



ANA PAULA DALLA CORTE  
FABIANO RODRIGUES PEREIRA  
THAIS CHAVES ALMEIDA  
NATIELE CAUMO MEZACASA

# SENSORIAMENTO REMOTO COM O GOOGLE EARTH ENGINE (GEE)

UM ENFOQUE PARA A  
ENGENHARIA FLORESTAL

1<sup>a</sup> EDIÇÃO

Ana Paula Dalla Corte  
Fabiano Rodrigues Pereira  
Thaís Chaves Almeida  
Natiele Caumo Mezacasa

**SENSORIAMENTO REMOTO COM O GOOGLE  
EARTH ENGINE (GEE)  
UM ENFOQUE PARA A ENGENHARIA FLORESTAL**

**1<sup>a</sup> Edição**

Curitiba  
Ana Paula Dalla Corte  
2025

## FICHA CATALOGRÁFICA

Copyright by Ana Paula Dalla Corte, Fabiano Rodrigues Pereira, Thaís Chaves Almeida, Natiele Caumo  
Mezacasa

**Autores:** Ana Paula Dalla Corte, Fabiano Rodrigues Pereira, Thaís Chaves Almeida, Natiele Caumo  
Mezacasa

**Capa:** Ana Paula Dalla Corte, Natiele Caumo Mezacasa

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)  
UNIVERSIDADE FEDERAL DO PARANÁ / SISTEMA DE BIBLIOTECAS  
BIBLIOTECA DE CIÊNCIAS FLORESTAIS E DA MADEIRA

Dalla Corte, Ana Paula

Sensoriamento remoto com o Google Earth Engine (GEE): um enfoque para a engenharia florestal / Ana Paula Dalla Corte, Fabiano Rodrigues Pereira, Thaís Chaves Almeida, Natiele Caumo Mezacasa. - Curitiba, 2025.

94 p. : il. color.; 30 cm.

Inclui bibliografias.

ISBN: 978-65-01-66265-7

1. Sensoriamento remoto. 2. Google Earth Engine. 3. Imagens de sensoriamento remoto. 4. Engenharia florestal - Sensoriamento remoto.  
I. Pereira, Fabiano Rodrigues. II. Almeida, Thaís Chaves.  
III. Mezacasa, Natiele Caumo. IV. Título.

CDD – 621.3678

CDU – 528.8

Bibliotecária: Berenice Rodrigues Ferreira – CRB 9/1160

# APRESENTAÇÃO

O livro “Sensoriamento Remoto com o *Google Earth Engine* (GEE): um enfoque para a Engenharia Florestal” foi confeccionado com o objetivo de oferecer aos alunos e profissionais da área uma literatura prática e acessível, permitindo que aprendam a utilizar a plataforma de forma tutorada, passo a passo. A proposta é que o material sirva como guia e referência de estudo, auxiliando na compreensão dos fundamentos do GEE e de suas aplicações voltadas ao campo florestal e ambiental.

No contexto do sensoriamento remoto moderno, sentimos a necessidade de que os profissionais estejam cada vez mais capacitados para utilizar essas tecnologias, desenvolvendo, ao menos, um entendimento básico de programação. Isso é fundamental diante do grande volume de dados disponíveis, da necessidade de análises automatizadas e da busca por resultados mais precisos e rápidos. Nesse cenário, o *Google Earth Engine* surgiu como uma ferramenta revolucionária, integrando programação, poder computacional em nuvem e acesso a bases de dados globais, tornando-se essencial para quem deseja se destacar na área.

Estamos muito felizes em disponibilizar este material, fruto da dedicação e do compromisso com o ensino, e acreditamos que ele será de grande valor para apoiar a formação técnica, científica e profissional daqueles que buscam dominar ferramentas modernas de sensoriamento remoto aplicadas à Engenharia Florestal.

Por fim, estamos imensamente felizes em contribuir para a disseminação do conhecimento sobre o *Google Earth Engine* na Engenharia Florestal, fortalecendo a integração entre ciência, tecnologia e prática profissional, e incentivando o uso de ferramentas inovadoras para a gestão sustentável dos recursos naturais.

Boa sorte na jornada!

Os autores.

# SUMÁRIO

1. ACESSO AO MATERIAL PARA OS EXERCÍCIOS GUIADOS.....	8
2. ACESSO AO GEE E SUA INTERFACE .....	9
3. TASSELED CAP.....	25
3.2. Janela temporal e ordenação por nebulosidade .....	25
3.3. Inspeção e enquadramento da cena.....	26
3.4. Composição RGB de referência (B4-B3-B2).....	26
3.5. Fundamentos do Tasseled Cap para Landsat 8 TOA.....	26
3.6. Cálculo das componentes por produto interno.....	26
3.7. Interpretação das três componentes principais.....	27
3.8. Configuração de paletas e faixas de visualização .....	27
3.9. Renderização das camadas TCT no mapa .....	27
3.10. Boas práticas e variações opcionais .....	28
3.11. Validação e comparação temporal/espacial .....	29
4. COMPOSIÇÃO DE IMAGENS, MANIPULAÇÃO DE HISTOGRAMA E CÁLCULO DE ÍNDICES DE VEGETAÇÃO .....	30
4.1. Importação da área de interesse (ROI) .....	30
4.2. Seleção da coleção de imagens Sentinel-2 .....	30
4.3. Pré-processamento radiométrico e recorte da cena .....	31
4.4. Composições coloridas .....	31
4.5. Cálculo de índices de vegetação .....	32
4.6. Visualização dos índices no mapa .....	34
4.7. Construção do histograma.....	35
5. ELABORAÇÃO DE MAPAS DE MUDANÇAS TEMPORAIS (DETECÇÃO DE MUDANÇAS) .....	38
5.1. Definição da área de estudo.....	38
5.2. Máscara de nuvens e sombras .....	38
5.3. Normalização da reflectância .....	38
5.4. Seleção temporal e composição de imagens .....	39
5.5. Cálculo do índice de vegetação NDVI .....	39
5.6. Detecção de mudança com $\Delta$ NDVI.....	40
5.7. Visualização e interpretação dos resultados .....	40

6. CLASSIFICAÇÃO DE USO DO SOLO COM SENSORES ÓPTICOS E SAR ( <i>Synthetic-aperture radar</i> ).....	43
6.1. Definição da área de estudo.....	43
6.2. Processamento das imagens ópticas Landsat-9.....	43
6.3. Correção radiométrica e geração do compósito óptico .....	44
6.4. Cálculo dos índices de vegetação .....	44
6.5. Processamento das imagens SAR Sentinel-1.....	44
6.6. Derivação de métricas SAR .....	44
6.7. Pré-processamento com filtros espaciais e texturas.....	46
6.8. Classificação supervisionada com <i>Random Forest</i> .....	46
6.9. Construção da Matriz de Confusão.....	48
7. CLASSIFICAÇÃO DE USO DO SOLO COM SATELLITE <i>EMBEDDING DATASET</i> .....	50
7.1. Definição da Área de Interesse (ROI).....	52
7.2. Preparação dos <i>embeddings</i> de satélite .....	52
7.3. Seleção de bandas preditoras e visualização .....	52
7.4. Definição das amostras de treinamento.....	53
7.5. Preparação do conjunto de treinamento e validação .....	54
7.6. Treinamento do classificador <i>Random Forest</i> .....	54
7.7. Validação por <i>hold-out</i> e métricas de acurácia.....	54
7.8. Classificação final da área de estudo .....	54
7.9. Estimativa de áreas por classe .....	55
7.10. Integração com dados do Sentinel-2.....	55
7.11. Análise de mudança temporal ( $\Delta$ NDVI) .....	55
7.12. Cálculo de áreas por classes de mudança .....	55
7.13. Similaridade do cosseno entre embeddings .....	56
7.14. Tabelas síntese e exportação de resultados .....	56
8. ANÁLISE DE SÉRIES TEMPORAIS DOS ÍNDICES DE VEGETAÇÃO.....	57
8.1. Escopo, parâmetros e objetivo .....	57
8.2. Qualidade dos dados e mascaramento atmosférico-geométrico .....	57
8.3. Escalonamento radiométrico das bandas .....	58
8.4. Seleção de bandas e construção dos índices .....	58
8.5. Definição da ROI e configuração do mapa .....	58
8.6. Montagem e filtragem da coleção Landsat 8 .....	58
8.7. Integração opcional com Landsat 9 .....	59

8.8. Produtos medianos para inspeção visual .....	59
8.9. Agregação mensal com metadados e controle de amostra .....	59
8.10. Reduções espaciais e gráficos mensais por índice.....	59
<b>9. AVALIAÇÃO DA ÁREA COM FOCO DE INCÊNDIO E ANÁLISE DA PRECIPITAÇÃO ....</b>	<b>60</b>
9.1. Definição da área de interesse (AOI) .....	61
9.2. Aquisição e filtragem do MODIS MCD64A1 (área queimada).....	61
9.3. Seleção da variável BurnDate e lógica de detecção .....	62
9.4. Estimativa de área queimada por imagem ( $\text{km}^2$ ) .....	62
9.5. Construção da série temporal e visualização de área queimada.....	62
9.6. Aquisição do CHIRPS em pentadas e agregação mensal.....	62
9.7. Série temporal de precipitação e inspeção gráfica.....	63
9.8. Totais sazonais da estação chuvosa (exemplos).....	63
9.9. Análise pontual de setembro de 2019: fogo e chuva.....	64
9.10. Pré-processamento e visualização Sentinel-2 (início de setembro/2019) .....	64
<b>10. DETECÇÃO DE MUDANÇAS COM SAR .....</b>	<b>66</b>
10.1. Definição da área de interesse (ROI) .....	66
10.2. Parâmetros temporais e espaciais.....	66
10.3. Carregamento das coleções Sentinel-1.....	66
10.4. Polarizações VV e VH .....	67
10.5. Suavização do speckle.....	68
10.6. Diferenças temporais .....	68
10.7. Criação de máscaras de perda .....	68
<b>11. ANÁLISE DE APP POR DECLIVIDADE COM SRTM .....</b>	<b>70</b>
11.1. Delimitação da área de estudo (ROI).....	70
11.2. Fonte altimétrica (SRTM) .....	71
11.3. Cálculo de declividade (graus).....	72
11.4. <i>Hillshade</i> para leitura de relevo .....	72
<b>12. ÁREAS URBANAS ARBORIZADAS .....</b>	<b>75</b>
12.1. Delimitação da área de estudo .....	75
12.2. Aquisição, mosaico e composição de falsa cor (SkySat).....	75
12.3. Segmentação SNIC: teoria e parametrização .....	75
12.4. Vetorização e junção de atributos.....	78
12.5. Preparação das amostras .....	78

12.6. Extração de preditores e amostragem .....	79
12.7. Treinamento do Random Forest (RF) .....	79
12.8. Cálculo de área por classe (hectares e km <sup>2</sup> ) .....	80
<b>13. USO DA COLEÇÃO MAPBIOMAS.....</b>	<b>82</b>
13.1. Preparação do Raster e escolha da coleção .....	82
13.2. Importação de limites administrativos .....	82
13.3. Filtragem espacial por municípios de interesse .....	83
13.4. Geometria combinada e recorte do raster .....	83
13.5. Cálculo da área por classe .....	84
13.6. Redução regional e agrupamento .....	84
13.7. Conversão em FeatureCollection .....	84
13.8. Visualização em tabela de áreas.....	84
13.9. Integração com a legenda oficial do MapBiomas e Exportação de Resultados	85
<b>14. UPSCALING GEDI.....</b>	<b>86</b>
14.1. Definição da área de estudo .....	86
14.2. Seleção de floresta plantada com MapBiomas .....	87
14.3. Filtragem e extração de dados GEDI L2A .....	87
14.4. Preparação dos dados ópticos (Landsat 9).....	88
14.5. Preparação dos dados de SAR (Sentinel-1).....	89
14.6. Extração de métricas texturais e focais .....	90
14.7. Construção do stack de preditores .....	90
14.8. Amostragem e preparação do dataset de modelagem.....	91
14.9. Treinamento dos modelos Random Forest .....	91
14.10. Avaliação dos modelos e análise de desempenho .....	92
<b>15. REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>94</b>

# 1. ACESSO AO MATERIAL PARA OS EXERCÍCIOS GUIADOS

O livro foi planejado de forma a combinar a fundamentação teórica com a prática aplicada, permitindo que o leitor acompanhe cada etapa dos procedimentos diretamente no *Google Earth Engine* (GEE). Para isso, todos os capítulos foram estruturados com *scripts* comentados e exemplos práticos, de modo que o aprendizado seja progressivo e dinâmico. A organização buscou alinhar conceitos fundamentais de sensoriamento remoto e geoprocessamento com exercícios que simulam situações reais de análise espacial, garantindo não apenas a compreensão conceitual, mas também a autonomia na execução.

Para facilitar o acesso e a replicação dos conteúdos, todos os *scripts* e arquivos complementares estão disponíveis em repositório público no GitHub. Assim, o leitor poderá baixar o material necessário e acompanhar cada tópico com os mesmos dados e códigos utilizados nos exemplos do livro. O repositório pode ser acessado pelo link: [https://github.com/anacorte/Livro\\_GEE](https://github.com/anacorte/Livro_GEE) (e deverá ser visualizado conforme Figura 1).



Figura 1 - Representação da estrutura dos arquivos disponíveis no Github.

Para acompanhar as atualizações e publicações, recomenda-se o acesso às nossas redes sociais, disponíveis em: <https://www.instagram.com/gpforesteyes/> e de nossa página na internet: <https://gpforesteyes.github.io/>.

## 2. ACESSO AO GEE E SUA INTERFACE

Este capítulo tem como objetivo fundamental guiar o leitor através do processo de acesso e familiarização com o ambiente de trabalho do *Google Earth Engine* (GEE) (AMANI et al., 2020). Serão abordados os requisitos para a criação de uma conta, a navegação na interface do *Code Editor*, a compreensão das funcionalidades de seus principais painéis e os conceitos básicos de programação em *JavaScript* essenciais para a interação com a plataforma. Adicionalmente, será detalhado o procedimento para o gerenciamento de dados, incluindo o upload e a exportação de assets.

### 2.1. Começando com o Google Earth Engine

#### Pré-requisitos para Acesso

O *Google Earth Engine* opera como uma plataforma baseada em nuvem, e o acesso a seus recursos é condicionado à posse de uma conta Google. Portanto, o principal pré-requisito é possuir (ou criar) uma conta de e-mail associada ao Google. Esta conta será o identificador principal do usuário dentro do ecossistema Google, permitindo a integração e o uso de diversos serviços, incluindo o GEE.

#### Processo de Registro

Uma vez que o pré-requisito de uma conta Google seja atendido, o registro para o *Google Earth Engine* é realizado através de uma página dedicada. Os usuários devem navegar até o link oficial de registro: <https://earthengine.google.com/signup/>.

É fundamental ressaltar que o GEE é atualmente gratuito para fins de pesquisa, educação e uso sem fins lucrativos. Este modelo de acesso democratiza o uso de ferramentas de sensoriamento remoto de escala planetária, tornando-o acessível a estudantes, pesquisadores e organizações que atuam nesses domínios.

Após a conclusão do registro e a verificação da conta, o usuário estará apto a acessar o *Code Editor* ([code.earthengine.google.com](https://code.earthengine.google.com)), o ambiente de desenvolvimento integrado (IDE) baseado em web para a API *JavaScript* do *Earth Engine*.

## 2.2. Explorando a Interface do GEE: O *Code Editor*

A janela principal do *Code Editor* é organizada em quatro painéis interconectados, cada um com funções específicas essenciais para o fluxo de trabalho no GEE de acordo com a Figura 2.



Figura 2 - Exemplo da interface do GEE.

### Painéis do *Code Editor*:

#### 1. Painel do Editor (Editor panel):

Localizado no centro superior da interface, é onde o código *JavaScript* é escrito. Este painel suporta funcionalidades básicas de edição, como auto-completar (acionado com Ctrl + Espaço), destaque de sintaxe e numeração de linhas. É importante salientar que o GEE não possui salvamento automático; portanto, é recomendável salvar o script frequentemente utilizando o botão Save ou o atalho Ctrl + S.

#### Botões de Controle Principal:

- Run: Executa o script presente no painel do Editor. O atalho Ctrl + Enter (ou Cmd + Enter no Mac) também pode ser utilizado.
- Save: Salva o script atual (atalho Ctrl + S). Recomenda-se salvar com frequência para evitar perda de trabalho.
- Get Link: Gera um link único para compartilhamento do script, útil para colaboração ou submissão de exercícios.

## **2. Painel Direito:**

Contém três abas fundamentais:

- **Console:** É a aba padrão exibida ao carregar o GEE. Nela são impressos resultados de operações, mensagens de erro, metadados de imagens e saídas de funções como `print()`. É uma ferramenta vital para a depuração e visualização de informações textuais.
- **Inspector:** Ao ativar essa aba, o cursor do mouse assume a forma de uma mira. Clicando sobre o mapa, o usuário obtém informações detalhadas sobre o pixel ou a feição naquela localização, incluindo coordenadas, valor de bandas e propriedades. Essa funcionalidade é essencial para análise exploratória de dados.
- **Tasks:** Gerencia processos assíncronos, como exportação de imagens ou vetores para o *Google Drive* ou *Google Cloud Storage*. Aqui é possível monitorar o progresso, cancelar tarefas ou executá-las novamente em caso de falha.

## **3. Painel Esquerdo:**

Organizado em três abas:

- **Scripts:** Lista todos os scripts salvos pelo usuário, organizados por proprietário, leitores, escritores e exemplos públicos. Clicar em um script abre seu conteúdo no painel do Editor.
- **Docs:** Apresenta uma documentação interativa e categorizada de todas as funções da API do GEE. É possível buscar por nome de função ou explorar por módulos (e.g., `ee.Image`, `ee.Reducer`). O uso de `Ctrl + Espaço` durante a edição também aciona uma lista de sugestões baseadas na documentação.
- **Assets:** Gerencia todos os dados carregados pelo usuário (*imagens*, *shapefiles*, *tabelas*). A partir daqui é possível importar ativos para o script atual, compartilhá-los com outros usuários ou excluí-los.

## **4. Mapa Interativo (*Interactive map window*):**

Localizado na parte inferior, é onde as camadas matriciais e vetoriais são visualizadas. O mapa opera de forma semelhante ao Google Maps, com controles de zoom e opções de base (mapa, satélite). Acima do mapa, encontram-se ícones para desenho de

geometrias (pontos, linhas, polígonos), que são automaticamente importadas para o script como variáveis do tipo ee.Geometry. As camadas carregadas podem ser gerenciadas via botão *Layers*, onde é possível ajustar transparência, ordem de exibição e parâmetros de visualização.

## 5. Outros Elementos da Interface

- **Barra de Pesquisa:** Localizada no topo, permite buscar por conjuntos de dados públicos (e.g., "Landsat 8", "MODIS", "NAIP") ou por locais geográficos (e.g., "Manaus, AM"). Ao selecionar um *dataset*, abre-se uma janela com metadados, incluindo bandas, resolução, período de disponibilidade e exemplo de código de importação.
- **Menu de Ajuda:** Acessível pelo ícone de interrogação no canto superior direito, oferece links para o guia do usuário, fórum de discussão, tour interativo das funcionalidades e atalhos de teclado.

## Fluxo Básico de Trabalho no *Code Editor*

1. Escrever o código no painel do *Editor*;
2. Executar com o botão *Run*;
3. Visualizar os resultados no mapa ou no *Console*;
4. Ajustar parâmetros ou geometrias, conforme necessário;
5. Salvar o *script*;
6. Exportar resultados via *Tasks*, se necessário.

## 2.3. Tipos de Dados no GEE

O *Google Earth Engine* (GEE) é otimizado para lidar com grandes volumes de dados geoespaciais, que podem ser amplamente categorizados em dois tipos principais: Dados Matriciais e Dados Vetoriais. Compreender a natureza e a manipulação desses tipos de dados é crucial para qualquer análise na plataforma.

### Dados Matriciais ou Raster: Imagens e Coleções de Imagens

No GEE, dados *raster* são representados como objetos ee.Image. Uma imagem é composta por uma ou mais bandas, e cada banda possui seu próprio nome, tipo de

dado, escala e projeção. Adicionalmente, cada imagem contém metadados armazenados como um conjunto de propriedades.

A plataforma oferece acesso direto a um catálogo de petabytes de imagens de satélite e conjuntos de dados geoespaciais, incluindo o catálogo completo do Landsat (EROS/USGS/NASA), MODIS, NAIP, Sentinel-1, além de dados de precipitação, elevação, temperatura da superfície do mar e dados climáticos CHIRPS. Além de carregar imagens do arquivo por um ID de imagem, é possível criar imagens a partir de constantes, listas ou outros objetos *Earth Engine*.

Uma ImageCollection (Coleção de Imagens) é uma pilha ou série temporal de imagens. O GEE permite carregar coleções de imagens usando um ID de coleção do *Earth Engine*, criar novas coleções a partir de listas de imagens usando ee.ImageCollection() ou ee.ImageCollection.fromImages(), e até mesmo mesclar coleções existentes.

## 1. Visualizando Metadados Raster

Após carregar uma imagem no *Code Editor*, é fundamental examinar seus metadados para entender suas características. A função print() aplicada a uma imagem exibirá informações sobre ela no Console.

- **Informações de Projeção:** É possível acessar informações detalhadas sobre a projeção de uma imagem utilizando a função .projection(). Isso inclui o Sistema de Referência de Coordenadas (CRS) e os parâmetros de transformação. Por exemplo, as imagens Landsat geralmente utilizam a projeção Google Mercator (EPSG:32618) como padrão do GEE.
- **Tamanho Nominal do Pixel:** O tamanho nominal do pixel da imagem (em metros) no nível mais baixo da pirâmide de imagens pode ser verificado com .projection().nominalScale(). Para imagens Landsat, o tamanho nominal do pixel é tipicamente de 30m.

## 2. Definindo Parâmetros de Visualização

Para que as imagens sejam exibidas de forma significativa no mapa interativo, é necessário definir parâmetros de visualização. Inicialmente, ao carregar uma imagem sem parâmetros específicos, ela pode não ser fácil de visualizar. No entanto, após o

carregamento, um botão "Layers" (Camadas) aparece no canto superior direito da janela do mapa. Clicar nele exibe uma lista de camadas com opções para ligar/desligar a exibição, ajustar a transparência e, através de um ícone de engrenagem, alterar as opções de exibição, como a combinação de bandas. Por exemplo, para exibir uma composição colorida infravermelha típica, as bandas B4, B3 e B2 podem ser usadas como RGB.

É mais conveniente e padronizado estabelecer os parâmetros de visualização antecipadamente no código, utilizando a função `Map.addLayer()`. Essa função aceita diversos parâmetros: `eeObject` (o objeto da imagem a ser exibido), `visParams` (parâmetros de visualização), `name` (nome da camada), `shown` (se a camada deve ser mostrada inicialmente) e `opacity` (transparência).

Os `visParams` (parâmetros de visualização) são definidos como um objeto de dicionário (usando chaves {}) e incluem opções como:

- **bands:** Uma lista de três nomes de bandas separadas por vírgulas para serem mapeadas para RGB, por exemplo, `['B4', 'B3', 'B2']` para uma composição infravermelha colorida.
- **min e max:** Valores que são mapeados para 0 e 255, respectivamente, para controle de contraste. Podem ser um número único ou uma lista de três números, um para cada banda.
- **gamma:** Fator(es) de correção gama para ajuste de brilho e contraste.
- **palette:** Uma lista de *strings* de cores no estilo CSS (apenas para imagens de banda única).
- **opacity:** Transparência da camada, de 0 (totalmente transparente) a 1 (totalmente opaco).
- **format:** "jpg" ou "png".

A especificação desses parâmetros pode ser feita diretamente na chamada `Map.addLayer()` ou definindo uma variável para armazenar todos os parâmetros de visualização, o que melhora a legibilidade do código.

## Dados Vetoriais

Pontos, linhas e polígonos são comumente classificados como objetos 'vetoriais' em sistemas GIS e sensoriamento remoto. No GEE, esses são referidos como geometrias. É possível criar uma geometria de diversas maneiras:

- **Desenho Interativo:** Utilizando as ferramentas de desenho na janela do mapa (botões no lado esquerdo da interface do *Code Editor*), é possível desenhar pontos, linhas ou objetos de área (polígonos). Esses itens desenhados são automaticamente armazenados como camadas na seção "Geometry Imports".
- **Definição por Coordenadas:** Geometrias podem ser criadas programaticamente usando funções como `ee.Geometry.Point()` para um ponto, `ee.Geometry.Rectangle()` para um retângulo ou `ee.Geometry.Polygon()` para um polígono, especificando as coordenadas.
- **Importação:** Também é possível importar dados vetoriais predefinidos de ativos (assets) do *Earth Engine* ou tabelas.
- **Pontos Aleatórios:** Para fins de amostragem, como na avaliação de precisão de classificação, podem ser gerados pontos aleatórios dentro de uma área de estudo usando `ee.FeatureCollection.randomPoints()`.

As geometrias vetoriais são frequentemente utilizadas para filtrar coleções de imagens (por exemplo, `ImageCollection.filterBounds(geometry)`) ou para recortar (clip) imagens para uma região específica de interesse (por exemplo, `myimage.clip(roi)`). Isso permite limitar a análise apenas à área relevante, otimizando o processamento.

## 2.4. Fundamentos de Programação em JavaScript no GEE

O *Google Earth Engine* (GEE) utiliza o *JavaScript* como linguagem principal em sua API dentro do ambiente *Code Editor*. Trata-se de uma linguagem originalmente desenvolvida para aplicações web, mas que, no contexto do GEE, é estendida com funções específicas para manipulação de dados geoespaciais em larga escala.

Dominar a sintaxe do *JavaScript* é fundamental para estruturar scripts de forma clara, eficiente e reproduzível. A seguir, apresentamos os principais elementos da linguagem conforme aplicados ao GEE.

## Regras de Sintaxe Essenciais

- **Finalização de comandos:** cada linha de código deve terminar com um ponto e vírgula (;)

```
var x = 10;  
var y = 5;  
var soma = x + y;
```

- **Declaração de variáveis:** utiliza-se a palavra-chave var

```
var my_grade = 100;
```

- **Parênteses para funções:** parâmetros são passados entre ()

```
print('Esta string será impressa no Console.');
```

- **Strings:** devem ser delimitadas por aspas simples ('')

```
var opinion = 'GEE é incrível';
```

- **Listas (arrays):** definidas entre colchetes []

```
var my_list = ['eggplant', 'apple', 'wheat'];  
print(my_list[0]); // imprime 'eggplant'
```

- **Dicionários (objetos):** definidos entre chaves {} e estruturados em pares chave:valor

```
var my_dict = {'food':'bread', 'color':'red', 'number':42};  
print(my_dict['color']); // imprime 'red'  
print(my_dict.color); // imprime 'red' (notação alternativa)
```

## Comentários no Código

Os comentários são indispensáveis para manter scripts legíveis e didáticos, especialmente em trabalhos colaborativos ou acadêmicos.

- **Linha única:** inicia com //

```
// Este é um comentário de linha única
```

- **Múltiplas linhas:** entre /\* ... \*/

```
/*  
Este é um comentário  
que ocupa várias linhas  
*/
```

## Operações Matemáticas

Podem ser realizadas tanto com símbolos aritméticos quanto com funções incorporadas da API:

```
print(3.subtract(2)); // saída: 1
```

```
print('Subtraindo dois de três é ', 3 - 2); // saída: Subtraindo dois de  
três é 1
```

## Definição e Uso de Funções

A modularização por funções facilita a reutilização de código e aumenta a clareza dos scripts.

- Função simples:

```
var my_hello_function = function(string) {  
    return 'Hello ' + string + '!';  
};  
print(my_hello_function('world'));  
// imprime "Hello world!"
```

- Função com operação matemática:

```
var add_function = function(a) {  
    return a + 3;  
};  
var a = 1;  
print(a + ' mais 3 é ' + add_function(a));  
// imprime "1 mais 3 é 4"
```

## Particularidades do *JavaScript* no GEE

Embora os princípios do *JavaScript* sejam universais, o GEE adiciona elementos específicos para processamento de dados espaciais:

- **Classes principais:** como ee.Image, ee.ImageCollection, ee.Geometry, ee.FeatureCollection.
- **Funções próprias:** como Map.addLayer() para visualização, print() para saída no Console, e Export.image.toDrive() para exportação de resultados.
- **Estrutura de dados imutável:** objetos ee não podem ser modificados diretamente; cada operação retorna um novo objeto.

## 2.5. Visualização e Análise Básica de Imagens

Uma das primeiras etapas no uso do *Google Earth Engine* (GEE) consiste em aprender a visualizar imagens e realizar manipulações básicas, tanto para inspeção exploratória quanto para análises iniciais. Nesta seção, apresentam-se os principais comandos em *JavaScript* utilizados no *Code Editor*.

## Referenciando Imagens

No GEE, uma imagem é representada por um objeto da classe ee.Image. Para carregá-la, é necessário especificar o ID do dataset disponível no catálogo.

```
// Exemplo: Carregar uma cena Landsat 5
var myimage = ee.Image('LANDSAT/LC08/C02/T1_TOA/LC08_221077_20250730');
```

Observação: os IDs das imagens podem ser localizados na aba Docs do *Editor* ou no catálogo do *Google Earth Engine Data Catalog* (<https://developers.google.com/earth-engine/datasets>).

## Centragem do Mapa e Controle de Zoom

O comando Map.setCenter(longitude, latitude, zoom\_level) ajusta a posição e escala da visualização no painel do mapa.

- **Zoom = 1:** visão global.
- **Zoom ≈ 5:** escala continental.
- **Zoom ≈ 10:** escala municipal.
- **Zoom = 15 – 20:** escala urbana/edificações.

Exemplo:

```
// Centralizar em Telêmaco Borba/PR com zoom de nível 10
Map.setCenter(-50.6164, -24.3244, 10);
```

## Adicionando Camadas ao Mapa

O comando Map.addLayer() é a forma padrão de visualizar imagens e outros objetos (vetores, geometrias, coleções).

### a) Visualização Padrão

```
Map.addLayer(myimage);
```

Na ausência de parâmetros adicionais, a imagem pode não apresentar cores adequadas, exigindo ajustes.

### b) Controles Interativos

Após carregar a camada, um botão “Layers” aparece no canto superior direito da janela do mapa, permitindo:

- habilitar/desabilitar camadas;

- alterar transparência; e
- reconfigurar parâmetros de exibição (bandas, estiramento de histogramas etc.).

### c) Uso de Parâmetros de Visualização (visParams)

O atalho Ctrl+Espaço no Editor mostra os parâmetros disponíveis para Map.addLayer.

Exemplo de uso básico com combinação de bandas:

```
// Exibir composição infravermelho colorido
Map.addLayer(myimage, {bands: ['B4', 'B3', 'B2']}, 'ColorIR composite');
```

### d) Definição de Parâmetros com Dicionário

```
Map.addLayer({
  eeObject: myimage,
  visParams: {bands: ['B4', 'B3', 'B2']},
  name: 'ColorIR composite 2'
});
```

### e) Organização com Variáveis

Para melhorar a legibilidade do código, recomenda-se armazenar os parâmetros de visualização em variáveis:

```
var vizParams = {
  bands: ['B4', 'B3', 'B2'],
  min: 0,
  max: 0.35,
  gamma: [0.95, 1.1, 1]
};
Map.addLayer(myimage, vizParams, 'Color IR composite');
```

## Recorte de Imagens (Clipping)

Em análises florestais, é comum limitar a visualização a uma região de interesse (ROI), seja para reduzir a área analisada, seja para focar em unidades de estudo (ex.: bacias hidrográficas, fragmentos florestais).

### a) Criando uma ROI via geometria

```
// Criar um ponto em Telêmaco Borba e gerar um buffer de 20 km
var roi = ee.Geometry.Point([-50.6164, -24.3244]).buffer(20000);
```

### b) Visualizando a ROI

```
Map.addLayer(roi, {}, 'ROI');
```

### c) Aplicando o recorte à imagem

```
// Exibir imagem recortada à ROI  
Map.addLayer(myimage.clip(roi), vizParams, 'Imagen recortada');
```

Dica: ROIs também podem ser desenhadas manualmente no mapa usando as ferramentas gráficas do *Code Editor*, ficando armazenadas em “Geometry Imports”.

## 2.6. Coleções de Imagens (ImageCollection)

No *Google Earth Engine*, uma *ImageCollection* é um conjunto estruturado de imagens, frequentemente representando séries temporais ou pilhas espaciais de um mesmo sensor. Por exemplo, a coleção LANDSAT/LC08/C01/T1\_TOA contém todas as imagens Landsat 8 de refletância de topo da atmosfera (TOA).

A manipulação de coleções é essencial em análises ambientais, permitindo explorar séries temporais de vegetação, monitorar eventos climáticos ou estudar mudanças de uso da terra.

## Criação de Coleções de Imagens

Coleções podem ser criadas de três formas principais:

### a) A partir de um ID de coleção já existente no catálogo:

```
var collection = ee.ImageCollection('LANDSAT/LC08/C02/T1_TOA');
```

### b) A partir de uma lista de imagens:

```
var img1 = ee.Image('LANDSAT/LC08/C02/T1_TOA/LC08_221077_20250730');  
var img2 = ee.Image('LANDSAT/LC08/C02/T1_TOA/LC08_222077_20250502');  
var newCollection = ee.ImageCollection([img1, img2]);
```

### c) Usando o método fromImages():

```
var newCollection2 = ee.ImageCollection.fromImages([img1, img2]);
```

### d) Mesclando coleções existentes:

```
var collection1 = ee.ImageCollection('LANDSAT/LC08/C02/T1_TOA')  
.filterBounds(roi) // Filtro espacial  
.filterDate('2025-04-01', '2025-08-01') // Filtro temporal  
.filter(ee.Filter.lt('CLOUD_COVER', 10)) // Filtro de nuvens
```

```

var collection2 = ee.ImageCollection('COPERNICUS/S2_SR_HARMONIZED')
    .filterBounds(roi) // Filtro espacial
    .filterDate('2025-04-01', '2025-08-01') // Filtro temporal
    .filter(ee.Filter.lt('CLOUDY_PIXEL_PERCENTAGE', 10)) // Filtro de nuvens
var merged = collection1.merge(collection2);

```

## Filtragem de Coleções de Imagens

As coleções podem ser refinadas por critérios espaciais, temporais ou por atributos dos metadados.

- **Límite espacial (filterBounds):**

```

var collection = ee.ImageCollection('LANDSAT/LC08/C02/T1_TOA');
var point = ee.Geometry.Point([-50.6164, -24.3244]);
var spatialFiltered = collection.filterBounds(point);

```

- **Intervalo temporal (filterDate):**

```

var start = ee.Date('2014-06-01');
var end = ee.Date('2014-10-01');
var temporalFiltered = collection.filterDate(start, end);

```

- **Atributos de metadados (filter):**

```

var pathrowFiltered = collection
    .filter(ee.Filter.eq('WRS_PATH', 15))
    .filter(ee.Filter.eq('WRS_ROW', 30))
    .filter(ee.Filter.lte('CLOUD_COVER', 10)); // até 10% de nuvens

```

## Organização e Acesso a Metadados

Os metadados das imagens armazenam informações importantes (data de aquisição, porcentagem de nuvens, ângulo solar, etc.).

- **Ordenação por propriedade:**

```

var sortedByCloud = collection.sort('CLOUD_COVER'); // da menor à maior
var recent = collection.sort('system:time_start', false).limit(10); // 10 mais recentes

```

- **Selecionando a primeira imagem da coleção filtrada:**

```

var bestImage = sortedByCloud.first();
var mostRecentImage = recent.first();

```

- **Impressão de metadados no Console:**

```

print('Coleção filtrada:', temporalFiltered);
print('Primeira imagem:', bestImage);

```

- **Tamanho da coleção:**  

```
print('Número de imagens:', collection.size());
```
- **Convertendo em lista para manipulação:**  

```
var asList = collection.toList(100);
print('Tamanho da lista:', asList.length());
```
- **Estatísticas de propriedades:**  

```
var sunStats = collection.aggregate_stats('SUN_ELEVATION');
print('Estatísticas de elevação solar:', sunStats);
```

## 2.7. Carregamento de Dados do Usuário (Assets)

O Google Earth Engine (GEE) possui um vasto repositório de imagens de satélite e dados geoespaciais prontos para uso. Entretanto, em muitos projetos é necessário trabalhar com dados próprios, como imagens geradas por drones, dados de inventário florestal ou resultados de pré-processamentos externos (e.g., correções atmosféricas realizadas em softwares especializados).

Para isso, o GEE disponibiliza a funcionalidade de upload de Assets — arquivos que ficam armazenados na pasta de usuário na nuvem, podendo ser acessados e utilizados em qualquer script.

### Formatos Suportados

- Atualmente, o GEE aceita apenas *rasters* nos formatos GeoTIFF e TFRecord, e vetores nos formatos Shapefile e CSV.
- Cada arquivo pode ter, no máximo, 10 GB.
- Para Shapefile, o upload deve ser feito em um arquivo .zip compactado ou selecionando os arquivos descritos na tela de carregamento.

### Procedimento de Upload

O carregamento de dados é realizado diretamente pela interface do *Code Editor*.

- Clique no botão “NEW” localizado na aba *Assets*, no painel esquerdo.
- Selecione a opção “*Image Upload*” ou “*Table Upload*”.

- c) Na caixa de diálogo que se abre, clique em “SELECT” e navegue até o arquivo armazenado localmente no seu computador.
- d) Atribua um Asset ID único.  
Exemplo: /users/nome\_usuario/novo\_arquivo.
- e) Para salvar dentro de uma pasta ou coleção já existente, selecione clicando ao lado de “Asset Name”.
- f) Clique em “OK” para iniciar o *upload*.

## Monitoramento da Tarefa

Após iniciar o processo de upload, o GEE cria uma tarefa de ingestão de Asset, que aparece na aba *Tasks*, no painel direito:

- Ao clicar sobre a tarefa, é possível visualizar informações detalhadas sobre seu progresso e status.
- Caso necessário, a operação pode ser cancelada clicando em *Cancel*.
- O tempo de ingestão depende do tamanho do arquivo e da velocidade de conexão à internet.

## Acesso ao Asset após o Upload

Quando a ingestão é concluída com sucesso:

- O arquivo passa a constar na aba *Assets*, identificado por um ícone correspondente ao tipo de dado (por exemplo, *raster* ou *vetor*).
- A partir daí, o usuário pode:
  - Visualizar uma prévia e consultar detalhes clicando diretamente sobre o Asset.
  - Importar o Asset para o script atual, clicando no ícone de Import.

Exemplo de uso em script:

```
// Importando um asset previamente carregado
var imagem_usuario = ee.Image('users/nome_usuario/novo_arquivo');

// Centralizando a visualização
Map.centerObject(imagem_usuario, 10);
```

```
// Exibindo na tela
Map.addLayer(imagem_usuario, {bands: ['B4', 'B3', 'B2'], min: 0, max:
3000}, 'Imagen do usuário');
```

### **3. TASSELED CAP**

Nesta aula, será desenvolvida uma aplicação prática do *Tasseled Cap Transformation* (CRIST, 1985) com foco em realçar a vegetação por meio de uma composição colorida específica. Para isso, será construída uma imagem combinando a banda vermelha (B4), a banda do infravermelho próximo (B5) e o componente *greenness* derivado da transformação. Essa composição permitirá destacar de maneira mais clara as áreas de vegetação, evidenciando o vigor e a densidade da cobertura vegetal em contraste com solos expostos e superfícies urbanas. Assim, o produto final da atividade será uma visualização sintética que amplia a interpretação da paisagem, ressaltando os padrões de vegetação a partir de um arranjo espectral e transformado. Serão descritas todas as ações para a confecção deste procedimento.

#### **3.1. Definição da área e seleção da coleção**

Iniciou-se declarando uma área de interesse (`roi`) e carregando a coleção LANDSAT/LC08/C02/T1\_TOA (CRAWFORD et al., 2023). Optou-se pelos dados TOA (*Top Of Atmosphere*) (CHANDER et al., 2009), nos quais os valores já se encontram em reflectância no topo da atmosfera, adequados para transformações espetrais clássicas como o *Tasseled Cap*. Em seguida, aplicou-se filterBounds(roi) para garantir que apenas cenas que cobrem a ROI fossem consideradas. Para o exemplo, será utilizada a cena inteira.

#### **3.2. Janela temporal e ordenação por nebulosidade**

Determinou-se uma janela temporal entre 15 de janeiro e 1º de março de 2024, visando capturar datas próximas ao período de referência do exercício. Em seguida, ordenaram-se as imagens por `CLOUD\_COVER`, priorizando as menos nubladas. Ao final, utilizou-se `first()` para recuperar a cena mais limpa segundo esse critério, assegurando um insumo de melhor qualidade para as etapas subsequentes.

### **3.3. Inspeção e enquadramento da cena**

Imprimiu-se no Console a imagem selecionada para verificação de metadados (ID, cobertura de nuvens, entre outros), e centralizou-se o mapa usando `Map.centerObject(image, 8)` . Esse enquadramento facilitou a conferência visual e a validação rápida da extensão espacial da cena e de sua adequação sobre a ROI.

### **3.4. Composição RGB de referência (B4-B3-B2)**

Adicionou-se uma visualização *true color* com as bandas B4 (red), B3 (green) e B2 (blue), com stretch (`min: 0.02` , `max: 0.4` ) apropriado para realçar contraste sem saturar alvos claros. Essa camada serviu como referência visual para comparar, depois, os padrões derivados do *Tasseled Cap* com a aparência espectral “natural” da cena.

### **3.5. Fundamentos do *Tasseled Cap* para Landsat 8 TOA**

Implementou-se a função tasseledCapL8TOA(img) conforme coeficientes propostos por BAIG et al. (2014) para dados TOA do OLI. Selecionaram-se as bandas espectrais B2, B3, B4, B5, B6 e B7, que cobrem do azul ao SWIR2. O *Tasseled Cap* realiza uma transformação linear (produto interno) dessas bandas para componentes que sintetizam variações espectrais dominantes relacionadas à luminosidade (*brightness*), vigor vegetal (*greenness*) e umidade/estrutura (*wetness*), além de componentes adicionais (*fourth*, *fifth*, *sixth*) menos utilizadas em aplicações de rotina.

### **3.6. Cálculo das componentes por produto interno**

Para cada componente, definiu-se um vetor de pesos (p. ex., `BRIGHT`, `GREEN` , `WET` ) e executou-se `b.multiply(COEF).reduce(ee.Reducer.sum())` . Esse procedimento implementou o \*\*produto escalar\*\* banda-a-banda que caracteriza o *Tasseled Cap*: cada pixel tem suas seis bandas ponderadas e somadas, retornando um valor escalar por componente. Ao final, concatenou-se as bandas em uma imagem multibanda contendo `brightness` , `greenness` , `wetness` , `fourth` , `fifth` e `sixth` . O resultado das principais bandas da transformação *Tasseled Cap* (TCT) está demonstrado na Figura 3.

### **3.7. Interpretação das componentes principais**

A *brightness* respondeu, em geral, ao albedo/clareza da superfície (solos expostos, áreas urbanas, superfícies secas). A *greenness* correlacionou-se com vigor e densidade de vegetação (maiores valores em áreas com forte resposta no NIR e absorção no vermelho). A *wetness* refletiu umidade e estrutura da cobertura (sensível à água livre, solo/vegetação úmidos e a contrastes entre NIR e SWIR). Essas interpretações, consolidadas na literatura, facilitaram diagnósticos de uso e cobertura do solo e monitoramento fenológico.

### **3.8. Configuração de paletas e faixas de visualização**

Definiram-se paletas de cores para guiar a leitura visual: uma paleta “geral” multicolorida e uma paleta graduada para *greenness* (tonalidades do amarelo-claro ao verde-escuro), favorecendo a percepção de gradientes de vegetação. As faixas de visualização (`min: -1` , `max: 2`) foram sugeridas como ponto de partida; enfatizou-se que o ajuste fino depende do contraste espectral da cena/bioma e pode requerer testes interativos no Inspector.

### **3.9. Renderização das camadas TCT no mapa**

Adicionaram-se ao mapa as camadas `brightness` , `greenness` e `wetness` com seus respectivos parâmetros de visualização. Essa etapa permitiu comparar lado a lado o *true color* e as componentes TCT, destacando padrões não triviais (p. ex., áreas de solo exposto com alta *brightness*, mosaicos agrícolas com *greenness* variável, várzeas/umidades elevadas realçadas em *wetness*).

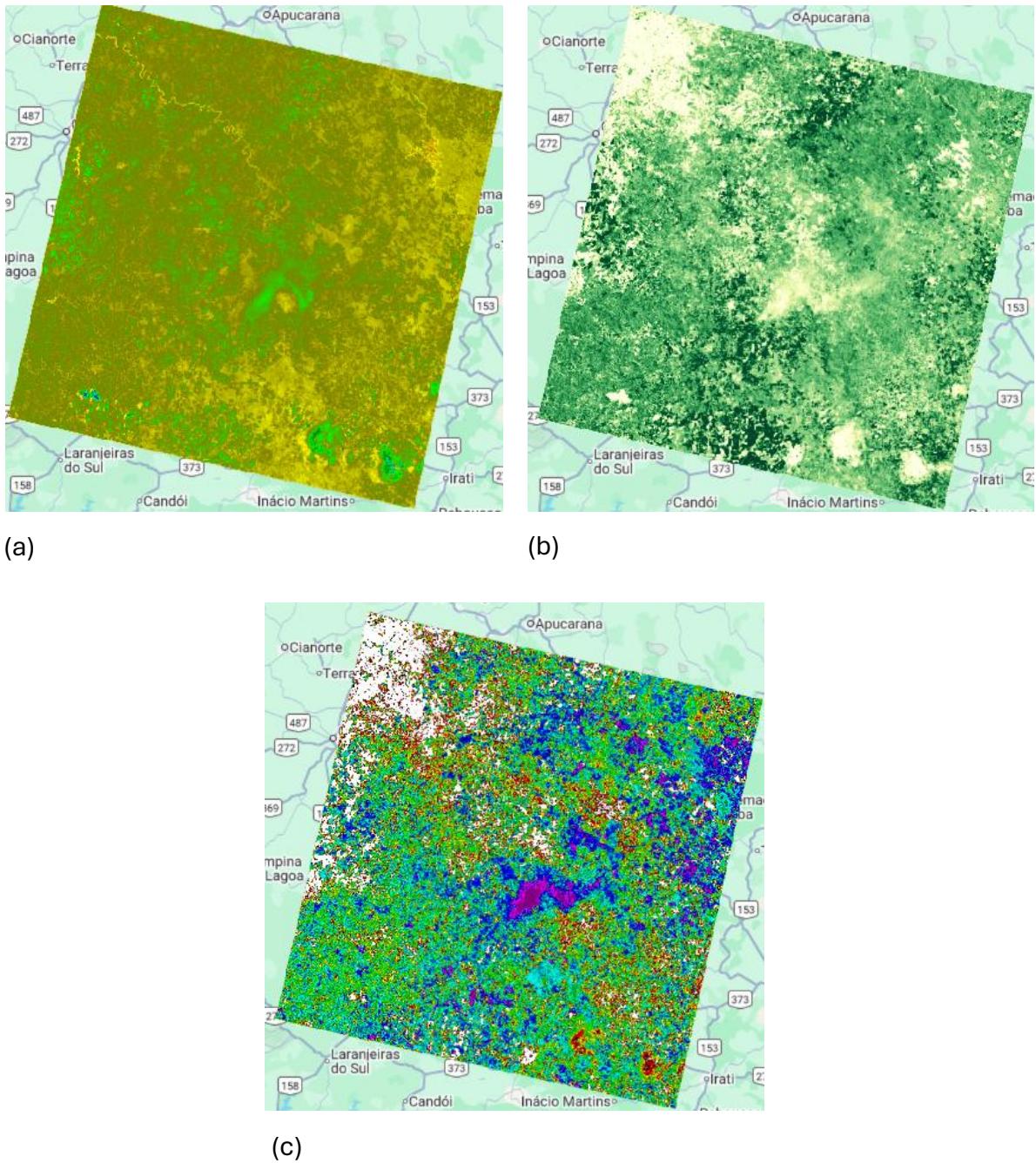


Figura 3 - Exemplo dos resultados da aplicação da transformação *Tasseled Cap* (TCT) e visualização das camadas principais (*Brightness* (a), *Greenness* (b), *Wetness* (c)).

### 3.10. Boas práticas e variações opcionais

Indicou-se a possibilidade de recortar (`clip(roi)` ) a imagem TCT para acelerar renderização e focar a análise. Também se apontou que os coeficientes são específicos para Landsat 8 TOA; ao trabalhar com L2/SR (reflectância de superfície), devem-se empregar coeficientes apropriados a SR. Por fim, reforçou-se que as componentes adicionais (*fourth*, *fifth*, *sixth*) podem trazer nuances úteis em

classificações/segmentações, embora, em aplicações operacionais, a tríade *brightness/greenness/wetness* costume ser suficiente.

### 3.11. Validação e comparação temporal/espacial

Sugeriu-se, para estudos aplicados, comparar as componentes TCT ao longo de séries temporais (monitorando sazonalidade/fenologia) ou entre áreas de referência (ROIs com rótulos conhecidos). Essas estratégias auxiliaram na calibração de limiares, no entendimento de assinaturas ambientais locais e na geração de indicadores robustos ligados à vegetação e à umidade superficial.

Por fim, cabe ressaltar que o TCT é uma transformação linear: sua eficácia depende da coerência radiométrica das entradas e da pertinência dos coeficientes ao sensor/nível de correção. Atmosfera remanescente, sombras topográficas e heterogeneidade de iluminação podem afetar as componentes. Assim, para produtos cartográficos e classificações, recomendou-se integrar o TCT com máscaras de nuvem/sombra, correções topográficas quando cabíveis, e validação com dados de campo ou rótulos independentes.

A Figura 4 apresenta a composição colorida com as bandas R, NIR e *greenness* na ordem RGB.

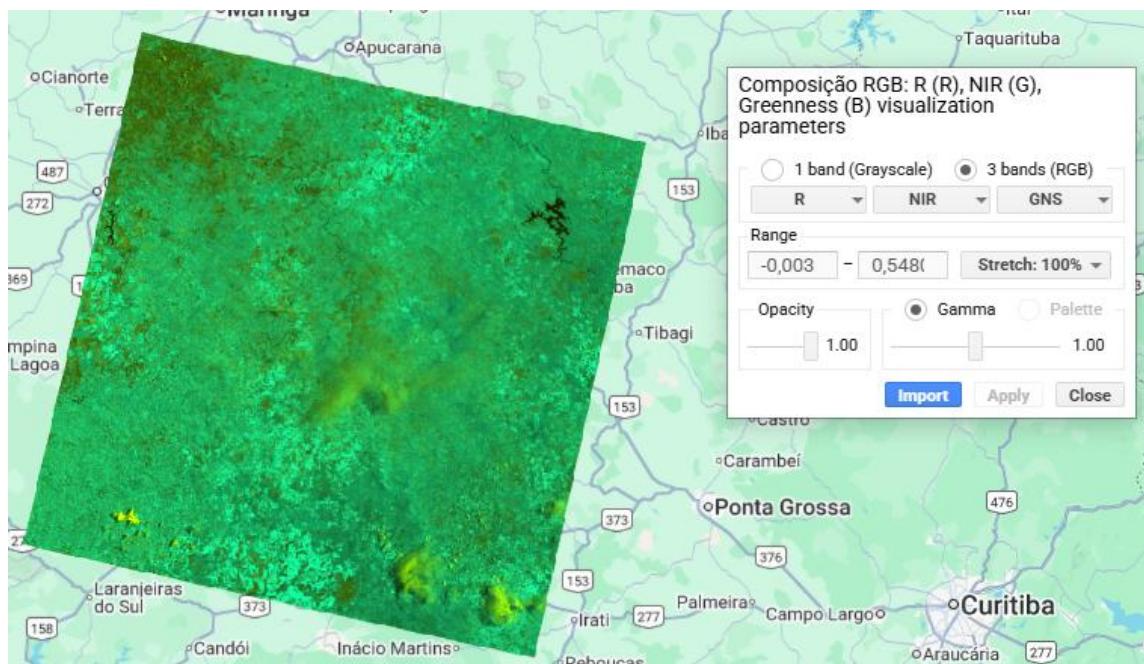


Figura 4 - Composição colorida com as bandas R, NIR e *greenness* na ordem RGB.

## **4. COMPOSIÇÃO DE IMAGENS, MANIPULAÇÃO DE HISTOGRAMA E CÁLCULO DE ÍNDICES DE VEGETAÇÃO**

Nesta aula, executou-se a importação da região ou área de interesse, depois foram selecionadas as imagens do satélite Sentinel-2 e realizamos alguns procedimentos de pré-processamento para a mesma. Fez-se também o recorte para a área de interesse e em seguida, elaborou-se a composição colorida utilizando algumas bandas. Por fim, alguns exemplos de índices de vegetação foram construídos e seus respectivos histogramas foram avaliados. A seguir serão detalhados os procedimentos.

### **4.1. Importação da área de interesse (ROI)**

No *script*, isso pode ser feito ao carregar um polígono armazenado no usuário do *Google Earth Engine* no *Assets*. Para os exemplos do livro, no entanto, visando facilitar as práticas e não precisar realizar ajustes nos *scripts*, serão utilizadas as coordenadas dos vértices do polígono de interesse para realizar a avaliação. Para tanto, o *script* inicia com essa indicação da ‘roi’.

```
var roi = ee.Geometry.Polygon([
  [
    [-50.50514984906117, -24.328943567577905], // SW
    [-50.42378235638539, -24.328943567577905], // SE
    [-50.42378235638539, -24.274811722892963], // NE
    [-50.50514984906117, -24.274811722892963], // NW
    [-50.50514984906117, -24.328943567577905] // fechar polígono
  ]
]);
```

### **4.2. Seleção da coleção de imagens Sentinel-2**

Em seguida, fez-se o acesso à coleção Sentinel-2 SR *Harmonized* (CLAVERIE et al., 2018), que contém imagens do sensor MSI com correção atmosférica de superfície (*Surface Reflectance*). Esse passo envolve aplicar filtros de data, de localização e de cobertura de nuvens, de forma a obter a imagem mais adequada para o período e para a região analisada. O critério adotado neste exercício foi selecionar a cena com menor

porcentagem de nuvem, garantindo melhor qualidade visual e menor interferência nos índices espectrais.

#### **4.3. Pré-processamento radiométrico e recorte da cena**

As imagens do Sentinel-2 são fornecidas em valores inteiros (Números Digitais – DN), geralmente multiplicados por um fator de escala. No caso da coleção SR, esse fator é 10.000, e por isso os valores precisam ser ajustados para refletâncias entre 0 e 1. Além disso, a cena foi cortada (*clipping*) para a ROI, assegurando que apenas os pixels dentro da área de estudo fossem processados e visualizados. Essa etapa foi fundamental para reduzir o volume de dados e concentrar a análise no espaço de interesse.

#### **4.4. Composições coloridas**

Após o pré-processamento, foram geradas diferentes composições coloridas para realçar características específicas da superfície. A composição RGB (B4, B3, B2) representa a visão em cores verdadeiras, semelhante à observada pelo olho humano. Já a composição NIR (B8, B4, B3) utiliza o infravermelho próximo e evidencia a vegetação saudável, que reflete fortemente nessa faixa espectral. Outra composição, usando as bandas SWIR, é aplicada para destacar diferenças de umidade e saúde da vegetação. Essas composições permitem interpretar aspectos distintos do ambiente e são amplamente utilizadas em sensoriamento remoto.

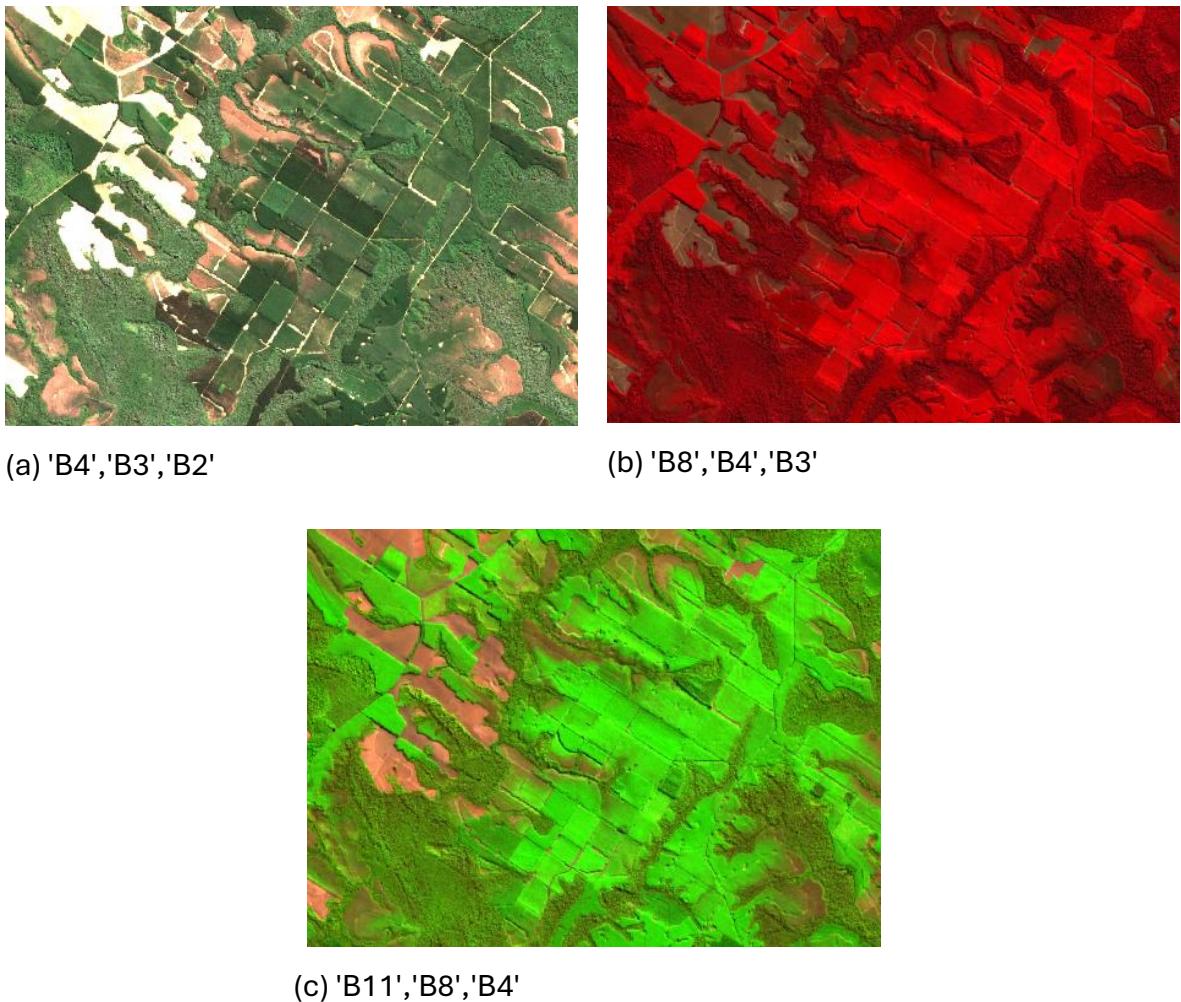


Figura 5 - Composição colorida das bandas do Sentinel-2 SR *Harmonized*

#### 4.5. Cálculo de índices de vegetação

Os índices de vegetação foram construídos a partir de expressões matemáticas que combinam bandas espectrais para realçar as condições da vegetação. O NDVI (*Normalized Difference Vegetation Index*), que em português é Índice de Vegetação por Diferença Normalizada é o mais conhecido e mede a diferença entre a alta reflectância do infravermelho próximo (NIR) e a baixa reflectância do vermelho (RED), indicando vigor vegetativo (ROUSE, 1973). O MCARI1 (*First Modified Chlorophyll Absorption Ratio Index*), ou Primeiro Índice de Razão de Absorção de Clorofila Modificado, é utilizado para estimar o conteúdo de clorofila da vegetação e o vigor foliar (HABOUDANE et al., 2004). O RDVI (*Renormalized Difference Vegetation Index*), ou Índice de Vegetação por Diferença Renormalizada, é usado para monitorar vigor e cobertura vegetal (ROUJEAN;

BREON, 1995). O EVI (*Enhanced Vegetation Index*), que em português é chamado de Índice de Vegetação Melhorado, visa aprimorar a sensibilidade em áreas de alta biomassa, reduzindo o efeito de saturação observado no NDVI (HUETE et al., 2002). Já o SAVI (*Soil Adjusted Vegetation Index*), traduzido como Índice de Vegetação Ajustado ao Solo incorpora um fator de correção que minimiza a influência do solo exposto, tornando-o útil em áreas de cobertura vegetal mais rala (HUETE, 1988). Cada índice traz informações complementares sobre a dinâmica da vegetação.

Índice	Fórmula	Descrição
EVI ( <i>Enhanced Vegetation Index</i> )	$2,5 \cdot \frac{NIR - R}{NIR + 6 \cdot R - 7,5 \cdot B + 1}$	Valores próximos de 1 → vegetação saudável e densa Valores médios → vegetação intermediária Valores próximos de 0 ou negativos → ausência de vegetação (solo exposto, água, áreas urbanas)
MCARI1 ( <i>First Modified Chlorophyll Absorption Ratio Index</i> )	$1,2 \cdot \left[ \frac{2,5 \cdot (NIR - R)}{-1,3 \cdot (NIR - G)} \right]$	Valores mais altos indicam maior absorção e, portanto, maior concentração de clorofila, o que geralmente está associado a vegetação saudável e vigorosa. Valores mais baixos indicam concentração de clorofila menor, o que pode corresponder a vegetação estressada, com menor atividade fotossintética, ou em fases de desenvolvimento mais precárias.
NDVI ( <i>Normalized Difference Vegetation Index</i> )	$\frac{NIR - R}{NIR + R}$	Valores próximos de 1 → vegetação densa e saudável Valores próximos de 0 → solo exposto, áreas urbanas ou vegetação rala Valores negativos → corpos d'água, neve, nuvens densas
SAVI ( <i>Soil Adjusted Vegetation Index</i> )	$1,5 * \frac{NIR - R}{NIR + R + 0,5}$	Valores próximos de 1 → vegetação vigorosa e densa. Valores intermediários → vegetação moderada. Valores próximos de 0 → pouco ou nenhum tipo de vegetação (solo exposto, áreas urbanas, corpos d'água).
RDVI ( <i>Renormalized</i> )	$\frac{NIR - R}{\sqrt{NIR + R}}$	Valores próximos de 1 indicam vegetação saudável; Valores próximos a 0 indicam pouca

<i>Difference Vegetation Index)</i>		vegetação; Utilizado para moderar limitações do NDVI em áreas com vegetação esparsa a moderada.
-------------------------------------	--	--

Em que: R é vermelho (Red); G é verde (Green); B é azul (Blue); NIR é infravermelho próximo (Near Infrared Reflectance); RE é borda do vermelho (Red Edge); L é o fator de correção do brilho do solo (L igual a 0,25 indica vegetação densa; 0,5 é indicativo de vegetação com densidade intermediária; quando L é igual a 1, a vegetação é rala ou com baixa densidade).

#### 4.6. Visualização dos índices no mapa

Cada índice calculado foi visualizado no mapa com paletas de cores que auxiliam na interpretação. Geralmente, valores negativos ou próximos de zero indicam corpos d'água, áreas urbanas ou solo exposto, enquanto valores positivos crescentes (até próximo de 1) representam vegetação com maior densidade e saúde (vigor). Essa etapa permite comparar espacialmente o comportamento dos diferentes índices, possibilitando a identificação de padrões de uso e cobertura da terra dentro da ROI. A Figura 6 apresenta os resultados dos índices de vegetação para a área de estudo.

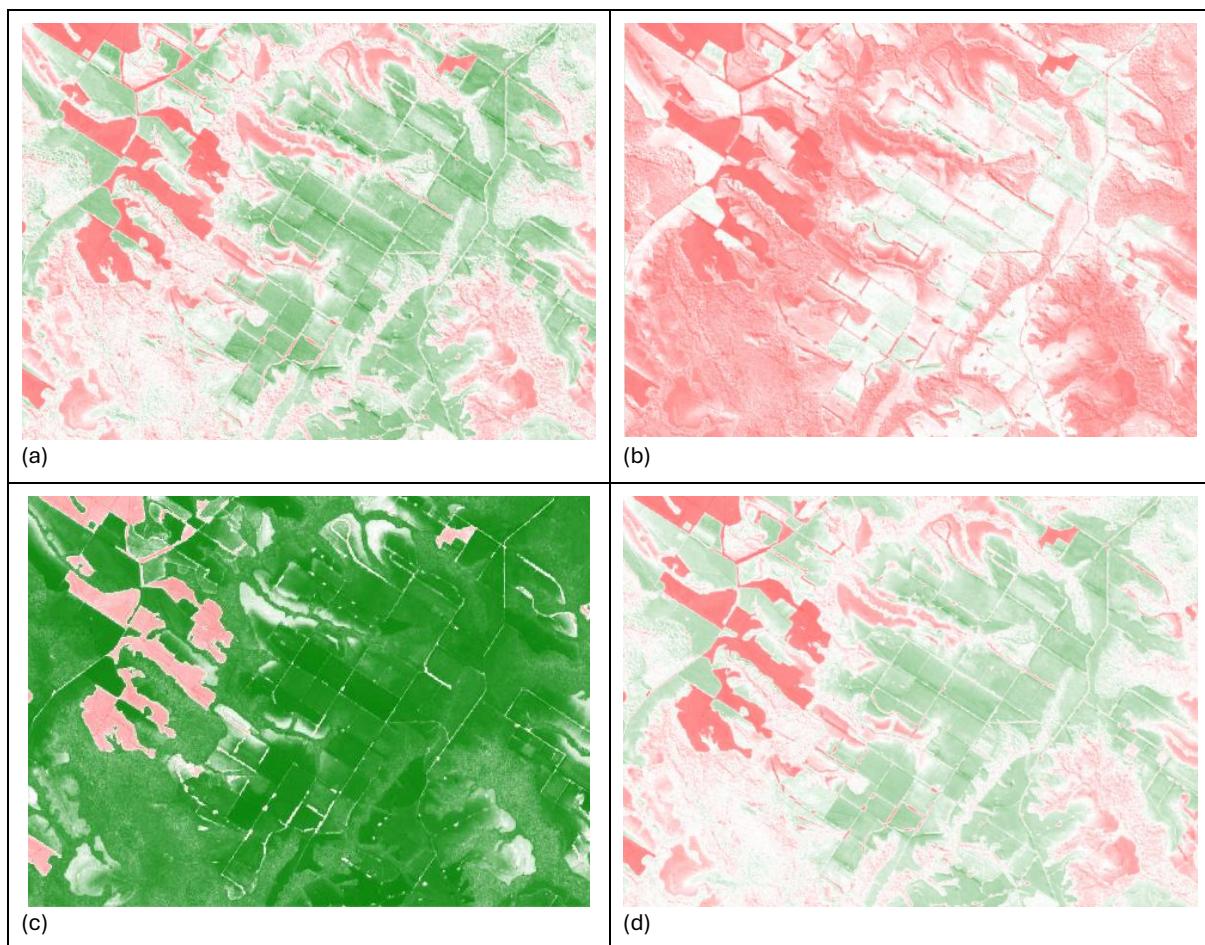




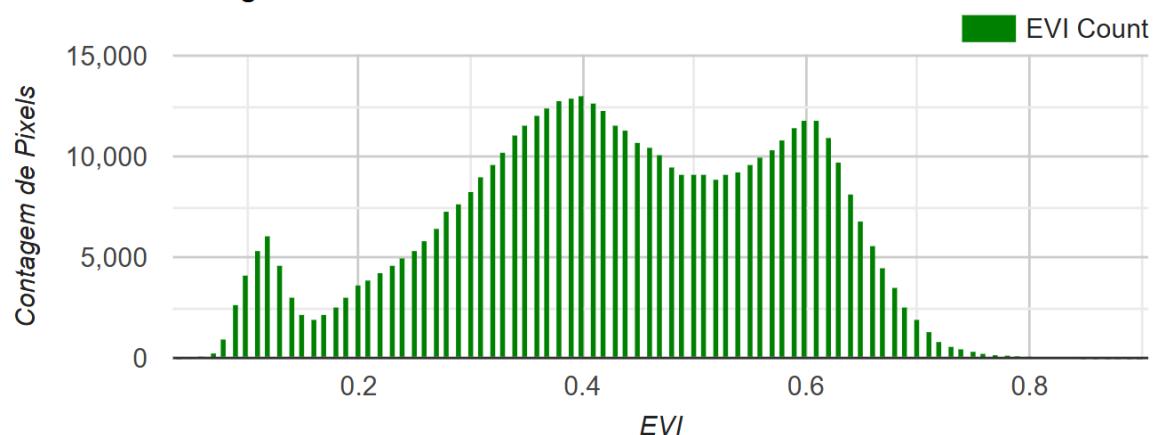
Figura 6 - Resultados dos índices de vegetação para a área de estudo; (a) EVI, (b) MCARI1; (c) NDVI, (d) RDVI. (e) SAVI

#### 4.7. Construção do histograma

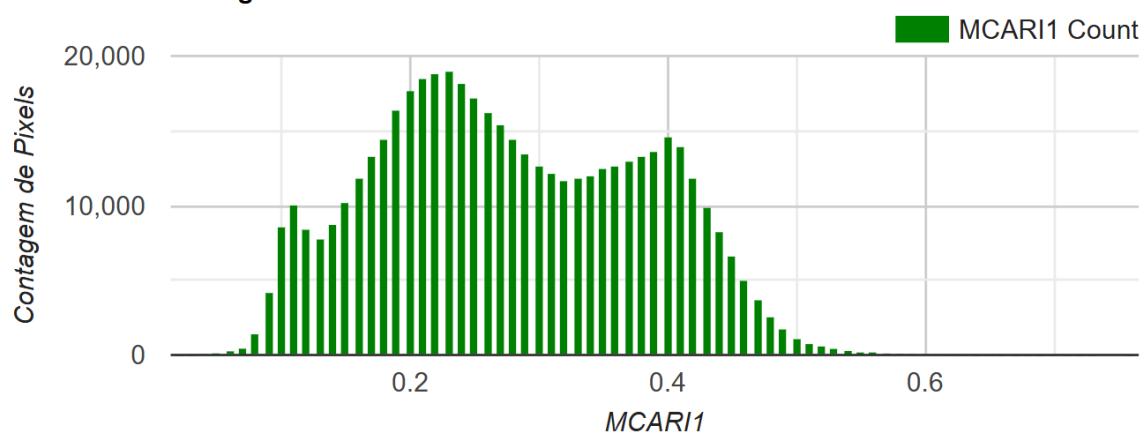
Em seguida, observou-se a distribuição estatística dos valores de um índice na área estudada, neste exemplo, usamos o histograma do NDVI que apresenta a frequência de pixels em diferentes intervalos de valores (Figura 7). Essa ferramenta estatística auxilia na interpretação da condição predominante da vegetação, permitindo observar, por exemplo, se a maior parte da área apresenta vegetação densa, intermediária ou rala. Também foi possível identificar valores extremos, que podem representar corpos d'água, áreas urbanas ou ruídos na imagem. Caso deseje selecionar outro índice de vegetação para explorar seu histograma, basta selecionar seu nome em substituição ao nome NDVI no código.

Desta forma, foi demonstrado um fluxo completo de análise de imagens orbitais, que vai desde a seleção de cenas até a extração de informações quantitativas sobre a vegetação. O exercício possibilita compreender os efeitos da escolha de composições coloridas, avaliar as diferenças entre alguns dos índices espectrais e interpretar estatisticamente a condição da cobertura vegetal. Essa abordagem alia técnicas de processamento digital de imagens com ferramentas de análise estatística, fornecendo subsídios para estudos de monitoramento ambiental, inventários florestais, agricultura de precisão e diversas outras aplicações na Engenharia Florestal.

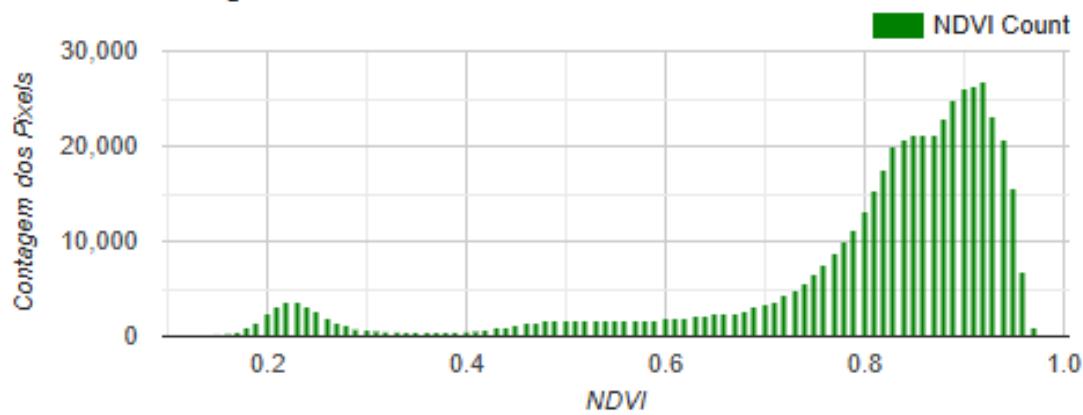
**Histograma de EVI**



**Histograma de MCARI1**



**Histograma do NDVI**



Continua...

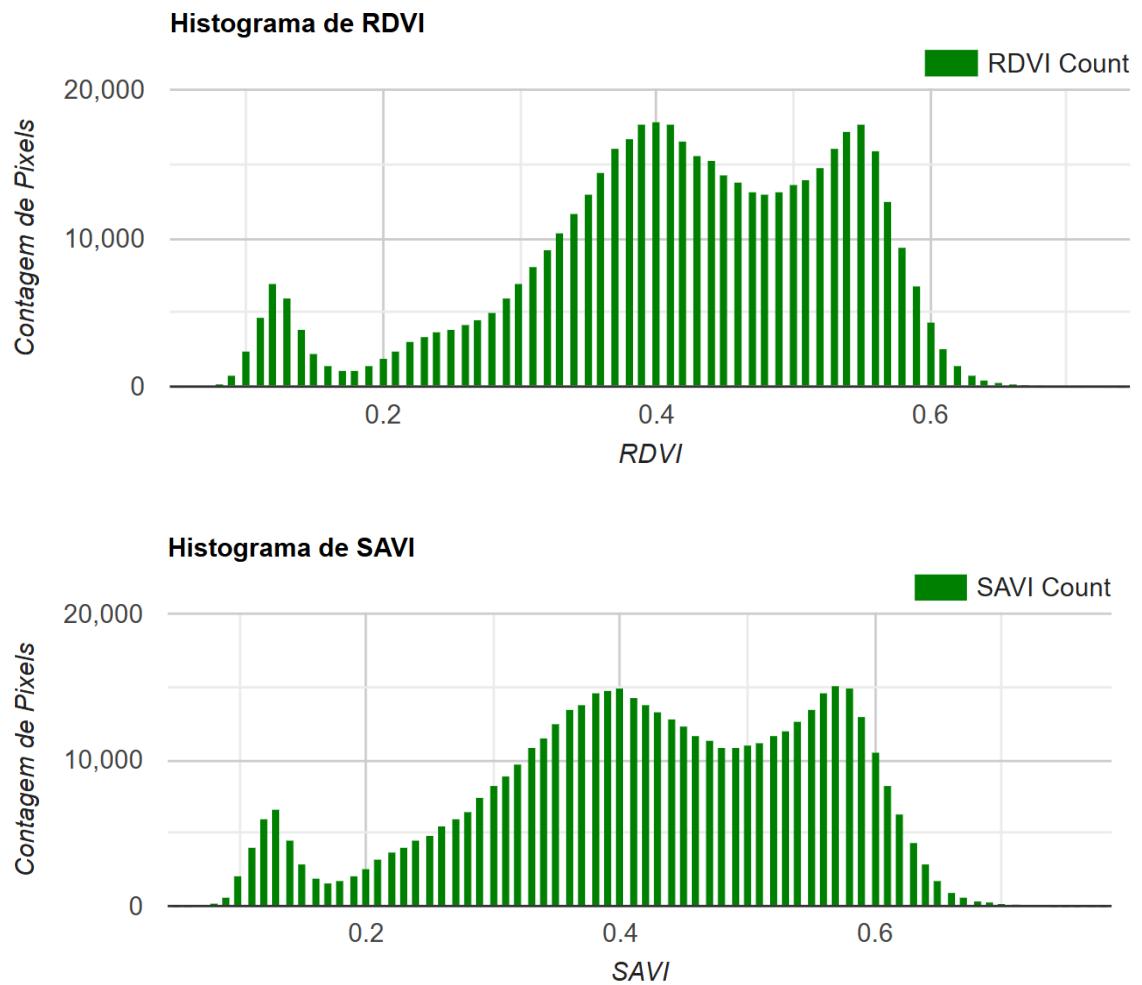


Figura 7 - Exemplo do histograma do NDVI gerado para a área de interesse.

## **5. ELABORAÇÃO DE MAPAS DE MUDANÇAS TEMPORAIS (DETECÇÃO DE MUDANÇAS)**

Nesta atividade, iniciou-se com a definição da região de interesse (ROI) a ser analisada. Em seguida, foram selecionadas imagens do satélite Sentinel-2 para dois períodos distintos e aplicados procedimentos de pré-processamento, como a máscara de nuvens e a conversão dos valores para reflectância. Posteriormente, elaborou-se composições representativas para cada período, recortadas à área de estudo. A partir dessas imagens, foi calculado o índice de vegetação NDVI para cada data e, por fim, obteve-se a diferença entre os períodos ( $\Delta$ NDVI), possibilitando avaliar ganhos e perdas de vegetação. A seguir, cada uma dessas etapas será detalhada.

### **5.1. Definição da área de estudo**

O primeiro passo foi delimitar a região de interesse (ROI – *Region of Interest*). Será utilizada a mesma área de interesse do exemplo anterior.

### **5.2. Máscara de nuvens e sombras**

As imagens de satélite estão sujeitas à presença de nuvens, sombras e cirros, que prejudicam a análise espectral da superfície. Para corrigir esse problema, o script aplicou uma função de máscara que combina a banda de probabilidade de nuvem (MSK\_CLDPRB) e a banda de classificação da cena (SCL). Assim, pixels classificados como nuvem ou sombra foram removidos, permitindo que apenas valores confiáveis de reflectância da superfície sejam usados nos cálculos subsequentes.

### **5.3. Normalização da reflectância**

Após aplicar a máscara, o script dividiu os valores da imagem por 10.000. Essa operação converteu os valores digitais originais em reflectância de superfície em escala real (0–1). Esse passo foi fundamental, pois índices de vegetação e outros cálculos espectrais são definidos teoricamente sobre reflectâncias, não sobre valores digitais brutos.

## 5.4. Seleção temporal e composição de imagens

O script definiu dois períodos distintos (junho a agosto de 2023 e junho a agosto de 2025) para análise. Em cada período, uma coleção de imagens Sentinel-2 foi filtrada pela área de estudo, pela data e por baixa cobertura de nuvens. Em seguida, aplicou-se a máscara e calculou-se a mediana da coleção, criando um compósito representativo do período. A escolha da mediana deu-se para reduzir ruídos residuais, eliminar efeitos de pixels anômalos e garantir uma síntese mais robusta do comportamento espectral da região. A Figura 8 apresenta as composições da coleção *Harmonized Sentinel-2 MSI* ('B4', 'B3', 'B2') para os anos 2023 e 2025.

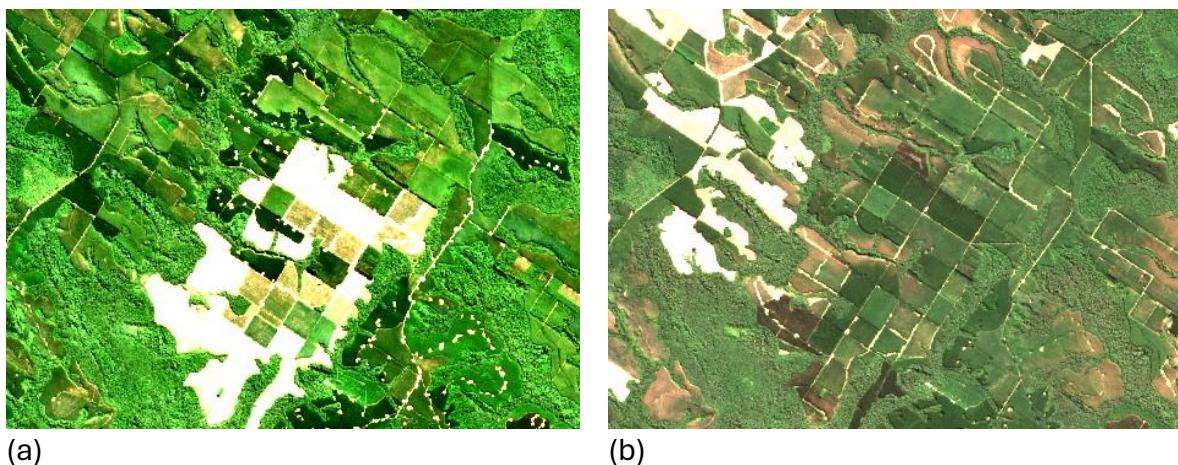


Figura 8 - Composição colorida da coleção *Harmonized Sentinel-2 MSI* ('B4', 'B3', 'B2').

## 5.5. Cálculo do índice de vegetação NDVI

O índice de vegetação por diferença normalizada (NDVI) foi calculado para cada composição temporal. Como já explicado, esse índice compara a reflectância do infravermelho próximo (B8) com a do vermelho (B4), variando de -1 a +1. Valores elevados indicam vegetação densa e saudável, enquanto valores baixos ou negativos indicam ausência de vegetação, água ou solo exposto. O NDVI é amplamente utilizado por sua simplicidade e eficiência na avaliação do vigor vegetativo, mas apresenta problemas relacionados à saturação quando de vegetação de porte florestal muito densa.

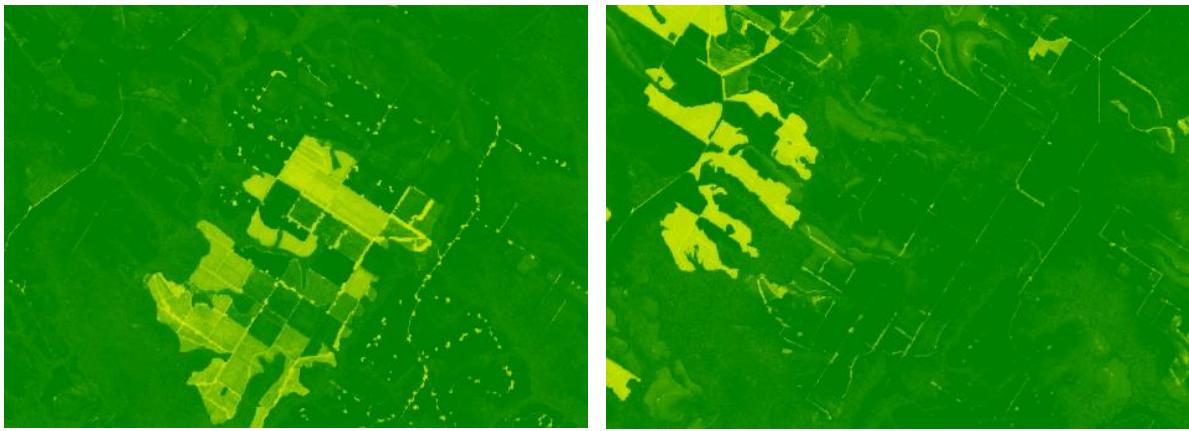
## **5.6. Detecção de mudança com $\Delta$ NDVI**

Para identificar alterações na vegetação entre 2023 e 2025, o script calculou a diferença entre os dois índices, o qual chamou-se de  $\Delta$ NDVI (Figura 9). Esse produto mostrou diretamente os ganhos e perdas de vegetação no período. Valores positivos significam aumento do vigor vegetativo (possível regeneração ou reflorestamento), enquanto valores negativos indicam redução (desmatamento, degradação, incêndios ou colheita florestal). Assim, o  $\Delta$ NDVI é uma métrica simples, mas que pode auxiliar muito no monitoramento temporal da cobertura florestal. Destaca-se que assim como foi realizada a mudança temporal para o NDVI, poderia ter sido realizado para outros índices de vegetação.

## **5.7. Visualização e interpretação dos resultados**

A etapa final do script definiu as paletas de cores para facilitar a interpretação dos mapas. O NDVI foi exibido em gradiente de vermelho (valores baixos) a verde (valores altos), enquanto o  $\Delta$ NDVI foi representado em escala de vermelho–branco–verde, indicando perda, estabilidade e ganho de vegetação, respectivamente. Além disso, as composições em cores naturais (RGB) para 2023 e 2025 são adicionadas, permitindo comparar visualmente a superfície terrestre e validar os resultados do índice.

Esse procedimento exemplificou um fluxo básico de análise de mudanças na vegetação a partir de imagens Sentinel-2. Ele poderia ser aplicado em estudos de desmatamento, regeneração natural ou monitoramento de áreas protegidas. Contudo, é importante considerar limitações: mudanças sazonais, variações atmosféricas residuais ou diferenças locais de clima podem influenciar o NDVI. Por isso, recomenda-se complementar a análise com dados de campo ou outros índices espectrais adicionais.



(a) NDVI 2023

(b) NDVI 2025



(b)  $\Delta$ NDVI

Figura 9 - Resultado do NDVI para os anos de 2023 e 2025 e do  $\Delta$ NDVI mostrando as alterações na vegetação.

Em seguida, estruturou-se um fluxo para calcular áreas em hectares a partir de uma imagem classificada. As classes foram definidas por intervalos do indicador (p.ex.,  $\Delta$ NDVI) conforme o dicionário classNames:

- 1) Perda forte ( $< TH1$ )
- 2) Perda moderada [ $TH1, TH2$ ]
- 3) Estável [ $TH2, TH3$ ]
- 4) Ganho moderado ( $TH3, TH4$ )
- 5) Ganho forte ( $> TH4$ )

A seguir, criou-se uma imagem contendo a área de cada pixel em hectares e essa informação foi associada à imagem das classes (Figura 10). Com base nessa combinação, aplicou-se um redutor zonal que somou as áreas correspondentes a cada classe dentro da região de interesse definida (ROI). Assim, obteve-se a área total ocupada por cada classe, em hectares, já levando em conta a resolução espacial da imagem utilizada. Posteriormente, os resultados foram organizados em uma tabela (FeatureCollection), contendo para cada classe o seu nome, a área total e a área arredondada. Essa tabela foi impressa no Console tanto em formato tabular (Tabela 1), facilitando a leitura, quanto em formato bruto, permitindo validações ou exportações futuras.

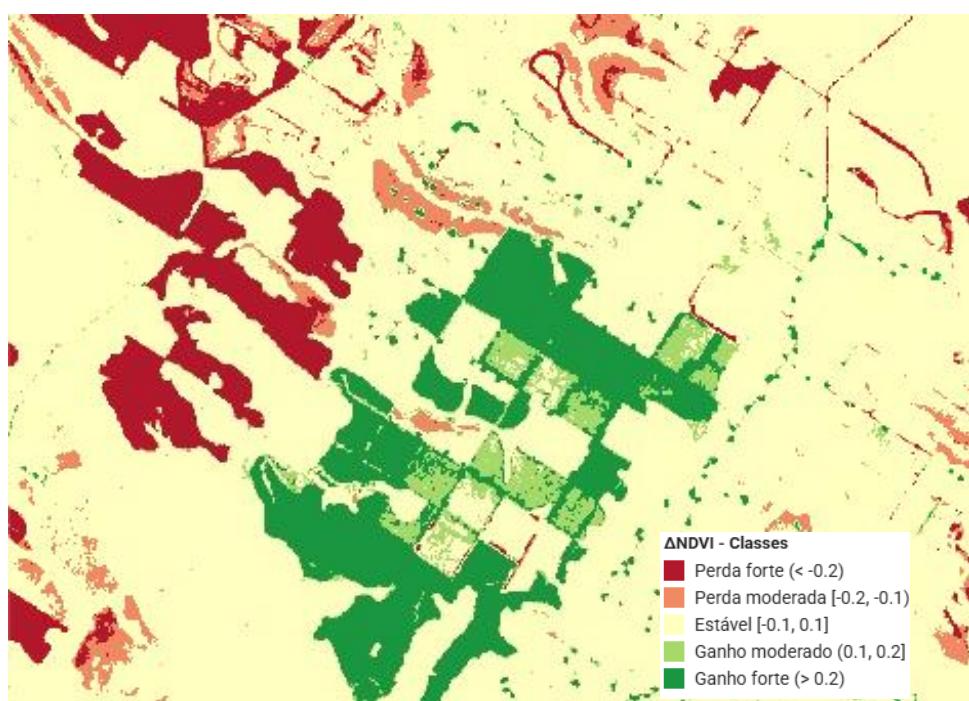


Figura 10 - Resultado do NDVI para os anos de 2023 e 2025 e do  $\Delta$ NDVI mostrando as alterações na vegetação.

Tabela 1 - Síntese das áreas por classe de mudanças conforme apontado na Figura 10.

Classe	Area_ha
Estável [-0.1, 0.1]	3,627.487
Ganho forte (> 0.2)	498.837
Ganho moderado (0.1, 0.2]	156.211
Perda forte (< -0.2)	402.547
Perda moderada [-0.2, -0.1]	266.58

## **6. CLASSIFICAÇÃO DE USO DO SOLO COM SENSORES ÓPTICOS E SAR (*Synthetic-aperture radar*)**

Nesta atividade, iniciou-se com a definição da região de interesse (ROI) a ser trabalhada. Em seguida, foram utilizadas imagens ópticas do satélite Landsat-9, sobre as quais foram aplicados procedimentos de pré-processamento, como a máscara de nuvens e a correção radiométrica, resultando em um compósito representativo. A partir da composição, calcularam-se índices espectrais de vegetação, como NDVI, EVI e SAVI. Paralelamente, foram incorporados dados de SAR do Sentinel-1, com as polarizações VV e VH (HAAS; BAN, 2017), a partir dos quais foram derivadas métricas adicionais, incluindo o SAR *Vegetation Index* (RVI) e relações entre polarizações (AGAPIOU, 2020; SOUDANI et al., 2021; NASIRZADEHDIZAJI et al., 2019; SCHLUND; ERASMI, 2020). Posteriormente, aplicou-se filtros de suavização, métricas de textura e estatísticas locais, tanto para dados ópticos quanto para os de SAR. Por fim, realizou-se a classificação supervisionada com o algoritmo *Random Forest* (BREIMAN, 2001) em três cenários distintos (somente SAR, somente óptico e a combinação SAR+óptico), permitindo a avaliação comparativa das acurárias. A seguir, cada uma dessas etapas será detalhada.

### **6.1. Definição da área de estudo**

O primeiro passo foi delimitar a região de interesse (ROI – *Region of Interest*). Será utilizada a mesma área de interesse do exemplo anterior.

### **6.2. Processamento das imagens ópticas Landsat-9**

Em seguida, foram selecionadas imagens do sensor OLI-2 a bordo do Landsat-9 (CHOATE et al., 2022). Para assegurar a qualidade dos dados, aplicou-se uma função de máscara que removeu pixels afetados por nuvens e sombras, utilizando a banda QA\_PIXEL. Essa etapa foi fundamental para minimizar a contaminação atmosférica e garantir que as reflectâncias representassem de forma fidedigna a superfície terrestre.

### **6.3. Correção radiométrica e geração do compósito óptico**

Após a aplicação da máscara, os valores digitais das bandas ópticas foram ajustados por fatores de escala, convertendo-os em reflectância de superfície. Em seguida, foi gerado um compósito representativo por meio da média das imagens filtradas no período de maio a agosto de 2025. Esse compósito permitiu sintetizar as condições espectrais médias da área de estudo durante a estação seca.

### **6.4. Cálculo dos índices de vegetação**

Calculou-se então os índices de vegetação: NDVI, EVI e SAVI. O NDVI foi utilizado para avaliar o vigor vegetativo de forma simples; o EVI foi aplicado para melhorar a sensibilidade em áreas densamente vegetadas e reduzir a influência atmosférica; e o SAVI foi calculado para compensar a interferência do solo exposto, especialmente em áreas de baixa cobertura vegetal. Esses índices serviram como variáveis auxiliares na análise da vegetação.

### **6.5. Processamento das imagens SAR Sentinel-1**

Além dos dados ópticos, foram incorporados dados de SAR provenientes do Sentinel-1, que opera em banda C. Foram selecionadas as polarizações VV e VH no modo IW e órbita descendente. As imagens foram filtradas para o mesmo intervalo temporal, construiu-se o mosaico e então, foram recortadas para os limites da área de interesse. Essa abordagem garantiu a integração de informações de sensoriamento ativo, que independe das condições de iluminação solar e apresenta maior resistência à presença de nuvens.

### **6.6. Derivação de métricas SAR**

A partir dos dados de SAR, foram extraídas métricas adicionais, como o SAR *Vegetation Index* (RVI), sensível à biomassa e à estrutura da vegetação, além de razões entre polarizações (VV/VH, VH/VV e combinações de diferenças). Essas métricas ampliaram a capacidade de discriminação de alvos, fornecendo informações sobre

rugosidade, umidade e estrutura da superfície terrestre. A Figura 11 apresenta os resultados das polarizações (VV e VH).

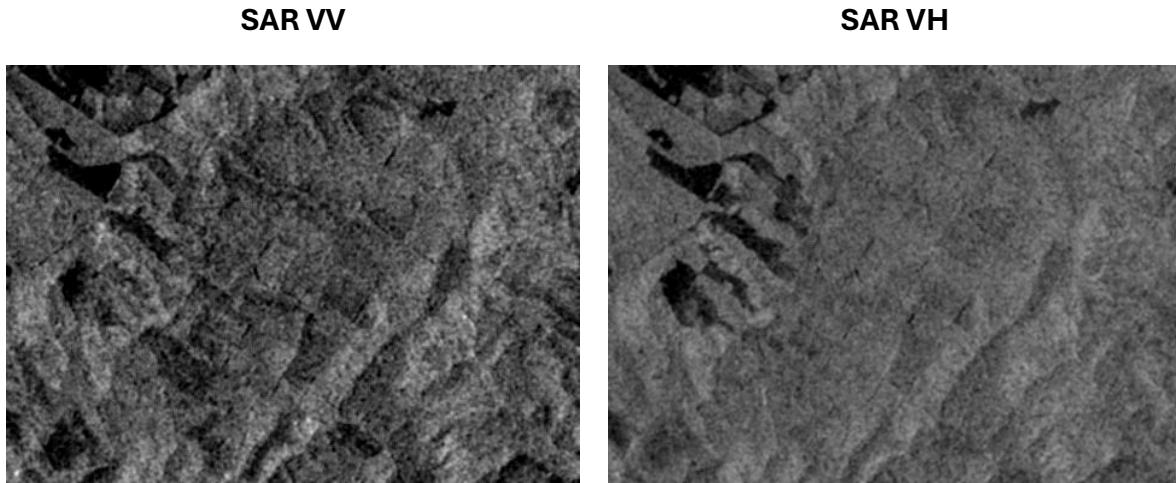


Figura 11 - Resultados das polarizações (VV e VH) para a área onde será realizada a classificação.

## 6.7. Pré-processamento com filtros espaciais e texturas

Tanto para os dados ópticos quanto para os de SAR, foram aplicados filtros espaciais com o objetivo de reduzir o ruído e realçar padrões locais. No caso das imagens SAR, utilizou-se o filtro de média focal para suavizar o *speckle* característico de sensores de SAR. Além disso, foram calculadas métricas de textura (GLCM) e estatísticas locais (máximo, mínimo, média, desvio-padrão) (IRONS; PETERSEN, 1981; HARALICK et al., 1973), que permitiram capturar variações estruturais e heterogeneidade espacial da cobertura.

## 6.8. Classificação supervisionada com *Random Forest*

Com os dados processados, foram utilizadas amostras de uso e cobertura do solo previamente definidas para treinar modelos de classificação Figura 12. Testaram-se três abordagens distintas: (i) apenas dados SAR, (ii) apenas dados ópticos e (iii) combinação óptico + SAR. Para cada cenário, os dados foram divididos em conjuntos de treino e teste, e aplicou-se o classificador *Random Forest*. O resultado dos três cenários testados para a classificação está apresentado na Figura 13. Em seguida, foram gerados mapas temáticos e avaliadas as matrizes de confusão e acurácia correspondentes.

Por fim, os mapas resultantes foram preparados para exportação em formato GeoTIFF, possibilitando análises posteriores em outros softwares e a comparação direta entre os métodos empregados.

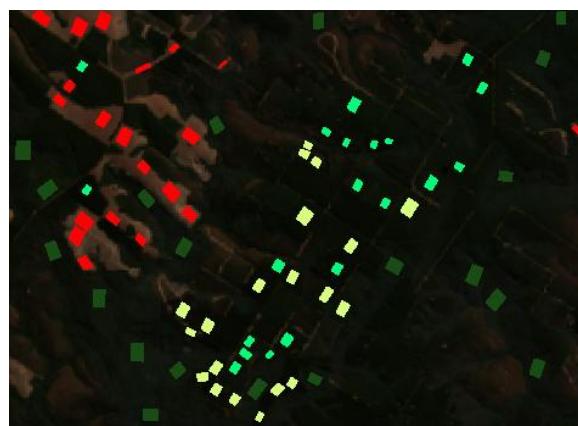
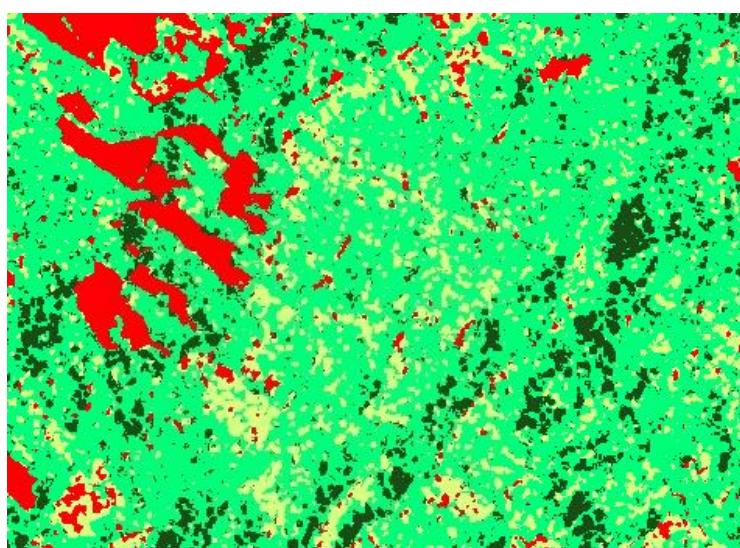
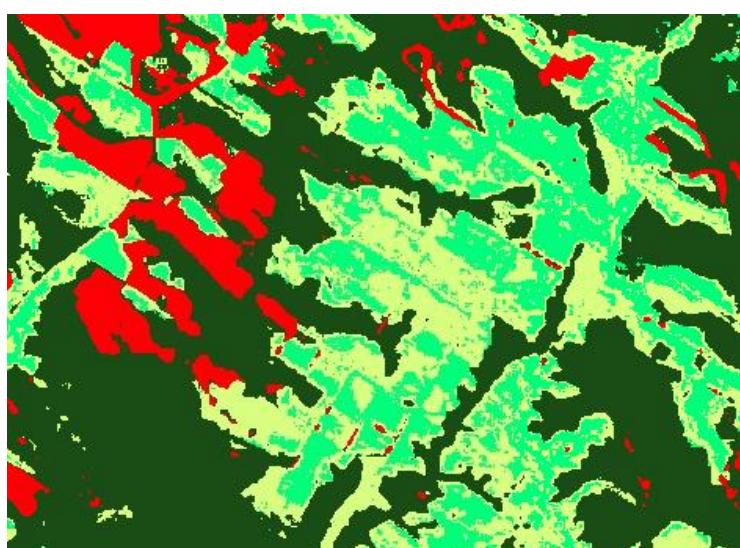


Figura 12 - Amostras coletadas para uso e cobertura do solo.

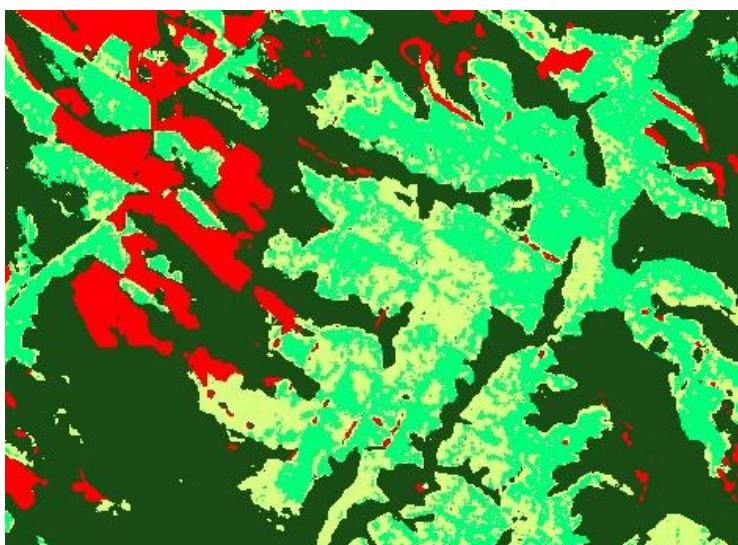


(i) apenas dados SAR



(ii) apenas dados ópticos

...continua



(iii) combinação óptico + SAR

Figura 13 - Resultados do classificador *Random Forest* para os três cenários.

## 6.9. Construção da Matriz de Confusão

Inicialmente, foi realizada a divisão das amostras coletadas em duas partes: conjunto de treinamento e conjunto de teste. O conjunto de treinamento foi utilizado para ajustar o modelo de classificação *Random Forest*, enquanto o conjunto de teste permaneceu reservado para avaliar o desempenho do classificador. Essa divisão foi feita de forma aleatória, garantindo que as amostras fossem representativas das diferentes classes de uso e cobertura do solo. Com o classificador ajustado, aplicou-se o modelo ao conjunto de teste. Esse procedimento teve como finalidade verificar a capacidade do algoritmo em generalizar padrões aprendidos no treinamento, avaliando se o modelo seria capaz de classificar corretamente amostras que não haviam sido utilizadas no ajuste. A seguir, foi elaborada a matriz de confusão, uma tabela que compara os valores previstos pelo modelo com os valores reais das amostras (Tabela 2). Essa matriz permitiu visualizar os acertos e os erros de classificação, mostrando quais classes apresentaram maior confusão entre si.

A partir da matriz de confusão, calculou-se a acurácia global, definida como a proporção de amostras corretamente classificadas em relação ao total de amostras avaliadas. Esse indicador forneceu uma visão geral do desempenho do classificador em todo o conjunto de dados. Além da acurácia global, foi calculado o coeficiente Kappa, que mede a concordância entre a classificação e os dados de referência considerando

a possibilidade de acertos ao acaso. Valores mais próximos de 1 indicaram forte concordância, enquanto valores próximos de 0 sugeriram que a classificação não foi melhor que uma atribuição aleatória.

Tabela 2 - Síntese dos valores do Kappa e Acurácia global

Modelo	Kappa	OA	
Óptico	0.96	0.972	
Óptico+SAR	0.959	0.971	
SAR	0.666	0.768	
..			

## **7. CLASSIFICAÇÃO DE USO DO SOLO COM SATELLITE EMBEDDING DATASET**

Nesta aula, iniciou-se com a importação da área de interesse (ROI) e, em seguida, carregou-se dados do Google *Satellite Embedding Dataset* anual (BROWN et al., 2025), definindo-se o período de 2024 e filtrando-se as imagens pela ROI para gerar um mosaico. Visualizou-se o espaço de *embeddings* por meio de uma composição RGB (bandas A01, A16 e A09) e preparou-se um conjunto de amostras para treinar um agrupador *k-means*, a partir do qual foram gerados mapas temáticos com 3, 5 e 10 classes. Posteriormente, calcularam-se as áreas por classe e exportaram-se os resultados em CSV e o *raster* clusterizado em GeoTIFF. Opcionalmente, estimou-se a mudança temporal entre 2023 e 2024 por similaridade do cosseno entre vetores de *embeddings*, com possibilidade de exportar a superfície de mudança. Como utilizou-se a coleção *Satellite Embedding V1* optou-se por explicar como esta coleção foi desenvolvida.

O *AlphaEarth Foundations* do Google é um modelo de incorporação geoespacial treinado em vários conjuntos de dados de observação da Terra (EO, na sigla em inglês). O modelo foi executado em séries temporais anuais de imagens, e os *embeddings* resultantes estão disponíveis como um conjunto de dados pronto para análise no *Earth Engine*. Com esse conjunto de dados, os usuários podem criar qualquer número de aplicativos de ajuste refinado ou outras tarefas sem executar modelos de aprendizado profundo computacionalmente caros (BROWN et al., 2025). Os *embeddings* são uma forma de compactar grandes quantidades de informações em um conjunto menor de recursos que representam semânticas significativas. O modelo *AlphaEarth Foundations* usa séries temporais de imagens de sensores, incluindo Sentinel-2, Sentinel-1 e Landsat, e aprende a representar de maneira exclusiva as informações mútuas entre fontes e destinos com apenas 64 números. O fluxo de dados de entrada contém milhares de bandas de imagens de vários sensores, e o modelo transforma essa entrada de alta dimensão em uma representação de baixa dimensão. Para mais detalhes, ver BROWN et al. (2025).

Conforme BROWN et al. (2025), um bom modelo mental para entender como o *AlphaEarth Foundations* funciona é uma técnica chamada análise de componentes

principais (PCA, na sigla em inglês). A PCA também ajuda a reduzir a dimensionalidade dos dados para aplicativos de aprendizado de máquina. Embora a PCA seja uma técnica estatística e possa compactar dezenas de bandas de entrada em alguns componentes principais, o *AlphaEarth Foundations* é um modelo de aprendizado profundo que pode usar milhares de dimensões de entrada de conjuntos de dados de séries temporais multissensoriais e aprender a criar uma representação de 64 bandas que captura de maneira exclusiva a variabilidade espacial e temporal desse pixel.

Os *embeddings* são uma forma de compactar grandes quantidades de informações em um conjunto menor de recursos que representam semânticas significativas. O modelo *AlphaEarth Foundations* usa séries temporais de imagens de sensores, incluindo Sentinel-2, Sentinel-1 e Landsat, e aprende a representar de maneira exclusiva as informações mútuas entre fontes e destinos com apenas 64 números. O fluxo de dados de entrada contém milhares de bandas de imagens de vários sensores, e o modelo transforma essa entrada de alta dimensão em uma representação de baixa dimensão (BROWN et al., 2025).

Um bom modelo mental para entender como o *AlphaEarth Foundations* funciona é uma técnica chamada análise de componentes principais (PCA, na sigla em inglês). A PCA também ajuda a reduzir a dimensionalidade dos dados para aplicativos de aprendizado de máquina. Embora a PCA seja uma técnica estatística e possa compactar dezenas de bandas de entrada em alguns componentes principais, o *AlphaEarth Foundations* é um modelo de aprendizado profundo que pode usar milhares de dimensões de entrada de conjuntos de dados de séries temporais multissensoriais e aprender a criar uma representação de 64 bandas que captura de maneira exclusiva a variabilidade espacial e temporal desse pixel (BROWN et al., 2025).

Um campo de *embedding* é a matriz contínua ou o "campo" de *embeddings* aprendidos. As imagens nas coleções de campos de *embedding* representam trajetórias espaço-temporais que abrangem um ano inteiro e têm 64 bandas (uma para cada dimensão de *embedding*) (BROWN et al., 2025).

O conjunto de dados de incorporação de satélite é uma coleção de imagens com imagens anuais de 2017 em diante (por exemplo, 2017, 2018, 2019, entre outros). Cada imagem tem 64 bandas em que cada pixel é o vetor de incorporação que representa a série temporal multissensorial do ano em questão (BROWN et al., 2025).

## **7.1. Definição da Área de Interesse (ROI)**

O primeiro passo foi delimitar a região de interesse (ROI – *Region of Interest*). Será utilizada a mesma área de interesse do exemplo anterior. Essa área espacial é o recorte no qual serão realizadas todas as análises posteriores. Em seguida, são definidos os parâmetros temporais, estabelecendo intervalos de datas para o ano (2024) e para o ano (2023). Essa abordagem foi essencial, pois as análises que serão realizadas irão comparar períodos distintos para identificar mudanças.

## **7.2. Preparação dos *embeddings* de satélite**

A base de dados utilizada para classificação é o Google *Satellite Embedding V1/Annual*, um conjunto de representações vetoriais (*embeddings*) derivadas de imagens de satélite (Figura 14). Esses *embeddings* condensam a informação espectral em bandas de alta dimensionalidade que servem como preditores de uso e cobertura da terra. O *script* filtra as imagens pelo período de interesse e pela ROI, criando composições anuais.

## **7.3. Seleção de bandas preditoras e visualização**

Após gerar os mosaicos anuais de *embeddings*, o *script* lista as bandas disponíveis e cria uma visualização em falso colorido (RGB - Figura 14a). Esse passo tem caráter exploratório, ajudando a verificar se os *embeddings* estão corretamente carregados e se existe coerência visual na região analisada.

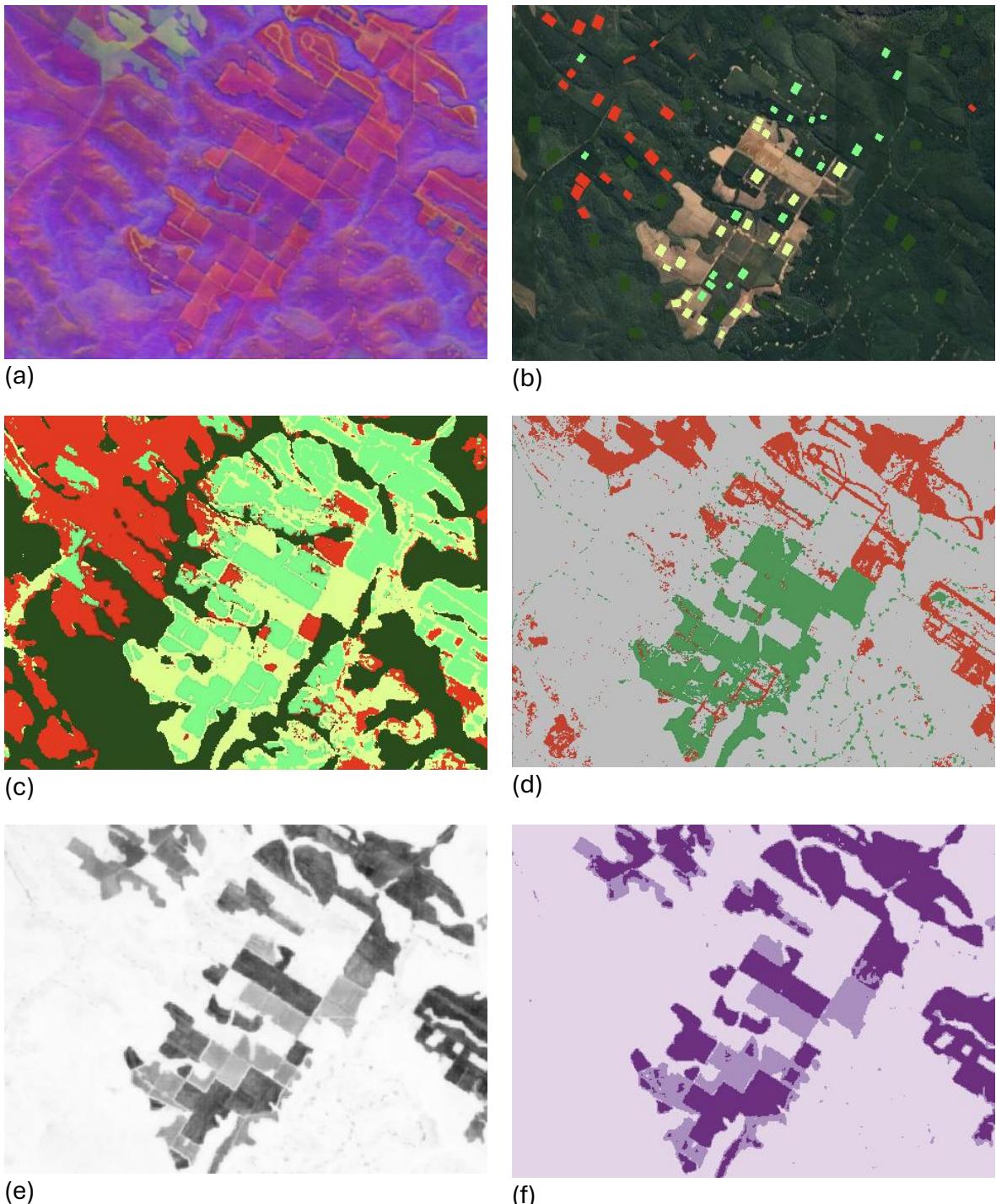


Figura 14 - Síntese do processamento com os dados do *Satellite Embedding Dataset*. (a) Embedding RGB; (b) Amostras; (c) Classificação; (d) Mudanças pelo NDVI (2023-2024); (e) Mudanças pela similaridade do cosseno entre embeddings; (f) Síntese das mudanças pela similaridade do cosseno.

#### 7.4. Definição das amostras de treinamento

A etapa seguinte utiliza uma *FeatureCollection* de amostras de uso do solo previamente delineadas pelo usuário. Essas amostras trazem a informação de

referência (*ground truth*), vinculada à propriedade "usodosolo". Elas são fundamentais para treinar o algoritmo de classificação supervisionada, fornecendo a ligação entre os atributos espectrais (*embeddings*) e as classes de interesse (floresta, agricultura, água e urbano).

## 7.5. Preparação do conjunto de treinamento e validação

O script extrai, para cada ponto de amostra, os valores das bandas dos *embeddings*. Esses registros compõem o conjunto de treinamento. Posteriormente, aplica-se uma divisão aleatória estratificada (70% treino, 30% validação), assegurando que o modelo seja treinado em parte dos dados e validado em dados independentes, evitando sobreajuste (*overfitting*).

## 7.6. Treinamento do classificador *Random Forest*

É aplicado o algoritmo *Random Forest* (RF), configurado com 300 árvores de decisão. O RF é um ensemble que combina múltiplas árvores para gerar classificações robustas e reduzir a variância. Ele é bastante utilizado em sensoriamento remoto por sua capacidade de lidar com dados ruidosos e variáveis altamente correlacionadas, como é o caso dos *embeddings*.

## 7.7. Validação por *hold-out* e métricas de acurácia

O conjunto de validação passa pelo modelo treinado, gerando previsões que são comparadas com as classes reais. Calcula-se a matriz de confusão, acurácia global, coeficiente de Kappa e as acurácias do produtor e do usuário para cada classe. Esses indicadores são fundamentais para aferir a confiabilidade do classificador, mostrando onde ocorrem acertos e erros.

## 7.8. Classificação final da área de estudo

Com o modelo validado, aplica-se o RF sobre todos os pixels do mosaico de *embeddings* de 2024, gerando o mapa final de classes de uso e cobertura da terra. O

resultado é visualizado em cores distintas para cada categoria, permitindo interpretação espacial e análise quantitativa da distribuição das classes.

### **7.9. Estimativa de áreas por classe**

Para transformar o mapa classificado em estatísticas, o *script* multiplica os pixels por sua área (em hectares) e agrupa os valores por classe. Isso gera tabelas com a área ocupada por cada uso da terra, permitindo análises comparativas e suporte à tomada de decisão em gestão territorial.

### **7.10. Integração com dados do Sentinel-2**

Além da classificação por *embeddings*, o *script* utiliza imagens Sentinel-2 SR harmonizadas para calcular o NDVI (Índice de Vegetação por Diferença Normalizada) em dois anos consecutivos. Esse índice é amplamente utilizado como *proxy* da biomassa verde e da atividade fotossintética, refletindo mudanças na cobertura vegetal.

### **7.11. Análise de mudança temporal ( $\Delta$ NDVI)**

A diferença entre o NDVI de 2024 e 2023 ( $\Delta$ NDVI) é calculada pixel a pixel. Um limiar (0,05) é aplicado para separar áreas que diminuíram, permaneceram estáveis ou aumentaram em vigor vegetativo. Esse método de detecção de mudança é direto e robusto, permitindo identificar processos de desmatamento, degradação ou regeneração.

### **7.12. Cálculo de áreas por classes de mudança**

Assim como na classificação, as áreas correspondentes a cada categoria de mudança (diminuição, estabilidade ou aumento do NDVI) são computadas em hectares. Isso gera uma visão quantitativa da dinâmica da vegetação no período analisado, indo além da simples classificação estática.

## 7.13. Similaridade do cosseno entre embeddings

Um procedimento adicional calcula a similaridade do cosseno entre os embeddings de 2023 e 2024. Essa métrica quantifica a intensidade da mudança multiespectral, distinguindo áreas de alta mudança estrutural daquelas relativamente estáveis. Os limiares  $T1=0,70$  e  $T2=0,90$  categorizam a intensidade em três níveis, refinando a interpretação da dinâmica ambiental.

## 7.14. Tabelas síntese e exportação de resultados

Todas as estatísticas geradas — áreas por classe, áreas por mudança e métricas de validação — são organizadas em tabelas síntese (Tabela 3). Essas tabelas podem ser exportadas para arquivos CSV, enquanto os mapas (classificação e mudanças) são exportados como *rasters* para o *Google Drive*. Isso garante que os resultados possam ser analisados posteriormente em outros softwares. Por fim, o *script* gera uma tabela formatada com ui.Chart no console do GEE, permitindo ao usuário visualizar de forma interativa as áreas calculadas. Essa etapa facilita a interpretação dos resultados, oferecendo uma visão consolidada tanto em mapas quanto em tabelas.

Tabela 3 - Síntese das áreas classificadas e das mudanças de uso entre 2023-2024.

rotulo	area_ha
Floresta	1,074.812
Agricultura	1,075.135
Água	1,067.889
Urbano	1,733.826
Diminuiu ( $\Delta NDVI < -0.05$ )	695.66
Não mudou ( $ \Delta NDVI  \leq 0.05$ )	3,604.398
Aumentou ( $\Delta NDVI > 0.05$ )	651.604

## **8. ANÁLISE DE SÉRIES TEMPORAIS DOS ÍNDICES DE VEGETAÇÃO**

Nesta atividade, iniciou-se pela definição da região de interesse (ROI) e pela configuração dos parâmetros temporais e da escala de análise; em seguida, carregaram-se as imagens do Landsat 8 L2/C2 (com opção de integração do Landsat 9), aplicou-se as máscaras de qualidade (QA\_PIXEL e QA\_RADSAT) e realizaram-se os devidos escalonamentos radiométricos das bandas ópticas e térmicas. A partir das bandas NIR, RED e BLUE, calcularam-se os índices de vegetação NDVI, EVI e SAVI ( $L = 0,5$ ) e produziram-se composições medianas para inspeção visual da área. Posteriormente, agregaram-se os índices mensalmente por pixel, registrando-se a contagem de cenas por mês e filtrando-se os meses com amostra insuficiente. Por fim, executaram-se reduções espaciais sobre a ROI para obter as séries temporais mensais (médias na área) e apresentaram-se gráficos individuais para cada índice, com opção de gráfico da contagem mensal de imagens para diagnóstico da robustez da série.

### **8.1. Escopo, parâmetros e objetivo**

Estabeleceu-se, de início, o objetivo de derivar séries mensais dos índices espectrais NDVI, EVI e SAVI para uma área de interesse (ROI), tomando como base o Landsat 8 Nível-2 Coleção 2 (com opção de complementar com Landsat 9). Definiram-se os parâmetros temporais (START, END), a escala espacial de processamento (SCALE = 30 m) e um limiar mínimo de imagens por mês (MIN\_IMAGES\_PER\_MONTH) para garantir robustez estatística na composição mensal e reduzir a influência de observações atípicas.

### **8.2. Qualidade dos dados e mascaramento atmosférico-geométrico**

Empregou-se a banda de qualidade QA\_PIXEL para mascarar *fill*, nuvem dilatada, *cirrus*, nuvens, sombreamento e neve, zerando-se todos esses bits indesejáveis. Em paralelo, utilizou-se QA\_RADSAT para excluir pixels com saturação radiométrica. Com isso, restringiram-se os cálculos subsequentes a observações mais confiáveis, mitigando contaminações atmosféricas e artefatos que poderiam distorcer os índices de vegetação.

### **8.3. Escalonamento radiométrico das bandas**

Aplicaram-se os fatores de escala e offset oficiais do produto L2/C2 às bandas ópticas SR\_B\* e às bandas térmicas ST\_B\*. Para as bandas de reflectância de superfície, multiplicou-se pelos coeficientes de escala e somou-se o offset, de modo a trazer os valores para reflectância adimensional. Para as bandas de temperatura de superfície, aplicou-se o fator correspondente para obter temperatura física (não utilizada nos índices, mas mantida de forma coerente no cubo de bandas). Esse passo padronizou as unidades e garantiu consistência física nas operações espectrais.

### **8.4. Seleção de bandas e construção dos índices**

Deram-se nomes explícitos às bandas necessárias: NIR = SR\_B5, RED = SR\_B4 e BLUE = SR\_B2. Em seguida, calcularam-se três índices: NDVI sensível à atividade fotossintética; EVI, que atenuou efeitos atmosféricos e de solo; e SAVI com  $L = 0,5$ , que compensou a influência do brilho do solo. Todos foram anexados como novas bandas ao mosaico mascarado e escalado.

### **8.5. Definição da ROI e configuração do mapa**

Definiu-se a ROI como uma *FeatureCollection* já existente no ambiente, extraindo-se sua geometria para operações espaciais e para os gráficos. Centralizou-se o mapa sobre a ROI para facilitar a inspeção visual e a depuração. Essa padronização assegurou que todos os cálculos de redução espacial fossem realizados estritamente dentro dos limites da área de estudo.

### **8.6. Montagem e filtragem da coleção Landsat 8**

Construiu-se a *ImageCollection* do Landsat 8 L2/C2/T1 filtrando-se por limites espaciais (ROI) e intervalo temporal (START-END). Mapeou-se a função de pré-processamento sobre cada imagem para devolver uma coleção homogênea, já mascarada, escalada e com índices calculados. Esse encadeamento garantiu que cada cena contribuísse de forma comparável às composições mensais.

## **8.7. Integração opcional com Landsat 9**

Previu-se a integração opcional com Landsat 9 L2/C2/T1, replicando o mesmo pré-processamento e mesclando as coleções. Quando ativado, esse passo aumentou a densidade temporal após 2021, o que elevou a chance de atingir o número mínimo de imagens/mês e reduziu a sensibilidade a datas pontuais problemáticas.

## **8.8. Produtos medianos para inspeção visual**

Geraram-se imagens medianas de NDVI, EVI e SAVI ao longo de todo o período, recortadas à ROI e exibidas no mapa com faixas [min, max] adequadas e paleta simples. Essa inspeção qualitativa serviu para verificar padrões espaciais estáveis, checar gradientes de vegetação e identificar rapidamente artefatos residuais que porventura escaparam ao mascaramento.

## **8.9. Agregação mensal com metadados e controle de amostra**

Estruturou-se uma função de agregação mensal que percorreu todos os meses do intervalo, filtrou as imagens de cada mês e computou a média por pixel do índice solicitado. Para cada imagem mensal, registraram-se metadados de tempo (system:time\_start no primeiro dia do mês), ano, mês e contagem de imagens utilizadas. Posteriormente, filtrou-se os meses com contagem inferior ao limiar MIN\_IMAGES\_PER\_MONTH, evitando que meses suportados por uma única cena (como o de 11/08/2021) distorcessem a série temporal.

## **8.10. Reduções espaciais e gráficos mensais por índice**

Para cada coleção mensal (NDVI, EVI e SAVI), aplicou-se um reduzir espacial (Reducer.mean) sobre a geometria da ROI à escala de 30 m, obtendo-se assim a média mensal do índice na área. Os resultados foram apresentados em gráficos individuais do tipo *LineChart*, com janela de visualização coerente (0–1), cores distintas e interpolação de nulos para manter a continuidade visual nos meses filtrados. Adicionalmente, previu-se um gráfico de contagem de imagens por mês para diagnóstico, permitindo avaliar a

robustez temporal da série e orientar eventuais ajustes de parâmetros (como ativar o Landsat 9 ou elevar o mínimo de imagens por mês – Vide Figura 15).

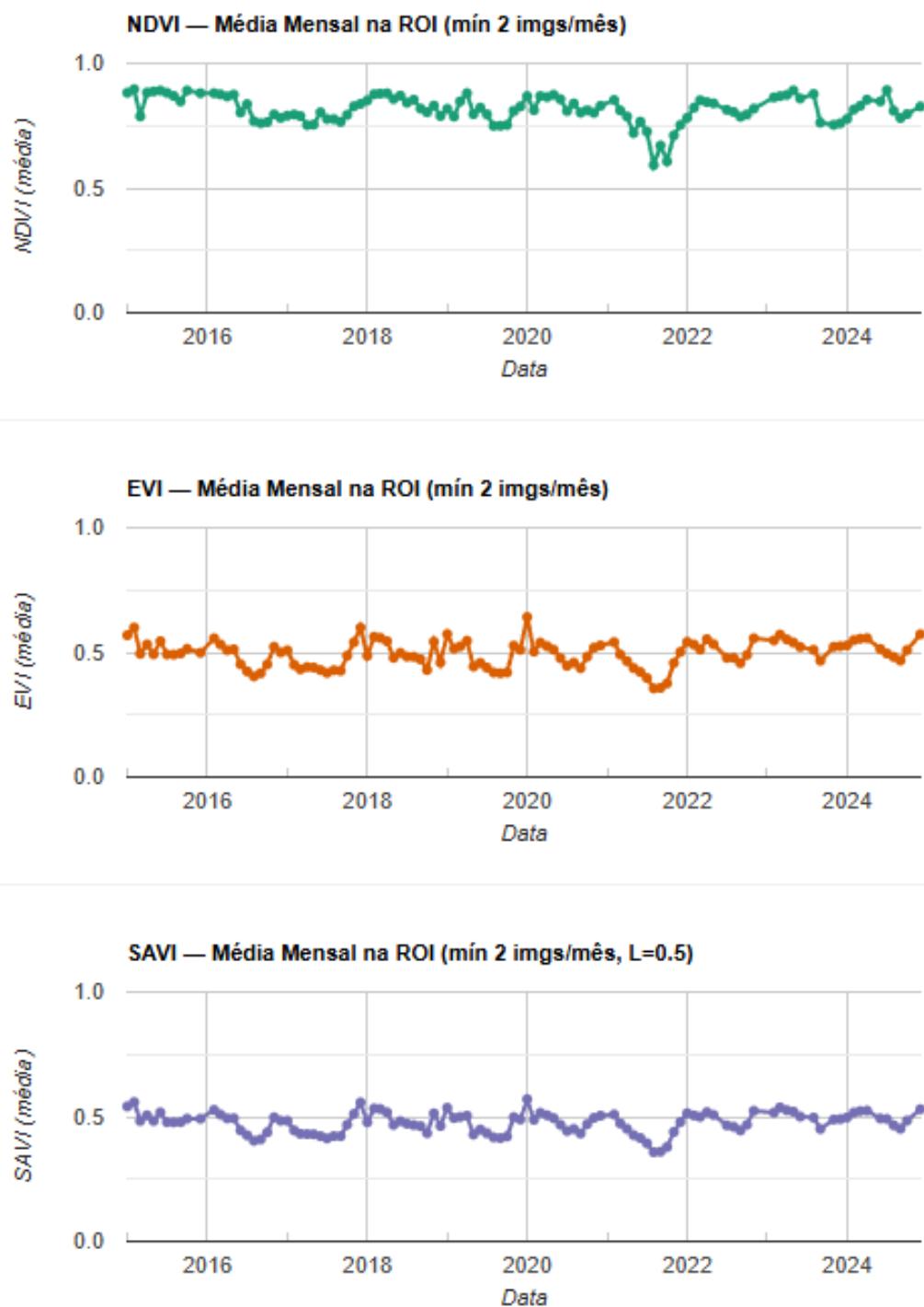


Figura 15 – Médias mensais dos índices de vegetação.

## **9. AVALIAÇÃO DA ÁREA COM FOCO DE INCÊNDIO E ANÁLISE DA PRECIPITAÇÃO**

Nesta atividade, definiu-se a área de interesse (AOI) e, em seguida, analisou-se a dinâmica de queimadas e precipitação entre 2015 e 2022. A coleção MODIS MCD64A1 (JUSTICE et al., 2002) foi filtrada e a banda *BurnDate* utilizada para mascarar pixels queimados, estimando-se a área queimada ( $\text{km}^2$ ) por imagem e construindo-se a série temporal correspondente. Paralelamente, agregou-se a precipitação total mensal, organizando-se uma coleção mensal e o respectivo gráfico, com exemplos de somatórios sazonais para a estação chuvosa. Selecionou-se o mês de setembro de 2019 onde mapeou-se especificamente a área queimada, quantificou-se sua extensão na AOI e calculou-se a precipitação média do mês. Complementarmente, obtiveram-se imagens Sentinel-2 do início de setembro de 2019; aplicou-se a máscara de nuvens/cirrus (QA60), escalou-se para reflectância e recortou-se à AOI, selecionando-se a melhor cena por menor nebulosidade e gerando-se também uma composição mediana para visualização (RGB e NIR). Por fim, adicionaram-se as camadas temáticas ao mapa e registraram-se informações de sanidade (número de imagens, ID e porcentagem de nuvens), integrando gráficos e métricas para relacionar áreas queimadas e precipitação.

### **9.1. Definição da área de interesse (AOI)**

Estabeleceu-se, inicialmente, o polígono que delimita a área de estudo (AOI), sobre o qual todas as operações subsequentes foram executadas. Neste exemplo, selecionou-se uma área no Brasil que sabidamente é alvo de focos de incêndios para demonstração do potencial da análise. Os limites das coordenadas encontram-se no código GEE. Procedeu-se à centralização do mapa e à exibição do contorno da AOI para garantir a referência espacial e a inspeção visual prévia, assegurando que os recortes e estatísticas fossem restritos ao limite definido.

### **9.2. Aquisição e filtragem do MODIS MCD64A1 (área queimada)**

Carregou-se a coleção MODIS MCD64A1 referente a áreas queimadas, restringindo-se o período de análise de 2015 a 2022. Essa filtragem temporal permitiu isolar eventos

de fogo ocorridos no intervalo de interesse, garantindo coerência cronológica para a comparação com as métricas de precipitação.

### **9.3. Seleção da variável *BurnDate* e lógica de detecção**

Selecionou-se a banda *BurnDate*, que registrou a ocorrência de queima dentro do mês de referência de cada composição. Adotou-se a regra segundo a qual valores positivos indicaram pixels queimados, o que viabilizou a criação de máscaras para separar áreas afetadas pelo fogo de áreas não queimadas, fornecendo base para quantificação de extensão.

### **9.4. Estimativa de área queimada por imagem ( $\text{km}^2$ )**

Aplicou-se a imagem de área de pixel para converter superfícies de  $\text{m}^2$  para  $\text{km}^2$ , restringindo os cálculos aos pixels marcados como queimados e recortados à AOI. Em seguida, realizou-se a agregação espacial por meio de redução regional, produzindo, para cada data, a estimativa da área queimada total, adequando-se a escala ao produto (500 m) para evitar vieses de amostragem.

### **9.5. Construção da série temporal e visualização de área queimada**

Organizou-se a coleção resultante contendo, para cada data, a área queimada estimada, preservando-se o carimbo temporal do sistema. Construiu-se, então, um gráfico de série temporal para inspecionar tendências (Figura 16a), sazonalidade e variabilidade interanual das queimadas na AOI, favorecendo a interpretação de padrões de ocorrência e intensidade do fogo.

### **9.6. Aquisição do CHIRPS em pentadas e agregação mensal**

A coleção CHIRPS *Pentad* refere-se aos dados de precipitação quase globais e com resolução de 0,05° fornecidos pelo conjunto de dados CHIRPS (*Climate Hazards Center InfraRed Precipitation with Station data*) em intervalos de cinco dias (pentadas). Esta base combina dados de satélite (infravermelho) com dados de estações meteorológicas *in situ*. Para tanto, carregou-se a coleção CHIRPS, restringindo-a ao mesmo intervalo

temporal da análise de queimadas e à AOI. Em seguida, agregaram-se as pentadas por soma mensal, formando uma coleção de totais mensais de precipitação, etapa essencial para harmonizar a granularidade temporal e permitir a comparação com as métricas mensais de fogo.

## 9.7. Série temporal de precipitação e inspeção gráfica

Derivou-se a série temporal mensal de precipitação média na AOI por meio de redução espacial apropriada, mantendo-se consistência com a escala do insumo. Em sequência, elaborou-se um gráfico de precipitação mensal (Figura 16b) para avaliar a distribuição anual e interanual das chuvas, identificando-se períodos chuvosos e secos e potenciais deslocamentos sazonais.

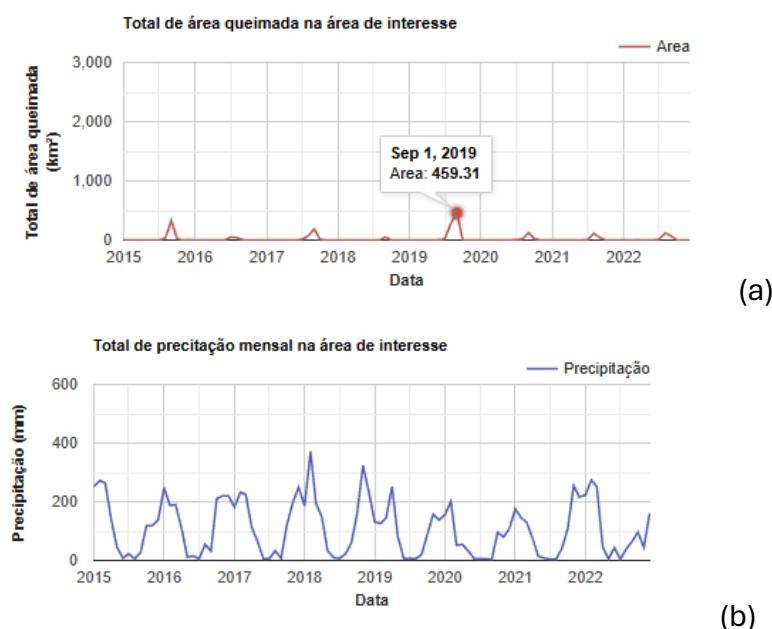


Figura 15 - Síntese das áreas queimadas e da precipitação por mês e ano.

## 9.8. Totais sazonais da estação chuvosa (exemplos)

Computaram-se exemplos de totais de precipitação em janelas sazonais que abrangeram a estação chuvosa, estabelecendo-se datas de início e término que cruzaram dois anos civis. Essa agregação ofereceu uma métrica robusta para comparar diferentes estações, permitindo inferências sobre acúmulo hídrico e possíveis relações com a extensão de queimadas.

## **9.9. Análise pontual de setembro de 2019: fogo e chuva**

Isolou-se o mês de setembro de 2019 para quantificação específica da área queimada, aplicando-se a máscara de pixels com *BurnDate* positivo e somando-se a área correspondente em km<sup>2</sup>. Em paralelo, calculou-se a precipitação média no mesmo mês, viabilizando um confronto direto entre a ocorrência de queimadas e o regime de chuvas naquele período crítico.

## **9.10. Pré-processamento e visualização Sentinel-2 (início de setembro/2019)**

Selecionaram-se imagens Sentinel-2 do início de setembro de 2019 (Figura 17), aplicando-se máscara de nuvens e *cirrus* (qualidade/QA), normalização para reflectância de superfície e recorte à AOI. Procedeu-se tanto à escolha da melhor cena (menor nebulosidade) quanto à composição mediana do período, gerando visualizações RGB e em infravermelho próximo (NIR) para contextualizar, com maior resolução espacial, as áreas queimadas e o estado da vegetação no intervalo analisado.

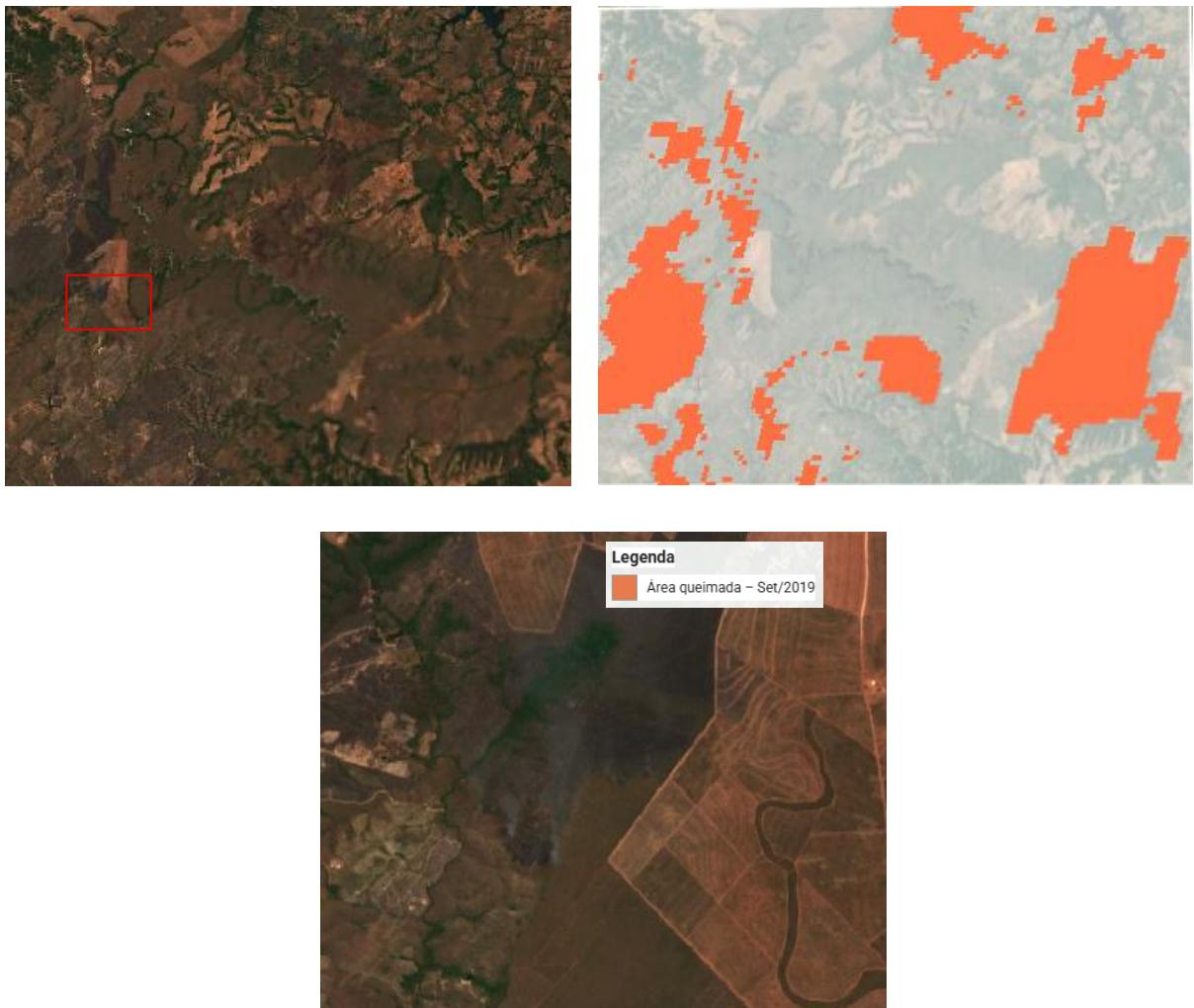


Figura 16 - Destaque para a imagem do início do mês de setembro de 2019 e áreas com foco de incêndio no interior da ROI.

Por fim, calculou-se a área queimada da série (2015-2022) avaliada acumulada por ano, conforme demonstra a Figura 18.

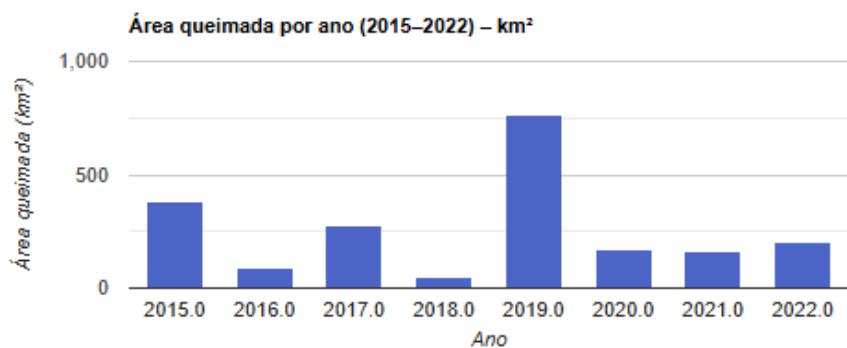


Figura 17 - Total da área queimada por ano da região avaliada.

## **10. DETECÇÃO DE MUDANÇAS COM SAR**

Nesta atividade, definiu-se a área de interesse (AOI) e realizou-se uma análise multitemporal de retroespalhamento do SAR Sentinel-1 (polarizações VV e VH) para os meses de agosto de 2019, 2020 e 2021. Foram extraídas imagens no modo IW, órbita descendente, com resolução espacial de 10 m, compondo-se mosaicos medianos mensais e aplicando-se suavização espacial para redução do *speckle*. A partir dessas imagens, construíram-se composições RGB multitemporais, que permitem identificar visualmente mudanças na estrutura da vegetação. Em seguida, calcularam-se as diferenças de retroespalhamento ( $\Delta$ VH) entre os intervalos 2019 $\rightarrow$ 2020 e 2020 $\rightarrow$ 2021, extraindo estatísticas de média e desvio-padrão para estabelecer limiares dinâmicos de detecção de perda. As áreas classificadas como perdas foram quantificadas em hectares, tanto por intervalo quanto no total acumulado (2019 $\rightarrow$ 2021), e os resultados organizados em tabela-síntese, permitindo relacionar padrões espaciais de alteração da cobertura com valores numéricos de superfície impactada.

### **10.1. Definição da área de interesse (ROI)**

Iniciou-se com a delimitação da região de estudo. Essa geometria é a referência espacial para todas as operações posteriores. Ao centralizar o mapa na ROI, garante-se que os resultados sejam visualizados apenas dentro dessa área específica, evitando análises fora do escopo definido.

### **10.2. Parâmetros temporais e espaciais**

Em seguida, define-se a lista de anos (2019, 2020, 2021), o mês de análise (agosto) e o raio de suavização espacial (50 metros). Esses parâmetros controlam tanto a janela temporal quanto o nível de suavização aplicado nas imagens, determinando a consistência entre anos e a redução de ruídos do SAR.

### **10.3. Carregamento das coleções Sentinel-1**

A função `s1Collection` busca imagens da coleção COPERNICUS/S1\GRD, filtrando apenas as que foram adquiridas no modo IW (*Interferometric Wide Swath*), com

resolução de 10 m, em órbita descendente, e contendo as polarizações VV ou VH. Essa filtragem garante comparabilidade entre os anos, evitando incluir cenas de geometrias de aquisição distintas.

#### 10.4. Polarizações VV e VH

O Sentinel-1 fornece duas polarizações principais: VV (onda transmitida e recebida na vertical) e VH (onda transmitida vertical, recebida horizontal). A polarização VV é mais sensível a superfícies rugosas e estruturas lineares, enquanto VH responde fortemente à presença de biomassa e heterogeneidade da vegetação, sendo mais usada em estudos de desmatamento e mudanças estruturais. Em seguida, fez-se a mediana mensal de agosto para cada ano. Essa abordagem minimiza efeitos de *outliers*, como interferências atmosféricas ou ruído de aquisição, criando um mosaico representativo de cada período. Assim, obtém-se três imagens anuais (2019, 2020, 2021) para cada polarização.

Para facilitar a detecção visual de mudanças, as imagens de 2019, 2020 e 2021 (VH) foram combinadas em uma composição colorida (R=2019, G=2020, B=2021) conforme representa a Figura 19. Essa técnica permite observar diferenças como deslocamentos de cores, indicando onde houve ganho ou perda de sinal de retroespalhamento entre anos.

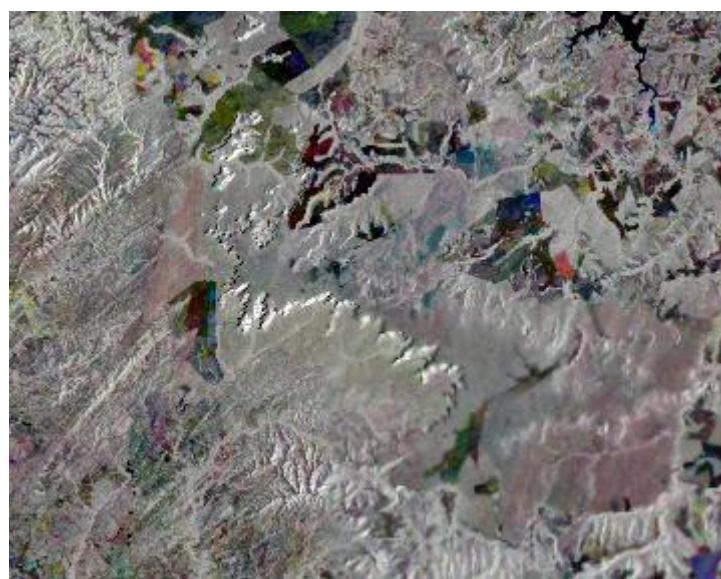


Figura 18 - Exemplo de composição colorida (R=2019, G=2020, B=2021)

## 10.5. Suavização do *speckle*

O SAR apresenta um ruído granular chamado \**speckle*\*, que dificulta análises quantitativas. Para reduzir esse efeito, aplica-se a média focal em um raio de 50 metros. Isso preserva tendências espaciais mais amplas, reduzindo a variabilidade *pixel a pixel*, o que aumenta a robustez das comparações temporais.

## 10.6. Diferenças temporais

A análise de mudanças é feita por subtração entre as imagens suavizadas. O script calcula  $\Delta VH$  de 2019 para 2020 e de 2020 para 2021. Valores positivos ou negativos dessas diferenças indicam, respectivamente, ganho ou perda no retroespalhamento, associados a alterações na cobertura do solo ou na vegetação.

Para caracterizar essas diferenças, o script extrai estatísticas de média e desvio-padrão dentro da ROI. Essas métricas são fundamentais para definir os limiares dinâmicos, evitando o uso de valores arbitrários que poderiam variar dependendo da região ou ano analisado.

Um limiar superior é calculado como média +  $1,5 \times$  desvio-padrão. Pixels cujo  $\Delta VH$  excede esse valor são considerados “mudanças significativas”, assumidas aqui como perdas (ex.: redução de biomassa, abertura do dossel ou desmatamento). Essa abordagem baseada em estatísticas locais se ajusta melhor às características de cada par de anos.

## 10.7. Criação de máscaras de perda

Com os limiares definidos, gerou-se máscaras binárias para identificar áreas que sofreram perda de cobertura. Essas máscaras são posteriormente exibidas no mapa em vermelho, destacando visualmente as regiões críticas detectadas pelo algoritmo.

Para quantificar os resultados, a função `areaHaFromMask` converte pixels classificados em perda em áreas (hectares), usando a banda de área de pixel do GEE. Esse cálculo permite interpretar os resultados em termos de métricas de superfície, tornando a análise aplicável para relatórios técnicos e tomada de decisão.

As perdas de 2019→2020 e de 2020→2021 foram unidas para calcular a perda acumulada no intervalo 2019→2021, conforme mostra a Figura 20. Essa etapa é importante para evitar dupla contagem de áreas e oferecer uma síntese clara da magnitude da perda em múltiplos anos (Tabela 4).

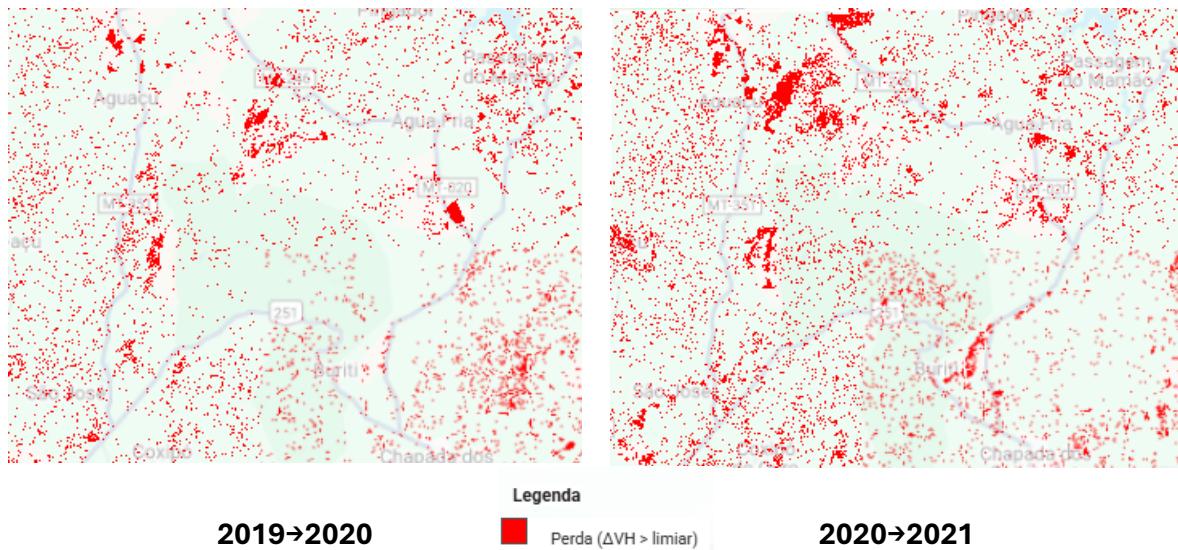


Figura 20 - Exemplo das perdas nos anos avaliados.

Os valores de perda por intervalo e o total acumulado são organizados em uma `FeatureCollection` exibida como tabela no console. Essa organização tabular fornece clareza na comparação entre períodos e facilita exportações para análises adicionais.

Por fim, o script adiciona uma legenda no mapa para interpretar corretamente as cores exibidas. A padronização da simbologia (vermelho = perda) garante consistência na comunicação dos resultados, seja para especialistas, seja para gestores ambientais. Esse cuidado fortalece a usabilidade do script como ferramenta prática de monitoramento.

Tabela 4 - Área de perda (hectares) para os períodos avaliados.

Intervalo	Área_ha
2019→2020	10,223.77
2020→2021	11,675.34
2019→2021	21,578.49

## 11. ANÁLISE DE APP POR DECLIVIDADE COM SRTM

Neste exercício, definiu-se a área de interesse (AOI) e utilizou-se o modelo digital de elevação SRTM para calcular as informações de declividade das áreas. Inicialmente, o SRTM bruto foi recortado à AOI e visualizado em escala hipsométrica. Em seguida, os dados foram reamostrados para 10 m e calculou-se a declividade do terreno, posteriormente classificada em cinco intervalos angulares (0–10°, 10–20°, 20–30°, 30–45° e acima de 45°). Para apoiar a interpretação visual, gerou-se também o *hillshade* do relevo e uma legenda colorida associada às classes de declividade. A partir dessa classificação, estimaram-se as áreas ocupadas por cada classe em hectares, organizando-se uma tabela-síntese no console que permite quantificar e comparar a distribuição espacial da declividade dentro da AOI, integrando análise visual e estatística do relevo. Cabe destacar que, conforme o Código Florestal Brasileiro (Lei nº 12.651/2012, art. 4º, inciso V), as áreas com declividade superior a 45° (equivalente a 100%) são consideradas Áreas de Preservação Permanente (APPs), devendo ser protegidas e mantidas com cobertura vegetal, independentemente da existência de cursos d’água. Assim, o mapeamento dessas classes de declividade não apenas subsidia análises geomorfológicas, mas também fornece suporte técnico e legal para a identificação de APPs e para o planejamento ambiental.

### 11.1. Delimitação da área de estudo (ROI)

Define-se explicitamente a região de interesse como um polígono em coordenadas geográficas (longitude/latitude). Essa ROI limita todo o processamento (cálculo de declividade, classes, somatórios de área, renderização e exportações), garantindo que estatísticas e visualizações sejam restritas ao recorte espacial desejado e evitando custos computacionais fora da área-alvo. Em seguida, fez-se a especificação de parâmetros globais: o CRS `EPSG:4674` (SIRGAS 2000, geográfico) e um teto de processamento `MAX\_PIXELS`. O CRS informado será usado na reprojeção da grade de trabalho; já `MAX\_PIXELS` protege operações agregadoras (ex.: `reduceRegion`) contra interrupção por exceder o limite padrão de pixels analisados.

## 11.2. Fonte altimétrica (SRTM)

O SRTM (*Shuttle SAR Topography Mission*) é um modelo digital de elevação (MDE) obtido a partir de uma missão conjunta da NASA e da NIMA (atual NGA – *National Geospatial-Intelligence Agency*), realizada em fevereiro do ano 2000 a bordo do ônibus espacial *Endeavour* (FARR et al., 2007). O mapeamento utilizou a técnica de interferometria de SAR de abertura sintética (InSAR) para gerar um modelo quase global da topografia terrestre, cobrindo aproximadamente 80% da superfície do planeta entre as latitudes 60°N e 56°S. O produto resultante fornece a altitude do terreno em relação ao nível médio do mar, com resoluções espaciais originais de 30 m (versão SRTMGL1) e 90 m (SRTMGL3), amplamente aplicadas em estudos ambientais, hidrológicos, geomorfológicos e de planejamento territorial. Por sua abrangência, consistência e disponibilidade gratuita, o SRTM tornou-se uma das bases altimétricas mais utilizadas no mundo para análise espacial e modelagem ambiental.

Portanto, neste exercício, carregou-se o modelo digital de elevação `USGS/SRTMGL1\_003`, com resolução nativa  $\sim 30$  m, elipsoide WGS84 e cobertura quase global (Figura 21). Esse MDE é a base para derivar variáveis geomorfológicas (declividade, orientação, etc.) via funções de terreno do *Earth Engine*, sendo suficientemente estável para análises regionais.

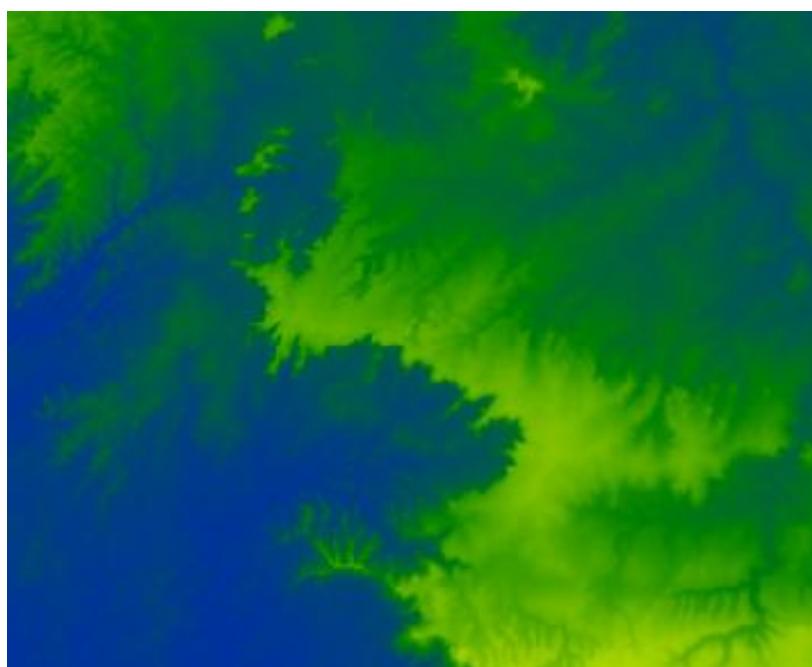


Figura 19 - Exemplo do SRTM da área de interesse.

Realizou-se a operação `resample('bilinear')` e `reproject({crs, scale:10})` sobre o SRTM, produzindo um *raster* em 10 m no SIRGAS 2000. A reamostragem bilinear suaviza gradientes (interpolação contínua) e pode atenuar ruído, mas não cria detalhes reais além da resolução nativa: trata-se de um refinamento de grade útil para alinhar escala de análise e cartografia, não para “ganhar” informação.

### 11.3. Cálculo de declividade (graus)

A função `ee.Terrain.slope` foi aplicada com operadores de derivada espacial sobre a superfície reamostrada (10 m), gerando declividade em graus. A conversão em graus facilita a comunicação técnica (0–90°) e a comparação com classes padronizadas (ex.: 0–10°, 10–20° etc.). O `clip(roi)` garantiu que apenas a área de interesse fosse mantida.

Ainda, construiu-se uma imagem categórica `slopeClass` por encadeamento de `where(...)` , com rótulos inteiros 1–5 mapeando faixas de declividade: 0–10°, 10–20°, 20–30°, 30–45° e ≥45°. Essa discretização traduziu um contínuo geomorfológico em classes interpretáveis para planejamento (ex.: mecanização, risco erosivo, engenharia de estradas).

A chamada `updateMask(slope10.mask())` propagou a máscara original (NoData/água sem elevação) para a classificação, mantendo consistência entre o *raster* contínuo (declividade) e o categórico. A banda resultante foi renomeada para `slope\_class` , facilitando inspeções e reduções por atributo.

Definiu-se uma paleta de cor (branco→vermelho) coerente com a progressão de declividade e uma lista de rótulos (`classLabels`) para ancorar a legenda. Essa escolha cromática cria um gradiente intuitivo: áreas planas em cores frias/claras e áreas íngremes em cores quentes, reforçando a leitura visual do relevo.

### 11.4. *Hillshade* para leitura de relevo

Gerou-se um sombreamento (`ee.Terrain.hillshade`) a partir do SRTM (30 m) e plota-se como camada de fundo (Figura 22). O *hillshade*, em tons de cinza 0–255, replicou as sombras projetadas conforme azimute/elevação solar padrão, adicionando

percepção de relevo (efeito 3D) e ajudando a validar visualmente padrões da declividade.

Sobrepoê-se `slopeClass` ao *hillshade*, definiu-se `min/max` e `palette`, adiciona-se o contorno da ROI como linha preta e centraliza-se o mapa na área. Essa composição (fundo sombreado + classes semitransparentes/contrastadas + contorno) foi um padrão eficaz para análise exploratória e comunicação com stakeholders.

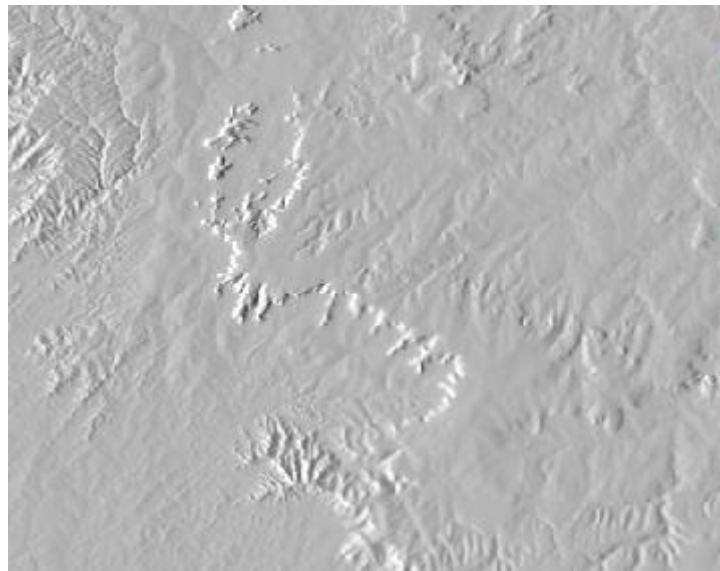


Figura 20 - *Hillshade* em tons de cinza 0–255 com a percepção de relevo (efeito 3D)

Criou-se um *raster* de áreas por *pixel* (`pixelArea()/10000`) e empilha-se com a banda categórica (`slopeClass`) para um `reduceRegion` com `Reducer.sum().group(...)` (Figura 23). A soma foi agrupada pelo valor da classe (Tabela 5), entregando áreas agregadas (ha) de cada faixa. O uso de `scale: slope10.projection().nominalScale()` assegurou consistência entre a escala de amostragem e a projeção efetiva da declividade.

Converte-se o dicionário agrupado em `FeatureCollection`, associando a cada rótulo inteiro um rótulo textual (ex.: 1 → “0–10°”) via `labelsDict`. O resultado é uma tabela legível com campos `classe`, `classe\_label` e `area\_ha`, adequada para auditoria, exportação e integração em relatórios.

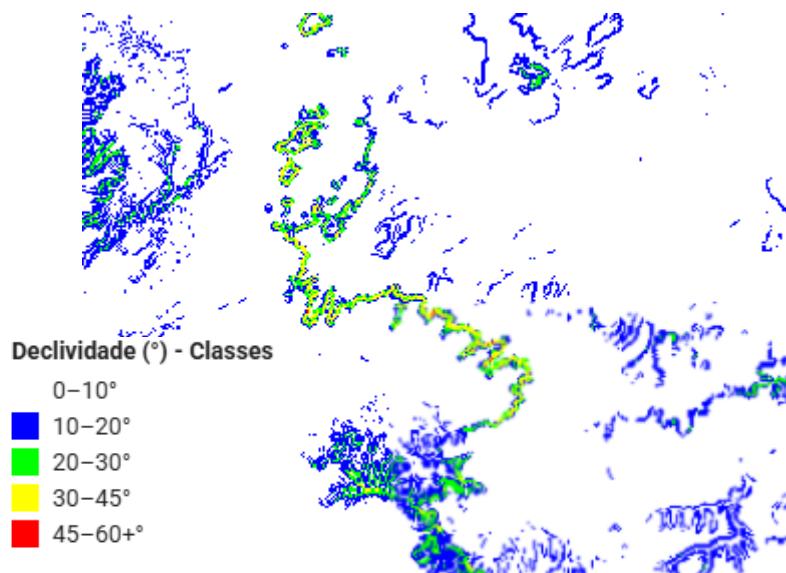


Figura 21 - Mapa com as classes de declividade para a área de interesse.

Tabela 5 - Síntese das áreas por classe de declividade (hectares)

==== Síntese das Áreas por Classe de Declividade (ha) ====  
JSOI

classe_label	area_ha
0-10°	184,785.906
10-20°	27,165.19
20-30°	9,556.216
30-45°	2,928.921
45-60+°	968.937

## 12. ÁREAS URBANAS ARBORIZADAS

Neste exemplo implementou-se uma abordagem de classificação orientada a objetos (GEOBIA) sobre imagens *SkySat* no *Google Earth Engine* (GEE). Em vez de classificar pixel a pixel, usou-se nesta aula um script que segmenta a cena em objetos espectralmente homogêneos por meio do algoritmo *Simple Non-Iterative Clustering* (SNIC) (ACHANTA; SUSSTRUNK, 2017), extraiu-se estatísticas intra-objeto (médias das bandas), treinou-se um modelo *Random Forest* com amostras vetoriais e, por fim, quantificou-se as classes (hectares e km<sup>2</sup>) em uma tabela formatada no console. Abaixo, descrevem-se os fundamentos e cada etapa operacional.

### 12.1. Delimitação da área de estudo

O processo iniciou com a definição de uma ROI (região de interesse) com os vértices de um polígono delimitado por coordenadas geográficas. Como utilizou-se imagens de alta resolução da SKYSAT, foi necessário selecionar uma região que tivesse imagem disponível. Selecionou-se, portanto, a região do entorno de New Bedford, Massachusetts (Costa Sul do estado), cidade histórica às margens do rio Acushnet e de Buzzards Bay.

### 12.2. Aquisição, mosaico e composição de falsa cor (*SkySat*)

Carregou-se a coleção pública SKYSAT/GEN-A/PUBLIC/ORTHO/MULTISPECTRAL (BUTLER, 2014), filtrada pela ROI. A função ‘mosaico’ compôs uma única imagem a partir das cenas filtradas (útil quando há múltiplos tiles/cenas sobre a área). Selecionou-se as quatro bandas B, G, R, N (NIR) e recortou-se ao limite da ROI. Para inspeção visual, utilizou-se uma composição de falsa cor (NIR-G-B) conforme Figura 24a, usual para destacar vegetação (NIR alto → tons mais claros/avermelhados), com parâmetros de mínimo e máximo apenas para visualização (não alteram os valores radiométricos).

### 12.3. Segmentação SNIC: teoria e parametrização

A Segmentação por Clustering espaço-espectral SNIC (*Simple Non-Iterative Clustering*) agrupou pixels contíguos em super-pixels com compromisso entre

homogeneidade espectral e compacidade espacial. A seguir os parâmetros que foram utilizados para a construção da segmentação:

- **size**: controla a densidade de sementes iniciais (quanto menor, maior o número de objetos e mais finos os detalhes);
- **compactness**: pondera a contribuição da distância espacial vs. espectral. Valores maiores favorecem objetos mais compactos; valores menores permitem maior ajuste aos gradientes espectrais;
- **connectivity**: define conectividade 4 ou 8; 8 é mais permissiva para contornos;
- **Neighborhood Size**: janela de processamento; valores maiores reduzem bordas de blocos em cenas extensas, ao custo de memória.

O SNIC retornou, entre outros, um raster de rótulos de clusters (clusters) e bandas de estatísticas intra-segmento (como B\_mean, G\_mean, R\_mean, N\_mean), que representam a assinatura média de cada objeto (Figura 24b). Essas médias suavizam o ruído intrínseco de alta resolução e reduzem a variância dos preditores para a etapa de classificação.



(a)



(b)

Figura 22 - Imagem SKYSAT com composição NIR-G-B (a); rótulos de clusters (clusters) e bandas de estatísticas intra-segmento (b).

Como apoio interpretativo, foi possível visualizar as médias por objeto em falsa cor (NIR\_mean, G\_mean, B\_mean), o que explicitou a suavização espectral obtida pela segmentação. Adicionalmente, extraiu-se um mapa de contornos dos segmentos para inspeção de sobre-segmentação (objetos excessivamente pequenos) ou sub-segmentação (objetos muito grandes que mesclam alvos distintos).

## 12.4. Vetorização e junção de atributos

O raster de rótulos foi vetorizado com `reduceToVectors`, convertendo cada cluster em polígono. A escolha do `scale` (p.ex., 2 m) nessa etapa atuou como uma simplificação geométrica, reduzindo a complexidade dos polígonos provenientes de um raster submétrico. Em seguida, usou-se `reduceRegions` para transferir as estatísticas calculadas no raster (p.ex., `B_mean_mean`) para os polígonos resultantes, formando um conjunto vetorial temático (um polígono por segmento, com médias espectrais como atributos) - vide Figura 25. Essa coleção pode ser exportada para o formato shapefile para edição ou inspeção externa (Por exemplo no QGIS).

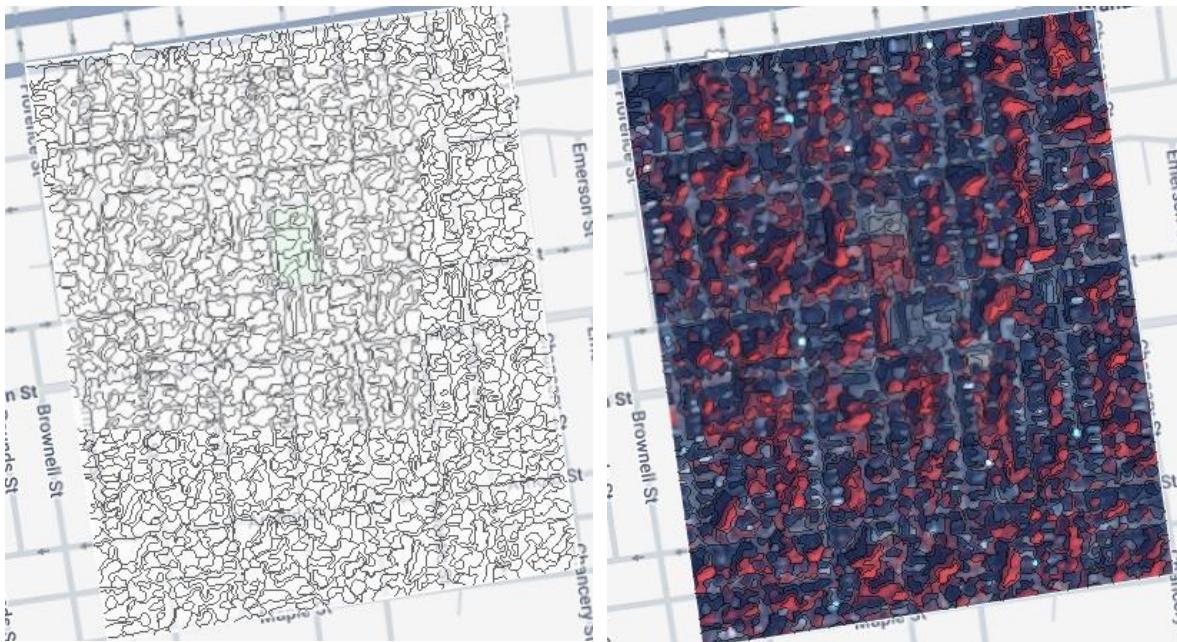


Figura 23 - Síntese da segmentação para a área em questão.

## 12.5. Preparação das amostras

As amostras de treinamento (`FeatureCollection`) devem conter um atributo categórico (no script, `classes`). Para simplificar o código recodificou o problema para duas classes: definiu-se um conjunto `classesParaDois` (p.ex., `{2}`) e atribuiu-se rótulo 2 às feições cuja classe original pertenciam ao conjunto; às demais, rótulo 1. Essa estratégia foi útil quando se deseja distinguir um alvo principal (p.ex., “áreas

arborizadas”) do restante (“demais áreas”), aumentando a robustez de classificação quando há poucos exemplos por classe ou quando classes secundárias se confundem espectralmente. Para tanto, utilizou-se o arquivo *shapefile* exportado com todos os segmentos e, no QGIS, construiu-se o conjunto de treinamento - Figura 25a (que está disponível ao no script e aberto ao público em: ‘users/anapaulacorte/amostra\_segmentos\_class’).

## 12.6. Extração de preditores e amostragem

Os preditores selecionados foram as bandas de média por segmento (B\_mean, G\_mean, R\_mean, N\_mean). Com sampleRegions, o script extraiu os valores desses preditores no local das amostras e carrega o rótulo classe2. Como cada amostra interceptava um conjunto de pixels/segmentos, usou-se médias por objeto tendendo a reduzir a variância intraclasse e a melhorar a separabilidade.

## 12.7. Treinamento do Random Forest (RF)

O Random Forest é um ensemble de árvores de decisão treinadas em amostras *bootstrap* e subconjuntos aleatórios de variáveis por divisão. Os principais hiperparâmetros usados:

- `numberOfTrees` = 200: aumenta a estabilidade do ensemble (até certo ponto) e reduz a variância
- `minLeafPopulation` = 2: controla o tamanho mínimo dos nós folha, evitando árvores excessivamente profundas (*overfitting*)
- `variablesPerSplit` = 2: número de preditores considerados por divisão; promove decorrelação entre árvores
- `bagFraction` = 0.7: fração de amostras usada em cada árvore (*bootstrapping* parcial)

Treinou-se o classificador com os preditores e o atributo-alvo classe2. O RF é apropriado para dados espectrais por lidar bem com não-linearidades, interações e ruído.

Aplicou-se o classificador à imagem de preditores por objeto (`segMeans.classify`). O resultado, inteiro (1 e 2), foi recortado à ROI e plotado com paleta de duas cores. A

legenda foi construída com ui.Panel e adicionada ao mapa para documentação visual da classe 1 vs. classe 2 (Figura 26b).

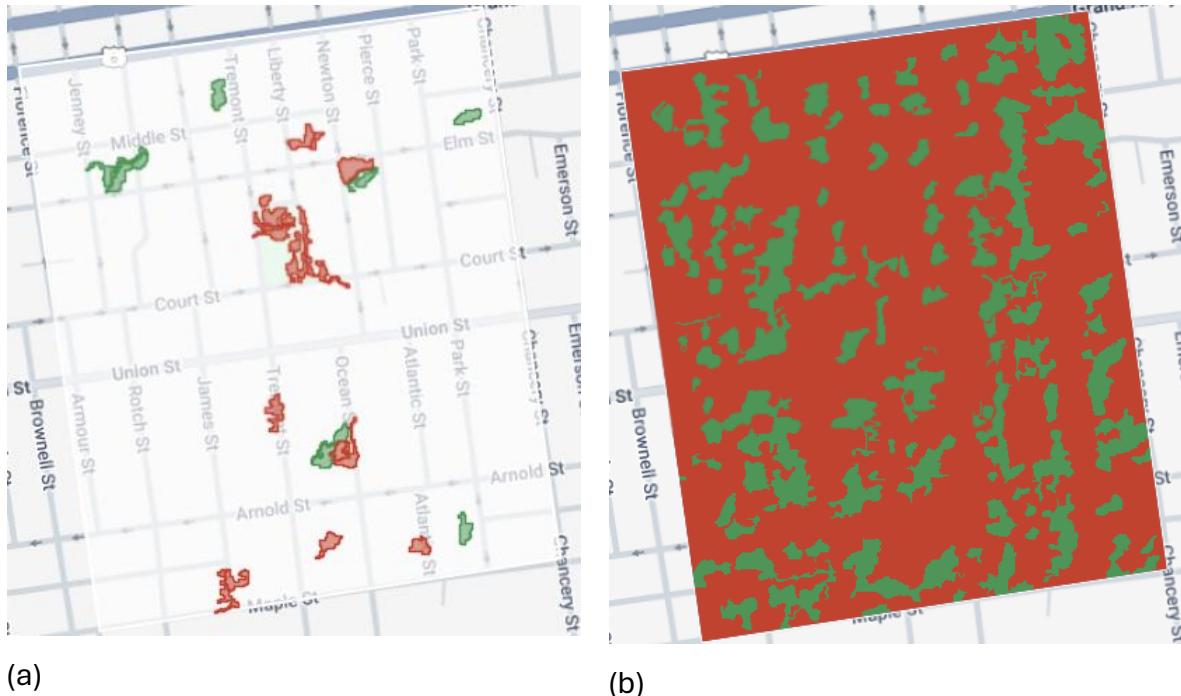


Figura 24 - Exemplo das amostras de input para a classificação dos segmentos (a) e resultado da classificação (b).

## 12.8. Cálculo de área por classe (hectares e km<sup>2</sup>)

A quantificação espacial baseou-se em ee.Image.pixelArea(), convertida para hectares (divisão por 10 000). O script demonstrou duas abordagens:

- **Redução agrupada:** empilhou-se areaHa com a imagem de classes e usou-se reduceRegion com Reducer.sum().group(...) para obter a soma de áreas por valor de classe em uma única passagem

- **Soma por máscara:** para cada classe, mascarou-se areaHa com os pixels da classe e soma-se a área com reduceRegion. Esta via é didática e facilita montar uma tabela customizada (apenas duas colunas: “classe” e “áreas”).

Para fins de compreensão dos valores, o script organizou os resultados em um FeatureCollection com duas colunas (classe, áreas) e imprimiu tanto a tabela “crua” quanto um ui.Chart do tipo Table. Isso tornou o resultado legível e exportável

(copiar/colar) e trouxe a síntese da distribuição de área entre as classes (De acordo com a Tabela 6).

Tabela 6 - Síntese das áreas das classes arborizada e demais áreas.

classe	areas
Áreas arborizadas	10.751
Demais áreas	32.051

## **13. USO DA COLEÇÃO MAPBIOMAS**

Nesta aula, será demonstrado o uso do *Google Earth Engine* em conjunto com a base de dados do *MapBiomas Collection 10* (MAPBIOMAS, 2025) para realizar uma análise de uso e cobertura da terra no ano de 2023 em municípios selecionados da Amazônia. O exercício irá abranger desde a importação e recorte do arquivo *raster* para a área de interesse, passando pelo cálculo das áreas ocupadas por cada classe temática, até a visualização dos resultados em forma de mapas e gráficos interativos. Por fim, será mostrado como exportar os dados obtidos em formato tabular (CSV) e *raster* (GeoTIFF), permitindo a utilização em análises posteriores em softwares de SIG e estatística.

### **13.1. Preparação do Raster e escolha da coleção**

O primeiro passo do script consistiu em carregar uma coleção de uso e cobertura da terra. Neste caso, utilizou-se o *MapBiomas Collection 10*, que é uma base consolidada de classificação temática multitemporal do território brasileiro, com resolução espacial de 30 metros. Essa coleção é construída a partir de imagens de satélite (como Landsat, Sentinel e outros) e algoritmos de classificação supervisionada. A seleção da banda de 2023 restringe a análise ao último ano disponível, garantindo consistência temporal.

Cada banda do *MapBiomas* representa o mapa temático de um ano específico, com classes codificadas por valores inteiros (ex.: floresta = 3, pastagem = 15, agricultura = 21 etc.). Essa padronização permite comparações temporais e análises estatísticas de mudanças no uso do solo. Trabalhar com a banda “classification\_2023” significa assumir um retrato atual da paisagem.

### **13.2. Importação de limites administrativos**

Para análise espacial, o script importou limites municipais disponibilizados pelo IBGE. Esses limites estão organizados em diferentes níveis (estado, município, setor censitário etc.), e aqui se utiliza o nível 3 (município). A escolha por polígonos oficiais assegura consistência cartográfica e permite comparabilidade entre diferentes estudos e escalas administrativas.

### 13.3. Filtragem espacial por municípios de interesse

A seleção dos municípios foi feita de duas formas: (i) por nomes em uma lista (ex.: Altamira, Apuí, Placas e Itaituba) ou (ii) por interseção com uma área de interesse desenhada manualmente no mapa (AOI). Essa flexibilidade foi essencial, pois permite tanto análises direcionadas (lista) quanto dinâmicas (interseção espacial).

### 13.4. Geometria combinada e recorte do raster

Uma vez selecionados os municípios, foi criada uma geometria combinada que representa a união espacial de todos eles (Figura 27). Esse polígono agregado foi então utilizado para realizar o *clipping* do *raster*, ou seja, recortar apenas os pixels do *MapBiomas* 2023 que estão dentro dos limites municipais selecionados. Isso reduziu o volume de dados e focou a análise no território de interesse.

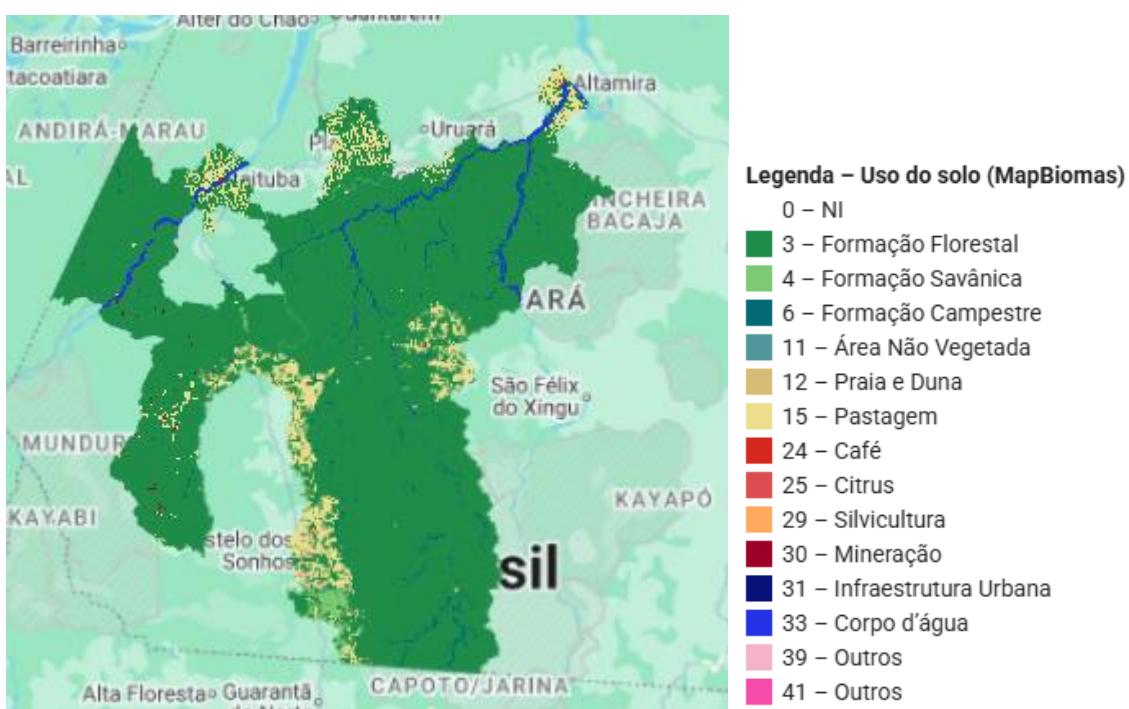


Figura 25 - Síntese do recorte do uso do solo do *MapBiomas* 2023 para os municípios selecionados.

### **13.5. Cálculo da área por classe**

O cálculo de áreas foi feito a partir do conceito de área por *pixel*. Cada *pixel* da imagem tem  $30\text{ m} \times 30\text{ m}$ , equivalendo a  $900\text{ m}^2$ . O *script* converteu esses valores para quilômetros quadrados (dividindo por  $1\text{e}6$ ). Em seguida, aplicou-se um *reducer* que soma as áreas por classe temática, agrupando-as de acordo com o código da classificação.

### **13.6. Redução regional e agrupamento**

A função *reduceRegion* foi aplicada sobre a geometria dos municípios, combinando as bandas de área do pixel e da classificação temática. O resultado foi um dicionário agrupado (*groups*), que contém para cada classe identificada a soma das áreas em  $\text{km}^2$ . Essa estrutura de dados é compacta e eficiente para análises estatísticas.

### **13.7. Conversão em *FeatureCollection***

Para facilitar a manipulação e visualização, os resultados foram convertidos em um *FeatureCollection*. Cada elemento corresponde a uma classe de uso do solo, com dois atributos principais: o código da classe (*class*) e a área ocupada em  $\text{km}^2$  (*area\_km2*). Essa transformação foi importante para gerar tabelas, gráficos e exportações.

### **13.8. Visualização em tabela de áreas**

A etapa seguinte organizou os dados em tabela com as respectivas classes adotadas no *MapBiomas* em  $\text{km}^2$ . Essa representação foi útil para comparação rápida entre categorias, destacando quais usos ocupam maior ou menor extensão territorial (Figura 28).

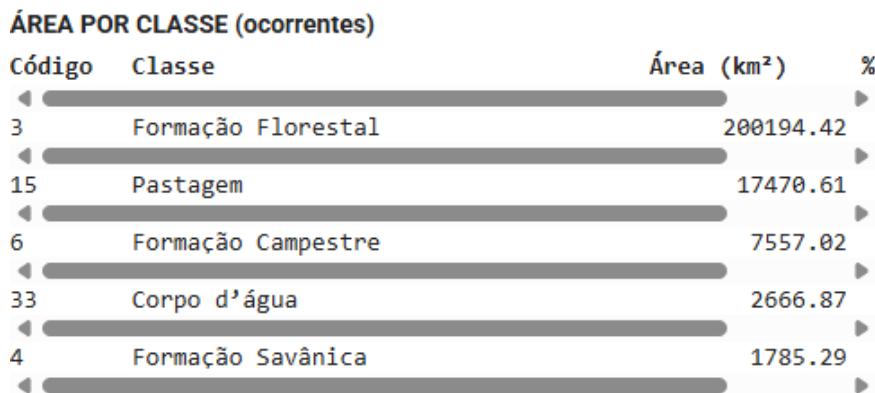


Figura 26 - Representação das classes e respectivas áreas.

### 13.9. Integração com a legenda oficial do MapBiomas e Exportação de Resultados

Para garantir consistência visual e comparabilidade, o script utilizou a paleta oficial de cores do *MapBiomas*. Essa padronização foi essencial em trabalhos científicos e técnicos, pois evita ambiguidades e facilita a leitura cartográfica. O mapa exibido mostrou a distribuição espacial das classes dentro dos municípios selecionados, com a simbologia reconhecida pela comunidade científica e técnica.

O *script* centralizou a visualização no conjunto de municípios selecionados e ajustou o nível de zoom para 7, adequado para análises regionais. Essa configuração garantiu que o usuário tenha uma visão panorâmica dos territórios, sem perder a capacidade de identificar detalhes locais.

Os resultados estatísticos de área foram exportados em formato CSV para o *Google Drive*. Esse formato é amplamente utilizado em planilhas (Excel, R, Python, QGIS, etc.), permitindo análises adicionais, integração com sistemas de informações geográficas (SIG) e geração de relatórios técnicos.

O raster recortado também foi exportado no formato GeoTIFF, que é um padrão aberto e compatível com softwares SIG. A opção *cloudOptimized: true* facilitou seu uso em plataformas na nuvem. Essa etapa assegurou que o mapa temático possa ser utilizado fora do GEE, em análises mais complexas ou em integração com outras camadas geoespaciais.

## 14. UPSCALING GEDI

Nesta aula, será desenvolvido um fluxo de integração de dados de sensoriamento remoto e LiDAR orbital (GEDI) para a estimativa da altura do dossel em florestas plantadas. O processo iniciou com a definição da área de estudo e a seleção da classe de interesse no *MapBiomas*, assegurando que as análises considerem apenas florestas plantadas. Em seguida, foram aplicados filtros de qualidade nos dados GEDI e extraídas métricas de altura para compor a variável de referência. Paralelamente, foram preparados conjuntos de dados ópticos (Landsat 9) e de SAR (Sentinel-1), complementados por índices espectrais e métricas texturais/focais. Todos esses produtos foram organizados em um *stack* de preditores, ao qual também foram adicionados *embeddings* derivados de imagens de satélite. A partir daí, realizou-se a amostragem sobre os *footprints* GEDI e a divisão em bases de treino e teste, que alimentam modelos de Random Forest para regressão. Por fim, os modelos foram avaliados estatisticamente e aplicados para mapear a altura do dossel em toda a área de estudo, gerando produtos espaciais e indicadores quantitativos de desempenho que permitem compreender a eficácia de cada abordagem de modelagem.

### 14.1. Definição da área de estudo

O código se inicia definindo a região de interesse (ROI) através de um polígono com coordenadas geográficas, que é transformado em um *FeatureCollection*. Em seguida, o contorno da ROI foi estilizado com a função `.style()` para visualização (Figura 29), exibindo apenas o limite sem preenchimento. O mapa é centralizado sobre a área com `Map.centerObject()`, garantindo que todas as operações futuras considerem apenas os pixels dentro dessa região.

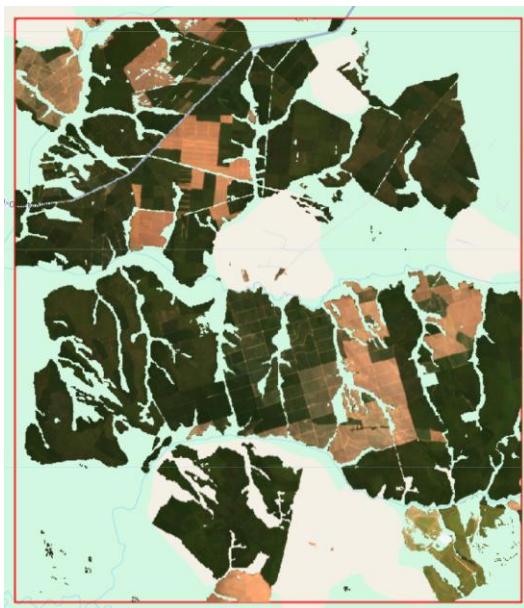


Figura 27 - Composição RGB para a região de interesse.

## 14.2. Seleção de floresta plantada com *MapBiomas*

Para focar apenas em áreas com florestas plantadas, acessou-se a imagem *MapBiomas* com `ee.Image()`, selecionou-se a banda correspondente ao ano de interesse com `.select()`, e criou-se uma máscara usando `.remap()` combinada com `.selfMask()`. O resultado manteve apenas os pixels da classe desejada e foi recortado para a ROI com `.clip()`. Essa máscara será aplicada sobre todos os conjuntos de dados para garantir que as análises considerem somente áreas de floresta plantada.

## 14.3. Filtragem e extração de dados GEDI L2A

O script acessa a coleção GEDI L2A (DUBAYAH et al., 2021) com `ee.ImageCollection()` e aplica filtros de qualidade por meio de funções como `l2a_quality_mask()`, `fullPower()` e `night()`. Cada função atualiza a máscara da imagem com `.updateMask()` para remover observações de baixa qualidade, beams incompletos, dados captados durante o dia ou de baixa qualidade. Após isso, foi selecionada a métrica `rh98` (altura do dossel) e as imagens foram combinadas em um mosaico. Finalmente, os pontos GEDI foram amostrados com `.sample()` e filtrados para remover valores nulos, permitindo gerar os *footprints* que servirão de referência para o ajuste dos modelos (Figura 30).



Figura 30- Amostras GEDI L2A de qualidade. Pontos cinza representam todas as amostras de qualidade, pontos azuis correspondem às amostras selecionadas para treino e pontos vermelhos às amostras selecionadas para teste.

#### 14.4. Preparação dos dados ópticos (Landsat 9)

As imagens Landsat 9 foram acessadas como `ee.ImageCollection()` e passaram por duas funções principais: `maskL8sr()`, que remove nuvens e sombras utilizando operações *bitwise* sobre a banda `QA_PIXEL`, e `applyScaleFactors()`, que ajustou os valores radiométricos. Depois, foi criada uma composição mediana com `.median()` para reduzir ruídos e fornecer uma imagem representativa do período analisado. A partir dessa composição, índices como NDVI, EVI e SAVI foram calculados (Figura 31), resultando em camadas que representam a vegetação sob diferentes perspectivas biofísicas.

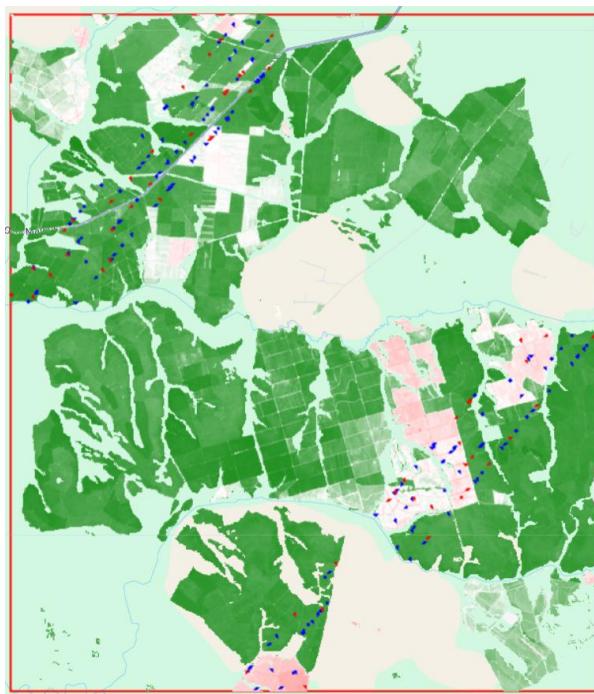


Figura 28 - Mediana do NDVI calculada para o período, considerando apenas as florestas plantadas da região de interesse.

#### 14.5. Preparação dos dados de SAR (Sentinel-1)

As imagens Sentinel-1 foram filtradas e aplicou-se máscaras de florestas plantadas usando `.updateMask()`. Foram selecionadas polarizações VV e VH (Figura 32) e calculados índices derivados, como RVI e COPOL, utilizando. Em seguida, a mediana temporal das imagens foi obtida com `.median()`. Esses produtos de SAR são capazes de capturar informações estruturais da vegetação que não dependem de condições atmosféricas ou luminosas, complementando os dados ópticos.

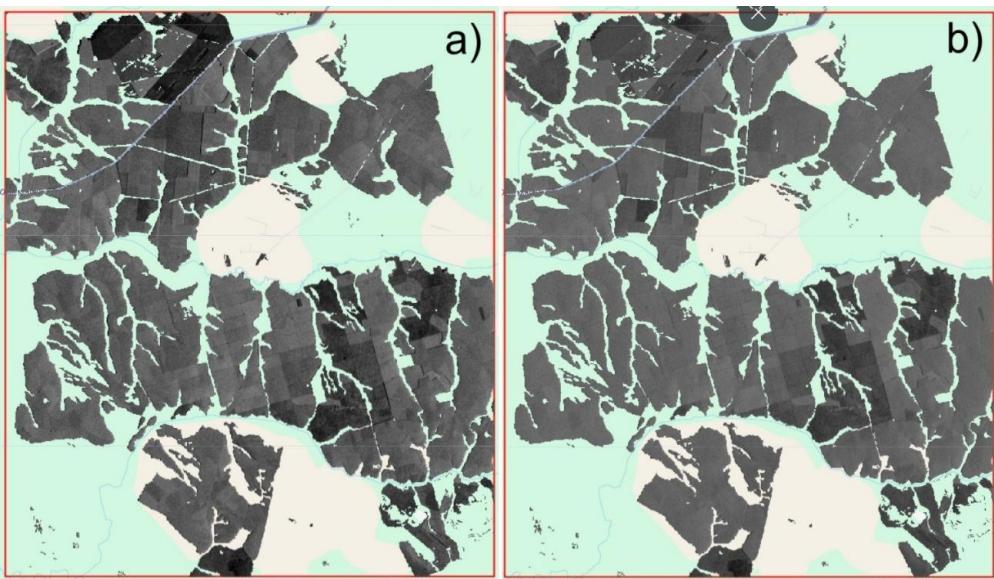


Figura 29 - Mapas da mediana do período para as bandas de SAR: a) VV e b) VH.

#### 14.6. Extração de métricas texturais e focais

Para enriquecer a base de dados e aumentar a robustez da modelagem, foram calculadas 18 métricas texturais usando `.glcmTexture()` para cada uma das bandas e dos índices dos conjuntos ópticos e de SAR, avaliando padrões de heterogeneidade espacial. Além disso, métricas focais (máximo, mínimo, média e desvio padrão) foram derivadas a partir de um *kernel* quadrado (3x3 pixels), permitindo que cada pixel incorpore informações de vizinhança. Essas operações ajudam o modelo a capturar características espaciais mais complexas da vegetação ao redor.

#### 14.7. Construção do stack de preditores

Todos os produtos (ópticos, SAR, texturas e métricas focais) foram combinados em um único *stack* de preditores (`predictors_optSAR`). Paralelamente, *embeddings* do *Google Satellite Embedding V1* foram adicionados como outro conjunto de preditores (`predictors_embeddings`). Essa organização possibilita o processo de modelagem, criando duas abordagens complementares: uma baseada em dados físicos (OptSAR) e outra em representações aprendidas (`embeddings`).

## 14.8. Amostragem e preparação do dataset de modelagem

Cada ponto GEDI recebeu um buffer de 25 m usando `.buffer()`, e os valores medianos dos preditores foram extraídos com `.reduceRegions()`. O resultado foi um *dataset* onde cada *footprint* contém a variável observada pelo instrumento LiDAR GEDI (RH98) e os valores correspondentes dos preditores. Amostras com valores nulos foram filtradas e um subconjunto aleatório de 200 observações foi selecionado aleatoriamente (`.randomColumn()`) para redução de esforço computacional. Posteriormente, os dados foram divididos em treino e teste, garantindo independência para a validação dos modelos.

## 14.9. Treinamento dos modelos Random Forest

Dois modelos Random Forest foram treinados com `.train()`: um utilizando o *stack* OptSAR e outro com *embeddings*. Ambos foram configurados para regressão com `.setOutputMode('REGRESSION')`. Após o treinamento, os modelos foram aplicados para toda as florestas plantadas na área de estudo com `.classify()`, gerando mapas contínuos de altura predita do dossel. As amostras de treino e teste foram adicionadas ao mapa para referência visual (Figura 33).

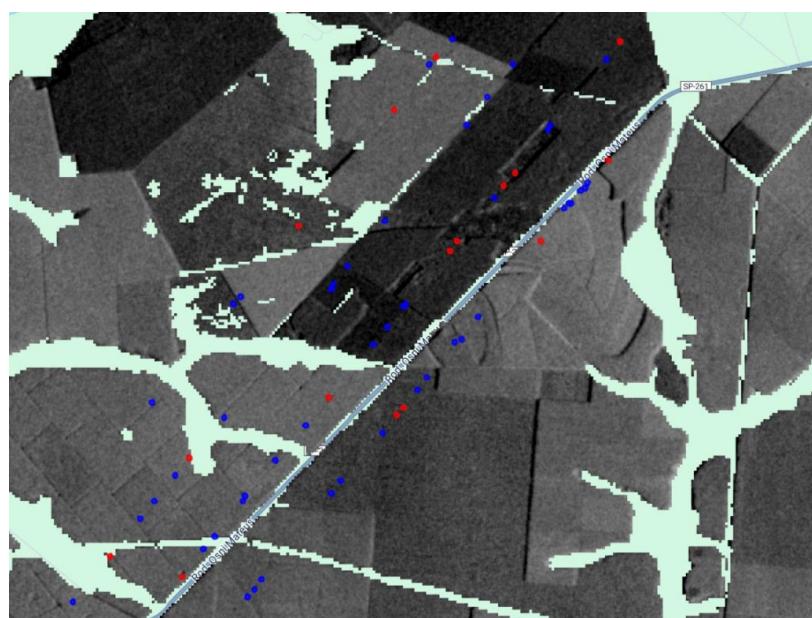


Figura 30 - Amostras selecionadas para modelagem: pontos azuis correspondem às amostras de treino e pontos vermelhos às amostras de teste.

## 14.10. Avaliação dos modelos e análise de desempenho

A avaliação envolveu comparar os valores observados e preditos usando `.classify()` nas amostras de teste. Gráficos de dispersão (Figura 34) foram gerados com `ui.Chart.feature.byFeature()` e incluíram linhas de regressão 1:1 e *trendlines*. Métricas como RMSE, rRMSE e R<sup>2</sup> foram calculadas com funções personalizadas (Tabela 7). Estatísticas descritivas dos conjuntos de treino, teste (Tabela 8) e mapas de predição (Figura 35) também foram geradas, resumindo o comportamento dos modelos e possibilitando interpretação quantitativa.

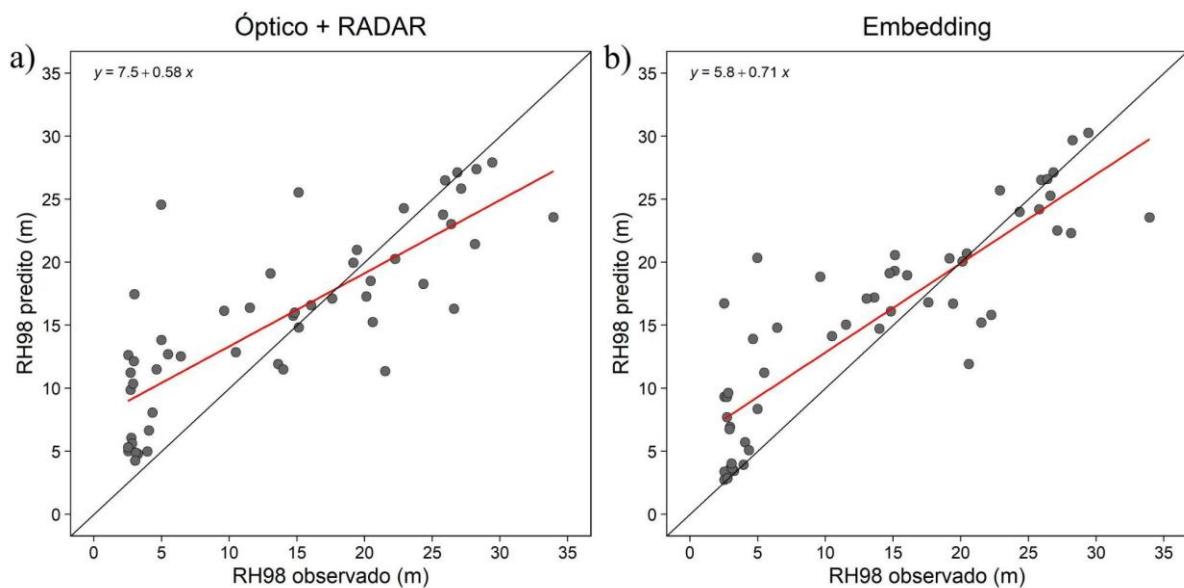


Figura 31 - Gráficos de observado versus predito para os modelos: a) Óptico + SAR e b) Embeddings. As linhas pretas indicam a relação 1:1 entre os valores observados e preditos, enquanto as linhas vermelhas representam a tendência da regressão entre as variáveis.

Tabela 7 - Métricas estatísticas de desempenho dos modelos ajustados e aplicados à base de teste.

==== Desempenho dos modelos ===

Modelo	RMSE_m	rRMSE_%	R2
OptSAR	7.41	51.83	0.49
Embedding	5.63	39.4	0.71
◀ ▶			

Tabela 8 - Estatísticas descritivas dos valores de RH98 das bases de treinamento e teste, bem como das previsões realizadas para as florestas plantadas da região de estudo, considerando cada abordagem de modelagem.

==== Estatísticas RH98 ===

Conjunto	RH98_mín	RH98_máx	RH98_média	RH98_desvio
Treinamento	2.39	41.73	16.28	11.15
Teste	2.53	37.02	14.3	10.51
Predição OptSAR	2.9	34.12	18.01	6.3
Predição Embedding	2.6	32.09	17.5	7.24

◀ ▶

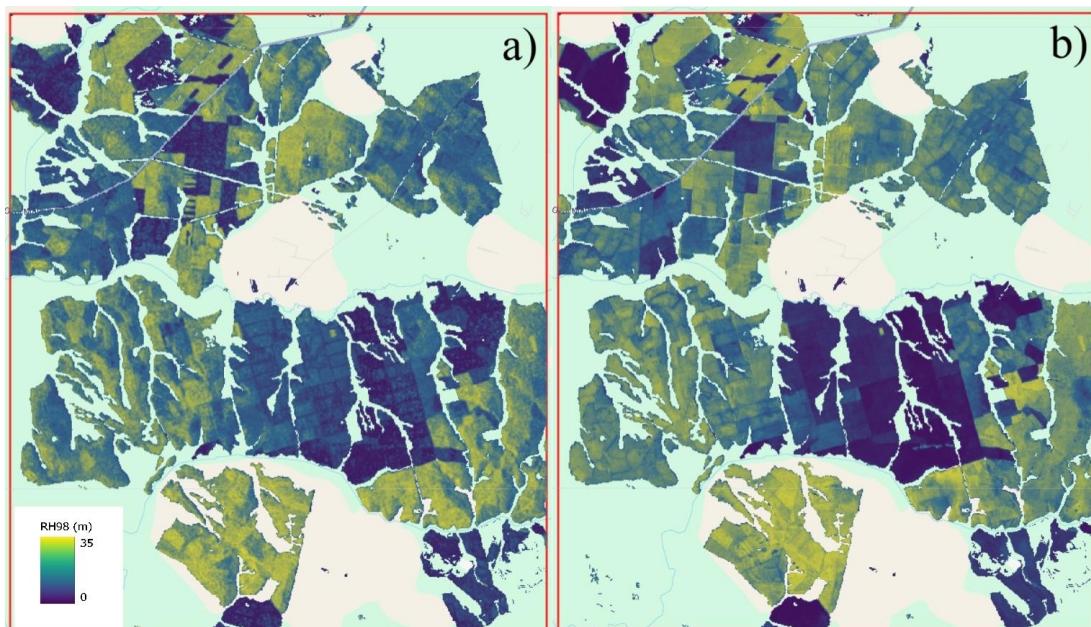


Figura 32 - Mapas de RH98 (altura de dossel) para as áreas de floresta plantada da região de estudo. a) Óptico + SAR e b) Embeddings.

## 15. REFERÊNCIAS BIBLIOGRÁFICAS

- ACHANTA, R.; SUSSTRUNK, S. Superpixels and polygons using simple non-iterative clustering. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. p. 4651-4660, 2017.
- AGAPIOU, A. Estimating proportion of vegetation cover at the vicinity of archaeological sites using sentinel-1 and-2 data, supplemented by crowdsourced openstreetmap geodata. **Applied sciences**, v. 10, n. 14, p. 4764, 2020.
- AMANI, M. *et al.* Google earth engine cloud computing platform for remote sensing big data applications: A comprehensive review. **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing**, v. 13, p. 5326-5350, 2020.
- BAIG, M. H. A. *et al.* Derivation of a tasseled cap transformation based on Landsat 8 at-satellite reflectance. **Remote Sensing Letters**, v. 5, n. 5, p. 423-431, 2014.
- BREIMAN, L. Random forests. **Machine learning**, v. 45, n. 1, p. 5-32, 2001.
- BROWN, C. F. *et al.* AlphaEarth Foundations: An embedding field model for accurate and efficient global mapping from sparse label data. **arXiv preprint arXiv:2507.22291**, 2025.
- BUTLER, D. Many eyes on Earth. **Nature**, v. 505, n. 7482, p. 143-144, 2014.
- CHANDER, G.; MARKHAM, B. L.; HELDER, D. L. Summary of current radiometric calibration coefficients for Landsat MSS, TM, ETM+, and EO-1 ALI sensors. **Remote sensing of environment**, v. 113, n. 5, p. 893-903, 2009.
- CHOATE, M. J. *et al.* Landsat 9 geometric characteristics using underfly data. **Remote Sensing**, v. 14, n. 15, p. 3781, 2022.
- CLAVERIE, M. *et al.* The Harmonized Landsat and Sentinel-2 surface reflectance data set. **Remote sensing of environment**, v. 219, p. 145-161, 2018.
- CRAWFORD, C. J. *et al.* The 50-year Landsat collection 2 archive. **Science of Remote Sensing**, v. 8, p. 100103, 2023.
- CRIST, E. P. A TM tasseled cap equivalent transformation for reflectance factor data. **Remote sensing of Environment**, v. 17, n. 3, p. 301-306, 1985.
- DUBAYAH, R. *et al.* GEDI L2A Elevation and Height Metrics Data Global Footprint Level V002 [Data set]. **NASA EOSDIS Land Processes DAAC**, 2021.
- FARR, T. G. *et al.* The shuttle radar topography mission. **Reviews of geophysics**, v. 45, n. 2, 2007.
- HAAS, J.; BAN, Y. Sentinel-1A SAR and sentinel-2A MSI data fusion for urban ecosystem service mapping. **Remote Sensing Applications: Society and Environment**, v. 8, p. 41-53, 2017.
- HABOUDANE, D. *et al.* Hyperspectral vegetation indices and novel algorithms for predicting green LAI of crop canopies: Modeling and validation in the context of precision agriculture. **Remote sensing of environment**, v. 90, n. 3, p. 337-352, 2004.
- HARALICK, R. M.; SHANMUGAM, K.; DINSTEIN, I. H. Textural features for image classification. **IEEE Transactions on systems, man, and cybernetics**, n. 6, p. 610-621, 2007.
- HUETE, A. *et al.* Overview of the radiometric and biophysical performance of the MODIS vegetation indices. **Remote sensing of environment**, v. 83, n. 1-2, p. 195-213, 2002.
- HUETE, A. R. A soil-adjusted vegetation index (SAVI). **Remote sensing of environment**, v. 25, n. 3, p. 295-309, 1988.
- IRONS, J. R.; PETERSEN, G. W. Texture transforms of remote sensing data. **Remote sensing of environment**, v. 11, p. 359-370, 1981.

- JUSTICE, C. O. *et al.* The MODIS fire products. **Remote sensing of Environment**, v. 83, n. 1-2, p. 244-262, 2002.
- MAPBIOMAS – Coleção 10 da série 2023 de Mapas de Cobertura e Uso da Terra do Brasil, 2025.
- NASIRZADEHDIZAJI, R. *et al.* Sensitivity analysis of multi-temporal Sentinel-1 SAR parameters to crop height and canopy coverage. **Applied Sciences**, v. 9, n. 4, p. 655, 2019.
- ROUJEAN, J. L.; BREON, F. M. Estimating PAR absorbed by vegetation from bidirectional reflectance measurements. **Remote sensing of Environment**, v. 51, n. 3, p. 375-384, 1995.
- ROUSE, J. W. Monitoring vegetation systems in the great plains with ERTS. In: **Third NASA Earth Resources Technology Satellite Symposium, 1973**. p. 309-317, 1973.
- SCHLUND, M.; ERASMI, S. Sentinel-1 time series data for monitoring the phenology of winter wheat. **Remote Sensing of Environment**, v. 246, p. 111814, 2020.
- SOUDANI, K. *et al.* Potential of C-band Synthetic Aperture Radar Sentinel-1 time-series for the monitoring of phenological cycles in a deciduous forest. **International Journal of Applied Earth Observation and Geoinformation**, v. 104, p. 102505, 2021.

