# 3.5-C 生态系统服务价值的贝叶斯网络构建与推理

## 3.5.1 生态系统服务价值（估值）与 InVEST 和 NWV 计算工具

### 3.5.1.1 生态系统服务

**生态系统（ecosystem，ecological system）**：由所有生物体（organisms）和它们相互作用的物理环境（physical environment，指自然环境中的非生物因素，如气候、地形、土壤等，对生物体和生态系统产生影响的各种自然条件）组成[1]458。这些生物和非生物成分通过营养循环（nutrient cycles）和能量流（energy flows）联系在一起。能量通过光合（photosynthesis）作用进入系统并融入植物组织。动物通过以植物或者彼此为食，对生态系统的物质和能量流动起到重要作用，也影响到植物和微生物生物量的数量。通过分解死亡的有机物质，分解者将碳释放回大气，并通过将储存在死亡生物质中的营养物质转化为易于被植物和微生物利用的形式，促进营养循环[2]。

**生态系统服务（eco-services，ecosystem services ）**：是自然环境和健康的生态系统为人类提供的多种多样的福利（benefits）。这些生态系统包括诸如农业生态系统（agroecosystems）、森林生态系统（forest ecosystem）、草地生态系统（grassland ecosystems）和水生生态系统（aquatic ecosystems）等，其提供有农作物自然授粉、空气清洁、极端气候调节和有利于人类身心健康等好处。总的来说，这些好处被称为生态系统服务，通常与食物和清洁饮用水供给（provision）、废物分解和粮食生态系统（food ecosystems）的复原力和生产力密不可分[3]。

**生物多样性（biodiversity）**：指一个地区可以找到的不同种类的生命，这些生命例如各种各样的动物、植物、真菌，甚至微生物（如细菌）等，并构成了一个自然世界。这些物种和有机体在生态系统中一起工作，形成一个错综复杂的网络，以保持平衡和维持生命。生物多样性的变化影响生态系统服务的供给，支持着人类所需要的一切，例如事物、清洁的水资源、药品和住所等[4]。

* 生态系统服务的类型

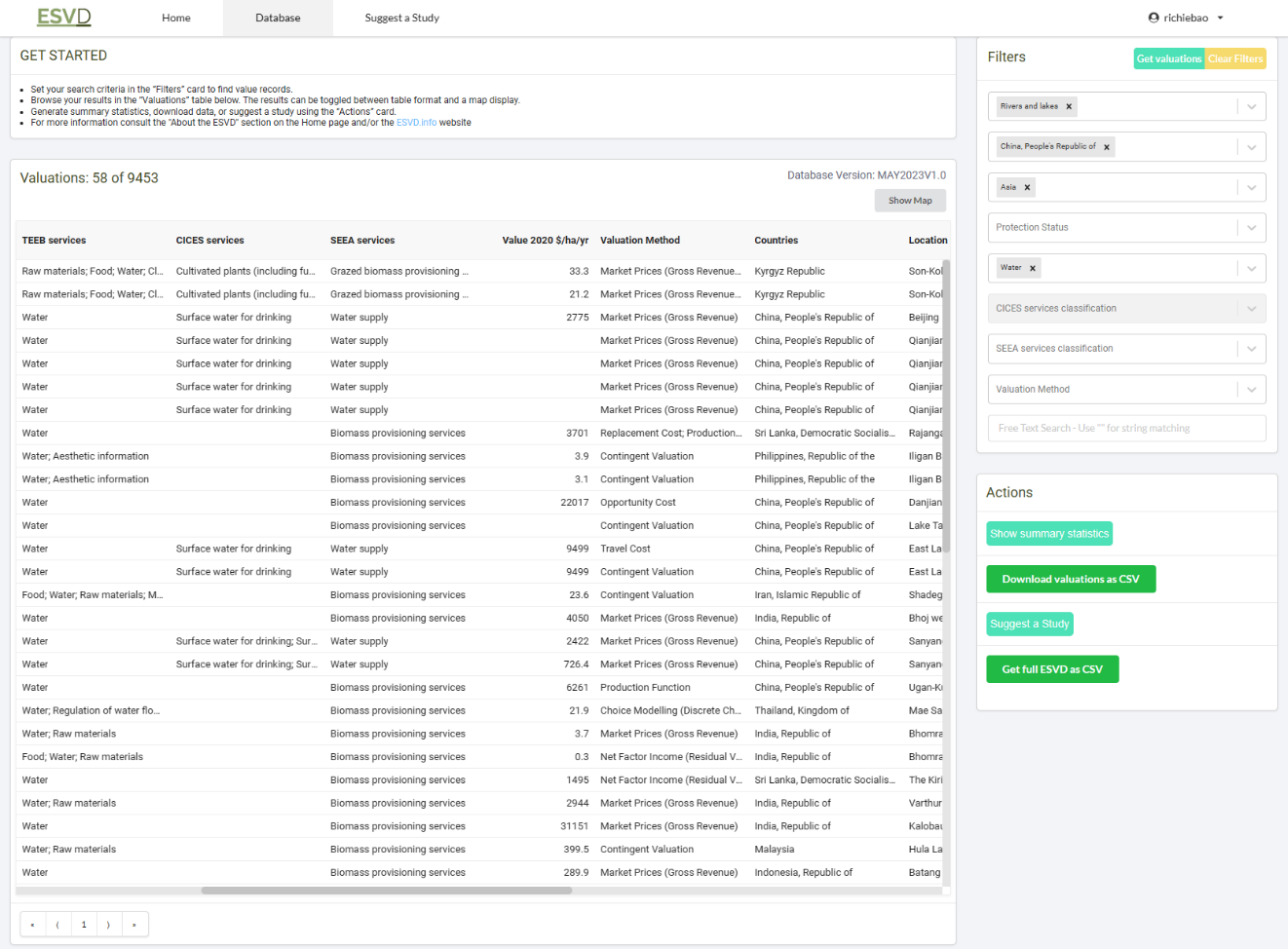
[2005年千年生态系统评估（Millennium Ecosystem Assessment，MA）报告](https://www.millenniumassessment.org/en/Index-2.html)①将生态系统服务定义为人类从生态系统中获得的利益，并将生态系统服务分为四类，调节服务（Regulating services）、供给服务（Provisioning services）、文化服务（Cultural services）和支持服务（Supporting services），其中支持服务是其它三类服务的基础。一个生态系统不一定同时提供上述4类服务，且不同的生态系统（例如森林、海洋、珊瑚礁或红树林等）提供的服务在性质和结果上各不相同。同时，一些服务直接影响邻近人类的生计，例如淡水、食物或审美价值等；而有些服务会引发气候变化、水土流失或自然灾害，间接的影响人类社会[5]。生态系统服务类型的内容如下：

| 生态系统服务的类型 | 支持服务（Supporting services | 供给服务（Provisioning services | 调节服务（Regulating services） | 文化服务（Cultural services） |
| --- | --- | --- | --- | --- |
| 内容 | 该功能是生态系统提供其它服务功能的基础，包括但不限于养分循环（nutrient cycling）、初级生产（primary production）、土壤形成（soil formation）、栖息地提供（habitat provision）等服务。 | 食物（包括海鲜和野味）、农作物、野生食物和香料（food (including seafood and game), crops, wild foods, and spices）原材料（包括木材、毛皮、薪柴、有机物、饲料和肥料）（raw materials (including lumber, skins, fuelwood, organic matter, fodder, and fertilizer)）遗传资源（包括作物改良基因和医疗保健）（genetic resources (including crop improvement genes, and health care)）生物矿物（biogenic minerals）医药资源（包括药品、化学模型以及测试和分析生物体）（medicinal resources (including pharmaceuticals, chemical models, and test and assay organisms)）能源（水电、生物质燃料）（energy (hydropower, biomass fuels)）观赏资源（包括时装、手工艺品、珠宝、宠物、崇拜、装饰和纪念品，如毛皮、羽毛、象牙、兰花、蝴蝶、观赏鱼、贝壳等）（ornamental resources (including fashion, handicrafts, jewelry, pets, worship, decoration, and souvenirs like furs, feathers, ivory, orchids, butterflies, aquarium fish, shells, etc.)） | 水和空气的净化（Purification of water and air） 碳封存和气候调节（Carbon sequestration and climate regulation）废物分解和解毒（Waste decomposition and detoxification）捕食对猎物数量的调节（Predation regulates prey populations）对病虫害的生物防治（Biological control pest and disease control）授粉（Pollination） 扰动调节（防洪）（Disturbance regulation, i.e. flood protection） | 文化（包括在书籍、电影、绘画、民俗、国家象征、广告等中使用自然作为主题）（cultural (including use of nature as motif in books, film, painting, folklore, national symbols, advertising, etc.)）精神和历史（包括将自然用于宗教或遗产价值或自然）（spiritual and historical (including use of nature for religious or heritage value or natural)）娱乐体验（包括生态旅游、户外运动和娱乐）（recreational experiences (including ecotourism, outdoor sports, and recreation)）科学和教育（包括利用自然系统进行学校旅行和科学发现）（science and education (including use of natural systems for school excursions, and scientific discovery)）治疗（包括生态疗法、社会林业和动物辅助疗法）（therapeutic (including eco-therapy, social forestry and animal assisted therapy)） |

MA 对生态系统、生态系统服务功能和人类福利（福祉）给出了如下定义：生态系统是植物、 动物和微生物群落及其无机环境的动态复合体和相互作用功能单元，树洞里一个临时的 小水洼，以及辽阔的海洋盆地，都可以被称之为一个生态系统；生态系统服务功能是指 人类从生态系统中获得的效益，这些效益包括供给功能（如粮食与水的供给）、调节功 能（如调节洪涝、干旱、土地退化，以及疾病等）、支持功能（如土壤形成与养分循环 等）和文化功能（如娱乐、精神、宗教以及其它非物质方面的效益）；人类福利具有多 重成分，包括维持高质量的生活所需的基本物质条件、自由权与选择权、健康、良好的 社会关系，以及安全等[6]。

**生态系统服务价值（估值）（Ecosystem Valuation）**：是为生态系统服务分配价值（货币化（monetary）、生物物理化（biophysical,）等）的一个经济过程，例如量化森林在减少洪水和侵蚀、固碳、为濒危物种提供栖息地和吸收有害化学物质等人类的福祉。生态系统服务货币化的途径一定程度上可以为政策制定者和自然保护主义者提供一种评估管理影响和比较潜在政策成本效益的分析工具[7]。生态系统服务包括水、木材、鱼类等可以以市场价格出售的直接使用价值；也包括通过减缓气候变化使整个人类社会受益的间接使用价值，及其它类型的价值等。在价值计算上，存在享乐估值法（Hedonic valuation methods）、旅行费用计算方法（Travel cost methods）、陈述偏好法（stated preference method）和福祉转移法（ benefit transfer methods）等[8]。

生态系统服务价值（估值）的文献不断涌现，集成相关研究成果可以构建生态系统服务价值数据库，例如由 [Luke Brander](http://lukebrander.com/) ②和[可持续发展基金会（Foundation for Sustainable Development，FSD）](https://fsd.nl/who-we-are/)③、[英国苏格兰农业学院（Scotlands Rural College，SRUC）](https://www.sruc.ac.uk/)④联合发布的 [ESVD（Ecosystem Services Valuation Database）](https://www.esvd.net/)⑤（以每年每公顷美元标准化）。ESVD 目前包含来自分布在所有生物群落、生态服务系统和地理区域 1,100 多项研究的 9,453 条价值记录，其用于估值的研究资料库超过 2,000 多项研究，且数量仍在持续增加，估值的记录数量也将继续增加。ESVD 的长期目标是提供有关生态系统和生物多样性经济效益及其损失成本可靠且易于获取的信息，以支持有关自然保护、生态系统修复的决策和可持续的土地管理。例如下图的 ESVD 搜索结果[9]：



### 3.5.1.2 生态系统服务计算工具

* InVEST

目前较为系统的生态系统服务计算工具以[InVEST（Integrated Valuation of Ecosystem Services and Tradeoffs）](https://naturalcapitalproject.stanford.edu/software/invest)⑥为主。InVEST是一套免费的开源软件模型（基于 Python 语言），用于绘制和评估自然界中维持和充实人类生活的商品和服务，为自然资源管理决策提供信息。该工具的模型模块化开发，已开发的模块有

1. 年产水量（Annual Water Yield）、
2. 碳储存和封存（Carbon Storage and Sequestration）、
3. 海岸带蓝碳（Coastal Blue Carbon）、
4. 海岸脆弱性模型（Coastal Vulnerability Model）、
5. 作物生产（Crop Production）、
6. 森林碳边缘效应（Forest Carbon Edge Effect）、
7. 生境质量（Habitat Quality）、
8. 生境风险评估（Habitat Risk Assessment）、
9. 养分输送比（NDR: Nutrient Delivery Ratio）、
10. 海上风能生产（Offshore Wind Energy Production）、
11. 作物授粉（授粉者丰度）（Crop Pollination (Pollinator Abundance)）、
12. 泥沙输移比例模型（SDR: Sediment Delivery Ratio）、
13. 季节性产水量（Seasonal Water Yield）、
14. 景区质量（Scenic Quality）、
15. 城市降温模型（Urban Cooling Model）、
16. 城市洪水风险缓解模型（Urban Flood Risk Mitigation model）、
17. 城市自然空间可达性模型（Urban Nature Access）、
18. 城市暴雨滞留模型（Urban Stormwater Retention Model）、
19. 参观: 休闲和旅游业（Visitation: Recreation and Tourism）、
20. 波浪能生产（Wave Energy Production）等 20 个模型，

以及辅助工具情景生成器（Scenario Generator: Proximity Based）等。

因为 InVEST 基于 Python 编程语言开发，其提供了一个 Python [应用程序编程接口（application programming interface，API ）](https://invest.readthedocs.io/en/latest/)⑦，方便在 Python 编程环境下安装调用。同时，[NatCap（natural capital） 研究团队](https://naturalcapitalproject.stanford.edu/about/people)⑧开发了一个地理数据处理的 Python 包[PyGeoprocessing](https://pygeoprocessing.readthedocs.io/en/latest/)⑨，提供了一组常用栅格、矢量数据和水文相关的处理工具。

* NWV（NVE）

因为生态系统服务（ecosystem services，ESS）没有被商业市场充分捕捉，也没有被充分量化，无法与市场上交易的商业服务相比较，造成生态系统服务在政策决定中往往得不到充分重视。忽视生态系统服务的价值可能会导致生态系统过度开发或不合理的政策和投资决策。佛兰德（Flemish）政府认识到生态系统服务的重要性，其环境部请[VITO](https://vito.be/en)⑩（清洁技术和可持续发展领域的独立研究机构）、安特卫普大学和阿姆斯特丹大学（universities of Antwerp and Amsterdam）的经济学家和生态学家证实生态系统服务的重要性。构建的[Nature Value Explorer（NVE）（Natuur waarde verkenner（荷兰语），NWV）](https://www.natuurwaardeverkenner.be/" \l "contact)⑪工具侧重于评估生态系统服务的实用方法，以便帮助绘制生态系统服务的社会经济价值。NWV 为在线网页工具的形式，仅服务于佛兰德地区，计算模型和代码并未开源，因此对世界上其它区域 NWV 仅适用于对生态系统服务计算工具这一构建方式的参考。

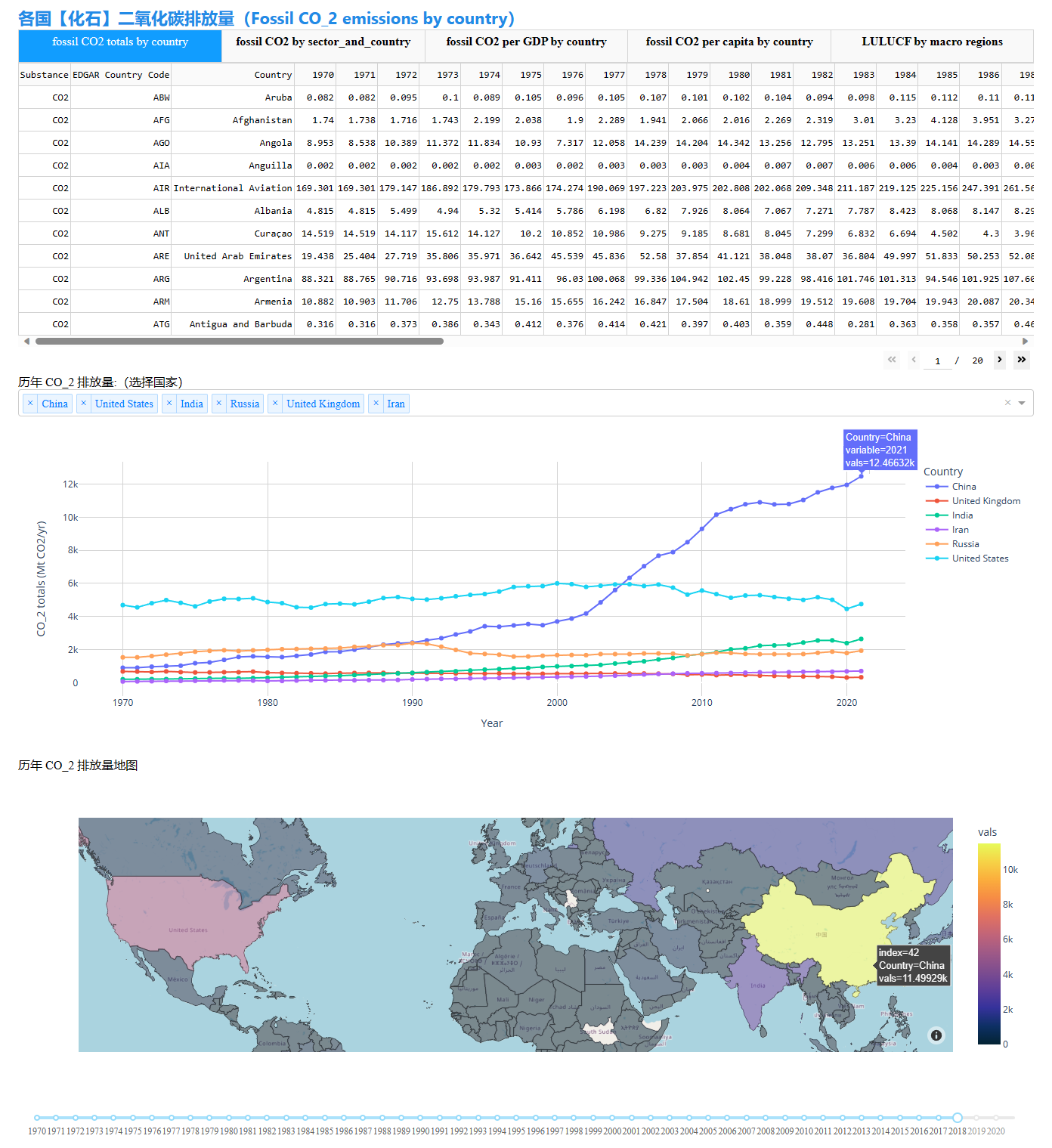
## 3.5.2 碳储存和封存/固碳（Carbon Storage and Sequestration）

全球气候变化导致极端气候事件频发，影响日渐深重，成为21世纪人类共同面临的最重大环境挑战[10]。从21世纪初至2019年，全球温室气体（greenhouse gas，GHG）排放呈上升趋势，其中我国等新兴经济体的二氧化碳排放量增加为主要原因。大气中温室气体浓度大幅度增加，增强了自然温室效应，这可能对地球上的生命产生负面影响[11]。《联合国气候变化框架公约》( United Nations Framework Convention on Climate Change，UNFCCC)建立了一项国际环境条约，以应对“人类对气候系统的危险干扰”，部分手段是稳定大气中的温室气体浓度。尽管如此，作为全球变暖的二氧化碳排放量仍在世界范围内增加。在此背景下，[全球大气研究排放数据库（Emissions Database for Global Atmospheric Research，EDGAR）](https://edgar.jrc.ec.europa.eu/)⑫就[政府间气候变化专门委员会（Intergovernmental Panel on Climate Change，IPCC）](https://www.ipcc.ch/)⑬的最新指南和最近的活跃数据，为世界各国提供了独立的温室气体估算，并在2022年9月发布了最新更新，可以获得1970-2021年期间化石燃料的排放数据，可以从[EDGAR](https://edgar.jrc.ec.europa.eu/report_2022)⑭下载[12]。

为了交互可视化二氧化碳排放数据，使用Python的[Dash（plotly）](https://dash.plotly.com/installation)⑮库，可交互的图表工具可视化分析数据，完成的代码可以从托管于GitHub的[USDA\_dashboard](https://github.com/richieBao/USDA_dashboard)⑯代码库下载，文件名为’dash\_fossil\_CO2\_emissions\_2022.py’。

# Import packages  
from dash import Dash, html, dash\_table, dcc, callback, Output, Input  
import pandas as pd  
import geopandas as gpd  
import plotly.express as px  
import plotly.graph\_objects as go  
import dash\_mantine\_components as dmc  
import numbers  
import json  
import numpy as np  
  
ghg\_fn='./data/EDGARv7.0\_FT2021\_fossil\_CO2\_booklet\_2022.xlsx'  
ghg\_df=pd.ExcelFile(ghg\_fn)  
print(ghg\_df.sheet\_names)  
# ['info', 'fossil\_CO2\_totals\_by\_country', 'fossil\_CO2\_by\_sector\_and\_countr', 'fossil\_CO2\_per\_GDP\_by\_country', 'fossil\_CO2\_per\_capita\_by\_countr', 'LULUCF by macro regions']  
fossil\_CO2\_totals\_by\_country=pd.read\_excel(ghg\_fn,sheet\_name='fossil\_CO2\_totals\_by\_country').round(3)  
fossil\_CO2\_by\_sector\_and\_country=pd.read\_excel(ghg\_fn,sheet\_name='fossil\_CO2\_by\_sector\_and\_countr').round(3)  
fossil\_CO2\_per\_GDP\_by\_country=pd.read\_excel(ghg\_fn,sheet\_name='fossil\_CO2\_per\_GDP\_by\_country').round(3)  
fossil\_CO2\_per\_capita\_by\_country=pd.read\_excel(ghg\_fn,sheet\_name='fossil\_CO2\_per\_capita\_by\_countr').round(3)  
LULUCF\_by\_macro\_regions=pd.read\_excel(ghg\_fn,sheet\_name='LULUCF by macro regions').round(3)  
keys\_year=[i for i in fossil\_CO2\_totals\_by\_country.columns if isinstance(i,numbers.Number)]  
  
countries\_fn=r'./data/countries.geojson'   
countries\_gdf=gpd.read\_file(countries\_fn)  
  
# Initialize the app - incorporate a Dash Mantine theme  
external\_stylesheets = [dmc.theme.DEFAULT\_COLORS]  
app = Dash(\_\_name\_\_, external\_stylesheets=external\_stylesheets)  
  
tabs\_styles = {  
 'height': '44px'  
}  
  
tab\_style = {  
 'borderBottom': '1px solid #d6d6d6',  
 'padding': '6px',  
 'fontWeight': 'bold'  
}  
  
tab\_selected\_style = {  
 'borderTop': '1px solid #d6d6d6',  
 'borderBottom': '1px solid #d6d6d6',  
 'backgroundColor': '#119DFF',  
 'color': 'white',  
 'padding': '6px'  
}  
  
# App layout  
app.layout = dmc.Container([  
 dmc.Title(r'各国【化石】二氧化碳排放量（Fossil CO\_2 emissions by country）', color="blue", size="h3"),  
 dcc.Tabs(id='tabs\_co2',value='tab-1',children=[  
 dcc.Tab(label='fossil CO2 totals by country', value='tab-1', style=tab\_style, selected\_style=tab\_selected\_style),  
 dcc.Tab(label='fossil CO2 by sector\_and\_country', value='tab-2', style=tab\_style, selected\_style=tab\_selected\_style),  
 dcc.Tab(label='fossil CO2 per GDP by country', value='tab-3', style=tab\_style, selected\_style=tab\_selected\_style),  
 dcc.Tab(label='fossil CO2 per capita by country', value='tab-4', style=tab\_style, selected\_style=tab\_selected\_style),  
 dcc.Tab(label='LULUCF by macro regions', value='tab-5', style=tab\_style, selected\_style=tab\_selected\_style),  
 ],style=tabs\_styles),  
 html.Div(id='tabs-content-inline'),  
 html.Label(r'历年 CO\_2 排放量:（选择国家）'),  
 dcc.Dropdown(  
 fossil\_CO2\_totals\_by\_country['Country'].tolist(),['China','United States','India','Russia','United Kingdom','Iran'],multi=True,id='countries4co2totals'   
 ),   
 dcc.Graph(id='co2\_totals'),   
 html.Label(r'历年 CO\_2 排放量地图'),   
 dcc.Graph(id='co2\_worldmap'),  
 dcc.Slider(id='year-slider',min=min(keys\_year),max=max(keys\_year),marks={x: {'label': str(x)} for x in range(min(keys\_year),max(keys\_year))},value=2018),  
 ],  
 fluid=True  
 )  
  
@app.callback(Output('tabs-content-inline', 'children'),  
 Input('tabs\_co2', 'value'))  
def render\_co2\_tables(tab):  
 if tab == 'tab-1':  
 return html.Div([  
 dash\_table.DataTable(data=fossil\_CO2\_totals\_by\_country.to\_dict('records'), page\_size=11, style\_table={'overflowX': 'auto'},columns=[{"name": str(i), "id": str(i)} for i in fossil\_CO2\_totals\_by\_country.columns]),  
 ])  
 elif tab == 'tab-2':  
 return html.Div([  
 dash\_table.DataTable(data=fossil\_CO2\_by\_sector\_and\_country.to\_dict('records'), page\_size=11, style\_table={'overflowX': 'auto'},columns=[{"name": str(i), "id": str(i)} for i in fossil\_CO2\_by\_sector\_and\_country.columns]),  
 ])  
 elif tab == 'tab-3':  
 return html.Div([  
 dash\_table.DataTable(data=fossil\_CO2\_per\_GDP\_by\_country.to\_dict('records'), page\_size=11, style\_table={'overflowX': 'auto'},columns=[{"name": str(i), "id": str(i)} for i in fossil\_CO2\_per\_GDP\_by\_country.columns]),  
 ])  
 elif tab == 'tab-4':  
 return html.Div([  
 dash\_table.DataTable(data=fossil\_CO2\_per\_capita\_by\_country.to\_dict('records'), page\_size=11, style\_table={'overflowX': 'auto'},columns=[{"name": str(i), "id": str(i)} for i in fossil\_CO2\_per\_capita\_by\_country.columns]),  
 ])  
 elif tab == 'tab-5':  
 return html.Div([  
 dash\_table.DataTable(data=LULUCF\_by\_macro\_regions.to\_dict('records'), page\_size=11, style\_table={'overflowX': 'auto'},columns=[{"name": str(i), "id": str(i)} for i in LULUCF\_by\_macro\_regions.columns]),  
 ])  
   
@app.callback(Output('co2\_totals', 'figure'),  
 Input('countries4co2totals', 'value'))   
def update\_co2\_totals\_by\_country(countries):  
 df=fossil\_CO2\_totals\_by\_country[fossil\_CO2\_totals\_by\_country.Country.isin(countries)]   
 df\_ys=df[keys\_year+['Country']]  
 df\_melted=df\_ys.melt('Country', value\_name='vals')  
 fig=px.line(df\_melted,x='variable',y='vals',color='Country',markers=True).update\_layout(plot\_bgcolor='rgba(0, 0, 0, 0)',yaxis\_title=r'CO\_2 totals (Mt CO2/yr)',xaxis\_title="Year")  
 fig.update\_yaxes(gridcolor='lightgrey')  
 fig.update\_xaxes(gridcolor='lightgrey')  
   
 return fig  
   
@app.callback(  
 Output('co2\_worldmap', 'figure'),  
 Input('year-slider', 'value'))  
def update\_co2\_worldmap(selected\_year):  
 df\_ys=fossil\_CO2\_totals\_by\_country[keys\_year+['Country','EDGAR Country Code']]  
 df\_melted=df\_ys.melt(['Country','EDGAR Country Code'], value\_name='vals')  
 df\_melted\_yr=df\_melted[df\_melted['variable']==selected\_year]  
 countries\_copy\_gdf=countries\_gdf.copy(deep=True)  
   
 def merging(row):   
 try:  
 key=row['ISO\_A3']  
 selection=df\_melted\_yr[df\_melted\_yr['EDGAR Country Code']==key]  
 return pd.Series(selection[['Country','vals']].values[0])  
 except:  
 return pd.Series([np.nan,np.nan])   
 countries\_copy\_gdf[['Country','vals']]=countries\_copy\_gdf.apply(merging,axis=1)  
   
 fig = go.Figure(px.choropleth\_mapbox(countries\_copy\_gdf,   
 geojson=countries\_copy\_gdf.geometry,  
 locations=countries\_copy\_gdf.index,  
 color='vals',  
 color\_continuous\_scale='thermal', # px.colors.diverging.BrBG, "Viridis",  
 mapbox\_style="open-street-map", # "carto-positron",  
 zoom=3, center = {"lat": 37.0902, "lon": -95.7129},  
 opacity=0.5,  
 hover\_data=["Country", "vals"],  
 # height=1000,  
 ))   
   
 return fig  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 app.run\_server(debug=True)

执行该代码，结果如下图。



[可查看联合国环境规划署（United Nations Environment Programme）对世界各国温室气体（）数据的可视化描述](https://www.unep.org/explore-topics/climate-action/what-we-do/climate-action-note/state-of-climate.html)⑰。

从数据可视化图表可以得知，我国的二氧化碳排放量在2005年超过美国并一路攀升，高达美国排放量的约2倍，在2021年达到 12466 亿吨（）。在2020年9月的第75届联合国大会上，我国郑重宣誓的排放力争于2030年前达到峰值，2060年前实现碳中和，即“双碳”目标。应对气候变化关键在于“控碳”，先实现碳达峰，再实现碳中和，后者更加重要，也更加困难。除通过调整产业结构、提高能源利用率、推广新能源等措施切实降低等温室气体排放外，同时通过生态系统服务，森林、草地、泥炭沼泽和其它陆生生态系统的碳汇能力，调节地球气候[13]。

1992年，UNFCCC 首次定义了碳汇（carbon sink），为从大气中清除的过程、活动或机制，即各种载体（森林、植被、水体、土壤、湿地等）通过其生态原理，吸收、储存和固定的过程，活动或机制，当其固碳量大于排放量时，称该过程为碳汇[14]。为了增强生态系统功能和生态产品供给能力，提升生态系统碳汇增量，除了各国政府采取切实的行动之外，国内外学者开展了广泛的研究探索。自1992年以来，碳汇相关研究发文量持续增长，研究内容涉及：在相关政策与理论研究方面的森林等生态系统的碳汇效能，低碳城市的发展策略，城乡绿色基础服务设施的碳汇能力建设任务与路径的探索；在碳汇计量与评估方面，相关研究从聚焦单一生态要素碳汇能力评估，发展到面向绿色空间全要素碳汇的综合评估，包括碳源、碳汇、碳储存和封存的计量监测，开发有[InVEST（Integrated Valuation of Ecosystem Services and Tradeoffs）](https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/en/index.html)⑥，[NWV（Nature Value Explorer）](https://www.natuurwaardeverkenner.be/)⑪等面向生态系统服务评估的综合模型（计算工具）；在碳汇影响机制方面，宏观层面倾向于蓝绿空间的数量与空间格局对碳汇效能的影响机制，及土地利用和覆盖（landuse and landcover，LULC）、人口规模、经济发展程度等经济社会因素对碳汇的驱动机制研究。微观层面倾向于单值植物、植物配置、土壤类型、全生命周期评估等；在以增汇为导向的城乡蓝绿空间优化途径方面，相关研究着眼于宏观系统构建增量和微观个体增效提质为蓝绿空间规划和建设提出优化策略[15]。

当前的研究涉猎广泛，从微观到宏观，从定性到定量，包含生态、环境、规划、经济、社会、计算机等众多学科交叉。在未来的研究上，可以进一步拓展大数据在气候变化和碳汇等领域的应用；用机器学习预测不同气候情境下各生态系统的碳汇量和变化；扩大生态系统的类型研究，从森林生态系统到湿地、草地、海洋和岩石风化等；并利用系统论的原理研究系统整体和部分之间的关系，统筹、协调和权衡生态系统和经济社会的矛盾；进一步探索全生命周期下碳汇功能评估[16]。

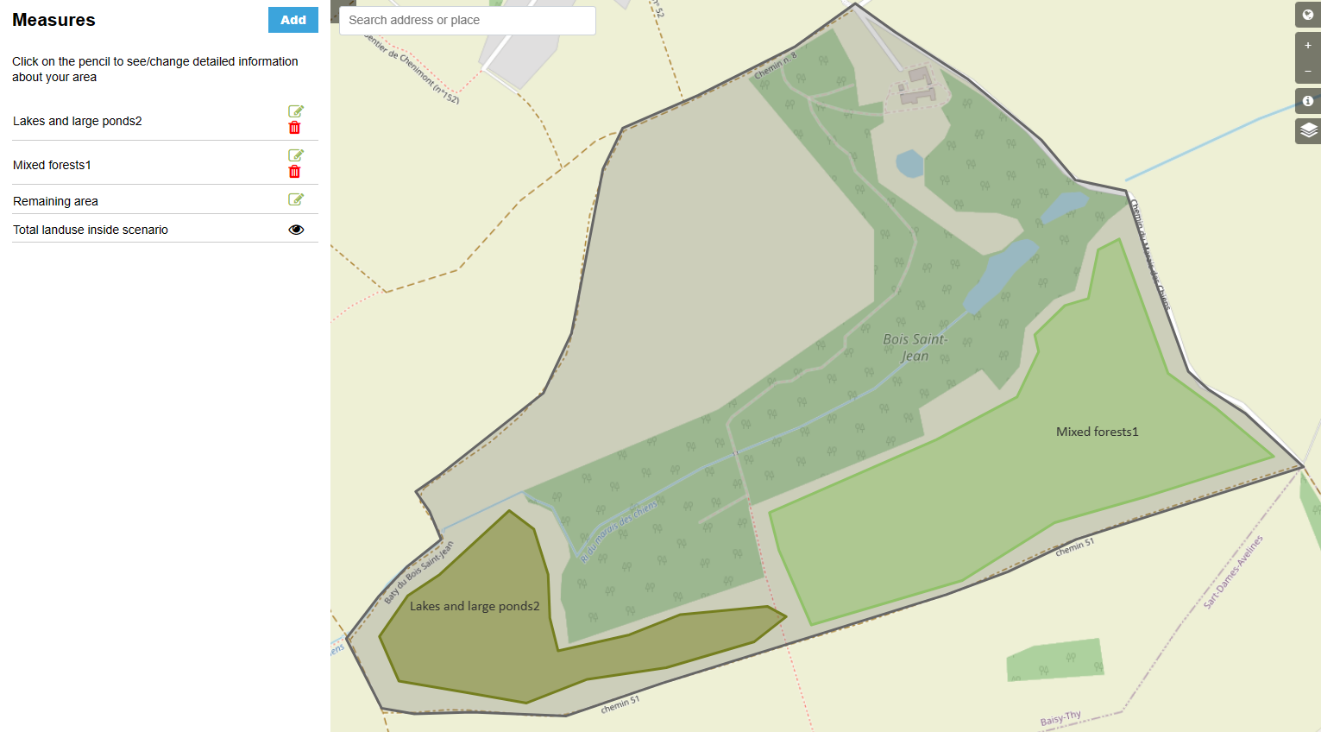
### 3.5.2.1 NWV 的碳相关计算

NWV 在调节服务（regulating services）中的调节物理、化学和生物环境（Regulating the physical, chemical and biological environment）下提供有对全球气候变化影响有关的[土壤中的碳封存](https://natuurwaardeverkenner.be/docs/manual_EN/regulerende_diensten/omgeving/carbonsoil)⑱和[生物质能（biomass）中的碳封存](https://natuurwaardeverkenner.be/docs/manual_EN/regulerende_diensten/omgeving/carbonbiomass)⑲计算。

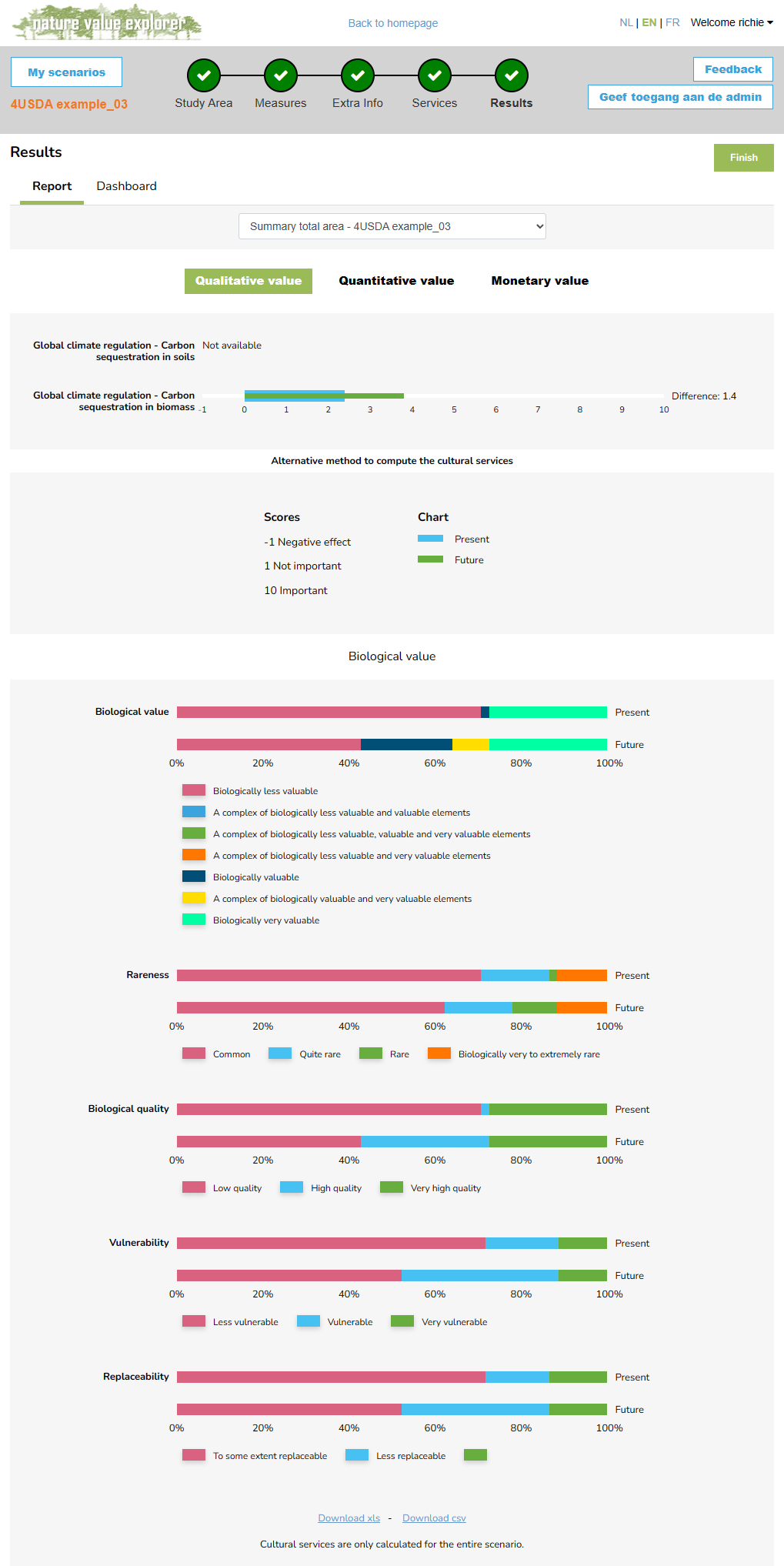
对于土壤的碳封存，NWV 采用多元回归方法评估比利时弗兰德斯（Flanders, Belgium）土壤有机碳（soil organic carbon，SOC）的空间分布及其对土壤特征的依赖性。基于这个回归模型，给定植被类型、土壤排水等级和土壤质地计算土壤潜在最大含碳量[17]。土壤排水和植被的变化都会改变潜在的最大含碳量。而年固碳潜力是潜在含碳量和实际含碳量之差的百分比[18,19]。碳封存的社会价值（货币化（monetary））计算使用缓解气候变化减排措施的成本作为指标。相关研究估计2010年比利时的减排成本为10至60欧元/吨（），且为了达到控制全球气温增高2° CO\_2CO\_2$）的较高值计算[21]。

对于生物质能中的碳封存，NWV 中的计算仅限于森林生态系统的吸收，使用特定物种的碳密度（carbon density）推导出每年每公顷森林的年度固碳量来计算其增量[22]。碳封存的货币化值同土壤的碳封存。

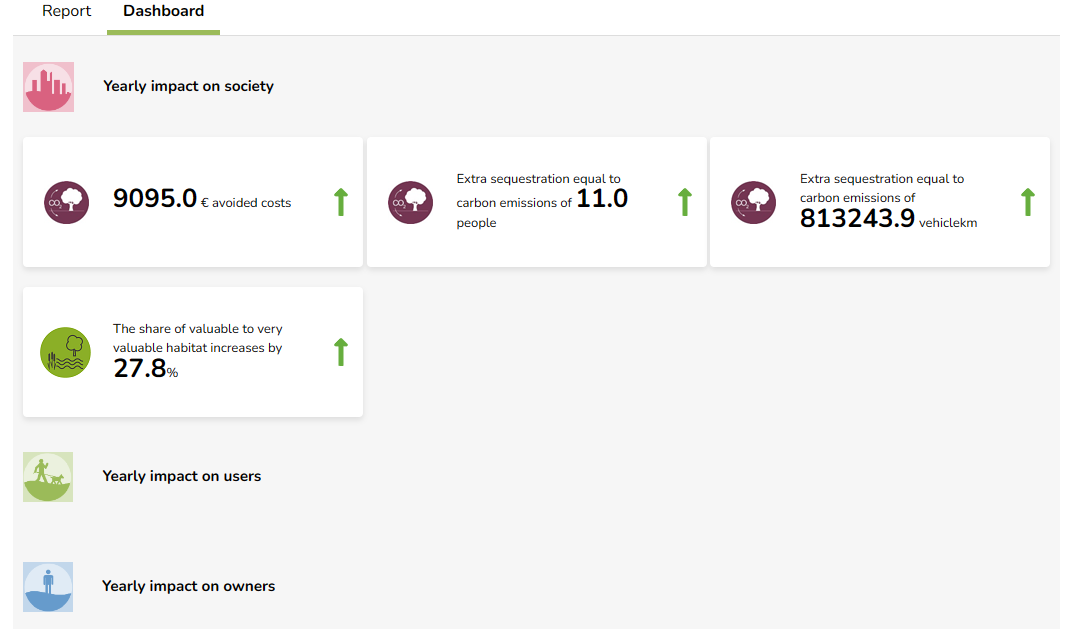
[NWV 生态系统服务在线计算](https://www.natuurwaardeverkenner.be/)⑳的数据为比利时弗兰德斯区域，下述圈画了一个区域进行计算，查看计算流程、输入条件和计算结果等情况，并仅计算土壤和生物质能的碳封存。在Measures阶段，假设新规划了一块混交林，和一个大面积的池塘；在Extra Info阶段，假设圈选区域当前不具有较高的生物多样性和存在受保护的物种，但在将来会采取额外的措施加强生物多样性并保护或恢复一些物种。同时，配置该区域可达，步道占比0.1%。



NWV 生成了可视化计算结果报告，并提供相关数据的下载服务。因为增加了混交林和池塘，生物价值（biological value）、珍稀物种（Rareness）、生物质量（Biological quality）均有所提升。



在年度的社会影响上，由于碳封存量的增加，可避免的减排成本（即碳封存的经济货币价值），避免的碳排放量，及栖息地的价值都在增加。



报告中给出了土壤和生物质能碳封存结果，及对应的货币价值，也可以从下载的数据文件中查看。从结果来看，每年土壤中的碳储量增加了 29.3 吨/年（），生物质能中增加了 9.7 吨/年（；土壤和生物质能碳封存的社会经济价值变化幅度为[3903,14287](欧元/年)。

| Quantitative valuation | Unit | Present |  | Future |  | Difference |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Low | High | Low | High | Low | High |
| Global climate regulation - Carbon sequestration in soils | tonnes C / year | 260.9 | 260.9 | 290.2 | 290.2 | 29.3 | 29.3 |
| Global climate regulation - Carbon sequestration in biomass | tonnes C / year | 12.2 | 12.2 | 21.9 | 21.9 | 9.7 | 9.7 |

| Monetary valuation | Unit | Difference future - present |  |
| --- | --- | --- | --- |
|  |  | Low | High |
| Global climate regulation - Carbon sequestration in soils | € / year | 2928 | 10719 |
| Global climate regulation - Carbon sequestration in biomass | € / year | 975 | 3568 |
| Total | € / year | 3903 | 14287 |

### 3.5.2.2 InVEST 的碳相关计算

目前 InVEST 工具提供有[碳储存和封存](https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/zh/carbonstorage.html) ㉑，[海岸带蓝碳（Coastal Blue Carbon）](https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/zh/coastal_blue_carbon.html)㉒和[森林碳边缘效应（Forest Carbon Edge Effect）](https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/zh/carbon_edge.html)㉓等碳相关计算模型。就 InVEST 的碳储存和封存模型，使用土地利用地图，及其对应包含有地上与地下生物量（biomass）、土壤及死的有机物质（dead organic matter）等4个碳库的存量来估算景观中当前储存的碳量，和随时间推移的碳封存量。并可使用碳封存的市场或社会价值，年变化率和折价率（discount rate）估计碳储存和封存做为调节服务对社会的价值[23]。InVEST 提供有[案例数据](http://releases.naturalcapitalproject.org/?prefix=invest/3.13.0/data/)㉔，提供的威拉米特（willamette）区域的土地利用/覆盖（LULC）数据包含有102个分类（含1个未知分类），碳库数据应该有一一对应分类的值。提供的碳库文件单位为 吨/公顷（t/ha），LULC 栅格高空分辨率为 30m，计算每一栅格单元的碳储量需要将碳库中各分类的每公顷碳储量转换为每单位的碳储量计算碳库中的碳储存地图，对应InVEST中的代码如下：

def \_generate\_carbon\_map(self,lulc\_path, carbon\_pool\_by\_type, out\_carbon\_stock\_path):  
 """Generate carbon stock raster by mapping LULC values to carbon pools.  
  
 Args:  
 lulc\_path (string): landcover raster with integer pixels.  
 out\_carbon\_stock\_path (string): path to output raster that will have  
 pixels with carbon storage values in them with units of Mg\*C  
 carbon\_pool\_by\_type (dict): a dictionary that maps landcover values  
 to carbon storage densities per area (Mg C/Ha).  
  
 Returns:  
 None.  
 """  
 lulc\_info = pygeoprocessing.get\_raster\_info(lulc\_path)  
 pixel\_area = abs(numpy.prod(lulc\_info['pixel\_size']))  
 carbon\_stock\_by\_type = dict([  
 (lulcid, stock \* pixel\_area / 10\*\*4)  
 for lulcid, stock in carbon\_pool\_by\_type.items()])   
 reclass\_error\_details = {  
 'raster\_name': 'LULC', 'column\_name': 'lucode',  
 'table\_name': 'Carbon Pools'}  
 utils.reclassify\_raster(  
 (lulc\_path, 1), carbon\_stock\_by\_type, out\_carbon\_stock\_path,gdal.GDT\_Float32, \_CARBON\_NODATA, reclass\_error\_details)

如果提供了当前和未来的LULC地图，就可以计算碳储量随时间变化的净变化（net change），即封存和损失（sequestration and loss），计算的方法为两个时期碳储存地图栅格差值。同时，也可以提供减少森林砍伐和退化 （reduced (carbon) emissions from deforestation and degradation ，REDD）政策情景下的LULC地图，用于支持政府、非政府组织和企业的一系列决策下 LULC 变化的场景分析（Scenario Analysis）。

将碳封存（不是碳储存）货币化转换为经济价值衡量标准，其计算公式为：，式中，是每公吨碳的价格，为每公吨元素碳的价格（不是），推荐用释放一吨碳造成的相关环境损害成本-碳的社会成本（social cost of carbon，SCC）进行价值估算[24,25,26]；为分析地块的碳封存量，以吨（tons）计；为未来的年份；为当前的年份；是碳价格的年度市场折价率（discount rate），反映了社会对眼前利益于未来利益的偏好，其值越低表明碳价格的波动越小，如果值为 0% 则碳的货币价值不折价。通常配置碳价格的年度市场折价率为一个较低值，例如美国政府推荐值为 7%，来表明影响气候变化的碳封存具有较高的社会经济价值；是碳价格的年变化率。因为碳排放对气候变化的影响随时间的推移带来相关的损害变化，这将改变碳封存的社会价值，例如配置年变化率大于 0% 意味着未来封存碳的社会价值小于现在封存的碳的价值。为了减小未来气候变化（例如温度升高 ）对全球经济造成的重大破坏，立即减少温室气体的排放是必要的，因此配置大于0的折价率表明减少碳排放，增加碳封存的重要性，以降低气候变化的风险。如果折价率配置小于 0%（例如 -2%） ，意味着未来碳封存的社会价值大于现在的碳封存价值。对应InVEST中的代码如下：

def \_calculate\_valuation\_constant(self,lulc\_cur\_year, lulc\_fut\_year, discount\_rate, rate\_change,price\_per\_metric\_ton\_of\_c):  
 """Calculate a net present valuation constant to multiply carbon storage.  
  
 Args:  
 lulc\_cur\_year (int): calendar year in present  
 lulc\_fut\_year (int): calendar year in future  
 discount\_rate (float): annual discount rate as a percentage  
 rate\_change (float): annual change in price of carbon as a percentage  
 price\_per\_metric\_ton\_of\_c (float): currency amount of Mg of carbon  
  
 Returns:  
 a floating point number that can be used to multiply a delta carbon  
 storage value by to calculate NPV.  
 """  
 n\_years = lulc\_fut\_year - lulc\_cur\_year  
 ratio = (  
 1 / ((1 + discount\_rate / 100) \*  
 (1 + rate\_change / 100)))  
 valuation\_constant = (price\_per\_metric\_ton\_of\_c / n\_years)  
 # note: the valuation formula in the user's guide uses sum notation.  
 # here it's been simplified to remove the sum using the general rule  
 # sum(r^k) from k=0 to N == (r^(N+1) - 1) / (r - 1)  
 # where N = n\_years-1 and r = ratio  
 if ratio == 1:  
 # if ratio == 1, we would divide by zero in the equation below  
 # so use the limit as ratio goes to 1, which is n\_years  
 valuation\_constant \*= n\_years  
 else:  
 valuation\_constant \*= (1 - ratio \*\* n\_years) / (1 - ratio)  
 return valuation\_constant

将 InVEST 的碳储存和封存计算代码迁移至 USDA库调用计算。

%load\_ext autoreload   
%autoreload 2   
import warnings  
warnings.filterwarnings('ignore')  
  
from usda.migrated\_project.invest.esv import u  
import usda.migrated\_project.invest as usda\_esv  
from usda.data\_visualization import generate\_colors,cmap\_patch\_build  
import usda.data\_visual as usda\_vis  
  
import pandas as pd  
import rioxarray as rxr  
import matplotlib.pyplot as plt  
import matplotlib as mpl  
import numpy as np

The autoreload extension is already loaded. To reload it, use:  
 %reload\_ext autoreload

配置参数，包括计算产生的过程文件和最终文件的保存路径workspace\_dir，碳库数据文件路径carbon\_pools\_path，线程数n\_workers，结果文件的后缀results\_suffix，LULC文件路径，包括当前（results\_suffix）、未来（lulc\_fut\_path）和用于场景分析的REDD情景（lulc\_redd\_path），当前和未来LULC所属年份lulc\_cur\_yea和lulc\_fut\_year，折价率discount\_rate，年变化率'rate\_change和碳价格price\_per\_metric\_ton\_of\_c等，可以通过help(usda\_esv.Carbon\_storage\_sequestration)获得所有参数的说明。

help(usda\_esv.Carbon\_storage\_sequestration)

Help on class Carbon\_storage\_sequestration in module usda.migrated\_project.invest.esv.\_carbon:  
  
class Carbon\_storage\_sequestration(builtins.object)  
 | Carbon\_storage\_sequestration(args)  
 |   
 | Carbon.  
 |   
 | Calculate the amount of carbon stocks given a landscape, or the difference  
 | due to a future change, and/or the tradeoffs between that and a REDD  
 | scenario, and calculate economic valuation on those scenarios.  
 |   
 | The model can operate on a single scenario, a combined present and future  
 | scenario, as well as an additional REDD scenario.  
 |   
 | Args:  
 | args['workspace\_dir'] (string): a path to the directory that will  
 | write output and other temporary files during calculation.  
 | args['results\_suffix'] (string): appended to any output file name.  
 | args['lulc\_cur\_path'] (string): a path to a raster representing the  
 | current carbon stocks.  
 | args['calc\_sequestration'] (bool): if true, sequestration should  
 | be calculated and 'lulc\_fut\_path' and 'do\_redd' should be defined.  
 | args['lulc\_fut\_path'] (string): a path to a raster representing future  
 | landcover scenario. Optional, but if present and well defined  
 | will trigger a sequestration calculation.  
 | args['do\_redd'] ( bool): if true, REDD analysis should be calculated  
 | and 'lulc\_redd\_path' should be defined  
 | args['lulc\_redd\_path'] (string): a path to a raster representing the  
 | alternative REDD scenario which is only possible if the  
 | args['lulc\_fut\_path'] is present and well defined.  
 | args['carbon\_pools\_path'] (string): path to CSV or that indexes carbon  
 | storage density to lulc codes. (required if 'do\_uncertainty' is  
 | false)  
 | args['lulc\_cur\_year'] (int/string): an integer representing the year  
 | of `args['lulc\_cur\_path']` used if `args['do\_valuation']`  
 | is True.  
 | args['lulc\_fut\_year'](int/string): an integer representing the year  
 | of `args['lulc\_fut\_path']` used in valuation if it exists.  
 | Required if `args['do\_valuation']` is True and  
 | `args['lulc\_fut\_path']` is present and well defined.  
 | args['do\_valuation'] (bool): if true then run the valuation model on  
 | available outputs. Calculate NPV for a future scenario or a REDD  
 | scenario and report in final HTML document.  
 | args['price\_per\_metric\_ton\_of\_c'] (float): Is the present value of  
 | carbon per metric ton. Used if `args['do\_valuation']` is present  
 | and True.  
 | args['discount\_rate'] (float): Discount rate used if NPV calculations  
 | are required. Used if `args['do\_valuation']` is present and  
 | True.  
 | args['rate\_change'] (float): Annual rate of change in price of carbon  
 | as a percentage. Used if `args['do\_valuation']` is present and  
 | True.  
 | args['n\_workers'] (int): (optional) The number of worker processes to  
 | use for processing this model. If omitted, computation will take  
 | place in the current process.  
 |   
 | Returns:  
 | None.  
 |   
 | Methods defined here:  
 |   
 | \_\_init\_\_(self, args)  
 | Initialize self. See help(type(self)) for accurate signature.  
 |   
 | ----------------------------------------------------------------------  
 | Data descriptors defined here:  
 |   
 | \_\_dict\_\_  
 | dictionary for instance variables (if defined)  
 |   
 | \_\_weakref\_\_  
 | list of weak references to the object (if defined)

args\_carbon={  
 'workspace\_dir':'I:\ESVs\carbon',  
 'carbon\_pools\_path':r'I:\data\invest\Carbon\carbon\_pools\_willamette.csv',  
 'n\_workers':6,  
 'results\_suffix':'willamette',  
 'lulc\_cur\_path':r'I:\data\invest\Carbon\lulc\_current\_willamette.tif',  
 'lulc\_fut\_path':r'I:\data\invest\Carbon\lulc\_future\_willamette.tif',  
 'lulc\_redd\_path':r"I:\data\invest\Carbon\lulc\_redd\_willamette.tif",  
 'do\_valuation':True,  
 'do\_redd':True,  
 'lulc\_cur\_year':2020,  
 'lulc\_fut\_year':2050,  
 'discount\_rate':0.07,  
 'rate\_change':0.05,  
 'price\_per\_metric\_ton\_of\_c':80,  
 }

碳库和 LULC 分类相对应，因此碳储存和封存的计算受到这两个数据及其相互牵制的影响。碳库数据通常使用已有研究测量结果，例如石羊河流域不同土地利用类型的碳密度（）[27]。

| 土地利用类型 | 地上碳密度（C\_above） | 地下碳密度（C\_below） | 土壤有机质（C\_soil） | 死亡有机质（C\_soil） |
| --- | --- | --- | --- | --- |
| 耕地 | 5.7 | 80.7 | 108.4 | 13 |
| 林地 | 42.4 | 115.9 | 236.9 | 13 |
| 草地 | 35.3 | 86.5 | 99.9 | 2 |
| 水域 | 0 | 0 | 0 | 0 |
| 建筑用地 | 1.2 | 0 | 0 | 0 |
| 未利用地 | 9.1 | 0 | 21.6 | 0 |

又或者[杨元合课题组发布青藏高原高寒草地土壤碳库及其动态变化专题数据](http://www.nesdc.org.cn/collection/view/6124789b7e28172cbed3d803)㉕等。

InVEST提供的威拉米特区域的碳库细分有102个类别，增加了碳储存和封存计算结果精度。该碳库为LULC类别的平均碳储存值，其理想数据源应为实地测量估算值。但是如果无法实现实地测量，可以使用一些通用的数据源，例如对于地上生物量中储存的碳 ，一个非常普遍使用的数据源是 IPCC于2006年制定的[国家温室气体清单指南（农业、森林和其它土地利用）](https://www.ipcc-nggip.iges.or.jp/public/2006gl/vol4.html)㉖。指南中的表格给出了多种土地利用类型地上生物量估算值，因为InVEST使用元素碳的质量，因此需要乘以一个转换系数将公吨的生物量转换为公吨的碳，该系数通常在0.43到0.51之间变化，可以从指南相关表格中查询；对于地下生物量中存储的碳，如果LULC主要为木质生物量（woody biomass），可以从指南中提供的根茎比（root to shoot）估算。但是如果LULC几乎不含木质生物量，却包含大量地下碳储量（例如草原灌丛等），通常需要本地测量数据。如果该数据不可获得，可从使用一些全球估算值，例如 IPCC 指南相关表格列出了每个气候区的总生物量（地上和地下）和地上生物量，可以同差值粗略估计地下生物量；对于土壤中存储的碳，指南中相应表格给出了按土壤类型划分的土壤碳储量估计值；储存在死亡有机物中的碳，指南中也给出了森林土地利用/覆盖类型中落叶的默认碳储量。对于非林地类型，枯落物接近为0。InVEST 也给出了其它一些碳库碳储存估算的[数据来源](https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/zh/data_sources.html" \l "belowground-biomass)㉗ [28]。

carbon\_pools\_willamette=pd.read\_csv(args\_carbon['carbon\_pools\_path'])  
print(carbon\_pools\_willamette.to\_string())

lucode LULC\_Name C\_above C\_below C\_soil C\_dead  
0 1 Residential 0-4 units/acre 15 10 60 1.0  
1 2 Residential 4-9 units/acre 5 3 20 0.0  
2 3 Residential 9-16 units/acre 2 1 5 0.0  
3 4 Residential >16 units/acre 0 0 0 0.0  
4 5 Vacant 10 20 10 5.0  
5 6 Commercial 0 0 0 0.0  
6 7 Commercial/Industrial 0 0 0 0.0  
7 8 Industrial 0 0 0 0.0  
8 9 Industrial & Commercial 0 0 0 0.0  
9 10 Residential & Commercial 0 0 0 0.0  
10 11 Urban non-vegetated unknown 0 0 10 0.0  
11 16 Rural structures 0 0 50 0.0  
12 18 Railroad 0 0 25 0.0  
13 19 Primary roads 0 0 50 0.0  
14 20 Secondary roads 0 0 25 0.0  
15 21 Light duty roads 0 0 35 0.0  
16 24 Rural non-vegetated unknown 0 0 0 0.0  
17 29 Main channel non-vegetated 0 0 0 0.0  
18 32 Stream orders 5-7 0 0 0 0.0  
19 33 Permanent lentic water 0 0 0 0.0  
20 39 Topographic Shadow 100 5 65 50.0  
21 40 Snow 0 0 0 0.0  
22 42 Barren 0 0 0 0.0  
23 49 Urban tree overstory 135 50 75 1.0  
24 51 Upland Forest open 75 45 85 20.0  
25 52 Upland Forest Semi-closed mixed 90 60 110 30.0  
26 53 Forest Closed hardwood 180 120 120 55.0  
27 54 Forest Closed mixed 200 130 130 65.0  
28 55 Upland Forest Semi-closed conifer 90 60 95 29.0  
29 56 Conifers 0-20 yrs 10 7 76 2.5  
30 57 Forest closed conifer 21-40 yrs 88 59 96 29.0  
31 58 Forest closed conifer 41-60 yrs 165 110 115 50.0  
32 59 Forest closed conifer 61-80 yrs 225 150 124 65.0  
33 60 Forest closed conifer 81-200 yrs 300 200 135 85.0  
34 61 Forest closed conifer older than 200 yrs 375 250 150 100.0  
35 62 Upland Forest Semi-closed hardwood 80 50 100 25.0  
36 66 Hybrid poplar 75 25 90 2.0  
37 67 Grass seed rotation 1 1 10 0.0  
38 68 Irrigated annual crop rotation 2 1 10 0.0  
39 71 Grains 3 2 10 0.0  
40 72 Nursery 10 3 90 1.0  
41 73 Berries & Vineyards 8 5 20 0.0  
42 74 Double cropping 5 2 10 0.0  
43 75 Hops 5 4 20 0.0  
44 76 Mint 2 1 10 0.0  
45 77 Radish seed 2 1 10 0.0  
46 78 Sugar beet seed 2 1 10 0.0  
47 79 Row crop 3 2 10 0.0  
48 80 Grass 1 1 10 0.0  
49 81 Burned grass 0 1 10 0.0  
50 82 Field crop 3 2 8 1.0  
51 83 Hayfield 5 4 23 1.0  
52 84 Late field crop 5 3 15 0.0  
53 85 Pasture 5 4 25 1.0  
54 86 Natural grassland 6 6 20 2.0  
55 87 Natural shrub 8 8 25 3.0  
56 88 Bare/fallow 1 1 10 0.0  
57 89 Flooded/marsh 10 5 20 0.0  
58 90 Irrigated perennial 5 5 15 0.0  
59 91 Turfgrass 1 1 10 0.0  
60 92 Orchard 125 5 115 1.0  
61 93 Christmas trees 13 28 95 2.0  
62 95 Conifer Woodlot 275 30 95 10.0  
63 98 Oak savanna 100 20 115 50.0  
64 101 Wet shrub 7 3 25 0.0  
65 102 Unknown 0 0 0 0.0

查看当前、未来和REDD情景的 LULC 数据。

在打印地图时，随机配置了 LULC分类颜色。

lulc\_cur=rxr.open\_rasterio(args\_carbon['lulc\_cur\_path'],masked=True).squeeze()  
lulc\_fut=rxr.open\_rasterio(args\_carbon['lulc\_fut\_path'],masked=True).squeeze()  
lulc\_redd=rxr.open\_rasterio(args\_carbon['lulc\_redd\_path'],masked=True).squeeze()  
  
print(lulc\_cur.rio.crs)  
lulc\_cur

EPSG:26910

<xarray.DataArray (y: 1889, x: 1325)>

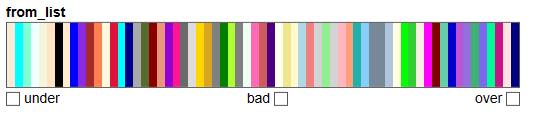
[2502925 values with dtype=float32]

Coordinates:

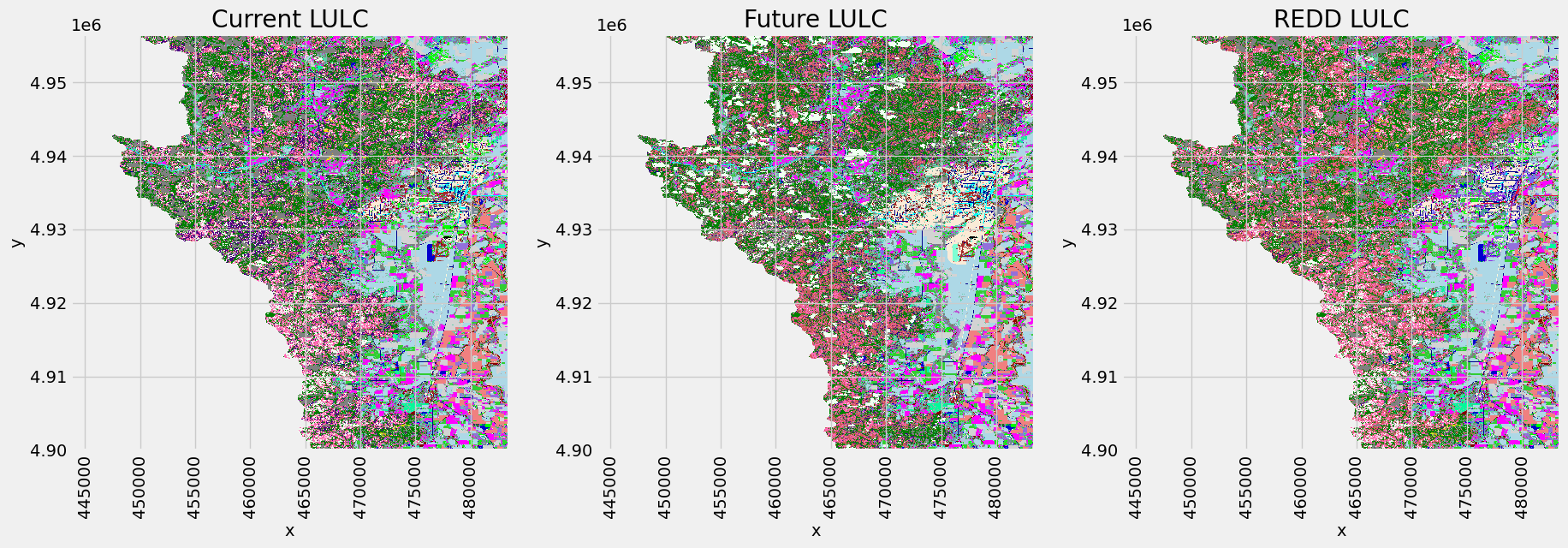
band int32 1

* x (x) float64 4.437e+05 4.438e+05 ... 4.834e+05 4.835e+05
* y (y) float64 4.957e+06 4.957e+06 4.956e+06 ... 4.9e+06 4.9e+06
* spatial\_ref int32 0
* Attributes: (12/13)
* AREA\_OR\_POINT: Area
* DataType: Generic
* RepresentationType: THEMATIC
* STATISTICS\_COVARIANCES: 294.9522998374362
* STATISTICS\_MAXIMUM: 95
* STATISTICS\_MEAN: 62.352993778709
* ... ...
* STATISTICS\_SKIPFACTORX: 1
* STATISTICS\_SKIPFACTORY: 1
* STATISTICS\_STDDEV: 17.174170144902
* STATISTICS\_VALID\_PERCENT: 65.59
* scale\_factor: 1.0
* add\_offset: 0.0

hex\_colors\_only,\_,\_=generate\_colors()  
labels\_cmap\_dict=dict(zip(carbon\_pools\_willamette.lucode,carbon\_pools\_willamette.LULC\_Name))  
labels\_cmap\_dict={k:[v,hex\_colors\_only[k]] for k,v in labels\_cmap\_dict.items()}  
cmap\_LC,norm,patches=cmap\_patch\_build(labels\_cmap\_dict,)  
cmap\_LC



f, axes=plt.subplots(1,3,figsize=(20,6))  
lulc\_cur.plot.imshow(ax=axes[0],add\_colorbar=False,cmap=cmap\_LC,norm=norm)  
lulc\_fut.plot.imshow(ax=axes[1],add\_colorbar=False,cmap=cmap\_LC,norm=norm)  
lulc\_redd.plot.imshow(ax=axes[2],add\_colorbar=False,cmap=cmap\_LC,norm=norm)  
  
axes[0].set\_title('Current LULC')  
axes[1].set\_title('Future LULC')  
axes[2].set\_title('REDD LULC')  
for ax in axes: ax.tick\_params(axis='x', labelrotation=90)  
plt.show()



调用usda\_esv.Carbon\_storage\_sequestration()方法计算碳储存和封存。为了方便查看结果数据和进行后续计算，在迁移InVEST代码时，增加了carbon\_results\_info属性，用于存储过程文件和结果文件保存路径。结果文件中，

tot\_c\_cur\_[Suffix].tif/tot\_c\_fut\_[Suffix].tif/tot\_c\_redd\_[Suffix].tif格式栅格文件为当前、未来和REDD方案每个像素（案例为）存储的碳量（碳储存）；

delta\_cur\_fut\_[Suffix].tif/delta\_cur\_redd\_[Suffix].tif为未来/REDD景观与当前景观之间的碳储存差异（封存），以公吨/像素（栅格单元）为单位。如果值为正则为封存的碳，如果为负值则表示有碳流失；

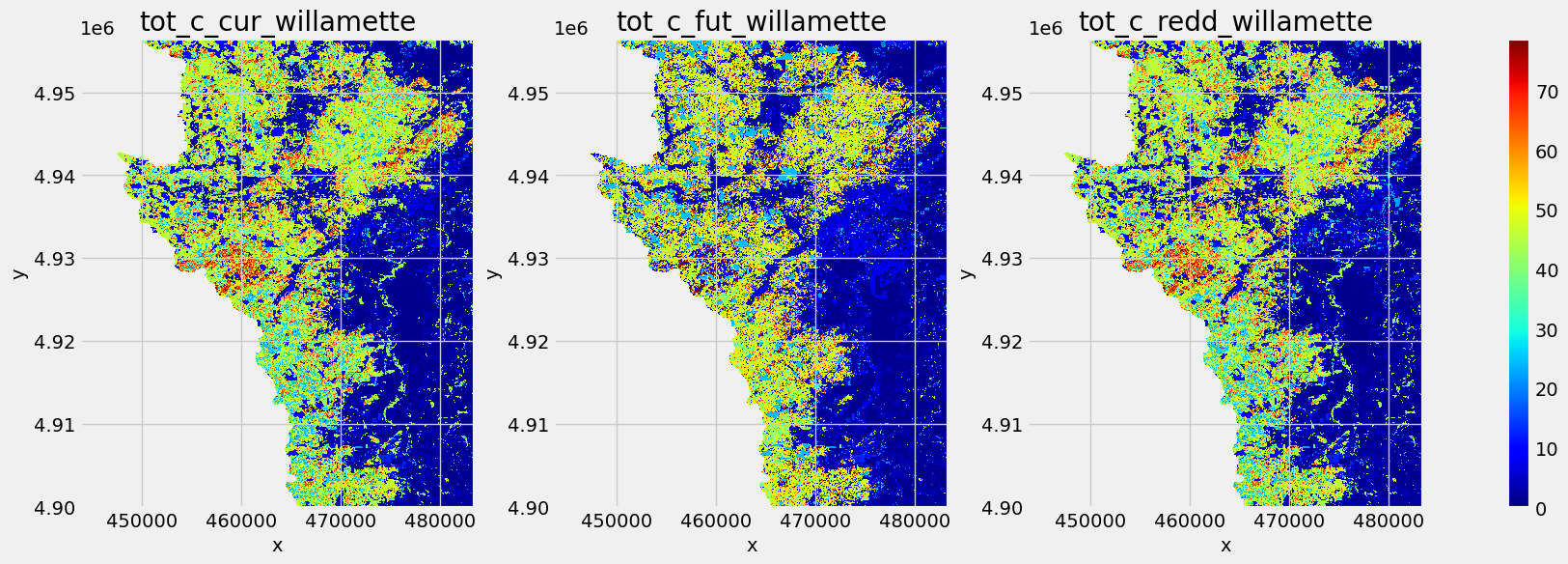
npv\_fut\_[Suffix].tif/npv\_redd\_[Suffix].tif为未来/REDD景观与当前景观之间的碳封存的社会经济价值（货币化），单位是货币/像素（栅格单元）。

carbon=usda\_esv.Carbon\_storage\_sequestration(args\_carbon)  
carbon.carbon\_results\_info['tifs\_to\_summarize']

cell size set={(30.0, -30.0)};  
raster size set={(1325, 1889)};  
valid lulc keys=['lulc\_cur\_path', 'lulc\_fut\_path', 'lulc\_redd\_path']  
valid scenarios=['cur', 'fut', 'redd']  
  
  
  
  
  
{'I:\\ESVs\\carbon\\delta\_cur\_fut\_willamette.tif',  
 'I:\\ESVs\\carbon\\delta\_cur\_redd\_willamette.tif',  
 'I:\\ESVs\\carbon\\npv\_fut\_willamette.tif',  
 'I:\\ESVs\\carbon\\npv\_redd\_willamette.tif',  
 'I:\\ESVs\\carbon\\tot\_c\_cur\_willamette.tif',  
 'I:\\ESVs\\carbon\\tot\_c\_fut\_willamette.tif',  
 'I:\\ESVs\\carbon\\tot\_c\_redd\_willamette.tif'}

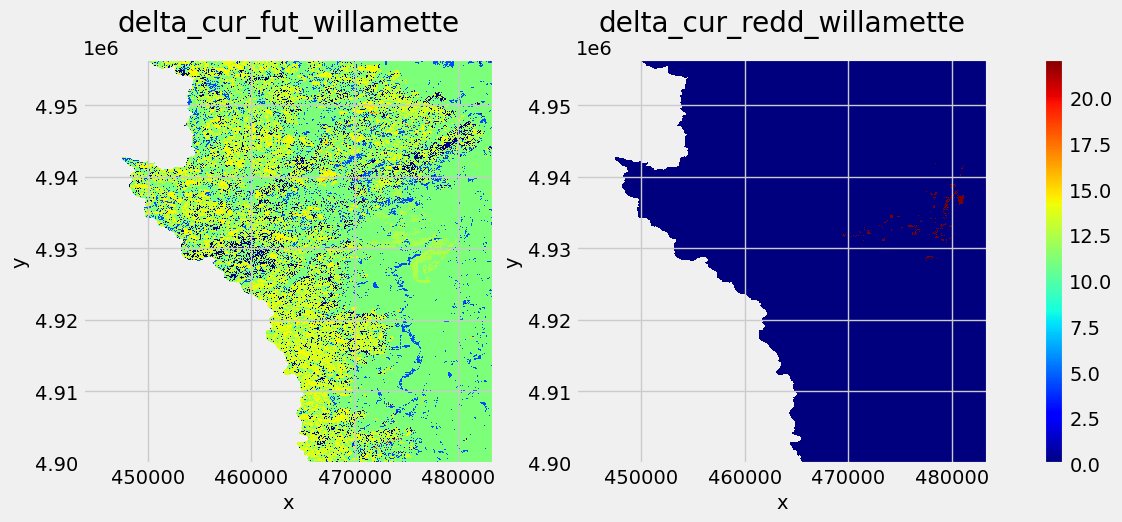
* 打印查看各时期和规划情景（REDD）下的碳储存

tot\_c\_cur\_willamette=rxr.open\_rasterio(r'I:\\ESVs\\carbon\\tot\_c\_cur\_willamette.tif',masked=True).squeeze()  
tot\_c\_fut\_willamette=rxr.open\_rasterio(r'I:\\ESVs\\carbon\\tot\_c\_fut\_willamette.tif',masked=True).squeeze()  
tot\_c\_redd\_willamette=rxr.open\_rasterio(r'I:\\ESVs\\carbon\\tot\_c\_redd\_willamette.tif',masked=True).squeeze()  
  
f, axes=plt.subplots(1,3,figsize=(20,6))  
cmap='jet'  
tot\_c\_cur\_willamette.plot.imshow(ax=axes[0],add\_colorbar=False,cmap=cmap)  
tot\_c\_fut\_willamette.plot.imshow(ax=axes[1],add\_colorbar=False,cmap=cmap)  
ax=tot\_c\_redd\_willamette.plot.imshow(ax=axes[2],add\_colorbar=False,cmap=cmap)  
  
axes[0].set\_title('tot\_c\_cur\_willamette')  
axes[1].set\_title('tot\_c\_fut\_willamette')  
axes[2].set\_title('tot\_c\_redd\_willamette')  
  
cax,kw = mpl.colorbar.make\_axes([ax for ax in axes.flat])  
plt.colorbar(ax, cax=cax, \*\*kw)  
plt.show()



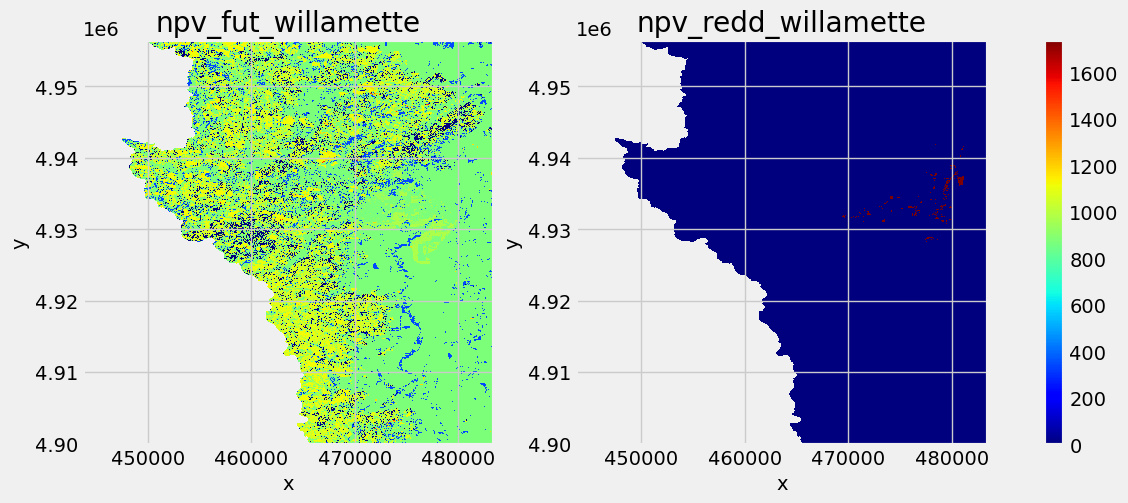
* 打印查看碳封存

delta\_cur\_fut\_willamette=rxr.open\_rasterio(r'I:\\ESVs\\carbon\\delta\_cur\_fut\_willamette.tif',masked=True).squeeze()  
delta\_cur\_redd\_willamette=rxr.open\_rasterio(r'I:\\ESVs\\carbon\\delta\_cur\_redd\_willamette.tif',masked=True).squeeze()  
  
f, axes=plt.subplots(1,2,figsize=(13,5))  
cmap='jet'  
delta\_cur\_fut\_willamette.plot.imshow(ax=axes[0],add\_colorbar=False,cmap=cmap)  
ax=delta\_cur\_redd\_willamette.plot.imshow(ax=axes[1],add\_colorbar=False,cmap=cmap)  
  
axes[0].set\_title('delta\_cur\_fut\_willamette')  
axes[1].set\_title('delta\_cur\_redd\_willamette')  
  
cax,kw = mpl.colorbar.make\_axes([ax for ax in axes.flat])  
plt.colorbar(ax, cax=cax, \*\*kw)  
plt.show()



* 打印查看碳封存的经济价值

npv\_fut\_willamette=rxr.open\_rasterio(r'I:\\ESVs\\carbon\\npv\_fut\_willamette.tif',masked=True).squeeze()  
npv\_redd\_willamette=rxr.open\_rasterio(r'I:\\ESVs\\carbon\\npv\_redd\_willamette.tif',masked=True).squeeze()  
  
f, axes=plt.subplots(1,2,figsize=(13,5))  
cmap='jet'  
npv\_fut\_willamette.plot.imshow(ax=axes[0],add\_colorbar=False,cmap=cmap)  
ax=npv\_redd\_willamette.plot.imshow(ax=axes[1],add\_colorbar=False,cmap=cmap)  
  
axes[0].set\_title('npv\_fut\_willamette')  
axes[1].set\_title('npv\_redd\_willamette')  
  
cax,kw = mpl.colorbar.make\_axes([ax for ax in axes.flat])  
plt.colorbar(ax, cax=cax, \*\*kw)  
plt.show()



## 3.5.3 生境质量（Habitat Quality）、 作物授粉（授粉者丰度）（Crop Pollination (Pollinator Abundance)）和作物生产（Crop Production）

### 3.5.3.1 生境质量（Habitat Quality）[29]

#### 1) 生境适宜性数值配置

InVEST 生境质量模型首先需要确定 LULC 数据中哪些类型应被考虑为生境，这可以用简单的二分法，值1表示为生境，而值0表示为非生境。二分法并不体现生境的重要性、稀有性或适宜性，所有生境类型均为值1，表征为平等的关系。模型的输入也假设不针对任何特定的物种，而是适用于更一般的生物多样性；如果要考虑生境的重要性、稀有性或适宜性，则值的配置为从0至1的一个连续值，1表示最高的生境适宜性。值越低则表示该 LULC 类型越不适宜于生物生存。对应 LULC 类型的生境（适宜性）配置，位于案例数据文件sensitivity\_willamette.csv下的HABITAT字段，如下代码（为二分法）。用（）表示 LULC 类型的生境适宜性。

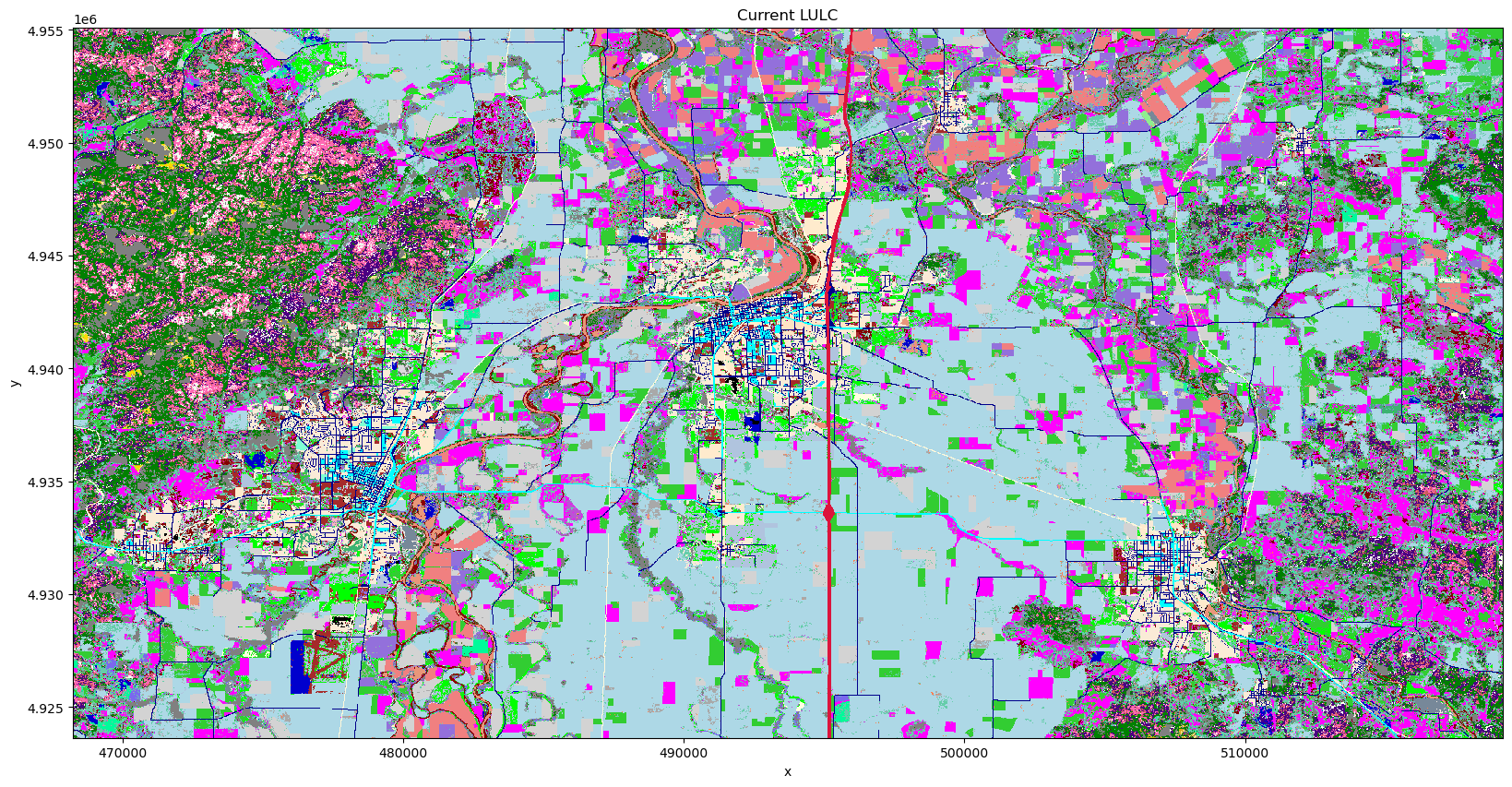
args\_habitat\_quality={  
 'workspace\_dir':'I:\ESVs\habitat\_quality',   
 'n\_workers':6,  
 'results\_suffix':'willamette',   
 'threats\_table\_path':r'I:\data\invest\HabitatQuality\threats\_willamette.csv',  
 'sensitivity\_table\_path':r'I:\data\invest\HabitatQuality\sensitivity\_willamette.csv',  
 'half\_saturation\_constant':0.05,  
 'lulc\_cur\_path':r'I:\data\invest\HabitatQuality\lulc\_current\_willamette.tif',  
 'lulc\_fut\_path':r'I:\data\invest\HabitatQuality\lulc\_future\_willamette.tif',  
 'lulc\_bas\_path':r'I:\data\invest\HabitatQuality\lulc\_baseline\_willamette.tif',  
 'access\_vector\_path':r'I:\data\invest\HabitatQuality\accessibility\_willamette.shp',   
 }

* 现状土地覆盖（Current Land Cover）

lulc\_cur=rxr.open\_rasterio(args\_habitat\_quality['lulc\_cur\_path'],masked=True).squeeze()  
print(lulc\_cur.rio.crs)

EPSG:26910

f, ax=plt.subplots(1,1,figsize=(20,10))  
lulc\_cur.plot.imshow(ax=ax,add\_colorbar=False,cmap=cmap\_LC,norm=norm)  
  
ax.set\_title('Current LULC')  
plt.show()



* 敏感性表格（Sensitivity Table）

sensitivity\_table=pd.read\_csv(args\_habitat\_quality['sensitivity\_table\_path'])  
print(sensitivity\_table.to\_string())

LULC NAME HABITAT crops railroad urban timber roads1 roads2 roads3  
0 1 Residential 0-4 units/acre 1 0.4 0.45 0.6 0.2 0.4 0.3 0.2  
1 2 Residential 4-9 units/acre 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
2 3 Residential 9-16 units/acre 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
3 4 Residential >16 units/acre 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
4 6 Commercial 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
5 7 Commercial/Industrial 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
6 8 Industrial 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
7 10 Industrial 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
8 11 Urban non-vegetated unknown 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
9 16 Rural structures 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
10 18 Railroad 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
11 19 Primary roads 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
12 20 Secondary roads 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
13 21 Light duty roads 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
14 24 Rural non-vegetated unknown 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
15 29 Main channel non-vegetated 1 0.7 0.75 0.9 0.5 0.7 0.6 0.5  
16 32 Stream orders 5-7 1 0.7 0.75 0.9 0.5 0.7 0.6 0.5  
17 33 Permanent lentic water 1 0.7 0.75 0.9 0.5 0.7 0.6 0.5  
18 39 Topographic Shadow 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
19 40 Snow 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
20 42 Barren 0 0.0 0.00 0.0 0.0 0.0 0.0 0.0  
21 49 Urban tree overstory 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
22 51 Upland Forest open 1 0.8 0.85 1.0 0.6 0.8 0.7 0.6  
23 52 Upland Forest Semi-closed mixed 1 0.7 0.75 0.9 0.5 0.7 0.6 0.5  
24 53 Forest Closed hardwood 1 0.6 0.65 0.8 0.4 0.6 0.5 0.4  
25 54 Forest Closed mixed 1 0.6 0.65 0.8 0.4 0.6 0.5 0.4  
26 55 Upland Forest Semi-closed conifer 1 0.7 0.75 0.9 0.5 0.7 0.6 0.5  
27 56 Conifers 0-20 yrs 1 0.5 0.55 0.7 0.3 0.5 0.4 0.3  
28 57 Forest closed conifer 21-40 yrs 1 0.5 0.55 0.7 0.3 0.5 0.4 0.3  
29 58 Forest closed conifer 41-60 yrs 1 0.5 0.55 0.7 0.3 0.5 0.4 0.3  
30 59 Forest closed conifer 61-80 yrs 1 0.6 0.65 0.8 0.4 0.6 0.5 0.4  
31 60 Forest closed conifer 81-200 yrs 1 1.0 1.00 1.0 0.8 1.0 0.9 0.8  
32 61 Forest closed conifer > 200 yrs 1 1.0 1.00 1.0 0.8 1.0 0.9 0.8  
33 62 Upland Forest Semi-closed hardwood 1 0.7 0.75 0.9 0.5 0.7 0.6 0.5  
34 66 Hybrid poplar 1 0.5 0.55 0.7 0.3 0.5 0.4 0.3  
35 67 Grass seed rotation 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
36 68 Irrigated annual rotation 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
37 71 Grains 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
38 72 Nursery 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
39 73 Berries & Vineyards 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
40 74 Double cropping 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
41 75 Hops 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
42 76 Mint 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
43 77 Radish seed 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
44 78 Sugar beet seed 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
45 79 Row crop 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
46 80 Grass 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
47 81 Burned grass 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
48 82 Field crop 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
49 83 Hayfield 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
50 84 Late field crop 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
51 85 Pasture 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
52 86 Natural grassland 1 0.4 0.45 0.6 0.2 0.4 0.3 0.2  
53 87 Natural shrub 1 0.4 0.45 0.6 0.2 0.4 0.3 0.2  
54 88 Bare/fallow 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
55 89 Flooded/marsh 1 0.7 0.75 0.9 0.5 0.7 0.6 0.5  
56 91 Irrigated perennial 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
57 90 Turfgrass 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
58 92 Orchard 1 0.3 0.35 0.5 0.1 0.3 0.2 0.1  
59 93 Christmas trees 1 0.5 0.55 0.7 0.3 0.5 0.4 0.3  
60 95 Conifer Woodlot 1 0.5 0.55 0.7 0.3 0.5 0.4 0.3

#### 2）生境威胁密度（habitat threat density，HTD）数据及其对生境质量的影响

威胁对生境质量影响的数据位于threats\_willamette.csv配置文件下，其中THREAT为威胁类型，DESCRIP字段为对威胁类型的描述，案例文件包括农业用地、建成区、木材采伐/种植园、铁路和三种类型的道路（一、二级公路和乡村公路）。每个威胁类型对应有现状和未来的栅格文件，用CUR\_PATH和FUT\_PATH字段指定栅格文件名（栅格文件和.csv配置文件位于同一路径下），用于模型计算时调用。字段MAX\_DIST是威胁的最大作用距离。WEIGHT是威胁的影响权重，表示为，值越趋近于1，对生境的相对破坏性也越大。

threats\_table=pd.read\_csv(args\_habitat\_quality['threats\_table\_path'])  
print(threats\_table.to\_string())

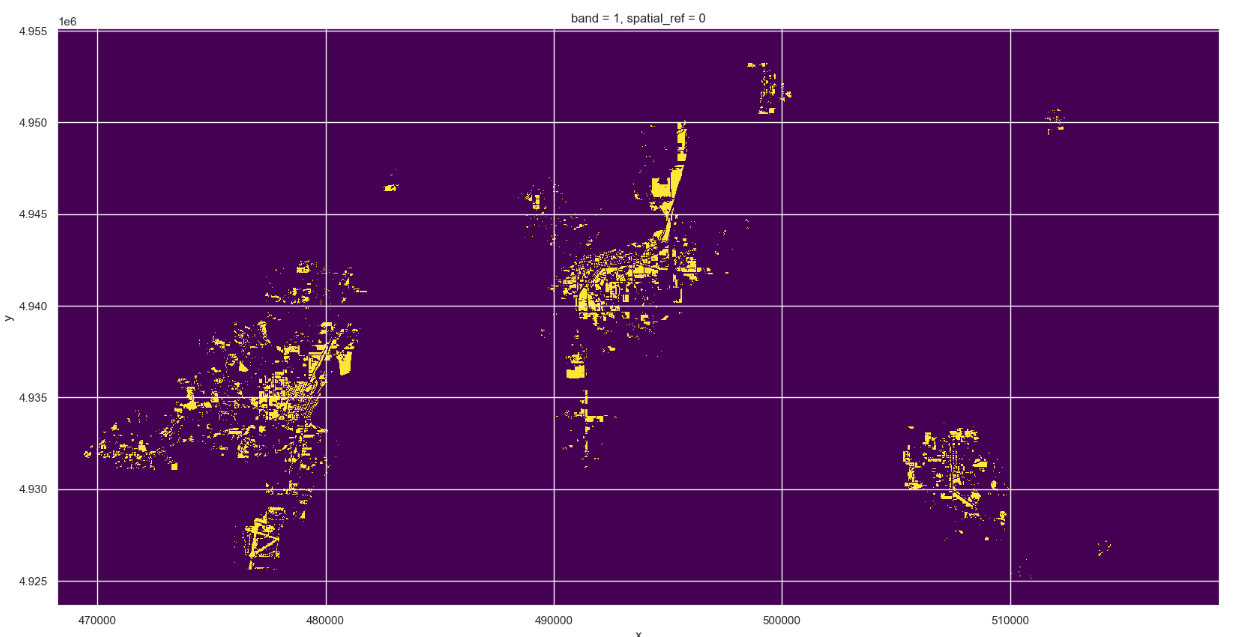
MAX\_DIST WEIGHT THREAT DECAY DESCRIP BASE\_PATH CUR\_PATH FUT\_PATH  
0 8.0 0.7 crops linear Agricultural lands NaN crops\_c.tif crops\_f.tif  
1 5.0 0.6 railroad exponential Railroad lines NaN railroad\_c.tif railroad\_f.tif  
2 10.0 1.0 urban exponential Urban/developed NaN urban\_c.tif urban\_f.tif  
3 6.0 0.5 timber exponential Timber harvesting/Plantations NaN timber\_c.tif timber\_f.tif  
4 3.0 1.0 roads1 linear Primary roads NaN roads1\_c.tif roads1\_f.tif  
5 1.0 0.7 roads2 linear Secondary roads NaN roads2\_c.tif roads2\_f.tif  
6 0.5 0.5 roads3 exponential Rural roads NaN roads3\_c.tif roads3\_f.tif

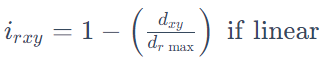
* 建成区威胁栅格数据示例

威胁地图（生境威胁密度）可以表示威胁强度，也可以只是用值1表示威胁的分布（没有威胁的区域用值0表示）。威胁地图栅格单元的威胁数值表示为。

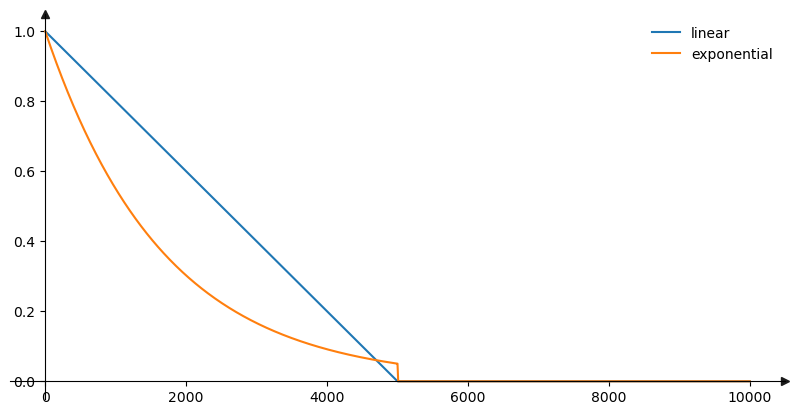
Urban=rxr.open\_rasterio(r'I:\data\invest\HabitatQuality\urban\_c.tif',masked=True).squeeze()  
print(np.unique(Urban.data))  
  
f, ax=plt.subplots(1,1,figsize=(20,10))  
Urban.plot.imshow(ax=ax,add\_colorbar=False)  
  
plt.show()

[0. 1.]



生境的栅格单元与威胁的栅格单元之间距离的远近，影响生境质量。通常随着距离的增加，威胁对生境的影响逐渐减弱，为了表征这种影响关系，可以使用线性函数或指数函数作为距离衰减函数（distance-decay function）计算威胁影响系数。其公式为：，式中，为栅格单元与威胁之间的直线距离，该距离的计算使用[pygeoprocessin库](https://pygeoprocessing.readthedocs.io/en/latest/api/pygeoprocessing.html#pygeoprocessing.distance_transform_edt)㉘的distance\_transform\_edt方法[30]；为威胁的最大作用距离，即MAX\_DIST字段值。为了方便观察距离衰减函数的变化趋势，打印图表如下。

fig, ax=plt.subplots(1, 1,figsize=(10,5))  
   
X=np.arange(0, 10000+10, 10)  
d\_max=5000  
  
y\_linear=[1-(x/d\_max) if x<=d\_max else 0 for x in X]  
ax.plot(X,y\_linear,label='linear')  
  
y\_exponential=[np.exp(-(2.99/d\_max)\*x) if x<=d\_max else 0 for x in X]  
ax.plot(X,y\_exponential,label='exponential')  
  
usda\_vis.plot\_style\_axis\_A(ax)  
plt.legend(loc='upper right',frameon=False)  
plt.show()

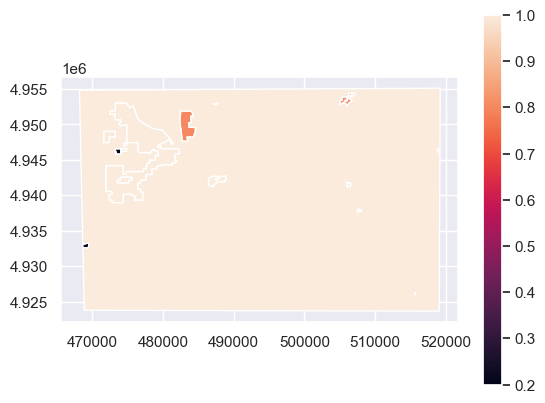


#### 3）可达性（accessibility）

如果限制进入生境区域，例如为生态保护区，或者人类无法接近的高海拔区域，那么威胁对生境的影响将会得到缓解。为了表达这一影响因素，绘制accessibility\_willamette.shp矢量地图（SHP 数据格式），其值域为。值1表示为完全可进入性，值趋近于0则表示可达性逐渐降低。案例给出的可达性文件包含4个区域，可达性为0.2、0.8和1。

args\_habitat\_quality=gpd.read\_file(args\_habitat\_quality['access\_vector\_path'])  
args\_habitat\_quality.plot(column='ACCESS',legend=True);  
args\_habitat\_quality

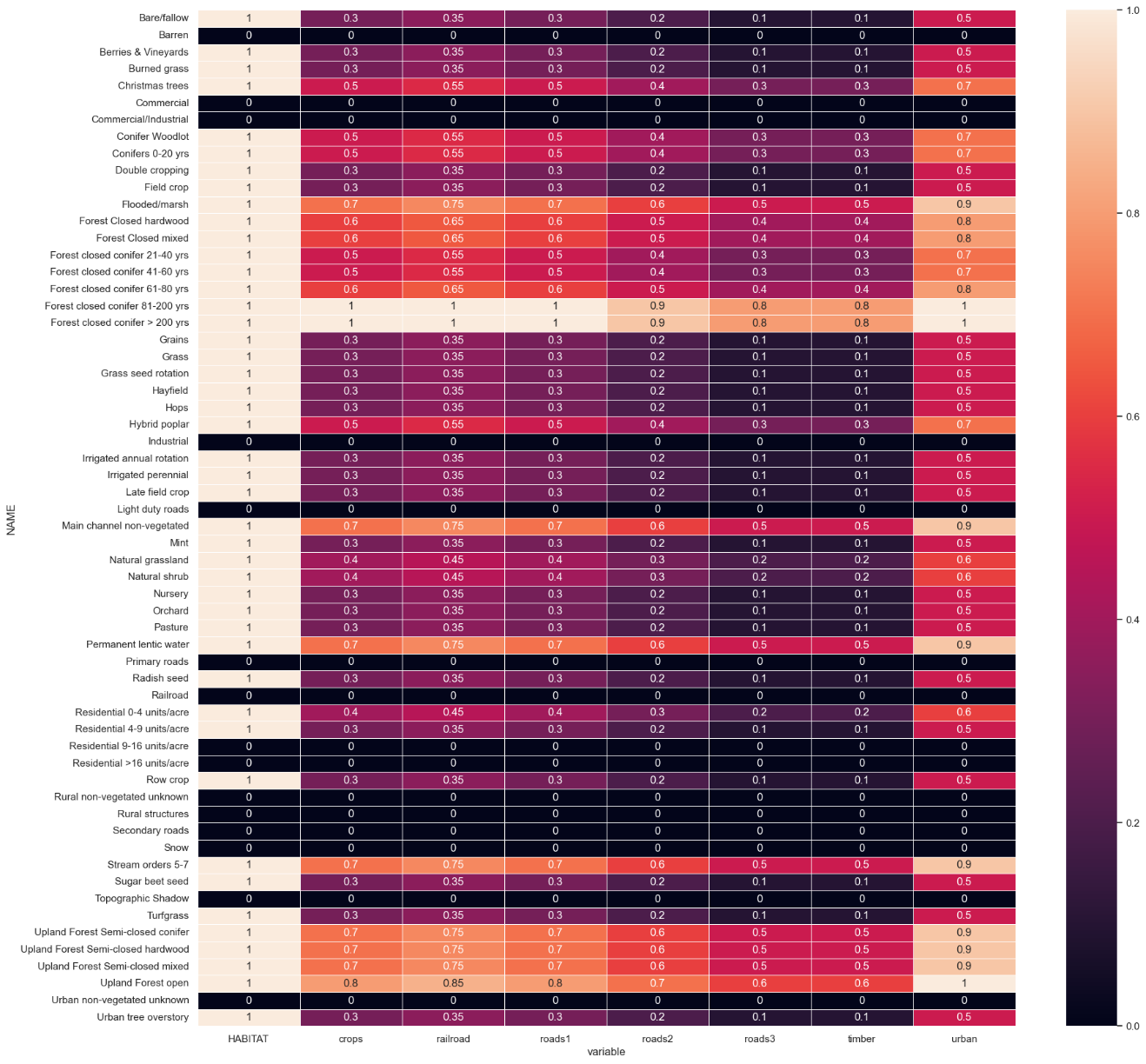
|  | **FID** | **ACCESS** | **geometry** |
| --- | --- | --- | --- |
| **0** | 0 | 0.2 | MULTIPOLYGON (((469563.515 4932894.663, 469559... |
| **1** | 1 | 0.8 | MULTIPOLYGON (((484322.863 4951233.698, 484346... |
| **2** | 2 | 1.0 | MULTIPOLYGON (((515669.675 4926249.355, 515671... |
| **3** | 3 | 1.0 | MULTIPOLYGON (((475648.118 4942540.232, 475644... |



#### 4）生境类型对每一威胁的相对敏感性（relative sensitivity）

敏感性数据位于sensitivity\_willamette.csv数据文件中，为前文的sensitivity\_table变量值。每一种生境类型对威胁的敏感性可能不同，因此通过配置该值来修正威胁对生境的影响，其值域为，式中，代表 LULC 类型（生境类型），代表某类威胁。为了更直观的观察生境类型对每一威胁的相对敏感性，使用热力图的方式图示数值强弱。

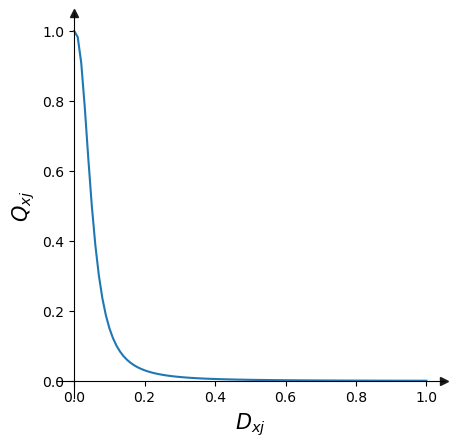
cols=['NAME','HABITAT','crops','railroad','urban','timber','roads1','roads2','roads3']  
data=sensitivity\_table[cols].melt('NAME', value\_name='vals')  
data\_pivot=data.pivot\_table(index='NAME', columns='variable', values='vals',aggfunc='mean') # data\_pivot=data.pivot(index='NAME', columns='variable', values='vals')  
  
f, ax=plt.subplots(figsize=(20, 20))  
sns.heatmap(data\_pivot, annot=True, linewidths=.5, ax=ax);



基于以上4个因素，计算为 LULC 或生境类型栅格单元的总威胁水平（total threat level），公式为，式中，索引威胁地图上的所有栅格单元。威胁权重为归一化的权重，通过方式计算。通过上述公式计算生境威胁水平得分后，可以由半饱和函数（half saturation function）转化为生境质量分值，其公式为：，式中，和为缩放参数（或常数），可参考配置，。当生境适宜性时，生境质量为0。生境质量随的增加而增加，随的增加而减小，且值永不会大于1。

下述代码打印了半饱和函数变化曲线，即当值增加时，值的变化。

z=2.5  
k=0.05  
fig, ax=plt.subplots(1, 1,figsize=(5,5))  
   
X=np.arange(0, 1+0.01, 0.01)  
ax.plot(X,1\*(1-X\*\*z/(X\*\*z+k\*\*z)))  
  
usda\_vis.plot\_style\_axis\_A(ax)  
plt.xlabel('$D\_{xj}$', fontsize=15)  
plt.ylabel('$Q\_{xj}$', fontsize=15)  
plt.show()



#### 生境稀有性（Habitat Rarity）

生境稀有性的计算公式为，式中，为当前 LULC 的第个类别的面积；为过去（基准，baseline）LULC 第个类别的面积。当值越趋近于1，说明第类生境（LULC）相对于基准生境的面积有大幅度减少，那么对于该类生境的保护可能需要增加重视。通过计算所有生境类型的后，可以获得总的生境稀有性，公式为，式中，如果栅格单元落入第类生境内，则，并入和计算，否则为，不计入。

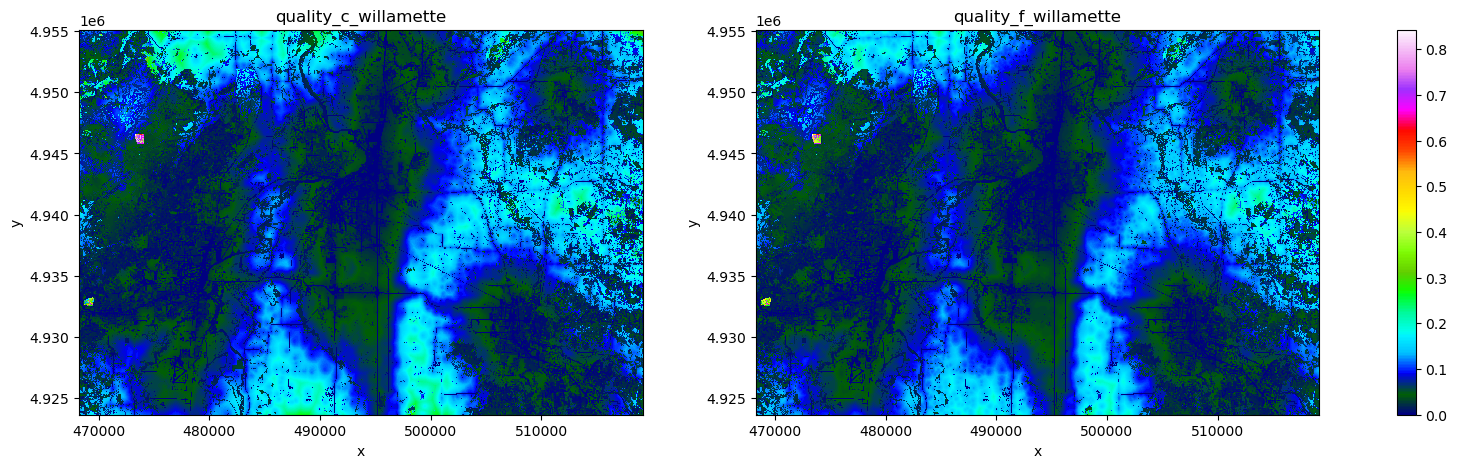
迁移InVEST 生境质量计算模型至USDA库，计算结果如下。

habitat\_quality=usda\_esv.Habitat\_quality(args\_habitat\_quality)  
habitat\_quality.habitat\_quality\_results\_info['paths\_dict']

The threat raster for crops could not be found for the land cover \_b. Skipping Habitat Quality calculation for this land cover.  
  
  
  
  
  
{'degsum': ['I:\\ESVs\\habitat\_quality\\deg\_sum\_c\_willamette.tif',  
 'I:\\ESVs\\habitat\_quality\\deg\_sum\_f\_willamette.tif'],  
 'quality': ['I:\\ESVs\\habitat\_quality\\quality\_c\_willamette.tif',  
 'I:\\ESVs\\habitat\_quality\\quality\_f\_willamette.tif'],  
 'rarily': ['I:\\ESVs\\habitat\_quality\\rarity\_c\_willamette.tif',  
 'I:\\ESVs\\habitat\_quality\\rarity\_f\_willamette.tif']}

读取和打印查看当前和未来景观的生境质量。

quality\_c\_willamette=rxr.open\_rasterio(r'I:\ESVs\habitat\_quality\quality\_c\_willamette.tif',masked=True).squeeze()  
quality\_f\_willamette=rxr.open\_rasterio(r'I:\ESVs\habitat\_quality\quality\_f\_willamette.tif',masked=True).squeeze()  
  
f, axes=plt.subplots(1,2,figsize=(20,5))  
cmap='gist\_ncar'  
quality\_c\_willamette.plot.imshow(ax=axes[0],add\_colorbar=False,cmap=cmap)  
ax=quality\_f\_willamette.plot.imshow(ax=axes[1],add\_colorbar=False,cmap=cmap)  
  
axes[0].set\_title('quality\_c\_willamette')  
axes[1].set\_title('quality\_f\_willamette')  
  
cax,kw = mpl.colorbar.make\_axes([ax for ax in axes.flat])  
plt.colorbar(ax, cax=cax, \*\*kw)  
plt.show()



### 3.5.3.2 作物授粉（授粉者丰度）（Crop Pollination (Pollinator Abundance)）[31]和作物生产（Crop Production）[32]

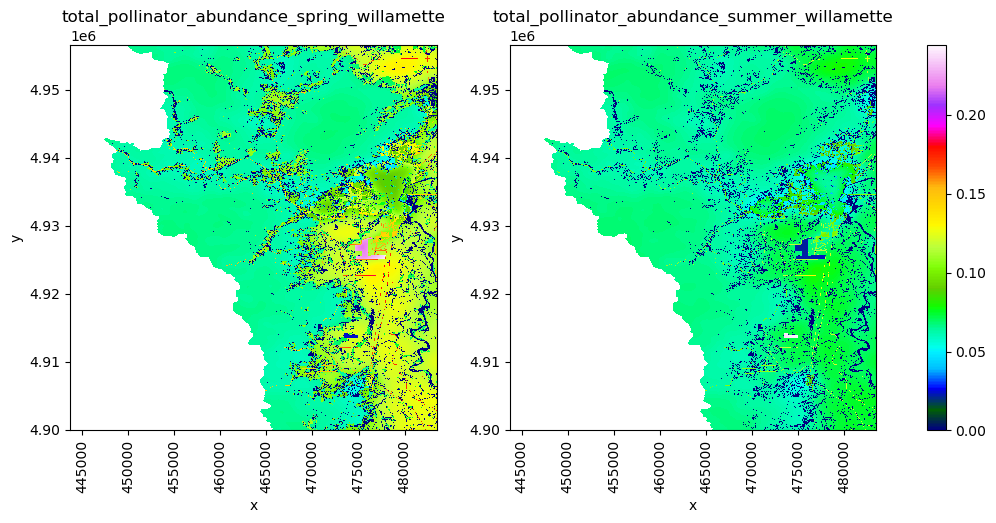
#### 1）作物授粉

对作物授粉的解释，可以查看 InVEST 用户指南，这里同样将该模型迁移至USDA库，调用计算如下。

args\_crop\_pollination={  
 'workspace\_dir':r'I:\ESVs\pollination',  
 'results\_suffix':'willamette',  
 'n\_workers':6,   
 'landcover\_raster\_path':r'I:\data\invest\pollination\landcover.tif',  
 'guild\_table\_path':r'I:\data\invest\pollination\guild\_table.csv',  
 'landcover\_biophysical\_table\_path':r'I:\data\invest\pollination\landcover\_biophysical\_table.csv',  
 'farm\_vector\_path':r'I:\data\invest\pollination\farms.shp',  
 }  
  
crop\_pollination=usda\_esv.Crop\_pollination(args\_crop\_pollination)

读取打印地图查看春夏季节每个栅格所有物种的总授粉者丰度。

total\_pollinator\_abundance\_spring\_willamette=rxr.open\_rasterio(r'I:\ESVs\pollination\total\_pollinator\_abundance\_spring\_willamette.tif',masked=True).squeeze()  
total\_pollinator\_abundance\_summer\_willamette=rxr.open\_rasterio(r'I:\ESVs\pollination\total\_pollinator\_abundance\_summer\_willamette.tif',masked=True).squeeze()  
  
f, axes=plt.subplots(1,2,figsize=(13,5))  
cmap='gist\_ncar'  
total\_pollinator\_abundance\_spring\_willamette.plot.imshow(ax=axes[0],add\_colorbar=False,cmap=cmap)  
ax=total\_pollinator\_abundance\_summer\_willamette.plot.imshow(ax=axes[1],add\_colorbar=False,cmap=cmap)  
  
axes[0].set\_title('total\_pollinator\_abundance\_spring\_willamette')  
axes[1].set\_title('total\_pollinator\_abundance\_summer\_willamette')  
  
cax,kw = mpl.colorbar.make\_axes([ax for ax in axes.flat])  
plt.colorbar(ax, cax=cax, \*\*kw)  
  
for ax in axes: ax.tick\_params(axis='x', labelrotation=90)  
plt.show()



#### 2）作物生产

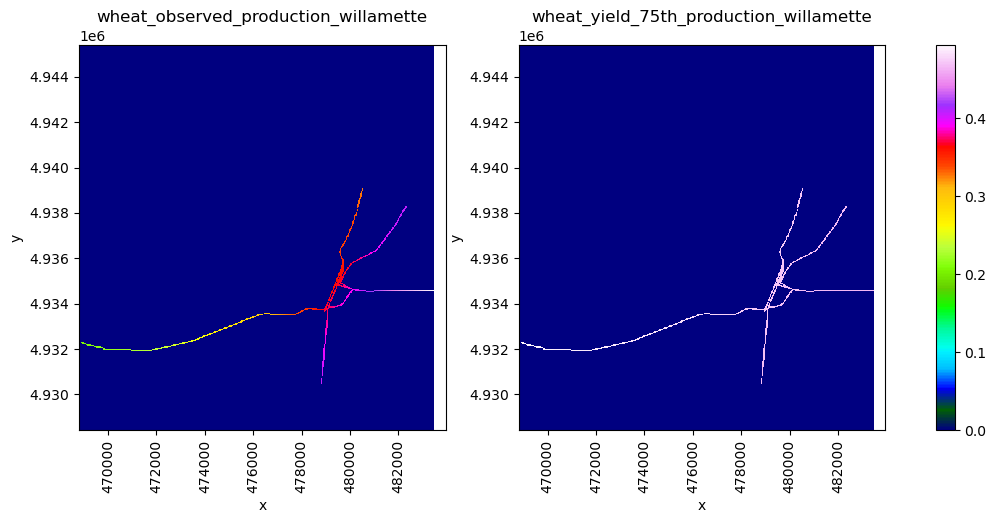
InVEST提供了两个作物生产模型，分别为百分位模型（Percentile Model）和回归模型（Regression Model），分别迁移至USDA库，对应Crop\_production\_percentile和Crop\_production\_regression类。分别使用两种方法计算。

* 百分位模型计算

args\_crop\_production\_percentile={  
 'workspace\_dir':r'I:\ESVs\crop\_PP',  
 'n\_workers':6,  
 'results\_suffix':'willamette',  
 'landcover\_raster\_path':r'I:\data\invest\CropProduction\sample\_user\_data\landcover.tif',  
 'landcover\_to\_crop\_table\_path':r'I:\data\invest\CropProduction\sample\_user\_data\landcover\_to\_crop\_table.csv',  
 'aggregate\_polygon\_path':r'I:\data\invest\CropProduction\sample\_user\_data\aggregate\_shape.shp',  
 'model\_data\_path':r'I:\data\invest\CropProduction\model\_data',  
 }  
  
crop\_production\_percentile=usda\_esv.Crop\_production\_percentile(args\_crop\_production\_percentile)

读取打印地图查看小麦的观测和百分位数为75的生产率（production rate），单位为公吨/像素每年。

wheat\_observed\_production\_willamette=rxr.open\_rasterio(r'I:\ESVs\crop\_PP\wheat\_observed\_production\_willamette.tif',masked=True).squeeze()  
wheat\_yield\_75th\_production\_willamette=rxr.open\_rasterio(r'I:\ESVs\crop\_PP\wheat\_yield\_75th\_production\_willamette.tif',masked=True).squeeze()  
  
f, axes=plt.subplots(1,2,figsize=(13,5))  
cmap='gist\_ncar'  
wheat\_observed\_production\_willamette.plot.imshow(ax=axes[0],add\_colorbar=False,cmap=cmap)  
ax=wheat\_yield\_75th\_production\_willamette.plot.imshow(ax=axes[1],add\_colorbar=False,cmap=cmap)  
  
axes[0].set\_title('wheat\_observed\_production\_willamette')  
axes[1].set\_title('wheat\_yield\_75th\_production\_willamette')  
  
cax,kw = mpl.colorbar.make\_axes([ax for ax in axes.flat])  
plt.colorbar(ax, cax=cax, \*\*kw)  
  
for ax in axes: ax.tick\_params(axis='x', labelrotation=90)  
plt.show()

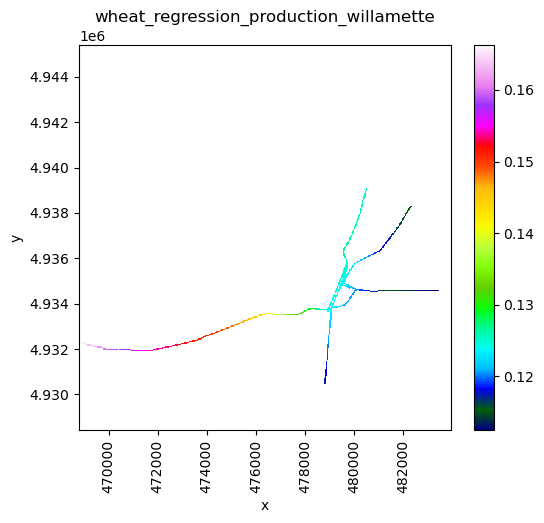


* 回归模型计算

args\_crop\_production\_regression={  
 'workspace\_dir':r'I:\ESVs\crop\_PR',  
 'n\_workers':6,  
 'results\_suffix':'willamette',  
 'landcover\_raster\_path':r'I:\data\invest\CropProduction\sample\_user\_data\landcover.tif',  
 'landcover\_to\_crop\_table\_path':r'I:\data\invest\CropProduction\sample\_user\_data\landcover\_to\_crop\_table.csv',  
 'fertilization\_rate\_table\_path':r'I:\data\invest\CropProduction\sample\_user\_data\crop\_fertilization\_rates.csv',  
 'aggregate\_polygon\_path':r'I:\data\invest\CropProduction\sample\_user\_data\aggregate\_shape.shp',  
 'model\_data\_path':r'I:\data\invest\CropProduction\model\_data',  
 }  
  
crop\_production\_regression=usda\_esv.Crop\_production\_regression(args\_crop\_production\_regression)

读取打印地图查看小麦的生产率。

wheat\_regression\_production\_willamette=rxr.open\_rasterio(r'I:\ESVs\crop\_PR\wheat\_regression\_production\_willamette.tif',masked=True).squeeze()  
  
f, ax=plt.subplots(1,1,figsize=(6,5))  
cmap='gist\_ncar'  
wheat\_regression\_production\_willamette.plot.imshow(ax=ax,add\_colorbar=True,cmap=cmap)  
ax.set\_title('wheat\_regression\_production\_willamette')  
  
ax.tick\_params(axis='x', labelrotation=90)  
plt.show()



## 3.5.4 生态系统服务价值的贝叶斯网络构建与推理

前文通过 InVEST 计算了生态系统服务价值中的碳储存和封存、生境质量、作物授粉和作物生产等4类服务，并获得了相应的计算结果。接下来希望能够分析这4类服务变量之间的相互关系，建立可以推断与权衡的模型。用[pgmpy库](https://github.com/pgmpy/pgmpy)㉙的BayesianNetwork方法构建离散型贝叶斯网络（Bayesian Network，BN）；根据前文计算的数据，对网络模型随机变量节点进行参数估计获得各个随机变量的条件概率分布（Conditional probability distribution，CPD）；进而应用建立的 BN 模型进行推理，给定某一随机变量的证据（evidence）估计其它随机变量的状态概率，例如如果假设一个栅格单元的碳储量很高，同时假设该栅格单元也具有很高的生境质量，那么该栅格单元为各个 LULC 类型的概率为多少？同时可以进行条件独立性和有效迹等分析。

### 3.5.4.1 数据准备——构建数据集

就生态系统服务价值的计算，并没有选择全部变量，仅选择了7个用以说明贝叶斯网络模型构建、参数估计和推理的方法。7个随机变量分别为当前的 LULC（lulc\_cur）、未来的 LULC (lulc\_fut)、碳储存量（tot\_c\_cur\_willamette）、碳封存的经济价值（npv\_fut\_willamette）、生境威胁建成区（Urba）、当前生境质量（quality\_c\_willamette）、未来生境质量（quality\_f\_willamette）、春季所有物种的总授粉者丰度（total\_pollinator\_abundance\_spring\_willamette）和小麦的生产率（wheat\_regression\_production\_willamette）。

%load\_ext autoreload   
%autoreload 2   
import warnings  
warnings.filterwarnings('ignore')  
  
import usda.geodata\_process as usda\_geoprocess  
import usda.pgm as usda\_pgm  
  
import rioxarray as rxr  
from matplotlib\_scalebar.scalebar import ScaleBar  
import matplotlib.pyplot as plt  
import matplotlib  
import numpy as np  
import pandas as pd  
import mapclassify  
from pgmpy.models import BayesianNetwork  
from pgmpy.estimators import BayesianEstimator, MaximumLikelihoodEstimator  
from pgmpy.inference import VariableElimination

data\_fn\_dict={'lulc\_cur':r'I:\data\invest\Carbon\lulc\_current\_willamette.tif',  
 'lulc\_fut':r'I:\data\invest\Carbon\lulc\_future\_willamette.tif',  
 'tot\_c\_cur\_willamette':r'I:\\ESVs\\carbon\\tot\_c\_cur\_willamette.tif',  
 'npv\_fut\_willamette':r'I:\\ESVs\\carbon\\npv\_fut\_willamette.tif',  
 'Urban':r'I:\data\invest\HabitatQuality\urban\_c.tif',  
 'quality\_c\_willamette':r'I:\ESVs\habitat\_quality\quality\_c\_willamette.tif',  
 'quality\_f\_willamette':r'I:\ESVs\habitat\_quality\quality\_f\_willamette.tif',  
 'total\_pollinator\_abundance\_spring\_willamette':r'I:\ESVs\pollination\total\_pollinator\_abundance\_spring\_willamette.tif',  
 'wheat\_regression\_production\_willamette':r'I:\ESVs\crop\_PR\wheat\_regression\_production\_willamette.tif'}

4类生态系统服务价值计算虽然都为威拉米特（willamette）区域，但是分析范围存在差异，因此并未采用栅格波段堆叠的方式合并这7个单独的栅格数据，而是采用了最大重叠区域内采样点提取各个栅格值的方法。首先需要计算所有栅格的最大重叠范围，定义rasters\_minimum\_bound()方法实现，基本逻辑就是获得各个栅格的边界坐标值（[left, bottom, right, top]），提取最大或最小值后建立叠合区域的矩形对象。

minimum\_bound=usda\_geoprocess.rasters\_minimum\_bound(list(data\_fn\_dict.values()))   
bound\_gdf=minimum\_bound['bound\_gdf']  
bound\_gdf

crs:  
{0: CRS.from\_epsg(26910), 1: CRS.from\_epsg(26910), 2: CRS.from\_epsg(26910), 3: CRS.from\_epsg(26910), 4: CRS.from\_epsg(26910), 5: CRS.from\_epsg(26910), 6: CRS.from\_epsg(26910), 7: CRS.from\_epsg(26910), 8: CRS.from\_epsg(26910)}

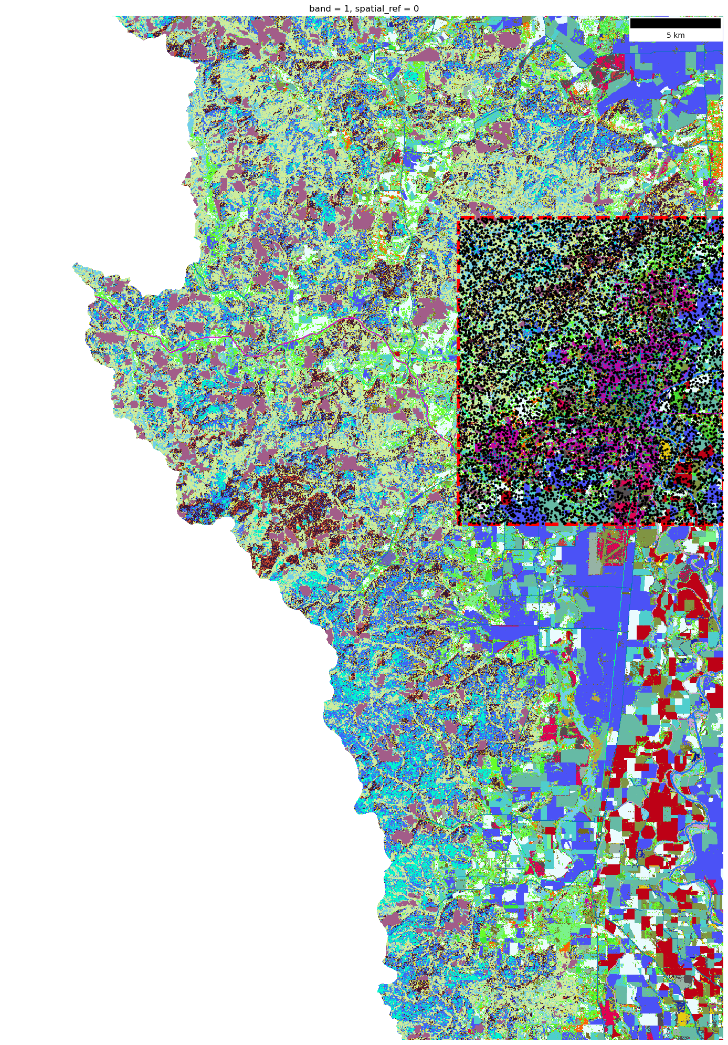
|  | **geometry** |
| --- | --- |
| **0** | POLYGON ((468803.127 4928406.906, 468803.127 4... |

定义random\_pts\_in\_geoBounds()方法获得随机采样点。实验中采样点数据配置为10,000个。

pts\_gdf=usda\_geoprocess.random\_pts\_in\_geoBounds(bound\_gdf,10\_000)

叠合打印随意一个 LULC 数据和最大重叠区域及随机采样点，核实计算结果是否正确。重叠区和采样点位于参考 LULC 栅格数据内，表明结果可以使用。

lulc\_cur\_path=rxr.open\_rasterio(data\_fn\_dict['lulc\_cur'],masked=True).squeeze()  
  
f, ax=plt.subplots(1,1,figsize=(20,20))  
  
np.random.seed(10)  
cmap=matplotlib.colors.ListedColormap (np.random.rand ( 256,3))  
lulc\_cur\_path.plot.imshow(ax=ax,add\_colorbar=False,cmap=cmap)  
bound\_gdf.plot(color='none',edgecolor='r',linewidth=5,ax=ax,linestyle='--')  
pts\_gdf.plot(ax=ax,markersize=5,color='k')  
  
ax.add\_artist(ScaleBar(1))   
ax.set\_axis\_off()  
plt.tight\_layout()  
plt.show()



定义extract\_raster\_vals\_at\_pts\_batch()方法用采样点采样所有栅格值，并对应 LULC 整数型索引增加实际 LULC 分类名称列lulc\_cur\_name和lulc\_fut\_name。

sample\_pt\_coords=[(p.x,p.y) for p in pts\_gdf.points.values]  
sample\_vals\_array,idx\_lst=usda\_geoprocess.extract\_raster\_vals\_at\_pts\_batch(sample\_pt\_coords,list(data\_fn\_dict.values()))

0%| | 0/9 [00:00<?, ?it/s]  
  
  
  
0it [00:00, ?it/s]

carbon\_pools\_willamette=pd.read\_csv(r'I:\data\invest\Carbon\carbon\_pools\_willamette.csv')  
LULC\_mapping=dict(carbon\_pools\_willamette[['lucode','LULC\_Name']].values)  
  
sample\_vals\_df=pd.DataFrame(sample\_vals\_array,columns=data\_fn\_dict.keys())  
sample\_vals\_df[['lulc\_cur\_name','lulc\_fut\_name']]=sample\_vals\_df.apply(lambda row:pd.Series([LULC\_mapping[int(row.lulc\_cur)],LULC\_mapping[int(row.lulc\_fut)]]),axis=1)  
sample\_vals\_unique\_df=sample\_vals\_df.drop\_duplicates()  
sample\_vals\_unique\_df

|  | **lulc\_cur** | **lulc\_fut** | **tot\_c\_cur\_willamette** | **npv\_fut\_willamette** | **Urban** | **quality\_c\_willamette** | **quality\_f\_willamette** | **total\_pollinator\_abundance\_spring\_willamette** | **wheat\_regression\_production\_willamette** | **lulc\_cur\_name** | **lulc\_fut\_name** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 6.0 | 6.0 | 0.000000 | 0.000000 | 1.0 | 0.000000 | 0.000000 | 0.088110 | -1.0 | Commercial | Commercial |
| **1** | 53.0 | 56.0 | 42.750000 | -2685.426514 | 0.0 | 0.014660 | 0.013035 | 0.055686 | -1.0 | Forest Closed hardwood | Conifers 0-20 yrs |
| **2** | 83.0 | 1.0 | 2.970000 | 375.039856 | 0.0 | 0.032672 | 0.032672 | 0.084252 | -1.0 | Hayfield | Residential 0-4 units/acre |
| **3** | 67.0 | 67.0 | 1.080000 | 0.000000 | 0.0 | 0.093392 | 0.079692 | 0.106351 | -1.0 | Grass seed rotation | Grass seed rotation |
| **4** | 1.0 | 1.0 | 7.740000 | 0.000000 | 0.0 | 0.015323 | 0.015414 | 0.117574 | -1.0 | Residential 0-4 units/acre | Residential 0-4 units/acre |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **9995** | 84.0 | 84.0 | 2.070000 | 0.000000 | 0.0 | 0.077150 | 0.073046 | 0.105832 | -1.0 | Late field crop | Late field crop |
| **9996** | 57.0 | 58.0 | 24.480000 | 1188.805420 | 0.0 | 0.020732 | 0.020690 | 0.055342 | -1.0 | Forest closed conifer 21-40 yrs | Forest closed conifer 41-60 yrs |
| **9997** | 71.0 | 1.0 | 1.350000 | 502.411865 | 0.0 | 0.038543 | 0.037324 | 0.101100 | -1.0 | Grains | Residential 0-4 units/acre |
| **9998** | 58.0 | 59.0 | 39.599998 | 877.451660 | 0.0 | 0.012720 | 0.002704 | 0.041639 | -1.0 | Forest closed conifer 41-60 yrs | Forest closed conifer 61-80 yrs |
| **9999** | 82.0 | 1.0 | 1.260000 | 509.488068 | 0.0 | 0.038406 | 0.033619 | 0.086102 | -1.0 | Field crop | Residential 0-4 units/acre |

9828 rows × 11 columns

离散化（Data discretization）连续型数据时，对不同的变量，根据其具体数值分布情况，做了不同的离散化处理，例如碳储存分类等区间分类为5级，碳封存的经济价值分类含按值0的划分，建成区分类为二分类等，具体如下代码所示。

sample\_vals\_discretization\_dict={}  
discretization\_classifiers={}  
  
# A-tot\_c\_cur\_willamette 当前 LULC 碳储存分类  
tot\_c\_cur\_disc=mapclassify.EqualInterval(sample\_vals\_unique\_df['tot\_c\_cur\_willamette'],k=5)  
discretization\_classifiers['totc']=tot\_c\_cur\_disc  
sample\_vals\_discretization\_dict['totc']=tot\_c\_cur\_disc.yb  
  
# B-npv\_fut\_willamette 碳封存的经济价值分类  
npv\_fut\_disc=mapclassify.UserDefined(sample\_vals\_unique\_df['npv\_fut\_willamette'],bins=[0,1000])  
discretization\_classifiers['npvfut']=npv\_fut\_disc  
sample\_vals\_discretization\_dict['npvfut']=npv\_fut\_disc.yb  
  
# C-Urban 生境威胁-建成区分类  
discretization\_classifiers['Urban']={'0':0,'1':1}  
sample\_vals\_discretization\_dict['Urban']=np.array(sample\_vals\_unique\_df['Urban']).astype(int)  
  
# D-quality\_c\_willamette 当前 LULC 生境质量分类  
quality\_c\_disc=mapclassify.UserDefined(sample\_vals\_unique\_df['quality\_c\_willamette'],bins=[0.05,0.1])  
discretization\_classifiers['qualityc']=quality\_c\_disc  
sample\_vals\_discretization\_dict['qualityc']=quality\_c\_disc.yb  
  
# E-quality\_f\_willamette 未来 LULC 生境质量分类  
quality\_f\_disc=mapclassify.UserDefined(sample\_vals\_unique\_df['quality\_f\_willamette'],bins=[0.05,0.1])  
discretization\_classifiers['qualityf']=quality\_f\_disc  
sample\_vals\_discretization\_dict['qualityf']=quality\_f\_disc.yb  
  
# F-total\_pollinator\_abundance\_spring\_willamette 春季授粉者总丰度分类  
total\_pollinator\_abundance\_spring\_disc=mapclassify.EqualInterval(sample\_vals\_unique\_df['total\_pollinator\_abundance\_spring\_willamette'],k=3)  
discretization\_classifiers['totalpollinator']=total\_pollinator\_abundance\_spring\_disc  
sample\_vals\_discretization\_dict['totalpollinator']=total\_pollinator\_abundance\_spring\_disc.yb  
  
# G-wheat\_regression\_production\_willamette 小麦生产率分类  
wheat\_regression\_production\_disc=mapclassify.UserDefined(sample\_vals\_unique\_df['wheat\_regression\_production\_willamette'],bins=[0,0.12])  
discretization\_classifiers['wheatproduction']=wheat\_regression\_production\_disc  
sample\_vals\_discretization\_dict['wheatproduction']=wheat\_regression\_production\_disc.yb  
  
sample\_vals\_discretization\_df=pd.DataFrame(data=sample\_vals\_discretization\_dict)  
sample\_vals\_discretization\_df[['lulccur','lulcfut']]=sample\_vals\_df[['lulc\_cur\_name','lulc\_fut\_name']]  
sample\_vals\_discretization\_df

|  | **totc** | **npvfut** | **Urban** | **qualityc** | **qualityf** | **totalpollinator** | **wheatproduction** | **lulccur** | **lulcfut** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 0 | 0 | 1 | 0 | 0 | 1 | 0 | Commercial | Commercial |
| **1** | 2 | 0 | 0 | 0 | 0 | 1 | 0 | Forest Closed hardwood | Conifers 0-20 yrs |
| **2** | 0 | 1 | 0 | 0 | 0 | 1 | 0 | Hayfield | Residential 0-4 units/acre |
| **3** | 0 | 0 | 0 | 1 | 1 | 2 | 0 | Grass seed rotation | Grass seed rotation |
| **4** | 0 | 0 | 0 | 0 | 0 | 2 | 0 | Residential 0-4 units/acre | Residential 0-4 units/acre |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **9823** | 0 | 0 | 0 | 1 | 1 | 2 | 0 | Pasture | Pasture |
| **9824** | 1 | 2 | 0 | 0 | 0 | 1 | 0 | Grass seed rotation | Grass seed rotation |
| **9825** | 0 | 1 | 0 | 0 | 0 | 1 | 0 | Forest closed conifer 81-200 yrs | Conifers 0-20 yrs |
| **9826** | 2 | 1 | 0 | 0 | 0 | 0 | 0 | Residential 0-4 units/acre | Residential 0-4 units/acre |
| **9827** | 0 | 1 | 0 | 0 | 0 | 1 | 0 | Forest Closed mixed | Forest Closed mixed |

9828 rows × 9 columns

离散化的区间划分结果，用于后续推理时查阅。

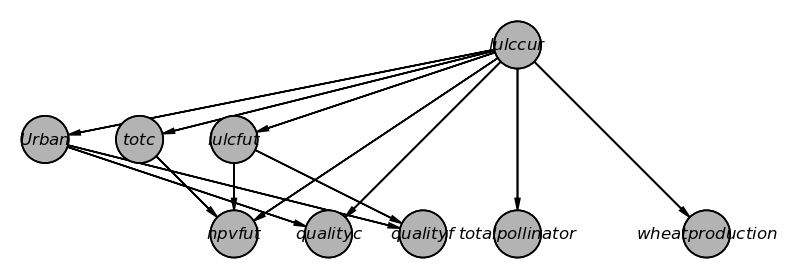
discretization\_classifiers

{'totc': EqualInterval  
   
 Interval Count  
 ----------------------  
 [ 0.00, 15.75] | 5976  
 (15.75, 31.50] | 404  
 (31.50, 47.25] | 2654  
 (47.25, 63.00] | 254  
 (63.00, 78.75] | 540,  
 'npvfut': UserDefined  
   
 Interval Count  
 ----------------------------  
 [-4419.10, 0.00] | 7937  
 ( 0.00, 1000.00] | 1342  
 ( 1000.00, 3156.00] | 549,  
 'Urban': {'0': 0, '1': 1},  
 'qualityc': UserDefined  
   
 Interval Count  
 --------------------  
 [0.00, 0.05] | 7906  
 (0.05, 0.10] | 1825  
 (0.10, 0.74] | 97,  
 'qualityf': UserDefined  
   
 Interval Count  
 --------------------  
 [0.00, 0.05] | 8106  
 (0.05, 0.10] | 1646  
 (0.10, 0.73] | 76,  
 'totalpollinator': EqualInterval  
   
 Interval Count  
 --------------------  
 [0.00, 0.05] | 1511  
 (0.05, 0.10] | 6842  
 (0.10, 0.16] | 1475,  
 'wheatproduction': UserDefined  
   
 Interval Count  
 ----------------------  
 [-1.00, 0.00] | 9746  
 ( 0.00, 0.12] | 15  
 ( 0.12, 0.16] | 67}

### 3.5.4.2 构建贝叶斯网络并参数估计

根据4类生态系统服务价值计算的关系建立贝叶斯网络图，确定随机变量节点和表征随机变量之间因果关系的边。

model\_ess=BayesianNetwork([  
 ('lulccur','totc'),  
 ('lulccur','npvfut'),  
 ('lulccur','qualityc'),  
 ('lulccur','totalpollinator'),  
 ('lulccur','wheatproduction'),  
 ('lulccur','Urban'),  
 ('lulccur','lulcfut'),  
 ('lulcfut','npvfut'),  
 ('lulcfut','qualityf'),  
 ('Urban','qualityf'),  
 ('Urban','qualityc'),  
 ('totc','npvfut'),  
 ])  
  
model\_ess\_daft=model\_ess.to\_daft(node\_pos={'lulccur':(6,3),'lulcfut':(3,2),'Urban':(1,2),'totc':(2,2),'npvfut':(3,1),'qualityc':(4,1),'qualityf':(5,1),'totalpollinator':(6,1),'wheatproduction':(8,1)})  
model\_ess\_daft.render(dpi=120);



参数估计采用了贝叶斯参数估计的方法。

model\_ess.fit(data=sample\_vals\_discretization\_df,   
 estimator=BayesianEstimator,  
 prior\_type="BDeu",  
 equivalent\_sample\_size=10,  
 complete\_samples\_only=False)

打印参数，即各个随机变量的 CPD（或称为随机概率表，conditional probability tables，CPT）。

print(f'Check model: {model\_ess.check\_model()}\n')  
for cpd in model\_ess.get\_cpds():  
 print(f'CPT of {cpd.variable}:')  
 print(cpd, '\n')

Check model: True  
  
CPT of lulccur:  
+---------------------------------------------------+-------------+  
| lulccur(Bare/fallow) | 0.00581268 |  
+---------------------------------------------------+-------------+  
| lulccur(Berries & Vineyards) | 0.00682915 |  
+---------------------------------------------------+-------------+  
| lulccur(Burned grass) | 0.00012047 |  
+---------------------------------------------------+-------------+  
| lulccur(Christmas trees) | 0.0084555 |  
+---------------------------------------------------+-------------+  
| lulccur(Commercial) | 0.0101835 |  
+---------------------------------------------------+-------------+  
| lulccur(Commercial/Industrial) | 0.0023567 |  
+---------------------------------------------------+-------------+  
| lulccur(Conifer Woodlot) | 0.000628704 |  
+---------------------------------------------------+-------------+  
| lulccur(Conifers 0-20 yrs) | 0.0109967 |  
+---------------------------------------------------+-------------+  
| lulccur(Double cropping) | 0.000831997 |  
+---------------------------------------------------+-------------+  
| lulccur(Field crop) | 0.0290898 |  
+---------------------------------------------------+-------------+  
| lulccur(Flooded/marsh) | 0.00530445 |  
+---------------------------------------------------+-------------+  
| lulccur(Forest Closed hardwood) | 0.0759489 |  
+---------------------------------------------------+-------------+  
| lulccur(Forest Closed mixed) | 0.146492 |  
+---------------------------------------------------+-------------+  
| lulccur(Forest closed conifer 21-40 yrs) | 0.0131312 |  
+---------------------------------------------------+-------------+  
| lulccur(Forest closed conifer 41-60 yrs) | 0.0444384 |  
+---------------------------------------------------+-------------+  
| lulccur(Forest closed conifer 61-80 yrs) | 0.0256338 |  
+---------------------------------------------------+-------------+  
| lulccur(Forest closed conifer 81-200 yrs) | 0.046573 |  
+---------------------------------------------------+-------------+  
| lulccur(Forest closed conifer older than 200 yrs) | 0.00754068 |  
+---------------------------------------------------+-------------+  
| lulccur(Grains) | 0.0326474 |  
+---------------------------------------------------+-------------+  
| lulccur(Grass) | 0.0157741 |  
+---------------------------------------------------+-------------+  
| lulccur(Grass seed rotation) | 0.0661908 |  
+---------------------------------------------------+-------------+  
| lulccur(Hayfield) | 0.0350869 |  
+---------------------------------------------------+-------------+  
| lulccur(Hybrid poplar) | 0.00134023 |  
+---------------------------------------------------+-------------+  
| lulccur(Industrial) | 0.0067275 |  
+---------------------------------------------------+-------------+  
| lulccur(Irrigated annual crop rotation) | 0.0177053 |  
+---------------------------------------------------+-------------+  
| lulccur(Irrigated perennial) | 0.0109967 |  
+---------------------------------------------------+-------------+  
| lulccur(Late field crop) | 0.00255999 |  
+---------------------------------------------------+-------------+  
| lulccur(Light duty roads) | 0.0324441 |  
+---------------------------------------------------+-------------+  
| lulccur(Main channel non-vegetated) | 0.00144188 |  
+---------------------------------------------------+-------------+  
| lulccur(Mint) | 0.00042541 |  
+---------------------------------------------------+-------------+  
| lulccur(Natural grassland) | 0.0153675 |  
+---------------------------------------------------+-------------+  
| lulccur(Natural shrub) | 0.0686303 |  
+---------------------------------------------------+-------------+  
| lulccur(Orchard) | 0.00937032 |  
+---------------------------------------------------+-------------+  
| lulccur(Pasture) | 0.0595838 |  
+---------------------------------------------------+-------------+  
| lulccur(Permanent lentic water) | 0.00815056 |  
+---------------------------------------------------+-------------+  
| lulccur(Railroad) | 0.00418634 |  
+---------------------------------------------------+-------------+  
| lulccur(Residential 0-4 units/acre) | 0.0751357 |  
+---------------------------------------------------+-------------+  
| lulccur(Residential 4-9 units/acre) | 0.00530445 |  
+---------------------------------------------------+-------------+  
| lulccur(Residential 9-16 units/acre) | 0.00296658 |  
+---------------------------------------------------+-------------+  
| lulccur(Residential >16 units/acre) | 0.00042541 |  
+---------------------------------------------------+-------------+  
| lulccur(Row crop) | 0.00591433 |  
+---------------------------------------------------+-------------+  
| lulccur(Rural non-vegetated unknown) | 0.0100818 |  
+---------------------------------------------------+-------------+  
| lulccur(Rural structures) | 0.00611762 |  
+---------------------------------------------------+-------------+  
| lulccur(Secondary roads) | 0.00855714 |  
+---------------------------------------------------+-------------+  
| lulccur(Stream orders 5-7) | 0.00601598 |  
+---------------------------------------------------+-------------+  
| lulccur(Sugar beet seed) | 0.000222117 |  
+---------------------------------------------------+-------------+  
| lulccur(Topographic Shadow) | 0.000222117 |  
+---------------------------------------------------+-------------+  
| lulccur(Turfgrass) | 0.00713409 |  
+---------------------------------------------------+-------------+  
| lulccur(Upland Forest Semi-closed conifer) | 0.000527057 |  
+---------------------------------------------------+-------------+  
| lulccur(Upland Forest Semi-closed hardwood) | 0.00184846 |  
+---------------------------------------------------+-------------+  
| lulccur(Upland Forest Semi-closed mixed) | 0.00195011 |  
+---------------------------------------------------+-------------+  
| lulccur(Upland Forest open) | 0.00042541 |  
+---------------------------------------------------+-------------+  
| lulccur(Urban non-vegetated unknown) | 0.0263453 |  
+---------------------------------------------------+-------------+  
| lulccur(Urban tree overstory) | 0.0118098 |  
+---------------------------------------------------+-------------+   
  
CPT of totc:  
+---------+----------------------+-----+-------------------------------+  
| lulccur | lulccur(Bare/fallow) | ... | lulccur(Urban tree overstory) |  
+---------+----------------------+-----+-------------------------------+  
| totc(0) | 0.577720207253886 | ... | 0.5597704813516098 |  
+---------+----------------------+-----+-------------------------------+  
| totc(1) | 0.03562176165803109 | ... | 0.08638826904686005 |  
+---------+----------------------+-----+-------------------------------+  
| totc(2) | 0.280440414507772 | ... | 0.27574115396876 |  
+---------+----------------------+-----+-------------------------------+  
| totc(3) | 0.03562176165803109 | ... | 0.008925725215173732 |  
+---------+----------------------+-----+-------------------------------+  
| totc(4) | 0.0705958549222798 | ... | 0.06917437041759641 |  
+---------+----------------------+-----+-------------------------------+   
  
CPT of npvfut:  
+-----------+------------------------+-----+-------------------------------+  
| lulccur | lulccur(Bare/fallow) | ... | lulccur(Urban tree overstory) |  
+-----------+------------------------+-----+-------------------------------+  
| lulcfut | lulcfut(Bare/fallow) | ... | lulcfut(Urban tree overstory) |  
+-----------+------------------------+-----+-------------------------------+  
| totc | totc(0) | ... | totc(4) |  
+-----------+------------------------+-----+-------------------------------+  
| npvfut(0) | 0.8420850240008487 | ... | 0.9999370177924738 |  
+-----------+------------------------+-----+-------------------------------+  
| npvfut(1) | 0.15790171586177634 | ... | 3.1491103763186906e-05 |  
+-----------+------------------------+-----+-------------------------------+  
| npvfut(2) | 1.3260137375023207e-05 | ... | 3.1491103763186906e-05 |  
+-----------+------------------------+-----+-------------------------------+   
  
CPT of qualityc:  
+-------------+----------------------+-----+-------------------------------+  
| Urban | Urban(0) | ... | Urban(1) |  
+-------------+----------------------+-----+-------------------------------+  
| lulccur | lulccur(Bare/fallow) | ... | lulccur(Urban tree overstory) |  
+-------------+----------------------+-----+-------------------------------+  
| qualityc(0) | 0.7585301837270341 | ... | 0.9932111337406654 |  
+-------------+----------------------+-----+-------------------------------+  
| qualityc(1) | 0.2224124158393244 | ... | 0.0033944331296673455 |  
+-------------+----------------------+-----+-------------------------------+  
| qualityc(2) | 0.01905740043364144 | ... | 0.0033944331296673455 |  
+-------------+----------------------+-----+-------------------------------+   
  
CPT of totalpollinator:  
+--------------------+----------------------+-----+-------------------------------+  
| lulccur | lulccur(Bare/fallow) | ... | lulccur(Urban tree overstory) |  
+--------------------+----------------------+-----+-------------------------------+  
| totalpollinator(0) | 0.12348877374784112 | ... | 0.15545638083094251 |  
+--------------------+----------------------+-----+-------------------------------+  
| totalpollinator(1) | 0.718048359240069 | ... | 0.689087238338115 |  
+--------------------+----------------------+-----+-------------------------------+  
| totalpollinator(2) | 0.1584628670120898 | ... | 0.15545638083094251 |  
+--------------------+----------------------+-----+-------------------------------+   
  
CPT of wheatproduction:  
+--------------------+----------------------+-----+-------------------------------+  
| lulccur | lulccur(Bare/fallow) | ... | lulccur(Urban tree overstory) |  
+--------------------+----------------------+-----+-------------------------------+  
| wheatproduction(0) | 0.9978411053540588 | ... | 0.9903304643502283 |  
+--------------------+----------------------+-----+-------------------------------+  
| wheatproduction(1) | 0.001079447322970639 | ... | 0.000531293167569865 |  
+--------------------+----------------------+-----+-------------------------------+  
| wheatproduction(2) | 0.001079447322970639 | ... | 0.009138242482201677 |  
+--------------------+----------------------+-----+-------------------------------+   
  
CPT of Urban:  
+----------+----------------------+-----+-------------------------------+  
| lulccur | lulccur(Bare/fallow) | ... | lulccur(Urban tree overstory) |  
+----------+----------------------+-----+-------------------------------+  
| Urban(0) | 0.9459196891191709 | ... | 0.9217405164169589 |  
+----------+----------------------+-----+-------------------------------+  
| Urban(1) | 0.05408031088082901 | ... | 0.07825948358304112 |  
+----------+----------------------+-----+-------------------------------+   
  
CPT of lulcfut:  
+---------------------------------------------------+-----+-------------------------------+  
| lulccur | ... | lulccur(Urban tree overstory) |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Bare/fallow) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Berries & Vineyards) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Christmas trees) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Commercial) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Commercial/Industrial) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Conifer Woodlot) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Conifers 0-20 yrs) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Double cropping) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Field crop) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Flooded/marsh) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Forest Closed hardwood) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Forest Closed mixed) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Forest closed conifer 21-40 yrs) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Forest closed conifer 41-60 yrs) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Forest closed conifer 61-80 yrs) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Forest closed conifer 81-200 yrs) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Forest closed conifer older than 200 yrs) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Grains) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Grass) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Grass seed rotation) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Hayfield) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Industrial) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Irrigated annual crop rotation) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Irrigated perennial) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Late field crop) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Light duty roads) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Main channel non-vegetated) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Mint) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Natural grassland) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Natural shrub) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Orchard) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Pasture) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Permanent lentic water) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Railroad) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Residential 0-4 units/acre) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Residential 4-9 units/acre) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Residential 9-16 units/acre) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Residential >16 units/acre) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Row crop) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Rural non-vegetated unknown) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Rural structures) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Secondary roads) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Stream orders 5-7) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Sugar beet seed) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Topographic Shadow) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Turfgrass) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Upland Forest Semi-closed conifer) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Urban non-vegetated unknown) | ... | 3.252815311652235e-05 |  
+---------------------------------------------------+-----+-------------------------------+  
| lulcfut(Urban tree overstory) | ... | 0.9984386486504069 |  
+---------------------------------------------------+-----+-------------------------------+   
  
CPT of qualityf:  
+-------------+-----------------------+-----+-------------------------------+  
| Urban | Urban(0) | ... | Urban(1) |  
+-------------+-----------------------+-----+-------------------------------+  
| lulcfut | lulcfut(Bare/fallow) | ... | lulcfut(Urban tree overstory) |  
+-------------+-----------------------+-----+-------------------------------+  
| qualityf(0) | 0.6652554332486593 | ... | 0.9925261584454411 |  
+-------------+-----------------------+-----+-------------------------------+  
| qualityf(1) | 0.3333333333333333 | ... | 0.0037369207772795228 |  
+-------------+-----------------------+-----+-------------------------------+  
| qualityf(2) | 0.0014112334180073386 | ... | 0.0037369207772795228 |  
+-------------+-----------------------+-----+-------------------------------+

### 3.5.4.3 网络分析和推理

* 独立性分析

对于该 BN 模型，所有可能给出的证据（evidence），可以做出1110个有效的独立性断言（valid independence assertions）。使用local\_independencies方法返回随机变量的局部独立性实例，为网络中任何节点独立于给定其父节点的非后代节点。例如，给定证据lulccur，随机变量lulcfut, wheatproduction, totc, npvfut, totalpollinator独立于随机变量Urban。

print(f'There can be made {len(model\_ess.get\_independencies().get\_assertions())}',  
 'valid independence assertions with respect to the all possible given evidence.')  
print('For instance, any node in the network is independent of its non-descendents given its parents (local semantics):\n',  
 f'\n{model\_ess.local\_independencies(list(model\_ess.nodes()))}\n')

There can be made 1110 valid independence assertions with respect to the all possible given evidence.  
For instance, any node in the network is independent of its non-descendents given its parents (local semantics):  
   
(totc ⟂ Urban, lulcfut, wheatproduction, qualityf, qualityc, totalpollinator | lulccur)  
(npvfut ⟂ Urban, wheatproduction, qualityf, qualityc, totalpollinator | lulcfut, lulccur, totc)  
(qualityc ⟂ lulcfut, wheatproduction, totc, qualityf, npvfut, totalpollinator | Urban, lulccur)  
(totalpollinator ⟂ Urban, lulcfut, wheatproduction, totc, qualityf, qualityc, npvfut | lulccur)  
(wheatproduction ⟂ Urban, lulcfut, totc, qualityf, npvfut, qualityc, totalpollinator | lulccur)  
(Urban ⟂ lulcfut, wheatproduction, totc, npvfut, totalpollinator | lulccur)  
(lulcfut ⟂ Urban, wheatproduction, totc, qualityc, totalpollinator | lulccur)  
(qualityf ⟂ wheatproduction, totc, npvfut, lulccur, qualityc, totalpollinator | Urban, lulcfut)

* 独立性断言（independence assertions）

判断给定的一个随机变量与另一个随机变量之间的独立性，可以给或不给定证据。例如，没有观测到任何证据时，随机变量totalpollinator和wheatproduction之间并不相互独立（False）；但给定证据lulccur时，两个随机变量之间相互独立（True）。

usda\_pgm.check\_assertion(model\_ess, independent=['qualityc'], from\_variables=['lulccur'], evidence=['lulcfur'])  
usda\_pgm.check\_assertion(model\_ess, independent=['npvfut'], from\_variables=['wheatproduction'], evidence=['lulccur'])  
  
usda\_pgm.check\_assertion(model\_ess, independent=['totalpollinator'], from\_variables=['wheatproduction'], evidence=[])  
usda\_pgm.check\_assertion(model\_ess, independent=['totalpollinator'], from\_variables=['wheatproduction'], evidence=['lulccur'])

(qualityc ⟂ lulccur | lulcfur): False  
(npvfut ⟂ wheatproduction | lulccur): True  
(totalpollinator ⟂ wheatproduction): False  
(totalpollinator ⟂ wheatproduction | lulccur): True

* 有效迹分析

检查一个随机变量是否对另一个随机变量有影响时，可以通过有效迹计算查看。例如，当没有观测到的证据时，wheatproduction与其它的所有变量之间均存在有效迹；但是，当观测到证据lulccur时，该变量和其它变量之间则无有效迹。

usda\_pgm.active\_trails\_of(model\_ess,query='wheatproduction', evidence=[])  
usda\_pgm.active\_trails\_of(model\_ess,query='wheatproduction', evidence=['lulccur'])

Active trails between 'wheatproduction' and {'totc', 'qualityf', 'qualityc', 'totalpollinator', 'Urban', 'lulcfut', 'lulccur', 'npvfut'} given no evidence.  
No active trails for 'wheatproduction' given the evidence {'lulccur'}.

* 推理

首先试验因果推理（预测），例如观测到证据 LULC 类型（lulccur）为商业用地（Commercial）时，碳封存的经济价值（npvfut）的状态更可能为状态 0，即值[4419.10, 0.00]区间，其状态概率为0.8373；当LULC类型为草地（Grass）时，可以发现状态为0的概率有所下降，而为状态 1 （区间(0.00, 1000.00]）的概率从0.097上升至0.2024，表明碳封存的经济价值可能有所增加。下述计算结果的其它推理分析同。

infer=VariableElimination(model\_ess)  
usda\_pgm.query\_report(infer, variables=['totc',], evidence={'lulccur':'Forest closed conifer 21-40 yrs',})  
usda\_pgm.query\_report(infer, variables=['npvfut'], evidence={'lulccur':'Commercial'})  
usda\_pgm.query\_report(infer, variables=['npvfut'], evidence={'lulccur':'Grass'})  
usda\_pgm.query\_report(infer, variables=['qualityc'], evidence={'Urban':0})  
usda\_pgm.query\_report(infer, variables=['qualityc'], evidence={'Urban':1})  
usda\_pgm.query\_report(infer, variables=['qualityc'], evidence={'totc':4})

+---------+-------------+  
| totc | phi(totc) |  
+=========+=============+  
| totc(0) | 0.5233 |  
+---------+-------------+  
| totc(1) | 0.1112 |  
+---------+-------------+  
| totc(2) | 0.2777 |  
+---------+-------------+  
| totc(3) | 0.0320 |  
+---------+-------------+  
| totc(4) | 0.0558 |  
+---------+-------------+  
--- Query executed in 0.0020 seconds ---  
  
+-----------+---------------+  
| npvfut | phi(npvfut) |  
+===========+===============+  
| npvfut(0) | 0.8373 |  
+-----------+---------------+  
| npvfut(1) | 0.0971 |  
+-----------+---------------+  
| npvfut(2) | 0.0655 |  
+-----------+---------------+  
--- Query executed in 0.0070 seconds ---  
  
+-----------+---------------+  
| npvfut | phi(npvfut) |  
+===========+===============+  
| npvfut(0) | 0.7411 |  
+-----------+---------------+  
| npvfut(1) | 0.2024 |  
+-----------+---------------+  
| npvfut(2) | 0.0566 |  
+-----------+---------------+  
--- Query executed in 0.0060 seconds ---  
  
+-------------+-----------------+  
| qualityc | phi(qualityc) |  
+=============+=================+  
| qualityc(0) | 0.7902 |  
+-------------+-----------------+  
| qualityc(1) | 0.1997 |  
+-------------+-----------------+  
| qualityc(2) | 0.0101 |  
+-------------+-----------------+  
--- Query executed in 0.0020 seconds ---  
  
+-------------+-----------------+  
| qualityc | phi(qualityc) |  
+=============+=================+  
| qualityc(0) | 0.9938 |  
+-------------+-----------------+  
| qualityc(1) | 0.0031 |  
+-------------+-----------------+  
| qualityc(2) | 0.0031 |  
+-------------+-----------------+  
--- Query executed in 0.0020 seconds ---  
  
+-------------+-----------------+  
| qualityc | phi(qualityc) |  
+=============+=================+  
| qualityc(0) | 0.8014 |  
+-------------+-----------------+  
| qualityc(1) | 0.1888 |  
+-------------+-----------------+  
| qualityc(2) | 0.0098 |  
+-------------+-----------------+  
--- Query executed in 0.0020 seconds ---

试验证据推理（解释），例如如果希望碳储量（totc）、生境质量（qualityc）和授粉者丰度（totalpollinator）的值都趋向于高值，即为状态 2 时，那么推断 LULC 各分类的概率可以得知，土地覆盖类型更可能为郁闭混交林（Forest Closed mixed），其状态概率为0.2031；而最不可能的土地覆盖类型有被烧过的草地（Burned grass）、针叶林（Conifer Woodlot）、杂交杨树（Hybrid poplar）等，其状态概率均为0。

usda\_pgm.query\_report(infer, variables=['lulccur'], evidence={'totc':2,'qualityc':2,'totalpollinator':2,},)

+---------------------------------------------------+----------------+  
| lulccur | phi(lulccur) |  
+===================================================+================+  
| lulccur(Bare/fallow) | 0.0129 |  
+---------------------------------------------------+----------------+  
| lulccur(Berries & Vineyards) | 0.0008 |  
+---------------------------------------------------+----------------+  
| lulccur(Burned grass) | 0.0000 |  
+---------------------------------------------------+----------------+  
| lulccur(Christmas trees) | 0.0072 |  
+---------------------------------------------------+----------------+  
| lulccur(Commercial) | 0.0003 |  
+---------------------------------------------------+----------------+  
| lulccur(Commercial/Industrial) | 0.0003 |  
+---------------------------------------------------+----------------+  
| lulccur(Conifer Woodlot) | 0.0000 |  
+---------------------------------------------------+----------------+  
| lulccur(Conifers 0-20 yrs) | 0.0005 |  
+---------------------------------------------------+----------------+  
| lulccur(Double cropping) | 0.0006 |  
+---------------------------------------------------+----------------+  
| lulccur(Field crop) | 0.0473 |  
+---------------------------------------------------+----------------+  
| lulccur(Flooded/marsh) | 0.0006 |  
+---------------------------------------------------+----------------+  
| lulccur(Forest Closed hardwood) | 0.0746 |  
+---------------------------------------------------+----------------+  
| lulccur(Forest Closed mixed) | 0.2031 |  
+---------------------------------------------------+----------------+  
| lulccur(Forest closed conifer 21-40 yrs) | 0.0004 |  
+---------------------------------------------------+----------------+  
| lulccur(Forest closed conifer 41-60 yrs) | 0.0530 |  
+---------------------------------------------------+----------------+  
| lulccur(Forest closed conifer 61-80 yrs) | 0.0151 |  
+---------------------------------------------------+----------------+  
| lulccur(Forest closed conifer 81-200 yrs) | 0.0124 |  
+---------------------------------------------------+----------------+  
| lulccur(Forest closed conifer older than 200 yrs) | 0.0123 |  
+---------------------------------------------------+----------------+  
| lulccur(Grains) | 0.0368 |  
+---------------------------------------------------+----------------+  
| lulccur(Grass) | 0.0007 |  
+---------------------------------------------------+----------------+  
| lulccur(Grass seed rotation) | 0.0522 |  
+---------------------------------------------------+----------------+  
| lulccur(Hayfield) | 0.0371 |  
+---------------------------------------------------+----------------+  
| lulccur(Hybrid poplar) | 0.0000 |  
+---------------------------------------------------+----------------+  
| lulccur(Industrial) | 0.0127 |  
+---------------------------------------------------+----------------+  
| lulccur(Irrigated annual crop rotation) | 0.0296 |  
+---------------------------------------------------+----------------+  
| lulccur(Irrigated perennial) | 0.0007 |  
+---------------------------------------------------+----------------+  
| lulccur(Late field crop) | 0.0005 |  
+---------------------------------------------------+----------------+  
| lulccur(Light duty roads) | 0.0430 |  
+---------------------------------------------------+----------------+  
| lulccur(Main channel non-vegetated) | 0.0011 |  
+---------------------------------------------------+----------------+  
| lulccur(Mint) | 0.0002 |  
+---------------------------------------------------+----------------+  
| lulccur(Natural grassland) | 0.0007 |  
+---------------------------------------------------+----------------+  
| lulccur(Natural shrub) | 0.0832 |  
+---------------------------------------------------+----------------+  
| lulccur(Orchard) | 0.0157 |  
+---------------------------------------------------+----------------+  
| lulccur(Pasture) | 0.0738 |  
+---------------------------------------------------+----------------+  
| lulccur(Permanent lentic water) | 0.0006 |  
+---------------------------------------------------+----------------+  
| lulccur(Railroad) | 0.0102 |  
+---------------------------------------------------+----------------+  
| lulccur(Residential 0-4 units/acre) | 0.0712 |  
+---------------------------------------------------+----------------+  
| lulccur(Residential 4-9 units/acre) | 0.0007 |  
+---------------------------------------------------+----------------+  
| lulccur(Residential 9-16 units/acre) | 0.0057 |  
+---------------------------------------------------+----------------+  
| lulccur(Residential >16 units/acre) | 0.0002 |  
+---------------------------------------------------+----------------+  
| lulccur(Row crop) | 0.0007 |  
+---------------------------------------------------+----------------+  
| lulccur(Rural non-vegetated unknown) | 0.0007 |  
+---------------------------------------------------+----------------+  
| lulccur(Rural structures) | 0.0101 |  
+---------------------------------------------------+----------------+  
| lulccur(Secondary roads) | 0.0008 |  
+---------------------------------------------------+----------------+  
| lulccur(Stream orders 5-7) | 0.0102 |  
+---------------------------------------------------+----------------+  
| lulccur(Sugar beet seed) | 0.0002 |  
+---------------------------------------------------+----------------+  
| lulccur(Topographic Shadow) | 0.0006 |  
+---------------------------------------------------+----------------+  
| lulccur(Turfgrass) | 0.0058 |  
+---------------------------------------------------+----------------+  
| lulccur(Upland Forest Semi-closed conifer) | 0.0010 |  
+---------------------------------------------------+----------------+  
| lulccur(Upland Forest Semi-closed hardwood) | 0.0006 |  
+---------------------------------------------------+----------------+  
| lulccur(Upland Forest Semi-closed mixed) | 0.0114 |  
+---------------------------------------------------+----------------+  
| lulccur(Upland Forest open) | 0.0007 |  
+---------------------------------------------------+----------------+  
| lulccur(Urban non-vegetated unknown) | 0.0265 |  
+---------------------------------------------------+----------------+  
| lulccur(Urban tree overstory) | 0.0128 |  
+---------------------------------------------------+----------------+  
--- Query executed in 0.0060 seconds ---

注释（Notes）：

① 2005年千年生态系统评估（Millennium Ecosystem Assessment，MA）报告，（<https://www.millenniumassessment.org/en/Index-2.html>）。

② Luke Brander，（<http://lukebrander.com/>）。

③ 可持续发展基金会（Foundation for Sustainable Development，FSD），（<https://fsd.nl/who-we-are/>）。

④ 英国苏格兰农业学院（Scotlands Rural College，SRUC），（<https://www.sruc.ac.uk/>）。

⑤ ESVD（Ecosystem Services Valuation Database，（<https://www.esvd.net/>）。

⑥ InVEST（Integrated Valuation of Ecosystem Services and Tradeoffs），（<https://naturalcapitalproject.stanford.edu/software/invest>）。

⑦ InVEST 应用程序编程接口（application programming interface，API ），（<https://invest.readthedocs.io/en/latest/>）。

⑧ NatCap（natural capital） 研究团队，（<https://naturalcapitalproject.stanford.edu/about/people>）。

⑨ PyGeoprocessing，（<https://pygeoprocessing.readthedocs.io/en/latest/>）。

⑩ VITO，（<https://vito.be/en>）。

⑪ Nature Value Explorer（NVE）（Natuur waarde verkenner（荷兰语），NWV），（<https://www.natuurwaardeverkenner.be/#contact>）。

⑫ 全球大气研究排放数据库（Emissions Database for Global Atmospheric Research，EDGAR，（<https://edgar.jrc.ec.europa.eu/>）。

⑬ 政府间气候变化专门委员会（Intergovernmental Panel on Climate Change，IPCC），（<https://www.ipcc.ch/>）。

⑭ EDGAR，（<https://edgar.jrc.ec.europa.eu/report_2022>）。

⑮ Dash（plotly），（<https://dash.plotly.com/installation>）。

⑯ USDA\_dashboard，（<https://github.com/richieBao/USDA_dashboard>）。

⑰ 联合国环境规划署（United Nations Environment Programme）对世界各国温室气体（）数据的可视化描述，（<https://www.unep.org/explore-topics/climate-action/what-we-do/climate-action-note/state-of-climate.html>）。

⑱ NWV土壤中的碳封存，（<https://natuurwaardeverkenner.be/docs/manual_EN/regulerende_diensten/omgeving/carbonsoil>）。

⑲ NWV生物质能（biomass）中的碳封存，（<https://natuurwaardeverkenner.be/docs/manual_EN/regulerende_diensten/omgeving/carbonbiomass>）。

⑳ NWV 生态系统服务在线计算，（<https://www.natuurwaardeverkenner.be/>）。

㉑ InVEST 碳储存和封存，（<https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/zh/carbonstorage.html)>）。

㉒ InVEST 海岸带蓝碳（Coastal Blue Carbon）， （<https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/zh/coastal_blue_carbon.html>）。

㉓ InVEST 森林碳边缘效应（Forest Carbon Edge Effect），（<https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/zh/carbon_edge.html>）。

㉔ InVEST 案例数据，（<http://releases.naturalcapitalproject.org/?prefix=invest/3.13.0/data/>）。

㉕ 杨元合课题组发布青藏高原高寒草地土壤碳库及其动态变化专题数据，（<http://www.nesdc.org.cn/collection/view/6124789b7e28172cbed3d803>）。

㉖ 国家温室气体清单指南（农业、森林和其它土地利用），（<https://www.ipcc-nggip.iges.or.jp/public/2006gl/vol4.html>）。

㉗ InVEST 其它一些碳库碳储存估算的数据来源，（<https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/zh/data_sources.html#belowground-biomass>）。

㉘ pygeoprocessin库distance\_transform\_edt距离计算方法，（<https://pygeoprocessing.readthedocs.io/en/latest/api/pygeoprocessing.html#pygeoprocessing.distance_transform_edt>）。

㉙ pgmpy库，（<https://github.com/pgmpy/pgmpy>）。

参考文献（References）:

[1] Chapin, F. Stuart, Pamela A. Matson, and Peter M. Vitousek. 2011. Principles of Terrestrial Ecosystem Ecology. 2nd ed. New York, NY: Springer.

[2] Ecosystem-Wikipedia, <https://en.wikipedia.org/wiki/Ecosystem>.

[3] Ecosystem service-Wikipedia, <https://en.wikipedia.org/wiki/Ecosystem_service>.

[4] What is biodiversity? Why it’s under threat and why it matters,<https://www.worldwildlife.org/pages/what-is-biodiversity>.

[5] Barbier, E. B. et al. The value of estuarine and coastal ecosystem services. Ecological Monographs vol. 81 (2011).

[6] 刘纪远，岳天祥，鞠洪波等主编.中国西部生态系统综合评估概要（千年生态系统评估国际合作项目）.2005.

[7] Ecosystem valuation-Wikipedia,<https://en.wikipedia.org/wiki/Ecosystem_valuation>.

[8] Seminar, J. B. & Boyd, J. Speaker Economic Valuation, Ecosystem Services, and Conservation Strategy.

[9] Brander, L.M. de Groot, R, Guisado Goñi, V., van ’t Hoff, V., Schägner, P., Solomonides, S., McVittie, A., Eppink, F., Sposato, M., Do, L., Ghermandi, A., and Sinclair, M. (2023). Ecosystem Services Valuation Database (ESVD). Foundation for Sustainable Development and Brander Environmental Economics.

[10] 石铁矛,李沛颖,汤煜.碳中和背景下城市碳汇功能及提升策略——以沈阳核心区为例[J].中国园林,2022,38(03):78-83.DOI:10.19775/j.cla.2022.03.0078.

[11] Crippa, M., Guizzardi, D., Banja, M., Solazzo, E., Muntean, M., Schaaf, E., Pagani, F., Monforti-Ferrario, F., Olivier, J., Quadrelli, R., Risquez Martin, A., Taghavi-Moharamli, P., Grassi, G., Rossi, S., Jacome Felix Oom, D., Branco, A., San-Miguel-Ayanz, J. and Vignati, E., CO2 emissions of all world countries - JRC/IEA/PBL 2022 Report, EUR 31182 EN, Publications Office of the European Union, Luxembourg, 2022, doi:10.2760/730164, JRC130363.

[12] IEA-EDGAR CO2, a component of the EDGAR (Emissions Database for Global Atmospheric Research) Community GHG database version 7.0 (2022) including or based on data from IEA (2021) Greenhouse Gas Emissions from Energy, www.iea.org/data-and-statistics, as modified by the Joint Research Centre.

[13] 张桂莲,仲启铖,张浪.面向碳中和的城市园林绿化碳汇能力建设研究[J].风景园林,2022,29(05):12-16.DOI:10.14085/j.fjyl.2022.05.0012.05.

[14] 武静,蒋卓利,吴晓露.城市蓝绿空间的碳汇研究热点与趋势分析[J].风景园林,2022,29(12):43-49.DOI:10.14085/j.fjyl.2022.12.0043.07.

[15] 王振坤,董心悦,邵明,姚朋.国内外城乡绿色空间碳汇研究进展与展望[J].风景园林,2023,30(02):115-122.DOI:10.14085/j.fjyl.202207310459.

[16] Huang, L., Chen, K. & Zhou, M. Climate change and carbon sink: a bibliometric analysis. Environmental Science and Pollution Research 27, 8740–8758 (2020).

[17] Meersmans, J., F. De Ridder, et al. 2008. “A multiple regression approach to assess the spatial distribution of Soil Organic Carbon (SOC) at the regional scale (Flanders, Belgium).” Geoderma 143(1-2): 1-13.

[18] Vos, P., Janssen, S., Verhees, L., de Wolff, J., Erbrink, H., 2012. Modellering van het effect van wegbegeleidend luchtgroen op de luchtkwaliteit. VITO Rapport nr 2012/RMA/R/112, VITO.

[19] Altor, A. E. and W. J. Mitsch, 2008. “Methane and carbon dioxide dynamics in wetland mesocosms: Effects of hydrology and soils.” Ecological Applications 18(5): 1307-1320

[20] Kuik O, Brander L, Tol RSJ. Marginal abatement costs of greenhouse gas emissions: A meta-analysis. Energy Policy 2009; 37:1395–1403.

[21] De Nocker, L; Michiels, H; Deutsch, F; Lefebvre, W; Buekers, J; Torfs R. 2010. Actualisering van de externe milieuschadekosten (algemeen voor Vlaanderen) met betrekking tot luchtverontreiniging en klimaatverandering; Studie uitgevoerd in opdracht van MIRA, Milieurapport Vlaanderen MIRA/2010/03; December 2010; 122 p. , https://www.milieurapport.be/

[22] Van de Walle et al. 2005 Growing stock-based assessment of the carbon stock in the Belgian forest biomass. Annals of Forest Science 62: 1-12

[23] Carbon Storage and Sequestration, <https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/en/carbonstorage.html>.

[24] Stern, N. 2007. The Economics of Climate Change: The Stern Review. Cambridge and New York: Cambridge University Press.

[25] Tol, RSJ. 2009. The Economic Effects of Climate Change.Journal of Economic Perspectives 23:29-51.

[26] Nordhaus, W. 2007a. Critical Assumptions in the Stern Review on Climate Change. Science 317 (5835): 201–202.

[27] 卿苗,赵军,冯超,黄治化,温媛媛,张伟婕.1980—2030年石羊河流域生态系统碳储存服务对土地利用变化的响应[J].生态学报,2022,42(23):9525-9536.

[28] InVEST 数据来源, <https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/zh/data_sources.html#belowground-biomass>.

[29] Habitat Quality\_InVEST, <https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/en/habitat_quality.html>.

[30] Meijster, Arnold, Jos BTM Roerdink, and Wim H. Hesselink. “A general algorithm for computing distance transforms in linear time.” Mathematical Morphology and its applications to image and signal processing. Springer, Boston, MA, 2002. 331-340.

[31] Crop Pollination (Pollinator Abundance),<https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/en/croppollination.html>.

[32] Crop Production,<https://storage.googleapis.com/releases.naturalcapitalproject.org/invest-userguide/latest/en/crop_production.html#the-model>.