INVARIANCE VS. ROBUSTNESS OF NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural networks achieve human-level accuracy on many standard datasets used in image classification. The next step is to achieve better generalization to natural (or non-adversarial) perturbations as well as known pixel-wise adversarial perturbations of inputs. Previous work has studied generalization to natural geometric transformations (e.g., rotations) as *invariance*, and generalization to adversarial perturbations as *robustness*. In this paper, we examine the interplay between *invariance* and *robustness*. We empirically study the following two cases: (a) change in adversarial robustness as we improve only the invariance using equivariant models and training augmentation, (b) change in invariance as we improve only the adversarial robustness using adversarial training. We observe that the rotation invariance of equivariant models (StdCNNs and GCNNs) improves by training augmentation with progressively larger rotations but while doing so, their adversarial robustness does not improve, or worse, it can even drop significantly on datasets such as MNIST. As a plausible explanation for this phenomenon we observe that the average perturbation distance of the test points to the decision boundary decreases as the model learns larger and larger rotations. On the other hand, we take adversarially trained LeNet and ResNet models which have good ℓ_∞ adversarial robustness on MNIST and CIFAR-10, and observe that adversarially training them with progressively larger norms keeps their rotation invariance essentially unchanged. In fact, the difference between test accuracy on unrotated test data and on randomly rotated test data upto $\pm\theta^\circ$, for all $\theta \in [0, 180]$, remains essentially unchanged after adversarial training . As a plausible explanation for the observed phenomenon we show empirically that the principal components of adversarial perturbations and perturbations given by small rotations are nearly orthogonal.

1 INTRODUCTION

Neural networks achieve state of the art, human-level accuracy on several standard datasets used in image classification. However, their performance in the wild depends on how well they can handle natural or non-adversarial transformations of input seen in real-world data as well as known deliberate, adversarial attacks created to fool the model.

Natural or non-adversarial transformations seen in real-world images include translations, rotations, and scaling. Convolutional Neural Networks (CNNs) are translation-invariant or shift-invariant by design. Invariance to other symmetries, and especially rotations, have received much attention recently, e.g., Harmonic Networks (H-Nets) by Worrall et al. (2016), cyclic slicing and pooling by Dieleman et al. (2016), Tranformation-Invariant Pooling (TI-Pooling) by Laptev et al. (2016), Group-equivariant Convolutional Neural Networks (GCNNs) by Cohen & Welling (2016), Steerable CNNs by Cohen & Welling (2017), Deep Rotation Equivariant Networks (DREN) by Li et al. (2017), Rotation Equivariant Vector Field Networks (RotEqNet) by Marcos et al. (2017), Polar Transformer Networks (PTN) by Esteves et al. (2018). For a given symmetry group G, a G-equivariant network learns a representation or feature map at every intermediate layer such that any transformation $g \in G$ applied to an input corresponds to an equivalent transformation of its representations. Any model can improve its invariance to a given group of symmetries through sufficient training augmentation. Equivariant models use efficient weight sharing and require smaller sample complexity to achieve better invariance. Equivariant models such as CNNs and GCNNs too generalize well to progressively larger random rotations but only when their training data is augmented similarly.

Adversarial attacks on neural network models are certain, deliberate changes to inputs that fool a highly accurate model but are unlikely to fool humans. Given any neural network model, Szegedy et al. (2013) show how to change the pixel values of images only slightly so that the change is almost imperceptible to human eye but makes highly accurate models misclassify. Szegedy et al. (2013) find these adversarial pixel-wise perturbations of small magnitude by maximizing the prediction error of a given model using box-constrained L-BFGS. Goodfellow et al. (2015) propose Fast Gradient Sign Method (FGSM) that applies the adversarial perturbation as $x' = x + \epsilon \text{ sign} \left(\nabla_x J(\theta, x, y)\right)$, where x is the input, y represents the target, θ represents the model parameters, and $J(\theta, x, y)$ is the loss function used to train the network. Moreover, Goodfellow et al. (2015) propose adversarial training, or training augmented with points (x', y), as a way to improve adversarial robustness of a model.

Subsequent work has introduced multi-step variants of FGSM, notably, an iterative method by Kurakin et al. (2017) and Projected Gradient Descent (PGD) attack by Madry et al. (2018). Given any model, these attacks produce adversarial perturbation for every test image x from a small ℓ_∞-ball around it, namely, each pixel value x_i is perturbed within $[x_i - \epsilon, x_i + \epsilon]$. PGD attack does so by solving an inner optimization by projected gradient descent over ℓ_∞-ball of radius ϵ around x to approximate the optimal perturbation. Adversarial training with PGD perturbations improves the adversarial robustness of models, and it is one of the best known defenses to make models robust to perturbations of bounded ℓ_∞ norm on MNIST and CIFAR-10 datasets, as shown in Madry et al. (2018) and Athalye et al. (2018).

Recent work has looked at simultaneous robustness to multiple adversarial attacks. Engstrom et al. (2017) show that adversarial training with PGD makes CNNs robust against perturbations of bounded ℓ_∞ norm but an adversarially chosen combination of a small rotation and a translation can still fool these models nevertheless. Recent work of Schott et al. (2019) shows that PGD adversarial training is a good defense against perturbations of bounded ℓ_∞ norm but can be broken with adversarial perturbations of small ℓ_0 or ℓ_2 norm that are also imperceptible to humans or have little semantic meaning for humans. Schott et al. (2019) show how to build models for MNIST dataset that are simultaneously robust to perturbations of small ℓ_0, ℓ_2 and ℓ_∞ norms.

1.1 Our results

In this paper, we study the simultaneous interplay between invariance of models to images randomly rotated by an angle between $[-\theta^\circ, +\theta^\circ]$, and their adversarial robustness to pixel-wise perturbations of ℓ_∞ norm at most ϵ. Unlike previous studies, we do not fix the magnitude ϵ (e.g., say $\epsilon = 0.3$) of pixel-wise perturbations nor limit ourselves to small rotations upto $\pm 30^\circ$. Instead, we compute the accuracy of a given model on input rotated by a random angle between $[-\theta^\circ, +\theta^\circ]$ for θ in the range $[0, 180]$. Similarly, we compute the accuracy of a given model for an adversarial attack of ℓ_∞ norm at most ϵ for ϵ in the range $[0, 1]$ (and the underlying dataset is normalized).

We empirically study the following: (a) change in ℓ_∞ adversarial robustness as we improve only rotation invariance using equivariant models (StdCNNs and GCNNs) and training augmentation with progressively larger rotations, (b) change in invariance as we improve only adversarial robustness using PGD adversarial training with progressively larger ℓ_∞-norm of pixel-wise perturbations.

We observe that the rotation invariance of equivariant models (StdCNNs and GCNNs) improves by training augmentation with progressively larger rotations but while doing so, their adversarial robustness does not improve, or worse, it can even drop significantly on datasets such as MNIST. On the other hand, we take adversarially trained LeNet and ResNet models which have good ℓ_∞ adversarial robustness on MNIST and CIFAR-10, and observe that adversarially training them with progressively larger norms keeps their rotation invariance essentially unchanged. In fact, the difference between test accuracy on unrotated test data and on randomly rotated test data upto $\pm\theta^\circ$, for all $\theta \in [0, 180]$, remains essentially unchanged after adversarial training with progressively larger bounds on the ℓ_∞ norm of pixel-wise perturbations. We support these two observations with plausible empirical reasons for their occurrence - distance to the decision boundary of the test points reducing in the first case, and orthogonality of adversarial perturbations with perturbations caused by small rotations of test points.

Related Work Schott et al. (2019) study simultaneous robustness to adversarial perturbations of small ℓ_0, ℓ_2, and ℓ_∞-norm. Tramèr & Boneh (2019) show an impossibility result by exhibiting data distribution where no model can have substantially better-than-random accuracy for binary classification simultaneously against both ℓ_∞ and ℓ_1 perturbations, and also against both ℓ_∞ perturbations and spatial perturbations, given by an adversarial permutation of coordinates that models an adversarially chosen combination of a small rotation and a small translation. They show an empirical validation of this claim on MNIST and CIFAR-10 datasets for simultaneous robustness against ℓ_∞ adversarial perturbation and an adversarially chosen combination of translation upto ± 3 pixels and rotation upto $\pm 30^\circ$.

2 Rotation invariance vs. ℓ_∞ adversarial robustness

In this section, we present our main result about the interplay between rotation invariance and ℓ_∞ adversarial robustness of models on MNIST, Fashion MNIST, and CIFAR-10 data. In Subsection 2.2, we study how the robustness of equivariant models (StdCNNs and GCNNs) to ℓ_∞ adversarial perturbations changes as they become more rotation invariant through training augmentation with random rotations of progressively larger degree. In Subsection 2.3, we study how the rotation invariance of LeNet and ResNet based architectures change as they become robust to ℓ_∞ adversarial perturbations through PGD adversarial training. The models used in Subsection 2.2 were shown to achieve good rotational invariance with augmentation by Cohen & Welling (2016). The LeNet and ResNet models used in Subsection 2.3 were shown to be adversarially robust by Madry et al. (2018) and reconfirmed in recent work by Athalye et al. (2018).

Before we begin our experiments we first give details of the plots that our experiments will yield.

Rotation invariance means that the predicted labels of an image and any of its rotations should be the same. Since most datasets are centered, we restrict our attention to rotations around the center of the image. One way to quantify rotation invariance is to look at the accuracy of a given model on randomly rotated test data. This is visualized by plotting the accuracy on test inputs rotated by a random angle between $[-\theta^\circ, +\theta^\circ]$, as θ varies in the range $[0, 180]$. We call this the *rotation invariance profile* of the given model.

Adversarial robustness means that the predicted labels of an image and its adversarial pixel-wise perturbation should be the same. Adversarial robustness of a given model to a fixed adversarial attack (e.g., FGSM, PGD) and a fixed ℓ_∞ norm (say, $0 \leq \epsilon \leq 1$) is quantified by the accuracy of the given model on perturbed test data with a perturbation of ℓ_∞ norm at most ϵ generated by the attack. The resulting plot of the accuracy of the model as ϵ varies over $[0, 1]$ is the *robustness profile* of the given model.

Convention used in the legends of our figure: We use the following convention in the legends of our plots. A coloured line labeled A/B indicates that the training data is augmented with random rotations from $[-A^\circ, A^\circ]$ and the test data is augmented with random rotations from $[-B^\circ, B^\circ]$. If A (resp. B) is zero it means the training data (resp. test data) is unrotated.

2.1 Improving rotation invariance

Any model improves its rotation invariance profile when its training data is augmented with progressively larger random rotations. Equivariant models (e.g., GCNNs, H-Nets, PTNs, RotEqNets) are designed to achieve better rotation invariance thought clever weight sharing. However, models such as GCNNs and H-Nets cannot achieve rotation invariance, if their training data is not sufficiently augmented. This appears to be folklore so we do not elaborate on this in the main paper. In Appendix E we confirm this on many popular rotation equivariant models and also show empirically that they become equivariant to rotations by training with a small sample of data augmented with random rotations.

2.2 Effect of rotation invariance on l_∞ adversarial robustness

For any fixed $\theta \in [0, 180]$, we take an equivariant model, namely, StdCNN or GCNN, and augment its training data by random rotations from $[-\theta^\circ, +\theta^\circ]$. Figures 1 and 2 show how the robustness profiles of StdCNN and GCNN models change on MNIST, Fashion MNIST, and CIFAR-10 datasets,

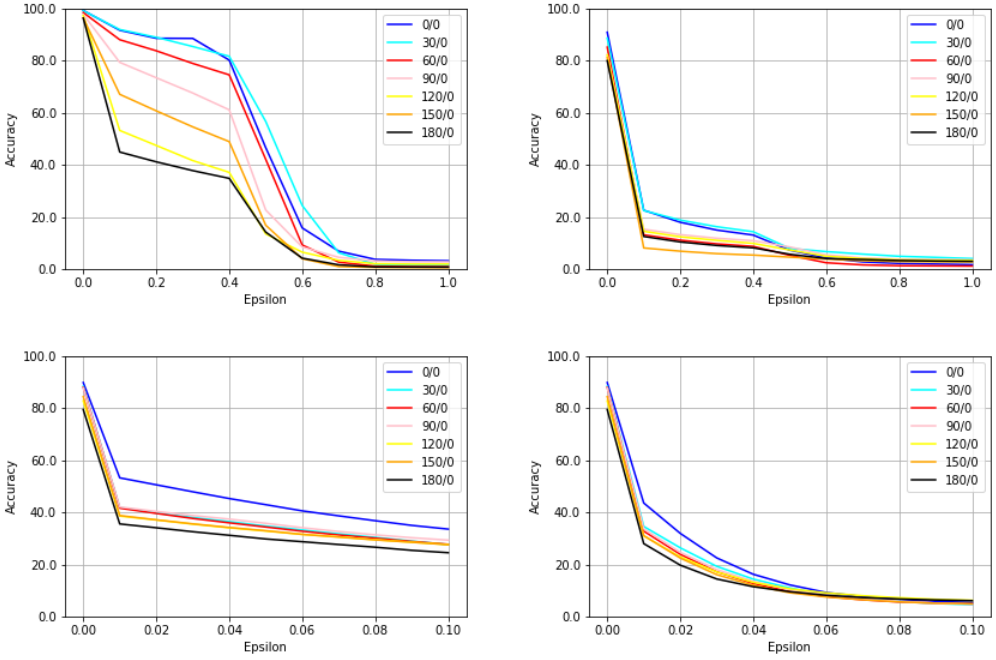

Figure 1: Robustness profiles of StdCNNs - (top left) PGD attack on MNIST, (top right) PGD attack on Fashion MNIST, (bottom left) FGSM attack on CIFAR-10, (bottom right) PGD attack on CIFAR-10.

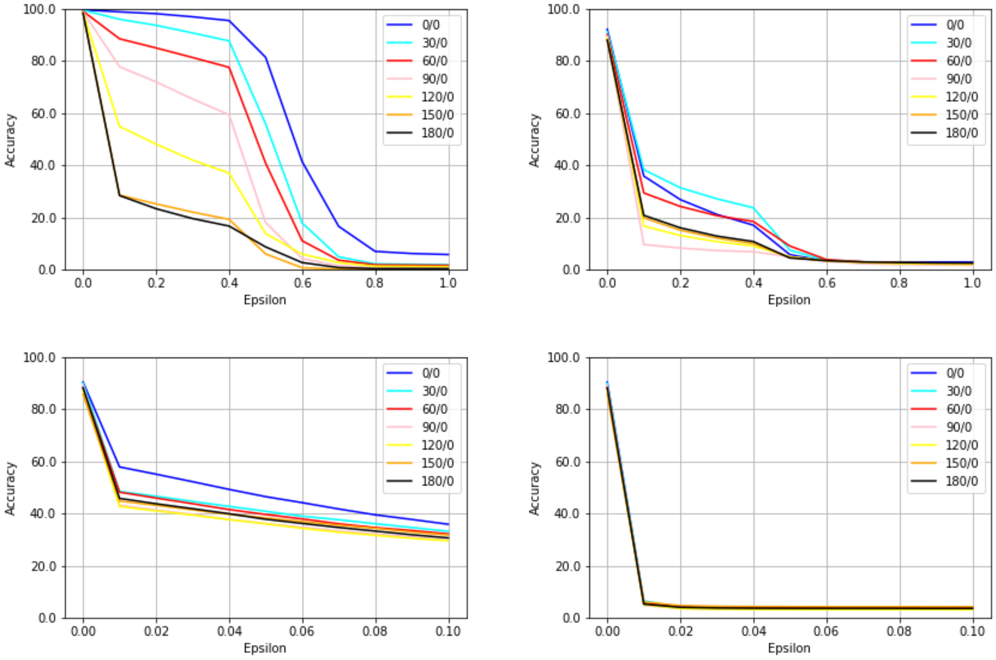

Figure 2: Robustness profiles of GCNNs - (top left) PGD attack on MNIST, (top right) PGD attack on Fashion MNIST, (bottom left) FGSM attack on CIFAR-10, (bottom right) PGD attack on CIFAR-10.

as we increase the degree θ used in training augmentation of the model (thereby improving their rotation invariance). We use PGD adversarial attack for MNIST and Fashion MNIST, FGSM and PGD attack for CIFAR-10 dataset. FGSM is a weaker attack for these models on CIFAR-10. In the above experiments, the test data is unrotated but adversarially perturbed. For completeness, Figures 9, 10 in Appendix B, show the robustness profile of the same networks when the test is also augmented with random rotations before applying adversarial perturbations.

Our experiments show that training augmentation with progressively larger rotations improves the rotation invariance of StdCNNs and GCNNs. For example the red line in Figure 20(bottom) for CIFAR-10 in Appendix E.2 shows that the accuracy of GCNNs drops below 70% when it is not trained with rotations but when the test has random rotations upto $\pm 30°$. However when the training data is augmented with random rotations upto $\pm 30°$ the accuracy improves to about 88% as can be seen from the black line in the same figure.

However training with rotations does not improve their ℓ_∞ adversarial robustness, and can even worsen as in the case of MNIST. On MNIST, the ℓ_∞ adversarial robustness of StdCNNs and GCNNs drops sharply when they are trained with larger rotation augmentations. When GCNN's are trained on MNIST with rotations upto $30°$ (light blue line in Figure 2 (top left)) their accuracy drops to 90% under a PGD attack with $\epsilon = 0.3$, whereas their accuracy is about 97% with the same PGD attack when they are not trained with random rotations (dark blue line in Figure 2 (top left)). We observe the same trend when the test is augmented with rotations, Figure 10 (left) in Appendix B. It is alarming to note that even seemingly innocuous training augmentation with random rotations can affect adversarial robustness of models.

2.3 Effect of ℓ_∞ adversarial training on rotation invariance

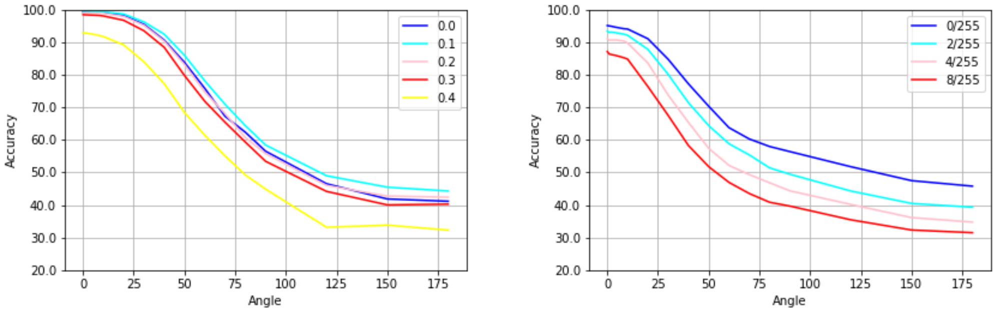

Figure 3: (left) Rotation invariance profiles of PGD adversarially trained LeNet based models from Madry et al. (2018) on MNIST,(right) Rotation invariance profile of PGD adversarially trained ResNet based models from Madry et al. (2018) on CIFAR-10. Different colored lines represent models adversarially trained with different ℓ_∞ budgets $\epsilon \in [0, 1]$.

The most common approach to improve adversarial robustness is adversarial training, i.e., training the model on adversarially perturbed training data. Adversarial training with PGD attack is one of the strongest known defenses on MNIST and CIFAR-10 datasets, see Athalye et al. (2018).

For any fixed $\epsilon \in [0, 1]$ and any fixed $\theta \in [0, 180]$, we adversarially train our models (LeNet and ResNet) with PGD adversarial perturbations with ℓ_∞ budget ϵ, and then plot their rotation invariance profiles. Each colored line in Figure 3 corresponds to a model adversarially trained with a different value of ϵ. Tsipras et al. (2019) point out that the test accuracy on the original data drops after adversarial training, and our plots confirm this - the y co-ordinate of the points where the curves begin decreases with increasing ϵ.

In Figure 3, we observe that the entire invariance profile essentially make a parallel shift downward for LeNet and ResNet based models on MNIST and CIFAR-10 datasets, respectively. In other words, adversarial training does not affect the relative drop in accuracy when test data is augmented with progressively larger rotations.

3 Decision boundary distance and Orthogonality

In order to understand what could be the possible reasons behind our empirical observations in Section 2.2 and Section 2.3 we perform more experiments.

3.1 Average perturbation distance to the boundary.

For each test image, adversarial attacks find perturbations of the test point with small ℓ_∞ norm that would change the prediction of the given model. Most adversarial attacks do so by finding the directions in which the loss function of the model changes the most. In order to explain why these networks become vulnerable to pixel-wise attacks as they learn more rotations, we see how the distance of the test points to the decision boundary changes as the networks learn larger rotations. This is abstractly depicted in Figure 4 where we show the distance of a test point x_0 to the boundary D_0 (resp. D_{180}) when the model is trained with zero (resp. 180°) rotations.

We use the L_2 attack vectors obtained by DeepFool (Moosavi-Dezfooli et al., 2016) for the datapoints under attack. We take the norm of this attack vector as an approximation to the shortest distance of the test point to the decision boundary. For each of the test points we collect the perturbation vectors given by DeepFool attack and report the average perturbation distance. We plot this average distance as the datasets are augmented with larger rotations.

Our experiments show that as the networks learn larger rotations with augmentation, the average perturbation distance falls. So as (symmetric) networks become invariant to rotations, they are more vulnerable to pixel-wise attacks.

The plots in Figures 5, 6, 7, 8 show this for StdCNNs and GCNNs on MNIST and CIFAR-10. Figures 17 and 18 in Appendix C show the plots for Fashion-MNIST. To make our point we plot the accuracy of these networks and also the average perturbation distance of the test points alongside in one figure. The blue line in Figure 5(left) shows the accuracy of a StdCNN on MNIST when both the training data and test data are augmented with θ, as θ ranges from 0 to 180. The green line in Figure 5(left) shows the accuracy of the StdCNN model when the train is augmented with random rotations upto θ, and the test is augmented with rotations upto θ and is also perturbed with PGD of ℓ_∞ norm 0.3. The red line shows the accuracy when the test is not augmented with rotations but is PGD perturbed with ℓ_∞ norm 0.3.

The red line Figure 5(right) shows the average perturbation distance of the unrotated test points when the network is trained with rotations upto θ. The green line shows the average perturbation distance of test points which are augmented with rotations upto θ - this is about 5 when θ is $0°$ (the point on the y-axis where the curves begin). As the network is trained with random rotations up to $180°$ the average perturbation distance of the augmented test drops below 3.5. Figure 5(left) shows that that the PGD accuracy has dropped from around 85% for the network at $0°$ to 30% at $180°$ (the corresponding green line on the left). The fact that the red line is above the green line is also reflected in Figure 11 in Appendix C. When the test is perturbed by PGD, the accuracy of the StdCNN with training data augmented with rotations is better when the test is not augmented with rotations than if the test were also augmented with rotations.

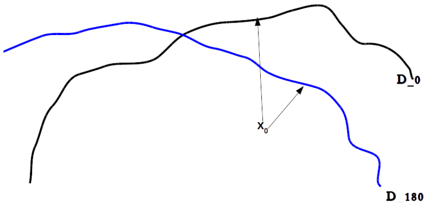

Figure 4: Distance of point x_0 to decision boundary D_{180} obtained by augmenting training set with random rotations in range $[-180°, 180°]$ is different compared to the decision boundary D_0 obtained with no training augmentation.

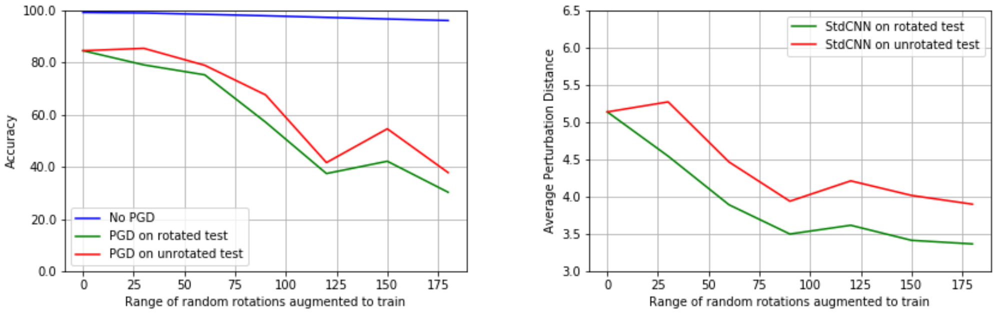

Figure 5: Accuracy of StdCNN on MNIST with/without PGD ($\epsilon = 0.3$), on rotated and unrotated test. Train/test if augmented are with random rotations in $[-\theta°, \theta°]$. (left) Accuracy, (right) Avg. Perturbation Distance.

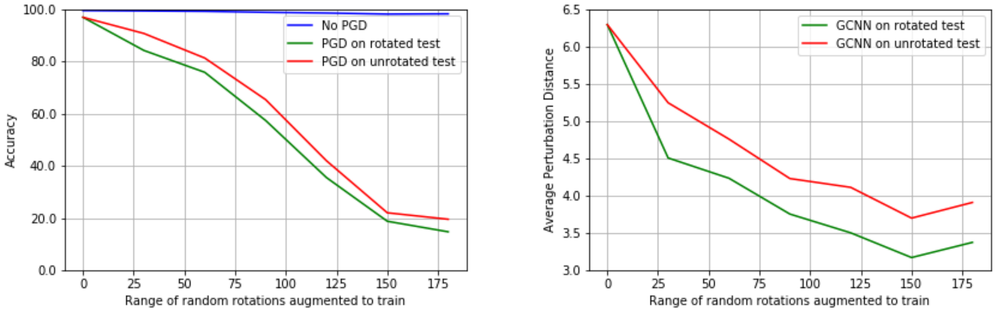

Figure 6: Accuracy of GCNNs on MNIST with/without PGD ($\epsilon = 0.3$) on rotated and unrotated test. Train/test if augmented are with random rotations in $[-\theta°, \theta°]$. (left) Accuracy, (right) Avg. Perturbation Distance.

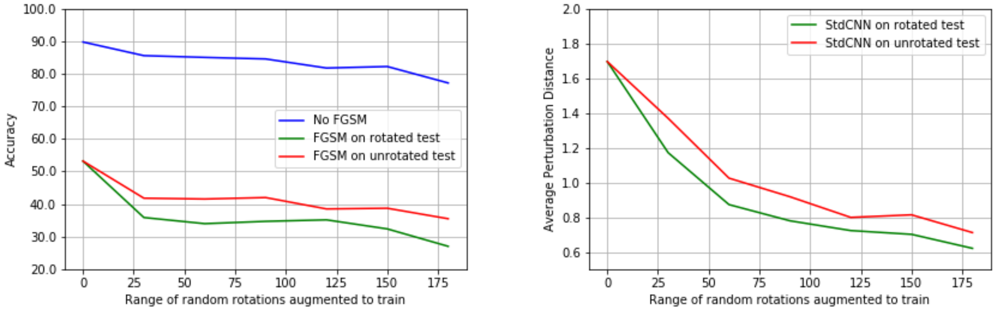

Figure 7: Accuracy of StdCNNs/VGG16 on CIFAR-10, with/without FGSM ($\epsilon = 0.01$) on rotated and unrotated test with $\epsilon = 0.01$. Train/test if augmented are with random rotations in $[-\theta°, \theta°]$. (left) Accuracy, (right) Avg. Perturbation Distance.

3.2 Orthogonality of Adversarial perturbations & invariant rotational perturbations

In general we expect neural networks to classify x and a small rotation x' of x, identically. This would be particularly so if the dataset has some inherent rotations and the labels remain the same under such rotations. Writing $x' = x + \Delta(x)$, we say $\Delta(x)$ is an invariant rotational perturbation

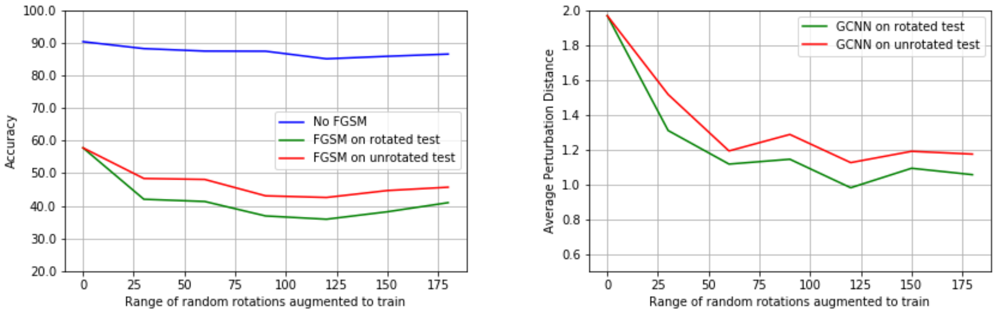

Figure 8: Accuracy of GCNNs/VGG16 on CIFAR-10, with/without FGSM ($\epsilon = 0.01$) on rotated and unrotated test. Train/test if augmented are with random rotations in $[-\theta°, \theta°]$. (left) Accuracy, (right) Avg. Perturbation Distance.

Table 1: MNIST: Principal angles between Top-5 SVD-subspace of PGD attack directions of test points with ℓ_∞ norm ϵ and Top-5 SVD-subspace of the difference directions between a test point and its $4°$ rotation.

ϵ	1	2	3	4	5
0.0	89.48	89.03	86.91	84.79	80.97
0.1	89.93	87.97	86.15	81.62	76.99
0.2	89.56	88.00	85.83	82.29	80.73
0.3	89.75	88.49	87.75	84.96	80.45
0.4	89.33	88.55	86.86	85.80	82.13

Table 2: CIFAR-10: Principal angles between Top-5 SVD-subspace of PGD attack directions of test points with ℓ_∞ norm ϵ and Top-5 SVD-subspace of the difference directions between a test point and its $2°$ rotation.

ϵ	1	2	3	4	5
0/255	89.99	89.40	88.27	86.98	85.99
2/255	89.76	88.76	88.31	87.15	86.33
4/255	89.78	89.02	88.30	86.78	86.41
8/255	89.69	89.13	88.11	87.25	86.42

for x. For a network trained to be equivariant to rotations, this rotational perturbation of x could well be orthogonal to an adversarial perturbation of x, since an adversarial perturbation changes the label whereas an invariant rotational perturbation does not change the label. We empirically show that this is the case for MNIST and CIFAR-10. For each test input find we find its PGD perturbation and stack these adversarial perturbations as rows of an adversarial matrix. We find the top 5 right singular vectors of this adversarial matrix. We rotate each input by $4°/2°$ for MNIST/CIFAR-10 and subtract the original image from it. We stack such invariant rotational perturbations as rows of an invariance matrix. We find the top 5 right singular vectors of this invariance matrix. We empirically observe that the principal angles between these two 5-dimensional subspaces are close to $90°$. This may explain why in Section 2.3 the rotation invariance profile just shifts down. For completeness we recall the definition of principal angles in Appendix D.

Table 1 gives the principal angles of MNIST trained on LeNet and Table 2 gives the principal angles for CIFAR-10 trained on ResNet as given by Madry et al. (2018).

4 Conclusion

We observe that as equivariant models (StdCNNs and GCNNs) are trained with progressively larger rotations their rotation invariance improves but at the cost of their adversarial robustness. On the other hand, adversarial training with perturbations of progressively increasing norms improves the robustness of LeNet and ResNet and also keeps the invariance unchanged. A plausible explanation of the first observation is that the average perturbation distance of the test points to the boundary decreases as StdCNNs and GCNNs learn larger rotations. A plausible explanation for the second observation is that the singular vectors of adversarial perturbations are orthogonal to the singular vectors of invariant rotational perturbations.

References

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, July 2018.

Taco S. Cohen and Max Welling. Group equivariant convolutional networks. *In Proceedings of the International Conference on Machine Learning (ICML)*, 2016.

Taco S. Cohen and Max Welling. Steerable CNNs. *In International Conference on Learning Representations*, 2017.

Sander Dieleman, Jeffrey De Fauw, and Koray Kavukcuoglu. Exploiting cyclic symmetry in convolutional neural networks. *In Proceedings of the International Conference on Machine Learning (ICML)*, 2016.

Logan Engstrom, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. A rotation and a translation suffice: Fooling CNNs with simple transformations. *arXiv preprint arXiv:1712.02779*, 2017.

Carlos Esteves, Christine Allen-Blanchette, Xiaowei Zhou, and Kostas Daniilidis. Polar transformer networks. *In International Conference on Learning Representations*, 2018.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *In International Conference on Learning Representations*, 2015.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2017.

Dmitry Laptev, Nikolay Savinov, Joachim M. Buhmann, and Marc Pollefeys. TI-pooling: transformation-invariant pooling for feature learning in convolutional neural networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 289–297, 2016.

Junying Li, Zichen Yang, Haifeng Liu, and Deng Cai. Deep rotation equivariant network. *arXiv preprint arXiv:1705.08623*, 2017.

Aleksander Madry, Aleksandar A Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *In International Conference on Learning Representations*, 2018.

Diego Marcos, Michele Volpi, Nikos Komodakis, and Devis Tuia. Rotation equivariant vector field networks. *In International Conference on Computer Vision*, 2017.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Lukas Schott, Jonas Rauber, Matthias Bethge, and Wieland Brendel. Towards the first adversarially robust neural network model on MNIST. In *International Conference on Learning Representations*, 2019.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

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

Florian Tramèr and Dan Boneh. Adversarial training and robustness for multiple perturbations. *CoRR*, abs/1904.13000, 2019.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019.

Maurice Weiler, Fred A. Hamprecht, and Martin Storath. Learning steerable filters for rotation equivariant cnns. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow. Harmonic networks: Deep translation and rotation equivariance. *arXiv preprint arXiv:1612.04642*, 2016.

Appendix A Details of experiments

All experiments performed on neural network-based models were done using MNIST, Fashion MNIST and CIFAR-10 datasets with appropriate augmentations applied to the train/validation/test set.

Data sets MNIST dataset consists of $70,000$ images of 28×28 size, divided into 10 classes. $55,000$ used for training, $5,000$ for validation and $10,000$ for testing. Fashion MNIST dataset consists of $70,000$ images of 28×28 size, divided into 10 classes. $55,000$ used for training, $5,000$ for validation and $10,000$ for testing. CIFAR-10 dataset consists of $60,000$ images of 32×32 size, divided into 10 classes. $40,000$ used for training, $10,000$ for validation and $10,000$ for testing.

Model Architectures For the MNIST and Fashion MNIST based experiments we use the network architecture of GCNN as given in Cohen & Welling (2016). The StdCNN architecture is similar to the GCNN except that the operations are as per CNNs. Refer to Table 3 for details. H-Nets architecture is as given in Worrall et al. (2016). RotEqNet architecture is as given in Marcos et al. (2017). PTN architecture is as given in Esteves et al. (2018).

For the CIFAR-10 based experiments we use the VGG16 architecture as given in Simonyan & Zisserman (2014) and its GCNN equivalent is obtained replacing the various layer operations with equivalent GCNN operations as given in Cohen & Welling (2016). This is similar to how we obtained a GCNN architecture from StdCNN for the MNIST based experiments. Input training data was augmented with random cropping and random horizontal flips.

For the adversarial training experiments we used the LeNet based architecture for MNIST and the ResNet architecture for CIFAR-10. Both these models are exactly as given in Madry et al. (2018).

Table 3: Architectures used for the MNIST and Fashion MNIST experiments

Standard CNN	**GCNN**
Conv(10,3,3) + Relu	P4ConvZ2(10,3,3) + Relu
Conv(10,3,3) + Relu	P4ConvP4(10,3,3) + Relu
Max Pooling(2,2)	Group Spatial Max Pooling(2,2)
Conv(20,3,3) + Relu	P4ConvP4(20,3,3) + Relu
Conv(20,3,3) + Relu	P4ConvP4(20,3,3) + Relu
Max Pooling(2,2)	Group Spatial Max Pooling(2,2)
FC(50) + Relu	FC(50) + Relu
Dropout(0.5)	Dropout(0.5)
FC(10) + Softmax	FC(10) + Softmax

Appendix B Effect of invariance on adversarial robustness after test augmentation

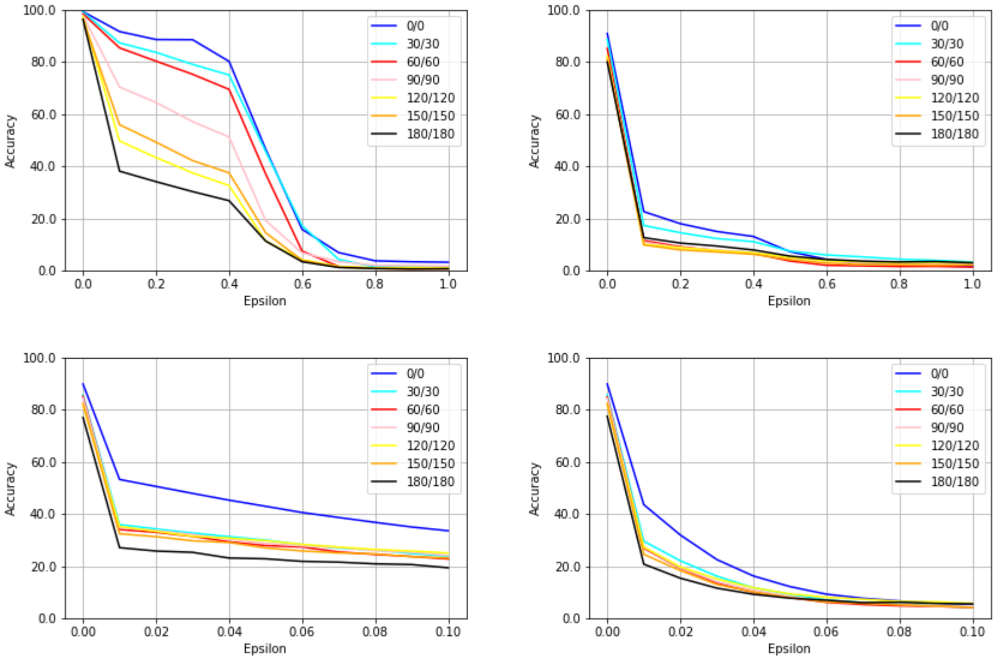

Figure 9: Robustness profile of StdCNN models. (top left) PGD attack on MNIST, (top right) PGD attack on Fashion MNIST, (bottom left) FGSM attack on CIFAR-10, (bottom right) PGD attack on CIFAR-10.

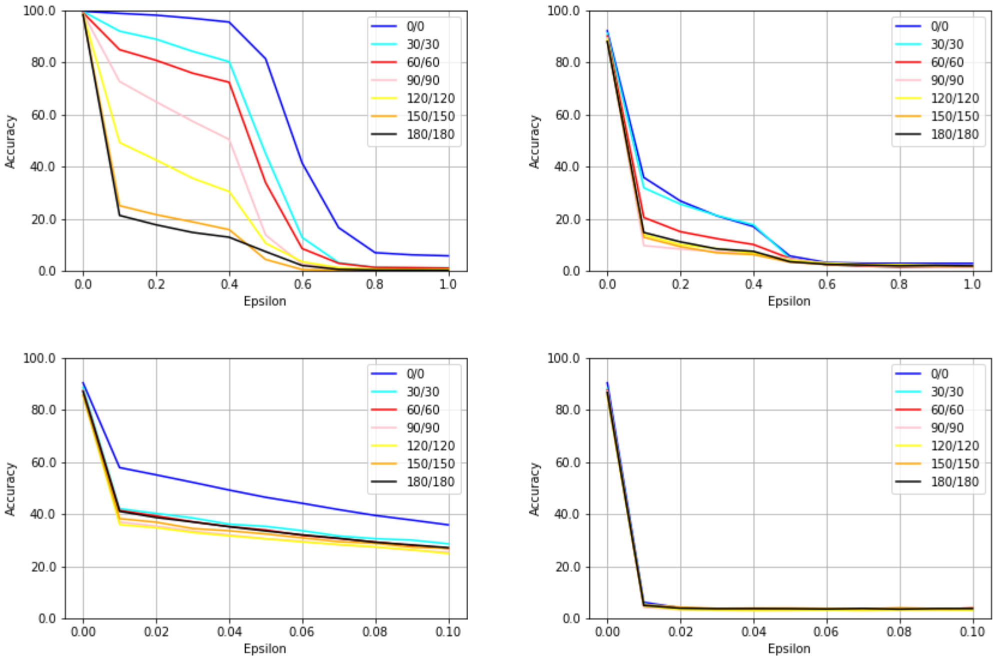

Figure 10: Robustness profile of GCNN models. (top left) PGD attack on MNIST, (top right) PGD attack on Fashion MNIST, (bottom left) FGSM attack on CIFAR-10, (bottom right) FGSM attack on CIFAR-10.

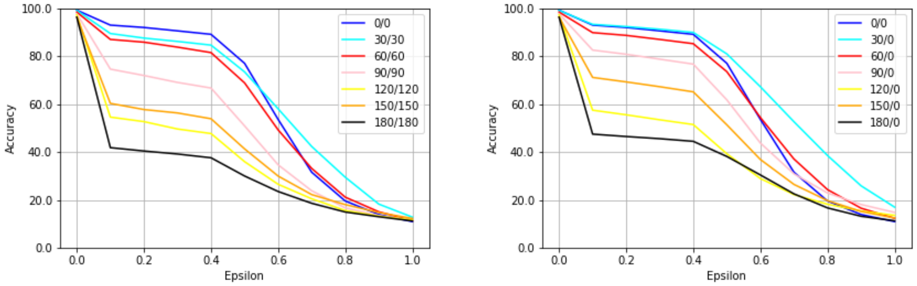

Figure 11: Robustness profile of StdCNN models on MNIST attacked with FGSM. (left) train and test augmented with $[-x^\circ, x^\circ]$ range (right) Only train augmented with $[-x^\circ, x^\circ]$ range and no test augmentation.

APPENDIX C DECISION BOUNDARY DISTANCE, ORTHOGONALITY

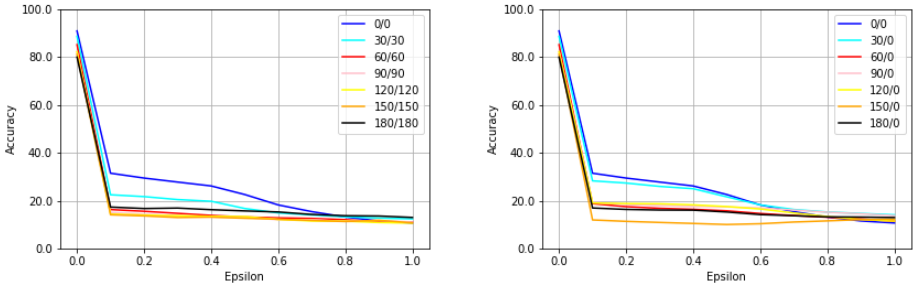

Figure 12: Robustness profile of StdCNN models on Fashion-MNIST attacked with FGSM. (left) train and test augmented with $[-x^\circ, x^\circ]$ range (right) Only train augmented with $[-x^\circ, x^\circ]$ range and no test augmentation.

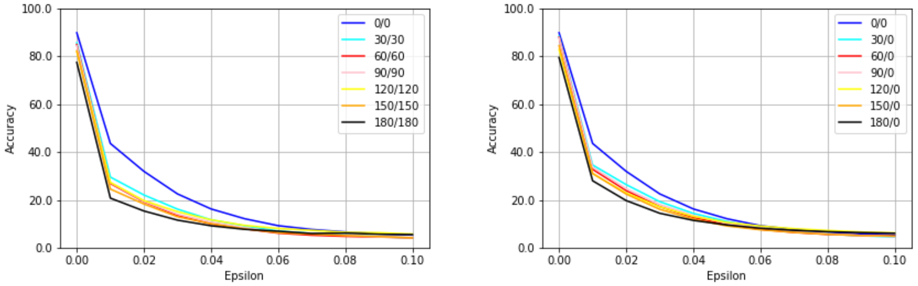

Figure 13: Robustness profile of StdCNN/VGG16 models on CIFAR-10 attacked with PGD. (left) train and test augmented with $[-x^\circ, x^\circ]$ range (right) Only train augmented with $[-x^\circ, x^\circ]$ range and no test augmentation.

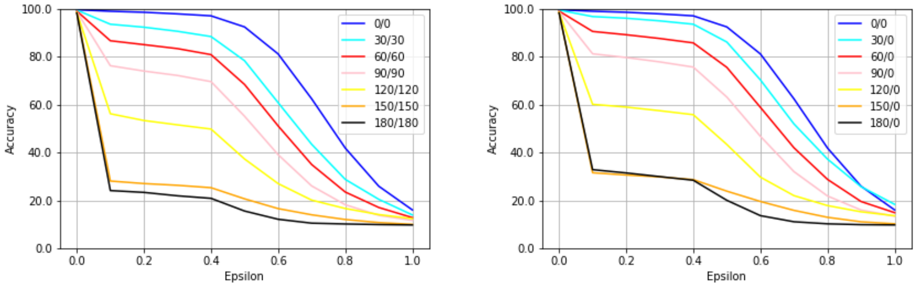

Figure 14: Robustness profile of GCNN models on MNIST attacked with FGSM. (left) train and test augmented with $[-x^\circ, x^\circ]$ range (right) Only train augmented with $[-x^\circ, x^\circ]$ range and no test augmentation.

Appendix D Principal angles

Recall that for two subspaces V,W, the first principal angle is defined as the minimum angle between two unit vectors $v_1 \in V$, $w_1 \in W$. The second principal angle is the minimum angle between unit vectors $v_2 \in V, w_2 \in W$, with $v_2 \perp v_1$ and $w_2 \perp w_1$. The other principal angles are defined similarly.

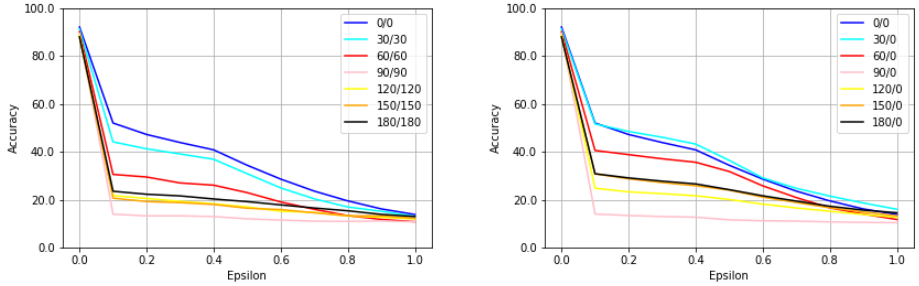

Figure 15: Robustness profile of GCNN models on Fashion-MNIST attacked with FGSM. (left) train and test augmented with $[-x^\circ, x^\circ]$ range (right) Only train augmented with $[-x^\circ, x^\circ]$ range and no test augmentation.

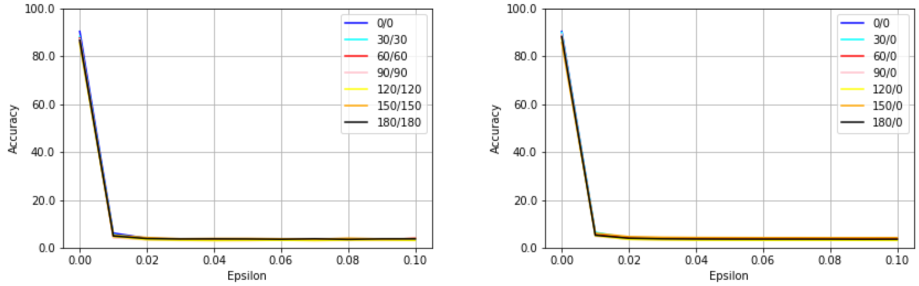

Figure 16: Robustness profile of GCNN/VGG16 models on CIFAR-10 attacked with PGD. (left) train and test augmented with $[-x^\circ, x^\circ]$ range (right) Only train augmented with $[-x^\circ, x^\circ]$ range and no test augmentation.

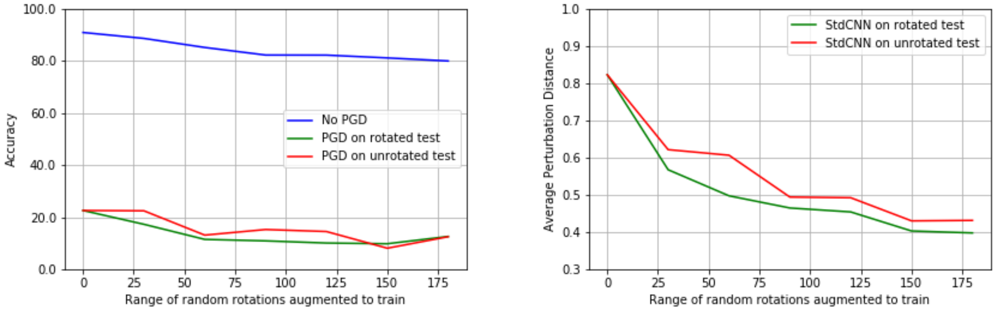

Figure 17: On Fashion MNIST, StdCNNs, Comparison of network with/without PGD on rotated and unrotated test with $\epsilon = 0.1$. Train/test if augmented are with random rotations in $[-x^\circ, x^\circ]$. (left) Accuracy, (right) Avg. Perturbation Distance.

Appendix E Rotation invariance of equivariant networks

We observe that (a) Rotation equivariant networks are robust to only small degrees of rotations away from the ones present in the training data - this was studied only for CNNs in Weiler et al. (2018), (b) applying data augmentation increases their invariance to rotations. (c) Rotation equivariant networks achieve state of the art results with smaller sample size for training.

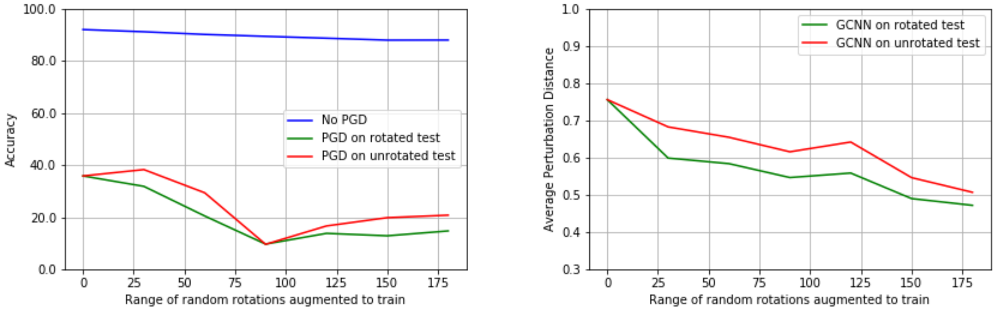

Figure 18: On Fashion MNIST, GCNNs, Comparison of network with/without PGD on rotated and unrotated test with $\epsilon = 0.1$. Train/test if augmented are with random rotations in $[-x°, x°]$. (left) Accuracy, (right) Avg. Perturbation Distance.

E.1 Robustness to rotations without training augmentation

We first train all the networks on MNIST and Fashion MNIST with no rotation augmentation and test against inputs augmented with varying range of random rotations from $\pm 0°$ to $\pm 180°$. Figure 19(top left) for MNIST and Figure 19(top right) for Fashion MNIST shows the performance of these networks to the entire range of rotations. We observe that even though these datasets are small and these networks are designed to be invariant to rotations, their performance drops as the range of random rotations in test data increases. For small range of rotations up to about $\pm 20°$, the performance of these networks on MNIST remains above 95% indicating that MNIST dataset has some natural rotation augmentation. On Fashion MNIST, the performance being above 85% when the test is augmented with small rotations up to $\pm 20°$. When the test is augmented with rotations beyond $25°$, the accuracy of these networks drops. We observe that PTN and RotEqNet are more robust than the other networks for MNIST and Fashion MNIST.

On the dataset CIFAR-10 we compare the performance of rotation-equivariant GCNNs with translation-equivariant StdCNNs. This is shown in Figure 19(bottom). GCNNs perform better, as expected, but the performance of both GCNNs and StdCNNs degrades to less than 70% accuracy when the test data is augmented with random rotations larger than $\pm 30°$.

E.2 Robustness with training augmentation

We train equivariant networks with input augmented with varying range of random rotations from $\pm 0°$ to $\pm 180°$. In Figure 20 we plot our results for MNIST (top left), Fashion MNIST (top right) and CIFAR-10 (bottom), for both StdCNN's and GCNN's. The red(GCNN) and blue(StdCNN) lines indicate networks for which no train augmentation is done. The y coordinate of a point on the black (resp. brown) line corresponding to $\theta°$ on the X-axis (for θ in the range of $[0, 180]$) indicates the accuracy of a GCNN (resp. StdCNN) with training data augmented with random rotations in the range $\pm\theta°$ and with test data also augmented with random rotations in the range $\pm\theta°$. For GCNN's we see a gap of about 4-5% on Fashion MNIST and also CIFAR-10, between the accuracies when trained and tested with no rotation augmentations ($\theta = 0$) and when trained and tested with $\pm 180°$ rotations ($\theta = 180$). It is only on MNIST that rotation-equivariant networks achieve almost the same performance ($\approx 1\%$ difference). For StdCNN's the gap is more pronounced, almost 10-15% on Fashion MNIST and CIFAR-10.

E.3 Sample Complexity of Networks

To understand the sample complexity of the networks, we perform two experiments. In the first we train the networks with varying sample sizes of training set and test them on the entire test set. And in the second experiment we do the same as the first with the inputs in train and test augmented with random rotations in the range $[-180°, 180°]$. From Figure 21 for MNIST, Figure 22 for Fashion MNIST and 23 for CIFAR-10, we can see that rotation equivariant networks achieve their best per-

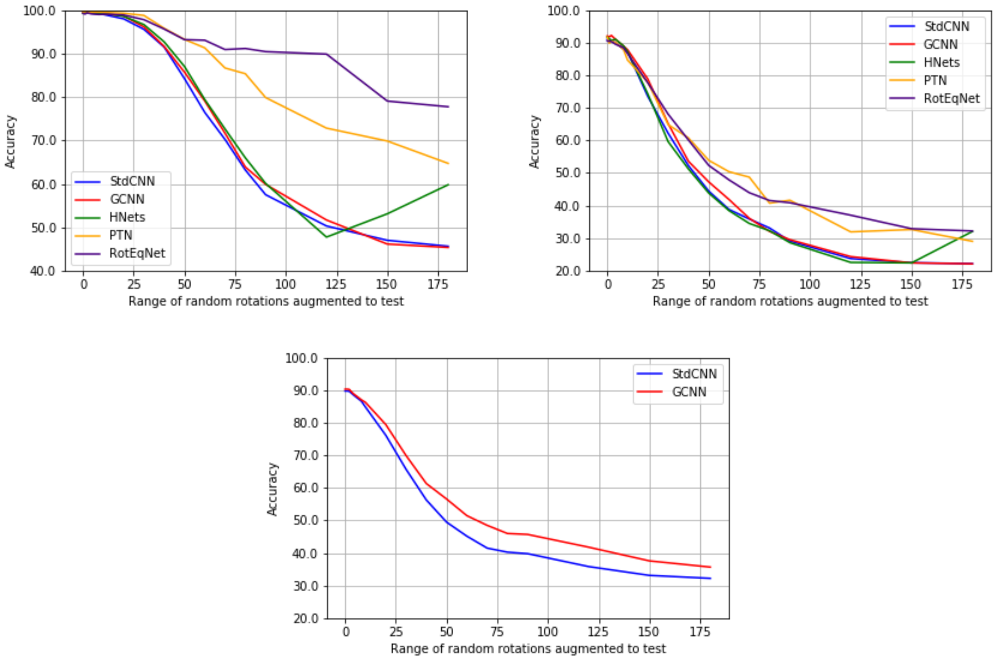

Figure 19: Rotation invariance profile of models with no training augmentation. (top left) MNIST, (top right) Fashion MNIST, (bottom) CIFAR-10 .

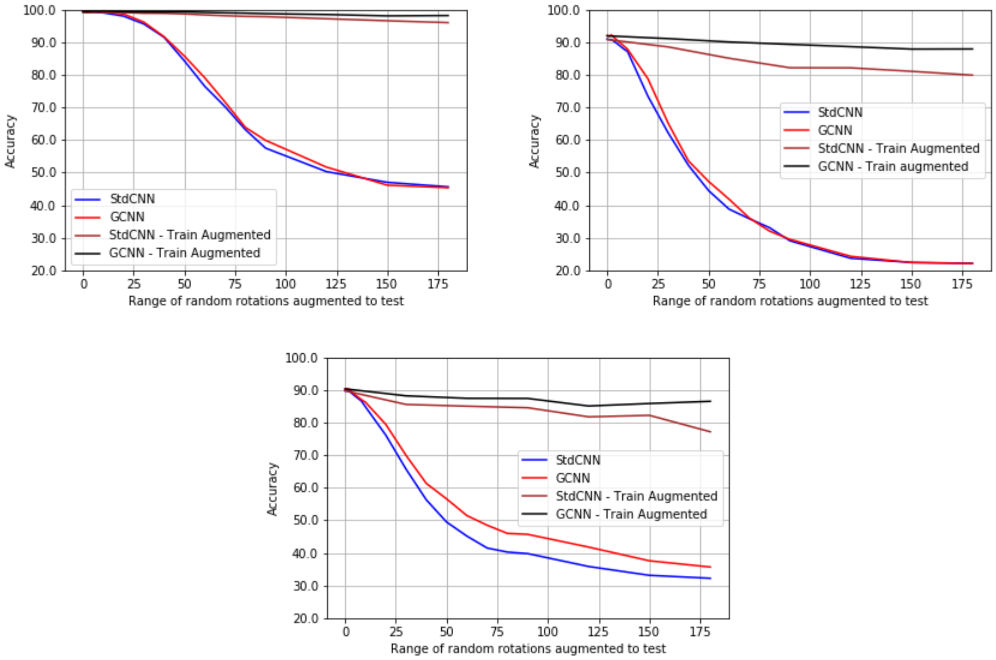

Figure 20: Networks trained with and without augmentation to dataset, random rotation augmentations in $[-x^\circ, x^\circ]$ range. (top left) MNIST (top right) Fashion MNIST (bottom) CIFAR-10.

formance safely using 10*k* - 30*k* training samples. This confirms that rotation equivariant networks can do well with smaller training sample size.

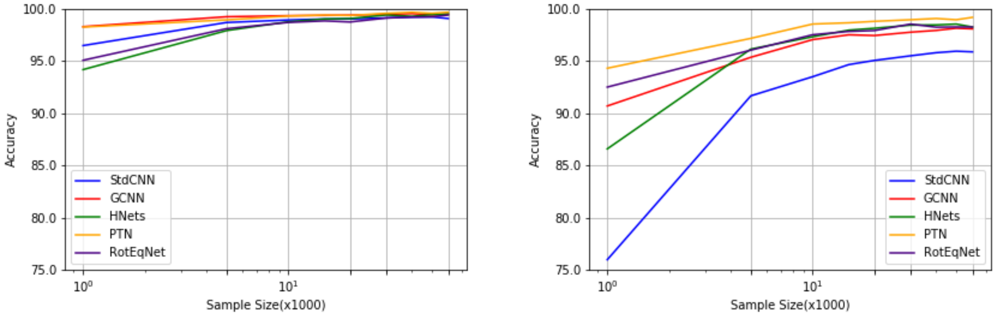

Figure 21: Networks trained with varying training sample size on X-axis. (left) Only MNIST, (right) MNIST train and test augmented with random rotations in $[-180°, 180°]$ range.

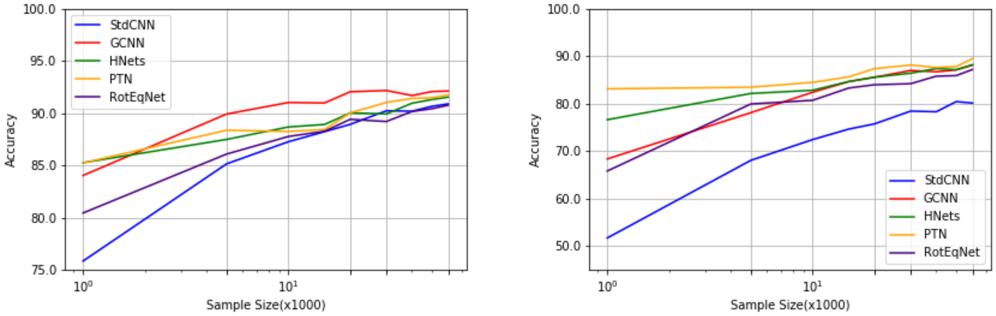

Figure 22: Networks trained with varying training sample size on X-axis. (left) Only Fashion MNIST, (right) Fashion MNIST train and test augmented with random rotations in $[-180°, 180°]$ range.

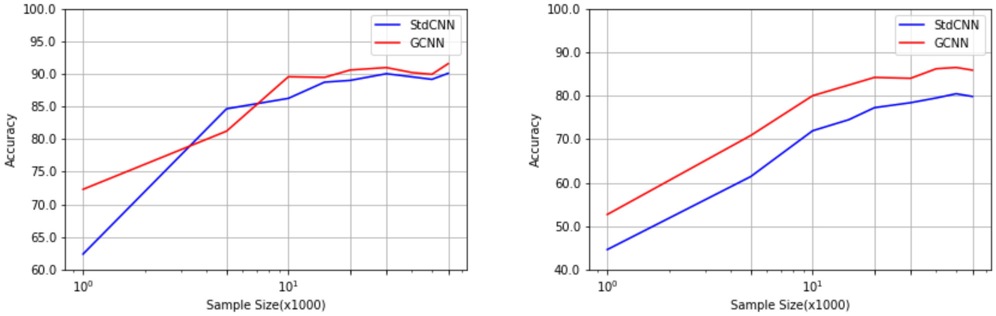

Figure 23: Networks trained with varying training sample size on X-axis. (left) Only CIFAR-10, (right) CIFAR-10 train and test augmented with random rotations in $[-180°, 180°]$ range.