



UNIVERSIDADE FEDERAL DA FRONTEIRA SUL
CAMPUS REALEZA

PAULO DE TARSO FERREIRA DOS SANTOS

DETECÇÃO DE SUPERNOVAS VIA REDES
NEURAIIS

REALEZA
2022

PAULO DE TARSO FERREIRA DOS SANTOS

DETECÇÃO DE SUPERNOVAS VIA REDES NEURAIS

Trabalho de Conclusão de Curso apresentado ao Curso de Licenciatura em Física da Universidade Federal da Fronteira Sul (UFFS), como requisito para obtenção do título de licenciado em física.

Orientador: Prof. Dr. Everton Artuso

Realeza, julho de 2022

Bibliotecas da Universidade Federal da Fronteira Sul - UFFS

Santos, Paulo de Tarso Ferreira dos
Detecção de Supernovas Via Redes Neurais / Paulo de
Tarso Ferreira dos Santos. -- 2022.
48 f.:il.

Orientador: Prof. Dr. Everton Artuso

Trabalho de Conclusão de Curso (Graduação) -
Universidade Federal da Fronteira Sul, Curso de
Licenciatura em Física, Realeza, PR, 2022.

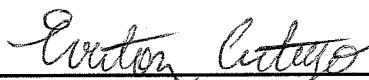
1. Supernovas. 2. Machine Learning. 3. Redes Neurais.
4. Invariância Rotacional. I. Artuso, Everton, orient.
II. Universidade Federal da Fronteira Sul. III. Título.

PAULO DE TARSO FERREIRA DOS SANTOS

**DETECÇÃO DE SUPERNOVAS VIA REDES
NEURAIS**

Trabalho de Conclusão de Curso apresentado ao Curso de Licenciatura em Física da Universidade Federal da Fronteira Sul (UFFS), como requisito para obtenção do título de licenciado em física.

Este trabalho foi defendido e aprovado pela banca em 27/ 07/2022



Orientador(a): Dr. Everton Artuso



Prof. Dr. Tobias Heimfarth



Prof. Dr. Eduardo de Almeida

Agradecimentos

A divindade criadora;

A minha família pelo dom da vida e educação que me deram;

Aos bons e maus professores que me ensinaram como lecionar;

Aos grandes amigos que aqui construí e me suportaram;

A Rúbiamara Tadiotto, pela leveza de nossa relação;

Ao Prof. Everton Artuso por todo amparo e compreensão;

Resumo

Com o presente trabalho buscou-se por apresentar uma ideia geral acerca do que são os eventos astronômicos conhecidos como supernovas, o motivo de investigá-los e uma maneira de realizar este movimento fazendo uso de heurísticas de *Machine Learning* e *Deep Learning* conhecidas como redes neurais convolucionais (CNN). Verificou-se como uma CNN pode ser utilizada para a identificação de supernovas e como as CNN que possuem propriedades de invariância rotacional conseguem produzir melhores resultados no que diz respeito á detecção destes eventos astronômicos. Observou-se como a metodologia de CNN com invariância rotacional, chamada de CAP e proposta por [Reyes et al. 2018], alcança resultados mais refinados quando comparados aos da modelagem chamada *Deep-HiTs* (que precedeu a metodologia CAP). Ainda, investigou-se os motivos de tal melhoria nos resultados, verificando-se que é justamente pelo fato da rede neural fazer uso de camadas cujas operações inferidas sob os dados rotacionados são capazes de realizar o processamento e predizer, corretamente, se os dados de uma determinada amostra avaliada corroboram para a hipótese de que um dado objeto observado é uma supernova (candidato) ou não (artefato). Por fim, ilustrou-se a maneira como se compararam os resultados de tais metodologias através de testes estatísticos.

Palavras-chave: Supernovas; machine learning; redes neurais; invariância rotacional.

Abstract

The work sought to present a general idea about what astronomical events known as supernovas are, the reason to investigate them and a way to carry out this movement using *Machine Learning* and *Deep Learning* heuristics is known as convolutional neural networks (CNN). It is verified how a CNN can be used to identify supernovae and how CNNs that have rotational invariance properties are able to produce better results with regard to the detection of these astronomical events. It was observed how the CNN methodology with rotational invariance, called CAP and proposed by [Reyes et al. 2018], achieves more refined results when compared to the *Deep-HiTs* modeling (which preceded the CAP methodology). Still, the reasons for such improvement in the results were investigated, verifying that it is exactly the fact that the network of layers of operations inferred under the rotated data are capable of performing the processing and correctly predicting the data of a sample data. recorded that corroborates an object verified for a supernova hypothesis (candidate) or not (artifact is observed for a supernova object). Finally, a way to compare the results of such methodologies through statistical tests was illustrated.

Keywords: supernovae; machine learning; neural networks; rotational invariance.

Lista de ilustrações

Figura 1 – Estrutura de cebola de uma estrela massiva	15
Figura 2 – Supernova tipo Ia	17
Figura 3 – Supernova remanescente Cas A.	18
Figura 4 – Neurônio <i>Perceptron</i>	21
Figura 5 – Rede neural com neurônios tipo <i>Perceptron</i>	22
Figura 6 – Rede neural com apenas um neurônio na camada oculta.	23
Figura 7 – Rede neural de perceptrons com um Δ peso adicionado.	24
Figura 8 – Redes neurais com uma e com várias camadas ocultas.	27
Figura 9 – Exemplo de campo receptível local.	30
Figura 10 – Exemplo de camadas de convolução	31
Figura 11 – Exemplos filtros de convolução (Kernels)	32
Figura 12 – Exemplo de rede neural convolucional	32
Figura 13 – Rotações de uma amostra	34
Figura 14 – Exemplos de operações de pooling	35
Figura 15 – Matriz de confusão	38
Figura 16 – métricas e suas equações	39
Figura 17 – Precisão e acurácia	39
Figura 18 – Arquitetura utilizada pela metodologia CAP.	40
Figura 19 – Resultados das médias do <i>Deep-HiTs</i> e CAP para 4 métricas e os respectivos <i>p</i> -valores encontrados.	44
Figura 20 – Mapas de calor para as metodologias <i>Deep-HiTs</i> e CAP	45

Lista de abreviaturas e siglas

CAP - *Cyclic average pooling*

CAPE - *Cyclic average pooling ensemble.*

CNN - *Convolutional neural network.*

CPU - *Central processing unit.*

GPU - *Graphics processing unit.*

HiTs - *High cadence Transient survey*

I.A - Inteligência artificial

LRP - *Layer relevance propagation*

ReLU - *rectified linear unit*

RGB - *REd, Blue and Green*

SNR - *Signal-to-noise ratio*

TCC - Trabalho de conclusão de curso

Sumário

1	INTRODUÇÃO	10
2	METODOLOGIA	12
3	FUNDAMENTAÇÃO TEÓRICA	14
3.1	SUPERNOVAS	14
3.2	REDES NEURAIIS	19
3.3	REDES NEURAIIS CONVOLUCIONAIS	29
3.4	REDES NEURAIIS CONVOLUCIONAIS COMO INVARIANTES ROTACIONAIS	33
3.4.1	INVARIÂNCIA ROTACIONAL	33
3.4.2	VISUALIZANDO E COMPREENDENDO AS DECISÕES DE UMA CNN . .	35
3.4.3	INVARIÂNCIA ROTACIONAL DE CNNs PARA A DETECÇÃO DE SUPER- NOVAS	37
4	RESULTADOS E DISCUSSÕES	43
5	CONSIDERAÇÕES FINAIS	47
	REFERÊNCIAS	48

1 INTRODUÇÃO

Com a evolução das pesquisas em astrofísica, diferencia-se as supernovas como sendo do tipo I, associadas a explosão termonuclear de uma estrela do tipo anã branca e tipo II, que é o resultado do colapso de uma estrela extremamente massiva. Supernova é a denominação que se dá para o fenômeno de explosão de uma determinada estrela, processo este que lança quantidades consideravelmente altas de matéria e energia para o espaço ao seu redor. Segundo [Horvath 2008], este objeto é um dos berços dos diferentes elementos químicos naturais (que não os mais elementares do universo como Hidrogênio, Lítio e Hélio), inclusive os que fazem parte da composição de nosso planeta e das formas de vida que aqui habitam. Tais elementos químicos são formados principalmente nos processos de fusão nuclear no interior de tais estrelas, mas também, pelo choque das suas camadas mais internas com as camadas mais externas que ocorre no momento da explosão. Tem-se, então, que debruçar-se sobre a investigação destes fenômenos é estudar sobre como os elementos que compõem nosso próprio organismo foram gerados. O universo é tão grande que, da perspectiva da astrofísica, ao observar eventos que se encontram em regiões muito distantes de nós, estamos, na realidade, observando um evento que ocorreu há milhões ou até bilhões de anos atrás (dependendo da distância), mesmo com a informação sobre tal evento viajando em nossa direção com a velocidade da luz. Nesse sentido ao investigar-se supernovas distantes, investiga-se, também, sobre como era o universo há bilhões de anos atrás, uma vez que as supernovas brilham tanto que podem ser observadas mesmo estando extremamente distantes de um observador. Em tal contexto, tem-se que observações e análises desses fenômenos realizadas nas últimas décadas apontam para um universo em expansão acelerada. Por sua vez, a investigação da ocorrência de supernovas vem a ser tema de diferentes pesquisas na atualidade. Porém, haja visto que tais eventos são raros e, conseqüentemente, difíceis de serem observados, acaba por ser necessário o desenvolvimento de técnicas para se identificar e detectar tais acontecimentos, observando-se diferentes regiões do universo com poderosos telescópios. Tais observações geram quantidades exorbitantes de dados para serem avaliados e é humanamente impraticável buscar por supernovas em meio tais dados sem utilizar algum tipo de ferramenta que acelere e otimize esse processo.

Com o avanço tecnológico das últimas décadas, emerge o que ficou conhecido como inteligência artificial (I.A), associado ao que uma máquina (computador) deve aprender para ser capaz de realizar tarefas complexas. De modo geral, se trata de uma linha de códigos (algoritmos) que processam um conjunto de informações e, após o processamento, expressam um resultado. Por exemplo, pode-se arquitetar uma linha de códigos para que seja capaz de identificar pássaros em imagens ou, então, realizar o reconhecimento facial

de uma pessoa para permitir o acesso dela ao celular. Logo, o chamado de aprendizado de máquina (*machine learning*) torna-se mais uma ferramenta para cientistas das mais diversas áreas conhecimento, por ser uma excelente ferramenta para se realizar análise de dados. É válido ressaltar-se que o *machine learning* é ideal para se tratar com grandes quantidades de dados. No escopo do aprendizado profundo (*deep learning*), [I. Goodfellow, Y. Bengio, and Courville, Aaron. 2016] expõe que as linhas de algoritmos são capazes de realizar tarefas ainda mais complexas. Aqui, as linhas são dispostas em camadas sucessivas de algoritmos, dando forma ao que se conhece por rede neural artificial, ou simplesmente, rede neural. Com isso, pode-se arquitetar diferentes estruturas (heurísticas) de redes neurais para resolver diferentes problemas de classificação de imagens, por exemplo. Nesse sentido, os astrônomos da modernidade avaliam grandes volumes de dados de objetos celestes que poderosos telescópios produzem, fazendo uso da alta capacidade de processamento que as redes neurais alcançam, levando a investigação em astronomia e astrofísica para uma nova era de pesquisa.

Sob esta luz, no presente trabalho busca-se por investigar a utilização de duas metodologias de redes neurais para a detecção de supernovas, propostas por [Reyes et al. 2018], explorando e elucidando a construção do arcabouço teórico que embasa tal ferramenta e suas propriedades. Para tal efeito, na seção 2 demonstra-se que a estratégia utilizada para realizar esse trabalho é pautada nos conceitos propostos pela metodologia de pesquisa exploratória. Na sequência, na seção 3.1 elucidada-se a ideia geral do que são as supernovas e como tais eventos ocorrem para, na sequência, na seção 3.2, enunciar o que são e como funcionam as redes neurais. Já na seção 3.3 é elencado um caso particular de rede neural chamada de rede neural convolucional (CNN) que conta com camadas que aplicam operações de convolução sobre os dados. Seguindo a metodologia das CNNs, na seção 3.4 investiga-se como as referidas redes podem ser arquitetadas para serem capazes de perceber transformações de rotação nos dados e realizar uma previsão correta. Ainda nesta seção, ilustra-se como a rede realiza esse movimento e o motivo da mesma ser melhor que os modelos que a precederam. Por fim, na seção 4 elenca-se os resultados obtidos por [Reyes et al. 2018] para duas metodologias diferentes de redes neurais, onde se compara tais resultados através de testes estatísticos.

2 METODOLOGIA

Como salienta [C. Selltiz. , L. S. Wrightsman. and S. W. Cook. 1965], existem estratégias de estudo que operam no intuito de descobrir as intuições e potencialidades que um determinado objeto de pesquisa pode promover, tal que essas ações ocorrem na busca de obter-se uma maior proximidade acerca da problemática investigada. Estas ações são denominadas como membros de uma metodologia de pesquisa chamada de metodologia exploratória. O ato de adotar tal estratégia, que não necessariamente conta com a formulação de hipóteses para se iniciar uma caminhada investigativa, permite que o investigando aumente seu repertório sobre um dado tema, inferindo ao mesmo a possibilidade de futuramente, com uma base teórica consolidada sobre a temática desejada, propor a formulação e a iniciação de novos trabalhos acerca da mesma, com maior riqueza de detalhes e de estruturação. Aqui, acaba por ser intrínseca a necessidade do uso de estratégias e planejamentos flexíveis para se realizar a pesquisa, de modo a investigar aspectos variados de um mesmo tema, que vão para além de uma descrição de caráter superficial. No caso particular desse trabalho, o norte problematizador se dá na investigação de métodos computacionais e estatísticos para a detecção de supernovas os quais, tanto a problemática, quanto os métodos estudados, não são trabalhados diretamente ou com aprofundamento nos Componentes Curriculares Regulares (CCR's) do curso de Licenciatura em Física.

Neste contexto, a pesquisa elaborada no presente Trabalho de Conclusão de Curso (TCC) ocorre no intuito de se iniciar-se uma jornada de estudos que tem por objetivo construir um arcabouço teórico necessário para se trabalhar com tais temas futuramente em um programa de pós-graduação. Seguindo a corroboração, [W. G. ZIKMUND. 2000] explana que, em geral, tais métodos de estudo com caráter exploratório podem ser utilizados para, entre outras possibilidades, descobrir novas ideias sobre um tema, para explorar possibilidades e, também, para realizar diagnósticos de fenômenos relacionados ao tema investigado. Ainda, o mesmo autor nos informa que as ações dessa metodologia podem ter resultados consideráveis ao serem aplicadas no momento inicial de uma pesquisa que abrange uma gama espessa de áreas do conhecimento ou de conceitos, possuindo a potencialidade de contribuir com a compreensão da natureza de um dado problema e/ou propiciar a construção de conhecimentos para produzir pesquisas futuramente. Sob esta luz, vale mencionar também que mesmo que o pesquisador possua uma bagagem teórica sobre o tema abordado, a pesquisa exploratória não deixa de ser uma prática a ser considerada no desenvolvimento da atividade, uma vez que um mesmo fenômeno pode possuir justificativas ou explicações diferentes ao ser analisado partindo de premissas diferentes das adotadas inicialmente.

[MATTAR. 2001] elucida que as estratégias que a pesquisa de caráter exploratório possui, além da gama de possibilidades, uma versatilidade que permite permear entre diferentes maneiras de se aprender algo sobre um dado tema. Entre outros métodos, o autor elenca principalmente a realização de um levantamento de experiências e referências secundárias sobre o objeto de pesquisa, a observação informal e o desenvolvimento de estudos de caso previamente selecionados. Nesse enredo, desenvolveu-se essa pesquisa partindo de práticas como estudos dirigidos para o levantamento de referencial bibliográfico, bem como, o acompanhamento de aulas, palestras e cursos disponíveis gratuitamente. [GIL, A. C. 1999] menciona que práticas semelhantes a estas podem ser organizadas levando em conta um planejamento mais maleável, uma vez que objetificam possibilitar uma visualização holística sobre o tema, dando margem para a percepção de novas ideias e maneiras para se estudar uma problemática. Desse modo, durante e após a realização das atividades mencionadas, realizaram-se, também, conversações com o orientador sobre as dúvidas que eventualmente surgiram, momentos estes que julgo como cruciais para a elaboração deste documento, uma vez que os mesmos promoveram ricos debates principalmente no que diz respeito a estruturação e organização do trabalho, as quais sofreram alterações ao longo do tempo.

3 FUNDAMENTAÇÃO TEÓRICA

3.1 SUPERNOVAS

Em [Horvath 2008] observa-se que as investigações iniciais de Fritz Zwicky (1898-1974) e Walter Baade (1839 - 1960) em 1930, levaram a percepção de que existem duas formas de explosão que são potentes o suficiente para ejetar elementos pesados para o universo, sendo denominada por tais autores como sendo supernovas de tipo I e tipo II, dependendo se ejetam ou não gás hidrogênio durante o processo. Com a evolução das pesquisas, diferencia-se as supernovas como sendo do tipo I, associadas a explosão termonuclear de uma estrela do tipo anã branca e tipo II, que são resultado do colapso de uma estrela extremamente massiva. Quando se trata da origem das estrelas, com o progredir de pesquisas em astrofísica e cosmologia, como as de [Sobrinho. 2013], sabe-se que estas são concebidas no interior de nuvens compostas por gases e poeiras onde, através da ação gravitacional entre esses elementos, compõem-se regiões onde a densidade de partículas e gases é elevada. Esse processo de formação pode ser influenciado pelo choque entre essas nuvens ou, então, pela onda de choque formada por uma supernova, objeto este que será apresentado um pouco mais adiante. Ainda, para que a pressão não vença a força gravitacional dentro das nuvens, a temperatura dos gases em dada região não pode ser elevada. A região do espaço que propicia tais condições é denominada como nebulosa escura e em seu interior é possível ocorrer a formação de protoestrelas. Dentro do processo evolutivo da vida de estrelas, a fase inicial é chamada de protoestrela, que se encontra numa posição antes do conjunto de fases chamado de sequência principal. Esta última é caracterizada pelo começo das reações nucleares de fusão do hidrogênio em hélio e é a fase mais duradoura da vida de uma estrela. Se tratando das protoestrelas, elas tem seu início com um conjunto de gases e partículas sendo aglutinados em regiões que ocupam dimensões comparáveis a de nosso sistema solar. Dado o agrupamento desses elementos, por ação gravitacional, localidades mais internas da nuvem tenderam a, de forma gradativa, sofrer um aumento de temperatura, emitindo radiação, dada a conversão de energia potencial em térmica. Nesse ponto, com a evolução temporal de milhares de anos, a temperatura da superfície de tais protoestrelas pode alcançar os $3 \cdot 10^3$ K (sem ocorrer fusão nuclear no interior) e, por sua vez, nesse momento a protoestrela tem alta intensidade luminosa. Mas como, em geral, elas estão escondidas entre as nebulosas escuras, para se observá-las, deve-se utilizar telescópios para mensurações na banda do infravermelho.

Para que se iniciem as reações de fusão nuclear, o interior da protoestrela deve ter uma temperatura que seja da ordem de 10^6 K, tal que o hidrogênio seja convertido em hélio. Aqui, quando ocorre um equilíbrio hidrostático entre a pressão de radiação,

liberada pelas reações de fusão, e a contração gravitacional, a protoestrela passa a ser uma estrela da fase estelar chamada de sequência principal. Como explica [Sobrinho. 2013], as protoestrelas que possuem massas menores que 0,08 massas solares não são capazes de criar as condições suficientes para que a fusão nuclear do hidrogênio aconteça e, por este motivo, não fazem parte da sequência principal. As protoestrelas que superam 200 massas solares também não fazem parte da sequência principal, uma vez que a pressão interna tende a superar a ação gravitacional, fazendo com que tal protoestrela sofra uma desintegração.

É sabido que o tempo que uma estrela irá permanecer na sequência principal é inversamente proporcional a massa da mesma, ou seja, quanto maior a massa, menos tempo ela permanecerá na sequência principal (convertendo hidrogênio em hélio). Neste arcabouço, quando não houver mais hidrogênio para ser “queimado” no centro da estrela, irá cessar, também, as reações de fusão nuclear do hidrogênio, caracterizando o estado de evolução da estrela para a fase de gigante vermelha. Então com a evolução temporal, se dá início às reações de fusão nuclear do hélio que irão gerar elementos como oxigênio e carbono, tal que, se uma estrela se encontrar neste estágio e tiver uma massa menor que oito massas solares, sua jornada se encerrará aqui, de modo que a estrela expulsará suas camadas mais externas, dando origem ao objeto conhecido como nebulosa planetária, possuindo uma estrela do tipo anã branca em seu interior. Já as estrelas que se encontram na fase de gigante vermelha e possuem massa superior a 8 massas solares, irão explodir em uma supernova. Na figura 1 vemos um esquema das reações nucleares.

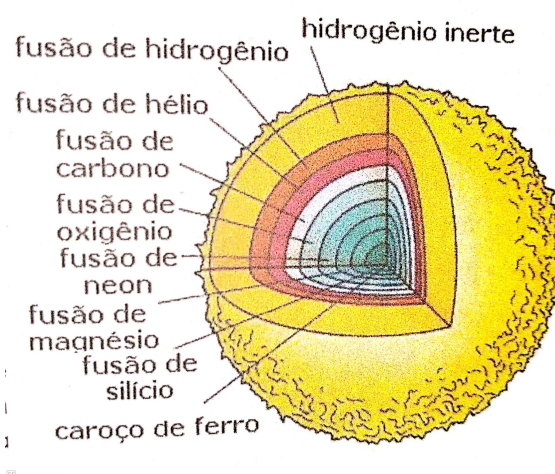


Figura 1 – Estrutura de cebola de uma estrela massa maior que 8 massas solares onde indica-se as reações de fusão nuclear que ocorrem no interior da estrela com a evolução temporal

Fonte: [Horvath 2008]

No escopo das estrelas com massa inicial superior a 8 massa solares, a fase inicial da vida destas é análoga as de estrelas com massa menor, convertendo hidrogênio em hélio e depois hélio em carbono e oxigênio. Porém, como o campo gravitacional é mais intenso

para estrelas mais massivas, o centro destas atinge temperaturas da ordem de $6 \cdot 10^6$ K, capazes de proporcionar a fusão do carbono em neônio, oxigênio, magnésio e sódio. Após acabarem estas reações nucleares, o núcleo da estrela colapsa e pode atingir a temperatura de até 10^9 K, permitindo a fusão neon, dando origem ao oxigênio e o magnésio. Quando a fusão nuclear consumir todo o neon, o núcleo irá colapsar mais uma vez, podendo alcançar a temperatura de $1,5 \cdot 10^9$ K, onde ocorre a fusão oxigênio resultando, principalmente, em enxofre. Após ocorrer a queima do oxigênio, o núcleo se colapsará novamente, de modo que se a temperatura de $2,7 \cdot 10^9$ K for alcançada, ocorre a fusão do silício, resultando em ferro e uma gama de outros elementos. Por fim, quando cessam as reações de fusão nuclear do silício, o núcleo, que nesse momento é formado principalmente por ferro, colapsa em menos de um segundo, superando temperaturas da ordem de $5,0 \cdot 10^9$ K o que gera uma grande e luminosa explosão, que denomina-se como supernova.

Sob a luz da cosmologia física, que estuda as grandes estruturas e a dinâmica do universo, encontra-se a hipótese do que é conhecido como "energia escura", que desde a década de 1990 vem sendo utilizada para descrever e interpretar determinados fenômenos observados, que corroboram com a ideia de que o universo está se expandindo a uma determinada aceleração. Evidentemente, por conta das dimensões, essas observações e medições devem ser imprescindivelmente precisas para compreendermos como e com que taxa o universo se expande e evolui no tempo. Da relatividade geral, sabemos que a evolução dessa taxa de expansão pode ser parametrizada em termos da equação cosmológica de estado, que busca por relacionar as grandezas de temperatura, densidade de energia, pressão e matéria combinada no vácuo em um ponto qualquer do universo.

Albert Einstein (1879 - 1955), em meados de 1915, conclui a formulação da Teoria da Relatividade Geral, explicitando a relação entre um conteúdo de matéria-energia e a geometria do espaço-tempo. Naquele período, por não se ter evidências e dados experimentais que corroborassem com o contrário, as crenças da comunidade científica iam em direção a um universo estático. Nesse sentido, o mesmo físico teórico buscou por determinar soluções para suas equações, que levassem em conta esse caráter estático pressuposto para o universo. De início, esse movimento não foi possível, pois em um universo dotado de matéria com pressão nula e densidade positiva (tal qual a matéria ordinária, conhecida até então), o colapso gravitacional seria iminente ou, então, o universo iria expandir-se incessavelmente. Em ambas as soluções, não teríamos um universo estático. No intuito de obter soluções estáticas, Einstein introduziu em suas equações uma constante cosmológica para suprir as características desejadas, mas com o surgimento de evidências que apontaram para um universo em expansão (descoberta do *redshift* de galáxias), a ideia de introduzir a mencionada constante foi deixada de lado, pelo menos temporariamente, já que na década de 1920, Friedman (1888 - 1925) e seus colaboradores, retomam as equações originais de Einstein, propondo soluções que levavam em conta um universo que era uniformemente dotado de matéria e radiação, podendo ser homogêneo e isotrópico.

As soluções encontradas para tais equações se relacionam com a constante cosmológica proposta por Einstein para satisfazer os critérios de um universo em expansão acelerada, unindo a teoria com os dados observados. Como menciona [Graef. 2012], com o avanço de técnicas observacionais, como a da radiação cósmica de fundo e medidas de supernovas, por exemplo, obtém-se fortes indícios de que o universo é plano, que se expande aceleradamente e que possui uma densidade crítica de $\approx 10^{-47} GeV^4$. Todavia, ao se mensurar essa densidade, observa-se que apenas 30% é em relação a matéria, sendo 25% fruto de matéria escura e 5% de matéria bariônica. Para que a expansão seja acelerada, foi necessário introduzir a constante cosmológica (que se comporta com um efeito repulsivo, tal qual um fluido com pressão negativa). O que existe, então, é a possibilidade desses 70% de densidade faltante estarem relacionados ao mesmo componente que é responsável pela expansão acelerada do universo. A este componente, nos referimos, atualmente, como energia escura. Então, quando calcula-se a equação de estado para o que chamamos de matéria escura, nos deparamos com um dos grandes desafios da ciência moderna e da cosmologia observacional, que é o de compreender características como o comportamento e natureza da energia escura. Existem estudos, também, que buscam por associar a constante cosmológica a energia associada ao vácuo.

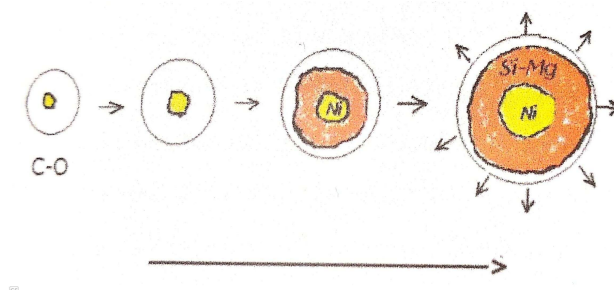


Figura 2 – Esquema que representa o resultado da fusão descontrolada do carbono, que produz níquel e também outros elementos como silício e magnésio

Fonte: [Horvath 2008].

Nesse cenário, temos que as supernovas são úteis pois podem ser utilizadas como referência a respeito de intensidades luminosas (a níveis astronômicos) quando se sabe qual o seu brilho intrínseco, tal como a vela padrão é pra nós na terra como um objeto com intensidade luminosa conhecida. Um tipo especial de supernova são as termonucleares, conhecidas como Ia, que são uma subcategoria das estrelas variáveis luminosas azuis. Na figura 2 vemos uma ilustração deste tipo de supernova. Estas, são tomadas como as velas padrão (intensidade conhecida) para as distâncias astronômicas, uma vez que sua luminosidade é muito forte e concisa. Por sua vez, ao olharmos para as relações entre a distância desses objetos e o seu *redshift* (que é a ocorrência da mudança do comprimento de onda da luz de um objeto para um comprimento de onda ainda maior), através de análises espectroscópicas, podemos investigar a história de sua expansão, e, conseqüentemente, mensurá-la. Então, como tais fenômenos são extremamente brilhantes, é possível observá-

los a grandes distâncias. Em astrofísica, observar objetos muito distantes implica em observar esses objetos no passado, dado o tempo necessário para que a luz percorra essa grande distância. Desse modo, as pesquisas em cosmologia física e astronomia buscam por encontrar um número cada vez maior de amostras de supernovas para investigar suas peculiaridades e características. Para que então ocorra a observação de supernovas, faz-se necessário investigar maneiras de se encontrar a ocorrência desse fenômeno, o que é um trabalho árduo haja visto que estes são eventos raros e que tem-se, literalmente, um universo a ser vasculhado. Estima-se, como explana [Horvath 2008], que a taxa de ocorrência de tal fenômeno, na Via Láctea, é da ordem de uma explosão a cada século. Na figura 3 observa-se o remanescente da supernova conhecida como Cas A. As cores observadas indicam diferentes elementos misturados pela explosão. Ainda, tal remanescente tem as características de uma supernova do tipo Ia, sendo originada da explosão de uma estrela do tipo anã branca (não massiva).

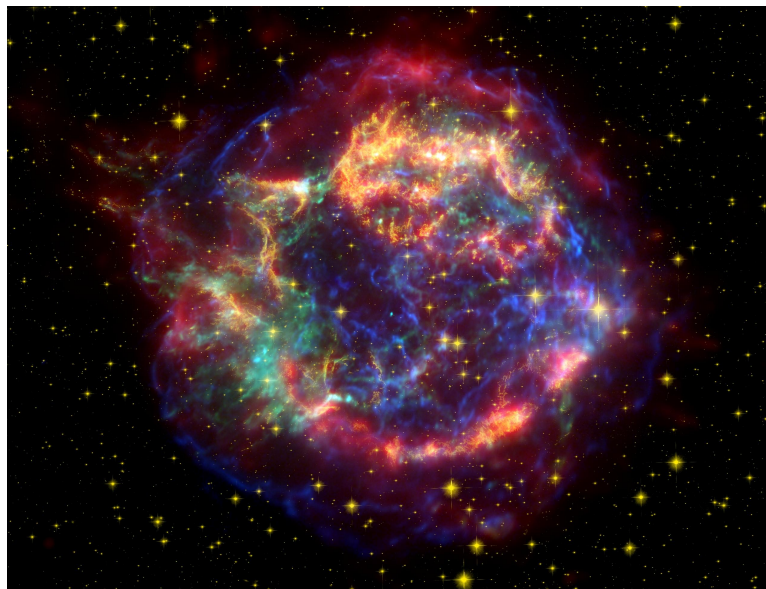


Figura 3 – Supernova remanescente Cas A, associada a uma explosão observada em 1680 da qual se tem poucos registros
Fonte: [Horvath 2008]

Com o advento da tecnologia aliado a sede de conhecimento sobre o cosmos, construiu-se ao longo das últimas décadas poderosos telescópios e observatórios que realizam diferentes mensurações em diferentes regiões do universo e, como consequência, a quantidade de dados a serem analisados cresce cada vez mais, demandando técnicas e máquinas cada vez mais avançadas para realizar tais processamentos. Na modernidade, a astronomia encontra-se em um novo momento no que diz respeito ao armazenamento de dados massivos. Uma das fontes de grandes quantidades de dados são, por exemplo, instalações de telescópios de grande escala, tal qual o *Large Synoptic Survey Telescope* (LSST). Este, que é um telescópio de 8,4 m com uma câmera de 3,2 Gigapixel, está previsto para iniciar suas atividades no final de 2022 e se encontra ao norte do Chile. O LSST é

um telescópio que irá escanear todo o céu do hemisfério sul com periodicidade de 3 dias, podendo obter dados de cerca de 50 bilhões de objetos celestes por década [Darin 2019].

O *High Cadence Transient Survey* (HiTS) é uma metodologia que teve por objetivo detectar e acompanhar transientes ópticos com escalas de tempo características de horas e até dias, mas com enfoque especial nas primeiras horas de explosões de supernovas, com o objetivo de detectar supernovas em seus estágios iniciais para estudar a astrofísica associada a esses fenômenos. O HiTS possui um *pipeline* (uma forma de canalização) personalizado para processar as imagens capturadas pelo telescópio e detectar transientes. Basicamente, o *pipeline* subtrai imagens de referência de novas imagens, detecta fontes e as classifica.

Na sequência, investiga-se a utilização do aprendizado de máquina para a detecção de supernovas em bancos de dados astronômicos, elucidando as contribuições de [Reyes et al. 2018], que propõe melhorias para o modelo *HiTS* ao utilizar redes neurais convolucionais e suas simetrias. Em [Cabrera-Vives et al. 2016] introduziu-se uma rede neural convolucional para classificar as fontes detectadas pelo *pipeline* HiTS como transientes verdadeiros ('candidatos a supernovas') ou falsos ('artefatos'). Depois disso, o modelo foi aprimorado com a introdução de invariâncias de rotações parciais, melhoramento da arquitetura e treinamento de algoritmos para produzir o chamado modelo *Deep-HiTS* presente em [Cabrera-Vives et al. 2017].

3.2 REDES NEURAIAS

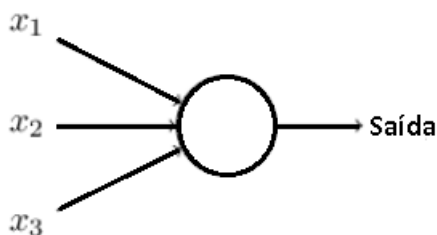
Nas últimas décadas a pesquisa e utilização da aprendizagem de máquina (*Machine Learning*) tem se tornado cada vez mais consolidadas. A realidade a qual estamos imersos tem se transformado rapidamente pelo emergente desenvolvimento tecnológico que nossa geração vive. Como fruto disso, tornou-se corriqueiro termos a nossa volta, por exemplo, a presença de celulares capazes de realizar um reconhecimento facial para liberar seu acesso, carros autônomos, softwares de sinalização de *Spam*, sistemas de recomendação de produtos/mercadorias e também marcação em fotos em redes sociais (automaticamente). Evidentemente, a ciência de modo geral não se faz indiferente frente a revolução que a tempestade de informações trazidas pelo *Machine Learning* promove e, por consequência, aduba o espaço para o surgimento de pesquisas, livros, artigos, palestras, conferências e empresas interessados em tal objeto. De acordo com o [Deep Learning Book 2022], o que o *Machine Learning* usa são algoritmos capazes de retirar informações de bancos de dados brutos e expressar tais informações via modelagem matemática, onde esta atua como uma facilitadora da compreensão de um conjunto muito numeroso de dados. Desenvolvido tal modelo matemático, pode-se aplicá-lo num novo conjunto de dados, avaliando as mesmas características de outrora, mas com novos valores. Nesse sentido, estamos realizando um

teste, que verifica a qualidade da tomada de decisões do modelo desenvolvido, podendo elaborar parâmetros que ajustem (ou calibrem) essa tomada de decisão na direção desejada. Vejamos com mais detalhes o que são esses parâmetros e demais objetos que compõem uma rede neural a seguir.

Entre os exemplos de algoritmos, temos os manuais de instruções de produtos, procedimentos experimentais listados em um roteiro didático de uma experimentação e receitas de bolo, que expressam uma sequência de ações e decisões lógicas que devem ser seguidas para se chegar em um determinado objetivo. Um exemplo mais elaborado de algoritmo é conhecido como Rede Neural Artificial, ou, simplesmente, Redes Neurais. Os estudos iniciais desses objetos datam de meados de 1950. Mesmo com a evolução do hardware com o passar dos anos, como se observa em [Deep Learning Book 2022], complicações envolvendo o volume de dados e como processar essa grande quantidade de informações utilizando *Machine Learning*, foram demais para as CPUs disponíveis antigamente. Mas nas últimas décadas, principalmente graças as empresas desenvolvedoras de vídeo-games e computadores, sedentas pelos melhores gráficos possíveis e força computacional, surgiram as Unidades de Processamento Gráfico (GPU), que, utilizando-se de programação paralela com essas GPUs, pode-se desenvolver simultânea e paralelamente, operações matemáticas envolvendo matrizes e vetores, que são personagens ativos nos modelos de Redes Neurais. Nesses anos de evolução científica, construiu-se um arsenal para analisar bancos de dados brutos e densos. Assim, tomando a capacidade de armazenamento poderosa que temos atualmente (*Big Data*) e somando-a com o processamento em paralelo citado, e, ainda, aliando estes a modelagem matemática presente no *Machine Learning*, temos o resultado que ficou conhecido como inteligência artificial (I.A.).

Na construção desse enredo, um dos principais objetos numa rede neural é o que podemos chamar de neurônios (na perspectiva matemática) ou, simplesmente, de nós. Os neurônios matemáticos são embasados nos biológicos, desde suas conexões e relações entre eles até o modo como essas comunicações evoluem temporalmente com um dado treinamento. De modo geral, entre 1980 e o início dos anos 90 ocorreram grandes avanços no que diz respeito a construção e modelagem das redes neurais, mas foi após a virada do século que o ferramental tecnológico e computacional passou a ter relevância para o mercado mundial, demandando a produção destes e, por consequência, o desenvolvimento de técnicas computacionais. Vamos inicialmente trabalhar com um modelo arquitetônico mais simples de rede neural, chamado de *Perceptron*. Essa arquitetura, projetada por Frank Rosenblatt (1928-1971), foi construída entre 1950 e 1960, e, mesmo não sendo muito utilizada atualmente, é um bom exemplo para visualizarmos como uma rede neural funciona.

Basicamente, um modelo de rede neural *Perceptron* é uma modelagem matemática capaz de receber informações em sua entrada e produzir uma única resposta em sua saída

Figura 4 – Neurônio *Perceptron*.

Fonte: Autor. 2022.

do tipo binária (sim ou não, por exemplo). Podemos observar um exemplo desse tipo de neurônio na figura 4, onde temos 3 informações de entrada, x_1 , x_2 e x_3 e uma informação de saída. Para se determinar a saída, utiliza-se a ideia de pesos w_1 , w_2 e w_3 , que são números reais que atribuem a importância que cada entrada tem para a saída. Nesse sentido, a saída do neurônio, que será 0 ou 1, pode ser expressa como

$$Saída = \sum_j w_j x_j, \quad (3.1)$$

sendo avaliada como maior ou menor que o valor de um parâmetro chamado de limiar (*threshold*), onde, assim como os pesos, o limiar é um número real que também acaba por ser um parâmetro para tal neurônio. Assim, matematicamente, teremos que

$$Saída = \begin{cases} 0 & \text{se } \sum_j w_j x_j \leq Limiar \\ 1 & \text{se } \sum_j w_j x_j > Limiar. \end{cases} \quad (3.2)$$

Vejamos um exemplo de seu uso. Imagine que no final desse mês haverá uma exposição de carros antigos e você, que está sem carro e procurando um antigo para comprar, está decidindo se irá ao evento e, para tal tomada decisão, leva em conta três fatores que lhe são relevantes: as condições climáticas, se a pessoa com quem você é casado(a) irá junto e se o centro de exposições é próximo ao ponto de ônibus. Esses fatores são representados pelas variáveis binárias x_1 , x_2 e x_3 . Nessa direção, se o tempo estiver bom teremos que $x_1 = 1$ e, se estiver ruim, que $x_1 = 0$. Analogamente, se sua companhia for junto, teremos que $x_2 = 1$ e, caso o contrário, que $x_2 = 0$. E o mesmo se aplica para x_3 . Suponha que sua paixão por carros antigos e a necessidade de se ter um veículo são consideráveis e você decide ir mesmo sem companhia e sem um ponto de ônibus próximo. Porém, se o tempo estiver fechado, você não será capaz de ir até lá. Construído esse enredo, você pode utilizar de uma rede tipo *Perceptron* para, com uma modelagem matemática, fazer a escolha de ir ou não ao evento. Uma maneira possível de fazer esse movimento, é assumir que $w_1 = 6$ para as condições climáticas e escolher que o peso valha 2 para w_2 e w_3 , de tal modo que o maior valor de w_1 quer dizer que as condições climáticas são realmente muito importantes nessa escolha, mais do que se terá companhia e se terá um ponto de ônibus próximo. Então, se adotar que o Limiar do neurônio da rede *Perceptron* é

5 e incluir esses parâmetros, o neurônio vai aplicar o modelo de escolha de decisões desejado, informando 1 quando o clima estiver limpo e 0 quando o clima for chuvoso. O que se tem é que, com esses parâmetros, não interfere no resultado se terá companhia ou não, bem como, se tem ou não ponto de ônibus próximo. Aqui, nota-se que oscilando-se o valor dos pesos e do limiar, é possível formular diferentes modelagens para a tomada de decisões. Se tomar o limiar com um valor igual a 3, o *Perceptron* vai decidir que você deve ir a exposição sempre que o clima estiver bom ou quando sua companhia for junto ou quando tiver ponto de ônibus próximo ao local, que seria uma modelagem diferente da primeira. Então, sabemos que, diminuir o limiar, implica na rede *Perceptron* tender a escolher a decisão de ir a exposição. Então, o modelo de rede *Perceptron* pode pesar tipos de evidências diferentes para tomar as decisões.

Vimos no exemplo acima como funciona uma rede *Perceptron* e como ela pode tomar decisões. Se montarmos uma rede mais complexa com vários neurônios, poderemos, então, resolver problemas mais complexos que envolvem escolhas mais sutis.

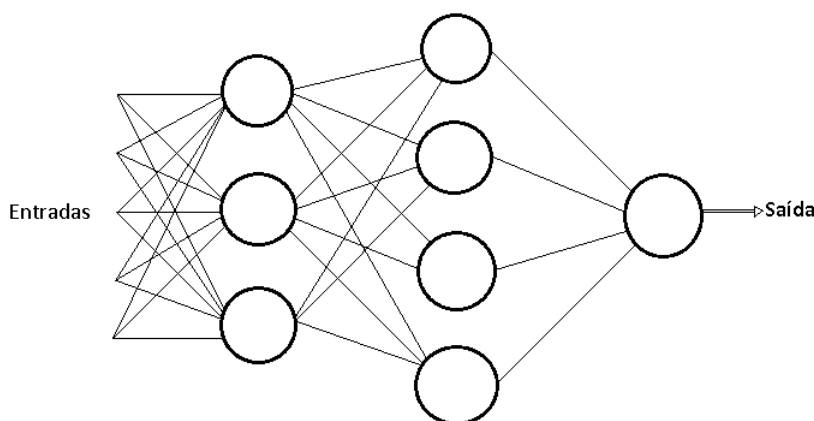


Figura 5 – Rede neural com muitos neurônios do tipo *Perceptron*.

Fonte: Autor. 2022.

Na figura 5, vemos a arquitetura de uma rede neural *Perceptron* maior, onde as camadas à esquerda são as de entrada, que irão receber as informações brutas, ao centro podemos ter as camadas chamadas de ocultas e no fim da rede, a camada mais à direita, temos a camada de saída. A primeira camada toma 3 simples decisões e atribui pesos nas informações da entrada. Na camada intermediária (e podem ser várias), cada neurônio vai tomar uma decisão tendo como base de informação uma ponderação das respostas que a primeira camada irá fornecer, sendo tal decisão mais complexa de ser tomada do que as que a primeira camada realiza. Seguindo a linha de raciocínio, a terceira camada acaba por tomar as decisões ainda mais sutis e assim sucessivamente, tal que quanto maior o

número de camadas, dispostas e arquitetadas adequadamente, mais sofisticado será o nível de tomada de decisões.

Sabe-se que, no limite de condição proposto por $\sum_j w_j x_j > Limiar$, podemos reescrever primeiramente $\sum_j w_j x_j$ como sendo um produto escalar dado por $w \cdot x$ de modo que $w \cdot x \equiv \sum_j w_j x_j$, (com w e x sendo os vetores com componentes compostas, respectivamente, por pesos e entradas). Ainda no intuito de melhorar a notação, elenca-se um novo parâmetro chamado de viés ou *Bias* do *Perceptron*, expresso pela letra b , onde $b \equiv -Limiar$. Nessa lógica, podemos reescrever (2.2) como sendo

$$Saída = \begin{cases} 0 & \text{se } w \cdot x + b \leq 0 \\ 1 & \text{se } w \cdot x + b > 0. \end{cases} \quad (3.3)$$

Refletindo sobre o conceito proposto pela ideia de viés, tem-se que este pode ser visualizado como uma medida do quanto é fácil conseguir um neurônio capaz de reproduzir uma saída com valor 1. Dito noutras palavras, o viés é o quanto é fácil fazer o neurônio disparar (ser excitado). Logo, tem-se que quanto maior o valor do viés, mais fácil será para a rede *Perceptron* emitir um valor 1 na saída e a recíproca também é verdadeira. No que tange o treinamento de uma rede neural tipo *Perceptron*, tem-se que este se dá no processo de fazer com que o modelo aprenda quais são os melhores (ideais) valores de pesos e também de vieses. Nesse sentido, o que se faz é apresentar os dados de entrada e os possíveis dados de saída ao modelo, de modo que, neste movimento de treinamento, o modelo aprenderá quais são os parâmetros de peso e viés que melhor constroem a trajetória do caminho que leva até a saída previamente desejada.

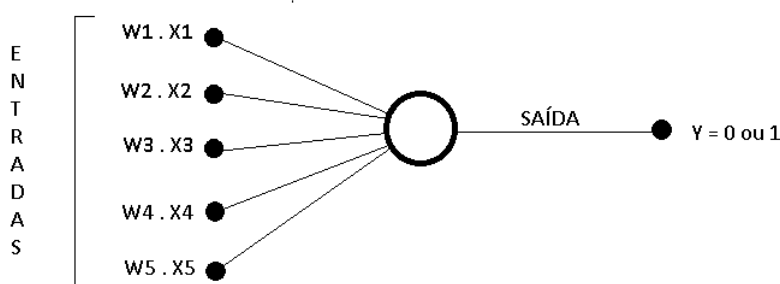


Figura 6 – Rede neural com apenas um neurônio *Perceptron* na camada oculta.

Fonte: Autor. 2022.

Se tivermos apenas uma camada oculta, como a da figura 6 temos que a rede neural *Perceptron* é uma rede neural de camada única que segue um modelo que ficou conhecido como *Feed-forward* que nos informa que as entradas serão encaminhadas para um neurônio, processadas por ele e, posteriormente, resultam uma informação na saída. Então, agrupando e arranjando várias camadas ocultas, tem-se uma rede neural profunda. Nesse sentido, o trabalho de uma rede neural, de modo resumido, gira em torno de operações

matemáticas que envolvem o conjunto de elementos de entrada (chamado de vetor de entrada) e o conjunto de valores de pesos de cada *Perceptron* (chamado de vetor peso). Então, pode-se adicionar outras camadas de caráter matemático ou estatístico que irão realizar operações distintas, dependendo do problema que se quer resolver utilizando a respectiva rede neural.

Um componente importante de uma rede neural artificial é conhecido por função de ativação. Para entender o que esse componente faz devemos refletir sobre como é possível elaborar algoritmos de aprendizagem para uma rede neural. Vamos supor que temos uma rede composta de *Perceptrons* e gostaríamos de usá-la para aprender a resolver um dado problema, por exemplo, analisar e identificar um dígito escrito a mão, onde as entradas da rede serão os dados do pixel dessa imagem após ela ser digitalizada. O que queremos é que nossa rede seja capaz de aprender os pesos e vieses adequados para que a saída da rede consiga classificar/identificar corretamente tal dígito. Então, para compreendermos como essa aprendizagem pode acontecer, vamos supor a seguinte situação: vamos fazer uma pequena alteração em algum peso ou viés da rede. A nossa intenção é que essa pequena alteração, seja no peso ou no viés, proporcione apenas uma alteração (também pequena) na saída da nossa rede. Como veremos a seguir, isso não acontece de maneira direta, mas é através desse processo que o movimento de aprendizagem torna-se possível. Vemos na figura 7 de maneira esquemática o que gostaríamos que acontecesse.

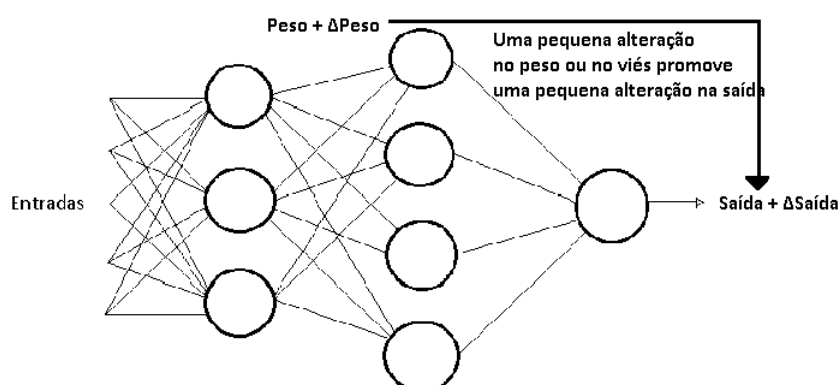


Figura 7 – Rede neural formada por neurônios *Perceptrons* com um acréscimo de peso/viés.

Fonte: Autor. 2022.

Porém, esse fato não é uma realidade. Imagine que nossa rede formada por *Perceptrons* está tendo um problema e acaba por identificar e reconhecer um dígito como sendo o número 6 quando na realidade é o número 5. O que pensamos em fazer logo de início é realizar uma pequena mudança nos vieses e pesos para que a rede consiga classificar corre-

tamente o dígito informado e fazer isso repetidas vezes até a rede entregar os resultados desejados. Todavia, não é isso que acontece numa rede formada por *Perceptrons*. O que acontece na realidade é que essa pequena alteração é capaz de alterar completamente o resultado expresso na saída, pois uma alteração no peso ou viés de um neurônio, mesmo que pequena, causara alterações em todo o restante da rede, podendo ficar muito complexo de resolver e controlar os problemas. Nesse cenário, uma maneira de contornar tal intempérie se dá através da introdução de um componente matemático chamado de função de ativação. Essas funções possibilitam que tais pequenas alterações nos vieses e pesos provoquem apenas as pequenas alterações na saída que desejamos na ideia inicial e é através desse movimento que uma rede neural é capaz de aprender. Em linhas gerais, são as funções de ativação que irão decidir se um neurônio irá ativar (ser excitado) ou não. Dito de outra forma, é como se essas funções avaliassem as informações que chegam na entrada do neurônio e decretam se esta é de fato relevante para a resposta que será informada na saída ou se ela deve ser desconsiderada. Com a nomenclatura adequada, temos que uma função de ativação é uma transformação (função) não linear diferenciável que é aplicada nos sinais de entrada produzindo uma saída também “transformada” que será enviada para a próxima camada como um sinal de entrada. Logo, quando não utilizamos uma função de ativação, nossos vieses e pesos realizam apenas transformações lineares, que são simples de serem resolvidas mas que é limitada pois não tem capacidade de resolver problemas mais complexos. Ainda, pode-se encarar uma rede neural que não conta com uma função de ativação como sendo um simples modelo de regressão linear ($AX = Y$, onde X é a imagem/dados de entrada, Y é a saída rotulada e A é o modelo, um vetor fruto de uma matriz que expressa um mapa linear cujos parâmetros devem ser determinados). Temos, então, que a intitulada função de ativação faz a transformação não linear nos dados do vetor de entrada, tornando a rede capaz de aprender e também aplicar decisões mais sofisticadas. O que ocorre é que a existência dessas funções permite a ocorrência do que é conhecido por propagação defasada (retro-propagação), se os gradientes forem informados com os erros que vão atualizar cada um dos vieses e pesos. De modo simplificado, essa retro-propagação se dá num algoritmo que emprega a regra da cadeia no sistema.

Assim como não existe apenas um tipo de arquitetura de rede neural, temos vários modelos de função de ativação e a escolha de qual delas utilizar varia conforme o problema a ser encarado. Entre as corriqueiramente utilizadas temos a função de etapa binária que é a mais simples por funcionar como uma espécie de qualificador binário, temos também a função linear que não depende do valor da entrada, ou seja, tem sua derivada constante, que é ideal para decisões mais simples. Ainda, temos a função de ativação conhecida como Sigmóide que é uma função suave (continuamente diferenciável) que vai tentar jogar os valores da saída para os extremos (0 ou 1) e que pode ter qualidades desejadas dependendo do problema a ser resolvido e, para encerrar os exemplos de função de ativação, temos a função *ReLU*. Essa função é definida como sendo a unidade linear retificada e é a função de

ativação mais utilizada em redes neurais hoje em dia. Levando em conta que ela é aplicada sob as camadas ocultas de uma rede neural e que ela não é uma função linear, temos dois fatos. Primeiro que pode-se copiar seus erros nas camadas anteriores com relativa facilidade e segundo que haverá várias camadas da rede ativadas pela função *ReLU*. Essa função ativadora tem a propriedade de não ativar todos os neurônios da camada simultaneamente. Chamada também de função de rampa, ela consegue realizar esse movimento pois toma os casos em que a entrada é um valor negativo e o identifica como zero de modo que o respectivo neurônio não será ativado, resultando numa facilitação do processamento (uma vez que o número de entradas a serem avaliadas é menor) e construindo uma arquitetura de neurônios mais espaçada. A função de ativação *ReLU* definida como a parte positiva do argumento de uma dada função, expressa como $f(x) = x^+ = \max(0, x)$.

Temos, então, que a arquitetura de uma rede neural pode ter formatos e organizações variáveis de acordo com a problemática a ser resolvida, onde, em geral, as camadas de entrada e saída tem um design direto, enquanto as camadas ocultas tem designs bem variados. Mas como via de regra, temos que a primeira camada é chamada de camada de entrada (ou simplesmente entrada) da rede neural e, analogamente, no final da rede tem-se a chamada a camada de saída (ou simplesmente saída) que pode ser apenas um neurônio ao invés de uma camada com vários neurônios. Já as camadas que se encontram entre a entrada e a saída, são conhecidas como camadas ocultas. Por exemplo, vamos supor que queremos que nossa rede neural aprenda a identificar o dígito 7 manuscrito. Para tal, pode-se, por exemplo, arquitetar uma rede neural que conta com 728 neurônios na entrada, 10 na camada de saída e uma camada oculta com 15 neurônios. Na sequência, temos que é necessário digitalizar essa imagem para que os algoritmos (neurônios) interpretem os *pixels* e as respectivas intensidades de cada pixel de tal imagem. Isso acontece pois podemos projetar nossa rede para interpretar, na camada de entrada, a intensidade luminosa de cada pixel que compõe a imagem digitalizada, de modo que, se a imagem estiver numa escala de cinza 28x28, teremos 728 neurônios na entrada com as intensidade podendo ser normalizadas entre 0 e 1.

Inicialmente os neurônios da camada de entrada da nossa rede realizarão o processo de codificar os *pixels* (dados a serem analisados) que estão em escala de cinza, onde 0 expressa a cor branca e 1 a cor preta. As imagens dos dígitos, após serem digitalizadas, podem, por exemplo, ter um tamanho de 28x28 *pixels*, o que implica em 728 neurônios, tendo na camada de entrada um neurônio para cada pixel a ser avaliado. Para melhor compreender o significado de uma “imagem” no referido contexto, devemos nos lembrar que interpretamos uma escalar como sendo um ponto adimensional, um vetor como um segmento de reta unidimensional e um ente matricial tal como um plano de duas dimensões. Ainda, podemos relacionar uma pilha de matrizes a um cubo com três dimensões e por fim, expressar uma quarta dimensão fazendo com que cada elemento membro dessas matrizes empilhadas tenham, também, uma pilha de mapas de *Features* (definiremos esse conceito

mais adiante). Por fim, temos nossa camada de saída, que contará com 10 neurônios, de modo que, se por exemplo o oitavo neurônio tiver uma saída que seja aproximadamente igual a 1, ou seja, se este neurônio for excitado, teremos, então, que a rede identificou que o dígito analisado se trata do número 7. Se o primeiro neurônio da camada de saída tivesse sido ativado, teríamos que a rede está nos informando que o dígito avaliado é o número zero. Se o décimo (último) neurônio da camada de saída tivesse sido excitado, teríamos que a rede identificou o número nove e assim se segue para os demais dígitos.

Identificar números escritos a mão, em geral, é uma tarefa fácil para um humano alfabetizado e com cérebro e olhos em perfeitas condições, tanto que ocorre de maneira quase que inconsciente. Porém, arquitetar uma estrutura de algoritmos capaz de realizar tal feito não é nada simples. Identificar, por intuição, que o número 8 pode ser visto como “uma circunferência em cima de outra circunferência”, por exemplo, é uma informação complexa de se transcrever para um algoritmo interpretar, pois cada pessoa possui a sua caligrafia com suas características. Então realizar essa identificação é algo complexo pois haverá incontáveis casos especiais e ressalvas para se considerar na hora do julgamento. O que ocorre é que as redes neurais encaram a problemática sob outra perspectiva, tomando uma gama espessa de caracteres manuscritos que são conhecidos por ela (chamados dados de treinamento) permitindo que ela desenvolva métodos sistemáticos para aprender inferindo as regras de como reconhecer esses manuscritos. Por obviedade, aumentando o número de dados de treinamento, a rede aprenderá cada vez mais, aprimorando sua precisão (se arquitetada corretamente). Isto posto, encerrando o enredo das camadas ocultas, não temos como reduzir a poucas palavras o processo de construção de seus designs em função de um conjunto de regras a serem seguidas. O que acontece, na realidade, é que os pesquisadores dessa área do conhecimento desenvolvem maneiras de investigar sobre como o design das camadas ocultas podem ser arquitetados para cada tipo de problema, o que culminou em espécimes de heurísticas de design, que nos auxilia a encontrar e reproduzir o comportamento ideal da nossa rede neural para resolver um determinado problema. Nesse sentido, o design escolhido para formar a arquitetura das redes com muitas camadas ocultas tem relevância considerável nos modelos de rede neural que desenvolvemos o que conhecemos como aprendizado profundo (*Deep Learning*).

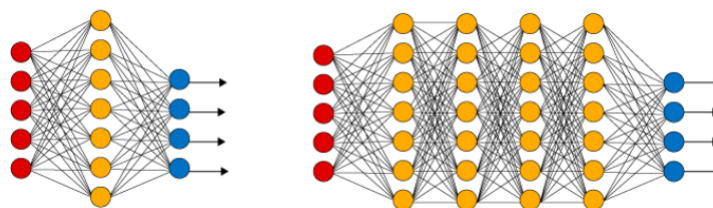


Figura 8 – Exemplo de redes neurais com uma e com várias camadas ocultas.

Fonte: [Deep Learning Book 2022]

O *Deep Learning*, que é uma ramificação do *Machine learning*, infere algoritmos

para realizar o processamento de uma gama espessa de dados, fazendo uso de camadas que são constituídas de neurônios matemáticos para realizar o processamento de dados, de modo que as informações percorrem cada camada. Aqui, cada camada é um algoritmo, em geral, simples e uniforme, que é ativado por uma função ativadora. Na figura 8 temos exemplos ilustrativos sobre a organização das camadas e suas conexões. Explicitando, os neurônios em vermelho expressam a camada de entrada, os amarelos a(s) camada(s) oculta(s) e, por fim, os azuis, que compõem a camada de saída. Nas camadas ocultas, a saída da camada anterior funcionará como uma entrada. Ainda na figura 8, vemos à esquerda uma rede neural simples, com apenas uma camada oculta, e à direita, temos uma rede neural de aprendizado profundo, com várias camadas ocultas que irão realizar operações diferentes. Uma característica interessante do *Deep Learning* é que ele é capaz de envolver *features*. No contexto dessa discussão que envolve o *Machine Learning* e, por sua vez, o reconhecimento de padrões, interpreta-se uma *feature* como sendo uma característica particular de um dado fenômeno que pode ser mensurada de modo que, como consequência, acaba por ser determinante a escolha de *features* que sejam informativas, independentes e distinguíveis (no sentido de tornar tal característica particular um fato discriminante) para que uma arquitetura de algoritmos consiga classificar e reconhecer padrões se baseando em tais características. Por esse fato, um conjunto de *features* é chamado, também, de relevâncias. A obtenção dessas *features* acontece com a utilização de um algoritmo que é capaz de “visualizar” automaticamente essa característica em meio ao conjunto de dados e, então, a julga como sendo significativa para os dados utilizados e destinados ao treinamento e ao aprendizado da rede neural profunda. Sendo assim, tem-se que as redes neurais profundas (*Deep Learning*) acabam por ser uma versão melhorada das redes neurais. Existem vários modelos para a estruturação de uma rede neural, que são diferentes pois foram projetados para resolver problemas também diferentes. Vale lembrar que nem todas as arquiteturas de redes neurais existentes são do tipo *Deep Learning*. Este último modelo tem por característica morfológica redes neurais com muitas camadas, como ilustra a figura 8.

Até o presente momento, elucidou-se redes neurais do tipo *Feed-Forward*, onde a saída de uma camada funciona como entrada para a camada subsequente. Noutros termos, tem-se a ideia de que aqui não existem *Loops* na rede, tal que as informações evoluem ao longo das camadas mas sempre seguindo para frente, nunca retrocedendo. Todavia, temos um tipo de modelagem para redes neurais onde esse *Looping* é permitido e são conhecidas como redes neurais recorrentes, onde os neurônios são excitados por um tempo finito. Essa excitação possivelmente vai excitar outros neurônios mas num período de tempo depois da primeira ativação. Nesse sentido, com o passar do tempo, teremos a formação de cascatas de excitações de neurônios, num processo análogo ao fenômeno de excitação de compostos luminescentes no processo de fluorescência, onde a emissão de radiação eletromagnética de tais compostos cessa imediatamente após a radiação incidente também

cessar-se. Logo, o que se tem é que a permissão do uso de *Loops* na rede não irá interferir negativamente no desempenho das redes neurais recorrentes, pois a saída de um dado neurônio irá afetar sua entrada apenas num momento ulterior, não imediatamente após ser excitado. E temos, também, as redes neurais que são conectadas de forma simétrica, que são semelhantes as redes recorrentes mas com as conexões tendo os mesmos pesos em todas as direções (proporcionando a simetria enunciada). Estas são algumas categorias específicas de possíveis arquiteturas de redes neurais. Entre essas três categorias gerais, temos uma arquitetura em especial que é conhecida como rede neural convolucional, a qual estudaremos e veremos exemplos de aplicações nas próximas seções.

Somando o poder de análise que temos hoje em dia com a fonte literalmente inesgotável de dados que é o nosso universo, temos um parque de diversões para os astrônomos da modernidade, que, com o advento da tecnologia, unem potentes telescópios com máquinas de alta capacidade de processamento, para analisar quantidades de dados impensáveis poucas décadas atrás. Nesse trajeto de desenvolvimento computacional que caminhamos até aqui, emergem as rede neurais convolucionais, que com o *Deep Learning*, constroem ferramentas, mecanismos e sistemas de inteligência artificial, tendo respaldo tanto na pesquisa quanto em setores comerciais e linhas de produção buscam por soluções de problemas com o uso da I.A.

3.3 REDES NEURAS CONVOLUCIONAIS

Já em meados de 1970 começavam-se os estudos de redes neurais com muitas camadas. Mas foi na metade dos anos 90 que [Lecun et al. 1998], enunciou essa temática modernista com maior riqueza de detalhes. Em linhas gerais, uma rede Rede Neural Convolucional (*Convolutional Neural Networks*), chamada também de CNN, é uma arquitetura especial de rede neural formada por muitas camadas de neurônios. Dito de outra forma, ela é um algoritmo de aprendizado profundo que é capaz de detectar uma imagem de entrada, atribuir certas “importâncias” para determinados aspectos e características dessa imagem e distingui-las umas das outras, de modo que essas importâncias/relevâncias são pesos e vieses que a rede pode aprender. A morfologia arquitetônica de uma CNN é semelhante aos padrões de conexão dos neurônios do cérebro humano, inspirada na estruturação do Córtex visual. Quando elencamos a ideia do que são redes neurais, vimos um exemplo de como ela pode reconhecer imagens de dígitos manuscritos, sob a luz do uso de redes em que camadas adjacentes são conectadas, por completo, umas nas outras, de modo que os neurônios da rede também estejam conectados a todos os neurônios nas camadas adjacentes. Temos, então, que as CNNs são uma modalidade de rede neural com certa especialidade pois conta com um conceito chamado de campo receptível local. Para compreender tal ideia devemos nos lembrar que, numa imagem digitalizada, cada pixel será um número que expressa qual a intensidade da cor, seja em escala de cinza para imagens em preto e branco, ou em

escala de RGB para imagens coloridas, por exemplo. Então o que se segue é que deve-se fazer a conexão dos neurônios da entrada com os neurônios da primeira camada oculta, mas diferentemente de outrora, os neurônios da entrada não são conectados a todos os neurônios da primeira camada oculta. O que acontece é que, a cada neurônio da camada oculta, conecta-se um conjunto de neurônios da entrada. Desse modo, cada neurônio da camada oculta estará conectado a uma “janela” de *pixels* da imagem de entrada. Por exemplo, podemos ter uma imagem de entrada composta por um quadro com 2500 *pixels* (50×50) e, nesta, escolher quadros com 5×5 *pixels*, que expressam uma região com 25 *pixels* para entrada, que são conectados a um único neurônio na camada oculta. Esse conjunto de *pixels* é chamado de campo receptível local. Na figura 9 vê-se um exemplo de campo receptível.

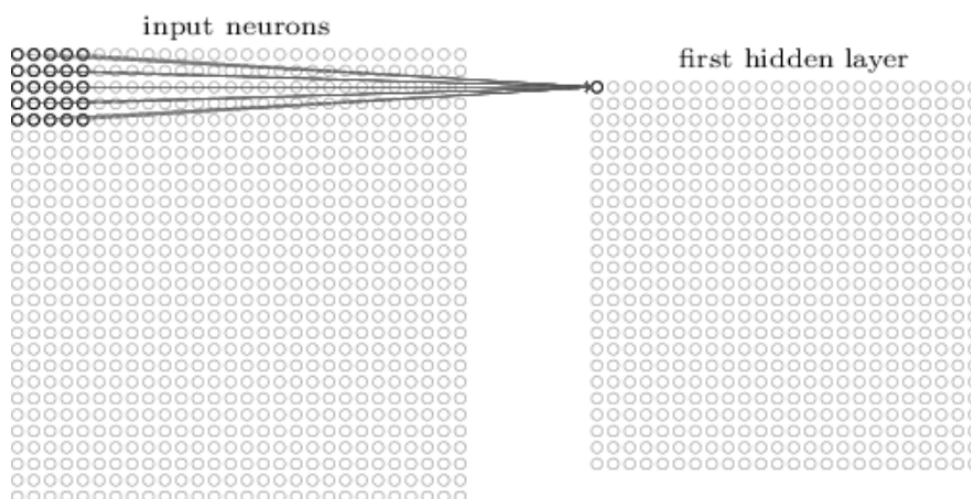


Figura 9 – Exemplo de campo receptível local com uma janela de 25 *pixels*.

Fonte: [Deep Learning Book 2022]

Com essa organização da rede, as conexões irão aprender os pesos ideais e o neurônio da camada oculta irá identificar qual o viés possivelmente ideal, o que culmina no fato de que cada neurônio da camada oculta irá aprender a avaliar especificamente um determinado campo receptível. Então, generalizando, podemos aplicar essa ideia de campo receptível local a toda a camada de neurônios de entrada, tendo um neurônio oculto para cada um desses campos. Podemos construir um campo receptível que translada para a direita movendo-se um pixel por vez (uma passada por vez), ou então, podemos realizar esse movimento aplicando um deslocamento de dois *pixels* ao invés de um, tendo deslocado um comprimento de duas passadas, que é um hiper-parâmetro conhecido por *stride length*. Como antes, o tamanho do *stride* bem como as dimensões do campo receptível local não são entes fixos, mas sim, manipuláveis, permitindo ser ajustados de acordo com a problemática a ser resolvida ou com a eficácia desejada. Por fim, a ideia é de se utilizar dos mesmos pesos e vieses para cada neurônio oculto da rede neural, num processo de compartilhamento das informações a respeito das relevâncias.

Uma convolução, matematicamente, se dá num operador linear que, ao ser aplicado em duas funções, retorna uma outra função. Esta terceira função irá computar o somatório do produto das duas primeiras, que resulta num modelo modificado de uma das duas primeiras funções. Dito de outra maneira, uma convolução se dá no ato de deslizar uma função sobre outra, aferindo-se a integral da soma dos produtos de tais funções, tendo como base a sobreposição das funções, fruto de deslizamento aplicado. Convoluções podem ser utilizadas para o processamento e análise de imagens, uma vez que tal operador conta com a característica de modificar a intensidade de um dado pixel para refletir as intensidades dos *pixels* que se encontram na vizinhança do pixel escolhido. Sendo assim, as convoluções são operações entre matrizes (imagem \times Kernel) que aletram a representação dos dados de modo a ser capaz de definir os filtros ideais a partir dos parâmetros do Kernel (peso, passada e/ou tamanho).

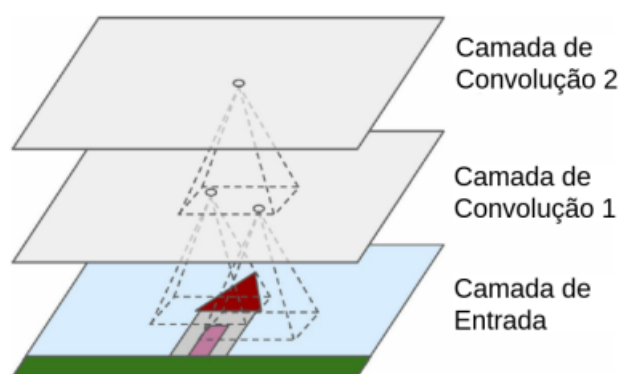


Figura 10 – Exemplo de camadas de convolução com campos receptíveis locais.

Fonte: [Géron. 2017.]

As CNNs possuem muitas partes que desempenham funções distintas. É costumeiro inferir nos dados de entrada camadas chamadas de convolução. Esta é constituída por muitos neurônios, de modo que cada um deles será destinado a aplicação de um filtro numa região da imagem. Isso ocorre fazendo a conexão de cada neurônio da camada de convolução com um campo receptível de *pixels* da camada anterior, tendo um peso específico sendo associado a cada conexão realizada, de modo que, ao combinar entradas de um neurônio, fazendo uso dos pesos que cada conexão gera, teremos uma saída que será encaminhada como uma entrada para a próxima camada. Estes pesos podem ser encarados como objetos matriciais que expressam o filtro de uma convolução, que avalia imagens no âmbito espacial (chamado também de *Kernel*).

Nas redes neurais do tipo *Feed-Forward*, por exemplo, as camadas de neurônios são totalmente conectadas (cada neurônio de uma camada é conectado a todos os neurônios da camada antecedente) mas nas rede neurais convolucionais somente um subgrupo de entradas é conectado a cada neurônio da próxima camada, de modo que, com essa alteração na arquitetura, a rede passa a analisar os campos receptivos locais. Nesse contexto, os neurônios de uma mesma camada serão unidos em mapas que são construídos através

Identidade	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Detecção de crista	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 4 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Afiado	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Desfoque de caixa (normalizado)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Desfoque gaussiano 3 x 3 (aproximação)	$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

Figura 11 – Exemplos da aplicação de filtros de convolução (Kernels) em uma imagem.
Fonte: [wikipedia. 2022.]

da união das saídas dos neurônios, abrangendo uma região da imagem que tenha sido processada por um filtro em comum. Nesse sentido, como explana [I. Goodfellow. Y. Bengio. and Courville, Aaron. 2016], diferente de arquiteturas tradicionais, as CNNs possuem uma menor quantidade de parâmetros para serem identificados e treinados, implicando em não precisar de longos períodos para treinamento ou precisar de máquinas extremamente poderosas. Ainda, como dentro de uma CNN existe a possibilidade de se compartilhar os parâmetros obtidos, a rede passa a ter a capacidade de aprender a identificar rotações de um dado objeto. Na figura 12 vemos um exemplo de estrutura de rede neural convolucional que busca por identificar um dígito manuscrito fazendo uso de camadas de convoluções e *pooling*, operação esta que será elucidada na seção 3.4.

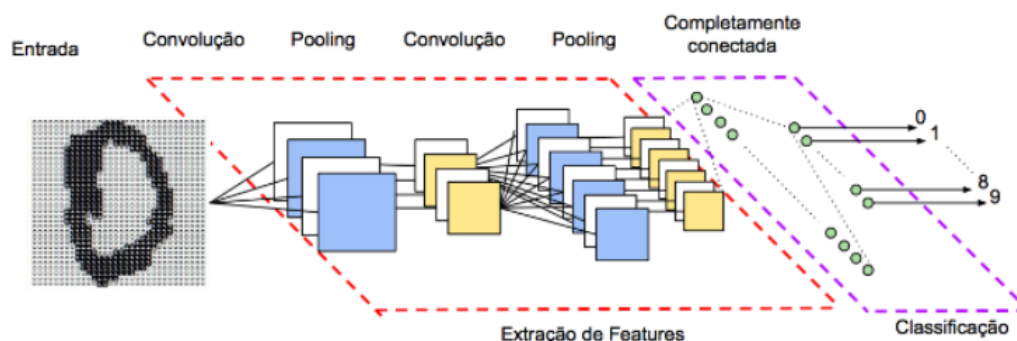


Figura 12 – Exemplo de rede neural convolucional e as diferentes camadas de operações para reconhecer um dígito manuscrito.

Fonte: [Vargas et. al. 2016.]

De modo geral, as camadas de convolução em uma rede neural convolucional tem a função de identificar os dados que realmente são significantes para a classificação de uma

imagem, reduzindo o número de dados a serem processados e otimizando o processamento, buscando uma forma mais fácil de se avaliar os dados da imagem, sem que ocorra a perda de *features* relevantes nesse processo de redução da imagem, que acaba por ser uma propriedade importante no que diz respeito ao ato de arquitetar estruturas de redes neurais que apresentam bons métodos de se aprender quando se trata de uma gama espessa de dados para avaliar. Por características como as supracitadas, as redes neurais convolucionais são uma das arquiteturas para *Deep Learning* mais poderosas com muitas possibilidades de aplicações. Entre as possíveis aplicações, investigaremos, aqui, um caso onde as redes neurais convolucionais são utilizadas para realizar a detecção de supernovas.

3.4 REDES NEURAS CONVOLUCIONAIS COMO INVARIANTES ROTACIONAIS

Elencou-se até aqui propriedades e características de redes neurais convolucionais e o conceito de supernovas. No contexto da detecção de supernovas, diversas ferramentas de *machine learning* tem sido aplicadas para a obtenção de melhores resultados de classificação, em especial as redes neurais, como observa-se nos trabalhos de [Qu et al. 2021], [Dieleman, Fauw e Kavukcuoglu 2016], [Dieleman, Willett e Dambre 2015], [Cohen e Welling 2016], [Cohen e Welling 2014] e [Cabrera-Vives et al. 2017], os quais exploram diferentes metodologias de redes neurais, trabalhos estes que foram investigados para construir uma visão geral sobre redes neurais convolucionais. Um caso particular de arquitetura de rede neural convolucional é observado em [Reyes et al. 2018], onde lança-se mão do uso de propriedades invariantes. Este trabalho é investigado a seguir, onde compara-se os resultados de duas metodologias aplicadas em um mesmo banco de dados.

3.4.1 INVARIÂNCIA ROTACIONAL

No escopo dos estudos da física teórica, um observável em um dado sistema físico é dito como invariante quando não altera-se a maneira como as leis físicas são interpretadas quando aplica-se certas transformações (rotação ou translação, por exemplo). É válido mencionar-se que se busca por metodologias de redes neurais que atuem com a premissa de serem capazes de perceber que as características principais de um objeto não se alteram á dadas transformações, pois as supernovas, que aqui serão os objetos a serem processados, são observáveis inscritos no Universo no qual vivemos. Sabe-se que as teorias cosmológicas corroboram com a hipótese chamada de Princípio Cosmológico, cujos resultados indicam que nosso universo é um sistema físico homogêneo e isotrópico. Holisticamente, para os fins aqui desejados, observa-se que para as escalas mais extremas do universo a homogeneidade tem como consequência uma distribuição de densidade média igual em todo o universo ao passo que da isotropia decorre um universo com a mesma aparência em todas as direções.

Unindo tais características para observações em larga escala, obtém-se como resultado um universo aproximadamente uniforme, onde não se tem direções ou regiões preferíveis entre outras para se realizar as observações. A estrutura que rege uma CNN convencional conta com camadas de convolução para obter a invariância em relação à translação, de modo que uma *feature* transladada para um posicionamento diferente na entrada, irá fornecer uma representação similar na saída. Tem-se, também, as operações chamadas de *pooling*, que buscam por obter a invariância translacional a nível local. Entre outros, um exemplo de conjunto de dados que apresenta as propriedades de invariância rotacional são imagens de galaxias e de estrelas. Na literatura, encontram-se alguns autores, em especial [Dieleman, Fauw e Kavukcuoglu 2016], que buscam por codificar de maneira direta a invariância presente nas estruturas de uma rede neural convolucional, no intuito de explorar as simetrias e gerar transformações cíclicas dos dados obtidos. Podemos definir essas transformações cíclicas a partir de rotações, no sentido anti-horário, $k \cdot 90^\circ$, para $k \in \{0, 1, 2, 3\}$.

Sob esta luz, a primeira operação será a de adicionar uma camada de corte cíclico na entrada, empilhando 4 “fatias”, rotacionadas ciclicamente de cada uma das amostras de entrada em um dado lote (ou canal). Desse modo, se o lote original inicialmente tiver um tamanho N , após passar por essa camada de cortes cíclicos, o novo tamanho desse lote será $4N$.

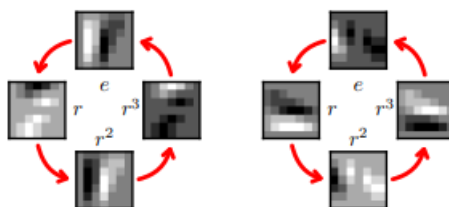


Figura 13 – Rotações de uma dada amostra.
Fonte: [Cohen e Welling 2014]

O que temos aqui é que as primeiras N amostras do lote estão relacionadas a imagem original e as amostras seguintes serão correspondentes as rotações de 90° , 180° e 270° respectivamente. Numa linguagem matemática mais refinada, podemos definir essa operação como

$$S(x) = [x, rx, r^2x, r^3x]^T \quad (3.4)$$

onde r representa a operação de rotação de 90° no sentido anti-horário e x o lote de entrada para a camada, de modo que com essa operação quatro imagens serão geradas e processadas independentemente pelas demais camadas do modelo. Assim, a camada de cortes cíclicos é utilizada por [Dieleman, Fauw e Kavukcuoglu 2016] para conseguir as CNNs invariantes sob a transformação de rotação para prever a morfologia de galaxias.

A segunda operação será uma camada de *pooling* que combinará as ativações das 4 cópias rotacionadas fazendo uso de uma função de permutação invariante. Respeitando o formalismo, o *cyclic pooling* é definido como uma operação sobre a entrada $x = [x_0, x_1, x_2, x_3]^T$ tal que $P(x) = p(x_0, r^{-1}x_1, r^{-2}x_2, r^{-3}x_3)$ em que p corresponde a operação de *pooling*.

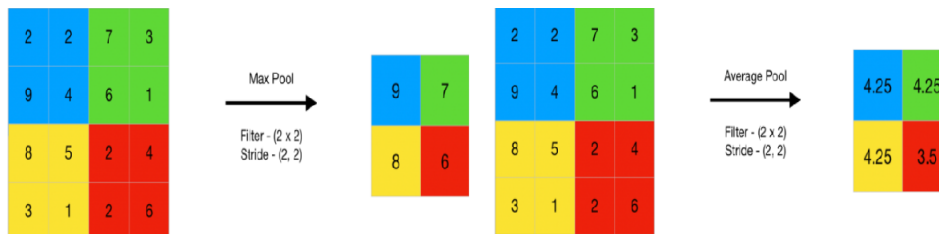


Figura 14 – Exemplos da operação pooling.

Fonte: Site geeks for geeks. 2022.

Na figura 14 vemos dois exemplos da operação *pooling*. Na primeira delas, à esquerda, temos o *max pooling*, que é um filtro que avalia os *pixels* de um dado campo receptível e identifica o de maior valor, assumindo este valor para todos os pixels membros de tal campo, resumindo toda a região delimitada pelo campo receptível á um único valor. Já à direita da figura 14, vemos um exemplo de *average pooling*, que, diferentemente do primeiro, realiza uma média entre os valores de cada pixel que compõe um campo receptível. No trabalho de [Reyes et al. 2018] o *cyclic pooling* é aplicado a cada novo lote não rotacionado. Desse modo, quando aplicamos um *cyclic pooling* depois da camada totalmente conectada não precisamos mais reordenar as covariáveis. Mesmo podendo introduzir uma camada de *pooling* em qualquer região do modelo após a camada de cortes cíclicos, é costumeiro introduzir esta antes da camada de saída, no intuito de obter uma rede que seja invariante sob a transformação de rotações.

3.4.2 VISUALIZANDO E COMPREENDENDO AS DECISÕES DE UMA CNN

Já faz algum tempo que métodos tem sido desenvolvidos com o objetivo de enxergar o que uma CNN aprende em uma dada tarefa de classificação. Um exemplo disso pode ser observado em [SIMONYAN, VEDALDI e ZISSERMAN 2013], onde se apresenta uma proposta para analisar a sensibilidade da rede, lançando mão do uso de cálculos de gradientes num processo conhecido por retro-programação. Este método proposto não da uma resposta direta e clara sobre o desempenho da rede, mas indica os elementos da entrada que devem ser alterados e em qual direção devemos fazer tais alterações, para que os mesmos pertençam (mais ou menos) a uma mesma classe determinada pela rede neural convolucional. Em todo caso, um modelo de classificação pode ser entendido como uma função que vai mapear uma série de entradas \mathbf{x} para um registro de saída $f(\mathbf{x})$ que será uma representação de um valor de confiança de uma determinada associação de classes. Ainda, operações de retro-propagação para o espaço de entrada são utilizadas para

interpretar a escolha tomada pela rede codificada nestes parâmetros. Essa estratégia irá construir um valor de relevância R_i para cada covariável (peso ou viés, por exemplo) de \mathbf{x} , de modo que essas relevâncias irão quantificar o quanto tais covariáveis contribuem para o valor informado na saída $f(\mathbf{x})$ (registro). O que tiramos disso é que valores positivos de R_i são encarados como covariáveis que contribuem para a decisão da rede, ao passo que valores negativos são interpretados como uma evidência contra a previsão, o que possui um potencial para diminuir o valor de $f(\mathbf{x})$. Nesse cenário, uma retro-propagação de relevâncias em camadas (LRP) irá retro-propagar a saída da rede, de camada em camada, buscando o espaço de entrada, obtendo os elementos da entrada que contribuem positiva ou negativamente para informação de classificação da saída. Essa retro-propagação de relevâncias é pautada num princípio de conservação que propõe que a soma dessas relevâncias $R_j^{(l+1)}$ em uma camada $(l + 1)$ deverá ser igual a soma das relevâncias $R_i^{(l)}$ na camada anterior l . Deste raciocínio, decorre o formalismo que expressa tal regra de conservação como

$$\sum_i R_{i \leftarrow j}^{(l)} = \sum_j R_j^{(l+1)}. \quad (3.5)$$

Existe uma série de operações matemáticas que podem ser desenvolvidas com o objetivo de aplicar a retro-propagação de $f(\mathbf{x})$ com camadas intermediárias que satisfazem a expressão (3.5), mas uma maneira simples é retro-propagar de forma proporcional em relação às ativações dos neurônios anteriores, tal qual um nó em uma corrente elétrica em um dado circuito, onde a presença desse nó faz a corrente se dividir proporcionalmente como sugerem a Leis de Kirchoff e Ohm. Nesse sentido, temos que

$$R_{i \leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{z_j} R_j^{(l+1)}. \quad (3.6)$$

Temos, então, que a relevância $R^{(l,l+1)}$ retro-propagada através do neurônio j na camada $l+1$ para o neurônio i situado na camada l , é equivalente a razão entre a ativação ponderada $z_{ij} = a_i w_{ij}$ do neurônio i , e a soma das ativações de *stride* que são provenientes camada l dada por $z_j = \sum_j z_{ij} + b_j$ para o neurônio j , adicionando-se mais um viés b_j . Mesmo que a adição desse viés quebre o princípio da conservação proposto pela equação (3.5), continua sendo uma aproximação razoável. Ainda, instabilidades numéricas conseguem ser superadas lançando-se mão da equação (3.7). Um exemplo de instabilidade é se $z_j \rightarrow 0$, pois, desse modo, $R_{i \leftarrow j} \rightarrow \infty$, e para evitar essa situação, é adicionado um hiper-parâmetro como um estabilizador, que estabelece uma regra, chamada de épsilon, tal que

$$R_{i \leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{z_j + \epsilon \operatorname{sign}(z_j)} R_j^{(l+1)}. \quad (3.7)$$

Uma outra operação matemática que também expressa uma regra de propagação, mas que não tem perda dessa relevância pra um estabilizador, é a regra conhecida por $\alpha\beta$. Esta, irá separar as ativações com pontuações positivas, os pesos e também os vieses z_{ij}^+ , das

com pontuações negativas z_{ij}^- da seguinte forma

$$R_{i \leftarrow j}^{(l,l+1)} = \begin{pmatrix} \alpha \frac{z_{ij}^+}{z_j^+} - \beta \frac{z_{ij}^-}{z_j^-} \end{pmatrix} R_j^{(l+1)}. \quad (3.8)$$

Em (3.8) deve ser verdadeiro que $\alpha - \beta = 1$ para que a regra de conservação seja satisfeita camada por camada, e, ao elencarmos esta, os valores usados para os parâmetros α e β definirão o modelo. Se utilizarmos $\alpha = 2$ e $\beta = 1$, por exemplo, o método será chamado de $LRP - \alpha_2\beta_1$. Ainda, a implementação de uma retro-propagação de relevâncias em camadas irá considerar camadas com operações de convolução, *drop-out*, *max pooling*, *average pooling*, múltiplas ativações e camadas totalmente conectadas.

3.4.3 INVARIÂNCIA ROTACIONAL DE CNNs PARA A DETECÇÃO DE SUPERNOVAS

Neste momento, apresenta-se a proposta elaborada por [Reyes et al. 2018] para aperfeiçoar o modelo de CNN elencado na metodologia *Deep-HiTs*, visando alcançar, entre outras peculiaridades, uma CNN que seja invariante perante às operações de rotação. Para tal feito, adicionou-se camadas de *cyclic average pooling* (CAP) antes da saída da rede. Para, então, comparar tais metodologias, utilizaram-se de testes de hipótese com os resultados obtidos em cada uma das metodologias.

As imagens (amostras) que compõem os dados de entrada utilizados, foram obtidos em um levantamento realizado em 2013 por meio da execução do *HiTs*. Neste conjunto de dados foram observados 40 campos celestes na banda *u* durante 4 noites por cerca de 2 horas. Em tal contexto, amostras negativas, chamadas de artefatos, são fruto de uma série de entidades que causam ruído em meio aos dados, como flutuações de fundo, astrometria ruim, a presença de outros objetos celestes que causam interferência e/ou *pixels* com deformidades. Na análise realizada, uma quantia de aproximadamente 800 mil amostras tiveram como resultado um valor negativo, e, para contornar o fato das supernovas serem eventos consideravelmente raros, fez-se necessário realizar um balanceamento no conjunto de dados, selecionando e incluindo dados positivos, chamados de candidatos, que foram simulados pautados em observações de objetos reais. Dessa forma, cada amostra será composta de 4 imagens com as dimensões de 21x21 empilhadas, onde cada camada dessa pilha representa uma *feature*, que é uma característica aferível e particular de certos objetos que compõem a imagem de entrada. Para entender esse conceito, façamos uma analogia. Se quiséssemos uma CNN que fosse capaz de identificar cubos em fotos digitalizadas, teríamos que arquitetar a rede de modo que ela fosse capaz de perceber os objetos cujas dimensões de altura, largura, e profundidade possuem o mesmo comprimento. Neste exemplo, a pilha (chamada também de mapa) de *features* teria três camadas, uma para cada dimensão avaliada. Pode-se fazer uma outra analogia semelhante para avaliar imagens coloridas,

onde a pilha de *features* seria composta pelas cores vermelho, verde e azul (RGB). No caso do trabalho investigado, tem-se que as quatro *features* utilizadas são, respectivamente, o *Template*, que é tomado como uma imagem de referência processada num instante de tempo $t_{inicial}$, a *Science*, que é a mesma amostra tomada como imagem de referência mas num instante de tempo futuro t_{final} , a *Difference*, que é a subtração entre o *Template* e a *Science*, evidenciando as mudanças espaciais que ocorrem em uma determinada região com o passar do tempo. Por fim, tem-se a última *feature* que é chamada de *Difference (SNR)*, que expressa a relação sinal-ruído durante as evoluções temporais e espaciais.

Das mais de 1.600.000 amostras de dados disponíveis, tomou-se uma quantia fixa de 1.200.000 para realizar o treinamento da rede, onde 100.000 dados foram utilizados para realizar a validação e outros 100.000 para fins de teste (estes dados são dispostos em plataformas públicas, tal qual o código *Deep-HiTs*). Com o melhor modelo obtido pelo treinamento e refinado via validação, realizou-se os testes. Ainda, para comparar tais heurísticas, lançou-se mão do uso das métricas de precisão, acurácia, *recall* e o registro de saída como medidas de desempenho.

Quando se trata de problemáticas que envolvem a classificação de dados, faz-se necessário a utilização de modelos de classificação, chamados também de métricas. Para construir tal cenário, é importante compreender certos conceitos, entre eles, a matriz de confusão. Este objeto faz o apontamento dos acertos e dos erros de um modelo, tomando como base de avaliação da qualidade a comparação dos erros/acertos com resultado que se espera.

		Detectada	
		Sim	Não
Real	Sim	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Figura 15 – Matriz de confusão

Fonte: [medium. 2022].

Na figura 15, observa-se uma matriz de confusão, onde a afirmação “verdadeiro positivo” indica uma classificação correta, “falso negativo” expressa que o modelo classificou como negativo um objeto de valor positivo, o que em estatística é tratado como erro tipo II. Tem-se, também, o “falso positivo”, enunciado como erro tipo I, que é o caso onde o modelo expressa uma classe como sendo positiva mas que, na realidade, é negativa e o “verdadeiro negativo” que é o caso em que o modelo prevê uma classificação negativa de um objeto que realmente tem valor negativo. Aqui, quando mensura-se esses termos e constrói-se a matriz de confusão, é possível computar as métricas de avaliação acerca da classificação.

MÉTRICAS	EQUAÇÃO
Acurácia	$\frac{VP + VN}{VP + VN + FP + FN}$
Precisão	$\frac{VP}{VP + FP}$
Recall	$\frac{VP}{VP + FN}$
F1-score	$2 * \frac{Precisão * Recall}{Precisão + Recall}$

Figura 16 – As quatro métricas utilizadas para comparar os modelos e suas respectivas equações.

Fonte: Adaptado do blog medium 2022.

A métrica conhecida por acurácia é atrelada a exatidão de um valor obtido quando comparado a um valor de referência e traz informações sobre a performance de modo geral de um dado modelo, indicando quantas classificações o modelo previu corretamente entre as positivas e negativas, (VP e VF na figurac16). Ainda, pode-se elencar uma mensuração como dotada de acurácia quando os valores obtidos são consideravelmente próximos do valor real (valor de referência). Já quando menciona-se sobre a precisão de um conjunto de dados, o que se tem é o quanto os dados coletados estão próximos uns dos outros. Tem-se, então, que quanto maior a precisão obtida numa mensuração, menor será a variabilidade entre os dados coletados (quanto maior a precisão, menor será a dispersão entre os pontos experimentais encontrados pelas sucessivas experimentações). Holisticamente, a precisão informa-nos, no conjunto de todas as classificações de caráter positivo, quantas o modelo classificou assertivamente.

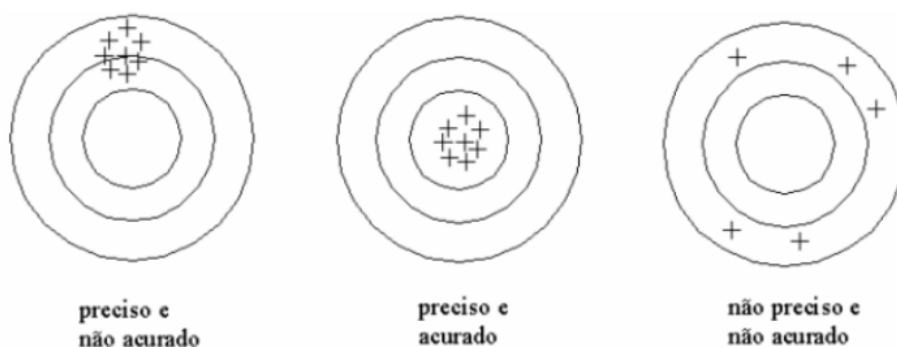


Figura 17 – Relações entre acurácia e precisão (ou não) em uma dada amostra.

Fonte: [CPE tecnologia. 2022.]

A métrica enunciada como *recall*, (atrelada á sensibilidade) expressa a relação entre as classificações positivas previstas corretamente e as classificações que são de fato positivas. Em outras palavras, ela nos demonstra o quanto o modelo classifica corretamente os dados positivos, sendo aplicada nos casos em que o objetivo é diminuir o número de falsos negativos. Por fim temos a *F1-score* como uma métrica que possibilita encarar as métricas de precisão e *recall* em conjunto, através do cálculo da chamada média harmônica

entre elas. Num caso aplicado, obter uma *F1-score* com valor baixo implica no fato de que ou a precisão ou a *recall* são também baixas.

Nesse cenário, foram testadas operações simétricas e cíclicas com o objetivo de implementar a invariância rotacional no *Deep-HiTs*, cujos resultados serão apresentados mais adiante para comparação. Então, construindo uma variação do *Deep-HiTs* que ficou conhecida como metodologia CAP, [Reyes et al. 2018] implementa um *cyclic average pooling* antes da camada de saída, o que significa que, ao invés de reordenar as covariáveis antes das camadas de saída (que é o que o *Deep-HiTs* faz), as *features* associadas a cada uma das rotações continuam a ser processadas de modo independente uma das outras, ou seja, é como se tivesse uma rede neural particular para avaliar cada uma das 4 rotações, que combinam seus resultados para expressar uma única informação na saída.

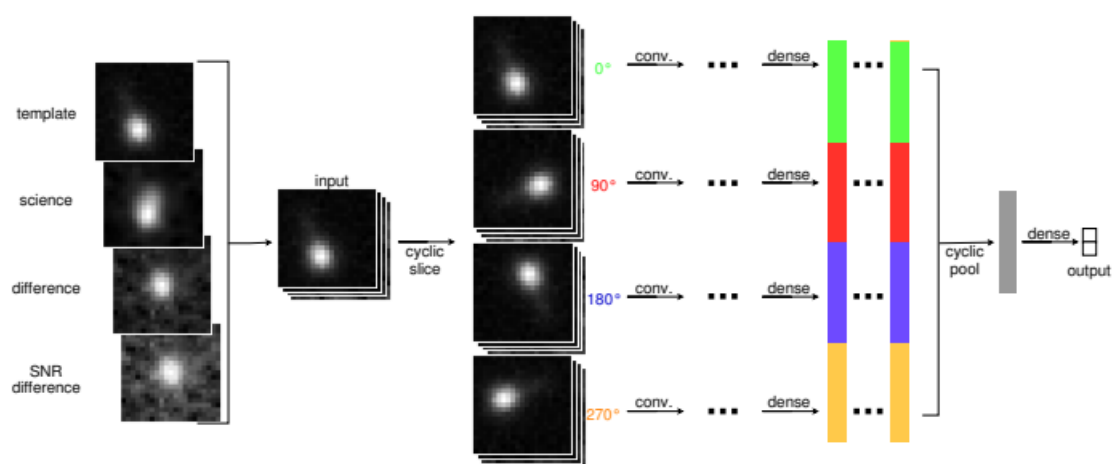


Figura 18 – Arquitetura utilizada pela modelagem CAP
Fonte: [Reyes et al. 2018].

Na figura 18 vemos a arquitetura do CAP. Ao lado esquerdo, temos as *features* associadas as imagens dos artefatos/candidatos e, na sequência, tem-se o empilhamento de tais *features* compondo a camada de entrada da CNN. A camada de entrada é composta por imagens com quadros de $21 \times 21 \times 4 = 1764$ *pixels*, onde 21×21 é o tamanho da matriz, e 4 o número de dimensões da matriz (associado ao número de *features*). Posteriormente, temos uma operação (*cyclic slice*) que vai prover a cópia das imagens rotacionadas em 90° , 180° e 270° , respectivamente. Com as cópias geradas, os dados obtidos são processados por duas camadas de convolução e, a partir desse momento, passam a ser processados por camadas de *max-pooling* e de convolução, sucessivas vezes. Ao chegarem na última camada de *max-pooling*, o tamanho da camada de saída desta é da ordem de $6 \times 6 \times 64 = 2304$ *pixels* para cada uma das rotações. Logo, para as 4 rotações, tem-se uma camada de saída com $4 \times 2304 = 9216$ *pixels* (por amostra!). Por fim os dados processados alcançam as duas últimas camadas (densas) compostas por 64 neurônios, concluindo com uma

camada *softmax* que irá fornecer a probabilidade de saída binária (0 para artefato e 1 para candidato). Dito de outra maneira, na figura 18, tem-se as quatro entradas de canais que passam pela camada de fatiamento cíclico, onde as rotações copiadas são empilhadas no decurso da dimensão do lote. Na sequência, essas rotações são processadas através das densas camadas convolucionais e de *pooling*, de maneira independente, até alcançarem a camada de *cyclic pooling*, onde todas as *features* de rotação, que são codificadas através das suas cores, são unidas de modo a serem analisadas pela camada de saída. Vale ressaltar que a operação CAP é aplicada antes da camada de saída da rede, onde, para cada amostra, as covariáveis provindas de cada rotação é calculada, e, dessa maneira, o registro de saída será sempre o mesmo, independente se há ou não rotações cíclicas na entrada. A proposta desse modelo é embasada no fato de que a detecção de supernovas deve ser independente de rotações angulares das amostras observadas (invariância rotacional).

Nesse escopo, para validar qualitativamente a conjectura de que o modelo CAP é invariável perante às transformações de rotação mais precisamente que a modelagem do *Deep-HiTs*, faz-se o cálculo do *desvio padrão médio de rotações cíclicas*, denotado por $\bar{\sigma}$, fazendo uso de aproximadamente cinco mil amostras de teste, onde essa medida do desvio é calculada por canal (relacionada a cada uma das quatro *features*) implicando em quatro valores diferentes para cada amostra. Logo, esses valores correspondem a cada um dos quatro canais e, para determinar $\bar{\sigma}$, primeiro deve-se determinar um canal c , bem como, sua i -ésima rotação, denotada por c_i para $i \in \{0, 1, 2, 3\}$. Por fim, calcula-se a variação do pixel para o seu respectivo mapa de calor h^{c_i} , fazendo

$$Var(h^{c_i}) = \frac{1}{N} \sum_{j=1}^N (h_j^{c_i} - \mu_j^c)^2, \quad (3.9)$$

com $h_j^{c_i}$ sendo o j -ésimo pixel e N o é número total de *pixels* do mapa de calor (neste caso, $N = 4$). Aqui, a variação é computada em relação ao mapa de calor médio por canal μ_j^c . Na última linha dos mapas de calor presentes na figura 20 da seção 4, onde se discute os resultados, encontram-se exemplos das médias dos mapas de calor tanto para o modelo *Deep-HiTs* quanto para o modelo CAP. Assim, calcula-se o $\bar{\sigma}$ determinando a raiz quadrada de $Var(h^{c_i})$ para cada rotação dos mapas de calor, para, por fim, calcular a média sobre as quatro rotações, tal que

$$\bar{\sigma} = \frac{1}{N_r} \sum_{i=1}^{N_r} \sqrt{Var(h^{c_i})}, \quad (3.10)$$

com $N_r = 4$ sendo o número de rotações na entrada da CNN.

Por fim, a estrutura da modelagem do CAP é idêntica em relação a do *Deep-HiTs* até a última camada convolucional, onde, nesse ponto, aplicasse as densas camadas de *cyclic average pooling*. Para facilitar a compreensão, pode-se imaginar que o que o *Deep-HiTs* faz é o que é ilustrado na primeira linha da figura 18 (sem a densa camada de *cyclic average*

pooling no final da rede, obviamente), que é o processamento dos dados sem nenhuma rotação aplicada.

4 RESULTADOS E DISCUSSÕES

Na figura 19 são elucidados os resultados obtidos pela metodologia do *Deep-HiTs*, elaborada por [Cabrera-Vives et al. 2017], e se compara com as metodologias CAP e CAPE propostas por [Reyes et al. 2018], onde apresentam-se as médias e os intervalos de confiança para 4 métricas. Todavia, para os fins previsto nestes trabalho, apresenta-se aqui a comparação e discussão dos resultados acerca do *Deep-HiTs* e CAP. Ainda, na mesma figura, apresentam-se tais resultados para as metodologias do *Deep-HiTs* e CAPE, permitindo realizar uma análise de tais modelagens. Para tal, avalia-se os resultados das médias e os respectivos desvios padrões, que são frutos de seis tentativas com iniciações diferentes para cada uma das métricas de acurácia, precisão, *recall* e o registro de saída (*F1-score*), para cada uma das três metodologias. Para comparar os resultados obtidos em relação a cada métrica para cada modelo, utilizou-se de um teste hipótese conhecido como teste t de Welch, que é uma adaptação do teste t de Student. Em geral esse tipo de teste de hipótese é aplicado para verificar a hipótese estatística de duas médias, com variâncias diferentes, serem iguais. O teste t de Welch funciona bem quando as amostras a serem analisadas são compostas por uma densa quantidade de dados, mesmo que estes não sejam distribuídos normalmente. O mencionado teste de hipótese também conta com a característica de funcionar com mais eficiência do que outros testes quando o número de elementos em cada amostra avaliada é diferente. Desse cenário estatístico, decorre o objeto chamado *p*-valor, que, holisticamente, é um indicador estatístico que serve como um termômetro para medir-se as diferenças entre as médias do CAP e do *Deep-HiTs* para cada métrica e, de acordo com o seu resultado, nos informará se tais diferenças são acidentais (fruto do acaso) ou se de fato essa diferença é fruto de uma melhoria em um dos modelos comparados, o que chama-se de uma diferença significativa. Nesse sentido, o *p*-valor será $< 0,01$ quando as diferenças entre as médias for significativa, ou seja, quando as médias são estatisticamente diferentes (isto é, aceita-se a validade da hipótese H_1) e quando *p*-valor $\geq 0,01$, poderá se considerar que as médias são estatisticamente iguais (em outras palavras, H_0 é válida), onde H_0 é a hipótese nula em que as médias do CAP e do *Deep-HiTs* são iguais e H_1 é a hipótese alternativa em que as médias dos modelos são significativamente diferentes.

O limiar do *p*-valor igual 10^{-2} nos diz que o modelo irá admitir 1 amostra com média fora do intervalo de confiança a cada cem amostras. Para, então, determinar se tal diferença é fruto de flutuações naturais da amostra ou se é uma diferença real (significativa), deve-se verificar se o resultado obtido para o *p*-valor se encontra dentro da região chamada de zona de rejeição de H_0 . O *p*-valor do teste de Welch sobre os resultados de CAP e *Deep-HiTs* expressa uma estatística significativa com valores menores que 10^{-2} (0,01),

exceto para a métrica *recall*. Em outras palavras, isso nos informa que os resultados ao aplicar o CAP são melhores quando comparados aos dos *Deep-HiTs*. Pode-se perceber, também, que aplicar o mesmo teste sobre CAP e CAPE (*cyclic average pooling ensemble*) não esboça resultados com estatística significativa, implicando no fato de que não tem grande distinção nos resultados ao utilizar um modelo ou outro e, por este fato, elucida-se apenas a comparação entre o *Deep-HiTs* e o CAP. Com essa elucidação, torna-se observável a melhoria dos resultados para as métricas utilizadas quando introduzimos a camada de *cyclic average pooling* em relação ao *Deep-HiTs*.

Modelos	Métricas			
	Acurácia	Precisão	Recall	F1-score
Deep-HiTs	99.45±0.02	99.37±0.04	99.55±0.06	99.45±0.02
Cyclic average Pool (CAP)	99.52±0.01	99.45±0.01	99.61±0.02	99.52±0.01
Cyclic average Pool ensemble (CAPE)	99.53±0.01	99.45±0.02	99.63±0.03	99.53±0.01
p-valor obtido com o teste de Welch (DH x CAP)	2.0×10^{-5}	4.2×10^{-3}	6.6×10^{-2}	2.2×10^{-5}
p-valor obtido com o teste de Welch (CAP x CAPE)	1.8×10^{-1}	8.9×10^{-1}	2.4×10^{-1}	1.8×10^{-1}

Figura 19 – Resultados obtidos pelas modelagens CAP e CAPE comparadas com o *Deep-HiTs* através do teste de hipótese de Welch para 4 métricas
Fonte: Adaptado de [Reyes et al. 2018].

Na figura 20, a título de exemplo, ilustra-se um caso onde, durante o teste, uma amostra intitulada SNR: 17.896 é prevista pela metodologia *Deep-HiTs*, à esquerda, como sendo um artefato, ao passo que a metodologia CAP, à direita, aponta essa amostra como sendo um exemplar de candidato a supernova (e que sabe-se que de fato esta o é, uma vez que as imagens aqui chamadas de candidatos à supernova expressam simulações pautadas em observações reais de supernovas). Para facilitar a visualização do que está acontecendo, [Reyes et al. 2018] propõe essa abordagem com mapas de calor que expressam, em vermelho, as regiões onde se concentram as relevâncias que vão contribuir positivamente para a rede predizer que uma amostra é um candidato a supernova e, em azul, regiões que se encontram as relevâncias que vão contribuir negativamente para a rede predizer que uma dada amostra é uma supernova. Avaliando tais mapas de calor, temos na primeira linha da figura 20 a amostra (em termos de suas *features*) que compõe a camada de entrada. Na segunda linha tem-se os mapas de calor da amostra, não rotacionada (0°), para cada uma das *features* avaliadas. Na terceira, quarta e quinta linhas tem-se os mapas de calor associados as *features* da amostra rotacionada em, respectivamente 90° , 180° e 270° em relação a primeira imagem, tal que $k \cdot 90^\circ$, $k = 0, 1, 2, 3$.

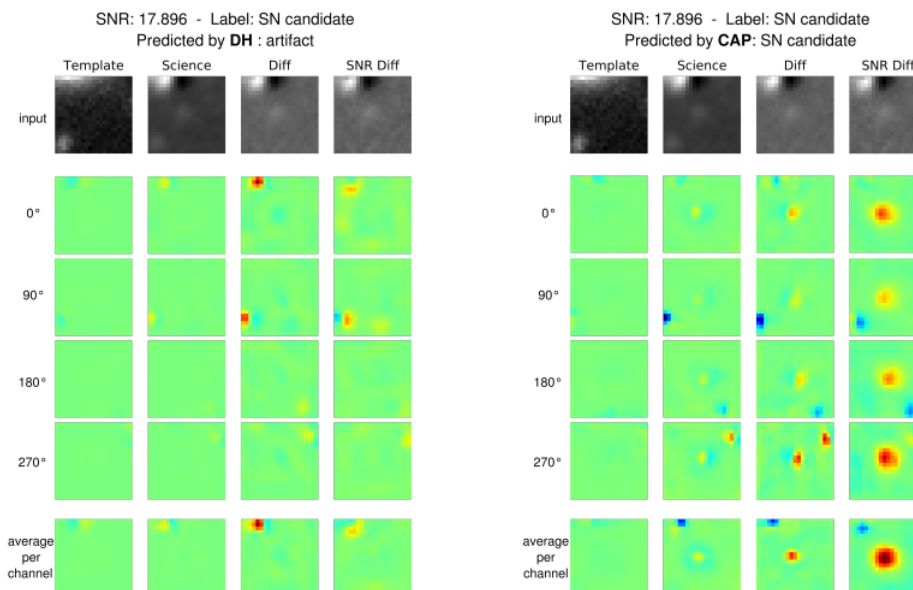


Figura 20 – Mapas de calor e suas rotações que expressam como as modelagens percebem quantidades diferentes de relevâncias para uma mesma amostra de dados.

Fonte: [Reyes et al. 2018].

Avaliemos, inicialmente, os mapas de calor para o *Deep-HiT*s (à esquerda). A tentativa aqui foi de aplicar rotações nas amostras para verificar se tal metodologia seria sensível a essas alterações. E o que se obtém é que o *Deep-HiT*s prevê que tal amostra se trata de um artefato. Já no lado direito da figura 20, vemos a mesma organização para os mapas de calor e para as rotações, mas para a modelagem do CAP, e o que se obtém é que tal metodologia percebe uma quantidade muito maior de relevâncias, principalmente as que contribuem positivamente com a decisão da rede em dizer que a amostra avaliada é um candidato. Em outro exemplo, o autor avalia uma amostra de dados de uma artefato e aponta que o *Deep-HiT*s prevê que tal objeto é uma supernova, enquanto o modelo CAP aponta para a previsão correta (que de fato tal objeto avaliado não é uma supernova), demonstrando que a metodologia CAP percebe melhor tanto as relevâncias negativas quanto as negativas para realizar suas previsões.

Observa-se, então, para o *Deep-HiT*s, que os mapas de calor demonstram que as relevâncias estão acumuladas no canto superior esquerdo da imagem não rotacionada associada a *feature Difference*. A interpretação desse fato é que a decisão do *Deep-HiT*s de predizer que se trata de uma amostra do tipo “artefato” é embasada pela fonte de luz observada naquela região da imagem *Difference*. Por outro lado, os mapas de calor para o CAP esboçam que as relevâncias se concentram no centro da imagem da *feature Difference* e da imagem *feature Difference (SNR)*, o que promove uma elucidação sobre a presença de uma supernova. Ainda, tem-se que a fonte de luz que pode ter confundido o modelo *Deep-HiT*s nesta amostra aparece em azul nos mapas de calor do CAP, informando-nos como relevâncias negativas, que indicam que a presença de fonte de luz, diminui o registro

de saída de previsão de uma supernova. Dentre as propriedades do modelo CAP que se observa, tem-se uma especial em que os mapas de calor que correspondem às rotações cíclicas são distribuídos de uma maneira mais uniforme, a qual pode-se observar e comparar, por exemplo, nas últimas colunas de mapas de calor da figura 20 para o *Deep-HiTs* e para CAP. Ainda, como corrobora [Reyes et al. 2018], distribuições mais uniformes de mapas de calor para as rotações cíclicas são uma indicação de que o modelo CAP é invariável em relação às rotações mais precisamente e, para verificar estatisticamente tal proposição, lança-se mão do cálculo do *desvio padrão médio de rotações cíclicas*, enunciado como $\bar{\sigma}$.

5 CONSIDERAÇÕES FINAIS

Com informações que constroem o arcabouço teórico deste movimento exploratório, respaldadas pela maneira como foram obtidas e posteriormente apresentadas neste trabalho, é válido afirmar-se que os objetivos foram alcançados, uma vez que o movimento de investigação da problemática possibilitou compreender: o que é e como ocorre uma supernova, os motivos de se estudar tais eventos, o conceito de rede neural e como esta pode ser utilizada para detectar as supernovas (são eventos raros) e a arquitetura de redes neurais convolucionais (CNN). Ainda, foi possível verificar como se obtém uma CNN ditas invariantes rotacionais e comparar os resultados desta com a modelagem que a precedeu, elucidando o caráter estatístico que se dá no momento de comparar-se tais metodologias

Tem-se, então, que a metodologia de aprendizado de máquina investigada (CAP) tem potencial para ser utilizada para realizar o processamento dos densos pacotes de dados que o LSST irá coletar após sua inicialização, sendo um objeto de pesquisa atual e com margem para novas possibilidades de investigação. É válido frisar-se, também, que tal modelagem de *machine learning* pode ser utilizada para outras finalidades como realizar a classificação morfológica de galáxias, descrita nos trabalhos de [Ivanishi e Izbicki 2017], propor estimativas acerca da irradiância solar, elucidado por [SILVA. 2021] e realizar abordagens de séries espectrais para regressões não paramétricas de alta dimensão [Lee e Izbicki 2017].

Como perspectiva para dar-se continuidade ao trabalho, pretende-se construir uma rede neural mais elementar, utilizando a plataforma *Tensorflow*, para elucidar, de maneira prática e didática, exemplos de como são construídas as linhas de algoritmos que compõem tal rede. Com esta finalidade, buscar-se-a por trazer a temática para a luz de discussões introdutórias a cerca do tema *Machine Learning* aplicado em física, uma vez que esta constrói seu arcabouço, por meio de sucessivas conjecturas e corroborações ao longo dos séculos, através de experimentações que geram uma infinidade de dados a serem avaliados. De modo geral, grandes quantidades de dados, desde que avaliados e obtidos adequadamente, geram resultados com intervalos de confiança razoavelmente aceitáveis. Haja visto que grandes volumes de dados necessitam de muito tempo para serem avaliados, tais técnicas computacionais podem ser elencadas para otimizar tal processo.

Referências

- C. Selltiz. , L. S. Wrightsman. and S. W. Cook. *Métodos de pesquisa das relações sociais*. [S.l.]: São Paulo: HERDER, 1965. Citado na página 12.
- CABRERA-VIVES, G. et al. Supernovae detection by using convolutional neural networks. *IJCNN*, p. 251–258, 2016. Citado na página 19.
- CABRERA-VIVES, G. et al. Deep-hits: Rotation invariant convolutional neural network for transient detection. *The Astrophysical Journal*, v. 836, n. 1, p. 97, 2017. Citado 3 vezes nas páginas 19, 33 e 43.
- COHEN, T.; WELLING, M. Learning the irreducible representations of commutative lie groups. *Proceedings of the 31st International Conference on Machine Learning*, v. 31, p. 1755–1763, 2014. Citado 2 vezes nas páginas 33 e 34.
- COHEN, T.; WELLING, M. Group equivariant convolutional networks. *International Conference on Machine Learning*, p. 2990–2999, 2016. Citado na página 33.
- DARIN, R. *The Planetary Society. Fundada por Carl Sagan , Louis Friedman e Bruce Murray em 1980. Utah, EUA: The planetary society, 2019*. [S.l.: s.n.], 2019. Citado na página 19.
- Deep Learning Book, D. *Data Science Academy. Deep Learning Book, 2022*. [S.l.: s.n.], 2022. Citado 4 vezes nas páginas 19, 20, 27 e 30.
- DIELEMAN, S.; FAUW, J. D.; KAVUKCUOGLU, K. Exploiting cyclic symmetry in convolutional neural networks. *International Conference on Machine Learning*, p. 1889–1898, 2016. Citado 2 vezes nas páginas 33 e 34.
- DIELEMAN, S.; WILLETT, K. W.; DAMBRE, J. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly notices of the royal astronomical society*, v. 450, n. 2, p. 1441–1459, 2015. Citado na página 33.
- GIL, A. C. *Métodos e técnicas de pesquisa social*. [S.l.]: 5.ed. São Paulo: Atlas,, 1999. Citado na página 13.
- GRAEF., L. l. *Um modelo para decaimento da energia escura*. [S.l.]: Dissertação (Mestrado em Física) - Instituto de Física, Universidade de São Paulo, São Paulo., 2012. Citado na página 17.
- HORVATH, J. *O ABCD da Astronomia e Astrofísica*. [S.l.]: LIVRARIA DA FISICA, 2008. ISBN 9788578610050. Citado 5 vezes nas páginas 10, 14, 15, 17 e 18.
- I. Goodfellow. Y. Bengio. and Courville, Aaron. *Deep Learning*. [S.l.]: MIT press., 2016. Citado 2 vezes nas páginas 11 e 32.
- IANISHI, P.; IZBICKI, R. Classificação morfológica de galáxias em conjuntos de dados desbalanceado. *Trends in Computational and Applied Mathematics*, 2017. Citado na página 47.

- LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, Nov 1998. ISSN 1558-2256. Citado na página 29.
- LEE, A. B.; IZBICKI, R. Spectral series approach to high-dimensional nonparametric regression. *Electronic Journal of Statistics*, p. 423–463, 2017. Citado na página 47.
- MATTAR., F. N. *Pesquisa de Marketing*. [S.l.]: 3.ed. São Paulo: Atlas,, 2001. Citado na página 13.
- QU, H. et al. Scone: Supernova classification with a convolutional neural network. *The Astronomical Journal*, v. 162, n. 67, 2021. Citado na página 33.
- REYES, E. et al. Enhanced rotational invariant convolutional neural network for supernovae detection. *IJCNN*, p. 1–8, 2018. Citado 12 vezes nas páginas 5, 6, 11, 19, 33, 35, 37, 40, 43, 44, 45 e 46.
- SILVA., O. o instituto federal do pernambuco. *Estimativa de irradiância solar utilizando redes neurais artificiais e imagens do céu*, 2021. Citado na página 47.
- SIMONYAN, K.; VEDALDI, A.; ZISSERMAN, A. **Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps**. *arXiv e-prints*, p. arXiv:1312.6034, dez. 2013. Citado na página 35.
- SOBRINHO., G. J. L. *Estrelas: origem, evolução e morte. Universidade da Madeira, Formação Contínua de docentes: Introdução à Astronomia: texto de apoio ao módulo 3, (17 pp.)*. [S.l.]: GAUMa, 2013. Citado 2 vezes nas páginas 14 e 15.
- W. G. ZIKMUND. *Business research methods*. [S.l.]: 5.ed. Fort Worth, TX: Dryden,, 2000. Citado na página 12.