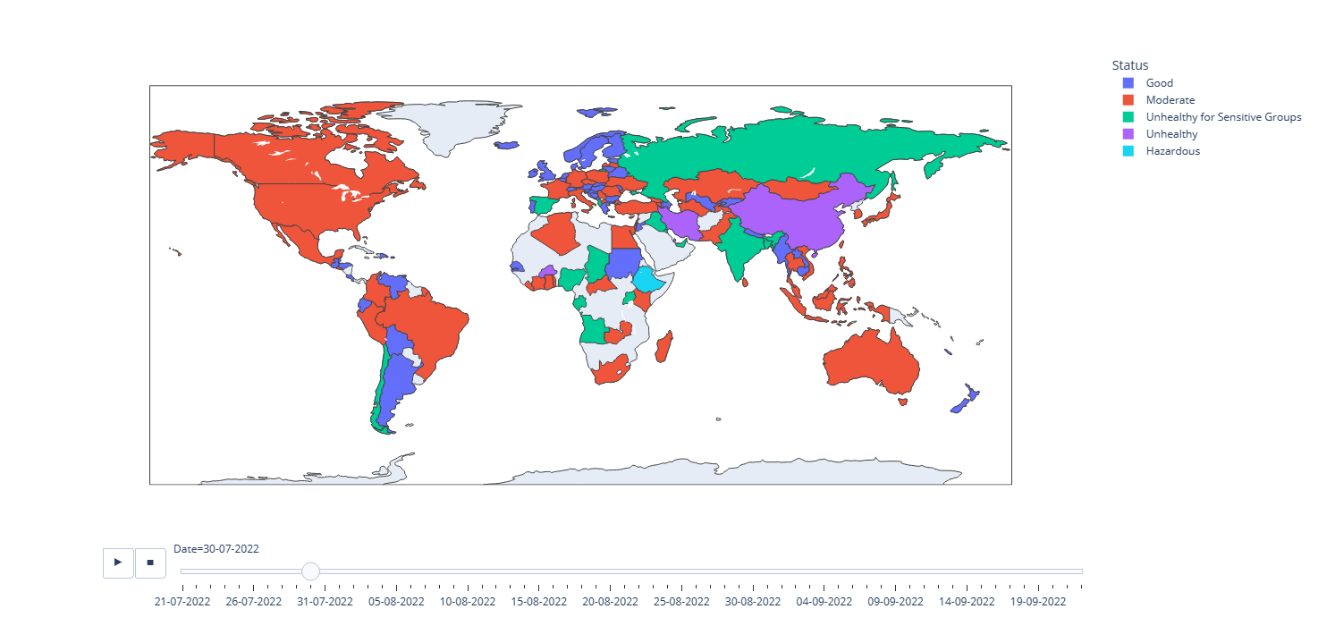
条形图并不能显示AQI数值变化的空间分布情况，可以使用px.choropleth打印地图观察实现。

fig=px.choropleth(aqi\_df,locations="Country",   
 locationmode='country names', color="AQI Value",   
 animation\_frame="Date",range\_color=[25,450], width=1400, height=700)  
fig.show()



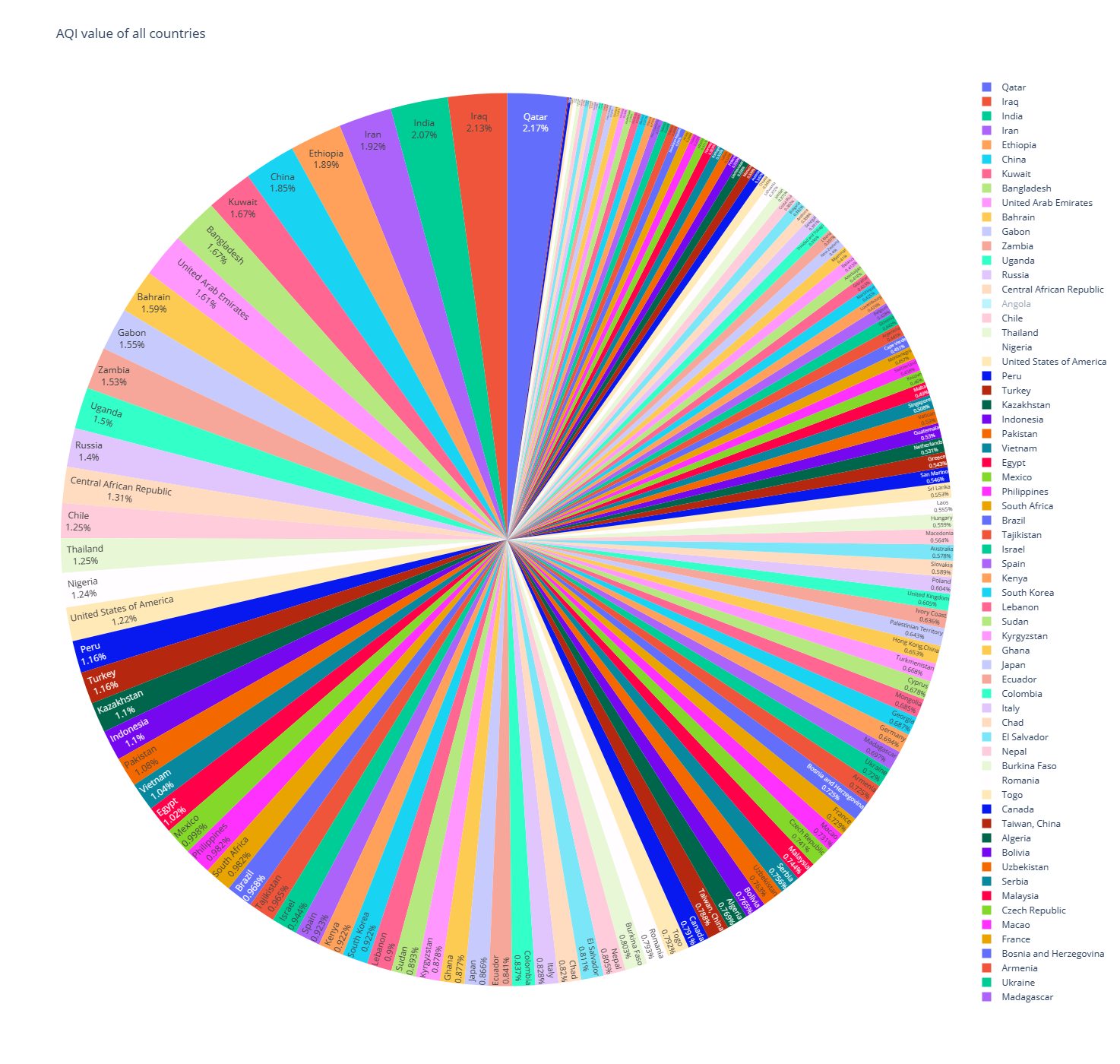
同样打印AQI分类类别，以Good（良好）、Moderate（中度）、Unhealthy for Sensitive Groups（对敏感群体不健康）、Unhealthy（不健康）、Very Unhealthy（非常不健康）和Hazardous（危险） 6类指示空间质量（对公众健康影响的空气污染程度）。

fig=px.choropleth(aqi\_df,locations="Country",   
 locationmode='country names', color="Status",   
 animation\_frame="Date",range\_color=[25,450], width=1400, height=700)  
fig.show()



通过饼状图统计63天AQI测量值各国所占比例，从结果可以看到从7月末到9月末，空气污染程度较高的前10个国家为：Qatar（卡塔尔）、Iraq（伊拉克）、India（印度）、Iran（伊朗）、Ethiopia（埃塞俄比亚）、China（中国）、Kuwait（科威特）、Bangladesh（孟加拉国）、United Arab Emirates（阿拉伯联合酋长国）和Bahrain（巴林）等。

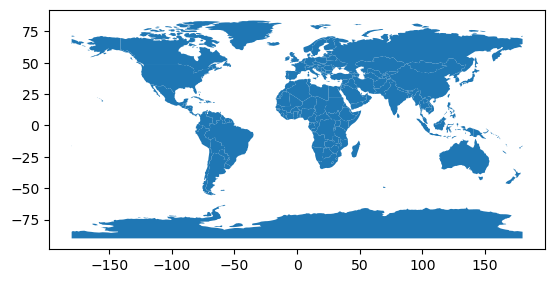
fig = px.pie(aqi\_df, names="Country" ,  
 values="AQI Value", title="AQI value of all countries",   
 hover\_data=["Status"],width=1400, height=1400)  
fig.update\_traces(textposition='inside', textinfo='percent+label')



* 世界地图国家边界

GeoPandas库提供了世界地图国家边界数据，可以直接读取。同样需要对不正确的国家名称进行修正。

import geopandas as gpd  
world=gpd.read\_file(gpd.datasets.get\_path('naturalearth\_lowres'))  
world["name"]=world["name"].replace({ 'Hong Kong':'Hong Kong,China',  
 'Taiwan': 'Taiwan, China'})  
  
world.plot();



首先需要明确AQI数据集中国家类别Country列，与世界地图国家边界（国界）地理数据中国家类别name列中的国家名是否一致。定义str\_listAINlistB()函数进行检查，打印AQI数据中出现而国界中未找到对应的国家或地区名有27个，这27个数据行将不被用于后续分析。

def str\_listAINlistB(listA,listB,lower=False):  
 '''  
 核实一个字符串列表元素（listA）是否在另一个字符串列表（listB）中，或被字符所包含，例如`abc`在'abc def'中也计入。  
  
 Parameters  
 ----------  
 listA : list(string)  
 字符传列表A.  
 listB : list(string)  
 字符传列表B.  
 lower : bool, optional  
 是否字符全部转小写，即区分大小写. The default is False.  
  
 Returns  
 -------  
 elements\_in : dict(string:int)  
 如果为0则不包含，否则包含.  
  
 '''   
 if lower:  
 listA=[i.lower() for i in listA]  
 listB=[i.lower() for i in listB]  
 elements\_in={}  
 for e in listA:   
 is\_in=e in listB   
 if is\_in:  
 elements\_in[e]=1  
 else:  
 is\_in\_sum=sum([e in i for i in listB])  
 if is\_in\_sum>0:  
 elements\_in[e]=1  
 else:  
 elements\_in[e]=0   
 return elements\_in  
   
c\_n\_aqi=aqi\_df.Country.unique()  
c\_n\_world=world.name.unique()   
c\_n\_in=str\_listAINlistB(c\_n\_aqi,c\_n\_world,lower=True)   
c\_n\_notin={k:v for k,v in c\_n\_in.items() if v==0}  
print(c\_n\_notin)

{'andorra': 0, 'bahrain': 0, 'bermuda': 0, 'bosnia and herzegovina': 0, 'cape verde': 0, 'cayman islands': 0, 'central african republic': 0, 'czech republic': 0, 'dominican republic': 0, 'french guiana': 0, 'gibraltar': 0, 'grenada': 0, 'guadeloupe': 0, 'guam': 0, 'hong kong,china': 0, 'ivory coast': 0, 'jersey': 0, 'liechtenstein': 0, 'macao': 0, 'malta': 0, 'martinique': 0, 'monaco': 0, 'palestinian territory': 0, 'reunion': 0, 'san marino': 0, 'singapore': 0, 'vatican': 0}

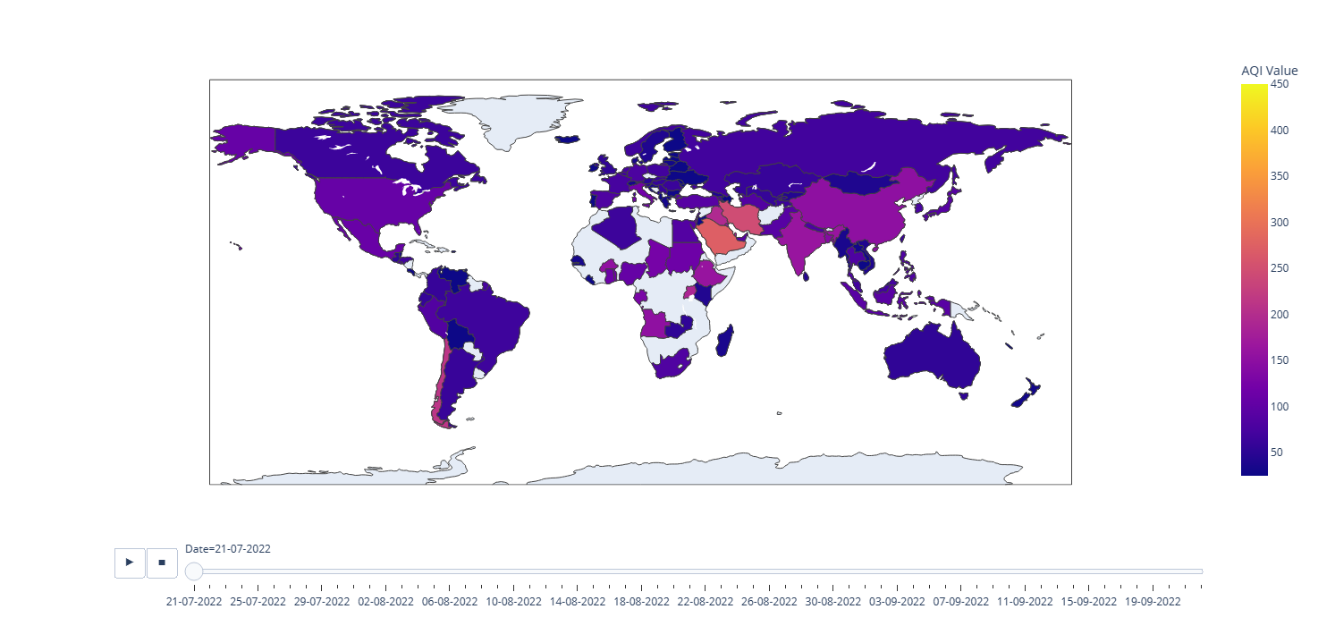
AQI数据集为非地理空间数据，需要按照国名合入国界地理信息数据，通过pd.merge方法，指定合并关键列进行合并。打印合并后的GeoDataFrame格式数据，观察数据缺失情况，其中蓝灰偏白区域没有数据。

aqi\_world\_df=pd.merge(aqi\_df,world,how='left',left\_on=['Country'],right\_on=['name'])  
aqi\_world\_gdf=gpd.GeoDataFrame(aqi\_world\_df,geometry='geometry',crs=world.crs)  
aqi\_world\_gdf.dropna(subset=['geometry'],inplace=True)  
aqi\_world\_gdf

|  | **Date** | **Country** | **Status** | **AQI Value** | **ts** | **pop\_est** | **continent** | **name** | **iso\_a3** | **gdp\_md\_est** | **geometry** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 21-07-2022 | Albania | Good | 14 | 2022-07-21 | 2854191.0 | Europe | Albania | ALB | 15279.0 | POLYGON ((21.02004 40.84273, 20.99999 40.58000... |
| **1** | 21-07-2022 | Algeria | Moderate | 65 | 2022-07-21 | 43053054.0 | Africa | Algeria | DZA | 171091.0 | POLYGON ((-8.68440 27.39574, -8.66512 27.58948... |
| **3** | 21-07-2022 | Angola | Unhealthy for Sensitive Groups | 113 | 2022-07-21 | 31825295.0 | Africa | Angola | AGO | 88815.0 | MULTIPOLYGON (((12.99552 -4.78110, 12.63161 -4... |
| **4** | 21-07-2022 | Argentina | Moderate | 63 | 2022-07-21 | 44938712.0 | South America | Argentina | ARG | 445445.0 | MULTIPOLYGON (((-68.63401 -52.63637, -68.25000... |
| **5** | 21-07-2022 | Armenia | Moderate | 76 | 2022-07-21 | 2957731.0 | Asia | Armenia | ARM | 13672.0 | POLYGON ((46.50572 38.77061, 46.14362 38.74120... |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **9988** | 22-09-2022 | United States of America | Unhealthy for Sensitive Groups | 109 | 2022-09-22 | 328239523.0 | North America | United States of America | USA | 21433226.0 | MULTIPOLYGON (((-122.84000 49.00000, -120.0000... |
| **9989** | 22-09-2022 | Uzbekistan | Moderate | 66 | 2022-09-22 | 33580650.0 | Asia | Uzbekistan | UZB | 57921.0 | POLYGON ((55.96819 41.30864, 55.92892 44.99586... |
| **9991** | 22-09-2022 | Venezuela | Good | 20 | 2022-09-22 | 28515829.0 | South America | Venezuela | VEN | 482359.0 | POLYGON ((-60.73357 5.20028, -60.60118 4.91810... |
| **9992** | 22-09-2022 | Vietnam | Moderate | 76 | 2022-09-22 | 96462106.0 | Asia | Vietnam | VNM | 261921.0 | POLYGON ((104.33433 10.48654, 105.19991 10.889... |
| **9993** | 22-09-2022 | Zambia | Unhealthy | 161 | 2022-09-22 | 17861030.0 | Africa | Zambia | ZMB | 23309.0 | POLYGON ((30.74001 -8.34001, 31.15775 -8.59458... |

8012 rows × 11 columns

fig=px.choropleth(aqi\_world\_gdf,locations="Country",   
 locationmode='country names', color="AQI Value",   
 animation\_frame="Date",range\_color=[25,450], width=1400, height=700)  
fig.show()



将预处理后的数据存入本地硬盘，方便后续调用。

aqi\_world\_gdf.to\_file(args.data.aqi\_world,driver='GeoJSON')

### 2.7.5.2 基于等级方式的空间动力学

#### 1）经典肯德尔等级相关系数（Kendall’s Tau）

Kendall’s Tau是一种非参数测量，用于衡量排名（等级）数据列之间的关系。Tau相关系数返回一个-1~1的值，其中0表示排名列间没有关系，1表示有完美的正相关关系，值越趋近于1关联性越强，反之趋近于-1则为负相关。以[Kendall’s Tau (Kendall Rank Correlation Coefficient)](https://www.statisticshowto.com/kendalls-tau/)⑥提供的案例数据为例，通过建立简单的数据样本，根据公式书写代码逐步计算。其公式为： Kendall’sTau=C+DC–D​，式中C为协和对数量（concordant pairs）； D为不协和对数量（disconcordant pairs）。区分协和对和不协和对的方式可以通过固定排序rank\_1列，比较rank\_2列每1行数据与其后数据大小的比较获得，如果大于为协和对；小于为不协和对；等于为无效对。如果不对rank\_1先排序，则可以通过对Xj​−Xi​Yj​−Yi​​值与0比较获得，式中， Xj​和Yi​为两个等级数列值，如果大于0为协和对；小于0为不协和对；等于0为无效对。

import pandas as pd  
  
rank\_dict=dict(name=list(map(lambda x:chr(x).upper(),range(97,97+12))),  
 rank\_1=list(range(1,13)),  
 rank\_2=[1,2,4,3,6,5,8,7,10,9,12,11])  
  
rank\_df=pd.DataFrame(rank\_dict)  
  
rank\_2=rank\_df.rank\_2.to\_list()  
concordant=[sum(map(lambda x:x>rank\_2[i],rank\_2[i:])) for i in range(len(rank\_2))]  
discordant=[sum(map(lambda x:x<rank\_2[i],rank\_2[i:])) for i in range(len(rank\_2))]  
rank\_df['concordant']=concordant  
rank\_df['discordant']=discordant  
rank\_df

|  | **name** | **rank\_1** | **rank\_2** | **concordant** | **discordant** |
| --- | --- | --- | --- | --- | --- |
| **0** | A | 1 | 1 | 11 | 0 |
| **1** | B | 2 | 2 | 10 | 0 |
| **2** | C | 3 | 4 | 8 | 1 |
| **3** | D | 4 | 3 | 8 | 0 |
| **4** | E | 5 | 6 | 6 | 1 |
| **5** | F | 6 | 5 | 6 | 0 |
| **6** | G | 7 | 8 | 4 | 1 |
| **7** | H | 8 | 7 | 4 | 0 |
| **8** | I | 9 | 10 | 2 | 1 |
| **9** | J | 10 | 9 | 2 | 0 |
| **10** | K | 11 | 12 | 0 | 1 |
| **11** | L | 12 | 11 | 0 | 0 |

通过上述对各行协和对和不协和对数量的统计，带入Kendall’s Tau公式计算结果值为0.848，表明rank\_1和ranki\_2列具有很强的正相关性。

c\_sum=sum(concordant)  
d\_sum=sum(discordant)  
Kendalls\_Tau=(c\_sum-d\_sum)/(c\_sum+d\_sum)  
print(f"concordant: {concordant}, c\_sum={c\_sum};\ndiscordant: {discordant}, d\_sum={d\_sum};\nKendall’s Tau={round(Kendalls\_Tau,3)}")

concordant: [11, 10, 8, 8, 6, 6, 4, 4, 2, 2, 0, 0], c\_sum=61;  
discordant: [0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0], d\_sum=5;  
Kendall’s Tau=0.848

giddy库提供了giddy.rank.Tau方法直接计算Kendall’s Tau，其公式⑦为：  
2τ=(n(n−1))/2c−d​，式中c即为协和对数量， d为不协和对数量， n为等级列行数（或解释为空间单元数），τ值位于[-1,1]区间。通过giddy库提供的方法计算结果为0.848，与上文计算结果同。

import giddy  
  
tau=giddy.rank.Tau(rank\_df["rank\_1"],rank\_df["rank\_2"])  
print(tau.concordant,tau.discordant,tau.tau.round(3),tau.tau\_p)

61.0 5.0 0.848 0.00012300294510098186

#### 2）空间Kendall’s Tau

空间Kendall’s Tau是将所有配对按指定的空间邻域分组，检查两组之间的等级相关性[9]，其公式为：τw​=ι′Wιι′(W∘S)ι​，式中， W为空间权重， S为协和矩阵（concordance matrix），ι为 (n,1)的统一向量。无效假设为等级交换的空间随机性。因为单独输入权重参数w，且使用libpysa（PySAL）库提供的block\_weights空间权重方法，因此τw​的推断计算将不受空间单元的数据行排列影响。block\_weights方法是根据定义的空间单元所属空间组（例如县界所属省界，国界所属大洲等）进行计算，例如libpysa库提供的案例⑧。

block\_weights空间权重

定义列表regimes为分组信息，包括类名为1，2，3等3个分组，空间权重值对应到该单元的邻里单元。

from libpysal.weights import block\_weights  
import numpy as np  
  
regimes=np.ones(25)  
regimes[range(10,20)]=2  
regimes[range(21,25)]=3  
print(f"空间分组（regime）:{regimes}")  
  
w=block\_weights(regimes)  
print(f"单元0的空间权重为：{w.weights[0]};\n单元0的邻里单元：{w.neighbors[0]}")  
print('所有空间单元的邻里单元：')  
w.neighbors

空间分组（regime）:[1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 2. 2. 2. 2. 2. 2. 2. 2. 2. 2. 1. 3. 3. 3.  
 3.]  
单元0的空间权重为：[1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0];  
单元0的邻里单元：[1, 2, 3, 4, 5, 6, 7, 8, 9, 20]  
所有空间单元的邻里单元：  
  
  
  
  
  
{0: [1, 2, 3, 4, 5, 6, 7, 8, 9, 20],  
 1: [0, 2, 3, 4, 5, 6, 7, 8, 9, 20],  
 2: [0, 1, 3, 4, 5, 6, 7, 8, 9, 20],  
 3: [0, 1, 2, 4, 5, 6, 7, 8, 9, 20],  
 4: [0, 1, 2, 3, 5, 6, 7, 8, 9, 20],  
 5: [0, 1, 2, 3, 4, 6, 7, 8, 9, 20],  
 6: [0, 1, 2, 3, 4, 5, 7, 8, 9, 20],  
 7: [0, 1, 2, 3, 4, 5, 6, 8, 9, 20],  
 8: [0, 1, 2, 3, 4, 5, 6, 7, 9, 20],  
 9: [0, 1, 2, 3, 4, 5, 6, 7, 8, 20],  
 20: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9],  
 10: [11, 12, 13, 14, 15, 16, 17, 18, 19],  
 11: [10, 12, 13, 14, 15, 16, 17, 18, 19],  
 12: [10, 11, 13, 14, 15, 16, 17, 18, 19],  
 13: [10, 11, 12, 14, 15, 16, 17, 18, 19],  
 14: [10, 11, 12, 13, 15, 16, 17, 18, 19],  
 15: [10, 11, 12, 13, 14, 16, 17, 18, 19],  
 16: [10, 11, 12, 13, 14, 15, 17, 18, 19],  
 17: [10, 11, 12, 13, 14, 15, 16, 18, 19],  
 18: [10, 11, 12, 13, 14, 15, 16, 17, 19],  
 19: [10, 11, 12, 13, 14, 15, 16, 17, 18],  
 21: [22, 23, 24],  
 22: [21, 23, 24],  
 23: [21, 22, 24],  
 24: [21, 22, 23]}

预处理的数据按照时间列组织每一样本的信息，为方便空间Kendall’s Tau计算，将数据形式转换为国家名（空间单元）Country和所属大洲（单元分组）continent列为行索引，以测量时间ts为列组织数据，核心使用的方法是pandas提供的pivot\_table方法。

import geopandas as gpd  
aqi\_world\_gdf=gpd.read\_file(args.data.aqi\_world)  
aqi\_world\_gdf

|  | **Date** | **Country** | **Status** | **AQI Value** | **ts** | **pop\_est** | **continent** | **name** | **iso\_a3** | **gdp\_md\_est** | **geometry** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 21-07-2022 | Albania | Good | 14 | 2022-07-21 | 2854191.0 | Europe | Albania | ALB | 15279.0 | POLYGON ((21.02004 40.84273, 20.99999 40.58000... |
| **1** | 21-07-2022 | Algeria | Moderate | 65 | 2022-07-21 | 43053054.0 | Africa | Algeria | DZA | 171091.0 | POLYGON ((-8.68440 27.39574, -8.66512 27.58948... |
| **2** | 21-07-2022 | Angola | Unhealthy for Sensitive Groups | 113 | 2022-07-21 | 31825295.0 | Africa | Angola | AGO | 88815.0 | MULTIPOLYGON (((12.99552 -4.78110, 12.63161 -4... |
| **3** | 21-07-2022 | Argentina | Moderate | 63 | 2022-07-21 | 44938712.0 | South America | Argentina | ARG | 445445.0 | MULTIPOLYGON (((-68.63401 -52.63637, -68.25000... |
| **4** | 21-07-2022 | Armenia | Moderate | 76 | 2022-07-21 | 2957731.0 | Asia | Armenia | ARM | 13672.0 | POLYGON ((46.50572 38.77061, 46.14362 38.74120... |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **8007** | 22-09-2022 | United States of America | Unhealthy for Sensitive Groups | 109 | 2022-09-22 | 328239523.0 | North America | United States of America | USA | 21433226.0 | MULTIPOLYGON (((-122.84000 49.00000, -120.0000... |
| **8008** | 22-09-2022 | Uzbekistan | Moderate | 66 | 2022-09-22 | 33580650.0 | Asia | Uzbekistan | UZB | 57921.0 | POLYGON ((55.96819 41.30864, 55.92892 44.99586... |
| **8009** | 22-09-2022 | Venezuela | Good | 20 | 2022-09-22 | 28515829.0 | South America | Venezuela | VEN | 482359.0 | POLYGON ((-60.73357 5.20028, -60.60118 4.91810... |
| **8010** | 22-09-2022 | Vietnam | Moderate | 76 | 2022-09-22 | 96462106.0 | Asia | Vietnam | VNM | 261921.0 | POLYGON ((104.33433 10.48654, 105.19991 10.889... |
| **8011** | 22-09-2022 | Zambia | Unhealthy | 161 | 2022-09-22 | 17861030.0 | Africa | Zambia | ZMB | 23309.0 | POLYGON ((30.74001 -8.34001, 31.15775 -8.59458... |

8012 rows × 11 columns

import numpy as np  
aqi\_world\_gdf\_pivot=aqi\_world\_gdf[['Country','AQI Value','continent','ts']].pivot\_table(index='ts',columns=['Country','continent'],aggfunc='mean', fill\_value=np.nan).T  
aqi\_world\_gdf\_pivot

|  |  | **ts** | **2022-07-21** | **2022-07-22** | **2022-07-23** | **2022-07-24** | **2022-07-25** | **2022-07-26** | **2022-07-27** | **2022-07-28** | **2022-07-29** | **2022-07-30** | **...** | **2022-09-13** | **2022-09-14** | **2022-09-15** | **2022-09-16** | **2022-09-17** | **2022-09-18** | **2022-09-19** | **2022-09-20** | **2022-09-21** | **2022-09-22** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Country** | **continent** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **AQI Value** | **Albania** | **Europe** | 15.5 | 16.666667 | 17.5 | 18.0 | 55.0 | 43.0 | 41.0 | 61.0 | 22.0 | 111.0 | ... | 7.0 | 11.0 | 13.0 | 28.0 | 11.0 | 5.0 | 5.0 | 5.0 | 8.0 | 9.5 |
| **Algeria** | **Africa** | 65.0 | 65.000000 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | ... | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 |
| **Angola** | **Africa** | 132.5 | 123.333333 | 107.0 | 107.0 | 117.0 | 102.5 | 123.0 | 152.0 | 117.0 | 142.0 | ... | 70.0 | 77.0 | 115.0 | 117.0 | 98.0 | 82.0 | 91.0 | 87.0 | 91.0 | 87.5 |
| **Argentina** | **South America** | 62.0 | 57.333333 | 55.5 | 42.0 | 74.0 | 39.0 | 33.0 | 23.0 | 31.0 | 32.0 | ... | 47.0 | 39.0 | 23.0 | 33.0 | 18.0 | 29.0 | 21.0 | 28.0 | 61.0 | 22.0 |
| **Armenia** | **Asia** | 77.0 | 52.666667 | 41.5 | 43.0 | 78.0 | 61.0 | 58.0 | 63.0 | 41.0 | 54.0 | ... | 76.0 | 69.0 | 90.0 | 62.0 | 45.0 | 52.0 | 67.0 | 64.0 | 65.0 | 59.0 |
| **...** | **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **United States of America** | **North America** | 111.5 | 89.666667 | 86.5 | 85.0 | 84.0 | 78.5 | 90.0 | 80.0 | 79.0 | 70.0 | ... | 162.0 | 151.0 | 105.0 | 364.0 | 122.0 | 92.0 | 92.0 | 98.0 | 110.0 | 104.5 |
| **Uzbekistan** | **Asia** | 59.0 | 90.000000 | 77.0 | 73.0 | 57.0 | 56.5 | 4.0 | 4.0 | 4.0 | 26.0 | ... | 57.0 | 89.0 | 65.0 | NaN | NaN | NaN | 40.0 | 78.0 | 95.0 | 54.5 |
| **Venezuela** | **South America** | 2.5 | 13.000000 | 20.0 | 4.0 | 5.0 | 16.0 | 31.0 | 21.0 | 2.0 | 13.0 | ... | 15.0 | 3.0 | 62.0 | 27.0 | 7.0 | 10.0 | 13.0 | 8.0 | 40.0 | 18.5 |
| **Vietnam** | **Asia** | 42.0 | 85.333333 | 103.0 | 92.0 | 84.0 | 74.0 | 107.0 | 118.0 | 83.0 | 75.0 | ... | 143.0 | 138.0 | 109.0 | 93.0 | 102.0 | 102.0 | 97.0 | 116.0 | 110.0 | 70.5 |
| **Zambia** | **Africa** | 46.5 | 67.333333 | 80.0 | 152.0 | 89.0 | 64.5 | 89.0 | 91.0 | 105.0 | 80.0 | ... | 176.0 | 118.0 | 116.0 | 99.0 | 107.0 | 117.0 | 184.0 | 155.0 | 151.0 | 147.5 |

114 rows × 64 columns

查看以测量时间为列的列名，包括有哪些测量时间。

aqi\_world\_gdf\_pivot.columns

DatetimeIndex(['2022-07-21', '2022-07-22', '2022-07-23', '2022-07-24',  
 '2022-07-25', '2022-07-26', '2022-07-27', '2022-07-28',  
 '2022-07-29', '2022-07-30', '2022-07-31', '2022-08-01',  
 '2022-08-02', '2022-08-03', '2022-08-04', '2022-08-05',  
 '2022-08-06', '2022-08-07', '2022-08-08', '2022-08-09',  
 '2022-08-10', '2022-08-11', '2022-08-12', '2022-08-13',  
 '2022-08-14', '2022-08-15', '2022-08-16', '2022-08-17',  
 '2022-08-18', '2022-08-19', '2022-08-20', '2022-08-21',  
 '2022-08-22', '2022-08-23', '2022-08-24', '2022-08-25',  
 '2022-08-26', '2022-08-27', '2022-08-28', '2022-08-29',  
 '2022-08-30', '2022-08-31', '2022-09-01', '2022-09-02',  
 '2022-09-03', '2022-09-04', '2022-09-05', '2022-09-06',  
 '2022-09-07', '2022-09-08', '2022-09-09', '2022-09-10',  
 '2022-09-11', '2022-09-12', '2022-09-13', '2022-09-14',  
 '2022-09-15', '2022-09-16', '2022-09-17', '2022-09-18',  
 '2022-09-19', '2022-09-20', '2022-09-21', '2022-09-22'],  
 dtype='datetime64[ns]', name='ts', freq=None)

如果测量值行中存在空值，将无法进行空间Kendall’s Tau计算，因此移除存在空值的行。原国家数含114个，有空值的行为5个，有效国家行为109个。

aqi\_world\_gdf\_pivot.dropna(inplace=True)  
aqi\_world\_gdf\_pivot.shape

(109, 64)

giddy库提供的giddy.rank.Tau经典Kendall’s Tau计算和giddy.rank.SpatialTau空间Kendall’s Tau计算，计算模式为一次提供两组等级数列，以2022-07-21和2022-09-22两列AQI测量值为例计算。首先计算经典的Kendall’s Tau值，其结果为0.527，p-value值几近为0，因此该两个时期AQI数据具有相关性。

from datetime import datetime  
str2datetime=lambda dt\_str:datetime.strptime(dt\_str,'%Y-%m-%d')  
dt1=str2datetime('2022-07-21')  
dt2=str2datetime('2022-09-22')

* giddy.rank.Tau经典Kendall’s Tau

import giddy  
aqi\_world\_tau=giddy.rank.Tau(aqi\_world\_gdf\_pivot[dt1],aqi\_world\_gdf\_pivot[dt2])

giddy\_rank\_tau\_info\_print=lambda tau:print(f"concordant: {tau.concordant};\ndiscordant: {tau.discordant};\ntau: {tau.tau};\ntau p-value: {tau.tau\_p}")  
giddy\_rank\_tau\_info\_print(aqi\_world\_tau)

concordant: 4471.0;  
discordant: 1375.0;  
tau: 0.5278322873099652;  
tau p-value: 4.1083562233353125e-16

空间Kendall’s Tau计算需要输入空间权重值，以国家所属大洲为分组计算block\_weights空间权重。为方便Country和continent数据提取，将其行索引值转换为列。

aqi\_world\_gdf\_pivot.index

MultiIndex([('AQI Value', 'Albania', 'Europe'),  
 ('AQI Value', 'Algeria', 'Africa'),  
 ('AQI Value', 'Angola', 'Africa'),  
 ('AQI Value', 'Argentina', 'South America'),  
 ('AQI Value', 'Armenia', 'Asia'),  
 ('AQI Value', 'Australia', 'Oceania'),  
 ('AQI Value', 'Austria', 'Europe'),  
 ('AQI Value', 'Azerbaijan', 'Asia'),  
 ('AQI Value', 'Bangladesh', 'Asia'),  
 ('AQI Value', 'Belarus', 'Europe'),  
 ...  
 ('AQI Value', 'Togo', 'Africa'),  
 ('AQI Value', 'Trinidad and Tobago', 'North America'),  
 ('AQI Value', 'Turkey', 'Asia'),  
 ('AQI Value', 'Uganda', 'Africa'),  
 ('AQI Value', 'Ukraine', 'Europe'),  
 ('AQI Value', 'United Arab Emirates', 'Asia'),  
 ('AQI Value', 'United Kingdom', 'Europe'),  
 ('AQI Value', 'United States of America', 'North America'),  
 ('AQI Value', 'Vietnam', 'Asia'),  
 ('AQI Value', 'Zambia', 'Africa')],  
 names=[None, 'Country', 'continent'], length=109)

aqi\_world\_gdf\_pivot.reset\_index(level=['Country','continent'],inplace=True)

aqi\_world\_gdf\_pivot

| **ts** | **Country** | **continent** | **2022-07-21 00:00:00** | **2022-07-22 00:00:00** | **2022-07-23 00:00:00** | **2022-07-24 00:00:00** | **2022-07-25 00:00:00** | **2022-07-26 00:00:00** | **2022-07-27 00:00:00** | **2022-07-28 00:00:00** | **...** | **2022-09-13 00:00:00** | **2022-09-14 00:00:00** | **2022-09-15 00:00:00** | **2022-09-16 00:00:00** | **2022-09-17 00:00:00** | **2022-09-18 00:00:00** | **2022-09-19 00:00:00** | **2022-09-20 00:00:00** | **2022-09-21 00:00:00** | **2022-09-22 00:00:00** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **AQI Value** | Albania | Europe | 15.5 | 16.666667 | 17.5 | 18.0 | 55.0 | 43.0 | 41.0 | 61.0 | ... | 7.0 | 11.0 | 13.0 | 28.0 | 11.0 | 5.0 | 5.0 | 5.0 | 8.0 | 9.5 |
| **AQI Value** | Algeria | Africa | 65.0 | 65.000000 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | ... | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 |
| **AQI Value** | Angola | Africa | 132.5 | 123.333333 | 107.0 | 107.0 | 117.0 | 102.5 | 123.0 | 152.0 | ... | 70.0 | 77.0 | 115.0 | 117.0 | 98.0 | 82.0 | 91.0 | 87.0 | 91.0 | 87.5 |
| **AQI Value** | Argentina | South America | 62.0 | 57.333333 | 55.5 | 42.0 | 74.0 | 39.0 | 33.0 | 23.0 | ... | 47.0 | 39.0 | 23.0 | 33.0 | 18.0 | 29.0 | 21.0 | 28.0 | 61.0 | 22.0 |
| **AQI Value** | Armenia | Asia | 77.0 | 52.666667 | 41.5 | 43.0 | 78.0 | 61.0 | 58.0 | 63.0 | ... | 76.0 | 69.0 | 90.0 | 62.0 | 45.0 | 52.0 | 67.0 | 64.0 | 65.0 | 59.0 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **AQI Value** | United Arab Emirates | Asia | 91.0 | 84.333333 | 89.5 | 174.0 | 115.0 | 289.0 | 108.0 | 120.0 | ... | 118.0 | 101.0 | 83.0 | 94.0 | 99.0 | 93.0 | 83.0 | 95.0 | 129.0 | 103.5 |
| **AQI Value** | United Kingdom | Europe | 55.0 | 46.333333 | 41.5 | 39.0 | 33.0 | 34.0 | 36.0 | 53.0 | ... | 54.0 | 63.0 | 54.0 | 41.0 | 47.0 | 36.0 | 38.0 | 46.0 | 52.0 | 57.0 |
| **AQI Value** | United States of America | North America | 111.5 | 89.666667 | 86.5 | 85.0 | 84.0 | 78.5 | 90.0 | 80.0 | ... | 162.0 | 151.0 | 105.0 | 364.0 | 122.0 | 92.0 | 92.0 | 98.0 | 110.0 | 104.5 |
| **AQI Value** | Vietnam | Asia | 42.0 | 85.333333 | 103.0 | 92.0 | 84.0 | 74.0 | 107.0 | 118.0 | ... | 143.0 | 138.0 | 109.0 | 93.0 | 102.0 | 102.0 | 97.0 | 116.0 | 110.0 | 70.5 |
| **AQI Value** | Zambia | Africa | 46.5 | 67.333333 | 80.0 | 152.0 | 89.0 | 64.5 | 89.0 | 91.0 | ... | 176.0 | 118.0 | 116.0 | 99.0 | 107.0 | 117.0 | 184.0 | 155.0 | 151.0 | 147.5 |

109 rows × 66 columns

* giddy.rank.SpatialTau空间Kendall’s Tau

from libpysal.weights import block\_weights  
w=block\_weights(aqi\_world\_gdf\_pivot["continent"])

np.random.seed(12345)  
aqi\_world\_tau\_w=giddy.rank.SpatialTau(aqi\_world\_gdf\_pivot\_m[dt1],aqi\_world\_gdf\_pivot\_m[dt2],w,999)

空间Kendall’s Tau计算结果为1，p-value为0.001，拒绝等级交换空间随机性的无效假设，表明按照大洲分组空间单元，两组等级数列间完美相关，有大量等级交换发生在同一大洲的不同国家间。

giddy\_rank\_spatialTau\_info\_print=lambda stau:print(f"concordant: {stau.concordant};\nconcordant spatial: {stau.concordant\_spatial};\ndiscordant: {stau.discordant};\ndiscordant spatial: {stau.discordant\_spatial};\ntau spatial: {stau.tau\_spatial};\ntau spatial psim: {stau.tau\_spatial\_psim}")  
giddy\_rank\_spatialTau\_info\_print(aqi\_world\_tau\_w)

concordant: 22538311.0;  
concordant spatial: 685;  
discordant: 6931375.0;  
discordant spatial: 0;  
tau spatial: 1.0;  
tau spatial psim: 0.001

* giddy.rank.Tau\_Regional 区域（分组）间和区域内Kendall’s Tau的分解

Kendall's τ的区域间和区域内分解提供了一个关于交换流动模式的中观视角。该方法不是考察整个研究区域内任何两个相邻空间单元之间的协和关系，而是区域内任何两个空间单元（邻里单元）之间的协和关系，例如Asia大洲，得出Asia大洲区域内的协和统计；也可以考察区域间例如Asia大洲和Europe大洲非邻里单元之间的协和关系，从而得出区域间的协和统计。进而可以寻找到承载最多等级交换的特定区域。如果有k个区域，就会有k个区域内协和统计量，和 (k−1)2个区域间协和统计量。可以将上述区域间和区域内协和统计量组织成一个 (k,k)矩阵，其中对角线元素是区域内协和统计量，非对角线元素是区域间协和统计量，其公式为： T=PHP′P(H∘S)P′​，式中， P是一个 (k,n)二元矩阵，如果空间单元i在j区域，则pj,i​=1，否则pj,i​=0； H是一个 (n,n)矩阵，对角线上的值为0，其它位置值为1； ∘∘ 为哈达玛积（Hadamard Product），当两个矩阵为同阶矩阵时，两矩阵的哈达玛积为对应位置元素相乘。

计算结果属性tau\_reg给出了区域间和区域内协和统计矩阵，较高的数值代表较低的交换流动性；负值则为较高的交换流动性。属性tau\_reg\_pvalues返回p-value。配置显著性水平为0.05，不满足要求的位置赋值为0，可以观察区域间和区域内有效协和统计矩阵值。

np.random.seed(12345)  
  
aqi\_world\_tau\_regional\_w=giddy.rank.Tau\_Regional(aqi\_world\_gdf\_pivot[dt1],aqi\_world\_gdf\_pivot[dt2],aqi\_world\_gdf\_pivot['continent'],999)

aqi\_world\_tau\_regional\_w.tau\_reg.round(3)

array([[ 0.255, 0.343, 0.619, 0.617, 0.889, -0.016],  
 [ 0.343, 0.568, 0.627, 0.626, 0.752, 0.261],  
 [ 0.619, 0.627, 0.538, 0.622, 0.426, 0.353],  
 [ 0.617, 0.626, 0.622, 0.733, 0.6 , 0.4 ],  
 [ 0.889, 0.752, 0.426, 0.6 , 0.333, 0.524],  
 [-0.016, 0.261, 0.353, 0.4 , 0.524, 0.19 ]])

aqi\_world\_tau\_regional\_w.tau\_reg\_pvalues

array([[0.023, 0.003, 0.055, 0.179, 0.01 , 0.001],  
 [0.003, 0.303, 0.005, 0.096, 0.071, 0.005],  
 [0.055, 0.005, 0.465, 0.097, 0.237, 0.051],  
 [0.179, 0.096, 0.097, 0.135, 0.458, 0.215],  
 [0.01 , 0.071, 0.237, 0.458, 0.518, 0.544],  
 [0.001, 0.005, 0.051, 0.215, 0.544, 0.114]])

aqi\_world\_tau\_regional\_w\_significant=aqi\_world\_tau\_regional\_w.tau\_reg \* (aqi\_world\_tau\_regional\_w.tau\_reg\_pvalues<0.05)  
aqi\_world\_tau\_regional\_w\_significant.round(3)

array([[ 0.255, 0.343, 0. , 0. , 0.889, -0.016],  
 [ 0.343, 0. , 0.627, 0. , 0. , 0.261],  
 [ 0. , 0.627, 0. , 0. , 0. , 0. ],  
 [ 0. , 0. , 0. , 0. , 0. , 0. ],  
 [ 0.889, 0. , 0. , 0. , 0. , 0. ],  
 [-0.016, 0.261, 0. , 0. , 0. , 0. ]])

* giddy.rank.Tau\_Local 局部Kendall’s Tau

局部Kendall’s Tau 是经典Kendall’s Tau的局部分解，提供了空间单元*r*的等级变化对整体交换流动水平的贡献[10]。对于空间单元*r*，其公式为：*τr*​=*n*−1*cr*​−*dr*​​，式中*cr*​是除了空间单元*r*外的空间单元与*r*空间单元的协和对数量； *dr*​是不协和对的数量。*τr*​值位于[-1,1]区间，值越大，单元*r*的交换流动性越低。

下述计算结果显示了排序后前后5行的局部Kendall’s Tau值，Bolivia值为-0.380，表明Bolivia在2022-07-21和2022-09-21两列AQI观测值间与很多国家交换了排名；相反，Puerto Rico值为0.991，凸显了Puerto Rico的AQI排名的高度稳定性。

tau\_r=giddy.rank.Tau\_Local(aqi\_world\_gdf\_pivot[dt1],aqi\_world\_gdf\_pivot[dt2])  
local\_tau=pd.DataFrame({"Country":aqi\_world\_gdf\_pivot['Country'].tolist(),"$\\tau\_r$":tau\_r.tau\_local})  
local\_tau.sort\_values(by=["$\\tau\_r$"]).round(3)

|  | **Country** | **$\tau\_r$** |
| --- | --- | --- |
| **12** | Bolivia | -0.380 |
| **108** | Zambia | -0.120 |
| **68** | Mongolia | -0.056 |
| **57** | Kyrgyzstan | 0.009 |
| **16** | Burkina Faso | 0.037 |
| **...** | ... | ... |
| **83** | Qatar | 0.907 |
| **47** | Iraq | 0.926 |
| **46** | Iran | 0.926 |
| **43** | Iceland | 0.981 |
| **82** | Puerto Rico | 0.991 |

109 rows × 2 columns

* giddy.rank.Tau\_Local\_Neighbor流动性关联的局部指标（Local indicator of mobility association，LIMA）

为了揭示空间在塑造每个空间单元间的交换流动模式方面的作用，使用局部Kendall’s Tau的两个空间变体：neighbor（邻居） set LIMA 和neighborhood （邻里）set LIMA[10]。邻里变体是局部Kendall’s Tau分解（相邻和不相邻）和空间Kendall’s Tau分解的共同结果。

**Neighbor set LIMA（邻居）**

Neighbor变体不像局部Kendall’s Tau的计算方式，分析一个目标空间单元r与所有其它空间单元之间的协和关系，而是分析目标空间单元r和其邻接空间单元之间的协和关系，其公式定义如下：τ~r​=∑b​wr,b​∑b​wr,b​sr,b​​，式中， wr,b​为目标单元r与邻接单元b之间的空间权重； sr,b​为空间单元r与邻接单元b的协和对矩阵。

aqi\_world\_tau\_local\_neighbor\_w=giddy.rank.Tau\_Local\_Neighbor(aqi\_world\_gdf\_pivot[dt1],aqi\_world\_gdf\_pivot[dt2],w,999)

aqi\_world\_tau\_local\_neighbor\_w.tau\_ln.round(3).shape

(109,)

使用pivot\_table转换GeoDataFrame数据时，并未链接geometry的地理数据信息，因此需要单独通过merge方法合并并配置投影，同时将Neighbor set LIMA计算结果置入其中，用于地图打印。

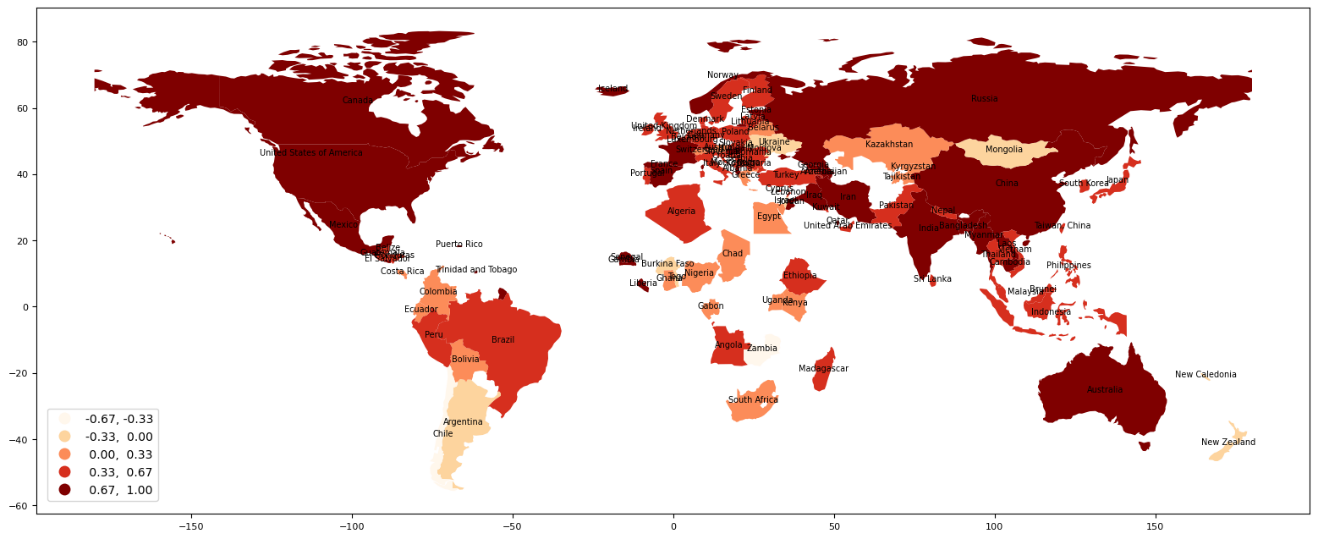
aqi\_world\_gdf\_pivot['tau\_ln']=aqi\_world\_tau\_local\_neighbor\_w.tau\_ln  
aqi\_world\_gdf\_pivot\_df=pd.merge(aqi\_world\_gdf\_pivot,world,how='left',left\_on=['Country'],right\_on=['name'])  
aqi\_world\_gdf\_pivot\_gdf=gpd.GeoDataFrame(aqi\_world\_gdf\_pivot\_df,geometry='geometry',crs=world.crs)  
aqi\_world\_gdf\_pivot\_gdf

|  | **Country** | **continent\_x** | **2022-07-21 00:00:00** | **2022-07-22 00:00:00** | **2022-07-23 00:00:00** | **2022-07-24 00:00:00** | **2022-07-25 00:00:00** | **2022-07-26 00:00:00** | **2022-07-27 00:00:00** | **2022-07-28 00:00:00** | **...** | **2022-09-20 00:00:00** | **2022-09-21 00:00:00** | **2022-09-22 00:00:00** | **tau\_ln** | **pop\_est** | **continent\_y** | **name** | **iso\_a3** | **gdp\_md\_est** | **geometry** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | Albania | Europe | 15.5 | 16.666667 | 17.5 | 18.0 | 55.0 | 43.0 | 41.0 | 61.0 | ... | 5.0 | 8.0 | 9.5 | 0.857143 | 2854191.0 | Europe | Albania | ALB | 15279 | POLYGON ((21.02004 40.84273, 20.99999 40.58000... |
| **1** | Algeria | Africa | 65.0 | 65.000000 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | 65.0 | ... | 65.0 | 65.0 | 65.0 | 0.411765 | 43053054.0 | Africa | Algeria | DZA | 171091 | POLYGON ((-8.68440 27.39574, -8.66512 27.58948... |
| **2** | Angola | Africa | 132.5 | 123.333333 | 107.0 | 107.0 | 117.0 | 102.5 | 123.0 | 152.0 | ... | 87.0 | 91.0 | 87.5 | 0.411765 | 31825295.0 | Africa | Angola | AGO | 88815 | MULTIPOLYGON (((12.99552 -4.78110, 12.63161 -4... |
| **3** | Argentina | South America | 62.0 | 57.333333 | 55.5 | 42.0 | 74.0 | 39.0 | 33.0 | 23.0 | ... | 28.0 | 61.0 | 22.0 | 0.000000 | 44938712.0 | South America | Argentina | ARG | 445445 | MULTIPOLYGON (((-68.63401 -52.63637, -68.25000... |
| **4** | Armenia | Asia | 77.0 | 52.666667 | 41.5 | 43.0 | 78.0 | 61.0 | 58.0 | 63.0 | ... | 64.0 | 65.0 | 59.0 | 0.382353 | 2957731.0 | Asia | Armenia | ARM | 13672 | POLYGON ((46.50572 38.77061, 46.14362 38.74120... |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **104** | United Arab Emirates | Asia | 91.0 | 84.333333 | 89.5 | 174.0 | 115.0 | 289.0 | 108.0 | 120.0 | ... | 95.0 | 129.0 | 103.5 | 0.500000 | 9770529.0 | Asia | United Arab Emirates | ARE | 421142 | POLYGON ((51.57952 24.24550, 51.75744 24.29407... |
| **105** | United Kingdom | Europe | 55.0 | 46.333333 | 41.5 | 39.0 | 33.0 | 34.0 | 36.0 | 53.0 | ... | 46.0 | 52.0 | 57.0 | 0.485714 | 66834405.0 | Europe | United Kingdom | GBR | 2829108 | MULTIPOLYGON (((-6.19788 53.86757, -6.95373 54... |
| **106** | United States of America | North America | 111.5 | 89.666667 | 86.5 | 85.0 | 84.0 | 78.5 | 90.0 | 80.0 | ... | 98.0 | 110.0 | 104.5 | 1.000000 | 328239523.0 | North America | United States of America | USA | 21433226 | MULTIPOLYGON (((-122.84000 49.00000, -120.0000... |
| **107** | Vietnam | Asia | 42.0 | 85.333333 | 103.0 | 92.0 | 84.0 | 74.0 | 107.0 | 118.0 | ... | 116.0 | 110.0 | 70.5 | 0.647059 | 96462106.0 | Asia | Vietnam | VNM | 261921 | POLYGON ((104.33433 10.48654, 105.19991 10.889... |
| **108** | Zambia | Africa | 46.5 | 67.333333 | 80.0 | 152.0 | 89.0 | 64.5 | 89.0 | 91.0 | ... | 155.0 | 151.0 | 147.5 | -0.411765 | 17861030.0 | Africa | Zambia | ZMB | 23309 | POLYGON ((30.74001 -8.34001, 31.15775 -8.59458... |

109 rows × 73 columns

根据LIMA数值分布，从地图中可以得出，负值的国家例如Mongolia、Chile等国家与大部分邻接的国家交换了排名（等级）；而正值较大的数值，例如China、United States of America等与邻接的国家几乎没有交换过等级。

import util\_misc  
  
util\_misc.gdf\_plot\_annotate(aqi\_world\_gdf\_pivot\_gdf,"tau\_ln","Country",annotate\_fontsize=7,scheme="equal\_interval",k=5,figsize=(20,8),legend\_bbox\_to\_anchor=(0.1, 0.22))



考虑到统计显著性水平（0.05），仅保留具有统计显著性的结果，可以从地图中得知，Chile为等级交换的热点，意味着Chile更倾向于与邻接国家交换等级而不是其它国家；相反，Russia等国家更倾向于与非邻接国家交换等级，而不是邻接国家。

aqi\_world\_tau\_local\_neighbor\_w.tau\_ln\_pvalues

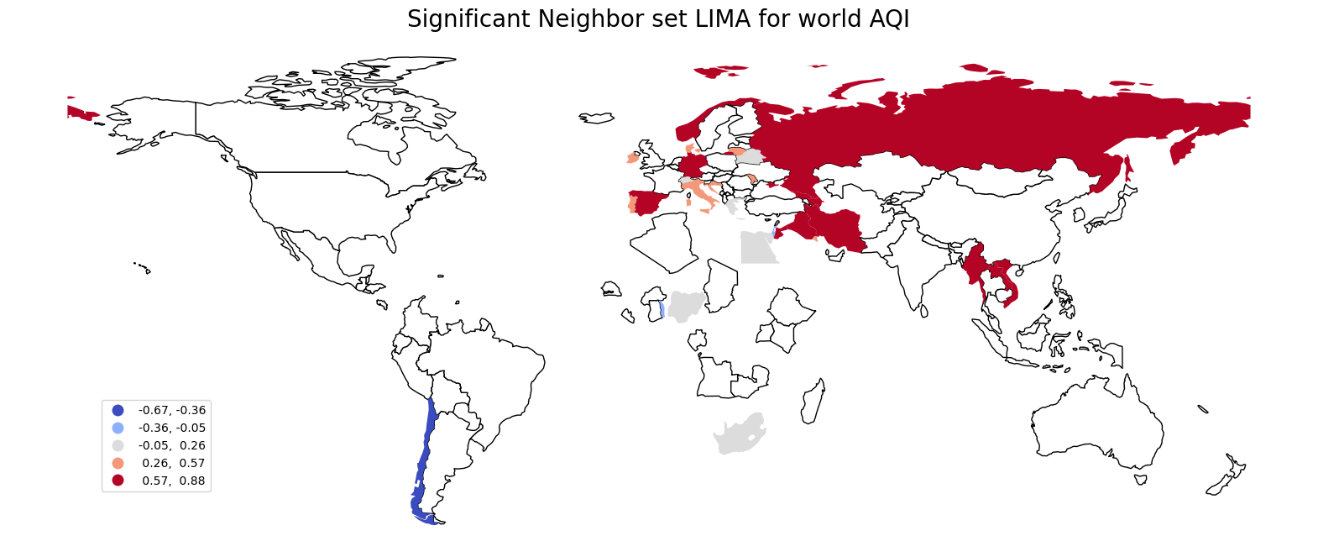
array([0.398, 0.346, 0.157, 0.372, 0.248, 0.594, 0.062, 0.001, 0.51 ,  
 0.001, 0.143, 0.481, 0.076, 0.431, 0.185, 0.084, 0.118, 0.186,  
 0.303, 0.144, 0.023, 0.135, 0.614, 0.417, 0.025, 0.478, 0.045,  
 0.356, 0.01 , 0.137, 0.287, 0.468, 0.213, 0.061, 0.07 , 0.236,  
 0.011, 0.043, 0.543, 0.002, 0.554, 0.634, 0.05 , 0.69 , 0.273,  
 0.17 , 0.007, 0.006, 0.012, 0.009, 0.002, 0.209, 0.048, 0.364,  
 0.553, 0.088, 0.048, 0.264, 0.001, 0.072, 0.548, 0.328, 0.019,  
 0.475, 0.35 , 0.499, 0.112, 0.01 , 0.454, 0.544, 0.011, 0.21 ,  
 0.096, 0.389, 0.226, 0.004, 0.026, 0.331, 0.595, 0.461, 0.236,  
 0.008, 0.926, 0.177, 0.255, 0.002, 0.104, 0.459, 0.149, 0.473,  
 0.005, 0.439, 0.021, 0.065, 0.084, 0.002, 0.245, 0.074, 0.587,  
 0.047, 0.693, 0.522, 0.17 , 0.002, 0.27 , 0.44 , 0.192, 0.02 ,  
 0.155])

saqi\_world\_tau\_local\_neighbor\_w\_significant=aqi\_world\_tau\_local\_neighbor\_w.tau\_ln \* (aqi\_world\_tau\_local\_neighbor\_w.tau\_ln\_pvalues<0.05)  
saqi\_world\_tau\_local\_neighbor\_w\_significant.round(3)

array([ 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0.765,  
 0. , 0.2 , 0. , 0. , 0. , 0. , 0. , 0. ,  
 -0. , 0. , 0. , 0. , -0.667, 0. , 0. , 0. ,  
 0.371, 0. , 0.429, 0. , 0.176, 0. , 0. , 0. ,  
 0. , 0. , 0. , 0. , 0.735, 0.6 , 0. , 0.257,  
 0. , 0. , 0. , 0. , 0. , 0. , 0.765, 0.765,  
 0.514, -0.147, 0.486, 0. , 0.882, 0. , 0. , 0. ,  
 0.529, 0. , 0.824, 0. , 0. , 0. , 0.486, 0. ,  
 0. , 0. , 0. , 0.371, 0. , 0. , 0.794, 0. ,  
 0. , 0. , 0. , 0.059, 0.714, 0. , 0. , 0. ,  
 0. , 0.429, 0. , 0. , 0. , 0.829, 0. , 0. ,  
 0. , 0. , 0.176, 0. , 0.829, 0. , 0. , 0.229,  
 0. , 0. , 0. , -0.294, 0. , 0. , 0. , 0. ,  
 0. , 0. , 0. , 0.647, -0. ])

aqi\_world\_gdf\_pivot\_gdf['sig\_wr']=saqi\_world\_tau\_local\_neighbor\_w\_significant

import matplotlib.pyplot as plt  
  
fig, ax=plt.subplots(nrows=1, ncols=1,figsize = (20,8))  
aqi\_world\_gdf\_pivot\_gdf[aqi\_world\_gdf\_pivot\_gdf["sig\_wr"]==0].plot(ax=ax, color='white',edgecolor='black')  
sig\_ln\_map=aqi\_world\_gdf\_pivot\_gdf[aqi\_world\_gdf\_pivot\_gdf["sig\_wr"] != 0].plot(ax=ax,column="sig\_wr",cmap='coolwarm',scheme='equal\_interval',legend=True)  
leg=ax.get\_legend()  
leg.set\_bbox\_to\_anchor((0, 0.1, 0.16, 0.2))  
sig\_ln\_map.set\_title("Significant Neighbor set LIMA for world AQI",fontdict={"fontsize":20})  
ax.set\_axis\_off()



**Neighborhood set LIMA（邻里）**

Neighborhood变体是对Neighbor变体的拓展，不只是分析目标空间单元和其邻接空间单元之间的协和关系，而是拓展到目标空间单元及其邻里空间单元组成的子集中任何两个空间单元之间的协和关系。

aqi\_world\_tau\_local\_neighborhood\_w=giddy.rank.Tau\_Local\_Neighborhood(aqi\_world\_gdf\_pivot[dt1],aqi\_world\_gdf\_pivot[dt2],w,999)

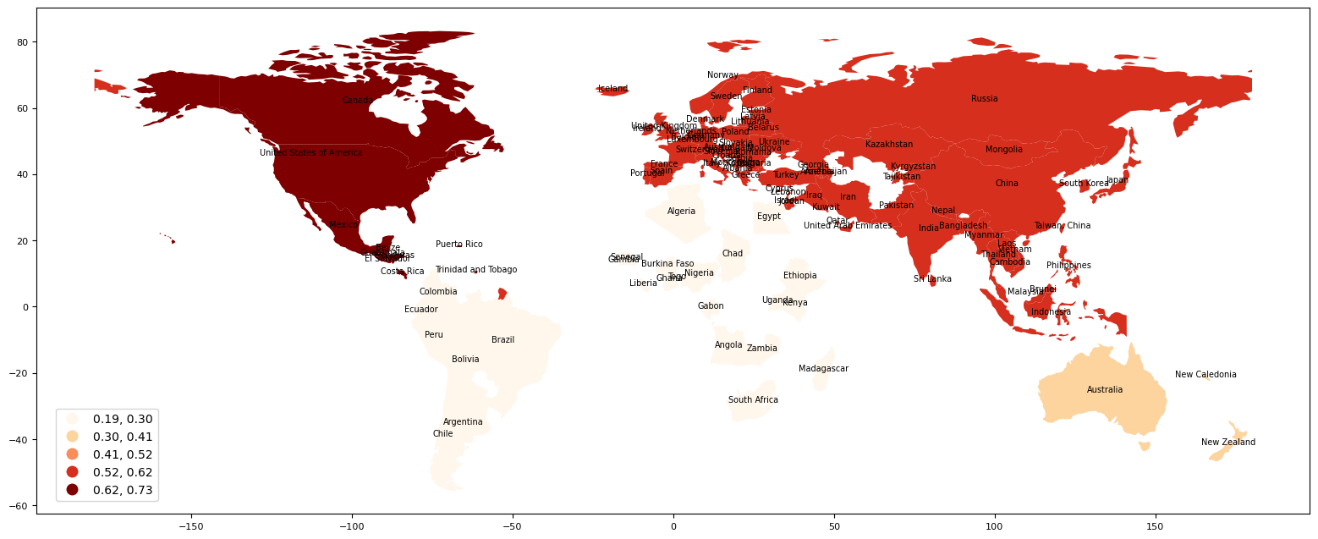
aqi\_world\_tau\_local\_neighborhood\_w.tau\_lnhood

array([0.53809524, 0.25490196, 0.25490196, 0.19047619, 0.56806723,  
 0.33333333, 0.53809524, 0.56806723, 0.56806723, 0.53809524,  
 0.53809524, 0.73333333, 0.19047619, 0.19047619, 0.56806723,  
 0.53809524, 0.25490196, 0.56806723, 0.73333333, 0.25490196,  
 0.19047619, 0.56806723, 0.19047619, 0.73333333, 0.53809524,  
 0.56806723, 0.53809524, 0.19047619, 0.25490196, 0.73333333,  
 0.53809524, 0.25490196, 0.53809524, 0.53809524, 0.25490196,  
 0.25490196, 0.56806723, 0.53809524, 0.25490196, 0.53809524,  
 0.73333333, 0.73333333, 0.53809524, 0.53809524, 0.56806723,  
 0.56806723, 0.56806723, 0.56806723, 0.53809524, 0.56806723,  
 0.53809524, 0.56806723, 0.56806723, 0.56806723, 0.25490196,  
 0.53809524, 0.56806723, 0.56806723, 0.56806723, 0.53809524,  
 0.56806723, 0.25490196, 0.53809524, 0.53809524, 0.25490196,  
 0.56806723, 0.73333333, 0.53809524, 0.56806723, 0.53809524,  
 0.56806723, 0.56806723, 0.53809524, 0.33333333, 0.33333333,  
 0.25490196, 0.53809524, 0.56806723, 0.19047619, 0.56806723,  
 0.53809524, 0.53809524, 0.73333333, 0.56806723, 0.53809524,  
 0.53809524, 0.25490196, 0.53809524, 0.53809524, 0.53809524,  
 0.25490196, 0.56806723, 0.53809524, 0.56806723, 0.53809524,  
 0.53809524, 0.56806723, 0.56806723, 0.56806723, 0.25490196,  
 0.73333333, 0.56806723, 0.25490196, 0.53809524, 0.56806723,  
 0.53809524, 0.73333333, 0.56806723, 0.25490196])

aqi\_world\_gdf\_pivot\_gdf['tau\_lnhood']=aqi\_world\_tau\_local\_neighborhood\_w.tau\_lnhood

从地图打印LIMA（邻里）结果来看，非洲和南美洲的国家邻里之间具有相对较高的等级交换；而北美具有较高稳定性，邻里之间的等级交换较少。

util\_misc.gdf\_plot\_annotate(aqi\_world\_gdf\_pivot\_gdf,"tau\_lnhood","Country",annotate\_fontsize=7,scheme="equal\_interval",k=5,figsize=(20,8),legend\_bbox\_to\_anchor=(0.1, 0.22))



aqi\_world\_tau\_local\_neighborhood\_w.tau\_lnhood\_pvalues

array([0.49 , 0.027, 0.016, 0.135, 0.299, 0.483, 0.418, 0.217, 0.357,  
 0.469, 0.438, 0.138, 0.334, 0.13 , 0.387, 0.505, 0.035, 0.357,  
 0.158, 0.023, 0.138, 0.375, 0.124, 0.125, 0.475, 0.292, 0.48 ,  
 0.135, 0.016, 0.109, 0.494, 0.016, 0.493, 0.457, 0.029, 0.008,  
 0.294, 0.431, 0.038, 0.444, 0.185, 0.134, 0.459, 0.464, 0.359,  
 0.324, 0.377, 0.396, 0.496, 0.212, 0.332, 0.282, 0.292, 0.224,  
 0.04 , 0.442, 0.334, 0.185, 0.227, 0.495, 0.276, 0.016, 0.495,  
 0.469, 0.04 , 0.31 , 0.155, 0.476, 0.166, 0.5 , 0.291, 0.323,  
 0.475, 0.569, 0.432, 0.016, 0.434, 0.315, 0.073, 0.33 , 0.398,  
 0.493, 0.267, 0.371, 0.451, 0.405, 0.019, 0.433, 0.439, 0.463,  
 0.022, 0.312, 0.449, 0.262, 0.482, 0.481, 0.286, 0.27 , 0.328,  
 0.043, 0.18 , 0.296, 0.024, 0.44 , 0.331, 0.452, 0.168, 0.262,  
 0.047])

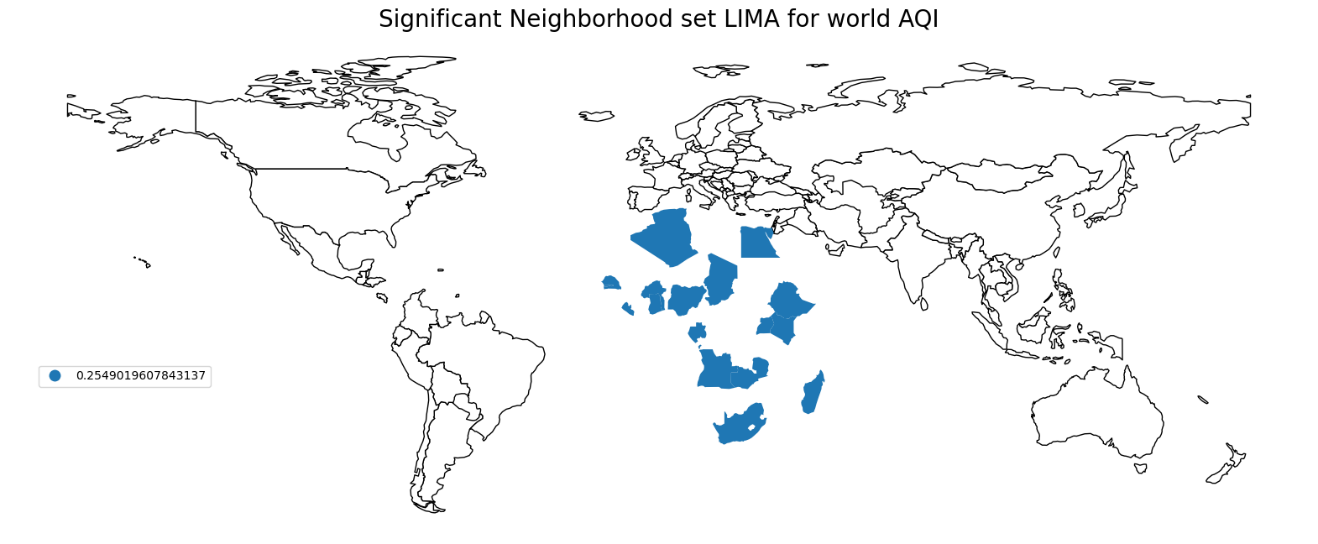
aqi\_world\_tau\_local\_neighborhood\_w\_significant=aqi\_world\_tau\_local\_neighborhood\_w.tau\_lnhood \* (aqi\_world\_tau\_local\_neighborhood\_w.tau\_lnhood\_pvalues<0.05)  
aqi\_world\_tau\_local\_neighborhood\_w\_significant.round(3)

array([0. , 0.255, 0.255, 0. , 0. , 0. , 0. , 0. , 0. ,  
 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0.255, 0. ,  
 0. , 0.255, 0. , 0. , 0. , 0. , 0. , 0. , 0. ,  
 0. , 0.255, 0. , 0. , 0.255, 0. , 0. , 0.255, 0.255,  
 0. , 0. , 0.255, 0. , 0. , 0. , 0. , 0. , 0. ,  
 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ,  
 0.255, 0. , 0. , 0. , 0. , 0. , 0. , 0.255, 0. ,  
 0. , 0.255, 0. , 0. , 0. , 0. , 0. , 0. , 0. ,  
 0. , 0. , 0. , 0.255, 0. , 0. , 0. , 0. , 0. ,  
 0. , 0. , 0. , 0. , 0. , 0.255, 0. , 0. , 0. ,  
 0.255, 0. , 0. , 0. , 0. , 0. , 0. , 0. , 0. ,  
 0.255, 0. , 0. , 0.255, 0. , 0. , 0. , 0. , 0. ,  
 0.255])

aqi\_world\_gdf\_pivot\_gdf['sig\_lnhood']=aqi\_world\_tau\_local\_neighborhood\_w\_significant

考虑统计显著性，非洲具有相对一定邻里空间的等级交换。

fig, ax=plt.subplots(nrows=1, ncols=1,figsize=(20,8))  
aqi\_world\_gdf\_pivot\_gdf[aqi\_world\_gdf\_pivot\_gdf["sig\_lnhood"] == 0].plot(ax=ax, color='white',edgecolor='black')  
sig\_ln\_map=aqi\_world\_gdf\_pivot\_gdf[aqi\_world\_gdf\_pivot\_gdf["sig\_lnhood"] != 0].plot(ax=ax,column="sig\_lnhood",categorical=True,legend=True)  
leg=ax.get\_legend()  
leg.set\_bbox\_to\_anchor((0, 0.15, 0.16, 0.2))  
sig\_ln\_map.set\_title("Significant Neighborhood set LIMA for world AQI",fontdict={"fontsize":20})  
ax.set\_axis\_off()



注释（Notes）：

① Clean Air Act，（美国）清洁空气法（<https://en.wikipedia.org/wiki/Clean_Air_Act_(United_States)>）。

② Kaggle，数据科学家和机器学习从业者的在线社区（<https://www.kaggle.com/>）。

③ Kaggle-Wikipedia，（<https://en.wikipedia.org/wiki/Kaggle>）。

④ Air Quality Index Dataset，（<https://www.kaggle.com/datasets/pratikshapandapkp/air-quality-index-dataset>）。

⑤ Plotly，是一个通过数据应用实现数据驱动决策的实践者，构建有支持多种编程语言的图表库（<https://plotly.com/python/#controls>）。

⑥ Kendall’s Tau (Kendall Rank Correlation Coefficient——Statistics How To网站提供了初级统计学的知识内容（<https://www.statisticshowto.com/kendalls-tau/>）。

⑦ Rank based Methods，（<http://pysal.org/notebooks/explore/giddy/Rank_based_Methods.html>）。

⑧ libpysal.weights.block\_weights空间权重，（<https://pysal.org/libpysal/generated/libpysal.weights.block_weights.html>）。

参考文献（References）:

[1] Air quality index, Wikipedia, <https://en.wikipedia.org/wiki/Air_quality_index>.

[2] “International Air Quality”. Archived from the original on 12 June 2018. Retrieved 20 August 2015.

[3] “NOAA’s National Weather Service/Environmental Protection Agency - United States Air Quality Forecast Guidance”. airquality.weather.gov. Retrieved 2021-04-28.

[4] “MACC Project - European Air Quality Monitoring and Forecasting”. Archived from the original on 2014-10-18. Retrieved 2014-10-12.

[5] Technical Assistance Document for the Reporting of Daily Air Quality – the Air Quality Index (AQI) (Report). U.S. Environmental Protection Agency. May 2016. EPA-454/B-16-002. Archived from the original on 2020-10-26. Retrieved 2020-08-20.

[6] David Mintz (September 2018). Technical Assistance Document for the Reporting of Daily Air Quality – the Air Quality Index (AQI). North Carolina: US EPA Office of Air Quality Planning and Standards. EPA-454/B-18-007. Retrieved 15 September 2021.

[7] Revised Air Quality Standards For Particle Pollution And Updates To The Air Quality Index (AQI). North Carolina: US EPA Office of Air Quality Planning and Standards. 2013.

[8] “ENVIRONMENTAL PROTECTION AGENCY 40 CFR Parts 50, 51, 52, 53 and 58 [EPA-HQ-OAR-2008-0699; FRL-9913-18-OAR] RIN 2060-AP38 – National Ambient Air Quality Standards for Ozone”. Environmental Protection Agency. Archived from the original on 2015-10-06. Retrieved 2015-10-05.

[9] Rey, Sergio J. 2014. “Fast Algorithms for a Space-Time Concordance Measure.” Computational Statistics 29 (3-4). Springer: 799–811.

[10] Rey, Sergio J. 2016. “Space–Time Patterns of Rank Concordance: Local Indicators of Mobility Association with Application to Spatial Income Inequality Dynamics.” Annals of the Association of American Geographers. Association of American Geographers 106 (4): 788–803.