

Manifold-Aware Regularization for Self-Supervised Representation Learning

by

M. Hadi Sepanj

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2025

© M. Hadi Sepanj 2025

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Ebrahim Bagheri
Professor, Dept. of Computer science, University of Toronto

Supervisor: Paul Fieguth
Professor, Dept. of Systems Design Engineering, University of Waterloo

Internal Members: Alexander Wong
Professor, Dept. of Systems Design Engineering, University of Waterloo
Bryan Tripp
Professor, Dept. of Systems Design Engineering, University of Waterloo

Internal-External Member: Pascal Poupart
Professor, Dept. of Computer Science, University of Waterloo

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Self-supervised learning (SSL) has emerged as a dominant paradigm for representation learning, yet much of its recent progress has been guided by empirical heuristics rather than unifying theoretical principles. This thesis advances the understanding of SSL by framing representation learning as a problem of *geometry preservation* on the data manifold, where the objective is to shape embedding spaces that respect intrinsic structure while remaining discriminative for downstream tasks. We develop a suite of methods—ranging from optimal transport-regularized contrastive learning (SinSim) to kernelized variance-invariance-covariance regularization (Kernel VICReg)—that systematically move beyond the Euclidean metric paradigm toward geometry-adaptive distances and statistical dependency measures, such as maximum mean discrepancy (MMD) and Hilbert–Schmidt independence criterion (HSIC).

Our contributions span both theory and practice. Theoretically, we unify contrastive and non-contrastive SSL objectives under a manifold-aware regularization framework, revealing deep connections between dependency reduction, spectral geometry, and invariance principles. We also challenge the pervasive assumption that Euclidean distance is the canonical measure for alignment, showing that embedding metrics are themselves learnable design choices whose compatibility with the manifold geometry critically affects representation quality. Practically, we validate our framework across diverse domains—including natural images and structured scientific data—demonstrating improvements in downstream generalization, robustness to distribution shift, and stability under limited augmentations.

By integrating geometric priors, kernel methods, and distributional alignment into SSL, this work reframes representation learning as a principled interaction between statistical dependence control and manifold geometry. The thesis concludes by identifying open theoretical questions at the intersection of Riemannian geometry, kernel theory, and self-supervised objectives, outlining a research agenda for the next generation of geometry-aware foundation models.

Acknowledgements

I acknowledge the use of generative AI (OpenAI's ChatGPT) solely for polishing the writing and improving clarity of expression. All ideas, analyses, and substantive content are entirely my own.

Dedication

To the silent architects of my becoming —
my parents, whose love and sacrifice are the soil from which I grew;
Paul, who beyond the compass of research, lent breadth to my sight and depth to my
thought, till my mind beheld unimagined horizons.

Those whose wisdom shaped me, whose patience carried me,
and whose belief burned brighter than my own.
To the boundless frontier of thought,
where questions are not obstacles but gateways,
and where the mind dares to imagine worlds
that the hands have yet to build.

To the future —
where human curiosity and artificial minds walk side by side,
not in rivalry, but in the shared pursuit of understanding the vastness within and beyond us.

To the apricity, a reminder of hope even in the abyss of despair,
And to every quiet hour,
every unseen struggle,
every whisper that urged me forward —
this work is the echo of that perseverance.

Table of Contents

Examining Committee	ii
Author’s Declaration	iii
Abstract	iv
Acknowledgements	v
Dedication	vi
List of Figures	xii
List of Tables	xiii
1 Introduction	1
1.1 Representation Learning: Foundations and Motivation	1
1.2 The Emergence of Self-Supervised Learning	2
1.3 From Early SSL to Modern Approaches	4
1.4 Motivation for This Thesis	5
1.5 Research Objectives and Contributions	6
1.6 Organization of the Thesis	7

2	Overview	9
2.1	General Framework of Self-Supervised Learning	9
2.2	Perspectives and Taxonomies of SSL	11
3	Sinkhorn-Regularized SimCLR	19
3.1	Motivation	19
3.2	Core Idea	20
3.3	Sinkhorn Regularization and Optimal Transport	21
3.4	SinSim	22
3.4.1	Network Architecture	22
3.4.2	Training Objective	22
3.4.3	Implementation Details	24
3.4.4	Theoretical Justification	25
3.5	Experiments	27
3.5.1	Experimental Setup	28
3.5.2	Linear Classification Performance	28
3.5.3	Non-Linear Classification Performance	29
3.5.4	Representation Visualization	29
3.5.5	Ablation Study	30
3.5.6	Implementation Details	33
3.6	Conclusion	36
4	Self-Supervised Learning Beyond Linear Correlation	37
4.1	Motivation	37
4.2	Core Idea	38
4.3	Overview and Intuition	39
4.4	The Neural Network Structure	40
4.5	Linear Correlation Loss Terms	41

4.5.1	Auto-Correlation of Samples	42
4.5.2	Cross-Correlation of Samples	42
4.5.3	Auto-Correlation of Features	43
4.5.4	Cross-Correlation of Features	44
4.6	Nonlinear Dependence Loss Terms	45
4.6.1	Auto-Dependence of Samples	45
4.6.2	Cross-Dependence of Samples	46
4.6.3	Auto-Dependence of Features	46
4.6.4	Cross-Dependence of Features	46
4.7	Overall Loss Function	47
4.7.1	Overall Loss Function	47
4.7.2	Special Cases of CDSSL	47
4.8	Experimental Results	49
4.8.1	Ablation Study	49
4.8.2	Evaluation and Comparison on Benchmarks	50
4.8.3	Evaluation by Nonlinear Classification	53
4.8.4	Domain Adaptation	53
4.8.5	Implementation Details	53
4.9	Conclusion	54
5	SSL in RKHS	55
5.1	Motivation and Core Idea	55
5.2	Review of VICReg	56
5.2.1	Data and Network Settings	56
5.2.2	VICReg Loss Function	56
5.3	Kernel VICReg	58
5.3.1	Covariance in RKHS	58
5.3.2	Kernelized Variance Regularization	60

5.3.3	Kernelized Covariance Regularization	60
5.3.4	Kernelized Invariance Term	61
5.3.5	Overall Kernel VICReg Loss	61
5.4	Discussions	62
5.4.1	Relation of Kernelized Variance Term with Kernel PCA	62
5.4.2	Connection to HSIC and Independence	64
5.4.3	Kernel Choice as Inductive Bias	64
5.5	Experimental Results	64
5.5.1	Comparison with Baselines	65
5.5.2	Further Analysis and Insights	66
5.5.3	Implementation Details	68
5.6	Conclusion	71
6	Context-Aware Augmentation	72
6.1	Motivation	72
6.2	Core Idea	73
6.3	Methodology	73
6.3.1	Modified SimCLR	74
6.3.2	Integration of Faster R-CNN	75
6.4	Results and Discussion	76
6.4.1	Results	76
6.4.2	Discussion	77
6.5	Implementation Details	78
6.6	Conclusion	78
6.6.1	Future Work	80

7	Manifold-Aware Batch-Level	81
7.1	Motivation	81
7.2	Core Idea	81
7.3	Methodology	82
7.3.1	Objective: VICReg with MMD-Enhanced Invariance	82
7.3.2	Theoretical Motivation for MMD-Enhanced Invariance in VICReg .	83
7.3.3	Parameter Selection and Kernel Choice	83
7.4	Experiments	84
7.4.1	Experimental Setup	84
7.4.2	Results and Analysis	85
7.4.3	t-SNE Visualization	86
7.4.4	Ablation Study on MMD Influence	87
7.4.5	Summary of Findings	88
7.5	Implementation Details	88
7.6	Conclusion	89
8	Conclusion	90
	References	95

List of Figures

1.1	Learning Paradigms	3
2.1	General SSL Framework	10
3.1	SinSim learned embeddings	30
3.2	Effect of Sinkhorn Regularization on SinSim-MNIST	31
3.3	Effect of Sinkhorn Regularization on SinSim-CIFAR-10	32
3.4	Effect of Sinkhorn iterations on SinSim-MNIST	33
3.5	Effect of Sinkhorn iterations on SinSim-CIFAR-10	34
3.6	Effect of λ on SinSim-MNIST	35
3.7	Effect of λ on SinSim-CIFAR-10	36
4.1	CDSSL framework	39
4.2	CDSSL optimal parameters	50
4.3	CDSSL embedding comparison	51
5.1	Kernel VICReg embeddings	67
6.1	Modified-SimCLR Schema	74
6.2	Modified SimCLR Downstream structure	76
7.1	t-SNE of VICReg	86
7.2	t-SNE of VICREG+MMD	87
7.3	MMD vs accuracy	88

List of Tables

3.1	SinSim accuracy	28
3.2	SinSim non-linear accuracy	29
4.1	CDSSL loss interpretations	49
4.2	CDSSL accuracy	52
4.3	CDSSL downstream	53
4.4	CDSSL domain adaptation	54
5.1	Kernel VICReg accuracy – Large-scale benchmarks	66
5.2	Kernel VICReg accuracy – Small-scale benchmarks	67
5.3	Kernel VICReg transfer learning	68
6.1	Modified SimCLR accuracy	77
6.2	Modified SimCLR downstream	77
7.1	VICReg+MMD accuracy	85

Chapter 1

Introduction

Recent years have witnessed extraordinary advances in artificial intelligence and machine learning, driven by the explosive availability of data and the rapid evolution of computational methods [94, 47]. From computer vision [106, 13] and natural language processing [20] to robotics [63, 75] and scientific discovery, data-driven models are increasingly central to automating decision-making and extracting insights from complex, high-dimensional environments. However, the success of these systems depends on how raw data is represented and understood by learning algorithms [10].

At the core of this transformation is the field of representation learning [81, 10, 64], which seeks to bridge the gap between raw sensory signals and the semantic or abstract information required for intelligent behavior. Hence, understanding, improving, and evaluating learned representations is an important research problem in modern AI. In this chapter, we introduce the foundations and motivation for representation learning, its impact on downstream machine learning tasks, and its evolving role in the context of self-supervised [5, 47, 62, 103] and deep learning [64, 53, 72, 118] frameworks.

1.1 Representation Learning: Foundations and Motivation

Representation learning lies at the heart of modern machine learning and artificial intelligence. It serves as the foundation for extracting, organizing, and encoding the underlying factors of variations in raw data [10]. At its core, *representation learning* is the process by which an algorithm transforms data—images, text, audio, or structured signals—into a

form that is more suitable for downstream tasks such as classification [124, 22, 2], regression [29, 35], clustering [119, 116], or decision-making [17, 91].

Traditional machine learning methods, such as support vector machines [23] or logistic regression [11], have long relied on carefully engineered features designed by human experts. These hand-crafted representations, while sometimes effective, are inherently limited by human intuition, domain expertise, and the curse of dimensionality [93]. As datasets have grown in size and complexity, and as applications have expanded into multiple domains, the inadequacy of manual feature engineering has become increasingly apparent.

Deep learning [64, 17, 124] marked a profound paradigm shift by automating the process of feature extraction. Neural networks, composed of multiple hierarchical layers, are capable of learning rich, multi-scale representations directly from raw data [31]. Each layer in a deep model can be interpreted as learning a more abstract, higher-level representation of the input, progressing from simple edges and colors in early layers (for images) to complex shapes and semantic concepts in deeper layers [64]. The success of deep learning across a wide range of applications can largely be attributed to its ability to learn powerful, data-driven representations that generalize well to unseen data [72].

The quality of learned representations is critical; representations that disentangle the true underlying factors of variation facilitate not only superior performance on supervised tasks, but also more efficient transfer learning [83, 126], robust generalization, and even better interpretability. Indeed, the effectiveness of modern AI systems is often bounded not by the choice of model architecture or optimization technique, but by the expressiveness and appropriateness of the representations they learn.

Representation learning provides the essential bridge between raw, high-dimensional data and the decision-making machinery of machine learning models [10]. Its centrality to AI arises from its capacity to automatically capture the intrinsic structure of data, thus enabling models to reason, generalize, and perform with minimal human intervention. In later chapters (Chapters 3–7), we show how such representations can be further shaped and regularized to respect both the semantic content of data and the underlying manifold structure.

1.2 The Emergence of Self-Supervised Learning

Machine learning approaches span a spectrum defined by the availability and nature of supervision: *supervised* learning [24, 80, 71] relies on large quantities of labeled data; *unsupervised* learning [55, 82, 9] explores patterns in completely unlabeled data; *semi-supervised*

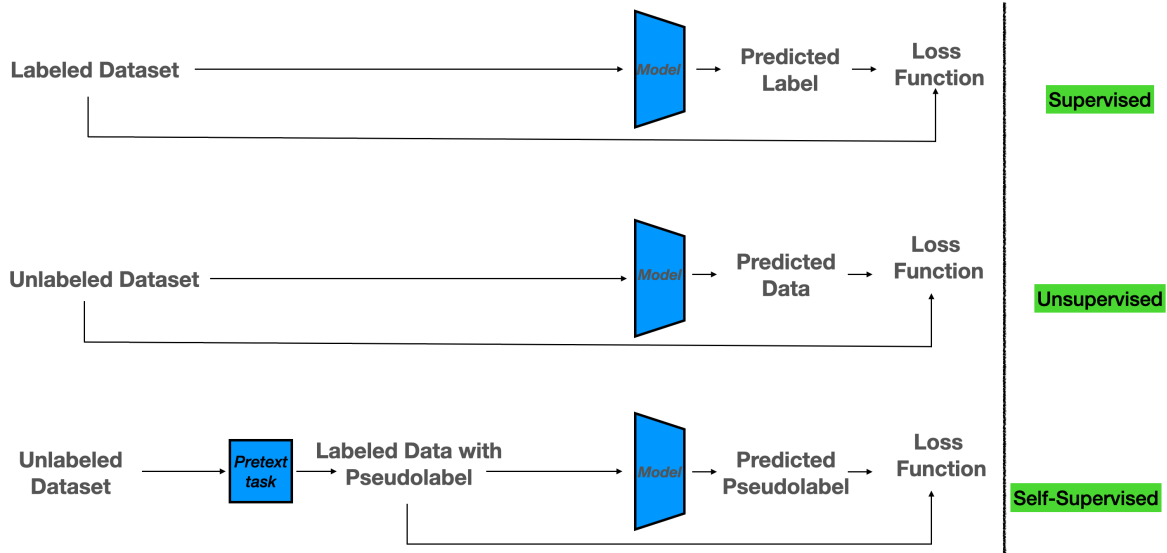


Figure 1.1: Contrasting supervised, unsupervised and self-supervised learning paradigms for training a model. Self-supervision methods introduce pretext tasks that generate pseudolabels for discriminative training.

learning [89, 125, 109] seeks to leverage limited labels in conjunction with abundant unlabeled samples.

In recent years, *self-supervised learning* (SSL) [54, 47, 73] has emerged as a powerful new paradigm that fundamentally reimagines how models acquire knowledge from data [47]. Unlike supervised learning, where human-annotated labels are required for each training example, self-supervised learning devises pretext tasks that exploit inherent structure or correlations within the data itself. Through the design of clever auxiliary objectives—such as predicting masked parts of an image, reconstructing context, or aligning different augmentations of the same input—SSL methods enable neural networks to learn useful and transferable representations without explicit human supervision. For an intuitive overview of supervised, unsupervised, and self-supervised learning, see Figure 1.1.

The key intuition [30] behind SSL is simple yet clever: rather than depending on manually crafted labels, the data themselves provides the supervision necessary for learning. This shift enables the utilization of massive, raw datasets, expanding access to large-scale training and reducing the cost and limitations associated with human annotation. As a result, self-supervised learning has led to big improvements in representation learning [32],

especially in domains where annotated data is scarce or expensive to obtain.

1.3 From Early SSL to Modern Approaches

The development of self-supervised learning has made its way through a remarkable journey, evolving from early generative frameworks [10, 112], to the dominance of contrastive [18, 52] and non-contrastive [46, 123] paradigms, and recently to the renewed interest in generative and predictive approaches [5, 7] that continue to expand the state of the art.

Early SSL methods, such as *autoencoders* [51] and *predictive coding* [88] frameworks, tried to learn useful representations by reconstructing input data or predicting future or missing elements. Autoencoders, for example, trained neural networks to encode input data into a compact latent space and decode it back to the original, with the objective of minimizing reconstruction error. Predictive coding extended this idea to temporal and sequential domains, encouraging models to anticipate forthcoming inputs from past context.

While these early approaches established the foundations for unsupervised representation learning, they were often limited by their reliance on low-level reconstruction objectives. Such objectives, while effective for capturing global data statistics, frequently failed to produce semantically rich or discriminative representations suitable for downstream tasks.

To address these shortcomings, the field moved towards *contrastive* learning methods. Notably, frameworks such as SimCLR [18] and MoCo [52] train models to maximize the agreement between different augmented views of the same input, while minimizing similarity with views from other samples. This contrastive objective proved effective at encouraging the emergence of powerful, transferable representations. However, contrastive methods often require large batch sizes and complex negative sampling strategies to avoid representational collapse.

The limitations of contrastive learning have inspired the development of *non-contrastive* approaches, exemplified by methods like BYOL [46], SimSiam [19], Barlow Twins [123], and VICReg [8]. These methods eliminate the explicit need for negative samples and instead rely on carefully designed regularization terms—such as variance preservation, feature decorrelation, and predictor architectures—to prevent trivial solutions. For instance, VICReg and Barlow Twins use decorrelation and variance-based losses to ensure that representations remain diverse and informative, addressing the pitfalls of both earlier generative methods and purely contrastive objectives.

This progression, from early autoencoders to modern regularization-based self-supervision, reflects an ongoing effort to discover more robust, efficient, and semantically meaningful ways for models to learn from unlabeled data.

1.4 Motivation for This Thesis

Despite the substantial progress in self-supervised learning, several fundamental challenges remain unresolved. Most contemporary SSL frameworks, while effective, still exhibit notable limitations that hinder their full potential:

- **Robust Invariance:** Many methods enforce invariance primarily at the pairwise level, aligning representations between specific augmented views of the same sample. However, this approach may fail to guarantee invariance across the wider set of transformations encountered in practice, leading to suboptimal generalization and non-robust representations.
- **Batch-Level Alignment:** Conventional SSL losses are often agnostic to the broader distributional structure of a batch. As a result, they may overlook higher-order dependencies and relationships among all samples, limiting the ability of models to capture comprehensive, cohesive feature spaces.
- **Nonlinear Structure:** The majority of SSL objectives are formulated in Euclidean space, relying on linear statistics (such as mean, variance, or covariance). This can be insufficient for capturing the complex, nonlinear manifolds that often underlie real-world data, potentially constraining the expressivity and adaptability of learned representations.

These challenges serve as the guide of this dissertation. Chapter 3 addresses the lack of global structure in contrastive learning via Sinkhorn-based optimal transport. Chapter 4 and Chapter 5 extend beyond linear correlation by incorporating statistical dependence and RKHS kernels. Chapter 6 reframes augmentations to better preserve semantic consistency, while Chapter 7 integrates batch-level distributional alignment through MMD. The central aim of this thesis is to address these pressing gaps in SSL. Each proposed method targets a distinct aspect of these open questions:

- By integrating batch-level distribution alignment techniques such as Maximum Mean Discrepancy (MMD), we encourage models to develop robust invariance across the entire set of augmented views within a batch.

- By kernelizing core SSL objectives, we enable learning in high-dimensional Reproducing Kernel Hilbert Spaces (RKHS), thereby equipping models to capture both linear and nonlinear dependencies in the data.
- Throughout, the emphasis is on designing self-supervised objectives that lead to more stable, generalizable, and semantically meaningful representations—bridging the gap between theory and practical robustness in representation learning.

1.5 Research Objectives and Contributions

The primary objective of this thesis is to advance the theoretical understanding and practical effectiveness of self-supervised representation learning. In pursuit of this goal, the following research questions guide the work:

- **How can batch-level alignment and distributional consistency be enforced in SSL frameworks to promote more robust invariance and richer representations?**
- **Can kernel methods be systematically integrated into SSL objectives to capture nonlinear dependencies and manifold structures in learned representations?**
- **What are the empirical impacts of these enhancements across datasets, in terms of representation quality and transferability?**

The thesis makes the following key contributions:

1. **A novel batch-level alignment mechanism:** We introduce the integration of Maximum Mean Discrepancy (MMD) into the VICReg framework, enabling alignment of feature distributions across the entire batch and promoting manifold-aware invariance (Chapter 7).
2. **Kernelized self-supervised objectives:** We propose a principled methodology for lifting SSL losses—variance, invariance, and covariance—into Reproducing Kernel Hilbert Space (RKHS), exemplified by the Kernel VICReg method. This allows for nonlinear, expressive feature learning without explicit mapping (Chapter 5).

3. **Unified correlation–dependence SSL framework:** We introduce *Correlation-Dependence Self-Supervised Learning (CDSSL)*, a principled and comprehensive framework that generalizes existing self-supervised methods by simultaneously addressing both linear correlations and nonlinear dependencies. By organizing the loss into eight interpretable components—spanning sample- and feature-level, auto- and cross-relations—we enforce diversity, disentanglement, and augmentation invariance in the learned representations (Chapter 4).
4. **Nonlinear dependence via HSIC in RKHS:** CDSSL leverages the Hilbert-Schmidt Independence Criterion (HSIC) to capture rich, nonlinear dependencies in both sample- and feature-wise embeddings within a Reproducing Kernel Hilbert Space. This enriches representation learning beyond linear statistical assumptions, enabling better generalization across datasets (Chapter 4).
5. **Geometry-aware regularization:** We propose *SinSim*, a Sinkhorn-regularized extension of SimCLR that integrates entropy-regularized optimal transport to impose global geometric structure on the representation space. By aligning feature distributions through Sinkhorn loss, SinSim enhances class separation and mitigates mode collapse, offering a principled alternative to statistical regularizers (Chapter 3).

Collectively, these contributions represent a step toward more robust, generalizable, and theoretically grounded self-supervised learning methods. The findings of this thesis have implications not only for SSL research but also for a wide range of applications that rely on effective unsupervised representation learning.

1.6 Organization of the Thesis

This thesis is structured to provide a cohesive and logical flow from motivation to contributions, deliberately foregoing a standalone background chapter in favor of integrating relevant literature and theoretical foundations within each major section. Chapter 3 introduces SinSim, a contrastive method with Sinkhorn regularization; Chapter 4 develops CDSSL to unify correlation and dependence; Chapter 5 extends these ideas into RKHS with Kernel VICReg; Chapter 6 proposes context-aware augmentation; and Chapter 7 presents MMD-enhanced VICReg for manifold-aware batch-level alignment. The thesis concludes in Chapter 8, which integrates these findings into a unified geometry-preserving framework for self-supervised learning.

The thesis opens with an introduction, motivating the importance of representation learning and the emergence of self-supervised learning (SSL). Immediately following, an overview section highlights the interconnections between the subsequent research chapters and articulates the overarching narrative that unifies the individual contributions.

Each of the main research chapters (3 - 7) is organized as follows:

- **Motivation and Core Idea:** Introduces the specific research question or gap being addressed.
- **Background and Related Work:** Presents the technical context and prior work most relevant to the proposed method.
- **Methodology:** Details the proposed approach, theoretical analysis, and design choices.
- **Experiments and Results:** Empirically evaluates the method on relevant benchmarks, with quantitative and qualitative analysis.
- **Discussion and Implications:** Interprets the results, draws connections to other chapters, and outlines avenues for future work.

A dedicated overview in chapter 2 summarizes how the individual works relate and collectively advance the field of self-supervised representation learning. The thesis concludes with a synthesis of the main findings, implications for the broader research community, and reflections on possible future directions.

Chapter 2

Overview

The purpose of this overview chapter is twofold. First, it aims to distill the fundamental principles, taxonomies, and mathematical underpinnings of self-supervised learning into a cohesive narrative. Second, it serves to explicitly link the subsequent chapters of this thesis by situating them within a unified framework.

This chapter provides the necessary scaffolding to understand how each methodological innovation addresses a specific gap or limitation in the broader SSL pipeline. By formalizing the general structure of SSL this chapter demonstrates that the research contributions that follow are not isolated explorations, but rather coordinated steps toward a more robust, expressive, and theoretically grounded approach to representation learning.

2.1 General Framework of Self-Supervised Learning

Self-supervised learning (SSL) provides a principled way to learn data representations without requiring human-annotated labels [47, 72]. Despite the diversity of existing methods, most SSL approaches can be abstracted into a common pipeline, characterized by a shared set of components and objectives. This section formalizes the typical structure of an SSL framework (see Figure 2.1) and highlights the principal elements that underlie its success.

Formalization

At its core, an SSL pipeline can be described by the following elements:

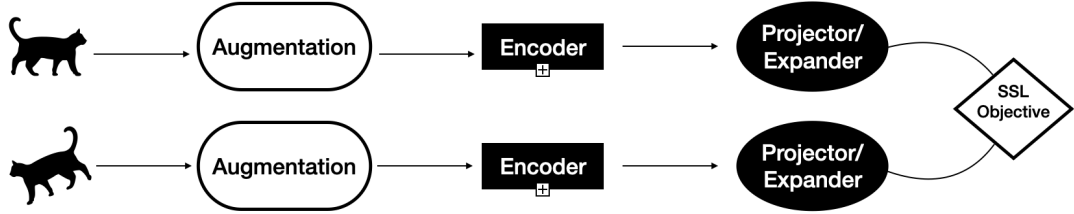


Figure 2.1: General SSL framework

- **Data:** Let $\{x_i\}_{i=1}^M \subset \mathbb{R}^d$ denote the unlabeled dataset, where each x_i resides in a high-dimensional input space $\mathbb{R}^d$ (e.g., images, signals, sequences).
- **Augmentations:** A set of stochastic transformations $\mathcal{T} = \{t^k\}$ is defined, where each $t^k : \mathbb{R}^d \rightarrow \mathbb{R}^d$ generates a perturbed or alternative view of the data. Given an input x_i , two augmentations $t^j, t^l \sim \mathcal{T}$ produce different views denoted by $x_i^j = t^j(x_i)$ and $x_i^l = t^l(x_i)$, where $j \neq l$ are temporary indices referring to the specific augmentation functions.
- **Batch Construction:** A batch of data $\{\mathbf{x}_i\}_{i=1}^b$ is sampled from the dataset. For each input, multiple stochastic augmentations (e.g., cropping, color jitter, noise) are applied, yielding sets of augmented views $\{\mathbf{x}_i^j, \mathbf{x}_i^l\}$.
- **Encoder and Projector / Expander:** An encoder network $f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^q$ parameterized by θ maps augmented data to an intermediate latent representation, which is often further processed by a head network $g_\phi : \mathbb{R}^q \rightarrow \mathbb{R}^p$ to produce the final feature vectors $z = g_\phi(f_\theta(\mathbf{x}))$. The architectural choice of the head is notable: in several frameworks such as SimCLR [18] and BYOL [46] the projector head reduces dimensionality ($p < q$), serving primarily to limit the information content of the representation space and focus learning on key invariances. In contrast, works like VICReg [8] and Barlow Twins [123] deliberately *expand* the representation ($p > q$) using an *expander* head.¹

¹This expander design is motivated by the need to decorrelate and regularize higher-dimensional features using variance and covariance penalties: having $p > q$ enables richer feature decorrelation (covariance regularization is more meaningful in a higher-dimensional space), encourages diversity in learned representations, and mitigates collapse by spreading information over more dimensions. In the context of VICReg, the expansion helps balance the three loss components—invariance, variance, and covariance—by providing sufficient capacity for decorrelation and ensuring that the variance and covariance objectives are effective. The choice between a projector ($p < q$) and expander ($p > q$) thus reflects different philosophies about

- **Objective Function:** A self-supervised loss $\mathcal{L}_{\text{SSL}}$ is defined to enforce desired properties (invariance, diversity, decorrelation, etc.) between the representations of positive pairs (different views of the same instance) and, optionally, negative pairs (views of different instances). A general mathematical formulation for the SSL objective is:

$$\mathcal{L}_{\text{SSL}} = \frac{1}{M} \sum_{i=1}^M \ell(z_i^j, z_i^l; \mathcal{Z}_{\text{neg}}) + \mathcal{R}(\mathcal{Z}), \quad (2.1)$$

where $\ell(\cdot)$ is a primary alignment (or contrastive) loss acting on positive (and potentially negative) pairs, $\mathcal{Z}_{\text{neg}}$ denotes negative samples if present, and $\mathcal{R}(\mathcal{Z})$ encompasses regularization terms (e.g., variance, covariance, or distributional alignment).

Well-known examples of this general structure include SimCLR (contrastive loss with negatives), BYOL (pairwise alignment without negatives and with a target network), Barlow Twins (redundancy reduction), and VICReg (explicit variance and covariance regularization).

Henceforth, we simplify the notation by referring to the first and second augmentations of the same batch of data as $\mathbf{x}$ and $\mathbf{x}'$, respectively. This convention aligns with the majority of self-supervised learning (SSL) frameworks discussed in this dissertation, which rely on paired augmentations for representation learning. This applies for h and z as well.

2.2 Perspectives and Taxonomies of SSL

Contrastive vs. Non-Contrastive SSL

Self-supervised learning (SSL) frameworks can be broadly categorized by whether they rely on explicit comparison with negative samples—a distinction that shapes both theoretical underpinnings and practical behavior.

Contrastive methods, exemplified by SimCLR [18], MoCo [52], and InfoNCE-based models [81], explicitly maximize agreement between representations of positive pairs (augmentations of the same sample) while simultaneously pushing apart representations of negative pairs (different samples). These approaches hinge on carefully constructed contrastive losses (e.g., NT-Xent [1], InfoNCE [114]) and often require large batch sizes or

where and how invariance and diversity should be enforced in the representation pipeline.

memory banks to provide sufficient negatives, which serve as anchors for the latent space geometry.

Non-contrastive methods, such as BYOL [46], SimSiam [19], Barlow Twins [123], and VICReg [8], remove negative pairs altogether. Instead, they employ architectural tricks (e.g., stop-gradient, asymmetric networks), redundancy reduction, or decorrelation objectives to avoid trivial solutions. These methods simplify the learning process and reduce computational demands but introduce subtler theoretical questions about why collapse is avoided.

Both aim to learn invariant, robust representations, but the mechanisms differ: contrastive methods enforce *relative* geometry, while non-contrastive methods encourage *absolute* structure within representations.

Alignment Granularity: Pairwise vs. Batch-Level

A complementary perspective on SSL methods is through the lens of alignment granularity—that is, *how* and *at what scale* representation alignment is enforced.

Pairwise alignment consists of most classical and modern SSL approaches, including SimCLR, BYOL, Barlow Twins, and VICReg. Here, objectives are constructed to encourage similarity (or dissimilarity) between individual pairs of augmented samples, typically via ℓ_2 loss, cross-entropy, or correlation measures. While pairwise approaches have achieved strong empirical performance, they focus exclusively on local relationships and may fail to capture higher-order dependencies or global structure.

Batch-level alignment aims to overcome these limitations by aligning entire distributions of representations within a batch. Techniques such as Maximum Mean Discrepancy (MMD) [44] and Hilbert-Schmidt Independence Criterion (HSIC) [45] provide principled, kernel-based means for matching feature distributions or enforcing independence between different parts of the representation. Batch-level objectives promote greater consistency, robustness, and invariance, capturing both local and global geometry of the data manifold. Recent work—including several methods in this thesis—explores these distributional alignment techniques to push beyond the constraints of pairwise objectives.

Linear vs. Nonlinear Representation Alignment

A further axis of taxonomy distinguishes between *linear* (Euclidean) and *nonlinear* (kernel) approaches to representation alignment.

Linear (Euclidean) alignment forms the basis for most established SSL methods, including SimCLR, BYOL, and VICReg. Objectives such as mean squared error, covariance regularization, and cross-correlation are computed directly in the learned feature space, implicitly assuming that linear operations are sufficient to capture relevant structure. While computationally efficient and well understood, these approaches may struggle to model highly nonlinear relationships—especially in settings where the underlying data manifold is complex or highly curved.

Nonlinear (kernelized) alignment addresses this gap by lifting the alignment objective into a Reproducing Kernel Hilbert Space (RKHS) [12, 39, 122], enabling the use of powerful, implicit feature mappings. Methods such as kernel VICReg (proposed in this thesis, chapter 5) reformulate variance, invariance, and decorrelation losses using double-centered kernel matrices and Hilbert-Schmidt norms, thereby capturing both local and global nonlinear dependencies. The kernel perspective unifies and generalizes many classical objectives and equips SSL with the flexibility to match the geometry of real-world data.

The choice between linear and nonlinear approaches has practical implications for expressivity, generalization, and robustness. This thesis investigates both, with an emphasis on bridging their strengths via principled mathematical formulations and empirical validation.

Manifold-Aware Representation Learning

While much of the prior self-supervised learning (SSL) literature has focused on pairwise alignment and statistical regularization within Euclidean spaces, this thesis introduces a unifying perspective grounded in the *manifold hypothesis*—namely, that high-dimensional observations concentrate near low-dimensional, smooth submanifolds embedded in the ambient input space [34]. From this perspective, the challenge of representation learning becomes one of identifying mappings that preserve and expose this intrinsic geometry, rather than merely optimizing pointwise objectives.

Why manifolds? The *manifold hypothesis* [10, 34] asserts that high-dimensional observations encountered in machine learning—such as images, signals, or protein surfaces—lie on or near a smooth, low-dimensional Riemannian submanifold [66] $\mathcal{M} \subset \mathbb{R}^d$. Embracing this inductive bias is not merely convenient—it yields powerful regularization mechanisms for self-supervised learning (SSL), grounding the learned representations in the intrinsic geometry of the data.

Local Isometry and Tangent Space Alignment

Let $f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^k$ be an encoder trained via SSL. A manifold-aware objective encourages f_θ to behave as a *local isometry* on $\mathcal{M}$: for nearby points $x, y \in \mathcal{M}$, the geodesic distance $d_{\mathcal{M}}(x, y)$ should satisfy

$$\|f_\theta(x) - f_\theta(y)\| \approx d_{\mathcal{M}}(x, y).$$

In differential terms, this corresponds to enforcing that the Jacobian $Df_\theta(x)$ approximately preserves the inner product on the tangent space $T_x\mathcal{M}$, i.e.,

$$\langle Df_\theta(x)v, Df_\theta(x)w \rangle \approx \langle v, w \rangle \quad \text{for all } v, w \in T_x\mathcal{M}.$$

Such local isometry preserves neighborhood structure and angles, mitigating representation collapse and encouraging discriminative features that remain faithful to the geometry of the underlying data.

Spectral Regularization via Laplacian Smoothness

Another foundational benefit arises from spectral geometry. If the data lie on a manifold $\mathcal{M}$, then the Laplace–Beltrami operator [14, 117] $\Delta_{\mathcal{M}}$ provides a natural basis $\{\phi_k\}$ for functions defined on $\mathcal{M}$:

$$-\Delta_{\mathcal{M}}\phi_k = \lambda_k\phi_k.$$

Laplacian-based regularization, commonly enforced via smoothness penalties or augmentation-invariance, encourages each component $f_j(x)$ of the learned representation to lie in the span of the low-frequency eigenfunctions $\{\phi_k\}_{k=1}^d$. This ensures that the embedding function varies slowly along $\mathcal{M}$, avoiding spurious high-frequency artifacts and capturing semantically meaningful global structure.

Recent analyses [6, 104] reveal that SSL objectives such as SimCLR, VICReg, and Barlow Twins implicitly recover spectral embeddings related to Laplacian Eigenmaps. These approaches minimize the Dirichlet energy $\int_{\mathcal{M}} \|\nabla_{\mathcal{M}} f_j\|^2 dV$, aligning SSL with classical manifold learning techniques.

Intrinsic vs. Extrinsic Distances

Geodesic distances $d_{\mathcal{M}}(x, y)$ often provide a more faithful measure of semantic similarity than ambient Euclidean distances $\|x - y\|$ [107]. For instance, points on opposite ends of a curved structure (e.g., a horseshoe) may appear close in Euclidean space but are

far apart along the manifold. Manifold-aware SSL avoids such distortions by embedding data in ways that preserve intrinsic distances, often through local graphs or kernel-based similarity measures. This preserves the relational structure and enhances generalization, as the learned features more accurately reflect meaningful data geometry.

Global Alignment through Batch-Level and Kernelized Objectives

SSL methods increasingly leverage *global alignment* by optimizing over entire batches or distributions. Methods like Barlow Twins and VICReg match batch-level statistics (e.g., cross-correlation matrices) and enforce feature decorrelation:

$$\mathcal{L}_{\text{decorrelation}} = \sum_{i \neq j} (\text{Corr}(z_i, z_j))^2,$$

which prevents feature redundancy and encourages a full-rank, isotropic embedding.

More sophisticated approaches, such as those using *Hilbert-Schmidt Independence Criterion* (HSIC) or *Maximum Mean Discrepancy* (MMD), align entire feature distributions in Reproducing Kernel Hilbert Spaces (RKHS):

$$\text{HSIC}(Z_1, Z_2) = \text{Tr}(K_{Z_1} H K_{Z_2} H), \quad \text{MMD}^2(P, Q) = \|\mu_P - \mu_Q\|_{\mathcal{H}}^2.$$

These kernelized losses are sensitive to higher-order structure and implicitly operate on the manifold $\mathcal{M}$ by comparing distributions with respect to geometry-aware kernels (e.g., Gaussian RBF). They go beyond pairwise alignment to preserve the global shape of the data, mitigating mode collapse and encouraging expressive, geometry-consistent features.

Optimal Transport and Sinkhorn Regularization

The Sinkhorn distance, a differentiable approximation of the Wasserstein distance [110], enables alignment of soft cluster assignments between views:

$$\mathcal{L}_{\text{SwAV}} = \sum_i - \sum_k P_{ik}^* \log Q_{ik},$$

where P^* is the balanced transport plan obtained via the Sinkhorn–Knopp algorithm [59]. This approach encourages diverse prototype usage and prevents degenerate collapse by distributing samples uniformly across the manifold’s latent modes.

The notion of **manifold-awareness** in SSL refers to the explicit modeling, regularization, and preservation of the underlying geometric structure throughout the learning pipeline. Instead of treating samples as i.i.d. points in $\mathbb{R}^d$, manifold-aware methods consider how local neighborhoods, curvature, and geodesic distances inform both alignment and diversity. This geometric framing yields several analytical advantages:

1. **Invariance Beyond Augmentations:** Classical SSL objectives enforce invariance by minimizing distances between augmented views under Euclidean metrics. However, these distances may distort true semantic proximity when the data lie on a curved manifold. A manifold-aware approach (Chapters 6 and 7) views according to their geodesic paths or intrinsic local structure, leading to better generalization and robustness under unseen or distributionally shifted transformations.
2. **Global Consistency via Distributional Matching:** By lifting objectives such as MMD into kernel spaces, we shift from pairwise matching to *distributional alignment*, inherently capturing global structure across batches. This is crucial for ensuring that entire regions of the data manifold—not just individual samples—are coherently mapped across augmentations (Chapters 3 and 4).
3. **Regularization in Tangent Spaces:** Variance and covariance regularization, when naively applied in Euclidean coordinates, can suppress or distort curvature. In contrast, when reformulated using kernel methods or nonlinear projections, these regularizers operate in spaces that more faithfully approximate the local tangent structure of the data manifold, preserving its richness while preventing collapse (Chapters 4 and 5).
4. **Nonlinear Alignment via RKHS Mappings:** The transition from linear to kernelized alignment, as in Kernel VICReg (Chapter 5), allows for modeling of higher-order, nonlinear dependencies in the representation space. This lifting into Reproducing Kernel Hilbert Space (RKHS) effectively encodes richer neighborhood structures and facilitates alignment in spaces where Euclidean distances are no longer meaningful.

Collectively, these insights form a coherent, geometrically motivated approach to SSL—one that transcends the limitations of traditional alignment paradigms. Each methodological contribution in this thesis can be viewed as a step toward **embedding the manifold structure into the learning process itself**, either by enforcing alignment of distributions that respect local topology (via MMD), penalizing dependencies that obscure factorized structure (via HSIC), or expanding the capacity of the model to encode curvature and nonlinear relations (via kernelization and expander networks).

This *manifold-aware lens* provides not only a theoretical justification for the proposed techniques but also a design philosophy: we regularize not just to prevent collapse, but to sculpt representations that remain faithful to the semantic and geometric properties of the data. In this way, the thesis positions itself at the intersection of statistical learning, information geometry, and representation theory—charting a unified path toward robust, expressive, and transferable SSL models.

We situate each core chapter of the thesis within the conceptual taxonomy introduced in this very chapter, specifically along the axes of *contrastive vs. non-contrastive learning*, *pairwise vs. batch-level alignment*, *linear vs. nonlinear alignment*, and *manifold-aware representation learning*. Each chapter targets specific facets of these distinctions, and together they progressively advance our goal of developing robust, expressive, and theoretically grounded self-supervised learning (SSL) models. Below, we outline how each chapter contributes to this overarching vision.

Chapter 3: Sinkhorn-Regularized SimCLR (SinSim). Returning to contrastive learning, this chapter enhances SimCLR with a *Sinkhorn-based optimal transport regularizer*. SinSim introduces **batch-level** structure via OT, while preserving **pairwise** contrastive loss. Although rooted in contrastive principles, it injects **manifold-awareness** and partial **nonlinear alignment** by solving a relaxed OT problem across the batch. This yields globally coherent embeddings that resist collapse and promote cluster separation. SinSim demonstrates that contrastive SSL can benefit from structure-aware regularization, offering a complementary route to achieving expressive and geometry-sensitive representations.

Chapter 4: Correlation-Dependence SSL (CDSSL). This chapter proposes a unified SSL framework, *CDSSL*, which blends feature-wise correlation (as in VICReg and Barlow Twins) with batch-level HSIC-based dependency maximization. CDSSL straddles both **contrastive and non-contrastive** design principles, combining **pairwise** feature alignment with **batch-level nonlinear** regularization. By addressing both feature redundancy and sample-wise structure, it yields highly **robust, expressive, and theoretically grounded** representations. It represents a culmination of insights from earlier chapters, integrating augmentation-awareness, manifold structure, and RKHS-based dependence into one coherent objective.

Chapter 5: Nonlinear Dependence in SSL via RKHS. This chapter deepens the theoretical backbone of SSL by analyzing representation alignment from a statistical depen-

dence perspective. Using the *Hilbert-Schmidt Independence Criterion (HSIC)* in RKHS, the model maximizes nonlinear dependence between views across batches. It is fully **non-contrastive**, **nonlinear**, and **batch-level**, directly promoting **expressiveness** by capturing high-order relationships beyond linear correlation. It also strengthens the thesis’s emphasis on **theoretical rigor**, offering a principled lens through which to understand and unify existing SSL objectives.

Chapter 6: Context-Aware Contrastive Learning. This chapter introduces a contrastive, pairwise SSL framework based on a novel *context-aware augmentation* strategy. Operating within the traditional instance discrimination setup, the method maintains linear pairwise alignment by ensuring that augmentations preserve semantically meaningful content. While remaining in the contrastive and pairwise regime, this chapter enhances **robustness** (by avoiding destructive transformations) and improves **expressiveness** (by encoding contextual semantics). Though not yet incorporating nonlinear or batch-level alignment, this method provides a solid foundation for further extensions, especially in promoting more faithful invariance.

Chapter 7: Manifold-Aware Batch-Level Alignment. Shifting away from pairwise contrastive losses, this chapter introduces *Maximum Mean Discrepancy (MMD)* into a VICReg-style non-contrastive framework, enabling alignment of entire batches of representations across views. This constitutes a move toward **batch-level**, **nonlinear** alignment and explicitly targets **manifold-awareness** by regularizing the structure of the feature space using kernel methods. By preserving global structure and reducing representational noise, the method improves both robustness and theoretical grounding through statistical regularization.

Chapter 3

Sinkhorn-Regularized SimCLR

In this chapter, we introduce **SinSim**, a principled extension of SimCLR that integrates Sinkhorn regularization from optimal transport theory [111, 33, 108]. By combining the NT-Xent contrastive loss [18] with an entropy-regularized Wasserstein distance [111, 57] computed via the Sinkhorn–Knopp algorithm [59, 25, 87], SinSim enforces a well-dispersed, geometry-aware latent space while preserving discriminative power. We begin by reviewing the theoretical underpinnings of Sinkhorn regularization, then detail the SinSim architecture and loss, and finally validate our approach through both theoretical analysis and empirical evaluations.

3.1 Motivation

Contrastive learning methods such as SimCLR rely on pairwise similarity maximization, which ensures that different augmentations of the same image remain close in the latent space. However, this approach does not necessarily promote a globally well-structured representation space, which can lead to poor generalization and mode collapse [41]. Without an explicit regularization mechanism, representations may cluster too tightly or fail to maintain meaningful global relationships among different samples [121].

While methods such as MoCo [52] and BYOL [46] have attempted to mitigate these issues by employing momentum encoders and redundancy reduction, the lack of an explicit global regularizer remains a limitation [46]. We conjecture—and subsequently demonstrate—that incorporating an optimal transport-based regularization term can explicitly structure the representation space while preserving discriminative power.

To address these challenges, we propose SinSim [102], a novel extension of SimCLR that integrates Sinkhorn regularization from optimal transport theory into the contrastive learning framework. Optimal transport provides a principled way to align distributions while preserving global structure, making it a natural choice for contrastive learning. The Sinkhorn loss, an entropy-regularized version of the Wasserstein distance, enforces a structured and well-dispersed latent space by promoting geometric consistency. This results in feature representations that are not only more discriminative but also more robust to variations in data.

This limitation directly echoes the broader challenges discussed in chapter 2, particularly the need for methods that impose global structure and prevent collapse in learned representations. SinSim represents one attempt to operationalize these principles.

3.2 Core Idea

SinSim extends the SimCLR framework by integrating Sinkhorn regularization from optimal transport theory [25, 111] into the contrastive learning objective. The key idea is to combine the standard contrastive loss, which encourages positive pairs to be close and negative pairs to be far apart, with a geometry-aware regularizer that shapes the overall structure of the learned feature space.

The Sinkhorn loss acts as an entropy-regularized Wasserstein distance [76, 27]. It measures the minimal transport cost required to align the distributions of features from different augmented views, while also including an entropy term for smoothness and computational efficiency. By including this loss, SinSim enforces a well-dispersed and globally consistent latent space, preserving the discriminative power of the representations.

This integration allows the network to not only maximize agreement between positive pairs, as in SimCLR, but also to organize the feature space in a way that reflects the global geometry of the data. As a result, SinSim learns features that are both robust and semantically meaningful. While this chapter focuses on geometry-aware regularization via optimal transport, later contributions (e.g., Chapter 4 on correlation-dependence SSL and Chapter 5 on kernelized objectives) extend this principle toward batch-level distributional alignment and nonlinear dependence capture.

3.3 Sinkhorn Regularization and Optimal Transport

Optimal transport (OT) provides a principled framework for comparing probability distributions in a geometrically meaningful way [3, 25, 111]. Unlike conventional similarity measures, OT considers the minimal *transport cost* required to morph one distribution into another. The classic Wasserstein distance [111, 57], also known as the Earth Mover’s Distance (EMD), quantifies this cost as

$$W_c(\mu, \nu) = \inf_{\gamma \in \Pi(\mu, \nu)} \sum_{i,j} c(x_i, y_j) \gamma_{ij}, \quad (3.1)$$

where $\Pi(\mu, \nu)$ represents the set of joint probability distributions (or *couplings*) with marginals μ and ν , $c(x_i, y_j)$ defines the cost to transport mass between samples, and γ is a transport plan whose entry γ_{ij} specifies the amount of mass transported from x_i to y_j .

Although powerful, the direct computation of the Wasserstein distance is computationally prohibitive due to its reliance on solving a linear program. To address this, Sinkhorn regularization introduces an entropy term that smooths the transport plan and enables efficient optimization:

$$W_\lambda(\mu, \nu) = \inf_{\gamma \in \Pi(\mu, \nu)} \sum_{i,j} c(x_i, y_j) \gamma_{ij} + \lambda H(\gamma), \quad (3.2)$$

where $H(\gamma) = -\sum_{i,j} \gamma_{ij} \log \gamma_{ij}$ is the entropy term and $\lambda > 0$ controls the trade-off between the transport cost and the degree of smoothing. This entropy regularization not only improves numerical stability but also leads to a differentiable approximation of the transport distance. The *Sinkhorn-Knopp algorithm* [25] iteratively normalizes the rows and columns of the transport plan to meet the marginal constraints, and it converges rapidly in practice, making it particularly suitable for deep learning applications.

The integration of optimal transport into representation learning has gained traction in recent years. Several works have explored the use of Wasserstein distance for deep learning tasks such as domain adaptation [108, 36, 120], generative modeling [48, 92, 48], and clustering [33, 26]. In self-supervised learning, optimal transport has been investigated as a tool for feature alignment [33, 86, 95] and structured representation learning [84, 4].

Contrastive learning methods traditionally rely on predefined similarity measures such as cosine similarity or Euclidean distance [62]. However, these metrics do not capture the global structure of learned representations. Recent efforts such as CDSSL [103], MMD-Based VICReg [101], VICReg [8] and Barlow Twins [123] introduced alternative regularization strategies based on variance-covariance matrices to enhance representation dispersion.

Our approach builds upon these insights by leveraging Sinkhorn-based optimal transport to enforce a structured latent space while maintaining the contrastive learning framework of SimCLR.

In contrast to prior work, our proposed method, **SinSim**, explicitly integrates Sinkhorn loss into SimCLR, bridging the gap between instance-based alignment and global structure regularization. We hypothesize that this hybrid approach not only prevents embedding collapse but also improves the overall robustness of learned representations.

3.4 SinSim

In this section, we detail the architecture and training procedure of our proposed **SinSim** framework, emphasizing the distinct roles of the intermediate representation h and the final embedding z .

3.4.1 Network Architecture

Our model comprises two primary components:

1. The **Encoder Network** f_θ transforms the input data $\mathbf{x}$ into intermediate representations h :

$$h = f_\theta(\mathbf{x})$$

These representations capture essential features of the input data.

2. The **Projection Head** g_ϕ maps h to the final embedding space z :

$$z = g_\phi(h)$$

The projection head is designed to facilitate the learning of embeddings suitable for contrastive objectives.

3.4.2 Training Objective

The training process involves two key components. First, there is the Contrastive Loss on final embedding, in which we apply a contrastive loss, such as the NT-Xent loss [18], on the embeddings z and z' obtained from different augmentations of the same input. This

loss encourages the model to bring positive pairs closer and push negative pairs apart in the embedding space:

$$\mathcal{L}_{\text{contrastive}} = -\log \frac{\exp(\text{sim}(z, z')/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(z_i, z_k)/\tau)} \quad (3.3)$$

where $\text{sim}(\cdot, \cdot)$ denotes cosine similarity, τ is a temperature parameter, and N is the batch size.

Next, to enforce a well-structured representation space and prevent collapse, we introduce a **Sinkhorn Regularization** on the intermediate representations h and h' corresponding to the embeddings z and z' . The Sinkhorn loss [25] is defined as

$$\mathcal{L}_{\text{Sinkhorn}} = \min_{\gamma \in \Pi(\mu, \nu)} \sum_{i,j} \gamma_{ij} c(h^{(i)}, h'^{(j)}) + \lambda H(\gamma) \quad (3.4)$$

where $h^{(i)}$ represent the intermediate representation of the i -th sample in the batch¹. The total loss is then a weighted combination of these two terms:

$$\mathcal{L} = \mathcal{L}_{\text{contrastive}} + \beta \mathcal{L}_{\text{Sinkhorn}} \quad (3.5)$$

where β controls the influence of the Sinkhorn regularization.

Rationale for Regularization on h and Clustering on z

Applying Sinkhorn regularization on the intermediate representations h rather than the final embeddings z offers several advantages:

- **Preservation of Information:** The representations h retain information about the input data, as they are directly produced by the encoder f_θ . Regularizing h ensures that the encoder learns to distribute information uniformly across the representation space, promoting diversity and preventing collapse.
- **Flexibility in Embedding Space:** The projection head g_ϕ is tasked with mapping h to z in a way that is conducive to the contrastive objective. By applying clustering objectives on z , we allow g_ϕ to focus on organizing the embeddings for better class separation without being constrained by the regularization applied to h .

¹Here, the superscript (i) refers to the sample index in the batch. It is distinct from augmentation indices; each augmented view is indexed separately as a different sample.

- **Computational Efficiency:** The dimensionality of h is typically lower (in the context of this chapter) than that of z , making the computation of the Sinkhorn loss more efficient when applied to h .

This overall design strategy, to regularize the intermediate representation, aligns with recent findings in contrastive learning literature [8], where decoupling the roles of different network components has been shown to enhance performance.

3.4.3 Implementation Details

Our training framework follows the SimCLR paradigm, where each input sample undergoes two distinct augmentations to create positive pairs. These augmented views are then processed through a ResNet-18 encoder, followed by a projection head, producing intermediate representations h, h' and projected embeddings z, z' .

The **NT-Xent loss** from (3.3) ensures contrastive alignment in the final embedding space, while the **Sinkhorn loss** (3.4) enforces a structured and well-dispersed representation space at the intermediate feature level.

During training, we initialize the model with random weights and update parameters using the Adam optimizer with a learning rate of 10^{-3} and weight decay of 10^{-6} . The encoder is trained for a total of 100 epochs on the MNIST dataset and 300 epochs for the rest of the datasets using a batch size of 512. Following pretraining, the encoder’s weights are frozen, and a linear classifier is trained on top of the learned representations for downstream evaluation.

The hyperparameters β and λ are tuned based on validation performance. The Sinkhorn distance is computed iteratively using 40 updates with the selected $\lambda = 0.05$ from validation to stabilize the optimal transport plan.

In practice, we compute the Sinkhorn loss using the *Sinkhorn–Knopp algorithm* [25], as detailed below:

1. **Cost Matrix Computation:** For a mini-batch of representations, compute the cost matrix

$$C_{ij} = \|z^{(i)} - z'^{(j)}\|^2.$$

2. **Initialization:** Initialize the coupling γ to an isotropic matrix (e.g., all entries equal to $1/N^2$).

3. **Iterative Scaling:** Update the coupling iteratively via

$$\gamma^{(t+1)} = \text{diag}(u^{(t)}) \gamma^{(t)} \text{diag}(v^{(t)}),$$

where the scaling vectors $u^{(t)}$ and $v^{(t)}$ are computed to enforce the row and column marginal constraints.

4. **Loss Computation:** Upon convergence, the stabilized transport plan is used to compute the final Sinkhorn loss.

3.4.4 Theoretical Justification

It is important to provide a justification for the use of Sinkhorn regularization in our framework. Recall that for two empirical distributions

$$P = \frac{1}{N} \sum_{i=1}^N \delta_{z^{(i)}}, \quad Q = \frac{1}{N} \sum_{i=1}^N \delta_{z'^{(i)}},$$

and a cost matrix with entries

$$C_{ij} = \|z^{(i)} - z'^{(j)}\|^2,$$

the entropy-regularized Wasserstein distance is defined as

$$W_\lambda(P, Q) = \min_{\gamma \in \Pi(P, Q)} \langle \gamma, C \rangle - \lambda H(\gamma), \quad (3.6)$$

where

$$H(\gamma) = - \sum_{i,j} \gamma_{ij} \log \gamma_{ij},$$

and $\Pi(P, Q)$ is the set of couplings with marginals P and Q .

Lemma 1 (Latent Space Dispersion). *Let P and Q be defined as above, and denote by γ^* the optimal coupling in (3.6). Then,*

$$W_\lambda(P, Q) \geq \frac{1}{N} \sum_{i=1}^N \|z^{(i)} - z'^{(i)}\|^2 - \lambda H(\gamma^*). \quad (3.7)$$

In particular, as $\lambda \rightarrow 0$, minimizing $W_\lambda(P, Q)$ forces the learned representations to align (by the “diagonal” cost) while still controlling dispersion in the latent space.

Proof. **1. Candidate Coupling:** Consider the coupling given by

$$\tilde{\gamma}_{ij} = \frac{1}{N} \delta_{ij},$$

where δ_{ij} is the Kronecker delta. This coupling satisfies the marginal constraints in $\Pi(P, Q)$.

2. Evaluation of the Objective: For $\tilde{\gamma}$, the cost term is

$$\langle \tilde{\gamma}, C \rangle = \frac{1}{N} \sum_{i=1}^N \|z^{(i)} - z'^{(i)}\|^2.$$

Its entropy is computed as

$$H(\tilde{\gamma}) = - \sum_{i=1}^N \frac{1}{N} \log \left(\frac{1}{N} \right) = \log N.$$

Thus, the regularized cost evaluated at $\tilde{\gamma}$ is

$$\langle \tilde{\gamma}, C \rangle - \lambda H(\tilde{\gamma}) = \frac{1}{N} \sum_{i=1}^N \|z^{(i)} - z'^{(i)}\|^2 - \lambda \log N.$$

3. Lower Bound via Optimality: Since γ^* minimizes the objective in (3.6), we have

$$\begin{aligned} W_\lambda(P, Q) &= \langle \gamma^*, C \rangle - \lambda H(\gamma^*) \\ &\leq \frac{1}{N} \sum_{i=1}^N \|z^{(i)} - z'^{(i)}\|^2 - \lambda \log N. \end{aligned}$$

However, note that for any $\gamma \in \Pi(P, Q)$, the cost term satisfies

$$\langle \gamma, C \rangle \geq \frac{1}{N} \sum_{i=1}^N \|z^{(i)} - z'^{(i)}\|^2,$$

since any deviation from the “diagonal” pairing increases the quadratic cost. In particular, for the optimal γ^* we obtain

$$\begin{aligned} W_\lambda(P, Q) &= \langle \gamma^*, C \rangle - \lambda H(\gamma^*) \\ &\geq \frac{1}{N} \sum_{i=1}^N \|z^{(i)} - z'^{(i)}\|^2 - \lambda H(\gamma^*). \end{aligned}$$

This establishes (3.7).

4. Interpretation: As $\lambda \rightarrow 0$, the penalty term $\lambda H(\gamma^*)$ becomes negligible, and the minimization of $W_\lambda(P, Q)$ forces the cost $\langle \gamma^*, C \rangle$ to approach the average diagonal cost. This structured alignment ensures dispersion in the latent space. $\square$

Lemma 2 (Prevention of Mode Collapse). *Assume that the contrastive loss $\mathcal{L}_{NT-Xent}$ is minimized, and let γ^* denote the optimal transport plan obtained via the Sinkhorn optimization. Then, for any $\lambda > 0$ (in particular, for sufficiently small λ), the strong convexity induced by the entropy term guarantees that*

$$\gamma_{ij}^* > 0 \quad \forall i, j. \quad (3.8)$$

Proof. **1. Optimality Conditions:** The inclusion of the entropy term in (3.6) makes the optimization problem strictly convex. Consequently, the optimal coupling γ^* is unique and is characterized by the Sinkhorn fixed-point equations.

2. Gibbs Form Representation: It is well established (see, e.g., [25]) that the solution γ^* can be written in Gibbs form as

$$\gamma_{ij}^* = \exp\left(\frac{f_i + g_j - C_{ij}}{\lambda}\right), \quad (3.9)$$

where f_i and g_j are dual potentials corresponding to the marginal constraints.

3. Strict Positivity: Since the exponential function is strictly positive for any finite argument, from (3.9) it follows that

$$\gamma_{ij}^* > 0 \quad \forall i, j.$$

Thus, the entropic regularization prevents any entry of the transport plan from collapsing to zero, thereby mitigating mode collapse. $\square$

3.5 Experiments

We evaluate **SinSim** on multiple benchmark datasets, including **MNIST**, **CIFAR-10**, **CIFAR-100**, and **STL-10**, to assess its effectiveness in self-supervised representation learning. Our approach is compared against **SimCLR** [18], **VICReg** [8], and **Barlow Twins** [123], which represent prominent contrastive and redundancy reduction methods.

Table 3.1: Linear Classification Accuracy (%)

Method	MNIST	CIFAR-10	CIFAR-100	STL-10
SimCLR	97.0	68.6	41.9	50.3
VICReg	96.1	72.4	45.5	52.4
Barlow Twins	95.4	72.1	46.1	51.3
SinSim (Ours)	98.4	70.7	43.2	52.6

Unlike these baselines, **SinSim** integrates Sinkhorn regularization, enforcing structured alignment in the latent space while preserving discriminative power. Through our experiments, we will demonstrate that **SinSim** not only enhances representation quality but also achieves comparable performance in both *linear and non-linear classification tasks*, benefiting particularly from the global structure imposed by optimal transport.

3.5.1 Experimental Setup

We perform experiments on the MNIST, CIFAR-10, CIFAR-100, and STL-10 datasets. All models use a ResNet-18 backbone with an additional projection head for the contrastive learning task. Following pretraining, the encoder is frozen and evaluated with both a linear classifier and a two-layer MLP.

3.5.2 Linear Classification Performance

Table 3.1 summarizes the linear classification accuracies across different self-supervised learning methods. Our proposed approach, **SinSim**, outperforms SimCLR on *every* dataset, and demonstrates competitive performance against VICReg and Barlow Twins, in particular with the proposed SinSim achieving the highest classification accuracy on both MNIST and STL-10.

While SinSim does not surpass VICReg on CIFAR-10, it still performs comparably, showing that Sinkhorn regularization does not degrade contrastive learning performance and provides an alternative to variance-covariance-based regularization strategies.

The improvement of SinSim over SimCLR, particularly on MNIST (+1.4%) and STL-10 (+2.3%), highlights the benefits of enforcing a more structured representation space via optimal transport. Moreover, while VICReg and Barlow Twins rely on statistical regularization (variance-covariance constraints) to encourage representation spread, Sinkhorn

Table 3.2: Non-Linear Classification Accuracy (%)

Method	MNIST	CIFAR-10	CIFAR-100	STL-10
SimCLR	97.7	70.8	42.1	52.6
VICReg	97.2	74.5	46.2	53.9
Barlow Twins	96.9	73.6	46.0	52.2
SinSim (Ours)	98.9	72.6	43.8	54.4

regularization explicitly models the geometry-aware distribution alignment, making SinSim a meaningful alternative in contrastive learning frameworks.

3.5.3 Non-Linear Classification Performance

We further assess the quality of the learned representations with a non-linear (MLP) classifier, shown in Table 3.2. With results very much consistent of those in Table 3.1, Table 3.2 demonstrates that the benefits of the SinSim strategy remain present with nonlinear classification, with SinSim still improving upon the baseline methods, suggesting that the global structure imposed by the Sinkhorn loss aids in learning more semantically meaningful features.

3.5.4 Representation Visualization

To further investigate the effect of Sinkhorn regularization on learned representations, we apply UMAP [77] to the embeddings extracted from different models. Figure 3.1 visualizes the latent space for MNIST and CIFAR-10, comparing SimCLR and SinSim.

Comparing the embeddings obtained using SimCLR (panels a, c) and SinSim (panels b, d), we observe:

- **Improved Class Separation:** SinSim produces more distinct clusters in MNIST and CIFAR-10, demonstrating that Sinkhorn regularization prevents mode collapse and encourages well-distributed representations.
- **Reduced Class Overlap:** In CIFAR-10, SimCLR embeddings (panel c) show overlap between different classes, whereas SinSim (panel d) results in more compact and well-separated clusters.

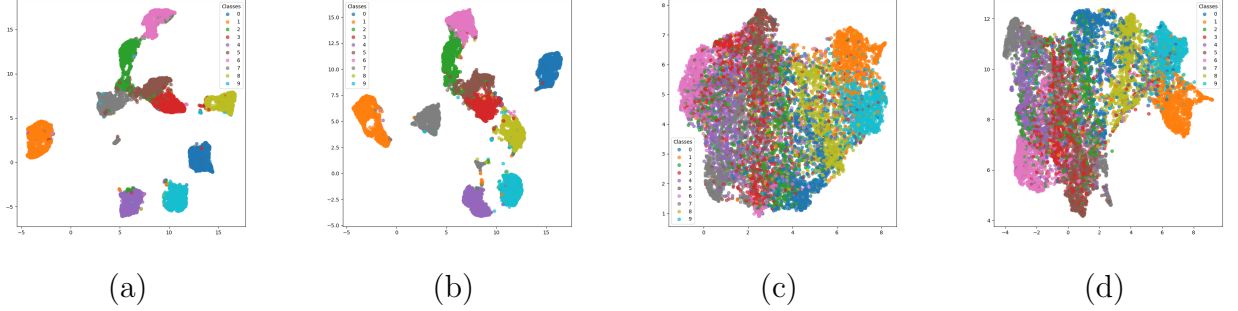


Figure 3.1: UMAP visualization of learned embeddings. (a) and (c) show the embedding space for SimCLR, while (b) and (d) illustrate SinSim’s feature representations. SinSim achieves better class separation and reduced overlap, suggesting that Sinkhorn regularization improves the structured alignment of representations.

These findings provide qualitative evidence that the Sinkhorn regularization enforces structured alignment in contrastive learning, leading to more robust, well-separated, and meaningful representations.

3.5.5 Ablation Study

Effect of Sinkhorn Regularization

To analyze the impact of Sinkhorn regularization on the representation learning process in contrastive learning, we conduct an ablation study by varying the strength β of the Sinkhorn regularization.

Figures 3.2 and 3.3 present the classification accuracy of a linear probe trained on the representations learned by SinSim as a function of β . Both SinSim and standard SimCLR (effectively SinSim at $\beta = 0$) are pretrained on the MNIST and CIFAR-10 datasets for 10 epochs, followed by the training of a linear classifier for 5 epochs.

The crux of our analysis is that the results unambiguously demonstrate an improvement in classification accuracy as β increases, suggesting that the presence (larger β) of Sinkhorn regularization enhances the feature space structure by promoting better alignment of representations. The classification accuracy improves significantly for moderate values of β , peaking around $\beta = 0.8$ for MNIST and $\beta = 0.9$ for CIFAR-10. This trend highlights the potential of optimal transport regularization in improving contrastive learning objectives by refining feature representation structure.

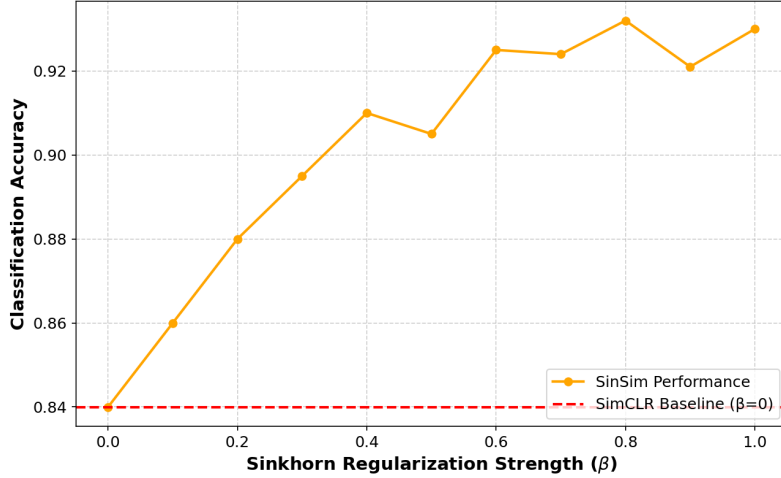


Figure 3.2: Effect of Sinkhorn Regularization on SinSim Performance, varying the Sinkhorn regularization strength (β) on classification accuracy when training on MNIST for **10 epochs**. The **solid orange curve** represents the classification accuracy achieved by the SinSim model as a function of β , while the **dashed red line** corresponds to the baseline performance of standard SimCLR (equivalent to SinSim at $\beta = 0$). Very clearly, incorporating Sinkhorn regularization consistently improves feature representations, leading to enhanced classification accuracy. This preliminary experiment primarily serves to capture the trend of Sinkhorn’s influence on contrastive learning rather than as an absolute performance benchmark.

Effect of Sinkhorn Iterations

The number of Sinkhorn iterations controls the convergence of the optimal transport plan. A small number of iterations may lead to an under-optimized transport plan, while too many iterations may introduce excessive regularization.

Figure 3.4 illustrates the classification accuracy across different values of iteration for MNIST, while Figure 3.5 shows the same effect on CIFAR-10. The results show that increasing the number of iterations improves performance up to an optimal value (around 40), after which performance slightly degrades. This trend suggests that while Sinkhorn optimization is beneficial, excessive iterations may lead to over-smoothing, potentially reducing the discriminative power of learned representations.

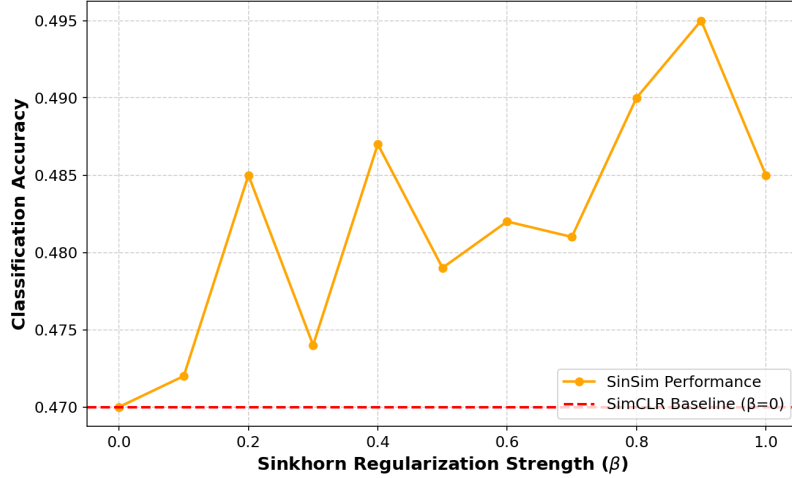


Figure 3.3: As in Figure 3.2, but here based on CIFAR-10 data. Unlike the consistent improvement observed in Figure 3.2 with MNIST, here the results based on CIFAR-10 exhibit significant fluctuations with β , with a more modest trend. Nevertheless, for all values of β SinSim outperforms the baseline, suggesting that Sinkhorn regularization definitely contributes to enhanced feature representations. This experiment primarily serves to capture the trend of Sinkhorn’s influence on contrastive learning in a more complex dataset.

Effect of Regularization Strength λ

The hyperparameter λ in the Sinkhorn regularization term controls the trade-off between entropy regularization and transport cost. A low λ results in minimal regularization, while a high λ may overly smooth the feature space, leading to a loss of discriminative power.

Figure 3.6 presents the classification accuracy as a function of λ for MNIST, and Figure 3.7 for CIFAR-10. The results for both datasets give consistent results, with a moderate value of λ (around $\lambda = 0.06$) achieving the best performance. For very small values, the impact of Sinkhorn regularization is too limited, while for large values ($\lambda > 0.09$), performance sharply drops, confirming that excessive smoothing negatively affects representation quality.

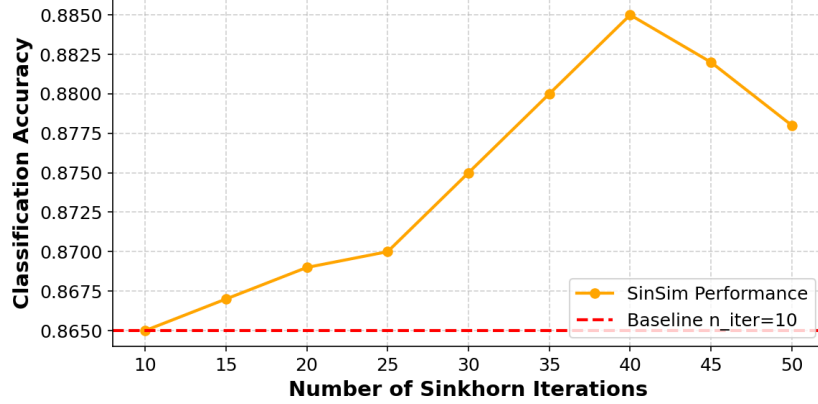


Figure 3.4: Effect of Sinkhorn iterations on SinSim classification accuracy on MNIST. Increasing the number of iterations improves accuracy up to 40, beyond which performance slightly decreases. The red dashed line represents a baseline SinSim performance at a default iteration count of 10.

3.5.6 Implementation Details

Datasets & augmentation. We use **MNIST**, **CIFAR-10**, **CIFAR-100**, and **STL-10**. Unless noted, images are transformed with a SimCLR-style pipeline:

- RandomResizedCrop(32, [0.2, 1.0]),
- RandomHorizontalFlip,
- RandomApply
- ColorJitter(0.4, 0.4, 0.4, 0.1), $p=0.8$),
- RandomGrayscale($p=0.2$),
- Channel-wise normalization to mean 0.5, std 0.5.

For the classifier phase, we use only ToTensor + normalization.

For MNIST, we adapt the first conv to single-channel input; for color datasets we keep three channels (the implementation uses a grayscale-to-3ch conversion to keep the ResNet stem unchanged).

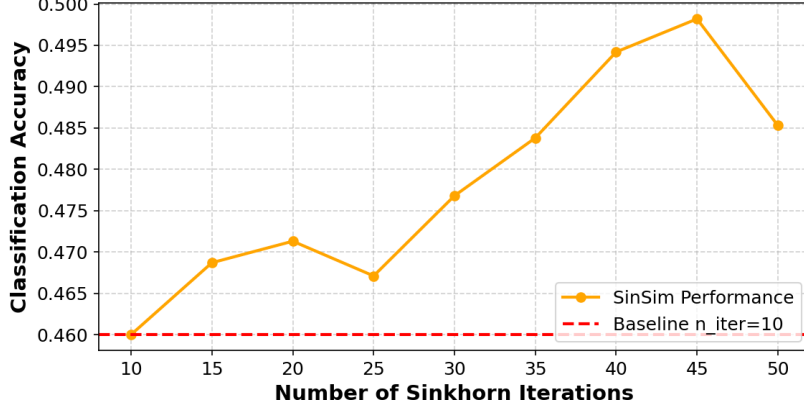


Figure 3.5: Effect of Sinkhorn iterations, as in Figure 3.4, but now assessed on CIFAR-10. The overall trend is very similar to that of MNIST, with slightly increased variability, but consistent overall conclusion.

Backbone and projection head. The encoder f_θ is a ResNet-18 with the final FC removed (`encoder.fc = Identity`), producing $h \in \mathbb{R}^{512}$. The projection head g_ϕ is a 2-layer MLP,

$$g_\phi(h) = \text{BN}(W_2 \text{ReLU}(\text{BN}(W_1 h))),$$

with $W_1 \in \mathbb{R}^{512 \times 512}$, $W_2 \in \mathbb{R}^{512 \times d}$, $d=128$, BatchNorm on both layers (the last BN is `affine=False`). We denote $z = g_\phi(h)$.

Objective. Pretraining minimizes

$$\mathcal{L} = \underbrace{\mathcal{L}_{\text{NT-Xent}}(z, z')}_{\text{contrastive, } \tau=0.5} + \beta \underbrace{\mathcal{L}_{\text{Sinkhorn}}(h, h')}_{\text{OT regularizer}},$$

where β corresponds in code to `lambda_sinkhorn` (default 0.8). We compute the contrastive term with cosine similarities and temperature $\tau=0.5$.

Entropic OT (Sinkhorn) implementation. Given a mini-batch of $\{h^{(i)}\}_{i=1}^N$ and $\{h'^{(j)}\}_{j=1}^N$, we form the quadratic cost matrix $C_{ij} = \|h^{(i)} - h'^{(j)}\|_2^2$, normalized by $\max C$ for numerical stability. We solve the entropically regularized OT with dual log-sum-exp updates (Sinkhorn-Knopp):

$$u \leftarrow -\lambda \log \sum_j \exp\left(\frac{-C_{ij}}{\lambda} + v_j\right), \quad v \leftarrow -\lambda \log \sum_i \exp\left(\frac{-C_{ij}}{\lambda} + u_i\right),$$

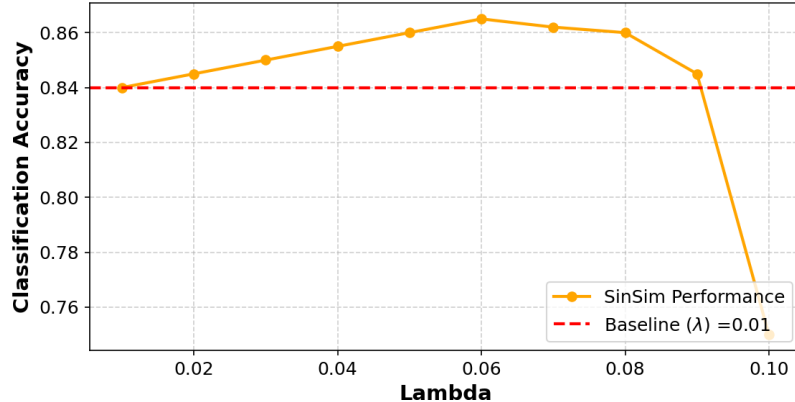


Figure 3.6: Effect of Sinkhorn regularization strength λ on SinSim classification accuracy on MNIST. Moderate values ($\lambda = 0.06$) yield the best performance, while overly large values reduce discriminative power. The red dashed line represents a default baseline at $\lambda = 0.01$.

where λ is the entropy parameter (code: `epsilon`, default 0.05). We center u, v each iteration to reduce drift and stop early if $\|u - u_{\text{prev}}\|_2 < 10^{-3}$ and $\|v - v_{\text{prev}}\|_2 < 10^{-3}$ or after at most 50 iterations. The transport plan has the usual Gibbs form $\gamma_{ij} \propto \exp((u_i + v_j - C_{ij})/\lambda)$. The reported Sinkhorn cost is the mini-batch average of $\langle \gamma, C \rangle$. (Complexity per batch is $O(N^2d)$ to build C and $O(TN^2)$ for T Sinkhorn iterations.)

Optimization and schedule. We use Adam with learning rate 10^{-3} and weight decay 10^{-6} , batch size 512, and train **100 epochs on MNIST** and **300 epochs on CIFAR-10/100 and STL-10**, unless stated otherwise. Weights are initialized randomly. After pretraining, we *freeze* the encoder f_θ and train a linear classifier ($512 \rightarrow \text{\#classes}$) for 10 epochs with Adam (10^{-3} , 10^{-6} weight decay). Before probing, we reset BatchNorm running statistics of the frozen encoder to avoid stale estimates carried over from contrastive augs.

Evaluation protocol. We report top-1 accuracy with a linear probe on frozen features. For non-linear evaluation, we additionally use a 2-layer MLP head (results reported in Sec. 3.5); training hyperparameters mirror the linear probe unless specified. For representation visualizations, we apply UMAP [77] on the frozen h -features (default parameters) and plot 2D embeddings.

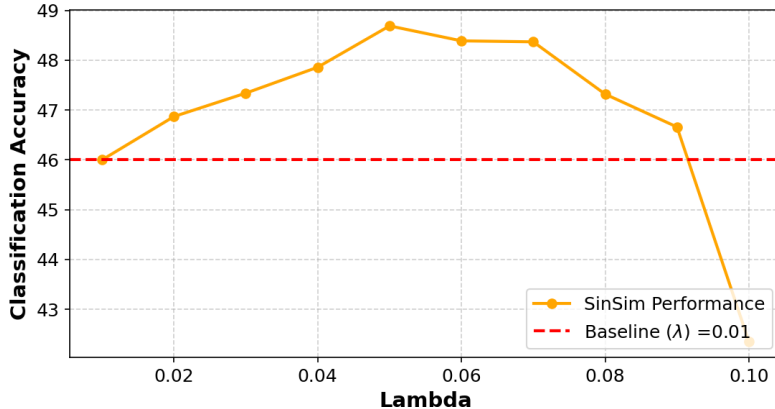


Figure 3.7: Effect of Sinkhorn regularization strength λ , as in Figure 3.6, but here tested on CIFAR-10, with results and overall conclusion similar to those found for MNIST.

3.6 Conclusion

We have introduced SinSim, a novel self-supervised learning framework that integrates Sinkhorn regularization into the SimCLR paradigm. By leveraging optimal transport theory, SinSim enforces a structured latent space while preserving the discriminative power of contrastive learning. Our results demonstrate that SinSim not only enhances representation structure but also improves classification performance, particularly in settings where contrastive learning alone struggles to maintain global alignment.

Through both theoretical insights and empirical evaluations, we have shown that Sinkhorn-based contrastive learning achieves superior representation quality, as measured by both linear and non-linear evaluation protocols. Moreover, our UMAP visualizations reveal that Sinkhorn regularization promotes a well-dispersed and semantically meaningful feature space, reducing representation collapse and improving class separation.

Future work will explore the extension of Sinkhorn regularization to other self-supervised methods and investigate its impact on larger-scale datasets and more complex architectures. A promising direction is examining its effectiveness in multimodal learning settings, such as vision-language models, where structured feature alignment is critical. Additionally, the structured feature representations encouraged by SinSim can benefit real-world applications such as medical imaging and robotics, where robust and geometry-aware representations are essential.

Chapter 4

Self-Supervised Learning Beyond Linear Correlation

4.1 Motivation

Self-supervised learning (SSL) [103, 73, 5] has fundamentally transformed the landscape of machine learning by enabling models to learn powerful and transferable representations from unlabeled data. This paradigm has proven especially valuable in domains where labeled data is scarce or expensive to obtain. Classic SSL frameworks, such as SimCLR [18], BYOL [46], and Barlow Twins [123], leverage data augmentation and carefully designed objective functions to promote invariance and reduce redundancy in the learned representations. Typically, these methods focus on linear statistical measures such as feature variance, covariance, and cross-correlation. While effective, this focus on linear dependencies overlooks the rich, nonlinear relationships that exist in high-dimensional data.

Many natural datasets are characterized by complex, nonlinear structures. When SSL models only account for linear dependencies, they may fail to capture the full spectrum of information present in the data, potentially leading to suboptimal representations that lack robustness and expressiveness for downstream tasks. Furthermore, existing methods tend to emphasize feature-wise statistics, with comparatively less attention given to the relationships among samples themselves. This imbalance can limit the model’s ability to distinguish between similar and dissimilar instances, especially in settings with subtle or highly structured intra-class and inter-class variations.

Recent advances have highlighted the value of moving beyond linear metrics, with approaches such as kernel-based SSL [68, 115] and nonlinear dependence measures beginning

to emerge. In particular, the Hilbert-Schmidt Independence Criterion (HSIC) [45] has proven to be a robust tool for quantifying nonlinear dependencies within a Reproducing Kernel Hilbert Space (RKHS) [105, 39], offering a principled way to capture complex interactions between features and between samples. However, a comprehensive SSL framework that unifies both linear and nonlinear dependencies, at both the feature and sample level, has remained absent from the literature.

This gap motivates the present work. By systematically addressing both linear correlations and nonlinear dependencies—and doing so across both feature and sample dimensions—it becomes possible to design self-supervised objectives that foster richer, more disentangled, and more transferable representations. Such a holistic approach can lead to improved generalization, robustness to spurious correlations, and better performance on a wide range of downstream tasks. The proposed Correlation-Dependence Self-Supervised Learning (CDSSL) framework is developed precisely to meet this need, providing a unified and extensible foundation for modern SSL.

Building on the theoretical foundations of correlation and dependence, we now present the network architecture and data preparation strategy employed in our proposed method. Consider a d -dimensional dataset $\{x_i\}_{i=1}^M \subset \mathbb{R}^d$. Using self-supervised learning (SSL), our objective is to learn an embedding space where similar instances are positioned close together, and dissimilar instances are placed far apart. This is achieved without labels, relying on data augmentation to generate similar instances.

4.2 Core Idea

The core idea underlying this work is that effective self-supervised representation learning should address both *linear* and *nonlinear* dependencies, not only among features, but also across samples within a dataset. Traditional approaches, such as SimCLR [18], Barlow Twins [123], and VICReg [8], focus primarily on minimizing linear correlations or maximizing variance to prevent collapse in the feature space. However, these approaches are inherently limited to linear relationships and often overlook the rich, complex dependencies present in high-dimensional data.

To address this limitation, we propose the Correlation-Dependence Self-Supervised Learning (CDSSL) framework, which explicitly unifies and extends existing methods by integrating both linear correlations and nonlinear dependencies into a single, comprehensive objective. CDSSL systematically categorizes dependencies into four groups—auto-correlation/dependence and cross-correlation/dependence, each considered at the sample and feature levels. By incorporating the Hilbert-Schmidt Independence Criterion

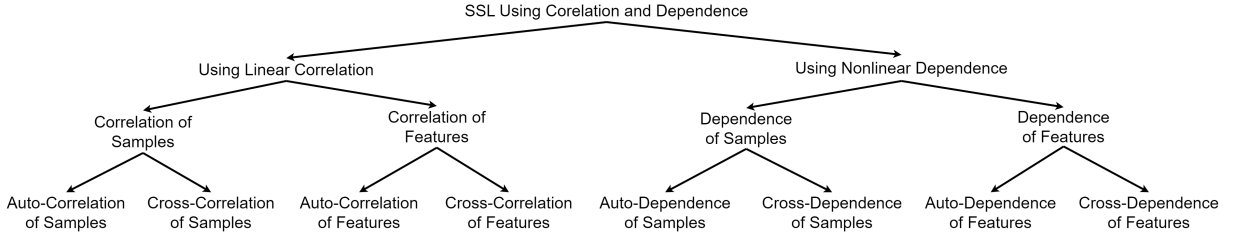


Figure 4.1: Overview of the CDSSL framework. This figure highlights the novel categorization of dependencies into linear correlation and nonlinear dependence, further divided into sample-wise and feature-wise interactions. By addressing auto- and cross-dependence at both levels, CDSSL unifies existing SSL methods while introducing robust measures, such as the HSIC, to enhance the quality of learned representations. This comprehensive framework bridges gaps in current methods, ensuring diversity and disentanglement in feature learning.

(HSIC) [45, 43, 39] as a measure of nonlinear dependence within a Reproducing Kernel Hilbert Space (RKHS), CDSSL is able to capture subtle, higher-order relationships that go beyond the reach of conventional linear metrics.

This unified treatment leads to eight loss components that encourage diversity, disentanglement, and invariance in the learned representations. By jointly minimizing linear correlations and nonlinear dependencies, CDSSL not only generalizes and subsumes many existing methods as special cases, but also achieves superior performance on a variety of benchmarks. This approach ultimately leads to richer and more transferable representations, as demonstrated by improved results on both in-domain and domain adaptation tasks.

4.3 Overview and Intuition

The motivation behind the Correlation-Dependence Self-Supervised Learning (CDSSL) framework arises from the observation that many contemporary self-supervised learning algorithms target only a subset of the complex relationships present in high-dimensional representations. While variance maximization and linear correlation minimization have become standard for mitigating feature redundancy and collapse, such approaches fail to address the more intricate, nonlinear dependencies that can persist both among features and among samples.

To overcome these limitations, CDSSL adopts a unified perspective: it seeks to disentangle and diversify representations by simultaneously regulating both linear and nonlinear dependencies, considering their effect across two axes—samples and features—and across two roles—auto (within the same view) and cross (between different augmentations). This framework is depicted schematically in Figure 4.1, which categorizes dependencies into a comprehensive taxonomy. Each branch in the figure corresponds to a distinct loss component within CDSSL, ensuring that all relevant types of redundancy and dependence are explicitly controlled during training.

Central to this methodology is the use of the Hilbert-Schmidt Independence Criterion (HSIC), which extends the notion of dependence beyond linear correlation and is capable of capturing higher-order relationships via kernel methods. By designing the loss function to incorporate both correlation and HSIC-based dependence terms, CDSSL provides a principled mechanism for learning representations that are simultaneously expressive, robust, and well-structured.

This chapter details the formulation of the CDSSL objective, its loss components, and the network architecture used to realize these goals.

4.4 The Neural Network Structure

In SSL, it is common to use an encoder neural network—of any backbone structure—followed by an expander network. The encoder, denoted by f_θ , provides the embedding; the expander, denoted by g_ϕ , expands the dimensionality. θ and ϕ are the learnable weights of the encoder and the expander, respectively. Let the q -dimensional output embedding of the encoder, referred to as the latent embedding, be denoted by $h \in \mathbb{R}^q$. We denote the p -dimensional output embedding of the expander by $z \in \mathbb{R}^p$ where usually $p \gg q$. The mapping of the encoder and the expander are

$$f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^q, \quad f_\theta : \mathbf{x} \mapsto h := f_\theta(\mathbf{x}), \quad (4.1)$$

$$g_\phi : \mathbb{R}^q \rightarrow \mathbb{R}^p, \quad g_\phi : h \mapsto z := f_\phi(h). \quad (4.2)$$

Here $z^{(i)} \in \mathbb{R}^p$ denotes the i -th output embedding in the batch where $i \in \{1, \dots, b\}$. Moreover, $z^{[i]} \in \mathbb{R}^b$ denotes the i -th embedding dimension (or feature) across all samples of batch where $i \in \{1, \dots, p\}$. We denote the j -th feature of $z^{(i)}$ by $z^{(i),[j]} \in \mathbb{R}$.

Corresponding Samples

Every batch in SSL contains two sets of data instances, $\{\mathbf{x}^{(i)}\}_{i=1}^b$ and $\{\mathbf{x}'^{(i)}\}_{i=1}^b$, where b is the mini-batch size and the corresponding instances in the two sets are two different augmentations from the same data instance in the dataset. Corresponding instances $\mathbf{x}^{(i)}$ and $\mathbf{x}'^{(i)}$ have similar patterns or belong to the same class without use of class labels since they are generated from same data input however with different augmentations. Consequently, in each mini-batch, there are pairs of latent embeddings $\{h^{(i)}\}_{i=1}^b$ and $\{h'^{(i)}\}_{i=1}^b$ and pairs of output embeddings $\{z^{(i)}\}_{i=1}^b$ and $\{z'^{(i)}\}_{i=1}^b$.

Non-Corresponding Samples

Assuming that the number of classes is large compared to the batch size, the probability of having two instances from the same class in the batch is small. The probability of having at least k samples from the same class in a batch of size b is

$$\mathbb{P} = \sum_{i=k}^b \binom{b}{i} \left(\frac{1}{c}\right)^i \left(1 - \frac{1}{c}\right)^{b-i}, \quad (4.3)$$

where c is the total number of classes. This probability is significantly low for datasets with many classes, such as ImageNet [28]. Therefore, when the number of classes is large compared to the batch size, the non-corresponding instances $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$ have different patterns or belong to different classes with high probability. As a result, the embeddings $z^{(i)}$ and $z^{(j)}$ should be different for $i \neq j$.

This assumption has been widely made in many contrastive learning approaches including SSL [81, 67] and few-shot learning [58]. In cases where the batch size is large or the number of classes is small—violating the assumption—the coefficients of the proposed loss terms involving non-corresponding embeddings can be made small. Alternatively, the patterns of samples in the batch can be compared, and the most distinct samples selected as non-corresponding samples.

4.5 Linear Correlation Loss Terms

This section focuses on the linear formulation of the proposed approach, detailing its structure and methodology.

4.5.1 Auto-Correlation of Samples

The correlation between the embeddings $z^{(i)}$ and $z^{(j)}$, $i \neq j$, should be minimized because $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$ have different patterns. Likewise, the correlation of $z'^{(i)}$ and $z'^{(j)}$, $i \neq j$, should be minimized. Therefore, the following *auto-correlation loss between samples* should be minimized:

$$\mathcal{L}_{\text{acs}} := \sum_{i=1}^b \sum_{j=1, j \neq i}^b c_{ij}^2 + \sum_{i=1}^b \sum_{j=1, j \neq i}^b c_{ij}''^2, \quad (4.4)$$

where c_{ij} and c_{ij}'' denote the linear correlation between embeddings $z^{(i)}$ and $z^{(j)}$ and also $z'^{(i)}$ and $z'^{(j)}$, respectively,

$$\mathbb{R} \ni c_{ij} := \frac{\sum_{k=1}^p \tilde{z}^{(i),[k]} \tilde{z}^{(j),[k]}}{\sqrt{\sum_{k=1}^p (\tilde{z}^{(i),[k]})^2} \sqrt{\sum_{k=1}^p (\tilde{z}^{(j),[k]})^2}}, \quad (4.5)$$

$$\mathbb{R} \ni c_{ij}'' := \frac{\sum_{k=1}^p \tilde{z}'^{(i),[k]} \tilde{z}'^{(j),[k]}}{\sqrt{\sum_{k=1}^p (\tilde{z}'^{(i),[k]})^2} \sqrt{\sum_{k=1}^p (\tilde{z}'^{(j),[k]})^2}}, \quad (4.6)$$

based on the centered embedding features:

$$\tilde{z}^{[k]} := z^{[k]} - \frac{1}{p} \sum_{\ell=1}^p z^{[\ell]}, \quad \tilde{z}'^{[k]} := z'^{[k]} - \frac{1}{p} \sum_{\ell=1}^p z'^{[\ell]}. \quad (4.7)$$

Note that c_{ij} is the (i, j) -th element of the $(b \times b)$ sample correlation matrix between samples of the batch $\{z^{(i)}\}_{i=1}^b$. Likewise, c_{ij}'' is the (i, j) -th element of the $(b \times b)$ correlation matrix over the batch $\{z'^{(i)}\}_{i=1}^b$.

The first term in Eq. (4.4) minimizes the cross-correlation of the different embeddings in the batch $\{z^{(i)}\}_{i=1}^b$ to its lowest value, i.e., zero, to uncorrelate them. The second term minimizes the cross-correlation of the different embeddings in the batch $\{z'^{(i)}\}_{i=1}^b$. Note that $c_{ii} = c_{ii}'' = 1$ always holds, hence Eq. (4.4) does not contain these terms.

4.5.2 Cross-Correlation of Samples

On the one hand, as the corresponding instances $\mathbf{x}^{(i)}$ and $\mathbf{x}'^{(i)}$ have similar patterns, the correlation of their output embeddings $z^{(i)}$ and $z'^{(i)}$ should be maximized. On the other hand, the cross-correlation between the non-corresponding embeddings $z^{(i)}$ and $z'^{(j)}$, for

$i \neq j$, should be minimized. Therefore, the following *cross-correlation loss between samples* should be minimized:

$$\mathcal{L}_{\text{ccs}} := \sum_{i=1}^b (1 - c'_{ii})^2 + \eta_{\text{ccs}} \sum_{i=1}^b \sum_{j=1, j \neq i}^b c'^2_{ij}, \quad (4.8)$$

where $\eta_{\text{ccs}} > 0$ is the regularization parameter and c'_{ij} denotes the linear correlation between embeddings $z^{(i)}$ and $z'^{(j)}$:

$$\mathbb{R} \ni c'_{ij} := \frac{\sum_{k=1}^p \tilde{z}^{(i),[k]} \tilde{z}'^{(j),[k]}}{\sqrt{\sum_{k=1}^p (\tilde{z}^{(i),[k]})^2} \sqrt{\sum_{k=1}^p (\tilde{z}'^{(j),[k]})^2}}, \quad (4.9)$$

in which $\tilde{z}^{(i),[k]}$ and $\tilde{z}'^{(j),[k]}$ are the i -th element and the j -th element of the $\tilde{z}^{[k]}$ and $\tilde{z}'^{[k]}$, respectively, and $\tilde{z}^{[k]} \in \mathbb{R}^b$ and $\tilde{z}'^{[k]} \in \mathbb{R}^b$ are the centered embedding features defined in Eq. (4.7). The c'_{ij} is the (i, j) -th element of the $(b \times b)$ sample correlation matrix between sample of the batches $\{z^{(i)}\}_{i=1}^b$ and $\{z'^{(i)}\}_{i=1}^b$.

The first term in Eq. (4.8) maximizes the cross-correlation of the corresponding embeddings to its highest value, i.e., one. Its second term minimizes the cross-correlation between the non-corresponding embeddings to its lowest value, i.e., zero, to uncorrelate them.

4.5.3 Auto-Correlation of Features

The correlation between the embedding features $z^{[i]}$ and $z^{[j]}$, for $i \neq j$, should be minimized to have low redundancy between the learned features. Likewise, the correlation of $z'^{[i]}$ and $z'^{[j]}$, for $i \neq j$, should be minimized. This decorrelates different features, within the batch, and it avoids collapse of the embeddings into zero vectors. Therefore, the following *auto-correlation loss between features* should be minimized:

$$\mathcal{L}_{\text{acf}} := \sum_{i=1}^b \sum_{j=1, j \neq i}^b c^2_{[ij]} + \sum_{i=1}^b \sum_{j=1, j \neq i}^b c''^2_{[ij]}, \quad (4.10)$$

where $c_{[ij]}$ and $c'_{[ij]}$ denote the linear correlation between embedding features $z^{[i]}$ and $z^{[j]}$ and the linear correlation between embedding features $z'^{[i]}$ and $z'^{[j]}$, respectively:

$$\mathbb{R} \ni c_{[ij]} := \frac{\sum_{k=1}^b \widehat{z}^{(k),[i]} \widehat{z}^{(k),[j]}}{\sqrt{\sum_{k=1}^b (\widehat{z}^{(k),[i]})^2} \sqrt{\sum_{k=1}^b (\widehat{z}^{(k),[j]})^2}}, \quad (4.11)$$

$$\mathbb{R} \ni c''_{[ij]} := \frac{\sum_{k=1}^b \widehat{z}'^{(k),[i]} \widehat{z}'^{(k),[j]}}{\sqrt{\sum_{k=1}^b (\widehat{z}'^{(k),[i]})^2} \sqrt{\sum_{k=1}^b (\widehat{z}'^{(k),[j]})^2}}, \quad (4.12)$$

in which $\widehat{z}^{(k)} \in \mathbb{R}^p$ and $\widehat{z}'^{(k)} \in \mathbb{R}^p$ are the centered embeddings:

$$\widehat{z}^{(k)} := z^{(k)} - \frac{1}{b} \sum_{\ell=1}^b z^{(\ell)}, \quad \widehat{z}'^{(k)} := z'^{(k)} - \frac{1}{b} \sum_{\ell=1}^b z'^{(\ell)}, \quad (4.13)$$

and $\widehat{z}^{(k),[i]}$ and $\widehat{z}^{(k),[j]}$ are the i -th element and the j -th element of the $\widehat{z}^{(k)}$ and $\widehat{z}'^{(k)}$, respectively. The $c_{[ij]}$ is the (i, j) -th element of the $(p \times p)$ feature correlation matrix between features in the batch $\{z^{(k)}\}_{k=1}^b$. Likewise, $c''_{[ij]}$ is the (i, j) -th element of the $(p \times p)$ correlation matrix over the batch $\{z'^{(k)}\}_{k=1}^b$.

The first and second terms in Eq. (4.10) minimizes the cross-correlation of the different embedding features in the batch $\{z^{(k)}\}_{k=1}^b$ and batch $\{z'^{(i)}\}_{i=1}^b$, respectively. $c_{[ii]} = c''_{[ii]} = 1$ always holds, hence Eq. (4.10) does not contain these terms.

4.5.4 Cross-Correlation of Features

The corresponding embedding features $z^{[i]}$ and $z'^{[i]}$ should be similar to have the same embedding features for instances with similar patterns. Therefore, the correlation between them should be maximized. Additionally, to minimize redundancy between the learned features across augmentations, the correlation between the embedding features $z^{[i]}$ and $z'^{[j]}$, for $i \neq j$, should be minimized. This decorrelates different features, across augmentations, and it avoids collapse of the embeddings into zero vectors. Overall, the following *cross-correlation loss between features* should be minimized:

$$\mathcal{L}_{\text{ccf}} := \sum_{i=1}^b (1 - c'_{[ii]})^2 + \eta_{\text{ccf}} \sum_{i=1}^b \sum_{j=1, j \neq i}^b c'^2_{[ij]}, \quad (4.14)$$

where $\eta_{\text{ccf}} > 0$ is the regularization parameter and $c'_{[ij]}$ denotes the linear correlation between embedding features $z^{[i]}$ and $z'^{[j]}$:

$$\mathbb{R} \ni c'_{[ij]} := \frac{\sum_{k=1}^b \widehat{z}^{(k),[i]} \widehat{z}'^{(k),[j]}}{\sqrt{\sum_{k=1}^b (\widehat{z}^{(k),[i]})^2} \sqrt{\sum_{k=1}^b (\widehat{z}'^{(k),[j]})^2}}, \quad (4.15)$$

in which $\widehat{z}^{(k),[i]}$ and $\widehat{z}'^{(k),[j]}$ are the i -th element and the j -th element of the $\widehat{z}^{(k)}$ and $\widehat{z}'^{(k)}$, respectively, and $\widehat{z}^{(k)} \in \mathbb{R}^p$ and $\widehat{z}'^{(k)} \in \mathbb{R}^p$ are the centered embedding features defined in Eq. (4.13). The $c'_{[ij]}$ is the (i, j) -th element of the $(p \times p)$ feature correlation matrix between features in the batches $\{z^{(k)}\}_{k=1}^b$ and $\{z'^{(k)}\}_{k=1}^b$.

The first term in Eq. (4.14) maximizes the cross-correlation of the corresponding embeddings to its highest value, i.e., one, and the second term minimizes the cross-correlation between the non-corresponding embeddings to its lowest value, i.e., zero, to uncorrelate them.

4.6 Nonlinear Dependence Loss Terms

This section explores the non-linear formulation of the proposed approach. In contrast to linear correlation, nonlinear dependence leverages kernel methods, most notably the Hilbert-Schmidt Independence Criterion (HSIC), to capture higher-order interactions and complex structures in the embedding space that would otherwise remain hidden.

4.6.1 Auto-Dependence of Samples

The samples in a batch, $\{z^{(i)}\}_{i=1}^b$ or $\{z'^{(i)}\}_{i=1}^b$ should exhibit sufficient richness to represent the structure of the data. In other words, assuming that the instances in the batch belong to different classes, the variance of embeddings in the feature space should be maximized. This can be measured by self-HSIC ($\text{HSIC}(X, X)$) which is also a measure of dependence. The following *auto-dependence loss between samples* should be minimized:

$$\begin{aligned} \mathcal{L}_{\text{ads}} &:= \\ &- \text{HSIC}(\{z^{(i)}\}_{i=1}^b, \{z^{(i)}\}_{i=1}^b) - \text{HSIC}(\{z'^{(i)}\}_{i=1}^b, \{z'^{(i)}\}_{i=1}^b) \\ &= -\text{tr}(\mathbf{K}_{z^{(i)}} \mathbf{H}_b \mathbf{K}_{z^{(i)}} \mathbf{H}_b) - \text{tr}(\mathbf{K}_{z'^{(i)}} \mathbf{H}_b \mathbf{K}_{z'^{(i)}} \mathbf{H}_b), \end{aligned} \quad (4.16)$$

where $\mathbf{H}_b$ is the $(b \times b)$ centering matrix, and $\mathbf{K}_{z^{(i)}} \in \mathbb{R}^{b \times b}$ and $\mathbf{K}_{z'^{(i)}} \in \mathbb{R}^{b \times b}$ are the kernel matrices over $\{z^{(i)}\}_{i=1}^b$ and $\{z'^{(i)}\}_{i=1}^b$, respectively. Minimizing this loss function maximizes the self-HSIC of data and enhances the richness of data.

4.6.2 Cross-Dependence of Samples

The dependence between the embeddings $\{z^{(i)}\}_{i=1}^b$ and $\{z'^{(i)}\}_{i=1}^b$ should be maximized, as $\{\mathbf{x}^{(i)}\}_{i=1}^b$ and $\{\mathbf{x}'^{(i)}\}_{i=1}^b$ share corresponding patterns due to augmentations. This ensures invariance to augmentation. The following *cross-dependence loss between samples* should be minimized:

$$\begin{aligned}\mathcal{L}_{\text{cds}} &:= -\text{HSIC}(\{z^{(i)}\}_{i=1}^b, \{z'^{(i)}\}_{i=1}^b) \\ &= -\text{tr}(\mathbf{K}_{z^{(i)}} \mathbf{H}_b \mathbf{K}_{z'^{(i)}} \mathbf{H}_b).\end{aligned}\tag{4.17}$$

Minimizing this loss function maximizes the dependence of corresponding instances across augmentations.

4.6.3 Auto-Dependence of Features

The embedding features $\{z^{[i]}\}_{i=1}^p$ or $\{z'^{[i]}\}_{i=1}^p$ should also exhibit sufficient richness to represent the structure of the data. To ensure diverse embedding features and prevent information collapse, the covariance of the embedding features should be maximized in the feature space, a property that can be quantified using self-HSIC. The following *auto-dependence loss between features* should be minimized:

$$\begin{aligned}\mathcal{L}_{\text{adf}} &:= -\text{HSIC}(\{z^{[i]}\}_{i=1}^p, \{z^{[i]}\}_{i=1}^p) \\ &\quad - \text{HSIC}(\{z'^{[i]}\}_{i=1}^p, \{z'^{[i]}\}_{i=1}^p) \\ &= -\text{tr}(\mathbf{K}_{z^{[i]}} \mathbf{H}_p \mathbf{K}_{z^{[i]}} \mathbf{H}_p) - \text{tr}(\mathbf{K}_{z'^{[i]}} \mathbf{H}_p \mathbf{K}_{z'^{[i]}} \mathbf{H}_p),\end{aligned}\tag{4.18}$$

where $\mathbf{H}_p$ is the $(p \times p)$ centering matrix and $\mathbf{K}_{z^{[i]}} \in \mathbb{R}^{p \times p}$ and $\mathbf{K}_{z'^{[i]}} \in \mathbb{R}^{p \times p}$ are the kernel matrices over $\{z^{[i]}\}_{i=1}^p$ or $\{z'^{[i]}\}_{i=1}^p$, respectively. Minimizing this loss function encourages the embedding features to capture diverse information, thereby preventing information collapse.

4.6.4 Cross-Dependence of Features

On the one hand, the dependence between the embedding features $\{z^{[i]}\}_{i=1}^p$ and $\{z'^{[i]}\}_{i=1}^p$ should be maximized to ensure invariant embedding features against augmentations. On the other hand, this dependence can be minimized if the features in one set are shuffled, promoting independence in such cases. The random permutations of features in one of them makes the structure of various features different from one another in a series of iterations.

This reduces the redundancy of learned embedding features. Overall, the following *cross-dependence loss between features* should be minimized:

$$\begin{aligned}\mathcal{L}_{\text{cdf}} &:= -\text{HSIC}(\{z^{[i]}\}_{i=1}^p, \{z'^{[i]}\}_{i=1}^p) + \text{HSIC}(\{z^{[i]}\}_{i=1}^p, \{z'^{\phi([i])}\}_{i=1}^p) \\ &= -\text{tr}(\mathbf{K}_{z^{[i]}} \mathbf{H}_p \mathbf{K}_{z'^{[i]}} \mathbf{H}_p) + \text{tr}(\mathbf{K}_{z'^{\phi([i])}} \mathbf{H}_p \mathbf{K}_{z'^{[i]}} \mathbf{H}_p),\end{aligned}\tag{4.19}$$

where $\phi([i])$ is a random permutation of the features and $\mathbf{K}_{z'^{\phi([i])}} \in \mathbb{R}^{p \times p}$ is the kernel matrix over $\{z'^{\phi([i])}\}_{i=1}^p$. Minimizing this loss function makes the embedding features invariant to augmentations and also avoids redundancy in the features.

It is noteworthy that we found out empirically that the embeddings, used in HSIC loss functions, are better to be normalized to have unit length.

4.7 Overall Loss Function

The proposed loss functions are combined to make an overall loss function for CDSSL. Here, we discuss the overall loss function and the intuition behind the terms of the loss.

4.7.1 Overall Loss Function

The overall proposed loss function in CDSSL is:

$$\mathcal{L} := \underbrace{\lambda_{\text{acs}} \mathcal{L}_{\text{acs}} + \lambda_{\text{ccs}} \mathcal{L}_{\text{ccs}} + \lambda_{\text{acf}} \mathcal{L}_{\text{acf}} + \lambda_{\text{ccf}} \mathcal{L}_{\text{ccf}}}_{\text{linear correlation}} + \underbrace{\lambda_{\text{ads}} \mathcal{L}_{\text{ads}} + \lambda_{\text{cds}} \mathcal{L}_{\text{cds}} + \lambda_{\text{adf}} \mathcal{L}_{\text{adf}} + \lambda_{\text{cdf}} \mathcal{L}_{\text{cdf}}}_{\text{nonlinear dependence}},\tag{4.20}$$

where the λ parameters are non-negative regularization hyperparameters. The first to fourth terms are linear correlation terms while the fifth to eighth terms are nonlinear dependence terms. Analysis of the terms of CDSSL loss function is provided in Table 4.1.

4.7.2 Special Cases of CDSSL

Some of the existing SSL methods can be considered as special cases of the proposed CDSSL loss function. The existing methods which are based on variance, covariance, and correlation are special cases of our loss function whereas the methods dealing with contrastive losses across samples or features may be considered to be special cases indirectly. In fact, it is possible to place the existing SSL methods in one of the eight nodes of the

tree in Fig. 4.1. Consequently, our proposed CDSSL loss function can be considered as a generalization of SSL algorithms.

For example, Barlow Twins [123] is cross-correlation of features. Therefore, Barlow Twins is a direct special case of the CDSSL loss function with $\lambda_{\text{ccf}} = 1$ and $\lambda_{\text{acs}} = \lambda_{\text{ccs}} = \lambda_{\text{acf}} = \lambda_{\text{ads}} = \lambda_{\text{cds}} = \lambda_{\text{adf}} = \lambda_{\text{cdf}} = 0$. Consequently, the performance of CDSSL is lower bounded by the performance of Barlow Twins. It is also noteworthy that the variance and covariance terms in the loss function of VICReg [8] are implicitly equivalent to cross-correlation of features, although they are weighted differently in its loss function.

Another special case of the CDSSL loss function is SSL-HSIC [68] which also uses HSIC. Its loss function is:

$$\mathcal{L} = -\text{HSIC}(Z, Y) + \gamma \sqrt{\text{HSIC}(Z, Z)},$$

where Z is the batch of data and Y is the identity or one-hot encoded label. According to Eq. 5 in [68], $\text{HSIC}(Z, Y)$ can be interpreted as $\text{HSIC}(Z, Z')$ in the language of our paper. Hence, the first term is in fact the cross dependence of samples ($\mathcal{L}_{\text{cds}}$). The second term is the auto-dependence of samples ($\mathcal{L}_{\text{ads}}$). SSL-HSIC [68] covers only cross-dependence of samples and auto-correlation of samples, but does not have any term on the features as in CDSSL.

Another special case of the CDSSL loss function is InfoNCE [81], which is also used in SimCLR [16]. According to [68], InfoNCE can be analyzed as a special case of SSL-HSIC with the cosine kernel. As discussed above, the SSL-HSIC itself is a special case of CDSSL with only cross-dependence of samples and auto-dependence of samples. Therefore, InfoNCE is also a special case of CDSSL with cross-dependence and auto-dependence of samples and the cosine kernel function.

Having a combination of non-zero λ values in the CDSSL loss function provides a superimposition of various approaches boosting the effectiveness of the learned embedding in SSL. This brings together the best of multiple worlds, some of which are existing methods in the literature and some of which are completely novel. For example, CDSSL has the merits of all Barlow Twins, VICReg, InfoNCE, SimCLR, SSL-HSIC, etc., as well as novel loss terms.

Table 4.1: Context and understanding of the CDSSL loss function terms.

Term	Description
$\mathcal{L}_{\text{acs}}$	Decorrelates the embeddings of non-corresponding samples within each batch.
$\mathcal{L}_{\text{ccs}}$	Correlates corresponding samples across augmentations and decorrelates non-corresponding samples across augmentations.
$\mathcal{L}_{\text{acf}}$	Decorrelates non-corresponding embedding features within each batch, avoiding redundancy and information collapse.
$\mathcal{L}_{\text{ccf}}$	Correlates corresponding embedding features across augmentations and decorrelates non-corresponding features across augmentations, avoiding redundancy and information collapse.
$\mathcal{L}_{\text{ads}}$	Increases the variance of embeddings of non-corresponding samples within each batch in the feature space.
$\mathcal{L}_{\text{cds}}$	Increases the dependence of corresponding samples across augmentations.
$\mathcal{L}_{\text{adf}}$	Increases the covariance of embedding features within each batch, avoiding redundancy and information collapse through the feature space.
$\mathcal{L}_{\text{cdf}}$	Increases the dependence of corresponding embedding features across augmentations and decreases dependence of non-corresponding features across augmentations, avoiding redundancy and information collapse.

4.8 Experimental Results

4.8.1 Ablation Study

The proposed CDSSL does contain a number of regularization hyperparameters in Eq. (4.20) which need to be set, however, setting these parameters is not actually a difficult task, precisely because we have a clear intuitive understanding of each loss term (Table 4.1). In Fig. 4.2 we demonstrate the results of grid-based search optimization over the parameter space, tested on two extreme cases: one with few classes (MNIST), and one with many classes (CIFAR-100). For visualization purposes, the hyperparameters are first brought to consistent levels, for fair comparison, before tuning the weights.

According to Fig. 4.2, cross-correlation and cross-dependence of samples play the most important roles among the terms. This makes sense as the augmented samples should be pushed to have similar embeddings. Comparing the optimal hyperparameters for MNIST and CIFAR-100 reveals that auto-correlation and auto-dependence of samples should have less impact when the number of classes is low. This is because the assumption of having non-corresponding samples in the batch, discussed in Section 4.4, is violated in cases

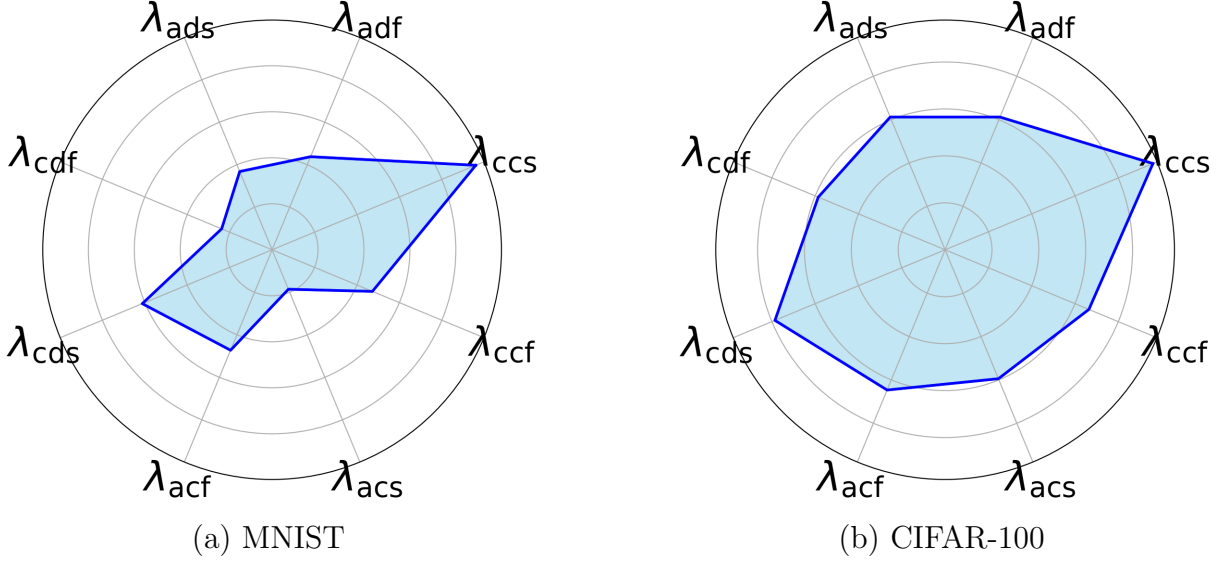


Figure 4.2: Optimal relative values of the regularization hyperparameters for the CDSSL loss function, obtained through grid search, on (a) MNIST with 10 classes and (b) CIFAR-100 with 100 classes. The results highlight the varying importance of different loss terms based on the dataset’s class complexity. Cross-correlation and cross-dependence of samples are more prominent in datasets with a larger number of classes.

with few classes. The other terms in the loss function have moderate impact on the loss regardless of the number of classes.

Figure 4.2 shows ablation for two cases, i.e., having few and many classes in the data. Therefore, for a new dataset, one can see which case is close and choose the weights of loss terms accordingly.

4.8.2 Evaluation and Comparison on Benchmarks

Evaluation by Linear Classification

All experiments in their paper have used the RBF kernel function, which is a universal kernel and thus is valid to be used in HSIC. The proposed algorithm was trained and tested on well-known benchmarks: MNIST [65], CIFAR-10, CIFAR-100 [61], and STL10 [21]. We have tested on multiple datasets with different distributions and domains. Having different datasets with different distributions showcases the statistical significance of our

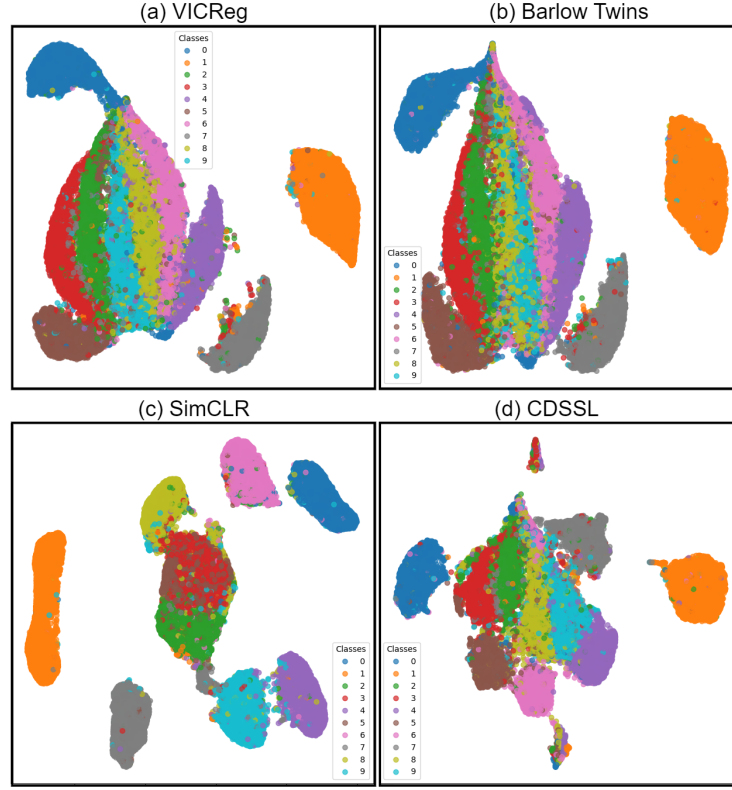


Figure 4.3: Four UMAP visualizations of the learned embeddings for the MNIST dataset. CDSSL in (d) shows better class separation and more isotropic distributions compared to VICReg and Barlow Twins, highlighting its ability to capture discriminative representations.

performance. Moreover, among the datasets we use are cases with few and many classes. We trained and tested VICReg [8], Barlow Twins [123], and SimCLR [18] as methods to compare to, all implemented with consistent settings for fair comparison. After training the SSL method, the expander was dropped and a linear logistic regression was trained and tested on the learned latent embeddings. The top-1 and top-5 accuracies are reported in Table 4.2. The performance of CDSSL is better than VICReg, Barlow Twins, and SimCLR in most of the cases.

Method	MNIST		CIFAR-10		STL10		CIFAR-100	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
VICReg	93.2	99.6	72.4	98.0	52.4	94.1	45.5	74.8
Barlow Twins	92.8	99.7	72.1	97.9	51.3	93.1	46.1	75.1
SimCLR	93.9	99.7	68.6	95.9	50.3	93.6	41.9	68.9
CDSSL	93.5	99.7	74.67	98.1	51.5	93.2	47.0	76.6

Table 4.2: Performance comparison of self-supervised learning on four benchmark datasets: MNIST, CIFAR-10, STL10, and CIFAR-100. Metrics are reported as top-1 and top-5 accuracy (%). CDSSL outperforms VICReg and Barlow Twins across most of the datasets, demonstrating its effectiveness in learning meaningful and transferable representations.

Visualization of Learned Embeddings

The UMAP visualizations [77] of the learned latent embeddings by CDSSL, VICReg, Barlow Twins, and SimCLR are illustrated in Fig. 4.3 for the MNIST and CIFAR-10 datasets. The visualizations clearly show that the classes in MNIST data have been better separated and more isotropically distributed by CDSSL than VICReg and Barlow Twins. The reason for this behavior is explained in the following.

CDSSL includes loss terms such as auto-correlation of samples and auto-dependence of samples that explicitly minimize the correlation/dependence among non-corresponding samples. This makes the classes separated in the embedding space while the features are also non-redundant because of the other loss terms. More inter-class variances results in better uniform propagation of the classes in the embedding space so their UMAP embedding shows isometric classes. This is while there is no explicit loss term in VICReg and Barlow Twins for repelling different (non-corresponding) samples from each other. Both VICReg and Barlow Twins work on the “features” to be non-redundant (uncorrelated) and non-collapsing but they do not explicitly deal with the “samples”. Hence, the inter-class variances are not as much as CDSSL.

This better separation of classes in the embedding space in CDSSL is also showing itself in the experiments. Tables 4.2 and 4.3 show that although VICReg is better than CDSSL in linear classification of STL-10 but it is outperformed by CDSSL in nonlinear classification. Also, Table 4.4 shows that CDSSL outperforms VICReg and Barlow in downstream tasks with larger and noticeable gaps. We will add these discussions in the camera-ready version.

Dataset	VICReg	Barlow Twins	SimCLR	CDSSL
MNIST	95.90	94.96	96.27	97.24
CIFAR-10	74.52	73.62	70.85	75.37
STL10	53.93	52.21	52.60	54.47
CIFAR-100	46.21	46.07	42.11	47.08

Table 4.3: Downstream comparison of nonlinear classification performance, on top of frozen representations. CDSSL provides a strikingly stronger performance, relatively to all compared methods, and for every dataset.

4.8.3 Evaluation by Nonlinear Classification

The proposed CDSSL was compared with VICReg, Barlow Twins, and SimCLR by nonlinear classification on four datasets in Table 4.3. The nonlinear classifier is a multilayer Perceptron with three layers. The resulting classifier, based on the learned latent embeddings, is strikingly superior for CDSSL, outperforming all of the baseline methods, supporting the generalization ability of CDSSL.

4.8.4 Domain Adaptation

We evaluated the cross-dataset generalization capabilities of the proposed CDSSL framework using the domain adaptation tasks shown in Table 4.4. In these experiments, models were trained on one dataset and the expander network discarded. A linear classifier was trained on top of the learned representations without fine-tuning to assess the generalization to unseen datasets. The used datasets in this experiment are MNIST, Digits Dataset [85], CIFAR-10, STL10, and CIFAR-100. Again, the results indicate that CDSSL consistently outperforms VICReg, Barlow Twins, and SimCLR across all tasks.

4.8.5 Implementation Details

Datasets. We evaluate CDSSL on **MNIST**, **CIFAR-10**, **CIFAR-100**, and **STL10**. Each mini-batch image is transformed twice independently to form two augmented views per sample.

Architecture. We use the encoder-expander design of an encoder $f_\theta: \mathbb{R}^d \rightarrow \mathbb{R}^q$ produces a latent $h \in \mathbb{R}^q$ with $q = 512$, followed by an expander $g_\phi: \mathbb{R}^q \rightarrow \mathbb{R}^p$ with $p = 1024$ (hidden

Domain Adaptation Task	VICReg	Barlow Twins	SimCLR	CDSSL
MNIST $\rightarrow$ Digits Dataset	67.7	66.8	71.1	78.3
CIFAR-10 $\rightarrow$ STL10	65.4	65.2	67.8	70.2
CIFAR-10 $\rightarrow$ CIFAR-100	18.3	17.2	18.6	21.4
CIFAR-100 $\rightarrow$ CIFAR-10	57.4	57.6	59.3	62.1

Table 4.4: Domain adaptation performance comparison between VICReg and CDSSL. A linear classifier was trained on top of the frozen representations to evaluate cross-dataset generalization.

width 1024) to obtain $z \in \mathbb{R}^p$. For MNIST we use a lightweight CNN encoder; for CIFAR-10/100 and STL10 we use a standard convolutional encoder with an attached projector head.

Objective. Training minimizes the CDSSL loss (Eq. (4.20)), i.e., the weighted sum of eight terms: linear correlation ($\mathcal{L}_{acs}, \mathcal{L}_{ccs}, \mathcal{L}_{acf}, \mathcal{L}_{ccf}$) and nonlinear dependence via HSIC ($\mathcal{L}_{ads}, \mathcal{L}_{cds}, \mathcal{L}_{adf}, \mathcal{L}_{cdf}$). For HSIC we use an RBF kernel and ℓ_2 -normalize embeddings used in the HSIC terms, which we found to stabilize optimization.

Optimization and schedule. We train with Adam (learning rate 2×10^{-4} , weight decay 10^{-6}), batch size 512, and a cosine annealing schedule over the pretraining horizon (300 epochs in our default runs). Unless stated otherwise, gradients are not clipped.

Evaluation. After self-supervised pretraining, we *freeze* the encoder f_θ and train a linear classifier (logistic regression head) on top of h for 5 epochs using Adam (10^{-3}). We report Top-1/Top-5 accuracy on the standard test splits.

4.9 Conclusion

We introduced CDSSL, a comprehensive framework that integrates both linear correlation and nonlinear dependence to enhance representation learning. By leveraging correlation and HSIC within a RKHS, CDSSL captures rich and diverse features while minimizing redundancy and information collapse. The results show that CDSSL outperforms state-of-the-art methods like VICReg and Barlow Twins, highlighting its capacity to generalize across datasets with varying distributions and complexities.

Chapter 5

SSL in RKHS

5.1 Motivation and Core Idea

Despite the impressive progress of self-supervised learning (SSL) methods such as SimCLR [18], BYOL [46], Barlow Twins [123], and VICReg [8], these frameworks typically operate in standard Euclidean space. Their geometric objectives—invariance, variance, and decorrelation—are all formulated in terms of ℓ_2 norms or linear covariance, implicitly assuming that the latent representations lie on a relatively simple, linearly-structured manifold.

However, after multiple layers of nonlinear transformation in deep networks, this assumption becomes questionable. The true data geometry may reside on a highly nonlinear manifold, poorly captured by Euclidean distances or second-order statistics alone [50]. Classical kernel methods offer a principled solution: by implicitly mapping data into a high-dimensional or even infinite-dimensional Reproducing Kernel Hilbert Space (RKHS), they make it possible to capture both local and global nonlinear relationships using only inner products computed via kernel functions [99]. In supervised learning, the “kernel trick” underpins the success of methods like kernel SVM [79], allowing the discovery of nonlinear decision boundaries without explicit feature construction.

This naturally raises the question: *Can self-supervised learning objectives also be systematically kernelized, lifting their Euclidean formulations into the richer geometry of RKHS?* We argue that this step is not only possible but also desirable. By kernelizing core SSL objectives, one can inherit the expressive power of RKHS and enable representation learning that is better aligned with the intrinsic data manifold.

In this chapter, we present a general approach for *kernelized self-supervised learning* (KSSL), illustrating the idea concretely through **Kernel VICReg**. We show how each term of the VICReg objective, invariance, variance preservation, and covariance decorrelation, can be reformulated in RKHS using double-centered kernel matrices and Hilbert–Schmidt norms [45]. Importantly, while we focus on VICReg as a concrete and popular example, the same kernelization strategy can be readily extended to other SSL objectives, including Barlow Twins and SimCLR.

The remainder of this chapter introduces the technical formulation of Kernel VICReg, demonstrating how invariance, variance, and covariance regularization are naturally lifted into RKHS, thus enabling powerful nonlinear self-supervised representation learning.

5.2 Review of VICReg

5.2.1 Data and Network Settings

The network is composed of an encoder network and an expander network. They pass through an encoder network to output the q -dimensional latent embeddings $\{h^{(i)}\}_{i=1}^b$ and $\{h'^{(i)}\}_{i=1}^b$ where $h^{(i)}, h'^{(i)} \in \mathbb{R}^q$. The latent embeddings pass through the expander network to obtain the p -dimensional output embeddings $\{z^{(i)}\}_{i=1}^b$ and $\{z'^{(i)}\}_{i=1}^b$ where $z^{(i)}, z'^{(i)} \in \mathbb{R}^p$ where $p > q$.

5.2.2 VICReg Loss Function

VICReg [8] is a non-contrastive SSL method that simultaneously enforces three complementary properties on a pair of views $\mathbf{x}$ and $\mathbf{x}'$ of the same sample:

1. **Invariance:** embeddings of corresponding views should be close;
2. **Variance preservation:** each dimension of the embedding space should maintain sufficient spread to avoid collapse;
3. **Covariance decorrelation:** different embedding dimensions should be de-correlated to encourage richness.

The VICReg loss function contains three terms reflecting the three preceding properties:

The **invariance loss** is defined as:

$$\mathcal{L}_{\text{inv}}(\mathbf{x}, \mathbf{x}') = \frac{1}{b} \sum_{i=1}^b \|z^{(i)} - z'^{(i)}\|_2^2, \quad (5.1)$$

which encourages paired views to have similar representations.

The **variance loss** seeks to ensure that no dimension collapses to zero variance, and is defined as

$$\mathcal{L}_{\text{var}}(\mathbf{x}) = \frac{1}{p} \sum_{j=1}^p [\gamma - \sigma_j]_+^2, \quad (5.2)$$

for threshold $\gamma > 0$ and where $[\cdot]_+ = \max(\cdot, 0)$. The sample standard deviation is found as:

$$\sigma_j = \sqrt{\mathbf{Var}(\{z^{(i,j)}\}_{i=1}^b) + \epsilon}, \quad (5.3)$$

where parameter $\epsilon > 0$ prevents degeneracy, and $z^{(i,j)}$ denotes the j -th dimension of $z^{(i)}$.

For the **covariance loss**, let

$$\mathbf{C} = 1/(b-1) \tilde{\mathbf{Z}}^\top \tilde{\mathbf{Z}} \in \mathbb{R}^{p \times p}, \quad (5.4)$$

be the empirical covariance of the zero-centered embeddings $\tilde{\mathbf{Z}} \in \mathbb{R}^{b \times p}$. VICReg penalizes off-diagonal correlations:

$$\mathcal{L}_{\text{cov}}(\mathbf{x}) = \frac{1}{p} \sum_{j=1}^p \sum_{k=1, k \neq j}^p \mathbf{C}_{jk}^2, \quad (5.5)$$

where $\mathbf{C}_{jk}$ denotes the (j, k) -th element of the matrix $\mathbf{C}$. Promoting a diagonal covariance matrix encourages different features to capture distinct aspects of the data.

The **Overall VICReg Loss** then is a weighted summation of the preceding terms:

$$\mathcal{L}_{\text{VICReg}} = \alpha \mathcal{L}_{\text{inv}}(\mathbf{x}, \mathbf{x}') + \beta (\mathcal{L}_{\text{var}}(\mathbf{x}) + \mathcal{L}_{\text{var}}(\mathbf{x}')) + \zeta (\mathcal{L}_{\text{cov}}(\mathbf{x}) + \mathcal{L}_{\text{cov}}(\mathbf{x}')), \quad (5.6)$$

with hyperparameters $\alpha, \beta, \zeta > 0$. VICReg thus avoids the need for negative samples or momentum encoders by combining these three regularizers in Euclidean space, laying the groundwork for our kernelized extension in the next section.

5.3 Kernel VICReg

Standard VICReg operates in Euclidean space by enforcing variance preservation, decorrelating feature dimensions, and ensuring consistency between views of the same sample. Our proposed Kernel VICReg extends these principles to the Reproducing Kernel Hilbert Space (RKHS), allowing for non-linear representations without explicit feature mappings. We rigorously derive the kernelized counterparts of VICReg’s variance, covariance, and invariance losses.

5.3.1 Covariance in RKHS

Before deriving the kernelized VICReg loss components, we establish the fundamental result that the covariance operator in RKHS is proportional to the double-centered kernel matrix. The formulations derived and explained in this section will be used in kernelizing the loss terms of VICReg.

Let $\phi(\mathbf{x})$ be the implicit feature mapping in RKHS $\mathcal{H}$ where $\phi(\cdot)$ is the pulling function into RKHS. The kernel function in the RKHS $\mathcal{H}$ is defined as [37]:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{H}}, \quad (5.7)$$

where $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ denotes the inner product in the RKHS $\mathcal{H}$.

The covariance operator in RKHS is defined as [97, 98]:

$$\mathbb{R}^{t \times t} \ni \mathbf{C}_{\phi}(\mathbf{x}) = \frac{1}{b} \sum_{i=1}^b (\phi(\mathbf{x}_i) - \bar{\phi}) (\phi(\mathbf{x}_i) - \bar{\phi})^{\top}, \quad (5.8)$$

where t is the dimensionality of the RKHS, $\phi(\mathbf{x}_i) - \bar{\phi}$ is the centered feature map of the batch and $\bar{\phi} = \frac{1}{b} \sum_{j=1}^b \phi(\mathbf{x}_j)$ is the mean feature vector of the batch in RKHS. If we let

$$\Phi := [\phi(x_1), \dots, \phi(x_b)] \in \mathbb{R}^{t \times b}, \quad (5.9)$$

then Eq. (5.8) can be restated in matrix form:

$$\mathbb{R}^{t \times t} \ni \mathbf{C}_{\phi}(\mathbf{x}) = \frac{1}{b} (\Phi \mathbf{H})(\Phi \mathbf{H})^{\top}, \quad (5.10)$$

where $\mathbf{H} := \mathbf{I}_b - \frac{1}{b} \mathbf{1}_b \mathbf{1}_b^{\top} \in \mathbb{R}^{b \times b}$ is the centering matrix.

Using the kernel trick [37], we define kernel matrix $\mathbf{K} \in \mathbb{R}^{b \times b}$ whose (i, j) -th element is:

$$\mathbf{K}(\mathbf{x})[i, j] = k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{H}}. \quad (5.11)$$

From Eq. (5.9), the kernel matrix can be stated as:

$$\mathbf{K}(x) = \Phi^\top \Phi. \quad (5.12)$$

The double-centered kernel matrix is [38]:

$$\widehat{\mathbf{K}}(\mathbf{x}) := \mathbf{H}\mathbf{K}(\mathbf{x})\mathbf{H} \stackrel{(5.12)}{=} \mathbf{H}\Phi^\top \Phi \mathbf{H} \stackrel{(s)}{=} (\Phi \mathbf{H})^\top (\Phi \mathbf{H}), \quad (5.13)$$

where step (s) considers that the centering matrix $\mathbf{H}$ is symmetric. The double-centered kernel matrix will be used in the following.

The following explains the relation of the covariance operator and double-centered kernel in RKHS. The squared Hilbert-Schmidt norm of the covariance matrix in RKHS is:

$$\begin{aligned} \|\mathbf{C}_\phi(\mathbf{x})\|_{\text{HS}}^2 &= \text{tr}(\mathbf{C}_\phi(\mathbf{x})^\top \mathbf{C}_\phi(\mathbf{x})) \\ &\stackrel{(5.10)}{=} \frac{1}{b^2} \text{tr}((\Phi \mathbf{H})(\Phi \mathbf{H})^\top (\Phi \mathbf{H})(\Phi \mathbf{H})^\top) \\ &\stackrel{(a)}{=} \frac{1}{b^2} \text{tr}((\Phi \mathbf{H})^\top (\Phi \mathbf{H})(\Phi \mathbf{H})^\top (\Phi \mathbf{H})) \\ &\stackrel{(5.13)}{=} \frac{1}{b^2} \text{tr}(\widehat{\mathbf{K}}(\mathbf{x})\widehat{\mathbf{K}}(\mathbf{x})) \\ &= \frac{1}{b^2} \text{tr}(\widehat{\mathbf{K}}(\mathbf{x})^2), \end{aligned} \quad (5.14)$$

where $\text{tr}(\cdot)$ denotes the trace of matrix and step (a) follows from the cyclic property of the trace operator. For easier computation in computer, it is possible to restate Eq. (5.14) using the Frobenius norm:

$$\begin{aligned} \|\mathbf{C}_\phi(\mathbf{x})\|_{\text{HS}}^2 &= \frac{1}{b^2} \text{tr}(\widehat{\mathbf{K}}(\mathbf{x})^2) = \frac{1}{b^2} \sum_{i \neq j} [\widehat{\mathbf{K}}(\mathbf{x})]_{i,j}^2 \\ &= \frac{1}{b^2} \left(\|\widehat{\mathbf{K}}(\mathbf{x})\|_F^2 - \sum_{i=1}^b [\widehat{\mathbf{K}}(\mathbf{x})]_{i,i}^2 \right). \end{aligned} \quad (5.15)$$

From Eq. (5.14), the covariance operator in RKHS is proportional to the centered kernel matrix:

$$\mathbf{C}_\phi(\mathbf{x}) \propto \frac{1}{b} \widehat{\mathbf{K}}(\mathbf{x}). \quad (5.16)$$

This key result enables the kernelization of VICReg's variance and covariance regularization terms, as discussed next.

5.3.2 Kernelized Variance Regularization

The variance regularization term in VICReg prevents representation collapse by ensuring that the variance along each feature dimension remains sufficiently large, i.e., above a threshold γ . In RKHS, the variance of feature dimensions corresponds to the eigenvalues of $\mathbf{C}_\phi(\mathbf{x})$. Since $\mathbf{C}_\phi(\mathbf{x})$ is proportional to $\widehat{\mathbf{K}}(\mathbf{x})$ according to Eq. (5.16), we define the kernelized variance loss as:

$$\mathcal{L}_{\text{var}}(\mathbf{x}) = \frac{1}{b} \sum_{i=1}^b \left(\left[\gamma - \sqrt{\frac{\lambda_i}{b}} + \epsilon \right]_+ \right)^2, \quad (5.17)$$

where $[\cdot]_+ := \max(\cdot, 0)$ is the standard Hinge loss, $\{\lambda_i\}_{i=1}^b$ are the eigenvalues of $\widehat{\mathbf{K}}(\mathbf{x})$, γ is a threshold for the minimum desired standard deviation, and ϵ is a small positive number preventing numerical instabilities. The proof of why λ_i/b is understood as a variance in RKHS will be developed in Section 5.4.1.

It is noteworthy that computing the eigenvalues here is not concerning in terms of time of computation because the double-centered kernel is $b \times b$ where b is the batch size, which is usually not a very large number.

5.3.3 Kernelized Covariance Regularization

To prevent redundancy in representations, VICReg penalizes off-diagonal elements of the covariance matrix (see Eq. (5.5)). Building on Eq. (5.15), the kernelized covariance loss can be defined as

$$\begin{aligned} \mathcal{L}_{\text{cov}}(\mathbf{x}) &= \|\mathbf{C}_\phi(\mathbf{x})\|_{\text{HS}} \\ &= \frac{1}{b} \sqrt{\left(\|\widehat{\mathbf{K}}(\mathbf{x})\|_F^2 - \sum_{i=1}^b [\widehat{\mathbf{K}}(\mathbf{x})]_{i,i}^2 \right)}. \end{aligned} \quad (5.18)$$

Because of the direct relation between covariance and correlation, this regularization enforces decorrelation between features in RKHS.

Note that it was empirically observed that using square root in Eq. (5.18) needs a smoother coefficient for the kernelized covariance regularization term in the overall loss function. One can use the squared Hilbert-Schmidt norm of the covariance matrix as the loss term; however, that will change the optimal regularization hyperparameters for combining the loss terms.

5.3.4 Kernelized Invariance Term

The invariance term of loss function minimizes the mean squared error between corresponding samples $\mathbf{x}$ and $\mathbf{x}'$, i.e., different views of the same sample. Consider the following $(b \times b)$ kernel matrices:

$$\begin{aligned}\mathbf{K}(\mathbf{x}, \mathbf{x})[i, j] &= k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{H}}, \\ \mathbf{K}(\mathbf{x}', \mathbf{x}')[i, j] &= k(\mathbf{x}'_i, \mathbf{x}'_j) = \langle \phi(\mathbf{x}'_i), \phi(\mathbf{x}'_j) \rangle_{\mathcal{H}}, \\ \mathbf{K}(\mathbf{x}, \mathbf{x}')[i, j] &= k(\mathbf{x}_i, \mathbf{x}'_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}'_j) \rangle_{\mathcal{H}}.\end{aligned}$$

Given kernel matrices $\mathbf{K}(\mathbf{x}, \mathbf{x})$ and $\mathbf{K}(\mathbf{x}', \mathbf{x}')$ for two augmented views and their cross-kernel matrix $\mathbf{K}(\mathbf{x}, \mathbf{x}')$, the distance of the views in RKHS is defined as [96]:

$$\mathcal{L}_{\text{inv}}(\mathbf{x}, \mathbf{x}') = \frac{1}{b} \text{tr}(\mathbf{K}(\mathbf{x}, \mathbf{x}) + \mathbf{K}(\mathbf{x}', \mathbf{x}') - 2\mathbf{K}(\mathbf{x}, \mathbf{x}')). \quad (5.19)$$

This loss term pushes the corresponding, i.e., augmented, instances toward each other in the RKHS and pulls away the non-corresponding instances away from each other in the RKHS. This enforces consistency across augmentations in RKHS.

5.3.5 Overall Kernel VICReg Loss

Combining all three regularization terms, the final Kernel VICReg loss is given by:

$$\mathcal{L}_{\text{Kernel-VICReg}} = \alpha \mathcal{L}_{\text{inv}}(\mathbf{x}, \mathbf{x}') + \beta (\mathcal{L}_{\text{var}}(\mathbf{x}) + \mathcal{L}_{\text{var}}(\mathbf{x}')) + \zeta (\mathcal{L}_{\text{cov}}(\mathbf{x}) + \mathcal{L}_{\text{cov}}(\mathbf{x}')), \quad (5.20)$$

where α, β, ζ are hyperparameters controlling the contributions of invariance, variance, and covariance. Note that the best values for hyperparameters $\alpha, \beta, \zeta, \gamma, \epsilon$ differ across the VICReg and kernel VICReg. The best hyperparameters can be found depending on the dataset, so as they are found in the original VICReg.

By reformulating VICReg in RKHS, our method enables self-supervised learning in high-dimensional implicit feature spaces without explicit feature extraction, making it a powerful framework for non-linear representation learning.

5.4 Discussions

5.4.1 Relation of Kernelized Variance Term with Kernel PCA

There is a close relation between the kernelized variance term, in kernel VICReg, and kernel Principal Component Analysis (kernel PCA) [97]. In standard PCA, the eigenvalues of the covariance matrix $\mathbf{C} \in \mathbb{R}^{p \times p}$ are calculated using the following eigenvalue problem [38, 40]:

$$\mathbf{C}\mathbf{u}_i = \eta_i \mathbf{u}_i, \quad (5.21)$$

where $\eta_i \in \mathbb{R}$ and $\mathbf{u}_i \in \mathbb{R}^p$ are the i -th eigenvalue and eigenvector of the covariance matrix $\mathbf{C}$, respectively.

In kernel PCA, the eigenvalue problem of the double-centered kernel matrix $\widehat{\mathbf{K}}(\mathbf{x}) \in \mathbb{R}^{b \times b}$ is considered [38]:

$$\widehat{\mathbf{K}}(\mathbf{x})\mathbf{v}_i = \lambda_i \mathbf{v}_i, \quad (5.22)$$

where $\lambda_i \in \mathbb{R}$ and $\mathbf{v}_i \in \mathbb{R}^b$ are the i -th eigenvalue and eigenvector of the double-centered kernel matrix $\widehat{\mathbf{K}}(\mathbf{x})$, respectively.

Each eigenvector $\mathbf{v}_i$ gives a principal direction in the feature space $\phi(\mathbf{U}) \in \mathbb{R}^{t \times b}$. According to the representation theory, any function in RKHS lies in the span of all points in the RKHS [78, 37]:

$$\phi(\mathbf{U}) = \sum_{i=1}^b \alpha_i \phi(\mathbf{x}_i) = \Phi \mathbf{A}, \quad (5.23)$$

where $\Phi \in \mathbb{R}^{t \times b}$ is defined in Eq. (5.9) and $\mathbf{A} := [\alpha_1, \dots, \alpha_b]^\top \in \mathbb{R}^{b \times b}$ is the matrix of coefficients in the linear combination.

On the one hand, the variance of the principal direction in the feature space is:

$$\begin{aligned} \mathbf{Var}(\phi(\mathbf{U})) &= \frac{1}{b} \|(\Phi \mathbf{H})^\top \mathbf{A}\|_2^2 \\ &= \frac{1}{b} \text{tr}(((\Phi \mathbf{H})^\top \mathbf{A})^\top ((\Phi \mathbf{H})^\top \mathbf{A})) \\ &= \frac{1}{b} \text{tr}(\mathbf{A}^\top \Phi \Phi^\top \mathbf{A}) \\ &\stackrel{(5.23)}{=} \frac{1}{b} \text{tr}(\mathbf{A}^\top \underbrace{(\Phi \mathbf{H})^\top (\Phi \mathbf{H})}_{\widehat{\mathbf{K}}(\mathbf{x})} \underbrace{(\Phi \mathbf{H})^\top (\Phi \mathbf{H})}_{\widehat{\mathbf{K}}(\mathbf{x})} \mathbf{A}) \\ &= \frac{1}{b} \text{tr}(\mathbf{A}^\top \widehat{\mathbf{K}}(\mathbf{x})^2 \mathbf{A}). \end{aligned}$$

For one of the coefficient vectors, this equation becomes:

$$\mathbf{Var}(\phi(\mathbf{u}_i)) = \frac{1}{b} \alpha_i^\top \widehat{\mathbf{K}}(\mathbf{x})^2 \alpha_i, \quad (5.24)$$

where trace is dropped because the trace of a scalar is equal to itself.

Assume that the coefficient α_i is the i -th eigenvector of the double-centered kernel matrix. Thus, the variance becomes:

$$\mathbf{Var}(\phi(\mathbf{u}_i)) = \frac{1}{b} \mathbf{v}_i^\top \widehat{\mathbf{K}}(\mathbf{x})^2 \mathbf{v}_i. \quad (5.25)$$

Squaring the double-centered kernel matrix in Eq. (5.22) gives:

$$\widehat{\mathbf{K}}(\mathbf{x})^2 \mathbf{v}_i = \lambda_i^2 \mathbf{v}_i. \quad (5.26)$$

Substituting Eq. (5.26) in Eq. (5.25) provides:

$$\begin{aligned} \mathbf{Var}(\phi(\mathbf{u}_i)) &= \frac{1}{b} \alpha_i^\top \lambda_i^2 \alpha_i = \frac{1}{b} \lambda_i^2 \alpha_i^\top \alpha_i \\ &= \frac{1}{b} \lambda_i^2 \|\alpha_i\|_2^2. \end{aligned} \quad (5.27)$$

The coefficient is usually normalized to have $\lambda_i \|\alpha_i\|_2^2 = 1$ (so that $\|\phi(\mathbf{u}_i)\|_{\mathcal{H}} = 1$):

$$\alpha_i = \frac{\mathbf{v}_i}{\sqrt{\lambda_i}} \implies \|\alpha_i\|_2^2 = \frac{1}{\lambda_i}, \quad (5.28)$$

where the eigenvector $\mathbf{v}_i$ is assumed to be normalized to have unit length, i.e., $\|\mathbf{v}_i\|_2 = 1$. According to Eq. (5.28), the Eq. (5.27) becomes:

$$\mathbf{Var}(\phi(\mathbf{u}_i)) = \frac{\lambda_i}{b}. \quad (5.29)$$

This proves why λ_i/b is used as variance in Eq. (5.17).

On the other hand, left-multiplying Eq. (5.22) by $\mathbf{v}_i^\top$ gives:

$$\mathbf{v}_i^\top \widehat{\mathbf{K}}(\mathbf{x}) \mathbf{v}_i = \mathbf{v}_i^\top \lambda_i \mathbf{v}_i = \lambda_i \mathbf{v}_i^\top \mathbf{v}_i = \lambda_i \|\mathbf{v}_i\|_2^2 \stackrel{(a)}{=} \lambda_i,$$

where (a) assumes that the eigenvector $\mathbf{v}_i$ is normalized to have unit length. Therefore, if $\|\mathbf{v}_i\|_2 = 1$, then:

$$\lambda_i = \mathbf{v}_i^\top \widehat{\mathbf{K}}(\mathbf{x}) \mathbf{v}_i. \quad (5.30)$$

Putting Eq. (5.30) in Eq. (5.29) provides:

$$\mathbf{Var}(\phi(\mathbf{u}_i)) = \frac{1}{b} \mathbf{v}_i^\top \widehat{\mathbf{K}}(\mathbf{x}) \mathbf{v}_i. \quad (5.31)$$

This analysis shows that the kernelized variance regularization in the proposed kernel VICReg can be considered as kernel PCA. A similar analysis can be discussed to analyze the variance regularization term in VICReg as PCA.

5.4.2 Connection to HSIC and Independence

The squared Hilbert–Schmidt norm of the RKHS covariance operator—used in our covariance loss—is closely related to the Hilbert–Schmidt Independence Criterion (HSIC), a well-established kernel-based dependence measure. In this view, our covariance loss can be interpreted as minimizing feature dependence in RKHS, encouraging the learning of diverse and disentangled features. This theoretical grounding strengthens the regularization effect of the kernel VICReg loss beyond simple decorrelation.

5.4.3 Kernel Choice as Inductive Bias

Different kernels induce different geometric priors. For example, the RBF kernel emphasizes local smoothness, the Laplacian kernel allows sharper decision boundaries, and the rational quadratic interpolates between them. Our experiments reveal that no single kernel is optimal across all datasets; instead, performance depends on the match between the dataset structure and the kernel-induced geometry. This makes Kernel VICReg not only robust but also adaptable to task-specific data distributions.

5.5 Experimental Results

We evaluate Kernel VICReg on a range of benchmark datasets to assess its ability to learn rich, non-linear representations in self-supervised settings. Our experiments span small-scale datasets (MNIST, CIFAR-10), mid-scale transfer learning (STL-10), and large-scale benchmarks (TinyImageNet, ImageNet100). For all experiments, we use a ResNet-18 backbone as the encoder, followed by a two-layer MLP projector. The model is trained using the Adam optimizer with an initial learning rate of 3×10^{-4} , batch size of 256, and

cosine learning rate scheduling. Each dataset is augmented following standard protocols: random cropping, horizontal flipping, and color jitter for natural image datasets.

To investigate the effect of different kernels in the Reproducing Kernel Hilbert Space (RKHS), we implement Kernel VICReg using four kernels: linear, radial basis function (RBF), Laplacian, and rational quadratic (RQ). The kernel matrices are computed batch-wise, with double-centering applied to ensure zero-mean embeddings in RKHS. We evaluate the learned representations using linear probing, where a logistic regression classifier is trained atop frozen embeddings, and transfer learning, where encoders pretrained on CIFAR-10 are evaluated on STL-10.

5.5.1 Comparison with Baselines

We compare Kernel VICReg against a diverse set of prominent self-supervised learning methods, including contrastive (SimCLR, MoCo), clustering-based (SwAV), and non-contrastive frameworks (BYOL, DINO, SimSiam, Barlow Twins, and VICReg). All baselines are implemented with the same ResNet-18 encoder to ensure fairness, and we reuse reported numbers or reproduce them when necessary using identical augmentation and optimization settings.

Table 5.1 summarizes the top-1 linear evaluation accuracy on ImageNet100 and TinyImageNet. While VICReg exhibits competitive performance on ImageNet100, it collapses on TinyImageNet due to its sensitivity to small datasets with high intra-class variance. In contrast, Kernel VICReg remains stable across all settings, with the Laplacian and RQ kernels achieving the best performance, demonstrating the robustness of RKHS-based regularization.

Table 5.2 presents results on MNIST and CIFAR-10. Kernel VICReg consistently outperforms its Euclidean counterpart, particularly on MNIST where the Laplacian kernel reaches 98.50% accuracy. On CIFAR-10, the RQ kernel achieves the best performance at 86.18%, indicating that kernel choice adapts to data complexity.

Finally, Table 5.3 reports transfer learning results on STL-10 using encoders pretrained on CIFAR-10. Kernel VICReg transfers better than VICReg highlighting its generalization capabilities in low-label regimes.

Table 5.1: Linear probing results on ImageNet100 and TinyImageNet using a ResNet-18 backbone. All models are trained in a self-supervised manner and evaluated by training a linear classifier on frozen embeddings. Performances of baseline methods on TinyImageNet are adapted from [69]. Standard VICReg, while competitive on ImageNet100, collapses on TinyImageNet due to high intra-class variability and limited data. In contrast, Kernel VICReg remains stable across both datasets, with Laplacian and RQ kernels yielding the strongest results, demonstrating the robustness of RKHS-based regularization.

Model	ImageNet100	TinyImageNet
SimCLR	78.64	37.83
SwAV	74.36	35.39
MoCo	79.62	41.23
BYOL	80.88	36.31
DINO	75.41	35.77
SimSiam	78.80	27.96
VICReg	79.77	Collapse
Barlow Twins	80.63	-
Kernel VICReg - Linear (ours)	77.34	37.48
Kernel VICReg - RBF (ours)	78.14	38.21
Kernel VICReg - Laplacian (ours)	79.92	40.12
Kernel VICReg - RQ (ours)	79.80	40.38

5.5.2 Further Analysis and Insights

To better understand how kernelization affects the structure of learned representations, we visualize the embedding spaces on the MNIST dataset using UMAP for three models: original VICReg, Kernel VICReg with a linear kernel, and Kernel VICReg with a Laplacian kernel (Figure 5.1). The UMAP plots reveal differences in cluster geometry across these methods. Representations from standard VICReg exhibit some class separation, but the clusters are elongated, and lack compactness, suggesting anisotropic variance and potential feature collapse. The red cluster in VICReg is separated and is constructing two clusters however this is not the case for our method.

Kernel VICReg with a linear kernel improves upon this, producing tighter and more separated clusters, indicating that even without explicit nonlinearity in the kernel, RKHS-based decorrelation provides better structure. However, the most striking improvement

Table 5.2: Linear probing results on MNIST and CIFAR-10 with ResNet-18 backbone. Kernel VICReg consistently improves upon standard VICReg, particularly on CIFAR-10 where the rational quadratic kernel achieves the highest accuracy (86.18%). On MNIST, the Laplacian kernel yields the best performance (98.50%), highlighting how different kernels act as inductive biases that align with dataset complexity and structure.

Model	MNIST	CIFAR-10
VICReg	97.15	83.41
Kernel VICReg - Linear	98.33	86.08
Kernel VICReg - RBF	92.68	81.13
Kernel VICReg - Laplacian	98.50	84.56
Kernel VICReg - RQ	97.46	86.18

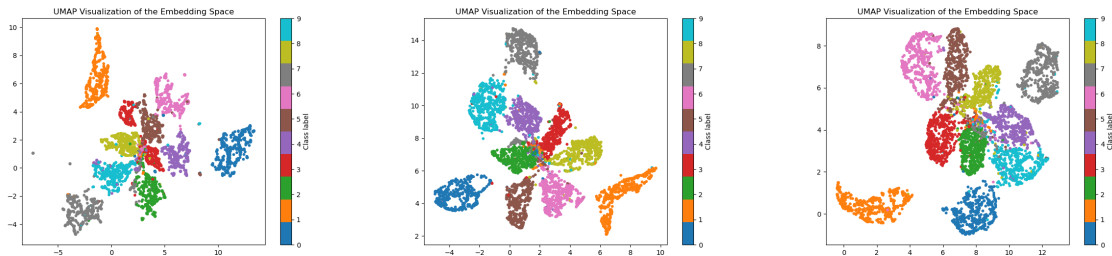


Figure 5.1: UMAP projections of MNIST embeddings from VICReg (left), Kernel VICReg with linear kernel (middle), and Kernel VICReg with Laplacian kernel (right). Laplacian kernel produces rounder, more isometric clusters.

appears with the Laplacian kernel: clusters become nearly circular and uniformly spaced, exhibiting strong isometry. This implies that the Laplacian-induced RKHS preserves pairwise relations and local structure more effectively, leading to embeddings with more consistent intra-class variance and improved inter-class margins.

These qualitative findings are reflected in quantitative performance. As shown earlier in Table 5.1, Kernel VICReg maintains stable and competitive accuracy across both large-scale (ImageNet100) and small-scale (TinyImageNet) benchmarks. Notably, VICReg collapses on TinyImageNet, consistent with its known sensitivity to datasets exhibiting high intra-class variance or insufficient regularization. In contrast, kernelized versions—especially with Laplacian and rational quadratic kernels—perform robustly, demon-

Table 5.3: Transfer learning results on STL-10 using embeddings pretrained on CIFAR-10 with a ResNet-18 backbone. Kernel VICReg improves over standard VICReg, with the rational quadratic (RQ) kernel achieving the highest accuracy (72.34%). These results highlight the stronger generalization capability of kernelized representations when transferred to new datasets, especially under limited-label regimes.

Model	STL-10 Accuracy
VICReg	69.82
Kernel VICReg - Linear	71.38
Kernel VICReg - RBF	67.21
Kernel VICReg - Laplacian	71.09
Kernel VICReg - RQ	72.34

strating the benefits of nonlinear geometric alignment in RKHS.

Furthermore, the results in Table 5.3 highlight the generalization strength of kernel-based methods in transfer learning. Embeddings trained on CIFAR-10 and evaluated on STL-10 show that the RQ and Laplacian kernels outperform VICReg and even the linear kernel variant. These findings support the hypothesis that kernel-induced representations better capture underlying data manifolds, resulting in improved performance on downstream tasks with distributional shifts.

Overall, Kernel VICReg offers a principled extension of VICReg that gracefully incorporates nonlinearity through RKHS-based loss formulations. The improved cluster geometry, resilience to collapse, and higher transfer accuracy together suggest that kernelized self-supervision is a promising direction for representation learning beyond Euclidean limitations.

5.5.3 Implementation Details

Datasets. We evaluate on **MNIST**, **CIFAR-10**, **CIFAR-100**, **STL10**, **TinyImageNet** (64×64), and **ImageNet100**. Each mini-batch forms two independent augmented views per sample.

Augmentations. Dataset-appropriate pipelines are used:

- **MNIST:** RandomRotation(20°), RandomAffine(translate=(0.2, 0.2)), ColorJitter(0.2, 0.2).
- **CIFAR-10/100:** ToTensor, RandomResizedCrop(32, [0.2, 1.0]), RandomGrayscale(0.2).
- **STL10:** ColorJitter(0.2, 0.2, 0.2, 0.1), RandomRotation(15°), RandomGrayscale(0.2), ToTensor.
- **TinyImageNet:** RandomResizedCrop(64, [0.2, 1.0]), RandomHorizontalFlip(0.5), RandomApply(ColorJitter(0.4, 0.4, 0.2, 0.1), $p=0.8$), RandomGrayscale(0.2), RandomRotation(15°), RandomApply(GaussianBlur($k=5$, $\sigma \in [0.1, 2.0]$), $p=0.5$), RandomSolarize(128, $p=0.2$), Normalize((0.4802, 0.4481, 0.3975), (0.2770, 0.2691, 0.2821)).
- **ImageNet100:** Resize(224), RandomResizedCrop(84, [0.6, 0.8]), RandomApply(ColorJitter(0.4, 0.4, 0.2, 0.1), $p=0.8$), RandomGrayscale(0.1), (optionally) mild GaussianBlur, RandomRotation(5°).

Backbone and heads. We follow the encoder–expander design of an encoder producing $h \in \mathbb{R}^{512}$ and an expander mapping to $z \in \mathbb{R}^{1024}$ (hidden width 1024). A small CNN is used for MNIST; a ResNet-style encoder is used for CIFAR-10/100, STL10, TinyImageNet, and ImageNet100.

Kernels. We instantiate Kernel VICReg with linear, RBF, Laplacian, and rational quadratic (RQ) kernels. Kernel matrices are computed per batch on expander outputs (z_1, z_2) and double-centered via $H = I - \frac{1}{b}\mathbf{1}\mathbf{1}^\top$. For RBF/Laplacian/RQ we use a median-heuristic bandwidth unless noted.

Objective and training loop. Training uses `train_kernelized` with our `KernelVICRegLoss`:

1. two augmented views $\rightarrow$ encoder+expander $\rightarrow z_1, z_2$;
2. build K_{11}, K_{22}, K_{12} on (z_1, z_2) ;
3. compute $\mathcal{L}_{\text{inv}}, \mathcal{L}_{\text{var}}, \mathcal{L}_{\text{cov}}$ in RKHS and backpropagate $\alpha\mathcal{L}_{\text{inv}} + \beta(\mathcal{L}_{\text{var}}^{(1)} + \mathcal{L}_{\text{var}}^{(2)}) + \zeta(\mathcal{L}_{\text{cov}}^{(1)} + \mathcal{L}_{\text{cov}}^{(2)})$.

We use Adam (lr 2×10^{-4}), weight decay 10^{-6} , batch size 512, and cosine annealing for 300 epochs.

Evaluation. We freeze the encoder and train a linear logistic-regression head for 5 epochs (Adam, 10^{-3}). We report Top-1/Top-5 accuracy on MNIST, CIFAR-10, CIFAR-100, STL10, TinyImageNet, and ImageNet100, and provide UMAP visualizations of frozen embeddings for qualitative analysis.

Kernel bandwidths and default settings. We use the following kernels on the projector outputs $z \in \mathbb{R}^p$. Unless stated otherwise, z is ℓ_2 -normalized per sample before forming kernels.

- **Linear:** $k(x, y) = x^\top y$. *No bandwidth.*

- **RBF (Gaussian):**

$$k(x, y) = \exp\left(-\frac{\|x-y\|_2^2}{2\ell^2}\right).$$

Default: $\ell = \text{median}_{i<j}\|x_i - x_j\|_2$ (batch median heuristic). If a fixed value is required, use $\ell = 1.0$ with unit-norm features.

- **Laplacian:**

$$k(x, y) = \exp\left(-\frac{\|x-y\|_2}{\ell}\right).$$

Default: $\ell = \text{median}_{i<j}\|x_i - x_j\|_2$. Fixed fallback: $\ell = 1.0$ with unit-norm features.

- **Rational Quadratic (RQ):**

$$k(x, y) = \left(1 + \frac{\|x-y\|_2^2}{2\alpha\ell^2}\right)^{-\alpha}.$$

Default: $\alpha = 1.0$, $\ell = \text{median}_{i<j}\|x_i - x_j\|_2$. Fixed fallback: $\alpha = 1.0$, $\ell = 1.0$ (unit-norm features).

- **Cosine (optional):**

$$k(x, y) = \frac{x^\top y}{\|x\|_2 \|y\|_2}.$$

Practical notes. (1) Recompute the median-heuristic bandwidth per batch and optionally smooth it with an EMA across iterations. (2) When z is unit-norm, typical pairwise distances lie in $[0, 2]$, so fixed $\ell \in [0.7, 1.3]$ is a reasonable starting range for MNIST, CIFAR-10/100, STL10, TinyImageNet, and ImageNet100. (3) For stability, clamp very small eigenvalues in the variance term with $\epsilon \in [10^{-6}, 10^{-4}]$.

5.6 Conclusion

We introduced *Kernel VICReg*, a principled extension of VICReg that lifts self-supervised learning objectives from Euclidean space to Reproducing Kernel Hilbert Space (RKHS). By reformulating invariance, variance, and covariance terms in kernel space using double-centered kernel matrices and Hilbert–Schmidt norms, our method captures complex non-linear structures without requiring explicit feature mappings.

Our empirical results across diverse datasets demonstrate the robustness and effectiveness of this kernelized formulation. Kernel VICReg outperforms its Euclidean counterpart, particularly in challenging regimes such as TinyImageNet, where standard VICReg collapses. Moreover, kernel-induced representations exhibit superior generalization in transfer learning tasks, as evidenced by improvements on STL-10. Visualization through UMAP further reveals that kernelization promotes more compact, isometric cluster structures, especially under the Laplacian kernel.

These findings suggest that kernel methods offer a natural and powerful means to enhance self-supervised learning. While our work focuses on VICReg, the framework readily extends to other SSL objectives such as Barlow Twins and SimCLR, opening promising avenues for future research in kernelized SSL. This study contributes to bridging classical kernel theory and modern representation learning by showing that the integration of RKHS structure meaningfully improves both stability and expressiveness in self-supervised models.

Chapter 6

Context-Aware Augmentation

6.1 Motivation

Data augmentation has always been a core component of self-supervised learning. Traditional approaches use random transformations such as cropping, flipping, or color jittering to create varied views of each input image [18, 72]. While these augmentations are effective at increasing data diversity, they lack an understanding of the underlying content and context of the image. As a result, conventional augmentation methods may fail to generate feature spaces that are both semantically meaningful and relevant to downstream tasks.

We argue that augmentation should go beyond randomness. The inherent structure of objects and their relationships in an image carries valuable information that is often ignored by standard augmentations. Human perception naturally focuses on objects and their spatial context to understand images, an ability that conventional machine learning pipelines have not fully exploited.

Motivated by this gap, we propose a shift towards context-aware augmentation. By leveraging object detection, it is possible to inform the augmentation process with explicit knowledge of objects and their spatial relationships in each image [100]. This enables the model to focus on more meaningful regions, capturing context and semantics that random perturbations often overlook. The goal is to produce a feature space that better reflects the true content of the data, leading to richer and more transferable representations for tasks such as classification, detection, and localization.

6.2 Core Idea

The central idea behind our approach is to move from purely random augmentations toward a more informed and context-driven augmentation pipeline. Instead of treating every part of an image equally, we use an object detector (specifically YOLOv8 [56]) to identify key objects and their locations within the image.

With this information, we generate augmentations that either focus on, or are guided by, these detected objects. For instance, instead of random cropping, we can crop around detected objects, adjust bounding box sizes, or modify augmentations based on objectness probability. This results in two types of augmented views: conventional (random) and context-aware (object-guided).

We integrate this object-aware augmentation process into the SimCLR framework [18]. By mixing both random and object-based augmentations, the model is exposed to a wider variety of semantically meaningful transformations. This helps the network learn feature representations that are not just robust to noise, but also sensitive to object-level context and spatial relationships. In downstream tasks like object detection or recognition, this leads to features that transfer better and capture more relevant information from each image.

6.3 Methodology

Contrastive learning [54], a fundamental concept in self-supervised representation learning, aims to train a model to distinguish between similar and dissimilar instances in a dataset. In our approach, we build upon this foundation to create a more robust self-supervised learning framework. As mentioned before, our research focuses on moving beyond traditional augmentation techniques that rely on random image perturbations. The intention is to transition to a more context-aware and semantically meaningful approach for feature space generation. Instead of random perturbations, we propose an augmentation technique that leverages the information encapsulated within objects present in an image. By doing so, we aim to create a feature space intricately tied to the content and spatial relationships of detected objects.

In order to assess the efficacy of integrating context-aware augmentation into a Self-Supervised Representation Learning (SSRL) approach, we designed the following experimental setup:

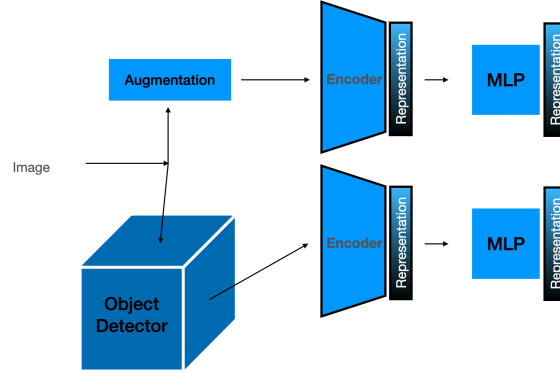


Figure 6.1: Proposed Modified-SimCLR Schema: The top-left augmentation employs a random traditional approach, while in a departure from SimCLR, we integrate a boundary box outlining a detected object as an alternative augmentation (bottom-left). The strictness of detection boundaries and the confidence threshold of the object detector are randomized. The rest of the figure depicts the standard SimCLR architecture.

1. We modified SimCLR to incorporate object detection-based augmentation with random objectness probability and varying bounding box strictness within a specified range.
2. We removed the MLP heads (which were initially used for the pretext task of image classification on ImageNet [28]) and retained the encoders, allowing us to obtain feature representations.
3. We leveraged the feature representations obtained in the previous steps as the backbone of a Faster R-CNN [42], for the purpose of multi-object detection on the COCO [70] dataset as the downstream task (Merely to demonstrate that the feature space quality has been improved, not with the primary objective of improving the accuracy of the multi-object detection algorithm).

6.3.1 Modified SimCLR

To validate our enriched feature space hypothesis, we modified the SimCLR approach and integrated an object detector, YOLOv8 [56], with SimCLR as illustrated in Fig. 6.1. YOLOv8 was chosen for its efficiency and accuracy in object detection. This modification

was intended to inject explicit object-level information into the self-supervised learning process. The integration of object detection allows the network to focus on relevant objects and their relationships, potentially leading to more semantically meaningful feature representations. Our modification resulted in two significant enhancements:

- **Improved Semantic Representations:** By incorporating object detection, the modified SimCLR architecture generated feature representations that were not only semantically meaningful but also contextually relevant to downstream tasks.
- **Enhanced Generalization:** The modified architecture demonstrated the potential to generalize better to various samples due to its incorporation of object-level knowledge.

6.3.2 Integration of Faster R-CNN

While the classification experiments validate that Modified-SimCLR produces more robust and semantically meaningful features, they do not directly assess whether these representations capture object-level semantics. Since our augmentation strategy explicitly leverages object detection, a natural next step is to evaluate the learned features on an object-centric downstream task. Faster R-CNN is an adopted benchmark for multi-object detection, making it an ideal testbed. Replacing its conventional convolutional backbone with Modified-SimCLR features allows us to test whether context-aware augmentation yields embeddings that transfer more effectively to detection and localization tasks.

Building upon the enriched feature space from modified-SimCLR, we explored the integration with Faster R-CNN [90].

Building upon the enriched feature space from modified-SimCLR, we explored the integration with Faster R-CNN [90]. This integration involved adapting the feature layers of modified-SimCLR to serve as the backbone of Faster R-CNN, facilitating multi-object detection.

The integration replaced the conventional convolutional backbone of Faster R-CNN with contextualized feature representation as illustrated in Fig. 6.2, resulting in improved multi-object detection performance compared to conventional Faster R-CNN models that relied solely on standard convolutional features.

We assessed the impact of our proposed approach on object detection using the COCO dataset [70]. The results indicated improvements compared to traditional random augmentation, showcasing the potential of considering context-aware augmentation to enhance the representation quality in SSRL performance.

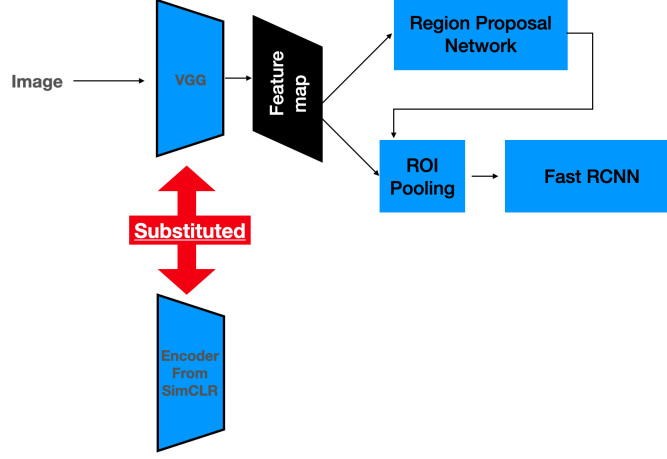


Figure 6.2: Employing the feature representation obtained from Modified-SimCLR (illustrated in Fig. 6.1), we retain the encoders and then discard the MLP heads in order to be used as the backbone for FasterRCNN. The rest of the network remains unchanged, following the original Faster-RCNN architecture [90].

6.4 Results and Discussion

In this section, we delve into the results obtained from our experiments, shedding light on the performance of our proposed method. Following that, we present a discussion and draw conclusions based on the outcomes.

6.4.1 Results

Our proposed approach to utilizing context-aware augmentation yielded improvement in classification accuracy compared to traditional augmentation methods for the SimCLR method. In addition, transferring the obtained feature representation to the multi-object detection downstream task resulted in enhancement comparing to transferring the feature representation obtained by traditional random augmentation-based SimCLR. These two experiments demonstrate the efficacy of the proposed method in enhancing the quality of feature representation. We achieved the following results as shown in Table 6.1 and Table 6.2 , showcasing the effectiveness of our contextually informed augmentation technique.

Table 6.1: Results on classification using SimCLR and Modified-SimCLR.

On ImageNet	SimCLR	Modified-SimCLR
Top1 Train Accuracy	70.8632	71.1293
Top1 Test Accuracy	67.1225 $\pm$ 0.2125	67.4751 $\pm$ 0.1237
Top5 Test Accuracy	96.9995 $\pm$ 0.0910	97.2857 $\pm$ 0.0749

Table 6.2: Results on multi-object detection using SimCLR and Modified-SimCLR feature representations as backbone. BB stands for Backbone.

On COCO	SimCLR_BB + FRCNN	Modified-SimCLR_BB + FRCNN
Average IoU	0.3172	0.3301
# of Detections	83/235	89/235
Detection Rate	0.3531	0.3787

It is noticeable that the results in the Table 6.2, illustrates that the performance in multi-object detection may not exhibit high accuracy. It is important to clarify that our objective is not primarily to propose an exceptional multi-object detector. Instead, we aim to showcase the superior quality of the feature representations derived from SimCLR.

The emphasis is on the quality of the feature representation rather than achieving peak performance in multi-object detection. Consequently, our approach prioritizes demonstrating the enhanced feature space quality through the SimCLR framework. It is noteworthy to mention that we did not extensively fine-tune the model on the downstream task dataset (COCO). By doing so, we aim to maintain reliability in the representations obtained from SimCLR trained on ImageNet, ensuring minimal alteration during the fine-tuning process.

6.4.2 Discussion

Our results demonstrate the efficacy of incorporating contextual information into the augmentation process for self-supervised learning. By leveraging object detection to guide the augmentation, we enhance the feature representation’s semantic meaning and context relevance.

The increase in classification accuracy signifies the potential of our approach to boost the performance of downstream tasks. Moreover, the reduction in variance indicates improved stability, essential for real-world applications. The enhancement in object detection performance further underscores the importance of contextually informed augmentation.

By aligning the feature space with object-level knowledge, we improve the detection accuracy and relevance to object-related tasks.

However, there are certain limitations to our approach. The effectiveness relies on the accuracy of the underlying object detection model. Noisy or inaccurate object detection can adversely affect the augmentation process. Albeit, our hunch is that, from another aspect it may increase the generalization power of the method which requires further research in that direction.

Our approach showcases promising results in enhancing the representation quality through contextually informed augmentation. Future work also will focus on refining our approach, exploring diverse datasets, and optimizing the integration of object detection for even better performance and broader applicability.

6.5 Implementation Details

Backbone & Head Encoder: ResNet-18 with `fc=Identity`. Projector: `Linear(512→512) + BN + ReLU + Linear(512→128) + BN (affine=false)`. Features for evaluation are taken *before* the projector.

Object Detector YOLOv8n (Ultralytics) pretrained on COCO; inference in train mode only. Default thresholds: `conf` $\sim$ Uniform[0.2, 0.6], `iou` = 0.5 (NMS). We keep the top-1 box per image (if any). If no box is found, we fall back to standard SimCLR augmentations.

Context-Aware View Given an image and its detected box $b = (x, y, w, h)$: (i) *Box jitter*: expand/contract by a random scale $s \sim \text{Uniform}[0.8, 1.25]$, clamp to image bounds. (ii) *Crop*: crop to the jittered box, then `RandomResizedCrop` to the target size. (iii) *Context mix*: with prob. $p_{\text{ctx}}=0.5$, blur or darken pixels outside the box before the crop (simple Gaussian blur or brightness $\times 0.5$). Finally apply standard SimCLR color/flip jitter. Each sample yields two views: one *random SimCLR* view and one *context-aware* view; with prob. 0.25 we use two random views to preserve diversity.

6.6 Conclusion

In this study, we introduced a novel approach to self-supervised representation learning by incorporating contextually informed augmentation. Our proposed method leverages object

detection to guide the augmentation process, creating a feature space intricately tied to the content and spatial relationships of objects within an image. The key contributions of our research include:

- **Contextual Augmentation:** We proposed a new augmentation technique that integrates object detection to generate more semantically meaningful and contextually relevant features, moving beyond traditional random image perturbations.
- **Improved Performance:** Our approach demonstrated improvements in classification accuracy and variance reduction compared to conventional augmentation methods. Moreover, it enhanced object detection performance, highlighting its potential in a broader range of computer vision tasks.
- **Stability and Consistency:** The reduction in variance indicates the stability and consistency our approach provides, making it a robust choice for practical applications.

Additionally, our research opens exciting avenues for future investigations:

- **Model Generalization:** Investigate further into the potential of our approach to generalize across various domains and datasets.
- **Semantic Segmentation Integration:** Explore incorporating object-level information into semantic segmentation tasks for improved segmentation accuracy.
- **Multi-Modal Data:** Extend our approach to handle multi-modal data, effectively integrating object detection across diverse data types.
- **Online Learning and Adaptability:** Develop a framework for online learning that adapts to changing object detection models and updates the feature space accordingly.

The integration of contextually informed augmentation into self-supervised representation learning is an advancement. By enriching the feature space with object-level knowledge, we bridge the gap between augmentation and the understanding of image content. This approach holds immense potential to enhance the quality and relevance of representations, ultimately leading to more effective and efficient machine learning models across various domains. As we continue to delve deeper into this paradigm, we are optimistic about its impact on the field of computer vision and beyond.

6.6.1 Future Work

To enhance the quality and transferability of the learned representations, future work will involve tuning and refining the hierarchical feature representation model. Incorporating object detection and optimizing the fine-grained and coarse-grained features is expected to yield more informed and elevated representations. Future evaluation will encompass full convergence to understand the true potential of our approach, considering aspects like learning curves, convergence speed, and generalization to unseen data. This comprehensive evaluation will provide valuable insights into the strengths and weaknesses of our method and guide future iterations and improvements.

Chapter 7

Manifold-Aware Batch-Level

7.1 Motivation

While the Variance-Invariance-Covariance Regularization (VICReg) framework has demonstrated substantial promise for self-supervised learning, its invariance objective is fundamentally pairwise, focusing on aligning representations only between pairs of augmented views. This approach, though effective at the local level, leaves a gap in ensuring global or batch-level consistency, especially when handling complex datasets with diverse transformations. In scenarios where the model must be robust to a wide variety of augmentations and data shifts, pairwise alignment can fall short: it may fail to enforce broader agreement among all augmented samples within a batch, leading to a less cohesive feature space and, ultimately, to reduced robustness and generalizability in the learned representations.

The lack of batch-level alignment may also hinder the model’s ability to fully capture the geometry of the underlying data manifold. Recent literature has emphasized that richer, more transferable features are often those that exhibit not just pairwise similarity, but a global alignment in distribution across multiple augmentations or views. As such, advancing beyond pairwise invariance is a critical next step in enhancing manifold awareness and representation stability in self-supervised learning.

7.2 Core Idea

To bridge this gap, we propose an extension to VICReg by integrating a Maximum Mean Discrepancy (MMD) term into the objective. MMD, as a kernel-based distribution align-

ment technique, enables comparison and alignment of entire feature distributions in a Reproducing Kernel Hilbert Space (RKHS), thus naturally supporting both linear and non-linear dependencies. By incorporating MMD into the VICReg framework, we move from a strictly pairwise alignment strategy to one that enforces batch-level consistency—encouraging the learned representations from all augmented samples in a batch to conform to a unified distribution. This approach enhances invariance across diverse transformations, reduces the risk of overfitting to individual augmentations, and encourages the network to capture the intrinsic structure of the data manifold.

In the following methodology section, we detail the formulation of the combined VICReg-MMD loss, discuss the theoretical motivation for batch-level alignment, and outline the practical considerations for kernel selection and implementation. This forms the basis for a more robust, manifold-aware self-supervised learning paradigm.

7.3 Methodology

Our proposed approach integrates Maximum Mean Discrepancy (MMD) [15] into VICReg’s to address the limitations of pairwise invariance by enforcing consistency across all samples in a batch. This section details the formulation of our objective function and the theoretical motivation behind using MMD to strengthen invariance in VICReg.

7.3.1 Objective: VICReg with MMD-Enhanced Invariance

VICReg aims to produce invariant features by minimizing the distance between representations of paired augmentations of the same input. However, this pairwise approach may be limited in capturing broader batch-level dependencies [60] and does not fully account for consistency across different transformations within a batch. MMD addresses this by aligning feature distributions across all augmented views in the batch.

To achieve this, we incorporate an MMD term into VICReg’s loss, forming an objective function that balances both pairwise and batch-level alignment:

$$\mathcal{L} = \lambda \mathcal{L}_{\text{inv}} + \mu \mathcal{L}_{\text{var}} + \nu \mathcal{L}_{\text{cov}} + \alpha \mathcal{L}_{\text{MMD}}, \quad (7.1)$$

where $\mathcal{L}_{\text{inv}}$ denotes the original pairwise invariance in VICReg, and $\mathcal{L}_{\text{MMD}}$ is the MMD term that aligns feature distributions across the batch. The parameter α controls the influence of MMD relative to other terms, balancing pairwise invariance with batch-level alignment.

7.3.2 Theoretical Motivation for MMD-Enhanced Invariance in VICReg

The addition of MMD to VICReg’s objective function offers several theoretical benefits:

Batch-Level Invariance Unlike pairwise objectives that enforce invariance only between specific views, MMD aligns all augmented samples in the batch. This caused a unified feature space where diverse transformations are consistently mapped. This batch-level consistency smooths variations among representations and contributes to a more robust, transformation-invariant representation [44].

Improved Generalization and Domain Adaptation By aligning feature distributions in a batch of different augmentations, MMD promotes a representation that is less susceptible to overfitting on individual transformations. This supports better generalization for unseen transformations [113]. Additionally, MMD has been shown to improve domain adaptability by aligning feature distributions in a manner that is stable across domains. It makes the learned representations more transferable across data settings [74].

Manifold Alignment and Higher-Order Dependencies MMD’s alignment in RKHS allows the model to capture both linear and non-linear dependencies across the batch, supporting a manifold-aware alignment of features that respects the intrinsic geometry of the data space. This is especially beneficial for complex, non-linear datasets, where capturing higher-order dependencies aids in representing the underlying structure of the data [44].

7.3.3 Parameter Selection and Kernel Choice

The effectiveness of MMD is sensitive to the choice of kernel. In our experiments, we employ a Gaussian kernel:

$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right), \quad (7.2)$$

where σ controls the scale of the kernel, affecting the degree to which local versus global dependencies are captured. Choosing an appropriate σ is critical for the effectiveness of MMD. A smaller σ emphasizes local dependencies, making MMD sensitive to small-scale

variations. A larger σ captures global dependencies and smooths out finer details. To balance these effects, we selected σ empirically testing a range of values. After experimentation, we selected $\sigma = 1$ as it provided the best results on the validation set, demonstrating a balanced contribution of MMD to batch-level alignment while maintaining the effectiveness of VICReg’s original objectives.

Combined Impact The combined VICReg-MMD objective leverages both pairwise and batch-level alignment. By enhancing batch-level invariance through MMD, our method achieves representations that are robust to diverse transformations and resilient to domain variations. This dual approach provides a balanced regularization strategy, yielding high-quality self-supervised representations suited for downstream tasks.

7.4 Experiments

To assess the effectiveness of our proposed VICReg with MMD-enhanced invariance, we conduct experiments on MNIST, CIFAR-10, and STL-10 datasets. These datasets vary in complexity, providing a robust evaluation of our method’s ability to capture invariance across simple and complex data distributions. We measure the quality of the learned representations by evaluating their performance on linear classification, and clustering accuracy. For evaluating the learned representations, we used a Multi-Layer Perceptron (MLP) classifier with two fully connected layers. The first layer maps the input dimension to 256 features, followed by a ReLU activation, and the second layer maps to 10 classes. For clustering, we used the k-Nearest Neighbors (k-NN) algorithm [49]. Our experiments are designed to highlight how batch-level alignment with MMD improves representation consistency and robustness compared to baseline VICReg.

7.4.1 Experimental Setup

We implemented a lightweight version of the VICReg architecture, using a three-layer convolutional network as the backbone. Each image in these datasets is processed through standard data augmentations, including random cropping, flipping, and color jittering. These augmentations introduce variations that our model learns to be invariant to, with MMD contributing to the alignment of the feature distributions across all augmented samples in the batch.

Dataset	VICReg		VICReg + MMD	
	Classifier	Clustering	Classifier	Clustering
MNIST	88.67%	81.65%	91.68%	88.26%
CIFAR-10	59.73%	48.96%	61.96%	50.36%
STL-10	53.78%	40.66%	60.22%	42.26%

Table 7.1: Performance comparison of VICReg and VICReg + MMD on MNIST, CIFAR-10, and STL-10, evaluated by linear classifier and clustering accuracy.

7.4.2 Results and Analysis

The results in Table 7.1 indicate that incorporating MMD into VICReg consistently improves representation quality across all datasets. The most significant improvements are observed on the STL-10 dataset, which contains high-dimensional and complex images, where MMD-enhanced VICReg achieves a classifier accuracy increase from 53.78% to 60.22% and a clustering accuracy increase from 40.66% to 42.26%. This improvement suggests that MMD’s batch-level alignment is especially effective in scenarios where invariance across complex transformations is critical.

Batch-Level Consistency Integrating MMD into VICReg enhances clustering performance, as demonstrated by improved metrics on complex datasets like CIFAR-10 and STL-10. This aligns with prior work highlighting the effectiveness of batch-level distribution alignment in capturing richer feature representations [44]. By enforcing consistency across the entire batch in a Reproducing Kernel Hilbert Space (RKHS), MMD reduces variability among augmentations, which is theoretically supported to promote more cohesive clustering [99]. While the observed increase in clustering accuracy provides empirical evidence of this benefit, further exploration is needed to fully characterize the relationship between batch-level alignment and feature space stability in diverse self-supervised settings.

Improved Robustness and Generalization The benefits of MMD’s batch-level alignment are evident in the linear classification results. By ensuring that augmented samples across the batch map to a cohesive feature distribution, MMD reduces variability, which enhances feature robustness and generalizability across transformations. This improvement is consistent across datasets, highlighting the advantage of enforcing batch-level invariance in a variety of data contexts.

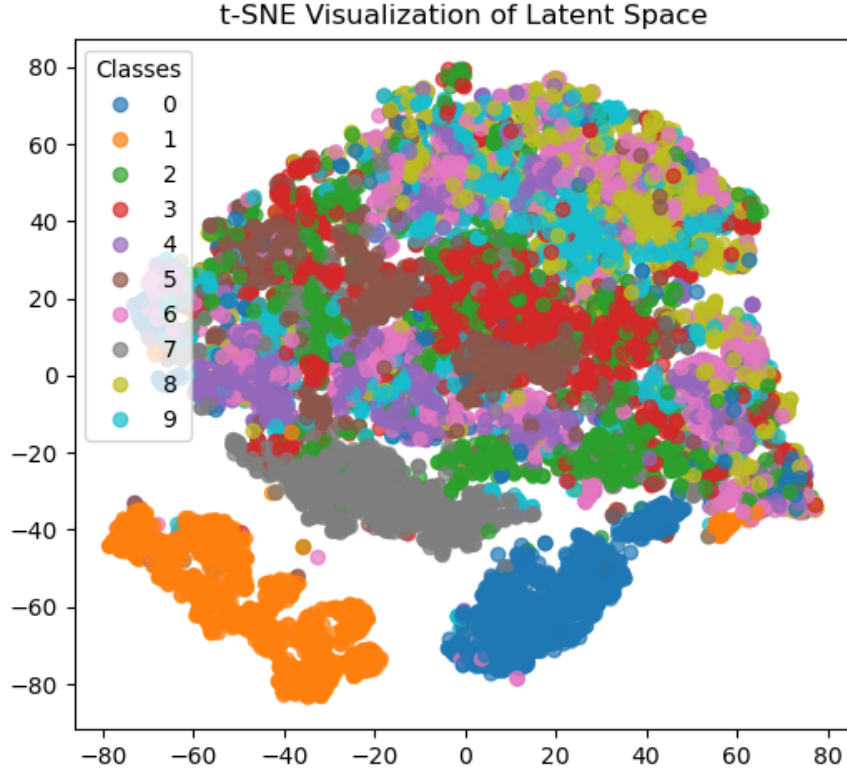


Figure 7.1: t-SNE visualization of MNIST representations with standard VICReg. The clusters are less distinct compared to the MMD-enhanced approach.

7.4.3 t-SNE Visualization

To illustrate the impact of MMD on representation structure, we visualize the learned features using t-SNE on the MNIST dataset. As shown in Figures 7.1 and 7.2, representations learned by VICReg + MMD exhibit more distinct clusters with less overlap between classes compared to the baseline VICReg. This improvement in cluster separability suggests that MMD’s batch-level alignment supports a more organized feature space, making the learned representations more interpretable and easier to classify.

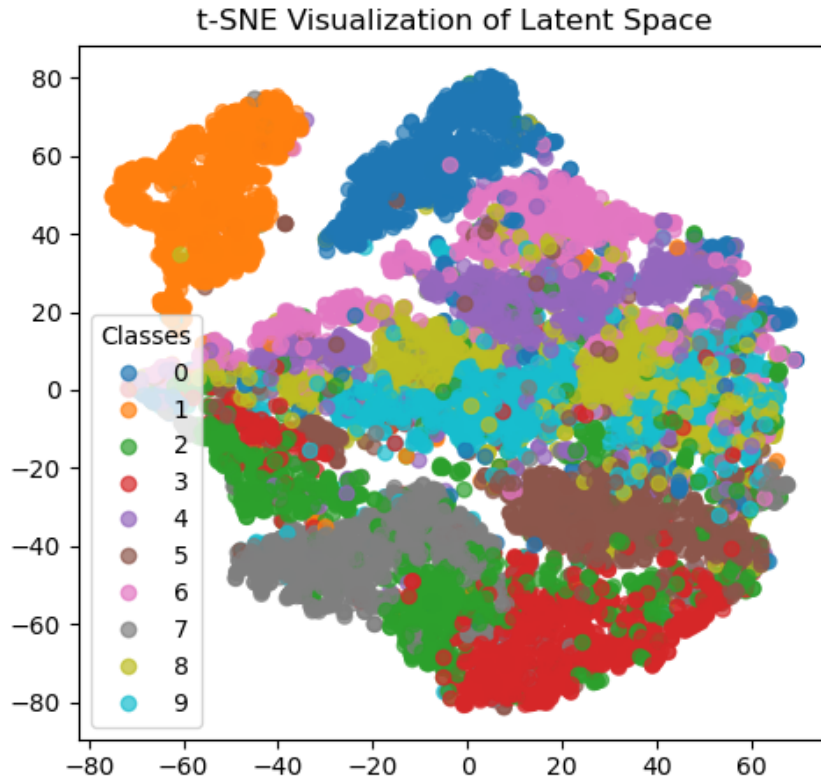


Figure 7.2: t-SNE visualization of MNIST representations with VICReg + MMD. The addition of MMD leads to more distinct, well-separated clusters.

7.4.4 Ablation Study on MMD Influence

We further conducted an ablation study to analyze the effect of the MMD term’s weight (α) on representation quality. Figure 7.3 shows that as α increased, clustering accuracy improved up to a certain threshold, beyond which performance plateaued. This suggests that MMD’s influence is most effective when balanced with other VICReg objectives, reinforcing that batch-level alignment enhances invariance without overwhelming pairwise alignment.

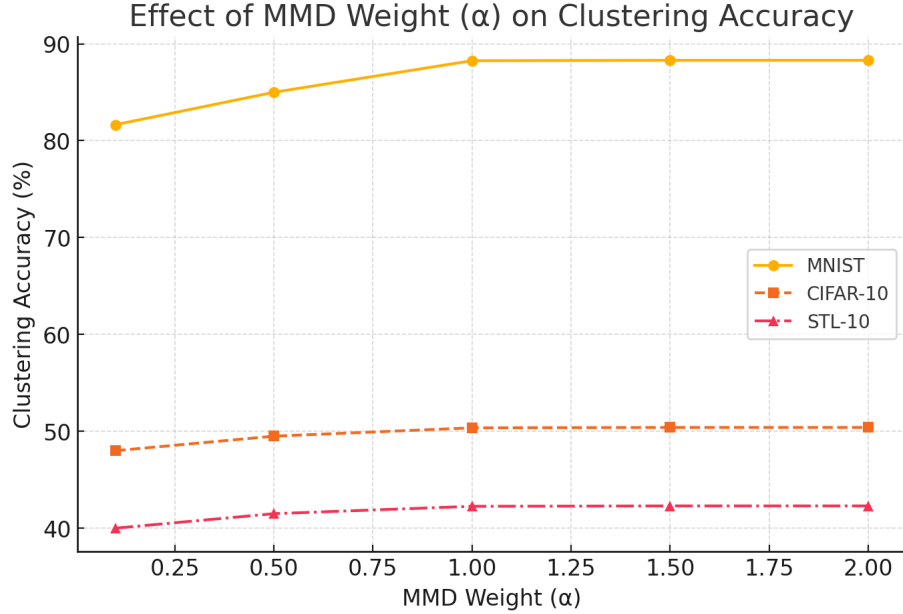


Figure 7.3: MMD weights vs clustering accuracy

7.4.5 Summary of Findings

Our experimental results validate the hypothesis that incorporating MMD into VICReg enhances representation quality by strengthening batch-level invariance. MMD’s ability to align feature distributions across the batch yields representations that are not only more invariant but also better suited for downstream tasks, particularly in complex data contexts. These results highlight the importance of batch-level consistency in self-supervised learning, demonstrating MMD’s role as a valuable addition to VICReg for manifold-aware, robust representation learning.

7.5 Implementation Details

Datasets & Augs We use **MNIST**, **CIFAR-10** and **STL-10**. SSL augs: `RandomResizedCrop`, `HorizontalFlip`, `ColorJitter(0.4,0.4,0.4,0.1)`, `RandomGrayscale(0.2)`, `ToTensor`, `Normalize`. MNIST is replicated to 3 channels for SSL; linear probe uses plain normalization.

Backbone & Heads ResNet-18 encoder (`fc=Identity`). Projector: `Linear(512→512)` + BN + ReLU, then `Linear(512→128)` + BN (`affine=false`). Linear probe: `Linear(512→C)` on frozen features.

Objectives Total loss: $\mathcal{L} = \mathcal{L}_{\text{NT-Xent}} + \lambda_{\text{MMD}} \cdot \text{MMD}^2$. NT-Xent on ℓ_2 -normalized projections (temperature τ). MMD on *encoder* features using an RBF kernel with median-heuristic bandwidth (per-batch).

Kernel Bands (defaults) RBF: $\sigma = \text{median distance}$, scale $s \in \{0.5, 1.0, 2.0\}$ (default $s=1.0$).

Optimization Adam (lr 1×10^{-3} , wd 1×10^{-6}). Batch size: 512 for MNIST/CIFAR-10/100; 256 for STL-10. Pretrain 300 epochs; linear probe 20 epochs. Defaults: $\tau=0.5$, $\lambda_{\text{MMD}}=0.1$.

7.6 Conclusion

We introduced an enhancement to VICReg by incorporating a Maximum Mean Discrepancy (MMD) term to address the limitations of pairwise invariance in self-supervised representation learning. By aligning feature distributions across the batch in a Reproducing Kernel Hilbert Space (RKHS), MMD promotes batch-level consistency, resulting in a unified and stable feature space.

Our experiments on MNIST, CIFAR-10, and STL-10 show that MMD improves representation quality, as evidenced by higher linear classification accuracy and enhanced clustering performance. These results underscore the benefits of capturing higher-order dependencies and suggest that batch-level alignment contributes to more robust and generalizable features.

Future work could explore MMD’s integration into other self-supervised frameworks, potentially enhancing the robustness of methods like SimCLR or BYOL. Additionally, testing this approach on larger and more complex datasets would further reveal the role of batch-level invariance. Our VICReg-MMD approach lays the groundwork for more effective manifold-aware representations, supporting a wide range of downstream applications.

Chapter 8

Conclusion

This thesis has advanced the theoretical foundations and practical methodologies of self-supervised learning (SSL), with a unifying perspective rooted in the *manifold hypothesis* and its implications for representation learning. By framing diverse SSL paradigms through the lens of geometric, statistical, and spectral constraints, we have established a coherent narrative in which each contribution addresses a distinct yet interconnected facet of the representation learning problem.

Summary of Contributions

We began by formalizing the geometric underpinnings of SSL, emphasizing that high-dimensional observations often reside on a smooth, low-dimensional manifold embedded in $\mathbb{R}^d$. Within this view, the quality of learned representations hinges on preserving manifold structure, both locally, via approximate isometry constraints, and globally, through spectral regularization. This perspective guided the design of methods that explicitly enforce desirable properties such as variance preservation, decorrelation, independence, and invariance at both the pairwise and batch levels.

Building on this theoretical grounding, we introduced novel regularization strategies that extend beyond linear statistical constraints. In particular:

- We proposed **Sinkhorn-Regularized SimCLR (SinSim)** [Chapter 3](#), integrating entropy-regularized optimal transport into a contrastive learning framework to enforce global geometric structure in the representation space while preserving discriminative power.

- We developed **Correlation–Dependence Self-Supervised Learning (CDSSL)** [Chapter 4](#), unifying linear correlation and nonlinear dependence regularization via the Hilbert–Schmidt Independence Criterion (HSIC), at both the feature- and sample-level, to enhance diversity and disentanglement in representations.
- We extended non-contrastive SSL into the **kernel domain** with **Kernel VICReg** [Chapter 5](#), reformulating variance, invariance, and covariance terms in a Reproducing Kernel Hilbert Space (RKHS) to capture complex, nonlinear dependencies.
- We incorporated **Maximum Mean Discrepancy (MMD)** into a VICReg-style framework for **manifold-aware batch-level alignment** [Chapter 7](#), encouraging invariance across entire augmented distributions rather than isolated pairs.
- We introduced a **context-aware augmentation** [Chapter 6](#) strategy within a SimCLR framework to preserve semantically meaningful content, improving robustness and invariance quality.

These contributions were systematically evaluated across benchmark datasets (e.g., MNIST, CIFAR-10/100, STL-10, ImageNet variants), showing consistent improvements in linear and non-linear evaluation, transfer learning, and robustness under distributional shifts. The methods proposed here not only achieve superior performance but also align with a principled understanding of manifold geometry and statistical dependency in SSL.

A Unified Geometry-Preserving Perspective on SSL

Across the diverse contributions presented in this thesis, ranging from optimal transport, regularized contrastive methods to kernelized dependency reduction—we can now reinterpret each as an instantiation of a broader *geometry-preserving self-supervised learning* (GP-SSL) framework. The core insight is that the effectiveness of an SSL method is determined not solely by its invariance objectives, but by the extent to which it aligns the learned representation space with the intrinsic geometry of the data manifold.

We formalize GP-SSL through the composite objective:

$$\mathcal{L}_{\text{GP-SSL}} = \mathcal{L}_{\text{inv}} + \lambda \mathcal{R}_{\text{dep}} + \mu \mathcal{R}_{\text{geo}}, \quad (8.1)$$

where:

- $\mathcal{L}_{\text{inv}}$ promotes invariance across positive pairs, enforcing semantic consistency.

- $\mathcal{R}_{\text{dep}}$ reduces statistical dependence among representation components, preventing collapse and encouraging diversity. In our work, this term has been instantiated as covariance minimization, HSIC-based nonlinear dependency reduction, and sparsity-inspired orthogonalization.
- $\mathcal{R}_{\text{geo}}$ preserves manifold structure under the learned embedding, as quantified by a geometry-aware metric. This term is where our contributions depart most strongly from the Euclidean default, introducing optimal transport distances, MMD-based kernel metrics, and manifold-adaptive similarity measures.

By articulating this shared structure, we reveal that seemingly disparate SSL innovations are in fact variations on a single principle: *learn invariances while preserving manifold geometry, using metrics and dependency measures that are themselves aligned with the data’s intrinsic structure*. This perspective lays the groundwork for systematic exploration of metric choice, dependency measures, and geometry-aware regularization in the next generation of SSL methods.

Broader Implications

Beyond empirical gains, this work offers a conceptual unification of diverse SSL approaches. By interpreting contrastive and non-contrastive paradigms as instances of manifold-preserving regularization under different statistical operators, we have shown that many seemingly disparate methods share a common theoretical foundation. This unified view facilitates a modular design philosophy: invariance objectives, dependency regularizers, and manifold priors can be composed and tuned to the topology of a target domain.

A further conceptual implication of this thesis is the questioning of an implicit yet pervasive assumption in self-supervised learning, that Euclidean distance (or its normalized form, cosine similarity) is the canonical metric for representation alignment. Such a choice presupposes that the embedding space is flat, homogeneous, and globally isotropic, conditions that rarely hold when the data manifold is curved, multi-scale, or topologically complex. This mismatch can result in geometric distortions whereby semantically distant points are brought artificially close, or vice versa. By incorporating kernelized dependence measures, optimal transport distances, and batch-level distributional alignment, the methods developed in this work move beyond the Euclidean paradigm, allowing the geometry of the learned space to adapt to the intrinsic structure of the data. This shift reflects a broader principle: the metric itself is a design choice, and its alignment with the manifold’s geometry is as crucial as the choice of the learning objective.

It also underscores the importance of geometry-aware evaluation metrics that extend beyond accuracy, measuring how faithfully learned embeddings respect the intrinsic structure of the data manifold, an essential step toward explainable and trustworthy representation learning.

Limitations and Open Theoretical Questions

While the methods and analyses presented in this thesis advance the state of self-supervised representation learning, several foundational questions remain unresolved. At a theoretical level, a precise characterization of the relationship between manifold preservation and generalization performance in high-dimensional, overparameterized models is still lacking. Current understanding is largely heuristic: we postulate that preserving task-relevant manifold geometry aids transfer and robustness, yet formal generalization bounds that explicitly depend on geometric distortion measures—such as geodesic deviation, tangent-space distortion, or spectral perturbations of the Laplace–Beltrami operator—remain underdeveloped.

Moreover, the definition of “task-relevant geometry” itself is inherently ambiguous. SSL seeks to construct embeddings invariant to nuisance transformations while sensitive to semantically meaningful variation; however, identifying this optimal invariance–sensitivity trade-off without access to downstream labels is a fundamentally ill-posed problem. In the absence of a principled criterion, current methods implicitly encode this balance through the choice of augmentations, kernels, and regularizers—decisions that remain more art than science.

Another open frontier lies in bridging local and global geometric constraints within a unified analytical framework. While local isometry ensures the preservation of neighborhood structure, and batch-level distributional alignment enforces global coherence, the interplay between these scales is not yet mathematically formalized. Similarly, while kernel methods and optimal transport provide powerful tools for aligning nonlinear structures, a comprehensive theory describing when these operations preserve or distort the manifold’s intrinsic topology is still missing.

Finally, the broader integration of manifold-aware SSL with theories of causal representation learning, information geometry, and statistical decision theory remains an aspirational goal. In particular, a deep connection likely exists between the Fisher–Rao metric of statistical manifolds, the geometry induced by SSL objectives, and the downstream Bayes risk; yet formalizing this link demands an interdisciplinary synthesis that the present work

only begins to suggest. Addressing these questions is essential for elevating SSL from an empirically driven field to one grounded in predictive, quantitative theory.

Future Directions

The research presented here opens multiple avenues for further investigation:

- **Adaptive manifold regularization.** Automatically selecting or learning kernels, transport costs, or dependency measures tailored to the data’s geometry. Extending the kernel selection strategies first introduced in Chapter 5.
- **Bridging generative and discriminative SSL.** Using generative priors, such as diffusion models, as manifold samplers to regularize discriminative embeddings. Building on the augmentation philosophy of Chapter 6, but extending it with generative priors.
- **Task-specific manifold shaping.** Extending these methods to domains where the manifold is partially known or constrained, such as scientific imaging or biophysical surfaces. Following the geometry-preserving strategies of SinSim (Chapter 3) and MMD-based SSL (Chapter 7).
- **Rank-aware regularization.** Incorporating spectral and rank constraints on representation matrices to further prevent collapse and encourage expressive embeddings.
- **Scalable spectral analysis.** Designing efficient approximations for eigenvalue-based regularizers suitable for large-scale models and datasets.

Closing Remarks

The central message of this thesis is that *Representation learning seeks to preserve the intrinsic geometry of the data manifold in a form that is optimally aligned with the invariances and discriminative structure required by downstream tasks*. By grounding SSL objectives in the mathematics of manifolds, kernels, and optimal transport, we provide both a theoretical lens and a practical toolkit for building models whose embeddings are not only discriminative but also faithful to the underlying data structure. As SSL continues to shape the next generation of machine learning systems, we hope that this work serves as a foundation for methods that are at once more interpretable, more robust, and more principled.

References

- [1] Wilhelm Ågren. The nt-xent loss upper bound. *arXiv preprint arXiv:2205.03169*, 2022.
- [2] Amer FAH Alnuaimi and Tasnim HK Albaldawi. An overview of machine learning classification techniques. In *BIO Web of Conferences*, volume 97, page 00133. EDP Sciences, 2024.
- [3] Patricia Álvarez, Miguel A Fernández, Alfonso Gordaliza, Alberto Mansilla, and Aquilino Molinero. Geometric road design factors affecting the risk of urban run-off crashes. a case-control study. *PLoS one*, 15(6):e0234564, 2020.
- [4] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. *arXiv preprint arXiv:1911.05371*, 2019.
- [5] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15619–15629, 2023.
- [6] Randall Balestriero and Yann LeCun. Contrastive and non-contrastive self-supervised learning recover global and local spectral embedding methods. *Advances in Neural Information Processing Systems*, 35:26671–26685, 2022.
- [7] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.
- [8] Adrien Bardes, Jean Ponce, and Yann Lecun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR 2022-International Conference on Learning Representations*, 2022.

- [9] Horace B Barlow. Unsupervised learning. *Neural computation*, 1(3):295–311, 1989.
- [10] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.
- [11] Joseph Berkson. Application of the logistic function to bio-assay. *Journal of the American statistical association*, 39(227):357–365, 1944.
- [12] Alain Berlinet and Christine Thomas-Agnan. *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media, 2011.
- [13] Irving Biederman. Human image understanding: Recent research and a theory. *Computer vision, graphics, and image processing*, 32(1):29–73, 1985.
- [14] Alexander I Bobenko and Boris A Springborn. A discrete laplace–beltrami operator for simplicial surfaces. *Discrete & Computational Geometry*, 38(4):740–756, 2007.
- [15] Karsten M Borgwardt, Arthur Gretton, Malte J Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alex J Smola. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14):e49–e57, 2006.
- [16] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.
- [17] Jian Chen and Song Lin. A neural network approach-decision neural network (dnn) for preference assessment. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 34(2):219–225, 2004.
- [18] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PmLR, 2020.
- [19] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15745–15753. IEEE Computer Society, 2021.
- [20] KR1442 Chowdhary. Natural language processing. *Fundamentals of artificial intelligence*, pages 603–649, 2020.

- [21] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.
- [22] Richard M Cormack. A review of classification. *Journal of the Royal Statistical Society: Series A (General)*, 134(3):321–353, 1971.
- [23] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
- [24] Pádraig Cunningham, Matthieu Cord, and Sarah Jane Delany. Supervised learning. In *Machine learning techniques for multimedia: case studies on organization and retrieval*, pages 21–49. Springer, 2008.
- [25] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.
- [26] Marco Cuturi and Arnaud Doucet. Fast computation of wasserstein barycenters. In *International conference on machine learning*, pages 685–693. PMLR, 2014.
- [27] Charles Dapogny, Franck Iutzeler, Andrea Meda, and Boris Thibert. Entropy-regularized wasserstein distributionally robust shape and topology optimization. *Structural and Multidisciplinary Optimization*, 66(3):42, 2023.
- [28] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE, 2009.
- [29] Devjit Dey, Md Samio Haque, Md Mojahedul Islam, Umme Iffat Aishi, Sajida Sultana Shammy, Md Sabbir Ahmed Mayen, Syed Toukir Ahmed Noor, and Md Jamal Uddin. The proper application of logistic regression model in complex survey data: a systematic review. *BMC Medical Research Methodology*, 25(1):15, 2025.
- [30] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 1422–1430, 2015.
- [31] Elizar Elizar, Mohd Asyraf Zulkifley, Rusdha Muharar, Mohd Hairi Mohd Zaman, and Seri Mastura Mustaza. A review on multiscale-deep-learning applications. *Sensors*, 22(19):7384, 2022.

- [32] Linus Ericsson, Henry Gouk, Chen Change Loy, and Timothy M Hospedales. Self-supervised representation learning: Introduction, advances, and challenges. *IEEE Signal Processing Magazine*, 39(3):42–62, 2022.
- [33] Kilian Fatras, Thibault Séjourné, Rémi Flamary, and Nicolas Courty. Unbalanced minibatch optimal transport; applications to domain adaptation. In *International conference on machine learning*, pages 3186–3197. PMLR, 2021.
- [34] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016.
- [35] Manuel Fernández-Delgado, Manisha Sanjay Sirsat, Eva Cernadas, Sadi Alawadi, Senén Barro, and Manuel Febrero-Bande. An extensive experimental survey of regression methods. *Neural Networks*, 111:11–34, 2019.
- [36] Rémi Flamary, Cédric Févotte, Nicolas Courty, and Valentin Emiya. Optimal spectral transportation with application to music transcription. *Advances in Neural Information Processing Systems*, 29, 2016.
- [37] Benyamin Ghojogh, Mark Crowley, Fakhri Karray, and Ali Ghodsi. Background on kernels. *Elements of Dimensionality Reduction and Manifold Learning*, pages 43–73, 2023.
- [38] Benyamin Ghojogh, Mark Crowley, Fakhri Karray, and Ali Ghodsi. Principal component analysis. In *Elements of Dimensionality Reduction and Manifold Learning*, pages 123–154. Springer, 2023.
- [39] Benyamin Ghojogh, Ali Ghodsi, Fakhri Karray, and Mark Crowley. Reproducing kernel hilbert space, mercer’s theorem, eigenfunctions, nyström method, and use of kernels in machine learning: Tutorial and survey. *arXiv preprint arXiv:2106.08443*, 2021.
- [40] Benyamin Ghojogh, Fakhri Karray, and Mark Crowley. Eigenvalue and generalized eigenvalue problems: Tutorial. *arXiv preprint arXiv:1903.11240*, 2019.
- [41] Nikolaos Giakoumoglou, Tania Stathaki, and Athanasios Gkelias. A review on discriminative self-supervised learning methods in computer vision. *arXiv preprint arXiv:2405.04969*, 2024.
- [42] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

- [43] Daniel Greenfeld and Uri Shalit. Robust learning with the hilbert-schmidt independence criterion. In *International Conference on Machine Learning*, pages 3759–3768. PMLR, 2020.
- [44] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.
- [45] Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with Hilbert-Schmidt norms. In *International conference on algorithmic learning theory*, pages 63–77. Springer, 2005.
- [46] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.
- [47] Jie Gui, Tuo Chen, Jing Zhang, Qiong Cao, Zhenan Sun, Hao Luo, and Dacheng Tao. A survey on self-supervised learning: Algorithms, applications, and future trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [48] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017.
- [49] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. Knn model-based approach in classification. In *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings*, pages 986–996. Springer, 2003.
- [50] Martin Hanik. *Geometric data analysis: Advancements of the statistical methodology and applications*. Freie Universitaet Berlin (Germany), 2023.
- [51] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.
- [52] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the*

IEEE/CVF conference on computer vision and pattern recognition, pages 9729–9738, 2020.

- [53] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.
- [54] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. *Technologies*, 9(1):2, 2020.
- [55] Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, and Jonathan Taylor. Unsupervised learning. In *An introduction to statistical learning: with applications in Python*, pages 503–556. Springer, 2023.
- [56] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. YOLO by Ultralytics, January 2023.
- [57] Leonid Kantorovich. On the transfer of masses (in russian). In *Doklady Akademii Nauk*, volume 37, page 227, 1942.
- [58] Siavash Khodadadeh, Ladislau Boloni, and Mubarak Shah. Unsupervised meta-learning for few-shot image classification. *Advances in neural information processing systems*, 32, 2019.
- [59] Philip A Knight. The sinkhorn–knopp algorithm: convergence and applications. *SIAM Journal on Matrix Analysis and Applications*, 30(1):261–275, 2008.
- [60] Panagiotis Koromilas, Giorgos Bouritsas, Theodoros Giannakopoulos, Mihalisis Nicolaou, and Yannis Panagakis. Bridging mini-batch and asymptotic analysis in contrastive learning: From infonce to kernel-based losses. *arXiv preprint arXiv:2405.18045*, 2024.
- [61] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, ON, Canada, 2009.
- [62] Pranjal Kumar, Piyush Rawat, and Siddhartha Chauhan. Contrastive self-supervised learning: review, progress, challenges and future research directions. *International Journal of Multimedia Information Retrieval*, 11(4):461–488, 2022.
- [63] Thomas R Kurfess et al. *Robotics and automation handbook*, volume 414. CRC press Boca Raton, FL, 2005.

- [64] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.
- [65] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [66] John M Lee. *Introduction to Riemannian manifolds*, volume 2. Springer, 2018.
- [67] Longchuan Li, Chaoyue Zhao, Shuqian He, Qiukai Qi, Shuai Kang, and Shugen Ma. Enhancing undulation of soft robots in granular media: A numerical and experimental study on the effect of anisotropic scales. *Biomimetic Intelligence and Robotics*, page 100158, 2024.
- [68] Yazhe Li, Roman Pogodin, Danica J Sutherland, and Arthur Gretton. Self-supervised learning with kernel dependence maximization. *Advances in Neural Information Processing Systems*, 34:15543–15556, 2021.
- [69] Zhehao Liang, Yanqing Luo, Marvin Beese, and David Jacob Drexlin. Multiple positive views in self-supervised learning. *openreview*, 2024.
- [70] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.
- [71] Bing Liu. Supervised learning. In *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*, pages 63–132. Springer, 2011.
- [72] Peng Liu, Lizhe Wang, Rajiv Ranjan, Guojin He, and Lei Zhao. A survey on active deep learning: From model driven to data driven. *ACM Computing Surveys (CSUR)*, 54(10s):1–34, 2022.
- [73] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. Self-supervised learning: Generative or contrastive. *IEEE transactions on knowledge and data engineering*, 35(1):857–876, 2021.
- [74] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015.

- [75] Rachel Macrorie, Simon Marvin, and Aidan While. Robotics and automation in the city: a research agenda. *Urban Geography*, 42(2):197–217, 2021.
- [76] Anton Mallasto, Augusto Gerolin, and Hà Quang Minh. Entropy-regularized 2-wasserstein distance between gaussian measures. *Information Geometry*, 5(1):289–323, 2022.
- [77] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.
- [78] Sebastian Mika, Gunnar Ratsch, Jason Weston, Bernhard Scholkopf, and Klaus-Robert Mullers. Fisher discriminant analysis with kernels. In *Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. no. 98th8468)*, pages 41–48. IEEE, 1999.
- [79] MN Murty and Rashmi Raghava. Kernel-based svm. In *Support vector machines and perceptrons: Learning, optimization, classification, and application to social networks*, pages 57–67. Springer, 2016.
- [80] Vladimir Nasteski. An overview of the supervised machine learning methods. *Horizons. b*, 4(51-62):56, 2017.
- [81] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- [82] Mohammad Ali Bagheri Orumi, M Hadi Sepanj, Mahmoud Famouri, Zohreh Azimi-far, and Alexander Wong. Unsupervised deep shape from template. In *International Conference on Image Analysis and Recognition*, pages 440–451. Springer, 2019.
- [83] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.
- [84] Giorgio Patrini, Rianne Van den Berg, Patrick Forre, Marcello Carioni, Samarth Bhargav, Max Welling, Tim Genewein, and Frank Nielsen. Sinkhorn autoencoders. In *Uncertainty in Artificial Intelligence*, pages 733–743. PMLR, 2020.
- [85] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

- [86] Michaël Perrot, Nicolas Courty, Rémi Flamary, and Amaury Habrard. Mapping estimation for discrete optimal transport. *Advances in Neural Information Processing Systems*, 29, 2016.
- [87] Khiem Pham, Khang Le, Nhat Ho, Tung Pham, and Hung Bui. On unbalanced optimal transport: An analysis of sinkhorn algorithm. In *International Conference on Machine Learning*, pages 7673–7682. PMLR, 2020.
- [88] Roshan Prakash Rane, Edit Szügyi, Vageesh Saxena, André Ofner, and Sebastian Stober. Prednet and predictive coding: A critical review. In *Proceedings of the 2020 international conference on multimedia retrieval*, pages 233–241, 2020.
- [89] YCAP Reddy, P Viswanath, and B Eswara Reddy. Semi-supervised learning: A brief review. *Int. J. Eng. Technol*, 7(1.8):81, 2018.
- [90] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.
- [91] Wei-Qing Ren, Yu-Ben Qu, Chao Dong, Yu-Qian Jing, Hao Sun, Qi-Hui Wu, and Song Guo. A survey on collaborative dnn inference for edge intelligence. *Machine Intelligence Research*, 20(3):370–395, 2023.
- [92] P Rubenstein, B Schölkopf, and I Tolstikhin. Learning disentangled representations with wasserstein auto-encoders. In *6th International Conference on Learning Representations (ICLR 2018)*. OpenReview. net, 2018.
- [93] Amir Salimi, Mansour Ziaii, Ali Amiri, Mahdieh Hosseini Zadeh, Sadegh Karimpouli, and Mostafa Moradkhani. Using a feature subset selection method and support vector machine to address curse of dimensionality and redundancy in hyperion hyperspectral data classification. *The Egyptian Journal of Remote Sensing and Space Science*, 21(1):27–36, 2018.
- [94] Chayna Sarkar, Biswadeep Das, Vikram Singh Rawat, Julie Birdie Wahlang, Arvind Nongpiur, Iadarilang Tiewsoh, Nari M Lyngdoh, Debasmita Das, Manjunath Bidarolli, and Hannah Theresa Sony. Artificial intelligence and machine learning technology driven modern drug discovery and development. *International Journal of Molecular Sciences*, 24(3):2026, 2023.
- [95] Morgan A Schmitz, Matthieu Heitz, Nicolas Bonneel, Fred Ngole, David Coeurjolly, Marco Cuturi, Gabriel Peyré, and Jean-Luc Starck. Wasserstein dictionary learning:

- Optimal transport-based unsupervised nonlinear dictionary learning. *SIAM Journal on Imaging Sciences*, 11(1):643–678, 2018.
- [96] Bernhard Schölkopf. The kernel trick for distances. *Advances in neural information processing systems*, 13, 2000.
 - [97] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Kernel principal component analysis. In *International conference on artificial neural networks*, pages 583–588. Springer, 1997.
 - [98] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998.
 - [99] Bernhard Schölkopf and Alexander J Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.
 - [100] Hadi Sepanj and Paul Fieguth. Context-aware augmentation for contrastive self-supervised representation learning. *Journal of Computational Vision and Imaging Systems*, 9(1):4–7, 2023.
 - [101] Hadi Sepanj and Paul Fieguth. Aligning feature distributions in vicreg using maximum mean discrepancy for enhanced manifold awareness in self-supervised representation learning. *Journal of Computational Vision and Imaging Systems*, 10(1):13–18, 2024.
 - [102] M Hadi Sepanj and Paul Fiegth. Sinsim: Sinkhorn-regularized simclr. *arXiv preprint arXiv:2502.10478*, 2025.
 - [103] M Hadi Sepanj, Benyamin Ghogh, and Paul Fieguth. Self-supervised learning using nonlinear dependence. *arXiv preprint arXiv:2501.18875*, 2025.
 - [104] James B Simon, Maksis Knutins, Liu Ziyin, Daniel Geisz, Abraham J Fetterman, and Joshua Albrecht. On the stepwise nature of self-supervised learning. In *International Conference on Machine Learning*, pages 31852–31876. PMLR, 2023.
 - [105] Bharath K Sriperumbudur, Kenji Fukumizu, and Gert RG Lanckriet. Universality, characteristic kernels and rkhs embedding of measures. *Journal of Machine Learning Research*, 12(7), 2011.
 - [106] George Stockman and Linda G Shapiro. *Computer vision*. Prentice Hall PTR, 2001.

- [107] Joshua B Tenenbaum, Vin de Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.
- [108] Vayer Titouan, Nicolas Courty, Romain Tavenard, and Rémi Flamary. Optimal transport for structured data with application on graphs. In *International Conference on Machine Learning*, pages 6275–6284. PMLR, 2019.
- [109] Jesper E Van Engelen and Holger H Hoos. A survey on semi-supervised learning. *Machine learning*, 109(2):373–440, 2020.
- [110] Cédric Villani. The wasserstein distances. In *Optimal transport: old and new*, pages 93–111. Springer, 2009.
- [111] Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2008.
- [112] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008.
- [113] Wei Wang, Haojie Li, Zhengming Ding, Feiping Nie, Junyang Chen, Xiao Dong, and Zhihui Wang. Rethinking maximum mean discrepancy for visual domain adaptation. *IEEE Transactions on Neural Networks and Learning Systems*, 34(1):264–277, 2021.
- [114] Chuhan Wu, Fangzhao Wu, and Yongfeng Huang. Rethinking infonce: How many negative samples do you need? *arXiv preprint arXiv:2105.13003*, 2021.
- [115] Yangzheng Wu and Michael Greenspan. Pseudo-keypoint RKHS learning for self-supervised 6DoF pose estimation. In *European Conference on Computer Vision*, pages 37–56. Springer, 2025.
- [116] Dongkuan Xu and Yingjie Tian. A comprehensive survey of clustering algorithms. *Annals of data science*, 2(2):165–193, 2015.
- [117] Guoliang Xu. Discrete laplace–beltrami operators and their convergence. *Computer aided geometric design*, 21(8):767–784, 2004.
- [118] Mingle Xu, Sook Yoon, Alvaro Fuentes, and Dong Sun Park. A comprehensive survey of image augmentation techniques for deep learning. *Pattern Recognition*, page 109347, 2023.

- [119] Rui Xu and Donald Wunsch. Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3):645–678, 2005.
- [120] Yingxue Xu, Guihua Wen, Yang Hu, and Pei Yang. Modeling hierarchical structural distance for unsupervised domain adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(9):8753–8765, 2024.
- [121] Yihao Xue, Siddharth Joshi, Eric Gan, Pin-Yu Chen, and Baharan Mirzasoleiman. Which features are learnt by contrastive learning? on the role of simplicity bias in class collapse and feature suppression. In *International Conference on Machine Learning*, pages 38938–38970. PMLR, 2023.
- [122] Ming Yuan and T Tony Cai. A reproducing kernel hilbert space approach to functional linear regression. 2010.
- [123] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning*, pages 12310–12320. PMLR, 2021.
- [124] Guoqiang Peter Zhang. Neural networks for classification: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 30(4):451–462, 2002.
- [125] Xiaojin Jerry Zhu. Semi-supervised learning literature survey. 2005.
- [126] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2020.