A Game Theoretic Approach to Situating Orthogonal Strategies of Making Deep Neural Network Image Classifiers Adversarially Robust

Word Count: 4514

Abstract

In recent years, deep learning has shown remarkable success in solving classification tasks. However, a major drawback of these deep learning models is their susceptibility to adversarial attacks, leading to incorrect classifications. This study examined different methods of creating classifiers that are resistant to adversarial attacks, focusing on the CIFAR-10 dataset to assess the accuracy of classifications on unaltered, single-step modified, and iteratively modified data. By using a high-level game theoretical framework, each technique was categorized into one of three main natural strategies: altering training data, denoising test data, or smoothing the training algorithm. Empirical testing was conducted to determine whether combinations of methods from these strategies yielded classifiers with higher accuracy. From experimentation, it was determined that non-optimal strategies may benefit greatly from combinations with strategies orthogonal to them; however, near-optimal strategies do not benefit and may even become less optimal when combined with other strategies.

Introduction

In today's world, deep learning classifiers play a pivotal role in automating diverse tasks across numerous industries (Biggio & Roli, 2017; Goodfellow et al., 2018; Kurakin et al., 2016). Their application often involves using supervised learning algorithms to train classifiers on real-world data, frequently leading to vulnerabilities in these classifiers, specifically their susceptibility to adversarial attacks (Goodfellow et al., 2014; Huang et al., 2011; Szegedy et al., 2013). Such adversarial attacks can manifest in two primary forms: poisoning, where adversaries manipulate training data, and evasion, where adversaries craft test data to induce misclassification (Biggio & Roli, 2017; Goodfellow et al., 2018). The latter is the focal point of this paper, particularly in scenarios where adversaries lack access to and cannot alter training data.

The susceptibility of classifiers to adversarial examples poses a significant challenge, particularly in critical domains such as defense, cybersecurity, healthcare, and autonomous vehicles, where misclassifications have the potential to result in severe injuries (Biggio & Roli, 2017; Chakraborty et al., 2021). Consequently, research in adversarial machine learning began, aimed to develop machine learning classifiers that are notably more resilient to adversarial attacks. This undertaking involved the dual objectives of enhancing the understanding of adversarial examples' impact on machine learning models and devising improved defense strategies to counter misclassification induced by adversarial attacks. The overarching goal was to fortify classifiers against adversarial examples, incorporating these insights into the training process to create new training algorithms to bolster the models' robustness (Biggio & Roli, 2017; Goodfellow et al., 2018; Huang et al., 2011).

After nearly a decade of exploration into adversarial machine learning, classifiers persist in remaining vulnerable to adversarial attacks orchestrated by bad actors (Biggio & Roli, 2017; Chakraborty et al., 2021). Despite the emergence of numerous methodologies intended to fortify classifiers, none have proven effective in completely eliminating the threat posed by adversarial examples (Biggio & Roli, 2017). Currently, the overarching objective of ongoing research within the domain of adversarial machine learning is the modification of classifiers to attain minimal susceptibility to adversarial attacks (Biggio & Roli, 2017; Goodfellow et al., 2018).

Game Theoretic Game Design

To model the interaction in adversarial machine learning, the utilization of game theory—a framework at the intersection of computer science, economics, and mathematics for decision-making among multiple agents—has proven effective (Dasgupta & Collins, 2019). This involves first creating representations of the game participants or players. One player is the defender, the classifier creator(s) responsible for ensuring its robustness to adversarial attacks while maintaining resilience to clean data (Dasgupta & Collins, 2019; Robey et al., 2023). The defender's objective is to ensure the classifier correctly classifies data most of the time. The other player, the adversary, introduces adversarial examples into test data to induce misclassification, driven by various objectives such as extracting information embedded during model training or causing the model to fail in classification (Dasgupta & Collins, 2019; Robey et al., 2023). Typically, researchers model the defender's and adversary's intentions as opposing, with the benefits they gain from achieving their desired outcomes considered equal (Dasgupta & Collins, 2019).

In the realm of game design for adversarial machine learning, the interaction between the defender and adversary is conceptualized as a two-player, non-cooperative game, resembling a

competition over a shared resource (Dasgupta & Collins, 2019; Gilmer et al., 2018). The prevalent model for this game is a zero-sum game, where the resource desired by the players (known as utility) lost by one player is equal to the utility gained by the other, making it a simplistic yet calculatable framework for addressing adversarial examples. This zero-sum model allows for the Nash Equilibrium, a technique for calculating player strategies based on rational decision-making, to be determined using the minimax theorem (Dasgupta & Collins, 2019; Gilmer et al., 2018). In this context, the defender seeks to minimize classification error, while the adversary aims to maximize it, revealing a strategy for defenders: to engage in "adversarial training," involving directly training classifiers on adversarial examples alongside clean data, enhancing their robustness by ensuring resilient decision boundaries (Robey et al., 2023).

However, this model may not precisely reflect the real-world dynamics between adversaries and defenders. In adversarial machine learning, a defender's successful avoidance of misclassification does not necessarily equate to the negation of the adversary's utility change, as their goals are not directly opposite to one another (Robey et al., 2023; Zuo et al., 2021). The algorithms used in adversarial training may not align with the attacks that real-world adversaries use, thus detrimentally affecting accuracy in more practical scenarios. An alternative model to the zero-sum game is the two-player Stackelberg game, where a leader selects a strategy, and a follower, aware of this choice, implements their own strategy, mirroring true adversarial machine learning scenarios. However, calculating the Nash Equilibrium in such games lacks the simplicity of the minimax theorem, making it far more difficult to find an equilibrium (Robey et al., 2023).

Adversary Strategies

In practical scenarios, adversaries can be categorized as either black-box or white-box, indicating their understanding of the targeted model (Ren et al., 2020; Sun et al., 2018). White-box adversaries typically possess comprehensive access to the model's training parameters and architecture. On the other hand, black-box adversaries operate with limited access, utilizing an interface solely for inputting data and obtaining output. This paper operates under the assumption that all adversaries are white box, grounded in evidence demonstrating that, in numerous instances, black box adversaries can reconstruct a functional model even with restricted access interfaces (Shokri et al., 2016; Tramèr et al., 2016). Moreover, empirical findings suggest that adversarial attacks crafted for misclassification on one model often translate to misclassification on others (Szegedy et al., 2013). This phenomenon implies that black box adversaries, despite their restricted information, can use similar adversarial attack methods as white box adversaries, challenging the traditional distinction between these categories (Chakraborty et al., 2021).

Given this context, adversaries have two strategies for adversarial attacks: evade via perturbed test data or poison training data (Chakraborty et al., 2021; Goodfellow et al., 2018). Both strategies aim to induce misclassification within a classifier. In the past, to make models robust to adversarial attacks, researchers have attempted to make models robust to one of the above strategies, as it is computationally infeasible to develop a utility function that encompasses both adversarial strategies (Goodfellow et al., 2018). Following this reasoning, this paper focuses on the former strategy, exploring methods to create models robust to adversarial data intended to evade accurate classification.

Defender Strategies

In practical contexts, defenders use three overarching strategies to enhance the robustness of a classifier against adversarial attacks, outlined as follows:

Altering Training Data

This strategy involves altering the training data to enhance the generalization of the resulting trained model. The most prominent method within this category is adversarial training, where adversarial examples, either simulated or derived from real-world scenarios, are incorporated into the training data. This modification allows for the model's decision boundaries to align more effectively with adversarial examples, thereby improving accuracy in classifying such data (Goodfellow et al., 2018; Ruan et al., 2021; Zhao et al., 2022).

Other methodologies include introducing additional "noise" to training data with the aim of enhancing the model generalization (Li et al., 2018). Despite attempts to implement these other methods, the prevailing approach remains the use of adversarial data for training models. This approach is the most favored today as it is perceived to have the highest accuracy, having undergone rigorous testing with various deep learning techniques and expansive datasets, leading to continuous advancements aimed at mitigating potential vulnerabilities arising from the incorporation of adversarial examples into training data (Goodfellow et al., 2018; Ruan et al., 2021).

Denoising Test Data

This strategy involves denoising test data to mitigate adversarial perturbations in data fed into models for classification. Denoising test data is the predominant approach, aimed at "removing" or "decreasing" adversarial perturbations, ultimately enhancing the model's resilience to adversarial perturbations, or "noise" (Sahay et al., 2018; Salman et al., 2020). This method has not demonstrated the resilience exhibited by classifiers trained using the other

prominent strategies involving altering the training data or the training algorithm. Despite this, approaches involving test data denoising have shown moderate classification accuracy (Sahay et al., 2018), as they effectively eliminate substantial adversarial noise.

Modify Training Algorithm

This strategy entails adjusting the machine learning model during training to make it more "smooth," thereby enhancing generalization and reducing susceptibility to "small" adversarial perturbations, where "small" perturbations refer to those imperceptible to humans, thus aligning the model's behavior more closely with human expectations (Carlini & Wagner, 2016a; Carlini & Wagner, 2016b; Papernot et al., 2015). The predominant approach within this strategy is known as Defensive Distillation, involving the training of a "student" model from the probability distribution outputted by a "teacher" model. The resultant "student" model incorporates decision-making functions that are highly resistant to minor perturbations, and generalize well to most types of adversarial noise (Papernot et al., 2015). While not as extensively pursued as the strategies of altering training data or denoising test data, results from the methods developed thus far within this strategy have been highly effective, with adversarial robustness reaching levels comparable to many prominent methods within the two other strategies (Carlini & Wagner, 2016b; Papernot et al., 2015).

Research Context

While various algorithms exist for implementing each strategy, this paper focuses on the most prominent ones within each strategy's scope, as these algorithms exhibit high robustness in diverse situations. Note that despite the existence of the Nash Equilibrium ensuring optimal strategies for accurate classification (Chakraborty et al., 2021), each mentioned strategy serves as an approximation, and as such, there is no singular "optimal" strategy universally guaranteeing

higher classification accuracy. Additionally, there is no metric available to determine an optimal strategy, as different situations where classification is needed will require different metrics to be maximized.

Research Objective

This paper focuses on concurrently using orthogonal methods to increase the adversarial robustness of machine learning classifiers on convolutional neural networks (CNNs), a type of machine learning algorithm used for image classification. This is done to develop methods of creating robust algorithms using these combined strategies that are more effective than the individual strategies themselves. To this end, the following question is proposed: how might a deep learning image classifier be made more robust to both clean data and adversarial examples using concurrent adversarial robustness strategies?

Regarding image classification, effective adversarial attacks involve perturbations measurable by the L0 norm (a metric for magnitude referring to the number of modified features in the present adversarial example), L2 norm (a metric for magnitude referring to the adversarial perturbation's Euclidean distance), or L^∞ norm (a metric for magnitude referring to the absolute value of adversarial perturbation's maximum vector component), with L^∞ being the most studied because of its ease-of-use in optimization algorithms (Carlini & Wagner, 2016b; Ren et al., 2020), so this study will utilize the L^∞ norm to measure the magnitude of modification within an adversarial example. Though this paper focuses on making

CNNs robust to evasion, the phenomenon known as model transferability ensures that the discussed strategies can also be applied to other machine learning classifiers, not solely CNNs, as the same types of adversarial attacks tend to work on all machine learning classifiers (Goodfellow et al., 2018).

Methodology

Experiment Objective

This study considers the orthogonality of various natural strategies of increasing adversarial robustness for image classification tasks. Specifically, the duality between adversarial robustness and clean data accuracy is considered. This research seeks to bridge the gap regarding the lack of understanding of the orthogonality of various adversarial machine learning techniques and introduces a method for comparing and quantifying the orthogonality of various techniques to enhance the robustness of image classifiers against adversarial attacks. This involves categorizing these techniques into broad high level "strategies". The proposed framework is then utilized to merge methods from these distinct "strategies" to improve adversarial robustness, with each combination evaluated based on two key metrics: adversarial robustness and the accuracy of classifying clean data. This approach aims to determine the effectiveness of the combinations used and identify the scenarios where certain strategies may be more effective in categorizing data for image classification than others.

The primary hypothesis is that distinct natural categories make models robust in orthogonal ways that are additive, allowing for greater robustness when combined. There is an expected tradeoff between overall classification accuracy and adversarial robustness, and thus both metrics will be evaluated during experimentation.

Assessed Classifiers

Training Dataset

The experimentation is performed using the CIFAR-10 dataset, a widely recognized benchmark for evaluating the robustness of image classification models. This dataset comprises 60,000 32x32-pixel color images of real-world animals and objects, evenly distributed across 10 classes, and is pre-partitioned into training and testing subsets (Krizhevsky, 2009). Due to model transferability (Goodfellow et al., 2018), this study basing its findings on one dataset is not problematic; the findings indicated in this study are extremely likely to be present in machine learning models trained on most other image datasets, especially when used in image classification.

Adversarial Robustness Methods

The approaches discussed earlier, which include altering training data, denoising test data, and developing smoothed classifiers each have a range of algorithms designed to implement them. In this study, models generated using single methods and combinations of two methods are evaluated and compared against each other in terms of their clean and adversarial data classification accuracies. The methods (listed as "Overarching Strategy":"Method") are detailed below:

Altering Training Data: FGSM Adversarial Training. This technique involves model training using both clean and adversarial data, the latter created with Fast Gradient Sign Method (FGSM). FGSM introduces minor distortions to clean training data to enhance model resilience against adversarial attacks (Goodfellow et al., 2014; Ren et al., 2020). The perturbation applied, measured by L^{∞}, is set at 0.1.

Altering Training Data: PGD Adversarial Training. This technique involves model training using both clean and adversarial data, the latter created with Projected Gradient Descent (PGD). PGD iteratively applies FGSM (but with smaller magnitude perturbations) to clean training data to enhance model resilience against adversarial attacks (Kurakin et al., 2016a; Ren et al., 2020). The perturbation, measured by L^{∞}, is set at 0.01, and is iteratively applied 10 times.

Denoising Test Data: FGSM Adversarial Noise Reduction. To assess the efficacy of defense tactics that use denoising, an autoencoder is trained using both adversarial examples generated by FGSM and clean data to remove adversarial noise from perturbed data. This method seeks to reconstruct original data from training data that been altered by FGSM (Sahay et al., 2018).

Denoising Test Data: PGD Adversarial Noise Reduction. To assess the efficacy of defense tactics that use denoising, an autoencoder is trained using both adversarial examples generated by PGD and clean data to remove adversarial noise from perturbed data. This method seeks to reconstruct original data from training data that been altered by PGD (Sahay et al., 2018).

Training Algorithm Smoothing: Defensive Distillation. Defensive distillation involves training a “student” classifier using "soft labels" (the classification probabilities outputted by another “teacher” model) (Carlini & Wagner, 2016a; Papernot et al., 2015). These “soft labels” help the final classifier develops smoother decision boundaries, reducing the effects of adversarial attacks.

Training Algorithm Smoothing: Random Self-Ensemble. This method combines randomness and ensemble learning to enhance resilience against adversarial attacks (Liu et al., 2017). By adding random noise layers (layers of Gaussian Noise with magnitude 0.1 are used in this study) to the network and ensembling predictions over these noises, RSE creates a robust and resilient model that can withstand adversarial attacks effectively.

CNN Architecture

A standard CNN architecture, designed to extract, process, and identify relevant features from images, is used in this study to allow for comparable results in image classification tasks. Four convolutional layers (needed to identify key patterns and features in images) equipped with 3x3 filters and using L2 regularization to lower the likelihood of overfitting are used, each followed by a Max Pooling layer and then a Dropout layer for downsampling. Following the final Max Pooling layer, a one-dimensional array is created from the feature maps via a Flatten layer, and this array is passed into two fully connected (dense) layers to further process the features (and each also containing L2 regularization and followed by a Dropout layer to prevent overfitting). Finally, an output layer consisting of 10 neurons (matching the number of labels in CIFAR-10) outputs the predicted classification probabilities for each label. All non-distillation models use a softmax activation function in their output layers to create class probabilities.

The models are assembled utilizing the Adam optimizer, configured with a learning rate of 0.0003 and the categorical cross-entropy loss function (highly accurate for multi-class classification problems). In the distilled models, the softmax cross-entropy loss function is used, with a temperature parameter set to 20, as these models do not include a softmax activation function in their output layer. All models undergo training on the same dataset, or, if adversarial

training is applied, they are trained on adversarially perturbed data derived from the same dataset.

Experimental Procedure

Twenty-two models are evaluated based on their classification accuracies: a model solely trained on clean data, six models trained using single individual methods, twelve models trained using combinations of two methods from distinct categories, and three models trained using combinations of two methods from the same category. Each model trained with Defensive Distillation, in this case, is a model trained using the standard model architecture outlined above, except both “student” and “teacher” models have a temperature parameter of 20.

This research did not use optimal hyperparameters for training and instead used a standard model architecture with all parameters being shared among models. The effectiveness of each technique could be further enhanced, but given that the model architectures used are identical, such a distinction will not change the conclusion regarding the accuracy gained by training classifiers on orthogonal methods of increasing classification robustness. For orthogonal method combinations to be deemed more effective, the findings should indicate higher adversarial data classification accuracy without tradeoffs involving a large degradation in clean data classification accuracy.

The following procedure is conducted to assess the resilience of classifiers in image classification tasks on the CIFAR-10 dataset. It measures the model's effectiveness in classifying images through its accuracy on predefined test datasets, which include clean data, FGSM-perturbed data, and PGD-perturbed data. The experiment is divided into three parts:

1. Clean Data

This initial phase evaluates the models' classification accuracy on a predefined, shared dataset of clean images. CIFAR-10 is already divided into training and test sets, which are used for model training and testing, respectively.

2. FGSM-Perturbed Data

In the second phase, the models' performance in classifying adversarial data is assessed. This adversarial test data is created by adding a single perturbation (of magnitude 0.1, measured by the L^∞ norm) to each image in the clean test data using the FGSM adversarial perturbation technique, similar to the adversarial examples generated during adversarial training.

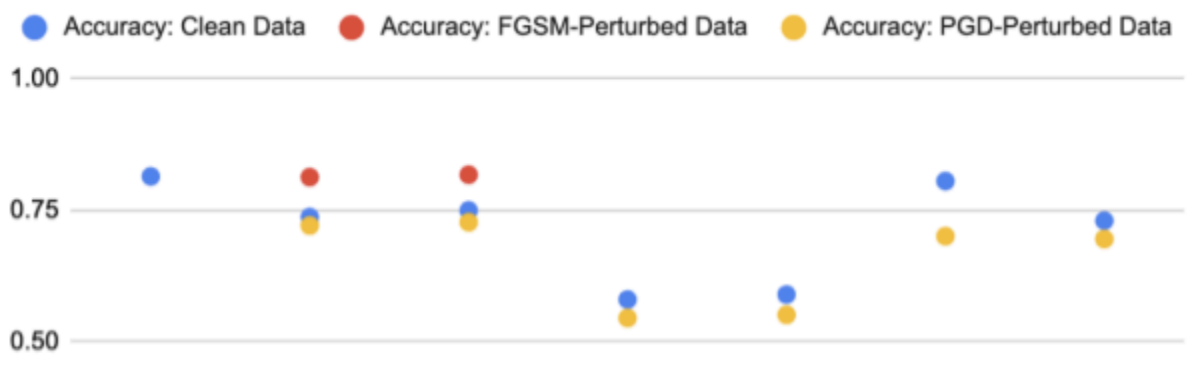
3. PGD-Perturbed Data

The third phase evaluates the models' performance on adversarial data generated by adding 10 iterative perturbations of magnitude 0.01, measured by the L^∞ norm via the PGD adversarial perturbation technique. This is also comparable to the generation of adversarial examples during adversarial training.

Results and Discussion

Experimental Results (Single Strategy Models)

Dataset: CIFAR-10



Unprecedented Robustness to PGD-Perturbed Adversarial Examples

During the experiment, there were significant differences in the expected and actual accuracies of the models when classifying PGD-generated adversarial test data. All classifiers showed greater resilience to PGD-perturbed test data compared to FGSM-perturbed test data.

Two main reasons likely explain this discrepancy in classification accuracy. First, the magnitude of FGSM perturbation was a single, fixed magnitude of 0.1, whereas the PGD perturbation contained 10 iterative fixed magnitude 0.01 perturbations. This means the total PGD perturbation magnitude was less than or equal to 0.1 (10 * 0.01), making PGD adversarial examples less likely to cause misclassifications in classifiers compared to FGSM adversarial examples. Second, the CIFAR-10 dataset contains images of 32x32x3 pixel dimensions, affecting the visibility of perturbations. Since the training dataset is of low resolution, FGSM-generated adversarial examples, which contain a single large perturbation, are more visible than PGD-generated adversarial examples, which are spread over multiple iterations.

Evaluating the Situated Singular Strategies

High Accuracy of Adversarial Training

Altering training data through adversarial training significantly enhances adversarial resilience, surpassing any other method tested during the experiment. However, it does not generalize as effectively as Defensive Distillation. While adversarial training retains a considerable portion of the clean data classification accuracy, it doesn't achieve this to the same extent as the unaltered classifier. The decline in classification accuracy suggests that the adversarial examples led the models to overfit adversarial example features, distorting model decision boundaries in an undesirable manner. This results in increased adversarial robustness

but not overall robustness in classification, leading to a decrease in clean data classification accuracy.

High PGD Adversarial Training Accuracy

Although the PGD perturbations likely had a much smaller L^∞ magnitude compared to FGSM perturbations, the PGD adversarially trained classifier exhibited adversarial robustness comparable to the FGSM adversarially trained classifier across all test data types. Unlike autoencoders, the adversarially trained models did not significantly decrease in accuracy during clean data classification, but still showed minor classification accuracy detriments. These decreases were roughly equivalent between PGD and FGSM adversarially trained models, suggesting that adversarial training adjusted the classifiers' decision boundaries to improve their ability to generalize to adversarial data, ensuring their robustness against adversarial attacks but simultaneously becoming less effective at clean data classification.

Overall Low Accuracy of Adversarial Noise Reduction

The approach of denoising test data to enhance adversarial robustness of models has potential, but struggles to generalize to the test data types. All experimental autoencoders, including those trained on PGD adversarial data, showed improved accuracy on FGSM and PGD adversarial data but reduced robustness against clean data. While autoencoders were capable of identifying and partially denoising the significant perturbations in FGSM-perturbed and

PGD-perturbed data, the data suggests that they also denoise relevant features rather than just the adversarial noise. Although denoising input data through autoencoders trained to remove adversarial noise can moderately improve the classification of adversarial data, it fails to sufficiently maintain classification robustness to ensure reliable and accurate denoising of input data.

High Generalization of Classifier Smoothing

The experiment also evaluated the strategy of altering classifiers through "smoothing." Defensive Distillation maintained clean data classification accuracy and exhibited high accuracy on PGD adversarial data. It had moderate accuracy classifying FGSM adversarial data, but not as much as when adversarial training or denoising test data. This indicates that Defensive Distillation can offer high adversarial robustness for data with minor feature perturbations, but is not as effective for adversarial data with large-magnitude perturbations. Random Self Ensemble performed at a similar accuracy as Defensive Distillation in classifying PGD-perturbed data, but displayed moderately higher accuracy in classifying FGSM-perturbed data and slightly lower accuracy in classifying clean data, indicating that like Defensive Distillation, Random Self Ensemble generalizes well, but perhaps smoothes the decision boundaries more than Defensive Distillation, thus degrading identification of relevant features in data further than Defensive Distillation.

Evaluating the Situated Combined Strategies

Dataset: CIFAR-10

# Adversarial Robustness Strategies Used	Adversarial Robustness Strategies Used	Clean Data Accuracy	FGSM-Perturbed Data Accuracy	PGD-Perturbed Data Accuracy
2 (Same Category)	(FGSM + PGD) Adversarially Trained Classifier	0.7221	0.8806	0.7158

2 (Same Category)	(FGSM + PGD) Adversarial Noise Reduction	0.5222	0.2670	0.4983
2 (Same Category)	Defensive Distillation + Random Self Ensemble	0.7336	0.2845	0.6930
2 (Different Categories)	(FGSM) Adversarially Trained Classifier + (FGSM) Adversarial Noise Reduction	0.6272	0.4604	0.6159
2 (Different Categories)	(FGSM) Adversarially Trained Classifier + (PGD) Adversarial Noise Reduction	0.6329	0.4734	0.6200
2 (Different Categories)	(FGSM) Adversarially Trained Classifier + Defensive Distillation	0.7464	0.7881	0.7314
2 (Different Categories)	(FGSM) Adversarially Trained Classifier + Random Self Ensemble	0.6665	0.6588	0.6653
2 (Different Categories)	(PGD) Adversarially Trained Classifier + (FGSM) Adversarial Noise Reduction	0.6410	0.4688	0.6241
2 (Different Categories)	(PGD) Adversarially Trained Classifier + (PGD) Adversarial Noise Reduction	0.6483	0.4771	0.6317
2 (Different Categories)	(PGD) Adversarially Trained Classifier + Defensive Distillation	0.7316	0.7656	0.7231
2 (Different Categories)	(PGD) Adversarially Trained Classifier + Random Self Ensemble	0.6831	0.6638	0.6802
2 (Different Categories)	(FGSM) Adversarial Noise Reduction + Defensive Distillation	0.5625	0.3159	0.5398
2 (Different Categories)	(FGSM) Adversarial Noise Reduction + Random Self Ensemble	0.6002	0.4383	0.5855
2 (Different Categories)	(PGD) Adversarial Noise Reduction + Defensive Distillation	0.5861	0.3178	0.5598
2 (Different Categories)	(PGD) Adversarial Noise Reduction + Random Self Ensemble	0.6045	0.4395	0.5888

Relative to Altering Training Data

Fig. 2: Classification Accuracies for Combined Methods Using FGSM Adversarial Training

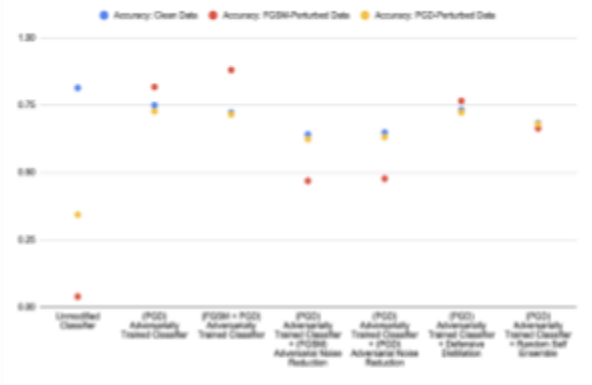


Fig. 3: Classification Accuracies for Combined Methods Using PGD Adversarial Training

As seen in the experimental data, the strategy of altering training data via adversarial training works optimally without being combined with an orthogonal strategy, suggesting that rather than being truly orthogonal to the other strategies, it is an exceedingly accurate approximation of the Nash Equilibrium, and thus, any combination of adversarial training with a different strategy decreases the accuracy of the approximation, thus decreasing robustness.

Relative to Denoising Test Data

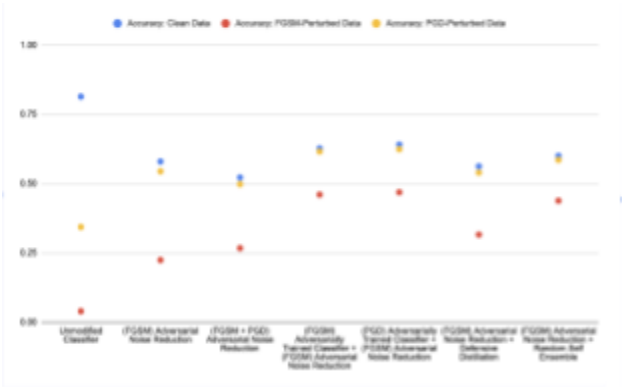


Fig. 4: Classification Accuracies for Combined Methods Using FGSM Adversarial Noise Reduction

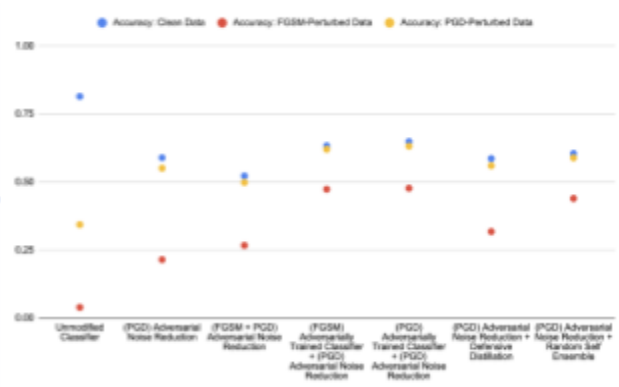


Fig. 5: Classification Accuracies for Combined Methods Using PGD Adversarial Noise Reduction

As seen in the experimental data, the strategy of denoising training data via an autoencoder was not optimal, and thus benefited from being combined with orthogonal

strategies, suggesting that when optimal strategies cannot be used for certain applications, orthogonal combinations of non-optimal strategies may be used as a suitable alternative.

Relative to Smoothing the Training Algorithm

Fig. 6: Classification Accuracies for Combined Methods Using Defensive Distillation

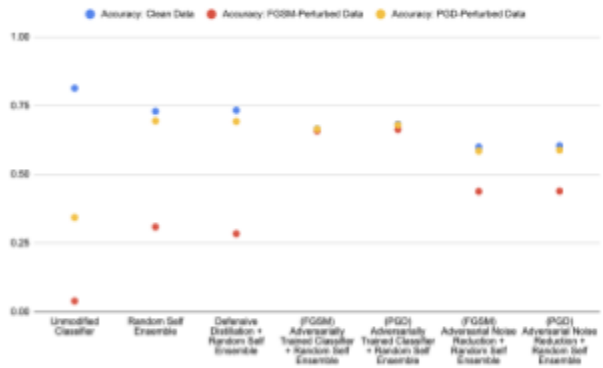


Fig. 7: Classification Accuracies for Combined Methods Using Random Self Ensemble

As seen in the experimental data, both methods within the strategy of smoothing the training algorithm benefited from being combined with a more optimal strategy, altering training data, but had mixed results for the different types of test data when combined with a less optimal strategy, denoising test data. This suggests that attempts to combine two strategies with near-optimal results on different types of test data may net a model with high—but not near-optimal—accuracy on those particular test data types.

Conclusion

The experimentation found that altering training data significantly boosted adversarial robustness, albeit at the expense of reduced accuracy on clean data. In contrast, denoising test data using autoencoders showed limited benefits for adversarial data but failed to generalize well to all data types, potentially compromising feature recognition. The strategy of adjusting classifiers through "smoothing" demonstrated moderate adversarial robustness, especially against adversarial examples with iterative perturbations, while generally preserving high classification accuracy on clean data. For the intents of developing classifiers trained using combinations of

orthogonal techniques, the experimentation revealed that strategies far from the Nash Equilibrium may benefit greatly from being combined with orthogonal strategies, but those close to the Nash Equilibrium are likely to stray further from the Nash Equilibrium rather than becoming more accurate when combined with another strategy.

The decision on which strategy to use for enhancing classification robustness should be based on the specific needs of the application. Adversarial training is the best choice when the priority is to accurately classify adversarial data, even if it means sacrificing some accuracy on clean data. On the other hand, classifier smoothing preserves high accuracy on clean data and offers moderate adversarial robustness (though not as optimal as adversarial training), but can still be subverted by high-magnitude single-step perturbations. This makes it suitable for applications where clean data classification is essential, where the occurrence of highly perturbed data is minimal. Currently, denoising strategies do not match the adversarial robustness provided by adversarial training, and hamper classification accuracy on clean data, making them less suitable for applications. Further research is needed to improve generalization and the ability of denoisers to maintain clean data accuracy. Regarding orthogonal combinations of strategies, further study must be done on the benefit of combining orthogonal strategies that are close approximations of the Nash Equilibrium and do not have significant deficits in classification accuracy for the test data types discussed.

References

Biggio, B., & Roli, F. (2017). Wild patterns: Ten years after the rise of adversarial machine learning. *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security.*

Carlini, N., & Wagner, D. A. (2016a). Defensive distillation is not robust to adversarial examples. *ArXiv*, *abs/1607.04311*.

Carlini, N., & Wagner, D. A. (2016b). Towards evaluating the robustness of neural networks. *2017 IEEE Symposium on Security and Privacy (SP)*, 39–57.

Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., & Mukhopadhyay, D. (2021). A survey on adversarial attacks and defences. *CAAI Transactions on Intelligence Technology*, *6*(1), 25–45.

Dasgupta, P., & Collins, J. B. (2019). A survey of game theoretic approaches for adversarial machine learning in cybersecurity tasks. *AI Mag.*, *40*, 31–43.

Gilmer, J., Adams, R. P., Goodfellow, I. J., Andersen, D. G., & Dahl, G. E. (2018). Motivating the rules of the game for adversarial example research. *ArXiv*, *abs/1807.06732*.

Goodfellow, I. J., Mcdaniel, P., & Papernot, N. (2018). Making machine learning robust against adversarial inputs. *Communications of the ACM*, *61*, 56–66.

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. *CoRR*, *abs/1412.6572*.

Huang, L., Joseph, A. D., Nelson, B., Rubinstein, B. I. P., & Tygar, J. D. (2011). Adversarial machine learning. *Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence*, 43–58.

Kurakin, A., Goodfellow, I. J., & Bengio, S. (2016). Adversarial machine learning at scale. *ArXiv*, *abs/1611.01236*.

Li, B., Chen, C., Wang, W., & Carin, L. (2018). Certified adversarial robustness with additive noise. *Neural Information Processing Systems*.

Liu, X., Cheng, M., Zhang, H., & Hsieh, C.-J. (2018). Towards robust neural networks via random self-ensemble. *Computer Vision – ECCV 2018*, 381–397.

Papernot, N., Mcdaniel, P., Wu, X., Jha, S., & Swami, A. (2015). Distillation as a defense to adversarial perturbations against deep neural networks. *2016 IEEE Symposium on Security and Privacy (SP)*, 582–597.

Ren, K., Zheng, T., Qin, Z., & Liu, X. (2020). Adversarial attacks and defenses in deep learning. *Engineering*.

Robey, A., Latorre, F., Pappas, G., Hassani, H., & Cevher, V. (2023). Adversarial training should be cast as a non-zero-sum game. *ArXiv*, *abs/2306.11035*.

Ruan, W., Yi, X., & Huang, X. (2021). Adversarial robustness of deep learning: Theory, algorithms, and applications. *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*.

Sahay, R., Mahfuz, R., & Gamal, A. E. (2018). Combatting adversarial attacks through denoising and dimensionality reduction: A cascaded autoencoder approach. *2019 53rd Annual Conference on Information Sciences and Systems (CISS)*, 1–6.

Salman, H., Sun, M., Yang, G., Kapoor, A., & Kolter, J. Z. (2020). Denoised smoothing: A provable defense for pretrained classifiers. *arXiv: Learning*.

Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2016). Membership inference attacks against machine learning models. *2017 IEEE Symposium on Security and Privacy (SP)*, 3–18.

Sun, L., Tan, M., & Zhou, Z. (2018). A survey of practical adversarial example attacks. *Cybersecurity*, *1*, 1–9.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., & Fergus, R. (2013). Intriguing properties of neural networks. *CoRR*, *abs/1312.6199*.

Tramèr, F., Zhang, F., Juels, A., Reiter, M. K., & Ristenpart, T. (2016). Stealing machine learning models via prediction apis. *Proceedings of the 25th USENIX Conference on Security Symposium*, 601–618.

Zhao, W., Alwidian, S. A., & Mahmoud, Q. H. (2022). Adversarial training methods for deep learning: A systematic review. *Algorithms*, *15*, 283.

Zuo, S., Liang, C., Jiang, H., Liu, X., He, P., Gao, J., Chen, W., & Zhao, T. (2021). Adversarial regularization as stackelberg game: An unrolled optimization approach. *Conference on Empirical Methods in Natural Language Processing*.