Teoria da Informação Aplicada ao Aprendizado Auto-Supervisionado com Múltiplas Vistas

1st Turi Rezende
Dept. de Ciência da Computação
Universidade Federal de Minas Gerais
Belo Horizonte, Brasil
turi@ufmg.br

2nd Wagner Meira Jr
Dept. de Ciência da Computação
Universidade Federal de Minas Gerais
Belo Horizonte, Brasil
meira@dcc.ufmg.br

3rd Mário Sérgio Alvim
Dept. de Ciência da Computação
Universidade Federal de Minas Gerais
Belo Horizonte, Brasil
msalvim@dcc.ufmg.br

***Abstract*—This work investigates the application of information theory to multiview self-supervised learning, proposing improvements to the *I-JEPA* model [1]. The modifications include incorporating *VICReg* [2] to enhance the diversity and decorrelation of representations and using Simplicial Embeddings [3] to improve separability and generalization. Experiments on *ImageNet-100* show that these enhancements increase classification accuracy, highlighting the relevance of information theory in improving self-supervised learning methods.**

***Index Terms*—self-supervised learning, information theory, joint-embedding predictive architecture.**

I. Introdução

Nos últimos anos, o aprendizado auto-supervisionado (*self-supervised learning, SSL*) emergiu como uma abordagem promissora para a extração de representações úteis a partir de dados não rotulados. Diferentemente dos métodos supervisionados tradicionais, que dependem de grandes quantidades de dados anotados, o SSL permite o aprendizado por meio da exploração de estruturas e padrões internos nos dados.

Dentro desse contexto, o aprendizado auto-supervisionado com múltiplas vistas (*multiview SSL*) é uma técnica de *SSL* que tem recebido atenção crescente. O paradigma de aprendizado *multiview* divide a variável de entrada em várias vistas, que compartilham informação semântica semelhante [4], [5]. Um conjunto de dados *multiview* usualmente consiste em dados capturados a partir de várias fontes, modalidades e formas. Esse mecanismo foi inicialmente aplicado a dados do mundo natural, combinando medições de imagem, texto, áudio e vídeo. Um exemplo é a identificação de uma pessoa ao analisar o fluxo de vídeo como uma vista e o fluxo de áudio como a outra. Embora diferentes vistas forneçam informações complementares sobre os mesmos dados, a integração direta delas nem sempre produz resultados satisfatórios devido a viéses [5]. Assim, o aprendizado de representação *multiview* envolve a identificação da estrutura subjacente dos dados a partir da integração das diferentes vistas em um espaço de características comum.

A teoria da informação fornece um arcabouço matemático fundamental para entender e aprimorar métodos de *SSL*. Em particular, princípios como a maximização da informação mútua (*InfoMax*) [6] e o gargalo de informação (*Information Bottleneck, IB*) [7] têm sido explorados para promover o aprendizado de representações que preservem aspectos essenciais dos dados, minimizando redundâncias e ruídos.

Neste trabalho, investigamos como a teoria da informação pode ser aplicada para o aprimoramento de métodos de aprendizado auto-supervisionado com múltiplas vistas. Em particular, propomos melhorias no modelo *I-JEPA* (*Image Joint-Embedding Predictive Architecture*) [1], um método de *SSL* que não utiliza *data augmentation* e se baseia na predição de regiões ocultas a partir de partes visíveis dos dados. As modificações propostas incluem a incorporação da função de custo *VICReg* (*Variance-Invariance-Covariance Regularization*) [2] para promover diversidade e descorrelação entre representações aprendidas e o uso de *Simplicial Embeddings (SEM)* [3], que normaliza as representações latentes, aumentando sua separabilidade e capacidade de generalização.

A fim de avaliar as melhorias propostas, realizamos experimentos no conjunto de dados *ImageNet-100*, comparando o desempenho do modelo original com as versões aprimoradas. A análise incluiu a avaliação do impacto das representações aprendidas na tarefa de classificação, além da medição direta da qualidade das representações por meio das métricas *Mutual Information Neural Estimation (MINE)* [8] e *LiDAR (Linear Discriminant Analysis Rank)* [9].

Os resultados obtidos demonstram que as modificações propostas proporcionam **ganhos expressivos na acurácia da tarefa de classificação subsequente**. Contudo, a análise da qualidade das representações por meio de *MINE* e *LiDAR* não apresentou correlação com os ganhos na tarefa de classificação, sugerindo que outras métricas devem ser estudas.

II. Referencial

Esta seção discute conceitos fundamentais relacionados ao *multiview SSL* e sua interpretação a partir da teoria da informação. Além disso, apresenta os principais trabalhos no campo de *multiview SSL*.

A. Aprendizado de Representações

O aprendizado de representações (*representation learning*) é um campo da aprendizagem de máquina que estuda como modelos computacionais podem aprender representações úteis dos dados automaticamente, sem a necessidade de engenharia manual de características (*feature engineering*). O objetivo

principal é encontrar transformações dos dados brutos que extraiam informações relevantes para uma determinada tarefa, como classificação ou regressão.

Dado um conjunto de dados $\mathcal{D} = \{x_i, y_i\}_{i=1}^N$, onde $x_i \in \mathcal{X}$ representa uma amostra de entrada e $y_i \in \mathcal{Y}$ representa um rótulo ou uma variável alvo associada, *representation learning* busca encontrar uma função $\Phi : \mathcal{X} \to \mathcal{Z}$ que mapeia os dados originais para um espaço latente $\mathcal{Z}$, extraindo informações relevantes.

Queremos encontrar Φ tal que:

$$z_i = \Phi(x_i), \quad z_i \in \mathcal{Z} \tag{1}$$

de modo que z_i seja uma representação útil de x_i para alguma tarefa subsequente, como a predição de uma variável-alvo associada $f : \mathcal{Z} \to \mathcal{Y}$.

O objetivo de *representation learning* pode ser formulado como a minimização de uma função de perda $\mathcal{L}$, que mede a discrepância entre a predição $f(z_i)$ e o valor verdadeiro y_i:

$$\min_{\Phi, f} \mathbb{E}_{(x,y)\sim p(x,y)} \left[\mathcal{L}(f(\Phi(x)), y)\right]. \tag{2}$$

B. Aprendizado Auto-Supervisionado

Diferentemente do aprendizado supervisionado, que depende de dados rotulados, o aprendizado auto-supervisionado (*self-supervised learning, SSL*) cria sinais de supervisão a partir dos próprios dados não rotulados. Esse paradigma tem sido amplamente utilizado para *representation learning*, pois não necessita de grandes conjuntos de dados anotados, que podem ser de difícil obtenção.

A principal motivação para o aprendizado auto-supervisionado é sua aplicabilidade ao pré-treinamento de modelos. Em muitas tarefas de aprendizado profundo, os conjuntos de dados anotados são escassos ou de obtenção cara, enquanto há uma abundância de dados não rotulados disponíveis. Modelos treinados inicialmente com *SSL* podem aprender representações robustas e generalizáveis a partir desses grandes volumes de dados, permitindo que um modelo ajustado posteriormente a um pequeno conjunto de exemplos rotulados alcance um desempenho competitivo.

Um exemplo clássico de *SSL* é o modelo de representação de texto *BERT* [10], que utiliza a modelagem de palavras mascaradas (*masked language modeling*) como tarefa auxiliar de pré-treinamento. Tal objetivo auto-supervisionado permite que o modelo aprenda representações semânticas ricas a partir de grandes quantidades de texto não rotulado, que podem ser posteriormente ajustadas para tarefas específicas, como classificação de texto, análise de sentimentos e tradução automática.

Além do pré-treinamento, o aprendizado auto-supervisionado também pode ser utilizado como um fim em si mesmo. Um exemplo disso são os modelos de geração de texto baseados em *transformers*, como o *GPT* [11]. Os modelos da família *GPT* são treinados para prever a próxima palavra em um contexto dado, utilizando o aprendizado auto-supervisionado como sua tarefa principal. Esse tipo de modelagem, chamada de *causal language modeling*, permite que o modelo aprenda dependências sequenciais nos dados e seja diretamente utilizado para geração de texto sem a necessidade de ajustes supervisionados.

1) Aprendizado Auto-Supervisionado com Múltiplas Vistas: o aprendizado auto-supervisionado com múltiplas vistas (*multiview self-supervised learning, multiview SSL*) explora diferentes visualizações de um mesmo dado para extrair representações robustas e informativas. Métodos *multiview SSL* são usualmente aplicados em cenários onde os dados possuem múltiplas modalidades, perspectivas ou transformações estruturadas.

Seja um conjunto de dados $\mathcal{D} = \{x_i\}_{i=1}^N$, onde cada amostra x_i pode ser observada sob diferentes vistas $v_1, v_2, \ldots, v_M$, denotadas como $x_i^{(v_1)}, x_i^{(v_2)}, \ldots, x_i^{(v_M)}$. O objetivo do *multiview SSL* é aprender uma representação latente z_i que seja invariante entre as diferentes vistas, preservando as informações úteis para tarefas posteriores.

Para isso, seja $f_\theta : \mathcal{X} \to \mathcal{Z}$ um codificador parametrizado que mapeia cada vista para uma representação latente. Definimos a perda, baseada na maximização da informação mútua entre duas vistas v_1 e v_2 da mesma amostra, como:

$$\mathcal{L}_{MI} = -I(z^{(v_1)}; z^{(v_2)}) \tag{3}$$

onde $I(\cdot;\cdot)$ representa a informação mútua entre as representações latentes. Entretanto, calcular a informação mútua de forma exata é computacionalmente inviável em muitos casos, especialmente quando as distribuições subjacentes das variáveis latentes são complexas ou desconhecidas. Por essa razão, técnicas baseadas em aproximação, como estimadores de informação mútua ou simplesmente a maximização da similaridade entre representações, são empregadas.

C. Princípio InfoMax

No contexto de *representation learning*, o Princípio InfoMax (*InfoMax Principle*) [6] propõe a maximização da informação mútua entre a entrada X e sua representação $f_\theta(X) = T$:

$$\max_{P(T|X)} I(X;T) \tag{4}$$

Este princípio sugere que a melhor representação é aquela que preserva o máximo de informação possível da entrada. Quando a dimensionalidade de T é menor que a de X, ocorre uma compressão da informação, onde o modelo f_θ aprende a extrair os aspectos mais significativos dos dados enquanto descarta redundâncias e ruídos.

D. Princípio InfoMax com Múltiplas Vistas

O Princípio InfoMax com Múltiplas Vistas (*multiview InfoMax*) busca maximizar a informação compartilhada entre diferentes vistas dos dados, garantindo que cada vista preserve o máximo de informação possível sobre a outra. A formulação geral é dada por:

$$\max_{P(Z_1|X_1);P(Z_2|X_2)} I(Z_1; Z_2) \tag{5}$$

onde $I(Z_1; Z_2)$ representa a informação mútua entre as representações das duas vistas.

1) Compressão implícita: a formulação

$$I(Z_1, Z_1) = H(Z_1) - H(Z_1|Z_2) \tag{6}$$

implica que a maximização de $I(Z_1, Z_2)$ envolve a minimização de $H(Z_1|Z_2)$. Dessa forma, podemos treinar modelos para extrair representações de Z_1 que sejam mais previsíveis a partir de Z_2. Isso naturalmente causa compressão: representações simplificadas são mais facilmente preditas. Contudo, $H(Z_1)$ atinge valores próximos de zero se não for regularizado, causando colapso das representações [12].

E. Gargalo de Informação

O princípio do Gargalo de Informação (*Information Bottleneck, IB*) [7] propõe uma abordagem para otimizar representações aprendidas, reduzindo informações irrelevantes e preservando apenas os aspectos essenciais para uma determinada tarefa. No contexto de aprendizado profundo, esse princípio equilibra a retenção da informação necessária para a predição, enquanto minimiza a redundância do sinal original.

A formulação do *IB* é dada por:

$$\mathcal{L} = \min_{P(T|X)} I(X;T) - \beta I(T;Y) \tag{7}$$

onde $I(X;T)$ representa a informação que a representação T retém sobre a entrada X, e $I(T;Y)$ mede a quantidade de informação relevante mantida sobre a variável alvo Y. O hiperparâmetro β controla o equilíbrio entre compressão e predição. Uma maior compressão reduz o risco de overfitting, mas pode eliminar informações relevantes para a tarefa.

No contexto de *SSL*, a formulação clássica do *IB* enfrenta desafios, pois não há um rótulo explícito Y para guiar a otimização. Para contornar essa limitação, abordagens baseadas no *IB* propõem maximizar a informação compartilhada entre diferentes vistas ou transformações dos dados de entrada, assumindo que essa informação comum é relevante para tarefas subsequentes.

F. Gargalo de Informação com Múltiplas Vistas

O Gargalo de Informação com Múltiplas Vistas (*multiview IB*) estende o *IB* para *multiview SSL*. Seja Y a variável-alvo de interesse, o paradigma *multiview* assume que existe ϵ_{info} tal que:

$$I(Y; X_2|X_1) \leq \epsilon_{info} \tag{8}$$
$$I(Y; X_1|X_2) \leq \epsilon_{info} \tag{9}$$

Dessa forma, quando e_{info} é pequeno, a informação compartilhada entre as vistas inclui detalhes relevantes para a tarefa de interesse. Essa suposição permite a separação da informação em relevante (compartilhada) e irrelevante (não compartilhada).

O objetivo do *multiview IB* é maximizar a informação compartilhada entre as vistas $I(X_2; Z_1)$, enquanto descarta a informação não compartilhada $I(Z_1; X_1|X_2)$ [13].

Sejam X_1 e X_2 duas vistas diferentes da mesma amostra e suas respectivas representações Z_1 e Z_2. A formulação do *multiview IB* é dada por:

$$\mathcal{L}_1 = I(Z_1; X_1|X_2) - \beta_1 I(X_2; Z_1) \tag{10}$$
$$\mathcal{L}_2 = I(Z_2; X_2|X_1) - \beta_2 I(X_1; Z_2) \tag{11}$$

onde β_1 e β_2 são multiplicadores de Lagrange que controlam o grau de compressão das representações. Combinando ambas as perdas, temos a seguinte formulação geral:

$$\mathcal{L} = -I(Z_1; Z_2) + \beta D_{\text{KL}}[P(Z_1|X_1)||P(Z_2|X_2)] \tag{12}$$

onde D_{KL} representa a divergência de Kullback-Leibler entre as distribuições das representações.

Embora representações melhores possam ser obtidas por meio do *multiview IB*, a maioria dos métodos de *multiview SSL* usa apenas o princípio InfoMax e maximiza $I(Z_1, Z_2)$. Isso ocorre porque o InfoMax é mais fácil de otimizar em espaços de alta dimensão.

G. Arquiteturas de Representação Conjunta

Arquiteturas de Representação Conjunta (*Joint-Embedding Architectures, JEAs*) são métodos de *multiview SSL* que exploram o mapeamento de diferentes vistas do mesmo dado para um espaço latente compartilhado.

1) Métodos Contrastivos: o aprendizado contrastivo explora a ideia de maximizar similaridades entre pares positivos (relacionados) e minimizar similaridades entre pares negativos (não relacionados).

- ***Contrastive Predictive Coding* [14]**: Abordagem de aprendizado auto-supervisionado que busca capturar dependências temporais ou estruturais nos dados, maximizando a informação mútua entre representações latentes de diferentes partes dos dados. A entrada é dividida em blocos $x_1, ..., x_k$ e um codificador é utilizado para mapear os blocos em representações latentes $z_1, ..., z_k$. Um modelo de contexto é usado para extrair uma representação de alto nível c_t para os embeddings $z_1, ..., z_t$. O objetivo é prever a representação de partes futuras z_{t+k} a partir de c_t e distinguir a representação objetivo z_{t+k} das demais $z_{j \neq t+k}$. O treinamento do CPC é realizado com a função de perda InfoNCE (*Information Noise-Contrastive Estimation*)

$$\mathcal{L} = -\mathbb{E}\left[\frac{\log f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right], \tag{13}$$

  que maximiza a similaridade entre pares positivos (representações de partes relacionadas dos dados, como segmentos temporais consecutivos) e minimiza a similaridade com pares negativos (representações de partes não relacionadas). A InfoNCE aproxima um limite inferior para a informação mútua.
- ***SimCLR [15]***: Propôs o uso de *data augmentation*, como rotações e cortes, para gerar diferentes vistas da mesma

imagem (pares positivos). Pares negativos são formados por diferentes imagens no mesmo *batch* de dados. O modelo utiliza uma rede base para extrair características, seguida por uma projeção não linear em um espaço latente. A função de perda contrastiva

$$\mathcal{L} = \frac{1}{2N} \sum_{k}^{N} [l(2k-1, 2k) + l(2k, 2k-1)], \qquad (14)$$

onde

$$l(i,j) = -\log \frac{\exp(sim(z_i, z_j)/\tau)}{\sum_{z_k \neq j} \exp(sim(z_i, z_k)/\tau)}, \qquad (15)$$

é então otimizada para maximizar a similaridade entre pares positivos enquanto minimiza a similaridade entre pares negativos no espaço projetado. Essa estratégia permite ao modelo aprender representações invariantes às transformações aplicadas.

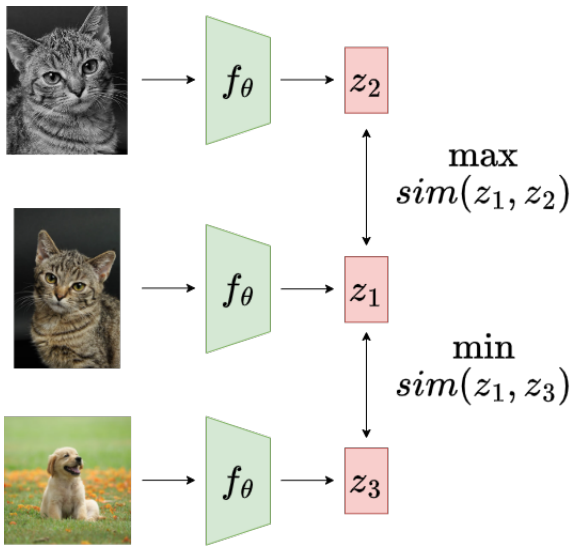

Fig. 1. *SimCLR*

2) Métodos Não Contrastivos: modelos não contrastivos eliminam a necessidade de pares negativos, explorando a maximização de similaridade entre representações de vistas do mesmo dado. Contudo, ao remover a característica contrastiva da função de perda, é necessário utilizar outras técnicas para evitar o problema conhecido como *representation collapse*. Esse fenômeno ocorre quando o modelo produz soluções triviais, que são representações idênticas ou invariantes para todos os dados de entrada, independentemente de suas características individuais, resultando em perda de informação relevante e incapacidade de distinguir entre diferentes exemplos.

- ***BYOL* [16]**: Utiliza uma rede principal (*online network*) e uma rede alvo (*target network*), onde a segunda é atualizada por média móvel exponencial (*Exponential Moving Average, EMA*) da primeira. O modelo emprega *data augmentation* para gerar duas vistas de cada amostra. A rede principal processa uma das vistas, enquanto a rede alvo processa a outra. O objetivo é alinhar as representações aprendidas para essas diferentes vistas, incentivando a invariância às transformações aplicadas. Para evitar *representation collapse*, o BYOL introduz um preditor no fim da *online network*, garantindo assimetria no treinamento.
- ***SiamSiam* [17]**: Simplifica o BYOL ao não utilizar *target network* ou atualizações por *EMA*. A assimetria é garantida apenas pelo preditor, que é aplicado somente em uma das vistas.
- ***DINO* [18]**: O *Self-distillation with no Labels (DINO)* utiliza duas redes, uma *teacher* e uma *student*, treinadas para alinhar representações de diferentes vistas dos mesmos dados de entrada. A rede *teacher* é atualizada por *EMA* dos pesos da *student* durante o treinamento. Ambas as redes recebem entradas diferentes geradas por técnicas de *data augmentation*. Diferentemente do BYOL, as saídas das redes são probabilísticas, e o modelo usa uma função de entropia cruzada

$$\mathcal{L} = \mathbb{E}_{p_1}[-\log p_2] \qquad (16)$$

para alinhá-las. Para evitar *representation collapse*, a rede *teacher* aplica uma normalização com mecanismo de *centering* e ajuste da temperatura no espaço de saída. Além disso, a atualização da rede *teacher* por *EMA* garante assimetria e dificulta a geração de representações triviais.

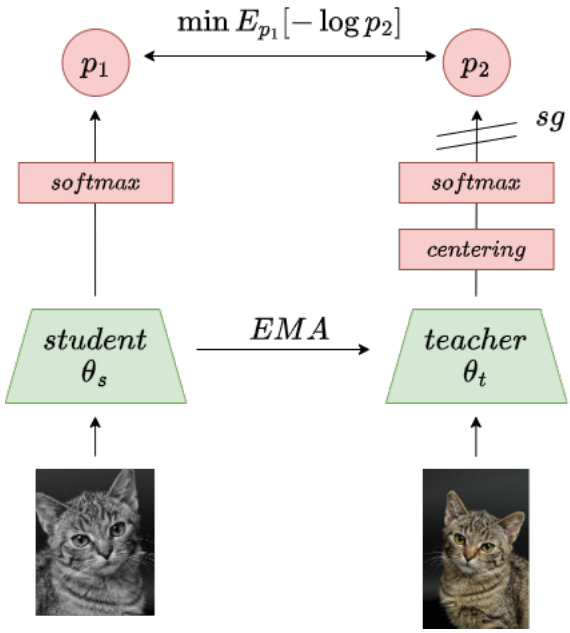

Fig. 2. *DINO*

- ***VICReg* [2]**: O *Variance-Invariance-Covariance Regularization* (*VICReg*) propõe um método não contrastivo baseado na maximização da similaridade entre representações de diferentes vistas sem a necessidade de uma *target network* ou *EMA*. Para evitar *representation collapse* e aumentar riqueza e diversidade das

representações, o VICReg introduz três termos na função de perda, referentes respectivamente à variância, similaridade e covariância. Dados $X = [x_1, ..., x_n]$ e $X' = [x'_1, ..., x'_n]$, *batches* de dados de duas vistas diferentes para os mesmos exemplos, uma rede neural f_θ é usada para extrair as representações latentes $Z = [z_1, ..., z_n]$ e $Z' = [z'_1, ..., z'_n]$ respectivamente. A função de perda para o treinamento possui os seguintes termos:

- $v(Z)$ (termo de variância) impõe uma dispersão mínima nas representações ao garantir que a variância ao longo de cada dimensão do espaço latente seja maior que um limiar. Isso evita que as representações colapsem para um único ponto.

$$v(Z) = \frac{1}{d}\sum_{j=1}^{d} \max\left(0, \gamma - \sqrt{Var(z^j) + \epsilon}\right) \quad (17)$$

  onde z^j denota o vetor composto pelo valor na dimensão j em cada um dos vetores no *batch*.
- $s(Z, Z')$ (termo de invariância) promove similaridade entre as representações das duas vistas do mesmo dado de entrada. Isso é feito minimizando a distância Euclidiana entre os embeddings correspondentes.

$$s(Z, Z') = \frac{1}{n}\sum_{i} ||z_i - z'_i||_2^2 \quad (18)$$

- $c(Z)$ (termo de covariância) regula a redundância entre dimensões diferentes dos embeddings, incentivando descorrelação ao minimizar a soma dos quadrados dos elementos fora da diagonal da matriz de covariância das representações.

$$C(Z) = \frac{1}{n-1}\sum_{i=1}^{n}(z_i - \overline{z})(z_i - \overline{z})^T \quad (19)$$

$$c(Z) = \frac{1}{d}\sum_{i \neq j}[C(Z)]_{i,j}^2 \quad (20)$$

A função de perda então é definida por:

$$\begin{aligned}\mathcal{L}(Z, Z') = \lambda_s s(Z, Z') \\ + \lambda_v[v(Z) + v(Z')] \\ + \lambda_c[c(Z) + c(Z')]\end{aligned} \quad (21)$$

Dessa forma, o VICReg promove a aprendizagem de representações ricas e informativas sem a necessidade de uma abordagem contrastiva.

H. Arquiteturas Generativas

Arquiteturas Generativas (*Generative Architectures*) aprendem representações reconstruindo regiões ausentes dos dados de entrada, incentivando o modelo a capturar informações relevantes a partir de contextos parciais.

- ***Masked Autoencoders* [19]**: Durante o treinamento, uma fração aleatória dos *patches* da entrada é mascarada e ignorada pelo codificador. O modelo é treinado para reconstruir os *patches* mascarados a partir dos *patches* visíveis, utilizando um decodificador que processa as representações latentes do codificador. Esse processo força o modelo a aprender padrões globais e contextuais a partir das partes observadas dos dados. Embora não seja um método diretamente *multiview*, as partes ausentes podem ser interpretadas como uma vista dos dados, que deve ser reconstruída a partir do contexto.

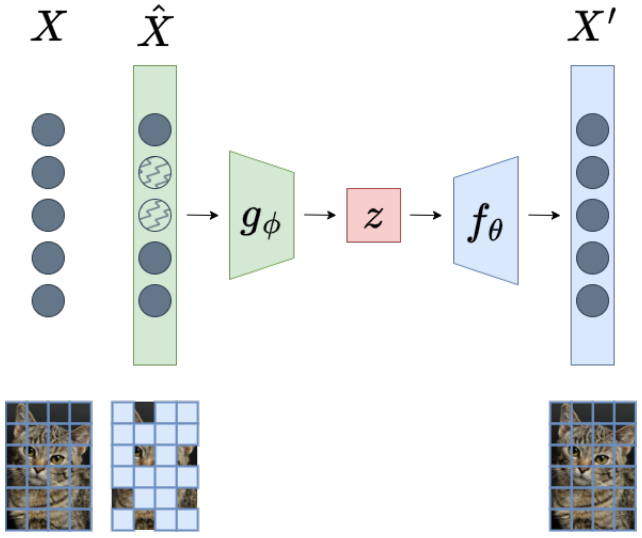

Fig. 3. *Masked autoencoders*

I. Arquiteturas de Predição de Representação Conjunta

Arquiteturas de Predição de Representação Conjunta (*Joint-Embedding Predictive Architectures, JEPAs*) são uma classe de métodos de *multiview SSL* que vão além do simples mapeamento de diferentes vistas do mesmo dado para um espaço latente compartilhado. Em vez disso, *JEPAs* exploram a estrutura interna dos dados ao modelar explicitamente relações entre regiões conhecidas e desconhecidas dentro de um exemplo. Isso é feito por meio da predição de representações latentes de partes ocultas (*targets*) a partir das representações de partes visíveis (*context*). O modelo é composto por duas redes principais:

- **Codificador (*Encoder*)**: Responsável por transformar a entrada em uma representação latente.
- **Preditor (*Predictor*)**: Toma como entrada a representação do contexto visível extraída pelo codificador e prevê a representação latente da parte oculta do dado.

Além disso, *JEPAs* não dependem de *data augmentation*, de forma a se tornarem uma abordagem atrativa para domínios onde a geração de vistas artificiais dos dados é limitada ou inviável.

Na saúde, por exemplo, sinais fisiológicos como eletrocardiogramas (ECG) e eletroencefalogramas (EEG) possuem estruturas temporais sensíveis, e a aplicação de transformações como distorções temporais ou adição de ruído pode afetar a interpretação clínica. Da mesma forma, em imagens médicas, como ressonâncias magnéticas e tomografias computadorizadas, a geração artificial de variações pode introduzir

artefatos que prejudicam a análise diagnóstica. No setor financeiro, onde séries temporais representam transações e padrões de mercado, a manipulação artificial dos dados pode levar a interpretações equivocadas e comprometer a modelagem de riscos.

Dessa forma, *JEPAs* oferecem uma solução robusta para aprendizado auto-supervisionado, especialmente em cenários onde preservar a integridade dos dados originais é importante.

- ***I-JEPA*** **[1]**: O *I-JEPA* (*Image Joint-Embedding Predictive Architecture*) é um método de *multiview SSL* baseado no paradigma *JEPA*. O *I-JEPA* aprende a prever representações latentes de regiões ocultas de uma imagem com base em representações das regiões visíveis e em informações posicionais das regiões ocultas, sem necessidade de reconstrução direta da imagem.
  O modelo é composto por um codificador (*encoder network*), um preditor (*predictor*) e uma rede-alvo (*target network*). O codificador extrai representações latentes das partes visíveis da entrada (*context*), enquanto a rede-alvo extrai representações das partes ocultas (*targets*). O preditor é treinado para estimar as representações ocultas a partir das representações visíveis e de informações posicionais das regiões ocultas. A rede-alvo é atualizada por *EMA* dos pesos do codificador durante o treinamento. Isso é feito para garantir alvos de predição estáveis para otimização, além de evitar *representation collapse* ao promover assimetria entre o codificador e a rede-alvo. As implementações do codificador, rede-alvo e preditor são baseadas no *Vision Transformer* (*ViT*) [20].
  Matematicamente, o *I-JEPA* é formulado da seguinte forma: seja uma imagem de entrada $x \in \mathbb{R}^{H \times W \times C}$, onde H e W são as dimensões espaciais e C é o número de canais. O objetivo é aprender uma representação z para regiões ocultas com base nas regiões visíveis.

  1) **Codificação das regiões visíveis**: O codificador f_θ transforma as regiões visíveis x_c (*context*) em uma representação latente:

  $$z_c = f_\theta(x_c), \tag{22}$$

  onde $z_c \in \mathbb{R}^d$ é a representação do contexto.
  2) **Codificação das regiões ocultas (rede-alvo)**: A rede-alvo f_ξ gera representações para as regiões ocultas x_t (*targets*):

  $$z_t = f_\xi(x_t), \tag{23}$$

  onde $z_t \in \mathbb{R}^d$ é a representação latente das regiões ocultas.
  3) **Predição das representações ocultas**: O preditor h_ϕ aprende a mapear z_c para z_t incorporando informações posicionais p_t, que codificam a posição das regiões ocultas na imagem original:

  $$\hat{z}_t = h_\phi(z_c, p_t), \tag{24}$$

  onde $\hat{z}_t$ é a predição da representação das regiões ocultas baseada no contexto e na informação posicional.
  4) **Função de perda**: O treinamento do modelo é baseado na minimização da distância L_2 entre $\hat{z}_t$ e z_t.

  $$\mathcal{L}_{IJEPA} = \|\hat{z}_t - z_t\|^2, \tag{25}$$

  5) **Atualização por *EMA***: Os pesos da rede-alvo são atualizados por média exponencial móvel dos pesos do codificador.

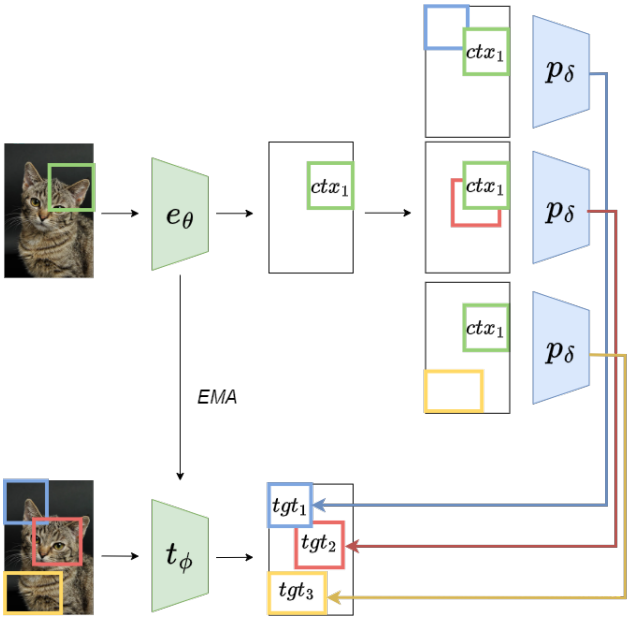

Fig. 4. *I-JEPA*

Esse processo permite que o *I-JEPA* aprenda representações sem reconstruir a imagem, focando diretamente nas características semânticas latentes.

J. Simplicial Embeddings

Simplicial Embeddings (*SEM*) [3] é um método que modifica a representação aprendida durante o pré-treinamento em aprendizado auto-supervisionado. Em vez de simplesmente mapear as representações para um espaço latente convencional, o *SEM* projeta essas representações em L simplexes de V dimensões, utilizando uma operação *softmax*. Esse procedimento impõe uma restrição estrutural à representação, promovendo um viés indutivo para a esparsidade e reduzindo a necessidade de normalizações adicionais.

Uma característica fundamental do *SEM* é a concatenação das representações projetadas em cada simplexo para formar a representação final do modelo. Formalmente, seja z_i uma das L projeções no espaço latente, a normalização SEM é definida como:

$$\bar{z}_i := \sigma_\tau(z_i), \quad \sigma_\tau(z_i)_j = \frac{e^{z_{ij}/\tau}}{\sum_{k=1}^{V} e^{z_{ik}/\tau}} \tag{26}$$

onde a operação *softmax* é aplicada para cada projeção. A concatenação dessas projeções normalizadas resulta na representação final:

$$\hat{z} := \text{Concat}(\bar{z}_1, \bar{z}_2, \ldots, \bar{z}_L) \tag{27}$$

Essa concatenação preserva a estrutura de simplicidade das projeções enquanto permite que o modelo explore uma representação mais expressiva e de alta dimensionalidade.

O *SEM* é particularmente interessante porque melhora a generalização das representações aprendidas, tanto em tarefas de classificação supervisionada quanto na robustez a distribuições fora do domínio. Durante o pré-treinamento, a temperatura do *softmax* controla o nível de esparsidade das representações, permitindo controle da expressividade do modelo. Ao aplicar *SEM* a métodos de *SSL* convencionais, como *BYOL*, *DINO* e *VICReg*, observou-se um aumento no desempenho de até 4% em conjuntos de dados como CIFAR-100 e ImageNet.

1) Redução do Erro de Generalização com SEM: a melhora na generalização do *SEM* pode ser explicada por três fatores principais:

1) **Regularização Implícita via Competição Entre Componentes:** A normalização *softmax* impõe uma restrição competitiva nos valores das representações latentes, onde um aumento no valor de uma dimensão precisa ser compensado por uma diminuição em outra. Isso reduz a redundância na representação e força o modelo a selecionar características mais discriminativas.
2) **Viés Indutivo para Esparsidade:** Como as projeções *SEM* tendem a ser esparsas, o modelo aprende a representar os dados com um subconjunto de características ativadas. Isso reduz o risco de *overfitting* ao evitar o aprendizado de características espúrias presentes apenas no conjunto de treinamento, contribuindo para um menor erro de generalização.
3) **Melhoria na Separabilidade das Classes:** Ao projetar representações em simplexes, o SEM cria um espaço latente onde as diferentes classes são mais separáveis, facilitando a aprendizagem de um classificador. A teoria do *SEM* demonstra que, conforme L e V aumentam, a generalização melhora, pois há um maior espaço disponível para expressar variações relevantes dos dados.

2) Impacto do SEM na Entropia do Espaço Latente: a aplicação do *SEM* também influencia diretamente a entropia das representações no espaço latente. Como cada projeção passa por uma normalização *softmax*, a distribuição de probabilidades resultante impõe uma estrutura competitiva entre as dimensões do vetor latente.

A entropia de cada simplexo pode ser definida como:

$$H(\bar{z}_i) = -\sum_{j=1}^{V} \sigma_\tau(z_i)_j \log \sigma_\tau(z_i)_j \tag{28}$$

onde valores menores de temperatura τ resultam em distribuições mais esparsas e menor entropia, concentrando a informação em poucas dimensões relevantes. Já valores maiores de τ promovem distribuições mais uniformes, aumentando a entropia e permitindo maior expressividade do modelo.

O controle da entropia através da temperatura do *softmax* possibilita um equilíbrio entre esparsidade e expressividade, influenciando diretamente a qualidade das representações aprendidas e sua capacidade de generalização.

Além disso, estudos empíricos mostraram que as representações obtidas por *SEM* apresentam maior alinhamento semântico com as categorias dos dados, o que pode ser útil para melhorar a interpretabilidade das representações auto-supervisionadas.

K. Avaliação do Aprendizado Auto-Supervisionado

A avaliação de modelos de aprendizado auto-supervisionado tem o objetivo de medir a qualidade das representações aprendidas. Neste trabalho, exploramos três abordagens distintas para avaliação: (i) estimativa da informação mútua entre representações de vistas correspondentes, utilizando *Mutual Information Neural Estimation* [8], (ii) avaliação da qualidade da representação com a métrica *LiDAR* [9], e (iii) avaliação indireta através da transferência das representações para a tarefa subsequente de interesse, com diferentes proporções de dataset, que quantifica o impacto final do aprendizado auto-supervisionado.

1) Mutual Information Neural Estimation: A informação mútua (*MI*) entre duas variáveis aleatórias Z_1 e Z_2 é definida como:

$$I(Z_1; Z_2) = \int_{Z_1 \times Z_2} \log \frac{dP_{Z_1 Z_2}}{dP_{Z_1} \otimes P_{Z_2}} dP_{Z_1 Z_2}, \tag{29}$$

onde $P_{Z_1 Z_2}$ é a distribuição conjunta e P_{Z_1}, P_{Z_2} são as distribuições marginais. O cálculo exato de $I(Z_1; Z_2)$ é desafiador para distribuições contínuas, mas pode ser estimado por redes neurais através da representação dual de Donsker-Varadhan para a divergência de Kullback-Leibler:

$$D_{\text{KL}}(P_{Z_1 Z_2} || P_{Z_1} \otimes P_{Z_2}) = \sup_T \mathbb{E}_{P_{Z_1 Z_2}}[T] - \log \mathbb{E}_{P_{Z_1} \otimes P_{Z_2}}[e^T]. \tag{30}$$

O *MINE* otimiza esse valor com uma rede neural parametrizada T_θ, estimando $I(Z_1; Z_2)$ como:

$$\hat{I}_\theta(Z_1; Z_2) = \sup_\theta \mathbb{E}_{P_{Z_1 Z_2}}[T_\theta] - \log \mathbb{E}_{P_{Z_1} \otimes P_{Z_2}}[e^{T_\theta}]. \tag{31}$$

A principal vantagem dessa abordagem é sua capacidade de capturar dependências não-lineares e de alta dimensão entre as representações aprendidas.

2) LiDAR (Avaliação por Linear Discriminant Analysis Rank): LiDAR é um método projetado para avaliar representações em arquiteturas *Joint Embedding* (*JE*) de *SSL*. Ele mede a qualidade das representações pela análise espectral da matriz de Análise Discriminante Linear (*LDA*), buscando distinguir características informativas das irrelevantes. O método se baseia nas matrizes de covariância

entre representações de exemplos diferentes (Σ_b) e entre representações de vistas diferentes do mesmo exemplo (Σ_w):

$$\Sigma_b = \mathbb{E}_x[(\mu_x - \mu)(\mu_x - \mu)^T], \tag{32}$$

$$\Sigma_w = \mathbb{E}_x \mathbb{E}_{x' \sim D_x}[(e(x') - \mu_x)(e(x') - \mu_x)^T] + \delta I. \tag{33}$$

A métrica *LiDAR* é então definida como:

$$\Sigma_{\text{LiDAR}} = \Sigma_w^{-\frac{1}{2}} \Sigma_b \Sigma_w^{-\frac{1}{2}}, \tag{34}$$

onde os autovalores λ_i de Σ_{LiDAR} representam a variabilidade ao longo das direções discriminativas. O *score* final é obtido por:

$$\text{LiDAR}(e) = \exp\left(-\sum_i p_i \log p_i\right), \quad p_i = \frac{\lambda_i}{\sum_j \lambda_j} + \varepsilon. \tag{35}$$

3) Avaliação por Transferência de Representação: A avaliação tradicional de representações *SSL* ocorre por meio de sua aplicação em uma tarefa supervisionada subsequente. Normalmente, modelos pré-treinados são ajustados em diferentes quantidades de dados rotulados, e seu desempenho é mensurado. Esse processo fornece uma estimativa confiável da qualidade do pré-treinamento auto-supervisionado, pois avalia a métrica final: o desempenho na tarefa subsequente de interesse.

III. Contribuição

Nesta seção, a metodologia seguida no trabalho será detalhada. Além disso, será discutida a contribuição científica oferecida.

A. Implementação do modelo I-JEPA

Nesta etapa, foi implementado o método *I-JEPA* [1]. A implementação utilizada foi baseada na versão original, disponibilizada no repositório oficial em https://github.com/facebookresearch/ijepa. O treinamento do modelo foi realizado utilizando o *ImageNet Large Scale Visual Recognition Challenge Dataset* (*ILSVRC Dataset, ImageNet-1k*), que contém imagens distribuídas em 1.000 classes. As imagens estão divididas em 3 conjuntos:

- **Treinamento**: Contém aproximadamente 1.280.000 imagens, com aproximadamente 1.280 amostras por classe.
- **Validação**: Composto por 50.000 imagens, com 50 amostras por classe.
- **Teste**: Composto por 100.000 imagens, com 100 amostras por classe. O conjunto de teste não está publicamente disponível.

Para reduzir o custo computacional, foi selecionado um subconjunto de 100 classes por meio de amostragem aleatória, denominado *ImageNet-100*. Além disso, considerando que o conjunto de teste oficial não é publicamente acessível, o conjunto de validação foi particionado em dois subconjuntos distintos: um para validação e outro para teste. Dessa forma, a distribuição final das amostras resultou em 128.000 imagens para **treinamento**, 2.500 para **validação** e 2.500 para **teste**.

O pré-processamento aplicado às imagens consistiu na normalização e redimensionamento para 224×224 pixels.

B. Proposição de melhorias no I-JEPA baseadas na teoria da informação

A principal contribuição apresentada neste trabalho é a proposição de melhorias no modelo *I-JEPA*, baseadas na teoria da informação.

1) Modificação da Função de Perda: Como discutido em II, métodos de *multiview self-supervised learning* (SSL) geralmente buscam maximizar a informação mútua entre as representações de diferentes vistas dos dados, conforme expresso por:

$$\max_{P(Z_1|X_1), P(Z_2|X_2)} I(Z_1; Z_2), \tag{36}$$

onde $I(Z_1; Z_2)$ denota a informação mútua entre as representações Z_1 e Z_2 extraídas a partir das vistas X_1 e X_2.

No caso do *I-JEPA*, essa otimização é realizada de forma indireta por meio da minimização da distância L_2:

$$\hat{z}_t = h_\phi(z_c, p_t), \tag{37}$$

$$\mathcal{L}_{\text{I-JEPA}} = \|\hat{z}_t - z_t\|^2, \tag{38}$$

onde h_ϕ representa o preditor, z_c a representação do contexto, z_t a representação-alvo e p_t a informação posicional associada a z_t.

Neste trabalho, propomos uma modificação da função de perda do *I-JEPA* para que ela forneça uma aproximação melhor da informação mútua $I(z_c, z_t)$. Para isso, integramos o *I-JEPA* ao *VICReg*, de modo a incorporar critérios baseado em variância, similaridade e descorrelação na otimização do modelo.

A função de perda do *VICReg* é expressa como:

$$\begin{aligned}\mathcal{L}(Z, Z') = \lambda_s s(Z, Z') \\ + \lambda_v[v(Z) + v(Z')] \\ + \lambda_c[c(Z) + c(Z')],\end{aligned} \tag{39}$$

sendo $s(\cdot)$ o termo de similaridade, $v(\cdot)$ o termo de variância e $c(\cdot)$ o termo de covariância.

Dadas duas vistas X e X' e suas respectivas representações Z e Z', o artigo [21] conecta o objetivo do VICReg com a maximização de informação mútua $I(Z, Z')$ da seguinte forma:

$$I(Z; Z') \leq \min(I(X, Z'), I(X', Z)), \tag{40}$$

onde a maximização de $I(X, Z')$ e $I(X', Z)$ resulta na maximização de $I(Z; Z')$. Assim sendo, $I(Z; X')$ pode ser definido como:

$$I(Z; X') = H(Z) - H(Z|X') \tag{41}$$

Assumindo que $(Z|X' = x_n) = \mathcal{N}(\mu(x_n), I + \Sigma(x_n))$ e manipulando a expressão, temos:

$$I(Z;X') \geq H(Z) + \frac{d}{2}\log 2\pi - \frac{1}{2}\mathbb{E}_{x,x'}\Big[(\mu(x)-\mu(x'))^2 + \log(|\Sigma(x)|\cdot|\Sigma(x')|)\Big] \quad (42)$$

A função de perda corresponpente à maximização de $I(Z, X')$ pode ser aproximada por meio da distribuição empírica dos dados:

$$\mathcal{L} \approx \frac{1}{N}\sum_{i=1}^{N}\Big(- H(Z) + \log\left(|\Sigma(x_i)|\cdot|\Sigma(x_i')|\right) + \frac{1}{2}(\mu(x_i)-\mu(x_i'))^2\Big) \quad (43)$$

onde o termo de regularização $\log\left(|\Sigma(x_i)|\cdot|\Sigma(x_i')|\right)$ corresponde ao controle da covariância das representações, e o termo de invariância $\frac{1}{2}(\mu(x_i) - \mu(x_i'))^2$ regula a distância entre representações para diferentes vistas. Usando os dois primeiros momentos para aproximar a entropia, obtemos a seguinte aproximação:

$$\mathcal{L} \approx \sum_{n=1}^{N} -\log\left(\frac{|\Sigma_Z(x_1,...,x_N)|}{|\Sigma(x_i)|\cdot|\Sigma(x_i')|}\right) + \frac{1}{2}(\mu(x)-\mu(x'))^2 \quad (44)$$

Assumindo que os autovalores de $\Sigma(x_i)$ e $\Sigma(x_i')$ e as diferenças entre $\mu(x_i)$ e $\mu(x_i')$ são limitados, a solução para o problema de otimização

$$\min_{\Sigma_Z}\left[\sum_{n=1}^{N} -\log\left(\frac{|\Sigma_Z(x_1,...,x_N)|}{|\Sigma(x_i)|\cdot|\Sigma(x_i')|}\right)\right] \quad (45)$$

envolve fazer com que Σ_Z seja uma matriz diagonal. Isso pode ser atingido minimizando os elementos fora da diagonal (corresponde ao termo de covariância do *VICReg*) e maximizando a soma dos elementos na diagonal (corresponde ao termo de variância do *VICReg*). O termo de similaridade do *VICReg* tem correspondência direta com $\frac{1}{2}(\mu(x_i) - \mu(x_i'))^2$.

A adaptação da função de perda do *I-JEPA* para incluir os termos do *VICReg* que apresentou os melhores resultados empíricos foi a seguinte:

$$\mathcal{L}(z_c, z_t, p_t) = \lambda_s\|h_\phi(z_c, p_t) - z_t\|^2 + \lambda_v v(\rho(z_c)) + \lambda_c c(\rho(z_c)), \quad (46)$$

onde ρ é uma transformação não-linear aprendida no treinamento.

2) Normalização Utilizando Simplicial Embeddings: A segunda melhoria proposta para o método *I-JEPA* neste trabalho é o uso de *SEM* para normalizar as representações:

$$\bar{z}_i := \sigma_\tau(z_i), \quad \sigma_\tau(z_i)_j = \frac{e^{z_{ij}/\tau}}{\sum_{k=1}^{V} e^{z_{ik}/\tau}} \quad (47)$$

$$\hat{z} := \text{Concat}(\bar{z}_1, \bar{z}_2, \ldots, \bar{z}_L) \quad (48)$$

A normalização foi aplicada aos outputs do codificador e da rede-alvo, antes do uso do preditor.

C. Avaliação das melhorias na tarefa de classificação subsequente

O método original e as modificações propostas foram avaliados na tarefa de classificação subsequente. A avaliação foi realizada seguindo os procedimentos descritos abaixo:

1) **Avaliação sem pré-treinamento:** O *Vision Transformer* foi avaliado diretamente na tarefa de classificação do *ImageNet-100*, utilizando diferentes proporções dos dados disponíveis: 10%, 25%, 50% e 100%. A implementação do *ViT* usada foi a *ViT-T* (*ViT Tiny*), o menor modelo disponível no repositório oficial do *I-JEPA*.
2) **Avaliação com pré-treinamento:** O *ViT* foi avaliado na tarefa de classificação utilizando as mesmas proporções de dados, mas agora com um pré-treinamento *I-JEPA* realizado com 100% dos dados disponíveis no *ImageNet-100*.
3) **Comparação entre modelos:** Para comparar os modelos porpostos e o orignal, a acurácia da classificação no conjunto de teste foi utilizada.

A tabela I contém os resultados comparativos. Os métodos

- *I-JEPA + SEM*
- *I-JEPA + VICReg*
- *I-JEPA + SEM + VICReg*

foram propostos neste trabalho.

Pré-treinamento	10%	25%	50%	100%
Nenhum	0.1079	0.2264	0.3460	0.4971
I-JEPA	0.3816	0.4953	0.5523	0.5982
I-JEPA + SEM	<u>0.3916</u>	0.4680	**0.5963**	<u>0.6604</u>
I-JEPA + VICReg	0.3582	<u>0.4898</u>	0.5691	<u>0.6604</u>
I-JEPA + SEM + VICReg	**0.4105**	**0.5433**	<u>0.5820</u>	**0.6866**

TABLE I
ACURÁCIA DE CLASSIFICAÇÃO PARA DIFERENTES MODELOS EM VÁRIOS PERCENTUAIS DE DADOS PARA FINE-TUNING.

É importante ressaltar que o objetivo deste trabalho não é fazer ajuste de hiperparâmetros para obter uma acurácia alta, mas sim comparar diferentes métodos de pré-treinamento sob as mesmas condições. Observando os resultados apresentados, é possível perceber que as modificações propostas proporcionaram um aumento na acurácia do modelo em todos os cenários, especialmente em comparação com a implementação

orignal. Além disso, o melhor modelo obtido consiste no *I-JEPA* implementado com *SEM* e *VICReg*, demonstrando que a combinação dos métodos foi efetiva.

D. Análise da qualidade das representações extraídas

A qualidade das representações extraídas pelos modelos foi analisada utilizando os métodos baseados na teoria da informação *MINE* [8] e *LiDAR* [9]. Esta análise foi conduzida com os seguintes objetivos:

1) **Estimativa da informação mútua entre z_c e z_t:** Utilizar o *MINE* para estimar a informação mútua entre os *embeddings* z_{obs} e z_{target}, fornecendo uma medida da dependência entre as representações de contexto z_c e as representações-alvo z_t aprendidas.
2) **Cálculo do LiDAR ao longo do treinamento:** Computar o *LiDAR* em diferentes pontos do processo de treinamento auto-supervisionado para avaliar a evolução das representações aprendidas ao longo do tempo.
3) **Correlação com o desempenho na tarefa de classificação:** Verificar se exitem correlações entre os resultados das métricas de qualidade das representações e o desempenho dos modelos pré-treinados na tarefa de classificação subsequente.

Abaixo, seguem os resultados obtidos para os métodos *I-JEPA + VICReg* e *I-JEPA + VICReg + SEM*.

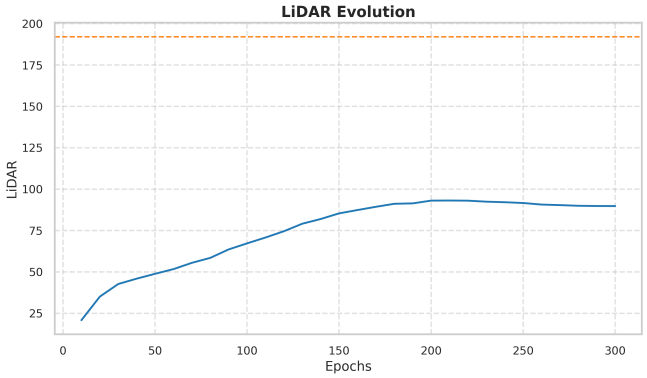

Fig. 5. *LiDAR* computado para vários *checkpoints* de treinamento para o método *I-JEPA + VICReg*

Analisando os gráficos, é possível perceber que o *I-JEPA + VICReg* apresentou melhores resultados para *MINE* e *LiDAR* do que o método *I-JEPA + VICReg + SEM*. Contudo, essa melhora não se propagou para os resultados obtidos na classificação, dado que *I-JEPA + VICReg + SEM* foi o melhor modelo. Uma possível justificativa para isso é que o valor ótimo de temperatura τ encontrado para *SEM* durante o treinamento foi de 0.5. Como visto em II, valores menores de temperatura τ resultam em distribuições mais esparsas e menor entropia, concentrando a informação em poucas dimensões relevantes. Consequentemente, a menor entropia induzida pelo *SEM* limita superiormente a informação mútua entre embeddings $I(Z_1, Z_2)$.

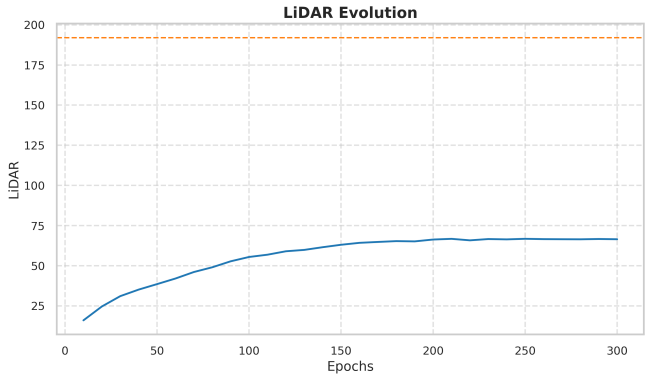

Fig. 6. *LiDAR* computado para vários *checkpoints* de treinamento para o método *I-JEPA + VICReg + SEM*

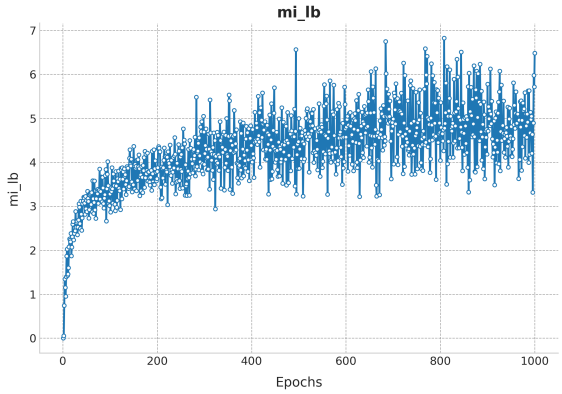

Fig. 7. *MINE* treinado para o método *I-JEPA + VICReg*

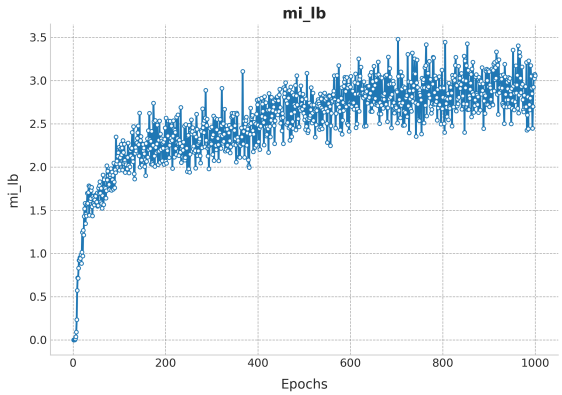

Fig. 8. *MINE* treinado para o método *I-JEPA + VICReg + SEM*

IV. Conclusão

Neste trabalho, exploramos a aplicação da teoria da informação ao aprendizado auto-supervisionado com múltiplas vistas, investigando como diferentes abordagens baseadas em maximização da informação mútua podem melhorar a qualidade das representações aprendidas. Em particular, propusemos modificações no modelo *I-JEPA*, incorporando princípios baseados em *VICReg* e *Simplicial Embeddings* para promover

representações mais estruturadas e informativas.

Os resultados experimentais demonstraram que as modificações propostas melhoraram consideravelmente a acurácia na tarefa de classificação subsequente, em comparação com a versão original do *I-JEPA*.

Além disso, avaliamos as melhorias propostas em diferentes configurações de dados e observamos que a combinação entre *VICReg* e *SEM* foi a melhor abordagem, alcançando os melhores resultados de acurácia. No entanto, constatamos que a redução da entropia induzida pelo *SEM* pode limitar a informação mútua entre as representações, causando um *trade-off* entre generalização e capacidade informativa.

Como direções futuras, sugerimos a exploração das técnicas propostas neste trabalho em outros domínios, especialmente naqueles em que não é simples utilizar *data augmentation*, como a área da saúde. Nessas aplicações, onde a integridade dos dados é fundamental para a interpretação, o *I-JEPA* se destaca por não gerar as vistas de forma artifical para o processo de aprendizado.

Além disso, considerando que a análise das representações por *MINE* e *LiDAR* não apresentou correlação direta com o desempenho na tarefa de classificação, é relevante explorar outras métricas para avaliar a qualidade das representações auto-supervisionadas.

Finalmente, os resultados obtidos reforçam a relevância da teoria da informação como ferramenta fundamental para o aprimoramento de métodos de aprendizado auto-supervisionado.

V. Referências

[1] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture, 2023.

[2] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning, 2022.

[3] Samuel Lavoie, Christos Tsirigotis, Max Schwarzer, Ankit Vani, Michael Noukhovitch, Kenji Kawaguchi, and Aaron Courville. Simplicial embeddings in self-supervised learning and downstream classification, 2022.

[4] Jing Zhao, Xijiong Xie, Xin Xu, and Shiliang Sun. Multi-view learning overview: Recent progress and new challenges, 2017.

[5] Xiaoqiang Yan, Shizhe Hu, Yiqiao Mao, Yangdong Ye, and Hui Yu. Deep multi-view learning methods: A review, 2021.

[6] R. Linsker. Self-organization in a perceptual network, 1988.

[7] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method, 2000.

[8] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: Mutual information neural estimation, 2021.

[9] Vimal Thilak, Chen Huang, Omid Saremi, Laurent Dinh, Hanlin Goh, Preetum Nakkiran, Joshua M. Susskind, and Etai Littwin. Lidar: Sensing linear probing performance in joint embedding ssl architectures, 2023.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[11] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training, 2018.

[12] Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning, 2022.

[13] Ravid Shwartz-Ziv and Yann LeCun. To compress or not to compress-self-supervised learning and information theory: A review, 2023.

[14] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019.

[15] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020.

[16] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning, 2020.

[17] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning, 2020.

[18] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021.

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners, 2021.

[20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[21] Ravid Shwartz-Ziv, Randall Balestriero, Kenji Kawaguchi, Tim G. J. Rudner, and Yann LeCun. An information-theoretic perspective on variance-invariance-covariance regularization, 2024.