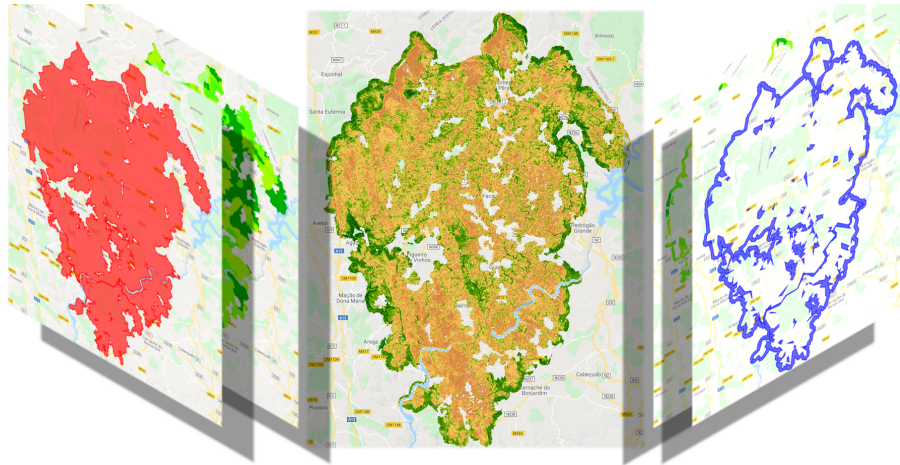




**INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA**

**Área Departamental de Engenharia de Electrónica e Telecomunicações e de Computadores**



**Modelo preditivo de recuperação da vegetação afetada por incêndios florestais**

**RUI PAULO MARQUES DOS SANTOS REIS**

Licenciado

Dissertação para obtenção do Grau de Mestre  
em Engenharia Informática e de Computadores

Orientadores : Doutora Matilde Pós-de-Mina Pato  
Doutor Nuno Miguel Soares Datia

Júri:

Presidente: Doutor Tiago Miguel Braga da Silva Dias

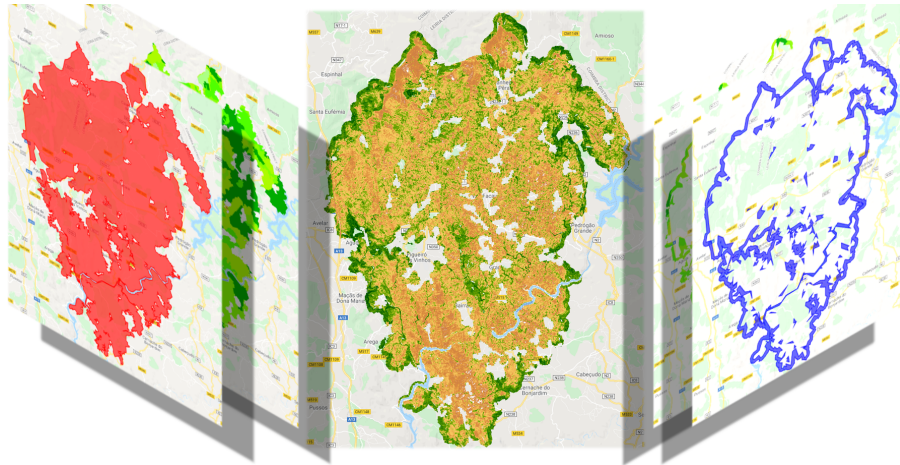
Vogais: Doutora Cristina Maria Sousa Catita  
Doutor Nuno Miguel Soares Datia

**Dezembro, 2019**



**INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA**

**Área Departamental de Engenharia de Electrónica e Telecomunicações e de Computadores**



**Modelo preditivo de recuperação da vegetação afetada por incêndios florestais**

**RUI PAULO MARQUES DOS SANTOS REIS**

Licenciado

Dissertação para obtenção do Grau de Mestre  
em Engenharia Informática e de Computadores

Orientadores : Doutora Matilde Pós-de-Mina Pato  
Doutor Nuno Miguel Soares Datia

Júri:

Presidente: Doutor Tiago Miguel Braga da Silva Dias

Vogais: Doutora Cristina Maria Sousa Catita  
Doutor Nuno Miguel Soares Datia

**Dezembro, 2019**

*Dedicado a Ti, a minha melhor metade, e aos nossos filhos.*

# Agradecimentos

As minhas primeiras palavras de profundo agradecimento são dirigidas à Doutora *Célia Gouveia*. É nos seus proverbiais ombros de gigante em que me suporto para escrever esta dissertação. Foi sua a ideia de realizar este trabalho e os seus artigos, e ensinamentos, foram fundamentais. Muito, muito obrigado!

No mesmo plano estão os meus orientadores, *Matilde Pós-de-Mina Pato* e *Nuno Datia*, que me honraram com o convite para fazer este trabalho, aturaram as minhas dúvidas, ensinaram-me a organizar as ideias e reviram este texto, várias vezes, apontando caminhos e dando as suas, sempre úteis e pertinentes, sugestões. Mesmo em momentos em que eu não compreendia a sua ausência, nunca se pouparam e estiveram sempre do meu lado. Foi um grande prazer trabalhar convosco e espero um dia poder retribuir, de alguma forma, a Vossa dedicação.

Não posso deixar de reconhecer o papel do *ISEL*, por ter aceite a minha candidatura no Mestrado em Engenharia Informática e de Computadores, por me ter acolhido e pelo muito que me ensinou. Do *ISEL* darei sempre testemunho de uma grande escola de engenharia.

A componente basilar de uma escola são os seus professores. Os meus Professores, generosos, formadores, homens e mulheres que nunca me negaram um ensinamento, a resposta a uma dúvida e sempre encorajaram a curiosidade. Estando, neste aspeto, duplamente grato aos meus orientadores, não quero deixar de referir todos por ordem alfabética: *Artur Ferreira; Fernando Sousa; Gonçalo Marques; Luis Assunção; Luis Morgado; Luis Osório; Paulo Trigo; e Walter Vieira*. A todos a minha profunda gratidão.

Não menos importante, agradeço-te a ti, *Carla Alexandra*. Não só por seres a minha companheira de todos os momentos, o chão que eu piso. Foste tu que me incentivaste a seguir viagem, deixaste-me o espaço necessário para ser eu a

descobrir o caminho, mesmo quando era claro que estava errado. Naturalmente, também tu foste minha Professora. Obrigado Meu Amor.

Finalmente Ricardo e Tiago, os meus adorados filhos. Sempre me incentivaram, nunca me reprimiram pelo tempo que estive ausente nem pelas promessas que não cumpri. Obrigado queridos!

# Resumo

Nos últimos anos, a ocorrência de fogos florestais em Portugal tem causado danos patrimoniais, destruição de habitats e, tragicamente, a perda de vidas humanas.

A estimativa da taxa de recuperação da vegetação numa área ardida é possível utilizando dados de deteção remota e, nomeadamente, o *Índice de Diferenças Normalizadas de Vegetação (NDVI)*.

Partindo de metodologias já definidas em trabalhos anteriores, propõe-se um procedimento que integre diversas fontes de dados cuja natureza permite melhorar os modelos preditivos existentes e não exige o conhecimento do histórico da vegetação no período anterior ao incêndio, usando em alternativa o comportamento de uma zona geometricamente adjacente como referencial o que se apresenta como uma importante vantagem no que se refere às necessidades de informação disponível.

Foram utilizados dados recolhidos pela missão *Sentinel 2* e o *Google Earth Engine*, como plataforma de desenvolvimento, na concretização da metodologia o que é, tanto quanto se conhece, uma técnica inovadora dada a escassez de trabalhos publicados nesta temática.

Deste trabalho resulta também uma biblioteca de código *Python* reutilizável, que suporta a interação com o *Google Earth Engine* na extração de dados agregados. Como prova de conceito, o modelo é aplicado às áreas afetadas pelos incêndios de 2017 de *Pedrogão Grande* e *Mata Nacional de Leiria*.

**Palavras-chave:** Fogo florestal, Deteção remota, NDVI, *Google Earth Engine*, *Python*.

# Abstract

In recent years, wildfires in Portugal lead to property damage, natural habitat destruction and the loss human lives.

The vegetation recovery rate can be estimated using remote sensing data, in particular the often-used Normalized Difference Vegetation Index (NDVI).

Taking advantage of methodologies set in previous work, by other authors, the proposed model relies on the integration of several data sources that leverage the existing estimators and will not be constrained by the availability of historical knowledge of the vegetation normal behavior prior to the wildfire episode. Instead, it will use the behavior of a geometrically neighboring area as a baseline. The fact that this method does not depend on the existence of historical data presents itself as a major advantage.

The model was implemented using *Sentinel 2* mission data and the *Google Earth Engine* platform, which is, to the author's knowledge, a novel technique given the shortage of publications citing their usage.

This work also produced a *Python* reusable library, which supports interaction with *Google Earth Engine* while extracting aggregate data. As a proof-of-concept, the model implementation will be applied to two areas affected by the 2017 wildfires: *Pedrogão Grande* and *Mata Nacional de Leiria*.

**Keywords:** Wildfire, Remote sensing, NDVI, *Google Earth Engine*, *Python*.

# Índice

<b>Lista de Figuras</b>	<b>x</b>
<b>Lista de Tabelas</b>	<b>xii</b>
<b>Lista de Listagens</b>	<b>xiii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Conceitos . . . . .	2
1.2 Trabalhos relacionados . . . . .	6
1.3 Objetivos . . . . .	7
1.4 Organização do documento . . . . .	9
<b>2 Estudo exploratório</b>	<b>10</b>
2.1 Fonte de dados de satélite . . . . .	10
2.2 Modelo matemático . . . . .	12
2.3 Ferramenta . . . . .	13
2.4 <i>Google Earth Engine</i> . . . . .	16
2.4.1 Aprendizagem e exploração . . . . .	17
2.4.2 Conceitos base . . . . .	19

<i>ÍNDICE</i>	ix
<b>3 Modelação e concretização</b>	<b>32</b>
3.1 Modelação . . . . .	32
3.1.1 Pré-processamento . . . . .	32
3.1.2 Cálculo . . . . .	35
3.2 Concretização . . . . .	36
3.2.1 Arquitetura . . . . .	36
3.2.2 Fontes de informação e ferramenta . . . . .	38
3.2.3 Pré-processamento . . . . .	43
3.2.4 Cálculo . . . . .	47
<b>4 Casos de estudo</b>	<b>50</b>
4.1 Pedrogão Grande . . . . .	51
4.2 Mata Nacional de Leiria . . . . .	52
4.3 Análise de sensibilidade . . . . .	53
4.3.1 Atualização da estimativa para <i>Pedrogão Grande</i> . . . . .	56
<b>5 Conclusões</b>	<b>59</b>
5.1 Trabalho futuro . . . . .	61
<b>Referências</b>	<b>63</b>
<b>A Nomenclatura Corine Land Cover</b>	<b>i</b>
<b>B Formato do ficheiro gerado pela classe <i>Modelo</i></b>	<b>iv</b>
<b>C Exemplo de produção de gráfico das diferenças de <i>NDVI</i></b>	<b>vii</b>
<b>D Artefactos de produção de figuras e resultados</b>	<b>ix</b>

# Lista de Figuras

1.1	Revisitas por coordenada da superfície terrestre do <i>Sentinel 2</i> . . . . .	3
1.2	Varrimentos da zona total de estudo da <i>Mata Nacional de Leiria</i> . . . . .	5
2.1	Visualização das duas séries temporais harmónicas. . . . .	13
2.2	<i>Workflow</i> desenvolvido na ferramenta <i>QGIS</i> . . . . .	14
2.3	Aspeto visual do cálculo do <i>NDVI</i> na ferramenta <i>QGIS</i> . . . . .	15
3.1	Representação das áreas de <i>Pedrogão Grande</i> de $G_{alvo}$ , $G_{total}$ e $G_{ref}$ . . . . .	33
3.2	Representação das áreas de <i>Pedrogão Grande</i> de $C_{alvo}$ e $C_{ref}$ . . . . .	34
3.3	Diagrama explicativo da avaliação das séries temporais. . . . .	34
3.4	Diagrama da arquitetura da solução. . . . .	37
3.5	Diagrama de classes <i>UML</i> - informação de eventos e geometria. . . . .	39
3.6	Diagrama de classes <i>UML</i> - informação de cobertura de superfície. . . . .	41
3.7	Diagrama de classes <i>UML</i> - compósitos. . . . .	43
3.8	Exemplo visual de <i>MVC</i> da área afetada de <i>Pedrogão Grande</i> . . . . .	45
3.9	Diagrama de classes <i>UML</i> - modelo. . . . .	46
4.1	Diferenças de <i>NDVI</i> para o incêndio de <i>Pedrogão Grande</i> . . . . .	51
4.2	Diferenças de <i>NDVI</i> para o incêndio da <i>Mata Nacional de Leiria</i> . . . . .	52
4.3	Médias de <i>NDVI</i> para a zona adjacente de <i>Pedrogão Grande</i> . . . . .	54
4.4	Médias de <i>NDVI</i> para a zona do incêndio de <i>Pedrogão Grande</i> . . . . .	55

4.5	NDVI para a área ardida e de referência de <i>Pedrogão Grande</i> . . . . .	56
4.6	Diferenças de NDVI para o incêndio de <i>Pedrogão Grande</i> a 20-10-2019.	57

# Lista de Tabelas

3.1	Conceitos genéricos e as propriedades da classe <i>Modelo</i> . . . . .	47
4.1	Valores observados em <i>Pedrogão Grande</i> desde 24-05-2019 . . . . .	58
A.1	Classificação CLC - Nível 1 . . . . .	i
A.2	Classificação CLC - Nível 2 . . . . .	ii
A.3	Classificação CLC - Nível 3 . . . . .	ii
D.1	Artefactos de produção de figuras e resultados . . . . .	ix

# Lista de Listagens

2.1	Inicialização da sessão no <i>GEE</i> . . . . .	19
2.2	Representação do coseno de um número utilizando <i>ee.Number</i> . . . . .	20
2.3	Obtenção do valor da representação guardada em <i>coseno</i> . . . . .	21
2.4	Representação guardada em <i>hoje</i> . . . . .	21
2.5	Variáveis <i>ontem</i> e <i>amanha</i> . . . . .	22
2.6	Obtenção do valor da representação guardada em <i>amanha</i> . . . . .	22
2.7	Representação guardada em <i>lista</i> . . . . .	23
2.8	Obtenção do valor da representação guardada em <i>lista</i> . . . . .	24
2.9	Função de mapeamento para processamento de uma data . . . . .	25
2.10	Representação guardada na variável <i>distribui</i> (truncada) . . . . .	25
2.11	Valor da representação guardada na variável <i>distribui</i> . . . . .	26
2.12	Representação guardada na variável <i>dicionario</i> . . . . .	27
2.13	Valor da representação guardada na variável <i>dicionario</i> . . . . .	27
2.14	Chaves do dicionário de metadados da variável <i>pais</i> . . . . .	28
2.15	Metadados associados ao <i>pais</i> . . . . .	28
2.16	Cálculo da área de Portugal Continental em <i>Km<sup>2</sup></i> . . . . .	29
2.17	Resultado da soma de duas bandas . . . . .	29
2.18	Resultado da redução da banda representada na variável <i>banda1</i> . . . . .	30
2.19	Resultado da redução da banda representada na variável <i>banda2</i> . . . . .	31
3.1	Troço de código de <i>zonasardidas.py</i> que obtém o histograma . . . . .	39

3.2	Troço de código que inclui o método estático <code>__gera_registo()</code> . . . . .	39
3.3	Método público <code>extrai()</code> . . . . .	40
3.4	Classe <code>Floresta</code> . . . . .	40
3.5	Cálculo das dimensões de todos os modelos de cobertura . . . . .	41
B.1	Conteúdo parcial de um ficheiro gerado pela classe <code>Modelo</code> . . . . .	iv
B.2	Constantes definidas na classe <code>Composito</code> . . . . .	vi
C.1	Função que gera um gráfico com as diferenças de <code>NDVI</code> . . . . .	vii

# 1

## Introdução

No Verão e Outono de 2017 ocorreram em Portugal incêndios florestais que tiveram um impacto muito profundo na sociedade portuguesa [2, 5]. A perda de vidas humanas e a destruição de habitats e património foram marcantes e as cicatrizes geradas pelo fenómeno perdurarão no tempo. Estes episódios de incêndio florestal têm sido objeto de investigação, científica e judicial, e poderão ter as mais diversas origens e responsáveis.

O facto da floresta mediterrânica ser particularmente sensível à ocorrência de fogos florestais e esta fragilidade ser potenciada pela alteração de padrões climáticos e por fenómenos de abandono de terrenos, ou mudanças na forma da sua exploração, que se vem agudizando desde há alguns anos contribuirá para a recorrência de fogos florestais na Península Ibérica [14].

O estudo do tempo de recuperação da vegetação após a ocorrência destes episódios é fundamental para prever e monitorizar a capacidade de regeneração das áreas ardidas [3, 14, 15]. A previsão do tempo de recuperação será essencial para uma gestão do coberto vegetal que pode atenuar o efeito dos futuros incêndios cuja erradicação total não é possível, nem desejável na sua versão benigna, por se tratar de um fenómeno natural.

Com este trabalho, pretende-se responder à seguinte questão: *Será possível estimar o tempo de recuperação da vegetação nas zonas queimadas, em função do tipo de cobertura vegetal, utilizando como referência o comportamento de uma zona geometricamente adjacente com o mesmo tipo de vegetação?*

## 1.1 Conceitos

Porque o domínio do problema é complexo, há que introduzir alguns conceitos que serão utilizados ao longo do texto e que são fundamentais para compreender a modelação e as soluções de concretização propostas.

**Deteção remota** - A exploração do espaço iniciou-se em Outubro de 1957 com o lançamento, pela União Soviética, do primeiro satélite artificial terrestre chamado *Sputnik*. Iniciada a corrida ao espaço, a tecnologia foi rapidamente progredindo dando origem a esforços da União Soviética (programas *Sputnik* e *Cosmos*) e dos Estados Unidos da América (programa *Explorer*) percursos do desenho e lançamento dos primeiros satélites meteorológicos.

É na década de 1960 que se começa a observar a terra do espaço. A perspectiva de se olhar para a atmosfera e superfície terrestres e poder, com base nessa observação, aprender sobre o nosso planeta e prever os fenómenos meteorológicos, levou a que se dotasse os satélites com instrumentos que pudessem substituir os sentidos humanos mas que fossem mais além. As primeiras abordagens começam com a inclusão de câmaras de televisão e sensores de radiação básicos que, no espaço de uma década, passaram a incluir os primeiros sensores multi-espectrais dedicados (e.g. AVHRR) [21].

Atualmente o conhecimento da Terra depende muito da informação recolhida por instrumentos complexos a bordo dos satélites terrestres. O termo deteção remota (*remote sensing*) refere-se ao processo de caracterização de uma dada área de superfície terrestre baseada na medição da radiação, absorvida e refletida, tipicamente a bordo de satélites artificiais. À data, o conceito abrange um conjunto mais diverso de veículos (aviões e aeronaves não tripuladas de baixa altitude) equipados com instrumentos análogos [28].

Os instrumentos são cada vez mais eficientes, tecnologicamente mais avançados, e a quantidade de informação multi-espectral gerada é crescente o que torna a necessidade de armazenamento, organização e exploração destes dados, por si só, um desafio. A título de exemplo, os dados multi-espectrais recolhidos pelos instrumentos *MSI-C* da missão *Sentinel-2* [8] no dia 5 de Janeiro de 2019, numa área retangular de  $3.245\text{km}^2$  (cobre, aproximadamente, as zonas que se irão estudar nos casos de estudo) têm um volume próximo de 1Gb.

**Varrimento e tempo de revisita** - A informação recolhida pelos instrumentos a bordo dos dois satélites que compõem a missão *Sentinel 2* é feita ciclicamente em faixas de 290Km de largura da superfície terrestre que se designam varrimentos.

O tempo de revisita é o período que medeia entre dois varrimentos consecutivos do mesmo ponto na superfície terrestre, nas mesmas condições operacionais, nomeadamente o mesmo ângulo de observação.

No caso da missão *Sentinel 2* o tempo de revisita é de 5 dias para um determinado ponto na superfície [8], podendo a mesma coordenada ser observada com maior frequência, porque existem varrimentos que se podem sobrepor, mas em ângulos de visão diferentes (ver Figura 1.1).

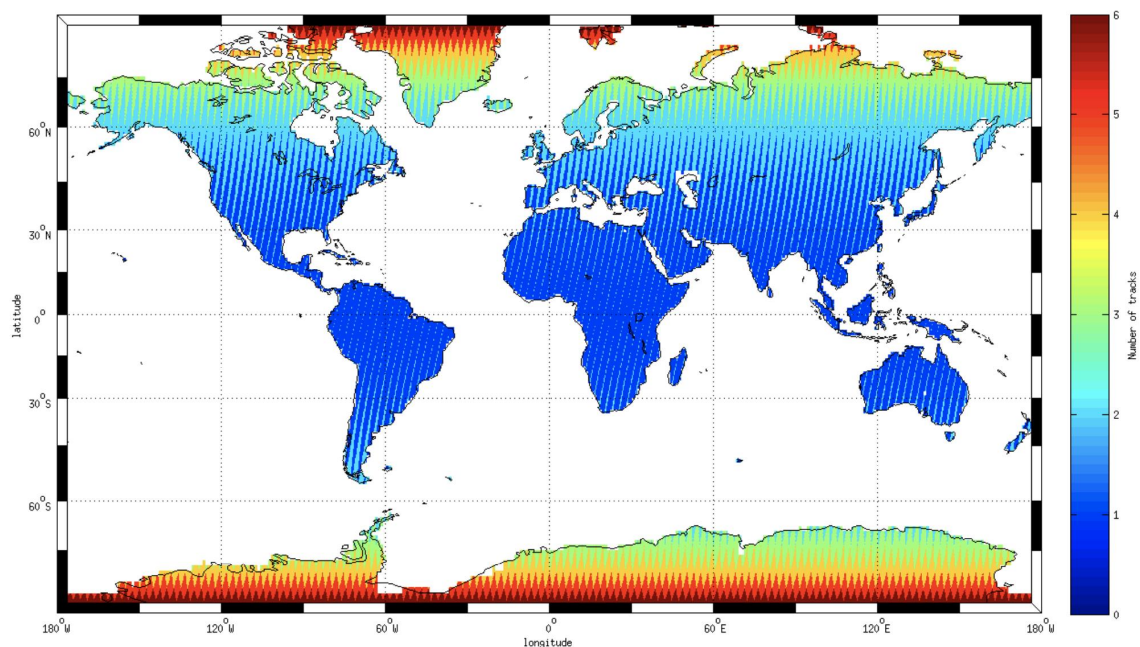


Figura 1.1: Número de revisitas por coordenada da superfície terrestre do *Sentinel 2* (Autor: Pascal Lacroix [8]).

**Medição da atividade fotossintética** - Na década de 1960, nos Estados Unidos da América, são realizados os primeiros estudos relativos ao comportamento sazonal no desenvolvimento das plantas nas grandes extensões de planície [27]. Nestes estudos iniciais, constata-se a dificuldade em determinar uma correlação direta entre a refletância e o comportamento sazonal das plantas (que atinge o seu máximo na Primavera e decai no final do Verão) em consequência da distorção ótica gerada pelos diversos ângulos de incidência do Sol nas múltiplas latitudes que são atravessadas por este extenso território.

Resulta destes estudos iniciais, após várias abordagens alternativas, um índice que é calculado como o rácio entre a diferença numérica de duas bandas espectrais (a banda próxima dos infravermelhos, *NIR*, e a banda visível do vermelho, *RED*) e a sua soma (ver Equação 1.1). A primeira utilização deste índice, designado *Normalized Difference Vegetation Index* [20] (*NDVI*), no estudo do comportamento sazonal ocorre em 1973 [26] sendo, desde essa data, o índice mais utilizado para avaliar fenómenos relacionados com a vegetação [19]:

$$NDVI = \frac{NIR - RED}{NIR + RED} \quad (1.1)$$

A utilização deste índice na aferição da atividade fotossintética [19, 29] decorre de dois factos: a clorofila tem tendência para absorver a radiação visível nos comprimentos de onda entre os 400nm e os 700nm (*RED*) quando ativa; por outro lado a organização celular das folhas das plantas reflete a radiação nos comprimentos de onda entre os 700nm e os 1100nm (*NIR*) e quantas mais folhas a planta tem, maior é a reflexão nestes comprimentos de onda.

Os valores de refletância são, por natureza, representados pela percentagem da radiação refletida (ou, de forma inversa, absorvida), traduzida em informação quantitativa entre 0 e 1. Desta forma, o índice assume valores entre -1 e 1: os valores de *NDVI* entre 0 e 1 corresponderão a zonas de vegetação, sendo mais densa e a fotossíntese mais ativa tenderá para valores mais próximos de 1; os valores negativos corresponderão a nuvens ou a zonas de neve.

Podem ocorrer problemas de avaliação nas zonas próximas de águas abertas que têm a tendência para apresentar valores de *NDVI* próximos de 0, mas num intervalo entre um valor ligeiramente negativo e um valor ligeiramente positivo.

**Compósitos de *NDVI*** - A medição do *NDVI* é muito sensível à presença de nuvens e aerossóis que provocam a ocultação e distorção ótica da luz refletida pela superfície terrestre, o que resulta na degradação da qualidade da informação recolhida. Para ultrapassar este problema será utilizado o método *Maximum Value Composite(MVC)* [17]. O método consiste em obter o valor máximo de *NDVI*, observado em cada coordenada, num intervalo de datas, como forma de aproximar o valor de *NDVI* verificado naquele período.

O intervalo de tempo será determinado pelo ciclo de recolha da informação. O satélite, ou constelação de satélites, em função do tipo de órbita que realizam à volta da Terra, terão um período fixo decorrido entre duas passagens consecutivas na mesma vertical como referido anteriormente. O ciclo de cálculo do *MVC*

será ajustado em função deste período de revisita.

Assim, um compósito *MVC* será calculado avaliando o valor máximo de *NDVI* observado em cada coordenada num intervalo de datas definido. Como a missão *Sentinel 2* tem um tempo de revisita de 5 dias (10 dias por cada satélite da constelação), o período escolhido neste trabalho para avaliar os compósitos *MVC* será de 10 dias para que, tendencialmente, cada coordenada tenha 2 valores de *NDVI* observados.

O número de varrimentos que contribuem para cada compósito é variável e depende das nuvens presentes no momento da observação o que, no limite, pode implicar que um dado compósito não tenha observações válidas e, desta forma, será descartado. Por outro lado os varrimentos contíguos podem gerar mais do que uma observação para a mesma coordenada e uma zona observada poderá não ser coberta por um só varrimento (ver Figura 1.2).

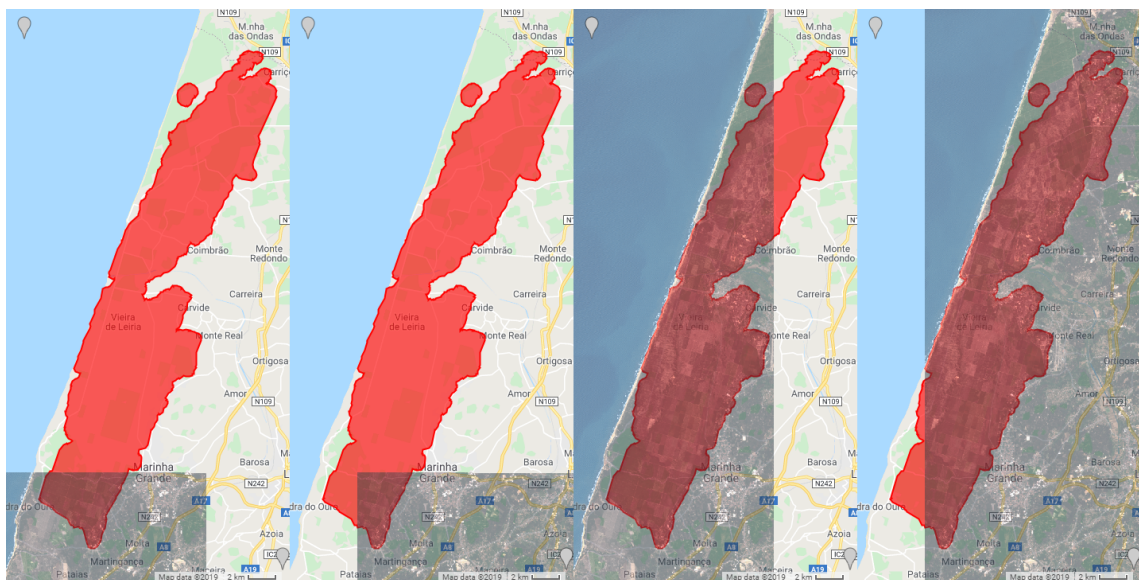


Figura 1.2: A vermelho a zona total de estudo da *Mata Nacional de Leiria*. Em sequência os varrimentos, a cinzento, correspondentes ao dia 03-05-2019. Pontos assinalados  $(-8.819979, 39.685283)$  e  $(-9.043139, 40.036838)$  WGS84.

**Informação georreferenciada** - A metodologia passará pelo tratamento de diversas fontes de informação que permitirão a modelação do processo de recuperação. Todas as fontes utilizadas terão como referencial uma projeção cartográfica da superfície terrestre. Não obstante, diferentes sistemas de informação geográfica poderão optar por referenciais diferentes. Na utilização de diferentes fontes

de dados o sistema de referência utilizado terá que ser comum ou, em alternativa, compatível ou convertível.

Outro desafio estará relacionado com a resolução espacial e a natureza da representação das diferentes fontes de informação. Uma fonte de informação georreferenciada tem duas formas básicas de representação: um mapa de pontos, naturalmente discreto, em que cada ponto equivale a um quadrado com uma dada dimensão da superfície terrestre; ou ser representada sob a forma de informação vetorial, tendencialmente contínua, em que cada polígono é projetado proporcionalmente na superfície.

Se no que se refere à informação vetorial a escala pode ser ajustada naturalmente entre duas fontes de dados, os mapas de pontos exigem mecanismos de redução de escala, por aproximação, ocorrendo uma potencial perda de informação.

## 1.2 Trabalhos relacionados

A metodologia a desenvolver tem a sua génese num conjunto de trabalhos anteriores que abordaram modelos de recuperação da vegetação após a ocorrência de incêndios florestais na Península Ibérica.

Gouveia et al. [14] estabelecem o racional de base para o cálculo do tempo de recuperação da vegetação por aplicação de um modelo de decaimento exponencial. Utiliza um conjunto de dados do instrumento *VEGETATION* da missão *SPOT*, com uma resolução de *1Km*. Foram utilizados compósitos de *NDVI*, utilizando períodos de 10 dias, com o método *MVC*, extraídos do repositório de informação multi-espectral da VITO<sup>1</sup>.

O trabalho identifica as áreas queimadas utilizando um algoritmo de agrupamento, *K-Means* [16] aplicado aos desvios dos valores de *NDVI* em relação à mediana, validando os resultados obtidos por comparação visual entre a representação gráfica dos agrupamentos obtidos e os mapas públicos oficiais que registam as áreas afetadas por incêndios florestais.

Este procedimento foi aplicado a duas zonas geográficas em Portugal, bem definidas, onde se sabia terem ocorrido fogos florestais no ano de 2003.

O tempo de recuperação é calculado utilizando um modelo de regressão mono-paramétrico que aproxima a variação do valor de *NDVI* em função do tempo *t*.

<sup>1</sup>Organização independente, sediada na Bélgica, que promove a investigação nas áreas das tecnologias limpas e desenvolvimento sustentável.

Porque se pretende determinar qual o tempo que a vegetação afetada demorará a recuperar o estado normal, é estabelecido um referencial utilizando uma função periódica  $NDVI^*(t)$  que corresponderá aos valores normais de  $NDVI$ , ao longo de um ciclo histórico anual, dado o comportamento sazonal da vegetação saudável. Por outro lado,  $NDVI(t)$  será o valor de  $NDVI$  observado num dado momento, e a medida da degradação da fotossíntese será

$$y(t) = NDVI(t) - NDVI^*(t) \quad (1.2)$$

Posteriormente, em Bastos et al. [3], utilizando dados da mesma origem, estende-se o âmbito do primeiro aplicando a metodologia a 7 fogos entre 2003 e 2005 concluindo-se pela generalização da sua aplicabilidade, fazendo uma análise de sensibilidade do método de cálculo: (i) em função dos pressupostos de arranque; (ii) à dimensão do período de análise; e (iii) à existência de lacunas nos dados.

Recentemente Gouveia et al. [15] parte de dados recolhidos pelo instrumento *MODIS* dos satélites *Aqua* e *Terra* com uma resolução de 250m. Foram utilizados compósitos de  $NDVI$ , em períodos de 16 dias, obtidos com o método *MVC*. O modelo foi aplicado com sucesso a áreas afetadas em 2003, 2005 e 2012, incluindo regiões afetadas por fogos recorrentes. Utilizando fontes de dados diferentes dos trabalhos anteriores, com maior resolução espacial e um horizonte temporal alargado, os autores validaram a aplicabilidade do modelo de recuperação.

Os trabalhos acima descritos utilizaram sempre produtos pré-calculados de  $NDVI$ . A aproximação à recuperação da vegetação utiliza uma estimativa teórica dos valores denominados como "ótimos" de  $NDVI$ , num cenário histórico no período pré-fogo. Uma abordagem alternativa é proposta por Diaz-Delgado et al. [6], e utiliza o valor de  $NDVI$  numa zona geometricamente adjacente à área afetada pelo incêndio. As diferenças são assim apuradas entre o  $NDVI$  medido nas zonas afetadas por um incêndio e os valores observados em zonas adjacentes àquelas. No referido trabalho as zonas adjacentes são determinadas de forma manual e é utilizado um modelo de regressão logarítmica para aproximar o tempo de recuperação da vegetação.

### 1.3 Objetivos

O objetivo deste trabalho é desenvolver uma metodologia para estimar o tempo de recuperação da vegetação após a ocorrência de um incêndio florestal, sem

recorrer a dados históricos relativos ao comportamento passado da vegetação. Desenvolveu-se uma metodologia, independente das fontes de dados e de ferramentas geográficas, concretizada num protótipo demonstrativo da exequibilidade da solução.

A metodologia conjuga as abordagens dos trabalhos anteriores, mas distingue-se por:

- A diferença de *NDVI* será determinada entre os valores médios observados, na mesma data, nas zonas ardida e adjacente;
- A determinação da zona adjacente será feita automaticamente com base num algoritmo proposto;
- Serão utilizados bandas multi-espectrais para o cálculo de *NDVI* em alternativa à utilização de produtos pré-calculados.

No que se refere ao cálculo do *NDVI*, terá que se calcular a diferença normalizada entre a radiação refletida próxima dos infravermelhos (*NIR*) e do vermelho (*RED*, ver a Equação 1.1).

A avaliação do *NDVI* utilizará uma fonte de dados de informação multi-espectral, adotando a técnica de cálculo de compósitos *MVC*, utilizando um intervalo de datas cíclico compatível com o tempo de varrimento do satélite.

A integração das diferentes fontes de informação terá que ser concretizada pela ferramenta, compatibilizando formatos e referenciais geográficos.

O volume de dados será, previsivelmente, muito significativo (aproximadamente 1Gb por dia avaliado nos casos de estudo), e terá impacto nas operações de seleção e processamento. Assim, na concretização ter-se-á que acautelar que:

- As operações de seleção de dados deverão ter uma sequência que permita reduzir a quantidade de dados a processar sem ter impacto no resultado final;
- Os algoritmos a conceber deverão, sempre que possível, permitir que o processamento ocorra de forma distribuída de modo a que os resultados sejam obtidos em tempo útil e não esgotem os recursos disponíveis;
- A modelação matemática do problema terá de refletir estas duas necessidades.

Como prova de conceito estima-se o tempo de recuperação para duas zonas afetadas pelos incêndios de 2017: *Pedrogão Grande* e a *Mata Nacional de Leiria*.

## 1.4 Organização do documento

Este documento está organizado em quatro partes, excluída a introdução. No Capítulo 2, *Estudo exploratório*, faz-se a narrativa de todo o trabalho exploratório das alternativas de investigação que resultaram na seleção das fontes de informação e do ecossistema de ferramentas de desenvolvimento. É também nesse capítulo que é feita uma introdução ao *Google Earth Engine* como plataforma de desenvolvimento.

No Capítulo 3, *Modelação e concretização*, na Seção 3.1, *Modelação*, procede-se ao desenvolvimento da metodologia, tentando dissociá-la tanto quanto possível da concretização, tendo por objetivo a sua aplicação futura noutros cenários de desenvolvimento. A materialização da modelação é abordada na Seção 3.2, *Concretização*, onde se faz a descrição da arquitetura aplicacional e da concretização da mesma. É também nesta secção que se abordam alguns problemas relacionados com a natureza da informação georreferenciada e da potencial existência de lacunas na informação multi-espectral em função de fenómenos ambientais.

Uma vez modelada e concretizada a metodologia, é no Capítulo 4, *Casos de estudo*, que se descrevem dois casos de uso da sua aplicação. Para suportar a utilidade da metodologia, é feita uma análise de sensibilidade comparando os resultados obtidos com uma abordagem alternativa, validando os pressupostos da modelação. No final da secção atualizam-se as séries temporais de *Pedrogão Grande* à data da entrega do documento de modo a validar e ajustar a previsão inicial.

Finalmente, no Capítulo 5, *Conclusões*, faz-se o resumo do trabalho realizado e apresentam-se caminhos futuros para o resultado desta dissertação.

Todos os resultados deste trabalho são reproduzíveis. No Apêndice D encontra-se a indicação de todos os artefactos de código usados no trabalho para produzir figuras e resultados. À exceção dos artefactos que dependem do *GEE*, e que exigem a configuração da máquina que executar os *scripts* e alguma familiaridade com o *GEE*, todos os *scripts* relacionados com o modelo de cálculo poderão ser executados num ambiente de *Python*.

# 2

## Estudo exploratório

Neste capítulo descreve-se o estudo exploratório efetuado para identificar as fontes de dados e ferramentas adequadas à concretização da metodologia e as diferentes abordagens matemáticas para a obtenção de um estimador para o tempo de recuperação da vegetação.

É feita, de forma narrativa, a descrição das experiências exploratórias realizadas e de como estas foram moldando a configuração final da metodologia proposta. Estas experiências determinaram: (i) as fontes de dados a utilizar; (ii) a adoção do modelo matemático de previsão e (iii) a ferramenta a utilizar na concretização da metodologia.

### 2.1 Fonte de dados de satélite

Foi explorada a possibilidade de recorrer a diversas fontes de dados, tendo como objetivo a utilização de dados multi-espectrais de alta resolução e a caracterização adequada das zonas geográficas ardidadas quanto à delimitação e tipo de cobertura do solo pré-existente.

A informação multi-espectral, recolhida utilizando os instrumentos a bordo de satélites, é disponibilizada não só na sua forma básica (e.g. bandas multi-espectrais), mas também sob a forma de produtos calculados, entre os quais o *NDVI*. Os produtos calculados são séries de valores obtidos por aplicação de algoritmos de

cálculo a informação multi-espectral em bruto. Estes dados são mantidos em repositórios, acessíveis aos utilizadores utilizando formatos padrão, não exigindo desta forma o conhecimento nem a execução dos algoritmos de cálculo.

A disponibilização de produtos calculados ocorre, sobretudo, porque as necessidades de armazenamento e processamento de dados para o cálculo são demasiado exigentes e não estão disponíveis para os recursos da generalidade dos utilizadores. Por exemplo, 1Gb de armazenamento em disco por cada dia avaliado para uma área retangular aproximada de apenas  $3.245\text{km}^2$  e aproximadamente 5 minutos (Intel Core i5-6200U com 8Gb RAM e SSD) para cálculo de uma média de *NDVI*.

Uma segunda motivação para a disponibilização de produtos calculados relaciona-se com a diversidade dos formatos de dados e algumas especificidades dos instrumentos de deteção remota que exigem um forte acoplamento entre os processos de cálculo e as fontes de dados.

A informação multi-espectral é muito sensível à presença de aerossóis e nuvens. Os valores de *NDVI* utilizados são, normalmente, agregados dos valores observados num determinado intervalo de datas. Estas séries de valores designam-se compósitos de *NDVI*.

Numa primeira abordagem estudou-se a possibilidade de utilizar os dados de maior resolução recorrendo a compósitos de *NDVI* já calculados. A missão *Sentinel 3* inclui um instrumento virtual designado *Synergy* [9] que publica produtos calculados com o objetivo de manter a compatibilidade e continuidade com a missão *SPOT* utilizada, nomeadamente, por Gouveia et al. [14]. Não obstante, na data em que se iniciou este trabalho, estes produtos apenas estavam disponíveis numa versão de teste e não cobriam o horizonte temporal pretendido.

Em alternativa estudou-se a possibilidade de utilizar os compósitos *MVC* a 10 dias, com uma resolução de 300m, da missão *Proba-V*, novamente numa perspetiva de continuidade dos trabalhos realizados anteriormente. Não obstante, estes dados não cumpriam com o objetivo de utilizar informação multi-espectral com uma resolução superior às fontes de dados utilizadas em trabalhos anteriores [3, 14, 15]. Acresce que a extração dos dados teria que ser feita manualmente utilizando o repositório *VITO*, o que dificultaria a conceção de um protótipo autónomo no que se refere à integração das fontes de dados.

A opção acabou por recair em dados de deteção remota, na sua forma base, como informação multi-espectral recolhida pelos instrumentos da missão *Sentinel 2* [8]. Esta escolha determinou que a metodologia proposta incluía o processo de cálculo

do *NDVI* e dos compósitos *MVC*. Convirá sublinhar que esta escolha foi também condicionada pela escolha da ferramenta de implementação como se descreve adiante.

## 2.2 Modelo matemático

Tentou-se um conjunto de abordagens alternativas ao modelo de cálculo proposto por Gouveia et al. [14] especialmente a partir do momento em que se decidiu a utilização dos dados da missão *Sentinel 2* e da ferramenta (ver Seção 2.3). As abordagens passaram por aproximar as séries temporais utilizando variantes de regressão linear.

A adoção de dados com um maior nível de detalhe e a utilização do *NDVI* observado numa zona geometricamente adjacente à afetada pelo incêndio, e não de um modelo teórico como referência para o cálculo do tempo de recuperação, dá origem à ideia de explorar o comportamento das duas séries temporais.

Recorde-se que nos trabalhos anteriores, o referencial é uma função periódica que corresponderá aos valores normais de *NDVI*, ao longo de um ciclo histórico anual do comportamento sazonal da vegetação saudável.

Assim as duas séries temporais de *NDVI* seriam: (i) uma referente à zona alvo afetada pelo incêndio e (ii) outra observada na zona geometricamente adjacente, de referência.

Os estimadores seriam obtidos utilizando um método de regressão e a estimativa do tempo de recuperação corresponderia à interceção entre os dois estimadores. Dito de outra forma, o ponto no tempo onde as duas séries temporais estimadas assumissem o mesmo valor de *NDVI* corresponderia ao tempo aproximado de recuperação da vegetação. A variável tempo teria a sua origem na primeira imagem de satélite viável após a ocorrência do incêndio.

A exploração deste modelo, embora com coeficiente de determinação superiores a 0.8 (ver Secção 4.3), seria muito enviesado pela ocorrência de *outliers* e anomalias resultantes de condições ambientais. Por outro lado, os melhores resultados eram obtidos utilizando uma regressão harmónica (ver Equação 4.1) cujo ponto de interceção obrigaria à utilização de um método iterativo de aproximação, por impossibilidade de resolução algébrica da igualdade entre os dois estimadores (ver Figura 2.1).

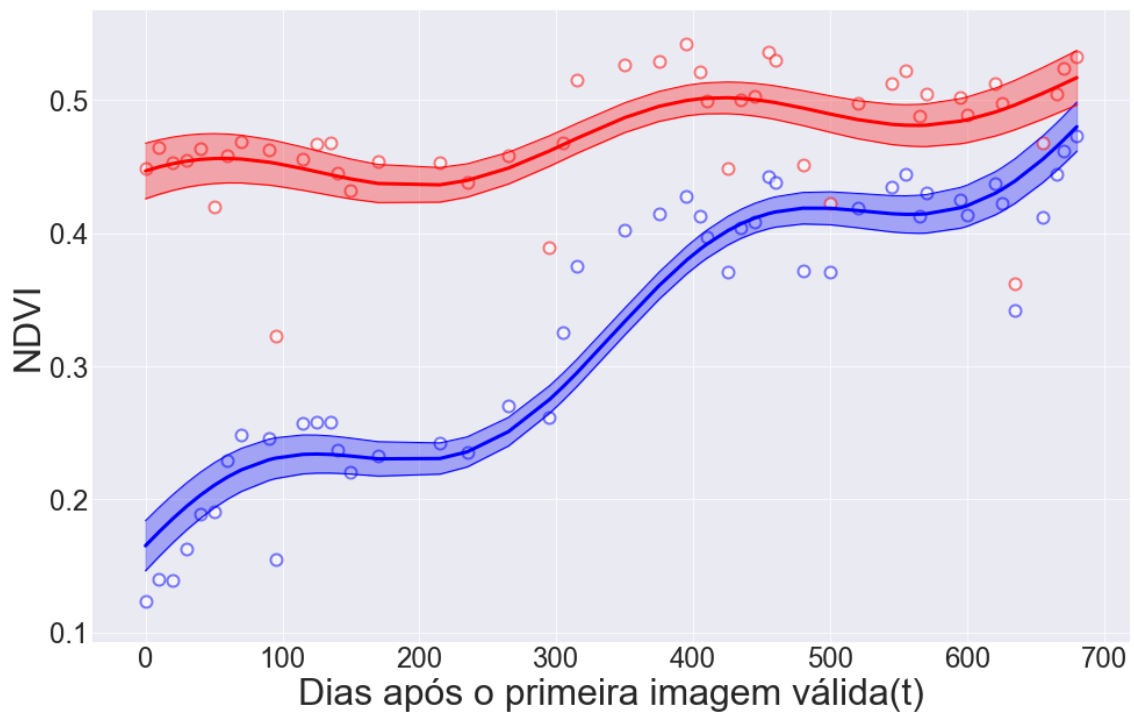


Figura 2.1: Visualização das duas séries temporais harmônicas alvo (azul) e de referência (vermelho)

Não obstante, verificou-se que existe coincidência no tempo na ocorrência de *outliers* em ambas as séries (entendidos como pontos fora dos intervalos de confiança). Isto parece indicar que, sendo as medições realizadas pelos mesmos instrumentos em condições ambientais semelhantes, *caeteris paribus*, pode considerar-se estes desvios são normais no que se refere à aferição do valor médio de *NDVI*. Optou-se assim por adotar o método de cálculo proposto por Gouveia et al. [14], aplicando o modelo de decaimento exponencial à diferença algébrica entre as duas séries temporais, relativa ao valor máximo observado, o que será desenvolvido na Subseção 3.1.2.

## 2.3 Ferramenta

Na abordagem ao trabalho foram exploradas duas ferramentas: (i) um sistema de informação geográfica, na sua forma tradicional, o *QGIS* [24] acompanhado da biblioteca *Geospatial Data Abstraction Library (GDAL)* [10]; e (ii) o *Google Earth Engine (GEE)* [13].

A experiência com a utilização do *QGIS* foi feita com dados da missão *Proba-V*

e recorreu-se ao repositório *VITO* para obter um compósito de *NDVI* de forma manual.

Este procedimento revelou imediatamente dois problemas na utilização desta fonte de informação: (i) o procedimento foi manual e, embora não tenha sido explorada a possibilidade de obter os dados automaticamente, passaria sempre por um procedimento segregado da restante solução para a concretização da metodologia; e (ii) o volume de dados envolvido foi de cerca de 1GB o que dificultaria ainda mais a integração dos dados de deteção remota, não obstante a ligação às restantes fontes de informação fosse mais simples porque poderia ocorrer apenas pontualmente.

Feita a integração de dados na ferramenta procedeu-se à delimitação da informação às áreas de interesse, automatizando o processo utilizando as funcionalidades do *QGIS* e da *GDAL*, integradas sob a forma de um *workflow* (ver Figura 2.2) que depende de um valor único que é a localização do incêndio.

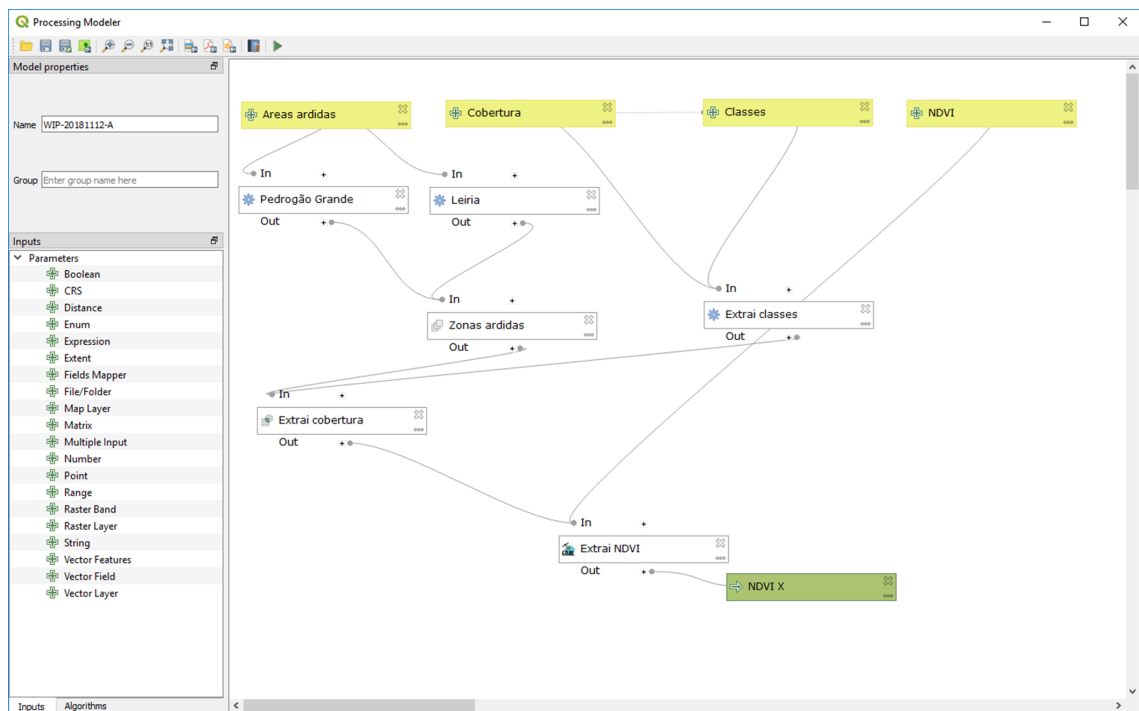


Figura 2.2: *Workflow* desenvolvido na ferramenta *QGIS*

O tempo de processamento do cálculo dos valores médios de *NDVI* e as operações geométricas de delimitação das áreas de interesse era longo (aproximadamente 5 minutos, num portátil Intel Core i5-6200U com 8Gb RAM e SSD) para um único compósito *MVC* (ver Figura 2.3) o que foi indicativo que o desenvolvimento de todo o processo seria demorado e complexo (obrigaria a um processo

*batch* de integração de dados).

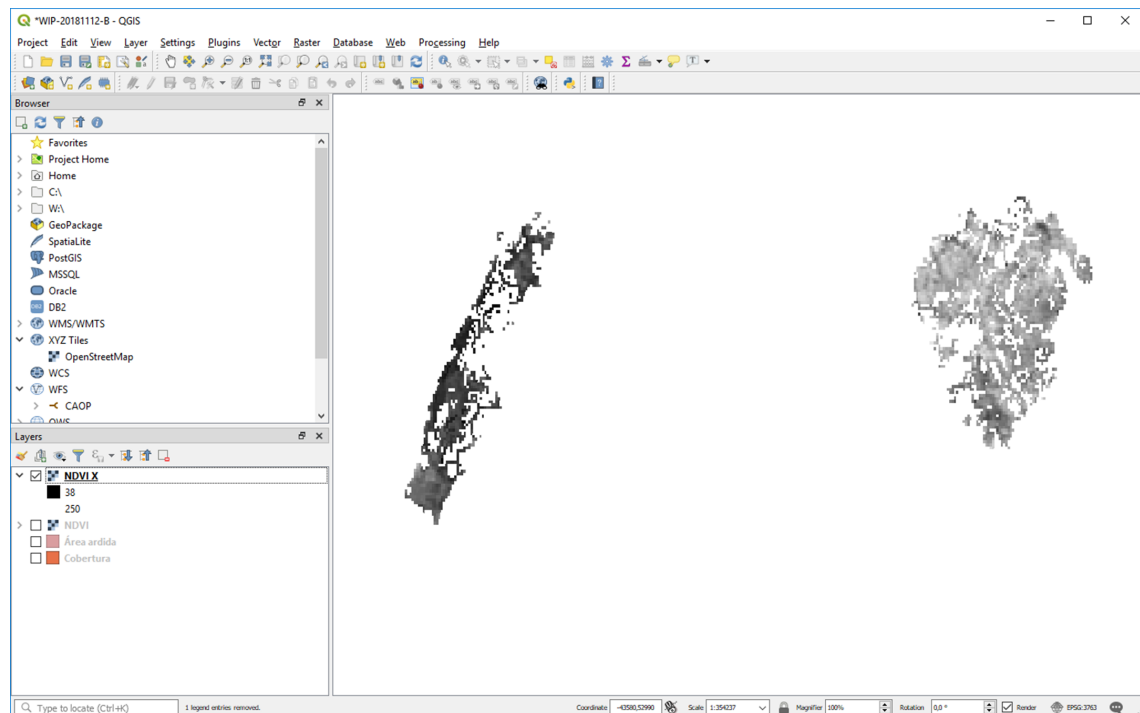


Figura 2.3: Aspeto do resultado visual da delimitação e cálculo do *NDVI* na ferramenta *QGIS*

Ainda se explorou a possibilidade de utilizar a *GDAL*, para manipulação da geometria e dos dados multi-espectrais, utilizando o *Python* para integrar o processo. As dificuldades de processamento com os recursos limitados de um computador pessoal eram evidentes e as funcionalidades implementadas pela *GDAL* têm uma curva de aprendizagem íngreme o que iria condicionar muito o tempo dedicado à investigação, conceção e concretização da metodologia.

A leitura de um trabalho designado "*Experiências na deteção de incêndios com o Google Earth Engine*" de autoria de Eduardo Fernandes, com afiliação no Departamento de Informática da Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, foi precursora da avaliação do *GEE*.

A exploração da ferramenta tornou clara a vantagem desta relativamente às outras soluções avaliadas, nomeadamente porque:

- As diversidade das fontes de dados integradas no *GEE* e as funcionalidades disponíveis ultrapassam as abordagens e ferramentas descritas anteriormente em facilidade de utilização e velocidade de acesso;

- Entre as muitas fontes de dados, é mantido o repositório com dados multi-espectrais de alta definição da missão *Sentinel 2*;
- Não obriga ao desenvolvimento de um processo *batch* de atualização de dados porque o *GEE* assegura a ingestão incremental das diferentes fontes de dados;
- O processamento ocorre de forma distribuída utilizando a infraestrutura da *Google*.

Foi este o racional que determinou que, na elaboração deste trabalho, a ferramenta de desenvolvimento, e repositório de dados, fosse o *GEE*.

A próxima secção descreve o *GEE* e os conceitos estruturantes que são necessários para compreender os detalhes da arquitetura da solução para a concretização da metodologia.

## 2.4 Google Earth Engine

O *GEE* é uma plataforma que suporta a manipulação de dados georreferenciados utilizando infraestruturas de armazenamento e processamento de dados da *Google* [13].

Entre outras valias, o *GEE* ultrapassa a necessidade de aguardar pelos ciclos de cálculo e disponibilidade de produtos calculados e a dificuldade de apreensão e apropriação dos formatos de dados utilizados pelas diferentes fontes de informação. Utiliza um modelo unificado de representação da informação georreferenciada, independentemente da sua origem, permitindo a utilização dos dados multi-espectrais em bruto, bem como a aplicação de algoritmos de cálculo, sem a necessidade de conhecer a organização original dos dados e beneficiando do poder de processamento da infraestrutura da *Google*.

De uma forma sucinta, a plataforma *GEE* é composta, à data, por um catálogo de dados, que agrega cerca de 600 conjuntos, de 50 fontes distintas obtidas a partir de cerca de 30 satélites e constelações. Estes dados são processados e catalogados pela *Google* em função dos seus ciclos de refrescamento e inclui, entre outras, duas das fontes de dados que irão ser utilizadas na concretização da metodologia proposta: (i) o *Corine Land Cover (CLC)* e (ii) os dados multi-espectrais da missão *Sentinel 2*. O catálogo, segundo estimativa da *Google*, tem um volume de  $20PB \equiv 20 \times 2^{50} \text{bytes}$  [11].

A API exposta suporta um vasto conjunto de operações sobre os dados do repositório (ou outros conjuntos de dados na área privada do utilizador) e:

- Tipos de dados que estruturam a informação sobre a forma de mapas de pontos ou vetores, de forma singular ou organizada em coleções;
- Operações que permitem a seleção de regiões baseadas em operações de conjuntos;
- Tipos canónicos de coleções (e.g. dicionários, listas, etc...) ou primitivos (e.g. números, cadeias alfanuméricas, etc...);
- Operações de filtragem e ordenação de dados aplicáveis a coleções de dados;
- Um tipo que estrutura bandas espectrais organizadas sob a forma de mapas de pontos multi-dimensionais. Estes podem ser manipulados por operações matemáticas simples (e.g. multiplicar uma banda por uma constante) ou dedicadas (e.g. diferença normalizada); e
- O poder computacional disponibilizado pela *Google*, utilizando o paradigma de distribuição de processamento *map / reduce*.

A linguagem utilizada na interface de desenvolvimento online, designada *Code Editor*, é o *Javascript*. O ambiente de *Javascript* interage com a plataforma do *GEE*, remota, e com bibliotecas locais que utilizam o *browser* como hospedeiro. Um *script* sabe distinguir entre os dois planos (cliente e servidor) mas há um conjunto de regras que têm de ser seguidas para que a interação entre os planos ocorra de forma harmoniosa [12]. Existem duas outras bibliotecas que permitem a interação com o *GEE*, fora do ambiente do *browser*, que utilizam o *Python* ou o *Node JS* como *framework* de desenvolvimento.

A *Google*, não obstante, não se compromete com uma garantia de *Service Level Agreement* quanto à disponibilidade ou comportamento do *GEE*, o que se revelou um desafio adicional à utilização desta ferramenta na elaboração deste trabalho.

### 2.4.1 Aprendizagem e exploração

Todo o trabalho de aprendizagem da plataforma, exploração das diversas fontes de dados e automação de tarefas foi realizada durante os primeiros meses utilizando o *Code Editor* e, conseqüentemente, o *Javascript*.

Várias dificuldades tornaram-se rapidamente evidentes na utilização da ferramenta, nomeadamente duas: (i) a construção dos algoritmos exige perceber que o mesmo *script* encapsula um fluxo de controle que é distribuído entre o processamento local e um ambiente remoto. Existe uma segregação estrita entre os dois ambientes, paradigmas de programação diferentes e um ambiente de desenvolvimento que isola o utilizador desta complexidade; (ii) a extração de dados do GEE é, no mínimo, desafiante quando se utiliza um *browser*.

Na realidade, o *Code Editor* tem utilidade na interação com o *Google Maps* para a visualização de informação ilustrativa e prototipagem rápida de mecanismos de interrogação e exploração do repositório.

Como facilitador do trabalho, foi desenvolvida uma biblioteca de artefactos que simplificou a reutilização de código entre tarefas e contribuiu muito para a compreensão da plataforma (ver Apêndice D. É esta primeira abordagem que é precursora da biblioteca em *Python* resultante deste trabalho.

Com o decorrer da exploração tornou-se evidente que na utilização do *Code Editor*: (i) a execução de processos longos utilizando o *browser* não consegue ser interativa (não é possível expor informação de progresso do processo ao utilizador); (ii) gera alertas do *browser* que não distingue se a execução demorada de um processo se trata de uma anomalia de desenvolvimento de *Javascript* ou um comportamento normal; e (iii) finalmente torna a extração de dados desafiante, embora a plataforma ofereça uma funcionalidade *batch* que se adequa a processos longos de extração de dados utilizando mecanismos de armazenamento remoto (e.g. *Google Drive* ou *Google Cloud*).

Este último aspeto tornou-se fundamental na definição da ferramenta. Concluiu-se que sem um esforço de desenvolvimento relevante, a exploração, armazenamento e realização de cálculos teria que ser sempre repetido na totalidade, o que seria ineficiente.

Finalmente, a concorrência pelos recursos do GEE, com os restantes utilizadores, refletiu-se na experiência de duas formas:

- A duração das tarefas realizadas utilizando o *Code Editor* passou a ser errática; e
- A ocorrência crescente de erros de ultrapassagem das quotas de memória e de capacidade de processamento, em cenários cada vez mais exigentes.

Assumiu-se então a decisão de reorientar a estratégia de desenvolvimentos adotando os seguintes pressupostos:

- Utilização do *Python* como plataforma de desenvolvimento;
- Reimplementação da biblioteca, entretanto construída em *Javascript*, procedendo à sua tradução para *Python*;
- Utilização do *Code Editor* apenas para tarefas de visualização e interação com o *Google Maps*;
- Separação do processo de integração com o *GEE* do modelo matemático de previsão; e
- Criação funcionalidades de interação com o utilizador.

As secções seguintes têm como objetivo a contextualização das funcionalidades mais relevantes do *GEE* utilizadas neste trabalho.

## 2.4.2 Conceitos base

O *GEE* é, antes de tudo, uma plataforma de computação distribuída que explora os paradigmas de programação mais adequados ao tratamento de grandes volumes de dados georreferenciados. Destacam-se a perspetiva funcional (não obstante a organização da API sob a forma de um *object model*) e a distribuição de processamento utilizando o paradigma *map / reduce*.

Tanto o *Code Editor*, baseado no *browser* e fazendo uso do *Javascript*, como a biblioteca de suporte em *Python*, o *GEE* tenta esconder a complexidade da plataforma simplificando a sua utilização.

Não se pretende nas próximas secções ser exaustivo quanto à descrição do *GEE* mas antes descrever as funcionalidades genéricas e os objetos que foram utilizados na concretização da arquitetura de *software* para suportar este trabalho.

### 2.4.2.1 Inicialização do *GEE*

Qualquer aplicação que utilize o *GEE* terá que começar por proceder à inicialização da plataforma que se traduz na autenticação, utilizando *OAuth*, e no estabelecimento de uma sessão no *GEE*.

---

```
import ee
```

```
ee.Initialize()
```

---

Listagem 2.1: Inicialização da sessão no *GEE*

Uma vez iniciada a plataforma pode-se começar a interação com o *GEE*.

#### 2.4.2.2 Valores numéricos - *ee.Number*

O objeto *ee.Number* encapsula um valor numérico e expõe um conjunto de funções (aritméticas, trigonométricas, etc...) que utilizam o seu valor como parâmetro.

Como exemplo, considere-se a codificação da constante PI. Pode-se representar o valor de PI, atribuí-lo a `numero` e representar o cálculo do seu coseno em `coseno`.

Sublinhe-se a natureza distribuída e simbólica das variáveis `numero` e `coseno` que no processo local em *Python* correspondem apenas a representações (i.e. *proxies*) que podem ser utilizadas na construção de uma operação complexa de *GEE* ou na obtenção do valor imediato como resultado da execução da representação no *GEE*.

---

```
import math

numero=ee.Number(math.pi)
coseno=numero.cos()
print(coseno)
```

---

```
ee.Number({
  "type": "Invocation",
  "arguments": {
    "input": 3.141592653589793
  },
  "functionName": "Number.cos"
})
```

---

Listagem 2.2: Representação do coseno de um número utilizando *ee.Number*

Utilizando a Listagem 2.2, como exemplo, nos próximos troços de código que enviem informação para a consola, por utilização de `print()`, será ilustrado o resultado separado por uma linha simples como neste caso.

Esta avaliação de `coseno` não implica qualquer chamada aos servidores do *GEE*. A variável `coseno` é apenas a representação de uma chamada à função `cos()` no *GEE* e `numero` é o argumento único da função.

A utilização de `getInfo()`, que é exposto por todos os objectos de *GEE*, envia a representação guardada em `coseno` para o servidor, onde será avaliada, e

devolvido o valor de apurado representado como um bloco de *JSON* (ver Listagem 2.3).

Observe-se que `resultado` é já um objeto primitivo de *Python*, e é o resultado da avaliação da expressão representada pela variável `coseno`, isto é  $\cos(\pi)$ .

---

```
resultado=coseno.getInfo()
print(resultado)
```

---

```
-1.0
```

---

Listagem 2.3: Obtenção do valor da representação guardada em `coseno`

Em resumo, ao nível do ambiente de execução local apenas são formadas as representações de uma operação na plataforma *GEE*, estas só são executadas remotamente com a utilização de `getInfo()` que devolve o resultado da operação implícita.

### 2.4.2.3 Manipulação de datas - *ee.Date*

O objeto *ee.Date* representa uma data de calendário e expõe um conjunto de funções que atuam sobre ela. Uma das funções, `fromYMD()`, é na realidade um dos seus construtores e codifica uma data a partir dos valores numéricos do ano, mês e dia.

O exemplo na Listagem 2.4 reforça a natureza simbólica dos objetos de *GEE* no ambiente local de *Python*.

---

```
hoje=ee.Date.fromYMD(2019,5,4)
print(hoje)
```

---

```
ee.Date({
  "type": "Invocation",
  "arguments": {
    "year": 2019,
    "month": 5,
    "day": 4
  },
  "functionName": "Date.fromYMD"
})
```

---

Listagem 2.4: Representação guardada em `hoje`

Para clarificar o funcionamento da criação destes *proxies*, considere-se mais duas representações do dia anterior e do dia posterior a `hoje`. Dada `hoje`, represente-se outras duas variáveis, utilizando `advance()`, exposta pelo objecto *ee.Date*,

que opera sobre as partes da data (e.g. ano, mês, dia, etc...), avançando (ou recuando) na data um dado número de unidades. Pode-se assim representar `ontem` e `amanha`, respetivamente.

---

```
ontem=hoje.advance(-1,"day")
amanha=hoje.advance(1,"day")
print(amanha)
```

---

```
ee.Date({
  "type": "Invocation",
  "arguments": {
    "date": {
      "type": "Invocation",
      "arguments": {
        "year": 2019,
        "month": 5,
        "day": 4
      },
    },
    "functionName": "Date.fromYMD"
  },
  "delta": 1,
  "unit": "day"
},
"functionName": "Date.advance"
})
```

---

#### Listagem 2.5: Variáveis `ontem` e `amanha`

Observando a representação de `amanha` na Listagem 2.5 é muito claro que `hoje` passou a ser o primeiro argumento da representação da função que será avaliada no *GEE*.

Avaliando a representação implícita na variável `amanha` (ver Listagem 2.6), obtém-se então o resultado.

---

```
print(amanha.getInfo())
```

---

```
{ 'type': 'Date', 'value': 1557014400000 }
```

---

#### Listagem 2.6: Obtenção do valor da representação guardada em `amanha`

O valor de um objecto *ee.Date* não é diretamente traduzido para um tipo primitivo de *Python*. Na Listagem 2.6, pode-se observar que a avaliação de `amanha` no *GEE* devolve um dicionário com dois campos: o tipo de dados e o seu valor sob forma numérica. Esta forma numérica corresponde a um *timestamp Unix* (milissegundos decorridos desde 1 de Janeiro de 1970).

### 2.4.2.4 A representação e operação sobre listas - *ee.List*

A representação de uma lista no *GEE* é suportada pelo objeto *ee.List*. Continuando o exemplo anterior, considere-se uma lista composta pelas representações de ontem, hoje e amanhã.

---

```
lista = ee.List([ontem, hoje, amanhã])
print(lista)
```

---

```
ee.List([
  {
    "type": "Invocation",
    "arguments": {
      "date": {
        "type": "Invocation",
        "arguments": {
          "year": 2019,
          "month": 5,
          "day": 4
        },
        "functionName": "Date.fromYMD"
      },
      "delta": -1,
      "unit": "day"
    },
    "functionName": "Date.advance"
  },
  {
    "type": "Invocation",
    "arguments": {
      "year": 2019,
      "month": 5,
      "day": 4
    },
    "functionName": "Date.fromYMD"
  },
  {
    "type": "Invocation",
    "arguments": {
      "date": {
        "type": "Invocation",
        "arguments": {
          "year": 2019,
          "month": 5,
          "day": 4
        }
      }
    }
  }
])
```

```

    },
    "functionName": "Date.fromYMD"
  },
  "delta": 1,
  "unit": "day"
},
"functionName": "Date.advance"
}
])

```

---

Listagem 2.7: Representação guardada em `lista`

Observando o valor da variável `lista` na Listagem 2.7 nota-se mais uma vez a natureza simbólica do objeto e a crescente complexidade da representação. Executando de forma explícita a operação no *GEE* na Listagem 2.8.

```
print(lista.getInfo())
```

```
[{'type': 'Date', 'value': 1556841600000}, {'type': 'Date', 'value': 1556928000000}, {'type': 'Date', 'value': 1557014400000}]
```

---

Listagem 2.8: Obtenção do valor da representação guardada em `lista`

Quando avaliado o objecto `lista`, que contém instâncias de *ee.Date*, o *GEE* devolve não valores primitivos mas três dicionários correspondente aos valores de `ontem`, `hoje` e `amanha` avaliados ao nível do serviço.

#### 2.4.2.5 Distribuição de processamento - *map*

O *GEE* é sobretudo uma plataforma de processamento distribuído que utiliza o paradigma *map / reduce* para paralelizar e agregar tarefas. Considere-se assim a necessidade de distribuição do processamento da variável `lista`. Esta contém a representação de três datas mas poderia conter estruturas de dados consideravelmente mais complexas (por exemplo, bandas multi-espectrais) que exigissem paralelizar o cálculo para gerir os recursos disponíveis e tornar o processo eficiente.

No *GEE* um objecto *ee.List* (bem como todos os objectos iteráveis) expõe `map()` que tem como argumento uma função. Existem regras para a construção desta função:

- Tem como argumento único o elemento da lista a processar;
- Devolve o resultado do processamento;

- Tem que ter no seu contexto de execução todos os dados sobre os quais opera, não podendo ser utilizado qualquer tipo de referência global. Dito de outra forma, uma função que sirva de argumento para `map()` tem que ser completamente isolada do restante ambiente de execução.

Considere-se assim uma `mapa()` que será utilizada para este efeito.

---

```
def mapa(data):
    return ee.Date(data).get("day")
```

---

Listagem 2.9: Função de mapeamento para processamento de uma data

Observe-se assim na Listagem 2.9:

- O parâmetro `data` será o elemento da `lista` a tratar a cada iteração;
- Na referência perde-se o tipo de dados de `data`. É feita a conversão de tipo para `ee.Date`;
- Para cada data, obtém-se o dia do mês utilizando `get()`; e
- A representação numérica do dia do mês é devolvida.

---

```
distribui=lista.map(mapa)
print(distribui)
```

---

```
ee.List({
  "type": "Invocation",
  "arguments": {
    "list": [
      {
        (...)
      }
    ],
    "baseAlgorithm": {
      "type": "Function",
      "argumentNames": [
        "_MAPPING_VAR_0_0"
      ],
      "body": {
        "type": "Invocation",
        "arguments": {
          "date": {
            "type": "Invocation",
```

```

    "arguments": {
      "value": {
        "type": "ArgumentRef",
        "value": "_MAPPING_VAR_0_0"
      }
    },
    "functionName": "Date"
  },
  "unit": "day"
},
"functionName": "Date.get"
}
}
},
"functionName": "List.map"
})

```

---

Listagem 2.10: Representação guardada na variável `distribui` (truncada)

Note-se a complexidade da representação da operação na Listagem 2.10. Invocado o *GEE*, é devolvida uma lista com os dias do mês de cada uma das datas contidas em `lista`, como se pode observar na Listagem 2.11.

É evidenciada também outra característica do processo de avaliação de uma operação distribuída utilizando `map()`. O tipo do resultado pode ser diferente do tipo do parâmetro da função, isto é, enquanto que o tipo de dados da `lista` são representações de datas, o resultado da operação é uma lista de números em consequência do resultado de `mapa()`.

---

```
print(distribui.getInfo())
```

---

```
[3, 4, 5]
```

---

Listagem 2.11: Valor da representação guardada na variável `distribui`

#### 2.4.2.6 Representação de dicionários - *ee.Dictionary*

A representação de um dicionário é encapsulada num objecto *ee.Dictionary*. Este expõe um conjunto de funções que exploram e manipulam o conteúdo do dicionário e alguns auxiliares que permitem conversões de valores contidos no dicionário em tipos de objectos do *GEE*.

Considere-se a representação de um dicionário simples na Listagem 2.12.

---

```
dicionario = ee.Dictionary({"um": 1, "dois": 2})
print(dicionario)
```

---

```
ee.Dictionary({
  "type": "Dictionary",
  "value": {
    "um": 1,
    "dois": 2
  }
})
```

---

Listagem 2.12: Representação guardada na variável `dicionario`

Executando `getInfo()` da representação, o *GEE* é devolvido o resultado (ver Listagem 2.13).

---

```
print(dicionario.getInfo())
```

---

```
{ 'dois': 2, 'um': 1 }
```

---

Listagem 2.13: Valor da representação guardada na variável `dicionario`

Tanto a representação do dicionário, que pode ser utilizada na composição de um pedido complexo de *GEE*, como a obtenção do valor representado por invocação do *GEE* são, depois dos exemplos anteriores, auto-explicativos. Note-se que, tratando-se de um tipo iterável, expõe `map()` que permite a distribuição de processamento dos seus elementos.

#### 2.4.2.7 Representação vetorial e metadados - *ee.Geometry*, *ee.Feature* e *ee.FeatureCollection*

Um conjunto de primitivas vetoriais, é representável num objeto *ee.Geometry*. Este expõe funções que permitem efetuar cálculos e transformações sobre a geometria. Estas funções podem operar apenas sobre a geometria (e.g. `centroid()` para calcular o centróide) mas também para realizar operações de conjunto entre uma ou mais geometrias (e.g. `intersection()` para obter a forma que resulta da interceção entre duas geometrias).

A representação de informação georreferenciada, na forma vetorial, é feita no *GEE* utilizando os objetos *ee.Feature* e estes são agrupados utilizando o objeto *ee.FeatureCollection*.

Um objecto *ee.Feature* é construído com dois parâmetros: (i) uma geometria que pode ser representada por um objeto *ee.Geometry* ou *None*; e (ii) um *ee.Dictionary* que representa os metadados associados à geometria.

Tipicamente os objetos *ee.Feature* são agrupados numa coleção que pode ter existência no âmbito de um dado *script* mas é, sobretudo, a forma do *GEE* manter informação vetorial no seu repositório.

Por exemplo, a coleção na Listagem 2.14 representa a descrição vetorial dos diversos países publicada pelo *Departamento de Estado dos Estados Unidos da América*, mantida numa das bases de dados do repositório do *GEE*. A função `first()` obtém o primeiro elemento da coleção, `propertyNames()` resulta numa lista dos nomes das chaves no dicionário de propriedades associado a cada geometria.

---

```
territorios=ee.FeatureCollection("USDOS/LSIB_SIMPLE/2017")
pais=territorios.first()
print(pais.propertyNames().getInfo())
```

---

```
['wld_rgn', 'country_na', 'abbreviati', 'system:index', 'country_co']
```

---

Listagem 2.14: Chaves do dicionário de metadados da variável `pais`

Conhecidas as chaves do dicionário, é possível obter os valores associados. Desta forma determina-se que o primeiro país cujo território é descrito por uma geometria na coleção é o *Chade* (ver Listagem 2.15). As propriedades são expostas por `get()` e os polígonos que descrevem o território por `geometry()`.

---

```
print("Pais: _%s" % (pais.get("country_na").getInfo()))
print("Divisao: _%s" % (pais.get("country_co").getInfo()))
```

---

```
Pais: Chad
Divisao: CD
```

---

Listagem 2.15: Metadados associados ao país

#### 2.4.2.8 Seleção de dados utilizando os metadados - *ee.Filter*

O *GEE* suporta a interrogação dos metadados utilizando objectos *ee.Filter*. Estes objectos podem ser utilizados em separado ou agrupados utilizando funções booleanas por ele expostos (e.g. `And()` ou `Or()`).

Considere-se assim que se pretende obter a geometria de Portugal Continental. Isto será possível utilizando as chaves `country_na` e `country_co`. A representação de Portugal poderia agora ser utilizada em operações de *GEE* mais

complexas ou para obter, por exemplo, a área do território do continente em quilômetros quadrados (ver Listagem 2.16).

---

```
portugal=territorios . filter (ee . Filter . And(\
    ee . Filter . eq ("country_co" , "PO") ,\
    ee . Filter . eq ("country_na" , "Portugal"))
print (portugal . geometry () . area () . getInfo () / 1e6)
```

---

88714.22214961443

---

Listagem 2.16: Cálculo da área de Portugal Continental em  $Km^2$

Note-se que o valor de `portugal.geometry().area()` é expresso em  $m^2$  motivo pelo qual se divide o resultado por `1e6`, ou seja,  $10^6$  no sentido de obter o valor em  $Km^2$ .

#### 2.4.2.9 Dados multi-espectrais - *ee.Image* e *ee.ImageCollection*

Os dados georreferenciados, sob a forma de um mapa de coordenadas discreto, são mantidos no *GEE* utilizando objetos *ee.Image* que podem ser construídos de diversas formas desde uma constante até um identificador único que permite obter uma determinada imagem de uma das bases de dados no repositório do *GEE*.

Este objeto é nuclear na representação e manipulação de dados multi-espectrais. Um objeto *ee.Image* pode representar ou uma única banda multi-espectral ou um conjunto de bandas. Cada coordenada tem associado um valor numérico e um conjunto de metadados variável. Os valores numéricos das coordenadas podem ser utilizados em cálculos quer em operações de agregação (i.e. *reduce*) quer naquilo que os autores designam por "*band math*". Este último tipo de aritmética passa por realizar operações entre duas bandas computando o resultado de uma operação aritmética sobre cada uma das coordenadas comuns.

---

```
imagem_um=ee . Image (1)
imagem_quatro=ee . Image (4)
print (imagem_um . add (imagem_quatro) . getInfo ())
```

---

```
{'type': 'Image', 'bands': [{'id': 'constant', 'data_type': {'type': 'PixelType', 'precision': 'int', 'min': 5, 'max': 5}, 'crs': 'EPSG:4326', 'crs_transform': [1.0, 0.0, 0.0, 0.0, 1.0, 0.0]}]}
```

---

Listagem 2.17: Resultado da soma de duas bandas

Como exemplo, considere-se uma operação entre duas bandas que, por simplificação, são constantes (i.e. todas as coordenadas têm um mesmo valor numérico associado). Então uma operação de soma entre as duas é apresentado na Listagem 2.17.

A banda resultante é também uma banda constante cujo valor mínimo e máximo são a constante 5. Será fácil extrapolar o que acontecerá com bandas multi-espectrais com dados variáveis.

Considere-se a Listagem 2.18 que demonstra este aspeto e a forma de utilizar objectos *ee.Filter* para interrogar uma coleção *ee.ImageCollection*.

---

```

imagens=ee.ImageCollection("COPERNICUS/S2")\
    .filterDate("2019-05-01", "2019-05-31")\
    .filterBounds(portugal.geometry())\
    .filter(ee.Filter.lt("CLOUDY_PIXEL_PERCENTAGE", 20))
banda1=imagens\
    .first()\
    .select("B4")
print(banda1.reduceRegion(reducer=ee.Reducer.minMax(),\
                        geometry=portugal.geometry(),\
                        scale=10,\
                        maxPixels=1e9).getInfo())

```

---

```

{'B4_max': 6186, 'B4_min': 392}

```

---

Listagem 2.18: Resultado da redução da banda representada na variável `banda1`

Traduzindo código na Listagem 2.18, pretende-se obter da coleção de dados da missão *Sentinel 2*, no mês de Maio de 2019, imagens que intercetem o território continental de Portugal e, destas, seleccionar as imagens com menos de 20% de cobertura de nuvens. Estas imagens serão representadas por `imagens`.

A variável `banda1` representará, para imagem mais antiga (i.e. a primeira) de `imagens`, a banda de refletância do vermelho (e.g. designada para na missão *Sentinel 2* "B4").

Dados os valores de refletância vermelha associados a todas as coordenadas representadas na banda, para a geometria de Portugal Continental, obter os valores mínimo e máximo (utilizando uma escala de 10 metros e avaliando um máximo de  $10^9$  coordenadas).

Por outro lado pode-se alterar ligeiramente o código na Listagem 2.18 para obter uma banda que tenha a mediana dos valores observados no mês, representado em `banda2` na Listagem 2.19.

Note-se que o objecto *ee.Image* tem algumas funções que podem simplificar a sintaxe, neste caso **reduce** (`ee.Reducer.median()`) poderia ser substituído por `median()`.

---

```
banda2=imagens\  
  .select("B4")\  
  .reduce(ee.Reducer.median())  
print(banda2.getInfo())
```

---

```
{'type': 'Image', 'bands': [{'id': 'B4_median', 'data_type': {'type': 'PixelType', 'precision': 'double', 'min': 0.0, 'max': 65535.0}, 'crs': 'EPSG:4326', 'crs_transform': [1.0, 0.0, 0.0, 0.0, 1.0, 0.0]}]}
```

---

Listagem 2.19: Resultado da redução da banda representada na variável `banda2`

Observe-se na Listagem 2.19 que o resultado contém uma única banda designada *"B4\_median"* resultante da concatenação do nome da banda original e da operação de agregação utilizada, isto é, `median()`.



## Modelação e concretização

### 3.1 Modelação

Nesta secção descreve-se a metodologia, que se pretende seja aplicável a qualquer ferramenta e fontes de informação georreferenciada no pré-processamento dos dados com o objetivo de produzir valores agregados que possam ser utilizados como entradas para o modelo de cálculo.

#### 3.1.1 Pré-processamento

Na preparação dos dados para utilização no cálculo, terá que se identificar quais as geometrias das zonas de estudo utilizando informação georreferenciada. Designe-se o conjunto de vetores representativo da geometria da zona ardida  $G_{alvo}$  (ver Figura 3.1-a).

Para determinar a zona adjacente, terá que se expandir  $G_{alvo}$  numa distância linear correspondente à largura da faixa que se utilize como referência (ver Figura 3.1-b). A largura da faixa adjacente será um parâmetro de entrada para o algoritmo de seleção de dados. Tratando-se de informação vetorial, a expansão da forma poderá ser realizada por aplicação de uma transformação linear. Se a fonte de informação for um mapa de pontos, então terá que se recorrer a algoritmos adequados para aproximar a mudança de escala do mapa de pontos.

O novo conjunto será designado  $G_{total}$  (ver Figura 3.1-b). A geometria da zona adjacente  $G_{ref}$  será, assim, a diferença geométrica (ver Figura 3.1-c):

$$G_{ref} \equiv G_{total} \setminus G_{alvo} \quad (3.1)$$

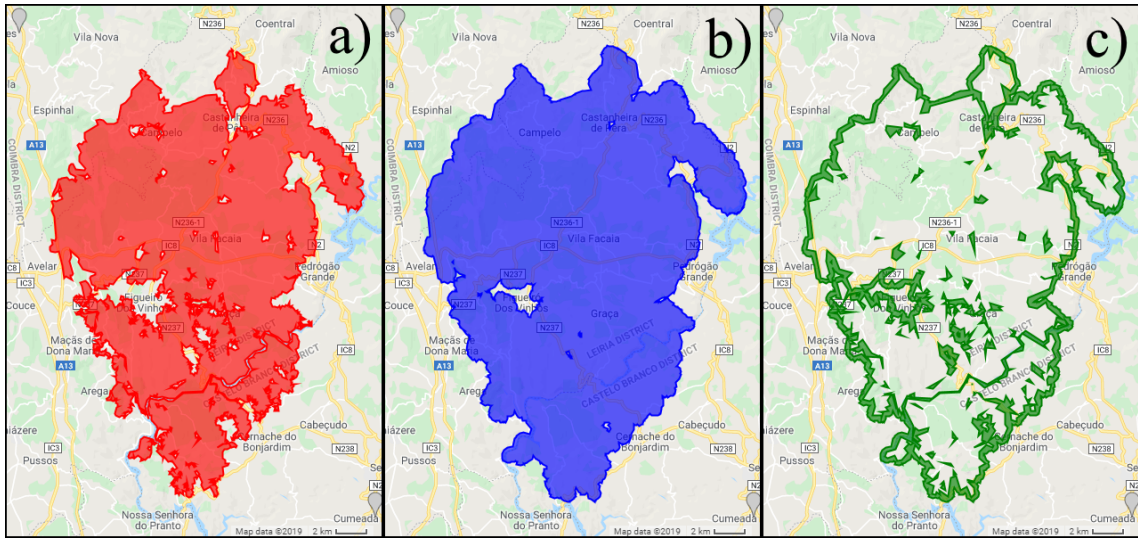


Figura 3.1: Representação, por exemplo, das áreas de *Pedrogão Grande* de  $G_{alvo}$  a vermelho,  $G_{total}$  a azul e  $G_{ref}$  a verde (da esquerda para a direita, ver a Equação 3.1). Pontos assinalados  $(-8.376035, 40.057843)$  e  $(-8.101377, 39.771869)$  WGS84

As características da cobertura pré-existente são diversas e existem áreas que não devem ser incluídas na avaliação do *NDVI*, por exemplo, superfícies de água ou zonas edificadas. Assim, determinadas as geometrias da zona ardida  $G_{alvo}$  e adjacente  $G_{ref}$ , há que seleccionar as coordenadas dentro destas zonas com o tipo de cobertura pré-existente pretendido (e.g. floresta).

Fazendo a intersecção geométrica destas duas zonas com uma fonte de informação de dados georreferenciados que descreva a cobertura da superfície pré-existente obtém-se  $C_{alvo}$  e  $C_{ref}$  (ver Figura 3.2). Estas áreas corresponderão às zonas para as quais se pretende efetivamente obter os compósitos *MVC* de *NDVI*.

Para as coordenadas na zona afetada pelo incêndio  $C_{alvo}$  e de referência  $C_{ref}$ , calcula-se o *NDVI* observado diariamente para o intervalo de datas em análise. Destes conjuntos de valores, produz-se os compósitos *MVC*.

Os dois compósitos serão denominados  $MVC_{alvo}(t)$  para a zona ardida e  $MVC_{ref}(t)$  para a zona adjacente de referência ( $t$  será a data do compósito). Estes conjuntos

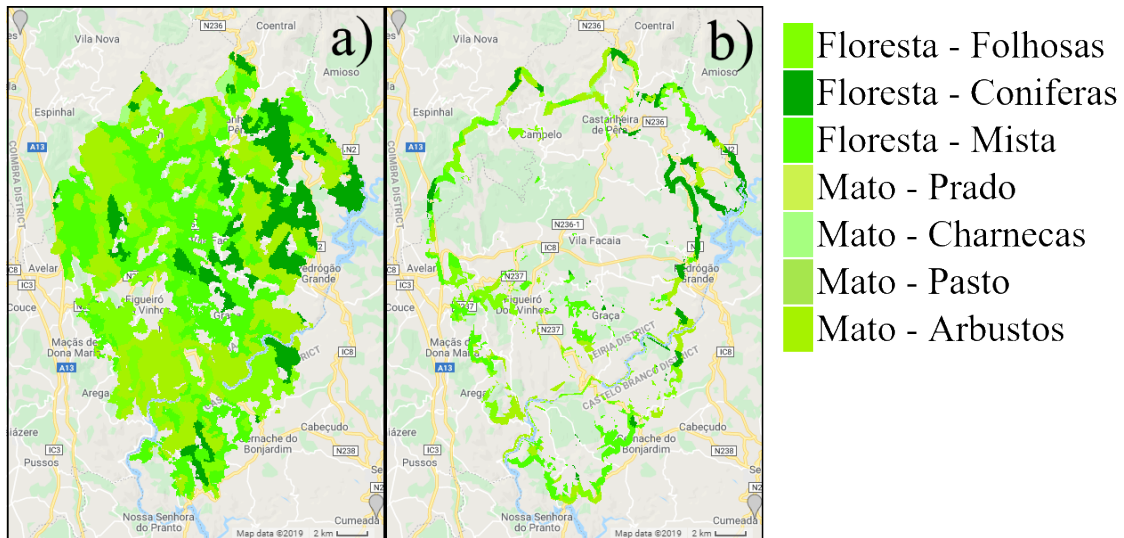


Figura 3.2: Representação, por exemplo, das áreas de *Pedrogão Grande* de  $C_{alvo}$  e  $C_{ref}$  (da esquerda para a direita). Os tons de verde correspondem a cada uma das categorias de vegetação pré-existente. Pontos assinalados  $(-8.376035, 40.057843)$  e  $(-8.101377, 39.771869)$  WGS84

terão os valores máximos de  $NDVI$  observados para cada coordenada, na respetiva zona, no intervalo de datas determinado por  $t$  e pelo número de dias  $k$  estabelecido com base no tempo de revisita do satélite.

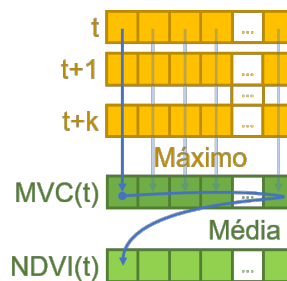


Figura 3.3: Diagrama explicativo da avaliação das séries temporais.

Os valores a utilizar no cálculo corresponderão à média de  $NDVI$ , observada em cada data  $t$  para todas as coordenadas válidas dos compósitos  $MVC$  (ver Figura 3.3) correspondentes à zona ardida e à zona adjacente, respetivamente:

$$NDVI(t) = \overline{MVC_{alvo}(t)} \quad (3.2)$$

$$NDVI^*(t) = \overline{MVC_{ref}(t)} \quad (3.3)$$

### 3.1.2 Cálculo

O cálculo utilizará as séries temporais dos valores de  $NDVI$  observados na zona ardida  $NDVI(t)$  (3.2) e na zona de referência  $NDVI^*(t)$  (3.3) resultantes do pré-processamento.

Expresse-se a diferença dos valores de  $NDVI$  observados na forma de (3.4).

$$y(t) = NDVI(t) - NDVI^*(t) \quad (3.4)$$

Representando a diferença expressa em (3.4), em função do valor máximo,  $a = NDVI(0) - NDVI^*(0)$ , que ocorrerá no primeiro momento após a ocorrência do incêndio para o qual exista um varrimento válido,  $t = 0$ , resultará:

$$\frac{y(t)}{a} = \frac{NDVI(t) - NDVI^*(t)}{NDVI(0) - NDVI^*(0)} \quad (3.5)$$

O modelo proposto por Gouveia et al. [14], utiliza o decaimento exponencial na definição do rácio de recuperação instantâneo de  $NDVI$  expresso por:

$$\frac{dy}{dt} = -b \times y \quad (3.6)$$

A solução da Equação diferencial 3.6 será:

$$y(t) = a \times e^{-b \times t} \quad (3.7)$$

Após simplificação, poderá ser reescrita como:

$$\ln\left(\frac{y(t)}{a}\right) = -b \times t \quad (3.8)$$

Utilizando a Equação 3.8, determinamos o coeficiente  $b$  utilizando uma regressão linear simples.

O modelo tem como parâmetro o *limiar*, proporção da diferença máxima de  $NDVI$  expressa na Equação 3.5 que se considera traduzir o ponto de recuperação da vegetação. Note-se que o modelo nunca permitirá que este limiar seja nulo ( $\ln(0)$  é indefinido). Só será possível estimar o tempo de recuperação para um limiar não nulo.

Nos trabalhos anteriores, o nível homólogo correspondia a 90% do  $NDVI$  médio do modelo teórico de referência. Neste modelo propõe-se uma abordagem

equivalente. A estimativa do tempo de recuperação será traduzida em função da proporção *limiar*, correspondente à percentagem de *NDVI* da diferença máxima verificada em (3.5).

Resolvendo a Equação 3.8 em ordem a  $t$ , obtém-se o estimador para o tempo previsto de recuperação da vegetação:

$$t = -\frac{\ln(\text{limiar})}{b} \quad (3.9)$$

Dito de outra forma, estima-se que no momento  $t$  o *NDVI* observado será equivalente, em valor, à proporção *limiar* da diferença máxima observada de *NDVI*, logo após o incêndio ( $t = 0$ ).

Ao momento estimado  $t$  atribui-se o significado semântico de *tempo de recuperação*, e o valor de *NDVI* implícito no *limiar*, será o *estado normal da vegetação*.

## 3.2 Concretização

Definida a metodologia, nas secções seguintes descreve-se a arquitetura da solução, que foi concretizada numa biblioteca em *Python*.

### 3.2.1 Arquitetura

A arquitetura da solução foi desenhada de forma a estabelecer a segregação entre a interação com o *GEE*, na fase de pré-processamento, e a implementação da metodologia de cálculo. Ambos os aspetos foram concretizados utilizando a linguagem *Python*.

As fontes de dados utilizadas estão alojadas no *GEE*. Todavia os dados multi-espectrais da missão *Sentinel 2* e os dados de classificação da cobertura de superfície pré-existente *CLC* (ver Subsubsecção 3.2.2.2) existem no repositório do *GEE*, mas os dados do Instituto da Conservação da Natureza e Florestas [18] (*ICNF*), de caracterização das áreas ardidadas em 2017 (ver Subsubsecção 3.2.2.1), tiveram que ser incorporados num repositório privado alojado na mesma plataforma. Na Figura 3.4 esta diferença é traduzida nas diferentes cores utilizadas para representar as fontes de informação.

O procedimento de inclusão de fontes de dados no repositório privado não é detalhado neste documento, mas trata-se de uma operação simples utilizando

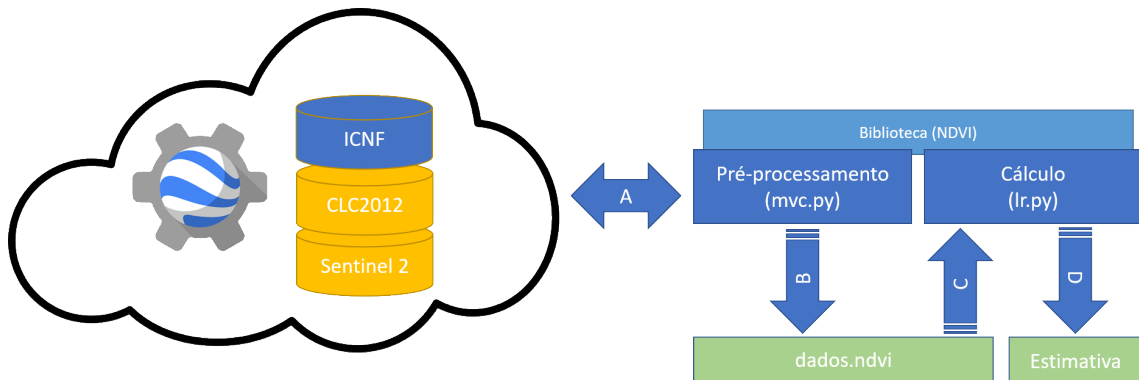


Figura 3.4: Diagrama da arquitetura da solução.

as funcionalidades disponibilizadas pelo *GEE* na interface de utilizador do *Code Editor*. Para fontes estáticas, como é o caso da base de dados do *ICNF*, esta solução é adequada.

A interação entre a biblioteca concretizada neste trabalho, designada *recveg*<sup>1</sup>, e o *GEE*, traduzida na Figura 3.4 como fluxo [A], ocorre por utilização da biblioteca disponibilizada pela *Google*. Traduz-se na inclusão do *namespace ee* em todos os artefactos de *Python* estabeleçam a ponte com o *GEE*. Todos os objetos prefixados por *ee* traduzem representações de entidades que suportam as funcionalidades do *GEE* (ver Subsecção 2.4.2).

Foi concebida a aplicação cliente, *mvc.py*<sup>2</sup>, que procede às operações necessárias para que a partir do fluxo [A] seja possível extrair os dados processados no *GEE*, fluxo [B], e persisti-los num ficheiro (e.g. *dados.ndvi*), que desta forma dissocia o procedimento do pré-processamento dos dados que são a entrada para metodologia de cálculo.

Para garantir a universalidade da utilização dos dados extraídos, o ficheiro é a representação de uma coleção de *Python* guardada num ficheiro utilizando a biblioteca *pickle* [23]. A biblioteca *pickle* serializa um objeto *Python* para um ficheiro binário que permite posteriormente a reconstrução do objeto a partir do conteúdo deste ficheiro.

Num cenário de utilização, uma vez produzido um ficheiro com dados correspondentes a um evento de incêndio utilizando o cliente *mvc.py*, pode-se proceder à utilização da segunda aplicação cliente, designada *lr.py*<sup>3</sup>, que recupera

<sup>1</sup>Consultar no repositório localizado em <https://github.com/a44447/tese>

<sup>2</sup>Consultar no repositório localizado em <https://github.com/a44447/tese/blob/master/mvc.py>

<sup>3</sup>Consultar no repositório localizado em <https://github.com/a44447/tese/blob/master/lr.py>

os dados guardados no ficheiro, correspondente ao fluxo [C] na Figura 3.4. Esta aplicação gera a estimativa e as Figuras 4.1 e 4.2.

Este procedimento poderá ser reproduzido em qualquer outro *script* que se pretenda uma vez que não foi estabelecida qualquer dependência entre a biblioteca *recveg* e o ficheiro gerado. O dicionário que é guardado no ficheiro tem os metadados descritivos para que o utilizador possa atribuir significado aos valores (ver Apêndice B).

O procedimento de cálculo será assim assegurado, neste caso de uso, por *lr.py* que utilizará alguns objetos incluídos na biblioteca *recveg* (e.g. constantes da classe *Composito*), dissociados do *GEE* para produzir as estimativas do tempo de recuperação da vegetação com base nos dados extraídos do *GEE* na fase de pré-processamento, fluxo [D] na Figura 3.4.

## 3.2.2 Fontes de informação e ferramenta

### 3.2.2.1 Identificação das áreas ardidas

A geometria das áreas ardidas no ano de 2017, foi compilada pelo *ICNF*. A base de dados contém a geometria das zonas onde ocorreram incêndios e, entre outra informação, um identificador único do incêndio e a data de ocorrência. Este repositório foi importado para o *GEE* e será a fonte de dados para  $G_{alvo}$  e  $G_{ref}$  da modelação.

Do conjunto de metadados desta fonte, modelou-se uma entidade, incluída na biblioteca *recveg*, designada *Evento* e uma funcionalidade de exploração destes dados designada *ZonasArdidas*, ambas concretizadas sob a forma de classes de *Python*.

A classe *ZonasArdidas*, não sendo necessária para suportar a metodologia, é no entanto útil na identificação dos eventos de incêndio na base de dados do *ICNF*. O método *distritos()*, parcialmente representado na Listagem 3.1, permite obter um dicionário com o histograma dos eventos ocorridos por distrito. O método *extrai()* permite obter zero ou mais eventos de incêndio em função de um critério de filtragem baseado nos campos *distrito*, *concelho*, *freguesia* e *local*. Note-se que o objetivo do trabalho não inclui a exploração da base de dados do *ICNF*, esta classe foi desenvolvida para garantir ao utilizador uma forma básica de interação. Como caso de uso, filtrados os eventos para um distrito obtido por invocação de *distritos()*, permitirá fazer o *drill-down* para outros níveis de organização administrativa (e.g. concelho).

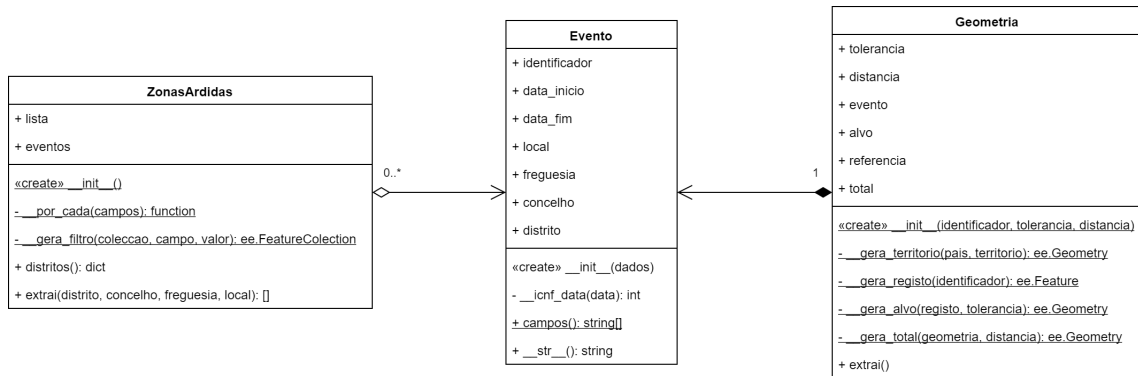


Figura 3.5: Diagrama de classes UML - informação de eventos de incêndio e geometria.

---

```

def distritos(self):
    (...)
    return ee.FeatureCollection(GEE.ICNF_COLECCAO)\
        .aggregate_histogram(GEE.ICNF_CAMPO_DISTRITO)\
        .getInfo()
  
```

---

Listagem 3.1: Troço de código de *zonasardidas.py* que obtém o histograma

Contribuí diretamente para a concretização da metodologia a classe *Geometria*. Esta é instanciada com base em três parâmetros: (i) o identificador único do evento na base de dados do *ICNF*; (ii) o grau de tolerância, em metros, a aplicar na simplificação da geometria (tem como efeito melhoria na performance de cálculo); e (iii) a distância linear, em metros, da faixa que se pretende definir como sendo a área de referência.

A classe *Geometria* contém ainda um conjunto de métodos privados estáticos que geram todas as representações de operações necessárias no *GEE* (ver exemplo na Listagem 3.2).

---

```

self.__registo = Geometria.__gera_registro(identificador)
(...)
@classmethod
def __gera_registro(cls, identificador):
    return ee.FeatureCollection(GEE.ICNF_COLECCAO)\
        .filter(ee.Filter.eq(GEE.ICNF_CAMPO_IDENTIFICADOR,
            identificador))\
        .first()
  
```

---

Listagem 3.2: Troço de código que inclui o método estático *\_\_gera\_registro()*

Na classe *Geometria* são expostas três representações correspondentes às geometrias alvo, referência e total em propriedades públicas que utilizam esta semântica. Expõe ainda uma instância da classe *Evento*, numa propriedade com o mesmo nome. Porque esta última necessita de obter dados do *GEE* só está disponível após a invocação de um método designado *extraí()* (ver Listagem 3.3).

---

```
def extraí(self):
    self.__evento = Evento(self.__registro.toDictionary(Evento.campos())
        .getInfo())
```

---

Listagem 3.3: Método público *extraí()*

Note-se novamente a distinção entre a construção da representação de uma operação de *GEE*, expressa no método descrito na Listagem 3.2, e a obtenção dos dados resultantes da execução da operação na Listagem 3.3, por utilização do método *getInfo()*. Esta distinção é abundantemente sublinhada na Seção 2.4.

### 3.2.2.2 Cobertura de superfície

A fonte de informação relativa à cobertura pré-existente será o *CLC* [4] para o ano de 2012 distribuído no âmbito do programa *Copernicus* da União Europeia, a cada 6 anos. Esta caracteriza a superfície terrestre quanto ao tipo de cobertura. Cada coordenada, correspondente a 100m de superfície, e é caracterizada por um código numérico que traduz uma de 44 classes de cobertura (ver Apêndice A). Integrada no repositório do *GEE* será a fonte de dados para  $C_{alvo}$  e  $C_{ref}$  da modelação.

Nem todas as classes de cobertura do solo são relevantes para a metodologia. Tratando-se da estimação do tempo de recuperação da vegetação, utilizando o *NDVI*, interessa reduzir o enviesamento eliminando todas as coordenadas que não correspondam a zonas em que existia vegetação antes do incêndio. As classes que traduzem áreas de vegetação correspondem aos códigos 3 no primeiro nível de *CLC* (ver Apêndice A).

---

```
class Floresta:
    Folhosas = 23
    Coniferas = 24
    Mista = 25

    @classmethod
    def lista(cls):
        return [cls.Folhosas, cls.Coniferas, cls.Mista]
```

---

Listagem 3.4: Classe *Floresta*

Para simplificar a utilização das diferentes classes de cobertura foram criadas duas enumerações (não existindo o conceito de enumerado em *Python* foi substituído por uma classe com propriedades numéricas<sup>4</sup>, ver Listagem 3.4): (i) uma correspondente aos valores das classes de floresta; e (ii) outra correspondente aos valores das classes de mato. A conjugação de ambas resulta em todos os valores das classes de vegetação. A semântica dada às constantes é semelhante aos conceitos que traduzem (ver classes *Floresta* e *Mato* na Figura 3.6).

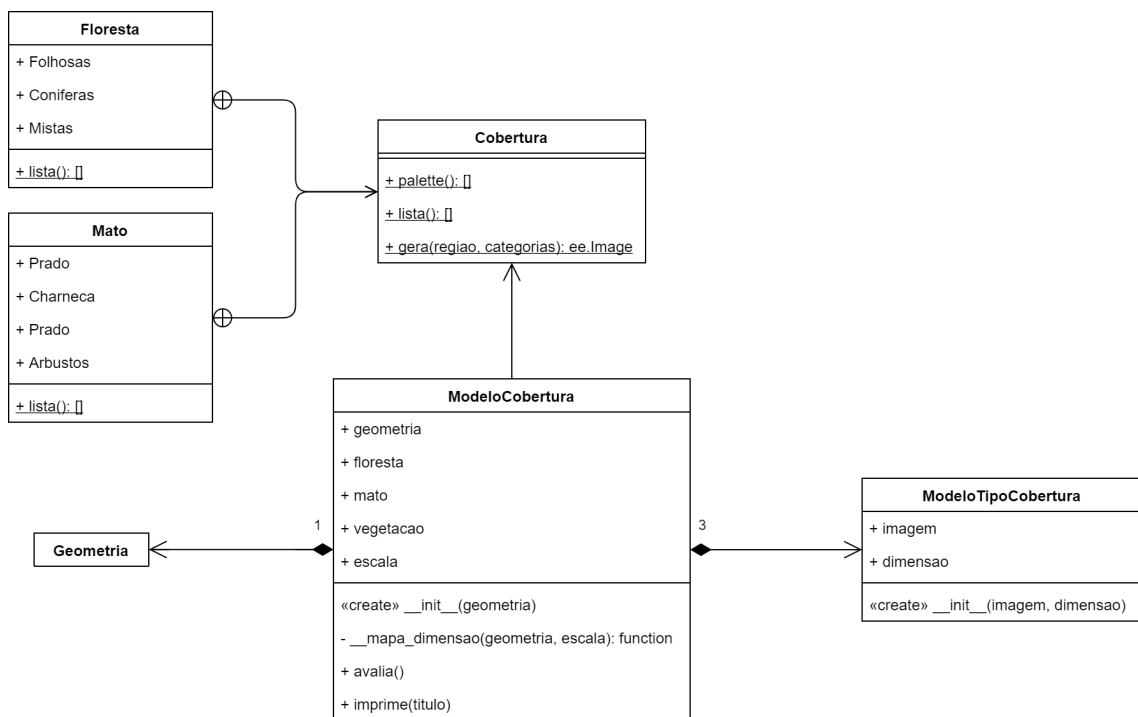


Figura 3.6: Diagrama de classes UML - informação de cobertura de superfície.

A classe de *Python* *ModeloCobertura* é uma composição de uma instância da classe *Geometria* e de três instâncias da classe *ModeloTipoCobertura* correspondentes aos tipos de cobertura (i.e. floresta, mato e vegetação). A classe *Cobertura* é estática e os seus métodos apenas produzem representações de *GEE*. Foi autonomizada de *ModeloCobertura* por uma questão de simplificação e dissociação formal.

```
@classmethod
```

```
def __mapa_dimensao(self, geometria, escala):
```

<sup>4</sup>A biblioteca foi inicialmente desenvolvida em *Python 2.7*, motivo pelo qual não se usou uma subclasse de *Enum*, disponível desde a versão 3.4

```

def __funcao_dimensao(imagem):
    return ee.Image(imagem).reduceRegion(\
        reducer=ee.Reducer.sum(),\
        geometry=geometria,\
        scale=escala,\
        maxPixels=1e9)

return __funcao_dimensao

def avalia(self):
    if self.__dimensoes is None:
        lista = ee.List([self.floresta.imagem, self.mato.imagem,\
            self.vegetacao.imagem])\
            .map(self.__mapa_dimensao(self.geometria, self.escala))\
            .getInfo()
    (...)

```

---

Listagem 3.5: Cálculo das dimensões de todos os modelos de cobertura

As dimensões de cada um dos modelos de cobertura, entendidas como o número de coordenadas válidas por modelo, são adequadas a demonstrar a utilização do paradigma *map / reduce*. A variável *lista* contém um objeto lista com as três imagens correspondentes aos três modelos de cobertura. A invocação do método *map()*, tendo como parâmetro o método *\_\_mapa\_dimensao()*, distribui o processamento da lista em função da gestão de recursos do *GEE*.

Cada elemento da lista será processado no método *\_\_mapa\_dimensao()* onde é realizada uma operação de redução que processa a soma dos valores associados a cada coordenada da imagem. Como estes só podem ser 1's ou 0's (com 1 representando as coordenadas válidas) a soma corresponderá ao número de coordenadas do *CLC* que contribui para o cálculo de *NDVI* na imagem.

### 3.2.2.3 Dados multi-espectrais

A fonte de informação fundamental será a informação da missão *Sentinel 2* [8], do programa *Copernicus* da União Europeia. Os instrumentos a bordo dos satélites (a constelação é composta por dois satélites) produzem 13 bandas multi-espectrais mais 3 bandas auxiliares [7].

Serão utilizadas as bandas do vermelho ( $\approx 665nm$  de comprimento de onda) e a próxima dos infravermelhos ( $\approx 833nm$  de comprimento de onda) no cálculo do *NDVI* que consiste na aplicação da diferença normalizada expressa na Equação 1.1.

A missão *Sentinel 2* tem um tempo de revisita de 5 dias (tempo que medeia entre duas passagens na mesma vertical, com as mesmas condições operacionais) e uma resolução de  $10m$  (para estas bandas de refletância). Estes dados fazem parte do repositório do *GEE* e serão utilizados na avaliação de  $MVC_{alvo}(t)$  e  $MVC_{ref}(t)$  da modelação e, conseqüentemente, na obtenção das séries temporais  $NDVI(t)$  e  $NDVI^*(t)$  (ver Equação 3.2 e 3.3).

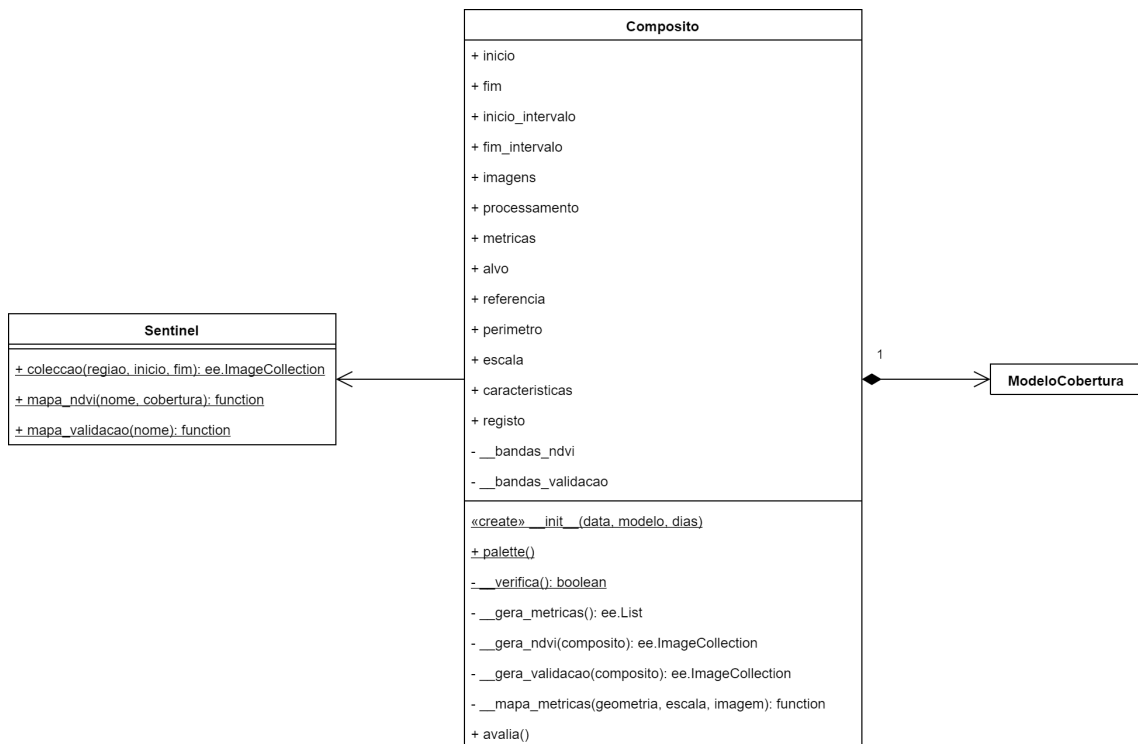


Figura 3.7: Diagrama de classes UML - compósitos.

Como é possível observar na Figura 3.7 a classe *Composito*, é uma composição de uma instância do *ModeloCobertura* e, pelos mesmos motivos apresentados relativamente à classe *Cobertura*, utiliza uma classe estática, designada *Sentinel*, que dissocia a representação de operações fundamentais relacionadas com a fonte de informação dos dados multi-espectrais do *Sentinel 2*. A composição de operações relacionadas com o cálculo dos compósitos foi concretizada em métodos privados estáticos da classe *Composito*.

### 3.2.3 Pré-processamento

**Determinação da geometria e cobertura** A área ardida  $G_{alvo}$  foi extraída da base de dados do ICNF. Procedeu-se à expansão de  $G_{alvo}$  em  $300m$  para obter  $G_{total}$

e realizou-se uma simplificação dos polígonos assegurando um erro máximo de  $100m$  (para eficiência de cálculo<sup>5</sup>). Calculou-se a diferença geométrica entre  $G_{total}$  e  $G_{alvo}$  para obter  $G_{ref}$ . Este processo que é assegurado pela classe *Geometria* (ver Figura 3.5).

No *GEE*, filtrou-se o *CLC* para os tipos de cobertura pretendidos, delimitados por  $G_{alvo}$  e  $G_{ref}$ . O algoritmo calcula para cada coordenada o valor 1 ou 0 (válida ou inválida) resultando os conjuntos  $C_{alvo}$  e  $C_{ref}$ . Esta funcionalidade é concretizada na classe *ModeloCobertura* (ver Figura 3.6).

**Cálculo dos compósitos de NDVI** O número de dias  $t$  tem origem na data do primeiro varrimento válido na zona ardida após a data de ocorrência do incêndio.  $MVC_{alvo}(t)$  e  $MVC_{ref}(t)$  serão os compósitos *MVC*, a 10 dias, que resultarão da composição das imagens recolhidas em  $[t, t + 10[$  nas coordenadas válidas de  $C_{alvo}$  e  $C_{ref}$ . Esta funcionalidade está implementada na classe *Composito* (ver Figura 3.7).

Os compósitos agregam os valores máximos de *NDVI* observados em cada coordenada utilizando os dados da missão *Sentinel 2*. O número de valores em cada compósito depende das condições de recolha para o dia  $t$ . A missão *Sentinel 2* sobrevoa a mesma zona geográfica a cada 5 dias (cada satélite da constelação faz uma passagem a cada 10 dias) [8] em varrimentos da superfície terrestre de  $290km$ .

A quantidade de varrimentos necessários para cada zona ardida será variável. Por exemplo, para *Pedrogão Grande* basta um único varrimento, para a *Mata Nacional de Leiria* são necessários 4 varrimentos (ver Figura 1.2).

A seleção dos varrimentos exclui, à priori, os que apresentam uma percentagem de cobertura por nuvens superior a 20%. Desta forma, são eliminadas imagens completas o que significa que um compósito *MVC*, para um dia, poderá não conter qualquer informação sendo, desta forma, descartado.

É feita uma segunda avaliação da cobertura de nuvens, utilizando a banda de qualidade [8]. Esta tem uma resolução de  $60m$  enquanto que as bandas *NIR* e *RED*, utilizadas para calcular o *NDVI*, têm uma resolução de  $10m$ . Assim pode ocorrer eliminação de dados por excesso. O efeito da nuvens está visível na Figura 3.8.

---

<sup>5</sup>A simplificação passa por fundir os vetores em conjuntos com um mínimo de  $100m$ , reduzindo desta forma o número de vetores, simplificando a geometria e a eficácia do cálculo.

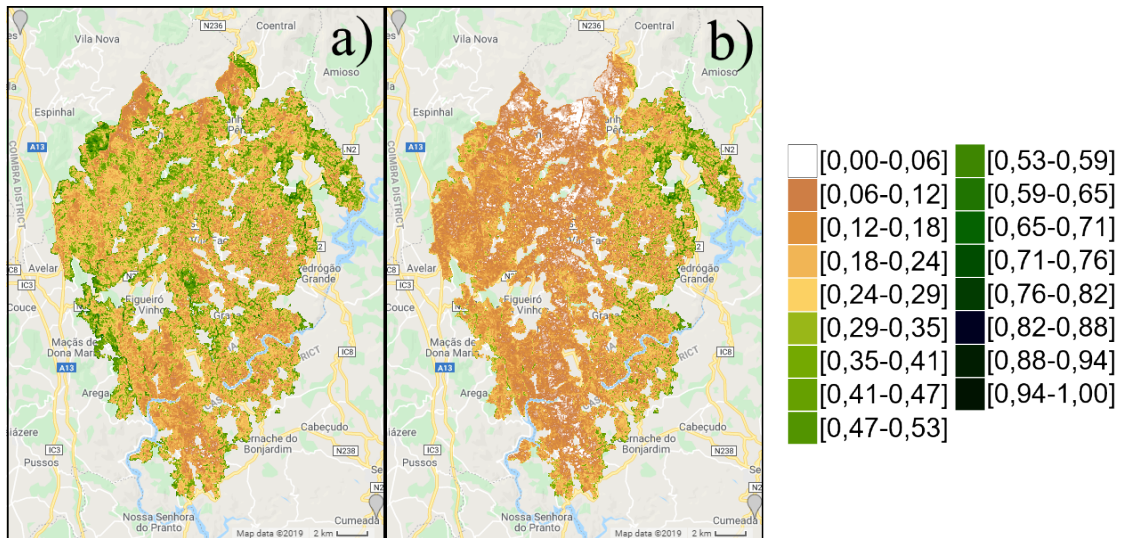


Figura 3.8: MVC da área afetada pelo fogo de *Pedrogão Grande* para os dias 24-09-2017 (esq.) e 04-10-2017 (dir.). Neste último, note-se a eliminação de valores em função da presença de nuvens (zonas brancas) e a diminuição dos valores de NDVI (zonas laranja). Pontos assinalados  $(-8.376035, 40.057843)$  e  $(-8.101377, 39.771869)$  WGS84

As duas condicionantes anteriores, tempo de revisita e eliminação de nuvens, determinam que, por exemplo, cada compósito MVC poderá ser constituído por 0 a 2 imagens no caso de *Pedrogão Grande* e 0 a 8 imagens para a zona da *Mata Nacional de Leiria*.

O algoritmo, desenvolvido utilizando o *GEE*, consolida a geração de  $NDVI(t)$  e  $NDVI^*(t)$ , dado o identificador de incêndio na base de dados do *ICNF*, a tolerância de simplificação da geometria e a distância da faixa de referência. Permite também guardar os dados utilizando um formato de ficheiro que permite segregar o pré-processamento do cálculo do tempo de recuperação. Todas estas funcionalidades estão concretizadas na classe *Modelo* (ver Figura 3.9).

A classe *Modelo* é uma composição de uma instância da classe *Geometria*, três instâncias de *ModeloCobertura* (floresta, mato e vegetação) e uma instância de uma classe designada *ModeloNotificacao* que permite notificar um cliente que implemente a classe *Modelo* dos eventos ocorridos na interação com o *GEE*. Note-se que para implementação da metodologia bastaria uma instância de *ModeloCobertura*. A classe *Modelo* foi construída desta forma para suportar a análises de sensibilidade que será avaliada adiante (ver Seção 4.3).

O construtor da classe *Modelo* aceita quatro parâmetros: (i) o identificador do incêndio na base de dados do *ICNF*; (ii) a tolerância (em metros) de simplificação

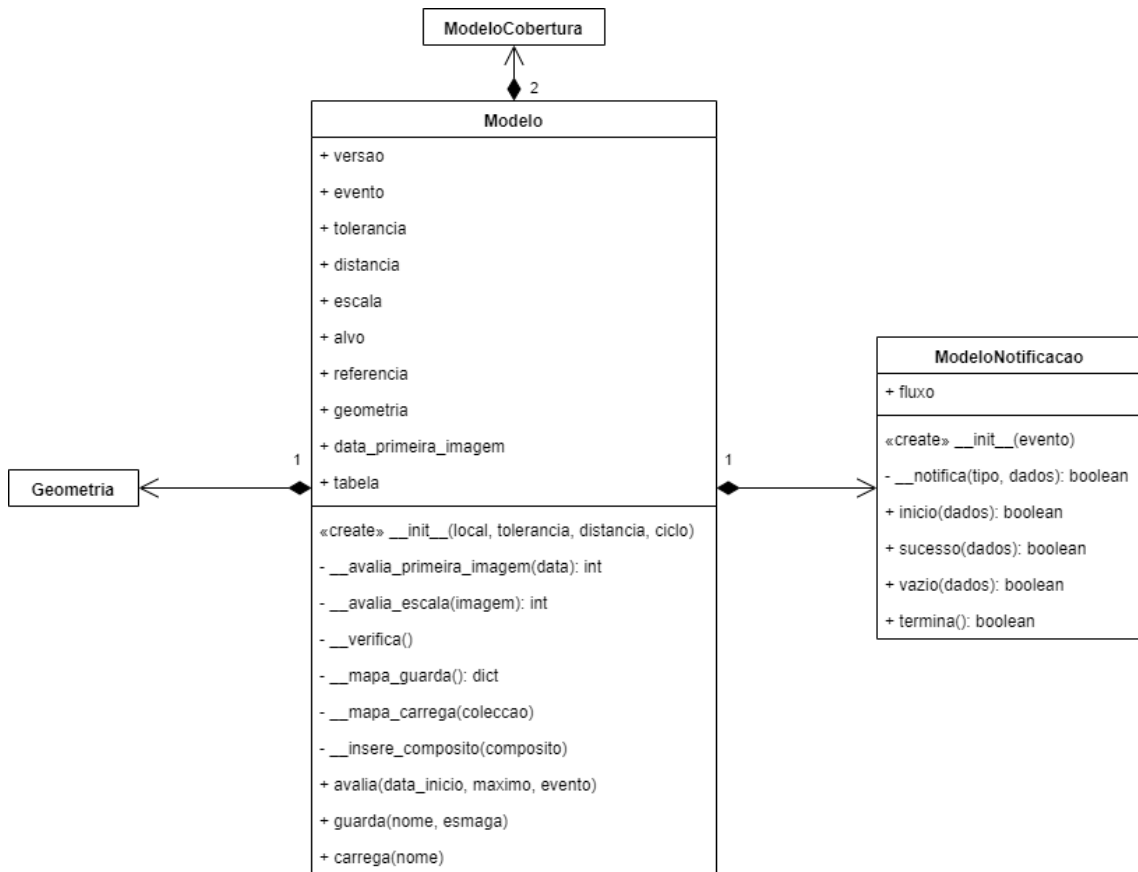


Figura 3.9: Diagrama de classes UML - modelo.

da geometria; (iii) a distância (em metros) correspondente à largura pretendida para a zona de referência; e (iv) o número de dias a considerar para cada intervalo de cálculo do MVC.

Algumas das propriedades expostas pela classe *Modelo*, correspondem aos conceitos apresentados como se resume na Tabela 3.1.

Note-se em particular que a modelação dos dados no ficheiro de destino é feita utilizando um dicionário e matrizes de dados para que o utilizador não fique dependente da biblioteca *recveg* para aplicar algoritmos de cálculo. Desta forma não se esconde a complexidade no caso das séries  $NDVI(t)$  e  $NDVI^*(t)$  optando-se por as expor da forma mais primitiva. Neste aspeto apenas se dotou a classe *Composito* de um conjunto de constantes numéricas que facilita a tarefa, mas que não obriga à sua utilização (ver Listagem B.2).

A utilização de cada uma das séries é feita com base em matrizes geradas utilizando o *numpy* [22] para facilitar a interação com outras ferramentas do ecossistema de *Python*.

Tabela 3.1: Conceitos genéricos e as propriedades da classe *Modelo*

Conceito	Classe <i>Modelo</i>
$G_{alvo}$	<i>geometria.alvo</i>
$G_{ref}$	<i>geometria.referencia</i>
$G_{total}$	<i>geometria.total</i>
$C_{alvo}$	<i>alvo.vegetacao.imagem</i>
$C_{ref}$	<i>referencia.vegetacao.imagem</i>
$t=0$	<i>data_primeira_imagem</i>
$NDVI(t)$	<i>tabela[:, Composito.NDVI_MEDIA_VEGETACAO_ALVO]</i>
$NDVI^*(t)$	<i>tabela[:, Composito.NDVI_MEDIA_VEGETACAO_REFERENCIA]</i>

O cliente desenvolvido para utilizar a classe *Modelo* é *mvc.py*. É uma interface de utilizador simples que acompanha o pré-processamento dos dados. Este processo pode ser extenso e, eventualmente, ser interrompido por indisponibilidade temporária do *GEE*. Para ultrapassar esta dificuldade dotou-se a classe *Modelo* de algumas funcionalidades adicionais:

- A capacidade de persistir os dados em ficheiro, é complementada com a possibilidade de fazer a atualização incremental deste para que os dados previamente obtidos não tenham que ser recalculados;
- Dissociando o cliente da lógica de implementação dos algoritmos de interação com o *GEE* existe um mecanismo de notificação de eventos, suportado pela classe *ModeloNotificacao* (ver Figura 3.9), que permite à aplicação cliente ser notificada de quatro eventos específicos: (i) o início do cálculo de um *MVC*; (ii) a indisponibilidade de informação para um dado *MVC*; (iii) o cálculo com sucesso de um *MVC*; e (iv) a conclusão do procedimento de recolha.

A interação com a aplicação cliente é baseada num mecanismo simples de passagem de uma referência a uma função de *callback*;

- A classe *ModeloNotificacao* atua igualmente como semáforo de controle de execução do processo de recolha de dados em função do resultado booleano da função que faz a notificação da aplicação cliente. Sendo *False* o processamento será interrompido.

### 3.2.4 Cálculo

**Ficheiro de dados** - O cálculo do tempo de recuperação será assim realizado a partir dos dados extraídos no pré-processamento e persistidos num ficheiro.

Como foi já referido, o formato dos dados é independente da biblioteca o que permitirá a sua utilização, sem qualquer restrição, desde que observada a estrutura do ficheiro.

Note-se igualmente dois aspetos fundamentais:

- A existência de um dicionário designado *metadados* que auxilia na leitura do conteúdo do ficheiro, dissociando este da biblioteca que o produziu; e
- A natureza *standard* das estruturas de dados utilizando apenas dicionários, matrizes, listas e tipos primitivos de dados.

A representação *JSON* (abreviada) do ficheiro que é gerado pelos algoritmos de pré-processamento encontra-se nos apêndices na Listagem B.1

No Apêndice C inclui-se um exemplo de utilização de um ficheiro gerado pelo pré-processamento para obtenção de um gráfico das diferenças de *NDVI*.

**Algoritmo de cálculo** - O algoritmo de cálculo, embora esteja modelado na biblioteca *recveg*, na classe *DecaimentoExponencial*, pretende-se que seja independente do pré-processamento.

Mais relevante que o código produzido, cuja simplicidade é evidente, será a descrição do procedimento de cálculo:

- Com as séries  $NDVI(t)$  e  $NDVI^*(t)$ , resultantes do pré-processamento, e sendo  $h$  o número de dias decorridos até à data atual, foi obtido por aplicação da Equação 3.8

$$Y = \ln\left(\frac{y(t)}{a}\right), \text{ com } t \in X \equiv [0, h[ \quad (3.10)$$

- Aplicou-se a regressão linear a  $Y = -b \times X$ ;
- Obteve-se o coeficiente  $b$ , e estimou-se  $\hat{y}(t) = a \times e^{-b \times t}$  que decorre da Equação 3.8

Note-se que  $\hat{y}(t) = 0$  não existe, apenas se pode calcular o momento em que a estimativa da diferença corresponde a um determinado limiar. Considere-se assim a Equação 3.5, pode-se concluir que esta diferença representa uma percentagem e, desta forma, pode-se arbitrar que a estimativa do tempo de recuperação será traduzida em função desta proporção. Assim, por exemplo, para um limiar de

10% da diferença verificada aquando da primeira imagem válida após o incêndio, resulta a equivalência:

$$\frac{NDVI(t) - NDVI^*(t)}{NDVI(t=0) - NDVI^*(t=0)} = 10\% \implies \frac{\hat{y}(t)}{a} = 0.1 \quad (3.11)$$

No próximo capítulo procede-se à aplicação da metodologia, e da sua concretização, a dois casos de estudo de incêndios ocorridos em 2017: (i) na zona de *Pedrogão Grande*; e (ii) na zona da *Mata Nacional de Leiria*.

Procede-se a uma análise de sensibilidade, utilizando o caso de estudo de *Pedrogão Grande*, desagregando as observações em função de duas categorias de cobertura pré-existente (mato e floresta), verificando o comportamento de cada uma das categorias em relação ao agregado total. É também realizado um estudo comparativo do método de cálculo com base no decaimento exponencial por contraponto a uma abordagem de aproximação das séries temporais harmónicas.

# 4

## Casos de estudo

Para o cálculo do tempo de recuperação nas áreas de estudo, sujeitas à metodologia proposta, parte-se do pressuposto de que, como referido na explicação do modelo de cálculo (ver Subseção 3.1.2), se considera um *limiar* que traduz a proporção da maior diferença de *NDVI* verificada logo após a ocorrência do incêndio. Note-se que não existe um valor ótimo para esta percentagem porque ao longo do ano, num cenário normal, o valor do *NDVI* varia (ver Figura 4.3). Estabelece-se assim que o valor de *NDVI* “normal” será de 10% daquela diferença. Este valor foi utilizado por Gouveia et. al [14] ao estabelecer o limiar homologado.

Em resumo, estabelece-se que a vegetação terá recuperado o seu estado normal uma vez que a diferença de *NDVI*, entre a zona afetada e a zona de referência, atinja 10% do máximo da mesma diferença, medida na primeira observação após o incêndio.

Os dados utilizados na avaliação destes casos de estudo foram recolhidos no dia 24-05-2019, por execução da aplicação *mvc.py*. Os ficheiros de resultados foram denominados *pedrogao.ndvi* e *leiria.ndvi* respetivamente<sup>1</sup>. Os resultados e gráficos apresentados foram obtidos por execução da aplicação *lr.py* para cada um dos ficheiros anteriores.

---

<sup>1</sup>Disponíveis no repositório localizado em <https://github.com/a44447/tese/dados>

## 4.1 Pedrogão Grande

A zona de *Pedrogão Grande* tem um tipo de cobertura vegetal dominado por floresta mista e mato.

Obtiveram-se resultados evidenciados na Figura 4.1 em que o diâmetro dos pontos verdes, relativos ao *NDVI* observado para a vegetação, é variável em função do número de varrimentos avaliados para o compósito *MVC*.

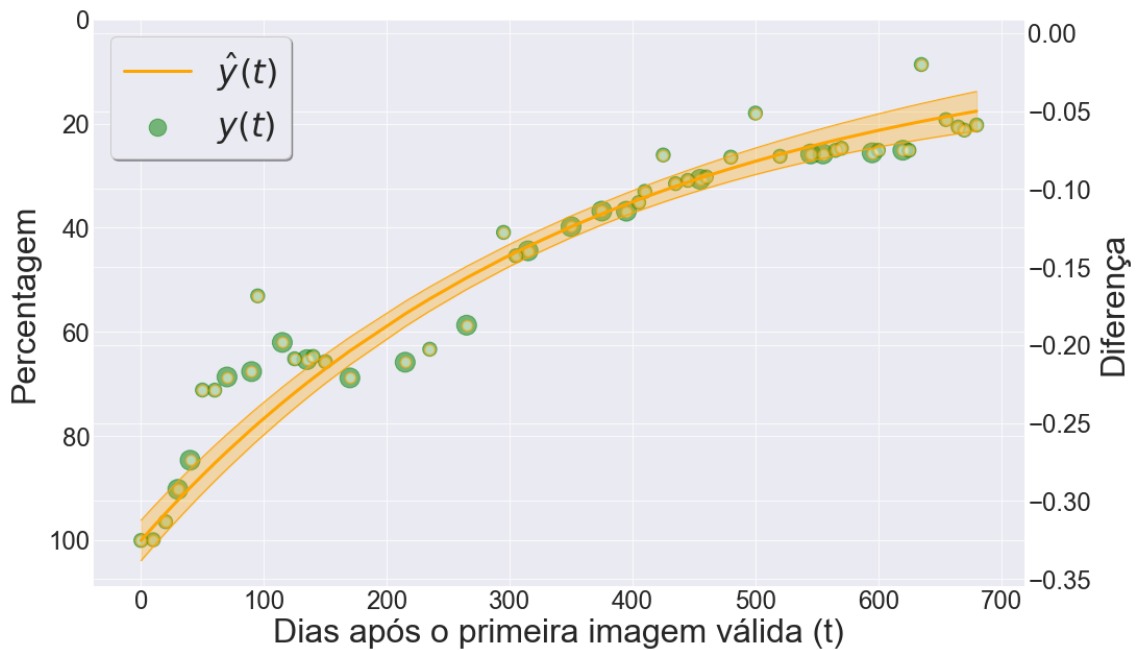


Figura 4.1: Diferenças de *NDVI*, entre a área afetada pelo fogo e a zona adjacente, para o incêndio de *Pedrogão Grande*.

São apresentados os intervalos para um nível de confiança de 95% (sombreado laranja), sendo que para este exemplo o coeficiente de determinação e o estimador foram:

$$R^2 = 0.9165, e \hat{y}(t) = -0.3253 \times e^{0.0028 \times t}$$

aplicando a Equação 3.8, resulta que

$$\ln(0.1) = 0.0028 \times t \implies t \approx 822$$

Assim, o modelo calcula que a vegetação terá recuperado quando atingido um valor de *NDVI* equivalente a 10% da diferença máxima verificada logo após o incêndio. Isto ocorrerá 822 dias após a primeira imagem válida (04-07-2017), o que corresponde ao dia 04-10-2019.

## 4.2 Mata Nacional de Leiria

A cobertura da *Mata Nacional de Leiria*, também conhecida como *Pinhal de Leiria*, é dominada por coníferas (*pinus pinaster*) cujas plantas adultas não recuperam naturalmente após a ocorrência de fogos florestais [3], está dependente da germinação de sementes localizadas no solo, abaixo da superfície, ou no topo das árvores que posteriormente caem para o solo [14].

Foi com a intervenção humana que que a *Mata Nacional de Leiria* foi constituída e, será também expectável, que seja com a ajuda humana que ela venha a recuperar o seu estado normal. Não é assim anormal que o tempo de recuperação estimado seja maior do que para *Pedrogão Grande* e que o comportamento do *NDVI* observado seja também mais errático.

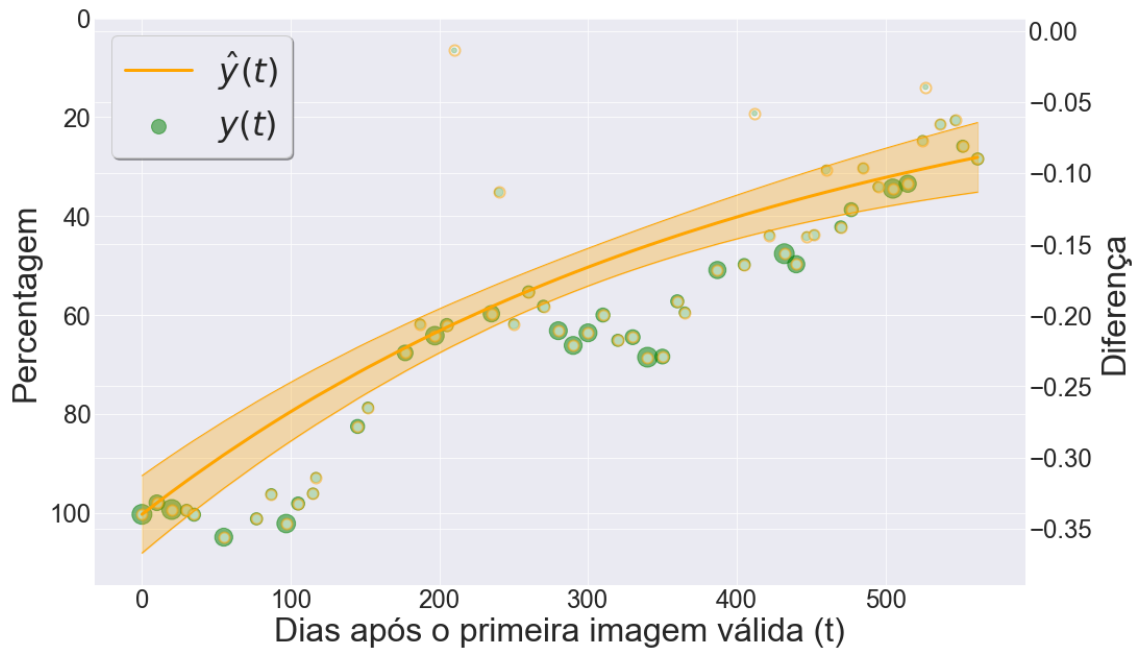


Figura 4.2: Diferenças de *NDVI*, entre a área afetada pelo fogo e a zona adjacente, para o incêndio da *Mata Nacional de Leiria*.

Obtiveram-se os resultados ilustrados na Figura 4.2 em que, à semelhança do gráfico anterior, o diâmetro dos pontos verdes, relativos ao *NDVI* observado para a vegetação, é variável em função do número de varrimentos avaliados para o composto *MVC*. São evidenciados os intervalos para um nível de confiança de 95% (sombreado laranja), sendo que para este exemplo o coeficiente de determinação

e os valores estimados foram:

$$R^2 = 0.7313, e \hat{y}(t) = -0.3401 \times e^{0.0024 \times t}$$

aplicando a Equação 3.8, resulta que

$$\ln(0.1) = 0.0024 \times t \implies t \approx 959$$

Assim, o modelo calcula que a vegetação terá recuperado quando atingido um valor de *NDVI* equivalente a 10% da diferença máxima verificada logo após o incêndio. Isto ocorrerá 959 dias após a primeira imagem válida (30-10-2017), o que corresponde ao dia 15-06-2020.

Porque o tipo de cobertura não regenera com a mesma velocidade, e o coeficiente de determinação é manifestamente mais baixo, será de esperar que a estimativa tenha uma qualidade inferior à da calculada para *Pedrogão Grande*.

Note-se ainda que, apesar da aplicação das diferenças de *NDVI* o efeito da sazonalidade não se reduz da mesma forma que em *Pedrogão Grande*. Isto acontece porque o *NDVI* observado na zona não afetada pelo incêndio é muito superior ao da zona ardida. Desta forma o efeito que se faz sentir é a predominância do comportamento da vegetação saudável.

### 4.3 Análise de sensibilidade

Os dados usados nesta análise de sensibilidade foram os mesmos usados para o caso de estudo de *Pedrogão Grande*.

Para a análise de sensibilidade foram calculados, para o incêndio de *Pedrogão Grande*, três conjuntos  $C_{alvo}$  e  $C_{ref}$  por tipo de cobertura: floresta (folhosas, coníferas e mista); mato (prado, charneca, pasto e arbustos); e as duas anteriores agregadas como uma única categoria de vegetação. Isto correspondeu à aplicação do procedimento de pré-processamento três vezes para diferentes tipos de cobertura.

Na concretização, a classe *ModeloCobertura* foi construída compondo três modelos de cobertura (floresta, mato e vegetação) com este objetivo (ver Figura 3.6) e o ficheiro gerado no pré-processamento pela classe *Modelo* inclui os dados relativos a estas séries adicionais. Obtiveram-se, assim, as séries  $NDVI(t)$  e  $NDVI^*(t)$  para os três tipos de cobertura (ver Figura 4.3 e Figura 4.4).

Utilizou-se, para comparação com o decaimento exponencial, uma regressão harmónica para um ciclo anual.

$$y(t) = b_0 + b_1 \times t + b_2 \times \sin\left(2\pi \times \frac{t}{365}\right) + b_3 \times \cos\left(2\pi \times \frac{t}{365}\right) \quad (4.1)$$

A forma gráfica destes dados encontra-se na Figura 4.3 e na Figura 4.4. o diâmetro dos pontos verdes é proporcional ao número de imagens que contribui para o compósito de MVC. Estão assinalados os intervalos para um nível de confiança de 95% (sombreado laranja).

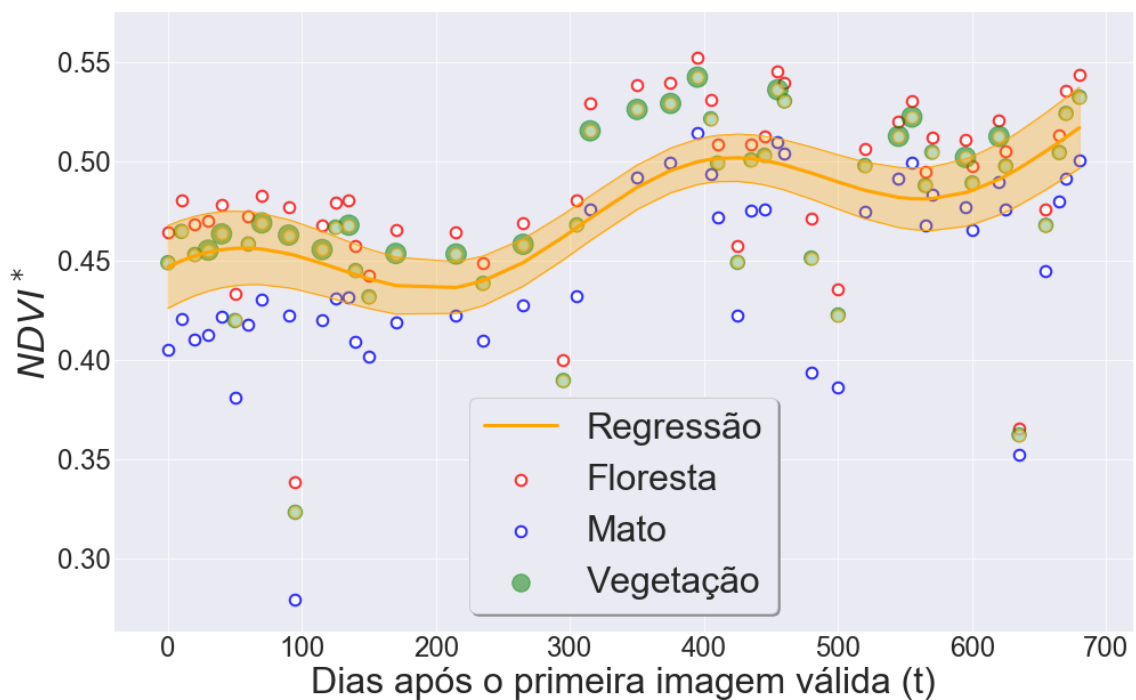


Figura 4.3: Médias de *NDVI* apuradas para a zona adjacente à geometria da área do incêndio de *Pedrogão Grande*

Os compósitos são calculados com base em quantidades de dados variáveis. O valor médio de *NDVI* da vegetação não corresponde, assim, a uma média aritmética simples dos valores determinados para as coberturas de floresta e mato.

Constata-se igualmente que, numa mesma data, os valores mais elevados de *NDVI* correspondem às zonas de floresta, o que confirma a correlação do *NDVI* com a biomassa produzida [27].

Todos os *outliers*, entendidos como os pontos mais afastados dos intervalos de confiança, estão associados a compósitos *MVC* com apenas uma imagem.

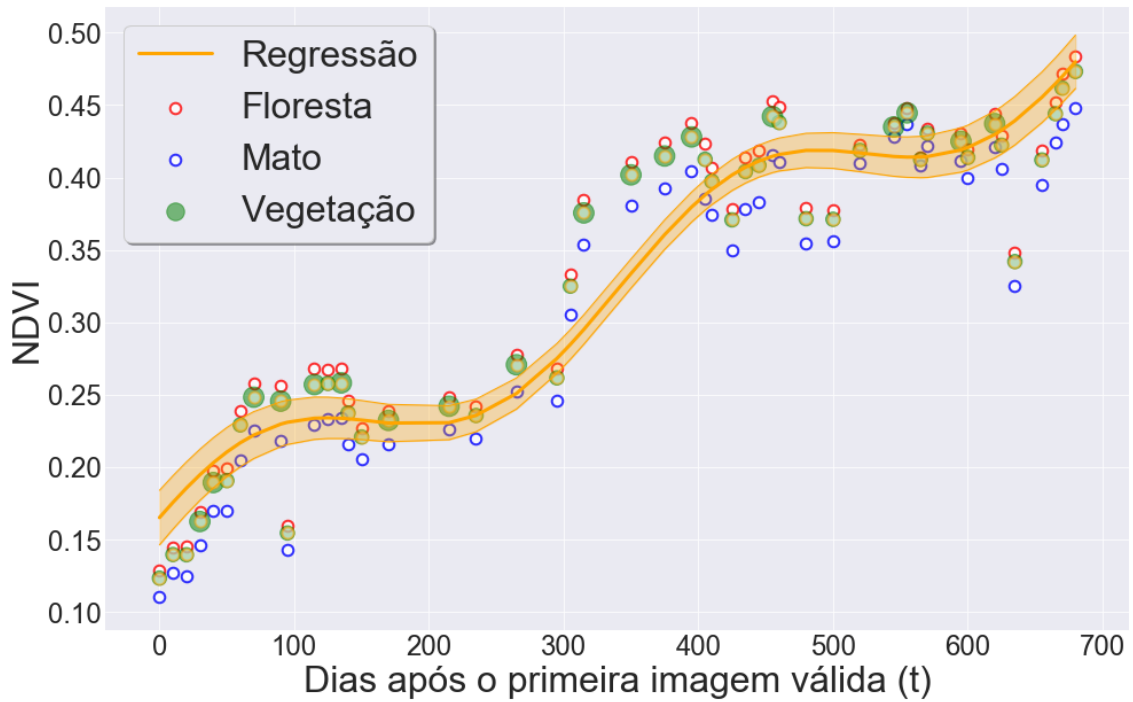


Figura 4.4: Médias de *NDVI* para a zona do incêndio de *Pedrogão Grande*

Existe coincidência no tempo na ocorrência de *outliers* em ambas as zonas e as medições foram realizadas pelos mesmos instrumentos, nas mesmas condições ambientais, tudo o mais constante. Esta relação é clara quando se recorre à Figura 4.5 e se observam os pontos que ficam fora dos intervalos de confiança assinalados. Na Figura 4.1 a influência da seca continua a verificar-se, mas os dados são mais consistentes e o efeito da sazonalidade é eliminado. Os *outliers*, apesar de persistirem, têm um efeito marginal e a dispersão é muito menor pelo que, neste caso, a aproximação usando a diferença de *NDVI* parece mais adequada.

A quantidade de *outliers* nas Figuras 4.3 e 4.4 é significativa, tendo em conta o número de amostras, o que gera alguma perturbação na estimação do *NDVI*. O coeficiente de determinação é relativamente elevado para o exemplo apresentado,  $R^2 \simeq 0.8949$ .

Embora a comparação só seja possível por analogia, verifica-se, no entanto, que o valor do coeficiente de determinação é mais elevado ( $R^2 \simeq 0.9165$ ) por aplicação da metodologia proposta aos mesmos dados (ver Seção 4.1), apontado para um melhor ajuste do estimador.

Observa-se também a existência de desvios causados por fenómenos ambientais. Note-se no exemplo a anomalia que ocorre até 300 dias após o incêndio

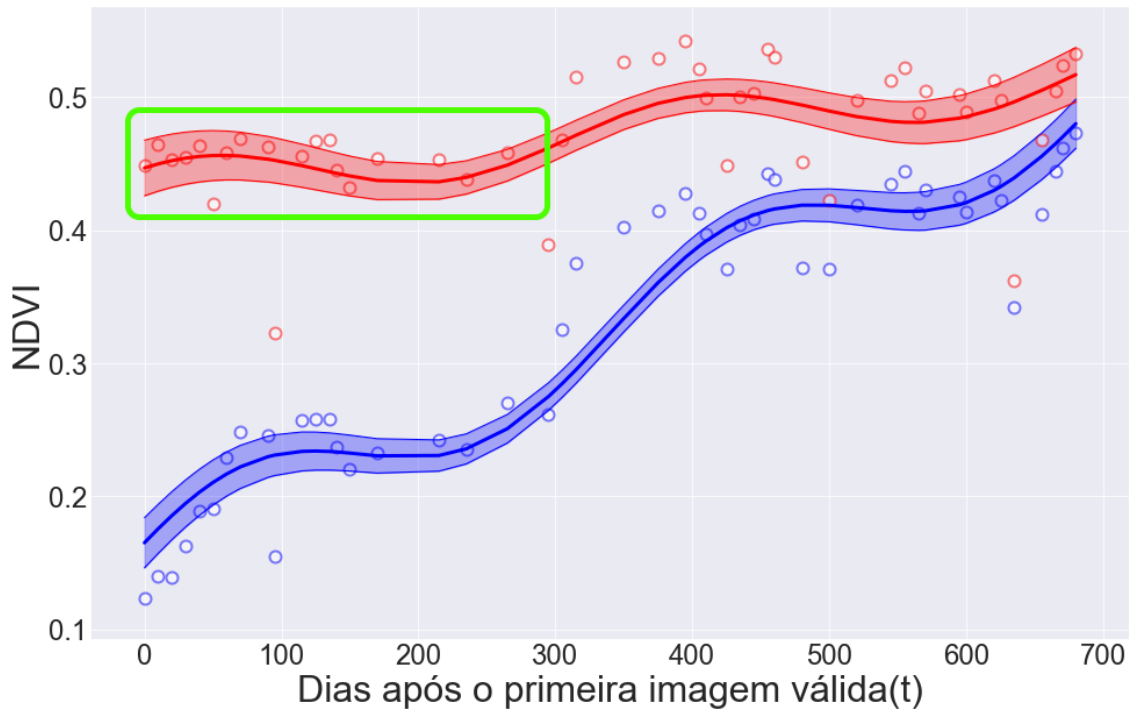


Figura 4.5: Valores de *NDVI* para a área ardida e de referência de *Pedrogão Grande*, com indicação do período de seca assinalado a verde.

consequência da seca ocorrida até Abril de 2018 [1] (ver Figura 4.5).

### 4.3.1 Atualização da estimativa para *Pedrogão Grande*

Na conclusão da redação, a 20-10-2019, procedeu-se à atualização dos dados que foram utilizados nas secções anteriores no que se refere a *Pedrogão Grande*. A atualização foi realizada, utilizando a aplicação *mvc.py* sobre os dados do ficheiro *pedrogao.ndvi* gerado a 24-05-2019. Os dados atualizados foram guardados no ficheiro *pedrogao-atual.py*<sup>2</sup>. Sobre estes dados foi executada a aplicação *lr.py* para obtenção de uma nova estimativa do tempo de recuperação e do gráfico na Figura 4.6. Nesta são apresentados os intervalos para um nível de confiança de 95% (sombreado laranja), sendo que o coeficiente de determinação e o estimador são agora:

$$R^2 = 0.92682, e \hat{y}(t) = -0.3253 \times e^{0.0026 \times t}$$

aplicando a Equação 3.8, resulta que

$$\ln(0.1) = 0.0026 \times t \implies t \approx 886$$

<sup>2</sup>Disponível no repositório localizado em <https://github.com/a44447/tese/dados>



Figura 4.6: Diferenças de *NDVI*, entre a área afetada pelo fogo e a zona adjacente, para o incêndio de *Pedrogão Grande* após atualização de dados a 20-10-2019.

Assim, o modelo calcula que a vegetação terá recuperado quando atingido um valor de *NDVI* equivalente a 10% da diferença máxima verificada logo após o incêndio, e que isto ocorrerá 886 dias após a primeira imagem válida (04-07-2017), o que corresponde ao dia 07-12-2019. Assim, o modelo com os dados atualizados, aponta agora para uma data 64 dias posterior à data calculada anteriormente na Seção 4.1 ( $\approx 8\%$  de acréscimo).

O motivo deste desvio poderá estar relacionado com condições ambientais adversas. No último período para o qual existem dados publicados, Setembro de 2018, todo o território nacional se encontrava em seca fraca a moderada [1]. Não sendo aparente que o ano de 2019 tenha sido particularmente chuvoso, e atendendo às notícias que têm sido veiculadas pelos media à data, será possível conjecturar que nos encontramos num período de seca que pode estar a atrasar a recuperação da vegetação atingida pelos incêndios de 2017.

É também possível que não se tendo concluído um ciclo fenológico completo em função da data de ocorrência do incêndio (que será Junho de 2020) poderá o efeito da sazonalidade estar a afetar a precisão do modelo de cálculo. Relacionado com este, outro motivo que poderá estar na origem deste ajustamento, será o *limiar* escolhido de 10%. Como foi referido não existe um valor fixo porque o valor de *NDVI* varia ao longo do ciclo fenológico da vegetação. Esta variação poderá ser

maior que o *limiar* escolhido. Note-se que visualmente o gráfico na Figura 4.6 pode suportar esta conjectura. Os valores nos últimos 150 dias (prazo decorrido entre 24-05-2019 e 20-10-2019) estão todos consistentemente dentro do intervalo de confiança, e abaixo da estimativa. O coeficiente de determinação aumentou ligeiramente, pelo que o erro total se reduziu.

Tabela 4.1: Valores observados em *Pedrogão Grande* desde 24-05-2019

Data MVC	%	$NDVI(t)$	$NDVI^*(t)$	$y(t)$
25-05-2019	18.6%	0.4835	0.5440	-0.0606
30-05-2019	18.6%	0.4941	0.5547	-0.0606
09-06-2019	18.3%	0.4841	0.5437	-0.0596
19-07-2019	15.4%	0.4634	0.5136	-0.0502
29-07-2019	15.6%	0.4552	0.5059	-0.0507
03-08-2019	15.5%	0.4471	0.4975	-0.0504
13-08-2019	14.5%	0.4684	0.5155	-0.0471
23-08-2019	13.5%	0.4429	0.4869	-0.0440
12-09-2019	13.8%	0.4762	0.5212	-0.0450
02-10-2019	12.9%	0.4867	0.5287	-0.0420

Na Tabela 4.1, inspecionando os valores observados desde a extração original, pode-se notar que a diferença verificada no último compósito (avaliado a 02-10-2019) é de 12,92%, próxima do limiar de 10%, o que pode apontar para que o valor escolhido para o parâmetro *limiar* poderá ser calibrado.

Finalmente outro parâmetro que poderá ser ajustado será o ciclo de cálculo dos compósitos MVC. A escolha de 10 dias poderá ser demasiado baixo, e poderá estar a ser um fator de perturbação no cálculo da regressão e, desta forma, nos parâmetros do estimador.

Não sendo já possível à data estender o âmbito do trabalho à exploração destas hipóteses, aprofundar estas questões será objeto de trabalho futuro.



## Conclusões

O estudo do tempo de recuperação da vegetação após a ocorrência de episódios de incêndios florestais é útil porque permite adotar políticas de gestão do coberto vegetal que permitam atenuar o efeito dos futuros incêndios.

A pergunta fundamental na origem deste trabalho é determinar se será possível estimar o tempo de recuperação da vegetação nas zonas queimadas, em função do tipo de cobertura vegetal, utilizando como referência o comportamento de uma zona geometricamente adjacente com o mesmo tipo de vegetação.

Para responder a esta pergunta, a metodologia proposta foi construída com base num modelo definido em trabalhos anteriores mas inova na abordagem à determinação dos valores de referência, utilizando o comportamento de áreas geometricamente adjacentes em alternativa a dados históricos da zona ardida. Recorrendo à utilização de uma ferramenta que inclui um vasto repositório de dados de deteção remota e elevada capacidade de processamento, a metodologia incorpora o cálculo de *NDVI* a partir das bandas multi-espectrais.

A concretização do modelo foi realizada utilizando o *GEE* e dados da missão *Sentinel 2* com uma resolução espacial 25 vezes superior às fontes de dados utilizadas nos trabalhos anteriores. A utilização destes instrumentos é, tanto quanto se sabe, pouco explorada constatando-se a escassez de trabalhos publicados.

A arquitetura da solução utiliza o *GEE* como repositório de dados e plataforma de processamento de dados georreferenciados. A concretização da metodologia deu origem a uma biblioteca de *Python* que dissocia a seleção, agregação e cálculo

dos valores de *NDVI*, utilizando a plataforma *GEE*, dos cálculos associados ao modelo matemático.

Explorou-se o poder de processamento e a riqueza da interface aplicacional do *GEE* em praticamente todas as suas vertentes. Foi feita a utilização de duas formas de interação com o *GEE*: (i) utilizando o *Javascript* no *Code Editor* suportado num *browser*; e (ii) desenvolvendo uma biblioteca de *Python* que interage com a *API* desenvolvida pela *Google* na mesma linguagem.

Os valores, obtidos por utilização do *GEE*, são persistidos num ficheiro estruturado usando objetos primitivos de *Python*. O ficheiro é construído com recurso à biblioteca *pickle* e inclui metadados descritivos, permitindo assim a sua utilização independente da biblioteca desenvolvida.

Na prova de conceito utilizam-se as regiões de *Pedrogão Grande* e da *Mata Nacional de Leiria* afetadas pelos incêndios de 2017. Para *Pedrogão Grande* estima-se que a vegetação recuperará 822 dias após a primeira imagem válida (04-07-2017), o que corresponde ao dia 04-10-2019, considerando o limiar de normalidade 10% da diferença máxima de *NDVI* verificada a 04-07-2017. De forma análoga estima-se que na *Mata Nacional de Leiria* a recuperação ocorrerá 959 dias após a primeira imagem válida (30-10-2017), o que corresponde ao dia 15-06-2020.

Não estando no âmbito do trabalho a discussão da natureza diferenciadora das duas zonas geográficas, e o diferente comportamento dos valores observados de *NDVI*, sabe-se no entanto que decorre da cobertura vegetal pré-existente. O facto da zona da *Mata Nacional de Leiria*, ter um período de recuperação mais longo está relacionado com o facto das coníferas não recuperarem naturalmente. Dependem da germinação natural e da replantação das árvores com intervenção humana para acelerar a regeneração da zona ardida e o controle de outras espécies invasoras [14]. Por outro lado a zona de *Pedrogão Grande* tinha como cobertura vegetal pré-existente espécies que recuperam naturalmente.

Comparou-se os resultados com a estimação do *NDVI* utilizando uma regressão harmónica e explicam-se as anomalias e *outliers* presentes nas séries temporais utilizadas nos dois cenários. Como o prazo estimado de recuperação da vegetação em *Pedrogão Grande* foi atingido no final da redação deste trabalho, procedeu-se à atualização das séries temporais e calculada uma nova estimativa do tempo de recuperação que resultou em 886 dias, o que significa que após atualização dos dados o modelo ajustou o tempo de recuperação em mais 64 dias (8% de desvio, correspondente a 07-12-2019).

De aplicação imediata, a qualquer região, o modelo proposto parece ser uma ferramenta importante que permitirá a definição de procedimentos ativos de gestão florestal como, por exemplo, em caso de regiões geridas com recurso a técnicas de fogo prescrito e outros que dependam de desflorestação planeada. A utilização de dados multi-espectrais de alta resolução permite avaliar com precisão faixas de terreno menos extensas, o que não era possível até agora com resoluções na ordem de centenas de metros.

Este trabalho deu origem a um artigo, apresentação oral e poster, com o mesmo título, na sessão *Sistemas Inteligentes de Monitorização de Recursos Naturais (SI-MRN)* do *INForum 2019 - Simpósio de Informática* [25]. Ao poster foi atribuído o prémio de *Melhor poster de estudante do INForum 2019*.

## 5.1 Trabalho futuro

A validação dos resultados poderá ainda ser desenvolvida, no enquadramento de trabalho futuro, por observação e amostragem no terreno bem como por comparação dos valores reais de *NDVI* verificados nas datas estimadas.

Embora no final do trabalho se tenha tentado validar os resultados obtidos para *Pedrogão Grande*, atualizando as séries temporais e realizando novo cálculo do tempo de recuperação, conjectura-se que o modelo poderá revelar alguma sensibilidade (i) a condições ambientais adversas, (ii) à calibração do parâmetro fundamental que é a percentagem da diferença máxima de *NDVI* verificada logo após a ocorrência do incêndio que estabelece o limiar de normalidade do nível de *NDVI* e (iii) o ciclo de avaliação dos compósitos *MVC*. A exploração destas hipóteses terá que ser objeto de trabalho futuro.

Embora tenha sido construídas aplicações cliente que produzem todos os resultados apresentados neste trabalho, será no âmbito de um novo projeto que será desenvolvida uma interface gráfica de utilizador que potencie a interação com a biblioteca desenvolvida, sem ser necessário o conhecimento de desenvolvimento em *Python*. Este trabalho terá que desenvolver a funcionalidade embrionária concretizada na classe *ZonasArdidas*, dotar a biblioteca de mecanismos assíncronos de interação com o *GEE* e a utilização de uma base de dados relacional em alternativa à geração de ficheiros.

Será também possível melhorar a biblioteca desenvolvida para incluir mecanismos mais robustos de cálculo incremental dos valores de *NDVI* e a aprendizagem

da plataforma permite também a exploração de outros cenários de estudos que envolvam a utilização de dados obtidos por detecção remota.

## Referências

- [1] APA, A. P. d. A. (2019). *Seca, relatório do estado do ambiente*. <https://rea.apambiente.pt/content/seca>. Acesso em 2019-06-04. (pp. 56 e 57)
- [2] Assembleia da República, C. T. I. (2018). *Avaliação dos incêndios ocorridos entre 14 e 16 de outubro de 2017 em portugal continental*. <https://www.parlamento.pt/Documents/2018/Marco/RelatorioCTI190318N.pdf>. Acesso a 2019-06-10. (p. 1)
- [3] Bastos, A., Gouveia, C., DaCamara, C., and Trigo, R. (2011). Modelling post-fire vegetation recovery in portugal. *Biogeosciences*, 8(12):3593–3607. (pp. 1, 7, 11, e 52)
- [4] Bossard, M., Feranec, J., Otahel, J., et al. (2000). CORINE land cover technical guide: Addendum 2000. *European Environmental Agency Technical Report No. 40*. Versão portuguesa e comentários de Fernanda Néry. (pp. 40 e i)
- [5] Departamento de Engenharia Mecânica, F.-U. C. (2017). *O complexo de incêndios de pedrogão grande e concelhos limítrofes, iniciado a 17 de junho de 2017, centro de estudos sobre incêndios florestais*. <https://www.portugal.gov.pt/download-ficheiros/ficheiro.aspx?v=3bb9773b-59fb-4099-9de5-a22fdcad1e3b>. Acesso a 2019-06-10. (p. 1)
- [6] Diaz-Delgado, R., Salvador, R., and Pons, X. (1998). Monitoring of plant community regeneration after fire by remote sensing. *Fire management and landscape ecology*, pages 315–324. (p. 7)
- [7] Drusch, M., Del Bello, U., Carlier, S., Colin, O., Fernandez, V., Gascon, F., Hoersch, B., Isola, C., Laberinti, P., Martimort, P., et al. (2012). Sentinel-2: ESA's

- optical high-resolution mission for GMES operational services. *Remote Sensing of Environment*, 120:25–36. (p. 42)
- [8] ESA (2019a). Sentinel 2 user guide. <https://earth.esa.int/web/sentinel/user-guides/sentinel-2-msi>. Acesso em 2019-01-15. (pp. 2, 3, 11, 42, e 44)
- [9] ESA (2019b). User guides - sentinel 3 synergy. <https://sentinel.esa.int/web/sentinel/user-guides/sentinel-3-synergy/product-types>. Acesso a 2019-06-24. (p. 11)
- [10] GDAL (2019). GDAL - geospatial data abstraction library. <https://www.gdal.org/>. Acesso em 2019-01-15. (p. 13)
- [11] Google (2019a). An easier way to browse, discover, and use earth engine datasets. <https://medium.com/google-earth/an-easier-way-to-browse-discover-and-use-earth-engine-datasets-77ac8bbd31de>. Acesso em 2019-01-14. (p. 16)
- [12] Google (2019b). Google earth engine guides, cliente vs. server. [https://developers.google.com/earth-engine/client\\_server](https://developers.google.com/earth-engine/client_server). Acesso em 2019-01-15. (p. 17)
- [13] Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., and Moore, R. (2017). Google earth engine: Planetary-scale geospatial analysis for everyone. *Remote Sensing of Environment*, 202:18–27. (pp. 13 e 16)
- [14] Gouveia, C., DaCamara, C., and Trigo, R. (2010). Post-fire vegetation recovery in portugal based on spot/vegetation data. *Natural Hazards and Earth System Sciences*, 10(4):673–684. (pp. 1, 6, 11, 12, 13, 35, 50, 52, e 60)
- [15] Gouveia, C., Páscoa, P., and DaCamara, C. (2018). Post-fire vegetation recovery in iberia based on remote-sensing information. In *Forest Fire*. IntechOpen. (pp. 1, 7, e 11)
- [16] Hartigan, J. A. (1975). *Clustering Algorithms (Probability & Mathematical Statistics)*. John Wiley & Sons Inc. (p. 6)
- [17] Holben, B. N. (1986). Characteristics of maximum-value composite images from temporal AVHRR data. *International journal of remote sensing*, 7(11):1417–1434. (p. 4)

- [18] ICNF (2018). Mapas ICNF. <http://www2.icnf.pt/portal/florestas/dfci/inc/mapas>. Acesso em 2018-11-10. (p. 36)
- [19] Myneni, R. B., Hall, F. G., Sellers, P. J., and Marshak, A. L. (1995). The interpretation of spectral vegetation indexes. *IEEE Transactions on Geoscience and Remote Sensing*, 33(2):481–486. (p. 4)
- [20] NASA (2018). Measuring vegetation (NDVI & EVI). [https://earthobservatory.nasa.gov/Features/MeasuringVegetation/measuring\\_vegetation\\_2.php](https://earthobservatory.nasa.gov/Features/MeasuringVegetation/measuring_vegetation_2.php). Acesso em 2018-11-05. (p. 4)
- [21] NASA (2019). NASA - TIROS. <https://science.nasa.gov/missions/tiros>. Acesso em 2019-01-15. (p. 2)
- [22] NumPy developers (2019). Numpy. <https://numpy.org>. Acesso a 2019-06-10. (p. 46)
- [23] Python Software Foundation (2019). pickle — python object serialization. <https://docs.python.org/3/library/pickle.html>. Acesso a 2019-06-10. (p. 37)
- [24] QGIS (2019). QGIS. <https://qgis.org/en/site/>. Acesso em 2019-01-15. (p. 13)
- [25] Reis, R. S., Gouveia, C., Datia, N., and Pato, M. P. M. (2019). Modelo preditivo de recuperação da vegetação afetada por incêndios florestais. In *INForum 2019 Atas do 11º Simpósio de Informática*, page 461–472. NOVA.FCT Editorial. (p. 61)
- [26] Rouse, J., Hass, R., and Schell, J. (1973). Third earth resources technology satellite (ERTS) symposium. *Monitoring vegetation systems in the great plains with ERTS*, 1:309–317. (p. 4)
- [27] Rouse Jr, J. W., Haas, R. H., Schell, J., and Deering, D. (1973). Monitoring the vernal advancement and retrogradation (green wave effect) of natural vegetation. *NASA/GSFCT Type II Report*. (pp. 3 e 54)
- [28] Salamí, E., Barrado, C., and Pastor, E. (2014). UAV flight experiments applied to the remote sensing of vegetated areas. *Remote Sensing*, 6(11):11051–11081. (p. 2)
- [29] Sellers, P. J. (1985). Canopy reflectance, photosynthesis and transpiration. *International Journal of Remote Sensing*, 6(8):1335–1372. (p. 4)



# Nomenclatura Corine Land Cover

A nomenclatura do *Corine Land Cover* [4] está organizada em três níveis. Cada código classificativo corresponde à composição de três dígitos, um por cada nível. Estão ainda previstos três códigos correspondentes a coordenadas parcial ou totalmente não caracterizadas.

O primeiro nível organiza as classificações em 5 grandes categorias, descritas em Tabela A.1.

Tabela A.1: Classificação CLC - Nível 1

Código	Descrição
1	Zonar artificializadas
2	Zonas agrícolas
3	Zonas florestais e semi-naturais
4	Zonas húmidas
5	Corpos de água

Para este trabalho utilizaram-se as classificações correspondentes ao código 3 do primeiro nível.

O segundo nível de agregação (ver Tabela A.2) corresponde a agrupamentos mais pequenos de informação e o primeiro dígito de cada código corresponde a um código de nível 1 descrito na Tabela A.1

Finalmente, a classificação atribuída a cada coordenada está na Tabela A.3.

Tabela A.2: Classificação CLC - Nível 2

Código	Descrição
11	Tecido urbano
12	Zonas industriais, comerciais ou de transportes
13	Minas, depósitos de resíduos ou zonas de construção
14	Zonas artificializadas não agrícolas com vegetação
21	Terras aráveis
22	Culturas permanentes
23	Pastagens
24	Zonas agrícolas heterogéneas
31	Florestas
32	Zonas com vegetação arbustiva e/ou herbácea
23	Zonas sem vegetação ou com vegetação esparsa
41	Zonas húmidas interiores
42	Zonas húmidas costeiras
51	Águas interiores
52	Águas costeiras

O campo código mantém a mesma lógica de organização correspondendo os dois primeiros dígitos à classificação de segundo nível (ver Tabela A.2).

Por questões relacionadas com o formato de cada coordenada, e nomeadamente o facto do mapa de cobertura ser distribuído em *GeoTIFF*, a cada código corresponde um valor entre 1 e 44 que é atribuído a cada coordenada.

Como referido anteriormente, há quatro valores que correspondem a anomalias: o valor 48 está reservado para coordenadas sem informação; os códigos 49 e 50 reservados para coordenadas das quais apenas se sabe que se trata de terra ou água, respetivamente; e, finalmente, o valor 255 é reservado para coordenadas intencionalmente não classificadas. Os códigos classificativos reservados são 999 para a ausência de dados, 990 para uma coordenada não classificada ou classificada parcialmente como terra e 995 para água.

Tabela A.3: Classificação CLC - Nível 3

Valor	Código	Descrição
1	111	Tecido urbano contínuo
2	112	Tecido urbano descontínuo
3	121	Unidades industriais ou comerciais
4	122	Rede rodoviária ou ferroviária e zonas associadas
5	123	Zonas portuárias
6	124	Aeroportos

Valor	Código	Descrição
7	131	Zonas de extracção mineira
8	132	Zonas de depósito de resíduos industriais ou urbanos
9	133	Zonas de construção
10	141	Zonas verdes urbanas
11	142	Equipamentos de desporto ou lazer
12	211	Terras aráveis não irrigadas
13	212	Terras permanentemente irrigadas
14	213	Arrozais
15	221	Vinhas
16	222	Pomares de árvores de fruto ou de baga
17	223	Olivais
18	231	Pastagens
19	241	Culturas anuais associadas a culturas permanentes
20	242	Sistemas culturais e parcelares complexos
21	243	Zonas principalmente agrícolas com zonas naturais importantes
22	244	Zonas agro-florestais
23	311	Florestas de folhosas
24	312	Florestas de coníferas
25	313	Florestas mistas de folhosas e conífera
26	321	Prados naturais
27	322	Charnecas ou matos
28	323	Vegetação esclerófila
29	324	Floresta ou vegetação arbustiva de transição
30	331	Praias, dunas ou areais
31	332	Rocha nua
32	333	Zonas de vegetação esparsa
33	334	Zonas ardidadas
34	335	Glaciares ou neves perpétuas
35	411	Pântanos ou pauis
36	412	Turfeiras
37	421	Sapais
38	422	Salinas
39	423	Zonas intermareais
40	511	Cursos de água
41	512	Planos de água
42	521	Lagoas costeiras
43	522	Estuários
44	523	Mar ou oceano



## Formato do ficheiro gerado pela classe *Modelo*

A biblioteca de *Python* que foi desenvolvida permite fazer a segregação entre o pré-processamento, realizado no *GEE*, e os cálculos que possam ser produzidos sobre os agregados. Esta funcionalidade elimina a dependência do *GEE* para tarefas que não necessitem do poder de processamento que ele garante, e que uma parte do trabalho não é condicionado pela disponibilidade de recursos do *GEE*. Um exemplo do conteúdo parcial do ficheiro gerado, em notação *JSON*, está na Listagem B.1.

---

```
{'alvo': {'escala': 100.0,
         'floresta': 20066.466666666698,
         'mato': 8096.376470588259,
         'vegetacao': 28162.843137255353},
 'comum': {'adjacente': 300,
           'data_incendio': 1497657600000L,
           'data_prineira_imagem': 1499167471880L,
           'escala': 10.0,
           'local': 'BL4171577',
           'tolerancia': 100,
           'versao': '1.0'},
 'metadados': {'alvo': ['escala', 'floresta', 'mato', 'vegetacao'],
               'comum': ['versao',
                        'local',
                        'tolerancia'],
```

```

'adjacente',
'data_incendio',
'data_prineira_imagem'],
'referencia': ['escala', 'floresta', 'mato', 'vegetacao'],
'tabela': ['Data_inicio',
'Data_fim',
'Imagens',
'Tempo_de_processamento_(microsec)',
'NDVI_alvo_floresta_m\xc3\xa9dia',
'NDVI_refer\xc3\xaancia_floresta_m\xc3\xa9dia',
'NDVI_alvo_mato_m\xc3\xa9dia',
'NDVI_refer\xc3\xaancia_mato_m\xc3\xa9dia',
'NDVI_alvo_vegeta\xc3\xa7\c3\xa3o_m\xc3\xa9dia',
'NDVI_refer\xc3\xaancia_vegeta\c3\xa7\c3\xa3o_m\c3\xa9dia',
'NDVI_alvo_floresta_mediana',
'NDVI_refer\xc3\xaancia_floresta_mediana',
'NDVI_alvo_mato_mediana',
'NDVI_refer\xc3\xaancia_mato_mediana',
'NDVI_alvo_vegeta\c3\xa7\c3\xa3o_mediana',
'NDVI_refer\xc3\xaancia_vegeta\c3\xa7\c3\xa3o_mediana',
'NDVI_alvo_floresta_pontos',
'NDVI_refer\xc3\xaancia_floresta_pontos',
'NDVI_alvo_mato_pontos',
'NDVI_refer\xc3\xaancia_mato_pontos',
'NDVI_alvo_vegeta\c3\xa7\c3\xa3o_pontos',
'NDVI_refer\xc3\xaancia_vegeta\c3\xa7\c3\xa3o_pontos']},
'referencia': {'escala': 100.0,
'floresta': 3360.556862745086,
'mato': 1178.3450980392124,
'vegetacao': 4538.901960784285},
'tabela': array([[1.49916747e+12, 1.49916747e+12, 1.00000000e+00, ...,
1.57431071e+05, 3.43167400e+06, 6.05335133e+05],
[1.50003127e+12, 1.50003127e+12, 1.00000000e+00, ...,
1.57343239e+05, 3.60396600e+06, 6.05334573e+05],
[1.50089547e+12, 1.50089547e+12, 1.00000000e+00, ...,
1.57311643e+05, 3.60919700e+06, 6.05094690e+05],
...,
[1.55662382e+12, 1.55662382e+12, 1.00000000e+00, ...,
1.57887800e+05, 3.66626600e+06, 6.04975882e+05],
[1.55705582e+12, 1.55705582e+12, 1.00000000e+00, ...,
1.57986078e+05, 3.69791900e+06, 6.07453788e+05],
[1.55791982e+12, 1.55791982e+12, 1.00000000e+00, ...,
1.59000306e+05, 3.69826700e+06, 6.10281965e+05]])}

```

---

Listagem B.1: Conteúdo parcial de um ficheiro gerado pela classe *Modelo*

De forma a auxiliar o utilizador, a classe *Composito* expõe um conjunto de constantes públicas que indexam os elementos da matriz *tabela* (ver Listagem B.2).

---

```
# Campos de cada registo
DATA_INICIO = 0
DATA_FIM = 1
IMAGENS = 2
TEMPO_PROCESSAMENTO = 3
NDVI_MEDIA_FLORESTA_ALVO = 4
NDVI_MEDIA_FLORESTA_REFERENCIA = 5
NDVI_MEDIA_MATO_ALVO = 6
NDVI_MEDIA_MATO_REFERENCIA = 7
NDVI_MEDIA_VEGETACAO_ALVO = 8
NDVI_MEDIA_VEGETACAO_REFERENCIA = 9
NDVI_MEDIANA_FLORESTA_ALVO = 10
NDVI_MEDIANA_FLORESTA_REFERENCIA = 11
NDVI_MEDIANA_MATO_ALVO = 12
NDVI_MEDIANA_MATO_REFERENCIA = 13
NDVI_MEDIANA_VEGETACAO_ALVO = 14
NDVI_MEDIANA_VEGETACAO_REFERENCIA = 15
NDVI_PONTOS_FLORESTA_ALVO = 16
NDVI_PONTOS_FLORESTA_REFERENCIA = 17
NDVI_PONTOS_MATO_ALVO = 18
NDVI_PONTOS_MATO_REFERENCIA = 19
NDVI_PONTOS_VEGETACAO_ALVO = 20
NDVI_PONTOS_VEGETACAO_REFERENCIA = 21
```

---

Listagem B.2: Constantes definidas na classe *Composito*



## Exemplo de produção de gráfico das diferenças de *NDVI*

Um exemplo de utilização de um ficheiro gerado na fase de pré-processamento para obter um gráfico representativo das diferenças de *NDVI* entre a área alvo e a área de referência. Cada ponto do gráfico terá um diâmetro variável em função do número de imagens que contribuíram para o compósito. Este *script* pode ser consultado no repositório<sup>1</sup>.

---

```
import pickle
import numpy as np
import matplotlib.pyplot as plt
from recveg import Composito, Utilidades

# =====
# Constantes
# =====
# Dimensao das fontes de texto
FONTE_NORMAL = 32
FONTE_PEQUENA = 24
# Espaco entre elementos
ESPACO = 24

def executa_calculo(nome_ficheiro):
    """
```

---

<sup>1</sup>Localizado no repositório em [http://github.com/a44447/tese/lr\\_diferenca.py](http://github.com/a44447/tese/lr_diferenca.py)

Algoritmo de produção do gráfico das diferenças de NDVI em função da diferença máxima observada

```
"""
ficheiro = open(nome_ficheiro, "rb")
modelo = pickle.load(ficheiro, encoding='latin1')
dados = modelo["tabela"]

tempo = dados[:, Composito.DATA_FIM].reshape(-1, 1)
tempo = (tempo - tempo[0]) / Utilidades.DIA
diametro = float(364) / \
    (float(max(dados[:, Composito.IMAGENS]) - \
        min(dados[:, Composito.IMAGENS]))+1)
diferenca = np.array(\
    dados[:, Composito.NDVI_MEDIA_VEGETACAO_ALVO] - \
    dados[:, Composito.NDVI_MEDIA_VEGETACAO_REFERENCIA]) \
    .reshape(-1, 1)
diferenca_maxima = diferenca[0]

plt.figure(1, figsize=(16, 10))
plt.style.use("seaborn-darkgrid")
plt.xlabel(u"Dias_apos_a_primeira_imagem_(t)", \
    fontsize=FONTE_NORMAL)
plt.ylabel(u"$\frac{NDVI(t)-NDVI^*(t)}{a}$", \
    fontsize=FONTE_NORMAL)
plt.xticks(fontsize=FONTE_PEQUENA)
plt.yticks(fontsize=FONTE_PEQUENA)

plt.scatter(tempo, \
    diferenca/diferenca_maxima, \
    c="green", \
    s=dados[:, Composito.IMAGENS] * diametro, \
    alpha=0.5, \
    label=u"Vegetacao")
plt.show()

# Atualizar com o nome do ficheiro que se pretende utilizar
executa_calculo("dados/pedrogao.ndvi")
```

---

Listagem C.1: Função que gera um gráfico com as diferenças de NDVI



## Artefactos de produção de figuras e resultados

Todas as figuras e resultados que foram produzidos para este trabalho com a utilização do *GEE* estão suportadas em artefactos de código. Na Tabela D.1 apresentam-

Tabela D.1: Artefactos de produção de figuras e resultados

Referência		Dados	Localização
Biblioteca	-	-	javascript/lib_fontes.js
Capa	Sim	-	javascript/capa.js
Figura 1.2	Sim	-	javascript/mosaico_leiria.js
Figura 2.1	Não	pedrogao.ndvi	lr_series.py
Figura 3.1	Sim	-	javascript/geometrias.js
Figura 3.2	Sim	-	javascript/coberturas.js
Figura 3.3	Sim	-	javascript/mvc_nuvens.js
Figura 4.1	Não	pedrogao.ndvi	lr.py
Figura 4.2	Não	leiria.ndvi	lr.py
Figura 4.3	Não	pedrogao.ndvi	lr_sensibilidade.py
Figura 4.4	Não	pedrogao.ndvi	lr_sensibilidade.py
Figura 4.5	Sim	pedrogao.ndvi	lr_series.py
Figura 4.6	Não	pedrogao-atual.ndvi	lr.py
Tabela 4.1	Não	pedrogao-atual.ndvi	lr_adenda.py

se as fontes que produzem cada uma das imagens. Em todos os artefactos encontra-se indicada a localização relativa no repositório<sup>1</sup>, onde poderão ser consultados. Algumas das figuras tiveram que ser compostas numa aplicação de processamento de imagem, para serem enquadradas na formatação do documento, sem que o conteúdo fosse de alguma forma adulterado. Estas têm o valor *Sim* na segunda coluna. Estão também indicados na Tabela D.1 os ficheiros de dados utilizados, quando aplicável. Estes últimos encontram-se na pasta *dados* do repositório, embora também eles fossem reproduzíveis usando a aplicação *mvc.py*.

---

<sup>1</sup>Repositório localizado em <https://github.com/a4447/tese>