

**UNIVERSIDADE DE SÃO PAULO**

Instituto de Ciências Matemáticas e de Computação

## Aplicação de redes neurais artificiais para classificação de falhas do LIGO

**Xirliane Muniz Vasconcelos**

Monografia - MBA em Inteligência Artificial e Big Data



**Xirliane Muniz Vasconcelos**

# **Aplicação de redes neurais artificiais para classificação de falhas do LIGO**

Monografia apresentada ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial

Orientador: Prof. Dr. Renato Tinós

**Versão original**

**São Carlos**

**2023**

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi  
e Seção Técnica de Informática, ICMC/USP,  
com os dados inseridos pelo(a) autor(a)

V331a Vasconcelos, Xirliane Muniz  
Aplicação de redes neurais artificiais para  
classificação de falhas do LIGO / Xirliane Muniz  
Vasconcelos; orientador Renato Tinós. -- São  
Carlos, 2023.  
71 p.

Trabalho de conclusão de curso (MBA em  
Inteligência Artificial e Big Data) -- Instituto de  
Ciências Matemáticas e de Computação, Universidade  
de São Paulo, 2023.

1. Classificação de imagens. 2. LIGO. 3. Redes  
neurais artificiais. 4. Redes neurais  
convolucionais. I. Tinós, Renato , orient. II.  
Título.

**Xirliane Muniz Vasconcelos**

**Application of artificial neural networks for the  
classification of LIGO glitches**

Monograph presented to the Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, as part of the requirements for obtaining the title of Specialist in Artificial Intelligence and Big Data.

Concentration area: Artificial Intelligence

Advisor: Prof. Dr. Renato Tinós

**Original version**

**São Carlos**

**2023**



## AGRADECIMENTOS

O que seria de mim sem o apoio e amor da minha família? Por isso, agradeço imensamente a ela por sempre estar ao meu lado. Aos meus pais, minhas irmãs, minhas primas, meus tios e minhas tias, expresso aqui a minha mais sincera e profunda gratidão.

O que seria de mim sem meus anjinhos de patas? Agradeço à todos que estão ao meu lado, me fazendo companhia e carinho, assim como à todos que estarão para sempre em meu coração.

O que seria de mim sem o refúgio daqueles que me acolhem nos momentos de crises? Agradeço à Juliana, à Luana, à Maiara por aguentarem minhas lamentações e angústias e por serem minhas confidentes com quem posso me amparar.

O que seria de mim sem a existência dos meus amigos? Agradeço-lhes por serem em seu completo significado do que é existir, apesar da minha escassa companhia, minha amizade e admiração por eles em nada diminuíram. À Mariana e à Cristiane, que sempre são tão incríveis e que me incentivam nos meus planos, agradeço pela dádiva de tê-las em minha vida.

O que seria de mim sem um farol para iluminar o meu caminho? Agradeço imensamente ao professor Renato por ser este guia nesta minha jornada. Obrigada pela orientação e pela atenção prestada ao longo do projeto, pois foram primordiais para que eu conseguisse concluí-lo.



*“Le savant n’est pas l’homme qui fournit les vraies réponses,  
c’est celui qui pose les vraies questions.”*

*Claude Lévi-Strauss*



## RESUMO

VASCONCELOS, X.M. **Aplicação de redes neurais artificiais para classificação de falhas do LIGO**. 2023. 71p. Monografia (MBA em Inteligência Artificial e Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

Com a primeira observação de uma onda gravitacional feita em 2016, através dos observatórios de ondas gravitacionais por interferometria laser (LIGO), foi estabelecido um novo campo da astronomia observacional através da detecção de ondas gravitacionais. Os detectores do projeto LIGO detectam dentre outros ruídos, os ruídos oriundos de fontes transientes e não Gaussianas, conhecidas como falhas. O projeto *Gravity Spy* classificou em famílias morfológicas essas falhas, definindo 22 classes distintas. Desta forma, o presente trabalho visa a aplicação de redes neurais artificiais para classificação das imagens disponibilizadas pelo projeto *Gravity Spy* com objetivo de classificar os diferentes tipos de classes das falhas. Utilizou-se diferentes tipos de arquiteturas de redes convolucionais para verificar a validade dos modelos na classificação das falhas. Alguns desses modelos de rede se mostraram ineficientes na classificação das falhas, porém algumas arquiteturas de redes convolucionais como as definidas como sendo o modelo de rede VI, o modelo de rede VII e o modelo de rede VIII apresentaram boa capacidade de generalização e resultados satisfatórios na classificação das classes de falhas, apesar do desbalanceamento do conjunto de dados utilizado.

**Palavras-chave:** Classificação de imagem. LIGO. Redes neurais artificiais. Redes neurais convolucionais.



## ABSTRACT

VASCONCELOS, X.M. **Application of artificial neural networks for the classification of LIGO glitches**. 2023. 71p. Monograph (MBA in Artificial Intelligence and Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

With the first observation of a gravitational wave in 2016 through the Laser Interferometer Gravitational-Wave Observatory (LIGO), a new field of observational astronomy was established through the detection of gravitational waves. The detectors in the LIGO project capture, among other noises, those originating from transient and non-Gaussian sources, known as glitches. The Gravity Spy project categorized these glitches into morphological families, defining 22 distinct classes. Thus, this study aims to apply artificial neural networks to classify images provided by the Gravity Spy project, with the goal of categorizing the different glitch classes. Various convolutional neural network architectures were employed to assess the validity of models in glitch classification. Some of these network models proved inefficient in classifying glitches; however, convolutional neural network architectures such as those defined as the VI model, the VII model, and the VIII model demonstrated good generalization ability and satisfactory results in classifying glitch classes, despite the imbalance in the utilized dataset.

**Keywords:** Image classification. LIGO. Artificial neural networks. Convolutional neural networks.



## LISTA DE FIGURAS

Figura 1 – <i>Layout</i> de um interferômetro a laser Michelson . . . . .	29
Figura 2 – Interferômetro a laser Michelson com cavidades de Fabry Perot . . . . .	30
Figura 3 – Interferômetro a laser Michelson com cavidades de Fabry Perot e espe- lhos de reciclagem de energia . . . . .	31
Figura 4 – Interferência construtiva entre duas ondas com mesma amplitude . . . . .	32
Figura 5 – Interferência destrutiva entre duas ondas com mesma amplitude . . . . .	32
Figura 6 – Interferência entre duas ondas com amplitudes e fases diferentes . . . . .	32
Figura 7 – Interferencia destrutiva no interferômetro do LIGO . . . . .	33
Figura 8 – Interferencia construtiva no interferômetro do LIGO . . . . .	33
Figura 9 – Modelos de redes neurais convolucionais utilizadas no artigo do Bahaadini	35
Figura 10 – Espectrograma de frequência em função do tempo gerado através do algoritmo Omicron . . . . .	37
Figura 11 – 22 classes das falhas do LIGO . . . . .	38
Figura 12 – (a) Acurácia do modelo I (b) Função de perda do modelo I . . . . .	47
Figura 13 – Matriz de confusão dos tipos de falhas de ondas gravitacionais do modelo I	48
Figura 14 – (a) Acurácia do modelo II (b) Função de perda do modelo II . . . . .	50
Figura 15 – Matriz de confusão dos tipos de falhas de ondas gravitacionais do modelo II	50
Figura 16 – (a) Acurácia do modelo III (b) Função de perda do modelo III . . . . .	52
Figura 17 – Matriz de confusão dos tipos de falhas de ondas gravitacionais do modelo III . . . . .	53
Figura 18 – (a) Acurácia do modelo IV (b) Função de perda do modelo IV . . . . .	54
Figura 19 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo IV . . . . .	55
Figura 20 – (a) Acurácia do modelo V (b) Função de perda do modelo V . . . . .	57
Figura 21 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo V . . . . .	57
Figura 22 – (a) Acurácia do modelo VI (b) Função de perda do modelo VI . . . . .	59
Figura 23 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo VI . . . . .	60
Figura 24 – (a) Acurácia do modelo VII (b) Função de perda do modelo VII . . . . .	62
Figura 25 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo VII . . . . .	62
Figura 26 – (a) Acurácia do modelo VIII (b) Função de perda do modelo VIII . . . . .	64
Figura 27 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo VIII . . . . .	65



## LISTA DE TABELAS

Tabela 1 – Quantidade de amostras das classes nos conjuntos de treinamento, teste e validação . . . . .	39
Tabela 2 – Arquiteturas dos modelos de redes convolucionais . . . . .	43
Tabela 3 – Matriz de Confusão . . . . .	45
Tabela 4 – Formato da saída do modelo de rede I . . . . .	47
Tabela 5 – Métricas das classes do modelo de rede I . . . . .	49
Tabela 6 – Formato do modelo de rede II . . . . .	49
Tabela 7 – Métricas das classes do modelo de rede II . . . . .	51
Tabela 8 – Formato do modelo de rede III . . . . .	52
Tabela 9 – Formato do modelo de rede IV . . . . .	54
Tabela 10 – Métricas das classes do modelo de rede IV . . . . .	56
Tabela 11 – Formato da saída do modelo de rede V . . . . .	56
Tabela 12 – Métricas das classes do modelo de rede V . . . . .	58
Tabela 13 – Formato do modelo de rede VI . . . . .	59
Tabela 14 – Métricas das classes do modelo de rede VI . . . . .	61
Tabela 15 – Formato do modelo de rede VII . . . . .	61
Tabela 16 – Métricas das classes do modelo de rede VII . . . . .	63
Tabela 17 – Formato do modelo de rede VIII . . . . .	64
Tabela 18 – Métricas das classes do modelo de rede VIII . . . . .	66
Tabela 19 – Métricas do conjunto de testes dos modelos . . . . .	67



## LISTA DE ABREVIATURAS E SIGLAS

AC	Acurácia
CNNs	Redes neurais convolucionais
DNNs	Redes neurais profundas
FN	Falso negativo
FP	Falso positivo
GANs	Redes adversária generativa
LIGO	Observatório de ondas gravitacionais por interferômetro laser
F1	F1-score
MF1	Macro F1-Score
MLPs	Perceptron multicamadas
RBF	Função de base radial
ReLU	Função de ativação linear retificada
SNR	Sinal-ruído
SVM	Máquina de vetores de suporte
TFN	Taxa de falso negativo
TFP	Taxa de falso positivo
TTP	Taxa de verdadeiro positivo
TTN	Taxa de verdadeiro negativo
VN	Verdadeiro negativo
VNP	Valor preditivo negativo
VP	Verdadeiro positivo
VPP	Valor preditivo positivo



## SUMÁRIO

1	<b>INTRODUÇÃO</b>	23
2	<b>FUNDAMENTOS E TRABALHOS RELACIONADOS</b>	25
2.1	<b>Redes Neurais Artificiais</b>	25
2.2	<b>Redes Neurais Convolucionais</b>	26
2.2.1	Camada de convolução	26
2.2.2	Camada de agrupamento ( <i>Pooling</i> )	27
2.2.3	Preenchimento com zeros ( <i>Zero padding</i> )	28
2.2.4	Camada conectada	29
2.3	<b>Interferômetros do LIGO</b>	29
2.4	<b>Interferometria</b>	31
2.5	<b>Trabalhos Relacionados</b>	34
3	<b>OS DADOS DO <i>GRAVITY SPY</i> E A APLICAÇÃO DE REDES NEURAIAS ARTIFICIAIS</b>	37
3.1	O conjunto de dados do <i>Gravity Spy</i>	37
3.2	A morfologia das falhas	38
3.3	Classificação usando CNN	42
4	<b>ANÁLISE DE RESULTADOS</b>	45
4.1	Critérios de avaliação do método proposto	45
4.2	Resultados dos Modelos	46
4.3	Modelo de Rede Convolucional I	47
4.4	Modelo de Rede Convolucional II	49
4.5	Modelo de Rede Convolucional III	52
4.6	Modelo de Rede Convolucional IV	53
4.7	Modelo de Rede Convolucional V	56
4.8	Modelo de Rede Convolucional VI	59
4.9	Modelo de Rede Convolucional VII	61
4.10	Modelo de Rede Convolucional VIII	64
4.11	Comparação dos modelos	66
5	<b>CONCLUSÃO</b>	69
	<b>REFERÊNCIAS</b>	71



## 1 INTRODUÇÃO

Com a primeira observação de uma onda gravitacional feita pelos observatórios de ondas gravitacionais por interferometria laser, feita de acordo com ABBOTT *et al.* (2016), foi provada empiricamente uma das previsões da teoria da relatividade geral de Einstein, consolidando ainda mais sua validade. Desde esta primeira observação, têm sido observadas diversas outras ondas gravitacionais originárias de diferentes eventos astronômicos, como fusões de buracos negros binários ou colisões de estrelas de nêutrons. Com estas observações, estabeleceu-se de forma contundente um novo campo da astronomia observacional, que é o da observação de fenômenos astronômicos através da detecção de ondas gravitacionais.

O projeto LIGO consiste em dois interferômetros localizados em Hanford, Washington, e outro em Livingston, Louisiana, que funcionam concomitantemente para a detecção de ondas gravitacionais. Os detectores usam a interferometria a laser para medir as pequenas ondulações no espaço-tempo causadas pela passagem das ondas gravitacionais.

Os detectores foram projetados para perceber alterações de distância da ordem de  $10^{-21}$  metros. Contrapondo essas distâncias com o diâmetro dos núcleos atômicos, que é da ordem de  $10^{-15}$  metros, tem-se uma ideia da sensibilidade exigida para a detecção de ondas gravitacionais. Visando garantir condições ideais para que essa sensibilidade seja efetiva, estes detectores possuem diversos mecanismos de isolamento para evitar interferências de ruídos instrumentais ou ambientais.

Apesar de todo o aparato para a medição e detecção de ondas gravitacionais ser isolado de perturbações que não sejam advindas de ondas gravitacionais, os detectores LIGO ainda sofrem perturbações que causam ruídos e que interferem na identificação precisa das ondas gravitacionais. Dentre estes ruídos detectados tem-se as fontes de ruído transientes e não Gaussianas, conhecidas como falhas (*glitches*), que por vezes acabam mascarando e até mesmo imitando os sinais de ondas gravitacionais, sendo um grande problema para a correta identificação de verdadeiras ondas gravitacionais.

Devido a gravidade destas falhas, diversos trabalhos dentro do meio científico tentam identificar, caracterizar e classificá-las. Dentre estes esforços o projeto *Gravity Spy* foi desenvolvido visando classificar em famílias morfológicas essas falhas que afligem os detectores de ondas gravitacionais. O projeto *Gravity Spy* tanto fez uso de aprendizado de máquina para classificação destas falhas, como contou com o auxílio de voluntários para identificar manualmente novas falhas através da plataforma Zooniverse ZONIVERSE (2023).

Os dados disponibilizados pelo projeto na plataforma Kaggle HARRAND (2020) consistem numa coleção de imagens de espectrogramas de frequência em função do tempo

de falhas do LIGO e suas respectivas características morfológicas que levaram à definição de 22 classes distintas de falhas.

Desta forma, entendendo a importância da identificação destas falhas para pesquisas atuais na observação de fenômenos astronômicos, ou seja, a identificação correta de falhas evita que estas interfiram na detecção de ondas gravitacionais pelos detectores, aumentando consequentemente a eficácia das observações. Neste trabalho, foram utilizados os dados disponibilizados pelo projeto *Gravity Spy* com as diferentes classes de falhas e aplicadas redes neurais artificiais para a classificação destas falhas. Assim, este trabalho pretende investigar o conjunto de falhas utilizando redes neurais artificiais visando um melhor algoritmo para a distinção das mesmas.

No segundo capítulo, são apresentados os fundamentos relacionados as redes neurais artificiais. Primeiramente, é explicado como funcionam as MLPs e posteriormente como evoluíram para o desenvolvimento das chamadas redes neurais convolucionais. Nesta parte, são discutidas as camadas que compõem uma rede neural convolucional, como a camada de convolução, a camada de agrupamento (*pooling*), o preenchimento com zeros (*zero padding*) e a camada conectada. Neste capítulo, ainda é discutido como funcionam os interferômetros do LIGO, assim como são explicados fundamentos da interferometria. Por fim, são abordados os trabalhos relacionados que aplicaram redes neurais artificiais no conjunto de dados do *Gravity Spy*.

No terceiro capítulo, são abordados os critérios para a geração das imagens do conjunto de dados do projeto do *Gravity Spy*, é explicada a morfologia das classes das falhas e como é feito o treinamento da classificação das falhas usando oito modelos de arquiteturas diferentes de redes convolucionais.

No quarto capítulo, primeiramente, são mostrados os critérios de avaliação dos modelos. Posteriormente são apresentados os resultados obtidos pelos treinamentos de cada um dos oito modelos propostos, finalizando o capítulo com uma comparação entre os resultados obtidos entre os diferentes modelos.

No quinto capítulo, é realizada a síntese dos resultados obtidos e inferida a conclusão, assim como são expostos possíveis desdobramentos para estudos futuros.

## 2 FUNDAMENTOS E TRABALHOS RELACIONADOS

### 2.1 Redes Neurais Artificiais

As redes neurais artificiais foram desenvolvidas tendo como premissa o funcionamento biológico do neurônio, segundo GOODFELLOW; BENGIO; COURVILLE (2016), elas são formadas por neurônios artificiais conectados entre si criando assim uma rede. Os pesos das conexões entre os neurônios podem ser desenvolvidos através de aprendizado supervisionado, no qual cada entrada é associada uma saída desejada, e a rede aprende a realizar a associação entre cada entrada e sua respectiva saída, este tipo de aprendizado é usado para aproximação de funções, regressão e classificação, ou através de aprendizado não supervisionado, no qual o processo de aprendizado de padrões não necessita de rótulos pré definidos, ou ainda através de aprendizado por reforço, no qual durante o processo de aprendizado são preditos punições ou recompensas para respectivos erros ou acertos cometidos pela rede.

As redes neurais artificiais mostram-se relevantes com aplicações nos mais diversos setores, tais como o setor financeiro, na concessão de empréstimos e crédito ou na detecção de transações fraudulentas; no setor da saúde, no auxílio de diagnósticos de doenças. Dentre as redes neurais o perceptron multicamadas (MLPs) pode ser usado para classificação e regressão e surge como generalização do perceptron de camada única. MLPs são redes neurais que possuem uma camada de entrada e uma ou mais camadas ocultas com nós computacionais formados usualmente por funções não-lineares.

MLPs são treinados de forma supervisionada e o cálculo de erro para uma unidade da camada oculta é feito através do algoritmo *Backpropagation* que requer duas fases que formam um ciclo:

- Uma de propagação da ativação da camada de entrada para a escondida e em seguida para a de saída, conhecida como passo à frente. Neste passo, o vetor de entrada é aplicado aos nós sensoriais da rede e propagado através da rede, camada por camada, mantendo os pesos sinápticos fixos.
- Outra em que o erro é calculado para as unidades de saída e retropropagado para as unidades ocultas e para as de entrada, conhecida como passo para trás. Neste passo pesos sinápticos são ajustados de acordo com uma regra de correção de erro que subtrai a resposta real da resposta desejada, produzindo um sinal, conhecido como sinal de erro, que é propagado de volta através da rede, em sentido contrário aos estímulos.

MLPs sofrem com o número de parâmetros, pois estes causam grande impacto em relação ao processamento e utilização de memória. A quantidade de parâmetros aprendidos e mantidos na memória por uma rede MLP durante o treinamento pode ser calculada por:

$$q \times l \times h + q + N \times q + N \quad (2.1)$$

onde  $q$  é a quantidade de neurônios na camada oculta,  $l$  é a largura da imagem de entrada da rede,  $h$  é a altura da imagem de entrada da rede e  $N$  é a quantidade de classes existentes a serem classificadas.

Desta forma, se uma rede MLP com uma camada oculta de 100 neurônios recebe como entrada uma imagem de tamanho  $24 \times 24$  que possui, portanto, 576 pixels e 3 classes distintas, isso implica que ela necessita que 58.003 parâmetros sejam aprendidos e mantidos na memória durante o treinamento da rede, gerando um grande custo computacional.

## 2.2 Redes Neurais Convolucionais

Na história das redes neurais artificiais, houve uma evolução das arquiteturas de redes, e após o desenvolvimento das MLPs, surgem as chamadas redes neurais convolucionais (CNNs). Uma rede convolucional é um tipo especializado de rede neural, treinada com retropropagação e empregada com sucesso em classificação de imagens.

Dentre alguns modelos de redes neurais convolucionais que se mostraram eficientes na classificação de imagens, destacam-se a AlexNet (2012), a VGGNet (2014), entre outras. O aprendizado de uma CNN ocorre de forma supervisionada, de acordo com GOODFELLOW; BENGIO; COURVILLE (2016) a terminologia de rede convolucional define: a entrada, que geralmente é uma matriz multidimensional de dados, por exemplo uma imagem em 3D pode ser caracterizada numa matriz com  $l$  (largura)  $\times$   $h$  (altura)  $\times$   $p$  (profundidade); o filtro (*kernel*), que geralmente é uma matriz multidimensional de parâmetros que são adaptados pelo algoritmo de aprendizado, por exemplo, para uma imagem 3D pode-se determinar um filtro  $k \times k \times p$ , com  $k$  sendo menor que  $l$  e  $h$  e que obedeça um determinado passo (*stride*) na imagem de entrada. Por exemplo, se o passo for igual a 1, temos que todos os pixels são filtrados pelo neurônio, caso o passo seja maior que 1 há um salto no número de pixels em determinada direção, e a saída é chamada de mapa de características que, empilhadas, formam um tensor que servirá de entrada da próxima camada.

### 2.2.1 Camada de convolução

As CNNs são chamadas desta forma devido ao fato de ser utilizada, em pelo menos uma de suas camadas, a operação matemática chamada convolução no lugar da usual multiplicação de matrizes. As redes convolucionais têm interações esparsas, ou

seja, utilizam filtros que por vezes são menores que o tamanho da entrada, mas com a profundidade igual à da entrada, isso acaba afetando o campo receptivo local, reduzindo assim os requisitos de memória do modelo e melhorando sua eficiência. Observa-se que este é um comportamento contrário ao das redes neurais tradicionais que usam multiplicação de matriz, onde cada parâmetro descreve a interação entre uma unidade de entrada com uma unidade de saída.

Além disso, as redes convolucionais usam o compartilhamento de parâmetros, isto é, elas usam o mesmo parâmetro para mais de uma função em um modelo. Em uma rede neural tradicional, cada elemento da matriz de peso é multiplicado por um elemento da entrada e usado exatamente uma única vez ao computar a saída de uma camada. Já com o compartilhamento de parâmetros, o valor do peso aplicado a uma entrada está vinculado ao valor de um peso aplicado em outro lugar. Assim, em uma rede neural convolucional, com o compartilhamento de parâmetros, ao invés de aprender um conjunto separado de parâmetros para cada local, a rede aprende apenas um conjunto, reduzindo os requisitos de armazenamento dos parâmetros do modelo.

O compartilhamento de parâmetros faz com que a camada tenha uma propriedade chamada equivariância à translação. Por exemplo ao processar imagens, a convolução cria um mapa 2-D de onde certas características aparecem na entrada, ao mover o objeto na entrada, sua representação se moverá na mesma quantidade na saída.

### 2.2.2 Camada de agrupamento (*Pooling*)

Importante ressaltar que as redes neurais convolucionais geralmente aplicam uma série de convoluções em paralelo. Isso ocorre porque a convolução com um único filtro pode extrair apenas um tipo de característica, embora frequentemente o desejado seja que cada camada da rede extraia muitos tipos de características, em muitos locais, por exemplo, uma imagem colorida tem intensidade de vermelho, verde e azul em cada pixel.

Uma função de agrupamento substitui a saída da rede em um determinado local por uma estatística resumida das saídas próximas. Por exemplo, a operação de agrupamento máxima (*Max pooling*) faz com que uma unidade de agrupamento gere a ativação máxima na região de entrada  $2 \times 2$ , reduzindo o tamanho da entrada, permitindo que o filtro opere em regiões maiores da imagem. Outras funções populares de agrupamento incluem a média de uma vizinhança retangular (*Average pooling*), que é uma operação de agrupamento que calcula o valor médio para *patches* de um mapa de características e o usa para criar um mapa de características reduzido; a norma L2 de uma vizinhança retangular que, em vez de tomar a ativação máxima de uma região  $2 \times 2$  de neurônios, toma a raiz quadrada da soma dos quadrados das ativações na região  $2 \times 2$ , condensando as informações da camada convolucional, ou o agrupamento global (*Global pooling*) que obtém um valor por canal, como se o tamanho do agrupamento fosse igual às dimensões laterais, por exemplo

numa entrada com  $30 \times 40 \times 100$ , a saída terá 100 dimensões.

O agrupamento ajuda a tornar a representação aproximadamente invariante à pequenas translações da entrada, ou seja, ao movimentar a entrada por uma quantidade pequena os valores da maioria das saídas agrupadas não mudam, sendo uma propriedade útil para saber se alguma característica está presente, sem saber sua localização exata. A vantagem do agrupamento se deve ao fato dele resumir as respostas em uma vizinhança inteira, melhorando a eficiência computacional da rede porque a próxima camada tem aproximadamente menos entradas para processar.

Desta forma, após realizada várias convoluções em paralelo para produzir um conjunto de ativações lineares, onde cada mapa de ativação linear é executado por meio de uma função, por exemplo a função de ativação linear retificada (ReLU), aplica-se uma função de agrupamento para obter um valor resumido estatisticamente dos resultados dessas ativações lineares.

### 2.2.3 Preenchimento com zeros (*Zero padding*)

Uma característica essencial de qualquer implementação de uma rede convolucional é a capacidade de ampliar preenchendo com zeros o vetor de entrada de forma que o tamanho de saída seja igual ao da entrada. Sem esse recurso, a largura da representação diminui em um pixel a menos que a largura do filtro em cada camada o que reduz rapidamente a extensão espacial da rede, fazendo com que por vezes opte-se por filtros menores o que acaba limitando o poder expressivo da rede. Assim, o preenchimento com zeros na entrada permite que a largura do filtro e o tamanho da saída sejam controlados de forma independentes.

As configurações de preenchimento com zeros possuem os seguintes casos especiais:

- Nenhum preenchimento com zero é usado, fazendo com que o filtro de convolução percorra apenas as posições em que todo o filtro esteja contido inteiramente na imagem, com isso todos os pixels na saída são uma função do mesmo número de pixels na entrada. Neste caso, há um comportamento mais regular, porém isso faz com que o tamanho da saída diminua a cada camada. Se a imagem de entrada tiver largura  $l$  e o filtro tiver largura  $k$ , a saída será de largura  $l - k + 1$ . Com isso, para filtros grandes, a taxa desse encolhimento é alta, limitando o número de camadas convolucionais que podem ser incluídas na rede.
- Preenchimento de zero apenas para manter o tamanho da saída igual ao tamanho da entrada. Neste caso, a rede não fica limitada em termos da quantidade de camadas convolucionais, pois a operação de convolução não modifica a possibilidade arquitetural disponível para a próxima camada. O porém desta abordagem é que os pixels de entrada próximos à borda influenciam menos pixels de saída do que os

pixels de entrada próximos ao centro, fazendo com que estes pixels da borda sejam sub-representados no modelo.

- Preenchimento de zeros suficientes para que cada pixel seja visitado  $k$  vezes em cada direção, resultando em uma imagem de saída de largura  $l + k - 1$ . Nesse caso, os pixels de saída próximos à borda são uma função de menos pixels do que os pixels de saída próximos ao centro. Isso pode dificultar o aprendizado de um único filtro com bom desempenho em todas as posições no mapa de características convolucionais.

#### 2.2.4 Camada conectada

A camada final na rede é uma camada totalmente conectada (*fully connected*) que conecta todos os neurônios da camada de agrupamento a cada um dos neurônios de saída.

### 2.3 Interferômetros do LIGO

Sabendo como funcionam as redes neurais convolucionais, outro fator importante deste trabalho é entender como ocorre a detecção de ondas gravitacionais através do projeto LIGO. O projeto LIGO possui dois interferômetros idênticos, o primeiro está instalado em Hanford, região desértica de Washington e outro em Livingston, região de florestas de Louisiana. Estes estão separados por 3002 quilômetros de distância, evitando que eles detectem as mesmas vibrações ao mesmo tempo de fontes oriundas das atividades humanas, como um caminhão passando numa estrada próxima, ou de causas naturais, como um terremoto em uma outra parte do planeta, evitando assim que estas camuflam os sinais indicadores de ondas gravitacionais. No entanto, esta distância garante que ambos os interferômetros possam sentir uma vibração de onda gravitacional, da ordem de  $10^{-21}$ , que serão sinais idênticos que ocorreram em um mesmo instante de tempo em ambos os locais.

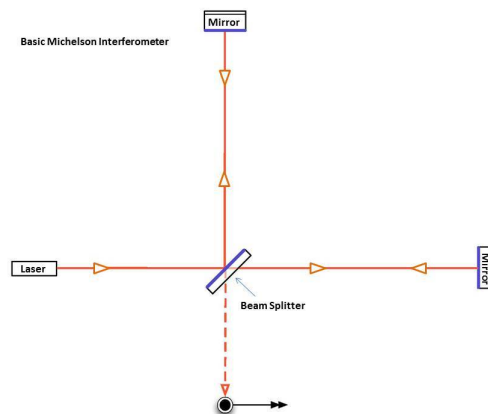


Figura 1 – *Layout* de um interferômetro a laser Michelson

Fonte: Caltech/MIT/LIGO Lab

Cada interferômetro do projeto LIGO possui a configuração interferométrica semelhante à usada por Michelson e Morley (Fig. 1) na década de 1880, que serviu para provar a não existência do éter e a propagação da luz (radiação eletromagnética) no vácuo. Sua configuração consiste em dois tubos de vácuo de aço de 4 km, cuja estrutura é feita de concreto de 30 cm de altura que protege os tubos do meio ambiente. Essa configuração é ideal, dado que o interferômetro não precisa apontar para locais específicos no espaço, pois a onda gravitacional pode vir de qualquer direção, inclusive do solo.

A configuração básica de um interferômetro a laser Michelson consiste em um laser, um divisor de feixe de laser, espelhos nas extremidades dos braços que refletem a luz e um fotodetector (o ponto preto) que registra o padrão de interferência dos feixes de luz refletidos agora sobrepostos. Além disso, algumas melhorias diferenciam um interferômetro LIGO de um interferômetro Michelson típico, seus braços medem 4 km de comprimento e para que sejam capazes de medir uma onda gravitacional são incluídas as chamadas "cavidades de Fabry Perot"(Fig. 2), aumentando a sensibilidade do interferômetro.

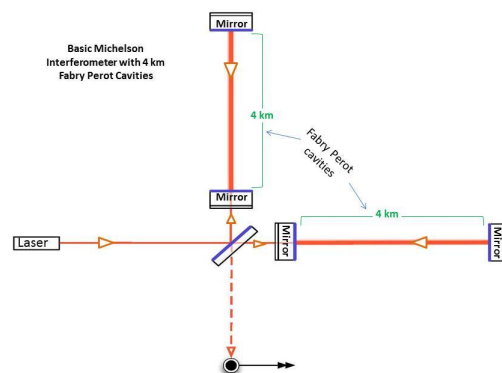


Figura 2 – Interferômetro a laser Michelson com cavidades de Fabry Perot

Fonte: Caltech/MIT/LIGO Lab

Na Fig. 2, pode ser visto que um espelho adicional é colocado para facilitar múltiplas reflexões do laser, mantendo-o circulando dentro do interferômetro por mais tempo e aumentando a distância percorrida em cada braço perto do divisor de feixe, a 4 km do espelho no final desse braço. Este espaço de 4 km de extensão constitui a cavidade Fabry Perot.

Depois de entrar no instrumento através do divisor de feixe, o laser em cada braço salta entre esses dois espelhos cerca de 300 vezes antes de se fundir com o feixe do outro braço. Essas reflexões servem para acumular a luz do laser dentro do interferômetro, o que aumenta a sensibilidade do LIGO, da mesma forma que servem para aumentar a distância efetiva percorrida por cada laser de 4 km para 1200 km. Isso é importante porque quanto

mais longos os braços do interferômetro, maiores as mudanças absolutas que as ondas gravitacionais fazem no comprimento total do braço.

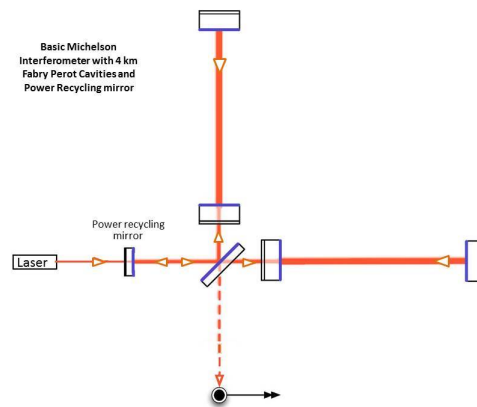


Figura 3 – Interferômetro a laser Michelson com cavidades de Fabry Perot e espelhos de reciclagem de energia

Fonte: Caltech/MIT/LIGO Lab

Além disso, para que o laser do LIGO aumente sua potência são usados espelhos de reciclagem (Fig. 3), fazendo com que a potência inicial do laser de cerca de 40 Watts chegue até perto de 750 kW. Por simplicidade na Fig. 3 aparece apenas um espelho de reciclagem de energia, porém são usados vários espelhos de reciclagem de energia. Para aumentar a potência do feixe de laser dentro das cavidades Fabry Perot, a luz do laser passa pelo lado transparente de um espelho de reciclagem de energia para o divisor de feixe e depois para os braços do interferômetro. O alinhamento do instrumento e os revestimentos do espelho garantem que quase toda a luz do laser que entra nos braços siga um caminho de volta para o lado reflexivo do espelho de reciclagem de energia, e não para o fotodetector. Como a energia do laser está constantemente entrando no interferômetro, o espelho de reciclagem de energia reflete continuamente a luz do laser que percorreu o instrumento de volta ao interferômetro. Este aumento na energia gerada pela reciclagem de energia resulta em franjas de interferência mais nítidas, tornando-as mais fáceis de serem identificadas quando se tratar de sinais indicadores de ondas gravitacionais.

## 2.4 Interferometria

O funcionamento dos interferômetros do projeto LIGO tem como base a interferometria laser. Primeiramente veja o que ocorre quando os picos e vales de duas ou mais ondas idênticas, isto é com mesma amplitude e frequência, se encontram. Nessa situação, seus picos se somam, formando uma onda maior, com tamanho igual à soma das suas amplitudes (alturas ou profundidades), esta é a chamada interferência construtiva (Fig. 4).

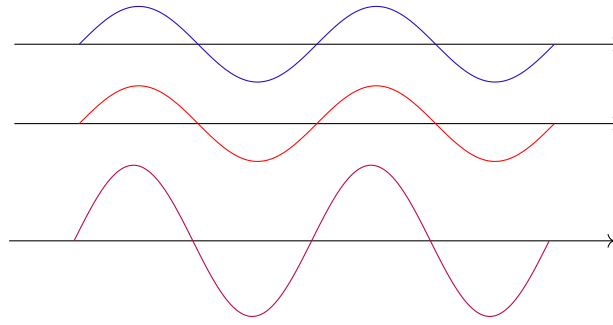


Figura 4 – Interferência construtiva entre duas ondas com mesma amplitude

Porém, no caso em que os picos de uma onda encontram os vales de outra onda idêntica, eles se anulam e há a chamada interferência destrutiva (Fig 5).

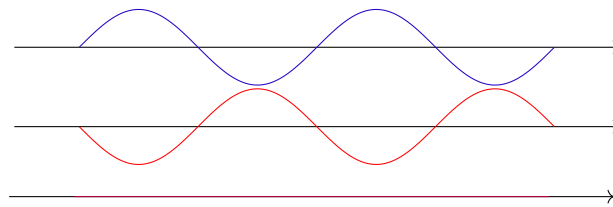


Figura 5 – Interferência destrutiva entre duas ondas com mesma amplitude

Além destes tipos, há o caso em que uma ou mais ondas não idênticas interagem ou o caso em que os picos e vales de uma onda não interagem completamente com os picos e vales da outra, neste caso forma-se um padrão de interferência mista (Fig. 6), cuja onda resultante é igual à soma das amplitudes das ondas em seus pontos de interseção.

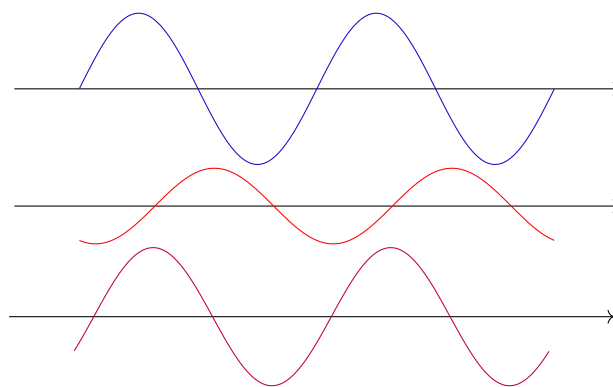


Figura 6 – Interferência entre duas ondas com amplitudes e fases diferentes

Nos interferômetros do LIGO, o que determina o alinhamento das ondas do feixe de laser quando elas se fundem é simplesmente a distância que elas percorrem antes de se fundirem. Se cada feixe percorrer exatamente a mesma distância ao longo de cada braço, quando os feixes forem fundidos novamente, suas ondas de luz interferirão de forma totalmente destrutiva (Fig. 7) e desta forma o detector não terá nenhuma

frequência de luz para detectar. Mas se, por algum motivo, os lasers não percorrerem as mesmas distâncias, suas ondas de luz não estarão mais sincronizadas quando se fundirem, produzindo algum padrão de interferência não destrutiva no fotodetector, sendo geradas franjas de interferência.

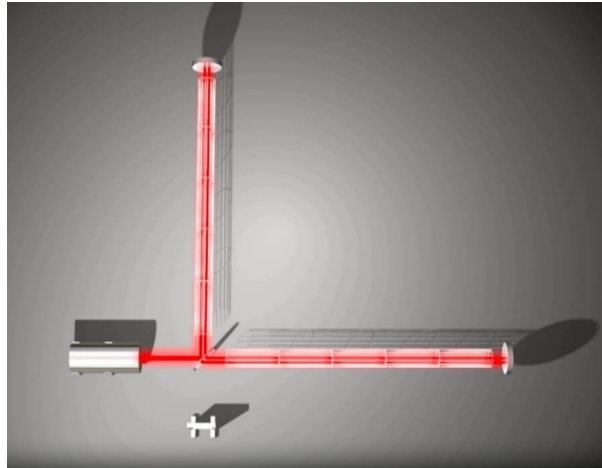


Figura 7 – Interferencia destrutiva no interferômetro do LIGO

Fonte: U.S. National Science Foundation (NSF)

Se os feixes estiverem percorrendo distâncias diferentes devido a uma onda gravitacional que esteja passando, fazendo com que os braços se alonguem e encurtem alternadamente, o fotodetector verá uma cintilação de luz, que dura apenas enquanto a onda gravitacional passante é detectável no interferômetro, variando em intensidade de nenhuma luz, onde quer que as ondas interfiram de forma totalmente destrutiva (Fig. 7), até a luz quase tão brilhante quanto o feixe de laser original, onde quer que as ondas interfiram de forma totalmente construtiva (Fig. 8).

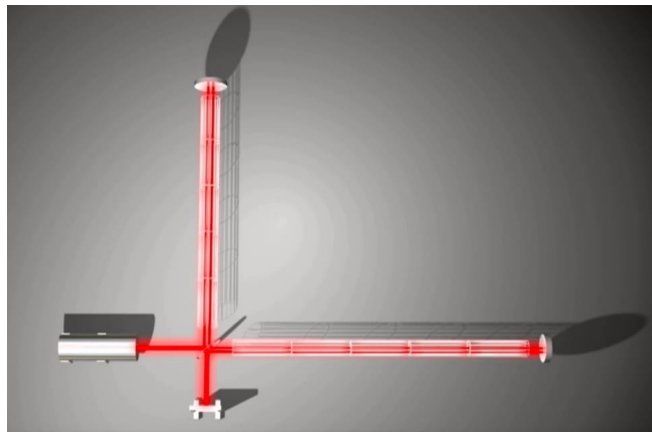


Figura 8 – Interferencia construtiva no interferômetro do LIGO

Fonte: U.S. National Science Foundation (NSF)

Esse padrão de interferência ocorre devido as ondas gravitacionais fazerem com que o próprio espaço se estique em uma direção e simultaneamente se comprima em uma direção perpendicular. Assim, ao passar uma onda gravitacional por um dos interferômetros do LIGO, ela fará com que um braço deste fique mais longo enquanto o outro fica mais curto, e vice-versa, para frente e para trás enquanto a onda estiver passando, fazendo com que a distância percorrida por cada feixe de laser também mude. Um feixe em um braço mais curto retornará ao divisor de feixe antes de um feixe em um braço mais longo, desta forma quando voltarem ao divisor de feixe, as ondas de luz estarão fora de fase resultando em uma cintilação de luz emergindo do interferômetro e produzindo as franjas de interferência.

## 2.5 Trabalhos Relacionados

Tendo em vista o funcionamento das CNNs, foi usado este tipo de rede neural no conjunto de dados disponibilizado pelo projeto *Gravity Spy*, com o intuito de que a rede aprenda a identificar os padrões das falhas e generalize as mesmas. Existem alguns estudos sobre esta base de dados usando métodos de aprendizado de máquina, como BAHADINI *et al.* (2018), que utiliza métodos como regressão logística, máquina de vetores de suporte (SVM) e redes neurais profundas (DNNs) e YAN; LEUNG; HUI (2022), que utiliza redes adversárias generativas (GANs).

O conjunto de dados do *Gravity Spy* utilizado por BAHADINI *et al.* (2018) possui 8.583 amostras de falhas, divididas em 6.008 amostras de treino, 1.288 de validação e 1.287 de teste. Devido o desbalanceamento das classes, estas foram forçadas a terem amostras proporcionalmente à sua distribuição em todos os conjuntos de treino, validação e teste.

O primeiro modelo de rede neural artificial utilizado em BAHADINI *et al.* (2018) é a regressão logística multinomial, devido ela não possuir hiperparâmetros não é utilizado o conjunto de validação. Deste modo, o modelo é apenas treinado usando o conjunto de treinamento e avaliado no conjunto de teste.

Nesta parte é analisado o desempenho da aplicação da regressão logística multinomial para diferentes resoluções, visando obter o melhor desempenho com o menor custo computacional. Dada que a resolução original dos espectrogramas é de  $470 \times 570$  pixels, segundo os resultados obtidos no artigo foi verificado que a resolução de  $140 \times 170$  pixels aparentemente compensa a precisão e custo computacional.

Além disso é mostrado que este modelo consegue aprender bem algumas classes, como *1080 Lines*, *Blip*, *Chirp*, enquanto que encontra dificuldade em classes como *None of the above*, *Paired doves* e *Wandering line*, sendo enfatizado que o modelo de regressão logística não consegue capturar as não linearidades existentes nos dados.

Depois é utilizada máquina de vetores de suporte (SVM) para o estudo da base de

dados, primeiro é utilizada SVM linear, onde os hiperparâmetros do modelo são ajustados por busca em grade e validação cruzada com *k-fold*, neste modelo o classificador é treinado usando apenas o conjunto de treinamento e o desempenho da classificação é verificado com o conjunto de teste. As acurácias de classificação da SVM linear, treinada e testada em imagens de diferentes durações, têm o melhor desempenho quando aplicada à chamada *view 2 (1s)* e o pior desempenho quando aplicada à *view 4 (4s)*, apresentando acurácias respectivamente de 96,19% e 93,16%.

Devido a complexidade presente nos dados é apresentada a SVM com *kernel*, que projeta as amostras para outro espaço de características onde podem ser separados linearmente. No artigo é usado a SVM com o *kernel* de função de base radial (RBF) para capturar a não linearidade das amostras do *Gravity Spy*. Para os melhores parâmetros encontrados do *kernel* RBF,  $C = 5,65$  e  $\gamma = 4e^{-5}$ , a classificação da SVM de *kernel* apresentou o melhor desempenho quando aplicada à *view 2 (1s)* apresentando acurácia de 97,12% e o pior desempenho quando aplicada à *view 4 (4s)*, com acurácia de 93,93%.

Posteriormente são usadas redes neurais profundas para tratamento da base de dados, para isso são usados dois modelos de arquiteturas de redes neurais convolucionais (Fig. 9) aplicadas para quatro diferentes durações.

Single view model	Merged-view model
Input $140 \times 170$	Input $280 \times 340$
$5 \times 5$ Convolutional layer (32) with reg.	$5 \times 5$ Convolutional layer (16) with reg.
$2 \times 2$ Maxpooling, 0.5 drop-out	$2 \times 2$ Maxpooling, 0.5 drop-out
$5 \times 5$ Convolutional layer (64) with reg.	$5 \times 5$ Convolutional layer (32) with reg.
$2 \times 2$ Maxpooling, 0.5 drop-out	$2 \times 2$ Maxpooling, 0.5 drop-out
Fully connected (256), 0.5 drop-out	$5 \times 5$ Convolutional layer (64) with reg.
Softmax (22)	$2 \times 2$ Maxpooling, 0.5 drop-out
	$5 \times 5$ Convolutional layer (64) with reg.
	$2 \times 2$ Maxpooling, 0.5 drop-out
	Fully connected (256), 0.5 drop-out
	Softmax (22)

Figura 9 – Modelos de redes neurais convolucionais utilizadas no artigo do Bahaadini

Fonte: BAHADINI *et al.* (2018)

É utilizado como inicializador para a camada convolucional a função *glorot uniforme* e usado a regularização L2. Com o *Keras Model Check Point*, o desempenho do modelo no conjunto de validação é verificado no final de cada época, atualizando os pesos com os melhores modelos caso haja melhora. No final de todas as iterações, os melhores pesos são carregados para o modelo final. No artigo o número de épocas é definido como 200 e o tamanho do lote é definido como 30. A acurácia da classificação das falhas para a *view 1*

(0,5 s) é 95,10%, para a *view 2* (1 s) é de 96,81%, para a *view 3* (2 s) é de 96,58%, para a *view 4* (4 s) é de 95,65% e para a *merged-view* é de 97,67%.

Por fim é desenvolvido um modelo chamado de *ensemble* final, combinando três classificadores básicos: dois modelos *merged-view*, um com o ponto de verificação do modelo definido em relação à precisão sobre o conjunto de validação e outro com o ponto de verificação do modelo definido em relação ao valor da função de perda sobre a validação e um modelos SVM com *kernel* treinado para visualização com duração de 1 segundo. Esse *ensemble* apresentou melhor desempenho que as abordagens tradicionais, apresentando acurácia de 98,21% no conjunto de testes.

Em YAN; LEUNG; HUI (2022) a classificação de espectrogramas é realizada usando redes adversárias generativas (GANs). No artigo é usada uma variante de GANs, a ProGAN, para gerar espectrogramas de alta resolução. Como resultados relevantes é apresentado o fato de que estes espectrogramas gerados pelo ProGAN são consistentes com os dados reais, aumentando a diversidade dos dados de treinamento, visando com isso superar os problemas da pequena quantidade de amostras e classes desequilibradas dos dados originais o que pode resultar consequentemente em uma melhor generalização na classificação das falhas.

Devido o aumento das amostras da base de dados com o uso da GAN é verificado que o desempenho obtido na classificação de aprendizado profundo é comparável ao obtido pelo modelo pré-treinado, sem necessitar do uso de técnicas como *transfer learning*. Além disso é confirmado que as classes com menor quantidade de amostras são impactadas positivamente ao incluir os espectrogramas gerados pela GAN para treinamento, reduzindo as flutuações na classificação destas classes menores.

### 3 OS DADOS DO *GRAVITY SPY* E A APLICAÇÃO DE REDES NEURAIAS ARTIFICIAIS

#### 3.1 O conjunto de dados do *Gravity Spy*

O conjunto de dados do *Gravity Spy* é composto por imagens de espectrogramas de frequência em função do tempo detectadas pelo projeto LIGO, como exemplo a imagem da classe *Blip* (Fig. 10). Conforme BAHADINI *et al.* (2018), estas falhas são medidas quando os ruídos transientes, detectados durante a medição de ondas gravitacionais nos detectores, excedem um certo limiar. As imagens são geradas através de um algoritmo, chamado *Omicron*, que identifica o excesso de energia transiente, se baseando na transformação Q ROBINET *et al.* (2020), onde ondas seno-Gaussianas são usadas para identificar excesso de ruído transitório nos dados e agrupá-los levando em consideração características como frequência do pico e relação sinal-ruído (SNR).

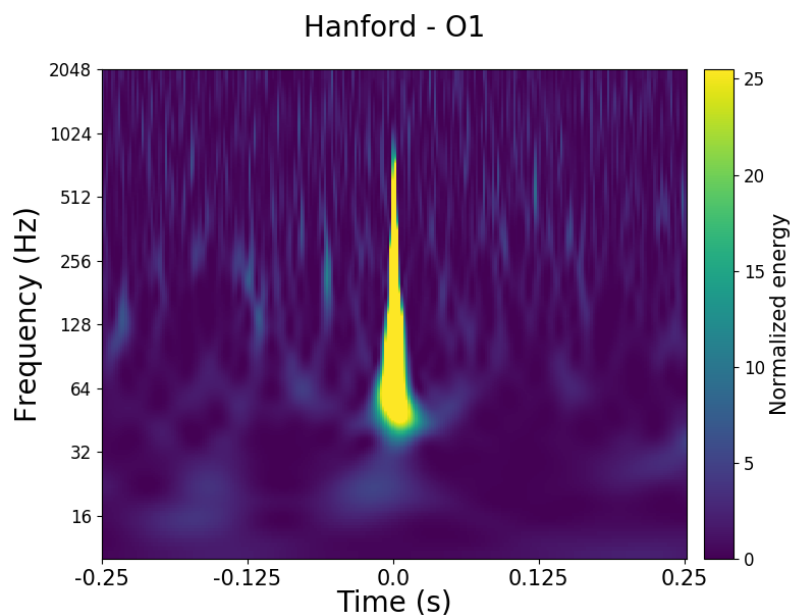


Figura 10 – Espectrograma de frequência em função do tempo gerado através do algoritmo Omicron

Fonte: Coughlin (2018)

Ao gerar estas imagens, não são consideradas as falhas que ocorreram fora dos períodos de tempo de observação do detector, ou com SNR abaixo de 7,5, dado que falhas abaixo desse limite são difíceis de classificar com base em sua morfologia devido a baixa resolutividade nas imagens e falhas cuja frequência de pico está fora da banda de frequência mais sensível do LIGO, de 10 Hz a 2048 Hz. Somente após aplicar esses filtros, as representações de frequência x tempo, chamadas de *Omega Scans* e baseadas na

transformação  $Q$ , são criadas e salvas como arquivos de imagem.

### 3.2 A morfologia das falhas

O conjunto de dados do *Gravity Spy* é composto por 22 diferentes classes de falhas detectadas durante a primeira (de setembro a dezembro de 2015) e segunda (de novembro de 2016 a agosto de 2017) rodadas de observação do LIGO.

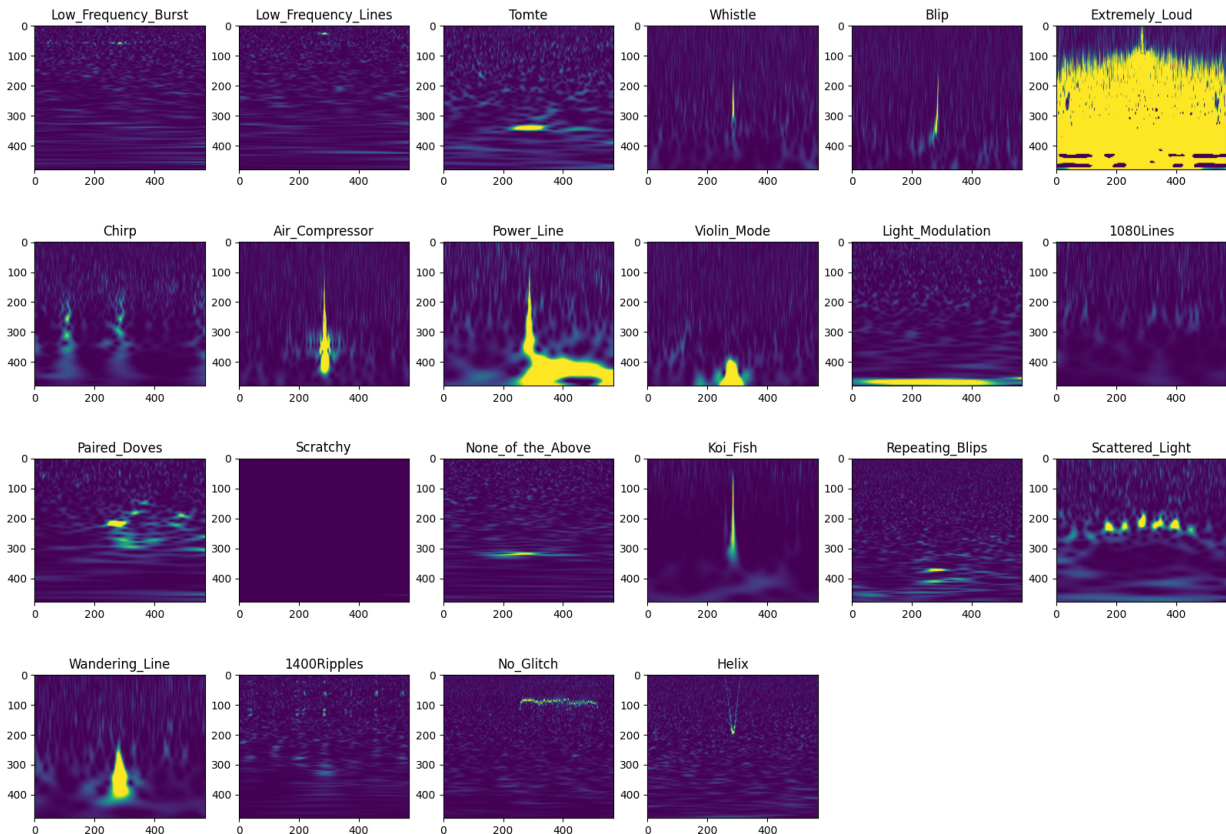


Figura 11 – 22 classes das falhas do LIGO

Apesar de os interferômetros serem projetados para evitar medições de eventos que não sejam provenientes de ondas gravitacionais, ainda assim são feitas medições que não condizem com ondas gravitacionais. Nestes casos, os fenômenos detectados geram as representações de frequência  $\times$  tempo das falhas que constituem a base de dados.

A base de dados utilizada é mais atualizada que a base utilizada em BAHADINI *et al.* (2018) e possui 31.941 amostras de falhas, sendo particionada com 22.407 amostras no conjunto de treino, 4.800 no de validação e 4.734 no de teste. Na Fig. 11, há exemplos de cada uma das 22 classes de falhas presentes nos conjuntos de dados.

Na Tab. 1 pode ser verificada a relação entre a quantidade de exemplos de cada classe em cada um dos conjuntos particionados. Nela, pode ser inferido que as classes do conjunto de dados não são balanceadas; a classe *Blip* é a que possui mais elementos,

totalizando 5.096 imagens deste tipo no conjunto de treinamento, enquanto que a classe *Paired Doves*, que possui 76 elementos, é a classe que possui menos elementos no conjunto de treinamento.

Classes	Conjuntos		
	Treinamento	Teste	Validação
<i>1080 Lines</i>	916	200	196
<i>1400 Ripples</i>	236	36	52
<i>Air Compressor</i>	164	36	32
<i>Blip</i>	5.096	1.092	1.100
<i>Chirp</i>	164	40	36
<i>Extremely Loud</i>	1.264	256	268
<i>Helix</i>	780	168	168
<i>Koi Fish</i>	1.992	408	424
<i>Light Modulation</i>	1.444	312	292
<i>Low Frequency Burst</i>	1.748	360	376
<i>Low Frequency Lines</i>	1.260	264	264
<i>No Glitch</i>	428	84	88
<i>None of the Above</i>	228	44	52
<i>Paired Doves</i>	76	16	16
<i>Power Line</i>	1.256	272	268
<i>Repeating Blips</i>	740	148	164
<i>Scattered Light</i>	1.232	268	272
<i>Scratchy</i>	948	200	200
<i>Tomte</i>	292	52	68
<i>Violin Mode</i>	1.136	256	256
<i>Wandering Line</i>	116	28	24
<i>Whistle</i>	832	180	184

Tabela 1 – Quantidade de amostras das classes nos conjuntos de treinamento, teste e validação

As 22 classes de falhas e suas características podem ser vistas a seguir:

### 1. *1080 Lines*

Observadas durante a segunda rodada de observação do detector LIGO de Hanford, consistem em um fluxo constante de falhas em torno de 1080 Hz. Nas imagens do *Omega Scan*, essas falhas se apresentam como uma sequência de pontos amarelos, às vezes conectados formando uma linha e em outras mais escassas. Interessante resaltar que após uma configuração feita no detector estas falhas foram reduzidas.

### 2. *1400 Ripples*

Essas são falhas de curta duração identificadas em torno de 1400 Hz. Elas aparecem tanto isoladas, isto é, uma única falha numa imagem, como podem aparecer às vezes em uma imagem de 4 segundos várias falhas ao mesmo tempo.

### 3. *Air Compressor*

Essas são falhas de curta duração que parecem ser uma linha grossa centrada na frequência de 50 Hz. No LIGO Hanford, descobriu-se que elas estavam relacionadas a motores de compressores de ar que ligavam e desligavam nas estações finais, esse problema foi resolvido com a substituição dos isoladores de vibração (pés de borracha) nos compressores de ar.

#### 4. *Blip*

São falhas curtas que geralmente têm uma duração de cerca de 40 ms, frequências entre 30 e 500 Hz e aparecem frequentemente no formato de gota estreita, vertical e simétrica no domínio da frequência do tempo.

#### 5. *Chirp*

Essas falhas possuem a forma característica de frequência de tempo criada por ondas gravitacionais de objetos compactos, que crescem em frequência ao longo do tempo. No conjunto de dados do *Gravity Spy*, essas falhas são sinais simulados de ondas gravitacionais adicionados fisicamente aos detectores para fins de teste e calibração.

#### 6. *Extremely Loud*

Esta é uma categoria abrangente para falhas que resultam de uma grande perturbação nos detectores. Elas geralmente abrangem grande parte do espectrograma e têm energias extremamente altas, incluindo também as falhas de alta energia de outras categorias que saturam na maior parte da imagem.

#### 7. *Helix*

Esta categoria contém falhas que se assemelham a um vórtice, que ocorrem em frequências intermediárias e geralmente aparecem agrupadas.

#### 8. *Koi Fish*

Estas falhas são semelhantes às falhas do tipo *Blip*, mas se assemelham a um peixe com a cabeça na extremidade de baixa frequência do gráfico, barbatanas peitorais em torno de 30 Hz e uma cauda fina em torno de 500 Hz.

#### 9. *Light Modulation*

A morfologia dessa falha nem sempre é a mesma, mas geralmente se parece com vários picos brilhantes próximos uns aos outros, possuindo normalmente ruído de baixa frequência ao mesmo tempo. Elas são causadas por flutuações de amplitude no sinal usadas para controlar o comprimento e o alinhamento de algumas das cavidades ópticas do LIGO.

#### 10. *Low Frequency Burst*

São falhas que abrangem ruídos altos, de curta duração e de baixa frequências. Embora várias morfologias constituam essa categoria, elas geralmente se assemelham a pequenas saliências com uma forma quase triangular que cresce de baixa frequência a um

pico e depois diminui em um ou dois segundos.

#### 11. *Low Frequency Line*

Estas falhas aparecem como linhas horizontais em baixas frequências, elas são distintas das do tipo *Low frequency bursts* dado que são mais duradouras e surgem da luz espalhada porque não variam visivelmente em frequência durante longos períodos.

#### 12. *No Glitch*

São falhas que descrevem espectrogramas sem uma estrutura de ruído transiente aparente visível no *Omega Scan*. A falta de visibilidade pode ser devido a vários fatores, incluindo problemas na exibição para falhas cuja frequência de pico é alta (2 kHz) e a duração é curta.

#### 13. *Paired Doves*

É constituída por uma coleção de falhas repetidas que se parecem com as falhas do tipo *Chirps* em baixas frequências e que se alternam entre aumentar e diminuir a frequência.

#### 14. *Power Line*

São falhas que geralmente parecem estreitas em frequência e duram cerca de 0,2 a 0,5 segundos no tempo, centralizadas em torno de 60 Hz ou um de seus harmônicos. Elas são devidas a falhas na rede elétrica dos Estados Unidos (corrente alternada na frequência de 60 Hz).

#### 15. *Repeating Blips*

Esta categoria são falhas do tipo *Blip* que se repetem nas imagens do *Gravity Spy*. Embora esta categoria englobe todas as cadências de repetição, muitas vezes elas se repetem a cada 0,25 ou 0,5 segundos.

#### 16. *Scattered Light*

Essas falhas vêm em muitas morfologias diferentes, embora muitas vezes sejam falhas de baixa frequência, longa duração e irregulares que se parecem com uma ou várias linhas curvas empilhadas umas sobre as outras.

#### 17. *Scratchy*

São uma série de falhas repetidas de curta duração (geralmente com 10-30 falhas por segundo) com frequências intermediárias, muitas vezes durando vários segundos. Elas ocorrem principalmente no LIGO de Hanford e algumas delas estão relacionados ao espalhamento de luz de um defletor usado para interceptar luz laser dispersa. Elas são de difícil reconhecimento, o que dificulta a aplicação de métodos de classificação através de aprendizado de máquina devido as suas morfologias serem espalhadas por uma ampla gama de espaço de características.

### 18. *Tomte*

Essas falhas são semelhantes ou possivelmente uma subclasse do tipo *Blips*. São 10 falhas de baixa frequência que geralmente são de forma triangular e se parecem com o chapéu usado por um gnomo de jardim.

### 19. *Violin Mode Harmonic*

Esta categoria aparece de forma breve se assemelhando a um ponto, e ocorre em 500 Hz e múltiplos harmônicos desta frequência. Como os espelhos principais do LIGO são suspensos por finas fibras de vidro que possuem ressonâncias como a de uma corda de violino, estas falhas ocorrem nas frequências dessas ressonâncias.

### 20. *Wandering Line*

Estas falhas são linhas que duram da ordem de minutos à uma hora e variam em frequência. Alguns exemplos de causas incluem motores (como bombas de vácuo) que não possuem frequências fixas e batidas entre sinais de frequência mais alta que apresentam alguma variação de frequência.

### 21. *Whistle*

Essas falhas possuem forma de W ou V em frequências mais baixas. Elas são causadas por sinais de radiofrequência, que são sinais em frequências de MHz, que interferem com os osciladores controlados por tensão do LIGO.

### 22. *None of the Above*

É um conjunto de falhas que não se enquadram nas outras 21 categorias possuindo muita variabilidade em sua morfologia.

## 3.3 Classificação usando CNN

A proposta de método a ser seguida é utilizar CNNs que sejam treinadas com esta base de dados e que sejam capazes de extrair as características discriminantes das imagens para classificar de forma correta as distintas classes das falhas. Tendo como base para o modelo I de CNN a arquitetura *merged-view*, presente em BAHADINI *et al.* (2018) (Fig. 9).

Nos treinamentos das redes a função de ativação usada em todas as camadas convolucionais foi a função ReLU, o número de épocas foi definido como 10 e o tamanho do lote de treinamento como 60.

Como forma de comparação, foram feitas alterações nos hiperparâmetros, nos tamanhos dos filtros e no número de camadas para verificar quais os resultados apresentam melhores métricas, deste modo, foram definidas oito diferentes arquiteturas de redes convolucionais para serem estudadas (Tab. 2).

Modelo I	Modelo II	Modelo III	Modelo IV
5x5 <i>Conv. layer</i> (16) 2x2 <i>Max Pooling</i> <i>Dropout</i>	5x5 <i>Conv. layer</i> (16) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (16) 2x2 <i>Max Pooling</i> <i>Dropout</i>	3x3 <i>Conv. layer</i> (16) 2x2 <i>Max Pooling</i>
5x5 <i>Conv. layer</i> (32) 2x2 <i>Max Pooling</i> <i>Dropout</i>	5x5 <i>Conv. layer</i> (32) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (32) 2x2 <i>Max Pooling</i> <i>Dropout</i>	3x3 <i>Conv. layer</i> (32) 2x2 <i>Max Pooling</i>
5x5 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> <i>Dropout</i>	5x5 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> <i>Dropout</i>	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i>
5x5 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> <i>Dropout</i> <i>Flatten</i>	5x5 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> <i>Flatten</i>	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> <i>Dropout</i> <i>Flatten</i>	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> <i>Flatten</i>
5x5 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> <i>Dropout</i> <i>Flatten</i> <i>Fully Connected</i> (256) <i>Softmax</i> (22)	<i>Fully Connected</i> (256) <i>Softmax</i> (22)	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> <i>Dropout</i> <i>Flatten</i> <i>Fully Connected</i> (256) <i>Softmax</i> (22)	<i>Fully Connected</i> (256) <i>Softmax</i> (22)
Modelo V	Modelo VI	Modelo VII	Modelo VIII
5x5 <i>Conv. layer</i> (16) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (16) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (16) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (16) 2x2 <i>Max Pooling</i>
5x5 <i>Conv. layer</i> (32) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (32) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (32) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (32) 2x2 <i>Max Pooling</i>
5x5 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i>	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i>
5x5 <i>Conv. layer</i> (128) 2x2 <i>Max Pooling</i> <i>Flatten</i>	3x3 <i>Conv. layer</i> (128) 2x2 <i>Max Pooling</i> <i>Flatten</i>	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> 3x3 <i>Conv. layer</i> (128)	3x3 <i>Conv. layer</i> (64) 2x2 <i>Max Pooling</i> 3x3 <i>Conv. layer</i> (128)
<i>Fully Connected</i> (256) <i>Softmax</i> (22)	<i>Fully Connected</i> (256) <i>Softmax</i> (22)	2x2 <i>Max Pooling</i> 3x3 <i>Conv. layer</i> (128) 2x2 <i>Max Pooling</i> <i>Flatten</i> <i>Fully Connected</i> (256) <i>Softmax</i> (22)	2x2 <i>Max Pooling</i> 3x3 <i>Conv. layer</i> (256) 2x2 <i>Max Pooling</i> <i>Flatten</i> <i>Fully Connected</i> (256) <i>Softmax</i> (22)

Tabela 2 – Arquiteturas dos modelos de redes convolucionais



## 4 ANÁLISE DE RESULTADOS

### 4.1 Critérios de avaliação do método proposto

Como critério de avaliação do método proposto foram levadas em contas algumas métricas. Primeiramente, foi utilizada a matriz de confusão na qual as linhas representam as classes verdadeiras e as colunas as classes preditas pelo classificador; assim, cada valor na matriz corresponde à quantidade de exemplos preditos em uma determinada classe associado à sua classe correta. Uma matriz de confusão deste tipo é representada na Tab. 3 em que VP são os verdadeiros positivos, FN são os falsos negativos, FP são os falsos positivos e VN são os verdadeiros negativos.

Verdadeiro	Predito	
	A	B
A	VP	FN
B	FP	VN

Tabela 3 – Matriz de Confusão

A partir da matriz de confusão deste tipo é possível obter as seguintes medidas de avaliação:

- Taxa de falso positivo (TFP)

$$TFP = \frac{FP}{FP + VN} \quad (4.1)$$

- Taxa de falso negativo (TFN)

$$TFN = \frac{FN}{VP + FN} \quad (4.2)$$

- Taxa de verdadeiro positivo (TTP), conhecida como sensibilidade ou *recall*.

$$TTP = \frac{VP}{VP + FN} \quad (4.3)$$

Mede a porcentagem de exemplos positivos classificados como positivos, assim nenhum exemplo positivo é deixado de fora. É usada para a análise da ocorrência de falsos negativos, sendo que quanto maior a ocorrência de falsos negativos, menor será a revocação.

- Taxa de verdadeiro negativo (TTN), conhecida como especificidade.

$$TNN = \frac{VN}{VN + FP} \quad (4.4)$$

Mede a porcentagem de exemplos negativos classificados como negativos, nela nenhum exemplo negativo é deixado de fora.

- Valor preditivo positivo (VPP), conhecida como precisão.

$$VPP = \frac{VP}{VP + FP} \quad (4.5)$$

Mede a porcentagem de exemplos classificados como positivos que são realmente positivo, desta forma nenhum exemplo negativo é incluído. É uma medida ideal para a análise da ocorrência de falsos positivos, assim quanto maior a ocorrência de falsos positivos, menor será a precisão.

- Valor preditivo negativo (VNP)

$$VNP = \frac{VN}{VN + FN} \quad (4.6)$$

- Acurácia (AC)

$$AC = \frac{VP + VN}{VP + VN + FP + FN} \quad (4.7)$$

- *F1-Score (F1)*

$$F1 = \frac{2}{\frac{1}{VPP} + \frac{1}{TTP}} \quad (4.8)$$

Ela é a média harmônica ponderada da precisão e da revocação.

- *Macro F1-Score (MF1)*

$$MF1 = \frac{F1_1 + F1_2 + \dots + F1_N}{N} \quad (4.9)$$

Para o caso de N classes, com  $F1_N$  sendo a F1 da N-ésima classe.

Com estas métricas é possível avaliar os resultados obtidos pela rede ao usar o conjunto de testes, verificando a validade dos modelos.

## 4.2 Resultados dos Modelos

A seguir, são apresentados os resultados dos treinamentos dos diferentes modelos de redes convolucionais.

### 4.3 Modelo de Rede Convolutacional I

No primeiro modelo, o formato das saídas de cada camada e a quantidade de parâmetros treinados por camada podem ser verificados na Tab. 4.

<i>Layer (type)</i>	<i>Output Shape</i>	<i>Param #</i>
5x5 <i>Convolutional2D</i>	(None, 276, 336, 16)	1216
2x2 <i>MaxPooling2D</i>	(None, 138, 168, 16)	0
Dropout	(None, 138, 168, 16)	0
5x5 <i>Convolutional2D</i>	(None, 134, 164, 32)	12832
2x2 <i>MaxPooling2D</i>	(None, 67, 82, 32)	0
Dropout	(None, 67, 82, 32)	0
5x5 <i>Convolutional2D</i>	(None, 63, 78, 64)	51264
2x2 <i>MaxPooling2D</i>	(None, 31, 39, 64)	0
Dropout	(None, 31, 39, 64)	0
5x5 <i>Convolutional2D</i>	(None, 27, 35, 64)	102464
2x2 <i>MaxPooling2D</i>	(None, 13, 17, 64)	0
Dropout	(None, 13, 17, 64)	0
<i>Flatten</i>	(None, 14144)	0
<i>Dense</i>	(None, 256)	3621120
<i>Dense</i>	(None, 22)	5654
<hr/>		
<i>Total params: 3.794.550 (14.48 MB)</i>		
<i>Trainable params: 3.794.550 (14.48 MB)</i>		
<i>Non-trainable params: 0 (0.00 Byte)</i>		

Tabela 4 – Formato da saída do modelo de rede I

A acurácia do treinamento foi de 22,80% e do conjunto de validação foi de 22,92%, como ilustrado na Fig. 12 (a). A função de perda, de acordo com os conjuntos de treino e de validação, podem ser verificados na Fig. 12 (b). Ela começa com 0,90 na primeira época do treinamento, chegando a 2,6 na última época, enquanto que no conjunto de validação varia em torno de 2,7.

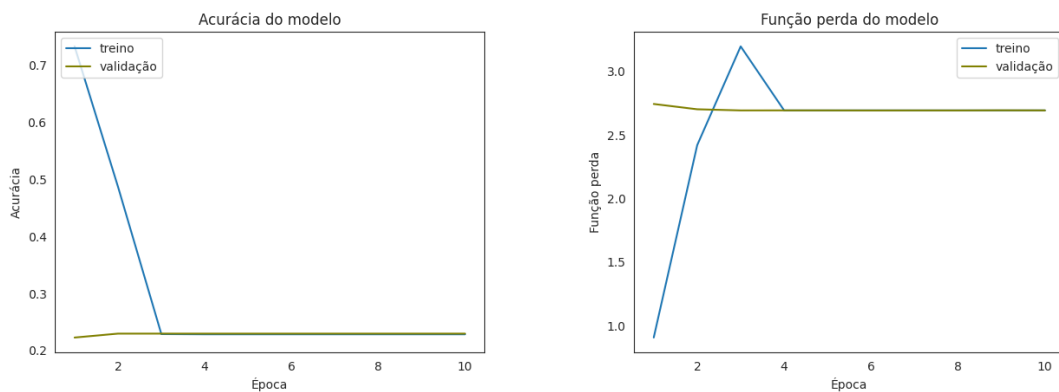


Figura 12 – (a) Acurácia do modelo I (b) Função de perda do modelo I

O modelo treinado foi utilizado no conjunto de teste para previsão e classificação

das falhas, sendo gerado a partir dos valores obtidos a matriz de confusão (Fig. 13) dos tipos de falhas das ondas gravitacionais.

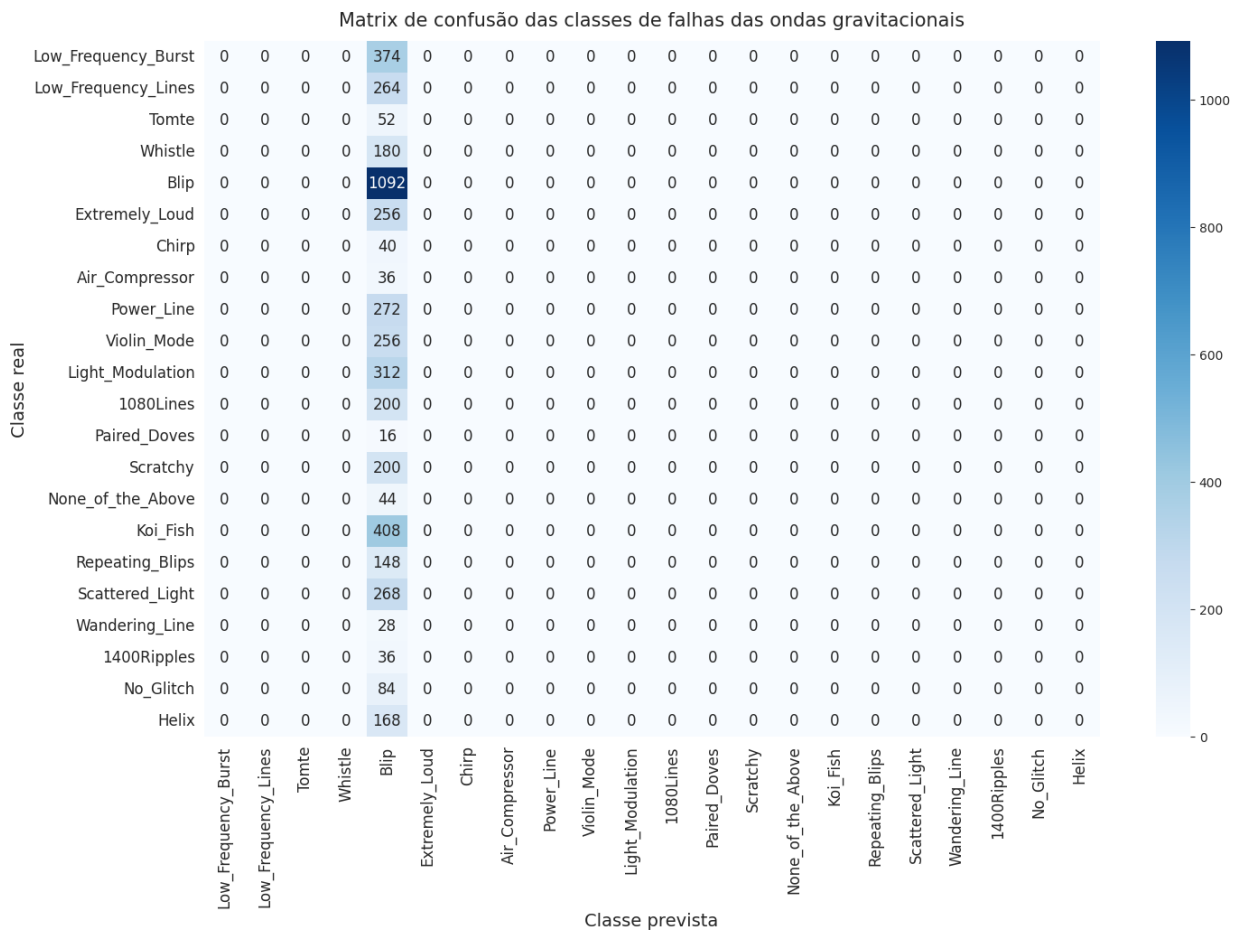


Figura 13 – Matriz de confusão dos tipos de falhas de ondas gravitacionais do modelo I

Com a matriz de confusão, obteve-se uma acurácia para o conjunto de teste de 23,07%, a precisão foi de 0,9%, o *recall* de 23,07% e a *MF1-score* do mesmo foi de 1,7%. A partir da matriz de confusão, também foi possível calcular as métricas como acurácia, precisão, *recall* e *f1-score* para cada uma das classes de falhas como consta na Tab. 5, verificando que o modelo está classificando com acurácia de 100% a classe *Blip* e 0% os outros modelos, indicando que o modelo não conseguiu identificar e generalizar as outras classes além da *Blip*.

De fato, ao verificar a matriz de confusão, é possível conferir que o modelo conseguiu identificar as 1.092 amostras do conjunto de teste da classe *Blip* de forma correta. No entanto, os elementos das demais classes foram todos preditos como sendo pertencentes também a classe *Blip*, demonstrando que durante o treinamento ocorreu *overfitting* no aprendizado desta classe. Isso provavelmente ocorreu devido ao desbalanceamento das classes, o que impossibilitou que o modelo conseguisse aprender os padrões das demais classes.

Classes de Falhas	Acurácia (%)	Precisão (%)	Recall (%)	F1-score (%)
<i>Low Frequency Burst</i>	0,00	0,00	0,00	0,00
<i>Low Frequency Lines</i>	0,00	0,00	0,00	0,00
<i>Tomte</i>	0,00	0,00	0,00	0,00
<i>Whistle</i>	0,00	0,00	0,00	0,00
<i>Blip</i>	100,00	23,07	100,00	37,49
<i>Extremely Loud</i>	0,00	0,00	0,00	0,00
<i>Chirp</i>	0,00	0,00	0,00	0,00
<i>Air Compressor</i>	0,00	0,00	0,00	0,00
<i>Power Line</i>	0,00	0,00	0,00	0,00
<i>Violin Mode</i>	0,00	0,00	0,00	0,00
<i>Light Modulation</i>	0,00	0,00	0,00	0,00
<i>1080 Lines</i>	0,00	0,00	0,00	0,00
<i>Paired Doves</i>	0,00	0,00	0,00	0,00
<i>Scratchy</i>	0,00	0,00	0,00	0,00
<i>None of the Above</i>	0,00	0,00	0,00	0,00
<i>Koi Fish</i>	0,00	0,00	0,00	0,00
<i>Repeating Blips</i>	0,00	0,00	0,00	0,00
<i>Scattered Light</i>	0,00	0,00	0,00	0,00
<i>Wandering Line</i>	0,00	0,00	0,00	0,00
<i>1400 Ripples</i>	0,00	0,00	0,00	0,00
<i>No Glitch</i>	0,00	0,00	0,00	0,00
<i>Helix</i>	0,00	0,00	0,00	0,00

Tabela 5 – Métricas das classes do modelo de rede I

#### 4.4 Modelo de Rede Convolutacional II

No segundo modelo, o formato das saídas de cada camada e a quantidade de parâmetros treinados por camada podem ser verificados na Tab. 6.

<i>Layer (type)</i>	<i>Output Shape</i>	<i>Param #</i>
5x5 <i>Convolutional2D</i>	(None, 276, 336, 16)	1216
2x2 <i>MaxPooling2D</i>	(None, 138, 168, 16)	0
5x5 <i>Convolutional2D</i>	(None, 134, 164, 32)	12832
2x2 <i>MaxPooling2D</i>	(None, 67, 82, 32)	0
5x5 <i>Convolutional2D</i>	(None, 63, 78, 64)	51264
2x2 <i>MaxPooling2D</i>	(None, 31, 39, 64)	0
5x5 <i>Convolutional2D</i>	(None, 27, 35, 64)	102464
2x2 <i>MaxPooling2D</i>	(None, 13, 17, 64)	0
<i>Flatten</i>	(None, 14144)	0
<i>Dense</i>	(None, 256)	3621120
<i>Dense</i>	(None, 22)	5654
<i>Total params: 3.794.550 (14.48 MB)</i>		
<i>Trainable params: 3.794.550 (14.48 MB)</i>		
<i>Non-trainable params: 0 (0.00 Byte)</i>		

Tabela 6 – Formato do modelo de rede II

A acurácia do treinamento foi de 91,82% e do conjunto de validação foi de 83,79%, como ilustrado na Fig. 14 (a). A função de perda, de acordo com os conjuntos de treino e

de validação, podem ser verificados na Fig. 14 (b), onde mostra que ela é alta na primeira época do treinamento, porém acaba caindo à 0,29 na última época. Enquanto que no conjunto de validação, ela varia de 0,54 à 0,91.

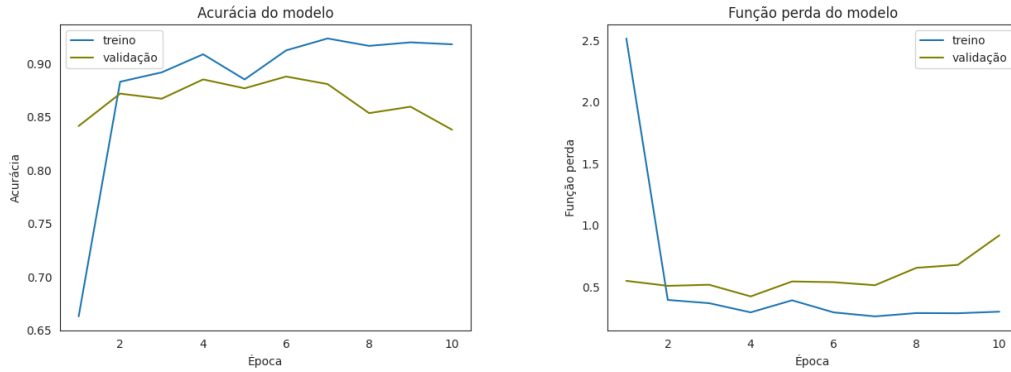


Figura 14 – (a) Acurácia do modelo II (b) Função de perda do modelo II

O modelo treinado foi utilizado no conjunto de teste para previsão e classificação das falhas, sendo gerado a partir dos valores obtidos a matriz de confusão (Fig. 15) dos tipos de falhas das ondas gravitacionais.

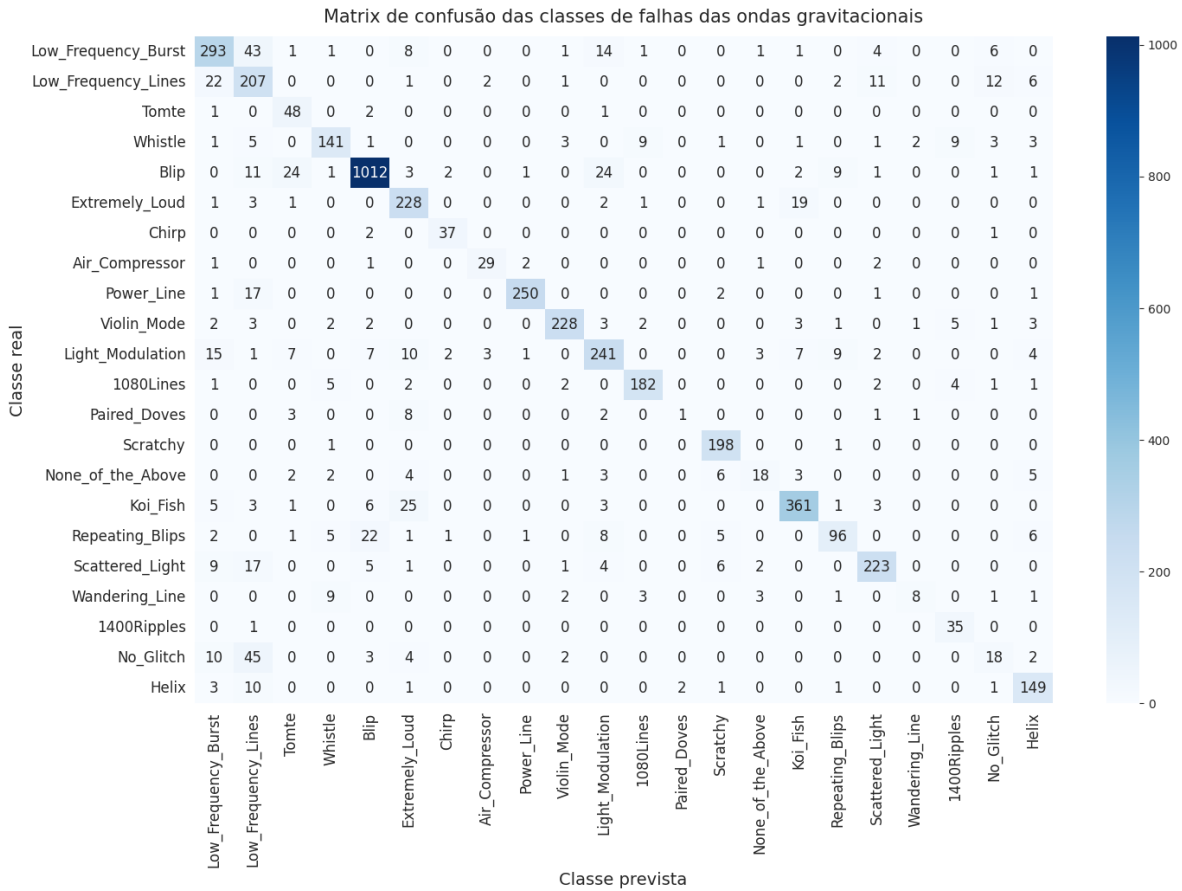


Figura 15 – Matriz de confusão dos tipos de falhas de ondas gravitacionais do modelo II

Com a matriz de confusão, obteve-se uma acurácia para o conjunto de teste de 84,56%, a precisão foi de 66,19%, o *recall* foi de 84,54% e a *MF1-score* do mesmo foi de 74,25% na última época do treinamento. A partir da matriz de confusão, também foi possível calcular as métricas como acurácia, precisão, *recall* e *f1-score* para cada uma das classes de falhas ilustradas na Tab. 7.

Classes de Falhas	Acurácia (%)	Precisão (%)	<i>Recall</i> (%)	<i>F1-score</i> (%)
<i>Low Frequency Burst</i>	81,39	79,84	78,34	79,08
<i>Low Frequency Lines</i>	78,41	56,56	78,41	65,71
<i>Tomte</i>	92,31	54,55	92,31	68,57
<i>Whistle</i>	78,33	84,43	78,33	81,27
<i>Blip</i>	92,67	95,20	92,67	93,92
<i>Extremely Loud</i>	89,06	77,03	89,06	82,61
<i>Chirp</i>	92,50	88,10	92,50	90,24
<i>Air Compressor</i>	80,56	85,29	80,56	82,86
<i>Power Line</i>	91,91	98,04	91,91	94,88
<i>Violin Mode</i>	89,06	94,61	89,06	91,75
<i>Light Modulation</i>	77,24	79,02	77,24	78,12
<i>1080 Lines</i>	91,00	91,92	91,00	91,46
<i>Paired Doves</i>	6,25	33,33	6,25	10,53
<i>Scratchy</i>	99,00	90,41	99,00	94,51
<i>None of the Above</i>	40,91	62,07	40,91	49,32
<i>Koi Fish</i>	88,48	90,93	88,48	89,69
<i>Repeating Blips</i>	64,86	79,34	64,86	71,38
<i>Scattered Light</i>	83,21	88,84	83,21	85,93
<i>Wandering Line</i>	28,57	66,67	28,57	40,00
<i>1400 Ripples</i>	97,22	66,04	97,22	78,65
<i>No Glitch</i>	21,43	40,00	21,43	27,91
<i>Helix</i>	88,69	81,87	88,69	85,14

Tabela 7 – Métricas das classes do modelo de rede II

As classes, como a classe *Blip*, que possuem maior número de exemplos no conjunto de treinamento, apresentam valores de acurácia e de precisão alto. A classe *Blip* possui uma acurácia de 92,67% e precisão de 95,20%, ou a classe *Power Line* que possui 1.256 amostras de treinamento e apresenta uma acurácia de 91,91% e uma precisão de 98,04%.

No entanto, classes como a *Scratchy*, que possui 948 elementos no conjunto de treinamento e 200 elementos no conjunto de testes, apresentou uma acurácia de 99% e uma precisão de 90,41%. A classe *Chirp*, que possui 164 elementos no conjunto de treinamento e 40 no conjunto de testes, apresentou uma acurácia de 92,5% e uma precisão de 88,10%.

O modelo apresenta métricas satisfatórias na maioria das classes, apresentando dificuldades em diferenciar as classes *Paired Doves*, *None of the Above*, *No Glitch*, *Repeating Blips* e *Wandering Line*.

## 4.5 Modelo de Rede Convolutacional III

No terceiro modelo, o formato das saídas de cada camada e a quantidade de parâmetros treinados por camada podem ser verificados na Tab. 8.

<i>Layer (type)</i>	<i>Output Shape</i>	<i>Param #</i>
3x3 <i>Convolutional</i>	(None, 278, 338, 16)	448
2x2 <i>MaxPooling2D</i>	(None, 139, 169, 16)	0
Dropout	(None, 139, 169, 16)	0
3x3 <i>Convolutional2D</i>	(None, 137, 167, 32)	4640
2x2 <i>MaxPooling2D</i>	(None, 68, 83, 32)	0
Dropout	(None, 68, 83, 32)	0
3x3 <i>Convolutional2D</i>	(None, 66, 81, 64)	18496
2x2 <i>MaxPooling2D</i>	(None, 33, 40, 64)	0
Dropout	(None, 33, 40, 64)	0
3x3 <i>Convolutional2D</i>	(None, 31, 38, 64)	36928
2x2 <i>MaxPooling2D</i>	(None, 15, 19, 64)	0
Dropout	(None, 15, 19, 64)	0
<i>Flatten</i>	(None, 18240)	0
<i>Dense</i>	(None, 256)	4669696
<i>Dense</i>	(None, 22)	5654
<i>Total params: 4.735.862 (18.07 MB)</i>		
<i>Trainable params: 4.735.862 (18.07 MB)</i>		
<i>Non-trainable params: 0 (0.00 Byte)</i>		

Tabela 8 – Formato do modelo de rede III

A acurácia do treinamento foi de 22,80% e do conjunto de validação foi de 22,92%, como ilustrado na Fig. 16 (a). A função de perda, de acordo com os conjuntos de treino e de validação da Fig. 16 (b), que explicita que a função de perda começa na primeira época de treinamento com 0,71 e termina a última época em 2,69. Enquanto que no conjunto de validação, ela oscila ao longo do treinamento e retorna ao valor próximo de 2,6 na última época.

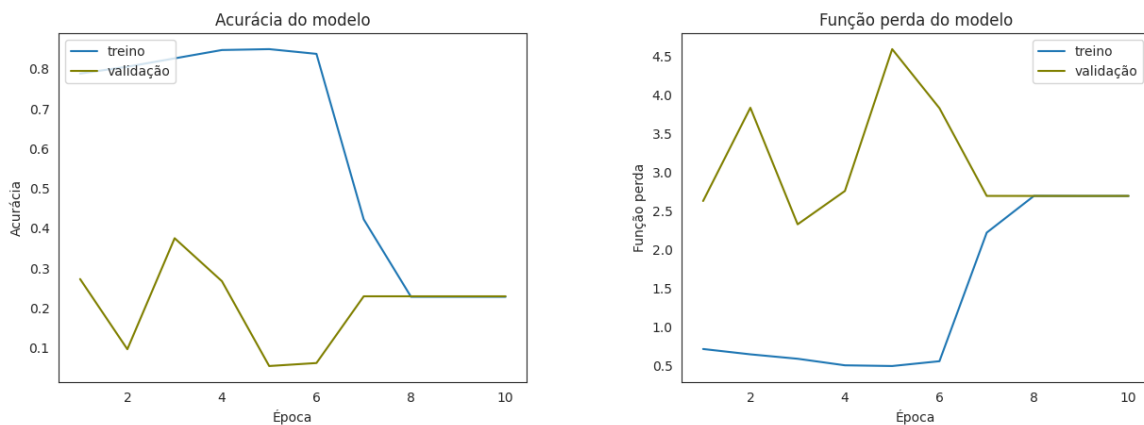


Figura 16 – (a) Acurácia do modelo III

(b) Função de perda do modelo III



<i>Layer (type)</i>	<i>Output Shape</i>	<i>Param #</i>
3x3 <i>Convolutional2D</i>	(None, 278, 338, 16)	448
2x2 <i>MaxPooling2D</i>	(None, 139, 169, 16)	0
3x3 <i>Convolutional2D</i>	(None, 137, 167, 32)	4640
2x2 <i>MaxPooling2D</i>	(None, 68, 83, 32)	0
3x3 <i>Convolutional2D</i>	(None, 66, 81, 64)	18496
2x2 <i>MaxPooling2D</i>	(None, 33, 40, 64)	0
3x3 <i>Convolutional2D</i>	(None, 31, 38, 64)	36928
2x2 <i>MaxPooling2D</i>	(None, 15, 19, 64)	0
<i>Flatten</i>	(None, 18240)	0
<i>Dense</i>	(None, 256)	4669696
<i>Dense</i>	(None, 22)	5654
<i>Total params: 4.735.862 (18.07 MB)</i>		
<i>Trainable params: 4.735.862 (18.07 MB)</i>		
<i>Non-trainable params: 0 (0.00 Byte)</i>		

Tabela 9 – Formato do modelo de rede IV

A acurácia do treinamento foi de 96,69% e do conjunto de validação foi de 88,65%, como ilustrado na Fig. 18 (a). A função de perda, de acordo com os conjuntos de treino e de validação da Fig. 18 (b), mostra que a função de perda diminui com o passar das épocas do treinamento chegando à 0,12 na última época. Enquanto que no conjunto de validação ela varia de 0,34 à 0,68.

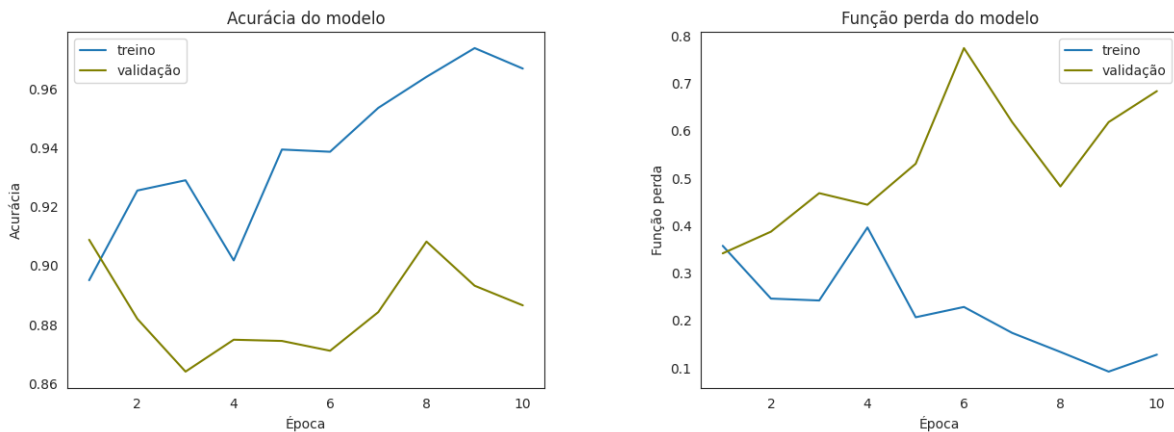


Figura 18 – (a) Acurácia do modelo IV (b) Função de perda do modelo IV

O modelo treinado foi utilizado no conjunto de teste para previsão e classificação das falhas, sendo gerado a partir dos valores obtidos a matriz de confusão (Fig. 19) dos tipos de falhas das ondas gravitacionais. Com a matriz de confusão, obteve-se uma acurácia para o conjunto de teste de 89,12%, a precisão foi de 73,81%, a *recall* foi de 89,12% e a *MF1-score* do mesmo foi de 80,75%. A partir da matriz de confusão também foi possível calcular as métricas como acurácia, precisão, *recall* e *f1-score* para cada uma das classes de falhas.

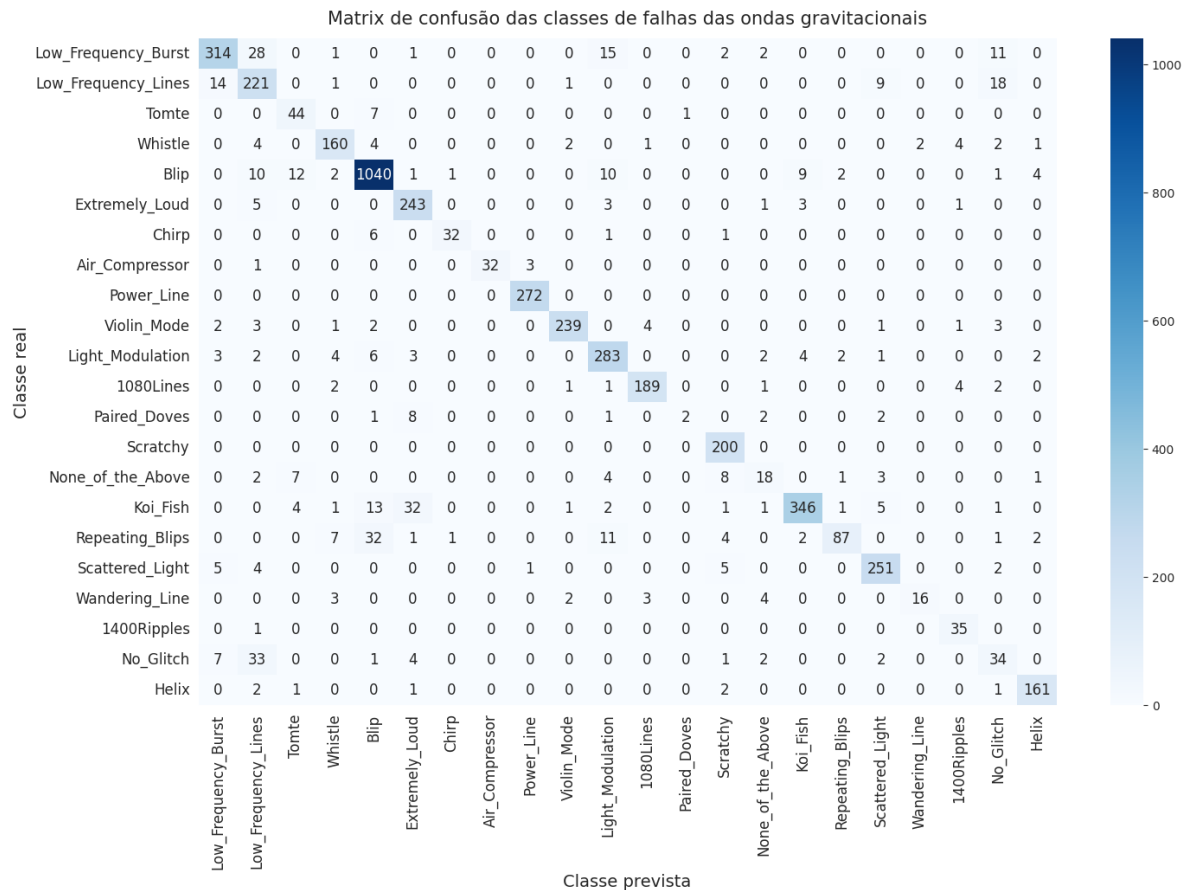


Figura 19 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo IV

Na Tab. 10, é possível verificar que a classe *Blip* que possui 5096 itens no conjunto de treinamento possui acurácia de 95,24% e precisão de 95,53%. A classe *Extremely Loud*, que possui 1.264 amostras de treinamento, apresenta acurácia de 94,94% e precisão de 82,65%. No entanto, classes menores, como a *Helix*, que possui 780 elementos no conjunto de treinamento e 164 elementos no conjunto de testes, apresentou uma acurácia de 95,83% e uma precisão de 94,15%. A classe *Scratchy*, que possui 948 elementos no conjunto de treinamento e 200 elementos no conjunto de testes, apresentou uma acurácia de 100% e uma precisão de 89,29%. A *Air Compressor*, com 164 amostras no conjunto de treinamento e 36 amostras no conjunto de testes, obteve acurácia de 88,89% e precisão de 100%, ou seja, predisse de forma correta todas as amostras do conjunto de teste.

O modelo apresenta métricas satisfatórias na maioria das classes, mas enfrenta dificuldades em diferenciar as classes *Paired Doves*, *None of the Above*, *No Glitch*, *Repeating Blips* e *Wandering Line*. Observa-se que estas duas últimas classes apresentam precisão alta, mas acurácia baixa. Ao verificar a matriz de confusão, nota-se que 32 elementos foram classificados incorretamente como pertencentes à classe *Repeating Blips*, quando na verdade são da classe *Blip*. Além disso, 4 elementos foram classificados incorretamente como pertencente à classe *Wandering Line*, quando deveriam ser classificados como *None*

of the Above, indicando que o modelo não conseguiu distinguir corretamente essas classes.

Classes de Falhas	Acurácia (%)	Precisão (%)	Recall (%)	F1-score (%)
<i>Low Frequency Burst</i>	87,22	91,01	83,96	87,34
<i>Low Frequency Lines</i>	83,71	69,94	83,71	76,21
<i>Tomte</i>	84,62	64,71	84,62	73,33
<i>Whittle</i>	88,89	87,91	88,89	88,40
<i>Blip</i>	95,24	93,53	95,24	94,37
<i>Extremely Loud</i>	94,92	82,65	94,92	88,36
<i>Chirp</i>	80,00	94,12	80,00	86,49
<i>Air Compressor</i>	88,89	100,00	88,89	94,12
<i>Power Line</i>	100,00	98,55	100,00	99,27
<i>Violin Mode</i>	93,36	97,15	93,36	95,22
<i>Light Modulation</i>	90,71	85,50	90,71	88,02
<i>1080 Lines</i>	94,50	95,94	94,50	95,21
<i>Paired Doves</i>	12,50	66,67	12,50	21,05
<i>Scratchy</i>	100,00	89,29	100,00	94,34
<i>None of the Above</i>	40,91	54,55	40,91	46,75
<i>Koi Fish</i>	84,80	95,05	84,80	89,64
<i>Repeating Blips</i>	58,78	93,55	58,78	72,20
<i>Scattered Light</i>	93,66	91,61	93,66	92,62
<i>Wandering Line</i>	57,14	88,89	57,14	69,57
<i>1400 Ripples</i>	97,2	77,78	97,22	86,42
<i>No Glitch</i>	40,48	44,74	40,48	42,50
<i>Helix</i>	95,83	94,15	95,83	94,99

Tabela 10 – Métricas das classes do modelo de rede IV

#### 4.7 Modelo de Rede Convolutacional V

No quinto modelo, o formato das saídas de cada camada e a quantidade de parâmetros treinados por camada podem ser verificados na Tab. 11.

<i>Layer (type)</i>	<i>Output Shape</i>	<i>Param #</i>
5x5 <i>Convolutional2D</i>	(None, 276, 336, 16)	1216
2x2 <i>MaxPooling2D</i>	(None, 138, 168, 16)	0
5x5 <i>Convolutional2D</i>	(None, 134, 164, 32)	12832
2x2 <i>MaxPooling2D</i>	(None, 67, 82, 32)	0
5x5 <i>Convolutional2D</i>	(None, 63, 78, 64)	51264
2x2 <i>MaxPooling2D</i>	(None, 31, 39, 64)	0
5x5 <i>Convolutional2D</i>	(None, 27, 35, 128)	204928
2x2 <i>MaxPooling2D</i>	(None, 13, 17, 128)	0
<i>Flatten</i>	(None, 28288)	0
<i>Dense</i>	(None, 256)	7241984
<i>Dense</i>	(None, 22)	5654
<i>Total params: 7.517.878 (28.68 MB)</i>		
<i>Trainable params: 7.517.878 (28.68 MB)</i>		
<i>Non-trainable params: 0 (0.00 Byte)</i>		

Tabela 11 – Formato da saída do modelo de rede V

A acurácia do treinamento foi de 94,73% e do conjunto de validação foi de 85,54%, conforme Fig. 20 (a). A função de perda, de acordo com os conjuntos de treino e de validação da Fig. 20 (b), que mostra que a função de perda diminui chegando à 0,18 na última época. Enquanto que no conjunto de validação ela varia em torno de 0,79 à 0,75.

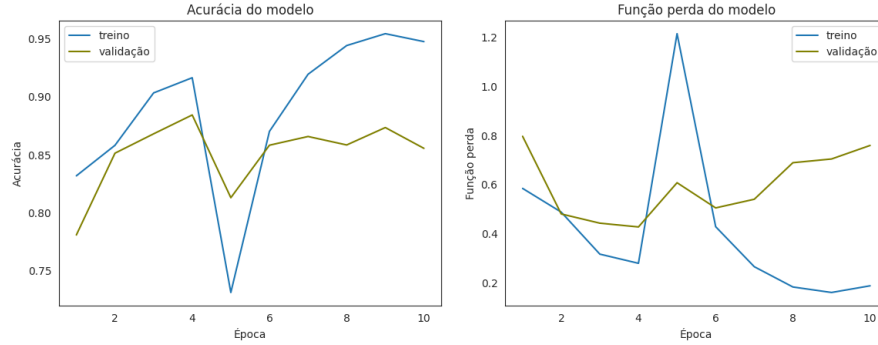


Figura 20 – (a) Acurácia do modelo V (b) Função de perda do modelo V

O modelo treinado foi utilizado no conjunto de teste para previsão e classificação das falhas, sendo gerado a partir dos valores obtidos a matriz de confusão (Fig. 21) dos tipos de falhas das ondas gravitacionais.

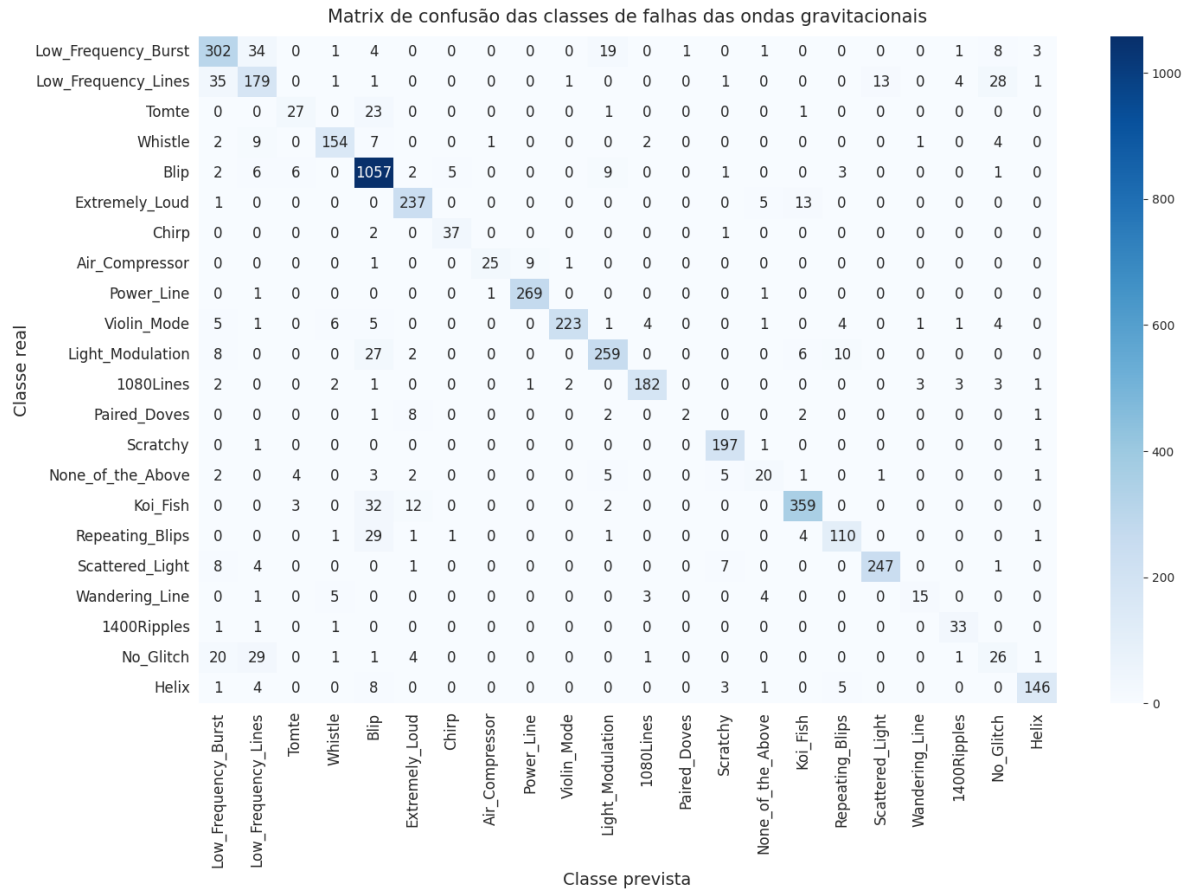


Figura 21 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo V

Com esta matriz de confusão, obteve-se que a acurácia do conjunto de testes foi de 86,73%, a precisão foi de 68,47%, o *recall* foi de 89,81% e a *MF1-score* do mesmo foi de 77,70%. A partir da matriz de confusão, também foi possível calcular as métricas como acurácia, precisão, *recall* e *f1-score* para cada uma das classes de falhas como pode ser verificado na Tab. 12.

Classes de Falhas	Acurácia (%)	Precisão (%)	<i>Recall</i> (%)	<i>F1-score</i> (%)
<i>Low Frequency Burst</i>	83,89	77,63	80,75	79,16
<i>Low Frequency Lines</i>	67,80	66,30	67,80	67,04
<i>Tomte</i>	51,92	67,50	51,92	58,70
<i>Whistle</i>	85,56	89,53	85,56	87,50
<i>Blip</i>	96,79	87,94	96,79	92,15
<i>Extremely Loud</i>	92,58	88,10	92,58	90,29
<i>Chirp</i>	92,50	86,05	92,50	89,16
<i>Air Compressor</i>	69,44	92,59	69,44	79,37
<i>Power Line</i>	98,90	96,42	98,90	97,64
<i>Violin Mode</i>	87,11	98,24	87,11	92,34
<i>Light Modulation</i>	83,01	86,62	83,01	84,78
<i>1080 Lines</i>	91,00	94,79	91,00	92,86
<i>Paired Doves</i>	12,50	66,67	12,50	21,05
<i>Scratchy</i>	98,50	91,63	98,50	94,94
<i>None of the Above</i>	45,45	58,82	45,45	51,28
<i>Koi Fish</i>	87,99	93,01	87,99	90,43
<i>Repeating Blips</i>	74,32	83,33	74,32	78,57
<i>Scattered Light</i>	92,16	94,64	92,16	93,38
<i>Wandering Line</i>	53,57	75,00	53,57	62,50
<i>1400 Ripples</i>	91,67	76,74	91,67	83,54
<i>No Glitch</i>	30,95	34,67	30,95	32,70
<i>Helix</i>	86,90	93,59	86,90	90,12

Tabela 12 – Métricas das classes do modelo de rede V

Nota-se que a classe *Blip*, que possui maior número de amostras no conjunto de treinamento, apresenta acurácia de 96,79% e precisão de 87,94% e a classe *Scattered Light*, que possui 1.232 amostras de treinamento, possui acurácia de 92,16% e precisão de 94,64%.

No entanto, classes menores como a *1080 Lines*, que possui 916 elementos no conjunto de treinamento e 200 elementos no conjunto de testes apresentou uma acurácia de 91,00% e uma precisão de 94,79% e a *Chirp*, que possui 164 amostras no conjunto de treinamento e 40 amostras no conjunto de testes, alcançou acurácia de 92,50% e precisão de 86,05%.

De modo que o modelo apresenta métricas satisfatórias na maioria das classes, apresentando dificuldades em diferenciar as classes *Low Frequency Lines*, *Paired Doves*, *Air Compressor*, *None of the Above*, *No Glitch*, *Tomte* e *Wandering Line*. Nota-se que a classe *Air Compressor* apresenta precisão alta, mas acurácia baixa. Ao verificar a matriz de confusão, observa-se que 9 elementos classificados incorretamente à ela, são pertencentes à

classe *Power Line*, como ambas estão centradas próximas da faixa de 50Hz-60Hz, pode ter havido a confusão devido à semelhança entre as classes..

#### 4.8 Modelo de Rede Convolutacional VI

No sexto modelo, o formato das saídas de cada camada e a quantidade de parâmetros treinados por camada podem ser verificados na Tab. 13.

Layer (type)	Output Shape	Param #
3x3 <i>Convolutional2D</i>	(None, 278, 338, 16)	448
2x2 <i>MaxPooling2D</i>	(None, 139, 169, 16)	0
3x3 <i>Convolutional2D</i>	(None, 137, 167, 32)	4640
2x2 <i>MaxPooling2D</i>	(None, 68, 83, 32)	0
3x3 <i>Convolutional2D</i>	(None, 66, 81, 64)	18496
2x2 <i>MaxPooling2D</i>	(None, 33, 40, 64)	0
3x3 <i>Convolutional2D</i>	(None, 31, 38, 128)	73856
2x2 <i>MaxPooling2D</i>	(None, 15, 19, 128)	0
<i>Flatten</i>	(None, 36480)	0
<i>Dense</i>	(None, 256)	9339136
<i>Dense</i>	(None, 22)	5654
<i>Total params: 9442230 (36.02 MB)</i>		
<i>Trainable params: 9442230 (36.02 MB)</i>		
<i>Non-trainable params: 0 (0.00 Byte)</i>		

Tabela 13 – Formato do modelo de rede VI

A acurácia do treinamento foi de 96,44% e do conjunto de validação foi de 89,08%, conforme Fig. 22 (a). A função de perda, de acordo com os conjuntos de treino e de validação, ilustrada na Fig. 22 (b), mostra que seu valor é de 0,14 na última época no conjunto de treinamento. Enquanto que no conjunto de validação ela varia de 0,40 à 0,78.

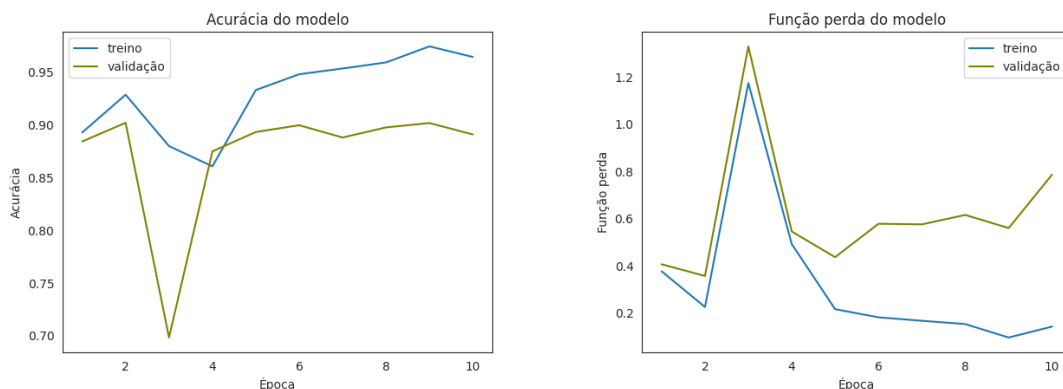


Figura 22 – (a) Acurácia do modelo VI (b) Função de perda do modelo VI

O modelo treinado foi utilizado no conjunto de teste para previsão e classificação das falhas, sendo gerado a partir dos valores obtidos a matriz de confusão (Fig. 23) dos tipos de falhas das ondas gravitacionais.

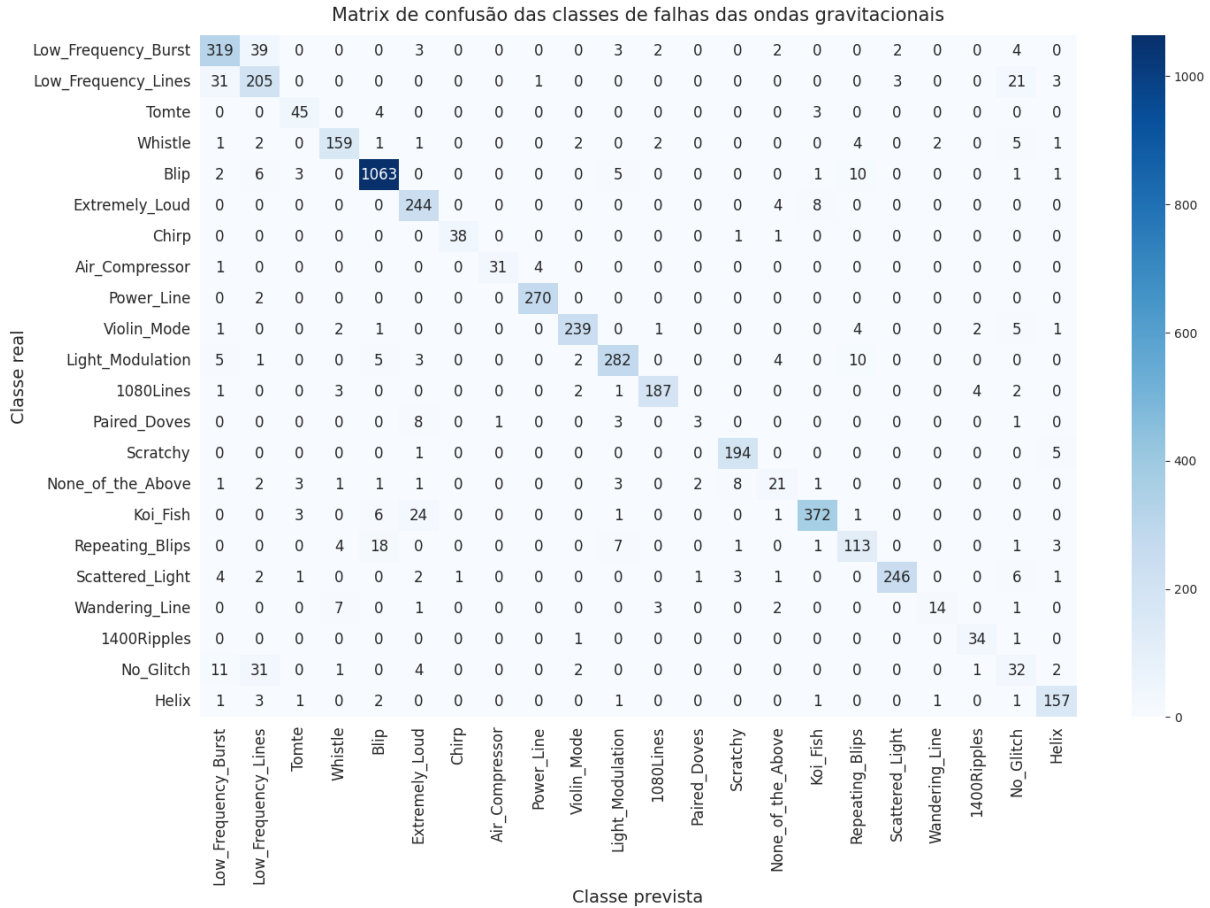


Figura 23 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo VI

Com a matriz de confusão, obteve-se uma acurácia para o conjunto de teste de 90,16%, a precisão foi de 75,37%, o *recall* de 90,15% e a *MF1-score* do mesmo foi de 82,10%. A partir da matriz de confusão também foi possível calcular as métricas como acurácia, precisão, *recall* e *f1-score* para cada uma das classes de falhas de acordo com a Tab. 14. Observa-se que a classe *Blip*, com 5096 itens no conjunto de treinamento, possui acurácia de 97,34% e precisão de 96,55%. A classe *Scattered Light* com 1.232 amostras de treinamento, apresenta acurácia de 91,79% e precisão de 98,01%.

Já classes como *Helix*, com 168 elementos no conjunto de testes, apresentou acurácia de 93,45% e precisão de 90,23% e a *Chirp*, que possui 164 amostras no conjunto de treinamento, alcançou acurácia de 95,00% e precisão de 97,44%. O modelo mostra dificuldades em diferenciar as classes *Paired Doves*, *None of the Above*, *No Glitch* e *Wandering Line*, esta última apresenta precisão alta, mas acurácia baixa. Ao verificar a matriz de confusão, observa-se que 7 elementos classificados incorretamente como

pertencente à ela são, na verdade, pertencentes à classe *Whistle*.

Classes de Falhas	Acurácia (%)	Precisão (%)	Recall (%)	F1-score (%)
<i>Low Frequency Burst</i>	88,6	84,39	85,29	84,84
<i>Low Frequency Lines</i>	77,65	69,97	77,65	73,61
<i>Tomte</i>	86,54	80,36	86,54	83,33
<i>Whistle</i>	88,33	89,83	88,33	89,08
<i>Blip</i>	97,34	96,55	97,34	96,94
<i>Extremely Loud</i>	95,31	83,56	95,31	89,05
<i>Chirp</i>	95,00	97,44	95,00	96,20
<i>Air Compressor</i>	86,11	96,88	86,11	91,18
<i>Power Line</i>	99,26	98,18	99,26	98,72
<i>Violin Mode</i>	93,36	96,37	93,36	94,84
<i>Light Modulation</i>	90,38	92,16	90,38	91,26
<i>1080 Lines</i>	93,50	95,90	93,50	94,68
<i>Paired Doves</i>	18,75	50,00	18,75	27,27
<i>Scratchy</i>	97,00	93,72	97,00	95,33
<i>None of the Above</i>	47,73	58,33	47,73	52,50
<i>Koi Fish</i>	91,18	96,12	91,18	93,58
<i>Repeating Blips</i>	76,35	79,58	76,35	77,93
<i>Scattered Light</i>	91,79	98,01	91,79	94,80
<i>Wandering Line</i>	50,00	82,35	50,00	62,22
<i>1400 Ripples</i>	94,44	82,93	94,44	88,31
<i>No Glitch</i>	38,10	39,51	38,10	38,79
<i>Helix</i>	93,45	90,23	93,45	91,81

Tabela 14 – Métricas das classes do modelo de rede VI

#### 4.9 Modelo de Rede Convolutacional VII

Layer (type)	Output Shape	Param #
3x3 <i>Convolutional2D</i>	(None, 278, 338, 16)	448
2x2 <i>MaxPooling2D</i>	(None, 139, 169, 16)	0
3x3 <i>Convolutional2D</i>	(None, 137, 167, 32)	4640
2x2 <i>MaxPooling2D</i>	(None, 68, 83, 32)	0
3x3 <i>Convolutional2D</i>	(None, 66, 81, 64)	18496
2x2 <i>MaxPooling2D</i>	(None, 33, 40, 64)	0
3x3 <i>Convolutional2D</i>	(None, 31, 38, 64)	36928
2x2 <i>MaxPooling2D</i>	(None, 15, 19, 64)	0
3x3 <i>Convolutional2D</i>	(None, 13, 17, 128)	73856
2x2 <i>MaxPooling2D</i>	(None, 6, 8, 128)	0
3x3 <i>Convolutional2D</i>	(None, 4, 6, 128)	147584
2x2 <i>MaxPooling2D</i>	(None, 2, 3, 128)	0
<i>Flatten</i>	(None, 768)	0
<i>Dense</i>	(None, 256)	196864
<i>Dense</i>	(None, 22)	5654
<i>Total params: 484470 (0.07 MB)</i>		
<i>Trainable params: 484470 (0.07 MB)</i>		
<i>Non-trainable params: 0 (0.00 Byte)</i>		

Tabela 15 – Formato do modelo de rede VII

No sétimo modelo, o formato das saídas de cada camada e a quantidade de parâmetros treinados por camada podem ser verificados na Tab. 15.

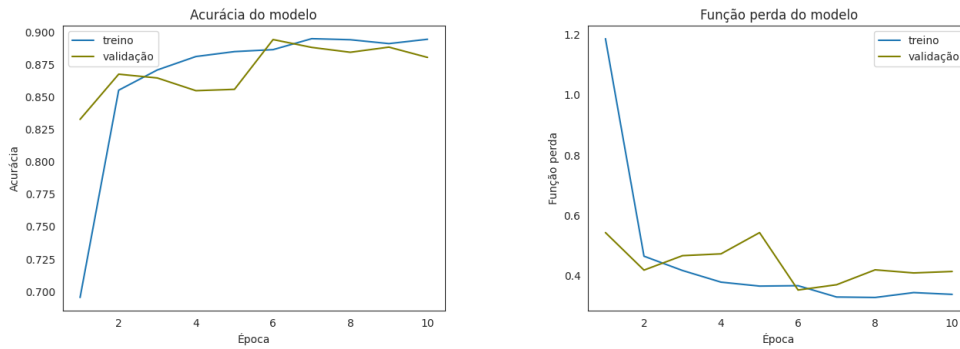


Figura 24 – (a) Acurácia do modelo VII (b) Função de perda do modelo VII

A acurácia do treinamento foi de 89,42% e do conjunto de validação foi de 88,02%, conforme Fig. 24 (a). A função de perda, de acordo com os conjuntos de treino e de validação, ilustrada na Fig. 24 (b), mostra que seu valor é de 0,33 na última época no conjunto de treinamento. No conjunto de validação, ela varia de 0,54 à 0,41.

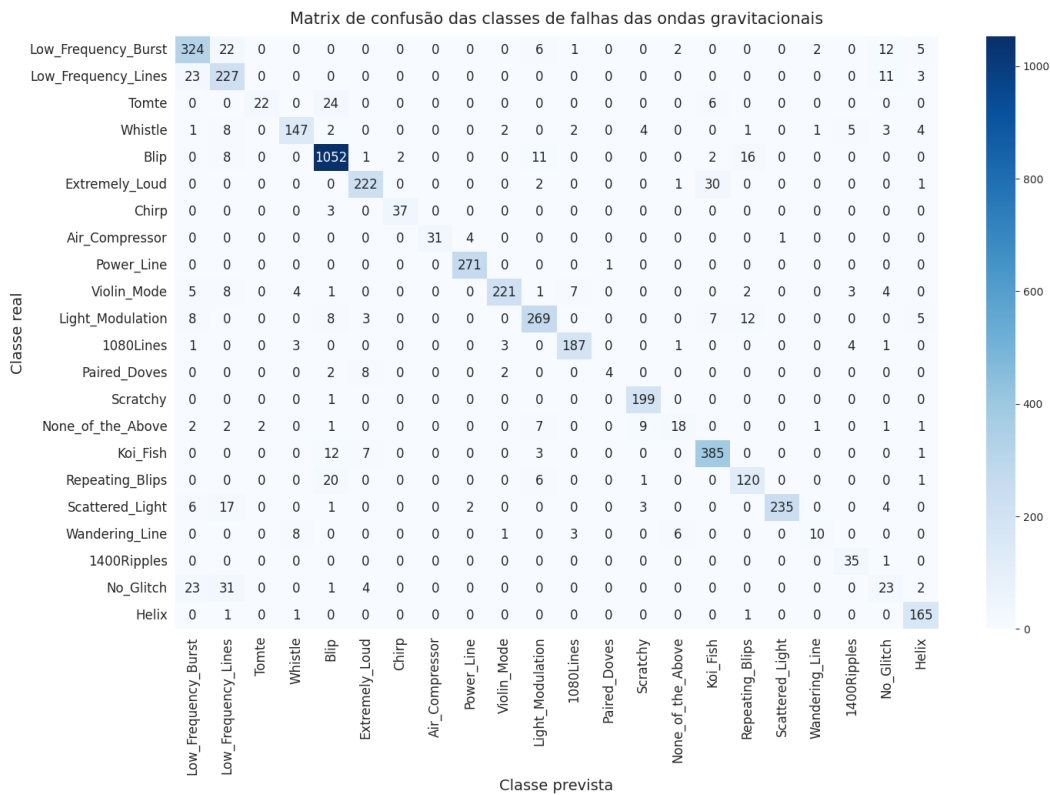


Figura 25 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo VII

O modelo treinado foi utilizado no conjunto de teste para previsão e classificação

das falhas, sendo gerado a partir dos valores obtidos a matriz de confusão (Fig. 25) dos tipos de falhas das ondas gravitacionais. Com a matriz de confusão, obteve-se uma acurácia para o conjunto de teste de 88,80%, a precisão foi de 72,27%, o *recall* foi de 88,56% e a *f1-score* do mesmo foi de 79,59%. A partir da matriz de confusão, também foi possível calcular as métricas como acurácia, precisão, *recall* e *f1-score* para cada uma das classes de falhas, como pode ser visto na Tab. 16.

Classes de Falhas	Acurácia (%)	Precisão (%)	<i>Recall</i> (%)	<i>F1-score</i> (%)
<i>Low Frequency Burst</i>	90,00	82,44	86,63	84,49
<i>Low Frequency Lines</i>	85,98	70,06	85,98	77,21
<i>Tomte</i>	42,31	91,67	42,31	57,89
<i>Whistle</i>	81,67	90,18	81,67	85,71
<i>Blip</i>	96,34	93,26	96,34	94,77
<i>Extremely Loud</i>	86,72	90,61	86,72	88,62
<i>Chirp</i>	92,50	94,87	92,50	93,67
<i>Air Compressor</i>	86,11	100,00	86,11	92,54
<i>Power Line</i>	99,63	97,83	99,63	98,72
<i>Violin Mode</i>	86,33	96,51	86,33	91,13
<i>Light Modulation</i>	86,22	88,20	86,22	87,20
<i>1080 Lines</i>	93,50	93,50	93,50	93,50
<i>Paired Doves</i>	25,00	80,00	25,00	38,10
<i>Scratchy</i>	99,50	92,13	99,50	95,67
<i>None of the Above</i>	40,91	64,29	40,91	50,00
<i>Koi Fish</i>	69,85	89,53	94,36	91,89
<i>Repeating Blips</i>	81,08	78,95	81,08	80,00
<i>Scattered Light</i>	87,69	99,58	87,69	93,25
<i>Wandering Line</i>	35,71	71,43	35,71	47,62
<i>1400 Ripples</i>	97,22	74,47	97,22	84,34
<i>No Glitch</i>	27,38	38,33	27,38	31,94
<i>Helix</i>	98,21	87,77	98,21	92,70

Tabela 16 – Métricas das classes do modelo de rede VII

É possível verificar que a classe *Blip*, que possui a maior quantidade de itens no conjunto de treinamento possui uma acurácia de 96,34% e uma precisão de 93,26%. Da mesma forma, a classe *Power Line*, que possui 1.256 amostras de treinamento, possui acurácia de 99,63% e precisão de 97,83%.

Entretanto, classes menores, como a *Scratchy*, que possui 200 amostras no conjunto de testes, alcançou acurácia de 99,50% e precisão de 92,13%. A *Air Compressor*, que possui 236 elementos no conjunto de treinamento e 36 no conjunto de testes, apresentou uma acurácia de 86,11% e uma precisão de 100,00%, ou seja, predisse corretamente todas as amostras do conjunto de teste.

De fato, o modelo apresenta métricas satisfatórias na maioria das classes, apresentando dificuldades em diferenciar as classes *Paired Doves*, *None of the Above*, *No Glitch*, *Wandering Line*, *Tomte* e *Koi Fish*. Estas duas últimas apresentam precisão alta, mas

acurácia baixa. Ao verificar a matriz de confusão, 24 elementos classificados incorretamente como pertencente à classe *Tomte* e 12 elementos classificados incorretamente pertencentes à classe *Koi Fish* são, na verdade, pertencentes à classe *Blip*, sendo que ambas estas classes possuem grandes semelhanças com a classe *Blip*, o que pode ter levado a essas classificações equivocadas.

#### 4.10 Modelo de Rede Convocucional VIII

No oitavo modelo, o formato das saídas de cada camada e a quantidade de parâmetros treinados por camada podem ser verificados na Tab. 17.

<i>Layer (type)</i>	<i>Output Shape</i>	<i>Param #</i>
3x3 <i>Convolutional2D</i>	(None, 278, 338, 16)	448
2x2 <i>MaxPooling2D</i>	(None, 139, 169, 16)	0
3x3 <i>Convolutional2D</i>	(None, 137, 167, 32)	4640
2x2 <i>MaxPooling2D</i>	(None, 68, 83, 32)	0
3x3 <i>Convolutional2D</i>	(None, 66, 81, 64)	18496
2x2 <i>MaxPooling2D</i>	(None, 33, 40, 64)	0
3x3 <i>Convolutional2D</i>	(None, 31, 38, 64)	36928
2x2 <i>MaxPooling2D</i>	(None, 15, 19, 64)	0
3x3 <i>Convolutional2D</i>	(None, 13, 17, 128)	73856
2x2 <i>MaxPooling2D</i>	(None, 6, 8, 128)	0
3x3 <i>Convolutional2D</i>	(None, 4, 6, 256)	295168
2x2 <i>MaxPooling2D</i>	(None, 2, 3, 256)	0
<i>Flatten</i>	(None, 1536)	0
<i>Dense</i>	(None, 256)	393472
<i>Dense</i>	(None, 22)	5654
<hr/>		
<i>Total params: 828662 (0.07 MB)</i>		
<i>Trainable params: 828662 (0.07 MB)</i>		
<i>Non-trainable params: 0 (0.00 Byte)</i>		

Tabela 17 – Formato do modelo de rede VIII

A acurácia do treinamento foi de 89,37% e do conjunto de validação foi de 88,29% conforme Fig. 26 (a).

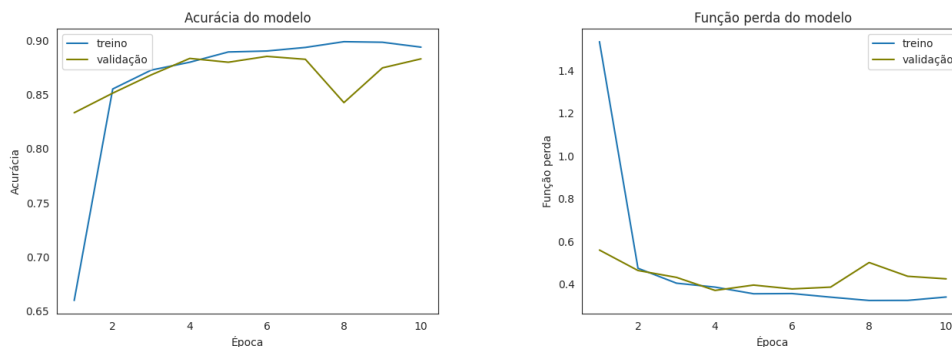


Figura 26 – (a) Acurácia do modelo VIII

(b) Função de perda do modelo VIII

A função de perda, de acordo com os conjuntos de treino e de validação da Fig. 26 (b), mostra que seu valor é de 0,33 na última época no conjunto de treinamento. Enquanto que no conjunto de validação ela varia de 0,56 à 0,42.

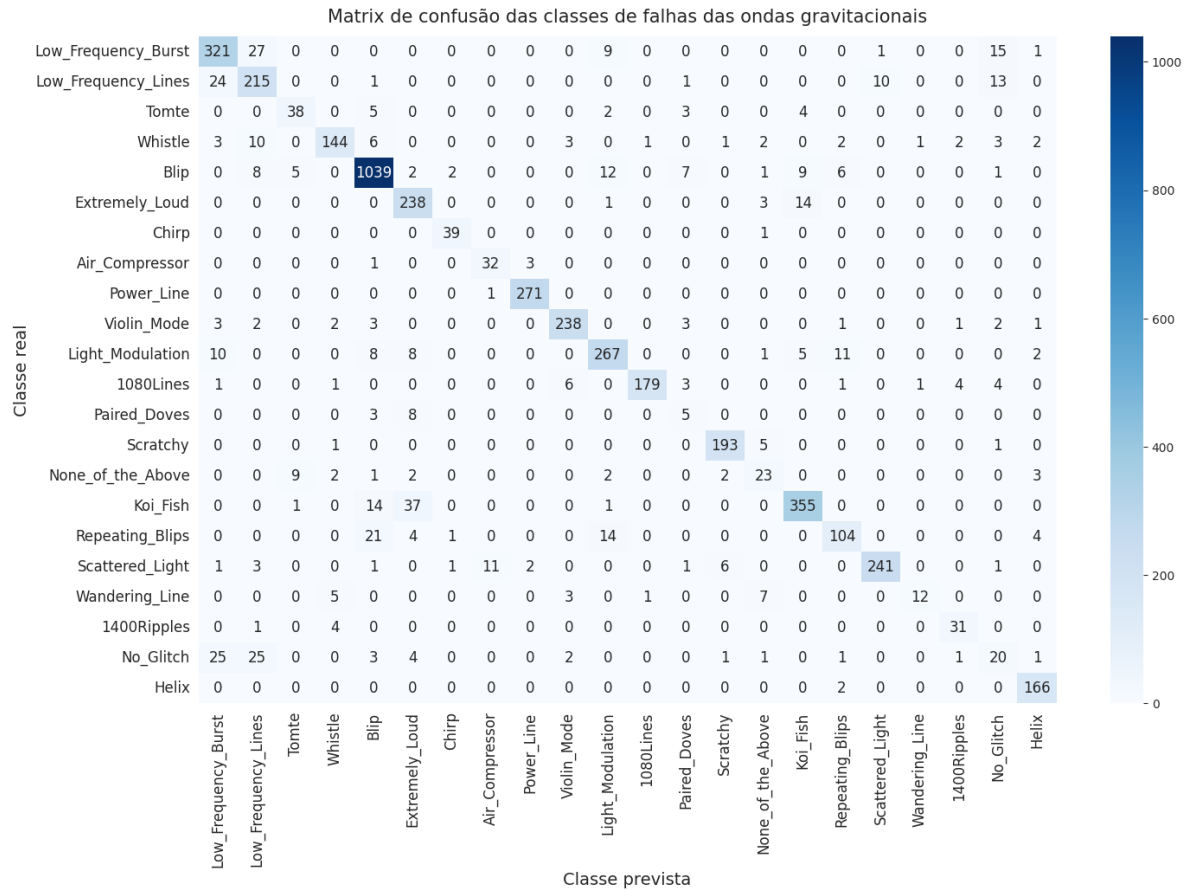


Figura 27 – Matriz de confusão dos tipos de falhas das ondas gravitacionais do modelo VIII

O modelo treinado foi utilizado no conjunto de teste para previsão e classificação das falhas, sendo gerado a partir dos valores obtidos a matriz de confusão (Fig. 27) dos tipos de falhas das ondas gravitacionais. Com a matriz de confusão, obteve-se uma acurácia para o conjunto de teste de 88,11%, a precisão foi de 71,74%, o *recall* foi de 88,07% e a *MF1-score* do mesmo foi de 79,07%, enquanto que a partir da matriz de confusão, também foi possível calcular as métricas como acurácia, precisão, *recall* e *f1-score* para cada uma das classes de falhas como pode ser visto na Tab. 18.

Constatou-se que a classe *Blip*, que possui a maior quantidade de itens no conjunto de treinamento, possui uma acurácia de 95,15% e uma precisão de 93,94%. Da mesma forma, a classe *Violin Mode*, que possui 1.136 amostras no conjunto de treinamento, apresenta acurácia de 92,97% e precisão de 94,44%.

Enquanto que classes menores como a *Chirp*, que possui 164 itens no conjunto de

Classes de Falhas	Acurácia (%)	Precisão (%)	Recall (%)	F1-score (%)
<i>Low Frequency Burst</i>	89,17	82,73	85,83	84,25
<i>Low Frequency Lines</i>	81,44	73,88	81,44	77,48
<i>Tomte</i>	73,08	71,70	73,08	72,38
<i>Whistle</i>	80,00	90,57	80,00	84,96
<i>Blip</i>	95,15	93,94	95,15	94,54
<i>Extremely Loud</i>	92,97	78,55	92,97	85,15
<i>Chirp</i>	97,50	90,70	97,50	93,98
<i>Air Compressor</i>	88,89	72,73	88,89	80,00
<i>Power Line</i>	99,63	98,19	99,63	98,91
<i>Violin Mode</i>	92,97	94,44	92,97	93,70
<i>Light Modulation</i>	85,58	86,69	85,58	86,13
<i>1080 Lines</i>	89,50	98,90	89,50	93,96
<i>Paired Doves</i>	31,25	21,74	31,25	25,64
<i>Scratchy</i>	96,50	95,07	96,50	95,78
<i>None of the Above</i>	52,27	52,27	52,27	52,27
<i>Koi Fish</i>	87,01	91,73	87,01	89,31
<i>Repeating Blips</i>	70,27	81,25	70,27	75,36
<i>Scattered Light</i>	89,93	95,63	89,93	92,69
<i>Wandering Line</i>	42,86	85,71	42,86	57,14
<i>1400 Ripples</i>	86,11	79,49	86,11	82,67
<i>No Glitch</i>	23,81	33,33	23,81	27,78
<i>Helix</i>	98,81	92,22	98,81	95,40

Tabela 18 – Métricas das classes do modelo de rede VIII

treinamento e 40 elementos no conjunto de testes, apresentou uma acurácia de 97,50% e uma precisão de 90,70% e a *Helix*, que possui 780 elementos no conjunto de treinamento e 168 amostras no conjunto de testes, alcançou acurácia de 98,81% e precisão de 92,22%.

Dessa forma, o modelo apresenta métricas satisfatórias na maioria das classes, apresentando dificuldades em diferenciar as classes *Paired Doves*, *None of the Above*, *No Glitch* e *Wandering Line*, tendo esta última precisão alta, mas acurácia baixa. Ao verificar a matriz de confusão, 7 elementos classificados de forma incorreta como pertencente à ela são, na verdade, pertencentes à classe *None of the Above*, para a qual o modelo apresenta dificuldade em classificar corretamente, talvez devido à falta de um padrão característico para essa classe.

#### 4.11 Comparação dos modelos

Ao utilizar o conjunto de testes nos modelos treinados, os resultados obtidos para acurácia, precisão, *MF1-score* e *recall* na última época de cada modelo podem ser verificados na Tab. 19.

Infere-se que os modelos I e III, pelo valor da acurácia do conjunto de validação de 22,92%, são modelos que não conseguiram generalizar as classes das falhas de forma efetiva. De fato, a precisão do modelo é de 0,9%, o que é refletido nas matrizes de confusão da Fig.

Modelos	Acurácia (%)			Precisão (%)	Recall (%)	MF1-Score (%)
	Treino	Validação	Teste			
Modelo I	22,80	22,92	23,07	0,9	23,07	1,7
Modelo II	91,82	83,79	84,56	66,19	84,54	74,25
Modelo III	22,80	22,92	23,07	0,9	23,07	1,7
Modelo IV	96,69	88,65	89,12	73,81	89,12	80,75
Modelo V	94,73	85,54	86,73	68,47	89,81	77,70
Modelo VI	96,44	89,08	90,16	75,37	90,15	82,10
Modelo VII	89,42	88,02	88,80	72,27	88,56	79,59
Modelo VIII	89,37	88,29	88,11	71,74	88,07	79,07

Tabela 19 – Métricas do conjunto de testes dos modelos

13 e da Fig. 17. Nessas matrizes, as 1.092 amostras do conjunto de teste da classe *Blip* são classificadas de forma correta, porém, os elementos das demais classes são todos previstos como pertencentes também a classe *Blip*. Ambos os modelos utilizaram camadas de *dropout* com parâmetro de 0,5 entre as camadas de *maxpooling* e de convolução, podendo ser este o motivo do baixo desempenho do treinamento junto com a influência do desbalanceamento das classes.

Os demais modelos II, IV, V, VI, VII e VIII possuem, respectivamente, valor da acurácia do conjunto de validação de 83,79%, 88,65%, 85,54%, 89,08%, 88,02% e 88,29%, indicando que estes modelos conseguiram generalizar as classes das falhas. Verificando o valor do *recall*, estes modelos apresentam, respectivamente, os valores de 84,54%, 89,12%, 89,81%, 90,15%, 88,56% e 88,07%, indicando que eles possuem bom desempenho em detectar as classes positivamente. Em relação ao valor da *MF1-score*, apresentam, respectivamente, os valores de 74,25%, 80,75%, 77,70%, 82,10%, 79,59% e 79,07%, indicando que, apesar de apresentarem um bom *recall*, têm perda no desempenho em conseguir acertar as classes previstas. Isso pode ser verificado nos valores de precisão, que são, respectivamente, 0,9%, 66,19%, 0,9%, 73,81%, 68,47%, 75,37%, 72,27% e 71,74%.

Dentre estes modelos, o que possui melhor acurácia ao ser aplicado no conjunto de testes foi o modelo VI, com acurácia de 90,16%. O modelo VI também apresenta a melhor *MF1-score* dentre os modelos, no valor de 82,10%. No entanto, na Fig. 22 (b), é visto que a função de perda do conjunto de validação na última época do treinamento foi de 78%, um valor alto indicando que as classes previstas no conjunto de validação não condizem neste percentual com as classes reais. Neste modelo, a maioria das classes apresentam individualmente ótimas métricas, explicitando que o modelo consegue prever de forma satisfatória o conjunto de testes, apresentando dificuldade em diferenciar as classes *Paired Doves*, *None of the Above*, *No Glitch* e *Wandering Line*. Sendo que a classe *Wandering Line* apresenta problema na acurácia, mas com boa precisão, devido à confusão com os elementos pertencentes à classe *Whistle*.

Por outro lado, os modelos VII e VIII apresentam, respectivamente nas Fig. 24 (b) e Fig. 24(b), valores mais baixos no conjunto de validação para a função perda na última época, respectivamente 41% e 42%. A acurácia obtida ao ser aplicado no conjunto de testes para o modelo VII foi de 88,80% e para o modelo VIII foi de 88,11. No entanto, a *MF1-score* obtida para cada um foi, respectivamente, de 79,59% e 79,07%, valores menores que o obtido no modelo VI.

O modelo VII possui ótimas métricas individuais para a maioria das classes, tendo dificuldade em diferenciar as classes *Paired Doves*, *None of the Above*, *No Glitch*, *Wandering Line*, *Tomte* e *Koi Fish*. As classes *Tomte* e *Koi Fish* possuem acurácias baixas, mas valores de precisão alta devido à classificação incorreta dos elementos pertencentes à classe *Blip*, que possuem fortes semelhanças com estas outras duas classes.

O modelo VII também apresenta métricas individuais satisfatórias para a maioria das classes, apresentando dificuldades em diferenciar as classes *Paired Doves*, *None of the Above*, *No Glitch* e *Wandering Line*. Notando que a classe *Wandering Line* apresenta boa precisão, apesar da baixa acurácia, fato que se deve a classificação dos elementos pertencente à classe *None of the Above*, que o modelo apresenta dificuldade em identificar os padrões.

Deste modo, concluí-se que especialmente esses modelos VI, VII e VIII apresentaram ótimos resultados após os treinamentos e aplicações nos conjuntos de testes, apresentando dificuldades apenas para encontrar padrões de determinadas classes específicas, que têm em comum serem classes que possuem pequena quantidade de amostras, tais como *Paired Doves*, que possui 76 elementos no conjunto de treinamento e 16 no de teste, *None of the Above*, que possui 228 elementos no conjunto de treinamento e 44 no de teste, *No Glitch*, que possui 428 elementos no conjunto de treinamento e 84 no de teste e *Wandering Line* que possui 116 elementos no conjunto de treinamento e 28 no de teste.

## 5 CONCLUSÃO

A proposta de classificar corretamente as classes de falhas utilizando os dados disponibilizados pelo projeto *Gravity Spy* através da aplicação de diferentes modelos de CNN's mostrou resultados satisfatórios, porém com algumas ressalvas a serem observadas devido à particularidade do conjunto de dados.

Com os valores obtidos dos modelos treinados, verifica-se que os modelos I e III se mostraram ineficientes na classificação das falhas, na medida que conseguiram identificar e classificar corretamente apenas as imagens pertencentes à classe *Blip*. Além disso, identificaram as imagens pertencentes às demais classes como sendo pertencente à classe *Blip*, apresentando uma acurácia de apenas 22,92%. A diferença desses modelos para os outros consiste no fato de que ambos modelos utilizam camadas de *dropout* com parâmetro de 0,5 entre as camadas de *maxpooling* e de convolução, podendo isso, junto com o desbalanceamento das classes, ter contribuído para o baixo desempenho do treinamento.

Porém, nos demais modelos, foi possível obter a classificação das falhas com uma acurácia maior. Dentre esses, os modelos IV, VI, VII e VIII possuem valor da acurácia do conjunto de validação acima de 88%. Deste modo, esses modelos apresentam uma boa capacidade de generalização após o treinamento. Além disso, os valores do *recall* desses modelos estão acima de 88%, indicando que eles possuem bom desempenho em detectar as classes positivamente. No entanto, a medida *MF1-score* possui um valor máximo de 82% indicando que esses modelos possuem um desempenho pior na precisão.

Dentre esses modelos, que apresentam acurácia no conjunto de validação acima de 88%, verifica-se que o modelo VI é o que possui a maior acurácia no conjunto de testes, assim como é o modelo com o melhor *MF1-score*, apesar de o conjunto de validação ter apresentado uma função de perda de 78%. Enquanto que os modelos VII e VIII que possuem valores mais baixo da função de perda, respectivamente 41% e 42%, apresentam acurácia no conjunto de testes respectivamente de 88,80% e de 88,11%, porém com *MF1-score* abaixo de 80%.

Deste modo, conclui-se que especialmente esses modelos VI, VII e VIII apresentaram excelentes resultados após os treinamentos e aplicações nos conjuntos de testes, evidenciando que eles conseguem prever de forma satisfatória as classes. Entretanto, algumas classes mostraram-se de difícil aprendizagem para os modelos, especialmente as classes *Paired Doves*, *None of the Above*, *No Glitch* e *Wandering Line*, que são classes com menor quantidade de amostras nos conjuntos de treinamento e testes.

Ressalta-se que os modelos treinados apresentaram, ao comparar os valores de acurácia e *f1-score* de cada uma das classes de falhas, uma tendência a classificar as

imagens como sendo pertencente às classes que possuem maior quantidade de amostras no conjunto de treinamento. Isso indica que o desbalanceamento do conjunto de dados interfere negativamente o treinamento dos modelos. Uma possibilidade para evitar que haja esta interferência seria utilizar a técnica de *undersampling*, diminuindo a quantidade de exemplos das classes com maior quantidade de exemplos, equilibrando com a quantidade de exemplos da classe que possui menos exemplos. No entanto, no caso do conjunto de dados utilizado, a classe com menor quantidade de exemplos é a classe Paired Doves, que possui 76 exemplos no conjunto de treinamento. Essa quantidade de exemplo talvez não seja suficiente para o treinamento dos modelos de forma que eles consigam aprender e generalizar a classificação das classes de falhas.

Apesar da proposta de classificar as falhas do LIGO através de redes CNN ter apresentado resultados satisfatórios com os modelos VI, VII e VIII, devido a limitação da quantidade de exemplos das classes no conjunto de dados, pode-se explorar outros tipos de técnicas mais avançadas de aprendizado de máquina, como redes adversárias generativas, para gerar amostras para as classes menores e assim obter melhores resultados na classificação das falhas.

## REFERÊNCIAS

- ABBOTT, B. P. *et al.* Observation of gravitational waves from a binary black hole merger. **Phys. Rev. Lett.**, v. 116, n. 6, p. 061102, 2016.
- BAHAADINI, S. *et al.* Machine learning for gravity spy: Glitch classification and dataset. **Information Sciences.**, v. 444, p. 172–186, 2018.
- COUGHLIN, S. **Updated Gravity Spy Data Set**. Zenodo, 2018. Acessado em 09/2023. Disponível em: <https://doi.org/10.5281/zenodo.1476551>.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. MIT Press, 2016. Acessado em 09/2023. Disponível em: <http://www.deeplearningbook.org>.
- HARRAND, R. **Gravity Spy Data Set**. Kaggle, 2020. Acessado em 09/2023. Disponível em: <https://www.kaggle.com/datasets/tentotheminus9/gravity-spy-gravitational-waves>.
- ROBINET, F. *et al.* Omicron: a tool to characterize transient noise in gravitational-wave detectors. **SoftwareX**, v. 12, p. 100620, 2020.
- YAN, J.; LEUNG, A. P.; HUI, D. C. Y. On improving the performance of glitch classification for gravitational wave detection by using generative adversarial networks. **Mon. Not. Roy. Astron. Soc.**, v. 515, n. 3, p. 4606–4621, 2022.
- ZOONIVERSE. 2023. Acessado em 09/2023. Disponível em: <https://www.zooniverse.org/projects/zooniverse/gravity-spy/>.