

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS DA NATUREZA

Kaik Fontana Octaviano

**BIOMA STATS: UM PACOTE R PARA ANÁLISE DE USO E OCUPAÇÃO DO
SOLO NO TERRITÓRIO BRASILEIRO**

BURI (SP)
2023

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS DA NATUREZA

Kaik Fontana Octaviano

**BIOMA STATS: UM PACOTE R PARA ANÁLISE DE USO E OCUPAÇÃO DO
SOLO NO TERRITÓRIO BRASILEIRO**

Trabalho de Conclusão de Curso apresentado
como exigência parcial para a obtenção do grau
de Bacharel em Engenharia Ambiental na
Universidade Federal de São Carlos.

Orientação: Prof. Dr. Iuri Emmanuel de Paula
Ferreira

Coorientação: Prof. Dr. Paulo Guilherme Molin

BURI (SP)
2023

Este trabalho foi Iniciação Científica financiada pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), pelo Edital 001/2021 da Pró-Reitoria de Pesquisa (ProPq) da Universidade Federal de São Carlos (UFSCar), e utilizado como Trabalho de Conclusão de Curso apresentado como exigência parcial para a obtenção do grau de Bacharel em Engenharia Ambiental na Universidade Federal de São Carlos.

RESUMO

Atualmente, os biomas brasileiros estão em risco mediante as diversas pressões impostas pela ação do Homem. Neste contexto, torna-se fundamental compreender os padrões de uso e ocupação do solo e a ocorrência de transições na paisagem natural. A disponibilidade de dados seriais acerca do uso e ocupação do solo possibilita o diagnóstico ambiental, a detecção de áreas vulneráveis à ação antrópica e o planejamento de políticas de conservação ambiental. O objetivo deste trabalho foi a o desenvolvimento do *Bioma Stats*, um programa computacional que visa a construção de relatórios automatizados sobre o uso e a ocupação do solo em território brasileiro. O *Bioma Stats* é implementado em linguagem computacional R, com código-fonte aberto e versões de livre distribuição. Espera-se que o programa seja uma ferramenta de impacto na área ambiental, contribuindo para estudos de uso e ocupação do solo, bem como na geração de informações úteis para a gestão ambiental e na construção de políticas públicas de combate ao desmatamento, na conservação do meio ambiente, na agricultura de precisão, na pecuária, entre outros.

Palavras-chave: Geotecnologia, Estatística Computacional, Linguagem R, Desmatamento.

1. INTRODUÇÃO

O estudo de dados relacionados à distribuição geográfica de recursos minerais, fauna, flora e atributos físicos da paisagem é uma tarefa importante no contexto do Brasil, país com dimensão continental e com grande demanda de informações para a tomada de decisões na resolução de problemas ambientais, rurais e urbanos. Com o avanço da Tecnologia da Informação, o acesso às informações geográficas e aos dados georreferenciados tem se tornado cada vez mais simples (CÂMARA et al., 2001).

Aliado a Geotecnologia, o Sistema de Informação Geográfica (SIG) realiza a integração de *softwares*, *hardwares*, procedimentos e usuários, possibilitando a coleta, o armazenamento, o processamento, a visualização e a análise de dados georreferenciados (TEIXEIRA, 1995). Dentre os *softwares* SIG utilizados para geoprocessamento, destaca-se o Quantum Geographic Information System (QGIS) como *software* livre e em constante desenvolvimento, capaz de suportar diversos formatos de dados, tais como: arquivos vetoriais (.shp, .dbf e .shx), imagens matriciais (Grid ArcInfo Binary, Grid ArcInfo ASCII, GeoTIFF, ERDAS IMAGINE, entre outros), banco de dados (.csv, .xlsx, .xml, .txt, .json, entre outros) e funcionalidades (QGIS DEVELOPMENT TEAM, 2022).

De forma alternativa aos *softwares* de SIG tradicionais, o sistema R tem se destacado como recurso para a manipulação e a análise de dados georreferenciados (R CORE TEAM, 2021). O R contém uma extensa variedade de pacotes estatísticos como, por exemplo, para a modelagem linear e não linear, para testes estatísticos clássicos, para análises de séries temporais, classificações e agrupamentos, para a visualização em tabelas e gráficos, entre outros, além de ser considerado extremamente extensível, destacando-se por não possuir custos de licença e por seu código ser aberto (R CORE TEAM, 2021). O R ainda conta com o ambiente de desenvolvimento integrado RStudio (RSTUDIO TEAM, 2021). A IDE RStudio foi desenvolvida em prol de facilitar a utilização da linguagem R, agrupando diversas funcionalidades e proporcionando maior interação com o usuário na visualização de comandos e funções, importação de dados, resultados e gráficos, além de criação de documentos (HENNING et al., 2016).

Os *softwares* de SIG processam variados tipos de dados georreferenciados, coletados através de técnicas integrativas que envolvem sensores, satélites, GNSS, e até trabalhos em

campo. Neste contexto destacam-se as imagens obtidas a partir de sensoriamento remoto, adquiridas por drone ou satélites, e que visam o monitoramento dos diferentes usos e ocupações da superfície terrestre. O estudo do uso e ocupação da Terra é de grande relevância na engenharia ambiental, uma vez que permite examinar aspectos como o desenvolvimento das cidades e infraestruturas, nível de preservação ou antropização da paisagem, e até mesmo atividades ilegais de desmatamento e mineração (IBGE, 2013).

A plataforma MapBiomias possui papel fundamental na classificação e distribuição de dados sobre o uso e ocupação do solo em território brasileiro (SOUZA JR et al., 2020). O projeto MapBiomias é uma iniciativa multi-institucional gratuita e colaborativa, composta por universidades, startups de tecnologia e ONGs, que por meio do uso de métodos avançados de processamento em nuvem, tecnologia operacional e *big data* (imagens *Landsat* da plataforma *Google Earth Engine*), concentra os principais dados georreferenciados acerca dos biomas brasileiros e transições na paisagem. O produto mais recente de cobertura do solo da plataforma (Coleção 7), que contempla dados de 1985 a 2020, utiliza imagens de diversos satélites, tais como Landsat 5 TM, Landsat 7 ETM+, Landsat 8 OLI, Sentinel-2A e Sentinel-2B, onde a frequência de aquisição das imagens varia de acordo com a resolução do satélite (MAPBIOMAS, 2022).

Os dados georreferenciados do MapBiomias podem ser visualizados e processados pelos SIGs, tal como o *software* QGIS, entretanto a coleta e análise de séries temporais de uso e ocupação do solo necessitam de pré-processamento, etapa que se realizada em *softwares* convencionais, como o QGIS, e exige do usuário diversas tarefas manuais e repetitivas, além de recursos como computadores potentes, salas com ambientação e mão de obra qualificada.

Nos estudos de progressão do uso e ocupação do solo, o pré-processamento sempre é conduzido de maneira similar, seguindo a mesma sequência lógica de passos: conexão com a base de dados, definição do recorte, reprojeção e aplicação de uma máscara, transformação dos dados em tabela, cálculo de estatísticas zonais, resumo e visualização dos dados através de técnicas estatísticas. Na abordagem convencional, muitos passos são repetidos, imagem a imagem, inflacionando o tempo gasto em tarefas manuais e a possibilidade de erros.

Há, portanto, demanda para a automação da análise de séries históricas de uso e ocupação do solo, que possa eliminar quase por completo o trabalho manual. Aqui, a primeira versão do programa *Bioma Stats* é apresentada. O *Bioma Stats* é um programa computacional implementado em linguagem de programação R, *open source* e de livre distribuição (R CORE

TEAM, 2021; RSTUDIO TEAM, 2021) que visa facilitar a obtenção e o tratamento estatístico das séries de uso e ocupação do solo nos biomas brasileiros. Suas funções permitem automatizar o cálculo de estatísticas zonais e fornecer análises ambientais customizadas a partir de dados obtidos pela plataforma MapBiomas.

2. OBJETIVOS

O objetivo geral foi introduzir uma nova ferramenta SIG para automatizar a elaboração de relatórios ambientais.

Os objetivos específicos do projeto consistiram em:

- Estudar a estrutura de dados georreferenciados e aplicações na análise do uso e ocupação do solo no território brasileiro
- Implementar um algoritmo capaz de agilizar a extração e o processamento de dados de uso e ocupação do solo através da linguagem R
- Fornecer um pacote computacional R a pesquisadores da área ambiental, promovendo análises de uso e ocupação do solo automatizadas.

3. FUNDAMENTAÇÃO TEÓRICA

a. Geotecnologia

Geotecnologia é a área do conhecimento que realiza o tratamento de dados georreferenciados a partir de técnicas matemáticas e computacionais, auxiliando diversas áreas do conhecimento, tais como: Cartografia; Energia e Planejamento Urbano; Transporte; Análise de Recursos Naturais; Agricultura; entre outros (CÂMARA et al., 2001). Dados espaciais se referem a qualquer tipo de informação que tem uma componente geográfica ou espacial, ou seja, que pode ser relacionada a uma posição no espaço. Isso pode incluir dados de satélite, mapas, imagens de sensoriamento remoto, entre outros (ARONOFF, 1989; BORGES, 1997). Desta forma, dados georreferenciados são dados espacializados que possuem uma localização geográfica específica, geralmente vinculados a um sistema de referência espacial e representados por coordenadas de latitude de longitude (CONCAR, 2010). Dentre as formas de

dados geoespaciais, destacam-se os dados vetoriais e matriciais (rasters) (TENNEKES and NOWOSAD, 2021).

Segundo Calles (2021), dados vetoriais são baseados em geometria vetorial, que utiliza de pontos, linhas e polígonos para descrever objetos e suas características no mundo real. Esses dados possuem propriedades relacionadas ao tamanho, forma, cor e espessura de um objeto, e podem ser armazenados em diferentes formatos de arquivo, como: shapefiles, KML, GeoJSON, entre outros. A aplicação de dados vetoriais na Geotecnologia é variada, incluindo limites de propriedades, de Áreas de Proteção Permanente (APPs), de Unidades de Conservação, rios, redes de transporte, mapas de cartografia, entre outros elementos que podem ser definidos a partir de suas geometrias e representados por pontos, linhas e polígonos.

Já dados matriciais, ou rasters, são representados a partir da organização de informação através de uma grade regular de células ou pixels, formando uma matriz bidimensional ou tridimensional. Cada célula representa uma pequena área no espaço, variando sua área de acordo com a resolução da imagem, e o valor armazenado indica uma propriedade ou característica do local representado pela célula. Dados matriciais são utilizados para representar fenômenos contínuos, tais como: elevação do terreno; temperatura; umidade; cobertura vegetal; e outros atributos que podem ser medidos em cada ponto do espaço (CÂMARA, 2003). A obtenção desses dados, geralmente, ocorre por meio de registros de sensores remotos, em aviões, drones e satélites, que capturam as informações sobre a superfície da Terra em forma de imagens e armazenam em diferentes formatos de arquivos (TIFF, JPEG, GeoTIFF, entre outros) (CÂMARA and MONTEIRO, 2005).

No Brasil, a Geoinformática tem sido amplamente utilizada em aplicações como o monitoramento de desmatamento (MAEDA et al., 2008; SZILASSI et al., 2006; JAIN et al., 2001), a gestão de recursos hídricos (MILLER et al., 2007; FERRAZ et al., 1999) e o planejamento urbano (PRENZEL, 2004; FASA and CHAN, 2000). Além disso, também é importante para a identificação de áreas vulneráveis a desastres naturais (EMBRAPA, 2014) e para a implementação de políticas públicas (SOARES e ESPINDOLA, 2008).

O geoprocessamento é realizado através dos SIGs, constituídos de ferramentas e técnicas para coletar, armazenar, processar, analisar e visualizar os dados georreferenciados (BURROUGH and MCDONNELL, 1998). Diversas informações podem ser encapsuladas em SIGs, tais como informações geográficas e mapas, dados de sensores e imagens, tabelas de

variáveis assessórias, informações demográficas, ecológicas e socioeconômicas (REIBEL, 2007).

Os SIGs são ferramentas importantes para a tomada de decisão em engenharia ambiental, uma vez que possibilitam a visualização de dados importantes no espaço, permitindo o diagnóstico ambiental e a modelagem; eles também permitem a simulação de cenários futuros.

b. Software R

Segundo Dorman (2022), o R pode ser utilizado como um Sistema de Informação Geográfica (SIG), pois conta com pacotes e recursos de visualização e processamento de dados georreferenciados, além de vantagens relevantes em tarefas que envolvam automação e reprodutibilidade, principalmente, quando comparado com abordagens tradicionais de SIG, tais como os softwares QGIS e ArcGIS (ARCGIS, 2022).

O processamento de dados georreferenciados através da linguagem R tomou grandes proporções após a criação do pacote *sp*, disponível em repositório CRAN desde 2005 (PEBESMA and BIVAND, 2005; PEBESMA and BIVAND, 2020). O pacote *sp* contém classes e métodos para a leitura e manipulação de dados espaciais, tais como: linhas, pontos, polígonos e matrizes espaciais (*raster*); para dados bidimensionais (2D) e tridimensionais (3D), o pacote contém classes que determinam informações referentes à localização espacial dos dados (x e y para 2D, e x , y e z para dados 3D), enquanto os métodos possibilitam a seleção dos dados, o alcance de coordenadas, o sistema de referência de coordenada dos dados (seja graus longitude/latitude, ou zona UTM), bem como a reprojeção do sistema de referência de coordenadas, visualização, impressão, entre outros (PEBESMA and BIVAND, 2005; BIVAND et al., 2013).

Pacotes como *rgdal*, *rgeos*, *mapview*, *ggplot* e *plotly*, por exemplo, dão suporte para o pacote *sp* na manipulação e visualização de dados espaciais (BIVAND, 2021). O pacote *rgdal* se encarrega de auxiliar na entrada/saída de arquivos vetoriais e *rasters* e nos processos relacionados à projeção e/ou transformação do sistema de referência de coordenada (BIVAND et al., 2020). Já o pacote *rgeos* realiza tarefas que envolvem alterações estruturais na geometria dos dados, em especial, vetores (BIVAND and RUNDEL, 2020). O pacote *mapview* possui funções capazes de fornecer a visualização interativa de mapas baseados em objetos espaciais (APPELHANS et al., 2020). Enquanto isso, os pacotes *ggplot2* (WICKHAM, 2016) e *plotly* (SIEVERT, 2020) são utilizados como base para a visualização estática e dinâmica, respectivamente, de gráficos de uso e ocupação do solo. Além dos pacotes listados acima, o

processamento dos dados em formato de séries temporais é possível a partir da utilização do pacote *dplyr*, capaz de filtrar e organizar os dados de matrizes espaciais em tabelas, possibilitando a criação dos gráficos estáticos e dinâmicos (WICKHAM et al., 2022).

c. Estatística Zonal

Em Geotecnologia, é comum a divisão do território analisado em “zonas” de avaliação. Essa divisão pode ocorrer por diversos critérios, como limites administrativos, temáticos, demográficos ou, até mesmo, por classificação de vegetação ou outro atributo da paisagem. Então, após a subdivisão do território, as zonas são avaliadas individualmente e seus dados são resumidos por estatísticas zonais.

Em seu estudo, Tomlin (1990) explicou sobre as operações zonais mais comuns, sendo: mínimo, máximo, maioria, média, soma e proporção zonal. A estatística zonal necessita de dois mapas de entrada, um contendo os atributos que se deseja obter a estatística, geralmente no formato *raster* e o outro mapa que contenha regiões, ou zonas, geralmente no formato *shapefile*. O produto desta operação será uma base de dados tabular onde para cada zona é gerado um conjunto de estatísticas (valor mínimo, máximo, média, desvio padrão, variância, soma de área, entre outros) obtidas a partir dos valores (atributos) do *raster* dentro de cada subdivisão (zona).

O conceito teórico de zonalidade para análise espacial, cujas restrições espaciais são delimitadas por áreas, apresenta dupla importância: possibilita a materialização em um SIG dos conceitos de paisagem (TRICART, 1977; TRICART and KIEWIETDEJONGE, 1992) e área-unidade (HARTSHORNE, 1978) e possibilita a conexão entre dados do meio físico-biótico e dados socioeconômicos, extremamente necessário para tarefas de ordenação territorial (BARBOSA, CLÁUDIO CLEMENTE et al., 1998).

As estatísticas zonais são calculadas nas mais variadas áreas da Ciência, como por exemplo na Ecologia, para estudar a relação da vegetação e da temperatura no processo de urbanização (JENERETTE et al., 2011; JENERETTE et al., 2007); Demografia e Meio Ambiente (HAYNES et al., 2015; HAYNES et al., 2017); e interpolação de áreas populacionais e controle de queimadas a partir de dados de uso e ocupação do solo (REIBEL, Michael et al., 2007; GHORBANZADEH, OMID et al., 2019), respectivamente.

4. MATERIAIS E MÉTODOS

a. Materiais

Os dados geográficos (unidades federativas e recorte dos biomas) foram obtidos a partir do banco de dados do Instituto Brasileiro de Geografia e Estatística (IBGE, 2022), enquanto os dados de uso e ocupação do solo do Projeto MapBiomas, compreendendo o período de 1985 a 2020, foram obtidos no formato de arquivos geoTIFF através do site do projeto MapBiomas – coleção 7.0 (MAPBIOMAS, 2022).

TABELA 1 - Fonte dos dados usados no programa *Bioma Stats*.

Nome	Formato	Escala	Resolução	Fonte
Uso e Cobertura do Solo (1985 e 2020)	Raster	1:100.000	30 m	MapBiomas (2022a)
Unidades Federativas do Brasil	Vetor	1:250.000	-	IBGE (2022)
Recorte Biomas do Brasil	Vetor	1:250.000	-	IBGE (2022)

Fonte: Autoria própria, 2023.

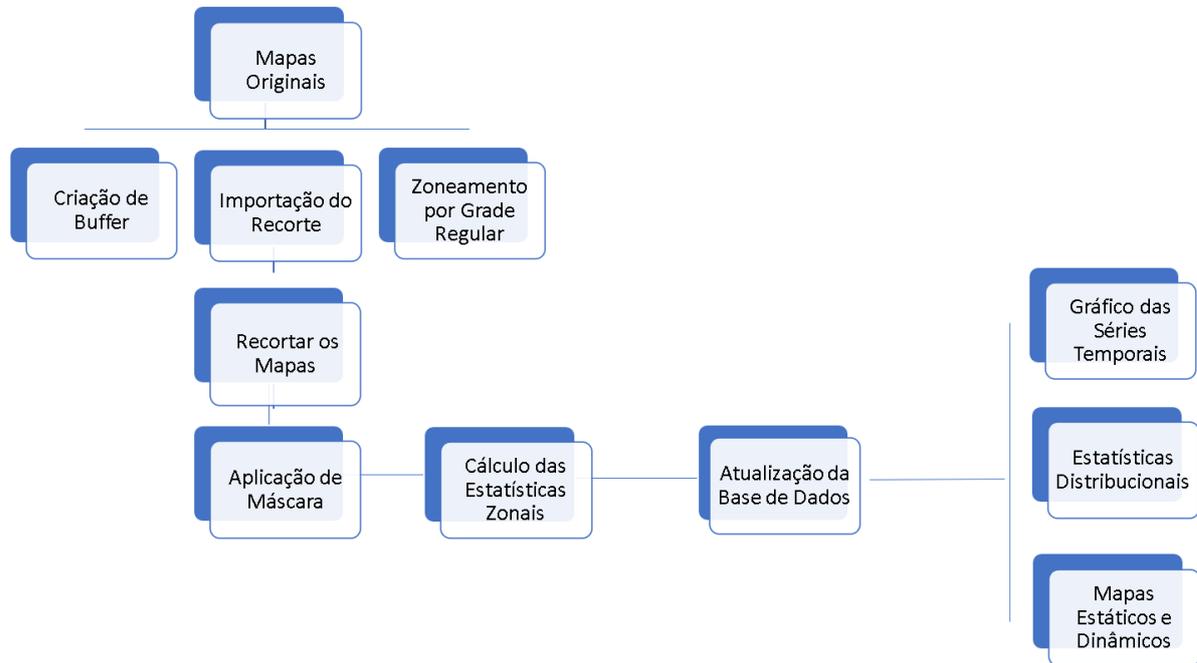
b. Programa Bioma Stats

O programa permite que o usuário faça o zoneamento do recorte de interesse e escolha a região-alvo de análise. Para o zoneamento, o usuário possui três opções: arquivos vetoriais (*shapefile*) de qualquer recorte em território nacional brasileiro; gerador automático de polígonos circulares, hexagonais e quadrados, centrados nas coordenadas fornecidas pelo usuário; e grades hexagonais de diferentes dimensões. O usuário também escolhe as classes de uso e ocupação do solo que pretende monitorar e o horizonte de tempo a ser avaliado. Ao final do processamento, o *Bioma Stats* emitirá como saída um relatório completo do uso e ocupação do solo e gráficos das séries temporais.

O *Bioma Stats* foi implementado em linguagem computacional R (R CORE TEAM 2021), com código-fonte aberto e versões de livre distribuição, via pacote computacional disponibilizado publicamente por repositório GitHub (GITHUB, 2022), através do link <<https://github.com/iep-ferreira/bioma-stats>>. O sistema RStudio (RSTUDIO TEAM, 2021) foi utilizado como IDE – Integrated Development Environment para a codificação do pacote.

A obtenção das séries históricas e estatísticas zonais ocorreu conforme o fluxo de trabalho detalhado na figura abaixo:

FIGURA 1 - Fluxo de trabalho do programa Bioma Stats.



Fonte: Autoria própria, 2022.

O fluxo de trabalho tem seu início com a delimitação de um recorte espacial por parte do usuário. Qualquer recorte em território nacional brasileiro pode ser usado como referência, desde que o usuário forneça um arquivo que possa ser lido como objeto espacial através do pacote *sp*. Para o uso do recorte fornecido pelo usuário, basta adicionar o arquivo no diretório de trabalho e informar ao programa seu caminho relativo em relação à pasta raiz do projeto.

O usuário também pode inicializar o programa a partir de coordenadas centrais, e escolher um objeto poligonal ou uma grade regular como recorte. Neste modo, o usuário fornece coordenadas geográficas (longitude e latitude, em UTM), a dimensão do objeto e a sua forma. O programa está habilitado para criar polígonos circulares, hexagonais e quadrados, centrados nas coordenadas fornecidas; e os polígonos são automaticamente salvos em uma subpasta do diretório de trabalho.

O modo “*grid*” permite associar um recorte qualquer a zoneamentos regulares na forma de colmeia (hexágonos). Neste modo, o usuário opta por usar recortes das unidades administrativas brasileiras, disponíveis através do próprio *Bioma Stats*, ou fornecer recortes de interesse a partir de documentos vetoriais (*shapefiles*). Então, o recorte selecionado é zoneado em hexágonos automaticamente a partir de suas coordenadas centrais. O usuário pode escolher

o diâmetro dos hexágonos (em km), sendo essa dimensão de referência ajustada para fornecer polígonos em coordenadas geográficas (UMT, em graus).

Tanto no modo que cria objetos poligonais a partir das coordenadas centrais quanto no “*grid*”, as dimensões dos polígonos são ajustadas pelo método *haversine*, o qual ajusta os resultados a partir da longitude (LONGLEY et al., 2015). Todos os objetos são tratados no sistema de coordenadas cartográficas UTM (*Universal Transverse Mercator*) considerando o sistema de referência geodésico WSG84 (*World Geodetic System 1984*). Logo após a criação de um objeto ou carregamento de um recorte, externamente, o usuário pode visualizar a área de interesse a partir de funcionalidades do pacote “*mapview*”. A visualização é dinâmica e o usuário pode usar os recursos de “*zoom in*” ou “*zoom out*” para ampliar ou diminuir a janela de visualização.

Após a definição dos recortes, o programa inicia o carregamento automático dos mapas provenientes da plataforma MapBiomias. Os mapas podem ser mantidos em servidor local ou serem carregados através de nuvem, via *url*. A equipe de desenvolvimento do Centro de Modelagem Estatística em Ciências Ambientais da UFSCar Lagoa do Sino (CeMECA - CCN) pretende disponibilizar um repositório com mapas fragmentados de acordo com as unidades administrativas do Brasil, bem como o código fonte do programa, facilitando o seu carregamento e, assim, diminuindo o tempo para *download* dos arquivos. A distribuição dos mapas será feita através de um novo protocolo de compartilhamento P2P (peer-to-peer) descentralizado, bioinspirado por colmeia de abelhas (FONTANA et al., em elaboração).

Após a leitura dos mapas, o programa *Bioma Stats* seleciona os mapas apenas na região de interesse, e depois aplica uma máscara para considerar apenas os pixels presentes dentro dos polígonos que definem o recorte de interesse. O processo é automático, sendo repetido para todos os mapas de cobertura, ano a ano, sem qualquer intervenção do usuário. Nessa etapa, o usuário fornece apenas o horizonte de tempo considerado no estudo, e recebe como resposta do sistema todos os mapas já recortados para a etapa de processamento dos dados.

Na etapa posterior, o programa recebe os dados, o horizonte de tempo e o polígono (*feature*) a ser analisado e devolve ao usuário uma tabela completa, com o cálculo de área para cada classificação de uso e ocupação e ano. No cálculo de área, o tamanho de cada *pixel* é ajustado automaticamente, considerando a longitude e latitude dos mesmos, pois em projeção geográfica os *pixels* mais periféricos apresentam maiores distorções. O ajuste da área é feito através da função “*área*” do pacote *raster* (HIJMANS, 2022).

O ajuste automático da área possibilita o cálculo de estatísticas zonais com mais rapidez e fluidez: não é necessário adotar projeções métricas que afetariam os contornos do mapa e aumentariam drasticamente o tempo de processamento; e as áreas são registradas de forma fiel, independente de utilizar o sistema UTM para a visualização dos *rasters*; a funcionalidade permite que o usuário analise recortes continentais, regionais e locais da mesma maneira, sem a preocupação de saber como a escala de observação afeta a qualidade dos resultados e qual projeção seria a mais adequada para o cálculo da área para o recorte escolhido em particular.

Após a etapa de processamento de dados, os mapas e tabelas de estatística zonais são enviados ao módulo de visualização de dados. Por meio deste, é possível visualizar mapas de uso e ocupação com alta qualidade, com cores previamente escolhidas por especialistas na área e dicionário importado da plataforma MapBiomas para a confecção automática das legendas. Os mapas são apresentados no modo estático ou dinâmico: no modo estático, o usuário visualiza o mapa de acordo com o ano escolhido, e para a sua confecção o programa utiliza recursos do pacote *ggplot2* (WICKHAM, 2016). No modo dinâmico, o usuário interage com o mapa a partir de um *slider*, selecionando o ano de interesse; e também é possível clicar no botão *play* para assistir uma animação da progressão espaço-temporal do uso e ocupação do solo. O modo dinâmico de visualização é confeccionado com recursos do pacote *plotly* para o R (SIEVERT, 2020).

O programa *Bioma Stats* também processa dados em formato de séries temporais. Através de recursos do pacote *dplyr*, pipelines são criados para filtrar e organizar os dados de matrizes espaciais em tabelas, o que permite a análise temporal a partir de gráficos de perfis estáticos e dinâmicos (WICKHAM et al., 2022). O sistema permite reclassificação de usos, como da cobertura natural preservada, e o estudo da distribuição dos atributos da paisagem ao longo do tempo.

5. RESULTADOS E DISCUSSÃO

Há três formas de iniciar o programa: criando um polígono com a função “*make_polygon*”, carregando um *shapefile* através da importação de um arquivo específico ou selecionando o formato de grades hexagonais (*grid*). As figuras abaixo exemplificam as opções que o usuário poderá utilizar.

FIGURA 2 - Polígono circular.



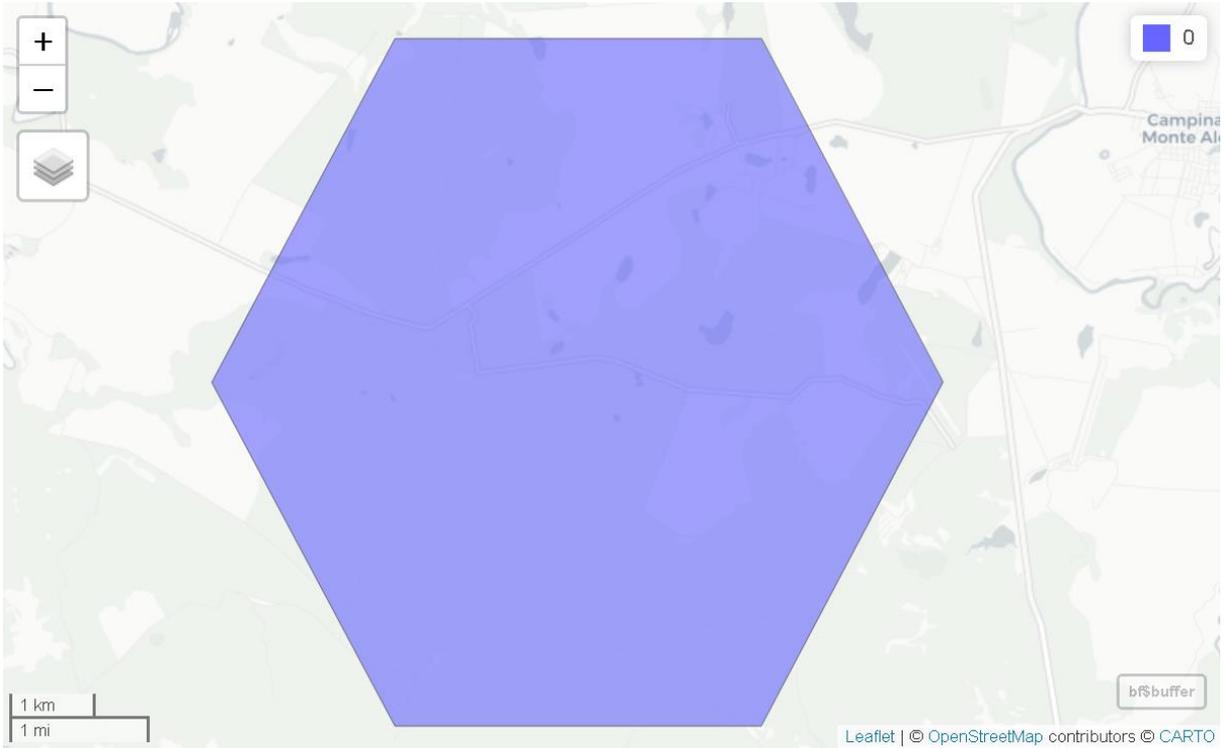
Fonte: Autoria própria, 2023.

FIGURA 3 - Polígono quadrado.



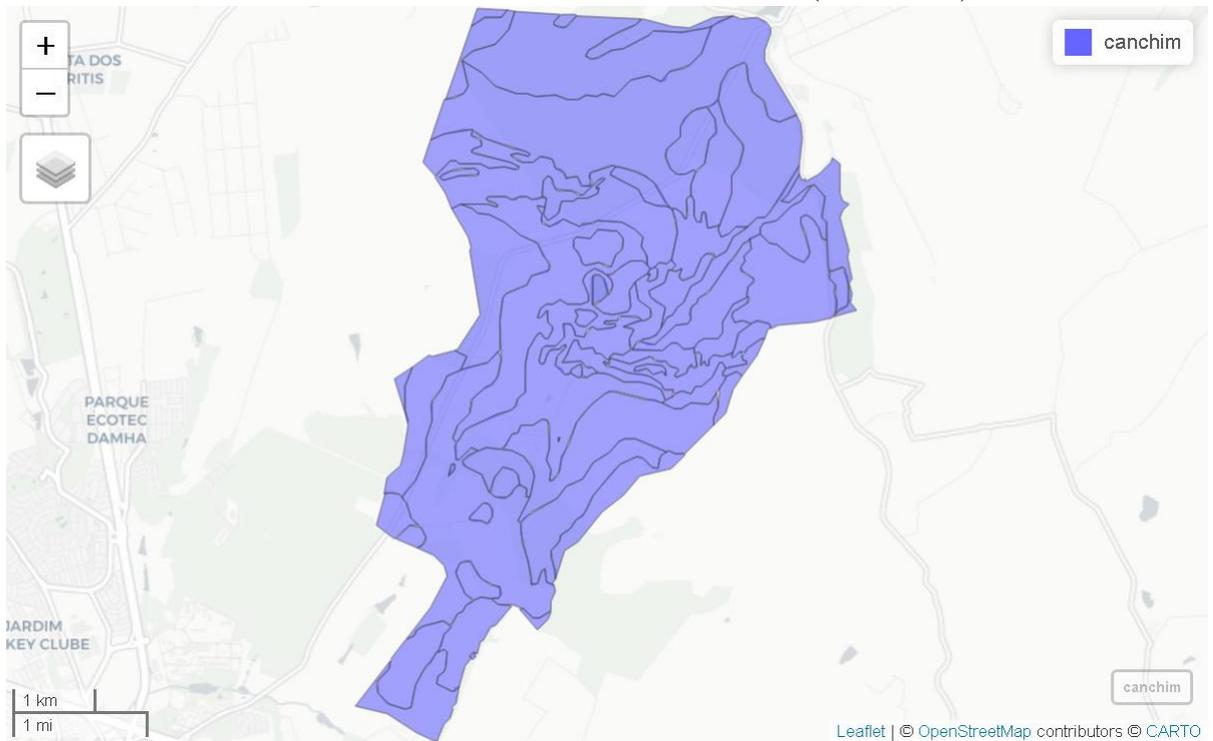
Fonte: Autoria própria, 2023.

FIGURA 4 - Polígono hexagonal.



Fonte: Autoria própria, 2023.

FIGURA 5 - Contorno da Fazenda Canchim (São Carlos).



Fonte: Autoria própria, 2023.

Para criar um objeto poligonal, o usuário deve informar a latitude e longitude centrais, a forma do polígono e a sua dimensão em quilômetros. Para inicializar os cálculos, o usuário deverá escolher o caminho do arquivo *shapefile*, o qual contém o recorte desejado. *Shapefiles* específicos podem ser importados para a pasta “./shape-example/”. O *shapefile* criado automaticamente pela função “*make_polygon*” será armazenado nesta pasta com o nome “*polygon.shp*”.

A fim de exemplo, o caso a seguir representou a aplicação da primeira forma de inicializar o programa, ou seja, criando um polígono hexagonal a partir das coordenadas centrais, além dos demais processamentos.

a. Caso 1 – Hexágono nos arredores da propriedade da UFSCar Lago do Sino.

Abaixo encontra-se as funções utilizadas para o processamento dos dados, bem como o detalhamento:

```
bf <- make_polygon(lat = -23.62, lon = -48.55, size = 5, shape = "hexagon"
)
```

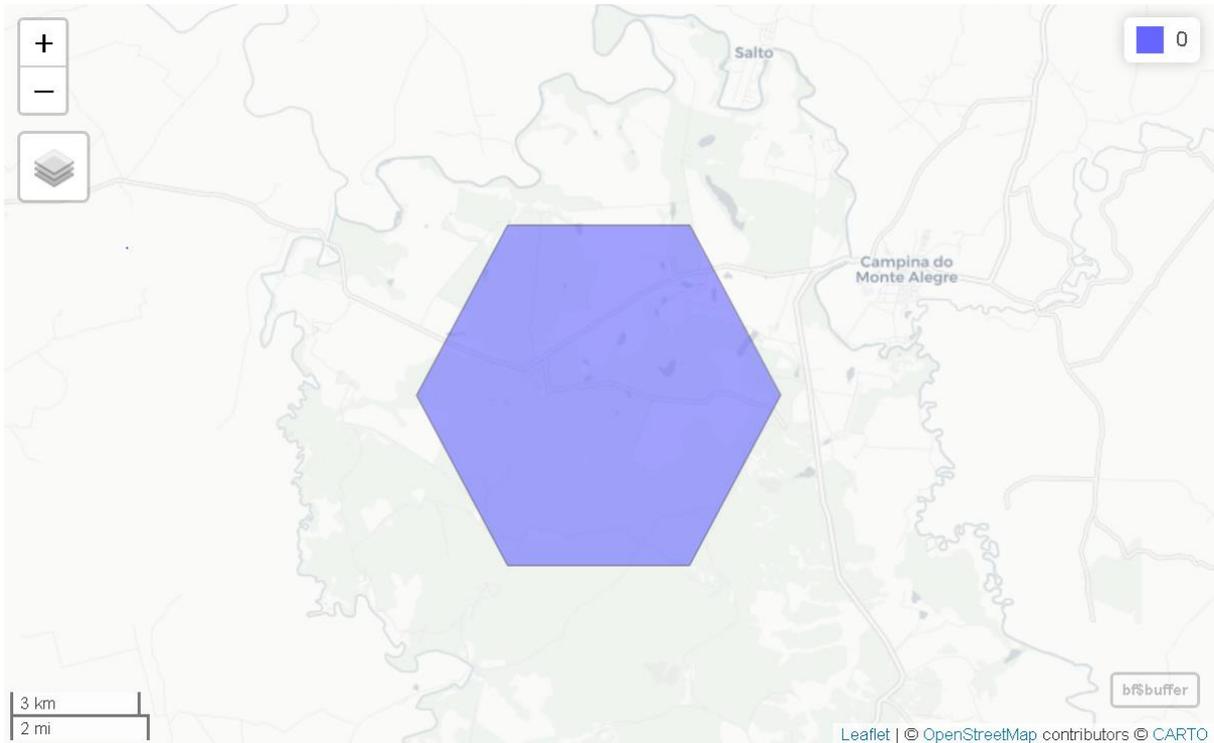
A função acima cria um polígono hexagonal com o diâmetro de 5km, centrado nas coordenadas que caracterizam a localização geográfica da UFSCar campus Lagoa do sino, onde situa-se o CeMECA.

Após criado o polígono, o comando “*mapview*” é utilizado para a visualização interativa do objeto criado.

```
mapview(bf$polygon)
```

A Figura 6 refere-se ao polígono hexagonal criado pela função acima. A partir do comando “*mapview*”, o usuário fica livre para manusear o objeto da maneira que quiser, podendo aproximar “*zoom in*” ou afastar “*zoom out*”, bem como arrastar o objeto para os lados.

FIGURA 6 - Polígono hexagonal nos arredores da UFSCar – Lagoa do Sino.



Fonte: Autoria própria, 2023.

A função abaixo define o caminho para o *shapefile* criado.

```
shape_path <- "./shape-example/polygon.shp"
```

O código acima está atribuindo o caminho do arquivo *shapefile* “*polygon.shp*” à variável “*shape_path*”. Mais especificamente, o caminho do arquivo *shapefile* é especificado como “*./shape-example/polygon.shp*”, que significa que o arquivo “*polygon.shp*” está localizado na subpasta “*shape-example*” do diretório atual (representado por “.”).

Após definido o caminho para o arquivo, inicia-se o recorte automático dos *rasters*, utilizando como máscara o *shapefile* criado.

```
recortes <- load_rasters(shape_path = shape_path)
```

A função “*load_rasters*” tem como objetivo recortar os arquivos *rasters* com base na extensão espacial do arquivo *shapefile* especificado anteriormente. O resultado dessa operação é que a variável “recortes” agora contém os dados *rasters* recortados, que correspondem à extensão espacial do arquivo *shapefile* especificado em “*shape_path*”.

Em seguida, realizam-se os cálculos das estatísticas zonais.

```
df_zonal <- get_area(data = recortes)
```

A função “*get_area*”, que recebe o objeto de dados “*data*”, tem como objetivo aplicar a estatística zonal para calcular a área ocupada por diferentes classes de uso da terra ao longo do tempo para os *rasters* recortados. O resultado dessa operação é que a variável “*df_zonal*” agora possui uma lista que inclui as informações de anos (1985 a 2020), área ocupada (uso e ocupação do solo), dados agregados em “*dataframe*” e um gráfico de progressão criado.

A partir da função abaixo, é possível realizar a impressão da tabela de cálculo de área.

```
head(df_zonal$aggregate_data)
```

O código extrai as primeiras linhas do objeto de dados chamado “*aggregate_data*”, o qual agrega informações sobre a área ocupada por cada classe de uso da terra em cada ano e seus respectivos “*ID’s*”, e que está contido no “*dataframe*” chamado “*df_zonal*”. A função “*head()*” é utilizada para exibir as primeiras linhas do objeto, por padrão as primeiras 6 linhas, para que se possa ter ideia de como os dados estão organizados. Além disso, o símbolo “*\$*” é usado para acessar uma coluna específica dentro de um objeto de dados, por exemplo, a coluna “*aggregate_data*”. O resultado deste código é a exibição de uma tabela (tabela 2) com as informações extraídas.

TABELA 2 - Cálculo de área das classes de uso e ocupação.

Land Class	Year	Area (ha)	The legend code ID number of MapBiomias
Forest Formation	1985	11.872	3
Savanna Formation	1985	0.603	4
Forest Plantation	1985	2.862	9
Wetland	1985	1.662	11
Grassland	1985	1.211	12
Pasture	1985	25.823	15

Fonte: Autoria própria, 2023.

Além da tabela gerada acima, é possível criar os seguintes gráficos e mapas:

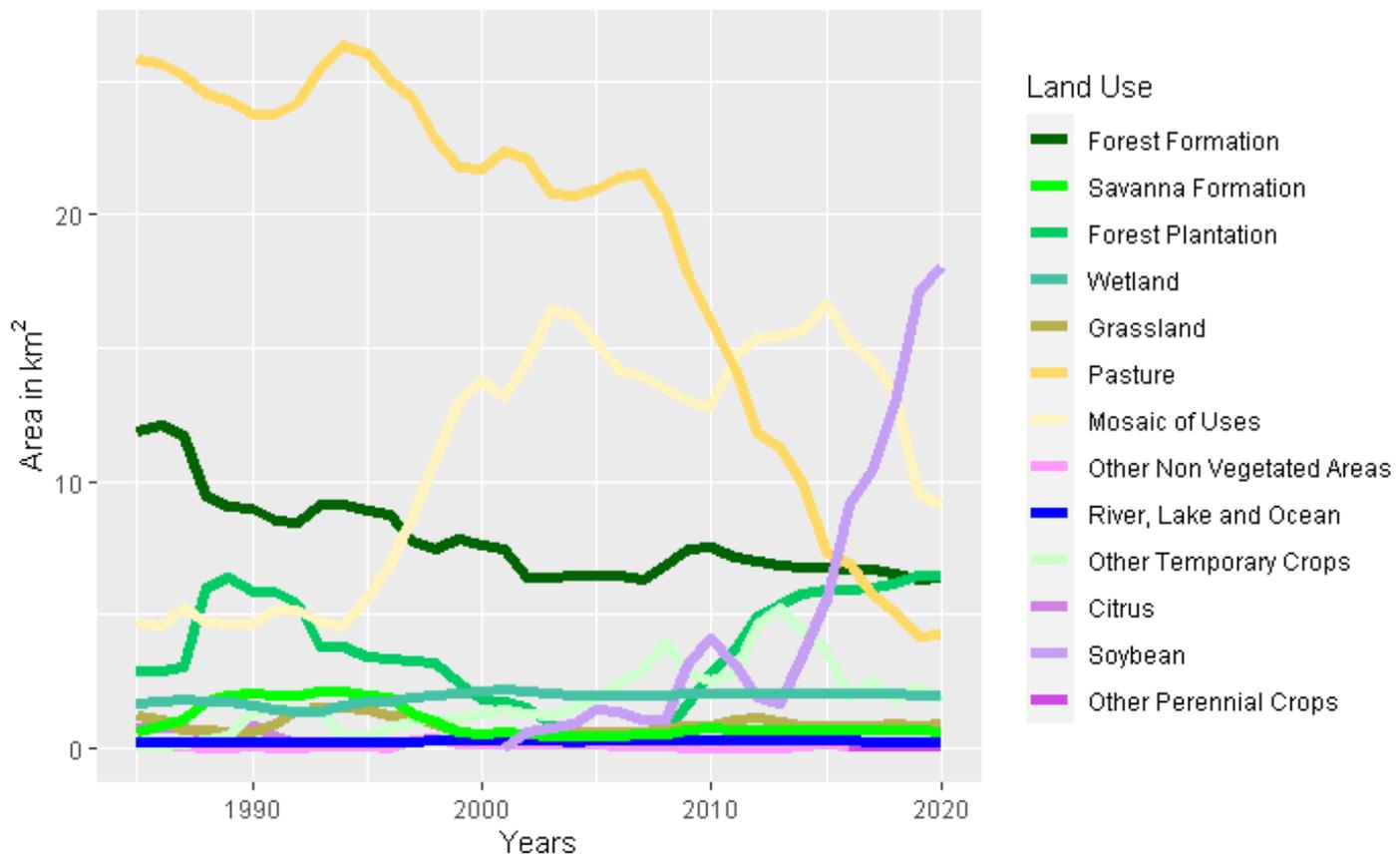
- Gráfico de progressão do uso e ocupação do solo ao longo dos anos.

```
df_zonal$time_series
```

O código gera um gráfico de progressão para o objeto de dados “*time_series*”, contido no *dataframe* “*df_zonal*”. O objeto “*time_series*” faz parte da função pré-definida anteriormente “*get_area*”, que utiliza o pacote *ggplot2*, para criar um gráfico de progressão (*time series plot*) da área ocupada por cada classe de uso da terra ao longo do tempo.

As principais transformações na cobertura do solo possuem como força impulsora as mudanças no uso e ocupação da paisagem. As políticas governamentais, a demanda do mercado internacional e o desenvolvimento econômico territorial levaram à intensificação do uso do solo e, por consequência, à transformação dos ecossistemas naturais em áreas de agricultura (BARBIER 2004; HEYCK 2019; NEPSTAD et al. 2014). A Figura 7 contempla o gráfico de uso e ocupação do solo ao longo do tempo. Com isso, podemos identificar que, ao longo dos anos, o uso do solo relacionado à pastagem diminuiu consideravelmente, entretanto, entre os anos de 2000 a 2010, nota-se o aumento do uso do solo relacionado aos mosaicos de agricultura e pastagem. Além disso, após o ano de 2013, pode-se afirmar que a produção de soja sobre a área de análise cresceu drasticamente, demonstrando uma mudança nos padrões de uso e ocupação da região e um possível aumento do desmatamento relacionado a esta cultura (GIBBS et al. 2015).

FIGURA 7 - Gráfico de Progressão do Uso e Ocupação ao longo do Tempo.



Fonte: A autoria própria, 2023.

Em seus estudos, Gama et al. (2021) utilizaram o método de transição para simular trajetórias de desmatamento até 2030 a partir do software Dinamica EGO. Entretanto, todo o pré-processamento das imagens de satélite, desde a criação e recorte de objetos vetoriais até a classificação do uso e ocupação do solo, teve de ser realizado a partir de softwares convencionais, tais como o ArcGIS, QGIS e o ENVI 5.3. Neste cenário, o *Bioma Stats* constitui uma ferramenta competitiva no pré-processamento, com a possibilidade de pré-processar as imagens de maneira quase que instantânea e agregar os dados necessários para a modelagem ambiental, a serem fornecidos ao dinâmica EGO, por exemplo, eliminando a necessidade de GIS convencionais e diminuindo a taxa de erros decorrentes de processamento manual.

- Mapa de uso e ocupação para um ano em particular.

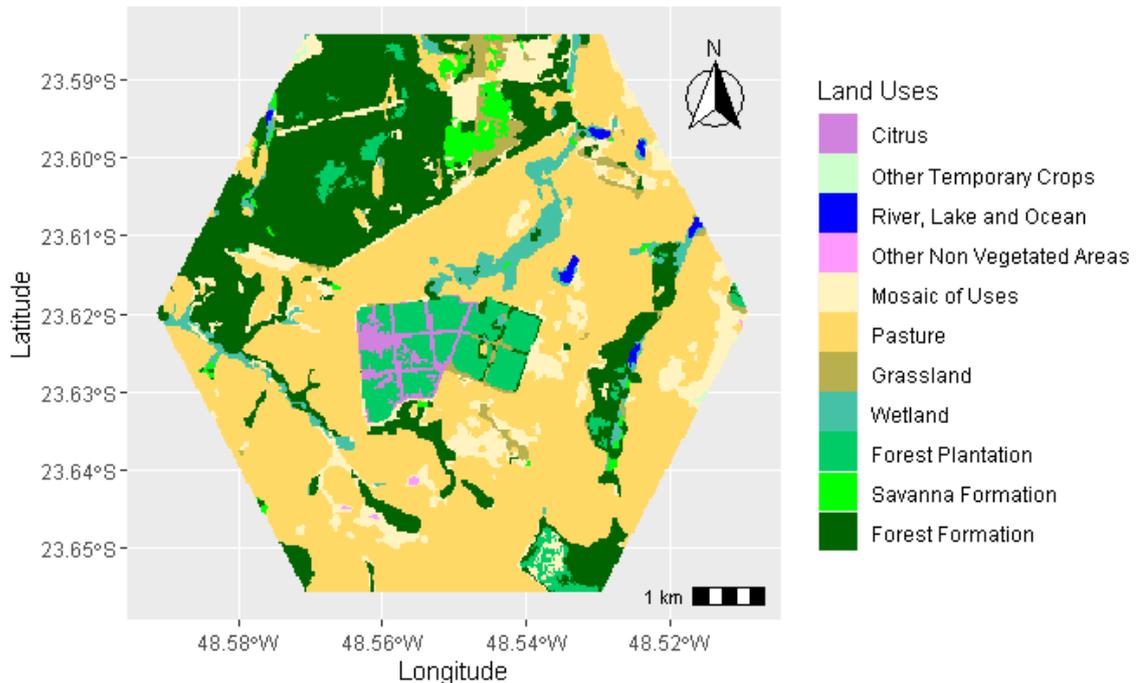
```
land_vis(data = recortes, year = 1986)
```

O código acima representa a função chamada “*land_vis*”, que recebe o objeto “*data*” e “*year*”, tem como objetivo criar um mapa de uso da terra com base no conjunto de recortes fornecidos pela função “*load_rasters*”. A função contém dois argumentos opcionais: “*data*” e “*year*”. O “*data*” é o conjunto de dados *rasters* recortados que contém informações sobre os tipos de uso da terra e “*year*” é o ano para qual o mapa será gerado. Vale ressaltar que, se nenhum conjunto de dados *raster* for fornecido para o argumento “*data*”, a função imprimirá uma mensagem de erro. Além disso, o ano escolhido deve estar entre 1985 e 2020, caso contrário, uma mensagem de erro será impressa.

A Figura 8 representa o recorte hexagonal de uso e ocupação do solo para o ano de 1986. Analisando o mapa obtido, pode-se afirmar que, no ano em destaque, a paisagem do local era predominantemente composta por pastagens, formações florestais e fragmentos de mosaico de agricultura e pastagem.

FIGURA 8 - Mapa hexagonal de Uso e Ocupação do Solo para 1986.

Year: 1986



Fonte: Autoria própria, 2023.

Outros programas de SIG também possuem a capacidade de realizar o processamento dos dados, como o cálculo das estatísticas zonais. Em seu trabalho, Da Silva (2020) utilizou o software QGIS para o mapeamento do uso e ocupação do solo na Bacia Hidrográfica do Rio Água Branca, ao longo do ano de 2018. A lógica para o cálculo das áreas e criação dos mapas é parecida com a lógica implementada ao programa *Bioma Stats*, entretanto boa parte das rotinas de processamento dos dados são feitas de forma manual, ou seja, imagem a imagem, mês a mês, tornando o processo, provavelmente, demorado, custoso e com alta chance de erro humano.

- Distribuição do uso e ocupação em ano específico.

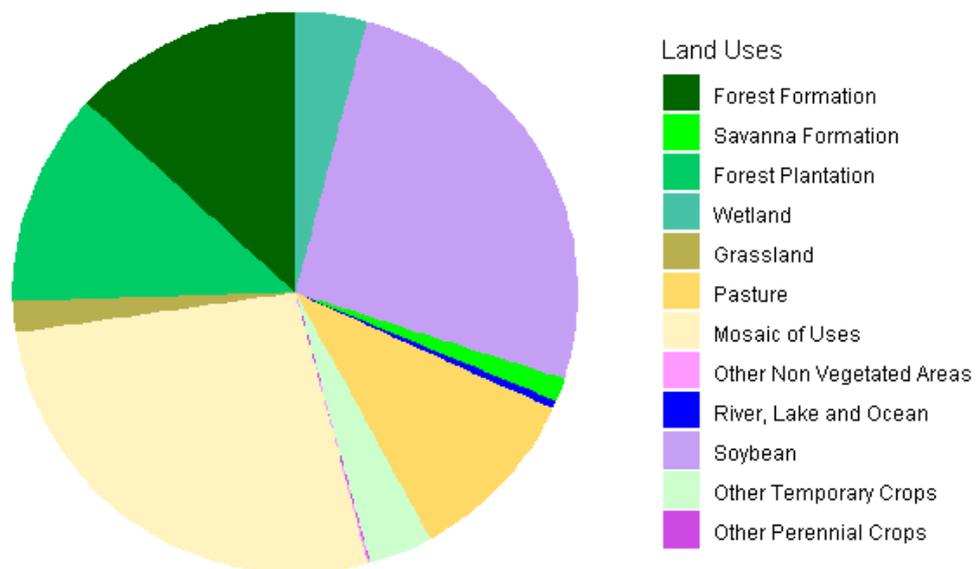
```
land_dist(df_zonal, year=2018, type="pie")
```

O código acima define a função “*land_dist*”, que possui três argumentos: “*data*”, “*year*” e “*type*”. A função utiliza do pacote *dplyr* para a manipulação dos dados. O objetivo da função é gerar um gráfico que mostre a distribuição da área de diferentes tipos de uso e ocupação do solo em um determinado ano. Sendo assim, o argumento “*data*” se remete ao processamento das informações sobre o uso da terra em diferentes anos (*df_zonal*), enquanto o argumento “*year*” é o ano para o qual deseja-se gerar o gráfico. Por fim, o argumento “*type*” especifica o tipo de gráfico a ser gerado, ou seja, gráfico de pizza (*pie*) ou gráfico de pizza (*barplot*).

A Figura 9 refere-se ao gráfico de pizza do uso e ocupação do solo para o ano 2000. Desta forma, podemos evidenciar que, ao passo que o uso relacionado à pastagem e formação florestal diminui, há um aumento expressivo do uso relacionado à mosaico de agricultura e pastagem e a produção de soja, possivelmente relacionado ao desmatamento de florestas para o plantio de culturas agrícolas.

FIGURA 9 - Gráfico de pizza do Uso e Ocupação do Solo em 2018.

Year: 2018



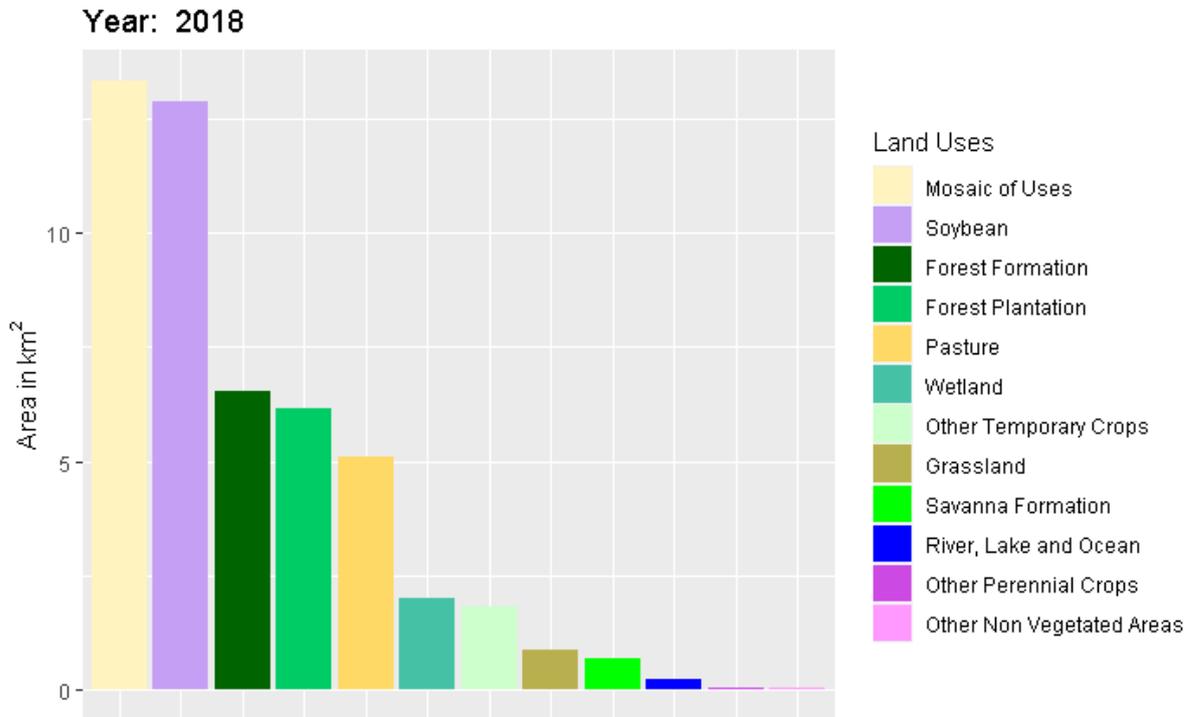
Fonte: Autoria própria, 2023.

- Distribuição do uso e ocupação em ano específico.

```
land_dist(df_zonal, year=2018, type="barplot")
```

A Figura 10 retrata o gráfico de barras do uso e ocupação do solo para o ano de 2015. Com isso, fica evidente a forte pressão exercida pela classificação referente ao uso e ocupação de mosaico de agricultura e pastagem. Entretanto, nota-se também que, há um grande aumento no uso do solo relacionado à produção de soja, possivelmente relacionado à rotação das culturas agrícolas ao longo dos anos.

FIGURA 10 - Gráfico de barras do Uso e Ocupação do Solo em 2018.



Fonte: Autoria própria, 2023.

6. CONSIDERAÇÕES FINAIS

O programa *Bioma Stats* permite a visualização de mapas e progressões de uso e ocupação do solo em poucos cliques. A abrangência é de todo o território nacional, e os mapas são produzidos seguindo as classificações, cores e legendas padrão da plataforma MapBiomias. O programa permite ao usuário definir recortes e baixar automaticamente os dados. Ajustes manuais não são necessários e, a depender do recorte, em poucos segundos a visualização é gerada. O usuário também pode exportar tabelas de atributos e estatísticas zonais. O programa é código-fonte aberto e permite modificações, além da livre distribuição sob a licença GNU GLP v3.0.

A utilização da linguagem de programação R para a construção do programa *Bioma Stats* apresentou bom desempenho. Além disso, a utilização de linguagem de programação para fins de geotecnologia se mostrou viável, podendo ser aplicável para diversas áreas do conhecimento.

O código do programa executou todos os processamentos e os resultados demonstraram grande relevância, podendo ser uma boa alternativa para a otimização de tempo em processamento de dados especializados.

Acredita-se que utilização do programa *Bioma Stats* desempenhará papel importante no mapeamento do uso e ocupação nos biomas brasileiros, além de auxiliar nas tomadas de decisão, pois tais ferramentas utilizadas no processamento poderão orientar na criação de políticas públicas de combate ao desmatamento, na conservação do meio ambiente, na agricultura de precisão, na pecuária, entre outros.

O *Bioma Stats* encontra-se em fase de prototipagem e ainda não possui Interface Gráfica de Usuário (GUI – Graphical User Interface). No momento, todas as funções são carregadas pelo usuário a partir de linhas de comando, em linguagem de programação R. Dessa forma, o programa encontra-se acessível somente aos usuários que possuem conhecimentos prévios em linguagem de programação R. Futuramente, Interface Gráfica do Usuário será construída via aplicativo *shiny*, priorizando garantir a melhor experiência ao usuário, com recursos interativos, organizados e gerados automaticamente. Assim, a partir de um clique, o usuário poderá obter um relatório detalhado acerca do uso e ocupação da paisagem em determinada localidade do Brasil.

O programa *Bioma Stats* é uma iniciativa do CeMECA (Centro de Modelagem Estatística em Ciências Ambientais), do CCN – UFSCar, como projeto de extensão colaborativo e sem fins lucrativos. A expectativa é implantar em breve módulos para: a análise de correlações seriais; cálculo de matrizes de transição e estudo por redes complexas; modelagem das séries temporais e previsão; análise do impacto de variáveis econômicas e socioambientais. Espera-se que o *Bioma Stats* seja uma ferramenta de impacto na área ambiental, ao contribuir para estudos do uso e ocupação do solo para os territórios brasileiros e gerar informações úteis para a gestão ambiental e na construção de políticas de desenvolvimento sustentável.

REFERÊNCIAS

ARONOFF, I. **Geographical information system: management perspective**. Ottawa: WDL Publications, 1989.

APPELHANS, T; DETSCH, F; REUDENBACH, C; WOELLAUER, S (2020) **mapview: interactive viewing of spatial data in R**. <https://CRAN.R-project.org/package=mapview>, R package version 2.9.0.

BARBIER, Edward B. Agricultural expansion, resource booms and growth in Latin America: implications for long-run economic development. **World Development**, v. 32, n. 1, p. 137-157, 2004.

BARBOSA, CLÁUDIO CLEMENTE et al. Operadores zonais em álgebra de mapas e sua aplicação a zoneamento ecológico-econômico. **Simpósio Brasileiro de Sensoriamento Remoto, IX**, 1998.

BIVAND, Roger S. Progress in the R ecosystem for representing and handling spatial data. **Journal of Geographical Systems**, v. 23, n. 4, p. 515-546, 2021.

BIVAND, RS; PEBESMA, E; GOMEZ-RUBIO, V., (2013). Applied spatial data analysis with R, Second edition. **Springer**, NY. <https://asdar-book.org/>.

BIVAND, R; KEITT, T; ROWLINGSON, B (2020) **rgdal: bindings for the 'geospatial' data abstraction** library. <https://CRAN.R-project.org/package=rgdal>, R package version 1.5-16.

BIVAND, R; RUNDEL, C (2020) **rgeos: interface to geometry engine—open source ('GEOS')**. <https://CRAN.R-project.org/package=rgeos>, R package version 0.5-3.

BORGES, K. A. B. **Uma extensão do modelo OMT para aplicações geográficas**. Dissertação (Mestrado em Administração Pública)—Fundação João Pinheiro, Belo Horizonte, 1997. Disponível em: <<http://www.dpi.inpe.br/cursos>> , 2002. Acesso em: 22 set. 2022.

CALLES, Michael Branion. A Crash Course in Geographic Information Systems (GIS) using R. In: CALLES, Michael Branion. **Chapter 3 Geographic Vector and Raster Data with R**. : Ações Gh, 2021. Cap. 3. p. 11-13.

CÂMARA, Gilberto; DAVIS, Clodoveu; MONTEIRO, Antônio Miguel Vieira. **Introdução à ciência da geoinformação**. 2001.

CÂMARA, Gilberto. Introdução a Sistema de Informação Geográfica. In: CÂMARA, Gilberto. Representação computacional de dados geográficos. Brasil: Researchgate, 2005. p. 35. Disponível em: https://bookdown.org/michael_bcalles/gis-crash-course-in-r/data.html Acesso em: 20 jan. 2023.

CÂMARA, Gilberto; MONTEIRO, Antônio Miguel Vieira. Fundamentos de geoprocessamento. In: CÂMARA, Gilberto; MONTEIRO, Antônio Miguel Vieira. **Conceitos Básicos em Ciência da Geoinformação**. **Brasil: Researchgate**, 2005. p. 10. Disponível em: <http://www.dpi.inpe.br/gilberto/livro/introd/cap2-conceitos.pdf>. Acesso em: 18 jan. 2023.

CONCAR – Comissão Nacional de Cartografia. **Plano de ação para implantação da Infraestrutura Nacional de Dados Espaciais**. Disponível em: < <https://www.inde.gov.br/pdf/PlanoDeAcaoINDE.pdf> >. Acesso em 30 dezembro 2022.

DA SILVA, Lorena Souza. Uso do geoprocessamento para mapeamento do uso e ocupação do solo com ênfase em métricas da paisagem: um estudo de caso na bacia hidrográfica do rio Água Branca. **Revista Acta Ambiental Catarinense**, v. 17, n. 1, p. 33-44, 2020.

DORMAN, Michael. **Introduction to Spatial Data Programming with R**. [S. l.], 21 fev. 2022. Disponível em: <http://132.72.155.230:3838/r/index.html#r-and-analysis-of-spatial-data>. Acesso em: 28 jun. 2022.

EMBRAPA, Empresa Brasileira de Pesquisa Agropecuária. **Geotecnologias e Geoinformação: O produtor pergunta, a Embrapa responde**. Brasília, DF, 2014. Disponível em: <<https://ainfo.cnptia.embrapa.br/digital/bitstream/item/107363/1/500P-Geotecnologias-e-geoinformacao-ed01-2014>>.

GAMA, L. H. O. M. .; PAIVA, P. F. P. R.; SILVA JUNIOR, O. M. da; RUIVO, M. de L. P. Environmental modeling and use of artificial intelligence for prognosis of deforestation: the case of Rebio do Gurupi-MA. **Research, Society and Development**, [S. l.], v. 10, n. 2, p. e13810211609, 2021. DOI: 10.33448/rsd-v10i2.11609. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/11609>. Acesso em: 7 apr. 2023.

GIBBS, H. K., L. RAUSCH, J. MUNGER, I. SCHELLY, D. C. MORTON, P. NOOJIPADY, B. SOARES-FILHO, P. BARRETO, L. MICOL, and N. F. WALKER. 2015. “Brazil’s Soy Moratorium.” *Science* 347 (6220): 377–78. <https://doi.org/10.1126/science.aaa0181>.

GHORBANZADEH, Omid et al. Spatial prediction of wildfire susceptibility using field survey gps data and machine learning approaches. **Fire**, v. 2, n. 3, p. 43, 2019.

GITHUB (2022) [cited 01 Jan 2022]. Available from: <https://github.com>.

HARTSHORNE, Richard. **Propósitos e natureza da geografia**/Richard Hartshorne; tradução original de Thomaz Newlands Neto; supervisão de Fábio MS Guimarães e LMC Bernardes.—. 1978.

HAYNES, David; MANSON, Steve; SHOOK, Eric. Terra Populus' architecture for integrated big geospatial services. **Transactions in GIS**, v. 21, n. 3, p. 546-559, 2017.

HAYNES, David et al. High performance analysis of big spatial data. In: **2015 IEEE International Conference on Big Data (Big Data)**. IEEE, 2015. p. 1953-1957.

HEYCK, Denis L. Surviving globalization in three Latin American communities. **University of Toronto Press**, 2019.

HENNING, E. et al. para além da computação estatística: o uso do ambiente R para o ensino de métodos numéricos. **RENOTE**, v. 14, n. 1, 2016.

IBGE – Instituto Brasileiro de Geografia e Estatística (Brasil). **Acesso e uso de dados geoespaciais**. Rio de Janeiro: IBGE, 2022. Disponível em: <https://biblioteca.ibge.gov.br/visualizacao/livros/liv101675.pdf>. Acesso em: 16 fev. 2022.

IBGE - Instituto Brasileiro e Geografia e Estatística (Brasil). **Cidades e Estados**. 2021. Disponível em: <https://www.ibge.gov.br/cidades-e-estados/to.html>. Acesso em: 16 fev. 2022.

IBGE. **Manual técnico de uso da terra**. 3.ed. Rio de Janeiro: IBGE, 2013.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA (IBGE). 2021. **Áreas Territoriais**. Disponível em <https://www.ibge.gov.br/geociencias/organizacao-do-territorio/estrutura-territorial/15761-areas-dos-municipios.html?=&t=sobre>. Acessado em: 25.out.2022.

JENERETTE, G. Darrel et al. Ecosystem services and urban heat riskscape moderation: water, green spaces, and social inequality in Phoenix, USA. **Ecological applications**, v. 21, n. 7, p. 2637-2651, 2011.

JENERETTE, G. Darrel et al. Regional relationships between surface temperature, vegetation, and human settlement in a rapidly urbanizing ecosystem. **Landscape ecology**, v. 22, n. 3, p. 353-365, 2007.

KLINK, Carlos A.; MACHADO, Ricardo B. Conservation of the Brazilian cerrado. **Conservation biology**, v. 19, n. 3, p. 707-713, 2005.

LONGLEY, Paul A. et al. Geographic information science and systems. John Wiley & Sons, 2015.

MAPBIOMAS. **MapBiomas General ‘Handbook’—Algorithm Theoretical Basis Document (ATBD)**. *Collection 7*, Jan. 2023, [online] Available: https://mapbiomas-br-site.s3.amazonaws.com/ATBD_Collection_7_v2.pdf.

NEPSTAD, Daniel et al. Slowing Amazon deforestation through public policy and interventions in beef and soy supply chains. **science**, v. 344, n. 6188, p. 1118-1123, 2014.

PEBESMA, E; BIVAND R (2020) **sp: classes and methods for spatial data**. <https://CRAN.R-project.org/package=sp>, R package version 1.4-2.

PEBESMA EJ; BIVAND RS (2005). “**Classes and methods for spatial data in R.**” R News, 5(2), 9–13. <https://CRAN.R-project.org/doc/Rnews/>.

QGIS DEVELOPMENT TEAM. (2022). **QGIS geographic information system**. QGIS Association. <https://www.qgis.org>.

R CORE TEAM. 2021. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.

REIBEL, Michael; AGRAWAL, Aditya. Areal interpolation of population counts using pre-classified land cover data. **Population Research and Policy Review**, v. 26, n. 5, p. 619-633, 2007.

HIJMANS, Robert J. (2022). raster: Geographic Data Analysis and Modeling. R package version 3.5-15. <https://CRAN.R-project.org/package=raster>.

RSTUDIO TEAM (2021). **RStudio: Integrated Development Environment for R**. RStudio, PBC, Boston, MA URL <http://www.rstudio.com/>.

SAADAT, Hossein et al. Land use and land cover classification over a large area in Iran based on single date analysis of satellite imagery. **ISPRS journal of Photogrammetry and Remote Sensing**, v. 66, n. 5, p. 608-619, 2011.

SCARIOT, Aldicir; FELFILI, Jeanine M.; SILVA, José Carlos Sousa. **Cerrado: ecologia, biodiversidade e conservação**. 2005.

SIEVERT, Carson. (2020). Interactive Web-Based Data Visualization with R, plotly, and shiny. Chapman and Hall/CRC. ISBN 9781138331457, <https://plotly-r.com>.

Soares, J. L. N.; Espindola, C. R. **Geotecnologias no planejamento de assentamentos rurais: premissa para o Desenvolvimento Rural Sustentável**. Presidente Prudente-SP. Revista NERA, ano 11, n.13, p. 108-116, 2008.

SOUZA JR, Carlos M. et al. **Reconstructing three decades of land use and land cover changes in brazilian biomes with landsat archive and earth engine**. Remote Sensing, v. 12, n. 17, p. 2735, 2020.

TEIXEIRA, Amândio. 1995. **Qual a melhor definição de SIG? FATOR GIS**, p. 20- 24.

TENNEKES, Martijn; NOWOSAD, Jakub. Dados espaciais em R. In: TENNEKES, Martijn; NOWOSAD, Jakub. **Mapas elegantes e informativos com tmap**. 2021. Cap. 2. p. 2-3. Disponível em: <https://r-tmap.github.io/tmap-book/>. Acesso em: 30 jan. 2023.

TRICART, Jean. **Ecodinâmica: Recursos naturais e meio ambiente**. IBGE/SUPREN, Rio de Janeiro, 91p, 1977.

TRICART, Jean et al. **Ecogeography and rural management: a contribution to the International Geosphere-Biosphere Programme**. Longman Group UK, 1992.

TOMLIN, C. Dana et al. **Geographic information systems and cartographic modeling**. Englewood Cliffs, NJ: Prentice Hall, 1990.

WICKHAM, Hadley (2016). ggplot2: Elegant Graphics for Data Analysis. **Springer-Verlag New York**. ISBN 978-3-319-24277-4, <https://ggplot2.tidyverse.org>.

WICKHAM, Hadley et al. (2022). dplyr: A Grammar of Data Manipulation. R package version 1.0.8. <https://CRAN.R-project.org/package=dplyr>.

ANEXO

Código-Fonte Documentado

Documentação do Código

Termos de Uso do Programa

Este programa utiliza um código-fonte desenvolvido por Kaik Octaviano Fontana, Cristiano Augusto de Souza, Ruth de Souza, Mateus Faria Santanta e Iuri Emmanuel de Paula Ferreira (Coordenador), do Centro de Ciências da Natureza da Universidade Federal de São Carlos, campus Lagoa do Sino.

O código-fonte deste programa está protegido por direitos autorais e é regido pela Licença GNU Affero General Public License (GNU AGPL) versão 3.0 ou posterior. Qualquer redistribuição ou modificação do código-fonte deve ser realizada de acordo com os termos desta licença.

É crucial destacar que a GNU AGPL estabelece a necessidade de que os autores do software sejam devidamente reconhecidos em qualquer redistribuição ou modificação efetuada no código-fonte original. Tal premissa se constitui como um requisito imprescindível para assegurar o respeito aos direitos autorais dos criadores do software, bem como para manter a transparência e a credibilidade do software, objetivos estes que são de grande relevância para toda a comunidade de usuários e desenvolvedores de software livre. Conseqüentemente, é fundamental observar tal exigência ao utilizar o programa.

A documentação deste programa foi confeccionada com a ajuda do modelo de linguagem GPT-4, desenvolvido pela OpenAI. O GPT-4 é uma ferramenta de inteligência artificial avançada que utiliza técnicas de aprendizado profundo para gerar texto natural em diversos idiomas e áreas de conhecimento. Agradecemos à OpenAI pelo fornecimento desta ferramenta e reconhecemos a sua contribuição na elaboração desta documentação.

Ao utilizar este programa, você concorda em cumprir os termos e condições estabelecidos pela Licença GNU Affero General Public License (GNU AGPL) versão 3.0 ou posterior. Para obter mais informações sobre a licença, consulte o arquivo "LICENSE" que acompanha este programa ou acesse o seguinte link: <https://www.gnu.org/licenses/agpl-3.0.html>.

É importante ressaltar que este programa é distribuído na esperança de que seja útil, mas SEM QUALQUER GARANTIA; sem sequer a garantia implícita de COMERCIALIZAÇÃO ou ADEQUAÇÃO A UM PROPÓSITO ESPECÍFICO. Consulte a Licença GNU Affero General Public License para obter mais detalhes.

Instalação de dependências

Este script em R instala e carrega uma série de pacotes necessários para trabalhar com dados geoespaciais, visualização e manipulação de dados. Os pacotes incluem: rgdal, sp, raster, ggplot2, gridExtra, rasterVis, leaflet.extras, rgeos, mapview, leaflet, broom, dplyr, plotly, sf, geosphere e ggsfatial.

```

# Definir pacotes necessários
list.of.packages <- c("rgdal", "sp", "raster", "ggplot2", "gridExtra",
                    "rasterVis", "leaflet.extras",
                    "rgeos", "mapview", "leaflet", "broom", "dplyr", "plotly",
                    "sf", "geosphere", "ggspatial")

# Verificar e instalar pacotes não instalados
new.packages <- list.of.packages[!(list.of.packages %in% installed.packages()[, "Package"])]
if(length(new.packages)) install.packages(new.packages)

# Carregar pacotes
lapply(list.of.packages, library, character.only = TRUE)

# Configurar opção para evitar a conversão automática de strings em fatores
options(stringsAsFactors = FALSE)

```

Função para criar polígono

Este script em R define funções para criar buffers de círculo, quadrado e hexágono com um tamanho especificado (em *km*) em torno de coordenadas geográficas fornecidas. A função principal `make_polygon()` cria e exporta um arquivo shapefile com o polígono a ser analisado.

```

# Função para criar um círculo
circle_coords <- function(lat, lon, adj_size){
  x <- y <- seq(0, 2*pi, length.out = 100)
  circle.coords <- cbind(adj_size * cos(x) + lon, adj_size * sin(x) + lat)
  polygon <- Polygon(circle.coords)
  return(polygon)
}

# Função para criar um hexágono
hexagon_coords <- function(lat, lon, adj_size){
  hexagon.coords <- rbind(c(adj_size + lon, lat), c(lon + adj_size/2, lat + adj_size*sqrt(3)/2), c(lon - adj_size/2, lat + adj_size*sqrt(3)/2), c(lon - adj_size, lat), c(lon - adj_size/2, lat - adj_size*sqrt(3)/2), c(lon + adj_size/2, lat - adj_size*sqrt(3)/2))
  polygon <- Polygon(hexagon.coords)
  return(polygon)
}

# Função para criar um quadrado
square_coords <- function(lat, lon, adj_size){
  square.coords <- rbind(c(lon + adj_size, lat + adj_size), c(lon + adj_size, lat - adj_size), c(lon - adj_size, lat - adj_size), c(lon - adj_size, lat + adj_size))
  polygon <- Polygon(square.coords)
  return(polygon)
}

# Função principal para criar e exportar um arquivo shapefile com o buffer

```

```

make_polygon <- function(lat = NULL, lon = NULL, size = NULL, shape = c("circle", "square", "hexagon")){
  if (is.null(lat) | is.null(lon)){
    stop("Please provide latit and longit.")
  }
  if(length(size)==0){
    stop("Please provide size, in km.")
  }
  shape <- match.arg(shape)

  buffer_crs<-CRS("+proj=longlat +datum=WGS84 +no_defs")
  adj_size <- distHaversine(c(lon, lat), c(lon + size*1000/111320, lat))/111320

  buffer_poly <- switch(shape,
    "circle" = circle_coords(lat, lon, adj_size),
    "square" = square_coords(lat, lon, adj_size),
    "hexagon" = hexagon_coords(lat, lon, adj_size)
  )
  polygons <- SpatialPolygons(list(Polygons(list(buffer_poly), ID = "polygon")),
    proj4string = buffer_crs)

  if (!dir.exists("./shape-example")){
    dir.create("./shape-example")
  }

  polygons <- as(polygons, "SpatialPolygonsDataFrame")
  writeOGR(polygons, "./shape-example", "buffer", driver = "ESRI Shapefile", overwrite_layer = TRUE)

  message("Shapefile created")
  message("Shapefile stored in ./shape-examples as buffer.shp")

  return(list("buffer" = polygons))
}

```

Função de carregamento dos rasters

A função `load_rasters` carrega arquivos shapefile e imagens raster de uso da terra para um período de tempo especificado. A função retorna uma lista contendo o shapefile carregado, o intervalo de tempo e os rasters recortados.

Descrição dos parâmetros:

- `shape_path`: Caminho para o arquivo shapefile que define a área de interesse;
- `start`: Ano inicial do intervalo de tempo para carregar as imagens raster (padrão: 1985).
- `end`: Ano final do intervalo de tempo para carregar as imagens raster (padrão: 2020).

A função realiza as seguintes etapas principais:

1. Carrega o arquivo shapefile e obtém seu sistema de referência de coordenadas (CRS).

2. Carrega as imagens raster de uso da terra para o intervalo de tempo especificado.
3. Verifica se o CRS do shapefile e das imagens raster são idênticos.
4. Recorta as imagens raster usando o shapefile.
5. Retorna uma lista contendo o shapefile, o intervalo de tempo e os rasters recortados.

```
# Função para carregar shapefiles e imagens raster de uso da terra
load_rasters <- function(shape_path = "path_to_shape", start = 1985, end =
2020){
  # Carrega o arquivo shapefile e obtém seu CRS
  s <- readOGR(shape_path)
  s_utm <- proj4string(s)

  # Carrega as imagens raster de uso da terra para o intervalo de tempo es
pecificado
  biome_rasters <- NULL
  for(i in start:end){
    path_raster <- sprintf("./mapbiomas/colecao-sete/brasil_coverage_%d.ti
f", i)
    biome_rasters[[i - (start - 1)]] <- raster(path_raster)
  }

  # Verifica se o CRS do shapefile e das imagens raster são idênticos
  biome_utm <- proj4string(biome_rasters[[1]])
  if(biome_utm != s_utm){
    stop("Error. CRSs differed from shp and rasters. CRS must be identical
!")
  }

  # Recorta e mascara as imagens raster usando o shapefile
  maps <- NULL
  map <- NULL
  for(i in start:end){
    maps[[i - (start - 1)]] <- crop(biome_rasters[[i - (start - 1)]], s)
    map[[i - (start - 1)]] <- mask(maps[[i - (start - 1)]], s)
  }

  # Retorna uma lista contendo o shapefile, o intervalo de tempo e os rast
ers mascarados
  return(list("shape" = s, "time_range" = c(start, end), "raster" = map))
}
```

Dicionário

Neste estudo, utilizamos um dicionário específico para classificar e visualizar diferentes tipos de cobertura do solo e uso da terra, conforme fornecido pelo Projeto MapBiomias. O MapBiomias é uma iniciativa colaborativa que envolve diversas instituições do Brasil, cujo objetivo é monitorar e analisar as mudanças na cobertura e uso da terra ao longo do tempo.

Land Use	Code	Color
Non Cataloged	0	FFFFFF

Forest	1	129912
Forest Formation	3	006400
Savanna Formation	4	00ff00
Mangrove	5	687537
Forest Plantation	9	00CC66
Non Forest Natural Formation	10	bbfcac
Wetland	11	45c2a5
Grassland	12	b8af4f
Other Non Forest Formations	13	bdb76b
Farming	14	ffffb2
Pasture	15	ffd966
Agriculture	18	e974ed
Temporary Crop	19	d5a6bd
Sugar Cane	20	c27ba0
Mosaic of Uses	21	fff3bf
Non Vegetated Area	22	ea9999
Beach, Dune and Sand Spot	23	dd7e6b
Urban Area	24	af2a2a
Other Non Vegetated Areas	25	ff99ff
Water	26	0000ff
Non Observed	27	D5D5E5
Rocky Outcrop	29	ff8C00
Mining	30	8a2be2
Aquaculture	31	29eee4
Salt Flat	32	968c46
River, Lake and Ocean	33	0000ff
Perennial Crop	36	f3b4f1
Soybean	39	c59ff4
Rice	40	982c9e
Other Temporary Crops	41	CCFFCC
Coffee	46	cca0d4
Citrus	47	d082de
Other Perennial Crops	48	cd49e4
Wooded Sandbank Vegetation	49	6b9932
Herbaceous Sandbank Vegetation	50	66ffcc
Cotton	62	660066
:-----: :-----: :-----:		

Cálculo de áreas

A função `get_area()` é usada para calcular a área ocupada por diferentes classes de uso da terra em um conjunto de dados específico, que representa as informações de uso da terra e cobertura do solo ao longo do tempo.

```
get_area <- function(data = NULL){
  # time span of study
  time_span <- data$time_range[2] - data$time_range[1]

  areas <- NULL
  # Loop para calcular a área de cada classe de uso da terra em cada ano
  for(yr in (data$time_range[1] - 1984):(time_span + 1)){
    areas_y <- area(data$raster[[yr]])
    v_r <- getValues(data$raster[[yr]])
  }
}
```

```

    areas[[yr]] <- tapply(areas_y, v_r, sum)
  } # end for

df <- NULL
# Criar um data frame com o ano, a área e a classe de uso da terra
for(yr in (data$time_range[1] - 1984):(time_span + 1)){
  aux <- data.frame(rep(yr+1984, dim(areas[[yr]])), round(areas[[yr]],
3))
  df <- rbind(df, cbind(names(areas[[yr]]), aux))
}
colnames(df) <- c("land_class", "year", "area")

# Associar o nome e a cor da classe de uso da terra
land_class_name <- NULL
land_class_color <- NULL
for(j in 1:length(df$land_class)){
  pos <- df$land_class[j] == dic$code
  land_class_name[j] <- dic$class[pos]
  land_class_color[j] <- dic$color[pos]
}
df$land_class_name <- land_class_name

# Criar um gráfico de séries temporais com a área ocupada para cada clas
se de uso da terra
p <- ggplot(df, aes(x=year, y=area, group=land_class_name, color=land_cl
ass_name) ) +
  scale_color_manual(values = land_class_color,
                    limits = land_class_name, name = "Land Use") +
  geom_line(size=2) + ylab(expression(paste("Area in ", km^2))) + xlab("
Years")

# Retornar os anos, área ocupada, dados agregados e a série temporal grá
fica
return(
  list(
    "Years" = data$time_range[1):(data$time_range[1]+time_span) ,
    "Occupied area" = areas, "aggregate_data" = df, "time_series" = p
  )
)
} # fim

```

A função `get_area()` recebe um objeto `data` como entrada, contendo informações sobre o período de tempo e os dados raster para cada ano. A função calcula a área ocupada por cada classe de uso da terra em cada ano, e retorna os anos, a área ocupada, os dados agregados e um gráfico de séries temporais que mostra as áreas ocupadas para cada classe de uso da terra ao longo do tempo.

Função para a visualização dos rasters

A função `land_vis()` é responsável por criar uma visualização de uso do solo para um determinado ano, utilizando o conjunto de dados fornecido. A visualização é construída com base em dados de raster e um dicionário de classes e cores de uso do solo, sendo exibida em um gráfico `ggplot`.

O código da função realiza várias etapas de processamento, como a verificação de dados nulos e limites do ano informado. Em seguida, realiza a extração de informações das classes de uso do solo e das cores correspondentes, converte o objeto Raster em um data frame e, finalmente, cria o gráfico ggplot com a visualização do uso do solo. O gráfico gerado inclui elementos como uma seta indicando o norte, rótulos dos eixos X e Y, título do gráfico e uma escala de cores personalizada com valores e rótulos manuais, além de uma legenda invertida.

```
# Função de visualização de uso do solo
land_vis <- function(data = NULL, year = 1985) {
  # Verifica se os dados são NULL e se o ano está dentro dos limites válidos
  if (is.null(data) == 1) {
    print("Erro. Dados são NULL!")
    return(NULL)
  }
  if (year < 1985 || year > 2020) {
    print("Ano fora dos limites!")
    return(NULL)
  }

  # Extrai o raster correspondente ao ano informado
  raster <- data$raster[[year - 1984]]

  # Cria um data frame com informações sobre as classes e áreas
  dd <- data.frame(table(getValues(raster)))
  colnames(dd) <- c("classe", "area")

  # Inicializa as variáveis para armazenar os nomes e cores das classes
  names_classes <- NULL
  cores_classes <- NULL

  # Obtém os nomes e cores das classes de uso do solo
  for (j in 1:length(dd$classe)) {
    pos <- dd$classe[j] == dic$code
    names_classes[j] <- dic$class[pos]
    cores_classes[j] <- dic$color[pos]
  }

  # Converte o objeto Raster para um data frame com colunas "value", "x" e "y"
  raster <- as(raster, "SpatialPixelsDataFrame")
  dat <- as.data.frame(raster)
  names(dat) <- c("value", "x", "y")

  # Cria o gráfico de visualização de uso do solo
  p <- ggplot(data = dat, aes(x = x, y = y, fill = factor(value))) +
    geom_tile() + coord_sf(crs = "+proj=longlat +datum=WGS84") +
    annotation_scale(width_hint = 0.2, style = "bar", location = "br", unit_category = "metric") +
    annotation_north_arrow(location = "tr", which_north = "true", pad_x =
unit(0.1, "in"), pad_y = unit(0.2, "in"), style = north_arrow_fancy_orient
eering) +
  xlab("Longitude") + ylab("Latitude") + ggtitle(paste("Ano: ", as.chara
```

```

cter(year))) +
  scale_fill_manual(values = cores_classes, name = "Usos do Solo", labels =
names_classes, guide = guide_legend(reverse = TRUE))

  return(p)
}

```

Função para análise da distribuição dos usos da terra

A função `land_dist()` tem como objetivo criar uma visualização da distribuição de uso do solo para um determinado ano, utilizando o conjunto de dados fornecido. A visualização pode ser exibida como um gráfico de barras ou um gráfico de pizza, dependendo do valor do parâmetro `type`. Se o tipo de gráfico for `pie`, a função cria um gráfico de pizza usando o `ggplot`. Caso o tipo de gráfico seja `barplot`, a função ordena os dados e cria um gráfico de barras.

```

# Função de distribuição de uso do solo
land_dist <- function(data = NULL, year = NULL, type = "barplot") {

  # Filtra os dados para o ano especificado
  df <- data$aggregate_data
  dat <- df %>% filter(year == year)

  # Obtém os nomes e cores das classes de uso do solo
  names_classes <- NULL
  cores_classes <- NULL
  for(j in 1:length(dat$land_class)) {
    pos <- dat$land_class[j] == dic$code
    names_classes[j] <- dic$class[pos]
    cores_classes[j] <- dic$color[pos]
  }

  # Adiciona colunas com cores, nomes das classes e percentual de área
  dat$cores <- cores_classes
  dat$names <- names_classes
  dat$percentual <- round(dat$area / sum(dat$area) * 100, 2)

  # Verifica o tipo de gráfico selecionado e cria o gráfico correspondente
  if (type == "pie") {
    p <- ggplot(dat, aes(x = "", y = area, fill = names)) +
      geom_bar(width = 1, stat = "identity") +
      coord_polar("y", start = 0) +
      scale_fill_manual(values = dat$cores, limits = dat$names, name = "Land
Uses") +
      theme_void() + labs(x = NULL, y = NULL) +
      ggtitle(paste("Year: ", as.character(year)))
  } else if (type == "barplot") {
    dat_ordenado <- dat %>% arrange(desc(area)) %>%
      mutate(names = factor(names, levels = names[order(area, decreasing =
TRUE)]))

    p <- ggplot(dat_ordenado, aes(x = names, y = area, fill = names)) +
      geom_bar(stat = "identity") +
      scale_fill_manual(values = dat_ordenado$cores, limits = dat_ordenado

```

```
$names, name = "Land Uses") +  
  labs(x = NULL, y = expression(paste("Area in ", km^2))) +  
  theme(axis.text.x = element_blank(), axis.ticks.x = element_blank())  
+  
  ggtitle(paste("Year: ", as.character(year)))  
} else {  
  print("Type not available yet!")  
  return(NULL)  
}  
  
plot(p)  
}
```