

## **PREVENDO EXPLOSÕES SOLARES COM DEEP LEARNING**

**Armando Augusto Dias Neto**

Centro Universitário Adventista de São Paulo.

**Daniel Augusto Latanzio**

Centro Universitário Adventista de São Paulo.

**Manoel Beserra Cavalcante Neto**

Centro Universitário Adventista de São Paulo.

**Alciomar Hollanda**

Centro Universitário Adventista de São Paulo.

## PREDICTING SOLAR FLARE WITH DEEP LEARNING

### ABSTRACT

Solar explosions are magnetic activities that cause eruptions, which when their classification reaches high levels, can cause serious damage to the telecommunications devices orbiting the planet Earth. To try to minimize possible damage, this work aimed to identify solar explosions through the algorithm in Deep Learning. A theoretical survey was carried out in Deep Learning, as well as a simulation algorithm for object recognition and their classifications. To construct the algorithm for analyzing the images collected from the Sun provided by the Joint Science Operations Center (JSOC), some technologies and tools were used. The percentage result of correct answers measured after applications with data related to the displayed problem, within the studied metrics were median. It is concluded that the application for analysis of images focused on the field of astrophysics is still in development and with the continuity in the increment and improvement of techniques in the predictive model, better results can be obtained.

Keywords: Solar explosion; Artificial intelligence; Deep Learning; forecast; image analysis.

## PREVENDO EXPLOSÕES SOLARES COM DEEP LEARNING

### RESUMO

Explosões solares são atividades magnéticas que causam erupções, que quando sua classificação atinge os níveis elevados, podem causar sérios danos aos aparelhos de telecomunicações que orbitam o planeta Terra. A fim de tentar minimizar os possíveis danos, este trabalho se propôs a identificar explosões solares através do algoritmo em Deep Learning. Foi realizado o levantamento teórico em Deep Learning, assim como um algoritmo de simulação para reconhecimento de objetos e suas classificações. Para a construção do algoritmo de análise das imagens coletadas do Sol fornecidas pelo Joint Science Operations Center (JSOC), foram utilizadas algumas tecnologias e ferramentas. O resultado percentual de acertos aferido após aplicações com dados relacionados a problemática exibida, dentro das métricas estudadas foram medianos. Conclui-se que a aplicação para análise de imagens voltadas ao ramo da astrofísica ainda está em desenvolvimento e com a continuidade no incremento e aprimoramento das técnicas no modelo preditivo, pode-se obter resultados melhores.

Palavras-chave: Explosão solar; Inteligência Artificial; Deep Learning; previsão; análise de imagens.

## INTRODUÇÃO

Compreendendo e conceitualizando as explosões solares como atividades magnéticas que causam erupções quando o sol está mais ativo, com seu nível de intensidade classificado como A, B, C, M ou X, sendo “X” o nível mais severo, e que, normalmente uma explosão com esse nível são filtrados pela atmosfera e podem acarretar sérios danos aos aparelhos de telecomunicações que orbitam o planeta Terra (OLIVEIRA, 2020).

Dado a compreensão dos fatos, entende-se que o problema existente necessita de acompanhamento, para que ações por entidades responsáveis, sejam tomadas, e essas só poderão ser eficazes, se uma previsão aproximada do momento da explosão for detectada.

Através da Inteligência Artificial (IA), com aplicação em *Deep Learning*, imagens fornecidas pelo *Joint Science Operations Center* (JSOC), foram treinadas e analisadas, resultando o momento aproximado em que explosões solares venham a acontecer.

O *Deep Learning* é um subcampo do ramo da IA, portanto, com o mesmo fundamento, busca desenvolver o aprendizado ainda mais profundo com diversas camadas de processamento, compostas por diversas alterações lineares e não lineares (BIG DATA, 2021). Por seu intenso e profundo processo de aprendizado de máquina, a criação de um modelo de reconhecimento de objetos com *Deep Learning* pode ser automatizada se a máquina obtiver sucesso em seu aprendizado segundo NGUYEN et al. (*apud* OLIVEIRA, 2020, p.04), e, portanto, adequado para nosso caso de estudo.

## METODOLOGIA

Foi realizado o levantamento bibliográfico, utilizando-se de palavras chaves para uma pesquisa eficaz e fontes confiáveis, como *Google* acadêmico, teses e dissertações e documentos acadêmicos.

Foi realizado também levantamento teórico, junto a professor Ms. Alciomar Hollanda, em *Deep Learning*, assim como um algoritmo de simulação para reconhecimento de objetos e suas classificações, com o objetivo de estudar e analisar o funcionamento de modelos classificatórios de *Deep Learning* para futura aplicação na problemática apresentada neste artigo.

Após aplicação do modelo de simulação de classificação, foi utilizada o conjunto de dados de explosões solares disponibilizado por Gradvohl *et al.* (2021), para o desenvolvimento do algoritmo de aprendizado profundo para predição de explosões solares a partir das imagens analisadas.

O algoritmo desenvolvido foi aplicado na linguagem Python, devido a sua usabilidade e flexibilidade em sistemas operacionais distintos, assim como sua simplicidade, mesmo sendo uma linguagem extremamente poderosa, de alto nível e de uso geral, possuindo uma sintaxe clara e objetiva, permitindo o foco total a solução do problema. Esta linguagem também nos proporciona diversos pacotes e funções para mineração de dados e criação de algoritmos de *Deep Learning*.

## DESENVOLVIMENTO

Para a construção do algoritmo de análise das imagens coletadas do Sol, foram utilizadas algumas tecnologias e ferramentas que serão explanadas perante o artigo.

### 3.1. Explosão Solar

O desenvolvimento da ciência e tecnologia nos dias de hoje possibilitaram a criação e envio de naves espaciais para órbita da Terra, assim como diferentes tipos de satélites, que em conjunto, nos disponibilizam acesso ao Sistema de Posicionamento Global (GPS), internet, controle de fluxo aéreo, sistemas elétricos, entre outros. Desse modo, fenômenos que ocorrem no Sol e disparam radiação pelo sistema solar, podem impactar gravemente aparelhos eletrônicos (ANDREWS, 2019).

São chamados de explosões solares, erupções que ocorrem no Sol a partir de regiões magneticamente instáveis e ativas no astro. Essas explosões são classificadas em nível de intensidade, podendo ser A, B, C, M ou X (Tabela 1).

Tabela 1 - Níveis de explosões solares.

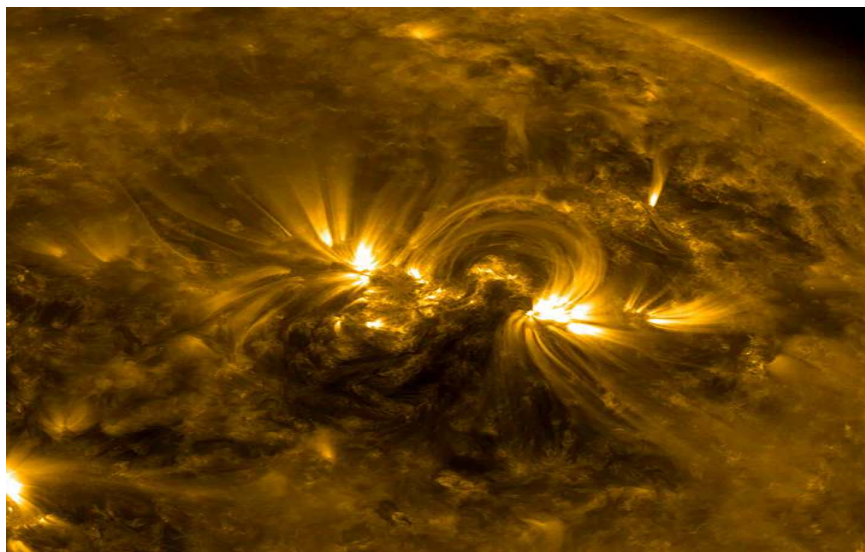
Classe.	Fluxo de pico de raios-X em W/m <sup>2</sup> .
A	$<10^{-7}$
B	$\geq 10^{-7}$ and $<10^{-6}$
C	$\geq 10^{-6}$ and $<10^{-5}$
M	$\geq 10^{-5}$ and $<10^{-4}$
X	$\geq 10^{-4}$

Fonte: OLIVEIRA, 2020

Explosões solares de classe X são o tipo mais intenso do fenômeno, liberando grandes quantidades de radiação pelo sistema solar, que podem impactar o planeta Terra, danificando componentes em órbita, causando problemas em comunicação de rádio e até mesmo apagões na rede elétrica, assim como trazer perturbações para o campo magnético

terrestre que podem levar semanas para se estabilizarem (CARDIAL, 2017). Na Figura 1 podemos observar uma região ativa no Sol.

Figura 1 – Explosão Solar



Fonte: *Solar Dynamics Observatory*, NASA (2017)

A Administração Nacional da Aeronáutica e do Espaço ou *National Aeronautics and Space Administration* (NASA) e uma companhia espacial americana realiza análises de

regiões ativas do Sol para cálculo de possíveis explosões solares e sua intensidade, desta forma podendo combater com antecedência seus possíveis impactos.

### 3.2. Inteligência Artificial

Em 1956 o termo Inteligência Artificial (IA) foi criado e utilizado pela primeira vez, surgindo quando a ideia de que computadores devem pensar e/ou agir de forma inteligente se instalou segundo RUSSELL; NORVIG (*apud* GONSALVES, 2018, p.25), e tem ganhado cada vez mais destaque, graças aos crescentes volumes de dados disponíveis, algoritmos avançados, e melhorias no poder e no armazenamento computacionais. Ao simular o modo de aprendizado e raciocínio humano, umas imensidões de possibilidades de aplicações dessa tecnologia são disponíveis, onde podemos destacar, principalmente, o reconhecimento facial, análise e reconhecimento de objetos em imagens, análise preditivas, modelos classificativos, Segurança e vigilância.

Durante o passar dos anos, algumas definições foram impostas à IA:

I. Pensar como seres humanos: “O novo e interessante esforço para fazer os computadores pensarem... máquinas com mentes, no sentido total e literal”. Segundo HAUGELAND (*apud* PACHECO, 2018, p.03).

II. Atuar como seres humanos: “A arte de criar máquinas que executam funções que exigem inteligência quando executadas por pessoas.” Segundo KURZWEIL (*apud* PACHECO, 2018, p.03).

III. Pensar racionalmente: “O estudo das faculdades mentais pelo seu uso de modelos computacionais.” Segundo CHARNIAK E MCDERMOTT (*apud* PACHECO, 2018, p.03).

IV. Atuar racionalmente: “A Inteligência Computacional é o estudo do projeto de agentes inteligentes.” Segundo POOLE et al. (*apud* PACHECO, 2018, p.03).

Uma vez entendido esses conceitos, podemos observar que a IA busca obter de que a capacidade de um computador seja o mais próximo do que um ser humano é capaz de fazer, com isso, foi desenvolvido o famigerado *Machine Learning*.

### 3.3. Machine Learning

Os seres humanos aprendem desde que nascem, observam e interagem com o ambiente ao redor, de maneira em que tudo que os sentidos captam, desde o momento em que nascem, se transformam em dados para o cérebro, e dessa forma, consigam se adaptar ao mundo, aprendendo a observar padrões, todos os experimentos realizados pelas pessoas viram dados no cérebro e que serão utilizados para o resto da vida (MELLO, 2007).

O aprendizado de máquina correlaciona com o que uma pessoa faz, obter, observar os dados do mundo e conseguir então realizar tarefas. Segundo MITCHELL (*apud* FANK, 2018, p.11), “*Machine Learning* é uma área da computação que busca melhorar a autonomia dos computadores através da sua própria experiência. Ou seja, como fazer com que algoritmos possam aprender a partir de uma base de dados de entrada.”

Para que o aprendizado da máquina ocorra, deve-se permitir que ela aprenda o modelo por si só, através dos métodos supervisionados e não supervisionados.

Nos métodos supervisionados será informado uma resposta para a máquina, ela terá uma entrada e uma saída associadas entre si, logo a máquina terá que relacionar esses valores,

nesse método, teremos a classificação dos dados em categorias, segundo HASTIE (*apud* SANTOS, 2018, p.22).

No método não supervisionados, a máquina não será informada sobre o que ela deverá aprender e nem terá uma variável resposta conhecida, será informado apenas os dados para rede e então, deverá aprender a separar por conta própria, com base nas características das imagens, segundo JAMES (*apud* SANTOS, 2018, p.22).

Dentre os métodos apresentados, o algoritmo desenvolvido nesse artigo apropria-se do método supervisionado, onde teremos os dados com a inserção também da saída resultante. Uma vez conceitualizado *Machine Learning*, e apresentado o método utilizado, podemos entranhar ainda mais no assunto, assim como a aplicabilidade que o *Deep Learning* visa fazer.

### 3.4. Deep Learning

*Deep Learning* (DL) é a aplicação de uma quantidade massiva de camadas de processamento em um algoritmo de rede neural (COPELAND, 2016). Em sua tradução para o português, como aprendizado profundo, é uma subárea de *Machine Learning* (Aprendizado de máquina) que por sua vez, é uma subárea da Inteligência Artificial. O DL é uma área que se inspira na forma em que o cérebro do ser humano aprende, como ele capta e trabalha com as informações que recebe do ambiente ao seu redor.

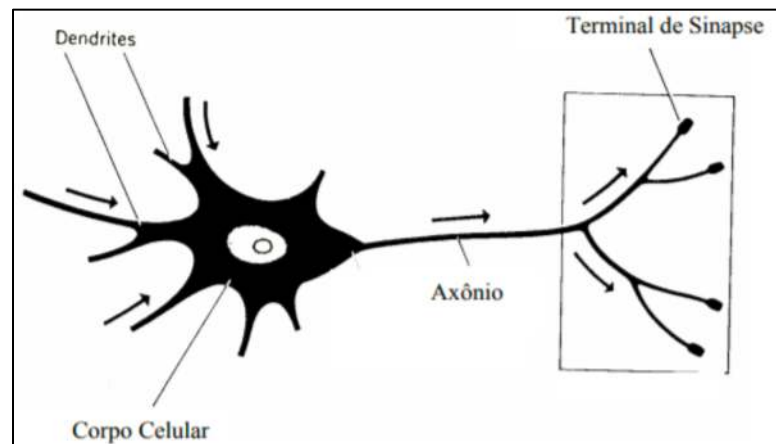
Para compreender melhor como DL funciona, a analogia com o cérebro é de suma importância, visto que ele seria uma imitação do processamento de informações do cérebro humano. Uma vez recebida uma informação externa, através de nossos sensores, como olhos, ouvidos etc., será transferida através de um sistema de neurônios, até chegar ao cérebro, onde finalmente irá processar as informações segundo ANITHA *et al.* (*apud* MIRANDA, 2012).

Entendemos então que a ideia de imitar o processo de aprendizado do cérebro humano, foca então na pergunta: e se nós conseguíssemos criar todo esse processo, de uma forma artificial, ou seja, como criar uma forma de imitar os neurônios, que passam informação e sabem exatamente que alerta passar até chegar no cérebro? Os neurônios devem saber qual informação passar para que incômodos sejam evitados nos seres humanos, então o sistema nervoso atua filtrando alguns sinais recebidos e ele também deve aprender qual o tipo de informação, irá permitir que os nossos neurônios levem até o cérebro, esse mesmo sistema foi a inspiração para a criação de neurônios que aprendem e conversem entre si, de forma totalmente artificial (MIRANDA *et al.*, 2012).

Em 1958, o psicólogo Frank Rosenblatt, inventou o *Perceptron*, que é uma forma simplificada de tentar imitar os neurônios biológicos segundo Yonenaga e Figueiredo (*apud* MIRANDA *et al.*, 2012). Seu sistema recebe informação pelo *Input*, que seria um vetor, análogo aos dendritos que recebem informação na célula humana, a informação recebida, seria transferida para dentro do que é conhecido como neurônio, análogo ao núcleo do neurônio biológico, dentro dele as informações são somadas e processadas por uma função

de ativação, e após ser processada ele passa a informação até o *Output* que retornará com o resultado, semelhante ao que o axônio faz (MIRANDA *et al.*, 2012), como apresentado na Figura 2.

Figura 2 - Neurônio Biológico.



Fonte: BIANCHINE, 2004

No modelo do neurônio biológico, os seres humanos estão em constante aprendizado, identificando e diferenciando o que é quente ou frio, seco ou húmido etc., ao longo da vida. O *Perceptron* pode aprender, passando vários e vários dados dentro dele. Para o reconhecimento de um objeto, ele conseguirá definir através das características recebidas no vetor (*Input*), dentro do neurônio ele irá classificar um peso para cada informação recebida e irá somá-las, o resultado dessa soma seria passado para uma função chamada de função de ativação que tem como objetivo de adicionar um pouco de não linearidade no processo, o resultado final é transmitido pela função de ativação e então comparado com o resultado esperado para aquelas características. O aprendizado seria justamente atualizar o peso dado para cada informação, visando que o resultado chegue o mais próximo do correto possível, segundo Yonenaga e Figueiredo (*apud* MIRANDA *et al.*, 2012).

O *Perceptron* não dava conta de tarefas muito complexas, então para melhorar ainda mais a técnica, novamente o cérebro serviu de inspiração, pois ele aprende com uma rede de neurônios extremamente complexos trabalhando e aprendendo juntos, logo, faz sentido se adicionássemos vários *Perceptrons*, imitando uma rede de neurônios, ou seja, vários deles conectados, seriam muito semelhante a rede neural biológica, seria mais complexa e teria mais neurônios para aprender, ela soma uma estrutura bem análoga aos neurônios conectados, onde cada parte é chamada de camada. (MIRANDA *et al.*, 2012).

As camadas são divididas em *Input*, camadas escondidas (neurônios conectados) e o *Output*, o número dessas camadas escondidas, definem o nome que a rede neural irá receber, se é contemplada apenas com uma camada escondida será conhecida como rede neural superficial, já uma rede com mais camadas escondidas, leva o nome de rede neural profunda, logo o que conhecemos como *Deep Learning*, um aprendizado baseado numa rede profunda de neurônios artificiais conectados entre si, segundo Valença *et al.* (*apud* MIRANDA *et al.*, 2012).

O aprendizado é bem semelhante do *Perceptron*, no entanto, não seria apenas pesos definidos por um neurônio, mas uma rede complexa com diversos neurônios, cada qual com seus pesos para aprender.

Atualmente os *Perceptrons* não são os únicos componentes da técnica de *Deep Learning*, existem também as camadas Convolucionais que compõe as redes neurais Convolucionais, assim como os *Perceptrons* são inspirados pelos neurônios, as Convolucionais são inspiradas nas células responsáveis pela visão (PACHECO, 2018).

Segundo PONTI E COSTA (2017), Rede Neural Convolutacional (CNN, do inglês *Convolutional Neural Network*) é o modelo de rede *Deep Learning* mais conhecido e utilizado nos últimos anos. Pois suas aplicações principais são o processamento de informações visuais, em especial imagens, onde a convolução filtra as imagens com base em sua estrutura bidimensional.

A etapa de convolução caracteriza-se pela passagem do núcleo (kernel) pela imagem (*input*) e o resultado desse processamento é denominado mapa de características (*output*), o qual permitirá o reconhecimento de padrões, inclusive de outros padrões da rede segundo GOODFELLOW (*apud* PACHECO, 2018, p.03).

### 3.5. Escolha da Deep Learning

Para trabalhar-se o problema de visão computacional e classificação de imagens, tais como trabalhados nesse artigo, podemos utilizar uma poderosa ferramenta da área da Inteligência Artificial conhecida como *Deep Learning*, através dela é possível criarmos algoritmos que podem reconhecer e classificar objetos em fotos ou vídeos, apresentando um considerável grau de performance, uma vez que essa ferramenta possui várias camadas de aprendizagem, que vai lhe dispondo um maior conhecimento e poder de classificação a cada camada aprofundada em seu treinamento.

Podemos compreendermos a famigerada análise de reconhecimento de imagens e seus resultados pelo trabalho de Vargas et al. (*apud* PACHECO, 2018, p.11)

***Um Estudo Sobre Redes Neurais Convolucionais e sua Aplicação em Detecção de Pedestres***, comparou três técnicas para a detecção de pedestres: *Features*, Histograma de Gradiente Orientados (HOG), e CNN, onde a CNN atingiu o melhor resultado. A abordagem *Features* teve 68,78% de precisão e 96,64% de acurácia; HOG teve 94,63% de precisão e 91,53% de acurácia; e a CNN teve 99,12% de precisão e 95,98% de acurácia.

### 3.6. Métricas de desempenho

O projeto descrito nesse artigo utilizou quatro métricas de avaliação de desempenho no modelo preditivo, sendo elas: *Precision*, *Recall*, *Accuracy* e *TSS*.

Abaixo, podemos observar a Tabela 3, apresentando a matriz de confusão relacionada as métricas que serão explicadas a seguir, onde Verdadeiros Positivos são a classificação correta da classe Positivo, Falsos Negativos quando o valor real era classe Positivo, mas o modelo previu classe Negativo, Falsos Positivos quando o valor real era classe Negativo, mas o modelo previu classe Positivo e Verdadeiros Negativos quando classificado corretamente a classe Negativo.

Tabela 3 – Matriz de confusão das métricas de desempenho.

		Detectada	
		Sim	Não
Real	Sim	Verdadeiro Positivo (TP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (TN)

Fonte: Próprio Autor.



### 3.6.1. Precision

Segundo Bobra e Couvidat (2015, p. 5), “A precisão caracteriza a capacidade do classificador de não rotular como positivo um exemplo que é negativo, e é definida como”.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (1)$$

Será resultante do cálculo de *precision* o percentual em que o modelo preditivo irá acertar a classificação positiva, quando esse, de fato ser positivo, para isso, temos a divisão de *True Positive* (TP), por *True Positive* (TP) somado ao *False Positive* (FP). Na imagem abaixo temos um exemplo prático de como a métrica *precision* trabalha para apresentar o seu resultado, no caso dado, teríamos 0,57.

Figura 3 – Exemplo de aplicação da métrica *Precision*.

Truth										
Prediction	Dog	Dog	Dog	No Dog	Dog	No Dog	Dog	No Dog	Dog	Dog
	✓	✗	✓		✓		✗		✓	✗
True Positive = 4					False Positive = 3					

Fonte: PRECISION, ... 2021











### 3.6.2. Recall

Segundo Bobra e Couvidat (2015, p. 5), “O recall (também conhecido como sensibilidade ou taxa de acerto) caracteriza a capacidade do classificador de encontrar todos os exemplos positivos”.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2)$$

Será resultante do cálculo de *recall* o percentual em que o modelo preditivo irá classificar corretamente o seu objeto de estudo, ou seja, a capacidade de reconhecer todas as imagens do objeto de estudo como corretas, sem a indicação de uma imagem como falsa, quando essa for verdadeira, para isso, temos a divisão de *True Positive* (TP), por *True Positive* (TP) somado ao *False Negative* (FN). Na imagem abaixo temos um exemplo onde o *recall* resultaria em 4/6=0,67.

Figura 4 – Exemplo de aplicação da métrica *Recall*.

Truth										
Prediction	Dog	Dog	Dog	No Dog	Dog	No Dog	Dog	No Dog	Dog	Dog
	✓	✗	✓		✓		✗		✓	✗

Recall is out of all **dog truth** how many you got it right?

Total Dog truth samples = 6      True Positive = 4

Fonte: PRECISION, ... 2021











### 3.6.3. Accuracy

A *accuracy* segundo Bobra e Couvidat (2015, p. 5), “é a razão entre o número de previsões corretas sobre o número total de previsões”.

$$\text{Accuracy} = \frac{TP+TN}{TP+FN+TN+FP} \quad (3)$$

Será resultante do cálculo de *accuracy* o percentual em que o modelo preditivo irá classificar corretamente as observações propostas a ele, para isso, temos a divisão de *True Positive* (TP) somado ao *True Negative* (TN), sobre a soma de *True Positive* (TP), *False Negative* (FN), *True Negative* (TN), *False Positive* (FP). Na imagem abaixo temos um exemplo onde a *accuracy* resultaria em 5/10=0,5.

Figura 5 – Exemplo de aplicação da métrica *Accuracy*.

Truth										
Prediction	Dog	Dog	Dog	No Dog	Dog	No Dog	Dog	No Dog	Dog	Dog
	✓	✗	✓	✗	✓	✓	✗	✗	✓	✗

Fonte: PRECISION, ... 2021

### 3.6.3. TSS

Segundo Bobra (2015, p.6), “O TSS é a diferença entre a taxa de recordação e alarme falso, ou seja, a recordação mais a especificidade menos um. Isso é equivalente a um menos a taxa de falsos alarmes menos a taxa de falsos negativos (definida como FN/(FN+TP)):

$$TSS = \frac{TP \times TN - FP \times FN}{P \times N} = \frac{TP}{TP+FN} - \frac{FP}{FP+TN} \quad (4)$$

A métrica é utilizada para fazer uma comparação entre os trabalhos publicados de explosões solares, seu melhor resultado é 1, quando o modelo preditivo teve o número máximo de acertos para positivos e negativos, para isso, temos a divisão de *True Positive* (TP)

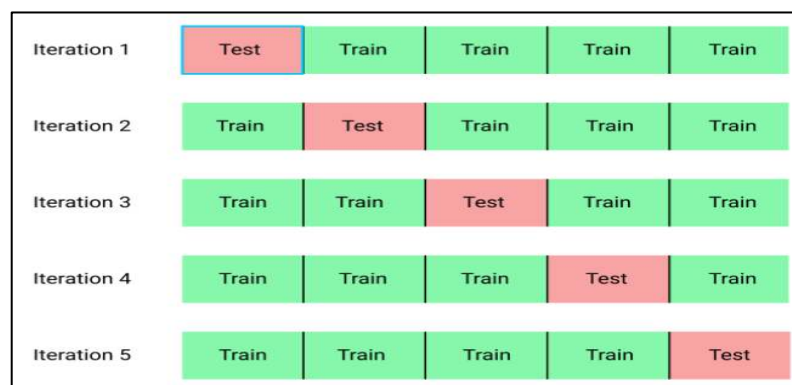
multiplicado ao *True Negative* (TN), subtraído pela multiplicação de *False Positive* (FP) e *False Negative* (FN), o resultado obtido dividido por a quantidade *positive* (P), multiplicado a quantidade *negative* (N) será o indicador TSS.

### 3.7. Cross Validation

Como técnica para avaliar o particionamento da base de dados para treinamento e testes, foi escolhido para o projeto o *Cross Validation* que segundo os estudos de Hyndman et al. (*apud* MARTINS, 2017, p.20), é uma maneira de percorrer sobre os dados, de forma que todos sejam utilizados como testes, ao contrário do que é fornecido pelo método *hold-out* (*in-sample* e *out-of-sample*), que se divide em apenas dados de treino e amostra, a prerrogativa do *Cross Validation* é também evitar o *Overfitting* que tem um bom desempenho quando utilizado dados de treino, mas que se apresenta com performance baixo quando se utiliza dados de testes.

Abaixo temos um exemplo ilustrativo do processo que o *Cross Validation* realiza, com a divisão aleatória do *dataset* em iterações e quantidade de amostras próximas em cada uma delas, onde um subconjunto é utilizado como dados para testes e os demais como treinamento. Para cada iteração a parte ainda não destinada como dados de teste é solicitada, visando garantir que todos sejam testados em algum momento:

Figura 6 – *Cross Validation*.



Fonte: SHAIKH, 2022

### 3.8. Resultados do trabalho

Com a premissa de estudar o funcionamento de modelos classificatórios para futura aplicação no problema apresentado no artigo, foi criado o *Sequential*, “Segundo a documentação oficial do *Keras*, o método *Sequential* cria uma estrutura linear e em pilha das camadas da rede neural. Isso definirá a forma da rede neural” (FALCÃO et al., 2019), contido na biblioteca *Tensorflow*. Nosso modelo criado foi alimentado com dados de treino compostos por imagens de carros e motos e suas respectivas classificações (0 e 1). Após o treinamento do modelo foram realizados alguns testes de funcionamento e assertividade, com imagens de carros e motos não presentes nos dados de treino, submetidos ao modelo. Aplicando as métricas apresentadas nesse artigo (*Accuracy*, *Precision*, *Recall* e TSS) no modelo preditivo piloto, temos os resultados respectivamente em: 67,55%, 74,60%, 71,42% e 0,85.

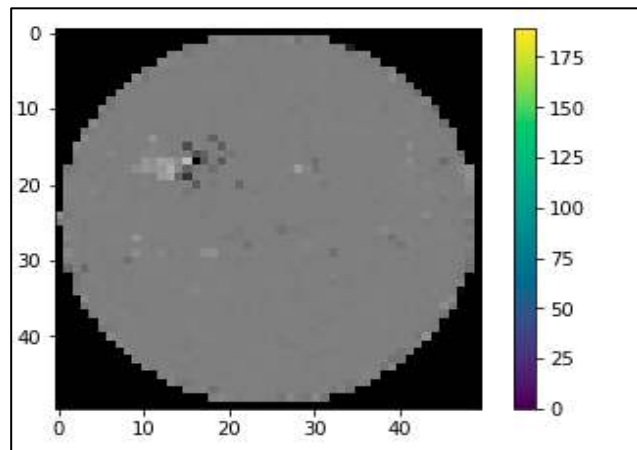
Após a elaboração do modelo de simulação descrito, foi implementada a aplicação do algoritmo de redes neurais em *Deep Learning* para identificar regiões ativas em imagens coletadas do Sol (magnetogramas), que visa prever uma possível explosão solar, através de treinos com uma série de 37 imagens para cada nível conhecido de explosões solares (B, C, M e X), a fim de reconhecer e premeditar essas explosões.

Nosso *Dataset* de magnetogramas foi fornecido pelo *Joint Science Operations Center* (JSOC) disponibilizado por Gradvohl et al. (2021), contendo 700 imagens de explosões de classe B, 626 imagens de classe C, 232 imagens de classe M e 37 imagens de classe X, e, portanto, para balanceamento entre as classes e uma limitação computacional para processamento de um grande conjunto de imagens, decidimos reduzir nosso *dataset* à 37 imagens para cada classe de explosão solar.

Foi observado que a resolução das imagens do nosso conjunto de dados eram de 4096 x 4096px, porém para um processamento de dados local, com limitado poder de processamento de imagens, decidimos redimensioná-las, e para isso, após uma série de experimentos para diferentes tamanho de redimensionamento, como 1000 x 1000px, 300 x 300px, 100 x 100px, entre outros, encontramos na dimensão de 50 x 50px valores que nos proporcionavam um processamento eficiente dos dados sem que houvesse perda de informação dos mesmos, assim como constatamos que com valores acima de 1000 x 1000px o treinamento do modelo era impedido por falhas na memória do computador.

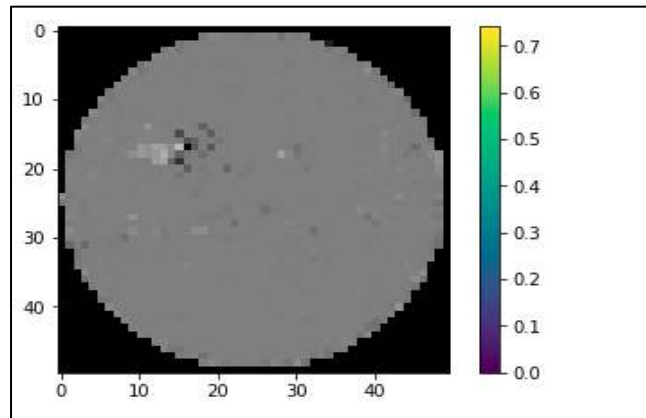
Considerando-se que uma imagem possui RGB que variam entre 0 e 255, o que deve ser analisado antes do treinamento do modelo, uma vez que, pode-se atribuir mais pesos ao aprendizado dos pixels de maior valor, à detrimento dos que possuem valores próximos ou igual à 0, uma vez que há uma grande diferença nos valores dos mesmos. Pensando nisso, optamos por normalizar nossos dados em escala de 0 a 1px, dividindo cada pixel da imagem. Para isso, a técnica utilizada foi dividir os pixels por 255, desta forma transformando o intervalo de valores deles, como pode ser identificado na Figura 7 e 8. Outros métodos de normalização de dados são conhecidos, porém não utilizados para esse projeto, como o método  $((V(\text{var}) - M(\text{var})) / (\text{Max}(\text{var}) - \text{Min}(\text{var})))$ , onde  $V(\text{var})$  = Valor da variável,  $M(\text{Var})$  = Média da Variável, Max e Min(var) são os valores máximos e mínimos da variável, respectivamente.

Figura 7 – Plot do magnetograma não normalizado.



Fonte: Próprio Autor

Figura 8 – Plot do magnetograma normalizado.



Fonte: Próprio Autor

O modelo preditivo de *Deep Learning* utilizado nesse experimento foi o mesmo utilizado no de carros e motos, *sequential* do *framework tensorflow*. Configuramos o modelo para apresentar 128 camadas de treinamentos nos dados e retornar como saída somente 2 camadas, as classes 0 e 1. Os modos de ativação das camadas definidas foram *relu* e *softmax* respectivamente. Como otimizador, definimos o hiper parâmetro Adam e para parâmetro de perda, foi utilizado o *Sparse Categorical Crossentropy* (SCC). Na Figura 9, podemos observar o algoritmo descrito do projeto desenvolvido.

Figura 9 – Modelo *Sequencial*.

```
def cria_modelo():

    #CRIANDO MODELO:
    modelo = keras.Sequential([
        keras.layers.Flatten(input_shape=(50, 50, 3)),
        keras.layers.Dense(128, activation='relu'),
        keras.layers.Dense(2, activation='softmax')
    ])

    # Configurando o modelo com otimizador, parametro de perda e metrica:
    modelo.compile(optimizer='adam',
                   loss='sparse_categorical_crossentropy',
                   metrics=['accuracy'])

    return modelo
```

Fonte: Próprio Autor

Para o treinamento e teste do modelo preditivo utilizamos a técnica *Cross Validation* que se baseia em dividir o conjunto de dados em diferentes partes (*folds*) e realizar o treinamento e teste do modelo em cada uma delas, garantindo que não haja um treinamento sobre dados inadequados, portanto, definimos 3 *folds* para os dados analisados, com utilização do método *StratifiedKfold* da biblioteca *model\_selection* do *framework sklearn* que separa os dados randomicamente. Para cada combinação de dados criada para o treinamento e teste do modelo, tem-se, nos mesmos as *features* (dados das imagens, como os pixels, propriamente ditos) e o *target* (categoria da imagem associada a cada dado, ou seja, a classe 0 ou 1), desta forma realizando-se um treinamento supervisionado do modelo, onde o mesmo recebe os

dados e uma resposta associada a eles. Por fim, as métricas de análise do modelo são a média delas, realizadas para cada parte do treinamento, tendo como resultados próximos ao que seguem: *Accuracy*: 52,64%, *Precision*: 50,63%, *Recall*: 53,44%, TSS: 0,2.

### 3.9. Trabalhos comparativos

Aplicado as técnicas estudadas em nosso modelo preditivo e dado os resultados apresentados no tópico anterior, apresentamos um trabalho comparativo da autora Bobra (2015) ao desenvolvido nesse projeto.

A apresentação desse trabalho paralelo tem como objetivo identificar que diferentes abordagens para o mesmo tema podem ser aplicadas e todos seguem com seus pontos positivos e negativos, mostrando as diversas possibilidades existentes.

Bobra; Couvidat (2015) segue com a abordagem voltada a análise de explosão solar, utilizando para isso dados já processados, portanto, não houve análise da imagem em si, seu *dataset* era composto por dados numéricos referentes as regiões ativas da imagem, seu algoritmo é um módulo *Scikit-Learn* de um SVM (*Machine Learnig*), também construído em linguagem Python.

Na Tabela 4 podemos observar os resultados das métricas dos modelos estudados nesse artigo, sendo eles o modelo desenvolvido para análise de imagens de explosões solares, automobilísticos e o modelo de explosões solares desenvolvido pela autora Bobra. Na primeira coluna da tabela apresenta as métricas utilizadas em ambos os projetos e em cada linha a sua frente, os resultados percentuais destes modelos. Podemos observar que na métrica *Accuracy*, o modelo de Bobra, teve um percentual de aproximadamente 43% em vantagem ao apresentado nesse artigo, em contrapartida na métrica *Precision*, o modelo preditivo de Bobra foi o que obteve o menor resultado percentual e tendo destaque ao modelo piloto com dados automobilísticos em 74,6% em seu resultado, já para a métrica *Recall*, o modelo de Bobra novamente se destacou com um resultado de 83,2%. O TSS mais alto foi conquistado pelo modelo piloto apresentado.

Tabela 4 – Apuração de resultados (*Solar Flare x Cars and Motorcycles x Solar Flares from M.G.Bobra*).

Metric	Solar Flare	Cars and Motorcycles	Solar Flare (from M.G. Bobra)
Accuracy	0,526	0,675	0,924
Precision (positive)	0,506	0,746	0,417
Recall (positive)	0,534	0,714	0,832
TSS	0,200	0,857	0,761

Fonte: BOBRA; COUVIDAT, 2015

### CONSIDERAÇÕES FINAIS

O trabalho apresentado tem a finalidade de transpor o conhecimento aprofundado em técnicas de *Deep Learning*, bem como sua aplicação em análise e reconhecimento de imagens para tomadas de decisões, segundo suas predições.

A problemática apresentada sobre as explosões solares, quando identificadas através do algoritmo proposto em *Deep Learning*, ajudará as organizações na decisão de meios para evitar os riscos apresentados por elas.

O resultado percentual de acertos aferido após aplicações com dados relacionados a problemática exibida, dentro das métricas estudadas foram medianos ao que esperávamos, no entanto, podemos concluir que a aplicação para análise de imagens voltadas ao ramo da astrofísica ainda está em desenvolvimento e com a continuidade no incremento e aprimoramento das técnicas no modelo preditivo, bem como uma possível modificação de parâmetros e/ou segmentação de dados para extração da região ativa dos magnetogramas (dados analisados) podemos obter resultados melhores.

O projeto de pesquisa foi concluído dentro da possibilidade do tempo desempenhado para esta e condições de hardware limitada, deixamos aberto para que em um tempo maior destinado ao projeto e com a possibilidade de um aumento significativo nos dados de treinos (talvez um processamento distribuído ou em nuvem), outros pesquisadores interessados no assunto deem continuidade em nossa pesquisa, para alcançar resultados ainda mais promissores, dado a certeza de possibilidade do aumento percentual apresentado no tópico de trabalhos comparativos.



## REFERÊNCIAS

ANDREWS, Robin George. **TEMPESTADES SOLARES PODEM SER PIORES SE VOCÊ MORAR PERTO DE CERTAS ROCHAS**. 2020. Disponível em: <<https://www.nationalgeographicbrasil.com/espaco/2019/05/tempestades-solares-geologia-regional-dano-rochas>>. Acesso em: 20 set. 2021.

**BIG DATA com R e Azure Machine Learning**. Produção do curso de Formação Cientista de Dados. Brasília: Data Science Academy, 2021. Disponível em: <<http://www.datascienceacademy.com.br>>. Acesso em: 27 Set. 2021.

BOBRA, M. G.; COUVIDAT, S.. SOLAR FLARE PREDICTION USINGSDO/HMI VECTOR MAGNETIC FIELD DATA WITH A MACHINE-LEARNING ALGORITHM. **The Astrophysical Journal**, [S.L.], v. 798, n. 2, p. 135, 8 jan. 2015. American Astronomical Society. <http://dx.doi.org/10.1088/0004-637x/798/2/135>.

CARDIAL, Richard. **Explosões Solares: um guia prático sobre intensidades e danos**. Um guia prático sobre intensidades e danos. 2017. Disponível em: <<https://www.galeriadometeorito.com>>. Acesso em: 13 set. 2021.

COPELAND, B. R. **Is Free Trade Good for the Environment?**. The American Economic Review, 2015.

FALCÃO, João Vitor Regis et al. REDES NEURAIIS DEEP LEARNING COM TENSORFLOW. **Re3C - Revista Eletrônica Científica de Ciência da Computação**, Minas Gerais, v. 14, n. 1, p. 1-12, 18 dez. 2019. Disponível em: <https://revistas.unifenas.br/index.php/RE3C/article/view/232>. Acesso em: 24 abr. 2022.

FANK, Elias Augusto. **Estudo comparativo entre plataformas de deep learning**. 2018. 54 f. TCC (Graduação) - Curso de Ciência da Computação, Campus Chapecó, Universidade Federal da Fronteira Sul, Chapecó, SC, 2018. Disponível em: <<https://rd.uffs.edu.br/handle/prefix/2097>>. Acesso em: 04 out. 2021.

GONSALVES, Lucas Gabriel. **Um estudo de aplicação de inteligência artificial em jogos**. 2018. 61 f. Trabalho de Conclusão de Curso (Tecnologia em Análise e Desenvolvimento de Sistemas) - Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2018.

GRADVOHL, André Leon S., LIMA, Ana Luísa Fogarin de Sousa, & LIU, Jianna. (2021). **Imagens de erupções solares em comprimento de onda de 1600 Angstrom (1.1)** [Conjunto de dados]. Zenodo. <https://doi.org/10.5281/zenodo.4483864>

MARTINS, Guilherme Cruvello da Silveira. **Análise de clusters, singular spectrum analysis e cross validation na previsão de velocidade do vento**. 2017. 46 f. Trabalho de Conclusão de Curso (Graduação de Estatística) - Instituto de Matemática e Estatística, Universidade Federal Fluminense, Niterói, 2017.

MELLO, Suely Amaral. Infância e humanização: algumas considerações na perspectiva histórico-cultural. **Perspectiva**, Florianópolis, v. 25, n. 1, p. 83-104, jun. 2007. Disponível em: <https://periodicos.ufsc.br/index.php/perspectiva/article/view/1630/1371>. Acesso em: 23 abr. 2022.



MIRANDA, André Pacheco et al. SISTEMA DE ANÁLISE DE ATIVOS ATRAVÉS DE REDES NEURAIS DE MÚLTIPLAS CAMADAS. **Administração da Universidade Federal de Santa Maria**, Santa Maria, Brasil, v. 5, n. 1, p. 145-162, abr. 2012.

NASA, Solar Dynamics Observatory. **Solar Golden Arches**. NASA, 2017. Disponível em: <<https://sdo.gsfc.nasa.gov/gallery/main/item/799>>. Acesso em: 29 Nov. 2021.

OLIVEIRA, Letícia Sousa de; GRADVOHL, André Leon Sampaio. **Automatic analysis of magnetograms for identification and classification of active regions using Deep Learning**. 2020. Repositorio da producao cientifica e intelectual da UNICAMP – Universidade Estadual de Campinas, 2020.

PACHECO, C. A. R., & Pereira, N. S. (2018). **Deep Learning Conceitos e Utilização nas Diversas Áreas do Conhecimento**. Revista Ada Lovelace, 2, 34–49. Recuperado de: <<http://anais.unievangelica.edu.br/index.php/adalovelace/article/view/4132>>.

PONTI, M. A.; COSTA, G. B. P.. **Tópicos em Gerenciamento de Dados e Informações**. ICMC - Universidade de São Paulo, São Carlos, SP – SBC, 1.ed, 2017.

PRECISION, Recall, F1 score, True Positive|Deep Learning Tutorial 19 (Tensorflow2.0, Keras & Python). Direção de Dhaval Patel. Produção de Dhaval Patel. Roteiro: Dhaval Patel. Estados Unidos: Codebasics, 2021. Son., color. Legendado. Disponível em: Disponível em: <https://youtu.be/2osIZ-dSPGE>. Acesso em: 28 fev. 2022.

SANTOS, H. G. **Comparação da performance de algoritmos de machine learning para a análise preditiva em saúde pública e medicina**. 2018. 206 f. Tese (Doutorado) - Faculdade de Saúde Pública, Universidade de São Paulo, São Paulo, 2018.

SHAIKH, Rahil. **Cross Validation Explained: Evaluating estimator performance**. Disponível em: <https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85>. Acesso em: 23 abr. 2022.