

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
CAMPUS DOIS VIZINHOS  
CURSO DE ESPECIALIZAÇÃO EM CIÊNCIA DE DADOS

BRUNA MATTIOLI DE OLIVEIRA

**BIBLIOTECAS PARA VISUALIZAÇÃO DE DADOS ESPACIAIS EM  
APLICAÇÕES WEB: ESTUDO COMPARATIVO**

TRABALHO DE CONCLUSÃO DE CURSO DE ESPECIALIZAÇÃO

DOIS VIZINHOS  
2022

BRUNA MATTIOLI DE OLIVEIRA

**BIBLIOTECAS PARA VISUALIZAÇÃO DE DADOS ESPACIAIS EM  
APLICAÇÕES WEB: ESTUDO COMPARATIVO**

**LIBRARIES FOR VISUALIZING SPATIAL DATA IN WEB  
APPLICATIONS: A COMPARATIVE STUDY**

Trabalho de Conclusão de Curso de Especialização apresentado ao Curso de Especialização em Ciência de Dados da Universidade Tecnológica Federal do Paraná, como requisito para a obtenção do título de Especialista em Ciência de Dados.

Orientador: Prof. Dr. Yuri Kaszubowski Lopes

DOIS VIZINHOS  
2022



4.0 Internacional

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

BRUNA MATTIOLI DE OLIVEIRA

## **BIBLIOTECAS PARA VISUALIZAÇÃO DE DADOS ESPACIAIS EM APLICAÇÕES WEB: ESTUDO COMPARATIVO**

Trabalho de Conclusão de Curso de Especialização apresentado ao Curso de Especialização em Ciência de Dados da Universidade Tecnológica Federal do Paraná, como requisito para a obtenção do título de Especialista em Ciência de Dados.

Data de aprovação: 11/novembro/2022

Yuri Kaszubowski Lopes  
Doutorado  
Universidade do Estado de Santa Catarina

Rafael Alves Paes de Oliveira  
Doutorado  
Universidade Tecnológica Federal do Paraná - Câmpus Dois Vizinhos

Rodolfo Adamshuk Silva  
Doutorado  
Universidade Tecnológica Federal do Paraná - Câmpus Dois Vizinhos

DOIS VIZINHOS  
2022

## **AGRADECIMENTOS**

Agradeço a meus pais por me darem a oportunidade de mais de uma vez estudar em universidades públicas. Agradeço a meu noivo por todo o apoio. Agradeço a meu orientador por me ajudar a ir além.

## RESUMO

O aumento do volume de dados espaciais e a evolução de softwares e aplicativos de geolocalização, gera a necessidade, cada vez maior, de extração e análise dos dados espaciais, visto que estes softwares são utilizados para a solução de problemas do mundo real, como localização, locomoção, logística, análise territorial e ambiental, entre outros. Este trabalho de conclusão de curso (TCC) tem como objetivo apresentar um estudo comparativo entre as principais bibliotecas para visualização de dados espaciais das linguagens de programação Python e R. Foi realizada uma pesquisa bibliográfica com a finalidade de levantamento das principais bibliotecas existentes para a visualização de dados espaciais no Python e R, então foram estabelecidos critérios de comparação entre elas para determinar as bibliotecas que atendem melhor cada necessidade. Como resultado da análise comparativa realizada entre as funcionalidades contidas nas bibliotecas, foram elaboradas tabelas que exibem os principais critérios de comparação entre as bibliotecas para visualização de dados espaciais com a finalidade de facilitar a análise e escolha destas pelos pesquisadores e programadores. Este trabalho servirá também como referencial teórico para futuros trabalhos na área de visualização de dados espaciais e facilitar a escolha da biblioteca que mais se adéqua às necessidades de pesquisadores e programadores.

**Palavras-chave:** Python, R, dados espaciais, visualização de dados espaciais, visualização de dados; dados geográficos

## ABSTRACT

Due to the increase in the volume of spatial data and the evolution of geolocation software and applications, it generates an increasing need for extraction and analysis of spatial data, since these are used to solve real-world problems, such as location, locomotion, logistics, territorial and environmental analysis, among others. This final paper aims to present a comparative study between the main libraries for visualizing spatial data from the Python and R programming languages. The methodology of this final paper is to carry out a research and survey of the main existing libraries for visualizing spatial data in Python and R, establish comparison criteria between them and determine the libraries that best meet each need. As a result, comparative tables of the main libraries and the functionalities defined by the comparison criteria will be made available. The result may serve as input for future works in the area of spatial data visualization, facilitating the choice of the library that best suits the needs of researchers and programmers.

**Keywords:** Python, R, spatial data, spatial data visualization, data visualization, geographic data

## LISTA DE FIGURAS

Figura 1 – Exemplos de tipos de objetos espaciais: (a) ponto simples; (b) ponto complexo; (c) linha simples; (d) linha complexa; (e) região simples; (f) região complexa. . . . .	14
Figura 2 – Exemplos consultas espaciais baseadas em topologia: (a) Point Query; (b) Window Query; (c) Containment Range Query. . . . .	15
Figura 3 – Exemplos consultas espaciais baseadas em topologia: (a) Intersection Range Query; (b) Consulta baseada em distância; (c) kNN. . . . .	15
Figura 4 – Quadro Geral 1 - Funcionalidades das bibliotecas Python . . . . .	18
Figura 5 – Quadro Geral 2 - Funcionalidades das bibliotecas Python . . . . .	19
Figura 6 – Quadro Geral 3 - Funcionalidades das bibliotecas R . . . . .	19
Figura 7 – Quadro Geral 4 - Funcionalidades das bibliotecas R . . . . .	20
Figura 8 – Exemplo de mapa no Leaflet . . . . .	31
Figura 9 – Exemplo de mapa no Tmap . . . . .	32
Figura 10 – Exemplo de mapa no Leafmap . . . . .	33
Figura 11 – Exemplo de mapa no Geemap . . . . .	33

## LISTA DE TABELAS

Tabela 1 – Bibliotecas - Informações Gerais . . . . .	22
Tabela 2 – Bibliotecas - Referências . . . . .	23
Tabela 3 – Bibliotecas - Funções básicas . . . . .	24
Tabela 4 – Bibliotecas - Geometria espacial . . . . .	25
Tabela 5 – Bibliotecas - Mapas com cores . . . . .	26
Tabela 6 – Bibliotecas - Mesclar e Sobrepor Dados . . . . .	27
Tabela 7 – Bibliotecas - Demais funções . . . . .	28
Tabela 8 – Bibliotecas - Sobreposição de camadas, mini mapa e grade . . . . .	29



## LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
DECOM	Departamento de Computação

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>10</b>
<b>1.1</b>	<b>CONTEXTO</b>	<b>10</b>
<b>1.2</b>	<b>PROBLEMA DE PESQUISA</b>	<b>10</b>
<b>1.3</b>	<b>OBJETIVOS</b>	<b>11</b>
1.3.1	OBJETIVO GERAL	11
1.3.2	OBJETIVOS ESPECÍFICOS	12
<b>1.4</b>	<b>JUSTIFICATIVA</b>	<b>12</b>
<b>1.5</b>	<b>MATERIAIS E MÉTODOS</b>	<b>12</b>
<b>1.6</b>	<b>ORGANIZAÇÃO DO TRABALHO</b>	<b>13</b>
<b>2</b>	<b>REVISÃO DE LITERATURA</b>	<b>14</b>
2.1	Tipos de dados espaciais	14
2.2	Tipos de consultas espaciais	14
2.3	Aplicações da análise de dados espaciais	16
<b>3</b>	<b>LEVANTAMENTO DAS BIBLIOTECAS PARA VISUALIZAÇÃO DE DADOS ESPACIAIS</b>	<b>18</b>
3.1	Metodologia do levantamento das bibliotecas	18
3.2	Tabelas Comparativas	19
<b>4</b>	<b>DESCRIÇÃO DAS BIBLIOTECAS MAIS RELEVANTES</b>	<b>30</b>
4.1	Crítérios e seleção das principais bibliotecas	30
4.2	Bibliotecas para o R	31
4.2.1	Leaflet	31
4.2.2	Tmap	31
4.3	Bibliotecas para o Python	32
4.3.1	Leafmap	32
4.3.2	Geemap	32
4.4	Análise Consolidada	34
<b>5</b>	<b>CONCLUSÃO E TRABALHOS FUTUROS</b>	<b>35</b>
5.1	Considerações Finais	35
5.2	Desafios Encontrados	35
5.3	Trabalhos Futuros	35
	<b>REFERÊNCIAS</b>	<b>37</b>

# 1 INTRODUÇÃO

## 1.1 CONTEXTO

Devido ao aumento no volume de criação de dados espaciais causado pelos avanços tecnológicos da internet e de aplicativos de geolocalização, assim como pela disponibilidade de smartphones, gera-se a necessidade, também cada vez maior, de extração e análise dos dados espaciais. Os dados espaciais são utilizados como base para a solução de diferentes problemas do mundo real que envolve a localização, locomoção, logística, análise territorial, análise ambiental, análise de mercado e de consumidores, entre outros (ALAM; TORGO; BIFET, 2021; GÜTING, 1994; SCHNEIDER; BEHR, 2006).

Além dos fatores apresentados acima, nos últimos anos houve um grande crescimento na utilização das linguagens Python e R para aplicações de Ciência de Dados, em especial, na Análise de Dados Espaciais. Segundo Alam, Torgo e Bifet (2021), a linguagem Python se tornou uma das plataformas mais populares para análise de dados na última década enquanto a linguagem R é uma das mais utilizadas na área de Ciência de Dados.

Dada a grande importância e necessidade das análises espaciais para a solução de diversos problemas do mundo real, houve também a criação de diversos pacotes nas linguagens citadas acima, possibilitando diferentes tipos de análise, como por exemplo a criação de mapas interativos, mapas de calor, mapas coropléticos e diversas funcionalidades, como pop-ups, pins, cores, zoom, cálculo de distâncias para permitir que os pesquisadores aprofundem suas análises com elementos espaciais.

Para utilização desses pacotes cria-se a necessidade do pesquisador ou programador identificar e entender quais linguagens e quais bibliotecas são mais aderentes à cada análise, levando em conta as funcionalidades existentes em cada uma delas. Contudo, realizar uma análise aprofundada e também testes e simulações em diversas bibliotecas para descoberta e entendimento do panorama de funcionalidades contidas nelas é muitas vezes custoso/trabalhoso. Sendo assim, os pesquisadores e programadores buscam na literatura formas de otimizar essa tomada de decisão sobre qual biblioteca utilizar de acordo com sua necessidade/funcionalidade requerida.

## 1.2 PROBLEMA DE PESQUISA

Diante da vasta utilização de dados espaciais para realizar análises e solucionar diversos problemas da sociedade atual, e da vasta quantidade de bibliotecas de visualização de dados espaciais, cada uma das bibliotecas possui características e funcionalidades diferentes entre si, que se aplicam melhor à determinados tipos de análises de dados espaciais (ALAM; TORGO; BIFET, 2021). Algumas se adéquam melhor à construção de diferentes tipos de mapas, outras têm mais funcionalidades para tornar os dados mais interativos ou ainda trazer mais

informações dentro dos mapas. Portanto, apesar de uma mesma análise poder ser implementada em diferentes bibliotecas, é possível obter diferentes tipos de mapas e mapas com diferentes informações e interações, o que pode auxiliar a análise e a apresentação dos resultados.

Devido a grande quantidade de bibliotecas *open-source* disponíveis, os pesquisadores da área de ciência de dados espaciais podem tirar proveito de estudos científicos experimentais e exploratórios que apontem quais as bibliotecas mais adequadas para cada análise. Levando em conta a problemática apresentada, este trabalho visa investigar as adequações e funcionalidades das principais bibliotecas para a visualização de dados espaciais existentes dentro das linguagens de programação Python e R e fornecer aos pesquisadores e profissionais da área de ciência de dados espaciais uma relação de funcionalidades disponíveis por biblioteca, com propósito de diminuir a complexidade/tempo/custo de tomada de decisão sobre qual biblioteca utilizar para um determinado problema.

Este TCC tem como base as seguintes questões de pesquisa:

- **Q1** - Quais as principais bibliotecas de visualização de dados espaciais para aplicações web em Python e em R? Quais são suas principais funcionalidades?
- **Q2** - Quais os benefícios da utilização de uma biblioteca específica em detrimento de outras?
- **Q3** - Quais são os pontos fortes das principais bibliotecas existentes no Python e no R para visualização de dados espaciais?
- **Q4** - Quais as melhores bibliotecas no Python e no R para visualização de dados espaciais de acordo com os principais tipos de análises?

Para responder as questões, foi realizada a pesquisa das funcionalidades na documentação de cada biblioteca a fim de identificar proficiências e lacunas destas.

A métrica utilizada para a análise dos dados coletados foi a comparação entre as 21 funcionalidades das bibliotecas analisadas, listadas abaixo: Mapa Interativo, Plataforma utilizada, Manipulações Geométricas, Pontos, Linhas, Polígonos, Mapa coroplético, Mapa de calor, Mesclar dados, Operações de sobreposição, Diferentes projeções, Diferentes estilos, Zomm interativo, Popups, Sobreposição de camadas, Grade, Mini mapa, GeoJSON, Renderizar objetos espaciais, Tipo de Imagem, Cluster.

## 1.3 OBJETIVOS

Os principais objetivos do trabalho são apresentados a seguir.

### 1.3.1 OBJETIVO GERAL

Este trabalho tem como objetivo geral elaborar um estudo comparativo entre as principais bibliotecas existentes no Python e no R para visualização de dados espaciais.

### 1.3.2 OBJETIVOS ESPECÍFICOS

Para atingir o objetivo geral, definiu-se os seguintes objetivos específicos:

- Apresentar o resultado do estudo comparativo em tabelas com critérios de comparação definidos em pesquisa, entre as diferentes bibliotecas existentes no Python e no R para visualização de dados espaciais;
- Facilitar a utilização destas bibliotecas provendo os pontos fortes da utilização destas;
- Aumentar a produtividade de pesquisadores e programadores, por meio da redução de tempo utilizado para decidir qual biblioteca tem maior aderência à sua necessidade.
- Apresentar a descrição das principais bibliotecas de cada linguagem.

### 1.4 JUSTIFICATIVA

Devido ao aumento constante na geração de dados espaciais nos últimos anos, a evolução dos aplicativos de geolocalização e também a necessidade da extração destes dados para a solução de diversos problemas do mundo real, entende-se a necessidade de disponibilizar um comparativo entre as principais bibliotecas das linguagens Python e R com funções para visualização de dados espaciais como forma de auxiliar trabalhos futuros no tema.

As bibliotecas possuem diferentes características e funcionalidades entre si ([ALAM; TORGO; BIFET, 2021](#)), diferindo qual tem melhor aplicação para diferentes tipos de trabalhos e análises. Algumas necessitam da possibilidade de criação de diferentes tipos de mapas, enquanto outras precisam de mapas que mostrem informações mais detalhadas.

Levando em conta a vasta quantidade de bibliotecas *open-source* disponíveis nas duas linguagens, os pesquisadores da área de ciência de dados espaciais podem extrair valor de estudos científicos experimentais e exploratórios que apontem quais as bibliotecas mais adequadas para cada análise.

### 1.5 MATERIAIS E MÉTODOS

Este TCC utilizou as linguagens de programação Python e R, que hoje são as linguagens mais relevantes da área de estudo de Análise e Visualização de Dados Espaciais, devido a enorme quantidade de recursos e bibliotecas existentes nessas linguagens para este fim.

Diante disso, neste TCC foi realizada a pesquisa e levantamento das bibliotecas existentes para as duas linguagens de programação. Em seguida foram estabelecidos critérios de comparação entre as bibliotecas para a visualização de dados espaciais e em seguida foi realizada a pesquisa documental das informações e funcionalidades de cada biblioteca.

A amostra consiste nas principais bibliotecas existentes das linguagens Python e R para visualização de dados espaciais. As bibliotecas foram coletadas através de pesquisa documental das bibliotecas e levantamento bibliográfico, os dados como: Nome da biblioteca, principais informações e principais funcionalidades foram tabulados para cada biblioteca.

## 1.6 ORGANIZAÇÃO DO TRABALHO

Este TCC possui 5 capítulos, iniciando com o capítulo atual de introdução, 5 outros capítulos, conforme detalhado a seguir:

**Capítulo 2.** Neste capítulo serão abordados os conceitos necessários para a compreensão dos tópicos abordados neste trabalho, incluindo: (i) Tipos de dados espaciais e (ii) Tipos de consultas espaciais.

**Capítulo 3.** Neste capítulo é realizado o mapeamento das principais bibliotecas para visualização de dados espaciais existentes nas linguagens Python e R. Contendo os seguintes tópicos: (i) Metodologia do levantamento das bibliotecas; (ii) Critérios de seleção; (iii) Comparação de características gerais em formato de tabela e (iv) Comparação das funcionalidades em formato de tabela.

**Capítulo 4.** Neste capítulo é realizada a seleção das principais bibliotecas mais relevantes de acordo com os critérios de cada linguagem, assim como, a descrição das bibliotecas mais relevantes da linguagem R e descrição das bibliotecas mais relevantes da linguagem Python.

**Capítulo 5.** Conclusões.

## 2 REVISÃO DE LITERATURA

### 2.1 Tipos de dados espaciais

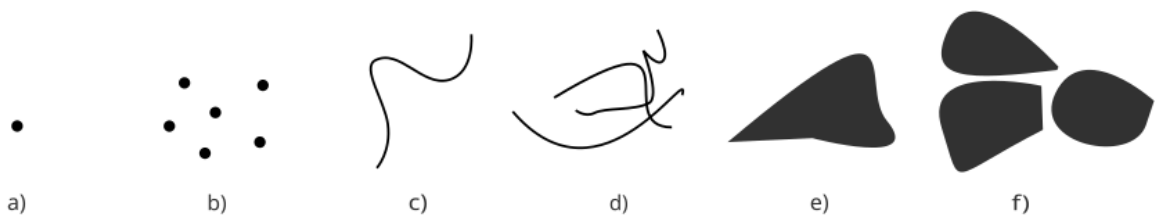
Os dados espaciais, também chamados de geográficos, representam fenômenos do mundo real através da localização geográfica e através de atributos descritivos. A localização geográfica é representada pelas coordenadas da posição de um dado em um sistema de coordenadas, no qual a coordenada é um número que representa uma posição relativa deste ponto a um ponto de referência.

Um dado espacial pode ser representado pelo modelo vetorial, ou pelo modelo matricial (raster) (GÜTING, 1994; CASANOVA et al., 2005). O modelo de dados vetorial representa o espaço como um conjunto de objetos geográficos, que é representado por pontos, linhas ou polígonos, referenciados geograficamente. Cada objeto geográfico também tem atributos descritivos como nome, população etc. O modelo de dados matricial representa um espaço geográfico, uma matriz formada por m linhas e n colunas, no qual cada célula possui um par de coordenadas que especifica sua localização geográfica.

Para objetos simples, as abstrações fundamentais utilizadas para representação são pontos, linhas, regiões (polígonos), para representações complexas: pontos complexos, linhas complexas ou regiões complexas, conforme a Figura 1 (GÜTING, 1994).

Segundo Schneider e Behr (2006), um ponto complexo pode incluir vários pontos, uma linha complexa pode incluir várias linhas, e uma região complexa pode ser uma região de várias partes, conforme a Figura 1.

Figura 1 – Exemplos de tipos de objetos espaciais: (a) ponto simples; (b) ponto complexo; (c) linha simples; (d) linha complexa; (e) região simples; (f) região complexa.



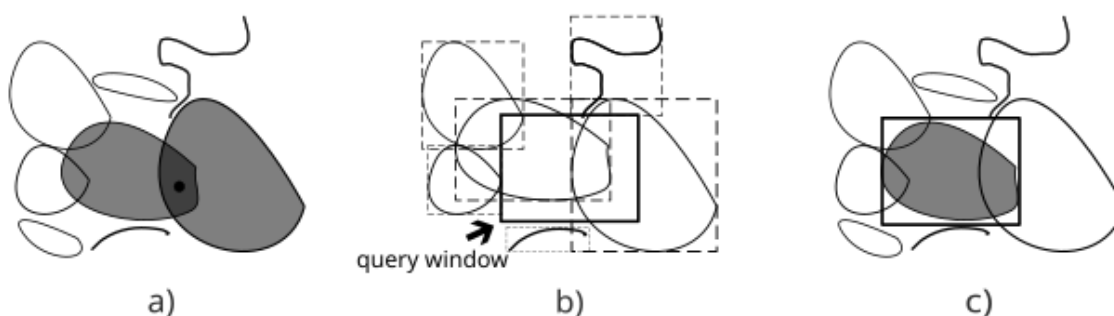
Fonte: Autoria Própria

### 2.2 Tipos de consultas espaciais

As consultas espaciais consistem na recuperação de dados espaciais em Banco de Dados Espaciais nos quais os dados são armazenados, a partir da utilização de relacionamentos topológicos de forma a selecionar objetos espaciais de interesse que atendam às restrições do usuário.

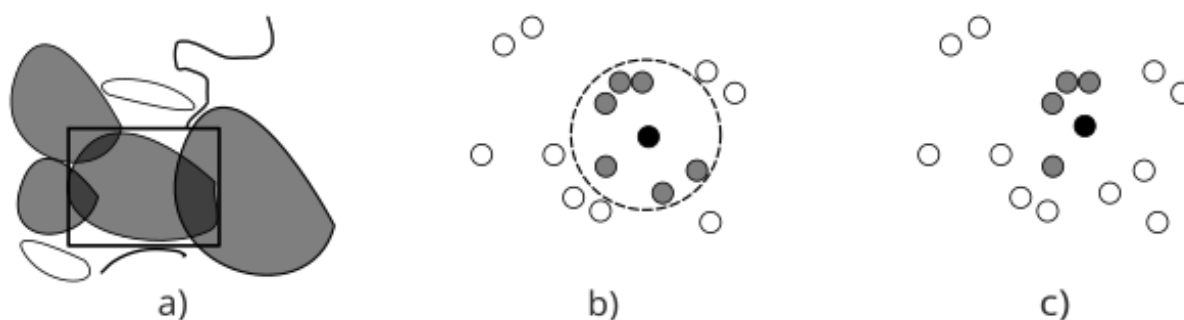
Existem diferentes tipos de consultas que utilizam critérios diferentes para selecionar objetos espaciais (CARNIEL, 2020; GAEDE; GÜNTHER, 1998). Entre eles, os que se destacam são: (i) *point query* (consulta de ponto), (ii) *window query* (consulta de janela), (iii) *containment range query* - CRQ (consulta de alcance de contenção), (iv) *Intersection Range Query* - IRQ (consulta de alcance de intersecção), (v) consulta baseada em distância, (vi) *k Nearest Neighbours* - kNN (k vizinhos mais próximos espacial) e (vii) *spatial join* (junção espacial).

Figura 2 – Exemplos consultas espaciais baseadas em topologia: (a) Point Query; (b) Window Query; (c) Containment Range Query.



Fonte: Autoria Própria

Figura 3 – Exemplos consultas espaciais baseadas em topologia: (a) Intersection Range Query; (b) Consulta baseada em distância; (c) kNN.



Fonte: Autoria Própria

Abaixo temos a descrição do que cada consulta retorna:

- **Point query:** Serão retornados todos os objetos espaciais que contenham um determinado ponto  $p$  em sua geometria, conforme a Figura 2.
- **Window query:** Serão retornados todos os objetos espaciais que contém uma intersecção com a janela de consulta, conforme a Figura 2.



- **Containment range query:** Serão retornados todos os objetos espaciais que contém todos os seus pontos dentro da janela de consulta, conforme a Figura 2.
- **Intersection range query:** Serão retornados todos os objetos espaciais que se intersectam com o objeto espacial de interesse, conforme a Figura 3.
- **Consulta baseada em distância:** Serão retornados objetos espaciais dentro de um raio de distância do objeto espacial de interesse, conforme a Figura 3.
- **kNN:** Serão retornados os k objetos espaciais mais próximos a um ponto ou uma região especificada, conforme a Figura 3.
- **Spatial join:** Serão retornados os pares de objetos espaciais que satisfazem uma dada relação ou uma combinação de relações espaciais (CARNIEL, 2020).

### 2.3 Aplicações da análise de dados espaciais

A análise de dados espaciais encontra aplicação em diferentes temas e assuntos. A principal aplicação para a análise de dados espaciais, hoje, é o uso da geolocalização nos smartphones, que se tornou essencial na vida cotidiana moderna. A partir desta, conseguimos consultar a localização de um local e a distância deste ponto até nós, assim como encontrar locais próximos. Porém, a análise de dados espaciais está presente para além deste contexto, nos campos da saúde, meio-ambiente, agricultura, finanças, planejamento urbano, entre outros. Os conceitos e técnicas presentes na análise de dados espaciais podem beneficiar inúmeros tipos de estudos em todos os campos de pesquisa (PANDEY et al., 2018; CARNIEL, 2020).

A análise de dados espaciais ganhou destaque também durante a epidemia do Coronavírus, no qual teve grande importância para detectar e entender o avanço da doença. De acordo com Alam, Torgo e Bifet (2021), o mapeamento realizado pela análise e visualização de dados espaciais na área de epidemiologia e saúde pública podem se tornar ferramentas essenciais para lidar com futuras pandemias. Outro caso citado por Alam, Torgo e Bifet (2021) são referentes aos incêndios que ocorreram na Austrália no ano de 2020, que foi o maior incêndio da história da Austrália; e neste cenário, a análise de dados espaciais é fundamental para detectar os incêndios no começo para que sejam mais fáceis de controlá-los.

Na área da saúde existem aplicações para a análise de dados espaciais, na qual esta pode ser utilizada para investigar como uma doença se dissemina em uma população, se tem características que se relacionam com o meio no qual as pessoas vivem, o que pode ser de grande auxílio aos gestores na tomada de decisão relacionada a saúde pública. No contexto de meio ambiente é possível aplicar a análise espacial à fluxos migratórios de animais, desmatamento e detecção de poluentes (ALAM; TORGO; BIFET, 2021).

Além disso, o uso das bibliotecas espaciais para aplicações é muito importante uma vez

que, segundo [Kyle et al. \(2021\)](#) tudo tem uma localização, seja relacionado a logística, mercado imobiliário, varejo, marketing, serviços financeiros, e isso torna a análise de dados espaciais e a utilização das bibliotecas espaciais centrais na solução da maior parte dos problemas de negócios.

No trabalho de [Kyle et al. \(2021\)](#) são utilizados dados relacionados ao consumo de café para entender o comportamento do consumidor com dados de tráfego de pedestres buscando entender quão longe os consumidores estão dispostos a ir para tomar café, qual localização tem maior visibilidade e de onde são a maior parte de consumidores de café a partir da análise com as bibliotecas GeoPandas e Folium da linguagem Python para análise de dados espaciais.

O trabalho citado acima mostra como todo tipo de negócio pode se beneficiar da extração e análise de dados a partir de bibliotecas espaciais para entender os consumidores, realizar a criação de marketing direcionado ao tipo de consumidor e entender a localização que trará mais visibilidade para um negócio. Outros exemplos de aplicação das bibliotecas espaciais estão presentes no trabalho [Wu \(2021\)](#), no qual o mesmo utiliza a biblioteca Leafmap da linguagem Python para realizar análises geoespaciais e criar mapas interativos.

### 3 LEVANTAMENTO DAS BIBLIOTECAS PARA VISUALIZAÇÃO DE DADOS ESPACIAIS

#### 3.1 Metodologia do levantamento das bibliotecas

Foram coletadas as principais bibliotecas existentes para visualização de dados nas linguagens Python e R, levando em conta as mais relevantes no momento no qual a pesquisa foi realizada. As bibliotecas foram coletadas via pesquisa nos sites google e github. Foram levantadas 32 bibliotecas, sendo 20 da linguagem Python e 12 da linguagem R.

Dessas bibliotecas foram extraídos os dados de ano da última atualização, site da documentação, informações de referência, além de 21 critérios de comparação descritos abaixo: Mapa Interativo, Plataforma utilizada, Manipulações Geométricas, Pontos, Linhas, Polígonos, Mapa coroplético, Mapa de calor, Mesclar dados, Operações de sobreposição, Diferentes projeções, Diferentes estilos, Zomm interativo, Popups, Sobreposição de camadas, Grade, Mini mapa, GeoJSON, Renderizar objetos espaciais, Tipo de Imagem, Cluster.

Os critérios acima foram determinados a partir da pesquisa exploratória nas bibliotecas para visualização de dados espaciais das linguagem Python e R, portanto foram coletados os critérios que mais se repetiram dentre as documentações de todas as bibliotecas, além disso, foi levado em conta também os critérios mais relevantes para as pesquisas dentro da área de visualização de dados espaciais.

As funcionalidades atendidas por cada biblioteca foram organizadas em 4 figuras 4, 5, 6, 7 com visualização clara de quais bibliotecas tem cada característica.

Figura 4 – Quadro Geral 1 - Funcionalidades das bibliotecas Python

Nome	R/Python	Mapa Interativo	Openstreet map	Googleearth	Manipulações Geométricas	Pontos	Linhas	Polígonos	Mapa coroplético	Mapa de calor	Cluster
Leafmap	Python	✓	✓		✓	✓	✓	✓	✓	✓	✓
Geemap	Python	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
GeoPandas	Python	✓	✓		✓	✓	✓	✓	✓	✓	✓
Folium	Python	✓	✓			✓	✓	✓	✓	✓	✓
Bokeh	Python	✓	✓	✓		✓	✓	✓		✓	
GeoPlot	Python	✓	✓			✓	✓	✓	✓	✓	
Plotly	Python	✓	✓	✓		✓	✓	✓	✓	✓	✓
Plotly.express	Python	✓	✓	✓		✓	✓	✓	✓	✓	✓
PySAL	Python	✓	✓		✓	✓	✓	✓	✓	✓	✓
Geoviews	Python	✓	✓			✓	✓	✓	✓	✓	
Eomaps	Python	✓	✓			✓	✓	✓			
PyMove	Python				✓	✓	✓	✓		✓	✓
Matplotlib	Python					✓	✓	✓	✓	✓	
Pydeck	Python	✓	✓	✓	✓		✓	✓		✓	
CartoPy	Python		✓		✓	✓	✓	✓			
Geoplotlib	Python				✓	✓	✓	✓	✓	✓	
Seaborn	Python					✓	✓	✓		✓	✓
Spaghetti	Python				✓	✓	✓	✓			✓
Shapely	Python				✓	✓	✓	✓			
Ggplot	Python						✓				

Figura 5 – Quadro Geral 2 - Funcionalidades das bibliotecas Python

Nome	R/Python	Miscelar dados	Operações de sobreposição	Diferentes projeções	Diferentes estilos	Zomm interativo	Popups	Sobreposição estática	Sobreposição dinâmica	Grade	Mini mapa
Leafmap	Python	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Geemap	Python		✓	✓	✓	✓	✓	✓		✓	✓
GeoPandas	Python	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Folium	Python		✓	✓	✓	✓	✓	✓	✓	✓	✓
Bokeh	Python	✓	✓	✓	✓	✓	✓	✓		✓	
GeoPlot	Python	✓	✓	✓	✓	✓		✓		✓	✓
Plotly	Python			✓	✓	✓	✓			✓	✓
Plotly.express	Python			✓	✓	✓	✓			✓	✓
PySAL	Python	✓	✓		✓	✓		✓		✓	
Geoviews	Python	✓	✓	✓	✓			✓	✓	✓	
Eomaps	Python		✓	✓	✓	✓		✓	✓	✓	
PyMove	Python	✓			✓	✓	✓			✓	✓
Matplotlib	Python		✓	✓	✓	✓	✓			✓	
Pydeck	Python				✓	✓				✓	
CartoPy	Python			✓	✓	✓		✓		✓	
Geoplotlib	Python				✓	✓				✓	
Seaborn	Python			✓	✓					✓	
Spaghetti	Python		✓					✓			
Shapely	Python	✓		✓							
Ggplot	Python				✓					✓	

Fonte: Autoria própria

Figura 6 – Quadro Geral 3 - Funcionalidades das bibliotecas R

Nome	R/Python	Mapa Interativo	Openstreet map	Googleeearth	Manipulações Geométricas	Pontos	Linhas	Polígonos	Mapa cloropético	Mapa de calor	Cluster
Leaflet	R	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Tmap	R	✓	✓		✓	✓	✓	✓	✓		✓
Ggplot2	R	✓	✓			✓	✓	✓		✓	
Ggmap	R		✓	✓	✓	✓	✓	✓			✓
Mapdeck	R	✓	✓			✓	✓	✓		✓	
PlotKML	R	✓		✓	✓		✓	✓			
Plotly	R	✓				✓	✓	✓	✓	✓	✓
Mapview	R	✓	✓				✓	✓			
Cartography	R	✓	✓		✓		✓	✓	✓		
Sf	R				✓		✓	✓			
RasterVis	R	✓					✓	✓			
Mapsf	R						✓	✓	✓		✓

Fonte: Autoria própria

### 3.2 Tabelas Comparativas

Nesta seção, as informações de cada biblioteca foram organizadas em 8 tabelas, agrupadas por tema de modo a facilitar a análise do leitor.

A [Tabela 1](#) apresenta informações gerais a respeito das bibliotecas coletadas, mostra informações como a linguagem relacionada à biblioteca, o ano de sua última atualização e o site da documentação. O ano de atualização da última versão das bibliotecas Python foi coletado do site <<https://pypi.org>> enquanto o ano da última versão das bibliotecas R foi coletado

Figura 7 – Quadro Geral 4 - Funcionalidades das bibliotecas R

Nome	R/Python	Miscelar dados	Operações de sobreposição	Diferentes projeções	Diferentes estilos	Zomm interativo	Popups	Sobreposição estática	Sobreposição dinâmica	Grade	Mini mapa
Leaflet	R		✓	✓	✓	✓	✓	✓	✓	✓	✓
Tmap	R		✓	✓	✓	✓	✓	✓		✓	✓
Ggplot2	R	✓	✓	✓	✓	✓				✓	
Ggmap	R		✓		✓	✓		✓		✓	
Mapdeck	R				✓	✓		✓		✓	
PlotKML	R		✓	✓	✓	✓				✓	
Plotly	R			✓		✓				✓	
Mapview	R				✓	✓	✓			✓	
Cartography	R				✓	✓					
Sf	R	✓	✓	✓						✓	
RasterVis	R				✓					✓	
Mapsf	R				✓	✓					

Fonte: Autoria própria

do site <<https://cran.r-project.org>>. Na Tabela 1 é possível perceber que cerca de 70% das bibliotecas de ambas as linguagens receberam suas últimas atualizações no ano de 2022, mesmo ano de publicação do TCC, o que demonstra que estamos lidando majoritariamente com as bibliotecas mais atuais de visualização de dados espaciais. A Tabela 2 apresenta a referência bibliográfica de cada biblioteca coletada e utilizada neste trabalho.

A Tabela 3 apresenta os critérios de “Mapa Interativo” e “Plataforma utilizada”. O critério “Mapa Interativo” refere-se a possibilidade do usuário interagir com a figura, podendo clicar sobre as regiões do mapa e obter alguma informação a respeito dela e também movê-lo a partir de cliques no mapa. Do total de 32 bibliotecas, 22 destas tem a função de criação de mapas interativos, sendo 12 dessas da linguagem Python e 10 da linguagem R. As bibliotecas do quadro que não possuem a função de mapa interativo possuem a função de criação de mapas não interativos.

O critério “Plataforma utilizada” refere-se a plataforma que a biblioteca utiliza, podendo ser Openstreetmap, Googleearth, ambas as anteriores ou nenhuma delas. Em relação a plataforma utilizada para gerar os mapas, da linguagem Python, 13 bibliotecas utilizam o Openstreetmap e apenas 7 utilizam o Googleearth, enquanto da linguagem R, 5 delas utilizam o Openstreetmap e apenas 3 utilizam o Googleearth.

A Tabela 4 apresenta os critérios de “Manipulações Geométricas”, “Pontos”, “Linhas” e “Polígonos”. O critério “Manipulações Geométricas” refere-se a possibilidade de realizar manipulações geométricas simples, em especial, o cálculo da distância entre dois pontos. Das bibliotecas coletadas, metade das bibliotecas realizam manipulações geométricas, sendo 10 da linguagem Python e 6 da linguagem R. Os critérios “Pontos”, “Linhas” e “Polígonos” referem-se a possibilidade de inserir pontos, linhas e polígonos nos mapas para destaque e apresentação de diferentes tipos de informações. Na linguagem Python, 17 bibliotecas estudadas permitem utilizar esses 3 elementos, enquanto 6 bibliotecas têm essa possibilidade na linguagem R.

A Tabela 5 apresenta os critérios “Mapa coroplético”, “Mapa de calor” e “Cluster”.

O critério “Mapa coroplético” refere-se a possibilidade do usuário inserir mapas coropléticos a partir da biblioteca. Mapas coropléticos consistem em criar um mapa com zonas e áreas geográficas coloridas onde as cores de cada área pertence a uma faixa de valor de uma variável dos dados (OLIVEIRA, 2019). Das bibliotecas analisadas, apenas metade tem a possibilidade de criar mapas coropléticos. O critério “Mapa de calor” refere-se a possibilidade do usuário inserir mapas de calor a partir da biblioteca. O mapa de calor existe a partir da generalização de um gráfico de dispersão, onde cada célula apresenta uma coloração de forma gradiente com base em valores (METSALU T.; VILO, 2015). Nesse tipo de mapa, as áreas com maior densidade, ou com maior volume de ocorrências são indicadas por cores mais quentes, mais próximas do vermelho, enquanto as regiões de menor densidade, ou menor número de ocorrências de um fenômeno são marcadas com cores mais frias, mais próximas do azul. Das bibliotecas analisadas, 19 delas tem a possibilidade de criar mapas de calor. O critério “Cluster” refere-se a possibilidade de trabalhar com clusterização, ou seja, agrupamento de dados a partir de algum critério definido, como por exemplo por semelhança ou por distância, dentro dos mapas. Das bibliotecas analisadas, 15 delas tem a possibilidade de formar clusters.

A Tabela 6 apresenta os critérios “Mesclar dados” e “Operações de sobreposição”. O critério “Mesclar dados” refere-se a possibilidade de combinar conjuntos de dados a partir de junções de atributos e junções espaciais. Das bibliotecas analisadas, apenas 10 delas têm a possibilidade de mesclar dados. O critério “Operações de Sobreposição” refere-se a possibilidade de sobrepor polígonos no mapa. 11 bibliotecas analisadas possuem essa função no Python e 6 no R.

A Tabela 7 apresenta os critérios “Diferentes projeções”, “Diferentes estilos”, “Zomm interativo” e “Popups”. O critério “Diferentes projeções” está relacionado com a possibilidade que a biblioteca traz de utilizar diferentes posições para as visualizações. A maioria da bibliotecas analisadas tem a função de criar diferentes projeções, sendo 14 da linguagem Python e 6 da linguagem R. O critério “Diferentes estilos” está relacionado com a possibilidade que a biblioteca traz de utilizar diferentes estilos de mapas para as visualizações. 28 das 32 bibliotecas analisadas fornecem ao usuário a função de trabalhar com diferentes estilos. O critério “Zomm interativo” refere-se a possibilidade de aumentar e diminuir o zoom de maneira interativa dentro de um mapa. 25 bibliotecas dentre as analisadas possuem essa funcionalidade. O critério “Popups” refere-se a possibilidade de criar popups com informações sobre um determinado local ou região para que o usuário possa consultar de forma interativa dentro do mapa. Já para a criação de popups, apenas 12 bibliotecas apresentaram essa funcionalidade.

A Tabela 8 apresenta os critérios “Sobreposição de camadas”, “Grade” e “Minimapa”. O critério “Sobreposição de camadas” traz a possibilidade da criação de camadas dentro dos mapas de forma estática, ou seja, através do código da linguagem, ou de forma dinâmica, a partir de cliques no próprio mapa. Das bibliotecas selecionadas, 15 delas apresentam a possibilidade de criar camadas de forma estática, enquanto 6 apresentam a possibilidade de criar camadas de forma dinâmica. O critério “Grade” refere-se a possibilidade de inserir grade nos mapas

para auxiliar a visualização. Das 32 bibliotecas analisadas, 27 têm a possibilidade de adicionar grade nos mapas. O critério “Minimapa” refere-se a possibilidade de inserir um minimapa em algum canto do mapa principal com informações geográficas mais gerais para ajudar o usuário a navegar pelo mapa principal com maior facilidade. Das bibliotecas coletadas, 9 delas têm a possibilidade de adicionar um minimapa.

Tabela 1 – Bibliotecas - Informações Gerais

<b>Biblioteca</b>	<b>Linguagem</b>	<b>Ano</b>	<b>Site da Documentação</b>
GeoPandas	Python	2022	< <a href="https://geopandas.org/">https://geopandas.org/</a> >
PySAL	Python	2022	< <a href="https://pysal.org/">https://pysal.org/</a> >
PyMove	Python	2022	< <a href="https://pymove.readthedocs.io/">https://pymove.readthedocs.io/</a> >
Shapely	Python	2022	< <a href="https://shapely.readthedocs.io/">https://shapely.readthedocs.io/</a> >
Folium	Python	2021	< <a href="https://python-visualization.github.io/folium">https://python-visualization.github.io/folium</a> >
GeoPlot	Python	2022	< <a href="https://residentmario.github.io/geoplot/">https://residentmario.github.io/geoplot/</a> >
Pydeck	Python	2021	< <a href="https://deckgl.readthedocs.io/">https://deckgl.readthedocs.io/</a> >
Plotly	Python	2022	< <a href="https://plotly.com/">https://plotly.com/</a> >
Geoplotlib	Python	2016	< <a href="https://andrea-cuttone.github.io/geoplotlib/">https://andrea-cuttone.github.io/geoplotlib/</a> >
CartoPy	Python	2022	< <a href="https://scitools.org.uk/cartopy/">https://scitools.org.uk/cartopy/</a> >
Geemap	Python	2022	< <a href="https://geemap.org/">https://geemap.org/</a> >
Seaborn	Python	2022	< <a href="https://seaborn.pydata.org/">https://seaborn.pydata.org/</a> >
Matplotlib	Python	2022	< <a href="https://matplotlib.org/">https://matplotlib.org/</a> >
Bokeh	Python	2022	< <a href="https://bokeh.org/">https://bokeh.org/</a> >
Ggplot	Python	2016	< <a href="https://yhat.github.io/ggpy/">https://yhat.github.io/ggpy/</a> >
Leafmap	Python	2022	< <a href="https://leafmap.org/">https://leafmap.org/</a> >
Spaghetti	Python	2022	< <a href="https://pysal.org/spaghetti/">https://pysal.org/spaghetti/</a> >
Eomaps	Python	2022	< <a href="https://eomaps.readthedocs.io/">https://eomaps.readthedocs.io/</a> >
Geoviews	Python	2022	< <a href="https://geoviews.org/">https://geoviews.org/</a> >
Ggplot2	R	2022	< <a href="https://cran.r-project.org/web/packages/ggplot2/">https://cran.r-project.org/web/packages/ggplot2/</a> >
Leaflet	R	2022	< <a href="https://cran.r-project.org/web/packages/leaflet/">https://cran.r-project.org/web/packages/leaflet/</a> >
Ggmap	R	2019	< <a href="https://cran.r-project.org/web/packages/ggmap/">https://cran.r-project.org/web/packages/ggmap/</a> >
Mapdeck	R	2020	< <a href="https://cran.r-project.org/web/packages/mapdeck/">https://cran.r-project.org/web/packages/mapdeck/</a> >
PlotKML	R	2022	< <a href="https://cran.r-project.org/web/packages/plotKML/">https://cran.r-project.org/web/packages/plotKML/</a> >
Tmap	R	2022	< <a href="https://cran.r-project.org/web/packages/tmap/">https://cran.r-project.org/web/packages/tmap/</a> >
MapView	R	2022	< <a href="https://cran.r-project.org/web/packages/mapview/">https://cran.r-project.org/web/packages/mapview/</a> >
Cartography	R	2021	< <a href="https://cran.r-project.org/web/packages/cartography">https://cran.r-project.org/web/packages/cartography</a> >
Mapsf	R	2022	< <a href="https://cran.r-project.org/web/packages/mapsf/">https://cran.r-project.org/web/packages/mapsf/</a> >
Plotly	R	2021	< <a href="https://cran.r-project.org/web/packages/plotly/">https://cran.r-project.org/web/packages/plotly/</a> >
Sf	R	2022	< <a href="https://cran.r-project.org/web/packages/plotly/">https://cran.r-project.org/web/packages/plotly/</a> >
RasterVis	R	2022	< <a href="https://cran.r-project.org/web/packages/rasterVis/">https://cran.r-project.org/web/packages/rasterVis/</a> >

Fonte: Autoria própria

Tabela 2 – Bibliotecas - Referências

<b>Biblioteca</b>	<b>Linguagem</b>	<b>Referência</b>
GeoPandas	Python	(JORDAHL et al., 2020)
PySAL	Python	(REY; ANSELIN, 2007)
PyMove	Python	(SANCHES, 2019)
Shapely	Python	(GILLIES, 2022)
Folium	Python	(VISUALIZATION, )
GeoPlot	Python	(BILOGUR, 2022)
Pydeck	Python	(DUBERSTEIN IB GREEN, 2021)
Plotly	Python	(INC., 2015)
Geoplotlib	Python	(CUTTONE SUNE LEHMANN, 2016)
CartoPy	Python	(Met Office, 2010 - 2015)
Geemap	Python	(WU, 2020)
Seaborn	Python	(WASKOM et al., 2017)
Matplotlib	Python	(HUNTER, 2007)
Bokeh	Python	(Bokeh Development Team, 2018)
Ggplot	Python	(LAMP, 2016)
Leafmap	Python	(QIUSHENG, 2021)
Spaghetti	Python	(GABOARDI; REY; LUMNITZ, 2021)
Eomaps	Python	(QUAST, 2022)
Geoviews	Python	(RUDIGER et al., 2020)
Ggplot2	R	(WICKHAM, 2016)
Leaflet	R	(GRAUL, 2016)
Ggmap	R	(KAHLE; WICKHAM, 2013)
Mapdeck	R	(COOLEY, 2020)
PlotKML	R	(HENGL et al., 2015)
Tmap	R	(TENNEKES, 2018)
Mapview	R	(APPELHANS et al., 2022)
Cartography	R	(GIRAUD; LAMBERT, 2016)
Mapsf	R	(GIRAUD, 2022)
Plotly	R	(INC., )
Sf	R	(PEBESMA, 2018)
RasterVis	R	(PERPIÑÁN; HIJMANS, 2022)

Fonte: Autoria própria



Tabela 3 – Bibliotecas - Funções básicas

<b>Biblioteca</b>	<b>Linguagem</b>	<b>Mapa Interativo</b>	<b>Plataforma utilizada</b>
GeoPandas	Python	Sim	Openstreetmap
PySAL	Python	Sim	Openstreetmap
PyMove	Python	Não	Nenhum
Shapely	Python	Não	Nenhum
Folium	Python	Sim	Openstreetmap
GeoPlot	Python	Sim	Openstreetmap
Pydeck	Python	Sim	Ambos
Plotly	Python	Sim	Ambos
Geoplotlib	Python	Não	Nenhum
CartoPy	Python	Não	Openstreetmap
Geemap	Python	Sim	Ambos
Seaborn	Python	Não	Nenhum
Matplotlib	Python	Não	Nenhum
Bokeh	Python	Sim	Ambos
Ggplot	Python	Não	Nenhum
Leafmap	Python	Sim	Openstreetmap
Spaghetti	Python	Não	Nenhum
Eomaps	Python	Sim	Openstreetmap
Geoviews	Python	Sim	Openstreetmap
Ggplot2	R	Sim	Openstreetmap
Leaflet	R	Sim	Ambos
Ggmap	R	Sim	Ambos
Mapdeck	R	Sim	Openstreetmap
PlotKML	R	Sim	Googleearth
Tmap	R	Sim	Openstreetmap
MapView	R	Sim	Openstreetmap
Cartography	R	Sim	Openstreetmap
Mapsf	R	Não	Nenhum
Plotly	R	Sim	Nenhum
Sf	R	Não	Nenhum
RasterVis	R	Sim	Nenhum

Fonte: Autoria própria

Tabela 4 – Bibliotecas - Geometria espacial

<b>Biblioteca</b>	<b>Linguagem</b>	<b>Manipulações Geométricas</b>	<b>Pontos</b>	<b>Linhas</b>	<b>Polígonos</b>
GeoPandas	Python	Sim	Sim	Sim	Sim
PySAL	Python	Sim	Sim	Sim	Sim
PyMove	Python	Sim	Sim	Sim	Sim
Shapely	Python	Sim	Sim	Sim	Sim
Folium	Python	Não	Sim	Sim	Sim
GeoPlot	Python	Não	Não	Sim	Não
Pydeck	Python	Sim	Não	Sim	Sim
Plotly	Python	Não	Sim	Sim	Sim
Geoplotlib	Python	Sim	Sim	Sim	Sim
CartoPy	Python	Sim	Sim	Sim	Sim
Geemap	Python	Sim	Sim	Sim	Sim
Seaborn	Python	Não	Não	Sim	Sim
Matplotlib	Python	Não	Sim	Sim	Sim
Bokeh	Python	Não	Sim	Sim	Sim
Ggplot	Python	Não	Não	Sim	Não
Leafmap	Python	Sim	Sim	Sim	Sim
Spaghetti	Python	Sim	Sim	Sim	Sim
Eomaps	Python	Não	Sim	Sim	Sim
Geoviews	Python	Não	Sim	Sim	Sim
Ggplot2	R	Não	Não	Sim	Sim
Leaflet	R	Sim	Sim	Sim	Sim
Ggmap	R	Sim	Sim	Sim	Sim
Mapdeck	R	Não	Sim	Sim	Sim
PlotKML	R	Sim	Não	Sim	Sim
Tmap	R	Sim	Sim	Sim	Sim
Mapview	R	Não	Não	Sim	Sim
Cartography	R	Sim	Não	Sim	Sim
Mapsf	R	Não	Não	Sim	Sim
Plotly	R	Não	Sim	Sim	Sim
Sf	R	Sim	Não	Sim	Sim
RasterVis	R	Não	Não	Sim	Sim

Fonte: Autoria própria

Tabela 5 – Bibliotecas - Mapas com cores

<b>Biblioteca</b>	<b>Linguagem</b>	<b>Mapa coroplético</b>	<b>Mapa de calor</b>	<b>Cluster</b>
GeoPandas	Python	Sim	Sim	Sim
PySAL	Python	Sim	Sim	Sim
PyMove	Python	Não	Sim	Sim
Shapely	Python	Não	Não	Não
Folium	Python	Sim	Sim	Sim
GeoPlot	Python	Sim	Sim	Não
Pydeck	Python	Não	Sim	Não
Plotly	Python	Sim	Sim	Sim
Geoplotlib	Python	Sim	Sim	Não
CartoPy	Python	Não	Não	Não
Geemap	Python	Sim	Sim	Sim
Seaborn	Python	Não	Sim	Sim
Matplotlib	Python	Sim	Sim	Não
Bokeh	Python	Não	Sim	Não
Ggplot	Python	Não	Não	Não
Leafmap	Python	Sim	Sim	Sim
Spaghetti	Python	Não	Não	Sim
Eomaps	Python	Não	Não	Não
Geoviews	Python	Sim	Sim	Não
Ggplot2	R	Não	Sim	Não
Leaflet	R	Sim	Sim	Sim
Ggmap	R	Não	Não	Sim
Mapdeck	R	Não	Sim	Não
PlotKML	R	Não	Não	Não
Tmap	R	Sim	Não	Sim
Mapview	R	Não	Não	Não
Cartography	R	Sim	Não	Não
Mapsf	R	Sim	Não	Não
Plotly	R	Sim	Sim	Sim
Sf	R	Não	Não	Não
RasterVis	R	Não	Não	Sim

Fonte: Autoria própria

Tabela 6 – Bibliotecas - Mesclar e Sobrepor Dados

<b>Biblioteca</b>	<b>Linguagem</b>	<b>Mesclar dados</b>	<b>Operações de sobreposição</b>
GeoPandas	Python	Sim	Sim
PySAL	Python	Sim	Sim
PyMove	Python	Sim	Não
Shapely	Python	Sim	Não
Folium	Python	Não	Sim
GeoPlot	Python	Sim	Sim
Pydeck	Python	Não	Não
Plotly	Python	Não	Não
Geoplotlib	Python	Não	Não
CartoPy	Python	Não	Não
Geemap	Python	Não	Sim
Seaborn	Python	Não	Não
Matplotlib	Python	Não	Sim
Bokeh	Python	Sim	Sim
Ggplot	Python	Não	Não
Leafmap	Python	Sim	Sim
Spaghetti	Python	Não	Sim
Eomaps	Python	Não	Sim
Geoviews	Python	Sim	Sim
Ggplot2	R	Sim	Sim
Leaflet	R	Não	Sim
Ggmap	R	Não	Sim
Mapdeck	R	Não	Não
PlotKML	R	Não	Sim
Tmap	R	Não	Sim
Mapview	R	Não	Não
Cartography	R	Não	Não
MapsF	R	Não	Não
Plotly	R	Não	Não
Sf	R	Sim	Sim
RasterVis	R	Não	Não

Fonte: Autoria própria

Tabela 7 – Bibliotecas - Demais funções

<b>Biblioteca</b>	<b>Linguagem</b>	<b>Diferentes projeções</b>	<b>Diferentes estilos</b>	<b>Zomm interativo</b>	<b>Popups</b>
GeoPandas	Python	Sim	Sim	Sim	Sim
PySAL	Python	Não	Sim	Não	Sim
PyMove	Python	Não	Sim	Sim	Sim
Shapely	Python	Sim	Não	Não	Não
Folium	Python	Sim	Sim	Sim	Sim
GeoPlot	Python	Sim	Sim	Não	Não
Pydeck	Python	Não	Sim	Não	Sim
Plotly	Python	Sim	Sim	Sim	Sim
Geoplotlib	Python	Não	Sim	Não	Sim
CartoPy	Python	Sim	Sim	Não	Sim
Geemap	Python	Não	Sim	Não	Sim
Seaborn	Python	Sim	Não	Não	Sim
Matplotlib	Python	Sim	Sim	Sim	Sim
Bokeh	Python	Sim	Sim	Sim	Sim
Ggplot	Python	Não	Não	Não	Sim
Leafmap	Python	Sim	Sim	Sim	Sim
Spaghetti	Python	Não	Não	Não	Não
Eomaps	Python	Sim	Não	Não	Sim
Geoviews	Python	Sim	Não	Não	Sim
Ggplot2	R	Sim	Sim	Não	Sim
Leaflet	R	Sim	Sim	Sim	Sim
Ggmap	R	Não	Sim	Não	Sim
Mapdeck	R	Não	Sim	Não	Sim
PlotKML	R	Sim	Sim	Não	Sim
Tmap	R	Sim	Sim	Sim	Sim
Mapview	R	Não	Sim	Sim	Sim
Cartography	R	Não	Sim	Não	Sim
Mapsf	R	Não	Sim	Não	Sim
Plotly	R	Sim	Sim	Não	Não
Sf	R	Sim	Não	Não	Não
RasterVis	R	Não	Não	Não	Sim

Fonte: Autoria própria

Tabela 8 – Bibliotecas - Sobreposição de camadas, mini mapa e grade

<b>Biblioteca</b>	<b>Linguagem</b>	<b>Sobreposição de camadas</b>	<b>Grade</b>	<b>Minimapa</b>
GeoPandas	Python	Ambas	Sim	Não
PySAL	Python	Estática	Sim	Não
PyMove	Python	Nenhuma	Sim	Sim
Shapely	Python	Nenhuma	Não	Não
Folium	Python	Ambas	Sim	Sim
GeoPlot	Python	Estática	Sim	Sim
Pydeck	Python	Nenhuma	Sim	Não
Plotly	Python	Nenhuma	Sim	Sim
Geoplotlib	Python	Nenhuma	Não	Não
CartoPy	Python	Estática	Sim	Não
Geemap	Python	Estática	Sim	Sim
Seaborn	Python	Nenhuma	Sim	Não
Matplotlib	Python	Nenhuma	Sim	Não
Bokeh	Python	Estática	Sim	Não
Ggplot	Python	Nenhuma	Sim	Não
Leafmap	Python	Ambas	Sim	Sim
Spaghetti	Python	Estática	Não	Não
Eomaps	Python	Ambas	Sim	Não
Geoviews	Python	Ambas	Sim	Não
Ggplot2	R	Nenhuma	Sim	Não
Leaflet	R	Ambas	Sim	Sim
Ggmap	R	Estática	Sim	Não
Mapdeck	R	Estática	Sim	Não
PlotKML	R	Nenhuma	Sim	Não
Tmap	R	Estática	Sim	Sim
Mapview	R	Nenhuma	Sim	Não
Cartography	R	Nenhuma	Não	Não
Mapsf	R	Nenhuma	Não	Não
Plotly	R	Nenhuma	Sim	Não
Sf	R	Nenhuma	Sim	Não
RasterVis	R	Nenhuma	Sim	Não

Fonte: Autoria própria

## 4 DESCRIÇÃO DAS BIBLIOTECAS MAIS RELEVANTES

### 4.1 Critérios e seleção das principais bibliotecas

Após coleta das bibliotecas para visualização de dados espaciais das linguagens R e Python e coleta dos critérios atendidos por elas, é possível selecionar as principais bibliotecas de ambas as linguagens, de forma a definir quais destas atendem a mais critérios e determinar quais servem a mais propósitos de visualização de dados espaciais. Para isso, os critérios de seleção abaixo foram estabelecidos baseados na maior usabilidade das bibliotecas para aplicações de visualização de dados espaciais.

- Criação de mapas interativos;
- Criação de manipulações geométricas;
- Criação de pontos, linhas e polígonos no mapa;
- Criação de mapa coroplético;
- Criação de clusters;
- Criação de diferentes projeções;
- Criação de diferentes estilos;
- Zoom interativo;
- Sobreposição estática;
- Minimapa.

Foram encontradas 2 bibliotecas para o R que atendem a esses critérios, sendo elas: Leaflet e Tmap e 2 bibliotecas para o Python que atendem a esses critérios, sendo elas: Leafmap e Geemap.

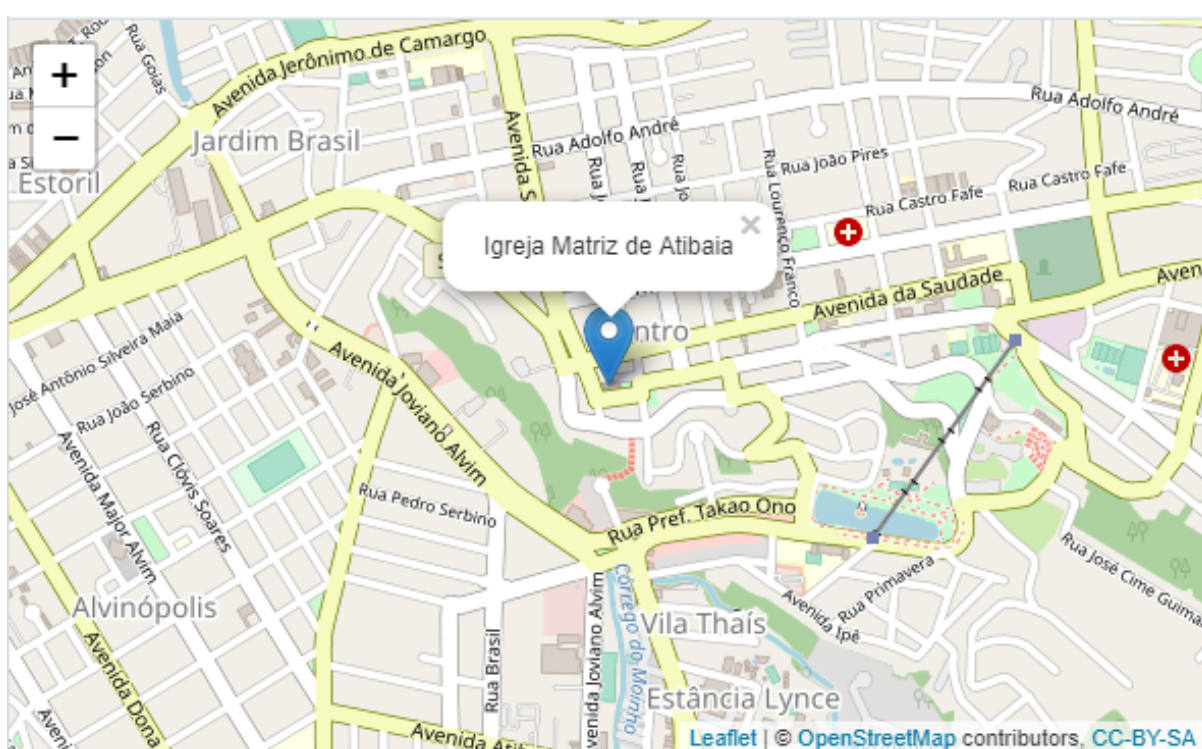
## 4.2 Bibliotecas para o R

### 4.2.1 Leaflet

Leaflet é a principal biblioteca de código aberto para a criação de mapas interativos, utilizando como base a linguagem R, esta é a que fornece mais funções e ferramentas para visualização de dados espaciais. Através desta, é possível criar e personalizar mapas interativos. A biblioteca atendeu a 20 dos 21 critérios estudados nesta pesquisa, não atendendo apenas ao critério de mesclagem de dados. Foi a biblioteca que atendeu a mais critérios dentre as bibliotecas do R e ficou em segundo lugar considerando as bibliotecas das linguagens Python também.

Segundo a documentação, essa biblioteca é utilizada por sites como de notícias The New York Times e The Washington Post, GitHub e Flickr, além de especialistas em GIS como OpenStreetMap, Mapbox e CartoDB.

Figura 8 – Exemplo de mapa no Leaflet



Fonte: Autoria própria

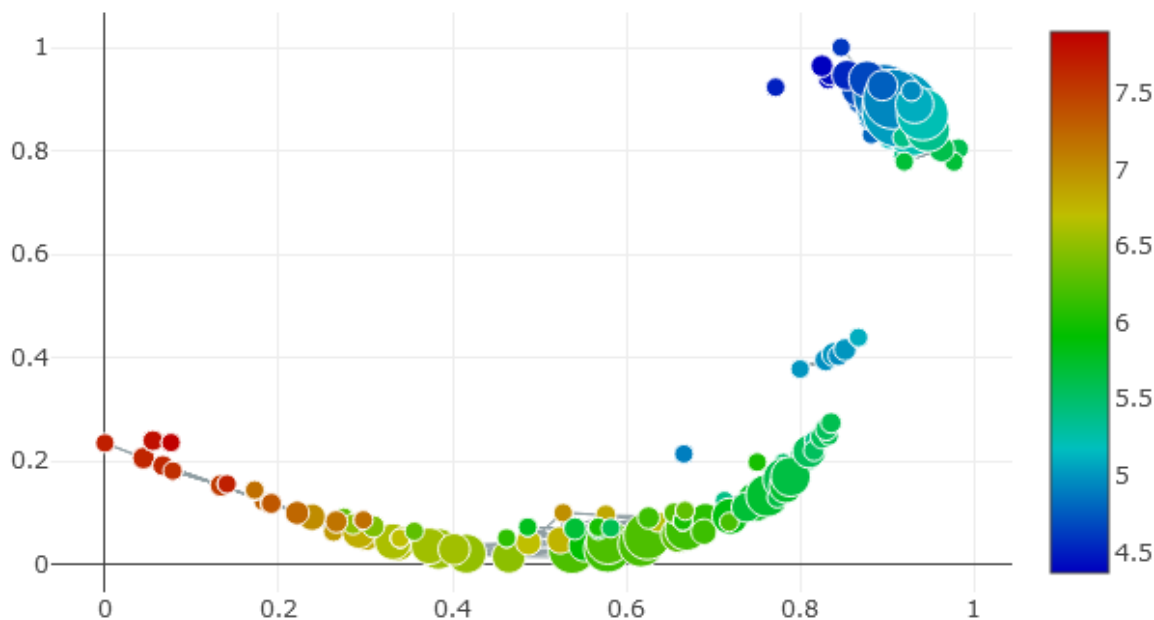
### 4.2.2 Tmap

Tmap, abreviação de Mapas Temáticos, apresenta mapas geográficos nos quais é possível realizar diversas análises visuais de dados espaciais. Este pacote oferece uma abordagem bastante flexível, baseada em camadas e fácil de usar para criar mapas coroplético, realizar manipulações geométricas e formar clusters.



Essa biblioteca é a segunda com mais funções relacionadas a visualização de dados espaciais no R, permite a criação de mapas interativos, manipulações geométricas, mapas coroplético, formação de clusters, diferentes projeções e estilos, assim como sobreposição estática e dinâmica.

Figura 9 – Exemplo de mapa no Tmap



Fonte: (TENNEKES, 2018)

### 4.3 Bibliotecas para o Python

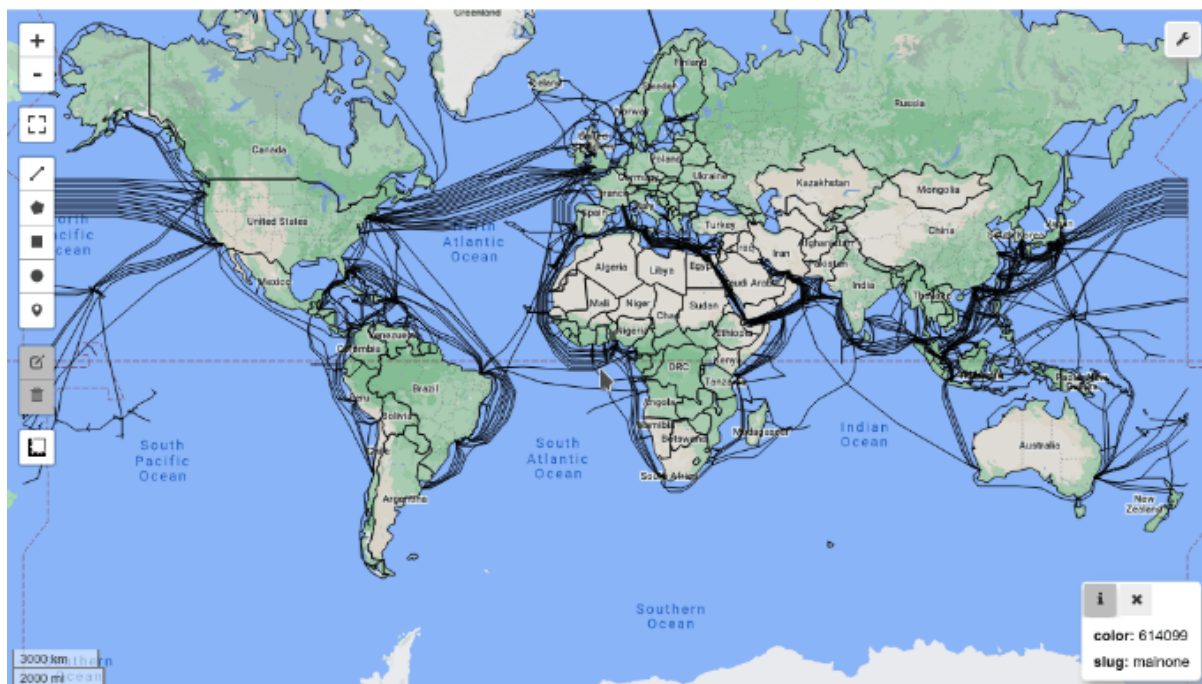
#### 4.3.1 Leafmap

A biblioteca Leafmap é um pacote da linguagem Python, voltada para análises geoespaciais e criação de mapas interativos. Esta se trata da biblioteca do Python que atendeu a mais critérios dentre os estudados neste TCC. A gama de funções permite que os usuários realizem análises e mapas para a visualização de dados espaciais com o mínimo de desenvolvimento de código, dentro do ambiente Jupyter.

#### 4.3.2 Geemap

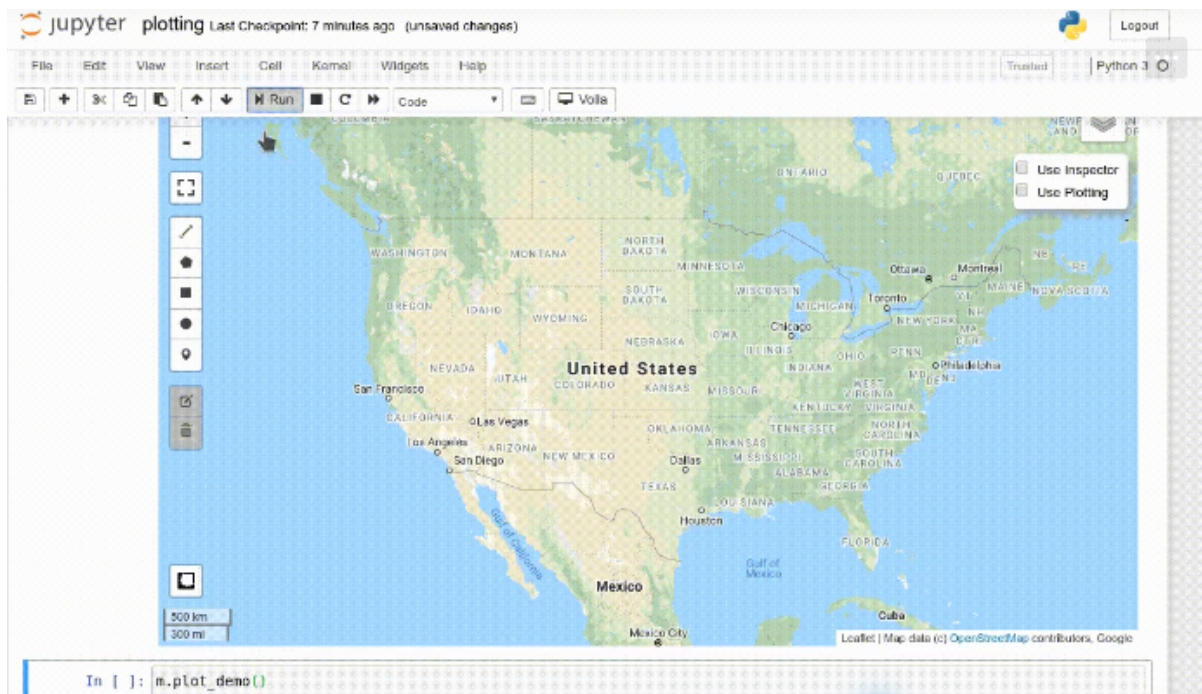
A biblioteca Geemap é um pacote do Python que permite realizar mapas interativos a partir da plataforma do Googleearth. É a segunda biblioteca mais completa da linguagem Python, segundo os critérios estudados neste TCC. Tem como maior diferencial das três bibliotecas anteriores, a possibilidade de trabalhar com a plataforma do Googleearth na criação de mapas interativos e utilização de ferramentas do Google no ambiente Jupyter.

Figura 10 – Exemplo de mapa no Leafmap



Fonte: (QIUSHENG, 2021)

Figura 11 – Exemplo de mapa no Geemap



Fonte: (WU, 2020)

#### 4.4 Análise Consolidada

Levando em conta as bibliotecas e as funcionalidades estudadas neste TCC, é possível perceber que as bibliotecas servem a diferentes propósitos, portanto, é recomendado ao programador ou pesquisador que este utilize as figuras 4, 5, 6, 7 de comparação entre as bibliotecas disponibilizadas no capítulo 3 para selecionar as bibliotecas que possuem as funcionalidades necessárias para a análise que estão desenvolvendo. Abaixo encontra-se uma análise de como as 4 bibliotecas selecionadas na seção 4.1 atendem melhor cada necessidade de pesquisa.

Para análises nas quais os pesquisadores procuram bibliotecas mais gerais para a criação de mapas, é possível utilizar a biblioteca Leafmap da linguagem Python ou a biblioteca Leaflet da linguagem R, dependendo da linguagem que o pesquisador tem maior familiaridade. O Leafmap permite a criação de mapas com o mínimo de desenvolvimento de código no ambiente Jupyter, atendendo a todas as funcionalidades estudadas neste TCC. Já o Leaflet foi a segunda biblioteca com mais funcionalidades atendidas neste TCC, na qual é possível criar mapas interativos com apenas 1 ou 2 linhas de código no R. Ambas são de fácil utilização e bastante intuitivas, portanto, permitem criar mapas e colocar diferentes funcionalidades nos mapas sem que seja necessário destinar muito tempo ao aprendizado de diferentes funcionalidades.

Para análises nas quais o pesquisador pretende criar mapas com diferentes temas, a biblioteca Tmap da linguagem R é a que melhor atende, pois possui diversas funcionalidades voltadas para este fim, a partir desta é possível criar mapas coropléticos ou agrupar os dados em clusters. Porém, se o pesquisador deseja criar mapas de calor, esta biblioteca não possui esta função, portanto ele pode optar pelas bibliotecas Leafmap, Geemap ou Leaflet.

As bibliotecas Leafmap, da linguagem Python, e Tmap, da linguagem R, não possuem a possibilidade de criar mapas utilizando o Googleearth, portanto, se o pesquisador quer ou precisa de maior flexibilidade, podendo criar mapas tanto no Openstreetmap quanto no Googleearth é mais interessante que seja escolhida a biblioteca Geemap do Python ou Leaflet do R.

As 4 bibliotecas citadas acima possuem a funcionalidade de sobrepor elementos de forma estática no mapa, porém, para análises nas quais o pesquisador deseja realizar a sobreposição dinâmica de elementos no mapa, é interessante que ele opte pela biblioteca Leafmap do Python ou Leaflet do R, uma vez que estas possuem esta funcionalidade.

Além da sobreposição estática, as 4 bibliotecas possuem também a funcionalidade de criar pontos, linhas e polígonos dentro dos mapas, assim como realizar manipulações geométricas nos mapas, criar mapas com diferentes projeções e diferentes estilos, utilizar o zoom interativo e adicionar popups, grade e minimapas a depender da necessidade de cada análise.

## 5 CONCLUSÃO E TRABALHOS FUTUROS

### 5.1 Considerações Finais

O presente TCC teve como principal objetivo elaborar um estudo comparativo entre as principais bibliotecas existentes no Python e no R para visualização de dados espaciais, apresentando um resultado comparativo entre elas, facilitando a utilização destas bibliotecas provendo os pontos fortes da utilização destas e assim, aumentar a produtividade de pesquisadores e programadores, por meio da redução do tempo utilizado para decidir qual biblioteca tem maior aderência à sua necessidade.

As linguagens escolhidas foram o Python e o R por se tratarem das linguagens mais relevantes na área de Análise e Visualização de dados espaciais, devido a quantidade de bibliotecas, recursos e ferramentas presentes nestas linguagens.

Para atingir o objetivo, foi realizada a coleta de bibliotecas para visualização de dados espaciais das linguagens R e Python. Foram determinados critérios de comparação entre as bibliotecas e coletado quais critérios cada biblioteca atendia. Após isso, foram estabelecidos critérios para a seleção das duas principais bibliotecas de cada linguagem abrangida neste TCC, e por fim, essas bibliotecas foram descritas com maior detalhamento.

O trabalho cumpriu seu objetivo de determinar quais bibliotecas tem maior aderência para cada tipo de análise e servirá de base para estudos mais aprofundados nas bibliotecas para visualização de dados espaciais, assim como, servirá de base para os pesquisadores tomarem a decisão quanto a qual biblioteca das linguagens estudadas atende melhor aos critérios da análise que precisa ser realizada.

### 5.2 Desafios Encontrados

A principal limitação encontrada consiste em depender da própria documentação das bibliotecas para coleta das funcionalidades que cada uma atende, o que dificulta bastante para as bibliotecas que apresentam uma documentação mais enxuta e com menos dados. Além disso, o fato de cada documentação utilizar um termo diferente para definir uma mesma funcionalidade acaba tornando o processo coleta mais extenso e complexo.

### 5.3 Trabalhos Futuros

O presente TCC trata de uma coleta e análise inicial de bibliotecas de visualização de dados espaciais, para as linguagens Python e R. Para futuros trabalhos na área, é possível realizar a construção de estudos de caso com as ferramentas escolhidas para aprofundar a análise das principais bibliotecas de cada linguagem. É possível também realizar testes de desempenho nas bibliotecas analisadas, realizando a comparação de quais tem melhor performance para os

diferentes objetivos. E por fim, é possível também estender a análise para o estudo de outras bibliotecas de visualização em outras linguagens além das abrangidas neste trabalho.

## Referências

- ALAM, M. M.; TORGO, L.; BIFET, A. A survey on spatio-temporal data analytics systems. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, dec 2021. ISSN 0360-0300. Just Accepted. Disponível em: <<https://doi.org/10.1145/3507904>>. Citado 3 vezes nas páginas 10, 12 e 16.
- APPELHANS, T. et al. **mapview: Interactive Viewing of Spatial Data in R**. [S.l.], 2022. R package version 2.11.0.9002. Disponível em: <<https://github.com/r-spatial/mapview>>. Citado na página 23.
- BILOGUR, A. **GeoPlot**. 2022. Disponível em: <<https://residentmario.github.io/geoplot/index.html>>. Citado na página 23.
- Bokeh Development Team. **Bokeh: Python library for interactive visualization**. [S.l.], 2018. Disponível em: <<https://bokeh.pydata.org/en/latest/>>. Citado na página 23.
- CARNIEL, A. C. Spatial information retrieval in digital ecosystems: A comprehensive survey. In: **Proceedings of the 12th International Conference on Management of Digital EcoSystems**. New York, NY, USA: Association for Computing Machinery, 2020. (MEDES '20), p. 10–17. ISBN 9781450381154. Disponível em: <<https://doi.org/10.1145/3415958.3433038>>. Citado 2 vezes nas páginas 15 e 16.
- CASANOVA, M. et al. **Bancos de Dados Geográficos**. [S.l.]: MundoGEO, 2005. Citado na página 14.
- COOLEY, D. **mapdeck: Interactive Maps Using 'Mapbox GL JS' and 'Deck.gl'**. [S.l.], 2020. R package version 0.3.4. Disponível em: <<https://CRAN.R-project.org/package=mapdeck>>. Citado na página 23.
- CUTTONE SUNE LEHMANN, J. E. L. A. geoplotlib: a python toolbox for visualizing geographical data. ago. 2016. Citado na página 23.
- DUBERSTEIN IB GREEN, K. B. A. **Pydeck**. 2021. Disponível em: <<https://deckgl.readthedocs.io/en/latest/>>. Citado na página 23.
- GABOARDI, J. D.; REY, S.; LUMNITZ, S. spaghetti: spatial network analysis in pysal. **Journal of Open Source Software**, The Open Journal, v. 6, n. 62, p. 2826, 2021. Disponível em: <<https://doi.org/10.21105/joss.02826>>. Citado na página 23.
- GAEDE, V.; GÜNTHER, O. Multidimensional access methods. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 30, n. 2, p. 170–231, jun 1998. ISSN 0360-0300. Disponível em: <<https://doi.org/10.1145/280277.280279>>. Citado na página 15.
- GILLIES, S. **shapely**. 2022. Disponível em: <<https://shapely.readthedocs.io/en/latest/#>>. Citado na página 23.
- GIRAUD, e. a. T. **mapsf: Thematic Cartography**. [S.l.], 2022. R package version 0.5.0. Disponível em: <<https://CRAN.R-project.org/package=mapsf>>. Citado na página 23.



- GIRAUD, T.; LAMBERT, N. cartography: Create and integrate maps in your r workflow. **JOSS**, The Open Journal, v. 1, n. 4, aug 2016. Disponível em: <<http://dx.doi.org/10.21105/joss.00054>>. Citado na página 23.
- GRAUL, C. **leafletR: Interactive Web-Maps Based on the Leaflet JavaScript Library**. [S.l.], 2016. R package version 0.4-0. Disponível em: <<http://cran.r-project.org/package=leafletR>>. Citado na página 23.
- GÜTING, R. H. An introduction to spatial database systems. **The VLDB Journal**, Springer-Verlag, Berlin, Heidelberg, v. 3, n. 4, p. 357–399, oct 1994. ISSN 1066-8888. Citado 2 vezes nas páginas 10 e 14.
- HENGL, T. et al. plotKML: Scientific visualization of spatio-temporal data. **Journal of Statistical Software**, v. 63, n. 5, p. 1–25, 2015. Disponível em: <<https://www.jstatsoft.org/v63/i05/>>. Citado na página 23.
- HUNTER, J. D. Matplotlib: A 2d graphics environment. **Computing in science & engineering**, IEEE, v. 9, n. 3, p. 90–95, 2007. Citado na página 23.
- INC., P. T. **Collaborative data science**. [S.l.]: Plotly Technologies. Citado na página 23.
- INC., P. T. **Collaborative data science**. Montreal, QC: Plotly Technologies Inc., 2015. Disponível em: <<https://plot.ly>>. Citado na página 23.
- JORDAHL, K. et al. **geopandas/geopandas: v0.8.1**. Zenodo, 2020. Disponível em: <<https://doi.org/10.5281/zenodo.3946761>>. Citado na página 23.
- KAHLE, D.; WICKHAM, H. ggmap: Spatial visualization with ggplot2. **The R Journal**, v. 5, n. 1, p. 144–161, 2013. Disponível em: <<https://journal.r-project.org/archive/2013-1/kahle-wickham.pdf>>. Citado na página 23.
- KYLE, J. et al. **Understand Consumer Behavior With Verified Foot Traffic Data**. 2021. Disponível em: <<https://github.com/SafeGraphInc/sigspatial-2021-cafe-analysis>>. Citado na página 17.
- LAMP, G. **ggplot**. [S.l.], 2016. Disponível em: <<https://yhat.github.io/ggpy/>>. Citado na página 23.
- Met Office. **Cartopy: a cartographic python library with a Matplotlib interface**. Exeter, Devon, 2010 – 2015. Disponível em: <<https://scitools.org.uk/cartopy>>. Citado na página 23.
- METSALU T.; VILO, J. Clustvis: A web tool for visualizing clustering of multivariate data using principal component analysis and heatmap. nucleic acids research. 2015. ISSN 13624962. Citado na página 21.
- OLIVEIRA, A. F. d. Uma arquitetura e implementação do módulo de visualização para biblioteca pymove. **Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará**, p. 64 f., 2019. Citado na página 21.
- PANDEY, V. et al. How good are modern spatial analytics systems? **Proc. VLDB Endow.**, VLDB Endowment, v. 11, n. 11, p. 1661–1673, jul 2018. ISSN 2150-8097. Disponível em: <<https://doi.org/10.14778/3236187.3236213>>. Citado na página 16.

- PEBESMA, E. Simple Features for R: Standardized Support for Spatial Vector Data. **The R Journal**, v. 10, n. 1, p. 439–446, 2018. Disponível em: <<https://doi.org/10.32614/RJ-2018-009>>. Citado na página 23.
- PERPIÑÁN, O.; HIJMANS, R. **rasterVis**. [S.l.], 2022. R package version 0.51.2. Disponível em: <<https://oscarperpinan.github.io/rastervis/>>. Citado na página 23.
- QIUSHENG, W. **Leafmap: A Python package for interactive mapping and geospatial analysis with minimal coding in a Jupyter environment**. 2021. Disponível em: <<https://github.com/giswqs/leafmap>>. Citado 2 vezes nas páginas 23 e 33.
- QUAST, R. **EOMaps: EOMaps v4.4.3**. Zenodo, 2022. Disponível em: <<https://doi.org/10.5281/zenodo.7153286>>. Citado na página 23.
- REY, S. J.; ANSELIN, L. PySAL: A Python Library of Spatial Analytical Methods. **The Review of Regional Studies**, v. 37, n. 1, p. 5–27, 2007. Citado na página 23.
- RUDIGER, P. et al. **holoviz/geoviews: Version 1.8.0**. Zenodo, 2020. Disponível em: <<https://doi.org/10.5281/zenodo.4046982>>. Citado na página 23.
- SANCHES, A. D. J. A. M. **Uma arquitetura e implementação do módulo de pré-processamento para biblioteca PyMove**. Dissertação (Bachelor's thesis) — Universidade Federal Do Ceará, 2019. Citado na página 23.
- SCHNEIDER, M.; BEHR, T. Topological relationships between complex spatial objects. **ACM Trans. Database Syst.**, Association for Computing Machinery, New York, NY, USA, v. 31, n. 1, p. 39–81, mar 2006. ISSN 0362-5915. Disponível em: <<https://doi.org/10.1145/1132863.1132865>>. Citado 2 vezes nas páginas 10 e 14.
- TENNEKES, M. tmap: Thematic maps in R. **Journal of Statistical Software**, v. 84, n. 6, p. 1–39, 2018. Citado 2 vezes nas páginas 23 e 32.
- VISUALIZATION python. **Folium**. Disponível em: <<https://python-visualization.github.io/folium/>>. Citado na página 23.
- WASKOM, M. et al. **mwaskom/seaborn: v0.8.1 (September 2017)**. Zenodo, 2017. Disponível em: <<https://doi.org/10.5281/zenodo.883859>>. Citado na página 23.
- WICKHAM, H. **ggplot2: Elegant Graphics for Data Analysis**. Springer-Verlag New York, 2016. ISBN 978-3-319-24277-4. Disponível em: <<https://ggplot2.tidyverse.org>>. Citado na página 23.
- WU, Q. geemap: A python package for interactive mapping with google earth engine. **The Journal of Open Source Software**, v. 5, n. 51, 2020. Disponível em: <<https://doi.org/10.21105/joss.02305>>. Citado 2 vezes nas páginas 23 e 33.
- WU, Q. Interactive mapping and geospatial analysis with leafmap and jupyter. In: **Proceedings of the 3rd ACM SIGSPATIAL International Workshop on APIs and Libraries for Geospatial Data Science**. New York, NY, USA: Association for Computing Machinery, 2021. (SpatialAPI '21). ISBN 9781450391030. Disponível em: <<https://doi.org/10.1145/3486189.3490015>>. Citado na página 17.