SELF-SUPERVISED ADVERSARIAL ROBUSTNESS FOR THE LOW-LABEL, HIGH-DATA REGIME

Sven Gowal*, Po-Sen Huang*, Aaron van den Oord, Timothy Mann & Pushmeet Kohli
DeepMind
London, United Kingdom
{sgowal,posenhuang}@google.com

ABSTRACT

Recent work discovered that training models to be invariant to adversarial perturbations requires substantially larger datasets than those required for standard classification. Perhaps more surprisingly, these larger datasets can be "mostly" unlabeled. *Pseudo-labeling*, a technique simultaneously pioneered by four separate and simultaneous works in 2019, has been proposed as a competitive alternative to labeled data for training adversarially robust models. However, when the amount of labeled data decreases, the performance of *pseudo-labeling* catastrophically drops, thus questioning the theoretical insights put forward by Uesato et al. (2019), which suggest that the sample complexity for learning an adversarially robust model from unlabeled data should match the fully supervised case. We introduce Bootstrap Your Own Robust Latents (BYORL), a self-supervised learning technique based on BYOL for training adversarially robust models. Our method enables us to train robust representations without any labels (reconciling practice with theory). Most notably, this robust representation can be leveraged by a linear classifier to train adversarially robust models, even when the linear classifier is not trained adversarially. We evaluate BYORL and *pseudo-labeling* on CIFAR-10 and IMAGENET and demonstrate that BYORL achieves significantly higher robustness in the low-label regime (i.e., models resulting from BYORL are up to two times more accurate). Experiments on CIFAR-10 against ℓ_2 and ℓ_∞ norm-bounded perturbations demonstrate that BYORL achieves near state-of-the-art robustness with as little as 500 labeled examples. We also note that against ℓ_2 norm-bounded perturbations of size $\epsilon = 128/255$, BYORL surpasses the known state-of-the-art with an accuracy under attack of 77.61% (against 72.91% for the prior art).

1 INTRODUCTION

As neural networks tackle challenges ranging from ranking content on the web (Covington et al., 2016) to autonomous driving (Bojarski et al., 2016) via medical diagnostics (De Fauw et al., 2018), it has becomes increasingly important to ensure that deployed models are robust and generalize to various input perturbations. Unfortunately, despite their success, neural networks are not intrinsically robust. In particular, the addition of small but carefully chosen deviations to the input, called adversarial perturbations, can cause the neural network to make incorrect predictions with high confidence (Carlini & Wagner, 2017a; Goodfellow et al., 2014; Kurakin et al., 2016; Szegedy et al., 2013). Starting with Szegedy et al. (2013), there has been a lot of work on understanding and generating adversarial perturbations (Carlini & Wagner, 2017b; Athalye & Sutskever, 2017), and on building models that are robust to such perturbations (Papernot et al., 2015; Madry et al., 2017; Kannan et al., 2018). Robust optimization techniques, like the one developed by Madry et al. (2017), learn robust models by trying to find the worst-case adversarial examples (by using gradient ascent on the training loss) at each training step and adding them to the training data.

Since Madry et al. (2017), various modifications to their original implementation have been proposed (Zhang et al., 2019; Pang et al., 2020; Huang et al., 2020; Qin et al., 2019). We highlight the simultaneous work from Carmon et al. (2019); Uesato et al. (2019); Zhai et al. (2019a); Najafi et al. (2019) that pioneered the use of additional unlabeled data using *pseudo-labeling*. While, theoretically,

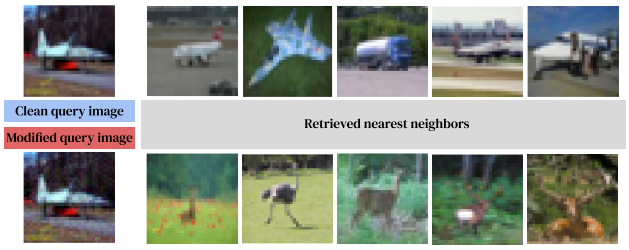


Figure 1: Dangers of using non-robust representation learning. We use a non-robust self-supervised learning technique to learn image representations (i.e., BYOL; Grill et al., 2020). The right-hand side shows CIFAR-10 images closest (in representation space using cosine similarity) to the query image on the left. The top row demonstrates that, when given an unmodified image of an airplane, the nearest matches resemble that query image either visually or semantically. The bottom row demonstrates that a seemingly identical image can be used to retrieve images of animals which are both visually and semantically far from the query image.

robustness can be achieved with only limited amount of labeled data, in practice, it remains difficult to train models that are both robust and accurate in the low-label regime. [1]

Finally, we note that there has been little work towards learning adversarially robust representations that allow for efficient training on multiple downstream tasks (with the exception of Cemgil et al., 2019; Kim et al., 2020). Learning good image representations is a key challenge in computer vision (Wiskott & Sejnowski, 2002; Hinton et al., 2006), and many different approaches have been proposed. Among them state-of-the-art methods include contrastive methods (Chen et al., 2020b; Oord et al., 2018; He et al., 2020) and latent bootstrapping (Grill et al., 2020). However, none of these recent works consider the impact of adversarial manipulations, which can render the widespread use of general representations difficult. As an example, Fig. 1 demonstrates the effect that a non-robust representation has on a content retrieval task, where two seemingly identical query images are matched to widely different images (i.e., their nearest neighbors in representation space).

In this paper, we tackle the issue of learning robust representations that are adversarially robust on multiple downstream tasks in the low-label regime. Our contributions are as follows:

- We formulate Bootstrap Your Own Robust Latents (BYORL), a modification of Bootstrap Your Own Latents (BYOL) (Grill et al., 2020) that enables the training of robust representations without the need for any label information. These representations allow for efficient training on multiple downstream tasks with a fraction of the original labels.
- Most notably, even with only 1% of the labels, BYORL comes close to or even exceeds previous state-of-the-art which uses all labels. For example, for ℓ_2 norm-bounded perturbations of size $\epsilon = 128/255$ on CIFAR-10, BYORL achieves 75.50% robust accuracy compared to 72.91% for the previous state-of-the-art using all labels. BYORL reaches 77.61% robust accuracy when using all available labels (and additional unlabeled data extracted from 80M-TINYIMAGES; Torralba et al., 2008).
- Finally, we show that the representations learned through BYORL transfer much better to downstream tasks (i.e., downscaled STL-10 (Coates et al., 2011) and CIFAR-100 (Krizhevsky et al., 2014)) than those obtained through *pseudo-labeling* and standard adversarial training. Importantly, we also highlight that classifiers trained on top of these robust representations do not need to be trained adversarially to be robust.

2 RELATED WORK

Adversarial robustness. Biggio et al. (2013) and Szegedy et al. (2013) observed that neural networks, while they achieve high accuracy on test data, are vulnerable to carefully crafted inputs perturbations, called adversarial examples. Since then, there has been several work on building stronger adversarial examples as well as defense against such adversarial examples (Carlini & Wagner, 2017b; Athalye & Sutskever, 2017; Goodfellow et al., 2014; Papernot et al., 2015; Madry

[1] In Uesato et al. (2018) and Carmon et al. (2019), robust accuracy drops by 10% when limiting the number of labels to about 10%.

et al., 2017; Kannan et al., 2018). Arguably, the most successful approach for learning adversarially robust models is adversarial training as proposed by Madry et al. (Athalye et al., 2018; Uesato et al., 2018). This classic version of adversarial training has been augmented in different ways – with changes in the attack procedure (e.g., by incorporating momentum; Dong et al., 2017), loss function (e.g., logit pairing; Mosbach et al., 2018) or model architecture (e.g., using attention; Zoran et al., 2020). We also highlight Zhang et al. (2019), who proposed TRADES which balances the trade-off between standard and robust accuracy. By construction, to the contrary of our proposed method, all aforementioned approaches use label information and are not capable of learning generic representations that might be useful to multiple downstream tasks.

Semi- and self-supervised learning. Since human annotations can be expensive, semi- and self-supervised learning approaches that leverage both labeled and unlabeled data have been proposed to improve model performance (Chapelle et al., 2009; Bachman et al., 2014; Berthelot et al., 2019; Laine & Aila, 2017; Miyato et al., 2018; Sajjadi et al., 2016; Xie et al., 2019). A common approach is to train networks to solve a manually-predefined pretext task (e.g., predicting the relative location of image patches) for representation learning, and later use the learned representation for a specific supervised learning task (Dosovitskiy et al., 2014; Doersch et al., 2015; Noroozi & Favaro, 2016). Recently, contrastive learning that uses different views of multiple augmented images has been an effective tool to learn rich representation from unsupervised data (Oord et al., 2018; Chen et al., 2020b; He et al., 2020; Tian et al., 2020), as these methods achieve comparable performance to fully-supervised models. While these works focus on improving standard generalization, we leverage representation learning, as proposed by Grill et al. (2020), to improve adversarial generalization.

Semi- and self-supervised learning for adversarial robustness. Schmidt et al. (2018) showed that learning adversarially robust models requires more data. As such, adversarial robustness with unlabeled data has recently drawn a lot of attention. We highlight the works by Uesato et al. (2019); Carmon et al. (2019); Zhai et al. (2019a) which leveraged labeled data to train a standard classifier that is in turn used to *pseudo-label* the remaining unlabeled data. However, as shown by Uesato et al. (2019); Carmon et al. (2019); Zhai et al. (2019a), when only 10% of the CIFAR-10 labels are available the robust accuracy drops significantly. In this paper, we focus on improving adversarial robustness in the low-label regime by leveraging unlabeled data (e.g., when 1%–10% of labels are available) to build robust representations. The result is a technique that significantly outperforms state-of-the-art *pseudo-labeling* techniques in the low-label regime and remains competitive with adversarial training when all labels are available. Chen et al. (2020a) also study adversarial pre-training on self-supervised tasks and demonstrate that they can train robust representations. However, to the contrary of our approach, their method does not preserve the robustness of their resulting representations on downstream tasks (unless robust fine-tuning is used). Hendrycks et al. (2019) combine supervised adversarial training with an additional self-supervised head. They demonstrate that they can improve on standard adversarial training, but do not learn general representations. We also highlight the recent work by Kim et al. (2020) which combines adversarial training with contrastive learning. Our method reaches comparable robust accuracy, but is more scalable (as it is not based on contrastive learning which requires large batch sizes). To the contrary of Kim et al., we also study the transferability of robust representations and focus on the low-label regime.

3 METHOD

In this section, we explain BYORL which elegantly combines adversarial training with BYOL. Hence, we start by giving a brief description of adversarial training and BYOL.

3.1 ADVERSARIAL TRAINING

Madry et al. (2017) formulate a saddle point problem whose goal is to find model parameters $\boldsymbol{\theta}$ that minimize the adversarial risk:

$$\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\max_{\boldsymbol{\delta}\in\mathbb{S}} l(f(\boldsymbol{x}+\boldsymbol{\delta};\boldsymbol{\theta}),y)\right] \tag{1}$$

where $\mathcal{D}$ is a data distribution over pairs of examples $\boldsymbol{x}$ and corresponding labels y, $f(\cdot;\boldsymbol{\theta})$ is a model parametrized by $\boldsymbol{\theta}$, l is a suitable loss function (such as the $0-1$ loss in the context of classification tasks), and $\mathbb{S}$ defines the set of allowed perturbations (i.e., the adversarial input set or threat model).

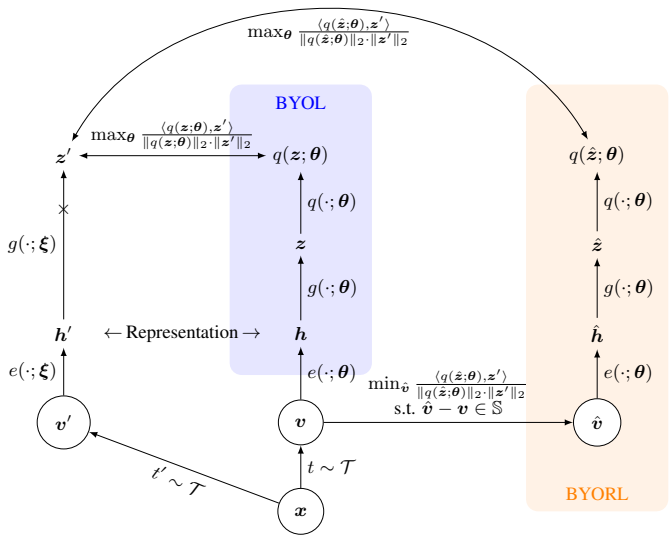


Figure 2: Flow diagram that highlights the difference between BYOL and BYORL. Whereas BYOL directly tries to maximize the cosine similarity between $q(\boldsymbol{z};\boldsymbol{\theta})$ and $\boldsymbol{z}'$, BYORL first executes an adversarial attack to retrieve an alternative image $\hat{\boldsymbol{v}}$.

Several methods (also known as "attacks") have been proposed to find adversarial examples (and effectively solve the inner maximization problem in Eq. 1). Classical adversarial training as proposed by Madry et al. (2017) uses Projected Gradient Descent (PGD),[2] which replaces the impractical $0-1$ loss l with the cross-entropy loss $\hat{l}$ and computes an adversarial perturbation $\hat{\boldsymbol{\delta}} = \boldsymbol{\delta}^{(K)}$ in K gradient ascent steps of size α as

$$\boldsymbol{\delta}^{(k+1)} \leftarrow \text{proj}_{\mathbb{S}}\left(\boldsymbol{\delta}^{(k)} + \alpha \nabla_{\boldsymbol{\delta}^{(t)}} \hat{l}(f(\boldsymbol{x} + \boldsymbol{\delta}^{(k)};\boldsymbol{\theta}), y)\right) \tag{2}$$

where $\boldsymbol{\delta}^{(0)}$ is chosen at random within $\mathbb{S}$, and where $\text{proj}_{\mathbb{A}}(\boldsymbol{a})$ projects a point $\boldsymbol{a}$ back onto a set $\mathbb{A}$. Finally, for each example $\boldsymbol{x}$ with label y, adversarial training minimizes the loss given by

$$\mathcal{L}_{\boldsymbol{\theta}}^{\text{AT}} = \hat{l}(f(\boldsymbol{x} + \hat{\boldsymbol{\delta}};\boldsymbol{\theta}), y) \approx \max_{\boldsymbol{\delta} \in \mathbb{S}} \hat{l}(f(\boldsymbol{x} + \boldsymbol{\delta};\boldsymbol{\theta}), y) \tag{3}$$

where $\hat{\boldsymbol{\delta}}$ is given by Eq. 2 and $\hat{l}$ is the softmax cross-entropy loss.

3.2 Bootstrap your own latents

Many successful self-supervised learning approaches learn image representations by identifying whether different views belong to the same image (Dosovitskiy et al., 2014; Wu et al., 2018). Whereas contrastive methods formulate this prediction problem into one of discrimination (i.e., from the representation of an augmented view, they learn to discriminate between the representation of another augmented view of the same image, and the representations of augmented views of other images), BYOL relies on two neural networks: an online and a target network, that interact and learn from each other. The goal of the online network is to predict the target network representation of the same image under different augmented views, where the target network itself is defined by an exponential moving average of the online network parameters. We selected BYOL as the basis of our proposed method not only because it is currently the most successful representation learning technique, but also because it is more amenable to adversarial training, to contrary of contrastive methods which require large batch sizes (Chen et al., 2020b; Oord et al., 2018) or memory banks He et al. (2020).

As shown in Fig. 2, the online network is composed of three stages: an encoder $e(\cdot;\boldsymbol{\theta})$, a projector $g(\cdot;\boldsymbol{\theta})$ and a predictor $q(\cdot;\boldsymbol{\theta})$. Omitting the predictor, the target network has the same architecture

[2]There exists a few variants of PGD which normalize the gradient step differently (e.g., using its sign or ℓ_2-norm depending on the threat model).

as the online network, but uses a different set of weights $\boldsymbol{\xi}$. As explained by Grill et al. (2020), in order to enhance representations while preventing their collapse, the target network's weights are allowed to change slowly throughout training. More precisely, given a decay rate $\tau \in [0, 1]$, after each training step, the parameters $\boldsymbol{\xi}$ are updated as $\boldsymbol{\xi} \leftarrow \tau\boldsymbol{\xi} + (1-\tau)\boldsymbol{\theta}$. Given an image $\boldsymbol{x}$, and two augmentations $t, t' \sim \mathcal{T}$ sampled from a set of augmentations (e.g., random crops or recolorizations), BYOL produces two augmented views $\boldsymbol{v} = t(\boldsymbol{x})$ and $\boldsymbol{v}' = t'(\boldsymbol{x})$. The first view passes through the online network, producing a representation $\boldsymbol{h} = e(\boldsymbol{x}; \boldsymbol{\theta})$ and a projection $\boldsymbol{z} = g(\boldsymbol{h}; \boldsymbol{\theta})$. The second view similarly passes through the target network, producing a target projection $\boldsymbol{z}' = g \circ e(\boldsymbol{v}'; \boldsymbol{\xi})$. Finally, given an online prediction $q(\boldsymbol{z}; \boldsymbol{\theta})$ (which should be predictive of the target projection), BYOL minimizes the loss

$$\mathcal{L}_{\boldsymbol{\theta}}^{\text{BYOL}} = \left\| \frac{q(\boldsymbol{z}; \boldsymbol{\theta})}{\|q(\boldsymbol{z}; \boldsymbol{\theta})\|_2} - \frac{\boldsymbol{z}'}{\|\boldsymbol{z}'\|_2} \right\|_2^2 = 2 - 2 \cdot \frac{\langle q(\boldsymbol{z}; \boldsymbol{\theta}), \boldsymbol{z}' \rangle}{\|q(\boldsymbol{z}; \boldsymbol{\theta})\|_2 \cdot \|\boldsymbol{z}'\|_2}. \tag{4}$$

At the end of training, everything but e and $\boldsymbol{\theta}$ is discarded and only the representation $e(\boldsymbol{x}; \boldsymbol{\theta})$ of an image $\boldsymbol{x}$ is used by downstream applications.

3.3 Bootstrap your own robust latents

We now introduce an effective and elegant approach to learn adversarially robust representations. Whereas previous (semi-)supervised techniques that are directly based on adversarial training require the presence of labels to produce adversarially robust neural networks (from which robust representations can be extracted at intermediate layers), our method can operate without any labels. As shown in the experimental section, the resulting representations can then be used to train linear classifiers (on top of these representation) that are intrinsically robust to adversaries. Our method, named Bootstrap Your Own Robust Latents or BYORL, consists of combining BYOL with adversarial training. A diagram summarizing BYORL is visible in Fig. 2.

For conciseness, we will denote by $\gamma = g \circ e$ the composition of the encoder and projector and by $\kappa = q \circ g \circ e$ the composition of the encoder, projector and predictor. Similarly to BYOL, BYORL starts by generating two views $\boldsymbol{v} = t(\boldsymbol{x})$ and $\boldsymbol{v}' = t'(\boldsymbol{x})$ of the same image $\boldsymbol{x}$. While the second view goes through the target network unmodified to produce a target projection $\boldsymbol{z}' = \gamma(\boldsymbol{v}'; \boldsymbol{\xi})$, the first view is further augmented via an adversarial attack. The goal of the adversarial attack is to maximize the disagreement between the online and target networks while respecting the threat model described by $\mathbb{S}$ (see subsection 3.1). To this end, we would like to find an optimal perturbation $\boldsymbol{\delta}^\star \in \mathbb{S}$ that minimizes the resulting cosine similarity between the online prediction $\kappa(\boldsymbol{v} + \boldsymbol{\delta}; \boldsymbol{\theta})$ and target projection $\boldsymbol{z}'$:

$$\boldsymbol{\delta}^\star = \arg\min_{\boldsymbol{\delta} \in \mathbb{S}} \frac{\langle \kappa(\boldsymbol{v} + \boldsymbol{\delta}; \boldsymbol{\theta}), \boldsymbol{z}' \rangle}{\|\kappa(\boldsymbol{v} + \boldsymbol{\delta}; \boldsymbol{\theta})\|_2 \cdot \|\boldsymbol{z}'\|_2}. \tag{5}$$

Like adversarial training, we can leverage PGD to approximate $\boldsymbol{\delta}^\star$ by $\hat{\boldsymbol{\delta}}$. Taking K steps of size α, resulting in $\hat{\boldsymbol{\delta}} = \boldsymbol{\delta}^{(K)}$, we have

$$\boldsymbol{\delta}^{(k+1)} \leftarrow \text{proj}_{\mathbb{S}} \left(\boldsymbol{\delta}^{(k)} + \alpha \nabla_{\boldsymbol{\delta}^{(t)}} \frac{\langle \kappa(\boldsymbol{v} + \boldsymbol{\delta}^{(t)}; \boldsymbol{\theta}), \boldsymbol{z}' \rangle}{\|\kappa(\boldsymbol{v} + \boldsymbol{\delta}^{(t)}; \boldsymbol{\theta})\|_2 \cdot \|\boldsymbol{z}'\|_2} \right) \tag{6}$$

where $\boldsymbol{\delta}^{(0)}$ is chosen at random within $\mathbb{S}$. Finally, we seek to maximize the agreement between the adversarially modified online prediction $\kappa(\boldsymbol{v} + \hat{\boldsymbol{\delta}}; \boldsymbol{\theta})$ and the target projection $\boldsymbol{z}'$ by updating the online weights $\boldsymbol{\theta}$ as to minimize the following loss:

$$\mathcal{L}_{\boldsymbol{\theta}}^{\text{BYORL}}(\boldsymbol{v}, \boldsymbol{v}') = 2 - 2 \cdot \frac{\langle \kappa(\boldsymbol{v} + \hat{\boldsymbol{\delta}}; \boldsymbol{\theta}), \gamma(\boldsymbol{v}'; \boldsymbol{\xi}) \rangle}{\|\kappa(\boldsymbol{v} + \hat{\boldsymbol{\delta}}; \boldsymbol{\theta})\|_2 \cdot \|\gamma(\boldsymbol{v}'; \boldsymbol{\xi})\|_2} \approx 2 - 2 \cdot \min_{\boldsymbol{\delta} \in \mathbb{S}} \frac{\langle \kappa(\boldsymbol{v} + \boldsymbol{\delta}; \boldsymbol{\theta}), \boldsymbol{z}' \rangle}{\|\kappa(\boldsymbol{v} + \boldsymbol{\delta}; \boldsymbol{\theta})\|_2 \cdot \|\boldsymbol{z}'\|_2}. \tag{7}$$

Here are a few additional considerations. First, we symmetrize the loss $\mathcal{L}_{\boldsymbol{\theta}}^{\text{BYORL}}$ in Eq. 7 by feeding $\boldsymbol{v}'$ to the online network and $\boldsymbol{v}$ to the target network. The adversarial attack is executed on $\boldsymbol{v}'$ instead of $\boldsymbol{v}$ and tries to minimize the cosine similarity between the online prediction $\kappa(\boldsymbol{v}' + \boldsymbol{\delta}; \boldsymbol{\theta})$ and target projection $\gamma(\boldsymbol{v}; \boldsymbol{\xi})$: $\mathcal{L}_{\boldsymbol{\theta}}^{\text{symmetric}}(\boldsymbol{v}, \boldsymbol{v}') = \mathcal{L}_{\boldsymbol{\theta}}^{\text{BYORL}}(\boldsymbol{v}, \boldsymbol{v}') + \mathcal{L}_{\boldsymbol{\theta}}^{\text{BYORL}}(\boldsymbol{v}', \boldsymbol{v})$. Second, one can observe that the adversarial attack is always performed through the online network. We could similarly perform the attack through the target network, but we found that training was less stable as batch

statistics (needed by batch normalization) were not representative of statistics induced by adversarial examples (as the online network would receive clean rather than adversarial images). Third, instead of the proposed method, we could imagine making two passes through the online network (for both the clean and adversarial images) and maximizing the agreement between both online predictions (in addition to maximizing the agreement with the target projection). We found that this increased the risk of representation collapse as this adds an incentive for the online network to output constant predictions (i.e., collapsed representations are the perfect defense against adversarial attacks).

4 EXPERIMENTS

We assess the performance of BYORL across multiple axes. We evaluate the robustness of the resulting representations by training robust linear classifiers on top of these representations. First, we compare to the performance of various classifiers (comparing BYORL with *pseudo-labeling*). Second, we study how these robust representations transfer to unseen new tasks. Finally, we also evaluate whether robustness transfers to downstream tasks – even when the final task is not treated as being adversarial.

4.1 SETUP AND IMPLEMENTATION DETAILS

We highlight here the most important components and defer some of the details to Appendix A.

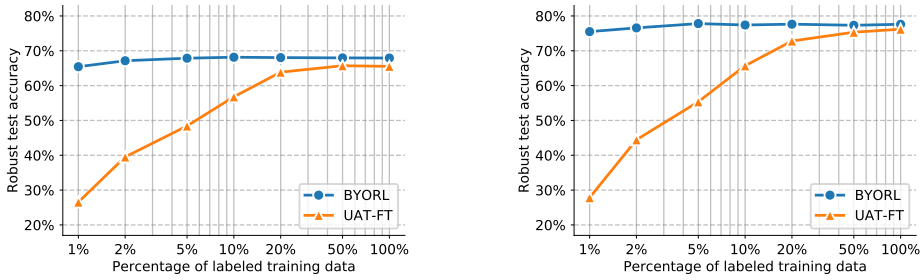
Architecture. We use a convolutional residual network (He et al., 2015) with 34 layers (Pre-Activation ResNet-34) as our encoder e. We also use wider (from $\times 1$ to $\times 4$) ResNets. The projector g and predictor q networks are MLPs with hidden dimension 4096 and output dimension 256.

Outer optimization. We use the LARS optimizer (You et al., 2017) with a cosine learning rate schedule (Loshchilov & Hutter, 2017) over 1000 epochs. We set the learning rate to 2 and use a global weight decay parameter of $5 \cdot 10^{-4}$. For the target network, the exponential moving average parameter τ starts from 0.996 and is increased to one during training. We use a batch size of 512.

Inner optimization. The inner minimization in Eq. 7 is implemented using K PGD steps (constrained by an ℓ_2 or ℓ_∞ norm-bounded ball). Unless specified otherwise, we set K to 40 and use an adaptive step size α (see Algorithm 1 in Croce & Hein, 2020). For ℓ_∞ and ℓ_2 norm-bounded perturbations, the gradients in Eq. 6 are first normalized to their sign or by their ℓ_2 norm, respectively.

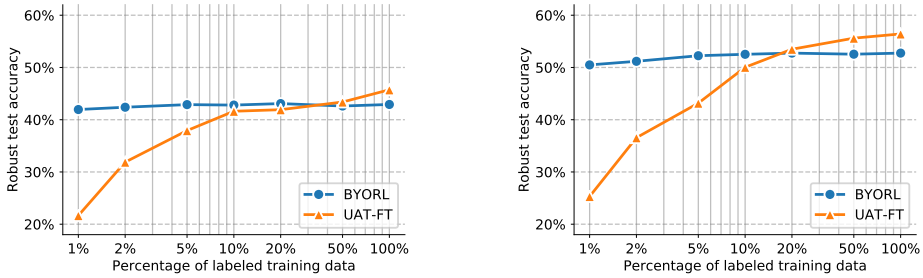
Evaluation protocol. We evaluate the performance of BYORL on CIFAR-10 against adversarial ℓ_2 and ℓ_∞ norm-bounded perturbations (CIFAR-100 and IMAGENET results are in the appendix). For that purpose, we train a linear classifier parametrized by coefficients $\boldsymbol{W}$ and offsets $\boldsymbol{b}$ on top of frozen BYORL representations, following the procedure described in Kolesnikov et al. (2019); Chen et al. (2020b). The linear model is either trained in a non-robust manner (i.e., $\min_{\boldsymbol{W},\boldsymbol{b}} \mathbb{E}_{(\boldsymbol{x},y)\in\mathcal{D}} \hat{l}(\boldsymbol{W}e(\boldsymbol{x};\boldsymbol{\theta}) + \boldsymbol{b}, y)$) or adversarially (i.e., $\min_{\boldsymbol{W},\boldsymbol{b}} \mathbb{E}_{(\boldsymbol{x},y)\in\mathcal{D}} \max_{\boldsymbol{\delta}\in\mathbb{S}} \hat{l}(\boldsymbol{W}e(\boldsymbol{x}+\boldsymbol{\delta};\boldsymbol{\theta}) + \boldsymbol{b}, y)$). We then compute the robust accuracy, which is the accuracy of the combined model $\boldsymbol{W}e(\cdot;\boldsymbol{\theta}) + \boldsymbol{b}$ against adversarial attacks (i.e., we count a successful attack as a misclassification): $1 - \mathbb{E}_{(\boldsymbol{x},y)\in\mathcal{D}} \max_{\boldsymbol{\delta}\in\mathbb{S}} l(\boldsymbol{W}e(\boldsymbol{x}+\boldsymbol{\delta};\boldsymbol{\theta}) + \boldsymbol{b}, y)$. In order to get faithful results, all models are evaluated using a strong attack which combines elements of the AutoAttack procedure (Croce & Hein, 2020) with the MultiTargeted attack (Gowal et al., 2019). Namely, we use a sequence of AutoPGD on the cross-entropy loss with 5 restarts and 100 steps, AutoPGD on the difference of logits ratio loss with 5 restarts and 100 steps, MultiTargeted on the margin loss with 20 restarts and 200 steps and Square (Andriushchenko et al., 2019), an efficient black-box attack, with 5000 queries.

Baseline. Throughout the experimental section, we compare BYORL with adversarial training (combined with *pseudo-labeling* to handle missing labels). *Pseudo-labeling* is currently, to the best of our knowledge, the most successful semi-supervised method for learning adversarially robust models (Carmon et al., 2019; Uesato et al., 2019; Zhai et al., 2019a; Najafi et al., 2019). More specifically, we use Unsupervised Adversarial Training with Fixed Targets (UAT-FT) (Uesato et al., 2019). When 100% of the labels are available UAT-FT is equivalent to classical adversarial training, as proposed by Madry et al. (2017). In settings where less than 100% of the labels are available, we train a separate non-robust model (with an architecture identical to the robust model being trained) on the available labeled data and use it to *pseudo-label* the rest of the unlabeled images. UAT-FT uses the same network architectures than those used by BYORL.



(a) CIFAR-10 only (b) CIFAR-10 and 80M-TINYIMAGES

Figure 3: Accuracy under ℓ_2 attack of size $\epsilon = 128/255$ for different CIFAR-10 models as a function of the ratio of available labels. Panel a restricts the available data to CIFAR-10 only (labeled and unlabeled), while panel b uses 500K additional unlabeled images extracted from 80M-TINYIMAGES.



(a) CIFAR-10 only (b) CIFAR-10 and 80M-TINYIMAGES

Figure 4: Accuracy under ℓ_∞ attack of size $\epsilon = 8/255$ for different CIFAR-10 models as a function of the ratio of available labels. Panel a restricts the available data to CIFAR-10 only (labeled and unlabeled), while panel b uses 500K additional unlabeled images extracted from 80M-TINYIMAGES.

4.2 RESULTS

Robustness on CIFAR-10. We evaluate BYORL and UAT-FT on a wide range of tasks across different threats for various amounts of available labels. As is typical in the literature (Rice et al., 2020; Augustin et al., 2020), we evaluate our models on CIFAR-10 against ℓ_2 and ℓ_∞ norm-bounded perturbations of size $\epsilon = 128/255$ and $\epsilon = 8/255$ (CIFAR-100 and IMAGENET are evaluated in the appendix). CIFAR-10 contains 60K images (i.e., 50K in the train set and 10K in the test set). As such, when we evaluate on 1% of labeled data, we only use 500 random labeled images from CIFAR-10 (we do not artificially balance the number of labels per class). As done in Carmon et al. (2019) and Uesato et al. (2019), we also explore the limits of BYORL in the setting where additional unlabeled data is available. This additional data is extracted from 80M-TINYIMAGES and consists of 500K unlabeled 32×32 images[3]. In settings without this additional data, we use a ResNet-34×2, whereas in settings with this additional data, we use a ResNet-34×4.

Fig. 3 shows the robust accuracy of BYORL and UAT-FT on the full CIFAR-10 test set against ℓ_2 norm-bounded perturbations (similar figures that show clean accuracy are available in Appendix B). We observe that linear classifiers trained on top of robust BYORL representations are more robust than those trained with UAT-FT. In particular, we highlight than when only 500 labeled images are available, BYORL remains competitive with state-of-the-art methods that use all labels: without additional data from 80M-TINYIMAGES, BYORL reaches 65.43% compared to 69.24% (Engstrom et al., 2019); with additional data from 80M-TINYIMAGES, BYORL reaches 75.50% compared to 72.91% (Augustin et al., 2020).

Fig. 4 shows the robust accuracy of BYORL and UAT-FT on the full CIFAR-10 test set against ℓ_∞ norm-bounded perturbations. Again, we can observe that BYORL remains competitive: in the low-label regime, BYORL surpasses UAT-FT by a significant margin (up to $2\times$ more accurate); in the high-label regime BYORL loses a few percentage points. Perhaps surprisingly, under both threat models (i.e., ℓ_∞ and ℓ_2), BYORL reaches peak performance with 5% of the labels (and beyond).

[3]We use the dataset from Carmon et al. (2019) available at https://github.com/yaircarmon/semisup-adv.

Table 1: Robust accuracy (under adversarial attack) obtained by finetuning a linear head on top of robust representations trained on CIFAR-10.

METHOD	NORM	RADIUS	STL-10 1%	STL-10 10%	STL-10 100%	CIFAR-100 1%	CIFAR-100 10%	CIFAR-100 100%
BYORL	ℓ_2	$\epsilon = 128/255$	33.85%	**53.23%**	**57.88%**	**10.09%**	**22.51%**	**28.24%**
UAT-FT			**37.71%**	42.16%	53.23%	3.68%	10.80%	14.43%
BYORL	ℓ_∞	$\epsilon = 8/255$	**24.18%**	36.30%	37.28%	**5.28%**	**10.12%**	**14.82%**
UAT-FT			23.47%	**36.52%**	**37.79%**	2.18%	4.88%	7.21%

Table 2: Clean (no perturbations) and robust (under adversarial attack) accuracy obtained when training robust and non-robust representations on CIFAR-10 against ℓ_2 norm-bounded perturbations of size $\epsilon = 128/255$. We evaluate the representations by finetuning a robust and non-robust linear head on CIFAR-10, STL-10 and CIFAR-100. Table 6 in the appendix shows ℓ_∞ perturbations.

TRAINING OF REPRESENTATION	TRAINING OF LINEAR HEAD	NORM	RADIUS	CIFAR-10 Clean	CIFAR-10 Robust
Robust (BYORL)	Robust (AT) on CIFAR-10	ℓ_2	$\epsilon = 128/255$	93.01%	**77.61%**
Robust (BYORL)	Non-robust on CIFAR-10			93.19%	77.09%
Non-robust (BYOL)	Robust (AT) on CIFAR-10			91.23%	0.05%
Non-robust (BYOL)	Non-robust on CIFAR-10			**94.76%**	0.00%
FINETUNING OF LINEAR HEAD				STL-10	
Robust (BYORL)	Robust (AT) on STL-10	ℓ_2	$\epsilon = 128/255$	76.49%	**57.88%**
Robust (BYORL)	Non-robust on STL-10			**77.54%**	57.66%
FINETUNING OF LINEAR HEAD				CIFAR-100	
Robust (BYORL)	Robust (AT) on CIFAR-100	ℓ_2	$\epsilon = 128/255$	48.34%	**28.24%**
Robust (BYORL)	Non-robust on CIFAR-100			**49.20%**	27.17%

Transfer to unseen tasks. We evaluate our robust representations on other classification datasets to assess whether the features learned on CIFAR-10 are generic and thus useful across image domains, or if they are CIFAR-10-specific. As a comparison, we test whether pre-logits activations resulting from training a model using UAT-FT can also be used for transfer learning. For both representations, we train a robust linear model using adversarial training (see subsection 3.1) with different label availability on STL-10 and CIFAR-100 against ℓ_∞ and ℓ_2 norm-bounded perturbations. Table 1 shows that BYORL representations result in equivalent or more robust models than UAT-FT representations. We note, however, that – at least on CIFAR-100 – the robust accuracy remains significantly lower than models trained directly on CIFAR-100.

Transfer without adversarial training. So far, the linear classifiers trained on top of BYORL representations were trained robustly using adversarial training. We now evaluate whether adversarial training is needed for downstream tasks. Conversely, we also verify that learning robust representations is needed to obtain robust linear classifiers. Table 2 shows the robust accuracy of four models: *(i)* an adversarially trained linear model on top of robust BYORL representations, *(ii)* a classically trained (not necessarily robust) linear model on top of robust BYORL representations, *(iii)* an adversarially trained linear model on top of non-robust BYOL representations, and *(iv)* a classically trained linear model on top of non-robust BYOL representations. Although not a guarantee in theory (Allen-Zhu & Li, 2020), we observe that the adversarial training of the linear classifier is, in practice, not necessary and that it is enough to train robust representations to obtain robust classifiers. Indeed, for all three considered downstream tasks (CIFAR-10, STL-10 and CIFAR-100), the resulting non-robustly trained linear classifiers are within a few percentage points of the robustly trained ones (similar results for IMAGENET are available in the appendix in Table 5).

5 CONCLUSION

In this work, we present BYORL, a modification of BYOL that enables us to train robust image representations. To the contrary of previous methods, BYORL does not require the presence of label information. In fact, it is even possible to use these robust representations to train adversarially robust classifiers on multiple downstream tasks (without the need to use adversarial training). Interestingly, classifiers using BYORL representations can be trained with as little as 500 labeled examples. Across all experiments, BYORL with 1% of labels (i.e., 500 labeled examples) matches or surpasses the performance of *pseudo-labeling* (implemented through UAT-FT) with 10-20% of labels (i.e., between 5K and 10K labeled examples).

ACKNOWLEDGMENTS

We would like to thank Jean-Baptiste Alayrac, Olivier Henaff, Florian Strub, Florent Altché for helpful discussions throughout this work.

REFERENCES

Zeyuan Allen-Zhu and Yuanzhi Li. Feature purification: How adversarial training performs robust deep learning. *arXiv preprint arXiv:2005.10190*, 2020. URL https://arxiv.org/pdf/2005.10190.

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square Attack: a query-efficient black-box adversarial attack via random search. *arXiv preprint arXiv:1912.00049*, 2019. URL https://arxiv.org/pdf/1912.00049.

Anish Athalye and Ilya Sutskever. Synthesizing robust adversarial examples. *arXiv preprint arXiv:1707.07397*, 2017.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420*, 2018. URL https://arxiv.org/pdf/1802.00420.

Maximilian Augustin, Alexander Meinke, and Matthias Hein. Adversarial Robustness on In-and Out-Distribution Improves Explainability. *arXiv preprint arXiv:2003.09461*, 2020. URL https://arxiv.org/pdf/2003.09461.

Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *NeurIPS*, 2014.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. MixMatch: A Holistic Approach to Semi-Supervised Learning. *arXiv:1905.02249*, 2019.

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 387–402. Springer, 2013.

Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao, and Karol Zieba. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016. URL https://arxiv.org/pdf/1604.07316.

Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pp. 3–14. ACM, 2017a.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy*, pp. 39–57. IEEE, 2017b.

Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, John C Duchi, and Percy S Liang. Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems*, pp. 11190–11201, 2019. URL https://papers.nips.cc/paper/9298-unlabeled-data-improves-adversarial-robustness.pdf.

Taylan Cemgil, Sumedh Ghaisas, Krishnamurthy (Dj) Dvijotham, and Pushmeet Kohli. Adversarially Robust Representations with Smooth Encoders. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/pdf?id=H1gfFaEYDS.

Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. *Semi-Supervised Learning*. MITPress, 2009.

Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. *arXiv preprint arXiv:2003.12862*, 2020a. URL https://arxiv.org/pdf/2003.12862.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020b. URL https://arxiv.org/pdf/2002.05709.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. Proceedings of Machine Learning Research, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011. URL http://proceedings.mlr.press/v15/coates11a.html.

Paul Covington, Jay Adams, and Emre Sargin. Deep neural networks for YouTube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, 2016.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. *arXiv preprint arXiv:2003.01690*, 2020. URL https://arxiv.org/pdf/2003.01690.

Jeffrey De Fauw, Joseph R Ledsam, Bernardino Romera-Paredes, Stanislav Nikolov, Nenad Tomasev, Sam Blackwell, Harry Askham, Xavier Glorot, Brendan O'Donoghue, Daniel Visentin, George van den Driessche, Balaji Lakshminarayanan, Clemens Meyer, Faith Mackinder, Simon Bouton, Kareem Ayoub, Reena Chopra, Dominic King, Alan Karthikesalingam, Cían O Hughes, Rosalind Raine, Julian Hughes, Dawn A Sim, Catherine Egan, Adnan Tufail, Hugh Montgomery, Demis Hassabis, Geraint Rees, Trevor Back, Peng T Khaw, Mustafa Suleyman, Julien Cornebise, Pearse A Keane, and Olaf Ronneberger. Clinically applicable deep learning for diagnosis and referral in retinal disease. In *Nature Medicine*, 2018. URL https://www.nature.com/articles/s41591-018-0107-6.pdf.

Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Computer Vision and Pattern Recognition*, 2015.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting Adversarial Attacks with Momentum. *arXiv preprint arXiv:1710.06081*, 2017. URL https://arxiv.org/pdf/1710.06081.

Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *Neural Information Processing Systems*, 2014.

Logan Engstrom, Andrew Ilyas, Hadi Salman, Shibani Santurkar, and Dimitris Tsipras. Robustness (python library), 2019. URL https://github.com/MadryLab/robustness.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Sven Gowal, Jonathan Uesato, Chongli Qin, Po-Sen Huang, Timothy Mann, and Pushmeet Kohli. An Alternative Surrogate Loss for PGD-based Adversarial Testing. *arXiv preprint arXiv:1910.09338*, 2019. URL https://arxiv.org/pdf/1910.09338.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning. *arXiv preprint arXiv:2006.07733*, 2020. URL https://arxiv.org/pdf/2006.07733.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. URL https://arxiv.org/pdf/1512.03385.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. *arXiv preprint arXiv:1906.12340*, 2019. URL https://arxiv.org/pdf/1906.12340.

Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

Lang Huang, Chao Zhang, and Hongyang Zhang. Self-Adaptive Training: beyond Empirical Risk Minimization. *arXiv preprint arXiv:2002.10319*, 2020. URL https://arxiv.org/pdf/2002.10319.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, 2015.

Harini Kannan, Alexey Kurakin, and Ian Goodfellow. Adversarial Logit Pairing. *arXiv preprint arXiv:1803.06373*, 2018.

Minseon Kim, Jihoon Tack, and Sung Ju Hwang. Adversarial Self-Supervised Contrastive Learning. *arXiv preprint arXiv:2006.07589*, 2020. URL https://arxiv.org/pdf/2006.07589.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. In *Computer Vision and Pattern Recognition*, 2019.

Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better ImageNet models transfer better? In *Computer Vision and Pattern Recognition*, 2019.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The CIFAR-10 dataset. 2014. URL http://www.cs.toronto.edu/kriz/cifar.html.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016. URL https://arxiv.org/pdf/1607.02533.

Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learnings. In *ICLR*, 2017.

Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Takeru Miyato, Shin ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning. *TPAMI*, 2018.

Marius Mosbach, Maksym Andriushchenko, Thomas Trost, Matthias Hein, and Dietrich Klakow. Logit Pairing Methods Can Fool Gradient-Based Attacks. *arXiv preprint arXiv:1810.12042*, 2018. URL https://arxiv.org/pdf/1810.12042.

Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *International Conference on Machine Learning*, 2010.

Amir Najafi, Shin-ichi Maeda, Masanori Koyama, and Takeru Miyato. Robustness to adversarial perturbations in learning from incomplete data. *arXiv preprint arXiv:1905.13021*, 2019. URL https://arxiv.org/pdf/1905.13021.

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, 2016.

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

Tianyu Pang, Xiao Yang, Yinpeng Dong, Kun Xu, Hang Su, and Jun Zhu. Boosting Adversarial Training with Hypersphere Embedding. *arXiv preprint arXiv:2002.08619*, 2020. URL https://arxiv.org/pdf/2002.08619.

Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. *arXiv preprint arXiv:1511.04508*, 2015.

Chongli Qin, James Martens, Sven Gowal, Dilip Krishnan, Krishnamurthy Dvijotham, Alhussein Fawzi, Soham De, Robert Stanforth, and Pushmeet Kohli. Adversarial Robustness through Local Linearization. *arXiv preprint arXiv:1907.02610*, 2019. URL https://arxiv.org/pdf/1907.02610.

Leslie Rice, Eric Wong, and J. Zico Kolter. Overfitting in adversarially robust deep learning. *arXiv preprint arXiv:2002.11569*, 2020. URL https://arxiv.org/pdf/2002.11569.

Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *NeurIPS*, 2016.

Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially Robust Generalization Requires More Data. In *Advances in Neural Information Processing Systems*. 2018. URL http://papers.nips.cc/paper/7749-adversarially-robust-generalization-requires-more-data.pdf.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.

Antonio Torralba, Rob Fergus, and William T. Freeman. 80 million tiny images: a large dataset for non-parametric object and scene recognition. *TPAMI*, 2008.

Jonathan Uesato, Brendan O'Donoghue, Aaron van den Oord, and Pushmeet Kohli. Adversarial Risk and the Dangers of Evaluating Against Weak Attacks. *arXiv preprint arXiv:1802.05666*, 2018.

Jonathan Uesato, Jean-Baptiste Alayrac, Po-Sen Huang, Robert Stanforth, Alhussein Fawzi, and Pushmeet Kohli. Are labels required for improving adversarial robustness? *arXiv preprint arXiv:1905.13725*, 2019. URL https://arxiv.org/pdf/1905.13725.

Laurenz Wiskott and Terrence J Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, 14(4):715–770, 2002.

Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020. URL https://arxiv.org/pdf/2001.03994.

Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance-level discrimination. *arXiv preprint arXiv:1805.01978*, 2018. URL https://arxiv.org/pdf/1805.01978.

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V. Le. Unsupervised Data Augmentation. *arXiv:1904.12848*, 2019.

Yang You, Igor Gitman, and Boris Ginsburg. Scaling SGD batch size to 32k for ImageNet training. *arXiv preprint arXiv:1708.03888*, 2017.

Runtian Zhai, Tianle Cai, Di He, Chen Dan, Kun He, John Hopcroft, and Liwei Wang. Adversarially Robust Generalization Just Requires More Unlabeled Data. *arXiv preprint arXiv:1906.00555*, 2019a. URL https://arxiv.org/pdf/1906.00555.

Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. *arXiv preprint arXiv:1905.03670*, 2019b. URL https://arxiv.org/pdf/1905.03670.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically Principled Trade-off between Robustness and Accuracy. *arXiv preprint arXiv:1901.08573*, 2019. URL https://arxiv.org/pdf/1901.08573.

Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In *European Conference on Computer Vision*, 2016.

Daniel Zoran, Mike Chrzanowski, Po-Sen Huang, Sven Gowal, Alex Mott, and Pushmeet Kohli. Towards robust image classification using sequential attention models. In *CVPR*, 2020.

A EXPERIMENTAL SETUP

Image augmentations. BYORL uses similar image augmentations to SimCLR (Chen et al., 2020b) and BYOL (Grill et al., 2020). First, a random patch of the image is selected and resized to 32×32 with a random horizontal flip, followed by a color distortion, consisting of a random sequence of brightness, contrast, saturation, hue adjustments, and an optional grayscale conversion. Because of the low image resolution of CIFAR-10 and CIFAR-100, we do not apply a final Gaussian blur and solarization to the patches.

Architecture. For consistency with prior work on adversarial robustness (Rice et al., 2020; Wong et al., 2020), we use a convolutional residual network (He et al., 2015) with 34 layers (Pre-Activation ResNet-34) as our encoder e. We also use wider (from $\times 1$ to $\times 4$) ResNets. The final representation $\boldsymbol{h}$ is the output of the final average pooling layer, which has a feature dimension of 2048 (when the width multiplier is $\times 1$). The projector g and predictor q networks are *multi-layer perceptrons*. They consist of a linear layer with output size 4096 followed by batch normalization (Ioffe & Szegedy, 2015), rectified linear units (ReLU) (Nair & Hinton, 2010), and a final linear layer with output dimension 256.

Outer optimization. We use the LARS optimizer (You et al., 2017) with a cosine decay learning rate schedule (Loshchilov & Hutter, 2017) over 1000 epochs, with a warm-up period of 10 epochs. We set the learning rate to 2 and use a global weight decay parameter of $5 \cdot 10^{-4}$. For the target network, the exponential moving average parameter τ starts from 0.996 and is increased to one during training using a cosine schedule. We use a batch size of 512 split over 32 Google Cloud TPU v3 cores. We train linear classifiers using the Adam optimizer (Kingma & Ba, 2014) for 50 epochs with an initial learning of $2 \cdot 10^{-3}$ (the learning rate is decayed by a factor $10\times$ after 25 epochs). For linear classifiers, we perform early stopping as suggested by Rice et al. (2020) using a separate set of 1024 validation images (we do the same when training UAT-FT models). Note that linear models trained on frozen BYORL representations do not really need early stopping (as the linear model do not overfit significantly). However, UAT-FT and adversarial training are prone to robust overfitting and obtain significantly higher robust accuracy using early stopping.

Inner optimization. The inner minimization in Eq. 7 is implemented using K PGD steps (constrained by an ℓ_2 or ℓ_∞ norm-bounded ball). In particular, unless specified otherwise, we set K to 40 and use an adaptive step size (as specified in Algorithm 1 in Croce & Hein, 2020). For ℓ_∞ and ℓ_2 norm-bounded perturbations, the gradients in Eq. 6 are first normalized to their sign or by their ℓ_2 norm, respectively. With this setup, training takes approximately 19 hours for a ResNet-34×2 and 3 days for a ResNet-34×4.

Evaluation protocol. We evaluate the performance of BYORL on CIFAR-10 and CIFAR-100 against adversarial ℓ_2 and ℓ_∞ norm-bounded perturbations. For that purpose, we train a linear classifier parametrized by coefficients $\boldsymbol{W}$ and offsets $\boldsymbol{b}$ on top of frozen BYORL representations, following the procedure described in Kolesnikov et al. (2019); Kornblith et al. (2019); Zhang et al. (2016); Chen et al. (2020b). The linear model is either trained in a non-robust manner (i.e., $\min_{\boldsymbol{W},\boldsymbol{b}} \mathbb{E}_{(\boldsymbol{x},y)\in\mathcal{D}} \hat{l}(\boldsymbol{W}e(\boldsymbol{x};\boldsymbol{\theta}) + \boldsymbol{b}, y)$) or adversarially (i.e., $\min_{\boldsymbol{W},\boldsymbol{b}} \mathbb{E}_{(\boldsymbol{x},y)\in\mathcal{D}} \max_{\boldsymbol{\delta}\in\mathbb{S}} \hat{l}(\boldsymbol{W}e(\boldsymbol{x} + \boldsymbol{\delta};\boldsymbol{\theta}) + \boldsymbol{b}, y)$). We then freeze the linear model and compute the accuracy of the combined model $\boldsymbol{W}e(\boldsymbol{x};\boldsymbol{\theta}) + \boldsymbol{b}$ against adversarial attacks (i.e., we count a successful attack as a misclassification). In other words, we approximate the result of Eq. 1 using an inner-maximization procedure (i.e., adversarial attack). In order to get faithful results, all models are evaluated using a strong attack which combines elements of the AutoAttack procedure (Croce & Hein, 2020) with the MultiTargeted attack (Gowal et al., 2019). Namely, we use a sequence of AutoPGD on the cross-entropy loss with 5 restarts and 100 steps, AutoPGD on the difference of logits ratio loss with 5 restarts and 100 steps, MultiTargeted on the margin loss with 20 restarts and 200 steps and Square (Andriushchenko et al., 2019), an efficient black-box attack, with 5000 queries.

B ADDITIONAL EXPERIMENTS

In this section, we perform ablation studies on network architectures, strength of the attack used during representation learning, and color augmentation strength.

Network size. In Table 3, we study the impact of model architectures on CIFAR-10 against ℓ_∞ norm-bounded perturbations of size $\epsilon = 8/255$. We train robust representations using wider ResNet, as in (Chen et al., 2020b; Grill et al., 2020), with 34 layers and a width multiplier 1, 2 and 4, and denoted by ResNet34$\times$1, ResNet34$\times$2, ResNet34$\times$4. We observe that models increase performance in both clean and robust accuracy when model size increases. Without using additional unlabeled data, the robust accuracy saturates when using ResNet34$\times$2. Hence, unless mentioned explicitly, in this paper, we use ResNet34$\times$2 for settings without additional data.

Table 3: Clean (no perturbations) and robust (under adversarial attack) accuracy obtained by networks of different sizes on CIFAR-10 against ℓ_∞ norm-bounded perturbations of size $\epsilon = 8/255$.

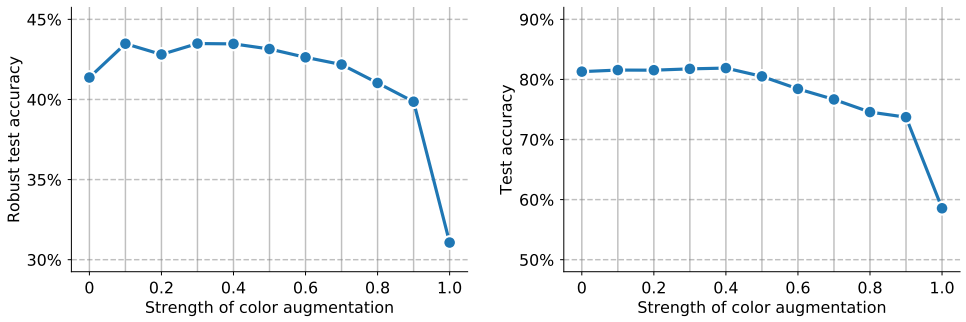
ARCHITECTURE	NORM	RADIUS	CIFAR-10	
			Clean	Robust
RESNET 34x1	ℓ_∞	$\epsilon = 8/255$	78.02%	41.30%
RESNET 34x2			80.80%	43.24%
RESNET 34x4			81.32%	43.19%

Attack strength. Table 4 shows results of training robust representations using different attack strengths on CIFAR-10 against ℓ_∞ norm-bounded perturbations of size $\epsilon = 8/255$. We study the effect of using standard PGD, as well as its adaptive variant named AutoPGD (Croce & Hein, 2020), with different step sizes using ResNet34$\times$2 models. We observe that more robust representations are obtained as the attack strength increases. Although this finding is similar to previous observations (Qin et al., 2019), we notice that robust representation learning is more prone to gradient masking (resulting from using weak attacks during training) and requires stronger attacks then typically used in classical adversarial training. Throughout the paper, unless mentioned explicitly, we used AutoPGD with 40 steps.

Table 4: Clean (no perturbations) and robust (under adversarial attack) accuracy obtained by representations trained against attacks of different strength on CIFAR-10 against ℓ_∞ norm-bounded perturbations of size $\epsilon = 8/255$.

ATTACK	STEPS	NORM	RADIUS	CIFAR-10	
				Clean	Robust
PGD	5	ℓ_∞	$\epsilon = 8/255$	85.26%	1.05%
	10			81.44%	37.88%
	20			80.29%	42.58%
	40			79.60%	43.41%
AUTOPGD	10	ℓ_∞	$\epsilon = 8/255$	80.49%	40.59%
	20			80.53%	43.27%
	40			80.47%	43.64%
	60			80.69%	43.64%

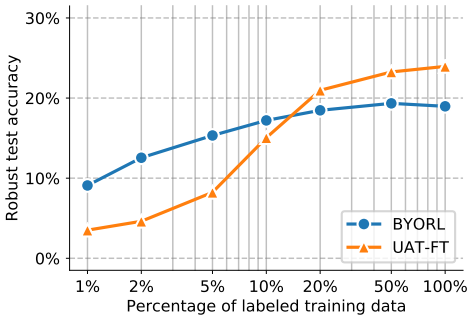
Impact of color augmentations. In Chen et al. (2020b) and Grill et al. (2020), the authors demonstrate the importance of using color augmentations, which is composed color jittering and color dropping. We follow the same setup as Chen et al. (2020b) and Grill et al. (2020), in Fig. 5, we perform an ablation study on the strength of color augmentations. We use AutoPGD with 20 steps on a ResNet34$\times$2 model. We observe that BYORL is relatively stable in both clean and robust accuracy for color augmentation strength between 0.1 and 0.5. In this paper, we had fixed the color augmentation strength to be 0.5 across all the experiments. A better choice in retrospect would have been 0.3 which provides an improvement of 0.34%.



(a) Accuracy under attack

(b) Clean accuracy

Figure 5: Accuracy on CIFAR-10 for different strengths of the color augmentation. Panel a shows the accuracy under ℓ_∞ attacks of size $\epsilon = 8/255$, while panel b shows the corresponding clean accuracy.



(a) CIFAR-100 only

Figure 6: Accuracy under ℓ_∞ attack of size $\epsilon = 8/255$ for different CIFAR-100 models as a function of the ratio of available labels. The available data is restricted to CIFAR-100 only (labeled and unlabeled).

CIFAR-100. We repeat the experiments in subsection 4.2 on CIFAR-100. For CIFAR-100, we expected BYORL to perform significantly better than UAT-FT as the number of classes is ten times larger than on CIFAR-10. However, Fig. 6 shows a more nuanced story and remains similar to Fig. 4. It is interesting to observe that BYORL tends to struggle with ℓ_∞ norm-bounded perturbations of size $\epsilon = 8/255$. As it has long been known that ℓ_∞ norm-bounded perturbations of size $\epsilon = 8/255$ are harder to cope with than ℓ_2 norm-bounded perturbations of size $\epsilon = 128/255$, we posit that, at equal model capacity, the pretext task (consisting of augmenting two different views of the same image) needs to be slightly modified to accommodate the more difficult threat. Indeed, the investigation on the impact of the color augmentation strength showed that it is possible to improve robust representations by reducing the strength of these augmentations.

IMAGENET. We repeat a subset of the experiments from subsection 4.2 on IMAGENET. For IMAGENET, we train BYORL representations with a ResNet 50×4 for 300 epochs. Following Grill et al. (2020) and Chen et al. (2020b), we add random blur to the image preprocessing pipeline. We also change the learning rate from 2 to 1.2 and the initial target network weight decay rate τ to 0.999. We use a batch size of 1024 split over 128 Google Cloud TPU v3 cores. All other parameters remain identical to other experiments.

Table 5 shows that BYORL is capable of learning robust representations for larger datasets such as IMAGENET. In particular, our best model achieves 45.44% robust accuracy against ℓ_∞ norm-bounded perturbations of size $\epsilon = 4/255$ (as evaluated by the combined set of attacks consisting of AutoPGD on the cross-entropy loss with 5 restarts and 100 steps, AutoPGD on the difference of logits ratio loss with 5 restarts and 100 steps). Note that classical adversarial training achieves 39.7% (when evaluated against PGD on the cross-entropy loss with 1 restart and 100 steps), while the current state-of-the-art

is 47.00% (Qin et al., 2019). While BYORL does not reach the state-of-the-art in robustness, it is remarkable to see it perform so well with only 1% of available labels and reach 31.57% robust accuracy (e.g., classical non-robust supervised finetuning achieves 25.4% top-1 accuracy when no adversarial perturbations are present as stated in Zhai et al., 2019b). Finally, these results also confirm the results from Table 2 whereby training a non-robust linear classifier still achieves significant robust accuracy.

Table 5: Clean (no perturbations) and robust (under adversarial attack) accuracy obtained when training robust representations on IMAGENET against ℓ_∞ norm-bounded perturbations of size $\epsilon = 4/255$. We evaluate the representations by training/finetuning a robust and non-robust linear head on IMAGENET with varying numbers of labels. For completeness, we also add results for ℓ_2 norm-bounded perturbations of size $\epsilon = 128/255$

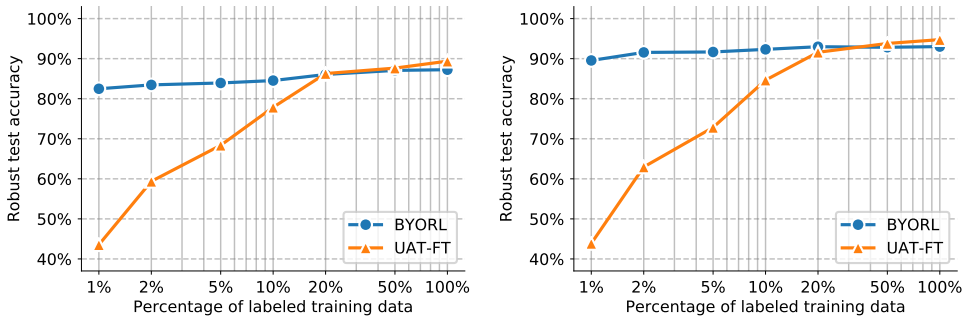
TRAINING OF LINEAR HEAD	NORM	RADIUS	IMAGENET (100%) Clean	IMAGENET (100%) Robust	IMAGENET (10%) Clean	IMAGENET (10%) Robust	IMAGENET (1%) Clean	IMAGENET (1%) Robust
Robust (AT)	ℓ_∞	$\epsilon = 4/255$	65.14%	45.44%	62.39%	41.58%	47.07%	31.57%
Non-robust			65.27%	43.83%	62.40%	41.06%	47.64%	31.99%
Robust (AT)	ℓ_2	$\epsilon = 128/255$	69.64%	65.48%	66.39%	61.90%	55.06%	51.00%

Transfer without adversarial training against ℓ_∞ norm-bounded perturbations. In subsection 4.2, we evaluate how robust accuracy degrades when linear classifiers trained on top of BYORL representations are not trained robustly. In Table 2, we evaluate models trained against ℓ_2 norm-bounded perturbations. In Table 6, we evaluate models trained against ℓ_∞ norm-bounded perturbations. In both cases, non-robust downstream classifiers are able to exhibit non-trivial levels of robustness that remain within a few percentage points of the robustly trained ones.

Table 6: Clean (no perturbations) and robust (under adversarial attack) accuracy obtained when training robust and non-robust representations on CIFAR-10 against ℓ_∞ norm-bounded perturbations of size $\epsilon = 8/255$. We evaluate the representations by training/finetuning a robust and non-robust linear head on CIFAR-10, STL-10 and CIFAR-100.

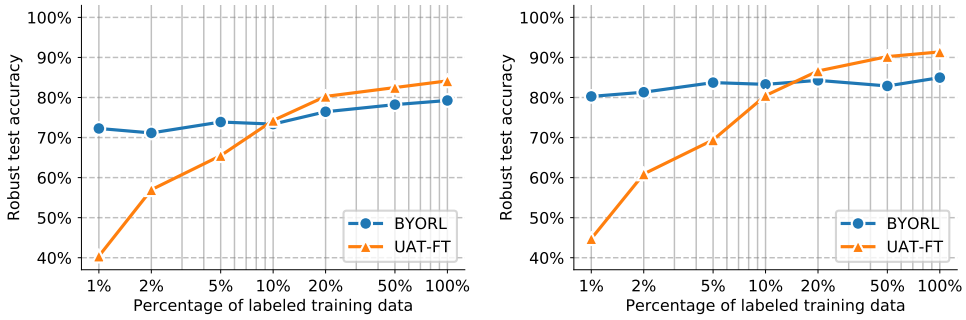
TRAINING OF REPRESENTATION	TRAINING OF LINEAR HEAD	NORM	RADIUS	CIFAR-10 Clean	CIFAR-10 Robust
Robust (BYORL)	Robust (AT) on CIFAR-10	ℓ_∞	$\epsilon = 8/255$	84.96%	**52.75%**
Robust (BYORL)	Non-robust on CIFAR-10			86.57%	50.94%
FINETUNING OF LINEAR HEAD				STL-10	
Robust (BYORL)	Robust (AT) on STL-10	ℓ_∞	$\epsilon = 8/255$	65.40%	**37.79%**
Robust (BYORL)	Non-robust on STL-10			**65.75%**	36.51%
FINETUNING OF LINEAR HEAD				CIFAR-100	
Robust (BYORL)	Robust (AT) on CIFAR-100	ℓ_∞	$\epsilon = 8/255$	37.80%	**14.82%**
Robust (BYORL)	Non-robust on CIFAR-100			**39.51%**	12.21%

C ADDITIONAL FIGURES



(a) CIFAR-10 only (b) CIFAR-10 and 80M-TINYIMAGES

Figure 7: Clean accuracy of same models presented in Fig. 3.



(a) CIFAR-10 only (b) CIFAR-10 and 80M-TINYIMAGES

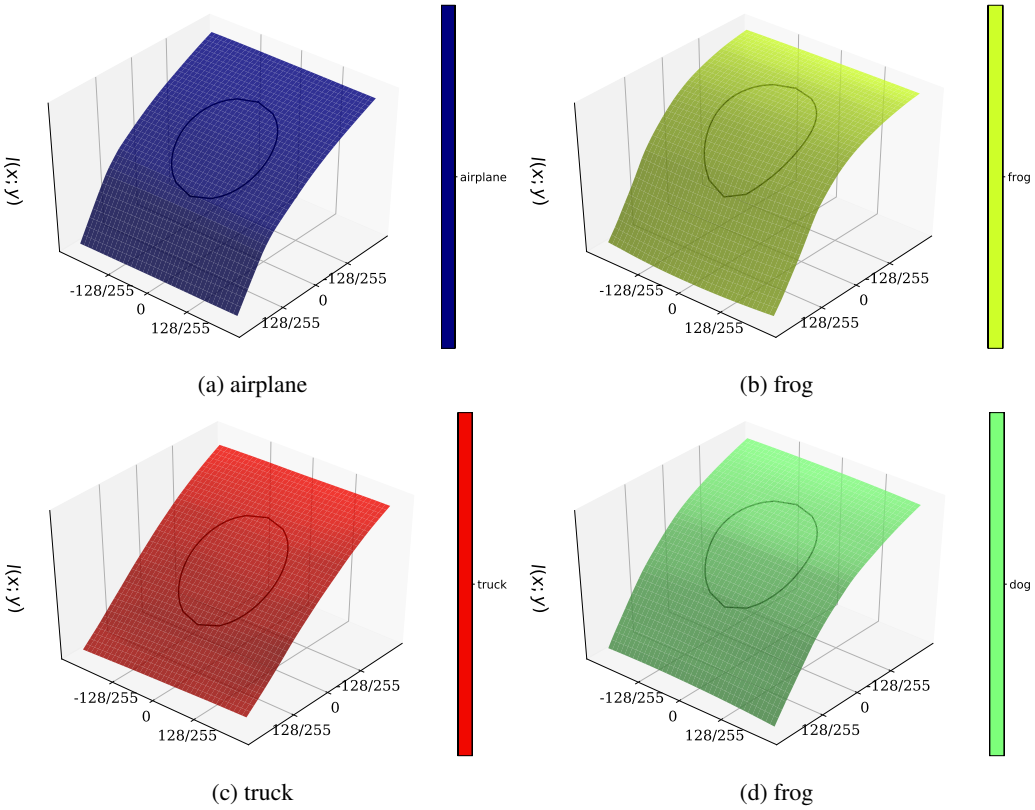
Figure 8: Clean accuracy of same models presented in Fig. 4.

D ANALYSIS OF RESULTING MODELS

In this section, we perform analysis on the adversarial loss landscape from the BYORL models in ℓ_2 and ℓ_∞ cases to further examine whether adversarial accuracy is due to gradient masking (Uesato et al., 2018; Athalye et al., 2018). The analysis made here complements the use of the black-box Square (Andriushchenko et al., 2019) attack used by AutoAttack within our evaluation pipeline. For generating a loss landscape, we vary the input along a linear space defined by the worse perturbations found by PGD (u direction) and a random Rademacher direction (v direction). The u and v axes represent the magnitude of the perturbation added in each of these directions respectively and the z axis represents the adversarial margin loss (Carlini & Wagner, 2017b).

ℓ_2 model. In Fig. 9, we show the adversarial loss landscapes of 4 randomly selected CIFAR-10 test images from the BYORL model trained against ℓ_2 perturbations on both CIFAR-10 and a subset of 80M-TINYIMAGES. The loss surfaces are generally smooth in Fig. 9, which provides evidence that the model performance is not due to gradient obfuscation. We also note that our rigorous evaluation already uses a black-box attack (i.e., Square Andriushchenko et al., 2019).

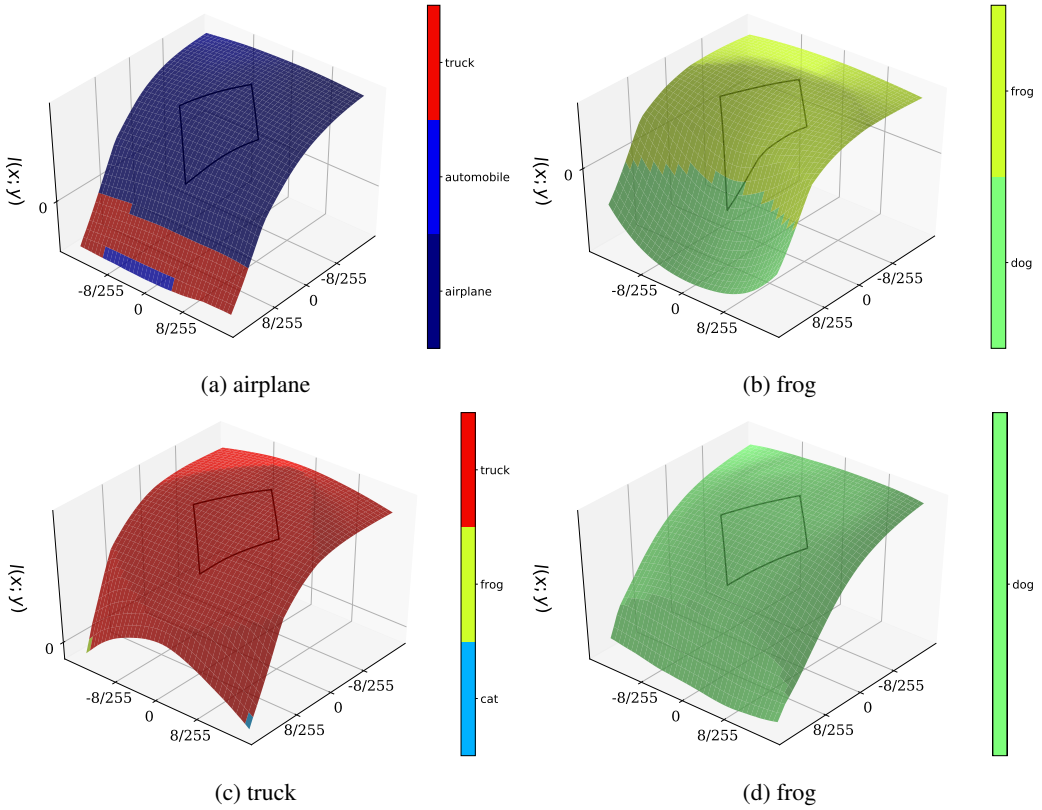
ℓ_∞ model. Fig. 10 shows the adversarial loss landscapes of 4 randomly selected CIFAR-10 test images from the BYORL model trained against ℓ_∞ perturbations on both CIFAR-10 and a subset of 80M-TINYIMAGES. We observe that the loss landscapes are generally smooth in Figure 10, which further suggests the model performance is not due to gradient obfuscation. We also note that our rigorous evaluation already uses a black-box attack (i.e., Square by Andriushchenko et al., 2019).



(a) airplane (b) frog

(c) truck (d) frog

Figure 9: Loss landscapes around different CIFAR-10 test images. It is generated by varying the input to the model, starting from the original input image toward either the worst attack found using PGD (u direction) or a random Rademacher direction (v direction). The loss used for these plots is the margin loss $z_y - \max_{i \neq y} z_i$ (i.e., a misclassification occurs when this value falls below zero). The model used is the BYORL model trained against ℓ_2 perturbations on both CIFAR-10 and a subset of 80M-TINYIMAGES. The circular-shape represents the projected ℓ_2 ball of size $\epsilon = 128/255$ around the nominal image.



(a) airplane (b) frog

(c) truck (d) frog

Figure 10: Loss landscapes around the clean image of different CIFAR-10 test images. It is generated by varying the input to the model, starting from the original input image toward either the worst attack found using PGD (u direction) or a random Rademacher direction (v direction). The loss used for these plots is the margin loss $z_y - \max_{i \neq y} z_i$ (i.e., a misclassification occurs when this value falls below zero). The model used is the BYORL model trained against ℓ_∞ perturbations on both CIFAR-10 and a subset of 80M-TINYIMAGES. The diamond-shape represents the projected ℓ_∞ ball of size $\epsilon = 8/255$ around the nominal image.