Are Adversarially Robust Deep Nets Always Better Transfer Learners?

Damon Falck
Mathematical Institute
University of Oxford

Abstract

Transfer learning uses high-level representations learned by a deep net as a starting point for training on a related task. The ability of transfer learners to quickly achieve high classification accuracy in a data-scarce setting makes them an essential part of modern deep learning. Recent works have shown that robustness of deep nets to small adversarial perturbations, while sometimes coming at the cost of accuracy on the source task, often results in improved performance when transferring to a new domain. In this report, we review the current literature on the effect of adversarial training on learned representations and transfer accuracy, and argue that the positive effects of robustness observed so far **only occur when there is a *human-interpretable* common structure** between source and target task. Specifically, we show empirically that there exist pathological downstream datasets on which transfer accuracy is inversely correlated with pre-trained model robustness. We also discuss the use of source dataset modification to produce better transfer learners without the need for adversarial training.

1 Introduction.

Deep neural networks have recently shown astounding accuracy on a large range of classification tasks. Training large networks, however, requires access both to plentiful data and to expensive computing resources; in many situations training from scratch on a new task is simply not feasible. However, it has been observed that training a large model on a 'generic' classification task such as ImageNet [Den+09] and then applying a few further rounds of optimization on the *new* task at hand is often highly efficient. In particular, starting training from a pre-trained generic model rather than a random initialization offers three advantages: higher *initial* accuracy, *faster* training, and higher *asymptotic* accuracy [Don+14; Yos+14]. These make transfer learning an extremely powerful technique in data-scarce settings.

A separate line of work [Bru+14] studies the resilience of neural networks to small adversarial input perturbations. While this is often considered from a security perspective, it has recently been observed [Sal+20; Utr+20] that such robustness tends to lead to better transfer learning performance when re-training on new tasks, particularly when all but the last layer of weights are *frozen* and only the last layer fine-tuned. This report is a discussion of the theoretical and empirical reasons for this behaviour, and the limitations of its effect.

Contributions. As well as reviewing current ideas, we make the following original contributions:

1. A more general theoretical model of single-source representation transfer learning for multi-label classification.
2. Demonstration of the existence of downstream datasets on which representation robustness hinders transfer learning.
3. Proposing the use of source dataset modification techniques to produce improved transfer learners without adversarial training.

Preprint. Work in progress.

2 Formal models of transfer learning

We start by presenting a simple model of the feature representations learned by a neural net. Consider multi-label classification, where a sample of labelled examples $(\boldsymbol{x},y) \in \mathcal{X} \times [k]$ is drawn from some distribution $\mathcal{D}$ (here $\mathcal{X}$ is some d-dimensional input space, e.g. images of a certain size) and the goal is to learn a classification function $C : \mathcal{X} \to [k]$ that maximises the out-of-sample accuracy $\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}[\mathbb{1}_{C(\boldsymbol{x})=y)}]$. For C a neural net, we may express the prediction on input $\boldsymbol{x}$ as

$$C(\boldsymbol{x}) = \operatorname*{argmax}_{i=1,\ldots,k} f_C(\boldsymbol{x})_i$$

where $f_C(\boldsymbol{x}) = WR(\boldsymbol{x}) + \boldsymbol{b} \in \mathbb{R}^k$ is the final-layer output of C on $\boldsymbol{x}$, i.e. an affine transformation of the penultimate-layer activations $R(\boldsymbol{x}) \in \mathbb{R}^r$.

Definition. *The **representation** of an input $\boldsymbol{x} \in \mathcal{X}$ under a neural net C is the vector $R(\boldsymbol{x}) \in \mathbb{R}^r$.*

Transfer learning aims to use a pre-trained neural net C as a starting point for performing some new, unseen classification task. We normally assume the new task has the same input space $\mathcal{X}$, but the number of classes k' may be different; in this case the output layer of the pre-trained network is first replaced with a (randomly initialized) layer of length k'. There are two primary regimes used in transfer learning:

1. **Re-training the entire model.** The network C may be fully re-trained on the target task, initializing with the pre-learned weights rather than randomly. Here the network may in principle still learn a completely different task (given enough data).
2. **Representation transfer and fine-tuning.** The alternative is to *freeze* the representation in C and fine-tune only the final-layer weights on the new task, thereby training a linear classifier on the representation learned for the source task. This paradigm is often referred to as *representation learning*.

Perhaps surprisingly, the second technique exhibits very good results on a wide array of tasks, and it is this technique which we will focus on in this report.

2.1 Tractability of representation transfer

Data. When is representation transfer effective?[1] Suppose there is a *source* (or *upstream*) classification task $\mathcal{T}_1$ consisting of a distribution $\mathcal{D}_1$ over $\mathcal{X} \times [k_1]$ for some input space $\mathcal{X}$ and number of classes $k_1 \in \mathbb{N}$, and a *target* (or *downstream*) classification task $\mathcal{T}_2$ consisting of a distribution $\mathcal{D}_2$ over $\mathcal{X} \times [k_2]$ for some $k_2 \in \mathbb{N}$. Moreover, assume there is some *true common representation* $R^* : \mathcal{X} \to \mathbb{R}^r$ so that

$$y = \operatorname*{argmax}_{i=1,\ldots,k_t} (W_t^* R^*(\boldsymbol{x}) + \boldsymbol{b_t^*} + \boldsymbol{\eta})_t \quad \forall (\boldsymbol{x},y) \sim \mathcal{D}_t \quad \text{for } t = 1,2$$

where $\boldsymbol{b}_t^* \in \mathbb{R}^{k_t}, W_t^* \in \mathbb{R}^{r \times k_t}$ are fixed 'true' weights and biases and $\boldsymbol{\eta}$ is a centered random noise variable drawn independently for each $\boldsymbol{x}, y$. (The stronger assumption that only the element of $W_1^* R^*(\boldsymbol{x}) + \boldsymbol{b_1^*}$ or $W_2^* R^*(\boldsymbol{x}) + \boldsymbol{b_2^*}$ corresponding to the true class is positive is often realistic and useful.) Intuitively, we assume that the source and target task are chosen to indeed have common structure.

Training. Samples $(\boldsymbol{x}_{t,1},y_{t,1}),\ldots,(\boldsymbol{x}_{t,n_t},y_{t,n_t}) \sim \mathcal{D}_t$, $t = 1,2$ are provided and a model is trained on the source dataset by minimising the in-sample risk:

$$\hat{\phi}, \hat{W}_1, \hat{\boldsymbol{b_1}} := \operatorname*{argmin}_{\phi \in \Phi, W_1 \in \mathbb{R}^{r \times k_1}, \boldsymbol{b_1} \in \mathbb{R}^{k_1}} \frac{1}{n_1} \sum_{i=1}^{n_1} \mathcal{L}_1(y_{1,i}, W_1 \phi(\boldsymbol{x}_{1,i}) + \boldsymbol{b_1})$$

(for some loss function $\mathcal{L}_1 : [k_1] \times \mathbb{R}^{k_1} \to \mathbb{R}$) where Φ is some class of functions (e.g. neural networks of a particular architecture). The model is then fine-tuned to the target task by doing

$$\hat{W}_2, \hat{\boldsymbol{b_2}} := \operatorname*{argmin}_{W_2 \in \mathbb{R}^{r \times k_2}, \boldsymbol{b_2} \in \mathbb{R}^{k_2}} \frac{1}{n_2} \sum_{i=1}^{n_2} \mathcal{L}_2(y_{2,i}, W_2 \hat{\phi}(\boldsymbol{x}_{1,i}) + \boldsymbol{b_2})$$

(for some other loss function $\mathcal{L}_2 : [k_2] \times \mathbb{R}^{k_2} \to \mathbb{R}$). Define then the *excess risk*

$$\mathrm{ER}(\hat{\phi}, \hat{W}_2, \hat{\boldsymbol{b}_2}) := \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}_2} \left[\mathcal{L}_2(\hat{W}_2 \hat{\phi} \boldsymbol{x}_{1,i} + \hat{\boldsymbol{b_2}}) - \mathcal{L}_2(W_2^* \phi(\boldsymbol{x}_{1,i}) + \boldsymbol{b_2^*}) \right].$$

Bounding the excess risk. By arguing that $\hat{R}$ and $\hat{R}^*$ are *close* in some sense, one may prove formal guarantees on $\mathrm{ER}(\hat{\phi}, \hat{W}_2, \hat{\boldsymbol{b}_2})$. Some progress has been made towards this, although none in the multi-class classification setting presented here;[2] [Du+21] show in the *regression* setting that:

[1] Our modelling ideas are similar to some in the existing literature [Den+21; TJJ21; Du+21] but tailored more closely to deep neural nets.

Theorem. *Under certain assumptions on the representation class Φ and the input distribution $\mathcal{D}_1 = \mathcal{D}_2$, for large enough n_1, n_2 the excess risk satisfies with probability $1-\delta$ for any δ*

$$\mathbb{E}_{W_2^*}[\mathrm{ER}(\hat{\phi}, \hat{W}_2, \hat{\boldsymbol{b_2}})] = \sigma^2 O\left(\frac{\mathcal{G}(\mathcal{F}_{\mathcal{X}}(\Phi))^2 + \log\frac{1}{\delta}}{n_1} + \frac{r + \log\frac{1}{\delta}}{n_2}\right),$$

where $\mathcal{G}(\mathcal{F}_{\mathcal{X}}))$ is the Gaussian width of the set $\mathcal{F}_{\mathcal{X}}$ of unit vectors in $\mathrm{span}([\phi(X), \phi'(X)])$ (here X is the matrix of source-task training inputs) for some $\phi, \phi' \in \Phi$ and σ^2 is the variance of the input noise terms.

Similar results were shown by [TJJ21] for when Φ is the class of *linear* representations, and the recent work [Den+21] has adapted this analysis to the binary classification setting (for linear representations), showing that for the loss function $\mathcal{L}(y, \hat{y}) = -y \cdot \hat{y}$ the excess risk is $O\left(\sqrt{\frac{r+\log n_1}{n_2}} + \sqrt{\frac{r^2 d}{n_1}}\right)$.

Effect of adversarial training. Deep neural nets have been shown to be highly sensitive to some small perturbations in the input data [Bru+14]. A standard method of inducing robustness to such adversarial perturbations is *adversarial training*, where the modified objective

$$\frac{1}{n}\sum_{i=1}^{n} \max_{\|\boldsymbol{\delta}\|_q \leqslant \varepsilon} \mathcal{L}(C(\boldsymbol{x}_i + \boldsymbol{\delta}), y_i)$$

is minimised for some small ε (and normally $q = 2$ or ∞) using projected gradient descent. While multiple recent experiments ([Sal+20; Utr+20]) have observed that adversarial source training results in better transfer learning on a number of downstream datasets (c.f. Section 4), only very recently has any progress been made in showing this theoretically. In particular, [Den+21] show that:

Theorem (Informal)**.** *For Φ the class of* linear *representations, under additional data assumptions*[3]

1. *ℓ_2-adversarial training decreases the excess risk if the source classification tasks have sufficiently varying signal-to-noise ratios (i.e. difficulties of classification) and are sufficiently diverse;*
2. *ℓ_∞-adversarial training decreases the excess risk if the source classification tasks have similar signal-to-noise ratios but the input data lies in a low-dimensional subspace of $\mathcal{X}$.*

Extending this result to non-linear representations (e.g. deep neural nets) remains an open problem.

3 Robustness and human-interpretability of representations

Let us discuss the representations themselves. Define a *feature* to be any function $f : \mathcal{X} \to \mathbb{R}$; thus a representation R consists of r learned features. For convenience we assume that all features we consider are shifted and scaled to have zero mean and unit variance, and from now on we consider only *binary* classification, with labels $y \in \{-1, 1\}$. Loosely following [Ily+19], we make the following definitions:

Definition. *A feature f is **useful** if it is correlated with the true label, i.e. if $\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}[y \cdot f(\boldsymbol{x})] > 0$. We say f is **ε-robust** if it is correlated with the true label under adversarial perturbations of size ε, i.e. $\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\inf_{\|\boldsymbol{\delta}\|\leqslant\varepsilon} y \cdot f(\boldsymbol{x}+\boldsymbol{\delta})\right] > 0$, and f is **robust** if it is ε-robust for some $\varepsilon > 0$.*

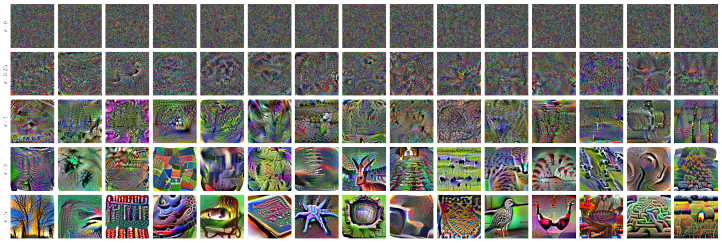
Note that robustness implies usefulness but not vice versa. In particular, there may be useful, non-robust features which are correlated with the true label but which stop being useful under even very small input perturbations. Crucially, these features still have good predictive power so will be learnt.

It is easy to see from this definition that applying ℓ_q-adversarial training with radius ε acts as a prior for learning predominantly ε-robust features (under the q-norm). [Ily+19] show this empirically by *disentangling* the robust from the non-robust features in a given dataset: given an input distribution $\mathcal{D}$ they construct modified distributions $\mathcal{D}_{\mathrm{robust}}, \mathcal{D}_{\mathrm{antirobust}}$ such that: **(a)** the only useful features on $\mathcal{D}_{\mathrm{robust}}$ are those that are ε-robust on $\mathcal{D}$, and **(b)** the only useful features on $\mathcal{D}_{\mathrm{antirobust}}$ are those that are *not* robust on $\mathcal{D}$. We adapt this technique for our own experiments in Section 4.

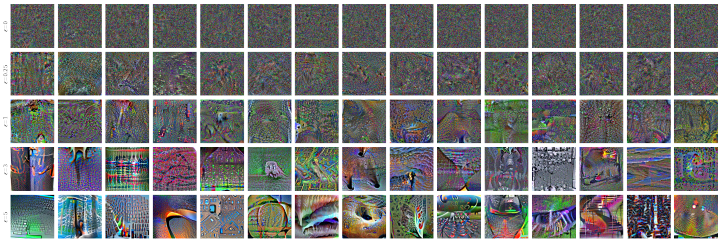
Human-interpretability. It has also been noted that: **(a)** robust features tend to be human-interpretable, whereas non-robust features are not, and **(b)** images with similar robust learned representations tend to be semantically similar, which is not observed for non-robust representations. [Eng+19] show this empirically for robust and non-robust models trained on the Restricted ImageNet dataset; we demonstrate the first point in the next section by visualising the features for a much wider range of robustness levels than earlier works.

[2] Other works consider *multiple source tasks*, each binary (or single-dimensional); this is however approximately equivalent due to the shared representation.

[3] Specifically, this is proven for multiple binary classification source tasks, with $x_i^{(t)} = \eta_i^{(t)} + y_i^{(t)} B a_t$ for each task t and each datapoint i; if B is orthogonal this implies that $y_i = \mathrm{sign}(a_t^T B_T (x_i^{(t)} - \eta_i^{(t)}))$ as is standard.



(a) ResNet18 features.



(b) ResNet50 features.

Figure 1: Random feature visualisations for ResNet18 and ResNet50 models pre-trained on ImageNet at a variety of robustness levels.

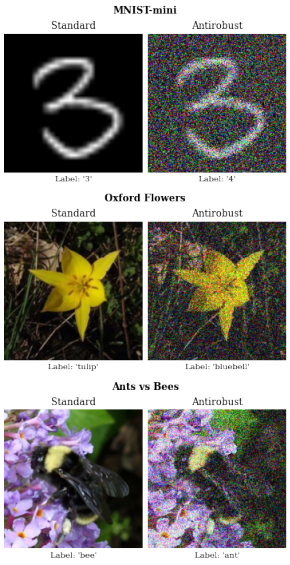


Figure 2: Examples of the 'anti-robust' variants of images in three of the downstream datasets.

4 Our experiments

The tendency of robust nets to learn human-perceptible features leads to the conjecture that these nets are better transfer learners. Indeed, a number of recent concurrent works made this observation; [Sal+20] observe that robust ImageNet models yield improved accuracy on several downstream classification tasks. [Utr+20] note the same behaviour, focusing more on comparing adversarial training techniques, and show that robust models are biased towards recognising shapes over textures (like humans).

In this section we replicate the findings of [Sal+20] on new datasets and we demonstrate that: **(a)** the human-interpretability of learned representations is strongly correlated with model robustness (we show this in finer detail than previous works); **(b)** there exist downstream datasets on which pre-trained model robustness induces *poorer* fine-tuning accuracy; and **(c)** removal of non-robust features from the source dataset may be used to train better transfer learners without adversarial training. Our experiments make use of several standard publicly-available software packages but all of the code used for our experiments is entirely our own.

4.1 Visualising feature representations by robustness level

We take ResNet18 and ResNet50 models [He+16] adversarially pre-trained on ImageNet by [Sal+20] with several different robustness levels and directly visualise their learned representations: for each component i, we choose a Gaussian-randomly generated noise image $\boldsymbol{x}_0$ as a seed and solve the optimization problem

$$\boldsymbol{x}_{\text{vis}} := \text{argmax}_{\boldsymbol{\delta}} R(\boldsymbol{x}_0 + \boldsymbol{\delta})_i$$

using gradient descent started from $\boldsymbol{x}_0$. This gives us the input image $\boldsymbol{x}_{\text{vis}}$ which approximately maximises this representation component. Figure 1 shows this for an arbitrary selection of components, and clearly demonstrates that robustness directly controls the human-interpretability of the representation.

4.2 Anti-robust downstream tasks

In the *'Standard'* plots in Figure 3 we show that pre-trained ImageNet model robustness correlates strongly with fine-tuning accuracy on the CIFAR-10 [Kri09], MNIST [Den12], 17-class Oxford flowers [NZ08], and 'Ants vs Bees' [Dut21] datasets, especially at the beginning of the fine-tuning process (which is equivalent to a data-scarce setting).

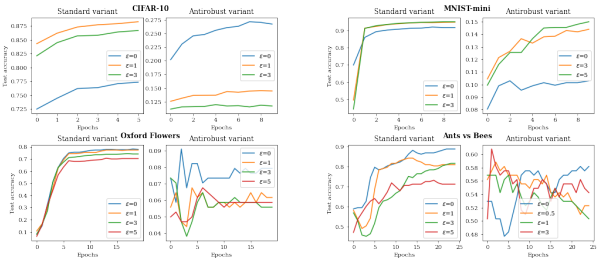
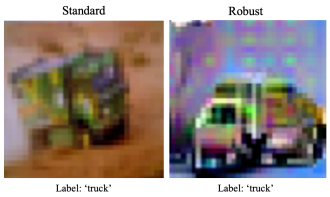
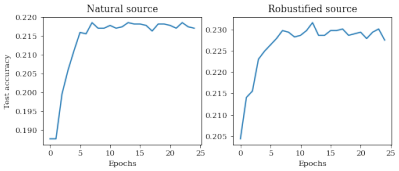


Figure 3: Transfer performance of ResNet18 ImageNet models of different robustnesses on a variety of downstream tasks and their antirobust variants. The pattern of interest is clearly visible on the CIFAR-10 dataset; due to a lack of sufficient time to fine-tune the antirobust dataset generation process the results are less convincing for the other downstream datasets, but we have included them for completeness.



(a) The 'robust' variant of a CIFAR-10 image.



(b) Transfer learning curves for ResNet18 models pre-trained on the standard and robust variants of CIFAR-10 and transferred to the Caltech101 dataset.

Figure 4

We next show that this is not always the case; that there exist downstream tasks where the only useful features are those that are *non-robust*, and therefore for which robust source models are worse transfer learners. Using the technique described in [Ily+19] we construct an *anti-robust* variant of each dataset as follows: we take a standardly-trained deep net C and for each input-label pair $(\boldsymbol{x},y)$ select a *target* class t uniformly at random, and apply a targeted adversarial perturbation to $\boldsymbol{x}$ so that it is mistakenly classified as t by C; that is, we choose

$$\boldsymbol{x}_{\text{adv}} := \text{argmin}_{\|\boldsymbol{x}'-\boldsymbol{x}\|\leqslant\varepsilon}\mathcal{L}(C(\boldsymbol{x}'),t)$$

for some small ε using projected gradient descent. We then re-label this example with t, so that the new training point is $(\boldsymbol{x}_{\text{adv}},t)$. Figure 2 shows a few examples of the training examples from generated anti-robust variants of standard downstream datasets, and the *'Antirobust'* plots in Figure 3 show the fine-tuning behaviour of the pre-trained models on these modified downstream tasks. We see that model robustness no longer improves downstream performance, often actively hindering it.[a]

4.3 Upstream dataset modification

Finally, we briefly explore the use of the dataset modification techniques from [Ily+19] for creating *improved source datasets* on which we can train good transfer learners while avoiding the computational expense of adversarial training. In particular, we take a pre-trained robust classifier C' and for each source datapoint $(\boldsymbol{x},y)$ choosing $\boldsymbol{x}_{\text{robust}}$ to minimise the representation distance $\|R(\boldsymbol{x})-R(\boldsymbol{x}_{\text{robust}})\|_2$ using gradient descent started from a randomly chosen ImageNet seed image $\boldsymbol{x}_0$. Figure 4a shows an example of one of these images and Figure 4b shows the transfer learning curves models trained on the robust and standard versions of CIFAR-10 and fine-tuned on Caltech101 [FFP06]. While the difference is subtle, these results show that source training on a modified dataset can have domain transfer benefits.

5 Discussion

These experiments shine an interesting light on the role of human-interpretable features in deep learning. We have shown that in theory it is possible for learning robust, and therefore human-interpretable, features to actually *hinder* performance on some downstream tasks. This could have far-reaching applications: if we can find more meaningful examples of downstream tasks for which human-perceptible representations fail we may start to develop a theory of what properties of learnt representations *are* desirable other than robustness. One particular application could be to steganalysis, which is a good example of a task that humans find difficult or impossible but neural networks have been shown to solve without difficulty.

Another line of thought these experiments open is the possibility of more advanced data manipulation techniques to create standard 'teaching' datasets that produce better transfer learners for a wider range of downstream tasks than e.g. ImageNet itself. Curriculum design like this could be an important part of training generalised AI further down the line.

[a] Apologies for the small figures; I have tried to ensure the resolution is sufficient that they may be zoomed in on.

References

[Bru+14] Joan Bruna et al. "Intriguing Properties of Neural Networks". In: *International Conference on Learning Representations*. 2014. URL: https://arxiv.org/abs/1312.6199.

[Den+09] Jia Deng et al. "Imagenet: A Large-Scale Hierarchical Image Database". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2009. URL: https://ieeexplore.ieee.org/document/5206848.

[Den+21] Zhun Deng et al. "Adversarial Training Helps Transfer Learning via Better Representations". In: *International Conference on Learning Representations*. 2021. URL: https://proceedings.neurips.cc/paper/2021/file/d3aeec875c479e55d1cdeea161842ec6-Paper.pdf.

[Den12] Li Deng. "The MNIST database of handwritten digit images for machine learning research". In: *IEEE Signal Processing Magazine* 29.6 (2012), pp. 141–142. URL: http://yann.lecun.com/exdb/mnist/.

[Don+14] Jeff Donahue et al. "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition". In: *International Conference on Machine Learning*. 2014. URL: http://proceedings.mlr.press/v32/donahue14.pdf.

[Du+21] Simon Shaolei Du et al. "Few-Shot Learning via Learning the Representation, Provably". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/pdf?id=pW2Q2xLwIMD.

[Dut21] Gaurav Dutta. *Ants & Bees Dataset*. https://www.kaggle.com/gauravduttakiit/ants-bees. 2021.

[Eng+19] Logan Engstrom et al. "Adversarial Robustness as a Prior for Learned Representations". In: *International Conference on Learning Representations*. 2019. URL: https://openreview.net/pdf?id=rygvFyrKwH.

[FFP06] Li Fei-Fei, Rob Fergus, and Pietro Perona. "One-shot learning of object categories". In: *IEEE transactions on pattern analysis and machine intelligence* 28.4 (2006), pp. 594–611. URL: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1597116&casa_token=8xZfnlmrpLEAAAAA:8UBpl6TQFWboYgWONt54C8cCu-8uWqz3-0gvcFtFEw3zNbj1fFP9A17FRCQekYEoP6sG62M&tag=1.

[He+16] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2016. URL: https://ieeexplore.ieee.org/iel7/8272884/8284746/08284852.pdf?casa_token=xmDh6D2lRjQAAAAA:EaonNsBDZVc5hKRJfo1inoGwE9NbaZ1cIcF6n7MABeMzztV0NLFHyxKf8SSA1BctdV5Hyr0.

[Ily+19] Andrew Ilyas et al. "Adversarial Examples Are Not Bugs, They Are Features". In: *Advances in Neural Information Processing Systems*. 2019. URL: https://proceedings.neurips.cc/paper/2019/file/e2c420d928d4bf8ce0ff2ec19b371514-Paper.pdf.

[Kri09] Alex Krizhevsky. "Learning Multiple Layers of Features from Tiny Images". In: (2009). URL: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

[NZ08] Maria-Elena Nilsback and Andrew Zisserman. "Automated Flower Classification over a Large Number of Classes". In: *Indian Conference on Computer Vision, Graphics and Image Processing*. 2008. URL: https://www.robots.ox.ac.uk/~vgg/data/flowers/17/index.html.

[Sal+20] Hadi Salman et al. "Do Adversarially Robust ImageNet Models Transfer Better?" In: *Advances in Neural Information Processing Systems*. 2020. URL: https://proceedings.neurips.cc/paper/2020/file/24357dd085d2c4b1a88a7e0692e60294-Paper.pdf.

[TJJ21] Nilesh Tripuraneni, Chi Jin, and Michael Jordan. "Provable Meta-Learning of Linear Representations". In: *International Conference on Machine Learning*. 2021. URL: http://proceedings.mlr.press/v139/tripuraneni21a/tripuraneni21a.pdf.

[Utr+20] Francisco Utrera et al. "Adversarially-Trained Deep Nets Transfer Better: Illustration on Image Classification". In: *International Conference on Learning Representations*. 2020. URL: https://openreview.net/pdf?id=ijJZbomCJIm.

[Yos+14] Jason Yosinski et al. "How Transferable are Features in Deep Neural Networks?" In: *Advances in Neural Information Processing Systems*. 2014. URL: https://proceedings.neurips.cc/paper/2014/file/375c71349b295fbe2dcdca9206f20a06-Paper.pdf.