Latent Boost: Enhancing Interpretability Through Loss-Defined Classification Objective in Structured Latent Spaces

Daniel Geißler[1,2], Bo Zhou[1,2], Mengxi Liu[1,2], and Paul Lukowicz[1,2]

[1] German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany
[2] RPTU Kaiserslautern-Landau, Kaiserslautern, Germany
daniel.geissler@dfki.de

Abstract. Supervised machine learning often operates on the data-driven paradigm, wherein internal model parameters are autonomously optimized to converge predicted outputs with the ground truth, devoid of explicitly programmed rules or a priori assumptions. Although data-driven methods have yielded notable successes across benchmark datasets, they inherently treat models as opaque entities, limiting interpretability and explanatory insights into their decision-making processes. Moreover, existing distance metric learning methods are primarily designed for clustering or retrieval tasks and are not effectively integrated into end-to-end supervised classification. In this work, we introduce *Latent Boost*, a training methodology that explicitly incorporates latent space structure into the classification objective. We combine probabilistic cross-entropy loss with distance metric learning through a weighted loss formulation, enabling simultaneous optimization of prediction accuracy and latent cluster organization. Thus, the model is not only optimized for classification of discrete data points but also enforces compact and well-separated class representations. By leveraging structural insights from intermediate latent representations, *Latent Boost* improves interpretability, as demonstrated by higher Silhouette scores, while accelerating training convergence. These benefits are achieved with minimal additional cost, making *Latent Boost* broadly applicable across datasets and architectures without data-specific adjustments. By enabling models to self-organize their internal representations during training, *Latent Boost* acts as a structural regularization mechanism aligning classification performance with transparent latent structure, positioning it as a general framework for interpretable and efficient classification systems.

Keywords: Latent Space Regularization, Supervised Metric Learning, Latent Space Interpretability

1 Introduction

Machine learning models are increasingly used across a wide range of domains, but their reliance on black-box architectures often obscures the internal decision-making processes, raising concerns about transparency and reliability [39,21].

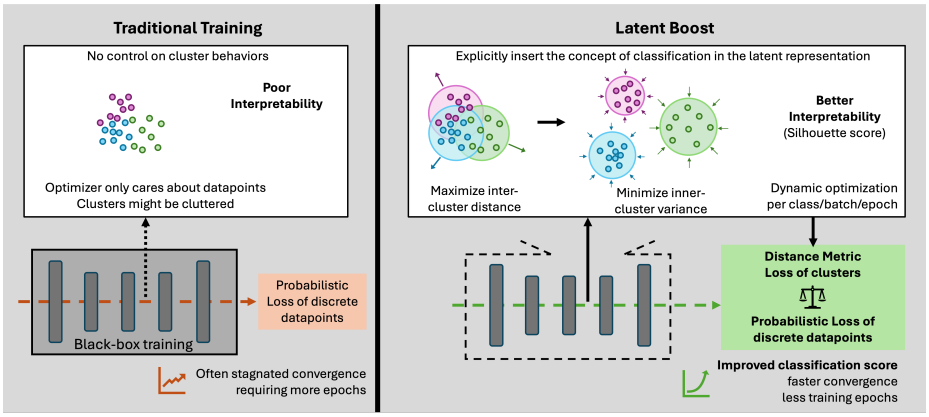


Fig. 1. In contrast to traditional training, relying on probabilistic loss only, *Latent Boost* injects distance metric information, obtained from the model's hidden latent representations, as addition into the training through balanced weighted sum equations.

Since Deep Neural Networks increasingly expand into sensitive domains such as medical, autonomous driving, or the avionics sector, the inability to interpret model decisions creates significant barriers to deployment and certification [2]. Trustworthy machine learning requires models not only to perform well but also to offer explanations that align with human understanding, ensuring that decisions are robust and free of unintended biases [17]. We therefore treat training not merely as parameter optimization but as a mechanism that shapes the organization of latent representations to simultaneously promote predictive accuracy and structural interpretability. This limitation is particularly evident in traditional data-driven classification training, which focuses solely on optimizing classification scores for discrete datasets while neglecting the structural organization of clusters within the continuous latent representation [43].

As a promising solution, latent representations, extracted from intermediate outputs of neural networks, contain rich information about the propagated data [11]. However, without explicit guidance during training, these representations often lack cohesion, with semantically similar instances failing to cluster effectively. This lack of organization can undermine the interpretability of latent spaces, as well as the ability of downstream tasks to utilize the encoded information efficiently [18]. Models trained in this way may struggle to generalize beyond the training data, especially in the presence of domain shifts through out-of-distribution deployments [36]. Addressing these deficiencies is critical for building robust systems capable of handling the ubiquitous challenges of real-world scenarios [42]. This lack of focus on latent structure not only limits interpretability but can also impede the model's ability to generalize effectively across varying domains and datasets [28]. To address these challenges, we propose an innovative approach that explicitly integrates the classification objective into the latent representation through distance metric learning.

In this work, we introduce a novel method, *Latent Boost*, that seamlessly combines latent cluster distance metrics with probabilistic training as motivated in Figure 1, fundamentally transforming the paradigm of structured latent representations to improve both transparency and performance. While conventional probabilistic approaches typically center on individual data samples, they often overlook the intricate relationships among data points, particularly within clusters, where the interdependencies and structural nuances are essential for capturing the underlying distribution patterns [52]. In other words, while the end-to-end classification performance might be satisfactory, metrics such as the F1-score are derived from discrete independent data points. Internally, the collective data may not form clear clusters, or the formed clusters of different classes could still be entangled in the continuous latent space, as the training processes only optimize for the probabilistic loss of the discrete dataset. In contrast, our innovative approach guarantees that semantically similar data points are thoroughly aligned in proximity, while dissimilar points are distinctly separated, thereby enhancing the model's capacity for nuanced differentiation. This not only improves the interpretability of the learned latent representations but also provides better alignment with the underlying data distribution, ensuring more meaningful and reliable predictions.

Although distance metric learning has established its efficacy in Machine Learning, particularly in clustering and pre-training tasks [30,26], its application in classification, especially when being trained from scratch, has been limited to simpler models like K-nearest Neighbors [10] and Support Vector Machines [9]. We propound that intermediate latent representations hold critical structural information essential for class distributions, and enhancing their cluster separation can significantly boost classification performance. *Latent Boost* therefore integrates distance metrics into the classification loss function, empowering the model to cultivate more meaningful and discriminative features without requiring explicit supervision. By incorporating distance-based loss into the weighted training process, *Latent Boost* bridges the gap between structural clustering methods and probabilistic classification, delivering a holistic framework for supervised learning.

Our key contributions include:

- We incorporate distance metric learning, traditionally developed for unsupervised clustering, into supervised classification through a weighted sum loss equation. This allows the model to be optimized for both classification metrics and interpretable latent representations.
- We propose a novel distance-based loss *Latent Boost* specifically for supervised classification, addressing previously overlooked nuances with dynamic adaptation and discriminative information density when training models from scratch.
- We demonstrate *Latent Boost* is a simple yet efficient approach to enhance performance and interpretability while shrinking computational demand through faster convergence.

2 Related Work

Distance Metric Learning has emerged as a crucial area in Machine Learning, offering a wide range of techniques aimed at improving performance in tasks of primarily unsupervised clustering and retrieval of latent representation information in general [30,49]. Such loss functions generally focus on minimizing intra-cluster variance and maximizing inter-cluster distances by learning a latent representation where similar points are closer together, and dissimilar points are further apart [26]. These functions penalize the model until the latent representation aligns with the desired distance metric priorities.

The structure and information density in latent representations also improve interpretability. Discriminative Dimension Selection can be utilized to enhance the interpretability and clustering performance, especially for K-means clustering by selectively retaining relevant features [33]. Similarly, works on adaptive feature selection and optimization have focused on developing efficient strategies to reduce the burden of complex dimensionality, improving clustering accuracy across several benchmarks [55,20,19]. Oppose to utilizing the Euclidean distance information, a boosting algorithm to effectively learn Mahalanobis distance metrics was proposed by Chang et al., demonstrating its effectiveness on popular datasets to capture intrinsic distance relationships [6]. Mahalanobis-based techniques have been widely adopted, with approaches such as Large Margin Nearest Neighbor [51] maximizing the margin between different classes, and Information-Theoretic Metric Learning [12], which minimizes the relative entropy between distance distributions.

Pairwise and Contrastive loss functions have shown remarkable improvements in distance metric learning, such as the Contrastive loss function for learning dimensionality-reducing embeddings [22]. This principle has been adopted in domains like zero-shot learning [50], cross-modal retrieval [48], and large-scale face recognition [35]. Contrastive learning has also been used in self-supervised settings, where methods such as SimCLR [7] leverage this loss to learn robust features without labels. The promising approaches of triplet-based methods for distance metric learning have been challenging to optimize due to the need for finding informative triplet anchor points [15]. To address this, semi-hard triplet mining methods have been developed, leading to more efficient training [44]. Advanced sampling strategies have been proposed to improve the performance of triplet-based learning systems [24]. Additionally, using proxy points to approximate original data points, as shown by Movshovitz-Attias et al., further improves convergence and stability in Triplet loss optimization [37]. Magnet loss has gained attention in recent years as an effective method for distance metric learning, particularly in dealing with high-dimensional data where traditional losses like Triplet loss face challenges [40]. Yu et al. introduce Semantic Drift Compensation as part of Magnet loss as a method to address catastrophic forgetting in class-incremental learning by estimating and compensating for the feature drift of previous tasks, significantly improving performance in embedding networks without requiring exemplars [54]. By focusing on the distribution of latent representations within each class, Magnet loss works to minimize the

overlap between class clusters, thus providing better discrimination, especially in face identification in noisy and large-scale datasets [13].

Metric learning with constraints has led to significant advances, with pairwise constraints being integrated into methods like Constrained Clustering via Metric Learning [3]. Similarly, Ding et al. extended this concept into semi-supervised clustering, demonstrating the utility of leveraging partial label information [14]. In sparse and high-dimensional data scenarios, sparse subspace clustering has been a successful approach [34].

Fairness in distance metric learning has also become a crucial area of focus, with works exploring how adversarial techniques can be used to ensure fairness in learned distance metrics [31]. These methods prevent demographic bias and ensure equitable performance across different groups, a significant consideration for real-world applications.

Methods like DeepCluster [4] and cluster-based contrastive learning [5] demonstrate how metric learning can generate meaningful representations for downstream tasks such as image retrieval. The innovation in these approaches lies in integrating clustering with deep learning techniques to build robust data representations. Hybrid models that integrate different learning techniques have also gained attention. For example, Lee et al. [32] employed stacked attention networks for cross-modal tasks, combining textual and visual data within joint latent spaces to enable multi-modal learning. However, those methods rest on unsupervised clustering techniques and are therefore unsuitable for our supervised multi-class classification approach.

3 Incorporating Distance Metrics in Multi-Class Classification

As the first novel contribution of this work, we incorporate distance metric information into supervised classification with probabilistic loss through a weighted sum loss function. Based on the previously referenced works from Section 2, we identified the Contrastive [8], Triplet [44], N-pair [46], and Magnet loss as the currently most relevant and recent choices. The theoretical background of each approach can be found in Appendix A. Such distance metrics have been hitherto primarily used for unsupervised clustering. Equation (1) outlines the total loss calculation on which our experiments rest. We introduce the hyperparameter λ to balance the weight between the two loss components. The range of λ can be continuously selected between 0 to 1, with 0 giving full weight to the probabilistic loss whereas 1 gives full weight to the distance metric loss. In order to isolate the influence of λ and the effect of each selected distance metric, we fixed the probabilistic loss within each experiment across epochs to shrink the overall experiment complexity. Moreover, with this approach the loss functions balance stays constant across the experiment, preventing any training inconsistencies through unexpected loss manipulations. Additionally, we selected the well-established soft-max cross-entropy as a counterpart, which matches the idea of utilizing the weighted sum equation for multi-class classification.

$$\begin{aligned} \mathcal{L}_{total} &= \lambda \cdot \mathcal{L}_{\text{dist}} + (1-\lambda) \cdot \mathcal{L}_{\text{cross-entropy}} \qquad \lambda \in [0,1], \\ &\text{with } \mathcal{L}_{\text{dist}} \in \{\mathcal{L}_{\text{contrast}}, \mathcal{L}_{\text{triplet}}, \mathcal{L}_{\text{N-pair}}, \mathcal{L}_{\text{magnet}}\}, \\ \mathcal{L}_{\text{cross-entropy}} &= -\sum_{\forall x} p(x) \log(q(x)) \end{aligned} \tag{1}$$

3.1 Experiment Setup

We selected three different experiment setups of varying complexity to explore the effects of the previously introduced distance metric losses through the weighted sum equations. The datasets used include Fashion MNIST [53], a collection of grayscale images representing 10 fashion categories, chosen for its simplicity and suitability for validating foundational improvements; CIFAR-10 [29], a dataset of natural color images spanning 10 object classes, selected to assess performance on more complex visual data; and CIFAR-100 [29], which significantly increases the challenge with its 100 fine-grained categories of color images, providing a rigorous evaluation of scalability and the ability to handle diverse, high-dimensional latent spaces. Each of the datasets has been split into train, validation and test split according to their state-of-the-art splits.

The model architecture trained on Fashion MNIST is a convolutional neural network (CNN) adapted for grayscale images. The network starts with a convolutional layer that takes single-channel (grayscale) 28x28 images and outputs a set of feature maps, followed by ReLU activation and max-pooling. The model employs two convolutional layers, with each followed by pooling and dropout of 25% to mitigate overfitting. The resulting feature maps are flattened and passed through fully connected layers. From the flattening layer, we extract the latent features. The final fully connected layer outputs the class probabilities for the 10 fashion categories.

The model architecture used for CIFAR-10 is based on the VGG-16 network [45]. VGG-16 is a deep convolutional network known for its success in image classification tasks on complex color images like CIFAR-10. This architecture consists of 16 layers, with 13 convolutional layers interspersed with ReLU activations and max pooling to downsample the feature maps, followed by three fully connected layers. The network extracts progressively richer features from the images, which are then classified into one of the 10 classes.

The model trained on CIFAR-100 is based on the ResNet-50 architecture [23] and comprises several key layers to efficiently process the input images. The architecture begins with a modified convolutional layer that accepts three input channels and employs a kernel size of 3 times 3, followed by batch normalization and ReLU activation. It includes four residual blocks (layer1 to layer4), each containing a series of convolutional layers, batch normalization, and skip connections to enhance gradient flow. The network concludes with an average pooling layer, which is flattened to extract the latent representations, and a fully connected layer that outputs the final class predictions, specifically tailored for the CIFAR-100 classification task.

All models were trained from scratch by initializing them with random weights. For each experiment, we extracted the latent representation after the convolutional part in the network architecture, meaning just before the classification section. For the CNN the latent representation has a dimension of 64, the VGG-16 model obtains 512 dimensions on that layer and the ResNet-50 generates a latent representation of 2048 dimensions. The choice to extract latent features from layers just before the classification head is based on their ability to capture high-level, task-specific representations that balance discriminative power and compactness, as these layers contain semantically rich information essential for classification tasks. Earlier layers typically emphasize low-level features and may lack the necessary information density required for effective downstream processing, making deeper layers more suitable.

Each setup was implemented in PyTorch [38] with a learning rate scheduler that reduced the learning rate by factor 5 when a plateau was reached after 10 epochs. The Adam optimizer was utilized for each experiment to efficiently work out the gradients [27]. We applied early stopping metrics with 20 epochs patience based on the validation accuracy to additionally compare the convergence speed next to the classification performance. Finally, we trained each experiment setup five times to provide meaningful results. We assigned each trial a different random seed, which we kept constant within the trials different runs across the variation of lambda to balance the distance metric with the classic cross-entropy.

3.2 Superiority of Magnet Loss

For our preliminary evaluation, we utilize the original hyperparameter settings for each distance metric loss function as shown in Table 5 (Appendix B) and balance them in the weighted sum loss.

The results of varying the hyperparameter λ across the different loss functions are reported across our three experiment setups, comparing the accuracy and Micro-F1 Scores. The metrics definitions are reported in Appendix C. We conducted each experiment across the range of 0.1 to 0.9 while keeping the value constant across each run. In Table 1, we present the most relevant λ values to compare and recognize trends. The aim of this evaluation is to highlight the overall potential of outperforming the baseline performance of the model through latent distance information. The results show that adjusting λ can improve performance, especially when applying the Magnet loss. On Fashion MNIST, Magnet loss outperforms other methods, achieving 89.52% accuracy and 0.8946 Micro-F1 scores at $\lambda = 0.75$. On CIFAR-10, Magnet loss also achieves the highest results, with an accuracy of 87.36% and a Micro-F1 score of 0.8738 at $\lambda = 0.75$. However, on CIFAR-100, the results were less conclusive due to its high complexity, but Magnet loss still showed some improvement over the baseline. Overall, the Magnet loss is the most robust existing method across our experiments, especially when selecting λ values in the range 0.5 to 0.9, since it properly improves classification performance across datasets by leveraging latent information and reducing intra-class variability with proper priority.

Table 1. Baseline comparison of Accuracy (↑) and Micro-F1 score (↑) across our three experiments using different loss functions and varying λ values; the baseline (λ=0.0) represents the standard model without additional loss integration. Metrics are mean ± std.

Dataset	λ	Loss	Accuracy	Micro-F1
Fashion-MNIST	**0.0**	Baseline	**88.59 ±0.15**	**0.8867 ±0.0020**
	0.25	Contrast	88.83 ±0.08	0.8884 ±0.0015
	0.25	Triplet	89.13 ±0.18	0.8909 ±0.0011
	0.25	N-pair	88.85 ±0.07	0.8883 ±0.0010
	0.25	Magnet	89.27 ±0.29	0.8928 ±0.0023
	0.5	Contrast	88.57 ±0.14	0.8850 ±0.0019
	0.5	Triplet	89.13 ±0.34	0.8915 ±0.0036
	0.5	N-pair	89.25 ±0.09	0.8923 ±0.0005
	0.5	Magnet	89.07 ±0.21	0.8911 ±0.0022
	0.75	Contrast	87.79 ±0.25	0.8773 ±0.0022
	0.75	Triplet	88.97 ±0.33	0.8900 ±0.0038
	0.75	N-pair	88.71 ±0.53	0.8869 ±0.0044
	0.75	Magnet	**89.52 ±0.34**	**0.8946 ±0.0040**
CIFAR-10	**0.0**	Baseline	**85.88 ±0.40**	**0.8586 ±0.0042**
	0.25	Contrast	86.01 ±0.99	0.8600 ±0.0102
	0.25	Triplet	86.81 ±0.25	0.8667 ±0.0025
	0.25	N-pair	84.52 ±0.71	0.8446 ±0.0067
	0.25	Magnet	86.91 ±0.69	0.8692 ±0.0073
	0.5	Contrast	86.87 ±0.56	0.8693 ±0.0049
	0.5	Triplet	86.88 ±0.64	0.8683 ±0.0059
	0.5	N-pair	86.43 ±0.78	0.8643 ±0.0081
	0.5	Magnet	86.72 ±1.77	0.8671 ±0.0174
	0.75	Contrast	84.74 ±0.44	0.8479 ±0.0045
	0.75	Triplet	86.95 ±0.42	0.8690 ±0.0039
	0.75	N-pair	85.91 ±0.40	0.8583 ±0.0042
	0.75	Magnet	**87.36 ±0.82**	**0.8738 ±0.0081**
CIFAR-100	**0.0**	Baseline	**61.77 ±0.49**	**0.6163 ±0.0064**
	0.25	Contrast	61.04 ±0.81	0.6088 ±0.0087
	0.25	Triplet	62.36 ±0.91	0.6213 ±0.0070
	0.25	N-pair	61.54 ±0.83	0.6139 ±0.0075
	0.25	Magnet	62.43 ±0.38	0.6242 ±0.0039
	0.5	Contrast	60.00 ±0.71	0.5984 ±0.0087
	0.5	Triplet	60.41 ±0.51	0.6022 ±0.0041
	0.5	N-pair	60.13 ±0.94	0.5991 ±0.0075
	0.5	Magnet	**62.75 ±0.51**	**0.6256 ±0.0057**
	0.75	Contrast	54.91 ±0.71	0.5482 ±0.0071
	0.75	Triplet	56.13 ±0.76	0.5597 ±0.0082
	0.75	N-pair	47.73 ±1.79	0.4694 ±0.0168
	0.75	Magnet	62.66 ±0.54	0.6242 ±0.0045

4 Latent Boost Evolution

Among the hitherto loss functions, Magnet loss provides superior potential in enhancing classification performance, wherefore we selected the Magnet loss as a basis for our *Latent Boost* approach to incorporate distance with classification metrics. However, those distance metric losses were initially developed for unsupervised clustering. Although we have repurposed them for supervised classification through Equation (1), there is room for improvement, as for the classification objective, they have several limitations and overlooks certain nuances. Specifically, no modifications or optimizations of the hyperparameters associated with the Magnet loss have been explored so far. Additionally, in the original formulation, the distance metric loss is computed across the full latent space. While this approach captures the complete available information, this might introduce noise and excessive computational demand if the latent representation includes dimensions that do not yet provide beneficial information. The following sections introduce our evolutionary development of *Latent Boost*, targeting the maximization of efficiency and explainability.

4.1 Condensed Distance Information

As a first step, we propose incorporating a dimensionality reduction step before calculating the *Latent Boost* batch loss by applying Principal Component Analysis (PCA) to reduce the dimensions of the latent vectors. PCA was selected for its balance of simplicity, computational efficiency, and ability to retain the maximum variance in the data, making it particularly suitable for preserving the structural integrity of the latent representations in high-dimensional spaces. Other dimensionality reduction techniques, such as t-SNE or UMAP, while effective for visualization, are less suited for downstream optimization tasks due to their nonlinear transformations and stochastic nature.

Let $\mathbf{W}_{\text{PCA}} \in \mathbb{R}^{d' \times d}$ denote the matrix of the top principal components, where d' is the reduced dimension and d is the original dimension. Each latent vector r_n is projected onto the lower-dimensional subspace as:

$$r'_n = \mathbf{W}_{\text{PCA}} r_n \tag{2}$$

To compute the class-wise cluster centroids used in the distance-based loss formulation, we first define the centroid $\mu_C \in \mathbb{R}^d$ of a class-specific cluster C in the original latent space as the mean of all latent vectors $r_i \in C$:

$$\mu_C = \frac{1}{|C|} \sum_{r_n \in C} r_n \tag{3}$$

Here, $|C|$ denotes the number of samples in cluster C, and r_n are the latent representations of those samples. This centroid reflects the central location of the latent representations within the cluster and serves as the anchor point for measuring intra- and inter-cluster distances.

Once the centroids μ_{r_n} (for the current class) and μ_{c_k} (for other classes c) are calculated in the original latent space, they are projected onto the same lower-dimensional subspace via PCA:

$$\mu'_{r_n} = \mathbf{W}_{\text{PCA}}\mu_{r_n}, \quad \mu'_{c_k} = \mathbf{W}_{\text{PCA}}\mu_{c_k} \tag{4}$$

The number of retained principal components dim is determined by ensuring a predefined threshold T of cumulative explained variance is met, as follows:

$$dim = \min\left\{ i \;\middle|\; \frac{\sum_{j=1}^{i} \frac{S_j^2}{m-1}}{\sum_{j=1}^{\text{max\_dim}} \frac{S_j^2}{m-1}} \geq T \right\} \quad \text{or} \quad dim = dim_{max} \text{ if no such } i \tag{5}$$

Here, S_j are the singular values from the Singular Value Decomposition (SVD) that is extracted from the PCA dimension reduction process, m is the number of samples, n is the number of available features, and $\text{max\_dim} = \min(m, n)$ is the maximum number of possible components. T represents the threshold for cumulative explained variance, which we set to 0.95 based on initial investigations.

After applying PCA, we compute the *Latent Boost* loss using the reduced latent representations r'_n and μ', including the distance information through a dense and efficient format.

4.2 Individual Cluster Variance

To ensure the model effectively adapts to the low dimensional representation for each cluster σ_C^2, the cluster variance is now computed based on the spread of points within each cluster individually. This allows the model to better account for the inherent variability and density differences within each cluster due to the filtered and compressed latent space information. Specifically, the variance represents the average squared distance of the points r_i in cluster C from the cluster mean μ_C. The formula for calculating the variance for a cluster C is given by:

$$\sigma_C^2 = \frac{1}{|C| - 1} \sum_{r_i \in C} \|\mathbf{r}_i - \boldsymbol{\mu}_C\|^2 \tag{6}$$

Here, $|C|$ denotes the number of points in cluster C and adjusts the degree of freedom. r_i represents the position of the i-th point in the cluster and μ_C is the centroid of the cluster from which we calculate the squared Euclidean distance between. This dynamic variance allows the model to adapt to clusters with varying densities, meaning clusters that are more spread out will have larger variance and be more forcefully compressed compared to tighter clusters.

4.3 Dynamic Inter and Intra Cluster Balance

Additionally, we introduce a hyperparameter β in the denominator of the loss function. The Magnet loss, as presented in Equation (13), is composed of two main components: intra-cluster variance minimization, controlled by α, and

inter-cluster separation, now influenced by β. To balance these competing objectives, we developed dynamic strategies to adjust α and β based on the current training epoch E, as formulated in Equation (7). Initially, the focus is on achieving tight clustering of intra-class samples by assigning larger values to α. Subsequently, β plays a greater role in encouraging separation between clusters. The update rule for α follows an exponential decay schedule due to the simple subtraction as a margin term. It starts at a value of $1 + \alpha_0$, where α_0 controls the initial strength, and gradually decreases by the factor $e^{-\frac{\mathrm{E}}{1.05 \cdot \mathrm{E}_{\mathrm{total}}}}$ towards the commonly set value of 1.0 as training progresses. This approach ensures that the focus on minimizing intra-cluster variance gradually weakens as the model learns to form tighter clusters.

In contrast, the update rule for β follows a linear schedule, starting from β_0, and decreases linearly until it reaches zero after the first 20% of the conventional training period. This linear decay ensures that the inter-cluster distance is encouraged early in training, but gradually becomes more significant as training progresses. This schedule ensures that β gradually diminishes and therefore increases the effect of distancing the clusters. This mechanism allows the model to be unconstrained by β in the early stages due to the multiplication factor and its initialization with 1.0, but ensures that the influence of the inter-cluster distance denominator increases for the overall loss.

Based on our experiments, the intra-cluster is mostly relevant during early epochs, whereas the exponential decrease of α helps to diminish the effect of intra-cluster variance distance later in the training progress. However, the inter-cluster importance needs to be linearly increased by decreasing β in order to move the clusters continuously and gradually farther apart till the model converges.

To ensure numerical stability across our experiments, we added ϵ and set it to $1 \cdot 10^{-8}$ as a diminutive constant to prevent division by zero and underflow of floating point numbers. Furthermore, ϵ is introduced as a minimum value for β to prevent it from diminishing completely and to avoid destabilization of the training process.

$$\alpha = 1 + \alpha_0 \cdot e^{-\frac{\mathrm{E}}{1.05 \cdot \mathrm{E}_{\mathrm{total}}}} \qquad \beta = \max\left(\beta_0 \cdot \left(1 - \frac{\mathrm{E}}{0.2 \cdot \mathrm{E}_{\mathrm{total}}}\right), \epsilon\right) \tag{7}$$

5 Full Potential of Latent Boost

Combining the previously introduced adaptations to the superior Magnet loss from our experiments, we can form the *Latent Boost* loss as stated in Equation (8). Finally, the *Latent Boost* loss is balanced through the λ hyperparameter with the cross-entropy loss to enhance multi-class classification. The full

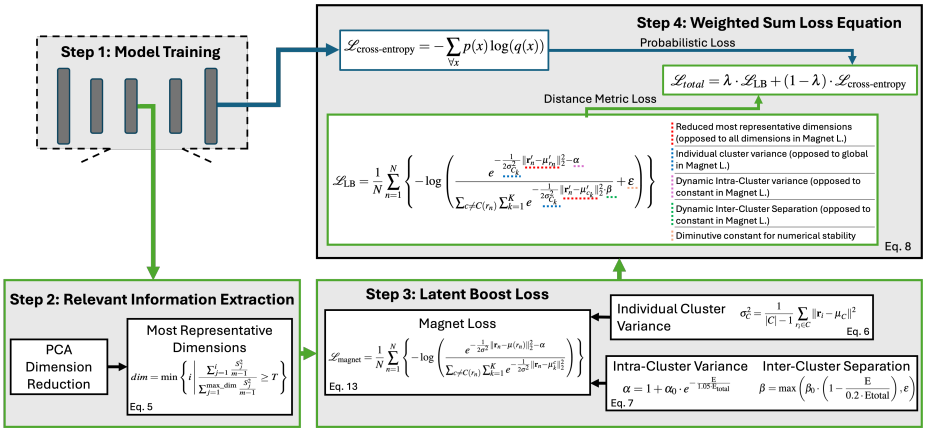


Fig. 2. Flowchart of the *Latent Boost* approach, summarizing the mathematical steps to embed distance-metric information into the classic probabilistic training.

equation as part of the main contribution of this work is expressed as follows:

$$\mathcal{L}_{\text{LB}} = \frac{1}{N}\sum_{n=1}^{N}\left\{-\log\left(\frac{e^{-\frac{1}{2\sigma_{C_k}^2}\|\mathbf{r}'_n-\boldsymbol{\mu}'_{r_n}\|_2^2-\alpha}}{\sum_{c\neq C(r_n)}\sum_{k=1}^{K} e^{-\frac{1}{2\sigma_{C_k}^2}\|\mathbf{r}'_n-\boldsymbol{\mu}'_{c_k}\|_2^2\cdot\beta}}+\epsilon\right)\right\},$$
$$\mathcal{L}_{\text{cross-entropy}} = -\sum_{\forall x} p(x)\log(q(x)) \tag{8}$$
$$\mathcal{L}_{total} = \lambda\cdot\mathcal{L}_{\text{LB}} + (1-\lambda)\cdot\mathcal{L}_{\text{cross-entropy}} \qquad \lambda\in[0,1]$$

In Figure 2, the whole process of *Latent Boost*, applied within each batch iteration, is visualized comprehensively. Within the traditional model training, we take the latent space data from intermediate layers. We reduce the dimensions using PCA and extract the most representative ones based on Equation (5). Afterwards, we calculate the *Latent Boost* based on the Magnet loss from Equation (13) (Appendix A) and the modifications of Equation (6) and Equation (7). In the final step, we combine the classic Cross-Entropy loss with the *Latent Boost* part through the weighted sum loss from Equation (8) with the λ parameter.

5.1 Improved Classification Performance

To evaluate our approach, we conducted the same experiment, based on our three datasets and model combinations, with the adapted *Latent Boost* distance metric. The selected λ values mimic the range of the preliminary experiment from Section 3.2 starting from 0.1 to 0.9. The results can be found in Table 2 and show the performance of *Latent Boost* across three datasets and model combinations. For Fashion MNIST, the best performance is achieved at $\lambda = 0.75$ with an accuracy of 90.86% and a Micro-F1 score of 0.9038. Similarly, on CIFAR-10,

Table 2. Complete performance evaluation of *Latent Boost* across the full set of λ values. Values for Accuracy and Micro-F1 with (Mean ± Std).

λ	**Fashion-MNIST**		**CIFAR-10**		**CIFAR-100**	
	Accuracy	Micro-F1	Accuracy	Micro-F1	Accuracy	Micro-F1
0.1	88.47 ±0.16	.889 ±.002	83.11 ±0.19	.834 ±.002	60.34 ±0.19	.611 ±.002
0.2	89.03 ±0.12	.891 ±.002	84.29 ±0.16	.843 ±.002	61.45 ±0.12	.617 ±.002
0.25	89.36 ±0.14	.892 ±.002	85.22 ±0.20	.846 ±.002	62.21 ±0.17	.622 ±.002
0.3	89.62 ±0.09	.893 ±.002	85.64 ±0.15	.849 ±.002	62.67 ±0.22	.626 ±.002
0.4	90.05 ±0.12	.898 ±.002	86.05 ±0.13	.851 ±.002	63.02 ±0.33	.628 ±.002
0.5	90.47 ±0.19	.901 ±.002	86.85 ±0.12	.855 ±.002	**63.04 ±0.48**	**.629 ±.002**
0.6	90.15 ±0.13	.902 ±.002	87.43 ±0.11	.860 ±.001	62.83 ±0.31	.628 ±.002
0.7	90.09 ±0.21	.902 ±.002	88.15 ±0.24	.864 ±.001	62.45 ±0.27	.627 ±.002
0.75	**90.86 ±0.21**	**.904 ±.003**	**88.44 ±0.28**	**.884 ±.002**	62.30 ±0.14	.626 ±.002
0.8	89.76 ±0.14	.895 ±.002	87.92 ±0.22	.880 ±.002	62.00 ±0.18	.623 ±.002
0.9	88.53 ±0.15	.890 ±.002	86.30 ±0.18	.873 ±.002	60.78 ±0.11	.620 ±.002

$\lambda = 0.75$ gives the highest accuracy (88.44%) and Micro-F1 score (0.8843), while higher values of λ slightly reduce performance. On CIFAR-100, the performance is less sensitive to λ, with the best results at $\lambda = 0.5$. Overall, λ values between 0.5 and 0.75 consistently deliver the best results, indicating an optimal balance for beneficial performance while structuring the latent spaces.

Additionally, as shown in Table 3, we track the accuracy, Micro-F1 Score, and the duration of training epochs. We calculated the average and standard deviation across our five experiment trials to compare our baseline without distance metrics and the classic Magnet loss addition. For the *Latent Boost*, we added two versions, first without the addition of PCA to isolate the effects of variance and the dynamic update rule and finally the full *Latent Boost* with all features. With that, we can manifest the performance of *Latent Boost* applied to the full high-dimensional space and obtain insights into the effects of extracting the necessary information from the dimension reduction process.

Latent Boost proves to consistently outperform the baseline and the classic Magnet loss results from the previous experiments of Table 1. For Fashion-MNIST and CIFAR-10, λ selection of 0.75 and 0.5 for the CIFAR-100 obtained the best performance equal to the original Magnet loss results. The Fashion MNIST shows a 2.56% increase in accuracy and a 1.93% increase in F1 Score, while CIFAR-10 and CIFAR-100 exhibit comparable improvements of around 2-3%. These gains are accompanied by reduced standard deviations, indicating a more stable and reliable convergence due to the additional structural information from latent representation. The tighter variability in performance indicates that *Latent Boost* is more consistent across datasets.

Another advantage of *Latent Boost* is its faster and more stable convergence next to the model performance. The number of epochs required for Fashion MNIST training is reduced by around 14%, while CIFAR-10 and CIFAR-100 show reductions of around 13% and 21%, respectively. This reduction not only

Table 3. Accuracy (↑), Micro-F1 Score (↑), and epoch duration (↓) between baseline ($\lambda = 0$), standard Magnet loss, the *Latent Boost* without PCA and the final *Latent Boost* with all features on the unseen test dataset (best λ selection); Δ compares *Latent Boost* vs Baseline. Values for Accuracy and Micro-F1 with (Mean ± Std)

Data	Metric	Baseline	Magnet	LB(-PCA)	LB	Δ
Fashion-MNIST	Accuracy	88.59 ±0.15	89.52 ±0.34	88.12 ±0.29	**90.86 ±0.21**	+2.56 %
	Micro-F1	.887 ±.002	.895 ±.004	.897 ±.004	**.904 ±.005**	+1.93 %
	Epochs	40.67 ±4.19	37.67 ±6.18	34.86 ±4.26	**34.67 ±3.09**	-14.76 %
CIFAR-10	Accuracy	85.88 ±0.40	87.36 ±0.82	88.12 ±0.47	**88.44 ±0.28**	+2.98 %
	Micro-F1	.859 ±.004	.874 ±.008	.881 ±.004	**.884 ±.002**	+2.99 %
	Epochs	76.33 ±8.65	72.67 ±1.70	71.94 ±3.72	**66.33 ±4.50**	-13.09 %
CIFAR-100	Accuracy	61.77 ±0.49	62.75 ±0.51	63.05 ±0.52	**63.04 ±0.48**	+2.06 %
	Micro-F1	.616 ±.006	.626 ±.006	.622 ±.006	**.629 ±.005**	+2.03 %
	Epochs	86.67 ±9.46	69.67 ±8.73	69.23 ±8.92	**68.67 ±6.02**	-20.74 %

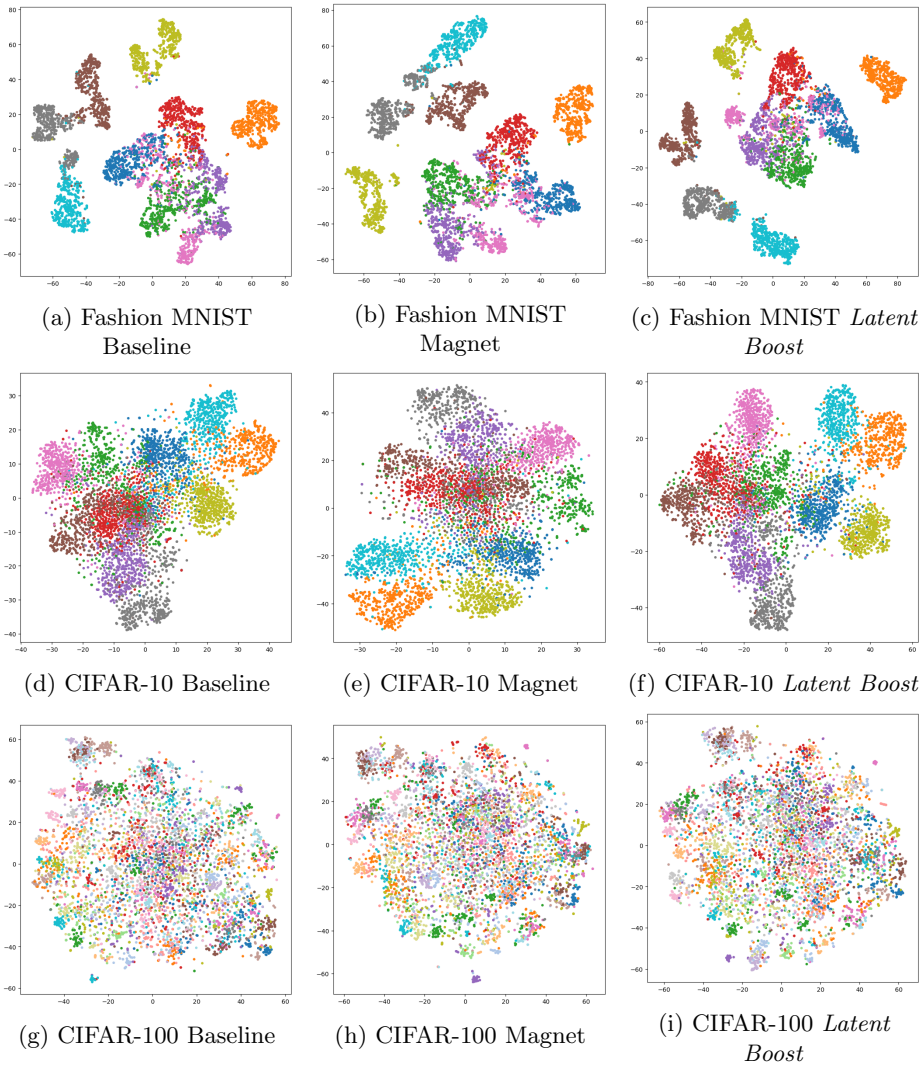
speeds up training but also offers potential sustainability benefits by decreasing computational resources and energy consumption.

As stated before, the threshold for selecting the number of principal components was decided to be at 95% of cumulative explained variance. For Fashion-MNIST, the number of principal components ranged around 45 to 50, with slight degradation to 40 components when the latent representation reaches a sufficient structure. For CIFAR-10, the number of principal components started around 65 and decreased to 40 components, stating that the latent representation is well structured throughout the epochs. For the final CIFAR-100 experiment, the principal components used were around 100 to 110 without a decent decrease over time. That being said, it reflects our findings of struggling to structure the large latent representation properly.

5.2 Improved Latent Representation Interpretability

We define interpretable classification as one in which class-specific latent representations form well-separated and structured clusters, facilitating human-interpretable organization of the latent space.

Qualitative Inspection: In Figure 3, we plotted the 2-dimensional representation of the high-dimensional latent space for each of our experiments. We passed the test dataset of each experiment through the model and extracted the latent representation the same way as previously proposed. However, we do not calculate the *Latent Boost* metric but rather utilize the data and shrink its dimension with classic and state-of-the-art TSNE [47] for visualization purposes. We utilized the TSNE according to best practices, keeping hyperparameters in its standard initialization and fixing the random seed. The visualizations from left to right show the latent representation for the best trial of baseline training without any distance metric information, the Magnet loss training from our preliminary experiments, and the *Latent Boost* approach. From top to bottom, we show each of the experiment models and dataset combinations.



(a) Fashion MNIST Baseline (b) Fashion MNIST Magnet (c) Fashion MNIST *Latent Boost*

(d) CIFAR-10 Baseline (e) CIFAR-10 Magnet (f) CIFAR-10 *Latent Boost*

(g) CIFAR-100 Baseline (h) CIFAR-100 Magnet (i) CIFAR-100 *Latent Boost*

Fig. 3. Comparison of baseline (λ=0), standard Magnet loss, and our *Latent Boost* approach across the three experiment setups.

Starting with the Fashion MNIST, even though the clusters were separated properly in the baseline already, we can see that in the *Latent Boost* clusters are formed with more concentrated density and classes such as *brown* and *grey* are well separated. In the scenario with CIFAR-10, a similar trend can be recognized. Even though there is much more confusion within the latent representation compared to the Fashion MNIST, clusters are more dense in the final stage of *Latent Boost*. Additionally, the main confusion between *green* and *red* class from the baseline could partially be resolved. Several classes exhibit more compact

cluster formation, and the boundaries between classes become more clearly distinguishable. For CIFAR-100, however, the visualization does not reveal a clear separation of clusters. This may partly stem from the difficulty of displaying 100 classes with similar color hues. Only peripheral regions of the latent space show small localized clusters, without a pronounced global structuring effect.

Quantitative Measurement: Since TSNE compresses the latent space and therefore loses the detailed information from the full dimensional representation, the visualizations can only be utilized for visual cross-comparison but do not suffice for compelling quantitative evaluation. To quantify the density of clusters and their separation from each other in the original dimension, we selected the Silhouette Score to measure the quality of the latent representation. Originally proposed by [41], the Silhouette Score is calculated for each data point by comparing the **cohesion** within its own cluster and the **separation** from the nearest neighboring cluster. It has been widely used for quantifying representation interpretability in recent works [1,25,16]. The score ranges from -1 to 1, with -1 indicating the location or assignment of samples to the wrong cluster, whereas 0 is between clusters and 1 is the best-case with clear allocation to the correct cluster. For a given data point i in a cluster C_i, the Silhouette Score $s(i)$ is calculated following Equation (9). $a(i)$ represents the cohesion as to how closely related a data point is to its own cluster, whereas $b(i)$ represents the separation, meaning the distance between a data point to its nearest neighbor cluster. To evaluate the impact of different training approaches on the latent representation, we calculated the Silhouette Scores for the previously discussed latent representations in Figure 3. The results, shown in Table 4, highlight that both the Magnet loss and *Latent Boost* methods improve cluster separation on the full dimension, relative to the baseline. *Latent Boost* yields the highest score in all cases, with particularly strong improvements observed for CIFAR-10.

$$\begin{gathered} \text{Silhouette Score} = \frac{1}{N} \sum_{i=1}^{N} \frac{b(i) - a(i)}{\max(a(i), b(i))} \\ a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, j \neq i} d(i,j) \qquad b(i) = \min_{C \neq C_i} \left(\frac{1}{|C|} \sum_{j \in C} d(i,j) \right) \end{gathered} \tag{9}$$

The results reflect the preliminary experiment hypothesis across the three datasets. Checking the improvement between baseline and *Latent Boost*, the Fashion MNIST experiment showed slight improvement, since the baseline latent space was already well separated. The CIFAR-10 achieved the greatest improvement in the dimensional representation with around 168 %, restating the model's complexity potential for improvement, especially in the high dimensions. For the CIFAR-100, even though some minor improvement could be recognized, the *Latent Boost* approach does not sufficiently support the training as expected. Since the Silhouette Score ranges in negative values for CIFAR-100, major confusion between the large set of classes still dominates the classification despite the slight increase.

Table 4. Silhouette Score to quantify cluster separation (larger values in range -1 to 1 represent greater separation); calculated on the models' latent representation without compression from the unseen test dataset; Δ compares *Latent Boost* vs Baseline.

Dataset	Baseline	Magnet	Latent Boost	Δ
Fashion MNIST	0.347	0.375	**0.395**	13.84 %
CIFAR-10	0.131	0.186	**0.351**	167.94 %
CIFAR-100	-0.280	-0.264	**-0.250**	10.71 %

While the Silhouette Score captures the quality of latent representation, it is important to note that the model is not directly trained on the latent representation data. Instead, the calculated metrics are injected as additional information through the loss function, helping the model to indirectly improve its internal representation structures. This approach leverages the power of the latent representation without requiring explicit supervision, allowing the model to develop more meaningful features.

6 Discussion

6.1 Overfitting Risk and Mitigation

Mitigating overfitting is essential to ensure that *Latent Boost* generalizes beyond the training data. *Latent Boost* computations were performed exclusively on training and validation data. Test data were used only for final evaluation, ensuring strict separation and preventing information leakage. Early stopping further limited overfitting by terminating training once validation performance plateaued, reducing unnecessary epochs and compute. Dropout regularization encouraged distributed representations by stochastically deactivating neurons during training. Although out-of-distribution evaluation was not conducted, it represents a relevant future direction to assess robustness under distributional shifts.

6.2 Dynamic Lambda Variations

The weighting factor λ balances classification performance and interpretability. Across experiments, values in the range 0.5–0.75 yielded the strongest improvements. We therefore used constant λ per experiment, as preliminary dynamic scheduling introduced instability and additional complexity. Future work may explore adaptive λ strategies that retain training stability while enabling finer control.

6.3 Training Resource Efficiency

Latent Boost requires several initial epochs with cross-entropy loss to form stable latent clusters before refinement. Although latent-space computations add per-epoch overhead, overall training converges faster and typically requires fewer

epochs. PCA-based dimensionality reduction further decreases latent size and computation, as *Latent Boost* operates only on the reduced dimensions (Eq. 5), whereas Magnet loss uses the full feature space. Benefits diminish when training is constrained to very few epochs.

6.4 Cluster Separability

Latent Boost assumes approximately hyperspherical and separable class clusters in latent space. This assumption is weaker for datasets with overlapping, irregular, or hierarchical structures, as observed in CIFAR-100, which may explain smaller gains. Future work could incorporate clustering or loss formulations that model hierarchical relations or cross-class dependencies to improve robustness across diverse latent geometries.

7 Conclusion

Overall, *Latent Boost* explicitly embeds the classification objective into the latent layer of an otherwise black-box model in the form of an additional loss term. This integration encourages the formation of better-separated clusters in the target latent space while simultaneously optimizing for prediction performance. As a result, the method achieves substantially improved latent interpretability, enhanced classification outcomes, and more efficient training convergence. While the uplift in classification performance is moderate, it is consistent across all benchmarks, exhibiting minimal deviation. Additionally, our approach facilitates faster convergence, aligning with the goals of sustainable machine learning by reducing both training time and computational resource consumption.

In conclusion, *Latent Boost* represents a significant advancement in harmonizing interpretability, efficiency, and performance within supervised learning. Its ability to explicitly structure latent representations during training highlights a promising direction toward more transparent and controllable machine learning systems, while future work may further extend this approach to more complex and less separable data regimes.

Declaration on Generative AI

GenAI tools were used solely for language refinement and phrasing improvements, while all scientific concepts, methodological decisions, analyses, and conclusions were developed and validated by the authors.

Acknowledgment

This work is supported by the European Union's HorizonEurope research and innovation program (HORIZON-CL4-2021-HUMAN-01) through the "SustainML" project (grant agreement No 101070408) and the German Federal Ministry of Education and Research (BMBF) through the "Cross-Act" project (Grant Agreement No. 01IW25001).

Appendix

A Distance Metric Learning Theoretical Background

Contrastive Loss is one of the simplest and most widely used loss functions for distance metric learning, introduced in the context of training Siamese networks [8]. The goal of Contrastive loss is to minimize the distance between pairs of samples that are similar and maximize the distance between pairs of samples that are dissimilar up to a certain margin. The Contrastive loss function is formulated in Equation (10), where $y_i \in \{0, 1\}$ is a binary indicator of similarity, with $y_i = 1$ for similar pairs and $y_i = 0$ for dissimilar ones. The Euclidean distance d_i is calculated between the latent representations of the two samples. The equation is applied across the total number of pairs N. Additionally, m works as a penalization, defining the margin as the minimum distance for dissimilar pairs.

$$\mathcal{L}_{\text{contrast}} = \frac{1}{2N} \sum_{i=1}^{N} \left(y_i \cdot d_i^2 + (1 - y_i) \cdot \max(0, m - d_i)^2 \right) \tag{10}$$

Triplet Loss was used by the FaceNet model architecture, introducing the triplets of samples [44] as shown in Equation (11). An anchor a_i, a positive sample p_i , and a negative sample n_i form one triplet. The goal is to ensure that the distance between the anchor and the positive sample is smaller than the distance between the anchor and the negative sample by at least a margin α. Compared to the Contrastive loss, the Triplet loss is based on the anchor and is not calculated solely on the pairwise samples. The Euclidean distance is calculated between the positive and negative sample to its anchor with m as the margin term.

$$\mathcal{L}_{\text{triplet}} = \frac{1}{N} \sum_{i=1}^{N} \max\left(0, d(a_i, p_i) - d(a_i, n_i) + m\right) \tag{11}$$

N-pair Loss shares similarity with Triplet loss and aims to converge more stable across the clusters [46]. Instead of using a single negative sample per triplet, N-pair loss optimizes the distance between the anchor and the positive sample while contrasting it with multiple negatives simultaneously. N-pair loss is particularly effective in multi-class classification tasks for large-scale datasets. The N-pair loss is defined in Equation (12) ($\mathbf{f}_i$ is the embedding of the anchor sample, $\mathbf{f}_i^+$ and $\mathbf{f}_i^-$ are positive and negative samples)

$$\mathcal{L}_{\text{N-pair}} = \frac{1}{N} \sum_{i=1}^{N} \log \left(1 + \sum_{i^+ \neq i^-} e^{\mathbf{f}_i^T \mathbf{f}_i^- - \mathbf{f}_i^T \mathbf{f}_i^+} \right) \tag{12}$$

Magnet Loss, introduced by Rippel et al. [40], is designed to address the challenges of high-dimensional and complex data distributions by grouping data into clusters instead of focusing on pairwise or triplet calculations. To benefit from cluster-based information, Magnet loss penalizes a sample based on

its distance to the centroid of the correct cluster and the other false clusters. The behavior of a Magnet is obtained by accounting for both intra-class compactness and inter-class separation. The Magnet loss function is formulated in Equation (13), where r_n is the latent representation of the sample, $\mu(r_n)$ is the mean of the cluster containing r_n, and μ_c are the means of other clusters. σ is the standard deviation, α is a margin term. N is the total number of samples whereas K represents the number of clusters.

$$\mathcal{L}_{\text{magnet}} = \frac{1}{N}\sum_{n=1}^{N}\left\{-\log\left(\frac{e^{-\frac{1}{2\sigma^2}\|\mathbf{r}_n-\boldsymbol{\mu}(r_n)\|_2^2-\alpha}}{\sum_{c\neq C(r_n)}\sum_{k=1}^{K} e^{-\frac{1}{2\sigma^2}\|\mathbf{r}_n-\boldsymbol{\mu}_k^c\|_2^2}}\right)\right\} \tag{13}$$

B Default Distance Metric Hyperparameters

Table 5 summarizes the original hyperparameter settings for various distance metric loss functions used in our preliminary experiments. For Contrastive loss, the positive and negative margins are set to 0.0 and 1.0, respectively, which determine the threshold for distinguishing between positive and negative pairs. The Triplet loss employs a margin of 0.05 and considers all possible triplets per anchor, ensuring a comprehensive evaluation of relative distances in the latent representations. N-Pair loss utilizes a MeanReducer as its reducer, which averages the distances, while Magnet loss is configured with an α value of 1.0, controlling the aggressiveness for forcing the latent space separation. These initial values were chosen based on established practices in the literature to ensure a fair comparison independently of the distance metric initialization or adaptation.

Table 5. Original hyperparameter setting for each distance metric utilized in the preliminary experiment.

Loss Function	Hyperparameter	Initial Value
Contrast	positive margin	0.0
	negative margin	1.0
Triplet	margin	0.05
	triplets per anchor	all
N-Pair	reducer	MeanReducer
Magnet	α	1.0

C Evaluation Metrics

Across our work, we utilized the Accuracy and the Micro-F1 Score as the main metrics to validate our approaches and experiments. We selected the Micro-F1 Score to aggregate the contributions of all classes to the overall results.

The Accuracy is given in percentage values and defined as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{14}$$

The Micro-F1 score is given in the range 0 to 1 and defined as:

$$\begin{aligned} \text{Micro-F1} &= 2 \cdot \frac{\text{Micro Precision} \cdot \text{Micro Recall}}{\text{Micro Precision} + \text{Micro Recall}} \\ \text{with Micro Precision} &= \frac{TP_{\text{total}}}{TP_{\text{total}} + FP_{\text{total}}}, \\ \text{Micro Recall} &= \frac{TP_{\text{total}}}{TP_{\text{total}} + FN_{\text{total}}} \end{aligned} \tag{15}$$

For both, the following abbreviations apply:

- TP = True Positives
- TN = True Negatives
- FP = False Positives
- FN = False Negatives

References

1. Adil M Bagirov, Ramiz M Aliguliyev, and Nargiz Sultanova. Finding compact and well-separated clusters: Clustering using silhouette coefficients. *Pattern Recognition*, 135:109144, 2023.
2. Hymalai Bello, Daniel Geißler, Lala Ray, Stefan Müller-Divéky, Peter Müller, Shannon Kittrell, Mengxi Liu, Bo Zhou, and Paul Lukowicz. Towards certifiable ai in aviation: landscape, challenges, and opportunities. *arXiv preprint arXiv:2409.08666*, 2024.
3. Mikhail Bilenko and Sugato Basu. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the 21st International Conference on Machine Learning (ICML)*, pages 11–18, 2004.
4. Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018.
5. 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.
6. Chin-Chun Chang. A boosting approach for supervised mahalanobis distance metric learning. *Pattern Recognit.*, 45:844–862, 2012.
7. Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1597–1607, 2020.
8. Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005.
9. Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
10. Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.
11. Jonathan Crabbé, Zhaozhi Qian, Fergus Imrie, and Mihaela van der Schaar. Explaining latent representations with a corpus of examples. *Advances in Neural Information Processing Systems*, 34:12154–12166, 2021.
12. Jason V Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S Dhillon. Information-theoretic metric learning. In *24th international conference on Machine learning*, pages 209–216, 2007.
13. Jiankang Deng, Jia Guo, Tongliang Liu, Mingming Gong, and Stefanos Zafeiriou. Sub-center arcface: Boosting face recognition by large-scale noisy web faces. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 741–757. Springer, 2020.
14. Chris Ding and Tao Li. Adaptive dimension reduction using discriminant analysis and k-means clustering. In *24th international conference on Machine learning*, pages 521–528, 2007.
15. Thanh-Toan Do, Toan Tran, I. Reid, B. V. Kumar, Tuan Hoang, and G. Carneiro. A theoretically sound upper bound on the triplet loss for improving the efficiency of deep distance metric learning. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10396–10405, 2019.

16. Peng Du, Fenglian Li, and Jianli Shao. Multi-agent reinforcement learning clustering algorithm based on silhouette coefficient. *Neurocomputing*, page 127901, 2024.
17. Birhanu Eshete. Making machine learning trustworthy. *Science*, 373:743 – 744, 2021.
18. Patrick Esser, Robin Rombach, and B. Ommer. A disentangling invertible interpretation network for explaining latent representations. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9220–9229, 2020.
19. Daniel Geißler, Mengxi Liu, Bo Zhou, Sungho Suh, and Paul Lukowicz. Spend more to save more (sm2): An energy and hardware-aware implementation of successive halving for sustainable hyperparameter optimization. In *International Conference on Architecture of Computing Systems*, pages 140–155. Springer, 2025.
20. Daniel Geißler, Bo Zhou, Mengxi Liu, Sungho Suh, and Paul Lukowicz. The power of training: How different neural network setups influence the energy demand. In *International Conference on Architecture of Computing Systems*, pages 33–47. Springer, 2024.
21. Daniel Geißler, Bo Zhou, Paul Lukowicz, and RPTU Kaiserslautern-Landau. Latent inspector: An interactive tool for probing neural network behaviors through arbitrary latent activation. In *IJCAI*, pages 7127–7130, 2023.
22. Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1735–1742, 2006.
23. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
24. Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8, 2017.
25. Ylber Januzaj, Edmond Beqiri, and Artan Luma. Determining the optimal number of clusters using silhouette score as a data mining technique. *International Journal of Online & Biomedical Engineering*, 19(4), 2023.
26. Mahmut Kaya and Hasan Sakir Bilge. Deep metric learning: A survey. *Symmetry*, 11(9):1066, 2019.
27. Diederik Kinga, Jimmy Ba Adam, et al. A method for stochastic optimization. In *International conference on learning representations (ICLR)*, volume 5. California;, 2015.
28. Shashank Kotyan, Pin-Yu Chen, and Danilo Vasconcellos Vargas. Linking robustness and generalization: A k* distribution analysis of concept clustering in latent space for vision models. *arXiv preprint arXiv:2408.09065*, 2024.
29. Alex Krizhevsky et al. Learning multiple layers of features from tiny images. *utoronto.edu*, 2009.
30. Brian Kulis et al. Metric learning: A survey. *Foundations and Trends in Machine Learning*, 5(4):287–364, 2013.
31. Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed Chi. Fairness without demographics through adversarially reweighted learning. *Advances in neural information processing systems*, 33:728–740, 2020.
32. Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216, 2018.

33. Hrang Kap Lian, Kitsana Waiyamai, Slavakis Konstantinos, and Choochart Haruechaiyasak. Discriminative dimension selection for enhancing the interpretability and performance of clustering output. In *2024 16th International Conference on Knowledge and Smart Technology (KST)*, pages 178–183. IEEE, 2024.
34. Guangcan Liu, Zhouchen Lin, and Yong Yu. Robust subspace segmentation by low-rank representation. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 663–670, 2010.
35. Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017.
36. Chaochao Lu, Yuhuai Wu, José Miguel Hernández-Lobato, and Bernhard Schölkopf. Invariant causal representation learning for out-of-distribution generalization. In *International Conference on Learning Representations*, 2021.
37. Yair Movshovitz-Attias, Alexander Toshev, Thomas Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 360–368, 2017.
38. Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.
39. Tehreem Qamar and Narmeen Zakaria Bawany. Understanding the black-box: towards interpretable and reliable deep learning models. *PeerJ Computer Science*, 9:e1629, 2023.
40. Oren Rippel, Manohar Paluri, Piotr Dollar, and Lubomir Bourdev. Metric learning with adaptive density discrimination. *arXiv preprint arXiv:1511.05939*, 2015.
41. Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.
42. Sergei Rybakov, M. Lotfollahi, Fabian J Theis, and F. A. Wolf. Learning interpretable latent autoencoder representations with annotations of feature sets. *bioRxiv*, 2020.
43. Emrullah ŞAHiN, Naciye Nur Arslan, and Durmuş Özdemir. Unlocking the black box: an in-depth review on interpretability, explainability, and reliability in deep learning. *Neural Computing and Applications*, pages 1–107, 2024.
44. Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823. IEEE, 2015.
45. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
46. Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1857–1865, 2016.
47. Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.
48. Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen. Adversarial cross-modal retrieval. In *25th ACM international conference on Multimedia*, pages 154–162, 2017.

49. Fei Wang and Jimeng Sun. Survey on distance metric learning and dimensionality reduction in data mining. *Data mining and knowledge discovery*, 29(2):534–564, 2015.
50. Qian Wang and Ke Chen. Zero-shot visual recognition via bidirectional latent embedding. *International Journal of Computer Vision*, 124:356–383, 2017.
51. Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. In *Journal of Machine Learning Research*, volume 10, pages 207–244, 2009.
52. Tianfu Wu and Xi Song. Towards interpretable object detection by unfolding latent structures. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6033–6043, 2019.
53. Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.
54. Lu Yu, Bartlomiej Twardowski, Xialei Liu, Luis Herranz, Kai Wang, Yongmei Cheng, Shangling Jui, and Joost van de Weijer. Semantic drift compensation for class-incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6982–6991, 2020.
55. Yujing Zhou and Dubo He. Multi-target feature selection with adaptive graph learning and target correlations. *Mathematics*, 12(3):372, 2024.