von Mises–Fisher Loss: An Exploration of Embedding Geometries for Supervised Learning

Tyler R. Scott*
University of Colorado, Boulder
tysc7237@colorado.edu

Andrew C. Gallagher
Google Research
agallagher@google.com

Michael C. Mozer
Google Research
mcmozer@google.com

Abstract

Recent work has argued that classification losses utilizing softmax cross-entropy are superior not only for fixed-set classification tasks, but also by outperforming losses developed specifically for open-set tasks including few-shot learning and retrieval. Softmax classifiers have been studied using different embedding geometries—Euclidean, hyperbolic, and spherical—and claims have been made about the superiority of one or another, but they have not been systematically compared with careful controls. We conduct an empirical investigation of embedding geometry on softmax losses for a variety of fixed-set classification and image retrieval tasks. An interesting property observed for the spherical losses lead us to propose a probabilistic classifier based on the von Mises–Fisher distribution, and we show that it is competitive with state-of-the-art methods while producing improved out-of-the-box calibration. We provide guidance regarding the trade-offs between losses and how to choose among them.

1. Introduction

Almost on a weekly basis, novel loss functions are proposed that claim superiority over standard losses for supervised learning in vision. At a coarse level, these loss functions can be divided into *classification based* and *similarity based*. Classification-based losses [5, 7, 10, 12, 14, 16, 36, 37, 38, 44, 45, 46, 58, 60, 61, 66, 67, 68, 77] have generally been applied to *fixed-set* classification tasks (i.e., tasks in which the the set of classes in training and testing is identical). The prototypical classification-based loss uses a softmax function to map an embedding to a probability distribution over classes, which is then evaluated with cross-entropy [5]. Similarity-based losses [2, 6, 9, 15, 17, 20, 23, 24, 25, 26, 32, 34, 35, 40, 41, 47, 48, 51, 54, 55, 56, 57, 59, 62, 63, 64, 69, 70, 71, 72, 74, 76, 78] have been designed specifically for *open-set* tasks, which include retrieval and few-shot learning. Open-set tasks refer to situations in which the classes at testing are disjoint from, or sometimes a superset of, those available at training. The prototypical similarity-based method is the triplet loss which discovers embeddings such that an instance is closer to instances of the same class than to instances of different classes [51, 72].

Recent efforts to systematically compare losses support a provocative hypothesis: on open-set tasks, classification-based losses outperform similarity-based losses by leveraging embeddings in the layer immediately preceding the logits [4, 39, 58, 61, 77]. The apparent advantage of classifiers stems from the fact that similarity losses require sampling informative pairs, triplets, quadruplets, or batches of instances in order to train effectively [4, 14, 38, 66, 67, 68, 77]. However, all classification losses are not equal, and we find systematic differences among them with regard to a fundamental choice: the *embedding geometry*, which determines the similarity structure of the embedding space.

Classification losses span three embedding geometries: *Euclidean*, *hyperbolic*, and *spherical*. Although some comparisons have been made between geometries, the comparisons have not been entirely systematic and have not covered the variety of supervised tasks. We find this fact somewhat surprising given the many large-scale comparisons of loss functions. Furthermore, the comparisons that have been made appear to be contradictory. The face verification community has led the push for spherical losses, claiming superiority of spherical over Euclidean. However, this work is limited to open-set face-related tasks [14, 36, 45, 46, 66, 67, 68]. The deep metric-learning community has recently refocused its attention to classification losses, but it is unclear from empirical comparisons whether the best-performing geometry is Euclidean or spherical [4, 39, 44, 77]. Independently, Khrulkov *et al*. [25] show that a hyperbolic prototypical network is a strong performer on common few-shot learning benchmarks, and additionally a hyperbolic softmax classifier outperforms the Euclidean variant on person re-identification.

*Work performed during an internship at Google Research.

Unfortunately, these results are in contention with Tian *et al*. [61], where the authors claim a simple Euclidean softmax classifier learns embedding that are superior for few-shot learning.

One explanation for the discrepant claims are confounds that make it impossible to determine whether the causal factor for the superiority of one loss over another is embedding geometry or some other ancillary aspect of the loss. Another explanation is that each bit of research examines only a subset of losses or a subset of datasets. Also, as pointed out in Musgrave *et al*. [39], experimental setups (e.g., using the test set as a validation signal, insufficient hyperparameter tuning, varying forms of data augmentation) make it difficult to trust and reproduce published results. The goal of our work is to take a step toward rigor by reconciling differences among classification losses on both fixed-set and image-retrieval benchmarks.

As discussed in more detail in Section 2.3, our investigations led us to uncover an interesting property of spherical losses, which in turn suggested a probabilistic spherical classifier based on the von Mises–Fisher distribution. While our loss is competitive with state-of-the-art alternatives and produces improved out-of-the-box calibration, we avoid unequivocal claims about its superiority. We do, however, believe that it improves on previously proposed stochastic classifiers (e.g., [41, 52]), in, for example, its ability to scale to higher-dimensional embedding spaces.

Contributions. In our work, (1) we characterize classification losses in terms of embedding geometry, (2) we systematically compare classification losses in a well-controlled setting on a range of fixed- and open-set tasks, examining both accuracy and calibration, (3) we reach the surprising conclusion that spherical losses generally outperform the standard softmax cross-entropy loss that is used almost exclusively in practice, (4) we propose a stochastic spherical loss based on von Mises–Fisher distributions, scale it to larger tasks and representational spaces than previous stochastic losses, and show that it can obtain state-of-the-art performance with significantly lower calibration error, and (5) we discuss trade-offs between losses and factors to consider when choosing among them.

2. Classification Losses

We consider classification losses that compute the cross-entropy between a predicted class distribution and a one-hot target distribution (or equivalently, as the negative log-likelihood under the model of the target class). The geometry determines the specific mapping from a deep embedding to a class posterior, and in a classification loss, this mapping is determined by a set of parameters learned via gradient descent. We summarize the three embedding geometries that serve to differentiate classification losses.

2.1. Euclidean

Euclidean embeddings lie in an n-dimensional real-valued space (i.e., $\mathbb{R}^n$ or sometimes $\mathbb{R}^n_+$). The commonly-used dot-product softmax [5], which we refer to as STANDARD, has the form:

$$p(y|\boldsymbol{z}) = \frac{\exp(\boldsymbol{w}_y^{\mathrm{T}}\boldsymbol{z})}{\sum_j \exp(\boldsymbol{w}_j^{\mathrm{T}}\boldsymbol{z})}, \tag{1}$$

where $\boldsymbol{z}$ is an embedding and $\boldsymbol{w}_j$ are weights for class j. The dot product is a measure of similarity in Euclidean space, and is related to the Euclidean distance by $||\boldsymbol{w}_j - \boldsymbol{z}||^2 = ||\boldsymbol{w}_j||^2 + ||\boldsymbol{z}||^2 - 2\boldsymbol{w}_j^{\mathrm{T}}\boldsymbol{z}$. (Classifiers using Euclidean distance have been explored, but gradient-based training methods suffer from the curse of dimensionality because gradients go to zero when all points are far from one another. Prototypical networks [54] do succeed using a Euclidean distance posterior, but the weights are determined by averaging instance embeddings, not gradient descent.)

2.2. Hyperbolic

We follow Ganea *et al*. [16] and Khrulkov *et al*. [25] and consider the Poincaré ball model of hyperbolic geometry defined as $\mathbb{D}_c^n = \{\boldsymbol{z} \in \mathbb{R}^n : c\|\boldsymbol{z}\|^2 < 1, c \geq 0\}$ where c is a hyperparameter controlling the curvature of the ball. Embeddings thus lie inside a hypersphere of radius $1/\sqrt{c}$. To perform multi-class classification, we employ the hyperbolic softmax generalization derived in [16], hereafter HYPERBOLIC:

$$p(y|\boldsymbol{z}) \propto \exp\left(\frac{\lambda^c_{\boldsymbol{p}_y}\|\boldsymbol{a}_y\|}{\sqrt{c}}\sinh^{-1}\left(\frac{2\sqrt{c}\langle -\boldsymbol{p}_y \oplus_c \boldsymbol{z}, \boldsymbol{a}_y\rangle}{\left(1 - c\left\|-\boldsymbol{p}_y \oplus_c \boldsymbol{z}\right\|^2\right)\|\boldsymbol{a}_y\|}\right)\right), \tag{2}$$

where $\boldsymbol{p}_j \in \mathbb{D}_c^n$ and $\boldsymbol{a}_j \in T_{\boldsymbol{p}_j}\mathbb{D}_c^n \setminus \{\boldsymbol{0}\}$[1] are learnable parameters for class j, $\lambda^c_{\boldsymbol{p}_j}$ is the conformal factor of $\boldsymbol{p}_j$, $\langle .\rangle$ is the dot product, and $\oplus_c$ is the Möbius addition operator. Further details can be found in [16, 25].

2.3. Spherical

Spherical embeddings lie on the surface of an n-dimensional unit-hypersphere (i.e., $\mathbb{S}^{n-1}$). The traditional loss, hereafter COSINE, uses cosine similarity [77]:

$$p(y|\boldsymbol{z}) = \frac{\exp(\beta\cos\theta_y)}{\sum_j \exp(\beta\cos\theta_j)}, \tag{3}$$

where $\beta > 0$ is an inverse-temperature parameter, $\|\boldsymbol{z}\| = 1$, $\|\boldsymbol{w}_j\| = 1\ \forall\ j$, and θ_j is the angle between $\boldsymbol{z}$ and $\boldsymbol{w}_j$. Note

[1] $T_{\boldsymbol{x}}\mathbb{D}_c^n$ denotes the tangent space of $\mathbb{D}_c^n$ at $\boldsymbol{x}$.

that, in contrast to STANDARD, the ℓ_2-norms are factored out of the weight vectors and embeddings, thus only the *direction* determines class association.

Many variants of COSINE have been proposed, particularly in the face verification community [14, 36, 45, 46, 66, 67, 68], some of which are claimed to be superior. For completeness, we also experiment with ArcFace [14], one of the top-performing variants, hereafter ARCFACE:

$$p(y|\boldsymbol{z}) = \frac{\exp(\beta \cos(\theta_y + m))}{\exp(\beta \cos(\theta_y + m)) + \sum_{j \neq y} \exp(\beta \cos \theta_j)}, \tag{4}$$

where $m \geq 0$ is an additive-angular-margin hyperparameter penalizing the true class. (Note that we are coloring the loss name by geometry; COSINE and ARCFACE are both spherical losses.)

Early in our investigations, we noticed an interesting property of spherical losses: $\|\boldsymbol{z}\|$ encodes information about uncertainty or ambiguity. For example, the left and right frames of Figure 1 show MNIST [33] test images that, when trained with COSINE, produce embeddings that have small and large ℓ_2-norms, respectively. This result is perfectly intuitive for STANDARD since the norm affects the confidence or peakedness of the class posterior distribution (verified in [42, 46]), but for COSINE, the norm has *absolutely no effect* on the posterior. Because the norm is factored out by the cosine similarity, there is no force on the model during training to reflect the ambiguity of an instance in the norm. Despite ignoring it, the COSINE model better discriminates correct versus incorrect predictions with the norm than does the STANDARD model (see COSINE and STANDARD rows of Table 1; note that the row for VMF corresponds to a loss we introduce in the next section).

Why does the COSINE embedding convey a confidence signal in the norm? One intuition is that when an instance is ambiguous, it could be assigned many different labels in the training set, each pulling the instance's embedding in different directions. If these directions roughly cancel, the embedding will be pulled to the origin.

Due to COSINE having claimed advantages over STANDARD, and also discarding important information conveyed by $\|\boldsymbol{z}\|$, we sought to develop a variant of COSINE that uses the ℓ_2-norm to explicitly represent uncertainty in the embedding space, and thus to inform the classification decision. We refer to this variant as the *von Mises–Fisher loss* or VMF.

2.3.1 von Mises–Fisher Loss

The von Mises–Fisher (vMF) distribution is the maximum-entropy distribution on the surface of a hypersphere, parameterized by a mean unit vector, $\boldsymbol{\mu}$, and isotropic concentration, κ. The pdf for an n-dimensional unit vector $\boldsymbol{x}$ is:

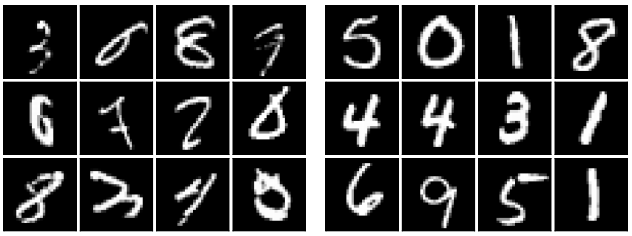

Figure 1. MNIST test images corresponding to embeddings with the (left) smallest $\|\boldsymbol{z}\|$ and (right) largest $\|\boldsymbol{z}\|$; trained with COSINE. The left grid clearly contains "noisier" or unorthodox digits.

	MNIST	Fashion MNIST	CIFAR10	CIFAR100
STANDARD	0.92	0.84	0.84	0.66
HYPERBOLIC	0.91	0.81	0.87	0.70
COSINE	0.93	0.84	**0.90**	**0.84**
ARCFACE	0.95	**0.89**	**0.90**	0.80
VMF	**0.97**	0.88	0.82	0.80

Table 1. The mean AUROC indicating how well the norm of an embedding, $\|\boldsymbol{z}\|$, discriminates correct and incorrect classifier outputs for five losses (rows) and four data sets (columns). Chance is 0.5; perfect is 1.0. Boldface indicates the highest value. Error bars are negligible across five replications. Although the embedding norm signals classifier accuracy for all losses, spherical losses yield the strongest signal.

$$p(\boldsymbol{x};\ \boldsymbol{\mu}, \kappa) = C_n(\kappa) \exp(\kappa \boldsymbol{\mu}^{\mathrm{T}} \boldsymbol{x}) \text{ with}$$
$$C_n(\kappa) = \frac{\kappa^{n/2-1}}{(2\pi)^{n/2} I_{n/2-1}(\kappa)}, \tag{5}$$

where $\boldsymbol{x}, \boldsymbol{\mu} \in \mathbb{S}^{n-1}$, $\kappa \geq 0$, and I_v denotes the modified Bessel function of the first kind at order v.

The von Mises–Fisher loss, hereafter VMF, uses the same form of the posterior as COSINE (Equation 3), although $\boldsymbol{z}$ and $\{\boldsymbol{w}_j\}$ are now vMF random variables, defined in terms of the deterministic output of the network, $\tilde{\boldsymbol{z}}$, and the learnable weight vector for each class j, $\tilde{\boldsymbol{w}}_j$:

$$\boldsymbol{z} \sim \text{vMF}\left(\boldsymbol{\mu} = \frac{\tilde{\boldsymbol{z}}}{\|\tilde{\boldsymbol{z}}\|}, \kappa = \|\tilde{\boldsymbol{z}}\|\right),$$
$$\boldsymbol{w}_j \sim \text{vMF}\left(\boldsymbol{\mu} = \frac{\tilde{\boldsymbol{w}}_j}{\|\tilde{\boldsymbol{w}}_j\|}, \kappa = \|\tilde{\boldsymbol{w}}_j\|\right). \tag{6}$$

The norm $\|.\|$ directly controls the spread of the distribution with a zero norm yielding a uniform distribution over the hypersphere's surface. The loss remains the negative log-likelihood under the target class, but in contrast to COSINE, it is necessary to marginalize over the the embedding and

weight-vector uncertainty:

$$\begin{aligned}\mathcal{L}(y, \boldsymbol{z};\ \boldsymbol{w}_{1:Y}) &= \mathbb{E}_{\boldsymbol{z},\boldsymbol{w}_{1:Y}}[-\log p(y|\boldsymbol{z}, \boldsymbol{w}_{1:Y})] \\ &= \mathbb{E}_{\boldsymbol{z},\boldsymbol{w}_{1:Y}}\left[-\log \frac{\exp(\beta \cos\theta_y))}{\sum_j \exp(\beta \cos\theta_j)}\right],\end{aligned} \tag{7}$$

where Y is the total number of classes in the training set. Applying Jensen's inequality, we obtain an upper-bound on $\mathcal{L}$ which allows us to marginalize over the $\{\boldsymbol{w}_j\}$ and obtain a form expressed in terms of an expectation over $\boldsymbol{z}$:

$$\begin{aligned}&\mathcal{L}(y, \boldsymbol{z};\ \boldsymbol{w}_{1:Y}) \leq \\ &\mathbb{E}_{\boldsymbol{z}}\left[\log\left(\sum_j \exp(\log C_n(\|\tilde{\boldsymbol{w}}_j\|)\right.\right. \\ &\left.\left. - \log C_n(\|\tilde{\boldsymbol{w}}_j + \beta\boldsymbol{z}\|))\right)\right] - \beta\mathbb{E}[\boldsymbol{w}_y]\mathbb{E}[\boldsymbol{z}],\end{aligned} \tag{8}$$

where $\mathbb{E}[\boldsymbol{z}] = (I_{n/2}(\kappa)/I_{n/2-1}(\kappa))\boldsymbol{\mu}_{\boldsymbol{z}}$. This objective can be approximated by sampling only from $\boldsymbol{z}$ and we find that during both training and testing, 10 samples is sufficient. At test time, vMF approximates $\mathbb{E}_{\boldsymbol{z},\boldsymbol{w}_{1:Y}}[p(y|\boldsymbol{z}, \boldsymbol{w}_{1:Y})]$ using Monte Carlo samples from each of $\boldsymbol{z}$ and $\{\boldsymbol{w}_j\}$.

To sample, we make use of a rejection-sampling reparameterization trick [11]. However, [11] computes modified Bessel functions on the CPU with manually-defined gradients for backpropagation, substantially slowing both the forward and backwards passes through the network. Instead, we borrow tight bounds for $I_{n/2}(\kappa)/I_{n/2-1}(\kappa)$ from [50] and $\log C_n(\kappa)$ from [31], which together make Equation 8 efficient and tractable to compute. We find that the rejection sampler is stable and efficient in embedding spaces up to at least 512D, adding little overhead to the training and testing procedures (see Appendix G). A full derivation of the loss is provided in Appendix A. Additional details regarding the bounds for $I_{n/2}(\kappa)/I_{n/2-1}(\kappa)$ and $\log C_n(\kappa)$ can be found in Appendix B.

The network fails to train when the initial $\{\tilde{\boldsymbol{w}}_j\}$ are chosen using standard initializers, particularly in higher dimensional embedding spaces. We discovered the failure to be due to near-zero gradients for the ratio of modified Bessel functions when the vector norms are small (see flat slope in Figure 2 for small κ). We derived a dimension-equivariant initialization scheme (see Appendix C) that produces norms that yield a strong enough gradient for training. We also include a fixed scale-factor on the embedding, $\tilde{\boldsymbol{z}}$, for the same reason. The points in Figure 2 show the scaling of initial parameters our scheme produces for various embedding dimensionalities, designed to ensure the ratio of modified Bessel functions has a constant value of 0.4. The expected norms are plotted as individual points with matching color

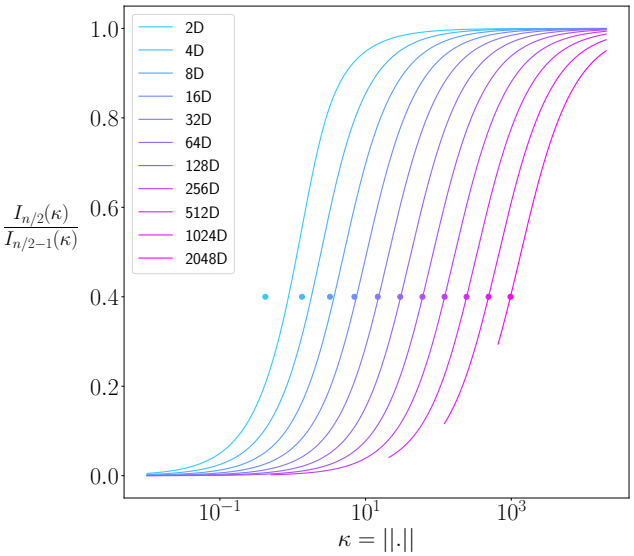

Figure 2. The ratio of modified Bessel functions versus κ for various embedding dimensionalities (colored curves). Our initializer for κ ensures the ratio of modified Bessel functions is constant regardless of the dimensionality. The value of κ provided by the initializer for each dimensionality is plotted as a single point. A perfect initializer would ensure the point sits exactly on the matching-colored curve. For this simulation, we initialized such that the y-axis had a constant value of 0.4.

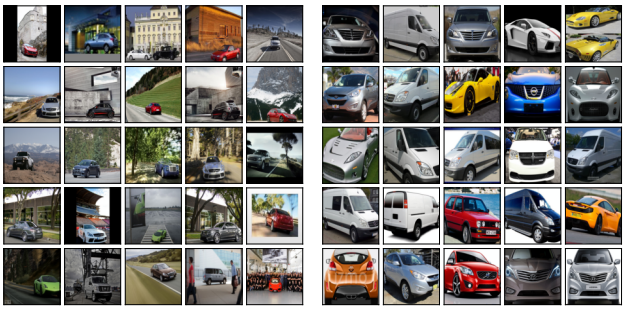

Figure 3. Cars196 test images corresponding to vMF embeddings with the (left) smallest $\kappa_{\boldsymbol{z}}$ and (right) largest $\kappa_{\boldsymbol{z}}$. Instances that are more difficult to classify or ambiguous correspond to small $\kappa_{\boldsymbol{z}}$.

and we find they produce a near-perfect fit for greater than 8D, lying on-top of their corresponding curves.

To demonstrate that vMF learns explicit uncertainty structure, we train it on Cars196 [28], a dataset where the class is determined by the make, model, and year of a photographed car. In Figure 3, we present images corresponding to embeddings whose distributions have the most uncertainty (smallest κ_z) and least uncertainty (largest κ_z) in the test set. vMF behaves quite sensibly: the most uncertain embeddings correspond to images of cars that are far from the camera or at poses where it's difficult to extract the make, model, and year; and the most certain embeddings correspond to images of cars close to the camera in a neutral pose.

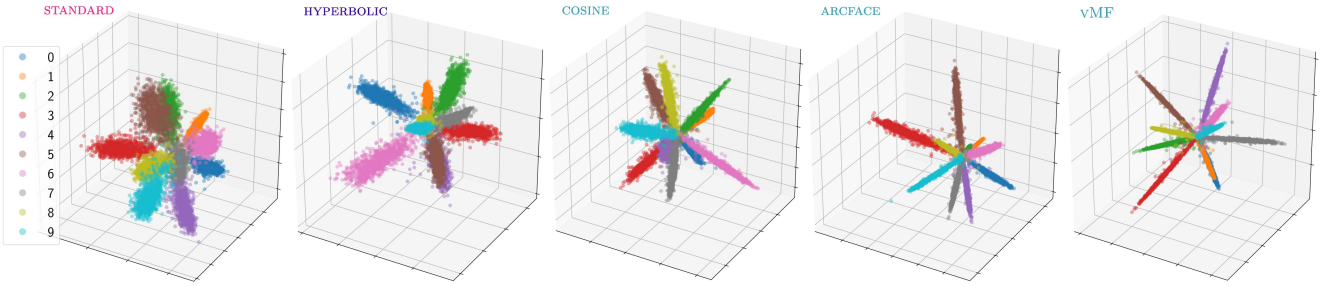

Figure 4. 3D embeddings of the MNIST test set for each of the five classification variants. Instances are colored by their ground-truth class. The plotted instances for vMF correspond to $\boldsymbol{\mu}_{\boldsymbol{z}}$. Note that HYPERBOLIC, COSINE, ARCFACE, and vMF are showing embeddings prior to the normalization/projection step. Best viewed in color.

	MNIST	Fashion MNIST	CIFAR10	CIFAR100
STANDARD	98.92 ± 0.03	90.31 ± 0.12	$\mathbf{94.13 \pm 0.05}$	69.21 ± 0.18
HYPERBOLIC	98.92 ± 0.03	90.31 ± 0.13	$\mathbf{94.11 \pm 0.05}$	69.85 ± 0.07
COSINE	98.99 ± 0.03	90.39 ± 0.09	93.99 ± 0.10	$\mathbf{70.57 \pm 0.54}$
ARCFACE	$\mathbf{99.13 \pm 0.02}$	$\mathbf{90.73 \pm 0.12}$	$\mathbf{94.15 \pm 0.05}$	69.08 ± 0.57
vMF	99.02 ± 0.04	$\mathbf{90.82 \pm 0.14}$	$\mathbf{94.00 \pm 0.12}$	$\mathbf{69.94 \pm 0.18}$

Table 2. Mean classification accuracy (%) of each loss across four fixed-set classification tasks. Error bars represent ± 1 standard-error of the mean. Boldface indicates the best-performing loss(es). Note that on average, the three spherical losses outperform HYPERBOLIC and STANDARD.

3. Experimental Results

We experiment with four fixed-set classification datasets—MNIST [33], FashionMNIST [75], CIFAR10 [29], and CIFAR100 [29]—as well as three common datasets for image retrieval—Cars196 [28], CUB200-2011 [65], and Stanford Online Products (SOP) [57]. MNIST and FashionMNIST are trained with 3D embeddings, CIFAR10 and CIFAR100 with 128D embeddings, and all open-set datasets with 512D embeddings. We perform a hyperparameter search for all losses on each dataset. The hyperparameters associated with the best performance on the validation set are then used to train five replications of the method. Reported test performance represents the average over the five replications. Additional details including the network architecture, dataset details, and hyperparameters are included in Appendix D.

3.1. Fixed-Set Classification

We begin by comparing representations learned by the five losses we described: STANDARD, HYPERBOLIC, COSINE, ARCFACE, and vMF. The latter three have spherical geometries. ARCFACE is a minor variant of COSINE claimed to be a top performer for face verification [14]. vMF is our probabilistic extension of COSINE. Using a 3D embedding on MNIST, we observe decreasing intra-class angular variance for the losses appearing from left to right in Figure 4. The intra-class variance is related to inter-class discriminability, as the 10 classes are similarly dispersed for all losses. The three losses with spherical geometry obtain the lowest variance, with ARCFACE lower than COSINE due to a margin hyperparameter designed to penalize intra-class variance; and vMF achieves the same, if not lower variance still, as a natural consequence of uncertainty reduction.

The test accuracy for each of the fixed-set classification datasets and the five losses is presented in Table 2. Across all datasets, spherical losses outperform STANDARD and HYPERBOLIC. Among the spherical losses, ARCFACE and COSINE are deficient on at least one dataset, whereas vMF is a consistently strong performer.

Table 3 presents the top-label expected calibration error (ECE) for each dataset and loss. The top-label expected calibration error approximates the disparity between a model's confidence output, $\max_y p(y|\boldsymbol{z})$, and the ground-truth likelihood of being correct [30, 49]. The left four columns show out-of-the-box ECE on the test set (i.e., prior to any post-hoc calibration). vMF has significantly reduced ECE compared to other losses, with relative error reductions of 40-70% for FashionMNIST, CIFAR10, and CIFAR100. The right four columns show ECE after applying post-hoc temperature scaling [19]. STANDARD and HYPERBOLIC greatly benefit from temperature scaling, with STANDARD exhibiting the lowest calibration error.

Post-hoc calibration requires a validation set, but many settings cannot afford the data budget to reserve a sizeable validation set, which makes out-of-the-box calibration a desirable property. For example, in few-shot and transfer learning, one may not have enough data in the target domain to both fine tune the classifier and validate.

Temperature scaling is not as effective when applied to spherical losses as when applied to STANDARD and HYPERBOLIC. The explanation, we hypothesize, is that the spher-

	ECE				ECE with Temperature Scaling			
	MNIST	Fashion MNIST	CIFAR10	CIFAR100	MNIST	Fashion MNIST	CIFAR10	CIFAR100
STANDARD	2.4 ± 0.2	12.4 ± 0.8	8.8 ± 0.1	20.6 ± 0.2	$\mathbf{0.4 \pm 0.1}$	$\mathbf{5.5 \pm 1.3}$	$\mathbf{2.7 \pm 0.2}$	$\mathbf{2.2 \pm 0.1}$
HYPERBOLIC	2.8 ± 0.1	13.2 ± 0.4	8.9 ± 0.1	22.0 ± 0.2	1.4 ± 0.1	7.2 ± 0.7	$\mathbf{2.8 \pm 0.1}$	2.6 ± 0.1
COSINE	$\mathbf{1.8 \pm 0.2}$	7.9 ± 0.5	8.9 ± 0.2	21.8 ± 1.2	1.6 ± 0.1	$\mathbf{4.2 \pm 0.1}$	6.4 ± 0.2	10.8 ± 0.8
ARCFACE	2.3 ± 0.1	11.0 ± 0.5	10.0 ± 0.1	26.4 ± 0.4	1.7 ± 0.1	7.2 ± 0.4	8.7 ± 0.1	15.7 ± 0.2
VMF	$\mathbf{1.6 \pm 0.1}$	$\mathbf{4.2 \pm 0.5}$	$\mathbf{5.9 \pm 0.2}$	$\mathbf{7.9 \pm 0.3}$	1.5 ± 0.1	5.0 ± 0.2	5.3 ± 0.2	8.0 ± 0.2

Table 3. Mean expected calibration error (%), computed with 15 equal-mass bins, before post-hoc calibration (leftmost four columns) and after temperature scaling (rightmost four columns) across the four fixed-set classification tasks. Error bars represent ± 1 standard-error of the mean. Boldface indicates the loss(es) with the lowest error.

	Cars196	CUB200-2011	SOP
STANDARD	21.3 ± 0.2	20.0 ± 0.2	39.7 ± 0.1
+ COSINE AT TEST	23.2 ± 0.1	21.4 ± 0.2	42.1 ± 0.1
HYPERBOLIC	22.9 ± 0.3	20.1 ± 0.3	41.0 ± 0.2
+ COSINE AT TEST	25.0 ± 0.4	21.8 ± 0.2	44.0 ± 0.1
COSINE	24.6 ± 0.4	$\mathbf{22.8 \pm 0.1}$	$\mathbf{44.3 \pm 0.1}$
ARCFACE	$\mathbf{27.4 \pm 0.2}$	$\mathbf{23.1 \pm 0.3}$	40.8 ± 0.3
VMF	$\mathbf{27.2 \pm 0.1}$	22.1 ± 0.1	38.3 ± 0.2

Table 4. Mean mAP@R (%) across the three open-set image retrieval tasks. Error bars represent ± 1 standard-error of the mean. Boldface indicates the best-performing loss(es). "+ Cosine at Test" replaces the default metric for the geometry (i.e., Euclidean for STANDARD and Poincaré for HYPERBOLIC) with cosine distance to compare instances.

ical losses incorporate a learned temperature parameter β (which we discuss below), which is unraveled by the calibration temperature. We leave it as an open question for how to properly post-hoc calibrate spherical losses.

3.2. Open-Set Retrieval

For open-set retrieval, we follow the data preprocessing pipeline of [4] where each dataset is first split into a train set and test set with disjoint classes. We additionally split off 15% of the training classes for a validation set, a decision that has been left out of many training procedures of similarity-based losses [39]. We evaluate methods using mean-average-precision at R (mAP@R), a metric shown to be more informative than Recall@1 [39]. For the stochastic loss, VMF, we compute $\mathbb{E}_{\boldsymbol{z}_{1:N}}[\text{mAP@R}(\boldsymbol{z}_{1:N}, y_{1:N})]$, where N is the number of test instances.

Table 4 presents the retrieval performance for each loss. As with fixed-set classification, there is no consistent winner across all datasets, but spherical losses tend to outperform STANDARD and HYPERBOLIC. Boudiaf *et al*. [4] find that retrieval performance can be improved for STANDARD by employing cosine distance at test time. Although no principled explanation is provided, we note from Figure 4 that $\|\boldsymbol{z}\|$ introduces a large source of intra-class variance in STANDARD and HYPERBOLIC. This variance is factored out automatically by the spherical losses. As shown in Table 4, cosine distance at test improves both STANDARD and HYPERBOLIC, though not to the level of the best performing spherical loss.

In contrast to other stochastic losses [41, 52], VMF scales to high-dimensional embeddings—512D in this case—and can be competitive with state-of-the-art. However, it has the worst performance on SOP which has 9,620 training classes; many more than all other datasets. In Section 2.3.1, we mentioned that to marginalize out the weight distributions, we successively apply Jensen's inequality to their expectations. The decision resulted in a tractable loss, but it is an upper-bound on the true loss and the bound very likely becomes loose as the number of training classes increases. We hypothesize the inferior performance is due to this design choice, and despite experimentation with curriculum learning techniques to condition on a subset of classes during training, results did not improve.

3.3. Role of Temperature

Due to spherical losses using cosine similarity, the logits are bounded in $[-1, 1]$. Consequently, it is necessary to scale the logits to a more suitable range. In past work, spherical losses have incorporated an inverse-temperature constant, $\beta > 0$ [14, 66, 68, 77]. Past efforts to turn β into a trainable parameter find that it either does not work as well as fixing it, or no comparison is made to a fixed value [45, 46, 67].

In COSINE, ARCFACE, and VMF, we compare a fixed β to a trained β, using fixed values common in the literature. Under a suitable initialization and parameterization of temperature, the trained β performs at least as well as a fixed value and avoids the manual search (Figure 5). In partic-

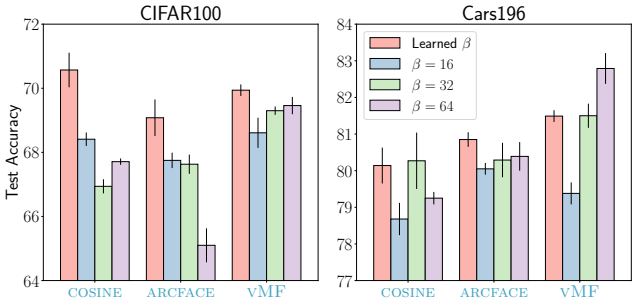

Figure 5. Comparison between a learned temperature and various values of a fixed temperature on (left) CIFAR100 and (right) Cars196. Learning the temperature performs at least as well as fixing it, with exception to vMF on Cars196.

ular, rather than performing constrained optimization in β, we perform unconstrained optimization in $\tau = \log(\beta)$. Details can be found in Appendix E.

4. Related Work

In Section 1, we dichotomized the literature in a coarse manner based on whether losses are similarity based or classification based. In this section, we focus on recent work relevant to the vMF, including losses functions using vMF distributions and stochastic classifiers.

A popular use of vMF distributions in machine learning has been clustering [3, 18]. Banerjee *et al*. [3] consider a vMF mixture model trained with expectation-maximization, and Gopal and Yang [18] propose a fully-Bayesian extension along with hierarchical and temporal versions trained with variational inference. In addition, vMF distributions have begun being applied in supervised settings. Hasnat *et al*. [21] suggest a supervised classifier for face verification where each term in the softmax is interpreted as the probability density of a vMF distribution. Zhe *et al*. [79] propose a similarity-based loss that is functionally identical to [21], except the mean parameters of the vMF distributions for each class are estimated as the maximum-likelihood estimate of the training data. Park *et al*. [43] describe the spherical analog to the prototypical network [54], but use a generator network to output the prototypes. The downside of these supervised losses compared to vMF is they assume the concentration parameter across all classes is identical and fixed, canceling $C_n(\kappa)$ in the softmax. Such a decision is mathematically convenient, but removes a significant amount of flexibility in the model. Davidson *et al*. [11] propose a variational autoencoder (VAE) with hyperspherical latent structure by using a vMF distribution. They find it can outperform a Gaussian VAE, but only in lower-dimensional latent spaces. We leverage the rejection-sampling reparameterization scheme they compose to train vMF.

Other work has sought losses that consider either stochastic embeddings or stochastic logits, but they suggest Gaussians instead of vMFs. Chang *et al*. [7] suppose stochastic embeddings, but deterministic classification weights, and train a classifier using Monte Carlo samples. They also add a KL-divergence regularizer between the embedding distribution and a zero-mean, unit-variance Gaussian. Collier *et al*. [10] propose a classification loss with stochastic logits that uses a temperature-parameterized softmax. They show it can train under the influence of heteroscedastic label noise with improved accuracy and calibration. Shi and Jain [53] convert deterministic face embeddings into Gaussians by training a post-hoc network to estimate the covariances. Their objective maximizes the mutual likelihood of same-class embeddings. Scott *et al*. [52] and Oh *et al*. [41] propose similarity-based losses that are the stochastic analogs of the prototypical network [54] and pairwise contrastive loss [20], respectively. Neither loss shows promise with high-dimensional embeddings. The loss from [52] struggles to compete with a standard prototypical network in 64D, and [41] omits any results with embeddings larger than 3D. Gaussian distributions suffer from the curse of dimensionality [11]; one possible explanation for the inferior performance compared to deterministic alternatives.

The work most closely related to ours is Kornblith *et al*. [27], which compares STANDARD, COSINE, and an assortment of alternatives including mean-squared error, sigmoid cross-entropy, and various regularizers applied to STANDARD. In contrast, we focus on multiple variants of spherical losses and comparing geometries. Their findings are compatible with ours.

5. Conclusions

In this work, we perform a systematic comparison of classification losses that span three embedding geometries—Euclidean, hyperbolic, and spherical—and attempt to reconcile the discrepancies in past work regarding their performance. Our investigations have led to a stochastic spherical classifier where embeddings and class weight vectors are von Mises–Fisher random variables. Our proposed loss is on par with other classification variants and also produces consistently reduced out-of-the-box calibration error. Our loss encodes instance ambiguity using the concentration parameter of the vMF distribution.

Consistent with the no-free-lunch theorem [73], we find there is no one loss to rule them all. Scanning arXiv, this conclusion is uncommon, and it is even rarer in published work. The performance jump claimed for novel losses often vanishes with rigorous, systematic comparsions in controlled settings—in settings where the network architecture is identical, hyperparameters are optimized with equal vigor, regularization and data augmentation is matched, and a held-out validation set is used to choose hyperparameters. Musgrave *et al*. [39] reach a similar conclusion: the gap be-

tween various similarity-based losses, as well as the spherical classification losses (COSINE and ARCFACE) was much smaller in magnitude than was claimed in the original papers proposing them. We are hopeful that future work on supervised loss functions will also prioritize rigorous experiments and step away from the compulsion to show unqualified improvements over state of the art.

Pedantics aside, we are able to glean some positive messages by systematically comparing performance across geometries and across different classification paradigms. We focus on specific recommendations in the remainder of the paper that address trade-offs among the embedding geometries.

Accuracy. Many losses are designed primarily with accuracy in mind. Across both fixed- and open-set tasks, we find that losses operating on a spherical geometry perform best. Our results support the following ranking of losses: STANDARD $\leq$ HYPERBOLIC $\leq \{$COSINE, ARCFACE, VMF$\}$. Additionally, our results corroborate two conclusions from Chen *et al*. [8]. First, smaller intra-class variance yields better generalization, and second, STANDARD focuses too much on separating classes by increasing embedding norms rather than reducing angular variance (e.g., Figure 4). Although the best of the spherical losses appears to be dataset dependent, the guidance to focus on spherical losses and perform empirical comparisons is not business as usual for practitioners, who treat STANDARD as the go-to loss. For practitioners using models with non-spherical geometries (STANDARD and HYPERBOLIC), we can still provide the guidance to use cosine distance at test time—discarding the embedding magnitude—which seems to reliably lead to improved retrieval performance.

An aspect of accuracy we do not consider, however, is the performance of downstream target tasks where a classification loss is used to pre-train weights. Kornblith *et al*. [27] discover that better class separation on the pre-training task can lead to worse transfer performance, as improved discriminability implies task specialization (i.e., throwing away inter-class variance necessary to relate classes). Spherical losses perform well on fixed-set tasks as well as on open-set tasks where the novel test classes are drawn from a distribution very similar to that of the training distribution, but transfer learning is typically a setting where STANDARD has been shown to be superior. We believe that it would be useful in future research to distinguish between near- and far-transfer, as doing so may yield distinct conclusions.

Calibration. In sensitive domains, deployed machine learning systems must produce trustworthy predictions, which requires that model confidence scores match model accuracy (i.e., they must be well calibrated). To our knowledge, we are the first to rigorously examine the effect of embedding geometry on calibration performance. Our findings indicate that when a validation set is available for temperature scaling, STANDARD consistently produces predictions with the lowest calibration error, but STANDARD generally underperforms in accuracy. Additionally, there does not exist a significant gap in out-of-the-box calibration performance across previously proposed losses from the three geometries. However, our novel VMF loss achieves superior calibration while maintaining state-of-the-art classification performance.

Future Work. We are exploring several directions for future work. First, [13] introduces a novel probability distribution—the Power Spherical distribution—with support on the surface of the hypersphere. They claim it has several key improvements over vMF distributions including a reparameterization trick that does not require rejection sampling, improved stability in high dimensions and for large values of the concentration parameter, and no dependence on Bessel functions. A stochastic classifier based on the Power Spherical distribution would likely improve the computational efficiency as well as the optimization procedure, particularly for high-dimensional embedding spaces. Second, we note that our formulation of VMF is a special case of a class of objectives based on the deep variational information bottleneck [1]: $I(Z,Y) - \gamma I(Z,X)$, where $\gamma = 0$ and the classification weights are also stochastic. Our objective thus lacks a regularizer attempting to compress the amount of input information contained in the embedding. Adding this regularization term may lead to improved performance and robustness. Third, all of the experimental datasets are approximately balanced and without label noise. Due to the stochasticity of the classification weights, VMF seems likely to benefit in supervised settings with long-tailed distributions or heteroscedastic label noise [10].

It is unfortunate that the field of supervised representation learning has become so vast that researchers tend to specialize in a particular learning paradigm (e.g., fixed-set classification, few-shot learning, transfer learning, deep metric-learning) or domain (e.g., face verification, person re-identification, image classification). As a result, losses are often pigeonholed to one paradigm or domain. The objective of this work is to lay out the space of losses in terms of embedding geometry and systematically survey losses that are not typically compared to one another. One surprising and important result in this survey is the strength of spherical losses, and the resulting dissociation of embedding norm and output confidence.

References

[1] Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep Variational Information Bottleneck. In *International Conference on Learning Representations*, 2017. 8

[2] Kelsey R. Allen, Evan Shelhamer, Hanul Shin, and Joshua B. Tenenbaum. Infinite Mixture Prototypes for Few-Shot Learning. In *International Conference on Machine Learning*, 2019. 1

[3] Arindam Banerjee, Inderjit S. Dhillon, Joydeep Ghosh, and Suvrit Sra. Clustering on the Unit Hypersphere Using von Mises–Fisher Distributions. *Journal of Machine Learning Research*, 6, 2005. 7

[4] Malik Boudiaf, Jérôme Rony, Imtiaz Masud Ziko, Eric Granger, Marco Pedersoli, Pablo Piantanida, and Ismail Ben Ayed. A Unifying Mutual Information View of Metric Learning: Cross-Entropy vs. Pairwise Losses. In *European Conference on Computer Vision*, 2020. 1, 6, 14

[5] John S. Bridle. Probabilistic Interpretation of Feedforward Classification Network Outputs, with Relationships to Statistical Pattern Recognition. In *Neurocomputing*. Springer, 1990. 1, 2

[6] Fatih Cakir, Kun He, Xide Xia, Brian Kulis, and Stan Sclaroff. Deep Metric Learning to Rank. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1

[7] Jie Chang, Zhonghao Lan, Changmao Cheng, and Yichen Wei. Data Uncertainty Learning in Face Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1, 7

[8] Beidi Chen, Weiyang Liu, Zhiding Yu, Jan Kautz, Anshumali Shrivastava, Animesh Garg, and Anima Anandkumar. Angular Visual Hardness. In *International Conference on Machine Learning*, 2020. 8

[9] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a Similarity Metric Discriminatively, with Application to Face Verification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2005. 1

[10] Mark Collier, Basil Mustafa, Efi Kokiopoulou, Rodolphe Jenatton, and Jesse Berent. A Simple Probabilistic Method for Deep Classification under Input-Dependent Label Noise. *arXiv e-prints 2003.06778 cs.LG*, 2020. 1, 7, 8

[11] Tim R. Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M. Tomczak. Hyperspherical Variational Auto-Encoders. *Conference on Uncertainty in Artificial Intelligence*, 2018. 4, 7

[12] Alexandre de Brébisson and Pascal Vincent. An Exploration of Softmax Alternatives Belonging to the Spherical Loss Family. In *International Conference on Learning Representations*, 2016. 1

[13] Nicola De Cao and Wilker Aziz. The Power Spherical Distrbution. In *International Conference on Machine Learning*, 2020. 8

[14] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1, 3, 5, 6

[15] Stanislav Fort. Gaussian Prototypical Networks for Few-Shot Learning on Omniglot. *NeurIPS Bayesian Deep Learning Workshop*, 2017. 1

[16] Octavian Ganea, Gary Becigneul, and Thomas Hofmann. Hyperbolic Neural Networks. In *Advances in Neural Information Processing Systems 31*, 2018. 1, 2

[17] Jacob Goldberger, Geoffrey E. Hinton, Sam Roweis, and Russ R. Salakhutdinov. Neighbourhood Components Analysis. In *Advances in Neural Information Processing Systems 17*, 2004. 1

[18] Siddharth Gopal and Yiming Yang. von Mises–Fisher Clustering Models. In *International Conference on Machine Learning*, 2014. 7

[19] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. In *International Conference on Machine Learning*, 2017. 5

[20] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality Reduction by Learning an Invariant Mapping. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2006. 1, 7

[21] Md. Abul Hasnat, Julien Bohné, Jonathan Milgram, Stéphane Gentric, and Liming Chen. von Mises–Fisher Mixture Model-based Deep learning: Application to Face Verification. *arXiv e-prints 1706.04264 cs.CV*, 2017. 7

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 14

[23] Junlin Hu, Jiwen Lu, and Yap Peng Tan. Discriminative Deep Metric Learning for Face Verification in the Wild. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 1

[24] Lukasz Kaiser, Ofir Nachum, Aurko Roy, and Samy Bengio. Learning to Remember Rare Events. In *International Conference on Learning Representations*, 2017. 1

[25] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic Image Embeddings. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2

[26] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese Neural Networks for One-Shot Image Recognition. *ICML Deep Learning Workshop*, 2015. 1

[27] Simon Kornblith, Honglak Lee, Ting Chen, and Mohammad Norouzi. What's in a Loss Function for Image Classification? *arXiv e-prints 2010.16402 cs.CV*, 2020. 7, 8

[28] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3D Object Representations for Fine-Grained Categorization. In *IEEE International Conference on Computer Vision Workshops*, 2013. 4, 5

[29] Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. 2009. 5

[30] Ananya Kumar, Percy Liang, and Tengyu Ma. Verified Uncertainty Calibration. In *Advances in Neural Information Processing Systems 32*, 2019. 5

[31] Sachin Kumar and Yulia Tsvetkov. von Mises–Fisher Loss for Training Sequence to Sequence Models with Continuous Outputs. In *International Conference on Learning Representations*, 2019. 4, 12

[32] Marc Law, Renjie Liao, Jake Snell, and Richard Zemel. Lorentzian Distance Learning for Hyperbolic Representations. In *International Conference on Machine Learning*, 2019. 1

[33] Yann LeCun and Corinna Cortes. MNIST Handwritten Digit Database, 2010. 3, 5

[34] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. DeepReID: Deep Filter Pairing Neural Network for Person Re-Identification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 1

[35] Jingtuo Liu, Yafeng Deng, Tao Bai, Zhengping Wei, and Chang Huang. Targeting Ultimate Accuracy: Face Recognition via Deep Embedding. *arXiv e-prints 1506.07310 cs.CV*, 2015. 1

[36] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. SphereFace: Deep Hypersphere Embedding for Face Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 3

[37] Pascal Mettes, Elise van der Pol, and Cees G. M. Snoek. Hyperspherical Prototype Networks. In *Advances in Neural Information Processing Systems 32*, 2019. 1

[38] Yair Movshovitz-Attias, Alexander Toshev, Thomas K. Leung, Sergey Ioffe, and Saurabh Singh. No Fuss Distance Metric Learning Using Proxies. In *IEEE International Conference on Computer Vision*, 2017. 1

[39] Kevin Musgrave, Serge Belongie, and Ser Nam Lim. A Metric Learning Reality Check. In *European Conference on Computer Vision*, 2020. 1, 2, 6, 7

[40] Maximilian Nickel and Douwe Kiela. Poincaré Embeddings for Learning Hierarchical Representations. In *Advances in Neural Information Processing Systems 30*, 2017. 1

[41] Seong Joon Oh, Kevin Murphy, Jiyan Pan, Joseph Roth, Florian Schroff, and Andrew Gallagher. Modeling Uncertainty with Hedged Instance Embedding. In *International Conference on Learning Representations*, 2019. 1, 2, 6, 7, 14

[42] Connor J. Parde, Carlos Castillo, Matthew Q. Hill, Y. Ivette Colon, Swami Sankaranarayanan, Jun-Cheng Chen, and Alice J. O'Toole. Deep Convolutional Neural Network Features and the Original Image. *arXiv e-prints 1611.01751 cs.CV*, 2017. 3

[43] Junyoung Park, Subin Yi, Yongseok Choi, Dong-Yeon Cho, and Jiwon Kim. Discriminative Few-Shot Learning Based on Directional Statistics. *arXiv e-prints 1906.01819 cs.LG*, 2019. 7

[44] Qi Qian, Lei Shang, Baigui Sun, Juhua Hu, Tacoma Tacoma, Hao Li, and Rong Jin. SoftTriple Loss: Deep Metric Learning Without Triplet Sampling. In *IEEE International Conference on Computer Vision*, 2019. 1

[45] Rajeev Ranjan, Ankan Bansal, Hongyu Xu, Swami Sankaranarayanan, Jun-Cheng Chen, Carlos D. Castillo, and Rama Chellappa. Crystal Loss and Quality Pooling for Unconstrained Face Verification and Recognition. *arXiv e-prints 1804.01159 cs.CV*, 2019. 1, 3, 6

[46] Rajeev Ranjan, Carlos D. Castillo, and Rama Chellappa. L2-Constrained Softmax Loss for Discriminative Face Verification. *arXiv e-prints 1703.09507 cs.CV*, 2017. 1, 3, 6

[47] Karl Ridgeway and Michael C. Mozer. Learning Deep Disentangled Embeddings with the F-Statistic Loss. In *Advances in Neural Information Processing Systems 31*, 2018. 1

[48] Oren Rippel, Manohar Paluri, Piotr Dollar, and Lubomir Bourdev. Metric Learning with Adaptive Density Discrimination. In *International Conference on Learning Representations*, 2016. 1

[49] Rebecca Roelofs, Nicholas Cain, Jonathon Shlens, and Michael C. Mozer. Mitigating Bias in Calibration Error Estimation. *arXiv e-prints 2012.08668 cs.LG*, 2021. 5

[50] Diego Ruiz-Antolín and Javier Segura. A New Type of Sharp Bounds for Ratios of Modified Bessel Functions. *Journal of Mathematical Analysis and Applications*, 443(2):1232 – 1246, 2016. 4, 12

[51] Florian Schroff, Dmitry Kalenichenko, and James Philbin. FaceNet: A Unified Embedding for Face Recognition and Clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015. 1

[52] Tyler R. Scott, Karl Ridgeway, and Michael C. Mozer. Stochastic Prototype Embeddings. *arXiv e-prints 1909.11702 stat.ML*, 2019. 2, 6, 7

[53] Yichun Shi and Anil K. Jain. Probabilistic Face Embeddings. In *IEEE International Conference on Computer Vision*, 2019. 7

[54] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical Networks for Few-Shot Learning. In *Advances in Neural Information Processing Systems 30*, 2017. 1, 2, 7

[55] Kihyuk Sohn. Improved Deep Metric Learning with Multi-Class N-Pair Loss Objective. In *Advances in Neural Information Processing Systems 29*, 2016. 1

[56] Hyun Oh Song, Stefanie Jegelka, Vivek Rathod, and Kevin Murphy. Deep Metric Learning via Facility Location. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1

[57] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep Metric Learning via Lifted Structured Feature Embedding. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1, 5

[58] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle Loss: A Unified Perspective of Pair Similarity Optimization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1

[59] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. Learning to Compare: Relation Network for Few-Shot Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1

[60] Eu Wern Teh, Terrance DeVries, and Graham W. Taylor. ProxyNCA++: Revisiting and Revitalizing Proxy Neighborhood Component Analysis. In *European Conference on Computer Vision*, 2020. 1

[61] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B. Tenenbaum, and Phillip Isola. Rethinking Few-Shot Image Classification: A Good Embedding is All You Need? *arXiv e-prints 2003.11539 cs.CV*, 2020. 1, 2

[62] Eleni Triantafillou, Richard Zemel, and Raquel Urtasun. Few-Shot Learning Through an Information Retrieval Lens. In *Advances in Neural Information Processing Systems 30*, 2017. 1

[63] Evgeniya Ustinova and Victor Lempitsky. Learning Deep Embeddings with Histogram Loss. In *Advances in Neural Information Processing Systems 29*, 2016. 1

[64] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching Networks for

One Shot Learning. In *Advances in Neural Information Processing Systems 29*, 2016. 1

[65] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, California Institute of Technology, 2011. 5

[66] Feng Wang, Jian Cheng, Weiyang Liu, and Haijun Liu. Additive Margin Softmax for Face Verification. *IEEE Signal Processing Letters*, 2018. 1, 3, 6

[67] Feng Wang, Xiang Xiang, Jian Cheng, and Alan L. Yuille. NormFace: L2 Hypersphere Embedding for Face Verification. In *ACM Multimedia Conference*, 2017. 1, 3, 6

[68] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. CosFace: Large Margin Cosine Loss for Deep Face Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 3, 6

[69] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep Metric Learning with Angular Loss. In *IEEE International Conference on Computer Vision*, 2017. 1

[70] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R. Scott. Multi-Similarity Loss with General Pair Weighting for Deep Metric Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1

[71] Xinshao Wang, Yang Hua, Elyor Kodirov, Guosheng Hu, Romain Garnier, and Neil M. Robertson. Ranked List Loss for Deep Metric Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1

[72] Kilian Q. Weinberger and Lawrence K Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. *Journal of Machine Learning Research*, 10, 2009. 1

[73] David H. Wolpert and William G. Macready. No Free Lunch Theorems for Optimization. *IEEE Transactions on Evolutionary Computation*, 1997. 7

[74] Chao-Yuan Wu, R. Manmatha, Alexander J. Smola, and Philipp Krähenbühl. Sampling Matters in Deep Embedding Learning. In *IEEE International Conference on Computer Vision*, 2017. 1

[75] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms. *arXiv e-prints 1708.07747 cs.LG*, 2017. 5

[76] Tongtong Yuan, Weihong Deng, Jian Tang, Yinan Tang, and Binghui Chen. Signal-to-Noise Ratio: A Robust Distance Metric for Deep Metric Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1

[77] Andrew Zhai and Hao Yu Wu. Classification is a Strong Baseline for Deep Metric Learning. In *British Machine Vision Conference*, 2019. 1, 2, 6

[78] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a Deep Embedding Model for Zero-Shot Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1

[79] Xuefei Zhe, Shifeng Chen, and Hong Yan. Directional Statistics-Based Deep Metric Learning for Image Classification and Retrieval. *arXiv e-prints 1802.09662 cs.CV*, 2018. 7