

LUCAS GABRIEL COIMBRA EVANGELISTA

**REDES NEURAIS CONVOLUCIONAIS PROFUNDAS APLICADAS AO
DIAGNÓSTICO DE TUBERCULOSE A PARTIR DE IMAGENS DE RAIO-X**

Trabalho de Conclusão de Curso apresentado
à banca avaliadora do Curso de Engenharia
de Computação, da Escola Superior de
Tecnologia, da Universidade do Estado do
Amazonas, como pré-requisito para obtenção
do título de Engenheiro de Computação.

Orientador(a): Profa. Dra. Elloá Barreto Guedes da Costa

Manaus – Junho – 2019

Universidade do Estado do Amazonas - UEA
Escola Superior de Tecnologia - EST

Reitor:

Cleinaldo de Almeida Costa

Vice-Reitor:

Cleto Cavalcante de Souza Leal

Diretora da Escola Superior de Tecnologia:

Ingrid Gadelha

Coordenador do Curso de Engenharia de Computação:

Salvador Ramos Bernardino da Silva

Coordenador da Disciplina Projeto Final:

Manoel S. Santos Azevedo

Banca Avaliadora composta por:

Data da Defesa: 17/06/2019.

Profa. Dra. Elloá Barreto Guedes da Costa (Orientadora)

Prof. Carlos Maurício S. Figueiredo, D.Sc.

Prof. Marcos Filipe Alves Salame, M.Sc.

CIP – Catalogação na Publicação

L864a EVANGELISTA, Lucas Gabriel Coimbra Evangelista

Redes Neurais Convolucionais Profundas Aplicadas ao Diagnóstico de Tuberculose a partir de Imagens de Raio-X; [orientado por] Profa. Dra. Elloá Barreto Guedes da Costa – Manaus: UEA, 2019.

240 p.: il.; 30cm

Inclui Bibliografia

Trabalho de Conclusão de Curso (Graduação em Engenharia de Computação).

Universidade do Estado do Amazonas, 2019.

CDU: _____

LUCAS GABRIEL COIMBRA EVANGELISTA

**REDES NEURAIS CONVOLUCIONAIS PROFUNDAS APLICADAS AO
DIAGNÓSTICO DE TUBERCULOSE A PARTIR DE IMAGENS DE RAIO-X**

Trabalho de Conclusão de Curso apresentado
à banca avaliadora do Curso de Engenharia
de Computação, da Escola Superior de
Tecnologia, da Universidade do Estado do
Amazonas, como pré-requisito para obtenção
do título de Engenheiro de Computação.

Aprovado em: 17/06/2019

BANCA EXAMINADORA

Profa. Dra. Elloá Barreto Guedes da Costa

UNIVERSIDADE DO ESTADO DO AMAZONAS

Prof. Carlos Maurício S. Figueiredo, D.Sc.

UNIVERSIDADE DO ESTADO DO AMAZONAS

Prof. Marcos Filipe Alves Salame, M.Sc.

Embrapa Amazônia Ocidental

Resumo

Diagnosticar a tuberculose é crucial para um tratamento adequado, uma vez que essa doença é uma das dez principais causas de mortes no mundo. Considerando uma abordagem assistida por computador, este trabalho apresenta os resultados obtidos de uma estratégia de reconhecimento inteligente de tuberculose em radiografias de tórax com redes neurais convolucionais profundas. Para alcançar este resultado, foram consideradas 5 arquiteturas canônicas deste tipo de rede sujeitas à diversos parâmetros e hiperparâmetros, contabilizando 47 modelos treinados e testados para esta tarefa, além de quatro *ensembles* com os melhores classificadores. O melhor desempenho individual verificado atinge a acurácia de 91,57%, enquanto o melhor desempenho geral foi observado em um *ensemble* das três melhores arquiteturas mais profundas, atingindo 93,82% de acurácia. Ambos os resultados superam classificadores obtidos pelo autor em momentos anteriores e também alguns trabalhos disponíveis na literatura, bem como mostram-se mais eficientes que especialistas humanos para esta tarefa neste mesmo conjunto de dados. Os modelos obtidos foram treinados com dados representativos, com tuberculose em diversos níveis, compreendendo pacientes com diferentes perfis e a partir de uma base de dados pública e gratuita, o que favorece a reproduzibilidade e a posterior aplicação em cenários práticos.

Palavras-Chave. Tuberculose, Redes Neurais Convolucionais, Aprendizado Profundo.

Abstract

Diagnose tuberculosis is crucial for proper treatment, as TB is one of the top ten causes of death in the world. Considering a computer-assisted approach, this paper presents the results of a strategy for the intelligent recognition of tuberculosis in chest radiographs with deep convolutional neural networks. To achieve this result, we considered 5 canonical architectures of this type of network, subjected to several parameters and hyperparameters, counting 47 trained and tested models for this task, as well as four ensembles with the best classifiers. The best individual performance verified reached the 91.57% accuracy, while the best overall performance was observed in one ensemble of the three best deepest architectures, reaching 93.82% accuracy. Both results outweigh classifiers obtained by the author at earlier times and also some papers available in the literature, as well as prove to be more efficient than human experts for this task in this same dataset. The models obtained were trained with representative data, with tuberculosis at various levels, comprising patients with different profiles and from a free public dataset, which favors reproducibility and subsequent application in practical scenarios.

Key-words. Tuberculosis, Convolutional Neural Networks, Deep Learning.

Agradecimentos

A Deus, Inteligência Suprema e causa primária de todas as coisas.

A Jesus, Sol da vida humana.

A Clarêncio, por retirar-me do umbral uma vez mais e secar minhas lágrimas.

A Chico Xavier, por ensinar-me a valorizar o trabalho e o tempo que corre.

A minha mãe, Lucinéia, e meu pai, Valberto, almas benditas que me aceitaram como filho nessa encarnação.

A meus irmãos, Luís e Erick, pelas risadas e conhecimentos em *memes* e jogos.

A Mayara, por me ensinar que estrelas gêmeas existem.

A todos os professoras da UEA, em especial a minha orientadora Elloá Guedes, me ensinaram sobre a disciplina, a dedicação pela docência, o afincô pelo método científico e ética no trabalho. Minha paixão por computação foi semeada por vocês.

A Giovanna, Janderson, Nicoli, Rafaela, Miranda, Marcos, Luiz, Leo, Isabella, Kazuo e Rodrigo, pela amizade e camaradagem dispensandas a mim sem medida.

Aos espíritas, representados nas figuras institucionais do Centro Espírita Jesus de Nazaré, da Cruzada dos Militares Espíritas e da Federação Espírita Amazonense. Voltei para casa mais uma vez e encontrei trabalho graças a todos vocês.

A Lorena, Couto, Cecília, Clara, Rayanna, Maria, Ciro, Miguel, Boo, Lucas, Stef, Natália, Kim, Sanderson e Erick, e a todos os companheiros espirituais do Jornal Mocidade representados na figura ímpar do amigo Elias da Silva. Muito obrigado pelo carinho e consolo que recebi até

aqui.

A comunidade vegana de Manaus, representada na imagem da Daniele e da Cecília, que me ensinam incessantemente a amar e proteger aqueles que não tem voz.

Aos amigos do ICTS e da TPV, representados na figura do Osmar, Anderson, Calixto, Ruan, André e Guilherme. Vocês me ensinam diariamente a ser um profissional melhor.

Aos invisíveis da sociedade: a cada trabalhador do campo e agricultor, que ajudou a levar alimento ao meu prato, a cada gari que varreu a rua, a cada motorista e cobrador de ônibus. Este trabalho também é de vocês.

Aos que lutam diariamente a batalha com a tuberculose, em todas as esferas a vida, seja o médico que vem com remédio salutar, seja para o doente aflito e agonizante, seja para a família que suporta o martírio e a dor, seja o pesquisador, em algum labotatório do mundo, que busca incessantemente novos meios de diagnóstico, tratamento e cura. A batalha contra a tuberculose é uma luta de todos.

Agradeço também à Fundação de Amparo à Pesquisa do Estado do Amazonas (FAPEAM) que, por meio dos Projetos PROTI Pesquisa 11/2017 e PPP 04/2017, colaborou para a consolidação da infraestrutura física e tecnológica do Laboratório de Sistemas Inteligentes da Escola Superior de Tecnologia da Universidade do Estado do Amazonas. Este trabalho de conclusão de curso é um dos produtos destes projetos, pois foi desenvolvido no referido laboratório, fez uso dos recursos computacionais ali disponíveis e foi melhorado grças às discussões e interações com o grupo de pesquisa nele sediado.

A vida não cessa. A vida é fonte eterna e a morte é o jogo escuro das ilusões.

André Luiz, por Chico Xavier.

Sumário

Lista de Tabelas	xii
Lista de Figuras	xiii
1 Introdução	1
1.1 Objetivos	2
1.2 Justificativa	3
1.3 Metodologia	3
1.4 Organização do Documento	4
2 Fundamentação Teórica	5
2.1 Tuberculose	5
2.2 <i>Deep Learning</i>	9
2.3 Redes Neurais Artificiais	10
2.3.1 Redes Neurais Convolucionais	14
2.3.2 Regularização de CNNs	18
2.3.3 <i>Dropout</i>	19
2.3.4 <i>Data Augmentation</i>	20
2.3.5 <i>Transfer Learning</i>	21
3 Trabalhos Relacionados	23
4 Solução Proposta	26
4.1 Tarefa de Aprendizado	26

4.2	Dados Experimentais	27
4.3	Modelos Considerados	30
4.4	Parâmetros e Hiperparâmetros dos Modelos	31
4.5	<i>Transfer Learning</i>	35
4.6	<i>Ensemble</i> de Classificadores	37
5	Resultados e Discussão	38
6	Considerações Finais	45
A	Exemplos de <i>Data Augmentation</i>	51
A.1	<i>Data Augmentation Hard</i>	51
A.2	<i>Data Augmentation Soft</i>	52

Lista de Tabelas

4.1	Tabela de modelos, parâmetros e hiperparâmetros para as configurações 1, 2 e 3.	34
4.2	Tabela de modelos, parâmetros e hiperparâmetros para as configurações 4, 5 e 6.	35
5.1	Tabela de resultados dos modelos para as configurações 1, 2 e 3.	39
5.2	Tabela de resultados dos modelos para as configurações 4, 5 e 6.	40
5.3	Resultados para as CNNs sujeitas às técnicas de <i>Transfer Learning</i>	40
5.4	Resultado para <i>Ensemble 1.1</i> com as três melhores CNNs.	42
5.5	Resultado para <i>Ensemble 1.2</i> com as três melhores CNNs.	42
5.6	Resultado para <i>Ensemble 2</i> com as três melhores CNNs.	42
5.7	Resultado para <i>Ensemble 3</i> com as três melhores CNNs.	43

Listas de Figuras

2.1	Mapa do panorama da TB no mundo, por 100 mil habitantes dentro da respectiva população, no ano de 2016. Fonte: (WHO, 2017).	7
2.2	Mapa do panorama da TB no Brasil. As cores da legenda representam a incidência do número de casos. Fonte: (BRASIL, 2016b).	8
2.3	Exemplo de variadas tarefas de <i>Visão Computacional</i> comumente endereçadas por técnicas de DL. Fonte: (KHAN et al., 2018).	10
2.4	Um neurônio abstrato. Fonte: (ROJAS, 1996).	11
2.5	Objetos linearmente separáveis. Fonte: (FACELI et al., 2011).	12
2.6	Arquitetura de um MLP com duas camadas ocultas. Fonte: (HAYKIN, 2008).	13
2.7	A operação de uma camada convolucional. (a)–(i) mostram os cálculos performedos a cada passo, enquanto o filtro percorre o mapa de características da entrada para calcular o valor corresponde no mapa de saída. O filtro 2×2 (em <i>verde</i>) é multiplicado com a região de mesmo tamanho (em <i>laranja</i>) com um mapa de caracerísticas de entrada 4×4 , e ao fim os resultados são somados para obter o valor de entrada (em <i>azul</i>) no mapa de saída de cada etapa convolucional. Fonte: (KHAN et al., 2018).	15
2.8	Exemplo de filtros aprendidos pela primeira camada de convolução e saída da ativação desses filtros ao apresentar uma face para a rede. Fonte: (DELICATO; PIRES; SILVEIRA, 2017).	16

2.9	A operação de uma camada de <i>pooling</i> , realizando um <i>max-pooling</i> , que retorna o maior valor obtido entre as saídas. (a)–(i) mostram o cálculo realizado em cada passo que a região de <i>pool</i> passa no mapa de características de entrada (em <i>laranja</i>) para calcular o valor correspondente no mapa de saída (em <i>azul</i>). Fonte: (KHAN et al., 2018).	17
2.10	Exemplo de arquitetura de CNN para reconhecimento de grafia – processamento de imagem. Fonte: (HAYKIN, 2008).	18
2.11	<i>Dropout</i> aplicado a uma rede neural com duas camadas internas. Fonte: (SRI-VASTAVA et al., 2014).	20
4.1	Quatro exemplos distintos que constam na base dados consolidada para este trabalho.	29
5.1	Perda e acurácia por época coletadas durante as fases de treinamento e validação da CNN GoogLeNet com melhor perfomance individual observada.	41
5.2	Matriz de Confusão da CNN GoogLeNet com melhor perfomance individual observada.	41
A.1	<i>Data augmentation Hard</i> com exemplo positivo para TB do Conjunto Shenzhen Hospital X-ray.	51
A.2	<i>Data augmentation Hard</i> com exemplo negativo para TB da Base de dados JSRT.	52
A.3	<i>Data augmentation Hard</i> com exemplo positivo para TB do Conjunto Montgomery County X-ray.	52
A.4	<i>Data augmentation Soft</i> com exemplo positivo para TB do Conjunto Shenzhen Hospital X-ray.	53
A.5	<i>Data augmentation Soft</i> com exemplo negativo para TB da Base de dados JSRT.	53
A.6	<i>Data augmentation Soft</i> com exemplo positivo para TB do Conjunto Montgomery County X-ray.	53

Capítulo 1

Introdução

A Tuberculose (TB) é uma doença respiratória que afeta principalmente os pulmões, e é uma das maiores causas de mortes em nível global, ultrapassando os índices conhecidos da HIV e malária (WHO, 2017). O diagnóstico para TB é feito através do estudo da história de adoecimento do paciente e pelo exame clínico; devendo ser confirmado por exames específicos, como no caso da baciloscopy, cultura do escarro e pelo raio-X da região torácica, podendo ser necessários outros exames, como a biópsia, dependendo do órgão afetado (HEEMSKERK et al., 2015).

A grande parte das mortes por TB poderia ser evitada com diagnóstico precoce e tratamento apropriado. Outras alternativas de diagnóstico já existem, como os sorodiagnósticos, que são simples, rápidos, baratos e não invasivos, no entanto, sua eficácia é questionada pela literatura (HEEMSKERK et al., 2015). Em contrapartida, os exames de raio-X possuem as mesmas características de destaque e são, no caso do Brasil, nacionalmente disponíveis pelo Sistema Único de Saúde (SUS).

Neste âmbito, as técnicas desenvolvidas pelo Aprendizado de Máquina (AM) têm alcançado resultados positivos em um vasto número de problemas do mundo real, devido à sua capacidade de generalização. Esta subárea da Inteligência Artificial almeja desenvolver métodos que automaticamente realizam o aprendizado de padrões complexos usando observações do mundo real, sem a definição explícita de regras ou lógica pelos programadores (KHAN et al., 2018). Segundo Chollet, AM é a busca por representações úteis de alguns dados de entrada, dentro de um espaço predefinido de possibilidades, usando a orientação de um sinal de resposta. Essa ideia simples

permite resolver uma gama extremamente ampla de tarefas intelectuais, do reconhecimento de fala à condução autônoma de automóveis (CHOLLET, 2017).

No entanto, à medida que se avança para problemas mais complexos, como reconhecimento de objetos e análise de texto, os dados se tornam de alta dimensionalidade, e os relacionamentos que se deseja capturar tornam-se altamente não-lineares. Para acomodar essa complexidade, pesquisas recentes em AM tentaram construir modelos que se assemelham às estruturas utilizadas por nossos cérebros. É essencialmente esse corpo de pesquisa, comumente chamado de aprendizado profundo – no original, *Deep Learning* (DL) –, que teve um amplo destaque na resolução de problemas de Visão Computacional, em especial. Para determinadas tarefas, esses algoritmos não apenas superam os resultados documentados no estado da arte, mas também rivalizam - ou até excedem - as precisões alcançadas pelos humanos (BUDUMA; LOCASCIO, 2017).

Deste modo, almejando endereçar a questão da detecção de tuberculose a partir de imagens de raio-X, este trabalho se propôs avaliar a utilização de modelos de DL para esta tarefa, conforme será detalhado nas seções a seguir.

1.1 Objetivos

O objetivo geral deste trabalho consiste em avaliar a utilização de redes neurais convolucionais para detectar tuberculose em radiografias torácicas. Para alcançar esta meta, alguns objetivos específicos precisam ser contemplados, a citar:

1. Formular um referencial teórico sobre as redes neurais convolucionais profundas;
2. Consolidar uma base de dados de imagens radiográficas de tórax;
3. Identificar tecnologias para implementação das redes neurais convolucionais profundas em ambiente computacional;
4. Identificar, propor e/ou modificar arquiteturas de redes neurais convolucionais para a tarefa considerada;

5. Treinar e testar os diferentes modelos propostos; e
6. Avaliar os resultados obtidos.

1.2 Justificativa

De acordo com a OMS, a TB é uma das maiores causas de mortes em nível mundial, ultrapassando os índices da HIV e malária (WHO, 2017). Levando isto em consideração, o presente trabalho de conclusão apresenta uma proposta de solução que colabora para o diagnóstico automatizado, diminuindo os custos, possibilitando a utilização em larga escala e com pouca intervenção humana, o que pode ser crucial especialmente para antecipar o início do tratamento medicamentoso, minimizando posteriores complicações. Além disso, por se basear nas imagens de raio-X, um exame de baixo custo e não invasivo, pode ser uma solução adequada para regiões remotas e com pouco acesso a profissionais de saúde, apoiando o diagnóstico.

Considerando a perspectiva do discente que desenvolve este trabalho, o projeto permite a prática de métodos, técnicas e tecnologias de vanguarda, numa área emergente da Aprendizagem Máquina, garantindo a continuidade de questões de pesquisa elencadas em um projeto de iniciação científica anterior, que tratou da problemática do diagnóstico de TB em imagens de radiografias torácicas apenas com modelos *Perceptron Multicamadas*. No mais, este trabalho também é importante pois colabora com os objetivos do *Grupo de Pesquisas em Sistemas Inteligentes*, da Escola Superior de Tecnologia (EST) da Universidade do Estado do Amazonas (UEA).

1.3 Metodologia

Para alcançar os objetivos propostos no escopo deste trabalho, a condução das atividades realizadas obedeceu à metodologia escrita a seguir:

1. Estudo dos conceitos relacionados às Redes Neurais Convolucionais e ao *Deep Learning*;

2. Levantamento do ferramental tecnológico para implementação das arquiteturas profundas das redes neurais convolucionais;
3. Consolidar uma base de dados representativa de imagens radiográficas torácicas acreditada por especialistas contendo exemplos de pacientes saudáveis e acometidos por TB;
4. Selecionar um conjunto de arquiteturas canônicas de redes neurais convolucionais aplicáveis ao problema em questão;
5. Treinar as redes identificadas para a tarefa de aprendizado considerada;
6. Testar as redes previamente treinadas com vistas a coletar métricas de desempenho;
7. Analisar os resultados e elencar os modelos mais adequados para o problema considerado;
8. Escrita da proposta de Trabalho de Conclusão de Curso;
9. Defesa da proposta de Trabalho de Conclusão de Curso;
10. Escrita do Trabalho de Conclusão de Curso; e
11. Defesa do Trabalho de Conclusão de Curso.

1.4 Organização do Documento

Este documento está organizado como segue. Inicialmente, a Seção 2 corresponde à fundamentação teórica, contemplando os conceitos relativos à tuberculose, AM e DL. A Seção 3, por sua vez, engloba uma análise dos trabalhos relacionados, os quais sintetizam o estado da arte para o diagnóstico de TB com AM e DL. Na Seção 4 é apresentada a solução proposta para a tarefa de diagnosticar TB em imagens radiográficas da região posteroanterior. Na seção 5 estão detalhados e discutidos os resultados obtidos. Por fim, as considerações finais podem ser encontradas na Seção 6, juntamente com uma visão geral dos próximos passos.

Capítulo 2

Fundamentação Teórica

Nessa seção é apresentada a fundamentação teórica que dá suporte à realização deste trabalho. A Seção 2.1 contempla os conceitos relacionados à tuberculose, considerando também estatísticas de mortalidade e planos de combate. Os fundamentos de *Deep Learning*, sua relação com Inteligência Artificial e Aprendizagem Máquina, assim como resultados mais relevantes de seu uso, são descritas na Seção 2.2. Em seguida, na Seção 2.3, são descritas as Redes Neurais Artificiais, considerando a perspectiva do ferramental matemático e computacional, com principal destaque para as Redes Neurais Convolucionais e conceitos relacionados, tais como regularização, *dropout* e *Transfer Learning*.

2.1 Tuberculose

Tuberculose (TB) é uma doença crônica infecciosa, que afeta geralmente os pulmões, causada principalmente pela bactéria *Mycobacterium tuberculosis* (HEEMSKERK et al., 2015). A forma principal de contágio é pelas vias aéreas de pessoas para pessoa, sendo que cerca de um terço da população mundial possui sua forma latente, o que significa que foram infectados pela TB, mas (ainda) não estão doentes e não podem transmiti-la (JAMISON et al., 2006).

De acordo com a Organização Mundial da Saúde (OMS), a TB é uma das maiores causas de mortes em abrangência mundial, passando os índices relativos a HIV e malária (WHO, 2017). Segundo Crofton, dois fatores determinam o risco de transmissão do bacilo da TB para um

receptor saudável: a concentração de partículas contaminadas suspensas no ar, e o período de exposição do indivíduo que respira esse ar contaminado (CROFTON, 2009).

Dados da Organização Pan-Americana da Saúde (PAHO, do inglês *Pan-American Health Organization*) apontam que a maior carga de TB recai sobre comunidades que já enfrentam desafios socioeconômicos, a exemplo de migrantes, refugiados, pessoas privadas de liberdade, pessoas que vivem em situação de rua, minorias étnicas, mineiros e outras que trabalham e vivem em ambientes sujeitos a riscos, além das mulheres marginalizadas, crianças e idosos. Mais de um terço dos acometidos pela TB não são diagnosticados ou notificados.

Há também um grande número de indivíduos que não recebem tratamento ou que este é de qualidade questionável (PAHO, 2017). O diagnóstico para a maioria dos pacientes suspeitos de TB em todo o mundo ainda é feito por baciloscopia para bacilos álcool-ácido resistentes. No entanto, existem outras alternativas como: cultura microbacteriana, que é um teste mais sensível; testes de amplificação de ácido nucleico, que se destacam por estar entre os maiores avanços da última década para um diagnóstico mais exato; e os sorodiagnósticos, que são simples, rápidos, baratos, não invasivos e comercializados com alegações de alta sensibilidade e especificidade – resultados estes questionados pela literatura (HEEMSKERK et al., 2015).

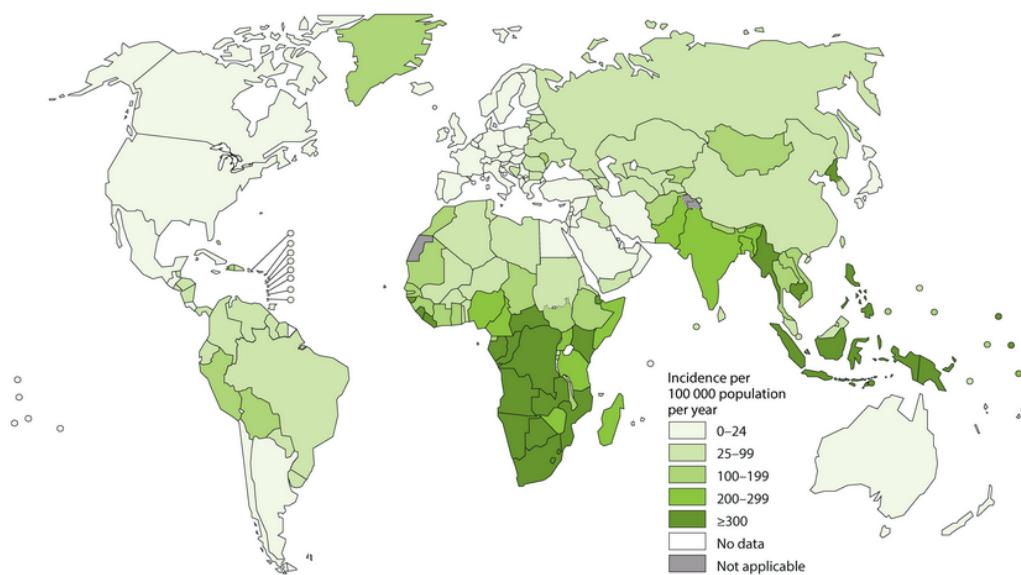
A grande parte das mortes por TB poderia ser evitada com diagnóstico precoce e tratamento apropriado. Milhões de pessoas são diagnosticadas com TB e tratadas com sucesso todos os anos, impedindo muitas de mortes: 53 milhões de óbitos evitados de 2000 a 2016, mas ainda há grandes barreiras na detecção e tratamento (AIT-KHALED et al., 2003).

Embora a TB seja uma doença antiga entre a humanidade (BARBERIS et al., 2017), e o número de infectados a nível global esteja caindo a uma taxa de 3% ao ano, o último *Global Tuberculosis Report* de 2017 da OMS relata que, em 2016, 6,3 milhões de novos casos de TB foram reportados, em contraste com os 6,1 milhões notificados em 2015, equivalendo a 61% da estimativa de incidência de 10,4 milhões; os últimos dados a respeito de tratamento mostram uma taxa de sucesso global de 83%, similar aos anos recentes. Foram registrados 476.774 casos de HIV-positivo com TB, dos quais 85% estavam em tratamento antiretroviral. Um total de 129.689 pessoas foram iniciadas no tratamento de drogaresistência de TB, um adicional pequeno

dos 125.629 casos reportados em 2015. De maneira geral, o sucesso no tratamento permanece baixo, a uma taxa global de 54%. Conforme o relatório conclui, fica evidente a alta carga da doença, e um progresso que ainda não é rápido o suficiente para atingir metas ou fazer grandes progressos no fechamento de lacunas persistentes no diagnóstico e tratamento (WHO, 2017).

A Fig. 2.1 ilustra o cenário da TB em âmbito mundial.

Figura 2.1: Mapa do panorama da TB no mundo, por 100 mil habitantes dentro da respectiva população, no ano de 2016. Fonte: (WHO, 2017).

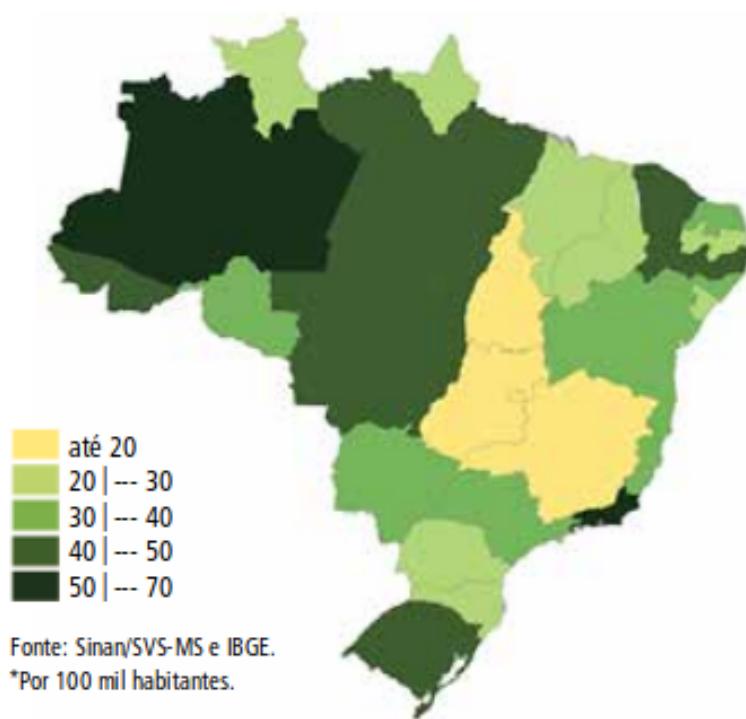


O Brasil é um dos países com o maior número de casos de TB no mundo e, desde 2003, essa doença têm sido considerada como uma prioridade na agenda política do Ministério da Saúde do Brasil. Ainda que a TB seja uma doença com diagnóstico e tratamento garantido de forma geral e gratuita pelo Sistema Unificado de Saúde, seu panorama ainda enfrenta barreiras: aproximadamente 69.000 novos casos e 4.500 mortes ocorrem todos os anos (BRASIL, 2016b).

A Fig. 2.2 representa o panorama da TB no Brasil.

De acordo com a OMS, o Brasil é um dos vinte e dois países que concentram 80% dos casos de TB no mundo (WHO, 2017). Neste grupo, o Brasil ocupa a décima sexta posição entre os novos casos e a vigésima segunda posição dentre os países identificados com alta prioridade de controle de TB. Devido à grande vulnerabilidade, os seguintes grupos foram identificados com alta prioridade para controle desta doença: (i) população carcerária, na qual a incidência de TB chega a ser 28 vezes maior do que a média; (ii) desabrigados ou sem-teto; (iii) pacientes

Figura 2.2: Mapa do panorama da TB no Brasil. As cores da legenda representam a incidência do número de casos. Fonte: (BRASIL, 2016b).



coinfectados com HIV-TB, com uma incidência de 3,6 por 100 mil habitantes; (iv) população indígena, com incidência 3 vezes maior que a média; e (v) profissionais da saúde. A população negra no Brasil também sofre com uma taxa 2,2 vezes maior que a taxa da população geral (BRASIL, 2016b).

Apesar dos dados representados, o número de casos de TB no Brasil têm diminuído como resultado de um trabalho desenvolvido pelo Ministério da Saúde Brasileiro, através do Programa Nacional de Controle da Tuberculose (PNCT), em parceria com os Estados, cidades e sociedade civil. Como parte do esforço geral para reduzir os índices de incidência e mortalidade, o PNCT elaborou um plano nacional visando o fim da TB como problema de saúde pública no Brasil, desejando alcançar o alvo de menos de 10 casos *per* 100 mil habitantes até o ano de 2035 (BRASIL, 2016a).

2.2 Deep Learning

Inteligência Artificial (IA) é uma área de estudo da Computação na qual máquinas mostram capacidades cognitivas, como aprender padrões, inferência, dedução, visão computacional, reconhecimento de fala, representação do conhecimento, percepção, etc. Mais coloquialmente, IA denota qualquer atividade pela qual máquinas imitam o comportamento inteligente tipicamente mostrado por seres humanos. De modo geral, toda IA recebe inspiração de elementos da Ciência da Computação, Matemática, e Estatística (RUSSEL; NORVIG, 2010).

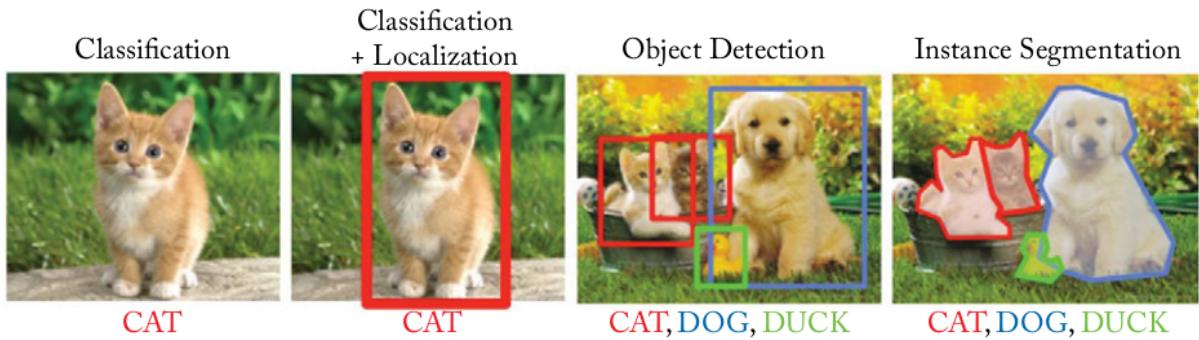
Aprendizagem Máquina (AM) é uma subárea da IA que permite que computadores aprendam a partir de dados sem serem explicitamente programados para tal. Em outras palavras, o objetivo da AM é desenvolver métodos que automaticamente realizam o aprendizado usando observações do mundo real – também chamadas de *training data* – sem a definição concreta de regras ou lógica pelos programadores (KHAN et al., 2018).

Deep Learning (DL), ou Aprendizado Profundo, é um subgrupo específico de técnicas da AM, nas quais o aprendizado é conseguido através de múltiplos níveis de representações e abstrações, sejam para imagens, sons, ou texto, reconhecendo padrões complexos a partir de uma grande quantidade de dados (GULLI; PAL, 2017; DENG; YU, 2014). A ideia de aprender a representação correta dos dados garante uma perspectiva de DL; outra perspectiva é de que a profundidade permite o computador aprender em múltiplos passos, o que vêm significativamente melhorando resultados de técnicas anteriores do estado da arte (GOODFELLOW; BENGIO; COURVILLE, 2016).

DL tem se destacado principalmente em problemas de Visão Computacional, especialmente em virtude do grande volume de imagens disponíveis e da existência de muitos rótulos para este tipo de dados, o que favorece as tarefas de Aprendizado Supervisionado (KHAN et al., 2018). A Fig. 2.3 demonstra múltiplas tarefas de Visão Computacional que são comumente endereçadas com DL.

O atual sucesso de DL é devido principalmente a algumas características do contexto contemporâneo, em especial a disponibilidade de boas bases de dados para treinamento – como a base de dados de imagens *ImageNet*, organizada de acordo com a hierarquia *WordNet* (LAB,

Figura 2.3: Exemplo de variadas tarefas de *Visão Computacional* comumente endereçadas por técnicas de DL. Fonte: (KHAN et al., 2018).



2016) –, e o custo baixo da disponibilidade de GPUs voltadas para cálculos numéricos de alta eficiência. Empresas como Google, Microsoft, Amazon, Apple, e muitas outras usam técnicas de DL todos os dias para analisar grandes quantidades de dados e aprender padrões ali existentes (GULLI; PAL, 2017).

O principal destaque envolvendo DL está relacionada à *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC), proposta pelo *Stanford Vision and Learning Lab* desde 2010, que avalia os algoritmos para detecção de objetos em imagens em larga escala. Uma das suas motivações é permitir que pesquisadores comparem seu progresso através de uma ampla variedade de objetos. Outra motivação é mensurar o progresso da visão computacional para indexação de imagens em grande escala para recuperação e anotação (LAB, 2016). As principais arquiteturas de DL foram propostas inicialmente para concorrerem no ILSVRC, como a *AlexNet*, primeira arquitetura de DL da competição e vencendo em sua primeira tentativa, e a *GoogLeNet*, uma das arquiteturas mais profundas já propostas (Krizhevsky; Sutskever; Hinton, 2012; Szegedy et al., 2015).

2.3 Redes Neurais Artificiais

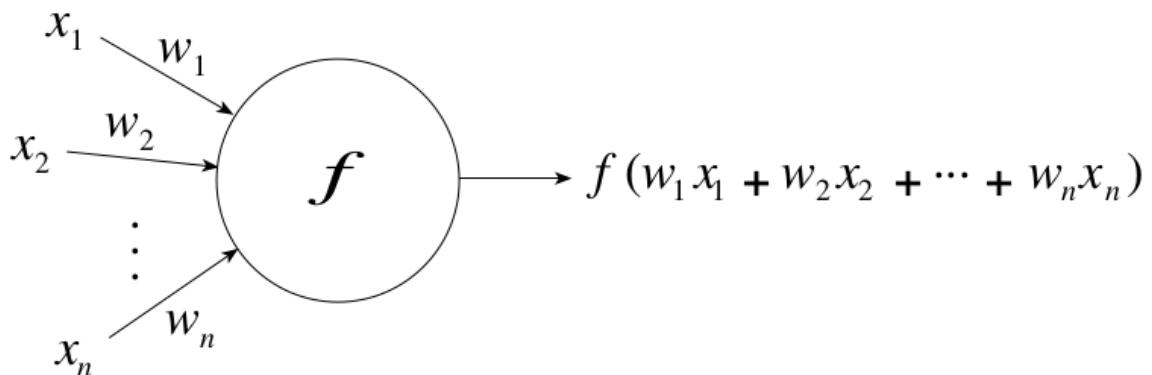
Redes Neurais Artificiais (RNAs) são sistemas computacionais que se inspiram no comportamento e estrutura do sistema nervoso, com o objetivo de simular a capacidade de aprendizado do cérebro humano na aquisição do conhecimento (BRAGA; CARVALHO; LUDERMIR, 2000). O poderio computacional das RNAs se embasa, primeiro, na sua distribuição paralela massiva

e, segundo, nas suas capacidades de aprendizado e, consequentemente, generalizar (HAYKIN, 2008). De acordo com Delicato et al., as RNAs são as protagonistas na atual revolução produzida pela AM (DELICATO; PIRES; SILVEIRA, 2017).

O sistema nervoso humano possui arquiteturas globais de varidade complexa, mas todas são compostas por blocos de construção, as células neurais ou *neurônios* (ROJAS, 1996). Similarmente, RNAs são compostas por unidades mínimas chamadas *neurônios*, que efetuam a soma ponderada de uma certa entrada e então sujeitam este resultado à variadas funções matemáticas, nomeadas *funções de ativação*, o que predispõe uma varidade grande de topologias possíveis (FACELI et al., 2011; ROJAS, 1996). Cada RNA é construída por vários neurônios, organizados em *camadas*, que trocam mensagens entre si – *disparam* – dentro de certas condições, interconectados através de *sinapses*, que estão associados a pesos (GULLI; PAL, 2017; ROJAS, 1996).

A Figura 2.4 mostra a estrutura de um neurônio com n entradas. Cada canal de entrada i pode transmitir um valor real x_i . A *função de ativação* f , calculada dentro do neurônio abstrato pode ser selecionada arbitrariamente. A partir dos pesos nos canais de entrada, a informação inicial x_i é multiplicada pelo correspondente w_i . A informação transmitida é integrada ao neurônio – usualmente através da adição de diferentes sinais – e então a função de ativação é calculada (ROJAS, 1996).

Figura 2.4: Um neurônio abstrato. Fonte: (ROJAS, 1996).

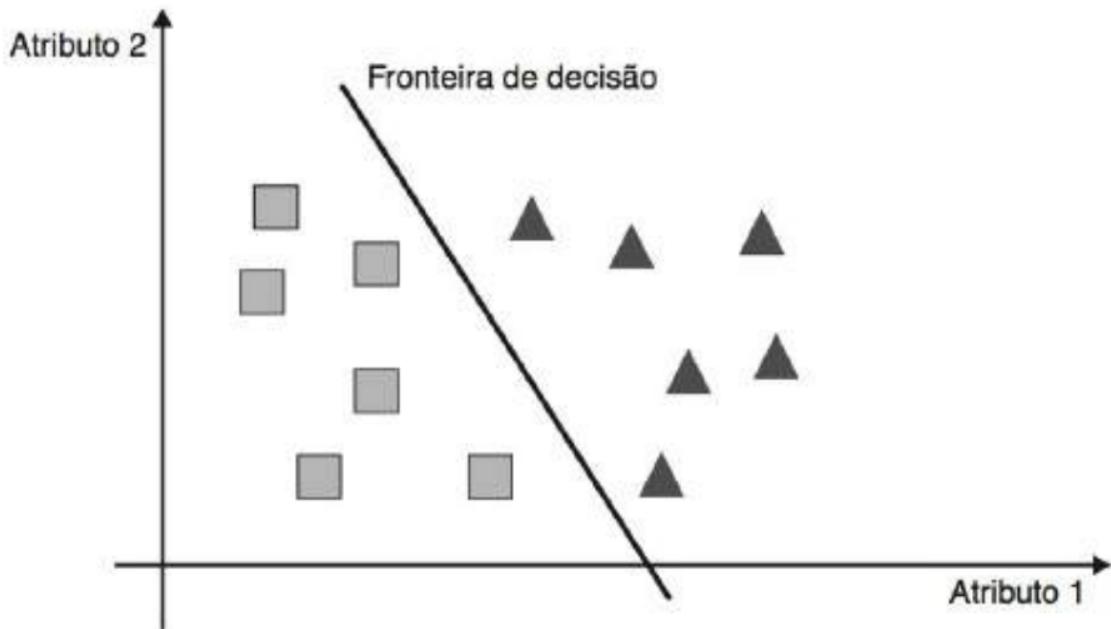


Diferentes seleções dos pesos w_1, w_2, \dots, w_n produzem diferentes resultados; consequentemente, existem três elementos que são particularmente importantes em qualquer modelo de

RNAs: (i) a estrutura dos neurônios; (ii) a topologia da rede; e, (iii) o algoritmo de aprendizado utilizado para encontrar os pesos da rede (ROJAS, 1996).

Uma rede neural de uma única camada, chamada de *perceptron*, consegue classificar apenas objetos que são linearmente separáveis, portanto, apenas separa o espaço de entrada em dois subespaços distintos (FACELI et al., 2011; ROJAS, 1996). A Figura 2.5 exemplifica dados com dois atributos que são linearmente separáveis. Se em vez de dois objetos apresentarem d atributos, o espaço de soluções será d -dimensional. Os objetos de duas classes distintas serão linearmente separáveis se houver um hiperplano que separe os dados das duas classes (FACELI et al., 2011).

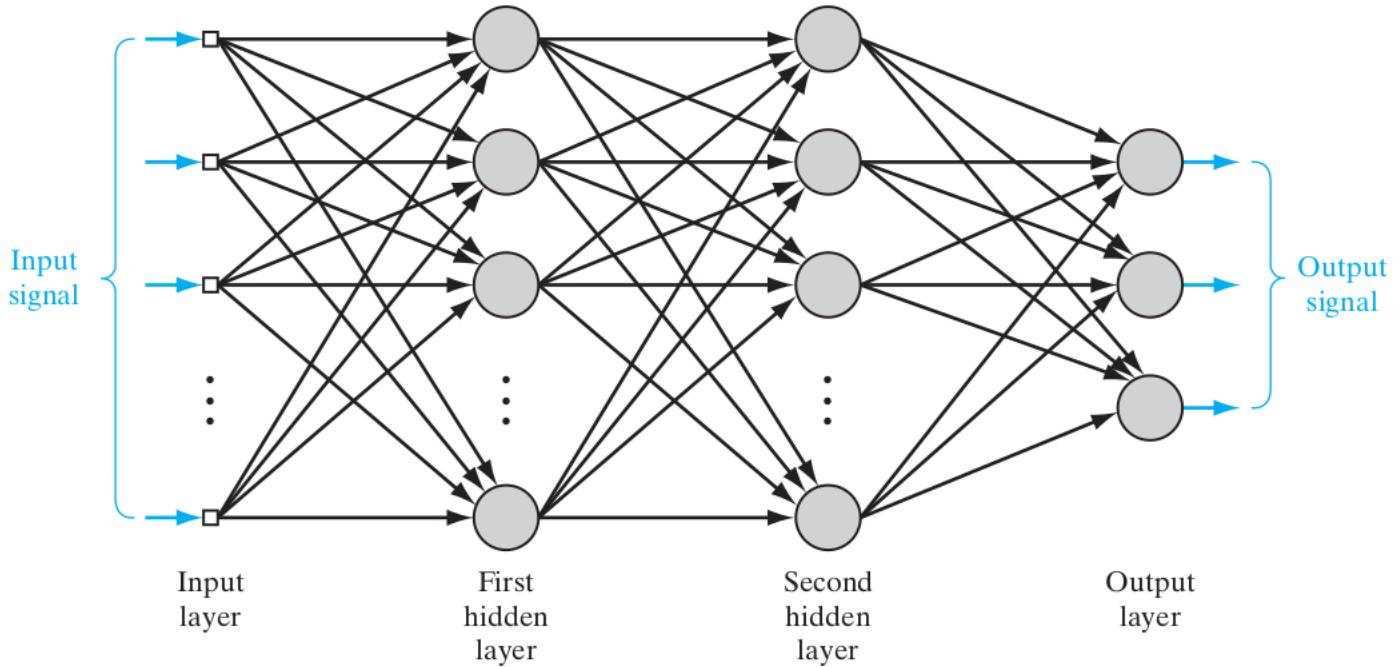
Figura 2.5: Objetos linearmente separáveis. Fonte: (FACELI et al., 2011).



Para resolver problemas não linearmente separáveis utilizando RNAs, a alternativa mais utilizada é adicionar uma ou mais camadas intermediárias (FACELI et al., 2011). Segundo Russell, um *modelo perceptron de várias camadas* (MLP, do inglês *Multi-Layer Perceptron*) pode aproximar qualquer função, dado neurônios suficientes (RUSSEL; NORVIG, 2010). A Figura 2.6 ilustra um MLP com duas camadas internas.

O modelo MLP possui três aspectos que ressaltam suas características básicas: (i) o modelo de cada neurônio na rede inclui uma função de ativação não-linear; (ii) a rede contém uma ou

Figura 2.6: Arquitetura de um MLP com duas camadas ocultas. Fonte: (HAYKIN, 2008).



mais camadas que são intermediárias, chamadas de camadas ocultas, estando entre a camada de entrada e a de saída; e, (iii) a rede apresenta um alto grau de conectividade, cuja extensão é determinada pelos pesos sinápticos da rede (HAYKIN, 2008).

Em contrapartida, essas mesmas características são também responsáveis pela dificuldade em se conhecer o comportamento da rede. A presença de uma forma distribuída de não-linearidade e uma alta conectividade da rede fazem com que a análise teórica de uma rede MLP difícil de entender (HAYKIN, 2008). À medida que a informação avança pelas camadas, o processamento realizado se torna mais complexo, e o uso de neurônios internos faz com que o processo de aprendizado seja mais difícil de visualizar (FACELI et al., 2011; HAYKIN, 2008). Segundo Marsland, o aprendizado nas RNAs acontecem na atualização de seus pesos (MARS-LAND, 2015), mas devido ao alto grau de conexões, a busca por novos pesos em um MLP precisa ser conduzida em um espaço maior de funções possíveis (HAYKIN, 2008; GOODFELLOW; BENGIO; COURVILLE, 2016).

A melhor técnica conhecida para atualização dos pesos é a técnica de *backpropagation* (KUBAT, 2015), que, segundo Haykin, representa um marco no estudo das RNAs, sendo considerado

um método computacionalmente eficiente para o treinamento de MLPs (HAYKIN, 2008; GOODFELLOW; BENGIO; COURVILLE, 2016). O algoritmo de *backpropagation* é dividido em duas fases, uma fase para frente (*forward*) e uma fase para trás (*backwards*). Na fase *forward*, o objeto é recebido pelos neurônios da camada de entrada, produzindo um valor de saída e passando adiante para as próximas camadas como valor de entrada, até chegar na camada de saída, onde os neurônios produzem um valor final que é comparado ao valor desejado para saída (FACELI et al., 2011; KUBAT, 2015). A diferença entre esses valores indica o erro cometido pela MLP. O valor do erro é então utilizado pela fase *backwards* para ajustar o valor dos pesos. O ajuste prossegue da camada de saída até a primeira camada intermediária. A derivada parcial da função de ativação define o ajuste dos pesos, utilizando o gradiente descendente – essa derivada mede a contribuição de cada peso no erro da rede para a classificação do dado objeto (FACELI et al., 2011). Conforme Rojas, o método de *backpropagation* não é somente mais geral do que as derivações analíticas, mas também mais fácil de acompanhar (ROJAS, 1996).

As aplicações das redes neurais MLPs já são consolidadas na literatura e abrangem as mais diversas áreas, inclusive com utilização em contextos práticos, tais como o reconhecimento de caracteres manuscritos, controle de trens de alta velocidade, identificação de perfis de clientes para análise de concessão de crédito, classificação de diferentes variedades de chás, determinação da velocidade com que poluentes penetram no solo, detecção de alteração do leite de gado por outras substâncias, dentre muitas outras (SILVA et al., 2017).

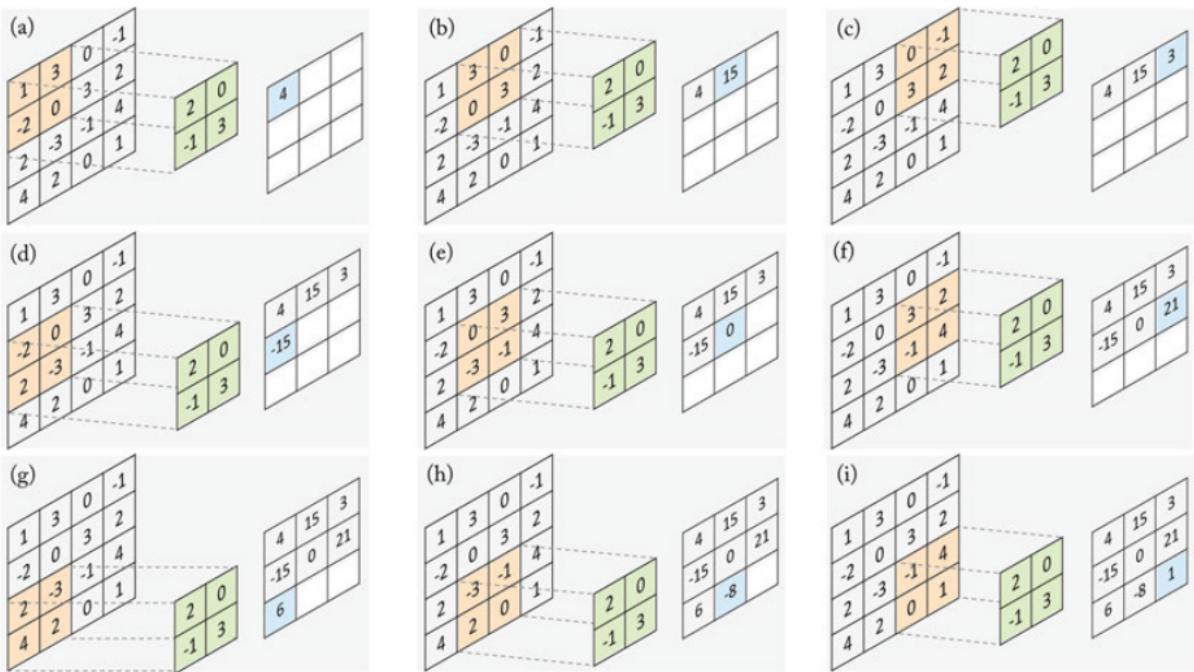
2.3.1 Redes Neurais Convolucionais

Rede Neural Convolucional (CNN, do inglês *Convolutional Neural Network*) é uma RNA com arquitetura de alimentação para frente especialmente projetada para processamento de sinais visuais (DELICATO; PIRES; SILVEIRA, 2017). Especificamente, CNNs são rede neurais *perceptron* multicamadas desenvolvidas para reconhecer formas bi-dimensionais com alto grau de invariância à escala, inclinação e outras formas de distorção, além de ser aplicado para analisar séries temporais, imagens, ou fala (HAYKIN, 2008; LECUN; BENGIO, 1995).

A característica principal de uma CNN são as *camadas convolucionais*, que possuem uma

série de filtros (também chamados de *kernels convolucionais*, ou somente *kernels*) que são convoluidos a partir de uma determinada entrada para gerar uma mapa de características como saída (KHAN et al., 2018). Uma convolução, de modo geral, é uma operação em duas funções (GOODFELLOW; BENGIO; COURVILLE, 2016), cujos passos são exemplificados detalhadamente na Figura 2.7.

Figura 2.7: A operação de uma camada convolucional. (a)–(i) mostram os cálculos performedos a cada passo, enquanto o filtro percorre o mapa de características da entrada para calcular o valor correspondente no mapa de saída. O filtro 2×2 (em verde) é multiplicado com a região de mesmo tamanho (em laranja) com um mapa de caracerísticas de entrada 4×4 , e ao fim os resultados são somados para obter o valor de entrada (em azul) no mapa de saída de cada etapa convolucional. Fonte: (KHAN et al., 2018).

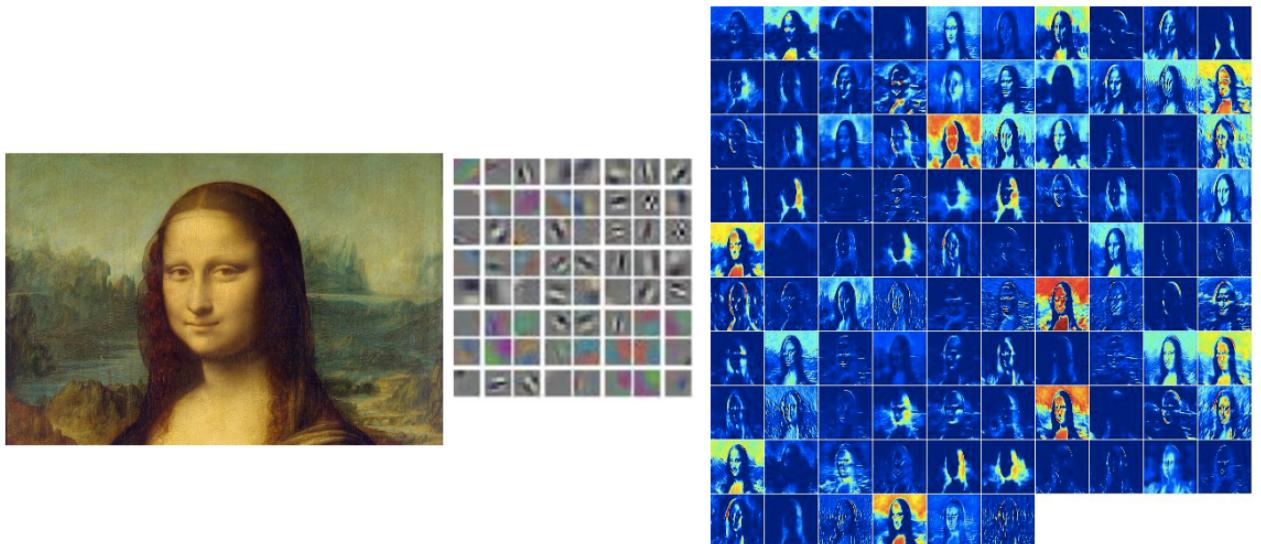


O método da convolução engloba três ideias importantes: (i) interações esparsas; (ii) compartilhamento de parâmetros; e, (iii) representações equivariantes. Mais especificamente, a convolução fornece um meio para trabalhar com entradas de tamanho variável (GOODFELLOW; BENGIO; COURVILLE, 2016; LECUN; BENGIO, 1995).

Além da convolução, CNNs usam a técnica de campos receptores locais, compartilhamento de pesos e subamostragem para garantir que as primeiras camadas extraiam e combinem características locais a partir de uma forma invariante de distorção (LECUN; BENGIO, 1995). Ao

longo de uma CNN, é construída uma hierarquia de características, na qual as camadas iniciais extraem características elementares, como bordas e cantos, que são combinadas sucessivamente para a extração de características de mais alto nível semântico nas camadas seguintes (DELICATO; PIRES; SILVEIRA, 2017). A Figura 2.8 exemplifica o resultado final desse processo de extração de características.

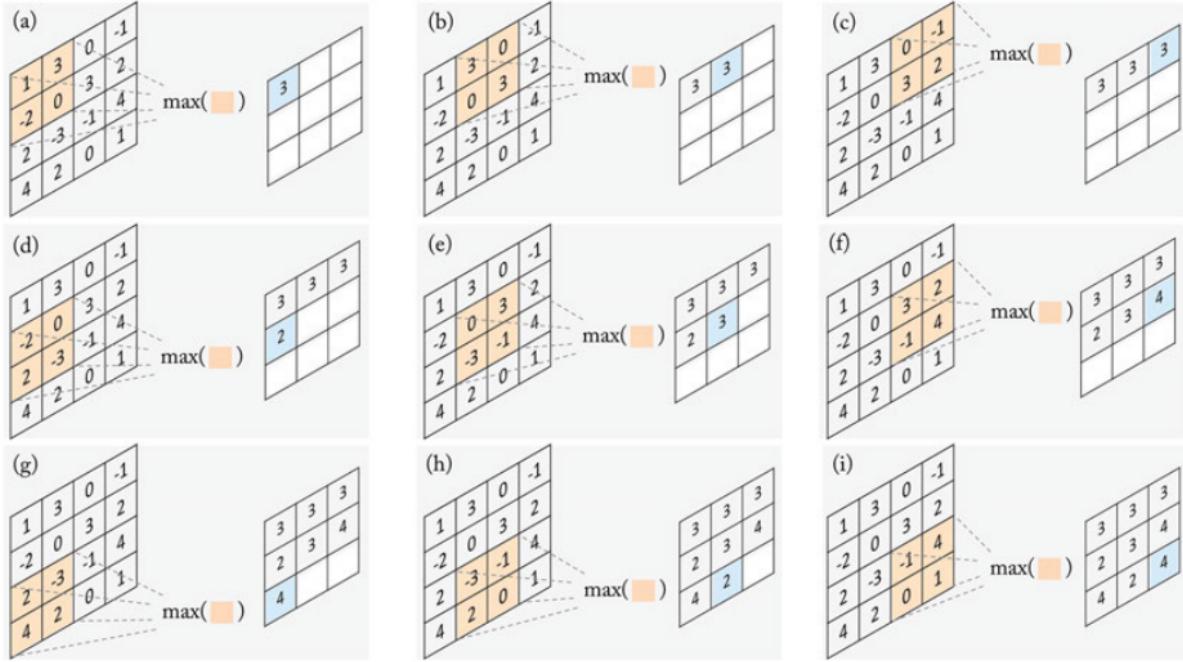
Figura 2.8: Exemplo de filtros aprendidos pela primeira camada de convolução e saída da ativação desses filtros ao apresentar uma face para a rede. Fonte: (DELICATO; PIRES; SILVEIRA, 2017).



Um típica camada de uma CNN consiste em três etapas. Na primeira etapa, a camada realiza várias convoluções em paralelo para gerar uma série de ativações lineares. Na segunda etapa, cada ativação linear passa por uma função de ativação não-linear. Na última etapa, é utilizado uma função de *pooling* para modificar a saída da camada. A função de *pooling*, ou camada de *pooling*, troca o valor de saída da camada através de uma interação entre os valores de saída pertencentes aos vizinhos mais próximos (GOODFELLOW; BENGIO; COURVILLE, 2016). A Figura 2.9 demonstra as operações de uma camada de *pooling*.

A Figura 2.10 ilustra a topologia que forma uma arquitetura de uma CNN, composta pela camada de entrada, quatro camadas internas, e uma camada de saída aplicada ao reconhecimento de dígitos manuscritos oriundos do *dataset* MNIST. A camada de entrada recebe uma matriz de 28×28 pixels correspondendo à imagem de um caractere manuscrito. Posteriormente,

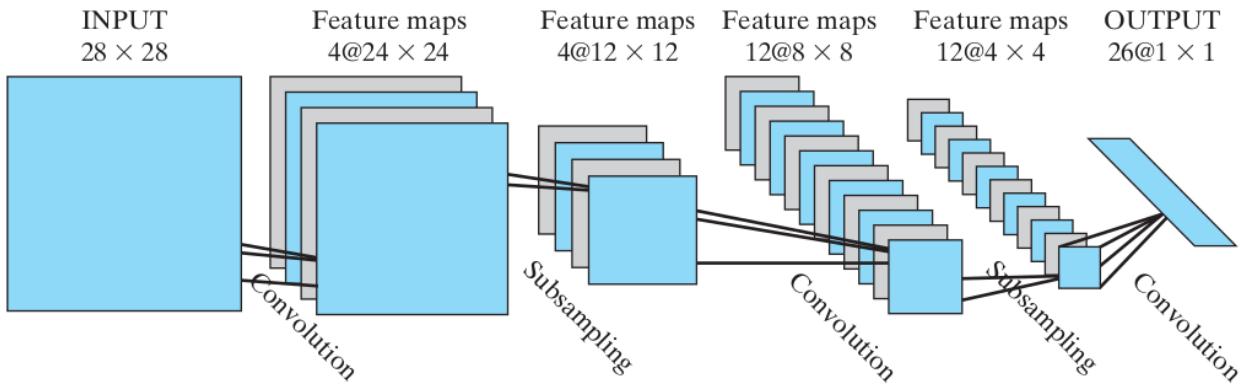
Figura 2.9: A operação de uma camada de *pooling*, realizando um *max-pooling*, que retorna o maior valor obtido entre as saídas. (a)–(i) mostram o cálculo realizado em cada passo que a região de *pool* passa no mapa de características de entrada (em laranja) para calcular o valor correspondente no mapa de saída (em azul). Fonte: (KHAN et al., 2018).



os *layouts* computacionais alternam entre convolução e subamostragem (HAYKIN, 2008).

1. A primeira camada interna, com 4 mapas de características de 24×24 neurônios cada, realiza a convolução;
2. A segunda camada interna, com 4 mapas de características de 12×12 neurônios cada, realiza a subamostragem e a média local;
3. A terceira camada interna, com 12 mapas de características de 8×8 neurônios cada, realiza a segunda convolução;
4. A quarta camada interna, com 4 mapas de características de 4×4 neurônios cada, realiza a segunda subamostragem e a média local; e
5. A camada de saída, com 26 neurônios, realiza a etapa final da convolução e retorna um vetor de probabilidade para cada uma das 26 letras do alfabeto.

Figura 2.10: Exemplo de arquitetura de CNN para reconhecimento de grafia – processamento de imagem. Fonte: (HAYKIN, 2008).



2.3.2 Regularização de CNNs

A complexidade da superfície de decisão de um classificador é proporcional ao número de parâmetros livres que ele possui, ou seja, o número de parâmetros utilizados para a determinação das superfícies de decisão. Intuitivamente pode parecer lógico que, para que um classificador seja poderoso, esse deve construir fronteiras de decisão detalhadas. Mas quando o número de parâmetros livres é grande, o classificador tende a se adaptar a detalhes específicos da base de treinamento, o que pode causar uma redução da taxa de acerto. Esse fenômeno é conhecido como *overfitting* (CAMPOS, 2000).

Como CNNs possuem uma quantidade grande de parâmetros, elas tendem ao *overfitting* durante a fase de treino. Isso, portanto, resulta em uma performance inferior em novos dados – geralmente nos dados de teste. Métodos de *regularização* visam evitar esse problema (KHAN et al., 2018).

Os métodos de regularização mais comumente aplicados no contexto das CNNs para evitar *overfitting* são:

1. A adição de *dropout*, cujo funcionamento será detalhado na seção a seguir;
2. O teste de diferentes arquiteturas, com a adição ou remoção de camadas;
3. A adição de regularização L1 ou L2, as quais visam impor restrições aos pesos da rede, que devem assumir valores pequenos, fazendo, por conseguinte, que a distribuição dos

valores de pesos mais regular. Com estas técnicas, associa-se um custo aos pesos grandes, em que:

- (a) Regularização L1: o custo é proporcional ao valor absoluto dos coeficientes dos pesos;
 - (b) Regularização L2: o custo é proporcional ao quadrado do valor dos coeficientes dos pesos;
4. O teste de diferentes hiperparâmetros, tais como o número de neurônios por camada, o valor da taxa de aprendizado, dentre outros, com vistas a obter uma configuração óptima; ou
5. Opcionalmente, a realização de engenharia de atributos, considerando a remoção ou adição de diferentes entradas (CHOLLET, 2017).

É importante salientar que a aplicação das técnicas de regularização mencionadas está sujeita à posterior avaliação do seu efeito mediante treinamento e teste. Estas práticas reforçam a construção de modelos mais confiáveis.

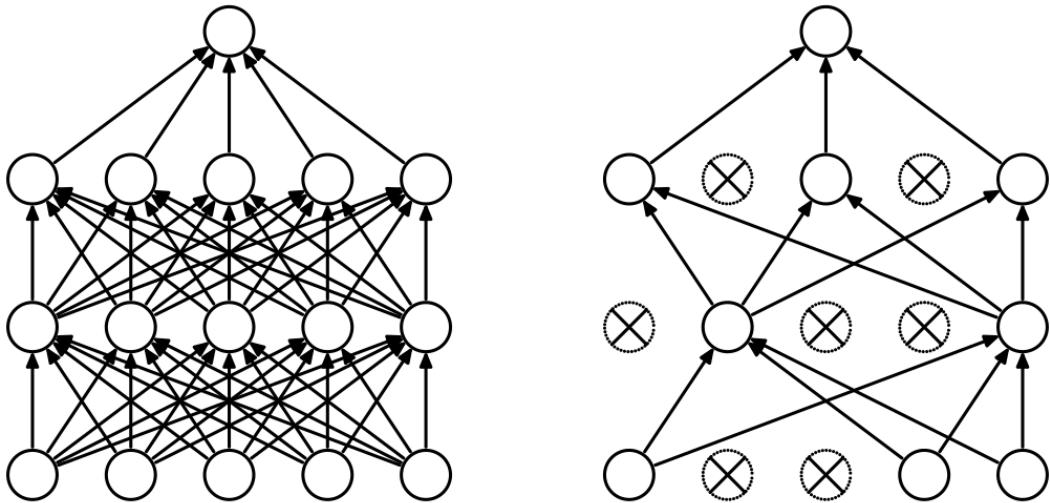
2.3.3 *Dropout*

A técnica de *dropout* é uma das técnicas de regularização mais efetivas para RNAs (CHOLLET, 2017). Ela é aplicada para melhorar o desempenho de redes neurais em uma ampla variedade de domínios de aplicação, incluindo classificação de objetos, reconhecimento de dígitos, reconhecimento de fala, classificação de documentos e análise de dados de biologia computacional. Isso sugere que o *dropout* é uma técnica geral e não é específica de nenhum domínio (SRIVASTAVA et al., 2014).

O termo *dropout* refere-se ao abandono de unidades (ocultas e visíveis) em uma rede neural. Ao descartar uma unidade, esta é removida temporariamente da rede, juntamente com todas as suas conexões de entrada e saída. A escolha de quais unidades descartar é aleatória. No caso mais simples, cada unidade é retida com uma probabilidade fixa p independente de outras unidades, onde p pode ser escolhido usando um conjunto de validação ou pode simplesmente ser

definido como 0,5, o que parece estar próximo do ideal para uma ampla gama de redes e tarefas. Para as unidades de entrada, no entanto, a probabilidade ótima de retenção é geralmente mais próxima de 1 do que de 0,5. (SRIVASTAVA et al., 2014).

Figura 2.11: *Dropout* aplicado a uma rede neural com duas camadas internas. Fonte: (SRIVASTAVA et al., 2014).



2.3.4 *Data Augmentation*

Data augmentation é uma técnica muito eficaz de aumentar o poder de generalização dos modelos de CNN (KHAN et al., 2018). A ideia básica por trás do *data augmentation* é de que o *overfitting* é causada por ter uma quantidade pequena de dados para aprender, impossibilitando o modelo ser treinado para generalizar novos dados. Se houvesse uma quantidade arbitrariamente grande de exemplos, um dado modelo estaria exposto a cada aspecto possível dos dados, ou seja, nunca ocorreria o *overfitting* (CHOLLET, 2017). *Data augmentation* utiliza a abordagem de gerar mais dados de treinamento a partir de amostras de treinamento existentes, aumentando as amostras por meio de várias transformações aleatórias que geram imagens de aparência confiável. O objetivo é que no tempo de treinamento, o modelo nunca verá a mesma entrada duas vezes. Isso ajuda a expor o modelo a mais aspectos dos dados e a generalizar melhor (CHOLLET, 2017).

As técnicas típicas de *data augmentation* aplicadas a problemas de Visão Computacional envolvem a realização de operações de modificação na imagem de entrada que não mudam

exatamente a semântica do seu conteúdo, tais como: rotações, inversões horizontais, zoom, leves distorções de cisalhamento (*shearing transformations*), dentre outras (CHOLLET, 2017).

2.3.5 Transfer Learning

Transfer Learning (TL) e seu domínio se refere à situação na qual o que já foi utilizado para treinar uma CNN é aproveitável para aprimorar a generalização em outra tarefa de aprendizado análoga (GOODFELLOW; BENGIO; COURVILLE, 2016). Isso se faz através do uso dos pesos obtidos após a etapa de treinamento. Tal abordagem permite o compartilhamento de diversos tipos de dados de entrada, dando possibilidade a aprender representações que capturam fatores subjacentes, um subconjunto dos quais pode ser relevante para cada tarefa específica (DENG; YU, 2014).

Seguindo essa linha de raciocínio, pesquisadores do ramo da Visão Computacional geralmente usam CNNs pré-treinadas para gerar representações de novas tarefas, onde a base de dados talvez não seja grande o suficiente para treinar uma CNN inteira do zero – também chamada de *feature extractor*. Outra tática comum é utilizar uma rede treinada com as imagens da base ImageNet e alterar camadas básicas da rede inteira para se adaptar à tarefa proposta – essa abordagem recebe o nome de *fine tuning* (GULLI; PAL, 2017; SARKAR; BALI; GHOSH, 201; LAB, 2016).

As possíveis vantagens de se utilizar *Transfer Learning* são (SARKAR; BALI; GHOSH, 201):

- **Melhor desempenho no *baseline*:** ao aumentar o conhecimento de um *learner* isolado (ou *learner* ignorante) com conhecimento de outra fonte de modelo, a performance do *baseline* talvez melhore devido à essa transferência de conhecimento;
- **Tempo no aprendizado do modelo:** ao utilizar o conhecimento de outro modelo talvez ajude também no aprendizado pleno da tarefa alvo, quando comparado com a tarefa do modelo alvo que aprende do zero. Isso, consequentemente, resulta numa melhoria no tempo necessário para que o modelo aprenda; e

- **Melhoria na perfomance final:** maior perfomance ao final do processo pode ser alcançada ao se utilizar *Transfer Learning*.

Capítulo 3

Trabalhos Relacionados

Sistemas especialistas (SE) consistem nas primeiras respostas adotadas para auxiliar no diagnóstico de TB. Sistemas especialistas são um ramo de Inteligência Artificial cuja ideia básica é transferir conhecimento humano para um computador (LIAO, 2005). Nesta perspectiva, um sistema especialista foi previamente proposto com vistas a replicar as decisões de especialistas experientes no diagnóstico de TB mediante sintomas dos pacientes e sua intensidade, como febre, dor abdominal, lesões de pele, teste sanguíneo, entre outros (IMIANVAN; OBI, 2011). Embora o resultado obtido seja relativamente recente, essa abordagem é considerada superada pela visão atual da literatura, por ser fortemente dependente de intervenção humana, amostras clínicas e testes diagnósticos (RUSSEL; NORVIG, 2010).

Os trabalhos de Elveren e Yumuşak (ELVEREN; YUMUŞAK, 2011), Er et al. (ER; TEMURTAS; TANRIKULU, 2010) e de El-Solh e colaboradores (EL-SOLH et al., 1999) baseiam-se na utilização de RNAs MLPs para distinção de pacientes acometidos por TB de indivíduos saudáveis a partir da dados oriundos de testes laboratoriais e exames preliminares – presença de tosse, pressão alta, presença de líquido dos pulmões, quantidade de leucócitos no sangue, hemoglobina, nível do fosfatase alcalina, etc, diferenciando-se quanto ao modo de treinamento das redes (busca de parâmetros por algoritmos genéticos, por exemplo) e arquitetura utilizada. Os resultados apontam acurácia de 93,30% e 93,93%, respectivamente (ER; TEMURTAS; TANRIKULU, 2010; EL-SOLH et al., 1999). No entanto, todos esses trabalhos apresentam limitações no tocante à necessidade de intervenção humana, custos envolvidos para a realização de exames,

dentre outros.

Abordagens mais autônomas consideradas pela literatura levam em conta que utilizar imagens digitais para o diagnóstico de TB como uma opção mais fácil e mais barata, principalmente do ponto de vista da pouca intervenção humana. Desta forma, muitas abordagens de reconhecimento de padrões foram desenvolvidas para a tarefa de segmentação de pulmões, supressão óssea, detecção das vias aéreas pulmonares, e extração de características relevantes (JAEGER et al., 2013).

Recentemente, com o avanço do *hardware* e *software* computacional assim como a crescente disponibilidade de dados, tornou-se possível realizar uma quantidade imensa de tarefas mais complexas (GOODFELLOW; BENGIO; COURVILLE, 2016). Algoritmos de DL são um exemplo disso, com o aumento das aplicações de Informática aplicadas na Saúde (RAVì et al., 2017), incluindo o campo da Radiologia como em detecção de lesões e doenças, classificação e diagnósticos, segmentação, e quantificação (MCBEE et al., 2018; HWANG et al., 2016; RAJPURKAR et al., 2017; LITJENS et al., 2017).

O trabalho de Lakhani e Sundaram é um dos pioneiros a propor uma solução para o diagnóstico de TB utilizando DL e CNN (LAKHANI; SUNDARAM, 2017). Os autores usaram quatro base de dados com exemplos rotulados e adotaram arquiteturas canônicas de CNNs, como AlexNet e GoLeNet, e também testaram a melhoria das métricas de avaliação com e sem *data augmentation* e *Transfer Learning*. Os melhores resultados foram alcançados com pesos pré-treinados e *data augmentation*, em um modelo baseado em *ensemble* de duas CNNs, com AUC de 0.99 e 98,9% de acurácia. Esses resultados são importantes, mas os autores reforçam que não substituem a interpretação médica-radiológica humana.

Se o número de cálculos computacionais for levado em consideração, o trabalho de Lakhani e Sundara, pode ser considerado como tendo um custo computacional elevado devido à profundidade dos modelos adotados (LAKHANI; SUNDARAM, 2017). Outro aspecto a se considerar é de que eles usaram imagens de quatro base de dados onde uma delas, provinda da Universidade Thomas Jefferson, que não é disponível ao público, comprometendo a reproduzibilidade dos resultados e dificultando comparar com resultados anteriores neste campo.

Similarmente, Hwang et. al propuseram uma CNN profunda alimentada com mais de 11 mil imagens de radiografias torácicas, coletadas a partir de três bases dados (*Shenzhen, Montgomery e KIT*) (HWANG et al., 2016). A CNN utilizada foi baseada numa arquitetura canônica de DL chamada *AlexNet*, que é treinada para reconhecimento em imagens (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). A rede neural proposta foi treinada utilizando *Transfer Learning* com pesos obtidos através da base de dados *ImageNet* (HWANG et al., 2016; LAB, 2016). O trabalho em questão se destaca por treinar a CNN separadamente para cada base de dados. Os melhores resultados foram alcançados nos exemplos constantes na base de dados do Instituto Coreano de Tuberculose (KIT, do inglês *Korean Institute of Tuberculosis*), com AUC de 0.967 e acurácia de 0.903. No entanto, o mesmo entrave é verificado, pois a base de dados KIT não está disponível para acesso público.

Considerando o estado da arte apresentado, este trabalho visa endereçar a utilização de CNNs para o problema de detecção de TB em imagens de raio-X torácicas utilizando bases de dados públicas e anotadas por especialistas, apresentando claramente as arquiteturas utilizadas e suas modificações, permitindo a reproduzibilidade dos resultados e uma comparação mais equânime com novas estratégias que venham a ser apresentadas em novos trabalhos da literatura.

Capítulo 4

Solução Proposta

Nesta seção são apresentados os elementos que compõem a solução proposta. Inicialmente, na Seção 4.1, será descrita como a detecção de TB em imagens de raio-X torácicas será endereçada considerando uma tarefa de Aprendizado de Máquina. Em seguida, a Seção 4.2 descreve os dados experimentais que serão utilizados, os quais irão fornecer experiência sobre a tarefa em questão. Os modelos de CNN que serão utilizados para aprendizado a cerca dos padrões nos dados são apresentados na Seção 4.3, seguidos dos parâmetros e hiperparâmetros a que os mesmos estarão sujeitos, descritos na Seção 4.4.

4.1 Tarefa de Aprendizado

Dada uma imagem de raio-X da região torácica, deseja-se saber se esta possui ou não sinais sugestivos de TB. Para prover experiência acerca desta tarefa, foram consideradas imagens representativas do problema oriundas de bases de dados consolidadas e acreditadas por especialistas. Assim, teve-se um cenário de tarefa de classificação binária mediante o paradigma de aprendizado supervisionado.

Os dados experimentais disponíveis foram divididos aleatoriamente seguindo o critério de validação *hold-out*: 80% dos dados são para fase de treino, 10% para validação e, consequentemente, 10% para teste; todos os *batchs* possuem exemplos saudáveis e com TB. Estes dados foram utilizados para treinar e testar algumas arquiteturas canônicas de CNNs: *LeNet*, *AlexNet*,

GoogLeNet, VGG e ResNet (LECUN et al., 2015; KRIZHEVSKY; SUTSKEVER; HINTON, 2012; SZEGEDY et al., 2015; SIMONYAN; ZISSERMAN, 2014; HE et al., 2016).

Considerando o objetivo desta tarefa, duas métricas de performance foram utilizadas para avaliar o desempenho das CNNs propostas na fase de teste: acurácia e *F-Score*. A primeira indica a proporção de classificações feitas corretamente pelo modelo, fornecendo uma visão mais simples e direta do desempenho, razão pela qual será adotada como métrica de referência. A segunda, em contrapartida, é uma métrica mais holística, por ser a medida harmônica entre os graus de precisão e revocação do modelo.

4.2 Dados Experimentais

Visando a obtenção de uma base de dados realística, radiografias da região posteroanterior foram coletadas a partir de base de dados públicas já existentes:

1. **Base de dados JSRT**: criada e mantida pela Sociedade Japonesa de Tecnologia Radiológica (*JSRT*, em inglês), essa base de dados contém 247 imagens, onde 154 são exemplos com nódulos pulmonares e 93 são saudáveis. Todas as imagens estão em alta resolução, 2048×2048 pixels em escala de cinza, com metadados em arquivo de texto que possuem gênero do paciente, dianóstico, e outras informações (SHIRAISHI et al., 2000);
2. **Conjunto Montgomery County X-ray**: as imagens nesta base de dados foram adquiridas através do programa de controle de TB do Departamento de Saúde e Serviços Humanos do Município de Montgomery, nos Estados Unidos. Contém 138 imagens de raio-X da região posteroanterior, onde 80 são os casos normais e 58 possuem TB, em diferentes estágios de manifestação (JAEGER et al., 2014); e
3. **Conjunto Shenzhen Hospital X-ray**: as imagens de raio-X deste conjunto foi coletada pelo Hospital Shenzhen, da China, como parte da rotina de acompanhamento dos pacientes. Possui 326 imagens normais e 336 imagens com TB, em vários estágios (JAEGER et al., 2014).

Após a coleta, as imagens e seus metadados foram analisados. Esta etapa revelou que as imagens não saudáveis da base de dados JSRT não continham descrições específicas dos nódulos pulmonares, não garantindo a confiabilidade de representarem caixas torácicas com TB, e portanto foram descartadas.

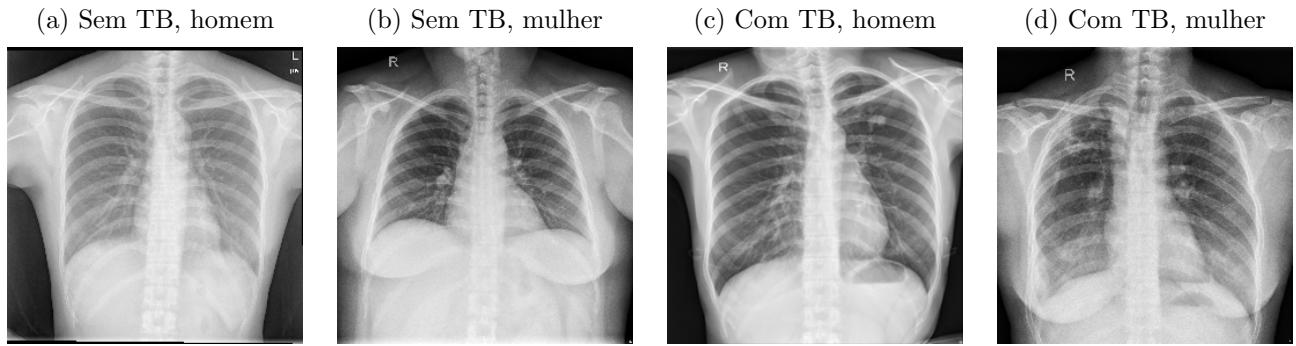
A próxima etapa considerou a padronização das imagens através do redimensionamento, pois as imagens eram muito grandes e estavam em dimensões diferentes. Visando o diagnóstico, a literatura sugere redimensionamento de 128×128 pixels até 1024×1024 pixels. Neste trabalho, foi utilizado o redimensionamento para 256×256 pixels devido ao tempo e custo computacional pelo modelo de Aprendizado de Máquina escolhido e para evitar *overfitting* a partir de uma possível grande quantidade de *features* irrelevantes no treino.

Ao final, a base de dados consolidada possuía 893 imagens, onde 394 são exemplos com TB de acordo com as descrições médicas que acompanhavam as imagens originais em arquivos de texto, e o restante foi composto por imagens saudáveis. Como pode-se constatar, esta base de dados é desbalanceada, uma vez que a quantidade de exemplos por classe difere de maneira razoável. Considerando esta característica, o *F-Score* adotado, do tipo *micro*, permite uma avaliação de desempenho mais adequada, razão pela qual foi considerado, pois leva em conta a proporção de exemplos por classe.

É importante salientar que esta base de dados é realística para o problema em questão, pois possui exemplos de manifestação de TB ao redor do globo, com diferentes intensidades, envolvendo pacientes masculinos e femininos com diversas idades e características físicas. A Figura A.6 ilustra alguns exemplos contidos na base de dados consolidada.

A base de dados foi utilizada em seu formato original e também mediante aumento artificial segundo as técnicas de *data augmentation*. Primeiramente, o conjunto de treino original sofreu mutações aleatórias (espelhamento vertical, rotação em 90° , rotação em 270° , rotação de 0° a 10° para esquerda ou direita), até alcançar a quantidade de 10.000 imagens modificadas. Essa primeira estratégia de aumento de dados foi denominada *data augmentation hard*. Porém, conforme sugestões posteriores, optou-se também por uma abordagem mais sutil neste aumento artificial de dados, visando refletir possíveis deslocamentos comuns no posicionamento

Figura 4.1: Quatro exemplos distintos que constam na base dados consolidada para este trabalho.



do paciente durante a captura deste tipo de imagem médica e também nas estratégias adotadas pelos profissionais ao observarem tais imagens, refletindo de maneira mais realística possíveis entradas. Assim, esta segunda abordagem, denominada *data augmentation soft*, contemplou o aumento da base de dados de treino até 10.000 exemplos com alterações mais sutis, tais como rotações entre 0° a 10° para esquerda ou direita. Os exemplos de imagens e suas mutações mediante as duas abordagens supracitadas podem ser vistas no Apêndice A.

Além disso, mesmo que *data augmentation* seja utilizada para melhorar a capacidade de generalização dos modelos de *Machine Learning*, especialmente nas tarefas de Visão Computacional, nem todas as técnicas disponíveis foram consideradas adequadas para uso neste trabalho devido aos riscos de erros de diagnósticos que podem ser inseridos. Portanto, em ambos os casos, evitou-se utilizar outras variações de *augmentation*, como espelhamento horizontal, zoom aleatório, recorte aleatório, por comprometerem a veracidade fisiológica da representação dos exemplos saudáveis e doentes. Inversões horizontais, em particular, violam a suposição de assimetria horizontal que pode confundir com a condição de *situs inversus*, na qual o paciente apresenta inversão completa de todos os órgãos torácicos e abdominais (EVANGELISTA; GUEDES, 2018a; MOUNT; POLLAK, 2007).

4.3 Modelos Considerados

A proposição de topologias eficientes de CNNs dentro do campo de DL é uma tarefa difícil, ainda mais para o cenário de imagens torácicas em escala de cinza. Em um trabalho anterior desenvolvido, foram propostos dois modelos de arquiteturas de alto nível, totalizando 9 CNNs diferentes (EVANGELISTA; GUEDES, 2018a). Em contraparida, nenhuma delas era tão profunda quanto os modelos canônicos existentes, que têm demonstrado excelente desempenho em competições de classificação de imagens e Visão Computacional (HAYKIN, 2008; CHOLLET, 2017; BUDUMA; LOCASCIO, 2017). Considerando esse fator, dando prosseguimento à pesquisa, foram utilizadas as seguintes arquiteturas:

- **LeNet.** É um dos primeiros sucessos na aplicação real de CNNs com *backpropagation*, quando em 1989 Yann LeCunn desenvolveu uma topologia profunda de CNN para reconhecimento de caracteres impressos e caligrafia (LECUN et al., 2015; CHOLLET, 2017);
- **AlexNet.** É a primeira arquitetura de CNN profunda apresentada no ILSVRC, alcançando o primeiro lugar na sua primeira tentativa com grande margem de diferença. Segundo Buduma e Locascio, a AlexNet colocou sozinha o aprendizado profundo no mapa para a visão computacional e revolucionou completamente o campo (KRIZHEVSKY; SUTSKEVER; HINTON, 2012; KHAN et al., 2018; BUDUMA; LOCASCIO, 2017);
- **GoogLeNet.** Com condinome *Inception*, é uma topologia profunda de CNN proposta em 2014 para o ILSVRC. A principal característica desta arquitetura é a melhor utilização dos recursos de computação dentro da rede. A profundidade e a largura da rede são grandes, mantendo o orçamento computacional constante. Para otimizar a qualidade, as decisões referentes à arquitetura foram baseadas no princípio de Hebbian e na intuição do processamento em escala múltipla. A versão particular usada na submissão para a competição ImageNet de 2014 é uma rede de 22 camadas de profundidade, cuja qualidade é avaliada no contexto de classificação e detecção (SZEGEDY et al., 2015; LAB, 2016);

- **VGG.** Essa arquitetura surgiu de um estudo do desempenho de CNNs com o aumento de sua profundidade. O resultado foi a melhoria da acurácia ao elevar a profundidade para 16 até 19 camadas, que era até então significativamente mais profundo do que as arquiteturas em voga na época (2014). Para reduzir o número de parâmetros é usado um filtro 3×3 em todas as camadas convolucionais (SIMONYAN; ZISSERMAN, 2014). Em 2014, a VGG obteve o primeiro e segundo lugar no ILSVRC, dentro das categorias localização e classificação, respectivamente (LAB, 2016).
- **ResNet.** Nome originário de *Residual Neural Network*, a ResNet faz parte de uma nova tendência no universo das RNAs ao acrescentar conexões residuais em um modelo. Uma conexão residual consiste em reinjetar representações prévias no fluxo seguinte de dados adicionando a resposta de saída de um tensor para a camada seguinte, o que ajuda na prevenção de perda de informação no decorrer do fluxo de processamento de dados (CHOLLET, 2017). Redes residuais são mais fáceis de serem otimizadas e podem ganhar acurácia a partir de um considerável aumento de profundidade. No dataset da ImageNet, com uma profundidade de 152 camadas – 8 vezes mais profunda do que o modelo VGG, mas ainda com uma complexidade menor –, a ResNet obteve uma taxa de erro de 3.57%. Esse resultado ganhou o primeiro lugar no ILSVRC 2015 (HE et al., 2016; LAB, 2016).

4.4 Parâmetros e Hiperparâmetros dos Modelos

Desenvolver um modelo envolve sempre ajustar a sua configuração (*tuning*): escolher o número de camadas internas, número de neurônios na camada de entrada, funções de ativação, taxa de aprendizado do otimizador, adicionar ou retirar *dropout*, etc. (CHOLLET, 2017). Os chamados *parâmetros* são todas as configurações que afetam diretamente os pesos de uma RNA, enquanto os *hiperparâmetros* envolvem as suas demais características e configurações de topologia (CHOLLET, 2017; BUDUMA; LOCASCIO, 2017). Os parâmetros e hiperparâmetros contemplados neste trabalho foram: otimizador, épocas e *patience*.

Um otimizador visa maximizar a performance de um modelo de AM ajustando iterativamente

seus parâmetros até que o erro seja minimizado. Três otimizadores foram empregados:

- **SGD:** *stochastic gradient descent*, o SGD é uma das abordagens mais recorrentes como otimizador, pois a superfície de erro é estimada apenas em relação a um único exemplo. Em vez de uma única superfície de erro estática, a superfície de erro é dinâmica. Como resultado, a descida nessa superfície estocástica melhora significativamente a capacidade de navegar por regiões planas (BUDUMA; LOCASCIO, 2017);
- **Adam:** *ADaptive Moment Estimation*, é um algoritmo com base em estimativas adaptativas de momentos de menor ordem. As atualizações são estimadas usando tanto o primeiro momento quanto o segundo momento do gradiente. É computacionalmente eficiente, tendo poucos requisitos de memória, é invariante para o reescalonamento diagonal dos gradientes e é bem adequado para problemas que são grandes em termos de dados e/ou parâmetros; o método também é apropriado para objetivos não estacionários e problemas com gradientes muito ruidosos e esparsos (KINGMA; BA, 2014; KHAN et al., 2018).
- **RMSprop:** é um método de taxa de aprendizagem adaptativa, não publicada, proposta por Geoff Hinton em seu curso no *Coursera*. É uma extensão de outros algoritmos para otimização baseada no gradiente descendente (AdaGrad, Adaptative Delta, etc), mas buscar reduzir a taxa de aprendizado agressiva e monotônica. De modo geral, adapta a taxa de aprendizagem dos parâmetros, performando atualizações maiores para parâmetros não frequentes e pequenas atualizações para parâmetros frequentes. Por essa razão, é bem adequado para lidar com dados esparsos (KHAN et al., 2018; RUDER, 2016; HINTON; SRIVASTAVA; SWERSKY, 2015).

A taxa de aprendizado, ou seja, a razão com a qual o modelo aprende, foi mantida fixa para todos os otimizadores em 10^{-3} . Este valor é tipicamente sugerido pelas bibliotecas que implementam redes neurais artificiais (GULLI; PAL, 2017).

A *patience* é um hiperparâmetro responsável por fiscalizar o desempenho de alguma métrica de performance da rede (acurácia durante treino, *loss* na validação, etc.), interrompendo a fase

de treino quando a condição especificada for cumprida. Em um primeiro momento, considerando que *data augmentation* foi utilizada, existiram quatro configurações de alto nível para os parâmetros e hiperparâmetros, envolvendo épocas, *patience* e otimizador, o que garantiu que os modelos que sofreram *data augmentation* viessem a ter mais possibilidade de aprender (época e *patience* maiores). As configurações possíveis foram:

1. **Configuração 1.** Épocas: 100, Paciência: 15 (inspecionando *loss* durante validação), Otimizador: Adam ou SGD, sem *data augmentation*;
2. **Configuração 2.** Épocas: 200, Paciência: 35 (inspecionando *loss* durante validação), Otimizador: Adam ou SGD, *data augmentation: hard*; e
3. **Configuração 3.** Épocas: 200, Paciência: 35 (inspecionando *loss* durante validação), Otimizador: Adam ou SGD, *data augmentation: soft*.

Considerando todas as combinações possíveis dos modelos, parâmetros e hiperparâmetros até então propostos, têm-se 30 CNNs a serem treinadas e testadas para o problema em questão. Estas CNNs encontram-se detalhadas na Tabela 4.1.

Tabela 4.1: Tabela de modelos, parâmetros e hiperparâmetros para as configurações 1, 2 e 3.

CNN	Épocas	<i>Patience</i>	Otimizador	<i>Data Augmentation</i>
LeNet	100	15	SGD	-
LeNet	100	15	Adam	-
AlexNet	100	15	SGD	-
AlexNet	100	15	Adam	-
GoogLeNet	100	15	SGD	-
GoogLeNet	100	15	Adam	-
VGG16	100	15	SGD	-
VGG16	100	15	Adam	-
ResNet	100	15	SGD	-
ResNet	100	15	Adam	-
LeNet	200	35	SGD	<i>Hard</i>
LeNet	200	35	Adam	<i>Hard</i>
AlexNet	200	35	SGD	<i>Hard</i>
AlexNet	200	35	Adam	<i>Hard</i>
GoogLeNet	200	35	SGD	<i>Hard</i>
GoogLeNet	200	35	Adam	<i>Hard</i>
VGG16	200	35	SGD	<i>Hard</i>
VGG16	200	35	Adam	<i>Hard</i>
ResNet	200	35	SGD	<i>Hard</i>
ResNet	200	35	Adam	<i>Hard</i>
LeNet	200	35	SGD	<i>Soft</i>
LeNet	200	35	Adam	<i>Soft</i>
AlexNet	200	35	SGD	<i>Soft</i>
AlexNet	200	35	Adam	<i>Soft</i>
GoogLeNet	200	35	SGD	<i>Soft</i>
GoogLeNet	200	35	Adam	<i>Soft</i>
VGG16	200	35	SGD	<i>Soft</i>
VGG16	200	35	Adam	<i>Soft</i>
ResNet	200	35	SGD	<i>Soft</i>
ResNet	200	35	Adam	<i>Soft</i>

Conforme os resultados foram sendo obtidos, decidiu-se, então, propor configurações específicas focadas apenas nas redes mais profundas (GoogLeNet, VGG16 e ResNet), gerando um novo conjunto de parâmetros e hiperparâmetros a serem avaliados:

1. **Configuração 4.** Épocas: 500, Paciência: 50 (inspecionando *loss* durante validação), Otimizador: Adam ou SGD, sem *data augmentation*;

2. **Configuração 5.** Épocas: 500, Paciência: 35 (inspecionando *loss* durante validação), Otimizador: Adam, *data augmentation: hard*; e
3. **Configuração 6.** Épocas: 500, Paciência: 35 (inspecionando *loss* durante validação), Otimizador: Adam ou SGD, *data augmentation: soft*.

De maneira análoga, obteve-se então 18 novas CNNs a serem treinadas e testadas. A Tabela 4.2 reúne seus respectivos valores de parâmetros e hiperparâmetros.

Tabela 4.2: Tabela de modelos, parâmetros e hiperparâmetros para as configurações 4, 5 e 6.

CNN	Épocas	<i>Patience</i>	Otimizador	<i>Data Augmentation</i>
GoogLeNet	500	50	Adam	-
GoogLeNet	500	50	RMSprop	-
VGG16	500	50	SGD	-
VGG16	500	50	Adam	-
GoogLeNet	500	50	Adam	<i>Hard</i>
GoogLeNet	500	50	Adam	<i>Soft</i>
VGG16	500	50	SGD	<i>Soft</i>
VGG16	500	50	Adam	<i>Soft</i>
ResNet	500	50	SGD	<i>Soft</i>
ResNet	500	50	Adam	<i>Soft</i>
ResNet	500	50	RMSprop	<i>Soft</i>

É importante enfatizar que todas as CNNs consideradas possuem como entrada os pixels das imagens redimensionadas para 256×256 , totalizando, na camada de entrada, 65.536 pixels, onde cada neurônio na camada inicial é um vetor com os valores RGB. Todas as redes mencionadas foram treinadas com pesos iniciais obtidos de maneira aleatória.

4.5 Transfer Learning

Como citado anteriormente, *Transfer Learning* é uma abordagem para adaptar e aplicar os conhecimentos adquiridos em outra tarefa relacionada à tarefa em questão. O mais comum é utilizar-se de arquiteturas treinadas em base de dados grandes, como a da ImageNet. A

vantagem de se utilizar a base de dados ImageNet é que ela possui cerca de 15 milhões de imagens (128×128 pixels) distribuídas em mais de 22 mil categorias, o que permite que um modelo aprenda diversas texturas, cores, padrões, características relevantes de seres humanos e animais, etc. No entanto, esta mesma característica também pode ser prejudicial ao modelo aplicado dependendo da tarefa-alvo no contexto de *Transfer Learning* (BUDUMA; LOCASCIO, 2017; LAB, 2016).

Para que haja uma transferência positiva de aprendizado entre uma tarefa origem e uma tarefa-alvo, é comum considerar domínios parecidos, que estejam, de alguma forma relacionados (SARKAR; BALI; GHOSH, 201). Considerando que a base de dados ImageNet contempla apenas imagens coloridas e que a base de dados da tarefa-alvo contém apenas imagens em tons de cinza, decidiu-se também considerar a transferência de aprendizado a partir de um modelo treinado em um domínio visualmente mais próximo, a partir da base de dados FashionMNIST.

A base de dados FashionMNIST possui 70.000 imagens em escala de cinza, com 28×28 pixels, divididas em 10 categorias – todas de peças de roupa. Acredita-se que, em comparação com a ImageNet, a base FashionMNIST seja mais parecida com a base de dados tratada por este trabalho (XIAO; RASUL; VOLLGRAF, 2017). Esta analogia já foi considerada em outro trabalho da literatura (GALINDO; LORENA, 2018). Consequentemente, houve duas abordagens com *Transfer Learning*:

1. **ImageNet.** Todas as 893 imagens de raio-X foram redimensionadas para o tamanho da base de dados da ImageNet, 128×128 pixels. Foram treinadas as arquiteturas GoogLeNet e VGG16, com otimizador SGD e Adam; e
2. **FashionMNIST.** Todas as 70.000 imagens desta base de dados foram redimensionadas para 256×256 pixels. Foram treinadas as arquiteturas GoogLeNet e VGG16, com otimizador Adam.

4.6 Ensemble de Classificadores

A técnica de *ensemble*, também chamada de comitê ou agrupamento, consiste em reunir as previsões de um conjunto de modelos diferentes para produzir melhores previsões. O *ensemble* baseia-se na suposição de que o agrupamento de diferentes bons modelos, treinados independentemente, tenha um bom desempenho por diferentes razões: cada modelo analisa aspectos ligeiramente diferentes dos dados para fazer suas previsões, obtendo parte da “verdade”, mas não de tudo. Nesse contexto, o agrupamento reúne as previsões dos modelos e gera uma nova previsão a partir da votação pela maioria (CHOLLET, 2017).

Em termos de Aprendizado de Máquina, se todos os modelos utilizados forem influenciados da mesma maneira, o agrupamento manterá esse mesmo viés. Se os modelos forem distorcidos de maneiras diferentes, os vieses se anularão mutuamente e o conjunto será mais robusto e mais preciso. Por essa razão, deve-se montar modelos que sejam tão bons quanto possível, sendo os mais diferentes possíveis. Isso normalmente significa usar arquiteturas muito diferentes ou até mesmo diferentes abordagens de Aprendizado de Máquina (CHOLLET, 2017).

Após a conclusão das etapas anteriores, que propõem modelos individuais, resolveu-se então agrupar aqueles com melhor desempenho em comitês, segundo as abordagens a seguir:

1. **Ensemble 1.** Comitê compreendendo as três CNNs com melhor desempenho individual;
2. **Ensemble 2.** Comitê composto apenas das três melhores CNNs com arquiteturas mais profundas; e
3. **Ensemble 3.** Comitê com o melhor integrante de cada arquitetura.

O conjunto de testes será submetido a cada comitê e os resultados obtidos serão aferidos com as métricas de desempenho previamente consideradas. É importante enfatizar que na montagem dos comitês não há necessidade de re-treino dos modelos, apenas de agregação de suas previsões.

Capítulo 5

Resultados e Discussão

Considerando a metodologia proposta, as 10 CNNs propostas foram então implementadas e passaram pelo processo de treinamento, respeitando as configuração de parâmetros e hiperparâmetros. Este treino foi conduzido em um servidor especialmente dedicado para tarefas de Deep Learning, disponível no Laboratório de Sistemas Inteligentes, o qual possui processador Intel Core i7, 16 GB de RAM e placa gráfica com 11 GB de GPU, o qual auxiliou na aceleração em hardware do treinamento. Esta etapa demorou aproximadamente 15 horas. Após a conclusão desta etapa, realizou-se o teste e as métricas de desempenho foram coletadas, as quais encontram-se sintetizadas na Tabela 5.1, a qual também contém o número de épocas até o *patience* limite, parando precocemente o treino quando alcançado. As letras *a*, *b* e *c* que acompanham a numeração das Configurações identificam os otimizadores utilizados – Adam, SGD e RMSprop, respectivamente.

Tabela 5.1: Tabela de resultados dos modelos para as configurações 1, 2 e 3.

CNN	Configuração	Épocas	TP	FP	FN	TN	Acurácia	F-Score
LeNet	1a	100	61	11	13	93	86,52%	0,8356
LeNet	1b	100	74	104	0	0	41,57%	0,5873
AlexNet	1a	100	59	9	15	95	86,52%	0,8310
AlexNet	1b	100	43	9	31	95	77,53%	0,6825
GoogLeNet	1a	100	38	4	36	100	77,52%	0,6552
GoogLeNet	1b	100	72	63	2	41	63,48%	0,6890
VGG16	1a	100	0	0	74	104	58,42%	0,0000
VGG16	1b	100	55	12	16	92	84,00%	0,7971
ResNet	1a	100	62	14	12	90	85,39%	0,8539
ResNet	1b	100	61	12	13	92	85,08%	0,8212
LeNet	2a	200	59	17	15	87	82,02%	0,7867
LeNet	2b	200	74	104	0	0	41,57%	0,5873
AlexNet	2a	200	64	19	10	85	83,71%	0,8153
AlexNet	2b	200	54	14	20	90	80,90%	0,7606
GoogLeNet	2a	200	61	21	13	83	80,90%	0,7821
GoogLeNet	2b	200	63	7	11	97	89,89%	0,8750
VGG16	2a	200	39	39	35	65	58,42%	0,5132
VGG16	2b	200	49	18	25	86	75,84%	0,6950
ResNet	2a	200	63	33	11	71	75,28%	0,7412
ResNet	2b	200	52	9	22	95	82,58%	0,7704
LeNet	3a	200	54	11	20	93	82,58%	0,7770
LeNet	3b	200	0	0	74	104	58,42%	0,0000
AlexNet	3a	200	55	11	19	93	83,15%	0,7857
AlexNet	3b	200	51	6	23	98	83,71%	0,7786
GoogLeNet	3a	200	60	8	14	96	87,64%	0,8451
GoogLeNet	3b	200	65	10	9	94	89,32%	0,8725
VGG16	3a	200	54	7	20	97	84,83%	0,8000
VGG16	3b	200	58	10	16	94	85,39%	0,8169
ResNet	3a	200	61	9	13	95	87,64%	0,8472
ResNet	3b	200	63	11	11	93	87,64%	0,8514

Tabela 5.2: Tabela de resultados dos modelos para as configurações 4, 5 e 6.

CNN	Configuração	Épocas	TP	FP	FN	TN	Acurácia	F-Score
GoogLeNet	4b	500	67	8	7	96	91,57%	0,8993
GoogLeNet	4c	500	65	31	9	73	77,53%	0,7647
VGG16	4a	500	20	18	56	84	58,42%	0,3509
VGG16	4b	500	34	49	42	53	48,88%	0,4277
GoogLeNet	5b	500	58	6	16	98	87,64%	0,8406
GoogLeNet	6b	500	60	5	14	99	89,33%	0,8633
VGG16	6a	500	20	18	56	84	58,43%	0,3509
VGG16	6b	500	34	49	42	53	48,88%	0,4277
ResNet	6a	500	58	7	16	97	87,08%	0,8345
ResNet	6b	500	60	7	14	97	88,20%	0,8511
ResNet	6c	500	57	6	17	98	87,08%	0,8321

Tabela 5.3: Resultados para as CNNs sujeitas às técnicas de *Transfer Learning*.

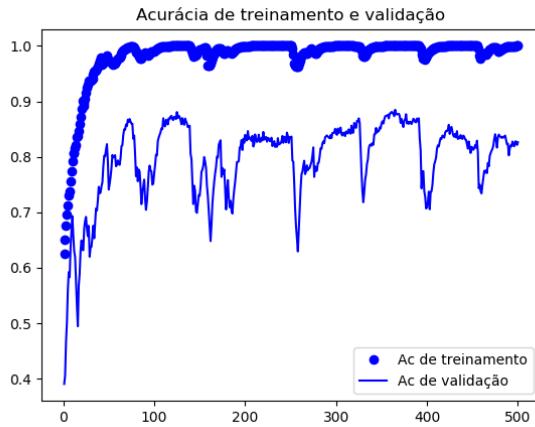
CNN	Dataset	Otimizador	TP	FP	FN	TN	Acurácia	F-Score
GoogLeNet	<i>ImageNet</i>	SGD	22	54	10	92	64,04%	0,4074
GoogLeNet	<i>ImageNet</i>	Adam	11	65	10	92	57,87%	0,2268
VGG16	<i>ImageNet</i>	SGD	54	22	5	97	84,83%	0,8000
VGG16	<i>ImageNet</i>	Adam	63	13	9	93	87,64%	0,8514
GoogLeNet	FashionMNIST	Adam	0	74	0	104	58,43%	0,0000
VGG16	FashionMNIST	Adam	0	74	0	104	58,43%	0,0000

Considerando a métrica de referência obtida e todos os resultados dos testes, foi possível identificar que a melhor estratégia (individual) foi com a CNN GoogLeNet de configuração 4, com 500 épocas, *patience* igual a 50 e otimizador Adam, alcançando acurácia de 0.9157. O desempenho desta CNN e sua matriz de confusão para o conjunto de testes encontram-se detalhados nas Figuras 5.1 e 5.2, respectivamente.

É válido reforçar que a versão da GoogLeNet utilizada neste trabalho foi a Inception-V3, que possui 23 milhões de parâmetros treináveis e 48 camadas internas. Esta rede é a terceira rede mais profunda dentre as utilizadas, atrás da ResNet e VGG16, e com a segunda maior quantidade de parâmetros treináveis, ficando atrás somente da VGG16. Isso leva a crer que a GoogLeNet, por estar num meio-termo entre parâmetros e profundidade quando comparada às

Figura 5.1: Perda e acurácia por época coletadas durante as fases de treinamento e validação da CNN GoogLeNet com melhor performance individual observada.

(a) Acurácia de treinamento e validação



(b) Perda de treinamento e validação

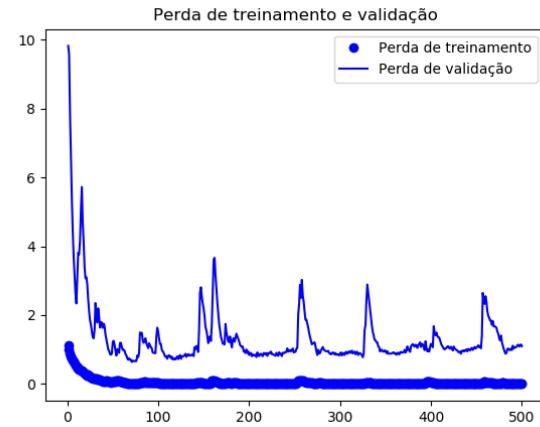
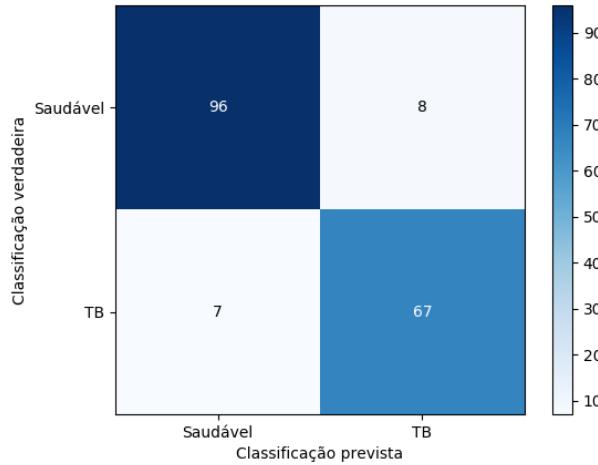


Figura 5.2: Matriz de Confusão da CNN GoogLeNet com melhor performance individual observada.



outras arquiteturas consideradas, consegue extrair mais características relevantes de TB com maior facilidade e em menos épocas.

Dentro do contexto de *Transfer Learning*, percebemos o baixo desempenho dessa abordagem, isso pode ser explicado devido à distância entre domínio de origem e domínio alvo. Tanto as imagens provenientes do *dataset* ImageNet quanto as imagens do FashionMNIST são muito diferentes de uma imagem de raio-X da região posteroanterior, o que exige, consequentemente, das arquiteturas utilizadas uma capacidade de reconhecimento diferente da necessária para se

reconhecer TB. Desde noções de profundidade, cores, formatos, texturas, classes, etc., são componentes que fazem com que as redes aprendam grupos diferencias de extração de características e abstração.

Em relação aos comitês, houve empate no desempenho de duas CNNs, o que ensejou um *ensemble* adicional para a primeira situação, com vistas a avaliar adequadamente todas as possibilidades propostas. Os resultados dos comitês encontram-se nas Tabelas 5.4, 5.5, 5.6 e 5.7. Nestas tabelas, os resultados individuais dos componentes do comitê foram repetidos apenas a título de referência.

Tabela 5.4: Resultado para *Ensemble 1.1* com as três melhores CNNs.

CNN	Configuração	TP	FP	FN	TN	Acurácia	F-Score
GoogLeNet	4b	67	8	7	96	91,57%	0,8993
GoogLeNet	2b	63	7	11	97	89,89%	0,8750
GoogLeNet	3b	65	10	9	94	89,33%	0,8725
<i>Ensemble 1.1</i>	-	57	17	13	91	83,15%	0,7917

Tabela 5.5: Resultado para *Ensemble 1.2* com as três melhores CNNs.

CNN	Configuração	TP	FP	FN	TN	Acurácia	F-Score
GoogLeNet	4b	67	8	7	96	91,57%	0,8993
GoogLeNet	2b	63	7	11	97	89,89%	0,8570
GoogLeNet	6b	60	5	14	99	89,33%	0,8633
<i>Ensemble 1.2</i>	-	56	18	10	94	84,27%	0,8000

Tabela 5.6: Resultado para *Ensemble 2* com as três melhores CNNs.

CNN	Configuração	TP	FP	FN	TN	Acurácia	F-Score
GoogLeNet	4b	67	8	7	96	91,57%	0,8993
ResNet	6b	60	7	14	97	88,20%	0,8511
VGG	<i>ImageNet – Adam</i>	63	13	9	93	87,64%	0,8514
<i>Ensemble 2</i>	-	68	6	5	99	93,82%	0,9252

Tabela 5.7: Resultado para *Ensemble 3* com as três melhores CNNs.

CNN	Configuração	TP	FP	FN	TN	Acurácia	F-Score
GoogLeNet	4b	67	8	7	96	91,57%	0,8993
ResNet	6b	60	7	14	97	88,20%	0,8511
VGG	<i>ImageNet – Adam</i>	63	13	9	93	87,64%	0,8514
AlexNet	1a	59	9	15	95	86,52%	0,8310
LeNet	1a	61	11	13	93	86,51%	0,8356
<i>Ensemble 3</i>	-	56	18	3	101	88,20%	0,8421

É interessante notar que, ao analisar os resultados obtidos com os comitês, percebe-se que o *Ensemble 2* supera o desempenho de todas as estratégias anteriormente consideradas, tornando a acurácia de 93,82% o novo patamar de melhor resultado obtido neste trabalho.

Esse resultado confirma as evidências da literatura quando discorre sobre a importância de se utilizar modelos bem treinados, mas com arquiteturas distintas entre si. Como supracitado, a GoogLeNet é o modelo com melhor resultado entre as abordagens individuais; a VGG16 é a rede com a maior quantidade parâmetros; e, a ResNet possui conexões residuais que a tornam a rede mais profunda de todas utilizadas, ainda que substancialmente menor devido ao uso de *pool* global médio em vez de camadas totalmente conectadas. Todas as divergências nas redes criam capacidades diferentes de absorção de características relevantes e produzem mapas de abstração diferentes que, quando reunidas no *ensemble*, conseguem obter juntas uma acurácia maior do que individualmente.

Os resultados obtidos aqui neste trabalho dão continuidade e superam resultados anteriores obtidos pelo autor no contexto de atividades de Iniciação Científica com a utilização das mesmas bases de dados sujeitas ao mesmo tipo de validação cruzada e com a mesma partição. Primeiramente, os melhores resultados até então obtidos contemplaram o uso de redes neurais artificiais *multi-layer perceptron*, com melhor acurácia observada em torno de 72% (EVANGELISTA; GUEDES, 2018b). Em seguida, foram avaliadas CNNs com arquiteturas obtidas conforme uma metodologia pré-estabelecida, as quais não seguiam arquiteturas canônicas. Nesta etapa, o melhor resultado contempla o uso de uma CNN com apenas uma camada convolucional, cerca de

66 milhões de parâmetros treináveis e desempenho aferido por acurácia igual a 88,76% (EVANGELISTA; GUEDES, 2018a). Como mencionado anteriormente, investigando a utilização das arquiteturas canônicas neste problema, os resultados deste trabalho de conclusão de curso culminaram em um modelo de agrupamento de redes com acurácia igual a 93,82%, uma melhoria percentual de 23,27% em relação ao primeiro trabalho e de 5,41% em relação ao segundo.

Jaeger et al. propuseram uma solução para diagnosticar TB a partir de segmentação pulmonar e extração de características relevantes com modelos de Aprendizagem Máquina, no qual obteve acurácia de 78,3% e 84% utilizando as bases de dados dos Conjuntos Montgmoery County e Shenzhen Hospital, respectivamente, ultrapassando a performance de especialistas humanos (JAEGER et al., 2014). Das 47 redes propostas e 4 comitês neste trabalho, 19 redes e 3 comitês superam os 84% de acurácia do trabalho supracitado, o que, por conseguinte, também supera o desempenho de especialistas humanos.

Como citado no Capítulo 3, Lakhani e Sundaram aplicaram quatro arquiteturas canônicas de CNNs e obtiveram 98,9% de acurácia. Ressalta-se, entretanto, que houve o uso de uma base de dados adicional, privada e oriunda da Universidade Thomas Jefferson (LAKHANI; SUNDARAM, 2017). Acredita-se que o presente trabalho não superou estas métricas de desempenho em virtude das características provenientes desta base privada, que pode ter fornecido novos exemplos com características cruciais para a melhoria na classificação geral.

Apesar disso, ressalta-se que o presente trabalho baseia-se inteiramente em bases de dados públicas e gratuitas e em arquiteturas canônicas de CNNs, o que favorece a sua reproduzibilidade. Embora o melhor desempenho obtido seja comparável ao estado da arte, não se tem a pretensão de que os resultados aqui apresentados venham a substituir a interpretação médica-radiológica humana. Outros trabalhos que avaliem os riscos e benefícios do uso destes modelos precisam ser cuidadosamente elaborados e avaliados.

Capítulo 6

Considerações Finais

O objetivo deste trabalho consistiu em avaliar o uso de redes neurais convolucionais no diagnóstico de tuberculose em imagens radiográficas torácicas. Para cumprir esse objetivo, primeiramente, foi consolidado um conjunto de dados com 893 imagens de bases diferentes, que apresentam características representativas da doença. Todas as imagens e seus respectivos metadados foram analisados, tratados e guardados para a etapa de aprendizagem.

Cumprida esta etapa, foram treinadas topologias canônicas de CNNs profundas, conhecidas pela alta acurácia em tarefas de Visão Computacional, são elas: *LeNet*, *AlexNet*, GoogLeNet, VGG16 e ResNet. Estabelecendo configurações diversas para parâmetros e hiperparâmetros, treinamento com e sem *Transfer Learning*, com e sem *data augmentation*, foi possível ter, no total, 47 variações dessas redes e 4 comitês com as melhores redes. Após a etapa de treinamento e teste, obteve-se 19 arquiteturas e 3 comitês que superaram valores alcançados pela literatura e por especialistas humanos. A CNN com o melhor desempenho individual, segundo os critérios de avaliação estipulados, obteve uma acurácia de 91,57% e F-score de 0,8993 para esta tarefa. Ao mesmo tempo, o melhor resultado geral foi observado em um *ensemble*, que alcançou uma acurácia de 93,82% e F-Score de 0,9252%.

Ressalta-se que o presente trabalho baseia-se inteiramente em bases de dados públicas e gratuitas e em arquiteturas canônicas de CNNs, o que favorece a sua reproduzibilidade. Embora o melhor desempenho obtido seja comparável ao estado da arte, não se tem a pretensão de que os resultados aqui apresentados venham a substituir a interpretação médico-radiológica

humana. Outros trabalhos que avaliem os riscos e benefícios do uso destes modelos precisam ser cuidadosamente elaborados e avaliados.

Para o graduando em Engenharia de Computação, este trabalho de conclusão de curso permitiu dar continuidade ao tema abordado na Iniciação Científica 2017-2018, obtendo melhorias na resolução do problema considerado. Do ponto de vista de sua formação, além de considerar métodos e técnicas de *Deep Learning*, colabora com estratégias que visam a minimização de um problema de saúde de escala global.

Em trabalhos futuros, sugere-se avaliar o desempenho dos modelos propostos em casos de tuberculose regionais e nacionais, visando aferir se as características aprendidas por estes modelos mostram-se relevantes para as manifestações locais desta enfermidade. Neste sentido, ressalta-se a importância de consolidar bases de dados de imagens médicas brasileiras para este tipo de tarefa.

Referências Bibliográficas

- AIT-KHALED, N. et al. *Tuberculosis: a manual for medical students*. Geneva: World Health Organization, 2003.
- BARBERIS, I. et al. The history of tuberculosis: from the first historical records to the isolation of Koch's bacillus. *Journal of preventive medicine and hygiene*, Pacini Editore, v. 58, n. 1, p. E9, 2017.
- BRAGA, A. de P.; CARVALHO, A. P. de Leon F. de; LUDELMIR, T. B. *Redes Neurais Artificiais: Teoria e Aplicações*. 1. ed. São Paulo, SP: LTC, 2000.
- BRASIL Ministério da Saúde do. *Brasil Livre da Tuberculose*. Brasil: Ministério da Saúde do Brasil, 2016. Disponível em <<http://portalarquivos.saude.gov.br/images/pdf/2017/fevereiro/24/Plano-Nacional-Tuberculose.pdf>>. Acessado em 10 de junho de 2019.
- BRASIL Ministério da Saúde do. *Panorama da Tuberculose no Brasil*. Brasil: Ministério da Saúde do Brasil, 2016. Disponível em <http://bvsms.saude.gov.br/bvs/publicacoes/panorama%20tuberculose%20brasil_2014.pdf>. Acessado em 10 de junho de 2019.
- BUDUMA, N.; LOCASCIO, N. *Fundamentals of deep learning: Designing next-generation machine intelligence algorithms*. Sebastopol, CA, USA: "O'Reilly Media, Inc.", 2017.
- CAMPOS, T. E. de. *Problemas de generalização*. São Paulo, SP: Universidade de São Paulo, 2000. Disponível em <<http://www.vision.ime.usp.br/~teo/publications/qualificacao/node8.html>>. Acessado em 10 de junho de 2019.
- CHOLLET, F. *Deep Learning with Python*. 1. ed. Shelter Island, New York: Manning Publications, 2017.
- CROFTON, J. *Clinical Tuberculosis*. 3a. ed. London, England: Macmillan, 2009.
- DELICATO, F. C.; PIRES, P. F.; SILVEIRA, I. F. *ANAIIS - Jornadas de Atualização em Informática 2017*. Porto Alegre, RS: Sociedade Brasileira de Computação, 2017.
- DENG, L.; YU, D. *Deep Learning: Methods and Applications*. 1a. ed. Boston, MA, USA: Now Publishers, 2014. (Foundations and Trends in Signal Processing 7.3-4).
- EL-SOLH, A. A. et al. Predicting active pulmonary tuberculosis using an artificial neural network. *Chest*, Elsevier, v. 116, n. 4, p. 968–973, 1999.
- ELVEREN, E.; YUMUŞAK, N. Tuberculosis disease diagnosis using artificial neural network trained with genetic algorithm. *Journal of medical systems*, Springer, v. 35, n. 3, p. 329–332, 2011.

- ER, O.; TEMURTAS, F.; TANRIKULU, A. Ç. Tuberculosis disease diagnosis using artificial neural networks. *Journal of medical systems*, Springer, v. 34, n. 3, p. 299–302, 2010.
- EVANGELISTA, L. G. C.; GUEDES, E. B. Computer-aided tuberculosis detection from chest x-ray images with convolutional neural networks. In: *Anais do XV Encontro Nacional de Inteligência Artificial e Computacional*. IBM Research Brasil, São Paulo: Sociedade Brasileira de Computação, 2018. p. 1–10.
- EVANGELISTA, L. G. C.; GUEDES, E. B. Detecção de tuberculose em imagens médicas com redes neurais artificiais. In: *Anais do VII Encontro Regional de Computação e Sistemas de Informação*. Manaus, Amazonas: FUCAPI, 2018. p. 1–9.
- FACELI, K. et al. *Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina*. 1a. ed. Rio de Janeiro, RJ: LTC, 2011.
- GALINDO, Y.; LORENA, A. C. Deep transfer learning for meteor detection. In: *Anais do XV Encontro Nacional de Inteligência Artificial e Computacional*. São Paulo, Brasil: SBC, 2018. p. 528–537.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. Cambridge: The MIT Press, 2016. v. 1.
- GULLI, A.; PAL, S. *Deep learning with Keras*. 1a. ed. Birmingham, UK: Packt Publishing, 2017.
- HAYKIN, S. O. *Neural Networks and Learning Machines*. 3. ed. Upper Saddle River, New Jersey, USA: Prentice Hall, 2008.
- HE, K. et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 770–778.
- HEEMSKERK, D. et al. *Tuberculosis in Adults and Children*. 1a. ed. New York, NY, USA: Springer International Publishing, 2015. (SpringerBriefs in Public Health 2).
- HINTON, G.; SRIVASTAVA, N.; SWERSKY, K. *Overview of mini-batch gradient descent*. Toronto, Canadá: Universidade de Toronto, 2015. Disponível em <http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf>. Acessado em 10 de junho de 2019.
- HWANG, S. et al. A novel approach for tuberculosis screening based on deep convolutional neural networks. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *Medical Imaging 2016: Computer-Aided Diagnosis*. Bellingham, Washington, USA, 2016. v. 9785, p. 97852W.
- IMIANVAN, A. A.; OBI, J. Fuzzy cluster means expert system for the diagnosis of tuberculosis. *Global Journal of Computer Science & Technology*, Global Journals Inc., v. 11, n. 6, p. 41–48, 2011.
- JAEGER, S. et al. Two public chest X-ray datasets for computer-aided screening of pulmonary diseases. *Quantitative Imaging in Medicine and Surgery*, v. 6, n. 4, p. 475–477, 2014.

- JAEGER, S. et al. Automatic screening for tuberculosis in chest radiographs: a survey. *Quantitative imaging in medicine and surgery*, AME Publications, v. 3, n. 2, p. 89, 2013.
- JAMISON, D. T. et al. *Disease Control Priorities in Developing Countries*. 2. ed. Washington, D.C., USA: World Bank Publications, 2006. 289–310 p.
- KHAN, S. et al. *A Guide to Convolutional Neural Networks for Computer Vision*. 1a. ed. San Rafael, California, USA: Morgan & Claypool, 2018. (Synthesis Lectures on Computer Vision).
- KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. Toronto, Ontario, Canada, 2012.
- KUBAT, M. *An Introduction to Machine Learning*. Estados Unidos: Springer, 2015.
- LAB, S. V. *IMAGENET*. Palo Alto, California, USA: Stanford University, 2016. Disponível em <<http://www.image-net.org/>>. Acessado em 10 de junho de 2019.
- LAKHANI, P.; SUNDARAM, B. Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology*, Radiological Society of North America, v. 284, n. 2, p. 574–582, 2017.
- LECUN, Y.; BENGIO, Y. Pattern recognition and neural networks. Murray Hill, New Jersey, New Providence, Nova Jersey, USA, 1995.
- LECUN, Y. et al. Lenet-5, convolutional neural networks. 2015.
- LIAO, S.-H. Expert system methodologies and applications — a decade review from 1995 to 2004. *Expert systems with applications*, Elsevier, v. 28, n. 1, p. 93–103, 2005.
- LITJENS, G. et al. A survey on deep learning in medical image analysis. *Medical image analysis*, Elsevier, v. 42, p. 60–88, 2017.
- MARSLAND, S. *Machine Learning – An algorithmic perspective*. 2. ed. Estados Unidos: CRC Press, 2015.
- MCBEE, M. P. et al. Deep learning in radiology. *Academic radiology*, Elsevier, 2018.
- MOUNT, D. B.; POLLAK, M. R. *Molecular and Genetic Basis of Renal Disease E-Book: A Companion to Brenner and Rector's The Kidney*. Palo Alto, California, USA: Elsevier Health Sciences, 2007.
- PAHO. *OMS publica orientação ética para proteger direitos dos pacientes com tuberculose*. 2017. Organização Pan-Americana da Saúde. Disponível em <<https://goo.gl/Gyb6qU>>. Acessado em 10 de junho de 2019.
- RAJPURKAR, P. et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.
- RAVI, D. et al. Deep learning for health informatics. *IEEE Journal of Biomedical and Health Informatics*, v. 21, n. 1, p. 4–21, 2017.

- ROJAS, R. *Neural Networks: A Systematic Introduction*. 1. ed. New York, NY, USA: Springer, 1996.
- RUDER, S. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.
- RUSSEL, S.; NORVIG, P. *Artificial Intelligence – A Modern Approach*. 3. ed. Nova Jersey: Prentice Hall, 2010.
- SARKAR, D.; BALI, R.; GHOSH, T. *Hands-On Transfer Learning with Python Implement Advanced Deep Learning and Neural Network Models Using TensorFlow and Keras*. 1. ed. Livery Place 35, Livery Street, Birmingham: Packt Publishing, 201.
- SHIRAI SHI, J. et al. Development of a digital image database for chest radiographs with and without a lung nodule: Receiver operating characteristic analysis of radiologists' detection of pulmonary nodules. *AJR*, n. 174, p. 71–74, 2000.
- SILVA, I. N. da et al. *Artificial Neural Networks – A practical course*. Suíça: Springer, 2017.
- SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- SRIVASTAVA, N. et al. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, JMLR.org, v. 15, n. 1, p. 1929–1958, 2014.
- SZEGEDY, C. et al. Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Boston, Massachusetts, USA: IEEE, 2015. v. 1, p. 1–9.
- WHO. *Global Tuberculosis Report 2017*. Geneva, Switzerland: World Health Organization, 2017. Disponível em <http://www.who.int/tb/publications/global_report/en/>. Acessado em 10 de junho de 2019.
- XIAO, H.; RASUL, K.; VOLLGRAF, R. *Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms*. 2017.

Apêndice A

Exemplos de *Data Augmentation*

A seguir imagens de raio-X e suas variações, em cada *dataset*, que exemplificam os dois tipos de *data augmentation* utilizados neste trabalho: **hard** e **soft**. As técnicas de *augmentation* foram realizadas até que fosse alcançada a quantidade de 10.000 exemplos por abordagem.

A.1 *Data Augmentation Hard*

A primeira abordagem de aumento da base de dados consistiu em espelhamento vertical, rotação em 90°, rotação em 270°, rotação de 0° a 10° para esquerda ou direita.

Figura A.1: *Data augmentation Hard* com exemplo positivo para TB do Conjunto Shenzhen Hospital X-ray.

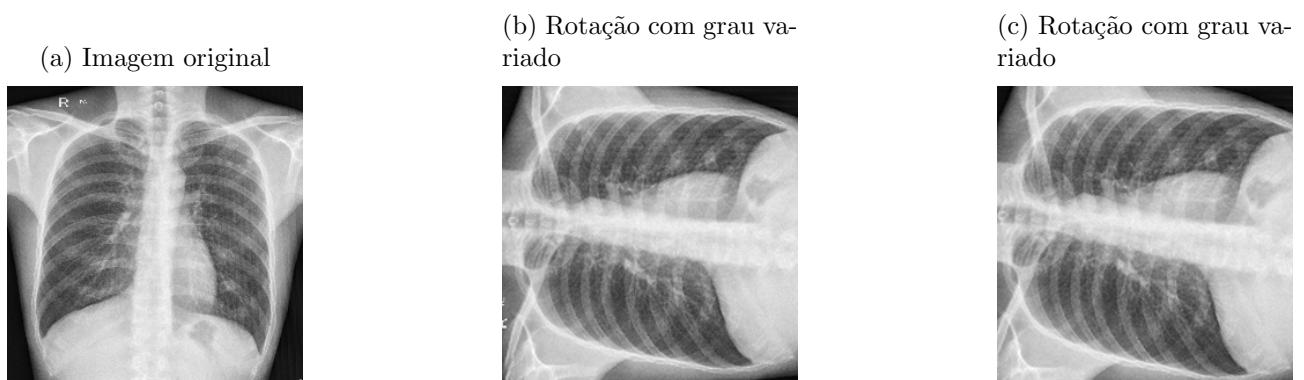


Figura A.2: *Data augmentation Hard* com exemplo negativo para TB da Base de dados JSRT.

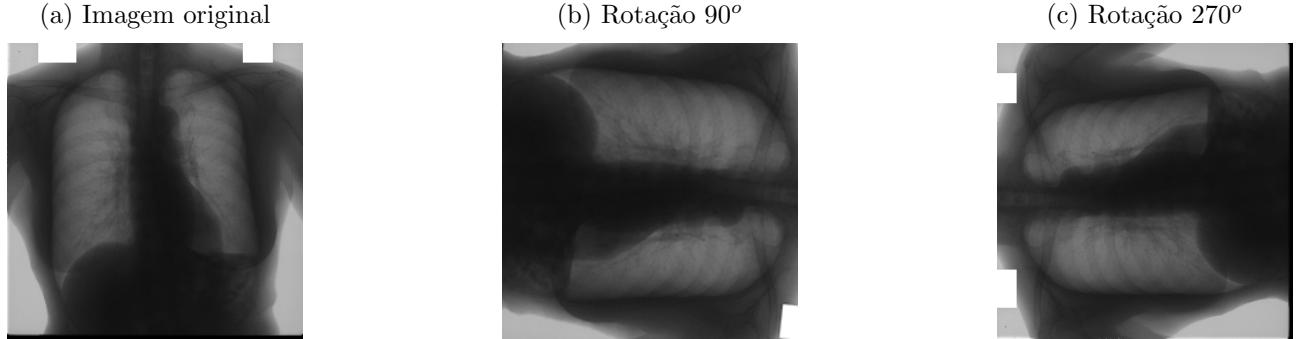
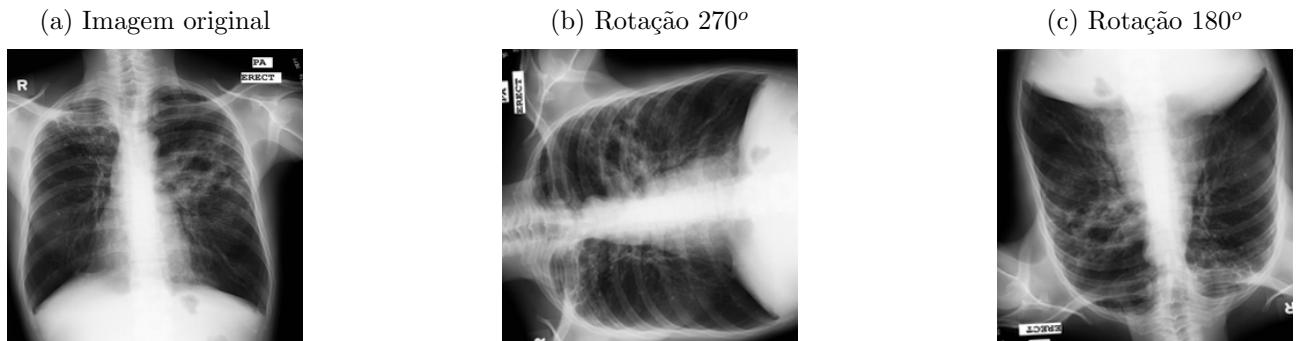


Figura A.3: *Data augmentation Hard* com exemplo positivo para TB do Conjunto Montgomery County X-ray.



A.2 Data Augmentation Soft

Diferentemente da primeira tentativa, a segunda abordagem contemplou variações mais sutis das imagens de entrada, na busca por conseguir reproduzir melhor cenários reais de um profissional de saúde ao analisar uma imagem médica. As técnicas utilizadas foram de rotações entre 0° a 10° para esquerda ou direita.

Figura A.4: *Data augmentation Soft* com exemplo positivo para TB do Conjunto Shenzhen Hospital X-ray.



Figura A.5: *Data augmentation Soft* com exemplo negativo para TB da Base de dados JSRT.

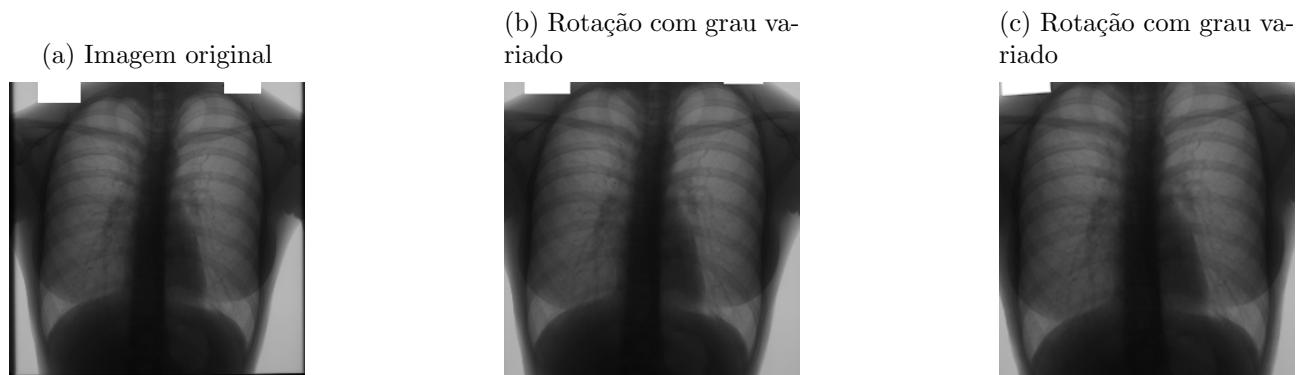


Figura A.6: *Data augmentation Soft* com exemplo positivo para TB do Conjunto Montgomery County X-ray.

