UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Estudo de representações de imagens de múltiplos domínios a partir de aprendizado profundo não supervisionado e semi-supervisionado

Gabriel Biscaro Cavallari

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC)

ICMC USP
SÃO CARLOS

Gabriel Biscaro Cavallari

Estudo de representações de imagens de múltiplos domínios a partir de aprendizado profundo não supervisionado e semi-supervisionado

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Moacir Antonelli Ponti

USP – São Carlos
Julho de 2022

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

C377e

Cavallari, Gabriel Biscaro
Estudo de representações de imagens de múltiplos domínios a partir de aprendizado profundo não supervisionado e semi-supervisionado / Gabriel Biscaro Cavallari; orientador Moacir Antonelli Ponti. -- São Carlos, 2022.
65 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2022.

1. aprendizado de características. 2. aprendizado profundo. 3. aprendizado semi-supervisionado. 4. aprendizado não supervisionado. 5. auto-supervisão. I. Ponti, Moacir Antonelli, orient. II. Título.

Bibliotecários responsáveis pela estrutura de catalogação da publicação de acordo com a AACR2:
Gláucia Maria Saia Cristianini - CRB - 8/4938
Juliana de Souza Moraes - CRB - 8/6176

Gabriel Biscaro Cavallari

A study of image representations from multiple domains using unsupervised and semi-supervised deep learning

Dissertation submitted to the Instituto de Ciências Matemáticas e de Computação – ICMC-USP – in accordance with the requirements of the Computer and Mathematical Sciences Graduate Program, for the degree of Master in Science. *FINAL VERSION*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Moacir Antonelli Ponti

USP – São Carlos
July 2022

Este trabalho é dedicado às crianças adultas que,
quando pequenas, sonharam em se tornar cientistas.
Em especial, ao pesquisadores do Instituto de Ciências Matemáticas e de Computação (ICMC).

AGRADECIMENTOS

Agradeço aos meus pais por sempre me apoiarem nos meus estudos, e também ao meu irmão Henrique, por todo o companheirismo.

Agradeço ao meu orientador Moacir pelo conhecimento, paciência e atenção.

A todos os amigos e colegas que de alguma forma me auxiliaram e me motivaram.

Agradeço também a FAPESP pelo financiamento desta pesquisa, através do processo 2019/02033-0.

RESUMO

CAVALLARI, G. B. **Estudo de representações de imagens de múltiplos domínios a partir de aprendizado profundo não supervisionado e semi-supervisionado**. 2022. 65 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

Sistemas atuais de visão computacional demonstram excelente desempenho em uma variedade de *benchmarks*, como detecção de objetos, reconhecimento e segmentação semântica de imagens. O treinamento dessas redes segue principalmente o paradigma de aprendizado supervisionado, em que são necessários muitos pares de entrada-saída para o treinamento. No entanto, grandes quantidades de dados rotulados manualmente são custosos e complexos de obter. Portanto, o aprendizado sem a necessidade de dados anotados é de grande importância para aproveitar a grande quantidade de dados visuais não rotulados geralmente disponíveis. Para enfrentar esse desafio, métodos de aprendizado não supervisionado e semi-supervisionado podem auxiliar na utilização de dados não rotulados para reduzir a dependência de grandes conjuntos de dados rotulados. Esta pesquisa tem como objetivo investigar diferentes arquiteturas e estratégias de treinamento que consideram uma situação em que se tem apenas dados não rotulados e dados rotulados limitados. Nossa hipótese é que essa estratégia melhora a generalização e a discriminação do espaço de características aprendido. Por meio de tarefas auxiliares, diferentes bases de dados e experimentos extensivos, concluímos que tanto o aprendizado semi-supervisionado quanto o auto-supervisionado seguido de ajuste fino geram representações discriminativas. Ainda, que essas representações tendem a ser mais robustas à ataques quando comparadas àquelas aprendidas em contextos puramente supervisionados.

Palavras-chave: aprendizado de características, aprendizado profundo, aprendizado semi-supervisionado, aprendizado não supervisionado, auto-supervisão.

ABSTRACT

CAVALLARI, G. B. **A study of image representations from multiple domains using unsupervised and semi-supervised deep learning**. 2022. 65 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

Modern computer vision systems demonstrate outstanding performance on a variety of challenging benchmarks, such as object detection, image recognition and semantic image segmentation. Training of such networks follows mostly the supervised learning paradigm, where sufficiently many input-output pairs are required for training. However, massive amounts of manually labeled data is both expensive and impractical to scale. Therefore, learning without requiring manual annotation effort is of crucial importance in order to successfully take advantage of the vast amount of unlabeled visual data that is available today. To address this challenge, unsupervised and semi-supervised learning methods could be a powerful paradigm for leveraging unlabeled data to mitigate the reliance on large labeled datasets. This research aims to investigate different architectures and training strategies that considers both unlabeled and limited labeled data. Our hypothesis is that this strategy improves the generalization and discrimination of the learned feature space. Through auxiliary tasks, different datasets and extensive experiments, we concluded that both semi-supervised and self-supervised learning followed by fine-tuning generate discriminative representations. Furthermore, these representations tend to be more robust to attacks when compared to those learned in purely supervised contexts.

Keywords: feature learning, deep learning, semi-supervised learning, unsupervised learning, self-supervision.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

CAPÍTULO

1

INTRODUÇÃO

A performance de tarefas que envolvem a classificação, agrupamento, ou busca de imagens é altamente dependente da escolha da representação de dados (ou *features*) para essas tarefas. Por esta razão, muito do esforço da área de reconhecimento de padrões visuais tem sido no projeto de métodos de pré-processamento e transformações dos dados originais que resultem em uma representação de dados (características) adequada para a tarefa em questão (BENGIO; COURVILLE; VINCENT, 2013).

Quando se trata de reconhecimento de padrões visuais é preciso definir qual conjunto de informações é mais relevante. Métodos de extração de características manuais têm a tarefa de gerar um espaço no qual são codificadas de maneira explícita: cor, textura, orientação, forma, entre outras características possíveis. Cada imagem será então representada por um vetor nesse espaço (PONTI; NAZARÉ; THUMÉ, 2016). Podemos ver esse processo como um mapeamento da matriz de pixels imagem X em um espaço com m dimensões, ou seja, uma função $f : X \rightarrow \mathbb{R}^m$. Definir esse mapeamento exige conhecimento acerca do problema em questão e escolher dentre uma grande variedade de possíveis descritores.

Uma abordagem alternativa para a extração de características manual é a utilização de métodos de Deep Learning por meio de redes como as CNNs (Convolutional Neural Networks), que são utilizadas em visão computacional com grande sucesso para uma grande variedade de tarefas (RAZAVIAN *et al.*, 2014). CNNs são compostas basicamente de camadas convolucionais, as quais processam matrizes ao invés de vetores. Quando são projetadas como classificadores de imagens, essas ainda possuem uma ou mais camadas densas no final, sendo a última uma camada que produz a distribuição de probabilidade para as classes do problema. Assim, a CNN aprende representações intermediárias da imagem de entrada via camadas convolucionais, as quais são então submetidas a uma camada densa que irá ser responsável por classificar a imagem. Seu treinamento é feito otimizando uma função objetivo de classificação, a qual depende dos rótulos. A vantagem desse método é que as representações são aprendidas diretamente a partir dos dados,

e portanto o modelo transforma a imagem de entrada de forma a obter as características mais discriminativas (GOODFELLOW *et al.*, 2016).

Porém, nem sempre temos disponíveis anotações para todas as classes em um problema de aprendizado de máquina. Apesar dos bons resultados obtidos com CNNs em tarefas de classificação, essas requerem um grande número de exemplos rotulados de treinamento, comumente da ordem de centenas de milhares (MELLO; FERREIRA; PONTI, 2017). O aprendizado profundo por meio exclusivamente de dados supervisionados, em particular classificação também tem desvantagens como o a capacidade de aprender as características da imagem mesmo não associadas ao conceito principal (NAZARÉ *et al.*, 2017). Assim, grande parte das atuais pesquisas em aprendizado de máquina têm focado em aprender boas características (ou *features*) de dados não rotulados (COATES; NG; LEE, 2011). Além disto, dados rotulados para aprendizado de máquina são muitas vezes difíceis e custosos de se obter. A utilização de dados não rotulados pode ser vista como uma promessa em termos de expandir vastamente a aplicabilidade de métodos de aprendizado (RAINA *et al.*, 2007).

Algoritmos para aprendizado não supervisionado podem ser utilizados para melhorar a performance de tarefas de classificação supervisionadas. Como apresentado em Raina *et al.* (2007), uma opção seria aprender a representação visual utilizando uma base de dados grande e não rotulada. Em seguida, utilizar o modelo não supervisionado aprendido para extrair características de um conjunto de dados rotulados e realizar uma tarefa de classificação. Uma vez que a representação foi aprendida no primeiro passo, ela pode ser aplicada repetidamente para diferentes tarefas de reconhecimento. Na prática, o pré-treinamento de redes neurais profundas é uma técnica comumente recomendada e provê comumente melhores resultados do que o aprendizado a partir de pesos aleatórios (em inglês, *from scratch*). Porém na metade final da década de 2010 tornou-se padrão o uso de redes neurais pré-treinadas a partir de grandes bases de dados supervisionadas – em particular ResNet e Inception projetadas para o dataset ImageNet – e então utilizadas como *backbones* em sistemas e arquiteturas diferentes (PONTI *et al.*, 2021).

Neste contexto, o uso de métodos não supervisionados ou semi-supervisionados é importante. A princípio, tarefas como a de reconstrução, como *autoencoders*, podem ser utilizadas para um pré-treinamento inicial seguido de ajuste fino ou então um treinamento em conjunto com uma função de custo supervisionada. Essa técnica foi considerada inicialmente nesse projeto, visto que são métodos baseados na tarefa de reconstrução (também chamados de de geradores). Porém como estes modelos também têm como objetivo gerar imagens realistas, eles acabam ignorando a capacidade de discriminar classes, e portanto métodos baseados em reconstrução passaram a ser menos utilizados a partir do final da década de 2010, em favor de métodos contrastivos ou de tarefas de auto-supervisão para o aprendizado de características (DONAHUE; SIMONYAN, 2019).

Em termos de métodos não supervisionados, tarefas de auto-supervisão (*self-supervision*) atingem atualmente estado-da-arte para o aprendizado de características (JING; TIAN, 2019).

Dentre as diversas tarefas de auto-supervisão existentes, exploramos duas específicas para este trabalho: a tarefa de predição de rotação (GIDARIS; SINGH; KOMODAKIS, 2018) e o método Barlow Twins (ZBONTAR *et al.*, 2021b). A tarefa de previsão de rotação se mostrou um método simples e funcional em relação aos métodos iniciais de auto-supervisão. Porém, estratégias mais recentes se mostraram mais robustas, entre elas a tarefa Barlow Twins.

Há na literatura uma lacuna com relação a dois pontos principais: 1) um estudo mais detalhado dos espaços de características formados por diferentes tarefas de auto-supervisão em comparação com as CNNs em termos da sua capacidade de separabilidade linear e robustez à ataques; 2) de cenários em que poucos rótulos estão disponíveis, mas que poderiam ser úteis em trazer mais informações para a representação aprendida pelos modelos de auto-supervisão;

Assim, este trabalho visa investigar duas diferentes tarefas de auto-supervisão e a comparação das características obtidas por estes modelos com modelos supervisionados, bem como uma estratégia de treinamento em conjunto simultâneo semi-supervisionado que pode ser beneficiado quando há poucos exemplos rotulados disponíveis.

1.1 Hipótese

A hipótese geral desta dissertação de mestrado é que a utilização de funções de custo auxiliares à tarefa classificação promove, em cenários de supervisão limitada, representações mais discriminativas e mais robustas com relação à ataques. Em particular, investigaremos tarefas de auto-supervisão, com a premissa de que tais tarefas permitam pré-treinar modelos de forma a obter boas representações iniciais para posterior transferência de aprendizado a tarefas supervisionadas.

Investigaremos dois cenários principais: (1) aprendizado auto-supervisionado seguido de ajuste-fino (*fine-tuning*) para tarefas de classificação e (2) aprendizado semi-supervisionado no qual as duas tarefas (classificação, com rótulos, e auto-supervisão, sem rótulos) são aprendidas em conjunto. Investigaremos a hipótese de que os cenários (1) e (2) obtém representações com melhor separação linear e robustez a ataques quando comparado ao uso apenas do aprendizado supervisionado.

1.2 Contribuições

Investigamos as tarefas auto-supervisionadas de predição de rotação e a tarefa Barlow Twins em três bases de dados de imagens de domínios diferentes (STL-10, Fashion-MNIST e Malaria). Exploramos diversas combinações de aprendizado supervisionado e auto-supervisionado, avaliando a separabilidade linear das representações aprendidas, além de investigar a robustez dos modelos a ataques de 1 pixel.

De modo geral, entre todos os estilos de treinamento explorados, o modelo semi-supervisionado foi o que obteve melhor acurácia, seguido pelos modelos que utilizaram *fine-tuning* de uma tarefa auto-supervisionada como pré-treinamento. Mostramos que os modelos que combinam tarefas supervisionadas com auto-supervisionadas podem aproveitar os dados não rotulados para melhorar a representação aprendida em termos de discriminação linear, além de permitir o aprendizado mesmo sob ataque.

Além disso, discutimos as escolhas em termos de auto-supervisão e casos de falha considerando os diferentes conjuntos de dados. Verificamos que diferentes tarefas de auto-supervisão têm desempenhos diferentes dependendo da base de dados, e observamos que a escolha da tarefa deve levar em consideração a natureza da base de dados. Os resultados dos experimentos realizados reforçam a hipótese inicial do trabalho.

1.3 Organização

Após essa introdução, são descritos conceitos fundamentais e trabalhos relacionados no Capítulo 2, mostrando as direções atuais da literatura no campo de aprendizado de características e levantando possíveis lacunas. A seguir, o Capítulo 3 descreve o método utilizado, bases de dados e configuração dos experimentos. Finalmente, os resultados são apresentados no Capítulo 4.

CAPÍTULO

2

CONCEITOS FUNDAMENTAIS E TRABALHOS RELACIONADOS

Neste capítulo são apresentados os conceitos utilizados na literatura relacionada a redes neurais profundas. Em particular, utilizaremos redes neurais convolucionais (CNNs), arquiteturas utilizadas tanto no aprendizado supervisionado como no aprendizado não supervisionado e semi-supervisionado. Para o aprendizado não supervisionado, utilizaremos tarefas de auto-supervisão (*self-supervision*). Adicionalmente, as *Support Vector Machines* (SVM, em português Máquinas de Vetores de Suporte) serão utilizadas para avaliar os espaços de características obtidos. Serão apresentados também os trabalhos relacionados sobre aprendizado de características de forma não supervisionada e semi-supervisionada.

2.1 Redes neurais convolucionais (CNNs)

Redes neurais convolucionais remetem a estrutura em camadas, comumente: convoluções, *poolings* e camadas totalmente conectadas. CNNs são normalmente utilizadas para classificação e, portanto, requerem que sejam fornecidos rótulos para as classes dos exemplos.

CNNs são organizadas de maneira a produzir transformações dos dados de entrada de forma sucessiva, até a última camada que produzirá a classificação. Iniciando com uma sequência de camadas convolucionais, por vezes também são aplicados operadores de redução de dimensionalidade espacial (ou *pooling*). Próximo ao final da rede, é possível ter uma ou mais camadas totalmente conectadas, as quais têm como objetivo aprender os pesos responsáveis por classificar a representação aprendida. A última camada de uma CNN é usualmente a camada para a qual se computa função de custo do tipo *cross-entropy*, empregando o classificador conhecido por *softmax*, que dá a distribuição de probabilidades associadas a cada uma das C classes, relativa ao exemplo de entrada (GOODFELLOW *et al.*, 2016).

Considerando um único exemplo cuja distribuição de probabilidade de classes real é

$\mathbf{y} = y_1, y_2, \ldots y_C$ e a predição é dada pela rede neural pela função $f(\mathbf{x}) = \hat{\mathbf{y}}$, temos a soma das entropias cruzadas (entre a predição e a classe real) de cada classe j:

$$\ell^{(\text{ce})} = -\sum_j y_j \cdot \log(\hat{y}_j + \varepsilon), \tag{2.1}$$

onde $\varepsilon << 1$ é uma variável para evitar $\log(0)$, e.g. $\varepsilon = 10^{-4}$.

Redes neurais profundas são formadas por camadas densas e/ou convolucionais. Nas camadas densas, ou totalmente conectadas, cada neurônio possui um peso associado a cada elemento do vetor de entrada. A Figura 1 ilustra uma rede neural densa.

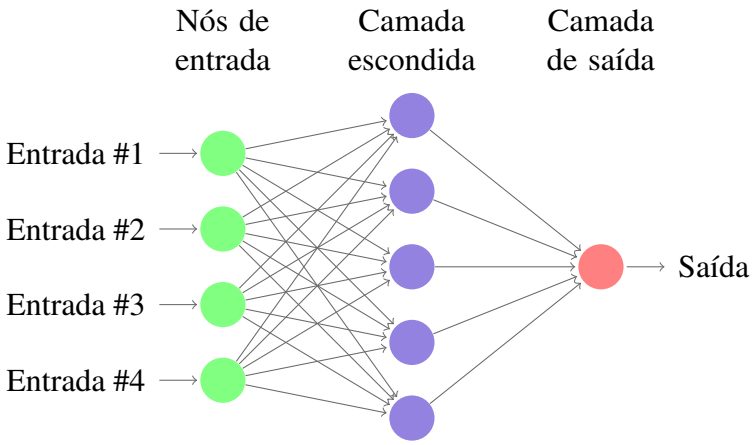


Figura 1 – Exemplo de uma rede neural com uma camada escondida densa.

Na camada convolucional cada neurônio é um filtro aplicado a uma imagem de entrada e cada filtro é uma matriz de pesos. A Figura 2 descreve o que é uma convolução.

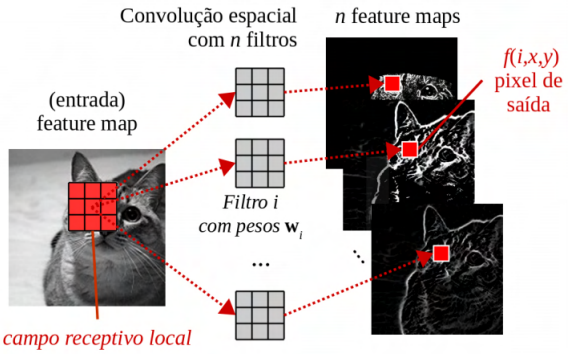


Figura 2 – Ao utilizar convolução, informações locais são processadas considerando cada posição (x, y) como centro: essa região é chamada de campo receptivo. Seus valores são então usados como entrada para um filtro i com parâmetros w_i, produzindo um único valor (pixel) no mapa de características $f(i,x,y)$ gerado como saída. Fonte: (PONTI; COSTA, 2018)

2.2 Support Vector Machines

Descreveremos brevemente o classificador *Support Vector Machines* (SVM), pois esse será utilizado como analisador dos espaços de características gerados pelos experimentos deste

trabalho. O SVM encontra a separação entre classes de um problema de classificação por meio de hiperplanos que definem as melhores fronteiras com base no critério de máxima margem. Como mostrado na Figura 3, ele encontra o hiperplano separador com base na maior margem entre os vetores de suporte, os quais são os exemplos de cada classe mais próximos ao hiperplano.

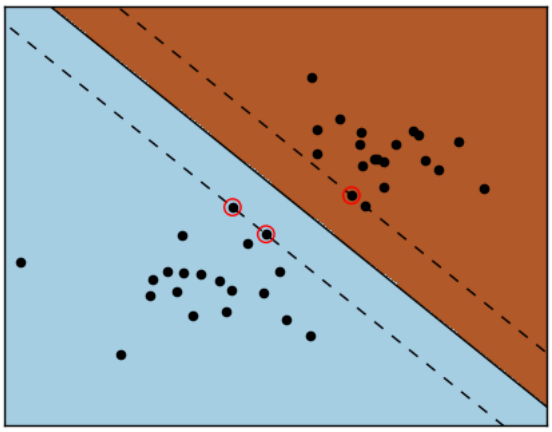


Figura 3 – Ilustração de separação através de um classificador SVM. Os pontos circulados em vermelho representam os vetores de suporte. A linha não tracejada representa o hiperplano separador.

Considerando um problema linearmente separável de duas classes, o SVM busca pelos vetores de suporte das classes positiva e negativa, respectivamente identificados por $\mathbf{x}_+$ e $\mathbf{x}_-$, de forma que:

$$|<\mathbf{w},\mathbf{x}_+>+b| = 1 \tag{2.2}$$

e

$$|<\mathbf{w},\mathbf{x}_->+b| = 1 \tag{2.3}$$

em que b é um fator constante para deslocar eventuais distâncias de forma a garantir a unidade da solução.

Com base nessa ideia, o SVM minimiza a seguinte função, sujeita a restrição $\mathbf{y_i}(<\mathbf{w},\mathbf{x_i}>+b) \geq 1$:

$$\min_{w,b} \frac{1}{2}||\mathbf{w}||^2 \tag{2.4}$$

Como o SVM é o classificador que encontra separações lineares, a acurácia de sua classificação pode ser vista como uma medida de separabilidade linear entre as classes dado um espaço de características. Utilizamos esse fato como premissa no desenvolvimento do trabalho.

2.3 Aprendizado de representações

O objetivo do aprendizado de representações é encontrar transformações dos dados de entrada que facilitem a extração de informações úteis ao criar classificadores ou outros preditores (BENGIO; COURVILLE; VINCENT, 2013). Em modelos probabilísticos, uma boa

representação é aquela que é capaz de capturar a distribuição dos fatores estruturais presentes nos dados de entrada observados. No caso de CNNs treinadas com rótulos, para extrair um vetor de características normalmente se utilizam as ativações da penúltima camada (anterior à *softmax*) como o vetor de características. Quando não temos os rótulos disponíveis, ainda é possível treinar modelos que sejam capazes de extrair vetores de características dos dados.

2.3.1 Não supervisionado

O aprendizado de representações de forma não supervisionada é um problema fundamental em aprendizado de máquina pois permite obter representações explorando propriedades dos dados diretamente, sem anotação.

2.3.1.1 Baseado em reconstrução

Chamados também de geradores, esses métodos utilizam os dados sem rótulos ao utilizar uma função objetivo de reconstrução. Um método clássico são os *Autoencoders* (HINTON; SALAKHUTDINOV, 2006), em que uma rede é treinada para reconstruir uma imagem de entrada, usando uma camada restrita na representação para forçar a abstração. Uma variação são os *Denoising Autoencoders* (VINCENT *et al.*, 2008), que treinam uma rede para desfazer ruídos aleatórios aplicados nos dados de entrada. Existem também técnicas para modelar a distribuição de probabilidade de imagens em redes profundas. Por exemplo, *Variational Autoencoders* (VAEs) (KINGMA; WELLING, 2013) empregam uma abordagem Bayesiana variacional para modelar a distribuição dos dados, dando origem a outras variações como VQ-VAE (OORD; VINYALS; KAVUKCUOGLU, 2017) e VQ-VAE-2 (RAZAVI; OORD; VINYALS, 2019). Outros modelos probabilísticos incluem *Restricted Boltzmann Machine* (RBMs) (SMOLENSKY, 1986) e *Deep Boltzmann Machines* (DBMs) (SALAKHUTDINOV; HINTON, 2009).

Alguns métodos recentes foram inspirados por *Generative Adversarial Networks* (GANs) (GOODFELLOW *et al.*, 2014). GANs geralmente consistem em dois tipos de redes: um gerador que gera imagens a partir de vetores latentes e um discriminador cuja função é distinguir se uma imagem foi gerada pela rede geradora. Com isto, o discriminador força o gerador a gerar imagens realistas, enquanto o gerador força o discriminador a melhorar sua capacidade de diferenciação. Durante o treinamento, as duas redes competem entre si e se fortalecem. Trabalhos posteriores incluem DCGAN (RADFORD; METZ; CHINTALA, 2015), BiGAN (DONAHUE; KRÄHENBÜHL; DARRELL, 2016), WGAN (ARJOVSKY; CHINTALA; BOTTOU, 2017), entre outros.

No entanto, para gerar imagens realistas, esses modelos devem prestar atenção significativa aos detalhes de baixo nível da imagem, e potencialmente ignorar a semântica de alto nível, como a capacidade de discriminar classes. As representações aprendidas através de modelos geradores têm sido usadas de diversas formas. Apesar dos primeiros sucessos no uso de GANs para o aprendizado não-supervisionado de representações, devido a problemas de treinamento

elas foram substituídas por abordagens baseadas em outras tarefas de auto-supervisão para aprendizado de características conforme constatado em Donahue e Simonyan (2019).

Mais especificamente avaliados no cenário de aprendizado de representações temos os trabalhos *Context Encoders* (PATHAK *et al.*, 2016) em que os autores propõem uma tarefa inspirada em pintura de imagem, em que um modelo é treinado para gerar o conteúdo de uma região removida arbitrariamente de uma imagem; *Colorization* (ZHANG; ISOLA; EFROS, 2016) em que é proposta uma tarefa de colorização de imagem para realizar a previsão dos canais de crominância de uma imagem, dada sua luminância e *Split-Brain Autoencoders* (ZHANG; ISOLA; EFROS, 2017) em que utilizam um *autoencoder* com uma divisão na arquitetura, resultando em duas sub-redes separadas, e cada sub-rede é treinada para prever um subconjunto dos canais de uma imagem em relação aos outros canais. Estes três últimos trabalhos citados comumente também são classificados como trabalhos da área de auto-supervisão, porém para melhor separação no texto decidimos colocá-los nesta seção pois são trabalhos que têm como objetivo uma tarefa de reconstrução, não sendo o objetivo único o aprendizado de características, mas também a qualidade visual das imagens geradas. Ou seja, a tarefa utilizada não é "descartada", como ocorre em outras tarefas em outros trabalhos da área de auto-supervisão.

Figura 4 – Ilustração de métodos comuns de reconstrução/geração de parte da imagem original: colorização, super-resolução e pintura de imagem. Dada a entrada original à esquerda, os modelos são solicitados a recuperá-la com diferentes entradas parciais dadas à direita. Adaptado de Liu *et al.* (2021).

2.3.1.2 Baseado em auto-supervisão (self-supervision)

É uma estratégia de treinamento não supervisionada para aprendizado de representações. Uma característica de métodos de aprendizado auto-supervisionado é buscar projetar um "gerador de problemas", de modo que os modelos capturem informações úteis sobre os dados, enquanto resolvem os problemas gerados. Esses problemas utilizam conhecimento prévio sobre a estrutura dos dados, e não de rótulos explícitos.

Esses problemas são considerados tarefas auxiliares (também chamadas de tarefas pretexto) para o modelo a ser treinado. A ideia é utilizar os próprios dados como supervisão para estas tarefas, as quais devem ser projetadas de forma que o entendimento em alto nível da imagem seja útil. Como resultado, as camadas intermediárias do modelo treinado para resolver estas tarefas pretexto codificam representações visuais semânticas de alto nível, que são úteis para

resolver tarefas subsequentes. Essas abordagens geralmente envolvem a alteração ou ocultação de certas partes dos dados de alguma maneira, e um modelo é treinado para prever ou gerar as informações ausentes ou modificadas. Jing e Tian (2019) e Liu *et al.* (2021) apresentam uma extensa revisão dos métodos de auto-supervisão.

Um passo importante para aprender boas tarefas auto-supervisionadas foi encorajar os modelos a aprenderem representações que fossem invariantes a transformações (comumente aplicadas como *data augmentation*) aplicadas às imagens. Um dos primeiros trabalhos a explorar esta questão foi *ExemplarCNN* (DOSOVITSKIY *et al.*, 2015), em que os autores propõem uma CNN para distinguir um conjunto de classes substitutas. Cada classe substituta é formada aplicando uma variedade de transformações a uma parte de uma imagem. A parte da imagem é amostrada aleatoriamente, e, após aplicar transformações nesta parte da imagem, obtém-se um conjunto de imagens que representarão esta nova classe substituta. Ao contrário do treinamento de uma rede de forma supervisionada, a representação resultante não é específica em relação às classes. Neste caso, a representação aprendida é robusta em relação às transformações que foram aplicadas durante o treinamento.

Figura 5 – Exemplos de tarefas que envolvem predição de posição relativa. Adaptado de Liu *et al.* (2021).

Trabalhos iniciais de auto-supervisão foram principalmente inspirados em tarefas pretexto para **predições de posição relativa**, focando em aprender posições relativas entre componentes locais da imagem. Em *ContextPrediciton* (DOERSCH; GUPTA; EFROS, 2015), os autores treinam uma CNN que prevê a localização relativa de dois *patches* de imagem não sobrepostos amostrados aleatoriamente. Por exemplo, se obtivermos 9 *patches* não sobrepostos de uma imagem e considerarmos o *patch* do meio como a âncora, pode-se tentar prever a localização relativa dos outros 8 *patches* em relação a esse *patch* do meio. Outro método, *Jigsaw* (NOROOZI; FAVARO, 2016), resolve quebra-cabeças considerando blocos de imagem obtidos da imagem. Nesse caso, dados alguns *patches* não sobrepostos retirados de uma imagem, o objetivo é reorganizar esses *patches*, como se estivesse resolvendo um quebra-cabeça. Outros trabalhos que envolvem predições de posição relativa são realizados em Mundhenk, Ho e Chen (2018) e Noroozi *et al.* (2018). Também é possível aprender representações úteis prevendo transformações de rotação simples com *RotNet* (GIDARIS; SINGH; KOMODAKIS, 2018), ao rotacionar aleatoriamente uma imagem em um dos quatro ângulos possíveis (0, 90, 180 ou 270°), e treinar um modelo para prever qual rotação foi aplicada.

Alguns trabalhos utilizam a **maximização da informação mútua**, em que o objetivo é aprender a associação entre duas variáveis com o objetivo de maximizá-la. Estes trabalhos se concentram principalmente em aprender as relações diretas entre as partes locais e o contexto global da imagem. Aqui, as posições relativas entre as partes locais são ignoradas. Um dos primeiros trabalhos a modelar informações mútuas com uma tarefa de aprendizagem contrastiva é o *Deep InfoMax* (HJELM *et al.*, 2018), em que os autores maximizam a informação mútua entre um *patch* local e o contexto global da imagem. Também controlam a distribuição de probabilidade da representação de forma adversária. Esse método é capaz de priorizar tanto a informação global quanto a local, sendo capaz de ajustar a adequação das representações aprendidas para tarefas de classificação ou de reconstrução. Em *AMDIM* (BACHMAN; HJELM; BUCHWALTER, 2019), obtém-se melhores resultados ao maximizar a informação mútua entre as características extraídas de diferentes visualizações de um contexto compartilhado de uma mesma imagem. Em *CMC* (TIAN; KRISHNAN; ISOLA, 2019) investiga-se um método em que a representação é aprendida ao maximizar a informação mútua entre diferentes visualizações de uma mesma cena (por exemplo, entre diferentes canais da imagem), de forma que as diferentes visualizações de uma mesma cena são mapeadas para pontos próximos, enquanto que visualizações de cenas diferentes são mapeadas para pontos distantes. É importante observar que usar o contexto global pode não ser o ideal quando o objetivo é aprender a discriminar classes, pois o fundo da imagem pode atrapalhar na identificação da classe correta.

Outros trabalhos se concentram na **discriminação baseada em *clusters***. Ao produzir pseudo-rótulos com o algoritmo K-means, *DeepCluster* (CARON *et al.*, 2018) foi um dos primeiros trabalhos a obter representações competitivas com modelos supervisionados, utilizando apenas dados não rotulados. Estratégias adicionais são apresentadas em *LA* (ZHUANG; ZHAI; YAMINS, 2019) e *ClusterFit* (YAN *et al.*, 2020). *SwAV* (CARON *et al.*, 2020) foi capaz de introduzir estratégias de agrupamento *online* (ou seja, sem necessidade de computar as representações de todas as imagens a cada iteração) e novas estratégias de transformações dos dados, não exigindo treinamento em dois estágios, que é demorado, e introduziu estratégias de transformações de dados mais eficientes, obtendo melhor eficiência computacional e melhor desempenho.

Métodos importantes foram desenvolvidos baseados no trabalho de discriminação de instância *InstDisc* (WU *et al.*, 2018), que trata cada instância (cada imagem) como uma classe distinta própria, e utiliza uma função *softmax* não paramétrica, calculada com uma adaptação do método *Noise-contrastive estimation* (GUTMANN; HYVÄRINEN, 2010). Estes métodos utilizam diferentes técnicas para maximizar a similaridade de representações obtidas de diferentes versões distorcidas de uma amostra usando alguma variante de redes siamesas. Como existem soluções triviais para esse problema, como uma representação constante, eles contam com diferentes estratégias para aprender as representações. CMC (TIAN; KRISHNAN; ISOLA, 2019), trabalho que também utiliza maximização de informação mútua, não utiliza apenas visualizações diferentes de uma imagem, mas também amostra outra imagem como exemplo negativo. Em

MoCo (HE *et al.*, 2020), mantém-se uma fila de amostras negativas e utiliza-se um *encoder* atualizado através da média móvel para melhorar a consistência da fila. *SimCLR* (CHEN *et al.*, 2020) aprende representações maximizando a similaridade entre visualizações transformadas de formas diferentes da mesma imagem no espaço latente, sem a necessidade de utilizar um banco de memória, mas utilizando *batches* grandes. *InfoMin* (TIAN *et al.*, 2020) explora a transformação de amostras positivas. *BYOL* (GRILL *et al.*, 2020) não utiliza o uso de amostras negativa, mas usa uma estratégia de média móvel exponencial para atualizar um *encoder*. *SimSiam* (CHEN; HE, 2021) usa uma operação de parada de gradiente em um lado da arquitetura para tornar o treinamento estável. *Barlow Twins* (ZBONTAR *et al.*, 2021a) calcula a matriz de correlação cruzada entre as saídas de duas redes idênticas recebendo como entrada versões distorcidas de uma mesma imagem, tornando a a matriz de correlação cruzada o mais próximo possível de uma matriz identidade. Estudos recentes como *MOCO*, *SimCLR*, *BYOL*, *Simsiam*, *SwAV* e *Barlow Twins* alcançaram resultados competitivos no conjunto de dados ImageNet em comparação com *baselines* supervisionadas.

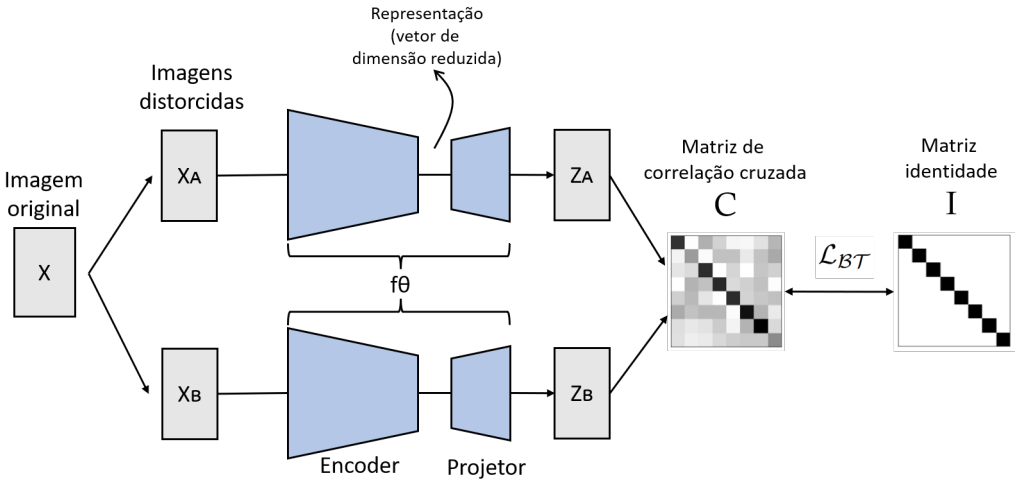


Figura 6 – Ilustração do método Barlow Twins. A função objetivo deste método tem com finalidade tornar a matriz de correlação cruzada igual à matriz identidade. A matriz de correlação cruzada é calculada entre as duas saídas de redes idênticas, que recebem versões distorcidas da mesma imagem. Isso faz com que as saídas (vetores) de versões distorcidas de uma mesma imagem sejam semelhantes, minimizando a redundância entre os componentes desses vetores. Adaptado de Zbontar *et al.* (2021a).

Formalmente, a função de custo utilizada pelo método Barlow Twins é dada pela Equação 2.5:

$$\mathscr{L}_{BT} \triangleq \sum_{i} (1 - C_{ii})^2 + \lambda \sum_{i} \sum_{j \neq i} C_{ij}^2 \tag{2.5}$$

em que λ é uma constante positiva que determina a importância entre o primeiro e o segundo termo. C é a matriz de correlação cruzada calculada entre as saídas das duas redes idênticas ao longo de um *batch* de imagens, dada pela Equação 2.6:

$$C_{ij} \triangleq \frac{\sum_b z^A_{b,i} z^B_{b,j}}{\sqrt{\sum_b \left(z^A_{b,i}\right)^2} \sqrt{\sum_b \left(z^B_{b,j}\right)^2}} \tag{2.6}$$

onde b indexa amostras de um *batch* de imagens e i, j indexam a dimensão vetorial das saídas das redes. C é uma matriz quadrada com a dimensionalidade de tamanho igual à da saída da rede, e com valores entre -1 (ou seja, anticorrelação perfeita) e 1 (correlação perfeita).

2.3.2 Semi-supervisionado

O aprendizado semi-supervisionado procura amenizar a necessidade de dados rotulados, permitindo que um modelo utilize também dados não rotulados. Os métodos de aprendizado semi-supervisionado fornecem uma maneira de explorar os padrões dos exemplos não rotulados, aliviando a necessidade de um grande número de rótulos. De maneira geral podemos classificar os métodos semi-supervisionados no paradigma de classificação de imagens em: métodos geradores (*generative*), regularizadores de consistência (*consistency regularization*), pseudo-rótulos e métodos híbridos (YANG *et al.*, 2021; OUALI; HUDELOT; TAMI, 2020). O paradigma de classificação de imagens consiste em, dado um conjunto de dados de treinamento com dados rotulados e não rotulados, a classificação semi-supervisionada tem como objetivo treinar um classificador a partir de dados rotulados e não rotulados, de modo que seja melhor do que o classificador supervisionado treinado apenas nos dados rotulados (SANTOS *et al.*, 2020). Assim como no aprendizado não supervisionado, as representações aprendidas de forma semi-supervisionada também podem ser utilizadas, uma vez aprendidas, em outras tarefas como classificação, detecção de objetos e segmentação semântica.

O protocolo padrão para a avaliação de algoritmos de aprendizado semi-supervisionado consiste em i) considerar uma uma base de dados rotulada; ii) levar em consideração apenas uma parcela dados com rótulos (por exemplo, 10%); iii) tratar os dados restantes como não rotulados. Apesar desta situação não refletir um cenário realista (OLIVER *et al.*, 2018), este ainda é o protocolo de avaliação padrão.

Muitos trabalhos da área de aprendizado semi-supervisionado que utilizam redes neurais profundas eram baseados em modelos geradores. Modelos geradores podem aprender representações implícitas dos dados, sendo capazes de modelar a distribuição dos dados. Com isso são capazes de então gerar novos dados com o conhecimento dessa distribuição. Como as GANs podem aprender a distribuição dos dados a partir de dados não rotulados, isto pode ser utilizado para facilitar o aprendizado semi-supervisionado, como por exemplo ao reutilizar as representações aprendidas pelo discriminador; ao usar dados gerados pelas GANs para regularizar um classificador; ou então ao utilizar os dados gerados como dados de treinamento adicionais.

Dentre os **métodos geradores** para o aprendizado semi-supervisionado de representações, Radford, Metz e Chintala (2015) mostram que é possível utilizar uma função de classificação

com uma função não supervisionada de uma GAN. *CatGAN* (SPRINGENBERG, 2016) modifica a função objetivo da GAN utilizada ao considerar também a informação mútua entre os exemplos observados e a predição das classes dada pela função supervisionada, fazendo com que o modelo force o discriminador a maximizar a informação mútua entre os exemplos e suas distribuições de classe previstas, em vez de treinar o discriminador para aprender uma classificação binária. Trabalhos posteriores como *SGAN* (ODENA, 2016) e *ImprovedGAN* (SALIMANS *et al.*, 2016) aprendem um gerador e um classificador simultaneamente, e há trabalhos que melhoram sobre essa ideia, como *GoodBadGAN* (DAI *et al.*, 2017), *LocalizedGAN* (QI *et al.*, 2018) e *CT-GAN* (WEI *et al.*, 2018). Em *CCGAN* (DENTON; GROSS; FERGUS, 2016) é utilizada uma função de custo adversária que aproveita dados não rotulados com base em *inpainting*, utilizando informações de contexto fornecidas pelas partes ao redor da imagem. Este método treina uma GAN em que o gerador deve gerar pixels dentro de uma parte ausente da imagem e o discriminador é treinado para diferenciar as imagens reais não rotuladas das imagens pintadas. *BiGAN* (DONAHUE; KRÄHENBÜHL; DARRELL, 2016) adiciona um *encoder* na arquitetura e *ALI* (DUMOULIN *et al.*, 2016) aprende um modelo para inferência durante o treinamento. *Triple-GAN* (LI *et al.*, 2017) adiciona um classificador independente ao invés de usar um discriminador como um classificador. Ainda, alguns métodos também são baseados em VAEs (*Variational Auto-Encoders*) como o trabalho pioneiro (KINGMA *et al.*, 2014) e alguns de seus sucessores *ADGM* (MAALØE *et al.*, 2016), *Disentangled VAE* (PAIGE *et al.*, 2017), entre outros.

Os **métodos de *consistency regularization*** são baseados na suposição de que, se uma perturbação for aplicada aos dados não rotulados, a predição não deve mudar significativamente. O modelo pode então ser treinado para ter uma predição consistente entre um determinado exemplo não rotulado e sua versão perturbada. Normalmente um termo relacionado à função de custo não supervisionada é aplicado à função de custo final. Uma estrutura comum presente nestes métodos é a estrutura de *Teacher-Student*. A estrutura de estudante aprende, enquanto que estrutura de professor gera exemplos para que a rede estudante aprenda, simultaneamente. Como o próprio modelo gera exemplos, eles podem ser incorretos e usados por ele mesmo, gerando um viés de confirmação. Este viés pode ser mitigado melhorando a qualidade dos exemplos gerados. Diferentes técnicas variam em como gerar os exemplos e também como melhorá-los, apresentadas no parágrafo abaixo.

LadderNetwork (RASMUS *et al.*, 2015) utiliza um método que tem dois *encoders*, um não corrompido e um corrompido, e um *decoder*. O dado então é passado pelos dois *encoders*, gerando uma predição limpa e outra corrompida. A predição corrompida é então utilizada pelo *decoder*, com objetivo de reconstruir o exemplo sem os ruídos, bem como as ativações limpas do *encoder* não corrompido. A função de custo final considera tanto as ativações do *encoder* não corrompido como as ativações reconstruídas, calculadas em relação a todas as camadas. Π-*model* (SAJJADI; JAVANMARDI; TASDIZEN, 2016) propõe um modelo simplificado, que remove o *encoder* corrompido, e utiliza a mesma rede porém com adição de transformações

dos dados de entrada e *dropout* na rede, que fará com que a rede gere predições distintas, e o objetivo é gerar predições consistentes entre as duas predições. *Temporal Ensembling* (LAINE; AILA, 2016) é semelhante ao Π-*model*, porém pra obter as predições dos rótulos ele calcula a Média Móvel Exponencial das predições de épocas passadas. Métodos subsequentes incluem *Mean Teacher* (TARVAINEN; VALPOLA, 2017), *Dual Students* (KE *et al.*, 2019) e *Fast-SWA* (ATHIWARATKUN *et al.*, 2019). *VAT* (MIYATO *et al.*, 2018) propõe o conceito de ataque adversarial para regularização de consistência. *UDA* (XIE *et al.*, 2020a) usa métodos avançados de transformação de dados como perturbações.

Há também os **métodos de pseudo-rótulos**. Os métodos de pseudo-rótulos diferem dos métodos de *consistency regularization*, pois estes últimos geralmente dependem da consistência entre as transformações dos dados, enquanto que os métodos de pseudo-rótulos dependem das predições dos pseudo-rótulos do conjunto não rotulado, que podem ser adicionados ao conjunto de dados de treinamento como dados rotulados. Esses pseudo-rótulos são então usados em conjunto com os dados rotulados, fornecendo algumas informações adicionais de treinamento, mesmo que os rótulos produzidos sejam frequentemente ruidosos ou fracos. *EntMin* (GRANDVALET; BENGIO, 2005) incentiva o modelo a fazer previsões de baixa entropia para dados não rotulados e, em seguida, utiliza os dados não rotulados em uma configuração de aprendizado supervisionado padrão. *Pseudo-label* (LEE, 2013) treina uma rede de forma supervisionada com dados rotulados e não rotulados simultaneamente. O modelo é treinado em dados rotulados de maneira supervisionada usual, e para dados não rotulados o mesmo modelo é usado para obter previsões para um *batch* de amostras não rotuladas. *Noisy Student* (XIE *et al.*, 2020b) utiliza uma variedade de técnicas ao treinar a rede do aluno, como transformação de dados e *dropout*. *Meta Pseudo Labels* (PHAM *et al.*, 2021) modifica o método de *Pseudo-label* ao atualizar a rede professora com *feedbacks* da rede aluno. Esses métodos também são chamados de *self-training*. Outros métodos treinam várias redes para a tarefa e exploraram o desacordo entre elas durante o processo de aprendizagem, como *Deep Co-training* (QIAO *et al.*, 2018) inspirados por Blum e Mitchell (1998) e *Tri-net* (DONG-DONGCHEN; WEIGAO, 2018), inspirados por *tri-training* (ZHOU; LI, 2005). Nesses casos duas ou três redes diferentes são treinadas simultaneamente e rotulam amostras não rotuladas para cada uma delas.

Modelos híbridos combinam as ideias dos métodos de pseudo-rótulos e *consistency regularization*. Aqui muitos métodos se baseiam no trabalho *MixUp* (ZHANG *et al.*, 2018), que estende o conjunto de dados de treinamento utilizando uma restrição que consiste em considerar que as interpolações lineares dos dados devem levar às interpolações lineares dos rótulos correspondentes. *ICT* (VERMA *et al.*, 2019) regulariza o modelo incentivando a previsão em uma interpolação de dois exemplos não rotulados para ser consistente com a interpolação das previsões nesses pontos. *MixMatch* (BERTHELOT *et al.*, 2019b) combina regularização de consistência e minimização de entropia em uma função de custo unificada. Esse modelo opera produzindo pseudo-rótulos para cada instância não rotulada e, em seguida, treinando os dados rotulados originais com os pseudo-rótulos para os dados não rotulados usando técnicas

totalmente supervisionadas. Outros métodos incluem *ReMixMatch* (BERTHELOT *et al.*, 2019a), *FixMatch* (SOHN *et al.*, 2020) e *DivideMix* (LI; SOCHER; HOI, 2020).

2.3.3 Trabalhos mais relacionados

Em termos de estratégia de treinamento semi-supervisionado e o uso de tarefas auxiliares como um termo regularizador da função de custo junto com a tarefa supervisionada, nosso trabalho está mais relacionado aos trabalhos *S4L* (ZHAI *et al.*, 2019), *SESEMI* (TRAN, 2019) e Hendrycks *et al.* (2019).

Em *S4L* (ZHAI *et al.*, 2019) os autores combinam tarefas da área de auto-supervisão com o aprendizado semi-supervisionado, propondo um aprendizado conjunto. Utilizam a tarefa de previsão de rotação e a tarefa de distinguir classes substitutas criadas a partir de transformações da imagem, juntamente com treinamento em uma parte de dados rotulados no conjunto de dados ImageNet. Em *SESEMI* (TRAN, 2019), o autor propõe um modelo com uma função de custo que explora a tarefa auto-supervisionada de prever rotações aplicadas às imagens não rotuladas juntamente com uma tarefa de classificação padrão utilizando um conjunto limitado de dados rotulados para treinar o modelo nas bases de dados SVHN, CIFAR-10 e CIFAR-100.

Diferente desses dois, exploramos neste trabalho não apenas imagens naturais e coloridas, mas também um domínio de imagem em tons de cinza (Fashion-MNIST) e também no conjunto de dados Malaria, um conjunto de dados que não é orientado a ângulos (diferentemente de fotografias que possuem viés de ângulo). Além de investigar a tarefa de rotação como base para obter uma rede semi-supervisionada (CAVALLARI; PONTI, 2021), avaliamos também uma tarefa de auto-supervisão do estado-da-arte (Barlow Twins).

Hendrycks *et al.* (2019) descobrem que a auto-supervisão pode aumentar a robustez do modelo contra exemplos adversários e ruídos nos rótulos e dados de entrada. Eles usam a tarefa de previsão de rotação como um termo auxiliar da função de custo para treinar o modelo semi-supervisionado na base de dados CIFAR-10. Em nosso trabalho, realizamos ataques de 1 pixel e avaliamos as representações resultantes.

2.3.4 Avaliação das representações aprendidas

Infelizmente, não há um critério geral para criar uma representação visual. No entanto, uma escolha natural é obter modelos capazes de discriminar atributos dos dados. Por exemplo, vários fatores como a forma do objeto, material do objeto e fontes de luz se combinam para criar efeitos complexos, como sombras, padrões de cores e reflexos nas imagens. Representações ideais separariam cada um desses fatores.

O protocolo atual mais utilizado para avaliar representações aprendidas de forma não supervisionada foi proposto por Zhang, Isola e Efros (2016): i) treina-se o modelo nas imagens do conjunto ImageNet, descartando os rótulos; ii) treina-se de forma supervisionada um classificador

linear tendo como entrada os vetores de características extraídos pelo modelo utilizando todo o conjunto de treinamento; iii) obtém-se a acurácia de classificação ao testar o modelo linear treinado no passo anterior em um conjunto de testes.

A maioria dos trabalhos em aprendizado semi-supervisionado utiliza o seguinte procedimento para avaliar a qualidade dos modelos: i) treina-se um modelo em um pequeno conjunto de dados rotulados; ii) treina-se um outro modelo no pequeno conjunto de dados rotulados juntamente com dados não rotulados; iii) comparam-se as acurácias obtidas pelos modelos i) e ii). A qualidade das representações aprendidas não são avaliadas pelo desempenho em tarefas subsequentes ou através de classificadores lineares, como nos trabalhos do aprendizado não supervisionado.

Ainda, é possível avaliar a qualidade das representações aprendidas através de alguns métodos qualitativos que utilizam visualizações (JING; TIAN, 2019). Entre esses métodos estão: i) visualização dos primeiros filtros aprendidos pelo modelo; ii) visualização dos *feature maps*, para descobrir para quais partes da imagem o modelo está dando mais atenção; e iii) visualização através de recuperação dos vizinhos mais próximos, pois imagens com aparência semelhante geralmente devem ficar próximas no espaço de características.

Neste trabalho optou-se pela utilização de diferentes bases de dados de domínios distintos para avaliação dos métodos estudados, além de cenários em que ocorre ataque de pixel nos dados do conjunto de treinamento. Além disso, utilizamos SVM para análise do espaço de características aprendido, além de visualizações com tSNE. Este trabalho não tem como objetivo propor um método do estado-da-arte em aprendizado de características, mas realizar avaliações dos espaços de características formados por diferentes tarefas, métodos e cenários de limitação de dados e ataques.

CAPÍTULO

3

MÉTODO

Considerando um conjunto de dados rotulados D_l contendo N_l pares de imagens e rótulos, e considerando um conjunto de dados não rotulados D_u contendo N_u imagens sem rótulos, os experimentos foram realizados em cinco etapas principais:

1. *Baselines supervisionados* ao realizar treinamento supervisionado em cenários de limitação de dados anotados, utilizando 1% ou 5% de dados rotulados em relação ao total de dados não rotulados. Os dados anotados são do conjunto D_l.

2. *Baselines não-supervisionados* ao treinar as tarefas auto-supervisionadas RotNet ou BarlowTwins, que não necessitam de rótulos explícitos, no conjunto de dados não rotulados D_u;

3. *Baselines semi-supervisionados* ao realizar *fine-tuning* utilizando 1% ou 5% de dados rotulados em relação ao total de dados não rotulados, a partir dos pesos congelados dos modelos treinados na etapa 2;

4. Método semi-supervisionado unificado que utiliza 1% ou 5% dos dados rotulados em relação ao total de dados não rotulados, simultaneamente com os dados não rotulados D_u, através de uma função de custo unificada e arquitetura siamesa;

5. *Fine-tuning* do método semi-supervisionado da etapa 4, utilizando 1% ou 5% de dados rotulados em relação ao total de dados não rotulados.

As três primeiras etapas foram realizadas para obtermos *baselines* para nosso estudo. Na quarta etapa estão os experimentos com nosso método semi-supervisionado proposto. A quinta etapa é realizada para analisar se o modelo aprendido na etapa 4 pode ser melhorado ainda mais. O método semi-supervisionado unificado será descrito com detalhes a seguir.

Nosso método semi-supervisionado considera um treinamento que simultaneamente utiliza dados rotulados e não rotulados, utilizando uma função de custo $\mathscr{L}$ que treina uma rede siamesa:

$$\mathscr{L}_{SS} = \lambda_l \cdot \ell_l(D_l) + \lambda_u \cdot \ell_u(D_u) \tag{3.1}$$

em que ℓ_l otimiza uma função de entropia-cruzada treinada com 1% ou 5% de dados rotulados em relação ao total de dados não rotulados, retirados do conjunto D_l, e ℓ_u otimiza a função de uma tarefa auto-supervisionada treinada no conjunto D_u. Os pesos λ_l e λ_u são número positivos. Esta função $\mathscr{L}$ pode ser usada com diferentes funções de custo auto-supervisionadas ℓ_u. A rede tem compartilhamento de pesos entre a tarefa supervisionada e tarefa de auto-supervisão na arquitetura principal.

Ao utilizar a tarefa de predição de rotação (RotNet), temos duas camadas de *softmax* separadas. Uma camada para a tarefa de classificação supervisionada (tendo como quantidade de neurônios a quantidade de classes a ser treinada), e a outra camada de *softmax* para a tarefa de rotação, contendo 4 neurônios, referentes às 4 rotações que podem ser aplicadas por este método (que correspondem a 0, 80,180, 270 graus de rotação aplicados). A figura 7 ilustra a arquitetura unificada utilizando a tarefa de predição de rotação.

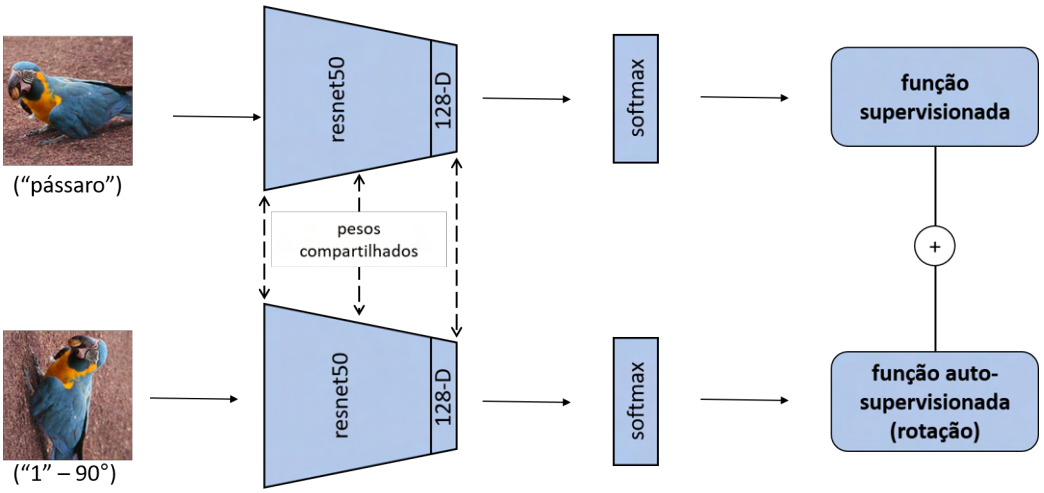


Figura 7 – Ilustração da arquitetura unificada utilizando a tarefa de predição de rotação. A função supervisionada compara imagens originais e seus rótulos, enquanto que a função auto-supervisionada de rotação compara imagens rotacionadas e o ângulo de rotação aplicado como "rótulo".

Ao utilizar a tarefa Barlow Twins, uma ramificação da arquitetura é responsável pela tarefa de classificação supervisionada, realizada pela camada de *softmax*. As outras duas ramificações da rede principal são utilizadas pela tarefa Barlow Twins, que recebem como entrada duas visualizações alteradas de uma mesma imagem, que serão passadas pelo *backbone* de uma *resnest50*, e então se conectar a uma MLP de três camadas de tamanho 2048, sendo que as duas primeiras camadas possuem *Batch Normalization* e *ReLu*, e a terceira não, semelhante à

proposição original do Barlow Twins, porém utilizamos camadas de tamanho 2048 ao invés de 8192 como no artigo original. A Figura 8 ilustra a arquitetura unificada ao utilizar a tarefa Barlow Twins. O *pipeline* de transformação da imagem consiste nas seguintes transformações: corte aleatório, redimensionamento, inversão horizontal, variação de cor, conversão para escala de cinza, desfoque gaussiano e solarização. As duas primeiras transformações (corte e redimensionamento) são sempre aplicadas, enquanto as cinco últimas são aplicadas aleatoriamente, com alguma probabilidade.

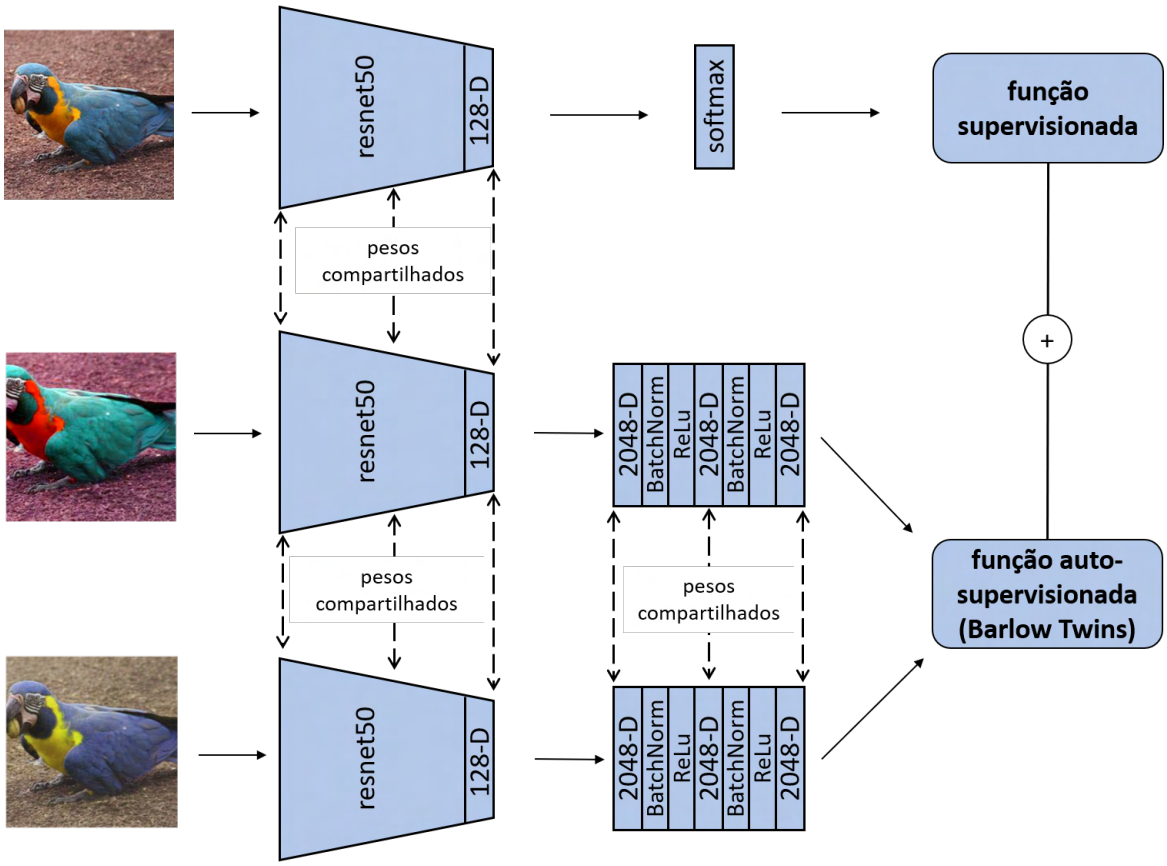


Figura 8 – Ilustração da arquitetura unificada utilizando a tarefa Barlow Twins. A função supervisionada compara imagens originais e seus rótulos, enquanto que a função auto-supervisionada Barlow Twins compara representações de duas visualizações alteradas de uma mesma imagem.

3.1 Base de dados

Avaliamos o desempenho do nosso método em três conjuntos de dados diferentes: STL-10, Fashion-MNIST e Malaria. STL-10 é um conjunto de dados desenvolvido para aprendizado semi-supervisionado e não supervisionado, contendo imagens coloridas de tamanho 96×96 de aviões, pássaros, carros, gatos, veados, cães, cavalos, macacos, navios e caminhões. A base de dados Fashion-MNIST tem tamanho 28×28 de imagens em tons de cinza de desenhos de peças de roupas e acessórios de moda. O conjunto de dados Malaria contém imagens de células de múltiplas resoluções com instâncias de células parasitadas e não infectadas a partir de imagens de

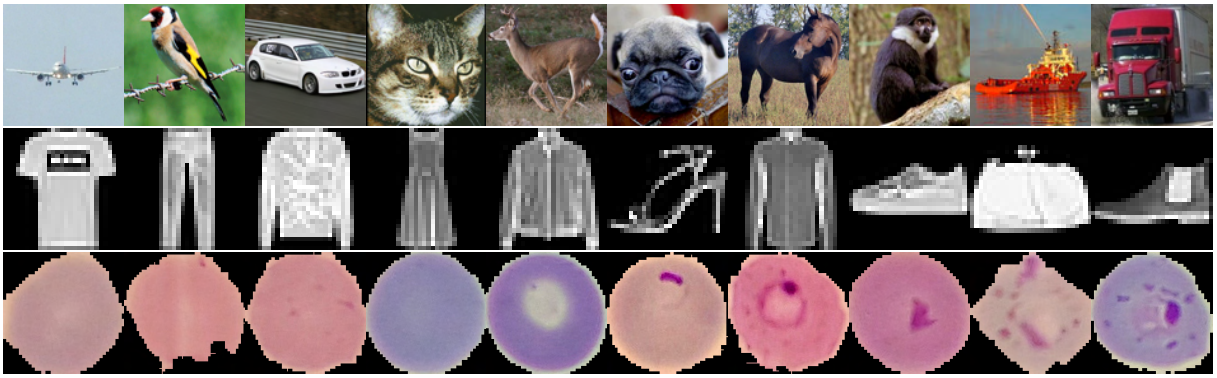


Figura 9 – Exemplos das classes das bases de dados STL-10, Fashion-MNIST e Malaria, respectivamente.

lâmina de esfregaço de sangue de células segmentadas. A Figura 9 mostra exemplos de imagens desses conjuntos de dados. Detalhes sobre cada uma delas são apresentados a seguir.

1 – Imagens Naturais

STL-10:[1] é uma base de dados criada com foco no desenvolvimento de algoritmos que aprendem características de forma não supervisionada ou semi-supervisionada. Possui dois conjuntos de imagens: um com 100 mil imagens não rotuladas, e outro com imagens rotuladas. O conjunto com imagens rotuladas possui 10 classes, com 500 imagens de treino por classe e 800 imagens de teste por classe. O conjunto de imagens não rotuladas é extraído de uma distribuição similar porém mais ampla do que o conjunto rotulado.

2 – Imagens de desenhos de peças de roupas

Fashion-MNIST: (XIAO; RASUL; VOLLGRAF, 2017), possui 10 classes, que consiste em imagens de desenhos de peças de roupas, criada para ser uma substituta imediata da base de dados MNIST - as imagens também são binárias, de tamanho 28x28, e contém um total de 60 mil imagens para treino e 10 mil imagens para teste.

3 – Imagens médicas

Malaria: (RAJARAMAN *et al.*, 2018), contém um total de 27.558 imagens de células, sendo metade com imagens de células parasitadas e a outra metade de imagens de células não infectadas.

Para a base de dados STL-10 consideramos as 100 mil imagens não rotuladas como conjunto não rotulado. Para os treinamentos que utilizam dados rotulados, utilizamos 1 mil imagens ou 5 mil imagens como imagens rotuladas (1% e 5% em relação ao total de imagens não rotuladas), retiradas do conjunto original de treinamento que contém 5 mil imagens. As 8 mil imagens do conjunto de teste não são utilizadas no treinamento.

Para a base de dados Fashion-MNIST, utilizamos as 60 mil imagens de treinamento como conjunto não rotulado, descartando-se os rótulos. Para os treinamentos que utilizam dados rotulados, utilizamos 600 imagens ou 3 mil imagens como imagens rotuladas (1% e 5%

[1] <https://cs.stanford.edu/~acoates/stl10/>

em relação ao total de imagens não rotuladas), retiradas do conjunto original de treinamento que contém 60 mil imagens. As 10 mil imagens do conjunto de teste não são utilizadas no treinamento.

Para a base de dados Malaria, que possui um total de 27.558 imagens, separamos metade para o conjunto de treinamento e metade para o conjunto de teste. Para os treinamentos que utilizam dados rotulados, utilizamos 137 imagens ou 685 imagens como imagens rotuladas (1% e 5% em relação ao total de imagens não rotuladas), e consideramos 13.779 imagens como conjunto não rotulado.

3.2 Configuração dos experimentos

Em todos os experimentos utilizamos como *backbone* a arquitetura ResNet50v2. Descartamos a camada final de classificação original, e adicionamos uma camada de *Global Average Pooling*, o que nos dá uma camada final de tamanho 2048. Após esta camada, adicionamos uma camada totalmente conectada, com ativação *relu*, de dimensão 128. Esta camada final de tamanho 128 é a que será avaliada nos experimentos. Exploramos três taxas de aprendizado para todos os experimentos: 0.01, 0.001 e 0.0001, com decaimento exponencial e^k em que $k = -0.01$, a partir da época 5% do total de épocas. Em todos os experimentos semi-supervisionados, utilizamos sempre $\lambda_l = 1.0$ e $\lambda_u = 1.0$.

Foi utilizado *batch* de tamanho 32 em todos os experimentos, com exceção dos experimentos que utilizam a tarefa Barlow Twins. Nesses casos específicos, como esta tarefa se beneficia de *batches* maiores, utilizamos *batch* de tamanho 200 para as bases de dados STL-10 e Fashion-MNIST, e de tamanho 120 para a base de dados Malaria, que foi o tamanho máximo dada a limitação de memória da placa de vídeo utilizada. No artigo original da tarefa Barlow Twins os autores obtêm os melhores resultados com *batch* de tamanho de 1024, porém a diferença na acurácia final ao utilizar 1024 ao invés dos tamanhos 128, 256 ou 512 é menor que 2%. O número de épocas foi escolhido de forma empírica para cada modelo, após a convergência das respectivas funções de custo, i.e. os valores computados da função de custo apresentaram variação baixa entre épocas subsequentes.

Todos os treinamentos foram executados 5 vezes, cada vez com uma semente diferente para a inicializações aleatória dos pesos. Para os experimentos que utilizam dados rotulados, cada treino foi realizado com uma partição aleatória dos dados rotulados. Isto foi realizado para garantir que os resultados não fossem influenciados pela qualidade da partição de dados escolhida. Além disso, utilizamos classes balanceadas em todas as partições rotuladas. Os mesmos conjuntos aleatórios de dados rotulados são utilizados para os treinamentos supervisionados e semi-supervisionados. Todos os modelos foram treinados do zero, com exceção dos modelos em que é realizado *fine-tuning*.

Para a base de dados STL-10, utilizamos o tamanho original de 96×96. Para a base de

dados Fashion-MNIST, as imagens foram redimensionadas para 96×96. Para a base de dados Malaria, as imagens foram redimensionadas para 128×128.

Para o caso do método semi-supervisionado unificado, em cada época o modelo vê todas as imagens não rotuladas N_u, enquanto as imagens rotuladas N_l são vistas pelo modelo um total de N_u / N_l vezes. Como $N_u > N_l$, em uma época o modelo verá todo o conjunto de imagens não rotuladas. Portanto, a rede será treinada com mais instâncias rotuladas por época, embora repetidas, em comparação com o método totalmente supervisionado.

3.3 Ataque de pixel

Avaliamos também o cenário em que os dados de treinamento rotulados contêm ruídos. Neste caso, os modelos foram treinados com dados que haviam ataque de pixel. O ataque de pixel foi realizado ao manipular manualmente as imagens, inserindo um pixel branco na mesma posição para todas as imagens de uma determinada classe. Para cada classe foi escolhida uma posição diferente. Como precisamos inserir o pixel branco na mesma posição em todas as imagens para uma dada classe, foi necessário o acesso aos dados de treinamento rotulados, pois o local do pixel a ser inserido depende da classe. Sendo assim, poderemos avaliar a robustez dos modelos em relação a casos quando temos as imagens de treinamento atacadas. A Figura 10 mostra exemplos de ataque de pixel. Os treinamentos dos modelos Rotação e Barlow Twins quando utilizam apenas dados não rotulados não sofrem ataque de pixel, pois precisamos ter acesso aos dados rotulados para realizar o ataque.

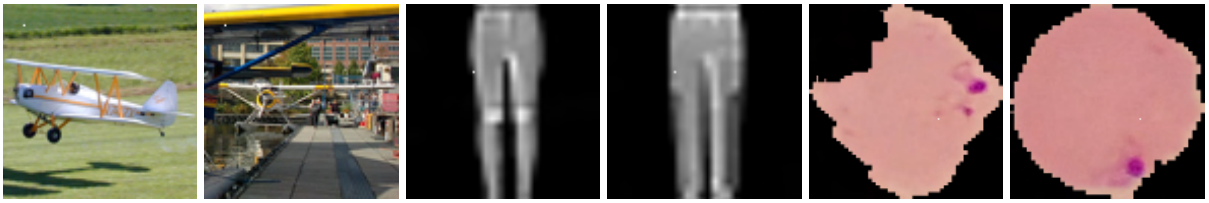


Figura 10 – Exemplo de ataque de pixel. Da esquerda para direta: duas imagens de aviões da STL-10, duas calças da Fashion-MNIST e duas imagens de amostras infectadas da Malaria. Figura melhor visualizada com *zoom*.

3.4 Avaliação

Para avaliar a capacidade discriminativa das representações aprendidas, utilizamos um SVM linear. O classificador SVM é um classificador raso com viés de baixa complexidade e baixa sensibilidade para ajuste de parâmetros, também tendo fortes garantias de aprendizado que o tornam útil como uma ferramenta para avaliar separabilidade linear de espaços de características (MELLO; PONTI, 2018). Como já citamos anteriormente, avaliaremos a camada de dimensão 128, sendo ela a camada após a última camada do *backbone* da resnet50.

Utilizando um modelo treinado nas etapas de 1 a 5 de nossos experimentos como extrator de características, consideramos como etapas para avaliação as descritas a seguir:

1. Extraímos as representações das imagens do conjunto de treinamento
2. Treinamos um SVM (linear, sem kernerls e com parâmetro $C = 1$) utilizando as representações extraídas na etapa 1
3. Extraímos as representações do conjunto de teste
4. Testamos o SVM treinado na etapa 2 nas representações extraídas do conjunto de teste

Testamos outros valores de C para o SVM, mas isso não alterou os resultados.

O conjunto de teste nunca foi visto durante o treinamento do modelo ou do SVM, e são usados apenas para obter as acurácias relatadas como resultado. O conjunto de treinamento corresponde a 1 mil ou 5 mil imagens (1% ou 5%) para a base de dados STL-10; 600 ou 3 mil imagens (1% ou 5%) para a Fashion-MNIST; 137 ou 685 imagens (1% ou 5%) para a Malaria; e são os mesmo conjuntos utilizados no treinamento dos modelos quando o modelo utiliza dados rotulados.

CAPÍTULO

4

RESULTADOS E DISCUSSÃO

Neste Capítulo apresentaremos os resultados dos experimentos realizados e a discussão. Na seção 4.1 discutimos qual foi o critério utilizado para a seleção dos modelos para teste. Na seção 4.2 apresentamos os resultados das avaliações com o SVMs e na seção 4.3 discutimos os resultados.

4.1 Selecionando a taxa de aprendizado / critério para seleção dos modelos para teste

Não utilizamos dados de validação, pois a quantidade de dados rotulados é muito pequena, o que não permite que tenhamos conjuntos de validação representativos. Por este motivo nossas escolhas de modelos para teste foram baseadas na função de custo de treinamento.

Como mencionado na seção 3.2, todo treinamento foi executado 5 (cinco) vezes (cada vez com inicialização aleatória de pesos e partições de dados diferentes). A Tabela 1 abaixo reporta a média e o desvio padrão do valor da função de custo da última época de treinamento desses cinco treinamentos. Como cada treinamento também foi executado com três taxas de aprendizado diferentes, elas também estão representadas na tabela, separadas por linhas. Na tabela abaixo e nas tabelas 2 e 3, abaixo do nome dos experimentos há 3 linhas reportando as médias do valor da função de custo da última época de treinamento para as taxas de aprendizados 0.01, 0.001 e 0.0001, respectivamente. Na primeira linha estão as médias para os treinamentos com taxa de aprendizado 0.01, na segunda linha para o treinamento com 0.001 e na terceira linha com 0.0001. Por exemplo, para a base de dados STL-10, ao realizar o experimento "Rotação", a menor média foi obtida ao utilizar o valor 0.0001 como taxa de aprendizado, indicada pelo valor 0.0139, destacada em negrito, como pode ser verificado na tabela abaixo. Assim, para este exemplo específico, avaliamos os 5 (cinco) modelos treinados no experimento "Rotação"com a taxa de aprendizado 0.0001, pois foi a taxa de aprendizado com a menor média do valor da

função de custo da última época de treinamento. O mesmo procedimento ocorre com os demais modelos, apresentados nas tabelas 2 e 3. Os resultados das avaliações dos modelos estão exibidos na próxima seção.

Tabela 1 – Média e desvio padrão do valor da função de custo da última época de treinamento para os modelos auto-supervisionados. Abaixo do modelo indicado, estão os resultados ao utilizar as taxas de aprendizados 0.01, 0.001 e 0.0001, respectivamente, representados em linhas separadas.

Taxa de aprendizado	STL-10	Fashion-MNIST	Malaria
	Rotação		
0.01	0.0653 ± 0.0334	0.0028 ± 0.0009	1.386 ± 0.000
0.001	0.0194 ± 0.0013	0.0011 ± 0.0002	0.010 ± 0.001
0.0001	$\mathbf{0.0139 \pm 0.0003}$	$\mathbf{0.0008 \pm 0.0002}$	$\mathbf{0.005 \pm 0.001}$
	Barlow Twins		
0.01	10.52 ± 0.15	$\mathbf{15.37 \pm 0.88}$	117.61 ± 61.74
0.001	$\mathbf{9.64 \pm 0.07}$	16.02 ± 1.8	100.42 ± 76.88
0.0001	14.27 ± 0.22	19.67 ± 0.4	$\mathbf{50.83 \pm 14.03}$

4.2 Acurácia e avaliação dos modelos

Como mencionado na seção anterior, selecionamos os 5 (cinco) modelos treinados com a taxa de aprendizado cuja média do valor da função de custo da última época de treinamento foi a menor. Retomando o exemplo da seção anterior, a média das acurácias de teste do SVM dos cinco treinamentos do modelo "Rotação"com a taxa de aprendizado 0.0001 foi 37.6, reportada na terceira linha da segunda coluna na tabela 4. As Tabelas 4 e 5 reportam a acurácia de teste dos experimentos.

4.3 Discussão

Nesta seção discutiremos os resultados obtidos, separados por base de dados para melhor compreensão.

4.3.1 STL-10

Quando utilizamos 1% de dados rotulados disponíveis, entre os modelos que utilizam a tarefa de Rotação, o modelo Semi-supervisionado foi o que obteve melhor acurácia, tanto no cenário sem ataque como com ataque. O mesmo se observa entre os modelos que utilizam a tarefa de Barlow Twins: o modelo semi-supervisionado foi o que teve melhor acurácia nos dois cenários.

Tabela 2 – Média e desvio padrão do valor da função de custo da última época de treinamento para os modelos que utilizam 1% de dados rotulados. Abaixo do modelo indicado, estão os resultados ao utilizar as taxas de aprendizados 0.01, 0.001 e 0.0001 (em cada linha de cima para baixo).

STL-10	STL-10 com ataque	Fashion-MNIST	Fashion-MNIST com ataque	Malaria	Malaria com ataque
Supervisionado 1%					
0.0334 ± 0.0501	0.0337 ± 0.0423	0.00723 ± 0.01463	0.00062 ± 0.00104	0.00114 ± 0.00107	0.00083 ± 0.00064
0.0099 ± 0.0092	0.0058 ± 0.0051	**0.00006 ± 0.00008**	**0.00003 ± 0.00002**	**0.00108 ± 0.00091**	**0.00029 ± 0.00039**
0.0029 ± 0.0026	**0.0045 ± 0.0025**	0.00022 ± 0.00013	0.00014 ± 0.00009	0.00204 ± 0.00197	0.00077 ± 0.00058
Rotação + Fine-tuning 1%					
0.0027 ± 0.0033	0.0015 ± 0.0007	0.00134 ± 0.00133	0.00140 ± 0.00266	0.5551 ± 0.3086	0.5574 ± 0.3033
0.0040 ± 0.0058	0.0036 ± 0.0048	0.00036 ± 0.00015	0.00038 ± 0.00022	**0.0006 ± 0.0003**	0.0005 ± 0.0004
0.0015 ± 0.0019	**0.0011 ± 0.0008**	**0.00027 ± 0.00007**	**0.00024 ± 0.00004**	0.0009 ± 0.0010	**0.0004 ± 0.0003**
Barlow Twins + Fine-tuning 1%					
0.0062 ± 0.0127	0.0097 ± 0.0059	0.00028 ± 0.00054	**0.00001 ± 0.00001**	0.0227 ± 0.0361	0.013 ± 0.0279
0.0006 ± 0.0011	0.0022 ± 0.0023	0.00021 ± 0.00013	0.00022 ± 0.00020	0.0109 ± 0.0149	0.0023 ± 0.0040
0.0011 ± 0.0015	**0.0012 ± 0.0013**	**0.00013 ± 0.00007**	0.00011 ± 0.00007	**0.0010 ± 0.0007**	**0.0006 ± 0.0007**
Rotação Semi 1%					
0.8290 ± 1.6035	0.0807 ± 0.0680	0.0040 ± 0.0020	0.0037 ± 0.0012	0.58581 ± 0.73515	1.52514 ± 0.90354
0.0135 ± 0.0010	**0.0143 ± 0.0013**	0.0012 ± 0.0002	0.0009 ± 0.0001	0.00121 ± 0.00068	0.00114 ± 0.00058
0.0143 ± 0.0004	0.0146 ± 0.0007	**0.0007 ± 0.0004**	**0.0008 ± 0.0001**	**0.00105 ± 0.00089**	**0.00102 ± 0.00095**
Barlow Twins Semi 1%					
13.04 ± 0.38	12.88 ± 0.19	**14.88 ± 1.52**	15.53 ± 2.25	77.64 ± 21.95	117.46 ± 53.06
11.63 ± 0.28	**11.50 ± 0.30**	15.32 ± 3.58	**14.66 ± 2.05**	100.91 ± 63.05	82.54 ± 33.41
17.04 ± 0.36	16.90 ± 0.32	18.68 ± 1.35	17.93 ± 0.64	**45.52 ± 11.32**	**61.61 ± 47.89**
Rotação Semi + Fine-tuning 1%					
0.4609 ± 1.0297	**0.00004 ± 0.00006**	**0.00003 ± 0.00004**	**0.00001 ± 0.00000**	0.0059 ± 0.0054	0.41792 ± 0.37689
0.0007 ± 0.0015	0.00027 ± 0.00036	0.00017 ± 0.00007	0.00008 ± 0.00002	0.0003 ± 0.0001	0.00014 ± 0.00017
0.0044 ± 0.0091	0.00171 ± 0.00246	0.00032 ± 0.00006	0.00029 ± 0.00007	**0.0001 ± 0.0001**	**0.00008 ± 0.00007**
Barlow Twins Semi + Fine-tuning 1%					
0.0024 ± 0.0030	**0.0039 ± 0.0055**	**0.0001 ± 0.0001**	**0.00006 ± 0.00003**	0.0018 ± 0.0021	0.0276 ± 0.0608
0.0053 ± 0.0094	0.0043 ± 0.0043	0.0002 ± 0.0000	0.00027 ± 0.00034	**0.0015 ± 0.0014**	0.0135 ± 0.0154
0.0036 ± 0.0053	0.0086 ± 0.0161	0.0004 ± 0.0003	0.00463 ± 0.00968	0.0023 ± 0.0017	**0.0028 ± 0.0050**

A maior acurácia obtida para a base de dados STL-10 para o caso 1% foi do método semi-supervisionado ao utilizar a tarefa Barlow Twins. Nota-se que o método de Barlow Twins + Fine-tuning atingiu acurácia estatisticamente semelhante no cenário sem ataque, porém no cenário com ataque o modelo semi-supervisionado obteve acurácia superior ao Fine-tuning. Isto pode indicar que o treinamento simultâneo foi capaz de auxiliar na robustez do modelo ao guiar a função de custo simultaneamente na tarefa auto-supervisionada, que utiliza dados não rotulados, e na tarefa de classificação comum, que utilizou dados supervisionados com ruído. Para a tarefa de Rotação Semi-supervisionada, ela obteve melhorar acurácia tanto no cenário sem ataque como com ataque em comparação com a tarefa Rotação + Fine-tuning.

Notamos que o fine-tuning do modelo semi-supervisionado piorou a acurácia em comparação com o modelo semi-supervisionado sem fine-tuning. Isto ocorreu com ambas as tarefas, tanto no caso 1% como no 5%, em ambos os cenários, sem e com ataque. Esta é uma indicação de que a continuação do treinamento com apenas os dados rotulados trouxe menor acurácia de teste, indicando que houve menor generalização .

Interessante notar que ambas as tarefas auto-supervisionadas sozinhas atingiram acurácia melhor do que o modelo supervisionado no cenário em que estão disponíveis apenas 1% de

Tabela 3 – Média e desvio padrão do valor da função de custo da última época de treinamento para os modelos que utilizam 5% de dados rotulados. Abaixo do modelo indicado, estão os resultados ao utilizar as taxas de aprendizado 0.01, 0.001 e 0.0001 (em cada linha de cima para baixo).

STL-10	STL-10 com ataque	Fashion-MNIST	Fashion-MNIST com ataque	Malaria	Malaria com ataque
Supervisionado - 5%					
0.4661 ± 1.0267	0.0017 ± 0.0024	0.00277 ± 0.00589	0.000270 ± 0.000362	0.13865 ± 0.30996	0.000173 ± 0.000371
0.0017 ± 0.0019	**0.0005 ± 0.0012**	**0.00003 ± 0.00004**	**0.000005 ± 0.000005**	**0.00001 ± 0.00000**	**0.000002 ± 0.000000**
0.0014 ± 0.0021	0.0011 ± 0.0018	0.00031 ± 0.00056	0.000070 ± 0.000125	0.00003 ± 0.00001	0.000006 ± 0.000002
Rotação + Fine-tuning 5%					
0.0005 ± 0.0005	**0.00026 ± 0.00015**	0.00074 ± 0.00108	0.000167 ± 0.000275	0.69320 ± 0.00004	0.69320 ± 0.00004
0.0011 ± 0.0018	0.00029 ± 0.00036	0.00079 ± 0.00110	0.000008 ± 0.000004	**0.00003 ± 0.00001**	**0.00002 ± 0.00001**
0.0007 ± 0.0009	0.00143 ± 0.00296	**0.00002 ± 0.00001**	**0.000004 ± 0.000003**	0.00006 ± 0.00005	0.00006 ± 0.00008
Barlow Twins + Fine-tuning 5%					
0.01485 ± 0.03100	0.00021 ± 0.00045	0.00246 ± 0.00536	0.001216 ± 0.001671	0.00096 ± 0.00109	0.00128 ± 0.00264
0.00043 ± 0.00070	**0.00002 ± 0.00003**	0.00019 ± 0.00026	**0.000003 ± 0.000002**	0.00016 ± 0.00015	0.00002 ± 0.00003
0.00045 ± 0.00071	0.00040 ± 0.00067	**0.00003 ± 0.00003**	0.000023 ± 0.000025	**0.00009 ± 0.00005**	**0.00001 ± 0.00001**
Rotação Semi 5%					
0.0363 ± 0.0056	0.0357 ± 0.0091	0.0035 ± 0.0005	0.7406 ± 1.6487	0.8330 ± 1.1379	1.5253 ± 0.9031
0.0135 ± 0.0009	**0.0127 ± 0.0006**	**0.0013 ± 0.0004**	0.0012 ± 0.0004	0.0009 ± 0.0005	**0.0011 ± 0.0008**
0.0149 ± 0.0007	0.0149 ± 0.0007	0.0014 ± 0.0004	**0.0008 ± 0.0002**	**0.0005 ± 0.0006**	0.0013 ± 0.0009
Barlow Twins Semi 5%					
12.63 ± 0.33	12.73 ± 0.29	21.44 ± 5.62	17.22 ± 3.01	112.74 ± 60.25	120.16 ± 90.98
11.38 ± 0.12	**11.32 ± 0.09**	**13.89 ± 1.27**	**14.92 ± 2.41**	112.18 ± 73.74	121.2 ± 72.03
17.09 ± 0.91	16.81 ± 0.35	18.62 ± 0.86	19.64 ± 0.93	**45.69 ± 8.75**	**53.56 ± 10.18**
Rotação Semi + Fine-tuning 5%					
0.0004 ± 0.0006	0.00007 ± 0.00011	0.000680 ± 0.001456	0.461175 ± 1.029825	0.27729 ± 0.37968	0.554572 ± 0.310015
0.0002 ± 0.0001	**0.00001 ± 0.00001**	**0.000004 ± 0.000003**	0.000007 ± 0.000008	0.00002 ± 0.00001	**0.000004 ± 0.000005**
0.0020 ± 0.0024	0.00011 ± 0.00012	0.000026 ± 0.000040	**0.000005 ± 0.000003**	**0.00001 ± 0.00001**	0.000006 ± 0.000002
Barlow Twins Semi + Fine-tuning 5%					
0.0075 ± 0.0064	0.0036 ± 0.0057	0.0033 ± 0.0046	**0.000003 ± 0.000004**	0.00233 ± 0.00402	0.00013 ± 0.00018
0.0056 ± 0.0056	**0.0009 ± 0.0016**	0.0145 ± 0.0321	0.000027 ± 0.000040	0.00219 ± 0.00349	0.00020 ± 0.00013
0.0011 ± 0.0014	0.0016 ± 0.0021	**0.0031 ± 0.0025**	0.000568 ± 0.001127	**0.00009 ± 0.00006**	**0.00001 ± 0.00000**

dados rotulados. Para o caso de 5% isto ocorreu apenas para a tarefa Barlow Twin. A tarefa de rotação sozinha obteve acurácia pior que o modelo supervisionado treinado em 5% dos dados rotulados.

Ao utilizar 5% de dados rotulados, o método semi-supervisionado com a tarefa de rotação obteve acurácia melhor que o modelo Rotação + Fine-tuning apenas no cenário sem ataque. Para o caso do Barlow Twins, esta vantagem existe para o cenário em que temos ataque, enquanto que o fine-tuning e o modelo semi-supervisionado ficam estatisticamente empatados.

4.3.2 Fashion-MNIST

Quando utilizamos 1% de dados rotulados disponíveis, entre os modelos que utilizam a tarefa de Rotação, o modelo Semi-supervisionado foi o que obteve melhor média de acurácia, tanto no cenário sem ataque como com ataque. O mesmo não ocorreu entre os modelos que utilizam a tarefa Barlow Twins. Neste caso o modelo Barlow-Twins + Fine-tuning foi o que teve melhor acurácia nos cenários sem e com ataque.

A maior acurácia obtida para a base de dados Fashion-MNIST para o caso 1% foi do método semi-supervisionado ao utilizar a tarefa Rotação.

Tabela 4 – Acurácia do SVM nos datasets STL-10, Fashion-MNIST e Malaria, quando disponíveis 1% de dados rotulados.

	STL-10	STL-10 com ataque	Fashion	Fashion com ataque	Malaria	Malaria com ataque
Supervisionado	36.7 ± 0.4	35.4 ± 0.5	79.8 ± 0.8	75.7 ± 3.1	63.4 ± 10.8	66.7 ± 17.1
Rotação	37.6 ± 1	-	70.9 ± 2.9	-	68 ± 1.4	-
Rotação + Fine-tuning	50.7 ± 1	51.1 ± 0.9	77.6 ± 0.9	77 ± 1.1	**88.4 ± 2.6**	**81.9 ± 1.7**
Rotação Semi	58.7 ± 1.3	56.5 ± 3.3	**83.1 ± 0.8**	**78.2 ± 0.7**	69.8 ± 2.8	66.3 ± 2.3
Rotação Semi + Fine-tuning	55.3 ± 2.6	46.8 ± 4.4	81.3 ± 1.4	76.4 ± 3.1	72.8 ± 1.8	74.6 ± 3
Barlow Twins	65.5 ± 0.7	-	71.7 ± 3	-	71.6 ± 6.9	-
Barlow Twins + Fine-tuning	**69.5 ± 0.6**	64.4 ± 0.4	81.1 ± 1.1	76.4 ± 1.4	67.5 ± 6.7	65.3 ± 4
Barlow Twins Semi	**70.3 ± 0.6**	**70.4 ± 0.2**	72 ± 3.7	73.6 ± 1.3	55.8 ± 5.2	55.8 ± 6.7
Barlow Twins Semi + Fine-tuning	62.8 ± 2.4	61.5 ± 3	79.3 ± 2.5	71.8 ± 3.1	63.8 ± 5.2	61.9 ± 6.2

Tabela 5 – Acurácia do SVM nos datasets STL-10, Fashion-MNIST e Malaria, quando disponíveis 5% de dados rotulados.

	STL-10	STL-10 com ataque	Fashion	Fashion com ataque	Malaria	Malaria com ataque
Supervisionado	52.2 ± 0.8	47.4 ± 3.8	86.5 ± 0.4	76.8 ± 1.3	**94.1 ± 0.2**	57.4 ± 2.4
Rotação	41.5 ± 1.2	-	78 ± 1.6	-	71.8 ± 2.5	-
Rotação + Fine-tuning	63.9 ± 2.6	60.5 ± 3.3	87.5 ± 0.1	**83 ± 0.5**	**93.9 ± 1.1**	**90.3 ± 0.7**
Rotação Semi	71 ± 0.6	60.2 ± 3.3	**88.3 ± 0.2**	**82 ± 1.1**	87.2 ± 3.5	60.5 ± 5.2
Rotação Semi + Fine-tuning	70.2 ± 0.3	58. ± 2.4	**88 ± 0.2**	**83 ± 0.5**	89 ± 3.1	71.6 ± 4.9
Barlow Twins	73.1 ± 0.4	-	77.3 ± 2.1	-	75 ± 7.6	-
Barlow Twins + Fine-tuning	**77.8 ± 0.4**	65.3 ± 1.8	86.7 ± 0.4	79.9 ± 0.8	89 ± 4	73.9 ± 9.6
Barlow Twins Semi	77 ± 0	**76.1 ± 1**	77.9 ± 1	78.6 ± 1.3	59.7 ± 2.7	61.5 ± 1.2
Barlow Twins Semi + Fine-tuning	71.1 ± 0.6	64.5 ± 0.5	84 ± 1.6	72.7 ± 6.4	90.7 ± 2.7	73.2 ± 7.1

Quando utilizamos 5% de dados rotulados disponíveis, obtemos resultados estatisticamente semelhantes entre os modelos que utilizam a tarefa de Rotação, com exceção da Rotação ao utilizar apenas os dados não rotulados, que obteve acurácia menor. Para os modelos que utilizaram a tarefa Barlow Twins com 5% de dados disponíveis, o modelo Barlow-Twins + Fine-tuning foi o que teve melhor acurácia nos cenários sem e com ataque.

Aqui nota-se que as tarefas auto-supervisionadas sozinhas não obtêm acurácia melhor que o método supervisionado em nenhum cenário, como ocorre para a base de dados STL-10.

Com a tarefa Barlow Twins, o Fine-tuning do modelo Semi-supervisionado trouxe melhoria para os cenários sem ataque tanto no caso 1% como no 5%. Interessante notar que com a tarefa Barlow Twins, apesar do fine-tuning do modelo Semi-supervisonado melhorar a acurácia nos cenários sem ataque, o mesmo não ocorre nos cenários com ataque, indicando que o treinamento simultâneo neste caso trouxe benefícios para o aprendizado mais robusto do modelo quando temos dados com ruídos.

4.3.3 Malaria

Quando utilizamos 1% de dados rotulados disponíveis, entre os modelos que utilizam a tarefa de Rotação, o modelo Rotação + Fine-tuning foi o que obteve melhor acurácia, tanto no cenário sem ataque como com ataque. O mesmo se observa entre os modelos que utilizam a tarefa Barlow Twins com 1%: o modelo Barlow Twins + Fine-tuning foi o que teve melhor acurácia nos dois cenários. Porém, nota-se aqui que o modelo treinado apenas na tarefa Barlow Twins atingiu maior acurácia do que os modelos que utilizaram 1% de dados rotulados, tanto em comparação com modelos que fazem fine-tuning como com os modelos semi-supervisionados. Ou seja, para a tarefa Barlow Twins a utilização de 1% de dados rotulados piorou a acurácia dos modelos que utilizaram os dados rotulados juntamente com os dados não rotulados.

A maior acurácia obtida para a base de dados Malaria para o caso 1% foi do método Rotação + Fine-tuning.

Ao contrário do que geralmente ocorreu com as outras bases de dados, aqui o Fine-tuning do modelo Semi-supervisionado traz uma melhoria para a acurácia em todos os cenários, em ambas as tarefas e tanto com 1% como com 5% de dados rotulados.

Nota-se aqui que o desvio padrão para os modelos supervisionados ao utilizar 1% de dados são grandes, indicando treinamentos instáveis em comparação com os outros experimentos em geral. O mesmo não ocorre no modelo supervisionado quando se utiliza 5% de dados rotulados.

Observamos que para essa base de dados a maioria dos modelos que utilizaram a tarefa de Rotação tiveram acurácia maior em comparação ao modelo supervisionado no caso 1%. Neste caso esta situação provavelmente ocorreu pois a quantidade de dados rotulados era extremamente pequena, pois, ao aumentar a quantidade de dados rotulados para 5%, os modelos que utilizaram

tarefas auto-supervisionadas não obtiveram melhor acurácia que o modelo supervisionado, no cenário sem ataque. Já no cenário com ataque, o modelo supervisionado teve a pior acurácia entre todos os modelos quando temos 5% de dados disponíveis. O modelo Rotação + Fine-tuning foi capaz de atingir mais de 90% de acurácia, quase chegando na acurácia do cenário sem ataque.

Aqui as tarefas auto-supervisionadas sozinhas atingem maior acurácia que o modelo supervisionado no caso 1%, mas não para 5%.

4.3.4 Observações / considerações

A tarefa Barlow Twins utiliza *data augmentations* que podem beneficiar cenários em que temos imagens naturais coloridas, talvez sendo o motivo pelo qual obtivemos bons resultados para a base de dados STL-10. Provavelmente alterando os tipos de *data augmentations* poderíamos obter resultados melhores para as bases de dados Fashion-MNIST e Malaria.

Ainda, há um parâmetro também que pode ser ajustado no modelo Semi-supervisionado que pode melhorar os resultados, que é a proporção de peso que atribuímos para a tarefa supervisionada e a tarefa de auto-supervisão na função de custo.

Porém, balancear e encontrar esses parâmetros (o conjunto de transformações utilizadas para a tarefa Barlow Twins e a atribuição dos pesos na função de custo) seria extremamente custoso computacionalmente. Apesar de talvez ideal, fugiria do escopo deste trabalho executar experimentos com as diversas combinações de parâmetros possíveis.

No geral, a tarefa Barlow Twins teve performance melhor para a base de dados STL-10. Todos os experimentos que utilizaram a tarefa Barlow Twins tiveram performance melhor que a tarefa de Rotação para a base de dados STL-10. No geral, a tarefa Rotação teve performance melhor para as bases de dados Fashion e Malaria. Os modelos Semi-supervisionados tiveram melhor desempenho no geral, seguido pelos modelos que utilizaram alguma tarefa de auto-supervisão + Fine-tuning.

4.3.5 Gráficos das funções de custo de treinamento

Na Figura 11 temos os gráficos das funções de treinamento para a base de dados STL-10 ao utilizar 5% de dados rotulados. Os gráficos dos demais treinamento seguem os padrões destes exemplo, ou seja, diminuição da função de custo com posterior variação baixa entre épocas subsequentes no final do treinamento, motivo pelo qual mostraremos apenas estes. Importante observar que a função de custo da tarefa de Barlow Twins tem valores altos em comparação com o treinamento supervisionado e com a tarefa Rotação.

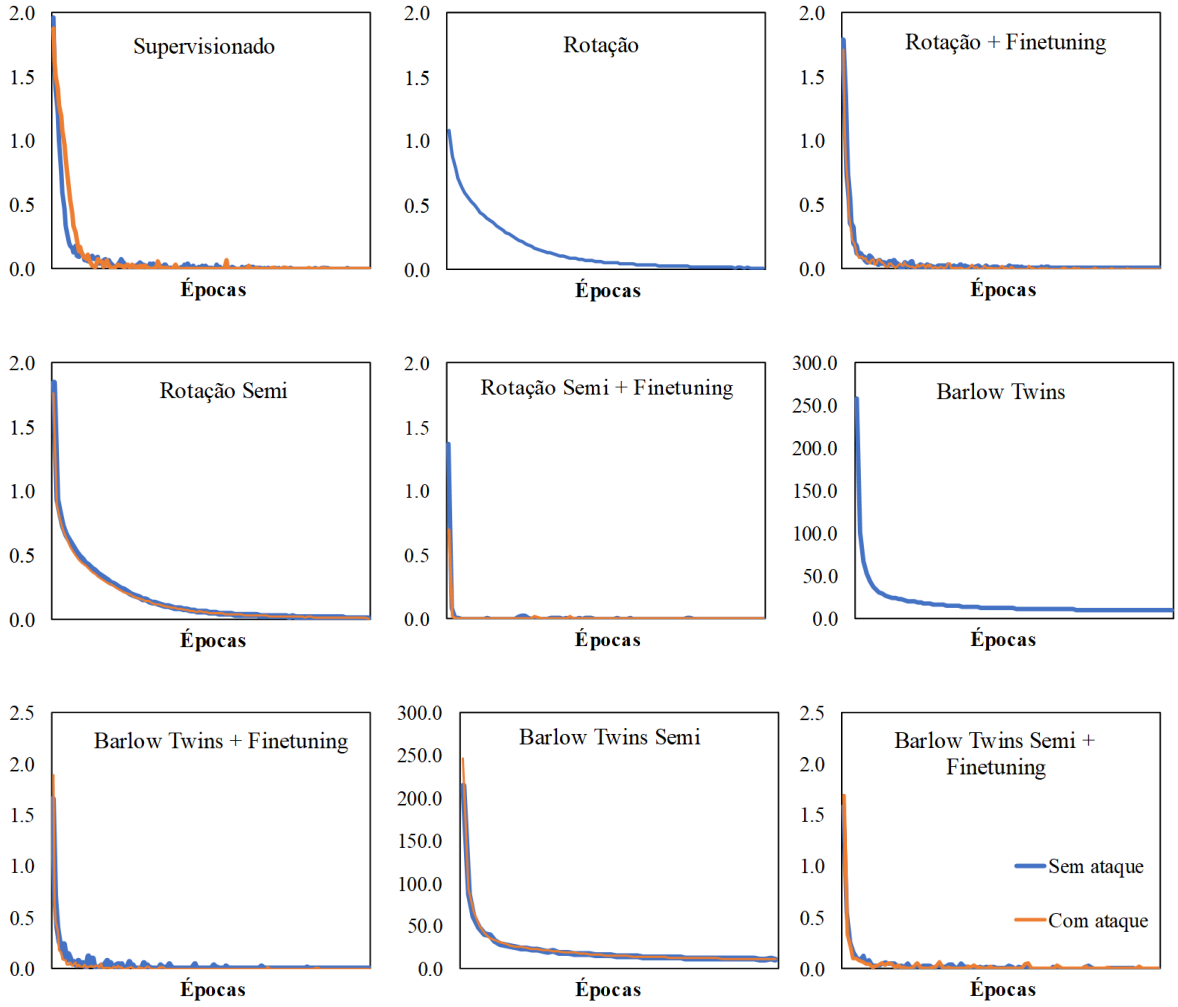


Figura 11 – Exemplos de gráficos das funções de custo para a base de dados STL-10 ao utilizar 5% de dados rotulados. Estão mostrados aqui os gráficos dos modelos que obtiveram os menores valores da função de custo na última época de treinamento. Porém, os demais modelos não mostrados aqui também seguem os padrões mostrados nos gráficos acima, razão pela qual mostraremos apenas estes exemplos.

4.3.6 Visualização do espaço de características com tSNE

Escolhemos alguns casos específicos para realizar a visualização do espaço de características aprendido, nos casos Barlow Twins 5% na base de dados STL-10, Rotação 1% na base de dados Fashion-MNIST e Rotação 5% na base de dados Malaria. Estão representados nas Figuras 12, 13 e 14, respectivamente.

Como podemos ver na Figura 12 o método semi-supervisionado sofre menos alterações no espaço de características ao comparar o método sem e com ataque, e as classes estão melhor separadas do que o espaço do caso supervisionado. O método fine-tuning obtém melhor separação entre as classes do que o método semi-supervisionado no caso sem ataque, porém ao introduzir ataque o método semi-supervisionado é mais robusto.

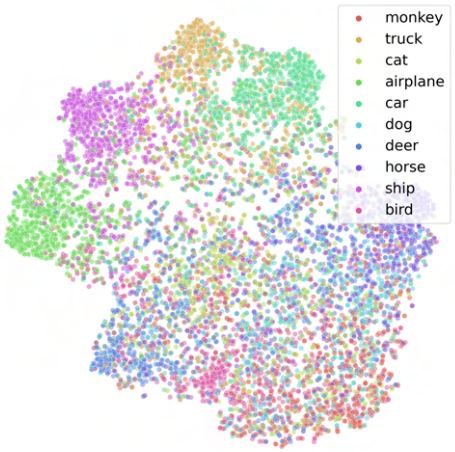
Na Figura 13 o método semi-supervisionado separa um pouco melhor as classes ca-

saco, pulôver e camisa em comparação com o método supervisionado e fine-tuning. O método supervisionado sofre mais na situação com ataque.

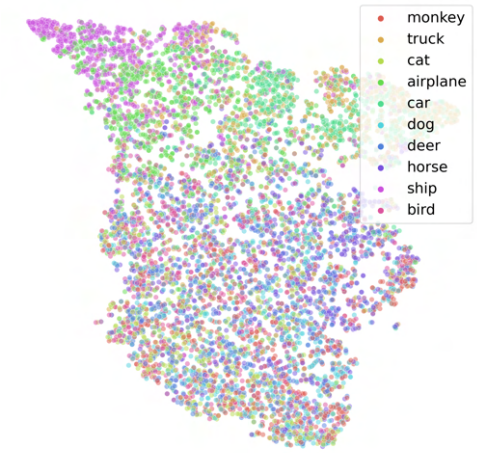
Na Figura 14 observamos que tanto o método supervisionado como o semi-supervisionado sofrem com o ataque, o que não ocorreu no método de fine-tuning.

Em geral, as visualizações estão consistentes com os resultados obtidos com o SVM.

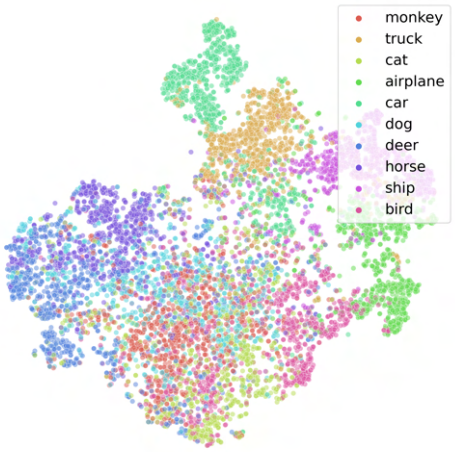
(a) Supervisionado 5% sem ataque



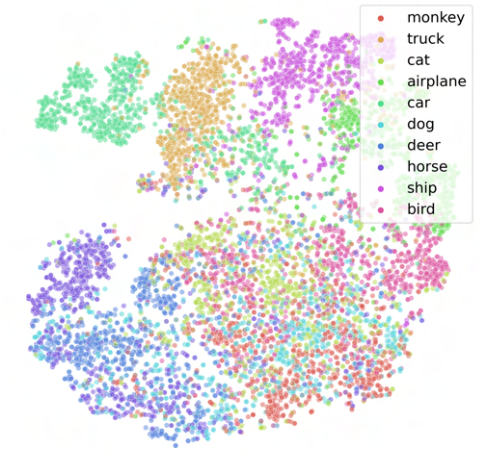
(b) Supervisionado 5% com ataque



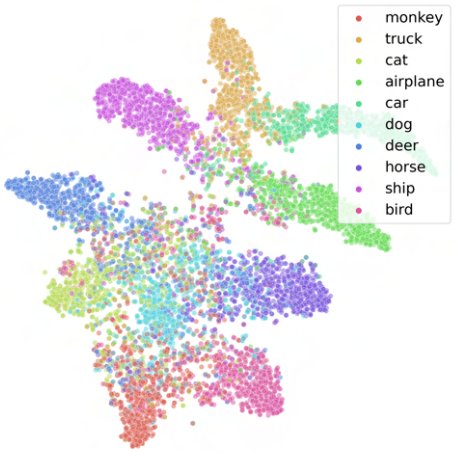
(c) Barlow Twins Semi 5% sem ataque



(d) Barlow Twins Semi 5% com ataque



(e) Barlow Twins Fine-tuning 5% sem ataque



(f) Barlow Twins Fine-tuning 5% com ataque

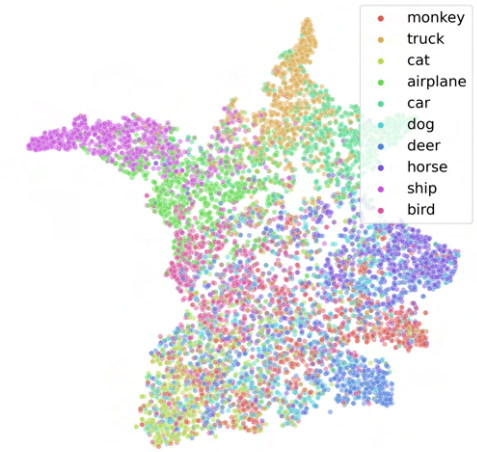
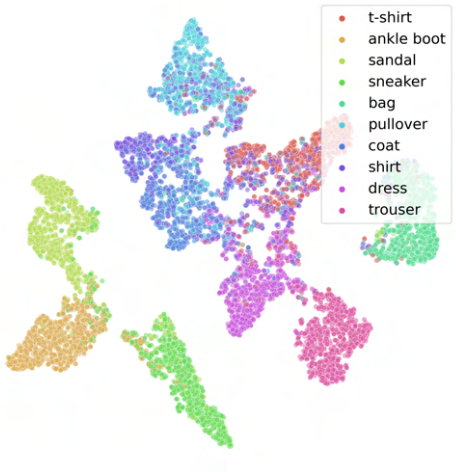
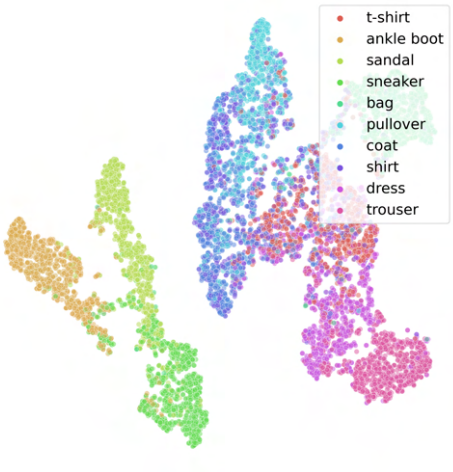


Figura 12 – Visualização do espaço de características com tSNE dos métodos Supervisionados, Barlow Twins Semi-supervisionado e Barlow Twins + Fine-tuning na base de dados STL-10 ao utilizar 5% de dados rotulados.

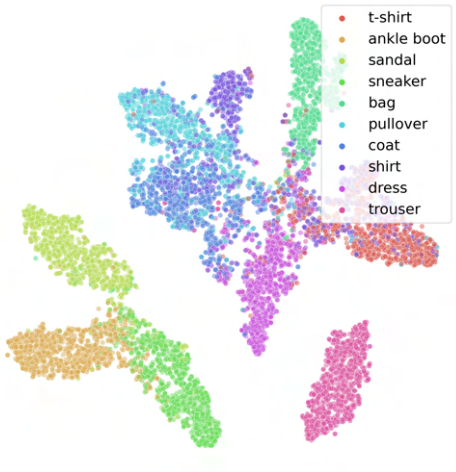
(a) Supervisionado 1% sem ataque



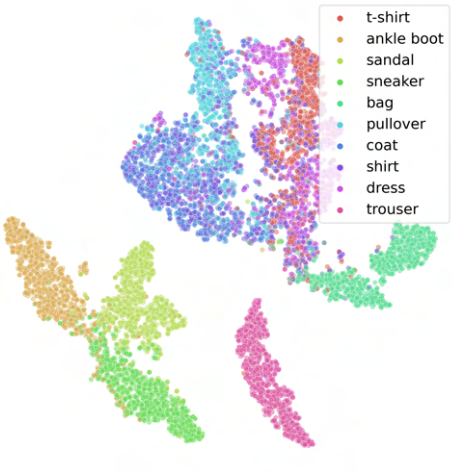
(b) Supervisionado 1% com ataque



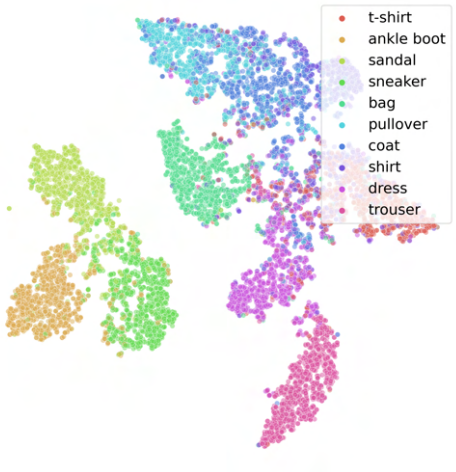
(c) Rotação Semi 1% sem ataque



(d) Rotação Semi 1% com ataque



(e) Rotação Fine-tuning 1% sem ataque



(f) Rotação Fine-tuning 1% com ataque

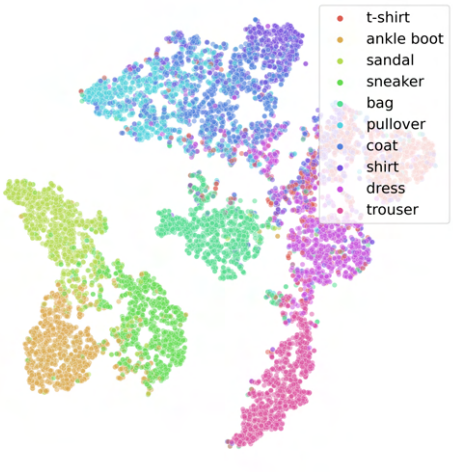
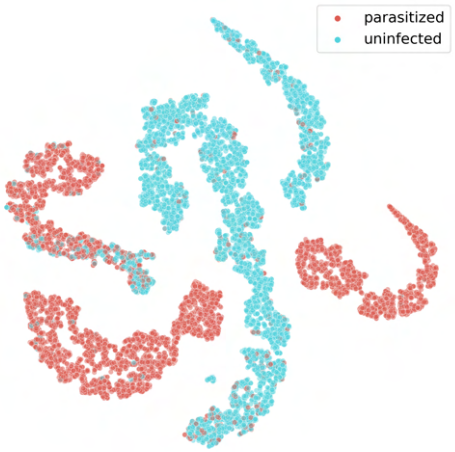
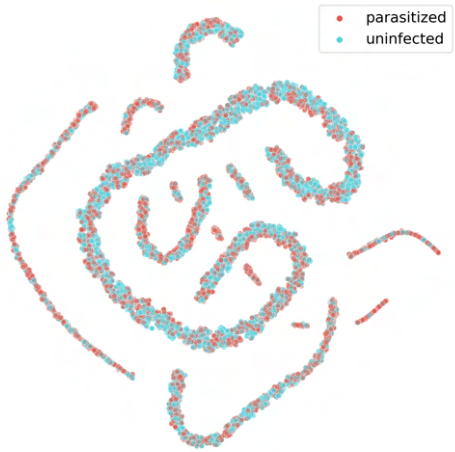


Figura 13 – Visualização do espaço de características com tSNE dos métodos Supervisionados, Rotação Semi-supervisionado e Rotação + Fine-tuning na base de dados Fashion-MNIST ao utilizar 1% de dados rotulados.

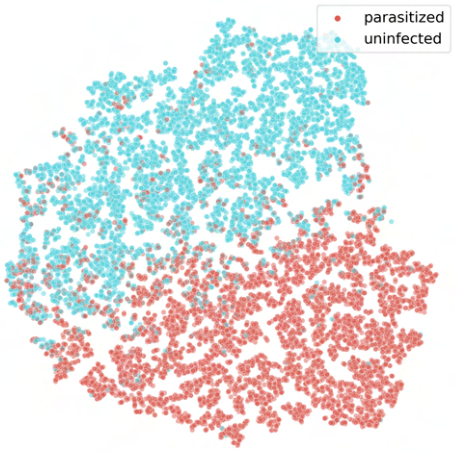
(a) Supervisionado 5% sem ataque



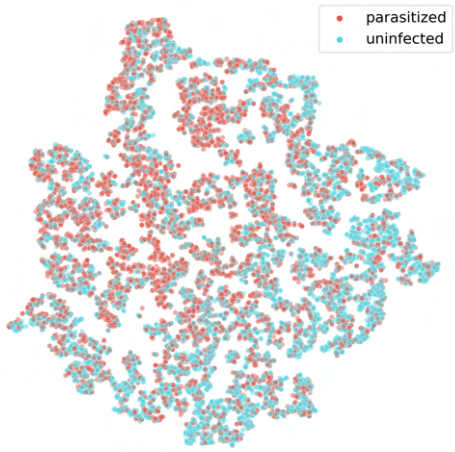
(b) Supervisionado 5% com ataque



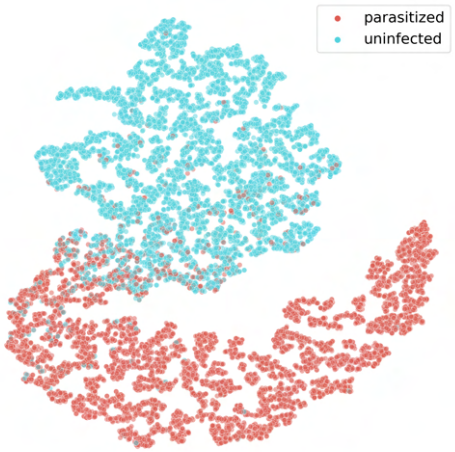
(c) Rotação Semi 5% sem ataque



(d) Rotação Semi 5% com ataque



(e) Rotação Fine-tuning 5% sem ataque



(f) Rotação Fine-tuning 5% com ataque

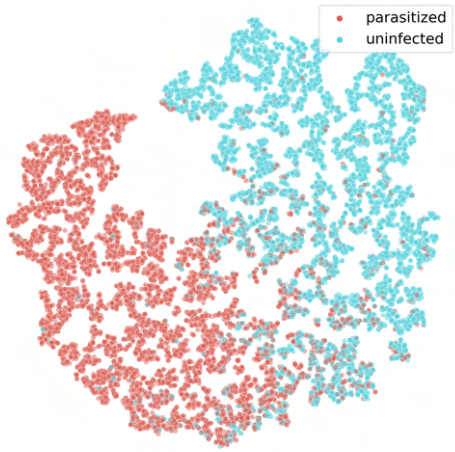


Figura 14 – Visualização do espaço de características com tSNE dos métodos Supervisionados, Rotação Semi-supervisionado e Rotação + Fine-tuning na base de dados Malaria ao utilizar 5% de dados rotulados.

CAPÍTULO

5

CONCLUSÃO

A obtenção de grandes bases de dados anotadas ainda é um desafio, especialmente para problemas reais em que dados anotados são escassos. Investigar diferentes estratégias de treinamentos que não necessitam de dados rotulados, ou que diminuam a necessidade de dados rotulados é importante. Nesse cenário, o aprendizado de representações de forma auto-supervisionada e semi-supervisionada aparecem como uma alternativa para auxiliar neste desafio.

Este projeto explorou o aprendizado de representações de forma semi-supervisionada e com tarefas da área de auto-supervisão. Aproveitando dados não rotulados em cenários de limitação de dados rotulados, foram exploradas duas tarefas de auto-supervisão: predição de rotação e a tarefa Barlow Twins. A tarefa de rotação é mais simples conceitualmente, enquanto que a tarefa Barlow Twins é mais atual e comparável ao estado-da-arte em representações de imagens utilizando dados não rotulados. O aprendizado de representações tem como um de seus objetivos melhorar a separabilidade das classes no espaço de características. Esta avaliação pode ser feita utilizando um classificador linear treinado nas características extraídas pelos modelos aprendidos. Neste trabalho utilizou-se o SVM como o classificador linear para esta análise.

Exploramos combinações de diversos estilos de treinamento. O cenário básico foi utilizar apenas os dados rotulados disponíveis. Outro cenário foi utilizar as tarefas auto-supervisionadas para realizar um pré-treinamento. Aproveitando este modelo pré-treinado, analisamos também o efeito ao realizar fine-tuning com dados rotulados deste modelo pré-treinado com auto-supervisão. Além disso, exploramos uma função de custo que considera tanto dados não rotulados como dados rotulados, ao utilizar uma arquitetura siamesa que treina uma tarefa auto-supervisionada em conjunto com a classificação comum. Por fim, realizamos um fine-tuning desta arquitetura semi-supervisionada com os dados rotulados para saber se melhoraria ainda mais o resultado.

De modo geral, entre todos os estilos de treinamento explorados, o modelo semi-supervisionado foi o que obteve melhor acurácia, seguido pelos modelos que utilizaram fine-tuning de uma tarefa auto-supervisionada como pré-treinamento.

Observamos que a escolha da tarefa deve levar em consideração a natureza da base de dados. Como verificado neste trabalho, diferentes tarefas de auto-supervisão têm desempenhos diferentes dependendo da base de dados. A tarefa Barlow Twins teve performance melhor para a base de dados STL-10. A tarefa Rotação teve performance melhor para as bases de dados Fashion e Malaria. Para as imagens da base de dados Malaria, há duas observações importantes: por elas não serem imagens que são orientadas pelo ângulo, a tarefa de rotação fica mais difícil. Mesmo assim, vemos benefícios ao incorporar dados não rotulados. Além disso, o ataque teve um impacto ainda maior para essa base de dados por conta da natureza das imagens dessa base, que contêm padrões similares ao ataque, piorando os resultados.

A hipótese inicial deste trabalho era a de que, quando comparado ao aprendizado supervisionado, métodos que utilizam auto-supervisão seguido de fine-tuning e métodos semi-supervisionados em que as tarefas são aprendidas em conjunto, obtêm representações com melhor separação linear e robustez ao ataque. Os resultados dos experimentos realizados reforçam a hipótese inicial do trabalho.

O aprendizado auto-supervisionado demonstrou ser útil em cenários semi-supervisionados, não apenas para melhorar os resultados numéricos, mas principalmente para aprender espaços mais discriminativos, bem como uma representação mais robusta em relação ao ataque. Nossos resultados mostraram que a escolha da tarefa auxiliar deve levar em consideração a natureza das imagens e pode não se adequar a todas as aplicações.

Tal como em trabalhos anteriores e recentes, a auto-supervisão surge como um método relevante para permitir a aprendizagem a partir de dados minimamente anotados. Mais do que isso, quando usado durante o processo de aprendizado, pode ajudar a melhorar a robustez contra ataques. Trabalhos futuros podem investigar outros tipos de tarefas auxiliares no contexto de aprendizagem semi-supervisionada, bem como robustez frente a outros cenários indesejados. Em particular, acreditamos que projetar tarefas auxiliares que sejam adequadas a cada base de dados e problema seja um caminho promissor.

Outros cenários que podem ser explorados incluem diferentes formas de investigar a partição de dados rotulados utilizados para os treinamentos. Trabalhos futuros podem explorar outras formas de selecionar as partições rotuladas. Uma opção seria selecionar os melhores exemplos com base em algum critério, ou então explorar também casos em que há desbalanceamento entre classes.

REFERÊNCIAS

ARJOVSKY, M.; CHINTALA, S.; BOTTOU, L. Wasserstein generative adversarial networks. In: PMLR. **International conference on machine learning**. [S.l.], 2017. p. 214–223. Citado na página 26.

ATHIWARATKUN, B.; FINZI, M.; IZMAILOV, P.; WILSON, A. G. There are many consistent explanations of unlabeled data: Why you should average. **ICLR**, 2019. Citado na página 33.

BACHMAN, P.; HJELM, R. D.; BUCHWALTER, W. Learning representations by maximizing mutual information across views. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2019. p. 15509–15519. Citado na página 29.

BENGIO, Y.; COURVILLE, A.; VINCENT, P. Representation learning: A review and new perspectives. **IEEE transactions on pattern analysis and machine intelligence**, IEEE, v. 35, n. 8, p. 1798–1828, 2013. Citado nas páginas 19 e 25.

BERTHELOT, D.; CARLINI, N.; CUBUK, E. D.; KURAKIN, A.; SOHN, K.; ZHANG, H.; RAFFEL, C. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. **ArXiv**, abs/1911.09785, 2019. Citado na página 34.

BERTHELOT, D.; CARLINI, N.; GOODFELLOW, I.; PAPERNOT, N.; OLIVER, A.; RAFFEL, C. A. Mixmatch: A holistic approach to semi-supervised learning. **Advances in Neural Information Processing Systems**, v. 32, 2019. Citado na página 33.

BLUM, A.; MITCHELL, T. Combining labeled and unlabeled data with co-training. In: **Proceedings of the eleventh annual conference on Computational learning theory**. [S.l.: s.n.], 1998. p. 92–100. Citado na página 33.

CARON, M.; BOJANOWSKI, P.; JOULIN, A.; DOUZE, M. Deep clustering for unsupervised learning of visual features. In: **Proceedings of the European Conference on Computer Vision (ECCV)**. [S.l.: s.n.], 2018. p. 132–149. Citado na página 29.

CARON, M.; MISRA, I.; MAIRAL, J.; GOYAL, P.; BOJANOWSKI, P.; JOULIN, A. Unsupervised learning of visual features by contrasting cluster assignments. 2020. Citado na página 29.

CAVALLARI, G. B.; PONTI, M. A. Semi-supervised siamese network using self-supervision under scarce annotation improves class separability and robustness to attack. In: **2021 34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)**. [S.l.: s.n.], 2021. p. 223–230. Citado na página 34.

CHEN, T.; KORNBLITH, S.; NOROUZI, M.; HINTON, G. A simple framework for contrastive learning of visual representations. In: PMLR. **International conference on machine learning**. [S.l.], 2020. p. 1597–1607. Citado na página 30.

CHEN, X.; HE, K. Exploring simple siamese representation learning. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2021. p. 15750–15758. Citado na página 30.

COATES, A.; NG, A.; LEE, H. An analysis of single-layer networks in unsupervised feature learning. In: **Proceedings of the fourteenth international conference on artificial intelligence and statistics**. [S.l.: s.n.], 2011. p. 215–223. Citado na página 20.

DAI, Z.; YANG, Z.; YANG, F.; COHEN, W. W.; SALAKHUTDINOV, R. R. Good semi-supervised learning that requires a bad gan. **Advances in neural information processing systems**, v. 30, 2017. Citado na página 32.

DENTON, E.; GROSS, S.; FERGUS, R. Semi-supervised learning with context-conditional generative adversarial networks. **arXiv preprint arXiv:1611.06430**, 2016. Citado na página 32.

DOERSCH, C.; GUPTA, A.; EFROS, A. A. Unsupervised visual representation learning by context prediction. In: **Proceedings of the IEEE International Conference on Computer Vision**. [S.l.: s.n.], 2015. p. 1422–1430. Citado na página 28.

DONAHUE, J.; KRÄHENBÜHL, P.; DARRELL, T. Adversarial feature learning. **arXiv preprint arXiv:1605.09782**, 2016. Citado nas páginas 26 e 32.

DONAHUE, J.; SIMONYAN, K. Large scale adversarial representation learning. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2019. p. 10541–10551. Citado nas páginas 20 e 27.

DONG-DONGCHEN, W.; WEIGAO, Z.-H. Tri-net for semi-supervised deep learning. In: **Proceedings of twenty-seventh international joint conference on artificial intelligence**. [S.l.: s.n.], 2018. p. 2014–2020. Citado na página 33.

DOSOVITSKIY, A.; FISCHER, P.; SPRINGENBERG, J. T.; RIEDMILLER, M.; BROX, T. Discriminative unsupervised feature learning with exemplar convolutional neural networks. **IEEE transactions on pattern analysis and machine intelligence**, IEEE, v. 38, n. 9, p. 1734–1747, 2015. Citado na página 28.

DUMOULIN, V.; BELGHAZI, I.; POOLE, B.; MASTROPIETRO, O.; LAMB, A.; ARJOVSKY, M.; COURVILLE, A. Adversarially learned inference. **arXiv preprint arXiv:1606.00704**, 2016. Citado na página 32.

GIDARIS, S.; SINGH, P.; KOMODAKIS, N. Unsupervised representation learning by predicting image rotations. **arXiv preprint arXiv:1803.07728**, 2018. Citado nas páginas 21 e 28.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A.; BENGIO, Y. **Deep learning**. [S.l.]: MIT press Cambridge, 2016. v. 1. Citado nas páginas 20 e 23.

GOODFELLOW, I. J.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. generative adversarial nets. In: **Advances in Neural Information Processing Systems 27**. [S.l.: s.n.], 2014. Citado na página 26.

GRANDVALET, Y.; BENGIO, Y. Semi-supervised learning by entropy minimization. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2005. p. 529–536. Citado na página 33.

GRILL, J.-B.; STRUB, F.; ALTCHÉ, F.; TALLEC, C.; RICHEMOND, P. H.; BUCHATSKAYA, E.; DOERSCH, C.; PIRES, B. A.; GUO, Z. D.; AZAR, M. G. *et al.* Bootstrap your own latent: A new approach to self-supervised learning. **arXiv preprint arXiv:2006.07733**, 2020. Citado na página 30.

GUTMANN, M.; HYVÄRINEN, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: JMLR WORKSHOP AND CONFERENCE PROCEEDINGS. **Proceedings of the thirteenth international conference on artificial intelligence and statistics**. [S.l.], 2010. p. 297–304. Citado na página 29.

HE, K.; FAN, H.; WU, Y.; XIE, S.; GIRSHICK, R. Momentum contrast for unsupervised visual representation learning. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2020. p. 9729–9738. Citado na página 30.

HENDRYCKS, D.; MAZEIKA, M.; KADAVATH, S.; SONG, D. Using self-supervised learning can improve model robustness and uncertainty. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2019. Citado na página 34.

HINTON, G. E.; SALAKHUTDINOV, R. R. Reducing the dimensionality of data with neural networks. **Science**, American Association for the Advancement of Science, v. 313, n. 5786, p. 504–507, 2006. Citado na página 26.

HJELM, R. D.; FEDOROV, A.; LAVOIE-MARCHILDON, S.; GREWAL, K.; BACHMAN, P.; TRISCHLER, A.; BENGIO, Y. Learning deep representations by mutual information estimation and maximization. **arXiv preprint arXiv:1808.06670**, 2018. Citado na página 29.

JING, L.; TIAN, Y. Self-supervised visual feature learning with deep neural networks: A survey. **arXiv preprint arXiv:1902.06162**, 2019. Citado nas páginas 20, 28 e 35.

KE, Z.; WANG, D.; YAN, Q.; REN, J.; LAU, R. W. Dual student: Breaking the limits of the teacher in semi-supervised learning. In: **Proceedings of the IEEE/CVF International Conference on Computer Vision**. [S.l.: s.n.], 2019. p. 6728–6736. Citado na página 33.

KINGMA, D. P.; MOHAMED, S.; REZENDE, D. J.; WELLING, M. Semi-supervised learning with deep generative models. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2014. p. 3581–3589. Citado na página 32.

KINGMA, D. P.; WELLING, M. Auto-encoding variational bayes. **arXiv preprint arXiv:1312.6114**, 2013. Citado na página 26.

LAINE, S.; AILA, T. Temporal ensembling for semi-supervised learning. **arXiv preprint arXiv:1610.02242**, 2016. Citado na página 33.

LEE, D.-H. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: **Workshop on challenges in representation learning, ICML**. [S.l.: s.n.], 2013. v. 3, p. 2. Citado na página 33.

LI, C.; XU, T.; ZHU, J.; ZHANG, B. Triple generative adversarial nets. **Advances in neural information processing systems**, v. 30, 2017. Citado na página 32.

LI, J.; SOCHER, R.; HOI, S. C. Dividemix: Learning with noisy labels as semi-supervised learning. **arXiv preprint arXiv:2002.07394**, 2020. Citado na página 34.

LIU, X.; ZHANG, F.; HOU, Z.; MIAN, L.; WANG, Z.; ZHANG, J.; TANG, J. Self-supervised learning: Generative or contrastive. **IEEE Transactions on Knowledge and Data Engineering**, IEEE, 2021. Citado nas páginas 13, 27 e 28.

MAALØE, L.; SØNDERBY, C. K.; SØNDERBY, S. K.; WINTHER, O. Auxiliary deep generative models. In: PMLR. **International conference on machine learning**. [S.l.], 2016. p. 1445–1453. Citado na página 32.

MELLO, R. F. de; FERREIRA, M. D.; PONTI, M. A. Providing theoretical learning guarantees to deep learning networks. **arXiv preprint arXiv:1711.10292**, 2017. Citado na página 20.

MELLO, R. F. de; PONTI, M. A. **Machine Learning: A Practical Approach on the Statistical Learning Theory**. [S.l.]: Springer, 2018. Citado na página 42.

MIYATO, T.; MAEDA, S.-i.; KOYAMA, M.; ISHII, S. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. **IEEE transactions on pattern analysis and machine intelligence**, IEEE, v. 41, n. 8, p. 1979–1993, 2018. Citado na página 33.

MUNDHENK, T. N.; HO, D.; CHEN, B. Y. Improvements to context based self-supervised learning. In: **CVPR**. [S.l.: s.n.], 2018. p. 9339–9348. Citado na página 28.

NAZARÉ, T. S.; COSTA, G. B.; CONTATO, W. A.; PONTI, M. Deep convolutional neural networks and noisy images. In: SPRINGER. **Iberoamerican Congress on Pattern Recognition**. [S.l.], 2017. p. 416–424. Citado na página 20.

NOROOZI, M.; FAVARO, P. Unsupervised learning of visual representations by solving jigsaw puzzles. In: SPRINGER. **European Conference on Computer Vision**. [S.l.], 2016. p. 69–84. Citado na página 28.

NOROOZI, M.; VINJIMOOR, A.; FAVARO, P.; PIRSIAVASH, H. Boosting self-supervised learning via knowledge transfer. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2018. p. 9359–9367. Citado na página 28.

ODENA, A. Semi-supervised learning with generative adversarial networks. **arXiv preprint arXiv:1606.01583**, 2016. Citado na página 32.

OLIVER, A.; ODENA, A.; RAFFEL, C. A.; CUBUK, E. D.; GOODFELLOW, I. Realistic evaluation of deep semi-supervised learning algorithms. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2018. p. 3235–3246. Citado na página 31.

OORD, A. van den; VINYALS, O.; KAVUKCUOGLU, K. Neural discrete representation learning. In: **NIPS**. [S.l.: s.n.], 2017. Citado na página 26.

OUALI, Y.; HUDELOT, C.; TAMI, M. An overview of deep semi-supervised learning. **arXiv preprint arXiv:2006.05278**, 2020. Citado na página 31.

PAIGE, B.; MEENT, J.-W. van de; DESMAISON, A.; GOODMAN, N.; KOHLI, P.; WOOD, F.; TORR, P. *et al.* Learning disentangled representations with semi-supervised deep generative models. **Advances in neural information processing systems**, v. 30, 2017. Citado na página 32.

PATHAK, D.; KRAHENBUHL, P.; DONAHUE, J.; DARRELL, T.; EFROS, A. A. Context encoders: Feature learning by inpainting. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2016. p. 2536–2544. Citado na página 27.

PHAM, H.; DAI, Z.; XIE, Q.; LE, Q. V. Meta pseudo labels. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2021. p. 11557–11568. Citado na página 33.

PONTI, M.; NAZARÉ, T. S.; THUMÉ, G. S. Image quantization as a dimensionality reduction procedure in color and texture feature extraction. **Neurocomputing**, Elsevier, v. 173, p. 385–396, 2016. Citado na página 19.

PONTI, M. A.; COSTA, G. B. P. da. Como funciona o deep learning. **arXiv preprint arXiv:1806.07908**, 2018. Citado nas páginas 13 e 24.

PONTI, M. A.; SANTOS, F. P. dos; RIBEIRO, L. S.; CAVALLARI, G. B. Training deep networks from zero to hero: avoiding pitfalls and going beyond. In: IEEE. **2021 34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)**. [S.l.], 2021. p. 9–16. Citado na página 20.

QI, G.-J.; ZHANG, L.; HU, H.; EDRAKI, M.; WANG, J.; HUA, X.-S. Global versus localized generative adversarial nets. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2018. p. 1517–1525. Citado na página 32.

QIAO, S.; SHEN, W.; ZHANG, Z.; WANG, B.; YUILLE, A. Deep co-training for semi-supervised image recognition. In: **Proceedings of the european conference on computer vision (eccv)**. [S.l.: s.n.], 2018. p. 135–152. Citado na página 33.

RADFORD, A.; METZ, L.; CHINTALA, S. Unsupervised representation learning with deep convolutional generative adversarial networks. **arXiv preprint arXiv:1511.06434**, 2015. Citado nas páginas 26 e 31.

RAINA, R.; BATTLE, A.; LEE, H.; PACKER, B.; NG, A. Y. Self-taught learning: transfer learning from unlabeled data. In: ACM. **Proceedings of the 24th international conference on Machine learning**. [S.l.], 2007. p. 759–766. Citado na página 20.

RAJARAMAN, S.; ANTANI, S. K.; POOSTCHI, M.; SILAMUT, K.; HOSSAIN, M. A.; MAUDE, R. J.; JAEGER, S.; THOMA, G. R. Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images. **PeerJ**, PeerJ Inc., v. 6, p. e4568, 2018. Citado na página 40.

RASMUS, A.; BERGLUND, M.; HONKALA, M.; VALPOLA, H.; RAIKO, T. Semi-supervised learning with ladder networks. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2015. p. 3546–3554. Citado na página 32.

RAZAVI, A.; OORD, A. van den; VINYALS, O. Generating diverse high-fidelity images with vq-vae-2. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2019. p. 14866–14876. Citado na página 26.

RAZAVIAN, A. S.; AZIZPOUR, H.; SULLIVAN, J.; CARLSSON, S. Cnn features off-the-shelf: an astounding baseline for recognition. In: IEEE. **Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on**. [S.l.], 2014. p. 512–519. Citado na página 19.

SAJJADI, M.; JAVANMARDI, M.; TASDIZEN, T. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2016. p. 1163–1171. Citado na página 32.

SALAKHUTDINOV, R.; HINTON, G. Deep boltzmann machines. In: **Artificial intelligence and statistics**. [S.l.: s.n.], 2009. p. 448–455. Citado na página 26.

SALIMANS, T.; GOODFELLOW, I.; ZAREMBA, W.; CHEUNG, V.; RADFORD, A.; CHEN, X. Improved techniques for training gans. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2016. p. 2234–2242. Citado na página 32.

SANTOS, F. P. D.; ZOR, C.; KITTLER, J.; PONTI, M. A. Learning image features with fewer labels using a semi-supervised deep convolutional network. **Neural Networks**, Elsevier, v. 132, p. 131–143, 2020. Citado na página 31.

SMOLENSKY, P. **Information processing in dynamical systems: Foundations of harmony theory**. [S.l.], 1986. Citado na página 26.

SOHN, K.; BERTHELOT, D.; CARLINI, N.; ZHANG, Z.; ZHANG, H.; RAFFEL, C. A.; CUBUK, E. D.; KURAKIN, A.; LI, C.-L. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. **Advances in Neural Information Processing Systems**, v. 33, p. 596–608, 2020. Citado na página 34.

SPRINGENBERG, J. T. Unsupervised and semi-supervised learning with categorical generative adversarial networks. **CoRR**, abs/1511.06390, 2016. Citado na página 32.

TARVAINEN, A.; VALPOLA, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2017. p. 1195–1204. Citado na página 33.

TIAN, Y.; KRISHNAN, D.; ISOLA, P. Contrastive multiview coding. **arXiv preprint arXiv:1906.05849**, 2019. Citado na página 29.

TIAN, Y.; SUN, C.; POOLE, B.; KRISHNAN, D.; SCHMID, C.; ISOLA, P. What makes for good views for contrastive learning? **arXiv preprint arXiv:2005.10243**, 2020. Citado na página 30.

TRAN, P. V. **Exploring Self-Supervised Regularization for Supervised and Semi-Supervised Learning**. 2019. Citado na página 34.

VERMA, V.; LAMB, A.; KANNALA, J.; BENGIO, Y.; LOPEZ-PAZ, D. Interpolation consistency training for semi-supervised learning. In: **IJCAI**. [S.l.: s.n.], 2019. Citado na página 33.

VINCENT, P.; LAROCHELLE, H.; BENGIO, Y.; MANZAGOL, P.-A. Extracting and composing robust features with denoising autoencoders. In: **Proceedings of the 25th international conference on Machine learning**. [S.l.: s.n.], 2008. p. 1096–1103. Citado na página 26.

WEI, X.; GONG, B.; LIU, Z.; LU, W.; WANG, L. Improving the improved training of wasserstein gans: A consistency term and its dual effect. **arXiv preprint arXiv:1803.01541**, 2018. Citado na página 32.

WU, Z.; XIONG, Y.; YU, S. X.; LIN, D. Unsupervised feature learning via non-parametric instance discrimination. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2018. p. 3733–3742. Citado na página 29.

XIAO, H.; RASUL, K.; VOLLGRAF, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. **arXiv preprint arXiv:1708.07747**, 2017. Citado na página 40.

XIE, Q.; DAI, Z.; HOVY, E.; LUONG, T.; LE, Q. Unsupervised data augmentation for consistency training. **Advances in Neural Information Processing Systems**, v. 33, p. 6256–6268, 2020. Citado na página 33.

XIE, Q.; LUONG, M.-T.; HOVY, E.; LE, Q. V. Self-training with noisy student improves imagenet classification. In: **Proceedings of the IEEE/CVF conference on computer vision and pattern recognition**. [S.l.: s.n.], 2020. p. 10687–10698. Citado na página 33.

YAN, X.; MISRA, I.; GUPTA, A.; GHADIYARAM, D.; MAHAJAN, D. Clusterfit: Improving generalization of visual representations. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2020. p. 6509–6518. Citado na página 29.

YANG, X.; SONG, Z.; KING, I.; XU, Z. A survey on deep semi-supervised learning. **arXiv preprint arXiv:2103.00550**, 2021. Citado na página 31.

ZBONTAR, J.; JING, L.; MISRA, I.; LECUN, Y.; DENY, S. Barlow twins: Self-supervised learning via redundancy reduction. **arXiv preprint arXiv:2103.03230**, 2021. Citado nas páginas 13 e 30.

____. Barlow twins: Self-supervised learning via redundancy reduction. In: **ICML**. [S.l.: s.n.], 2021. Citado na página 21.

ZHAI, X.; OLIVER, A.; KOLESNIKOV, A.; BEYER, L. S4l: Self-supervised semi-supervised learning. In: **Proceedings of the IEEE international conference on computer vision**. [S.l.: s.n.], 2019. p. 1476–1485. Citado na página 34.

ZHANG, H.; CISSE, M.; DAUPHIN, Y. N.; LOPEZ-PAZ, D. mixup: Beyond empirical risk minimization. **International Conference on Learning Representations**, 2018. Citado na página 33.

ZHANG, R.; ISOLA, P.; EFROS, A. A. Colorful image colorization. In: SPRINGER. **European conference on computer vision**. [S.l.], 2016. p. 649–666. Citado nas páginas 27 e 34.

____. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2017. p. 1058–1067. Citado na página 27.

ZHOU, Z.-H.; LI, M. Tri-training: Exploiting unlabeled data using three classifiers. **IEEE Transactions on knowledge and Data Engineering**, IEEE, v. 17, n. 11, p. 1529–1541, 2005. Citado na página 33.

ZHUANG, C.; ZHAI, A. L.; YAMINS, D. Local aggregation for unsupervised learning of visual embeddings. In: **Proceedings of the IEEE/CVF International Conference on Computer Vision**. [S.l.: s.n.], 2019. p. 6002–6012. Citado na página 29.


ICMC
USP
SÃO CARLOS