Stationary Representations: Optimally Approximating Compatibility and Implications for Improved Model Replacements

Niccolò Biondi Federico Pernici Simone Ricci Alberto Del Bimbo
DINFO (Department of Information Engineering), University of Florence, Italy,
MICC (Media Integration and Communication Center),
name.surname@unifi.it

Abstract

Learning compatible representations enables the interchangeable use of semantic features as models are updated over time. This is particularly relevant in search and retrieval systems where it is crucial to avoid reprocessing of the gallery images with the updated model. While recent research has shown promising empirical evidence, there is still a lack of comprehensive theoretical understanding about learning compatible representations. In this paper, we demonstrate that the stationary representations learned by the d-Simplex fixed classifier optimally approximate compatibility representation according to the two inequality constraints of its formal definition. This not only establishes a solid foundation for future works in this line of research but also presents implications that can be exploited in practical learning scenarios. An exemplary application is the now-standard practice of downloading and fine-tuning new pre-trained models. Specifically, we show the strengths and critical issues of stationary representations in the case in which a model undergoing sequential fine-tuning is asynchronously replaced by downloading a better-performing model pre-trained elsewhere. Such a representation enables seamless delivery of retrieval service (i.e., no reprocessing of gallery images) and offers improved performance without operational disruptions during model replacement. Code available at: https://github.com/miccunifi/iamcl2r.

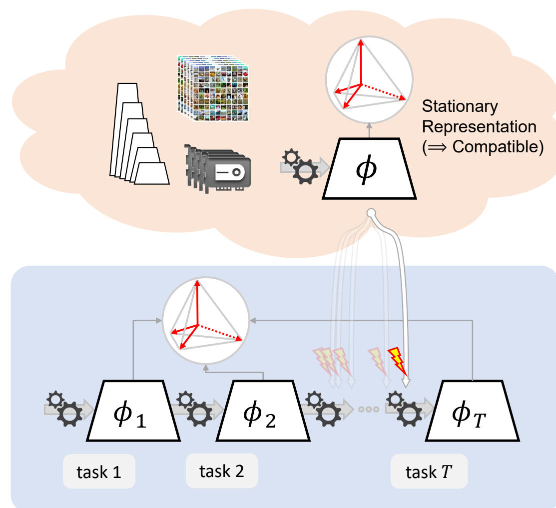


Figure 1. Improved Asynchronous Model Compatible Lifelong Learning Representation (IAM-CL^{2}R pronounced "*I am clear*"). In the process of lifelong learning, a model is sequentially fine-tuned and asynchronously replaced with improved third-party models that are pre-trained externally. Stationary representations ensure seamless retrieval services and better performance, without the need to reprocess gallery images.

1. Introduction

By learning powerful internal feature representations from data, Deep Neural Networks (DNNs) [1–4] have made tremendous progress in some of the most challenging search tasks such as face recognition [5–9], person re-identification [10–12], image retrieval [13–15] and this significance also extends to a variety of other data modalities [16, 17]. Although all of the works mentioned above have focused on learning feature representations from *static* and, more recently, *dynamic* datasets [18–21], the now-standard practice is downloading and fine-tuning representations from models pre-trained elsewhere [22, 23]. These "third-party" pre-trained models often incorporate new data, utilize alternative architectures, adopt different loss functions or more in general provide novel methodologies. Whether applied individually or combined, these advancements aim to encapsulate the field's rapid progress within a single unified model [24]. This greatly facilitates the exploitation of internally learned semantic representations, particularly as models, datasets, and computational infrastructure continue to expand in size, complexity, and cost [25, 26].

The challenge of fully exploiting such standard practice in retrieval/search systems has to deal with the underlying problem of *compatible learning* [27–29]. That is the desire to align the representation of different models trained with different data, initialization seeds, loss functions, or alternative architectures—either individually or in combination. In such applications, maintaining alignment is crucial to minimize the need for repeated reprocessing of gallery images for feature extraction each time a new pre-trained model becomes available [24]. Reprocessing is not only computationally intensive but may also be unsustainable for extensive gallery sets [25, 26, 30] or unfeasible if the original images are no longer accessible due to privacy concerns [31]. This holds across various typical galleries: social networks update millions of images every month, while in robotics and automotive domains, the update rate can be as rapid as hundreds of images every second. Similarly, in textual domains, books can be structured into chapters, paragraphs, and sentences, enabling the capture of semantic relationships between these segments. While a similar organizational principle can be structured for the web with LLMs [17, 32], the challenge lies in the impracticality of reprocessing such extensive content with each advancement in representation models. Although recent research has shown the effectiveness of compatible representation learning [27–29, 33–41], there is still a lack of comprehensive theoretical understanding about compatibility.

This paper introduces a theorem that demonstrates how the stationary representations proposed in [42, 43] optimally approximate compatibility according to the two inequality constraints of its formal definition as provided in [27]. This not only establishes a solid foundation for future works, but also presents implications that can be exploited fine-tuning third-party models without the need of reprocessing gallery images. Specifically, we show that a continuously fine-tuned model can be asynchronously replaced by downloading a higher-performing, pre-trained model from an external source. Due to stationarity (and therefore optimal compatibility), such a replacement provides seamless retrieval services with improved performance, eliminating the need for image gallery reprocessing. We refer to this scenario as Improved Asynchronous Model Compatible Lifelong Learning Representation (IAM-CL^{2}R pronounced "*I am clear*"). Fig. 1 illustrates the relationship between sequential fine-tuning and model replacement. Furthermore, as will be elaborated in the related work section, our foundation draws connections with the Neural Collapse phenomenon [44] and its associated theory.

Our second contribution is related to a specific challenge that arises: the tendency of the old and the new replaced models to align at their first-order statistics, an inherent property of stationary representation. Consequently, cross-entropy based prediction errors alone, when fine-tuning the representation, may not fully capture higher-order dependencies. To address this issue while preserving compatibility, we show that learning stationary representations using a convex combination of the cross-entropy loss and the infoNCE loss [45] is equivalent to training under one of the compatibility inequality constraints in [27]. This combined loss, termed Higher-Order Compatibility (HOC), distinguishes itself from the use of cross-entropy alone by capturing higher-order dependencies and optimally approximating compatibility.

2. Related Work

Neural Collapse. Neural Collapse (NC) is an empirical phenomenon that demonstrates the alignment between features and the classifier in a symmetric configuration [44]. Specifically, each class feature vector and its corresponding class prototype vector align with each other (i.e., collapse onto the same vector), forming a regular Simplex geometry in a subspace of the representation space. This particular configuration, which results in maximal separation of the collapsed vectors, is also referred to as a regular Simplex ETF (Equiangular Tight Frame). As training progresses and the training phase goes beyond zero classification error, the network increasingly approaches collapse. Notably, this also agrees with the double descent generalization regime observed within the same training phase [46]. The two phenomena together indicate a form of stable steady-state for the internal representations of Deep Neural Networks.

Prior to the observation of neural collapse, other research applied the steady-state of the Simplex geometry directly from the beginning of training. The fixed classifier with mutually orthogonal prototypes, introduced in [47], firstly demonstrates no degradation in classification performance. Building on this initial model, the regular polytope fixed classifiers—such as the d-Simplex, d-Cube, and d-Orthoplex—advance the concept further by observing stationary and maximally separated representations, as introduced in [48] and further detailed in [42]. Prior to these developments, [49] delved into the early energy-based investigations of symmetric and maximal separation in the representation space. The distinction between the natural emergence of a regular Simplex ETF and intentionally fixing the regular Simplex geometry at the beginning of training is that prior fixing can preserve regions in the representation space for future classes, as introduced in [50] and more recently in [51] and [52]. Our work takes advantage of this preservation for future classes, allowing third-party representation models to be trained from scratch and fine-tuned, while mitigating the interference in the representation space of the classes involved in both processes.

As neural collapse is related to the interaction between the neural network's final and penultimate layers, it offers a tool to examine training dynamics and convergence, as introduced in [53] and [54] under the name of Unconstrained Feature Model (UFM) and Layered Peeled Model (LPM), respectively. In both [55] and [56], the favorable convergence of fixing the final classifier according to the UFM is demonstrated. In [54], it is shown that training on imbalanced datasets does not necessarily result in NC. Additional observations from [57] suggest that NC can emerge in both imbalanced and long-tail scenarios when the classifier is fixed to a d-Simplex geometry. Further detailed results on NC are presented in [58]. Our proof is based on the assumption from the UFM and LPM that the backbone has sufficient expressiveness to allow for the independent study of each feature. Our proof is also based on the assumption of d-Simplex fixed classifier, whose inherent symmetry allows to reduce the extent of the analysis to a single pairwise class interaction, as it causes all the interactions to be identical.

Compatible Representations Learning. Compatible representations broadly refer to the ability to align different learned representations, as discussed in [59–63]. The distinction outlined in [27] is that the alignment of models should be achieved without wasting the information learned from new data. This capability is typically evaluated in a query and gallery setting, where query and gallery features are extracted from two different representation models. The model for the query is trained using an extended dataset that includes additional data not present in the one used for training the gallery's model. The study in [27] further presents a method called Backward Compatible Training (BCT), which applies regularization to a new model using the classifier from the previous learning phase. This approach implicitly aligns the current improved model with the previously trained classifier, which is kept fixed. Several other methods have adopted this basic working principle: The fundamental aspect of this principle is that the challenge of model alignment is primarily demanded by the new model, which must learn from both the additional and the old data how to compensate for the inadequate representations of the previously learned models. Conversely, as also recently highlighted in [41], methods such as [64] or the more recent [28, 65, 66] train a lightweight transformation to convert old representations into new ones for backward compatibility. However, these methods do not entirely eliminate the re-processing cost. As the number of chained mappings increases, the entire chain necessitates re-evaluation each time the representation model is updated. This makes them unsuitable for sequential learning and large gallery-sets. While its primary focus is on classification, the study in [37] is one of the first methods employing sequential chaining transformations for aligning representations within a common reference space. The works in [39] and [38] bypass the use of chaining transformations, focusing instead on aligning representations for compatibility purposes in lifelong learning scenarios. Both approaches leverage auxiliary losses to ensure similarity among previously learned representations. Additionally, [39] achieves alignment with an absolute reference through the use of fixed classifiers, in line with the neural collapse phenomenon.

The work in [41] argues that there is an inherent trade-off in the definition of compatibility introduced in [27], which inspires them to "hold" incompatible information of the new model on additional orthogonal dimensions to avoid this conflict. Their argument seems to be in line with the recent work [29] and [39] based on stationarity in which (nearly) orthogonal dimensions are pre-allocated from the beginning using a regular d-Simplex fixed classifier. In this paper, we establish a formal relationship among compatibility, neural collapse, and stationarity, showing that stationarity provides an optimal approximation to the compatibility definition formulated in [27].

3. Theoretical Results

3.1. Stationarity and Compatibility

Preliminaries. Let $\mathcal{G} = \{\mathbf{x}_i\}_{i=1}^{N_g}$ be a gallery-set composed of a set of N_g images $\mathbf{x}_i \in \mathbb{R}^D$ with class labels from $\mathcal{Y} = \{y_i\}_{i=1}^{L}$ and $\Phi^{\mathcal{G}} = \{\phi(\mathbf{x}_i) \in \mathbb{R}^d \,|\, \forall \mathbf{x}_i \in \mathcal{G}\}$ be the set of feature vectors of the gallery-set $\mathcal{G}$ obtained with representation model ϕ. Let $\mathcal{Q} = \{\mathbf{x}_i\}_{i=1}^{N_q}$ be a query-set composed of N_q images $\mathbf{x}_i \in \mathbb{R}^D$ and $\Phi^{\mathcal{Q}} = \{\phi(\mathbf{x}_i) \in \mathbb{R}^d \,|\, \forall \mathbf{x}_i \in \mathcal{Q}\}$ be the set of feature vectors of the query-set $\mathcal{Q}$ obtained with ϕ. Visual search is performed using a distance function $\mathrm{d}(\cdot,\cdot)$ to identify the closest gallery features to the query features.

Let $\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_T$ be a sequence of T tasks, where each task $\mathcal{T}$ is composed of labeled images $\mathbf{x}_i$ of class $y_i \in \mathcal{K}$ with $\mathcal{K}$ the set of classes in $\mathcal{T}$. At task t, the model ϕ_t is fine-tuned starting from the previous representation model ϕ_{t-1}. Compatibility between the current model ϕ_t and a previous model ϕ_k, with $k < t$, is achieved when the feature vector of any query image obtained with ϕ_t, the set $\Phi_t^{\mathcal{Q}}$, can be compared with feature vectors in $\Phi_k^{\mathcal{G}}$ without reprocessing the gallery-set. The following provides a formal definition of compatibility [27]:

Definition 1 (Compatibility) *Given two representation models ϕ_t and ϕ_k, with ϕ_t learned after ϕ_k, ϕ_t and ϕ_k are compatible according to the distance function $\mathrm{d}(\cdot,\cdot)$ if it holds:*

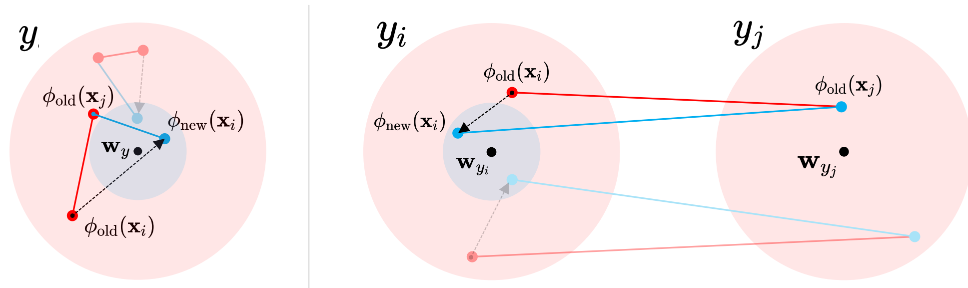
$$\mathrm{d}\big(\phi_k(\mathbf{x}_i), \phi_t(\mathbf{x}_j)\big) \leq \mathrm{d}\big(\phi_k(\mathbf{x}_i), \phi_k(\mathbf{x}_j)\big) \quad \forall\,(i,j) \in \big\{(i,j)\,|\,y_i = y_j\big\} \tag{1a}$$

and

$$\mathrm{d}\big(\phi_k(\mathbf{x}_i), \phi_t(\mathbf{x}_j)\big) \geq \mathrm{d}\big(\phi_k(\mathbf{x}_i), \phi_k(\mathbf{x}_j)\big) \quad \forall\,(i,j) \in \big\{(i,j)\,|\,y_i \neq y_j\big\} \tag{1b}$$

$$\textit{with } k < t, \quad t = (2,3,\ldots,T), \;\; k = (1,2,\ldots,T-1).$$

Main Result. In this paragraph, we state and prove that learning stationary feature representations according to a d-Simplex fixed classifier necessarily implies optimal approximation of the compatibility as defined in Eqs. 1a and 1b.



(a) Same class

(b) Different class

Figure 2. Key concepts and relationships underlying Theorem 1. Distances in feature space of two distinct samples within their hyperballs before and after model update, with the update process represented by a dotted arrow. (*a*): Distances between samples $\mathbf{x}_i$ and $\mathbf{x}_j$ of the same class y before (red) and after (cyan) model update. (*b*): Distances between samples $\mathbf{x}_i$ of class y_i and $\mathbf{x}_j$ of class y_j, before (red) and after (cyan) model update. Compatibility is verified by computing the expected lengths of the segments and verifying if they satisfy the inequalities of the compatibility definition. A transparently colored instance shows counter-intuitive distance behavior. Expectation reveals the underlying pattern of approximation.

The formulation involves examining the expected distance between feature points before and after a learning update in a high-dimensional space, where the feature points are assumed to be distributed in hyperballs (i.e., high dimensional ball) centered at the prototypes of the d-Simplex fixed classifier. This abstraction allows for mathematical manipulation and analysis of the cluster as a single entity rather than individual points.

Theorem 1 (Stationarity $\implies$ Compatibility) *Let* $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_K]$ *be the* $d \times K$ *matrix of a* d*-Simplex fixed classifier with* K *pre-allocated classes. Given two tasks,* $\mathcal{T}_k$ *and* $\mathcal{T}_t$*. The task* $\mathcal{T}_t$ *is derived from* $\mathcal{T}_k$ *by incorporating an additional training set* $\Delta\mathcal{T}$*, such that* $\mathcal{T}_t = \mathcal{T}_k \cup \Delta\mathcal{T}$*. The combined task,* $\mathcal{T}_t$*, comprises a set of classes each denoted by* y*, where* $y \in \{1, 2, \dots, K_t\}$ *and* $K_t < K$*. Under the assumption that learning the new task* $\mathcal{T}_t$ *causes the hyperball* $\mathcal{B}_k(\mathbf{w}_y)$ *with radius* r_k^y *to shrink into a smaller hyperball* $\mathcal{B}_t(\mathbf{w}_y)$*, i.e.,* $r_t^y \leq r_k^y$ *for all* y *in the set* $\{1, 2, \dots, K_k\}$*, then it necessarily follows that* ϕ_t *and* ϕ_k *optimally approximate the compatibility inequality constraints as defined in Def. 1 in expectation.*

The proof is available in the Appendix.

Discussion. The Theorem relies on two main assumptions: the use of a d-Simplex fixed classifier [42] and the model's sufficient expressiveness, as described in the UFM abstraction [53, 54]. The latter assumption enables us to consider features independently[1]. While the former allows focusing on a single pairwise class interaction, since interactions with all other classes are symmetrically similar and cannot change. Fig. 2 illustrates the key concepts and relationships presented in Theorem 1.

Without loss of generality, the Theorem considers two distinct hyperballs of different radius $\mathcal{B}_{\text{new}}(\mathbf{w}_y)$ and $\mathcal{B}_{\text{old}}(\mathbf{w}_y)$ representing the semantic clusters of a generic class y, respectively before and after a generic learning update. The assumption that features are distributed in hyperballs stems from the margin-based softmax loss[2] introduced in [67]. This interpretation has since been utilized in various studies, such as SphereFace [68] and ArcFace [8]. Besides the margin formulation, empirical evidence, such as Neural Collapse [44], shows that class features not only cluster around their associated prototypes but also, with sufficient training epochs, collapse into them, resulting in hyperballs tightening around the prototypes. Due to the stationarity property induced by the d-Simplex classifier $\mathcal{B}_{\text{new}}(\mathbf{w}_y)$ and $\mathcal{B}_{\text{old}}(\mathbf{w}_y)$ hyperballs have the same center in the representation space on the classifier prototype $\mathbf{w}_y$. After the learning step, $\mathcal{B}_{\text{new}}(\mathbf{w}_y)$ has a shorter radius (i.e., adding new information improves the discrimination capability of the model [69–72]).

In particular, Fig. 2a shows the case in which feature vectors are from samples of the same class. As defined in Eq. 1a compatibility requires that, after updating, the distance between $\phi_{\text{new}}(\mathbf{x}_i)$ (in the cyan hyperball) and $\phi_{\text{old}}(\mathbf{x}_j)$ (in red hyperball) is less than or equal to distance between $\phi_{\text{old}}(\mathbf{x}_i)$ and $\phi_{\text{old}}(\mathbf{x}_j)$. The figure displays two configurations: one where the condition is met and another where it is not met (shown in transparent colors).

Fig. 2b shows the case in which the feature vectors are

[1]Essentially, the Neural Collapse phenomenon, which is observed across various networks and datasets, *also appears* in a two-layer neural network when assuming input feature independence (i.e., a UFM). This equivalence supports the assumption that: 1) real network backbones are typically expressive enough to learn features as independent entities, and 2) UFM can be used as a tool to study neural networks properties.

[2]The margin enforces the confinement of features within a hyperball or a hyperdisc (the local approximation of a hypercap) around class prototypes. A disc in high-dimensional space can be considered a hyperball when referring to its filled volume.

from samples of different classes. As defined in Eq. 1b compatibility requires that, after updating, the distance between $\phi_{\text{new}}(\mathbf{x}_i)$ of class y_i (in the cyan hyperball centered in $\mathbf{w}_{y_i}$) and $\phi_{\text{old}}(\mathbf{x}_j)$ of class y_j (in the red hyperball centered in $\mathbf{w}_{y_j}$) is greater than or equal to than the distance between $\phi_{\text{old}}(\mathbf{x}_i)$ and $\phi_{\text{old}}(\mathbf{x}_j)$. The Theorem establishes that, on average, this condition cannot be optimally satisfied and that stationarity is the best approximation achievable under the given constraints. A detailed justification for this is provided in the proof of Theorem 1, with a clearer and more focused exposition presented as a Corollary in the Appendix.

Informally, the proof of the Theorem starts with the premise that, upon retraining a model, the probability of finding a class feature near the corresponding class prototype from the old model—an indicator of compatibility between the two models—is nearly zero. Subsequently, the proof establishes that the optimal approximation for a compatible representation is obtained when the average distance between the same hyperball in two distinct learned models is minimized. This minimization occurs when the two corresponding hyperballs are centered at the same class prototype and when adding more classes does not alter this distance, i.e., the stationarity condition.

Our formulation calculates the average distance between hyperballs based on the Ball Line Picking problem, which determines the expected length of a line segment that connects two random points inside a hyperball [73–78]. Differently from that problem, our theorem considers a line segment connecting two random points in two distinct hyperballs, each with a different radius. Specifically, we analyze the cases as shown in Fig. 2. These hyperballs represent the "class-state" before and after the learning step during each model update. Closed-form solutions are not available for this problem, except in a specific two-dimensional case [79].

3.2. Stationarity and Higher-Order Alignment

A specific challenge arises when fine-tuning stationary learned representation models, for example in the IAM-CL^{2}R setting of Fig. 1. In this case the old and the new models align at the first-order statistics, an inherent property of stationarity [42]. The consequence is that cross-entropy based prediction errors may not fully capture higher-order dependencies in representation space. We conjecture that simple cross-entropy mostly focuses on prediction errors related to the forgetting of the internal representation which may not promote compatibility when the representation model is largely aligned. To address this problem, we show that adding the infoNCE loss function [45, 80] is equivalent to training with the cross-entropy loss under one of the compatibility constraints while capturing higher-order dependencies.

The loss for training at task t the stationary representation model ϕ_t assumes the form [42]:

$$\mathcal{L}_{\text{SCE}}(\phi_t) = \\ = -\sum_B \log\left(\frac{\exp\left(\mathbf{W}_{y_i}^\top \phi_t(\mathbf{x}_i)\right)}{\sum_{j=1}^{K_t}\exp\left(\mathbf{W}_j^\top \phi_t(\mathbf{x}_i)\right) + \sum_{j=K_t+1}^{K}\exp\left(\mathbf{W}_j^\top \phi_t(\mathbf{x}_i)\right)}\right) \tag{2}$$

where $\mathbf{W}_j^\top \in \mathbb{R}^d$ denotes the j-th column of the d-Simplex classifier matrix $\mathbf{W} \in \mathbb{R}^{d\times K}$, being K the number of pre-allocated classes, $K_t = |\bigcup_{i=1}^t \mathcal{K}_i|$ the number of classes learned until time t with $K_t < K$, and B is a mini-batch of samples of $\mathcal{T}_t$. The first term in the denominator accounts for the classes learned until t. The second term accounts for future classes, preserving dedicated regions in the representation space. This ensures that adding new classes minimally impacts the representation of previously learned classes [29, 39, 50, 81]. We train the representation model ϕ_t with the following convex combination, namely:

$$\mathcal{L}_{\text{HOC}}(\phi_t) = \lambda\mathcal{L}_{\text{SCE}}(\phi_t) + (1-\lambda)\,\mathcal{L}_{\text{NCE}}(\phi_t,\phi_{t-1}), \quad \text{with} \quad \lambda \in [0,1] \tag{3}$$

where: $\mathcal{L}_{\text{SCE}}(\phi_t)$ is the cross-entropy loss of Eq. 2, and

$$\mathcal{L}_{\text{NCE}}(\phi_t,\phi_{t-1}) = -\sum_B \log\left(\frac{\Delta\big(\phi_{t-1}(\mathbf{x}_i),\phi_t(\mathbf{x}_i)\big)}{\sum_{j\neq i}\Delta\big(\phi_{t-1}(\mathbf{x}_i),\phi_t(\mathbf{x}_j)\big)}\right) \tag{4}$$

with

$$\Delta\big(\phi_{t-1}(\mathbf{x}_i),\phi_t(\mathbf{x}_j)\big) = \exp\left(\tau\cdot\frac{\phi_{t-1}(\mathbf{x}_i)\phi_t(\mathbf{x}_j)}{||\phi_{t-1}(\mathbf{x}_i)\phi_t(\mathbf{x}_j))||}\right) \tag{5}$$

is the contrastive loss [45, 80] based on τ-scaled cosine similarity between $\phi_{t-1}(\mathbf{x}_i)$ and $\phi_t(\mathbf{x}_j)$. We show that training the representation model with the $\mathcal{L}_{\text{HOC}}$ of Eq. 3 is both: (1) able to capture higher-order dependencies between old and new model representations and (2) equivalent to learning under the compatibility constraints in Def. 1a. We refer to this loss as the Higher-Order Compatibility loss ($\mathcal{L}_{\text{HOC}}$).

Through Theorem 1 presented in the previous section, we establish that the constraint of Eq. 1a cannot be exploited in combination with the constraint of Eq. 1b. Based on this result we show that, under no specific conditions, the constrained optimization problem using solely the inequality constraint of Eq. 1a:

$$\begin{aligned} \operatorname*{argmin}_{\phi_t} \quad & \mathcal{L}_{\text{SCE}}(\phi_t) \\ \text{s.t.} \quad & \mathrm{d}\big(\phi_k(\mathbf{x}_i),\phi_t(\mathbf{x}_j)\big) - \mathrm{d}\big(\phi_k(\mathbf{x}_i),\phi_k(\mathbf{x}_j)\big) \leq 0 \\ & \forall\, y_i = y_j \end{aligned} \tag{6}$$

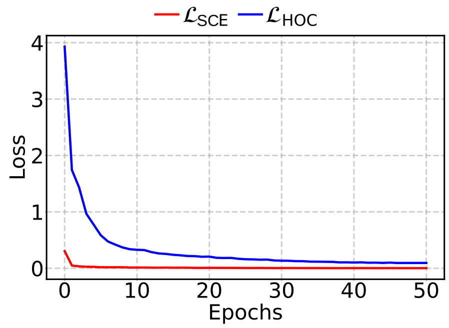


Figure 3. Training loss of a d-Simplex fixed classifier during a model update. Values are the cross-entropy loss of Eq. 2 (red line) and the loss of Eq. 3 (blue line). Models are trained on MNIST.

can be transformed into a tractable form. Rooted in the work of [82], this transformation not only provides an approach to solve the tractability issue but, within the context of compatibility, it also allows preserving the optimality as outlined in the proof of Theorem 1. As shown in [82], the model for a constrained problem like Eq. 6 can be equivalently learned with a convex combination of the cross-entropy loss and the Kullback-Leibler divergence function.

On the other hand, as discussed in [83], the contrastive loss $\mathcal{L}_{\text{NCE}}(\phi_t, \phi_{t-1})$ can be approximated as the Kullback-Leibler divergence between the product of the marginals of the joint distribution of ϕ_t and ϕ_{t-1}. Moreover, $\mathcal{L}_{\text{NCE}}(\phi_t, \phi_{t-1})$ also approximates the mutual information between ϕ_t and ϕ_{t-1}, thereby enabling to capture higher-order dependencies between consecutive updates of the model. As a consequence, training with the loss in Eq. 3 is equivalent to the optimal classifier for the constrained optimization problem stated in Eq. 6 and at the same time, thanks to the term $\mathcal{L}_{\text{NCE}}(\phi_t, \phi_{t-1})$, takes into account higher-order variations between $\phi_{t-1}(\mathbf{x}_i)$ and $\phi_t(\mathbf{x}_j)$. In the following, we call training the representation model using d-Simplex with $\mathcal{L}_{\text{HOC}}$ as d-Simplex-HOC.

In Fig. 3, we illustrate the effects of $\mathcal{L}_{\text{HOC}}$ compared to the cross-entropy loss. We use a toy example with the LeNet++ CNN architecture [84] with the d-Simplex fixed classifier. The model is initially trained on the first five MNIST classes and then fine-tuned on all ten classes. The cross-entropy training error (red curve) converges rapidly to low values. In contrast, the convergence with the $\mathcal{L}_{\text{HOC}}$ loss (blue curve) is more gradual, which allows for the capture of richer information during back-propagation.

4. Experimental Verification

Referring to the IAM-CL^{2}R learning scenario presented in Fig. 1, this section provides empirical evidence to verify the practical implications of the theoretical results discussed earlier.

4.1. Datasets and Settings

Pre-trained Models. We pre-train our models in a supervised manner using the ImageNet32 [85]. Three distinct models are pre-trained on ImageNet32 with 100, 300, and 600 classes. The model trained with 100 classes is used to initialize the model before fine-tuning on the sequence of tasks. The other two models are used to simulate the practice of downloading and fine-tuning pre-trained models and serve as third-party models that will replace the current one undergoing fine-tuning.

Fine-tuning. We replicate the fact that dataset size for training third-party models is typically significantly larger than the dataset size used for fine-tuning [86]. According to this, pre-trained models are fine-tuned with a reduced version of CIFAR100 [87] denoted in this paper as CIFAR100R.

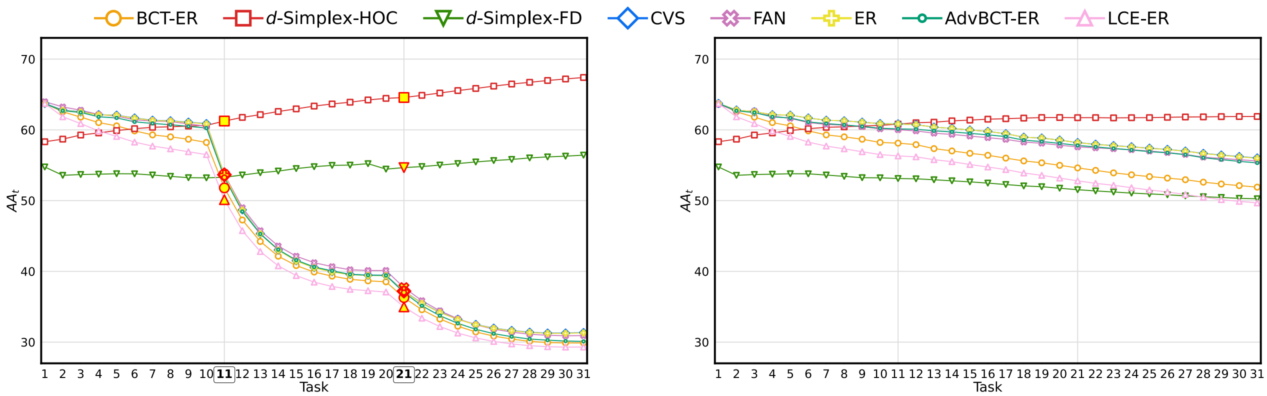
We considered two distinct task sequences consisting of 7 and 31 tasks each. We fine-tune the pre-trained model with an initial task comprising 10 classes. Subsequently, for the sequences of 7 and 31 tasks, the respective tasks contain 15 and 3 classes each. The fine-tuning process incorporates incoming task data, consisting of 300 images per class, and utilizes an episodic memory that stores 20 images from each class of previous tasks.

Model Replacement. In our experiments, we verify the impact of replacing the current fine-tuned model with two improved models pre-trained elsewhere. The two replacements occur while fine-tuning on CIFAR100R: at the third and fifth tasks in the shorter sequence, and at the eleventh and twenty-first tasks in the longer sequence. We also consider the challenging scenario of improved model replacement considering more sophisticated network architectures.

The d-Simplex fixed classifier is pre-allocated with a number of classes K, ensuring enough space to accommodate future classes for both pre-training and fine-tuning. Class assignments for pre-training are made from left to right, and for fine-tuning, from right to left. This straightforward convention is used to ensure that classes assigned for pre-training and fine-tuning remain distinct, without overlap. Other non overlapping assignment methods could also be used.

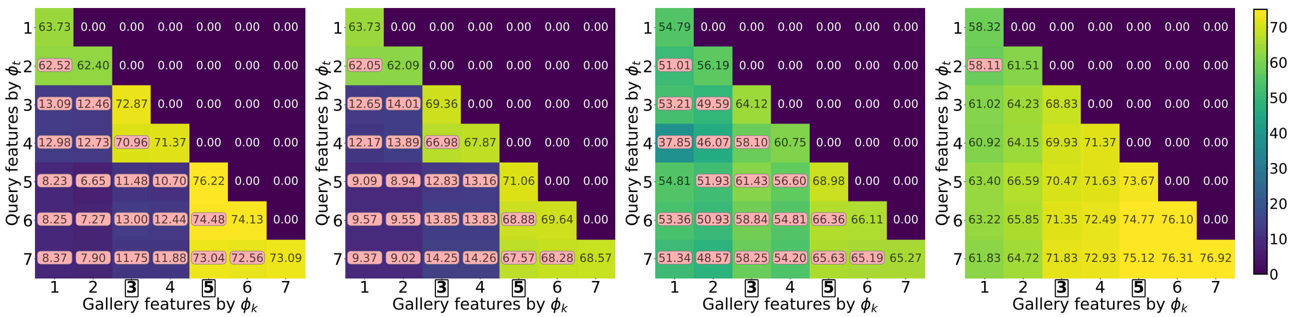
Network Architectures. We use ResNet18 [88] as network architecture. In the scenario using more sophisticated network architectures, we initially replace ResNet18 with SENet18 [89], followed by a subsequent replacement with a RegNetY_400MF [90].

Hyper-parameters. The ResNet18, SENet18, and RegNetY_400MF models were pre-trained on ImageNet32 using the following hyper-parameters: 300 epochs, a batch size of 128, and an initial SGD optimizer learning rate of 0.1, which was adjusted using a Cosine Annealing schedule. For each task used for fine-tuning, the model is trained for 70 epochs with a batch size of 128, starting with a learning rate of 0.001 that was reduced by a factor of 10 after the 50th and 64th epochs. The d-Simplex was pre-allocated with $K = 1024$



(a) Two replacements (b) No replacements

Figure 4. Average multi-model Accuracy (AA_t) evaluated across 31 tasks using CIFAR100R/10, showing: *(a)* model replacements at tasks 11 and 21 (indicated by yellow markers); *(b)* no model replacement.



(a) CVS [38] (b) BCT-ER [27] (c) d-Simplex-FD [39] (d) d-Simplex-HOC (in this paper)

Figure 5. Compatibility Matrices for d-Simplex-HOC, CVS, BCT-ER, and d-Simplex-FD on CIFAR100R/10 across 7 tasks. Model replacements at tasks 3 and 5 are highlighted in bold. Entries failing to meet compatibility criteria as defined in [27] are marked with a light-red background.

classes (i.e., $d = K - 1$).

Performance Evaluation. The evaluation focuses on the open-set recognition task, in which separated datasets for training and evaluation are required. The standard 1:N search protocol, applicable to re-identification and similar tasks [27], is employed in the evaluation. To ensure strict separation between datasets, the CIFAR10 dataset is utilized for evaluation during fine-tuning with CIFAR100R. Specifically, the test set of CIFAR10, comprising 10,000 images, is used as the gallery set, while its training set of 50,000 images serves as the query set.

Following [27] and [29], we measure performance progression across the two sequences of tasks using two established metrics: Average Compatibility (AC) and Average multi-model Accuracy (referred shortly as to AA_t). The metric AC quantifies the extent of compatibility across all possible pairs of model combinations by providing a normalized count of times in which compatibility is achieved. Conversely, AA_t calculates the mean accuracy across all combinations of the previously learned models until task t, providing an overall measure of accuracy.

4.2. IAM-CL^{2}R: Comparative Results

We performed a comparative analysis of d-Simplex-HOC against FAN [37], CVS [38], d-Simplex-FD [39], and the lifelong adapted versions of BCT [27] (BCT-ER), LCE [28] (LCE-ER), and AdvBCT [36] (AdvBCT-ER). The experiments also incorporate a baseline method, Experience Replay (ER), in which the model is fine-tuned using cross-entropy loss on data of the new task and an episodic memory. Ablation studies of IAM-CL^{2}R with the d-Simplex-HOC are provided in the Appendix.

Replacing: Same Architecture, Expanded Data. Fig. 4 presents the Average multi-model Accuracy at task t (AA_t) for learning scenarios with model replacement as depicted in Fig. 4a and for those without as depicted in Fig. 4b. The experiment involves fine-tuning a ResNet18 model across 31 tasks. The comparison provides insights into the performance benefits that can be obtained by replacing models when representations are trained in a compatible manner. The d-Simplex-HOC effectively incorporates improvements from model replacements, showing increased performance

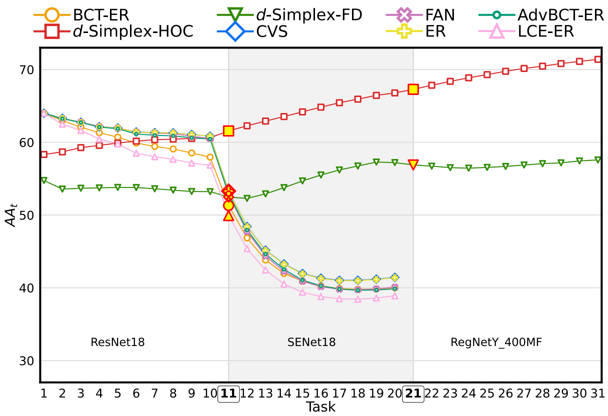


Figure 6. Plots of Average multi-model Accuracy (AA_t) for 31 tasks on CIFAR100R/10, showing the impact of model replacements with different network architectures at tasks 11 and 21.

compared to the case without model replacement, as indicated in Fig. 4b. The d-Simplex-FD demonstrates a similar capability, though to a reduced extent. The other methods have a clear performance decay after model replacements and end up with a worse performance than the case without replacement. This can be attributed to the fact that after replacement, fine-tuning is applied to a model obtained by retraining the network from scratch, leading to an entirely different representation.

Further performance details, as indicated by the self and cross-test accuracy values, are shown according to the compatibility matrices [27, 29]. Fig. 5 shows these values for CVS, BCT-ER, d-Simplex-FD, and d-Simplex-HOC in the 7 tasks sequence. The values reveal that the d-Simplex-HOC effectively leverages the improved expressive power of the models after replacement, except in one instance. This exception, where the model is not compatible and the cross-test accuracy falls below the self-test accuracy, is shown in Fig. 5d. Both CVS and BCT-ER score near zero cross-tests accuracy after model replacements as indicated by the values in the blue sub matrix blocks shown in Fig. 5a and Fig. 5b. This leads to mostly non-compatible representations. Although both d-Simplex-HOC and d-Simplex-FD utilize the d-Simplex fixed classifier to learn stationary representations, the former shows better performance. This can be attributed to the high-order alignment achievable through the HOC loss. To provide a full evaluation of compatibility, the AA_t of Fig. 4 is complemented with the Average Compatibility AC in Tab. 1. We also report the Average multi-model Accuracy AA_7 and AA_{31} for methods compared at the end of the 7-th and 31-th task, respectively. It is observed that, in both instances, all models—with the exception of d-Simplex-HOC—fail to achieve significant compatibility performance.

Replacing: Different Architectures, Expanded Data. Fig. 6 shows the performance of the evaluated methods when

	7 tasks		31 tasks	
METHOD	AC	AA_7	AC	AA_{31}
ER baseline	×	36.22	<0.01	31.30
FAN [37]	×	36.32	<0.01	30.79
BCT-ER [27]	×	35.59	×	29.88
LCE-ER [28]	×	34.89	×	29.30
AdvBCT-ER [36]	×	35.73	×	30.10
CVS [38]	×	36.31	0.01	31.34
d-Simplex-FD [39]	0.05	56.58	0.21	56.27
d-Simplex-HOC	**0.95**	**68.13**	**0.65**	**67.40**

Table 1. Compatibility metrics with CIFAR100R/10 for 7 tasks with model replacements at task 3 and task 5, and 31 tasks with model replacements at task 11 and task 21. "×" indicates the case in which compatibility is not achieved.

the original ResNet18 is replaced first by a SENet18 and then by a more expressive RegNetY_400MF. It is observed that the change of network architecture not only does not adversely affect compatibility in the d-Simplex-HOC but takes advantage of their more expressive representation power. In particular, direct comparison of Fig. 6 with Fig. 5 shows that d-Simplex-HOC improves performance gradually with each model replacement. This is in contrast to d-Simplex-FD, which does not demonstrate the same trends leading to a plateau around the 20-th task. Given the different feature sizes before and after the second model replacement with the RegNetY_400MF architecture—512 and 384, respectively—all methods except d-Simplex-HOC and d-Simplex-FD require non-trivial extensions to adapt to the changed feature size. According to this, for these methods, evaluation cannot be reported.

5. Conclusion

In this paper, we have investigated the concept of learning compatible representations through the principle of stationarity. We demonstrated that stationary representations optimally approximate compatibility according to its definition. We demonstrated that better model alignment through higher-order dependencies can be obtained by training with a loss derived from one of the compatibility inequality constraints. Finally, empirical evidence confirmed that stationary representations enable uninterrupted retrieval service allowing for fine-tuning and model replacement to occur concurrently and asynchronously with limited interference.

Acknowledgment: This work was partially supported by the European Commission under European Horizon 2020 Programme, grant number 951911 - AI4Media.

We acknowledge the CINECA award under the ISCRA initiative, for the availability of high-performance computing resources and support (ISCRA-C ID: HP10C4TIIM).

References

[1] 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. 1

[2] 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.

[3] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813, 2014.

[4] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. 1

[5] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014. 1

[6] Yi Sun, Ding Liang, Xiaogang Wang, and Xiaoou Tang. DeepID3: Face recognition with very deep neural networks. *arXiv preprint arXiv:1502.00873*, 2015.

[7] Rajeev Ranjan, Carlos D Castillo, and Rama Chellappa. L2-constrained softmax loss for discriminative face verification. *arXiv preprint arXiv:1703.09507*, 2017.

[8] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019. 4

[9] Qiang Meng, Shichao Zhao, Zhida Huang, and Feng Zhou. MagFace: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1

[10] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In *European Conference on Computer Vision*, 2018. 1

[11] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[12] Ergys Ristani and Carlo Tomasi. Features for multi-target multi-camera tracking and re-identification. In *European Conference on Computer Vision*, pages 6036–6046, 2018. 1

[13] Filip Radenović, Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondřej Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5706–5715, 2018. 1

[14] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning cnn image retrieval with no human annotation. *IEEE transactions on pattern analysis and machine intelligence*, 41(7):1655–1668, 2018.

[15] Wei Chen, Yu Liu, Weiping Wang, Erwin M Bakker, Theodoros Georgiou, Paul Fieguth, Li Liu, and Michael S Lew. Deep learning for instance retrieval: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 1

[16] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

[17] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 2

[18] Lucas Caccia, Rahaf Aljundi, Nader Asadi, Tinne Tuytelaars, Joelle Pineau, and Eugene Belilovsky. New insights on reducing abrupt representation change in online continual learning. In *International Conference on Learning Representations*, 2021. 1

[19] MohammadReza Davari, Nader Asadi, Sudhir Mudur, Rahaf Aljundi, and Eugene Belilovsky. Probing representation forgetting in supervised and unsupervised continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16712–16721, June 2022.

[20] Tommaso Barletti, Niccolo' Biondi, Federico Pernici, Matteo Bruni, and Alberto Del Bimbo. Contrastive supervised distillation for continual representation learning. *International Conference on Image Analysis and Processing*, 2022.

[21] Nader Asadi, MohammadReza Davari, Sudhir Mudur, Rahaf Aljundi, and Eugene Belilovsky. Prototype-sample relation distillation: towards replay-free continual learning. In *International Conference on Machine Learning*, pages 1093–1106. PMLR, 2023. 1

[22] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019. 1

[23] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1

[24] Colin Raffel. Building machine learning models like open source software. *Communications of the ACM*, 66(2):38–40, 2023. 1, 2

[25] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021. 1, 2

[26] Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *arXiv preprint arXiv:2206.14486*, 2022. 1, 2

[27] Yantao Shen, Yuanjun Xiong, Wei Xia, and Stefano Soatto. Towards backward-compatible representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6368–6377, 2020. 2, 3, 7, 8

[28] Qiang Meng, Chixiang Zhang, Xiaoqiang Xu, and Feng Zhou. Learning compatible embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9939–9948, 2021. 3, 7, 8

[29] Niccolo Biondi, Federico Pernici, Matteo Bruni, and Alberto Del Bimbo. Cores: Compatible representations via stationarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2, 3, 5, 7, 8

[30] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, 2019. 2

[31] W Nicholson Price and I Glenn Cohen. Privacy in the age of medical big data. *Nature medicine*, 25(1):37–43, 2019. 2

[32] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020. 2

[33] Chien-Yi Wang, Ya-Liang Chang, Shang-Ta Yang, Dong Chen, and Shang-Hong Lai. Unified representation learning for cross model compatibility. In *31st British Machine Vision Conference 2020, BMVC 2020*. BMVA Press, 2020. 2

[34] Binjie Zhang, Yixiao Ge, Yantao Shen, Shupeng Su, Chun Yuan, Xuyuan Xu, Yexin Wang, and Ying Shan. Towards universal backward-compatible representation learning. *arXiv preprint arXiv:2203.01583*, 2022.

[35] Rahul Duggal, Hao Zhou, Shuo Yang, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Compatibility-aware heterogeneous visual search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10723–10732, 2021.

[36] Tan Pan, Furong Xu, Xudong Yang, Sifeng He, Chen Jiang, Qingpei Guo, Feng Qian, Xiaobo Zhang, Yuan Cheng, Lei Yang, et al. Boundary-aware backward-compatible representation via adversarial learning in image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15201–15210, 2023. 7, 8

[37] Ahmet Iscen, Jeffrey Zhang, Svetlana Lazebnik, and Cordelia Schmid. Memory-efficient incremental learning through feature adaptation. In *European Conference on Computer Vision*, pages 699–715. Springer, 2020. 3, 7, 8

[38] Timmy S. T. Wan, Jun-Cheng Chen, Tzer-Yi Wu, and Chu-Song Chen. Continual learning for visual search with backward consistent feature embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16702–16711, June 2022. 3, 7, 8

[39] Niccolo Biondi, Federico Pernici, Matteo Bruni, Daniele Mugnai, and Alberto Del Bimbo. Cl2r: Compatible lifelong learning representations. *ACM Transactions on Multimedia Computing, Communications and Applications*, 18(2s):1–22, 2023. 3, 5, 7, 8

[40] Frederik Träuble, Julius Von Kügelgen, Matthäus Kleindessner, Francesco Locatello, Bernhard Schölkopf, and Peter Gehler. Backward-compatible prediction updates: A probabilistic approach. *Advances in Neural Information Processing Systems*, 34:116–128, 2021.

[41] Yifei Zhou, Zilu Li, Abhinav Shrivastava, Hengshuang Zhao, Antonio Torralba, Taipeng Tian, and Ser-Nam Lim. Btˆ 2: Backward-compatible training with basis transformation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11229–11238, 2023. 2, 3

[42] Federico Pernici, Matteo Bruni, Claudio Baecchi, and Alberto Del Bimbo. Regular polytope networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2021. 2, 4, 5

[43] Federico Pernici, Matteo Bruni, Claudio Baecchi, and Alberto Del Bimbo. Maximally compact and separated features with regular polytope networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 2

[44] Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020. 2, 4

[45] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2, 5

[46] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019. 2

[47] Elad Hoffer, Itay Hubara, and Daniel Soudry. Fix your classifier: the marginal value of training the last weight layer. In *International Conference on Learning Representations*, 2018. 2

[48] Federico Pernici, Matteo Bruni, Claudio Baecchi, and Alberto Del Bimbo. Fix your features: Stationary and maximally discriminative embeddings using regular polytope (fixed classifier) networks. *arXiv preprint arXiv:1902.10441*, 2019. 2

[49] Weiyang Liu, Rongmei Lin, Zhen Liu, Lixin Liu, Zhiding Yu, Bo Dai, and Le Song. Learning towards minimum hyperspherical energy. *Advances in neural information processing systems*, 31, 2018. 2

[50] Federico Pernici, Matteo Bruni, Claudio Baecchi, Francesco Turchini, and Alberto Del Bimbo. Class-incremental learning with pre-allocated fixed classifiers. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 6259–6266. IEEE, 2021. 2, 5

[51] Yibo Yang, Haobo Yuan, Xiangtai Li, Zhouchen Lin, Philip Torr, and Dacheng Tao. Neural collapse inspired feature-classifier alignment for few-shot class-incremental learning. In *The Eleventh International Conference on Learning Representations*, 2022. 2

[52] Da-Wei Zhou, Fu-Yun Wang, Han-Jia Ye, Liang Ma, Shiliang Pu, and De-Chuan Zhan. Forward compatible few-shot

class-incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9046–9056, 2022. 2
[53] Dustin G Mixon, Hans Parshall, and Jianzong Pi. Neural collapse with unconstrained features. *Sampling Theory, Signal Processing, and Data Analysis*, 20(2):1–13, 2022. 2, 4
[54] Cong Fang, Hangfeng He, Qi Long, and Weijie J Su. Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training. *Proceedings of the National Academy of Sciences*, 118(43):e2103091118, 2021. 2, 3, 4
[55] Florian Graf, Christoph Hofer, Marc Niethammer, and Roland Kwitt. Dissecting supervised contrastive learning. In *International Conference on Machine Learning*, pages 3821–3830. PMLR, 2021. 2
[56] Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features. *Advances in Neural Information Processing Systems*, 34:29820–29834, 2021. 2
[57] Yibo Yang, Shixiang Chen, Xiangtai Li, Liang Xie, Zhouchen Lin, and Dacheng Tao. Inducing neural collapse in imbalanced learning: Do we really need a learnable classifier at the end of deep neural network? *Advances in Neural Information Processing Systems*, 35:37991–38002, 2022. 3
[58] Vignesh Kothapalli, Ebrahim Rasromani, and Vasudev Awatramani. Neural collapse: A review on modelling principles and generalization. *arXiv preprint arXiv:2206.04041*, 2022. 3
[59] Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 991–999, 2015. 3
[60] Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John Hopcroft. Convergent learning: Do different neural networks learn the same representations? In *Feature Extraction: Modern Questions and Challenges*, pages 196–212. PMLR, 2015.
[61] Liwei Wang, Lunjia Hu, Jiayuan Gu, Zhiqiang Hu, Yue Wu, Kun He, and John Hopcroft. Towards understanding learning representations: To what extent do different neural networks learn the same representation. *Advances in neural information processing systems*, 31, 2018.
[62] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR, 2019.
[63] Yamini Bansal, Preetum Nakkiran, and Boaz Barak. Revisiting model stitching to compare neural representations. *Advances in neural information processing systems*, 34:225–236, 2021. 3
[64] Ken Chen, Yichao Wu, Haoyu Qin, Ding Liang, Xuebo Liu, and Junjie Yan. R3 adversarial network for cross model face recognition. In *CVPR*, pages 9868–9876. Computer Vision Foundation / IEEE, 2019. 3
[65] Weihua Hu, Rajas Bansal, Kaidi Cao, Nikhil Rao, Karthik Subbian, and Jure Leskovec. Learning backward compatible embeddings. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3018–3028, 2022. 3
[66] Vivek Ramanujan, Pavan Kumar Anasosalu Vasu, Ali Farhadi, Oncel Tuzel, and Hadi Pouransari. Forward compatible training for large-scale embedding retrieval systems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19386–19395, 2022. 3
[67] Weiyang Liu et al. Large-margin softmax loss for convolutional neural networks. *ICML*, 2016. 4
[68] Weiyang Liu et al. Sphereface: Deep hypersphere embedding for face recognition. *CVPR*, 2017. 4
[69] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017. 4
[70] Gabriele Prato, Simon Guiroy, Ethan Caballero, Irina Rish, and Sarath Chandar. Scaling laws for the out-of-distribution generalization of image classifiers. *ICML 2021 Workshop on Uncertainty and Robustness in Deep Learning.*, 2021.
[71] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003, 2021.
[72] Ethan Caballero, Kshitij Gupta, Irina Rish, and David Krueger. Broken neural scaling laws. In *The Eleventh International Conference on Learning Representations*, 2023. 4
[73] Bernhard Burgstaller and Friedrich Pillichshammer. The average distance between two points. *Bulletin of the Australian Mathematical Society*, 80(3):353–359, 2009. 5
[74] Herbert Solomon. *Geometric probability*. SIAM, 1978.
[75] Maurice G. (Maurice George) Kendall. *Geometrical probability*. Griffin's statistical monographs & courses ; no. 10. C. Griffin, Hafner Pub. Co, London ; New York, 1963.
[76] Luis Antonio Santaló Sors and Luis A Santaló. *Integral geometry and geometric probability*. Cambridge university press, 2004.
[77] HR Kirby and John David Murchland. *Average Distance Calculations Between and Within Zones: Some Issues at the Interface of Continous and Discrete Models*. University of Leeds, Institute for Transport Studies, 1982.
[78] Rodney Vaughan. Approximate formulas for average distances associated with zones. *Transportation science*, 18(3):231–244, 1984. 5
[79] David Fairthorne. The distances between random points in two concentric circles. *Biometrika*, 51(1/2):275–277, 1964. 5
[80] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive representation distillation. In *International Conference on Learning Representations*, 2020. 5
[81] Yibo Yang, Haobo Yuan, Xiangtai Li, Zhouchen Lin, Philip Torr, and Dacheng Tao. Neural collapse inspired feature-classifier alignment for few-shot class-incremental learning. In *ICLR*, 2023. 5
[82] Heinrich Jiang, Harikrishna Narasimhan, Dara Bahri, Andrew Cotter, and Afshin Rostamizadeh. Churn reduction via distillation. In *International Conference on Learning Representations*, 2021. 6
[83] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th*

European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16, pages 776–794. Springer, 2020. 6

[84] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, pages 499–515. Springer, 2016. 6

[85] Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. A downsampled variant of imagenet as an alternative to the cifar datasets. *arXiv preprint arXiv:1707.08819*, 2017. 6

[86] Vinay Venkatesh Ramasesh, Aitor Lewkowycz, and Ethan Dyer. Effect of scale on catastrophic forgetting in neural networks. In *International Conference on Learning Representations*, 2021. 6

[87] A. Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Technical report, Univ. Toronto, 2009. 6

[88] 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. 6

[89] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 6

[90] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, and Piotr Dollár. Designing network design spaces. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10428–10436, 2020. 6