

Experiências na deteção de incêndios com o Google Earth Engine

Eduardo Fernandes *

Orientadores:

Professor Fernando Birra†

Professor João Moura Pires ‡

Departamento de Informática, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa

Resumo

Os incêndios trazem consigo inúmeras consequências negativas para as zonas que afetam, que podem perdurar por vários anos. Para minorar os aspetos negativos é necessário um bom plano de recuperação que adapte a resposta às necessidades de cada zona. Este processo passa por realizar um balanço da área ardida.

Em Portugal o balanço das áreas ardidas é realizado manualmente através da análise de imagens de satélite. Isto causa alguma demora na obtenção de resultados, e gera erros uma vez que a análise de uma imagem tem associada sempre alguma subjetividade.

Uma forma alternativa de realizar este balanço consiste na utilização de algoritmos que, a partir de dados, nos possam dar informações sobre a área ardida. Existem várias ferramentas para implementar estes algoritmos. Google Earth Engine é apelativo uma vez que este oferece simultaneamente o acesso aos dados de satélite de forma integrada e expedita, o acesso a recursos computacionais elásticos e uma API de alto nível para este tipo de dados e de processamento.

O presente projeto de investigação tem como objetivo avaliar o Google Earth Engine (GEE) num contexto de uma aplicação real (a deteção de área ardida). Foram por isso realizadas várias soluções que classificassem are ardida e não ardida usando a plataforma. A experiência servirá como base para fazer uma avaliação do GEE, na implementação de soluções de *remote sensing*.

Conclui-se que o GEE é de facto uma boa ferramenta, na área da deteção remota por livrar o utilizador da gestão e armazenamento de dados e por fornecer APIs capazes de processar uma enorme quantidade de dados remotamente.

Keywords: Incêndios, Área Ardida, Google Earht Engine, Machine Learning

1 Introdução

No ano de 2017 os incêndios reclamaram a vida de mais de 100 pessoas [Branco and Porto 2017] e queimaram mais de 418 mil hectares [de Gestão de Áreas Públicas e de Proteção Florestal 2017]. Parte importante do processo de recuperação passa por fazer um balanço da área ardida. Em Portugal o Instituto da Conservação da Natureza e das Florestas (ICNF) é responsável por realizar este levantamento, que até à data é feito de forma manual, trazendo assim dois problemas associados: demora na produção de resultados e o

*efa.fernandes@campus.fct.unl.pt

†e-mail: fpb@fct.unl.pt

‡e-mail: jmp@fct.unl.pt

erro humano. Uma forma de minimizar estes dois aspetos é recorrer a técnicas de *remote sensing* para detetar área ardida.

Em 2010¹ a Google tornou pública a plataforma Google Earth Engine (GEE)² [Gorelick et al. 2017], que coloca a computação e os dados do lado dos servidores da Google. Por este motivo não é necessário descarregar dados, nem ter hardware poderoso para realizar as computações que queremos executar. Isto vem facilitar a análise de dados de satélite ao fornecer APIs de alto nível em Python e em Javascript para comunicar com a plataforma.

O presente projeto de investigação tem dois objetivos. Por um lado, usar o GEE na deteção de área ardida em Portugal, e por outro avaliar o GEE nesse contexto.

Inicialmente o relatório aborda como é feita a deteção de área ardida até ao momento. Seguidamente é apresentada a plataforma Google Earth Engine. Depois desta visão geral chega a altura de apresentar a forma como o trabalho foi abordado, bem como apresentar o trabalho realizado. Finalmente há uma secção onde se discute os resultados obtidos e possível trabalho futuro.

2 Estado da arte

A deteção de área ardida assenta na deteção de resíduos no solo e na deteção de alterações na vegetação. Após um incêndio o solo fica coberto de cinza. Porém, esta rapidamente se dissipa com as condições climatéricas. A vegetação também sofre alterações, contudo demora mais tempo a regressar à normalidade. Estas mudanças têm efeitos característicos na assinatura eletromagnética da zona. Para salientar as perturbações típicas de uma área ardida, há um conjunto de índices já estudados e testados [Huang et al. 2016], [Chen et al. 2016], [Santos 2018].

- Normalized Difference Vegetation Index (NDVI)
- Enhanced Vegetation Index (EVI)
- Soil Adjusted Vegetation Index (SAVI)
- Moisture Stress Index (MSI)
- Burned Area Index (BAI)
- Burn Ratio (BR)

Este último conta com algumas variantes, NBR, NBR1, NBR2, NBR3 e NBR4. Para além dos próprios índices, também se pode utilizar a sua variação ao longo do tempo. O NDVI é um dos índices mais conhecidos e mais usados. Este tira partido da diferença de absorção da luz no infravermelho e no vermelho por diferentes tipos de vegetação. Está definido da seguinte forma:

$$NDVI = \frac{NIR - RED}{NIR + RED}$$

Em anexo encontra-se uma lista de como são calculados os índices apresentados.

¹<http://blog.google.org/2010/12/introducing-google-earth-engine.html>

²<https://earthengine.google.com>

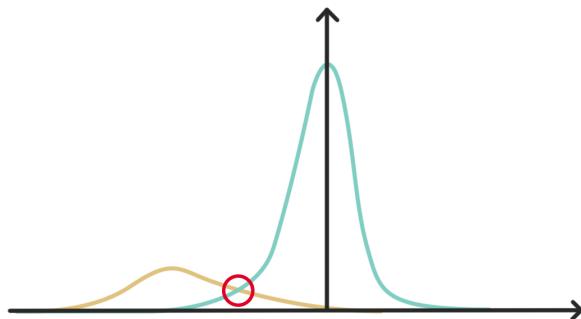


Figura 1: Exemplo de histograma de frequência para um índice

No estudo [Chen et al. 2016] são utilizadas imagens do projeto Landsat, a partir das quais são calculados os índices (NDVI, EVI, VFC, DI) relativos a cada pixel e, com base na escolha de um limite, classifica-se uma área como ardida ou não ardida. Para escolher estes limites é realizada uma amostragem estratificada de pixels das imagens, com base na classificação de área ardida, área não ardida. A partir destes dados geram-se histogramas de frequência do índice em questão como o observado na figura 1.

Através da análise do histograma é escolhido o intervalo onde se encontra o limite que maximiza a precisão dos resultados. Este intervalo encontra-se na intersecção das duas curvas salientada na figura 1 por um círculo vermelho. Posteriormente é elaborado um gráfico que mostra como varia a *accuracy* dos resultados no intervalo selecionado. Finalmente, é escolhido o valor que maximiza *accuracy*. Esta abordagem revela-se simples e fácil de implementar e foi por isso escolhida como o ponto de partida para o corrente trabalho.

Noutros estudos como [Brewer et al. 2005] [Santos 2018] utilizam-se técnicas de aprendizagem automática para classificação de área ardida. Mais concretamente [Santos 2018], aborda o comportamento de classificadores aliados ao processamento paralelo para perceber se estes são apropriados para o uso em tarefas de previsão em florestas. O trabalho usa como área de estudo uma zona em Castelo de Paiva, Portugal. São comparadas três fontes de imagens de satélite: MOODIS, Landsat e Sentinel. Prova-se que os melhores resultados são alcançados a partir de dados do Sentinel-2 devido à sua superior resolução espacial. A partir das imagens são calculados vários índices, posteriormente é utilizada uma regressão LASSO [Tibshirani 1996] para selecionar os mais relevantes. Os índices selecionados são usados como *features* de entrada para os classificadores testados, a saber: Gradient Boosted Trees, Support Vector Machines, K-Nearest Neighbours e Artificial Neural Networks [Gron 2017]. Para cada classificador há uma versão standard, que corre em CPU, e uma outra versão acelerada, que corre em paralelo num GPU. Conclui-se que as versões aceleradas são de facto mais rápidas e não apresentam diferenças significativas nos resultados. Isto revela-se como uma mais valia porque o Google Earth Engine tira partido de um ambiente de computação intrinsecamente paralelo.

Para este trabalho serão usadas imagens do Sentinel-2, uma vez que esta missão apresenta melhores resultados em tarefas de classificação quando comparado com outras como Landsat e MOODIS [Zhu et al. 2015] [Santos 2018]. As imagens obtidas pelo Sentinel-2 são compostas por 13 bandas (tabela 1) do espectro eletromagnético. A resolução varia consoante a banda, sendo a maior resolução de 10m x 10m e a menor resolução de 60m x 60m. Esta missão tem frequentes revisitas, a latitudes médias obtendo-se imagens a cada dois a três dias enquanto que no equador são necessários dez dias [SUHET 2013].

Tabela 1: Bandas Sentinel-2

Resolução	Número da banda	Comprimento de onda central (nm)	Descrição
10m x 10m	2	490	Azul
	3	560	Verde
	4	665	Vermelho
	8	842	NIR
20m x 20m	5	705	Red Edge 1
	6	740	Red Edge 2
	7	783	Red Edge 3
	8b	865	Red Edge 4
	11	1610	SWIR 1
60m x 60m	12	2190	SWIR 2
	1	443	Aerossóis
	9	945	Vapor de água
	10	1375	Cirrus

2.1 Google Earth Engine

O Google Earth Engine [Gorelick et al. 2017] é uma plataforma na *cloud* que tem como objetivo a análise de dados geo-espaciais. Esta análise é realizada criando scripts, em Python ou em Javascript, que em parte correm no lado do cliente e em parte no lado dos servidores. O desenvolvimento de scripts em Python requer a instalação do interpretador de Python bem como todas as dependências necessárias para utilizar o Google Earth Engine. Para desenvolver em JavaScript é apenas necessário um browser, isto porque a Google disponibiliza um ambiente de desenvolvimento interativo baseado em tecnologias web. A utilização do GEE requer a inscrição na plataforma seja primeiro aceite, depois disso poder-se-á usufruir gratuitamente deste serviço.

O GEE tem todas as ferramentas necessárias para detetar área ardida. Permite o acesso a vários *datasets*, dos quais se destacam os seguintes projetos: LandSat, Sentinel e MOODIS. É importante referir que todas estas fontes de informação são públicas e de acesso livre. Também é possível um utilizador importar as suas próprias coleções de imagens. Existem essencialmente dois tipos de dados no GEE: imagens (dados raster) e *features* (dados vetoriais). Estes dois tipos de dados encontram-se agrupados em coleções que podem ser filtradas por data e zona e até reduzidas a uma só imagem, calculando valores como o valor médio ou máximo num pixel. As imagens são conjuntos de bandas. As bandas podem representar inúmeros conceitos como a intensidade de luz no infravermelho, um índice como o NDVI ou uma classificação (queimado/ não queimado) isto porque numa mesma imagem diferentes bandas podem ter diferentes resoluções e tipos de dados. A uma imagem é possível, adicionar e remover bandas, realizar cálculos pixel a pixel com bandas para obtermos índices, somar, subtrair, e dividir por outras imagens e recortar segundo um polígono, entre muitas outras. Isto viabiliza o cálculo da variação de índices na zona do incêndio.

Por fim é necessário referir que a plataforma fornece uma lista de 16 classificadores já implementados que facilmente podemos utilizar para classificar área ardida.

O editor de código online³ para desenvolver scripts em JavaScript oferece um ambiente interativo onde é possível mapear as imagens geradas durante a execução dos scripts, gerar gráficos, aceder a documentação, ter repositórios de código facilmente partilhados com outros utilizadores e inclusive gerar interfaces simples de modo a tornar um script interativo. Por todas estas razões este foi o ambiente de programação escolhido para desenvolver o trabalho.

³<https://code.earthengine.google.com>

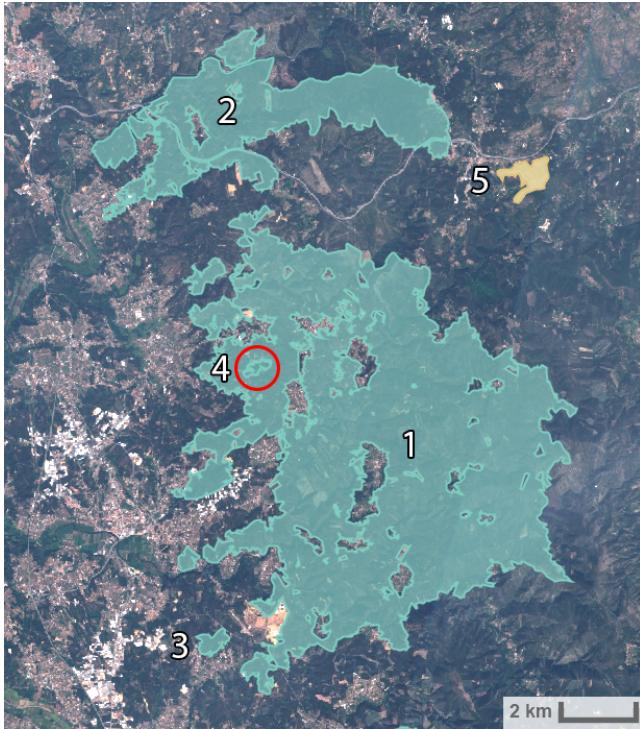


Figura 2: Zona escolhida com as áreas ardidas ardidas coloridas

3 Visão geral da abordagem seguida

Selecionou-se o ano de 2016 para realizar este trabalho. Nesse ano o concelho de Águeda no distrito de Aveiro, ficou marcado pelos incêndios que, segundo os dados do ICNF, ali devastaram um total de 9067 ha. Esta zona foi então escolhida como área de estudo, que se apresenta na Figura 2) que englobasse a totalidade da área ardida. A zona tem 40627 ha, sendo que ardeu um total de 8991 ha, 23% de área ardida. Houve um total de cinco incêndios na zona. Nos dados há informação da data de início e de conclusão para quatro deles (incêndios 1,2,3 e 4), a área do incêndio sem data (incêndio 5) não foi considerada (figura 2).

Ao longo do trabalho são implementados vários métodos que classificam área ardida e não ardida. Precisamos de ferramentas que nos ajudem a discriminar as melhores classificações quando comparadas com os dados de referência do ICNF. Vamos tomar estes dados como totalmente corretos, sendo a nossa *ground truth*. Segundo [Foody 2002] a análise de mapas de classificação usa como principal métrica uma matriz de confusão, em conjunto com outros índices que podem ser extraídos da mesma tal como a *accuracy* e o *kappa*.

Uma matriz de confusão é uma tabela gerada para avaliar um conjunto de classificações. No caso específico de classificar área ardida temos 2 classes, área ardida ou não ardida, o que dá origem à tabela 2. A matriz de confusão dá-nos uma ideia dos pontos fracos do método de classificação, compreendendo se por exemplo a implementação tem tendência a gerar mais falsos positivos do que outra. Serão usadas matrizes de confusão que usam percentagens em vez de frequências para facilitar a comparação.

Accuracy é a métrica mais intuitiva e simples, pois apenas divide o número de classificações corretas pelo número total de classificações (To).

$$Accuracy = \frac{VP + VN}{To}$$

Porém, quando aplicada à zona de estudo, um classificador que constantemente classificasse todos os pixels como não ardidos teria uma *accuracy* de 77% uma vez que esta é a percentagem de área não ardida. Esta métrica não tem em conta a possibilidade de a classificação ter sido conseguida por sorte. Segundo [Ben-David 2008] o *Kappa* de Cohen (*kappa*) tenta corrigir o grau de concordância subtraíndo a porção de classificações corretas que possam ter sido geradas por acaso.

O *kappa* está definido como:

$$K = \frac{p_0 - p_e}{1 - p_e}$$

onde p_0 é a probabilidade total de concordarem e P_e é a probabilidade de concordarem por sorte.

Esta métrica varia de -1 (total discordia) passando por 0 (classificação aleatória) até 1 (total concordância) [Ben-David 2008]. Sendo que valores superiores a 0.7 já são considerados como bons.

Todas estas ferramentas são úteis para comparar métodos de classificação, mas perde-se demasiada informação no processo ao reduzir uma imagem a apenas alguns números. Estes números não conseguem mostrar como se distribui especialmente o erro. Para este tipo de detalhe é necessário a análise qualitativa das imagens classificadas comparadas com a classificação oficial. Para ajudar a esta tarefa, optou-se por criar imagens de confusão que utilizam o código de cores definido na tabela 2. Podemos observar um exemplo deste tipo de imagem na figura 5. A área ardida estará sempre com cor castanha enquanto que área não ardida terá cor verde. Caso a zona em questão tenha sido classificada de forma incorreta a cor será menos saturada, caso contrário a cor será mais saturada.

Tabela 2: Código de cores para as imagens classificadas

		Referencia	
		Ardida	Não Ardida
Classificação	Ardida	(VP) Verdadeiro Positivo	(FP) Falso Positivo
	Não Ardida	(FN) Falso Negativo	(VN) Verdadeiro Negativo

4 Trabalho desenvolvido

Nesta secção será apresentado o trabalho desenvolvido ao longo do semestre. Serão descritas as várias experiências realizadas assim como os resultados alcançados. O trabalho foi desenvolvido de forma iterativa começando com uma base simples mas funcional, com o principal objetivo de compreender a plataforma. A partir do script inicial foram sendo acrescentadas novas funcionalidades umas vezes por necessidade outras apenas para testar hipóteses.

Para poder avaliar os métodos de classificação é necessária uma base com a qual comparar os resultados obtidos. O ICNF disponibiliza um ficheiro no formato shapefile que contem polígonos com os incêndios relativos a cada ano⁴. Dos cinco incêndios que ocorreram na zona de estudo um não tem nem data de início nem de conclusão. Por este motivo os dados relativos à zona afetada por este incêndio foram descartados.

Como fonte de dados foi utilizada a coleção de imagens do Sentinel-2 disponibilizada pelo GEE⁵. Para importar a coleção é apenas necessária uma linha de código.

⁴<http://www2.icnf.pt/portal/florestas/dfci/inc/info-geo>

⁵<https://code.earthengine.google.com/dataset/COPERNICUS/S2>

Na base de todas as experiências está a criação de duas imagens a partir das quais serão extraídos os índices a serem usados para a classificação. Uma resulta da média das imagens Sentinel-2 disponíveis dez dias antes do primeiro incêndio ter começado e outra resulta da média das imagens disponíveis dez dias depois do último incêndio ter terminado. A média é realizada pixel a pixel para todas as bandas das imagens em questão.

4.1 Classificação com base num limite da diferença de NDVI

Numa primeira fase e com o objetivo de compreender melhor a plataforma foi criado um script simples que classificava uma zona como ardida ou não ardida com base na diferença de NDVI que esta apresentava através de um limite baseado no trabalho realizado em [Chen et al. 2016]. Após calcular o delta NDVI com base nas duas imagens, foi gerado o histograma da figura 3, que nos mostra que é possível separar as duas classes com base num limite e que este se situará na intersecção das duas curvas. Seguidamente foi calculada a *accuracy* para vários limites e os resultados obtidos encontram-se no gráfico na figura 4, levando a escolha de -0.24 como limite. Foi obtida uma *accuracy* de 0.9229 e um kappa igual a 0.7719. É possível observar a matriz de confusão na tabela 3. Todo o processo, inclusive a criação de gráficos, é realizado no ambiente interativo de programação online.

Tabela 3: Matriz de confusão da classificação o melhor limite para o delta NDVI

		Ground Truth	
		Ardido	Não Ardido
Classificação	Ardido	17,7%	2,8%
	Não Ardido	4,9%	74,6%

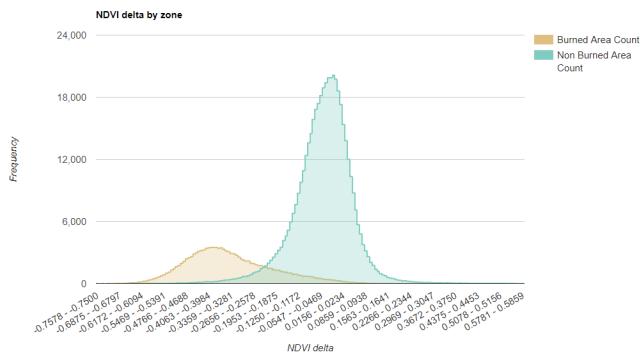


Figura 3: Histograma de frequência do delta NDVI em zonas ardidas e não ardidas

4.2 Classificação com base em limites para outros índices

O NDVI não é o melhor índice no que diz respeito à deteção de área ardida uma vez que a colheita de uma cultura também gera uma diferença grande de NDVI podendo assim criar um falso positivo. Posto isto, o próximo passo envolveu utilizar outras diferenças de índices. Foram selecionados EVI, NBR e NBR2 mantendo a escolha manual de um limite. Repetindo o processo descrito na secção anterior obtemos os resultados que se encontram na tabela 4. Os melhores resultados foram obtidos usando o NBR, enquanto que EVI ao contrário do que acontecia em [Chen et al. 2016] foi

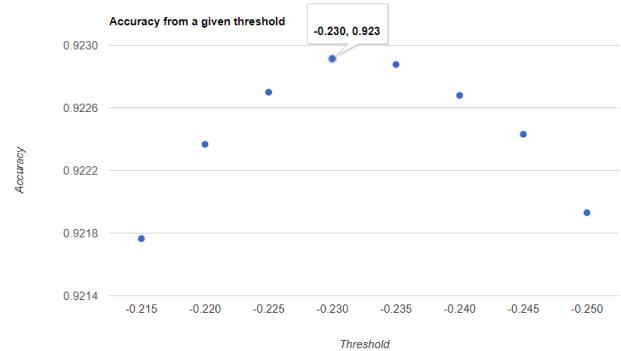


Figura 4: Variação de accuracy consoante o limite selecionado

o pior. É possível que isto se deva às diferentes características das duas regiões.

Tabela 4: Comparação de classificação que usam os melhores limites para cada índice

Índice	Limite	Accuracy	Kappa
NBR1	-0,240	0,9236	0,7768
NDVI	-0,240	0,9229	0,7719
NBR2	-0,685	0,8942	0,6913
EVI	-0,235	0,8908	0,6824

4.3 Classificadores

A escolha manual de índices é claramente o ponto fraco da abordagem até este ponto uma vez que apenas tem em consideração um índice de cada vez, uma solução possível seria fazer uma média ponderada das classificações dos índices por exemplo. Porém uma melhor solução seria utilizar aprendizagem automática para classificar as áreas ardidas. O GEE disponibiliza uma variedade de classificadores já implementados que nos podem ajudar resolver este problema, com muito pouco esforço. Para declarar e treinar um classificador é apenas necessário uma linha de código assumindo que já temos os dados de treino numa variável *trainingData*.

```
var classifier = ee.Classifier.cart()
  .train(trainingData, classProperty);
```

O GEE disponibiliza 16 classificadores, é por isso necessário fazer uma seleção dos mais interessantes para a tarefa em questão. Para realizar esta seleção foram testados todos os classificadores disponíveis sem qualquer alteração aos parâmetros por defeito, usando os mesmo dados para todos 14974 pontos (3%) dos dados disponíveis e 14966 pontos (3%) para validação. Esta implementação não garante que hajam dados de treino que também se encontrem nos dados de validação. O tamanho destes datasets deve-se ao facto do tempo de computação e a memória que o GEE disponibiliza ser limitada. Uma vez que não é preciso deferir limites manualmente para cada índice podemos facilmente adicionar mais índices para os classificadores terem em conta. Para este teste foram usados NDVI, MIRBI, NBR, BAI, pois na literatura estes indicadores revelam-se importantes. É ainda utilizado o NBR4 por se revelar promissor em [Santos 2018]. Os classificadores recebem como input o todos os índices antes, depois do incêndio e a diferença perfazendo um total de 15 *features*.

Os resultados encontram-se na tabela 5 e na figura 6. Não foi possível obter resultados dos seguintes classificadores: gmoLine-

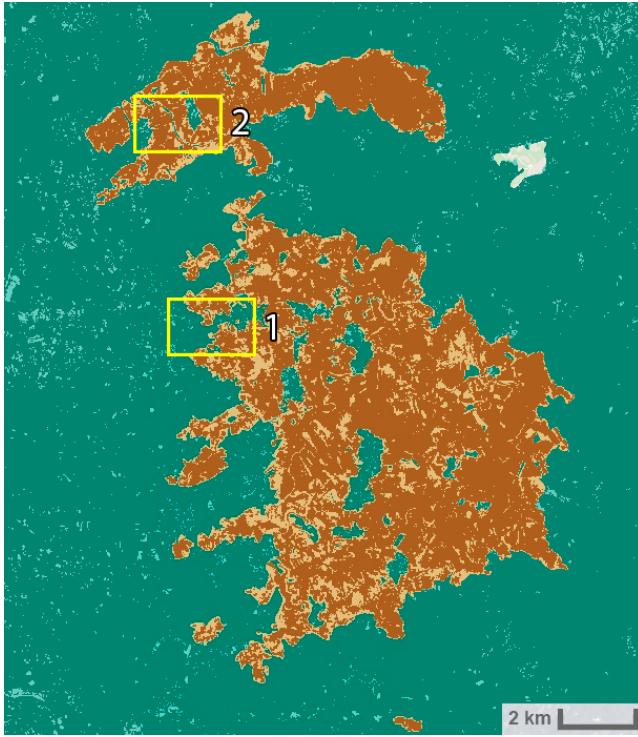


Figura 5: Imagem classificada usando o melhor limite para o delta NDVI

arRegression, ikpamir, pegasosGaussian, pegasosLineare pegasosLinear. O primeiro apenas realiza regressão linear e os restantes geravam um erro interno ao servidor. Os classificadores dos quais foi possível obter resultados, destacam-se pela positiva: cart, continuousNaiveBays, gmoMaxEnt, randomForest e svm. Estes classificadores superaram o método anterior de classificação, podemos observar a confusion matrix na tabela 6. No lado oposto a estes encontram-se o naiveBayes e o winnow. Apesar da accuracy do winnow chegar a 0.75 o kapa evidencia que apenas está a acertar na classificação por sorte. Observando a matriz de confusão (tabela 7) conclui-se que este classifica todos os pixels como área não ardida. O minimum distance e o perceptron obtiveram resultados menos maus mas ainda assim inferiores à classificação usando limites escolhidos manualmente.

Uma vez que os parâmetros são os definidos por defeito pela plataforma todos os classificadores que se destacaram são candidatos para serem usados em outros testes. No entanto o SVM tem tendência a gerar problemas, quer de excesso de memória, quer de excesso de tempo de computação. Por isso apenas serão testados: cart, continuousNaiveBays, gmoMaxEnt e randomForest.

4.4 Smoothing pré e pós classificação

A análise das imagens classificadas mostra que há algum ruído criando pequenas zonas isoladas com falsos positivos e falsos negativos, é possível observar este fenômeno na figura 7. Para resolver este problema temos que utilizar algum método de redução de ruído, porém o GEE só oferece um método básico onde um kernel fixo envolve uma imagem. Surgem duas alternativas: ou após a imagem ser classificada haver então a redução de ruído, ou acrescentar versões dos índices com a redução de ruído ao classificador. A primeira alternativa é a mais simples e direta. A segunda alternativa fornece ao classificador informação sobre a vizinhança do

Tabela 5: Resultados da Comparação de Classificadores

Classificador	Accuracy	Kappa
svm	0,9761	0,9318
gmoMaxEnt	0,9747	0,9276
cart	0,9740	0,9266
continuousNaiveBayes	0,9699	0,9147
randomForest	0,9656	0,9031
perceptron	0,8486	0,4814
winnow	0,7681	0,0026
minimumDistance	0,7530	0,3186
naiveBayes	0,2946	-0,1138

Accuracy e Kappa

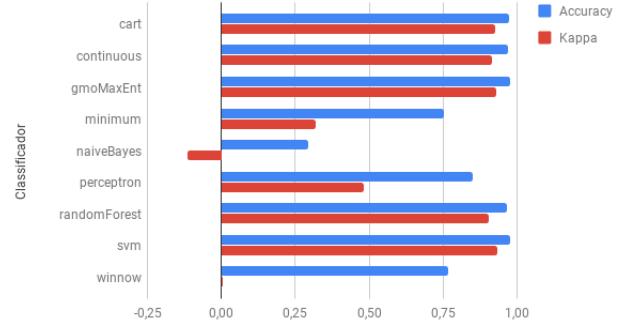


Figura 6: Accuracy e Kapa para os classificadores testados

pixel em questão e cabe a este definir a importância que deve dar a esta informação. Infelizmente esta segunda abordagem também tem pontos negativos porque requer o dobro da memória, uma vez que todas as *features* passaram a ter uma *feature* "irmã" que nos fornece informação sobre a vizinhança. Uma escolha importante passa por selecionar um kernel para realizar a remoção de ruído. Para a solução que reduz o ruído depois da imagem ser classificada foi usado um kernel gaussiano com raio de 30m e sigma de 20m, e para a outra solução foi usado um kernel gaussiano de 3 pixels por 3 pixels mas com o valor central igual a 0 uma vez que esse valor encontra-se na fature irmã.

Os resultados obtidos na tabela 8 favorecem a redução de ruído antes da classificação, porém quando observamos as imagens classificadas conclui-se que a nível de redução de ruído a primeira abordagem é claramente superior. Infelizmente a prestação destas duas alternativas demonstra piores resultados quando comparando com uma implementação sem redução de ruído. É um preço a pagar pela simplificação de certas zonas que não deveriam ser simplificadas. Ao observar as imagens: 7, 8 e 9. Conclui-se que o método que adiciona *features* para tentar reduzir ruído, não cumpre esse objetivo. Já o método mais simples apesar de menos preciso reduz o ruído. Não será utilizada redução de ruído nos seguintes testes para ser possível comparar métricas com outras secções.

4.5 Stratified Random Sampling baseado no landcover

O próximo passo foi relativo a forma como eram obtidas as amostras para treinar os classificadores. Até ao momento eram apenas escolhidos aleatoriamente uma percentagem de pontos da totalidade da imagem, sem qualquer preocupação com o quão representativa esta amostra era quando comparada com a totalidade dos dados. Isto pode induzir o classificador em erro devido a falta

Tabela 6: Matriz de confusão para o classificador cart

		Ground Truth	
		Ardido	Não Ardido
Classificação	Ardido	22,2%	0,3%
	Não Ardido	0,4%	77,1%

Tabela 7: Matriz de confusão para o classificador winnow

		Ground Truth	
		Ardido	Não Ardido
Classificação	Ardido	0,0%	0,0%
	Não Ardido	23,2%	76,8%

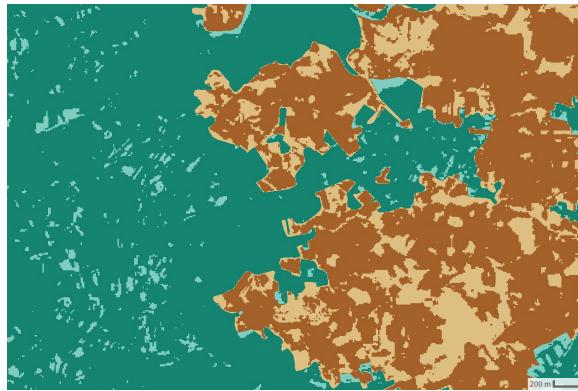


Figura 7: Imagem classificada usando o cart, sem qualquer tipo de redução de ruído



Figura 8: Imagem classificada usando o cart, com redução de ruído após a classificação



Figura 9: Imagem classificada usando o cart, com redução de ruído antes da classificação

ou excesso de informação durante a fase de treino. Foi então adotado uma seleção de pontos que tem como base o tipo de terreno de cada pixel, para que todos os tipos de terreno sejam representados durante o treino. Como fonte de informação do *landcover* foi utilizado o GlobCover: Global Land Cover Map disponível em https://code.earthengine.google.com/dataset/ESA/GLOBCOVER_L4_200901_200912_V2_3. Usando esta forma de amostragem cada tipo de terreno com as suas variações queimado, não queimado terá uma proporção de dados de teste igual a proporção em que está na totalidade dos dados.

Observando a tabela 9 podemos concluir, tendo particular atenção ao kappa, que o cart e o randomForest foram os classificadores que mais ganharam com a amostragem estratificada. É possível observar a matriz de confusão do cart na tabela 10. Nos restantes classificadores não houve uma diferença significativa quer na *accuracy* quer no kappa.

4.6 Aplicar máscaras

Há zonas que nunca devem ser consideradas como área ardida como por exemplo, rios, lagos, autoestradas entre outras. Para estas zonas é possível criar uma máscara e definir que nunca sejam classificadas como ardidas. Durante a análise das imagens foi encontrada uma zona mal classificada. Esta zona está assinalada na figura 5 com o número 2, podemos ver na figura 10 que realmente há uma linha bem definida que na *ground truth* está classificada como não ardida. Analisando a imagens de satélite (figura 11) é possível observar uma auto-estrada. Surge então a necessidade de gerar máscaras para zonas quer nunca deverão ser classificadas como ardidas. No caso em questão apenas foi criada uma pequena máscara para a auto-estrada por isso não há alterações significativas nas métricas utilizadas. Porem é possível ver o resultado da aplicação da máscara na figura 12. A qualidade dos resultados conseguidos pela utilização de máscaras depende da qualidade das máscaras.

Todos os scripts desenvolvidos encontram-se disponíveis em <https://github.com/EduardoFAFernandes/BurnedAreaExperimentsWithGEE>.

5 Conclusão e Trabalho Futuro

Durante o projeto foi possível comparar vários métodos de classificação de área ardida. Foram utilizados vários índices e classificadores, comparados 2 tipos de amostragem, e 2 tipos de redução de ruído. Sendo que os melhores resultados foram alcançados usando o classificador svm, com os índices NDVI, MIRBI, NBR, NBR4 e BAI e respetivos deltas como *features* de entrada e realizando uma amostragem estratificada para obter os dados de treino. Esta implementação obteve uma *accuracy* de 0.9877 e um kappa igual a 0.9705, a tabela 10 representa a sua matriz de confusão.

Tabela 8: Comparação de métodos de redução de Ruído

		Método de redução de ruído		
Classificador		Após Classificação	Pré Classificação	Nenhum
Cart	Accuracy	0,9586	0,9663	0,9740
	Kapa	0,8890	0,9124	0,9265
random Forest	Accuracy	0,9532	0,9601	0,9655
	Kapa	0,8744	0,8963	0,9030
continuous	Accuracy	0,9531	0,9707	0,9698
	Kapa	0,8733	0,9225	0,9147
NaiveBayes	Accuracy	0,9536	0,9660	0,9746
	Kapa	0,8743	0,9093	0,9275

Tabela 9: Comparação de tipos de Amostragem

		Tipo de Amostragem	
Classificador		Aleatória	Estratificada
gmoMaxEnt	Accuracy	0,9746	0,9705
	Kapa	0,9275	0,9283
Cart	Accuracy	0,9740	0,9887
	Kapa	0,9265	0,9730
random Forest	Accuracy	0,9655	0,9785
	Kapa	0,9030	0,9486
continuous	Accuracy	0,9698	0,9658
	Kapa	0,9147	0,9171

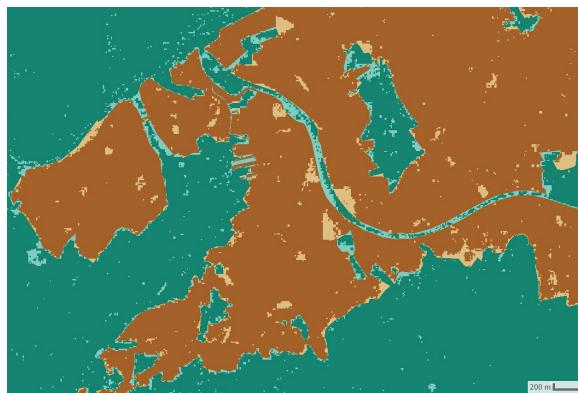


Figura 10: Imagem classificada usando o cart



Figura 11: Imagem de Satélite da auto estrada

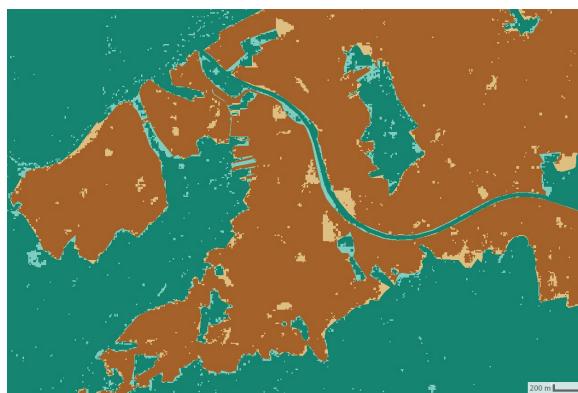


Figura 12: Imagem classificada usando o cart com uma mascara para a auto-estrada

O Google Earth Engine revelou-se numa mais valia durante o desenvolvimento deste projeto. Ao providenciar a infraestrutura necessária co-localizada com os dados, a plataforma consegue facilitar o acesso e a análise de dados de satélite. A documentação disponível está dirigida para iniciantes na área de *remote sensing* e dispõe de inúmeros exemplos bem apresentados. A forma de escrever código é algo diferente da norma, isto deve-se ao facto de o código não correr localmente. Porém o GEE também tem algumas limitações. As mais restritivas são as do limite de tempo de computação e memória que facilmente são atingidas com scripts mais completos. Os erros que são gerados do lado de servidor muitas vezes são pouco descritivos o que pode tornar a depuração do código num desafio. Por último, a natureza paralela da plataforma torna-a algo inflexível. No caso da redução de ruído, não foi possível implementar outras técnicas devido à falta de ferramentas.

Este projecto explorou o editor de código online do Google Earth Engine, este editor integra a parte interativa do GEE. A plataforma tem uma parte de *batch computation* com menos restrições. Seria por isso interessante explorar esta faceta do GEE para tentar escalar as implementações a áreas maiores. Outra ideia interessante seria, utilizar a API para realizar uma aplicação talhada às necessidades de quem está a operar no terreno o precisa dos dados de área ardida.

Referências

BEN-DAVID, A. 2008. Comparison of classification accuracy using cohen's weighted kappa. *Expert Systems with Applications* 34, 2, 825 – 832.

BOSCHETTI, L., STEHMAN, S. V., AND ROY, D. P. 2016. A stratified random sampling design in space and time for regional to global scale burned area product validation. *Remote Sensing of Environment* 186, 465 – 478.

BRANCO, C., AND PORTO, A. R., 2017. Fogos já mataram 107 pessoas desde junho de 2017. desde 2000 morreram 208 pessoas.

Tabela 10: Matriz de confusão do cart usando amostragem estratificada para selecionar dados de treino

		Ground Truth	
		Ardido	Não Ardido
Classificação	Ardido	29,0%	0,4%
	Não Ardido	0,7%	69,9%

BREWER, C. K., WINNE, J. C., REDMOND, R. L., OPITZ, D. W., AND MANGRICH, M. V. 2005. Classifying and mapping wildfire severity. *Photogrammetric Engineering & Remote Sensing* 71, 11, 1311–1320.

CHEN, W., MORIYA, K., SAKAI, T., KOYAMA, L., AND CAO, C. 2016. Mapping a burned forest area from landsat tm data by multiple methods. *Geomatics, Natural Hazards and Risk* 7, 1, 384–402.

DE GESTÃO DE ÁREAS PÚBLICAS E DE PROTEÇÃO FLORESTAL, D. 2017. 9.º relatório provisório de incêndios florestais - 2017 01 de janeiro a 16 de outubro. Tech. rep., Departamento de Gestão de Áreas Públicas e de Proteção Florestal, 10.

FOODY, G. M. 2002. Status of land cover classification accuracy assessment. *Remote Sensing of Environment* 80, 1, 185 – 201.

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*.

GRON, A. 2017. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, 1st ed. O'Reilly Media, Inc.

HUANG, H., ROY, D. P., BOSCHETTI, L., ZHANG, H. K., YAN, L., KUMAR, S. S., GOMEZ-DANS, J., AND LI, J. 2016. Separability analysis of sentinel-2a multi-spectral instrument (msi) data for burned area discrimination. *Remote Sensing* 8, 10.

SANTOS, L. D. 2018. *GPU Accelerated Classifier Benchmarking for Wildfire Related Tasks*. Master's thesis, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa.

SUHET. 2013. *Sentinel-2 User Handbook*.

TIBSHIRANI, R. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)* 58, 1, 267–288.

ZHU, Z., WANG, S., AND WOODCOCK, C. E. 2015. Improvement and expansion of the fmask algorithm: cloud, cloud shadow, and snow detection for landsats 4–7, 8, and sentinel 2 images. *Remote Sensing of Environment* 159, 269 – 277.

6 Anexos

6.1 Formulas usadas para o calculo dos índices

$$NDVI = \frac{NIR - RED}{NIR + RED}$$

$$EVI = \frac{2.5(NIR - RED)}{NIR + 6 * RED - 7.5 * BLUE + 1}$$

$$SAVI = (1 - 0.5) * \frac{NIR - RED}{NIR + RED + 0.5}$$

$$MSI = \frac{SSWIR}{NIR}$$

$$MIRBI = 10 * SSWIR - 9.8 * LSWIR + 2$$

$$BR = \frac{NIR}{LSWIR}$$

$$NBR1 = \frac{NIR - LSWIR}{NIR + LSWIR}$$

$$NBR2 = \frac{2 * NIR}{SSWIR + LSWIR}$$

$$NBR3 = \frac{SSWIR - LSWIR}{SSWIR + LSWIR}$$

$$NBR4 = \frac{NIR - LSWIR}{((NIR + LSWIR) * GREEN) + 1}$$

$$BAI = \frac{1}{(0.1 - RED)^2 + (0.06 - NIR)^2}$$