On Separability of Self-Supervised Representations

Vikash Sehwag [1] **Mung Chiang** [2] **Prateek Mittal** [1]

Abstract

Increasingly, self-supervised techniques are achieving competitive accuracy compared to supervised representations, on multiple downstream image classification tasks. While accuracy is a good predictor of performance, it fails to provide deeper insights into *how well separable self-supervised representations are*? In this paper, we investigate the separability of self-supervised representations under two settings and compare them with representations learned form supervised training. First, we calculate the margin of each data point from the learned classifier. Since calculating margin is intractable for non-linear classifiers, we further leverage adversarial perturbations to measure separability in our second framework. In both experiments, we observe that self-supervised representations are generally less separable than supervised representations, even in cases when the former achieves higher accuracy than the latter. We validate our results across five state-of-the-art self-supervised training techniques and seven different datasets. Finally, we propose a new performance metric, named *effective-accuracy*, which encapsulates both accuracy and separability of learned representations.

1. Introduction

Self-supervised training of neural networks aims to learn high-quality feature representations in the absence of data labels (Doersch et al., 2015; Larsson et al., 2016; Gidaris et al., 2018; Kolesnikov et al., 2019). In the recent years, self-supervised training methods have improved significantly where they achieve competitive performance with supervised training for deep neural networks (Tian et al., 2020; He et al., 2020; Chen et al., 2020b;a; Misra & Maaten, 2020; Tian et al., 2019; Donahue & Simonyan, 2019; Oord et al., 2018). A common approach to measure the quality of self-supervised representations is to measure accuracy on downstream tasks. For example, for self-supervised representations learned with ImageNet dataset, a common approach is to measure classification accuracy using linear classifiers for downstream datasets like ImageNet (Deng et al., 2009), CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009). While accuracy is predictive of the quality of self-supervised representation, it fails to provide a deeper understanding of the difference between self-supervised and supervised representations. In this work, we ask, *how good are self-supervised representations compared to representations learned with supervised training?*

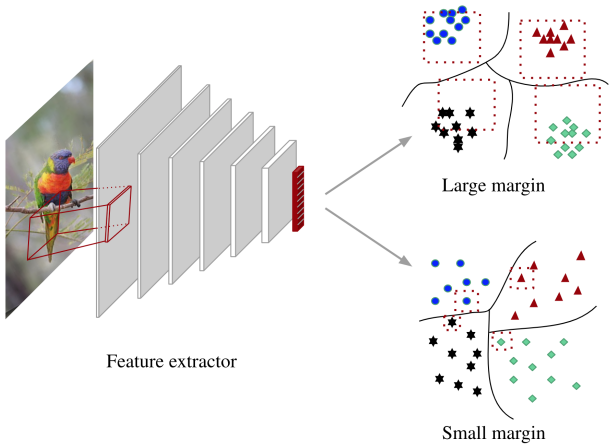


Figure 1. Self-supervised learning aims to train a feature extractor such that the extracted features can achieve high accuracy for the downstream task/dataset . However, even with similar accuracy, learned features can differ in their separability, i.e., margin from the classifier boundaries. We observe that even when self-supervised features achieve competitive accuracy, they tend to achieve poor separability compared to supervised features.

We aim to answer this question by delving deeper into the *separability* of learned representations. Earlier works employ dimensionality reduction techniques like t-sne (Maaten & Hinton, 2008) to visualize the quality of learned representations. While helpful, such visualization techniques are both lossy and have inherent stochasticity (Wattenberg et al., 2016).

In contrast, we aim to capture the separability of representation in the *feature space itself*. In particular, we first train a

[1]Princeton Universty, USA [2]Purdue University, USA. Correspondence to: Vikash Sehwag <vvikash@princeton.edu>.

Presented at the ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning.

classifier on the learned self-supervised/supervised features and then measure the margin, i.e., minimum distance, of each data-point from this classifier. We provide a visualization of this procedure in Figure 1. Note that better separability (margin) is highly desirable to additionally improve generalization, robustness to label noise, and robustness to adversarial perturbations (Elsayed et al., 2018).

Our experimental results demonstrate the following intriguing observation: while self-supervised representations can achieve competitive accuracy, they achieve *poor* separability compared to supervised representations in most cases. We validate this observation across 7 different datasets and 5 different self-supervised training techniques.

However, calculating margin is intractable for non-linear classifiers (Elsayed et al., 2018). To circumvent this challenge, we leverage techniques from adversarial attacks literature. In particular, we argue that with increasing margin comes higher robustness to adversarial perturbations (visualization in Figure 1). We benchmark the accuracy of each self-supervised technique across multiple perturbation budgets and make a similar observation as observed with margin. Finally, we propose an *effective accuracy* metric which not only captures the accuracy but also the separability of learned representations.

Key Contributions. We focus on the separability of both supervised and self-supervised representations. We first calculate margins and provide a rigorous benchmark of separability of representations from five state-of-the-art self-supervised training techniques. Next, we propose a formulation based on adversarial perturbation to measure separability and propose a new performance metric, named *effective-accuracy* which encapsulates both accuracy and separability of learned representations. We experiment with 7 different datasets to validate our findings.

2. Methodology and experimental setup

Given a feature extractor $\phi : \mathbb{R}^{w\times h\times 3} \rightarrow \mathbb{R}^n$, which is trained either with supervised or self-supervised methods, we first extract features for each dataset. We operate in this features space, where we train a classifier ($f : \mathbb{R}^n \rightarrow \mathbb{R}^c$) on these extracted features. The c-dimensional output of the classifier are logits where the class with the maximum logits value is the predicted class for an input feature.

Calculating margin. Given the classifier f and a data-point x in the features space, our objective is to calculate its smallest distance to the classifier. However, an analytic formulation of margin is only tractable for a linear classifier. We calculate this distance in each of the l_1, l_2, and l_∞ space using following formulation for a linear classifier $f(x) = wx + b$ (Elsayed et al., 2018).

$$m_p(x) = \min_{i\in[1,c], i\neq t} \frac{f(x)_t - f(x)_i}{||w_t - w_i||_q}$$

where t is predicted class, i.e., class with highest logit value. Note the denominator requires dual-norm (q) of the norm (p) in which we aim to measure the distance. Thus for distance in l_1, l_2, and l_∞, we use l_∞, l_2, and l_1 norm in the denominator respectively. In addition, we only calculate the margin for correctly classified examples.

However, calculating the margin of a non-linear classifier is not tractable. Here we leverage the techniques developed in adversarial attacks literature where we argue that a classifier with a better margin will also be robust to larger adversarial perturbations. Thus for each classifier, we evaluate the accuracy across multiple perturbation budgets and analyze how fast it decays with increasing perturbation budget. Accuracy for representations with poor separability will degrade quickly with increasing perturbation budget.

Effective-accuracy. With effective accuracy we aim to also embed the information on separability along with the achieved accuracy. We first measure the normalized area-under-the-curve (norm-AUC), i.e., AUC divided by maximum accuracy and maximum perturbation budget. It represents the stability of the classifier under adversarial perturbations. We calculate effective-accuracy by multiplying norm-AUC by maximum accuracy. In summary, we use

$$\textit{effective-accuracy} = \textit{accuracy} \times \textit{normalized-area-under-the-curve}.$$

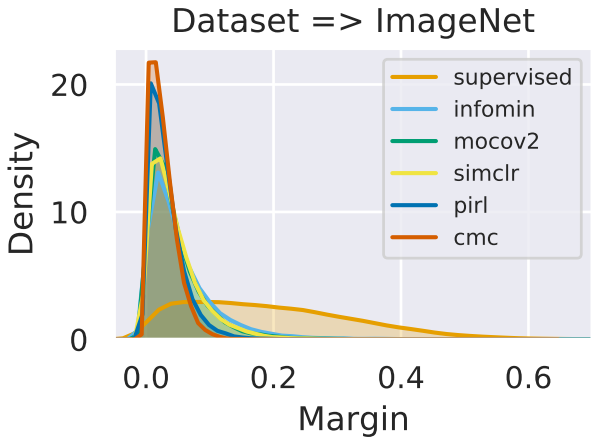
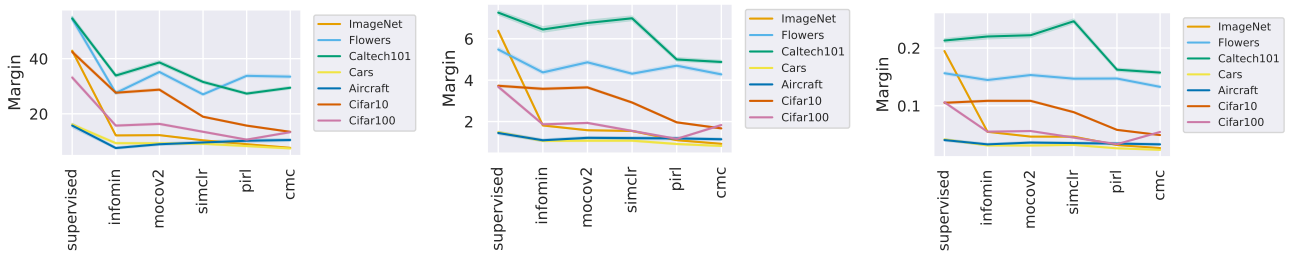


Figure 2. Margins in l_∞-norms for ImageNet dataset. Even though self-supervised methods like InfoMin, MoCo, and SimCLR achieve large improvement in accuracy, they still achieve much poorer margins compared to supervised training.

2.1. Setup

Features standardization. Unlike the input space, the dynamic range of features differs for each choice of dataset



(a) l_1 (b) l_2 (c) l_∞

Figure 3. Margins along L_p-norms foo multiple datasets and training methods. Self-supervised training methods are ordered by their accuracy on the ImageNet dataset.

or feature extractors. It makes it difficult to compare margins or adversarial robustness of features across datasets or feature extractors. To alleviate this issue, we standardize the features to have zero mean and unit norm. This choice also simplified the hyper-parameter search as we no longer require very high learning rates (He et al., 2020; Chen et al., 2020b). However, we observe a small decrease in performance, though consistently, across most features extractors.

Hyper-parameter search. We find that the classification performance over the features depends significantly on the choice of hyper-parameters, in particular on learning rate and weight decay. To achieve the best results, we perform a search over learning rate in {0.01, 0.05, 0.1, 0.5, 1.0, 5.0, 10.0} and weight decay in {1e-5, 1e-4, 1e-3, 1e-2} using the released or a custom validation set and use the best parameters in the final training setup.

Choice of self-supervised training methods. We experiment with five state-of-the-art self-supervised training methods, namely, InfoMin (Tian et al., 2020), MoCo (Chen et al., 2020b; He et al., 2020), SimCLR (Chen et al., 2020a), PIRL (Misra & Maaten, 2020), and CMC (Tian et al., 2019). We standardize the choice of network architecture, where we use a ResNet-50 network for each of them. We use the best performing publicly available checkpoints for respective methods. We provide more details in Appendix B.

Choice of datasets. We standardize the choice of the dataset where each of the self-supervised methods trains a ResNet-50 network on ImageNet dataset. However, to evaluate the quality of learned representations, we work with seven different datasets, namely ImageNet, CIFAR-10 (Krizhevsky et al., 2009), CIFAR-100 (Krizhevsky et al., 2009), Oxford Flowers (Nilsback & Zisserman, 2008), Caltech101 (Fei-Fei et al., 2004), FGVC Aircrafts (Maji et al., 2013), and Stanford cars (Krause et al., 2013).

Ordering of self-supervised methods for comparison. When visualizing results, we order the self-supervised methods along their performance on ImageNet dataset under linear classification (ssl, 2020 (accessed June 14, 2020), i.e., the order is InfoMin, Moco, SimCLR, PIRL, and CMC.

3. Experimental results

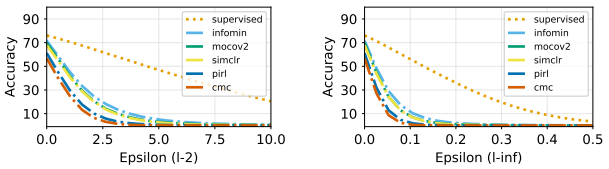
In this section, we first calculate margins for each feature extractor and dataset combination and analyze the separability of their representations. Next, we demonstrate that an analysis based on adversarial perturbations can also be used to analyze separability. Finally, we present our results on effective-accuracy which encapsulates both accuracy and separability.

3.1. Calculating margin

We first extract the features for each dataset from each feature extractor and then train a linear classifier on them. Figure 2 presents these results when we calculate the margin of each correctly classified data point from the classifier.

Note that the feature extractors, both supervised and self-supervised, are trained on ImageNet. Thus we start our analysis with ImageNet where we plot the distribution of margin achieved for each choice of feature extractor. We use l_∞ distance in this plot and plot the data with l_1 and l_2 distance in Appendix C.

Our key observations are the following. Note that our results should be viewed in conjunction with their accuracy on the ImageNet dataset. Supervised features achieve a 76.1% top-1 accuracy while self-supervised techniques improve



(a) l_2-norm based perturbation (b) l_∞-norm based perturbation

Figure 4. Adversarial accuracy for ImageNet with features extracted from different training methods. As highlighted with margins, supervised representations are more separable thus achieve higher robustness under adversarial perturbations. We provide results for all dataset in appendix C.

Table 1. Effective-accuracy (benign-accuracy) of different features extractors and datasets. Effective-accuracy encapsulates both benign-accuracy and separability. Even when from self-supervised methods achieve higher benign accuracy, they achieve poor effective-accuracy compared to features from supervised networks. This indicates the poor separability of representations learned by even state-of-the-art self-supervised training techniques.

(a) Using l_2-norm based perturbations

	Supervised	InfoMin	Moco	SimCLR	PIRL	CMC
ImageNet	**62.2 (76.1)**	27.6 (71.6)	24.4 (69.7)	23.6 (67.9)	16.1 (61.1)	12.9 (56.7)
Flowers	**76.2 (91.1)**	67.7 (88.5)	73.1 (89.8)	67.6 (86.9)	68.2 (85.6)	655 (85.2)
Caltech101	**80.7 (91.5)**	71.6 (88.4)	74.7 (90.0)	76.8 (90.0)	56.7 (78.5)	60.5 (81.1)
Aircrafts	**16.0** (48.4)	12.3 (46.3)	14.7 (**50.6**)	1.0 (1.0)	11.3 (40.0)	12.2 (43.5)
CIFAR10	**64.7** (91.5)	60.4 (92.4)	63.7 (**93.1**)	54.8 (90.6)	37.2 (85.5)	31.4 (82.6)
CIFAR100	**50.1** (74.2)	29.9 (75.1)	31.5 (**75.8**)	0.9 (0.9)	17.4 (64.4)	25.9 (59.6)
Cars	**17.4 (49.8)**	10.9 (43.8)	11.2 (43.8)	11.2 (43.2)	7.2 (32.0)	6.2 (30.2)

(b) Using l_∞-norm based perturbations

	Supervised	InfoMin	Moco	SimCLR	PIRL	CMC
ImageNet	**68.5 (76.1)**	40.5 (71.6)	36.6 (69.7)	36.0 (67.9)	25.8 (61.1)	21.7 (56.7)
Flowers	**83.1 (91.1)**	79.0 (88.5)	81.9 (89.8)	78.6 (86.9)	76.8 (85.6)	75.2 (85.2)
Caltech101	**85.7 (91.5)**	81.3 (88.4)	82.8 (90.0)	84.6 (90.0)	67.4 (78.5)	70.8 (81.1)
Aircrafts	**23.4** (48.4)	19.7 (46.3)	23.1 (**50.6**)	1.0 (1.0)	17.3 (40.0)	19.2 (43.5)
CIFAR10	76.8 (91.5)	75.3 (92.4)	**77.5** (**93.1**)	71.5 (90.6)	54.8 (85.5)	48.5 (82.6)
CIFAR100	**59.5** (74.2)	43.3 (75.1)	45.2 (**75.8**)	0.9 (0.9)	27.7 (64.4)	36.9 (59.6)
Cars	**25.8 (49.8)**	17.6 (43.8)	18.1 (43.8)	18.4 (43.2)	11.8 (32.0)	10.5 (30.2)

the top-1 accuracy from 56.8% to 71.8% with CMC and InfoMin, respectively.

Self-supervised representations are less separable than supervised ones. As fig. 2 clearly highlights, the margin values for the supervised features are much higher than any other self-supervised ones. In particular, supervised features have an average margin of 0.19 while self-supervised methods have an average margin of maximum 0.05 (achieved for InfoMin). Note that though they have a large difference in the margin, the accuracy of InfoMin and supervised features has a difference of only 4.3%.

Relatively small improvement in separability for self-supervised techniques. While self-supervised techniques, from CMC to InfoMin have improved the accuracy significantly, the average margin has only been improved from 0.03 to 0.05 (in comparison to an average margin of 0.19 for supervised features). Note that compared to the improvements in accuracy, which is quite close to accuracy with supervised features, the improvements in margin have been relatively small.

We present more detailed results with different datasets in Figure 3, where we plot the average margin for each feature extractor. Note that we still use the feature extractor trained on the ImageNet dataset. The broad trend across different dataset also support our aforementioned observations. For most datasets, we observe a sharp decrease in margin from supervised to self-supervised features. Note that we are using exactly the same network architecture (ResNet50) for each method.

3.2. Encapsulating accuracy and separability in a single metric: effective accuracy

In this section, we present our results on using adversarial perturbations a tool to measure the separability of learned representations. In the feature space, we first train a classifier and then measure its accuracy when the features are perturbed adversarially. For consistent comparison with our earlier results on margins, we use a linear classifier in this experiment. However, we observe a similar trend when using a multi-layer classifier.

Figure 4 shows these results for ImageNet dataset for both l_∞ and l_2-norm based perturbations. It supports our hypothesis that with increasing separability, the adversarial robustness of the classifier increases. As the supervised representations achieve better margins, they also show more stability to adversarial perturbations. We observe the same effect for both l_∞ and l_2 based perturbations.

Next, we aggregate the data from the accuracy-perturbation plots into a single metric, named *effective accuracy*. We take a conservative approach where we use a maximum perturbation of 5.0 and 0.1 for l_2 and l_∞-norms, respectively. For SimCLR features, we find the data standardization leads to a failure to converge for Aircrafts and CIFAR100 dataset. We present these results in Table 1. Along with supporting our observations with margins, effective accuracy also highlights the following intriguing phenomenon.

Supervised learning can achieve better separability even when it under-performs. For datasets like Aircrafts, CIFAR10, CIFAR100 we observe that supervised learning achieves higher effective accuracy, thus better separability while achieving less accuracy than self-supervised representations.

4. Discussion

In this work, we take a deeper look into the separability of both supervised and self-supervised representations. We first measure separability using margins and later use adversarial perturbations as an alternative when calculating margins becomes intractable. Our results highlight the intriguing observation that even when self-supervised representation achieves competitive accuracy with supervised representations, it achieves poor separability. In future work, we aim to improve self-supervised training techniques to achieve better effective-accuracy.

References

Self-Supervised Image Classification on ImageNet, 2020 (accessed June 14, 2020). URL https://paperswithcode.com/sota/self-supervised-image-classification-on.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020a.

Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020b.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Doersch, C., Gupta, A., and Efros, A. A. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1422–1430, 2015.

Donahue, J. and Simonyan, K. Large scale adversarial representation learning. In *Advances in Neural Information Processing Systems*, pp. 10541–10551, 2019.

Elsayed, G., Krishnan, D., Mobahi, H., Regan, K., and Bengio, S. Large margin deep networks for classification. In *Advances in neural information processing systems*, pp. 842–852, 2018.

Fei-Fei, L., Fergus, R., and Perona, P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pp. 178–178. IEEE, 2004.

Gidaris, S., Singh, P., and Komodakis, N. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Kolesnikov, A., Zhai, X., and Beyer, L. Revisiting self-supervised visual representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 1920–1929, 2019.

Kornblith, S., Shlens, J., and Le, Q. V. Do better imagenet models transfer better? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2661–2671, 2019.

Krause, J., Deng, J., Stark, M., and Fei-Fei, L. Collecting a large-scale dataset of fine-grained cars. 2013.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Larsson, G., Maire, M., and Shakhnarovich, G. Learning representations for automatic colorization. In *European Conference on Computer Vision*, pp. 577–593. Springer, 2016.

Maaten, L. v. d. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov): 2579–2605, 2008.

Maji, S., Rahtu, E., Kannala, J., Blaschko, M., and Vedaldi, A. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

Misra, I. and Maaten, L. v. d. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6707–6717, 2020.

Nilsback, M.-E. and Zisserman, A. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pp. 722–729. IEEE, 2008.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Tian, Y., Krishnan, D., and Isola, P. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.

Wattenberg, M., Viégas, F., and Johnson, I. How to use t-sne effectively. *Distill*, 1(10):e2, 2016.

A. Details on datasets

We consider a suit of image classification datasets in this work. We provide details on them in Table 2. Similar to previous works (Kornblith et al., 2019), we use mean per-class accuracy for Flowers, Caltech101, and Aircraft datasets.

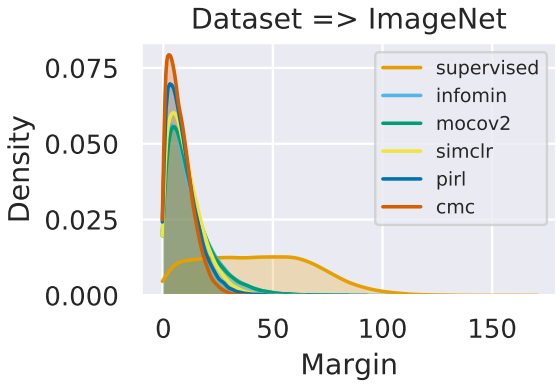
Table 2. Details on datasets used in this work.

Dataset	Images (train/val)	Classes
ImageNet	1.2M/50,000	1000
Flowers	2,040/6,149	102
Caltech101	3,060/6,084	102
Aircrafts	6,667/3,333	100
CIFAR10	50,000/10,000	10
CIFAR100	50,000/10,000	1000
Cars	8,144/8,041	196

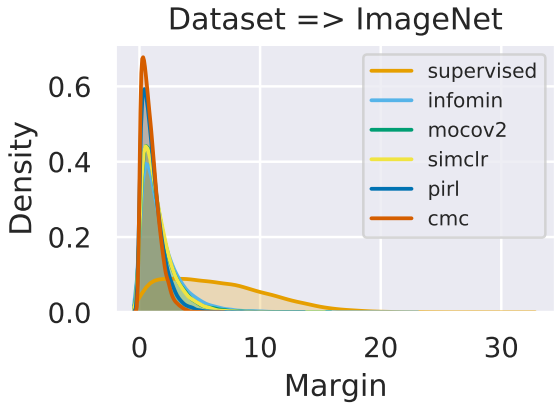
B. Details on Self-supervised methods

We work with the five state-of-the-art self supervised training techniques, namely InfoMin, MoCo, SimCLR, PIRL, and CMC. Each of these methods uses a contrastive loss in their training mechanism while difference in the choice of transformations (SimCLR, PIRL), use of memory bank and moment encoder (MoCo), using mutliple color-spaces (CMC, InfoMin). We refer the reader to Tian et al. 2020 for more details on each of these methods.

C. Additional experimental results

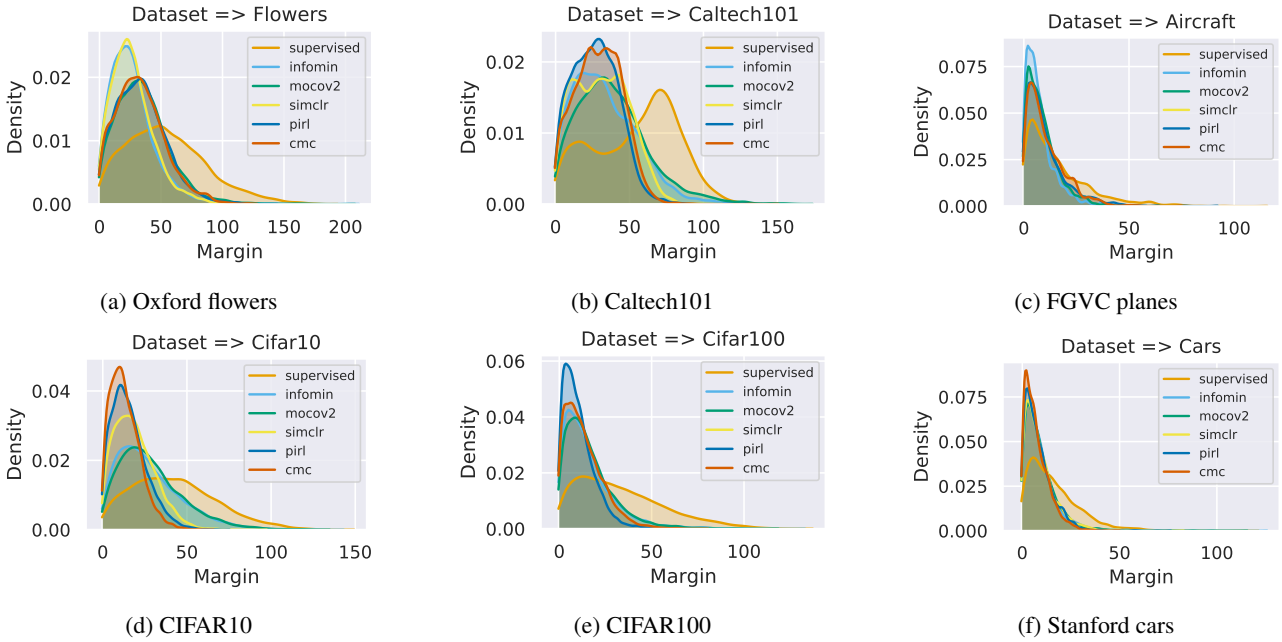


(a) l_1



(b) l_2

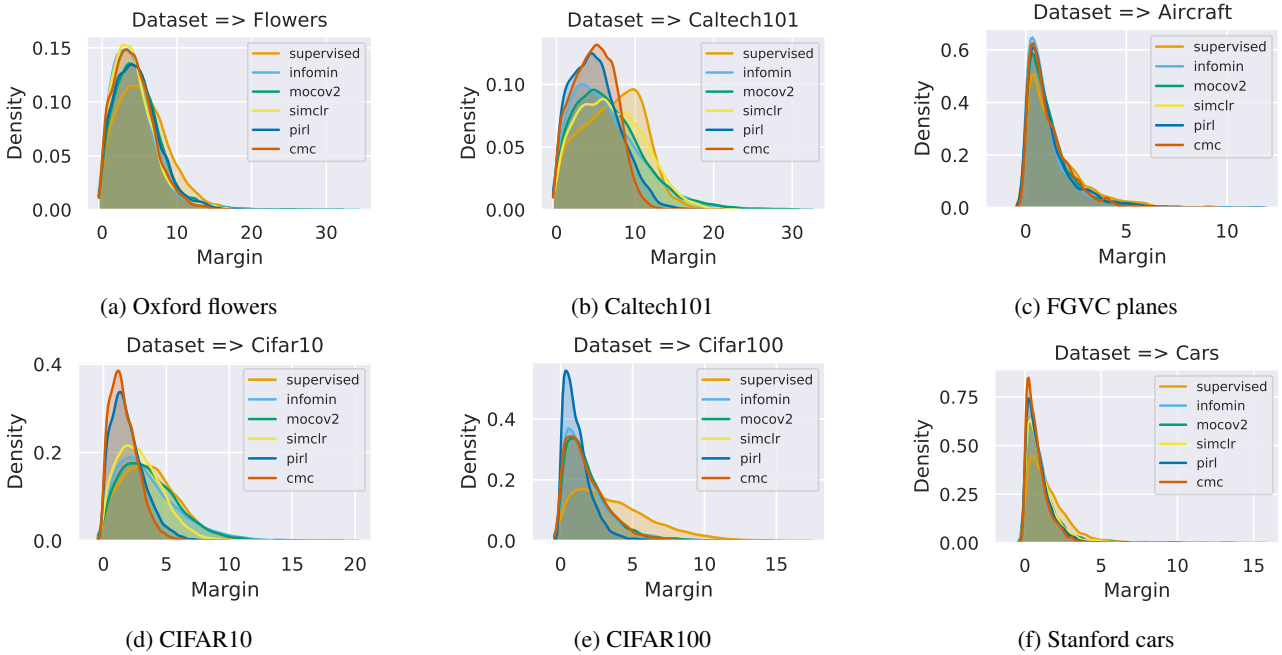
Figure 5. Margin for imagenet dataset in the l_1 and l_2 space.



(a) Oxford flowers (b) Caltech101 (c) FGVC planes

(d) CIFAR10 (e) CIFAR100 (f) Stanford cars

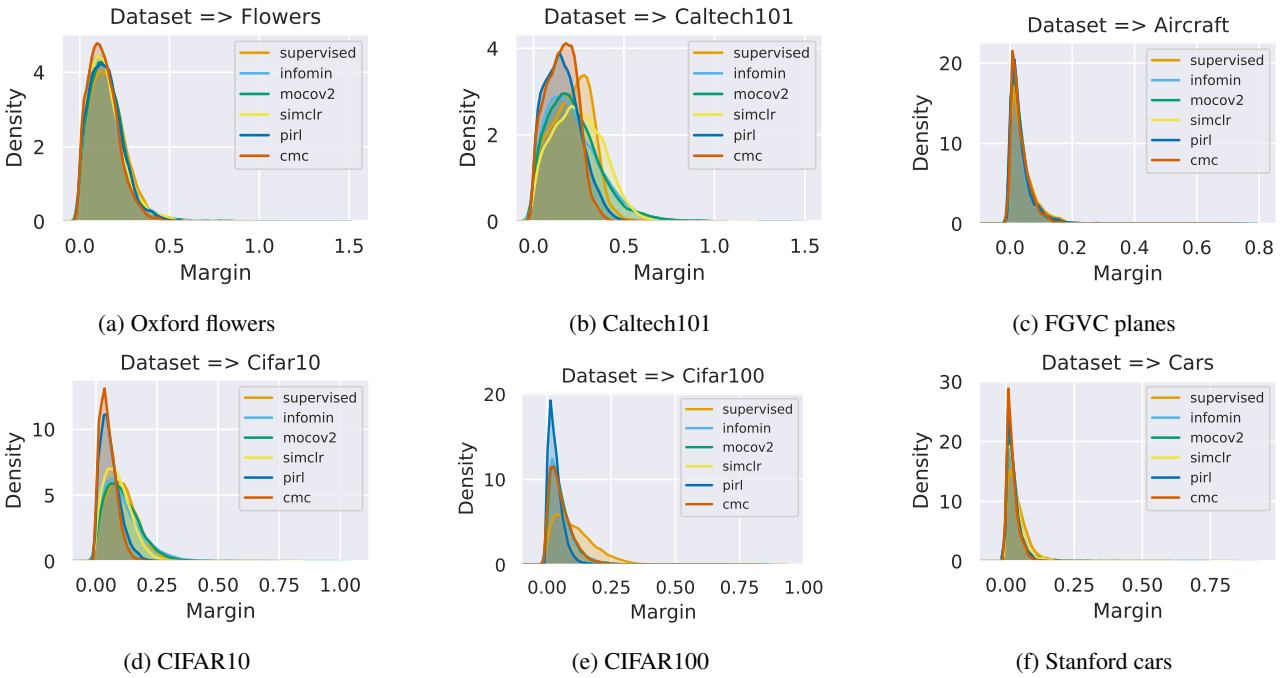
Figure 6. Distribution of margins for all dataset in the l_1 space.



(a) Oxford flowers (b) Caltech101 (c) FGVC planes

(d) CIFAR10 (e) CIFAR100 (f) Stanford cars

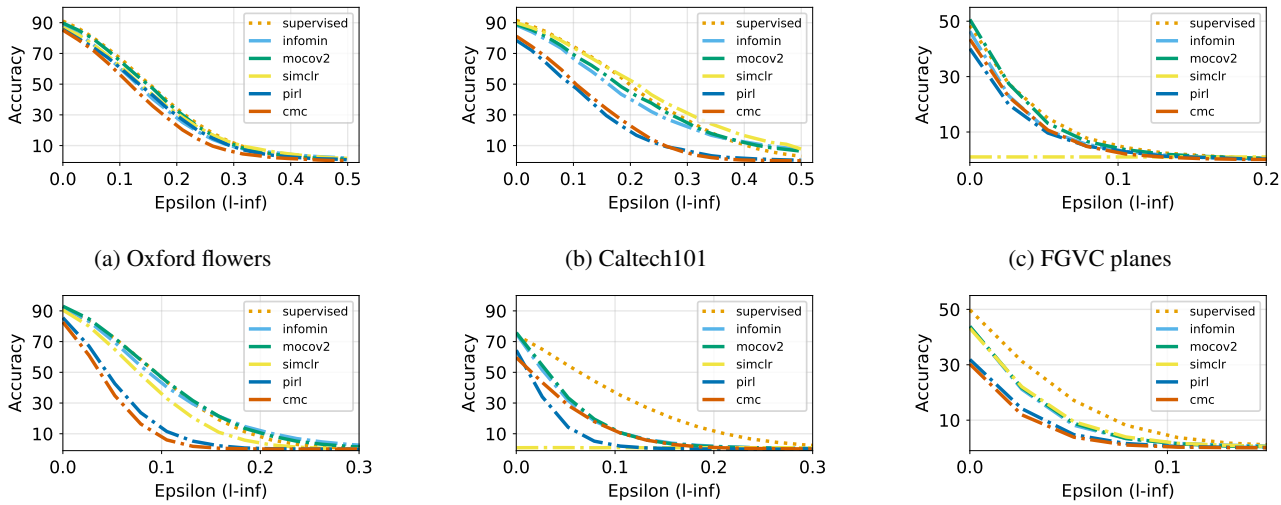
Figure 7. Distribution of margins for all dataset in the l_2 space.



(a) Oxford flowers (b) Caltech101 (c) FGVC planes

(d) CIFAR10 (e) CIFAR100 (f) Stanford cars

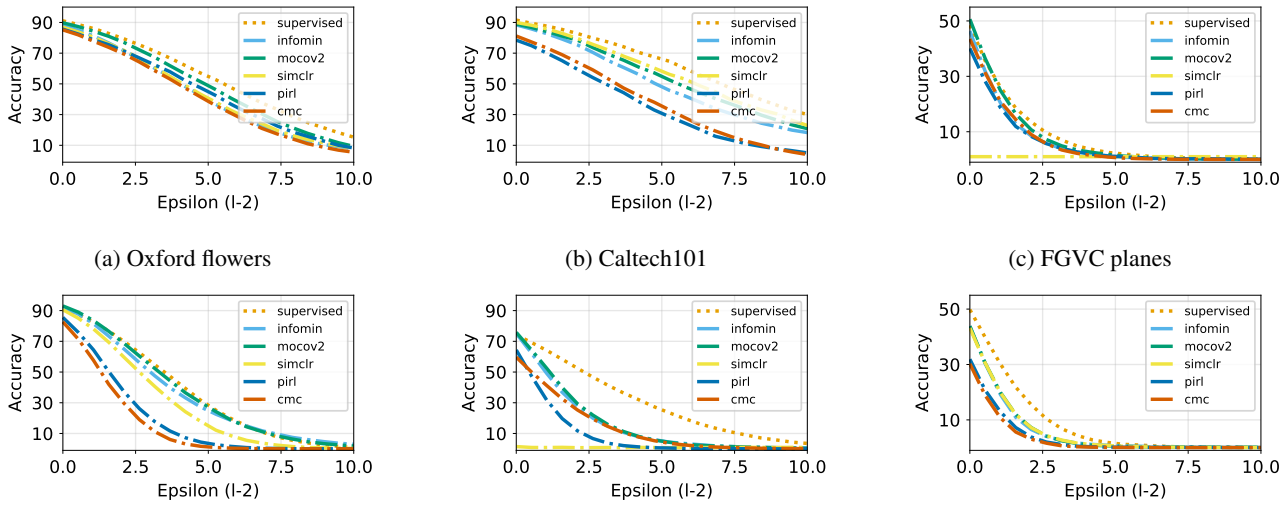
Figure 8. Distribution of margins for all dataset in the l_∞ space.



(a) Oxford flowers (b) Caltech101 (c) FGVC planes

(d) CIFAR10 (e) CIFAR100 (f) Stanford cars

Figure 9. Accuracy under l_∞-norm based adversarial perturbations for different datasets.



(a) Oxford flowers (b) Caltech101 (c) FGVC planes

(d) CIFAR10 (e) CIFAR100 (f) Stanford cars

Figure 10. Accuracy under l_2-norm based adversarial perturbations for different datasets.