

---

# GAME THEORETIC MIXED EXPERTS FOR COMBINATIONAL ADVERSARIAL MACHINE LEARNING

**Ethan Rathbun, Kaleel Mahmood, Sohaib Ahmad, & Caiwen Ding**

Department of Computer Science and Engineering  
University of Connecticut  
{first.last}@uconn.edu

**Marten van Dijk**

CWI, Amsterdam  
The Netherlands  
marten.van.dijk@cwi.nl

## ABSTRACT

Recent advances in adversarial machine learning have shown that defenses considered to be robust are actually susceptible to adversarial attacks which are specifically tailored to target their weaknesses. These defenses include Barrage of Random Transforms (BaRT), Friendly Adversarial Training (FAT), Trash is Treasure (TiT) and ensemble models made up of Vision Transformers (ViTs), Big Transfer models and Spiking Neural Networks (SNNs). A natural question arises: how can one best leverage a combination of adversarial defenses to thwart such attacks? In this paper, we provide a game-theoretic framework for ensemble adversarial attacks and defenses which answers this question. In addition to our framework we produce the first adversarial defense transferability study to further motivate a need for combinational defenses utilizing a diverse set of defense architectures. Our framework is called Game theoretic Mixed Experts (GaME) and is designed to find the Mixed-Nash strategy for a defender when facing an attacker employing compositional adversarial attacks. We show that this framework creates an ensemble of defenses with greater robustness than multiple state-of-the-art, single-model defenses in addition to combinational defenses with uniform probability distributions. Overall, our framework and analyses advance the field of adversarial machine learning by yielding new insights into compositional attack and defense formulations. Corresponding code to replicate our results can be found [here](#).

## 1 INTRODUCTION

Machine learning models have been shown to be vulnerable to adversarial examples Goodfellow et al. (2014); Papernot et al. (2016). Adversarial examples are inputs with small perturbations added, such that machine learning models misclassify the example with high confidence. Addressing the security risks posed by adversarial examples are critical for the safe deployment of machine learning in areas like health care Finlayson et al. (2019) and self driving vehicles Qayyum et al. (2020). However, current defenses and attacks in adversarial machine learning have trended towards a cat and mouse dynamic where in new defenses are continually being proposed and then broken Carlini & Wagner (2017); Tramer et al. (2020); Mahmood et al. (2021a); Sitawarin et al. (2022) by improved attacks.

In parallel to attack and defense development, studies have also been conducted on the transferability of adversarial examples Liu et al. (2016); Mahmood et al. (2021b); Xu et al. (2022). Transferability refers to the phenomena where adversarial examples generated for one model are also misclassified by a different machine learning model. However, to the best of our knowledge no analyses have been done on the transferability of adversarial examples designed to attack specific defenses. From these observations several pertinent questions arise:

- 
1. *Do adversarial examples generated for one specific defense transfer to other defenses?*
  2. *Based on adversarial transferability, can a game theoretic framework be developed to determine the optimal choices for both attacker and defender?*
  3. *Can randomized defense selection yield higher robustness than a single state-of-the-art defense?*

These are precisely the questions our paper seeks to answer. We break from the traditional dynamic of adversarial machine learning which focuses on the single best attack and defense. We instead take a multi-faceted approach and develop a game theoretic framework to answer the above questions. Specifically, we provide the following contributions: Most importantly, we formulate a practical, game-theoretic framework for finding the optimal strategies for an attacker and defender who each employ a set of state-of-the-art adversarial attacks and defenses. Motivated by this framework, we develop two new white-box attacks called the Momentum Iterative Method over Expectation (MIME) and the Auto Expectation Self-Attention Gradient Attack (AE-SAGA) in order to create a stronger adversary. These attacks are necessary for targeting certain randomized defenses and for adapting to multi-defense strategies. Lastly, we analyze the adversarial transferability of current defenses like Trash is Treasure Xiao & Zheng (2020), Barrage of Random Transforms Raff et al. (2019), Friendly Adversarial Training Zhang et al. (2020) and other new architectures like SNNs Rathi & Roy (2021b); Fang et al. (2021) and ViTs Dosovitskiy et al. (2020). We further leverage the low transferability between these classifiers to find those which are best suited for a combined, ensemble defense such as the one developed in our game-theoretic framework.

## 2 ADVERSARIAL MACHINE LEARNING DEFENSES

Here we summarize the state-of-the-art defenses we analyze in this paper. In the following subsections, we give an overview of each defense and our reasons for choosing said defense. It is important to note our analyses encompass a broad range of different defenses, including ones based on randomization, ones based on adversarial training and ones based on exploiting model transferability. In addition, we also consider diverse architectures including Big Transfer models (BiTs), Vision Transformers (ViTs) and Spiking Neural Networks (SNNs). Despite our broad range, we do not attempt to test every novel adversarial defense. It is simply infeasible to test every proposed adversarial machine learning defense, as new defenses are constantly being produced. However, based on our game theoretic design and open source code (which will be provided upon publication), any new defense can easily be tested and integrated into our proposed framework.

### 2.1 BARRAGE OF RANDOM TRANSFORMS

Barrage of Random Transforms (BaRT) Raff et al. (2019) utilize a set of image transformations in a random order and with randomized transformation parameters to thwart adversarial attacks. Let  $t_j^i(x)$  represent the  $i^{th}$  transformation used in the  $j^{th}$  order in the sequence. A BaRT defense using  $n$  image transformations randomly alters the input  $x$ :

$$t(x) = t_{\mu_n}^{\omega_n} \circ t_{\mu_{n-1}}^{\omega_{n-1}} \circ \dots \circ t_{\mu_1}^{\omega_1}(x) \quad (1)$$

where  $\omega$  represents the subset of  $n$  transformations randomly selected from a set of  $N$  total possible transformations and  $\mu$  represents the randomized order in which the  $n$  transformations are applied. In Equation 1 the parameters of each image transformation  $t_j^i(x)$  are also randomized at run time, further adding to the stochastic nature of the defense. In this paper, we work with the original BaRT implementation which includes both differentiable and non-differentiable image transformations.

**Why we selected it:** Many defenses are broken soon after being proposed Tramer et al. (2020). BaRT is one of the few defenses that has continued to show robustness even when attacks are specifically tailored to work against it. For example, most recently BaRT achieves 29% robustness on CIFAR-10 against a customized white-box attack Sitawarin et al. (2022). It remains an open question whether using BaRT with other randomized approaches (i.e. selecting between different defenses) can yield even greater robustness.

---

## 2.2 FRIENDLY ADVERSARIAL TRAINING

Training classifiers to correctly recognize adversarial examples was originally proposed in Goodfellow et al. (2014) using FGSM. This concept was later expanded to include training on adversarial examples generated by PGD in Madry et al. (2018). In Zhang et al. (2020) it was shown that Friendly Adversarial Training (FAT) could achieve high clean accuracy while maintaining robustness to adversarial examples. This training was accomplished by using a modified version of PGD called PGD- $K-\tau$ . In PGD- $K-\tau$ ,  $K$  refers to the number of iterations used for PGD. The  $\tau$  variable is a hyperparameter used in training which stops the PGD generation of adversarial examples earlier than the normal  $K$  number of steps, if the sample is already misclassified.

**Why we selected it:** There are many different defenses that rely on adversarial training Madry et al. (2018); Zhang et al. (2019); Wang et al. (2019); Maini et al. (2020) and training and testing them all is not computationally feasible. We selected FAT for its good trade off between clean accuracy and robustness, and because we wanted to test adversarial training on both Vision Transformer and CNN models. In this regard, FAT is one of the adversarial training methods that has already been demonstrated to work across both types of architectures Mahmood et al. (2021b).

## 2.3 TRASH IS TREASURE

One early direction in adversarial defense design was model ensembles Pang et al. (2019). However, due to the high transferability of adversarial examples between models, such defenses were shown to not be robust Tramer et al. (2020). Trash is Treasure (TiT) Xiao & Zheng (2020) is a two model defense that seeks to overcome the transferability issue by training one model  $C_a(\cdot)$  on the adversarial examples from another model  $C_b(\cdot)$ . At run time both models are used:

$$y = C_a(\psi(x, C_b)) \quad (2)$$

where  $\psi$  is an adversarial attack done on model  $C_b$  with input  $x$  and  $C_a$  is the classifier that makes the final class label prediction on the adversarial example generated by  $\psi$  with  $C_b$ .

**Why we selected it:** TiT is one of the newest defenses that tries to achieve robustness in a way that is fundamentally different than pure randomization strategies or direct adversarial training. In our paper, we further develop two versions of TiT. One version is based on the original proposed CNN-CNN implementation. We also test a second mixed architecture version using Big Transfer model and Vision Transformers to try and leverage the low transferability phenomena described in Mahmood et al. (2021b).

## 2.4 NOVEL ARCHITECTURES

In addition to adversarial machine learning defenses, we also include several novel architectures that have recently achieved state-of-the-art or near state-of-the-art performance in image recognition tasks. These include the Vision Transformer (ViT) Dosovitskiy et al. (2020) and Big Transfer models (BiT) Kolesnikov et al. (2020). Both of these types of models utilize pre-training on larger datasets and fine tuning on smaller datasets to achieve high fidelity results. We also test Spiking Neural Network (SNNs) architectures. SNNs are a competitor to artificial neural networks that can be described as a linear time invariant system with a network structure that employs non-differentiable activation functions Xu et al. (2022). A major challenge in SNNs has been matching the depth and model complexity of traditional deep learning models. Two approaches have been used to overcome this challenge, the Spiking Element Wise (SEW) ResNet Fang et al. (2021) and transferring weights from existing CNN architectures to SNNs Rathi & Roy (2021a). We experiment with both approaches in our paper.

**Why we selected it:** The set of adversarial examples used to attack one type of architecture (e.g. a ViT) have shown to not be misclassified by other architecture types (e.g. a BiT or SNN) Mahmood et al. (2021b); Xu et al. (2022). While certain white-box attacks have been used to break multiple undefended models, it remains an open question if different architectures combined with different defenses can yield better performance.

### 3 ADVERSARIAL ATTACKS

In our paper we assume a white-box adversarial threat model. This means the attacker is aware of the set of all defenses  $D$  that the defender may use for prediction. In addition,  $\forall d \in D$  the attacker also knows the classifier weights  $\theta_d$ , architecture and any input image transformations the defense may apply. To generate adversarial examples the attacker solves the following optimization problem:

$$\max_{\delta} \sum_{d \in D} L_d(x + \delta, y; d) \quad \text{subject to: } \|\delta\|_p \leq \epsilon \quad (3)$$

where  $D$  is the set of all possible defenses (models) under consideration in the attack,  $L_d$  is the loss function associated with defense  $d \in D$ ,  $\delta$  is the adversarial perturbation, and  $(x, y)$  represents the original input with corresponding class label. This is a more general formulation of the optimization problem allowing the attacker to attack single or multi-model classifiers. The magnitude of this perturbation  $\delta$  is typically limited by a certain  $l_p$  norm. In this paper we analyze the attacks and defenses using the  $l_\infty$  norm.

Static white-box attacks such as the Projected Gradient Descent (PGD) Madry et al. (2018) attack often perform poorly against randomized defenses such as BaRT or TiT. In Xiao & Zheng (2020) they tested the TiT defense against an attack designed to compensate for randomness, the Expectation over Transformation attack (EOT) attack Athalye et al. (2018). However, it was shown that the EOT attack performs poorly against TiT (e.g. 20% or worse attack success rate). For attacking BaRT, in Sitawarin et al. (2022) they proposed a new white-box attack to break BaRT. However, this new attack requires that the image transformations used in the BaRT defense be differentiable, which is a deviation from the original BaRT implementation.

**Attack Contributions:** It is crucial for both the attacker and defender to consider the strongest possible adversary when playing the adversarial examples game. Thus, we propose two new white-box attacks for targeting randomized defenses. The first attack is designed to work on individual randomized defenses and is called the Momentum Iterative Method over Expectation (MIME). To the best of our knowledge, MIME is the first white-box attack to achieve a high attack success rate (> 70%) against TiT. MIME is also capable of achieving a high attack success rate against BaRT, even when non-differentiable transformations are implemented as part of the defense. Our second attack, is designed to generate adversarial examples that work against multiple type of defenses (both randomized and non-randomized defenses) simultaneously. This compositional attack is called, the Auto Expectation Self-Attention Gradient Attack (AE-SAGA).

#### 3.1 MOMENTUM ITERATIVE METHOD OVER EXPECTATION

We develop a new attack white-box attack specifically designed to work on defenses that inherently rely on randomization, like Barrage of Random Transforms (BaRT) Raff et al. (2019) and Trash is Treasure (TiT) Xiao & Zheng (2020). Our new attack is called the Momentum Iterative Method over Expectation (MIME). The attack ‘‘mimes’’ the transformations of the defender in order to more precisely model the gradient of the loss function with respect to the input after the transformations are applied. To this end, MIME utilizes two effective aspects from earlier white-box attacks, momentum from the Momentum Iterative Method (MIM) Dong et al. (2018) attack and repeated sampling Athalye et al. (2018) from the Expectation Over Transformation (EOT) attack:

$$x_{adv}^{(i)} = x_{adv}^{(i-1)} + \epsilon_{\text{step}} g^{(i)} \quad (4)$$

where the attack is computed iteratively with  $x_{adv}^{(0)} = x$ . In Equation 4  $g^{(i)}$  is the momentum based gradient of the loss function with respect to the input at iteration  $i$  and is defined as:

$$g^{(i)} := \gamma g^{(i-1)} + \mathbb{E}_{t \sim T} \left[ \frac{\partial L}{\partial t(x_{adv}^{(i)})} \right] \quad (5)$$

where  $\gamma$  is the momentum decay factor and  $t$  is a random transformation function drawn from the defense’s transformation distribution  $T$ . In Table 1 we show experimental results for the MIME attack on CIFAR-10 randomized defenses (TiT and BaRT). It can clearly be seen that MIME has a better attack success rate than both APGD Croce & Hein (2020) and MIM Dong et al. (2018).

Attack	BaRT-1	BaRT-5	BaRT-10	TiT (BiT/ViT)	TiT (VGG/RN)
MIME-10	<b>3.18%</b>	15.5%	43.2%	10.1%	24.9%
MIME-50	4.3%	<b>8.22%</b>	<b>23.2%</b>	<b>8.3%</b>	<b>23.3%</b>
MIM	6.7%	39.5%	59.5%	52%	58.9%
APGD	8.9%	47.7%	70.8%	68.2%	40.7%
Clean	98.4%	95.3%	92.5%	90.1%	76.6%

Table 1: Performance of the MIME attack against CIFAR-10 randomized defenses: Trash is Treasure (TiT) and Barrage of Random Transforms (BaRT). In our experiments  $\epsilon_{max} = 0.031$  for all attacks. It can clearly be seen that MIME outperforms both APGD and MIM on these two randomized defenses. Further defense and attack implementation details are given in Table 3

### 3.2 AUTO EXPECTATION SELF-ATTENTION GRADIENT ATTACK

The use of multi-model attacks are necessary to achieve a high attack success rate when dealing with ensembles that contain both CNN and non-CNN model architectures like the Vision Transformer (ViT) Dosovitskiy et al. (2020) and Spiking Neural Network (SNN) Fang et al. (2021). This is because adversarial examples generated by single model white-box attacks generally do not transfer well between CNNs, ViTs and SNNs Mahmood et al. (2021b); Xu et al. (2022). In addition it is an open question if multi-model attacks can be effective against the current state-of-the-art defenses. In this paper, we expand the idea of a multi-model attack to include not only different architecture types, but also different defenses. The generalized form of the multi-model attack is found in Equation 3.

For a single input  $x$  with corresponding class label  $y$ , an untargeted multi-model attack is considered successful if  $(\forall d \in D, C_d(x + \delta) \neq y) \wedge (\|\delta\|_p \leq \epsilon)$ . One formulation of the multi-model attack is the Auto Self-Attention Gradient Attack (Auto SAGA) which was proposed in Xu et al. (2022) to iteratively attack combinations of ViTs, SNNs and CNNs:

$$x_{adv}^{(i+1)} = x_{adv}^{(i)} + \epsilon_{step} * \text{sign}(G_{blend}(x_{adv}^{(i)})) \quad (6)$$

where  $\epsilon_{step}$  was the step size used in the attack. In the original formulation of Auto-SAGA,  $G_{blend}$  was a weighted average of the gradients of each model  $d \in D$ . By combining gradient estimates from different models, Auto-SAGA is able to create adversarial examples that are simultaneously misclassified by multiple models. One limitation of Auto-SAGA attack is that it does not account for defenses that utilize random transformations. Motivated by this, we can integrate the previously proposed MIME attack into the gradient calculations for Auto-SAGA. We denote this new attack as the Auto Expectation Self-Attention Gradient Attack (AE-SAGA). Both SAGA and AE-SAGA use the same iterative update (Equation 6). However, AE-SAGA uses the following gradient estimator:

$$G_{blend}(x_{adv}^{(i)}) = \gamma G_{blend}(x_{adv}^{(i-1)}) + \sum_{k \in D \setminus R} \alpha_k^{(i)} \phi_k^{(i)} \odot \frac{\partial L_k}{\partial x_{adv}^{(i)}} + \sum_{r \in R} \alpha_r^{(i)} \phi_r^{(i)} \odot (\mathbb{E}_{t \sim T} [\frac{\partial L_r}{\partial t(x_{adv}^{(i)})}]) \quad (7)$$

In Equation 7 the two summations represent the gradient contributions of sets  $D \setminus R$  and  $R$ , respectively. Here we define  $R$  as the set of randomized defenses and  $D$  as the set of all the defenses being targeted. In each summation  $\phi$  is the self-attention map Abnar & Zuidema (2020) which is replaced with a matrix of all ones for any defense that does not use ViT models.  $\alpha_k$  and  $\alpha_r$  are the associated weighting factors for the gradients for each deterministic defense  $k$  and randomized defense  $r$ , respectively. Details of how the weighting factors are derived are given in Xu et al. (2022).

## 4 TRANSFERABILITY EXPERIMENTS

Adversarial transferability refers to the phenomena in which adversarial examples generated to attack one model are also misclassified by a different model. Adversarial transferability studies have been done on a variety of machine learning models Liu et al. (2016); Mahmood et al. (2021b); Xu et al. (2022). However, to the best of our knowledge, adversarial transferability between different state-of-the-art defenses has not been conducted. This transferability property is of significant interest because a lack of transferability between different defenses may indicate a new way to improve adversarial robustness.

In Table 2 we show the different single defenses we analyze in this paper and the best attack on each of them from the set of attacks (MIM Dong et al. (2018), APGD Croce & Hein (2020) and MIME

Defense	Best Attack	Clean Acc	Robust Acc
B1	MIME	98.40%	3.40%
B5	MIME	95.30%	15.00%
B10	MIME	92.50%	43.50%
RF	APGD	81.89%	52.00%
VF	APGD	92.36%	25.00%
ST	APGD	91.54%	0.00%
SB	APGD	81.16%	1.60%
BVT	MIME	90.10%	8.60%
VRT	MIME	76.60%	26.20%

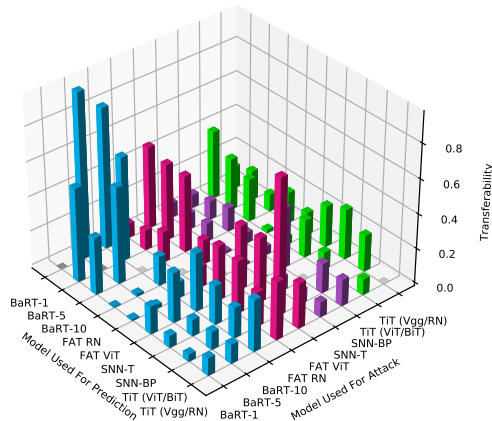


Table 2: Single defense implementations for CIFAR-10 with the corresponding strongest attack on the defense and the clean accuracy of the defense. The robust accuracy is measured using 1000 adversarial examples. The examples are class wise balanced and also correctly recognized by all the defenses in their original clean form. All attacks are done using  $\epsilon_{max} = 0.031$ . Complete attack and defense implementation details are given in our supplementary material.

Figure 1: Visual representation of the transferability of adversarial examples between defenses for CIFAR-10. The blue, green, pink, and purple bars represent adversarial examples generated using the best attack for BaRT, TiT, FAT, and SNN classifiers respectively. Full numerical tables used to generate this figure are given in our supplemental material.

(proposed in this work). In Figure 1 we visually show the transferability results of these attacks for CIFAR-10. We give detailed discussions of these results in our supplementary material and briefly summarize of the key takeaways from these experiments.

In general adversarial examples generated using the best attack on one defense *do not* transfer well to other defenses. For example, only 0.8% of the adversarial examples generated by the BaRT-1 defense transfer to the FAT ViT defense. The average transferability for the 8 different defenses shown in Figure 1 is only 21.62% and there is no single model attack that achieves strong performance (i.e.  $> 50\%$ ) across all defenses. These results in turn motivate the development of a game theoretic framework for both the attacker and defender. For the attacker, this prompts the need to use multi-model attack like AE-SAGA that was proposed in Section 3, as no single attack (APGD, MIM or MIME) is ubiquitous. For the defender these results highlight the opportunity to increase robustness by taking advantage of the low levels of transferability between defenses through the implementation of a randomized ensemble defense.

## 5 GAME THEORETIC MIXED EXPERTS

In this section we derive our framework, Game theoretic Mixed Experts (GaME), for approximating a Nash equilibrium in the adversarial examples game. In comparison to other works Meunier et al. (2021) Pinot et al. (2020) Pal & Vidal (2020) Balcan et al. (2022) Le et al. (2022) we take a more discretized approach and solve the approximate version of the adversarial examples game. This ultimately leads to the creation of a finite, tabular, zero-sum game that can be solved in polynomial time using linear programming techniques. In relation to our work, a similar adversarial game framework was proposed in Sengupta et al. (2018) but did not include comprehensive defender and attacker threat models. Specifically, we develop our framework under an adversary that can employ state-of-the-art single model and multi-model attacks and a defender that can utilize both randomization and voting schemes.

## 5.1 THE ADVERSARIAL EXAMPLES GAME

We build upon and discretize the adversarial examples game explored in Meunier et al. (2021). The adversarial examples game is a zero-sum game played between two players: the attacker,  $p_A$ , and the defender  $p_D$ . Let  $\mathcal{X}$  be the input space and  $\mathcal{Y}$  the output space of  $p_D$ 's classifiers, and let  $\Theta$  represent the space of classifier parameters. Additionally, let  $P_{\epsilon, \mathcal{X}} = \{x \in \mathcal{X} : \|x\|_p \leq \epsilon\}$  be the set of valid adversarial perturbations for norm  $p$  and  $\epsilon \in \mathbb{R}^+$ .

Let  $A_\epsilon^* = \{(f : \Theta \times \mathcal{X} \times \mathcal{Y} \rightarrow P_{\epsilon, \mathcal{X}})\}$  be the set of all valid attack functions. The goal of  $p_A$  is to choose  $a \in A_\epsilon^*$ , which maximizes the expected loss of  $p_D$ 's classifier,  $\theta$ , given some pair of input and ground truth label  $(x, y) \in \mathcal{X} \times \mathcal{Y}$ . The goal of  $p_D$  is to minimize this loss through its choice of  $\theta \in \Theta$ . We can thus formulate the adversarial examples game as a mini-max optimization problem:

$$\inf_{\theta \in \Theta} \sup_{a \in A_\epsilon^*} \mathbb{E}_{(x, y) \sim \mathcal{X} \times \mathcal{Y}} [L(x + a(\theta, x, y), y; \theta)] \quad (8)$$

Due to the vastness of  $\Theta$  and  $A_\epsilon^*$ , solving this optimization problem directly is currently computationally intractable. To this end, in the next subsections we will formulate  $\text{GaME}_1$  and  $\text{GaME}_n$  which discretize  $\Theta$  and  $A_\epsilon^*$  by enlisting a set of state-of-the-art attacks and defenses.

## 5.2 $\text{GAME}_1$

The goal of the  $\text{GaME}$  framework is to find an approximate solution to the adversarial examples game through the implementation of a set of attacks and defenses which will serve as experts for  $p_A$  and  $p_D$  respectively.

Let  $A' \subset A_\epsilon^*$  be a subset of all valid adversarial attack functions chosen by  $p_A$ . Additionally, let  $D \subset \Theta$  be a set of defense classifiers chosen by  $p_D$ . We further impose that all  $a \in A'$  are white-box attacks (see Section 3 for our adversarial threat model) and that  $A', D$  are finite, i.e.  $|A'| \leq N_a$  and  $|D| \leq N_d$  for some  $N_a, N_d \in \mathbb{N}$ .

It is important to note that each  $a \in A'$  is a function of some classifier,  $\theta \in \Theta$ , in addition to the input and ground truth label. Due to this it is possible for  $p_A$  to choose to attack defense  $d \in D$  with attack  $a \in A'$ , while  $p_D$  chooses to evaluate the sample using defense  $d' \in D$  where  $d \neq d'$ . Therefore, for convenience, we will define a new, more general set of attack strategies for  $p_A$ :

$$A \subseteq \{(f : \mathcal{X} \times \mathcal{Y} \rightarrow P_{\epsilon, \mathcal{X}}) : f(x, y) = a_i(U, x, y), a_i \in A', U \subseteq D\} \quad (9)$$

where we extend the definition of  $A' \subseteq A_\epsilon^*$  to attack functions that can take subset of defense parameters  $U \subseteq D$  as input (see Equation 3 for our multi-model attack formulation). This comes into play with multi-model attacks like AE-SAGA. Thus we will let  $D$  be the strategy set of  $p_D$ , and  $A$  be the strategy set of  $p_A$ . We can then reformulate a discretized version of the adversarial examples game as follows:

$$\min_{d \in D} \max_{a \in A} \mathbb{E}_{(x, y) \sim \mathcal{X} \times \mathcal{Y}} [L(x + a(x, y), y; d)] \quad (10)$$

In the above two formulations we optimize over the set of pure strategies for the attacker and defender. However, as previously explored in Araujo et al. (2020) Meunier et al. (2021), limiting ourselves to pure strategies severely inhibits the strength of both the attacker and defender. Thus we create the following mixed strategy vectors for  $p_A, p_D$ :

$$\lambda^A \in \{r \in [0, 1]^{|A|} : \|r\|_1 = 1\}, \quad \lambda^D \in \{r \in [0, 1]^{|D|} : \|r\|_1 = 1\} \quad (11)$$

here  $\lambda^A$  and  $\lambda^D$  represent the mixed strategies of  $p_A$  and  $p_D$  respectively. Let each  $a_i \in A$  and  $d_i \in D$  correspond to the  $i^{\text{th}}$  elements of  $\lambda^A$  and  $\lambda^D$ ,  $\lambda_i^A$  and  $\lambda_i^D$ , respectively. Formally:

$$\mathbb{P}(\{a_i \in A : a = a_i\}) = \lambda_i^A, \quad \mathbb{P}(\{d_i \in D : d = d_i\}) = \lambda_i^D \quad (12)$$

where  $a \in A$  and  $d \in D$  are random variables. With these mixed strategy vectors we can then reformulate the adversarial examples game as a mini-max optimization problem over  $p_D$ 's choice of  $\lambda^D$  and  $p_A$ 's choice of  $\lambda^A$ :

$$\begin{aligned} & \min_{\lambda^D} \max_{\lambda^A} \mathbb{E}_{(x, y) \sim \mathcal{X} \times \mathcal{Y}} [\mathbb{E}_{(a, d) \sim A \times D} [L(x + a(x, y), y; d)]] = \\ & \min_{\lambda^D} \max_{\lambda^A} \mathbb{E}_{(x, y) \sim \mathcal{X} \times \mathcal{Y}} \left[ \sum_{a_i \in A} \lambda_i^A \sum_{d_i \in D} \lambda_i^D [L(x + a_i(x, y), y; d_i)] \right] \end{aligned} \quad (13)$$

For continuous and or non-finite  $\mathcal{X}$ ,  $D$ , and  $A$  solving the above optimization problem is currently computationally intractable. Thus we can instead approximate the mini-max optimization by taking  $N$  Monte-Carlo samples with respect to  $(x_j, y_j) \in \mathcal{X} \times \mathcal{Y}$ :

$$\min_{\lambda^D} \max_{\lambda^A} \frac{1}{N} \sum_{j=0}^N \sum_{a_i \in A} \lambda_i^A \sum_{d_i \in D} \lambda_i^D [L(x_j + a_i(x_j, y_j), y_j; d_i)] \quad (14)$$

For convenience we will denote  $r_{d_i, a_i} = \frac{1}{N} \sum_{j=0}^N [L(x_j + a_i(x_j, y_j), y_j; d_i)]$ . Colloquially  $r_{d_i, a_i}$  represents the expected robustness of defense  $d_i$  when evaluating adversarial samples generated by attack  $a_i$ . From a game theoretic perspective,  $r_{d, a}$  is the payoff for  $p_D$  when they play strategy  $d$  and  $p_A$  plays strategy  $a$ . The payoff for  $p_A$  given strategies  $d, a$  is  $-r_{d, a}$ . Our mini-max optimization problem can then be simplified to:

$$\min_{\lambda^D} \max_{\lambda^A} \sum_{a_i \in A} \lambda_i^A \sum_{d_i \in D} \lambda_i^D r_{d_i, a_i} \quad (15)$$

Using all of this we can create a finite, tabular, zero-sum game defined by the following game-frame in strategic form:

$$\langle \{p_A, p_D\}, (A, D), O, f \rangle \quad (16)$$

where  $O = \{r_{d, a} \forall a \in A, d \in D\}$  and  $f$  is a function  $f : A \times D \rightarrow O$  defined by  $f(d, a) = r_{d, a}$ . Since this is a finite, tabular, zero-sum game, we know that it must have a Nash-Equilibrium as proven in Nash (1951). Let  $R$  be the payoff matrix for  $p_D$  where  $R_{d, a} = r_{d, a}$ . It then becomes the goal of  $p_D$  to maximize their guaranteed, expected payoff. Formally,  $p_D$  must solve the following optimization problem:

$$\max_{r^*; \lambda^D} r^* \quad \text{subject to } \lambda^D R \geq (r^*, \dots, r^*) \text{ and } \|\lambda^D\|_1 \leq 1 \quad (17)$$

This optimization problem is a linear program, the explicit form of which we provide in the supplemental material. All linear programs have a dual problem, in this case the dual problem finds a mixed Nash strategy for  $p_A$ . This can be done by changing the problem to a minimization problem and transposing  $R$ . In the interest of space we give the explicit form of the dual problem in the supplemental material as well. These linear programs can be solved using polynomial time algorithms.

### 5.3 GAME<sub>n</sub>

GAME<sub>n</sub> is a more general family of games of which GAME<sub>1</sub> is a special case. In GAME<sub>n</sub>, for  $n > 1$ ,  $p_D$  can calculate their final prediction based upon the output logits of multiple  $d \in D$  evaluated on the same input  $x$ . In order to do this,  $p_D$  must also choose a function to map the output of multiple defenses to a final prediction. Formally, the strategy set of  $p_D$  becomes  $D = D' \times F$ , where  $F$  is a set of prediction functions and  $D'$  is defined as follows.

$$D' \subseteq \{U : U \subseteq D, |U| \leq n\} \quad (18)$$

Multi-model prediction functions can increase the overall robustness of an ensemble by requiring an adversarial sample to be misclassified by multiple models simultaneously Mahmood et al. (2022). In this paper we will focus on two voting functions: the majority vote Raff et al. (2019), and the largest softmax probability vote Sitawarin et al. (2022). We will refer to these as  $f^h$  and  $f^s$  respectively:

$$f^h(x, U) = \arg \max_{y \in \mathcal{Y}} \sum_{d \in U} \mathbb{1}\{y = \arg \max_{j \in \mathcal{Y}} d_j(x)\} \quad (19)$$

$$f^s(x, U) = \arg \max_{y \in \mathcal{Y}} \frac{1}{|U|} \sum_{d \in U} \sigma(d(x)) \quad (20)$$

where  $\mathbb{1}$  is the indicator function,  $\sigma$  is the softmax function, and  $d_j(x)$  represents the  $j^{\text{th}}$  output logit of defense  $d$  when evaluated on input  $x$ . Solving GAME<sub>n</sub> is the same as solving GAME<sub>1</sub>, except  $|D|$  will be much larger. Notationally the mini-max optimization problem in terms of  $r_{d, a}$  remains the same as we have simply redefined  $D$ , however we can redefine  $r_{d, a}$  as follows:

$$r_{(U_i, f_j), a_k} = \sum_{l=0}^N [L(f_j(U, (x_l + a_k(x_l, y_l))), y_l)] \quad (21)$$

where  $(U_i, f_j) \in D = D' \times F$ . Similarly to GAME<sub>1</sub>, in GAME<sub>n</sub>  $r_{(U_i, f_j), a_k}$  represents the expected robust accuracy, i.e. the payoff, for  $p_D$  if they play strategy  $(U_i, f_j)$  and  $p_A$  plays strategy  $a_k$ .



## 6 GAME EXPERIMENTAL SETUP

For our experimental results, we test on two datasets, CIFAR-10 Krizhevsky et al. and Tiny-ImageNet Le & Yang (2015). For CIFAR-10 we solved instances of  $\text{GaME}_n$  using the following defenses: BaRT-1 (B1), BaRT-5 (B5), ResNet-164-FAT (RF), ViT-L-16-FAT (VF), SNN Transfer (ST), Backprop SNN (SB), and TiT using ViT and BiT (BVT). For Tiny ImageNet we solved instances of  $\text{GaME}_n$  utilizing: BaRT-1, BaRT-5, ViT-L-16-FAT, and TiT using ViT and BiT.

We then attacked every random transform defense with MIME, every non-random-transform defense with APGD, and each pair of defenses with AE-SAGA. This came to a total of 28 attacks on CIFAR-10 and 10 attacks on Tiny ImageNet. Every attack was run with respect to the  $l_\infty$  norm. The hyper parameters for each attack are seen in Table 3:

Attack	$\epsilon$	$\epsilon_{step}$	Attack Steps	$N$	$\gamma$	Fitting Factor	$\alpha$ Learning Rate
APGD	.031	.005	20	-	-	-	-
MIM	.031	.0031	10	-	.5	-	-
MIME	.031	.0031	10	10	.5	-	-
AE-SAGA	.031	.005	40	4	.5	50	10000

Table 3: Attack parameters for each of the attacks used in the paper. Here  $\gamma$  represents the momentum decay rate,  $\epsilon$  represents the maximum allowed perturbation magnitude,  $\alpha$  represents the weights used in the AE-SAGA algorithm, and  $N$  represents the number of EOT samples taken. Note for APGD that the  $\epsilon_{step}$  value presented is only the initial value and is subject to change according to the attack’s algorithm.

AE-SAGA has the largest number of iterations since the added complexity of attacking two models simultaneously requires additional attack steps to converge. MIME was given less attack steps due to the computational cost of the attack.

For each attack we first chose an arbitrary, but class-wise balanced set of 1000 clean images from the testing set of each respective data set. From this subset we generated 1000 adversarial examples for each attack. We used 800 of these samples to create the payoff matrix  $R$ , then evaluated the ensemble using the remaining 200, class-wise balanced samples from each attack.

## 7 EXPERIMENTAL RESULTS

Figure 2 shows a visual representation of the performance of our  $\text{GaME}$  generated defenses when compared to single model defenses. Here, and throughout the paper, we measure the robustness of each ensemble as the lowest robust accuracy achieved by any single attack in our study. From a game-theoretic perspective this can be seen as the minimum, guaranteed utility for the defender. Of all the defenses in our study the FAT ResNet-164 had the highest robust accuracy on CIFAR-10 at 50%. On Tiny ImageNet BaRT-5 had the highest robust accuracy at 10.62%. Compared to these defenses our  $\text{GaME}$  generated ensemble achieved 63.5% robustness on CIFAR-10 and 44.5% robustness on Tiny ImageNet, leading to a 13.5% increase and 33.88% increase in robustness on each respective dataset.

In addition to this, our ensemble is able to maintain a high level of clean accuracy in spite of optimizing for robustness in isolation. In particular, the  $\text{GaME}$  framework is able to maintain 96.2% clean accuracy and 72.6% clean accuracy on CIFAR-10 and Tiny ImageNet respectively. This is only out performed by BaRT-1 with a clean accuracy of 98.4% on CIFAR-10 and the BiT-ViT Trash is Treasure defense with a clean accuracy of 76.97% on Tiny ImageNet. It is important to note that in both these cases the robustness of the  $\text{GaME}$  framework is higher.

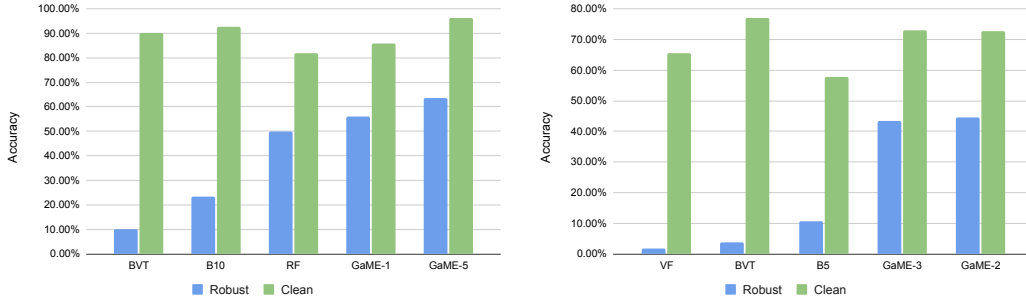


Figure 2: (Left: CIFAR-10; Right: Tiny ImageNet) Comparison of the robust and clean accuracy of the top three most robust single model defenses to the top two most robust GaME generated ensembles on CIFAR-10 and Tiny ImageNet.

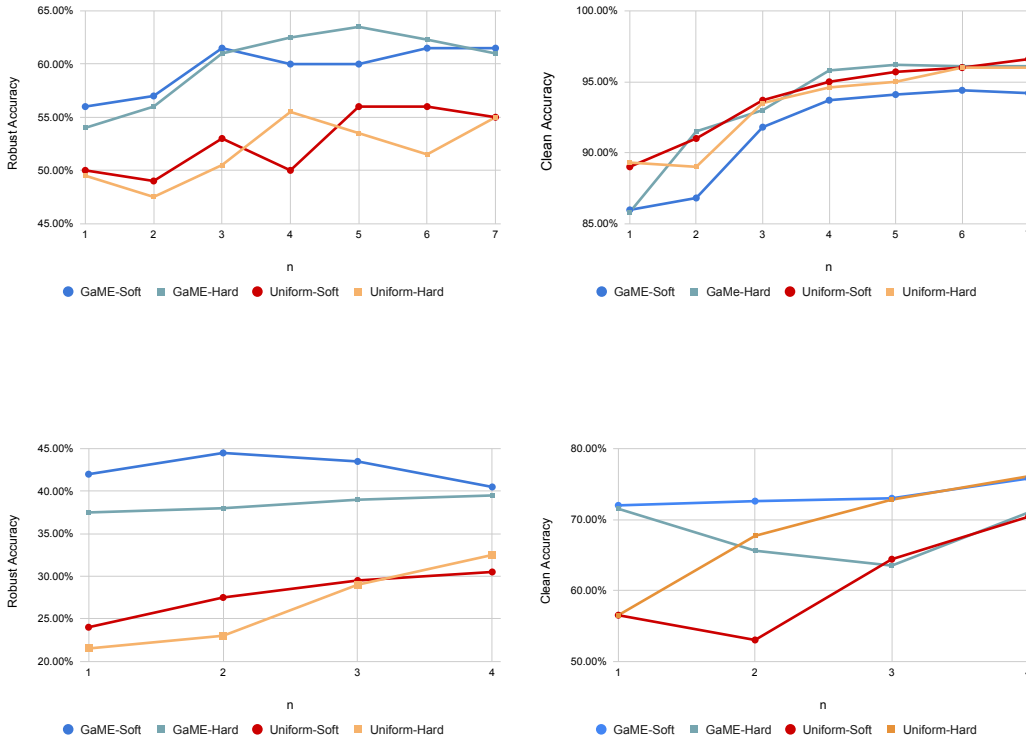


Figure 3: (Top: CIFAR-10; Bottom: Tiny ImageNet) Here GaME-Soft and GaME-Hard denote GaME generated ensembles utilizing  $f_s$  and  $f_h$  as voting functions respectively. Additionally Uniform-Soft and Uniform-Hard denote ensemble defenses with uniform mixed strategies utilizing  $f_s$  and  $f_h$  as voting functions respectively. We plot both the robust (Left) and clean (Right) accuracy of all ensembles as a function of  $n$ , the maximum voting ensemble size.

In Figure 3 we compare the robust and clean accuracy for GaME generated ensemble defenses and those with uniform probability distributions over all defender strategies. We further expand the study to consider all values of  $1 \leq n \leq |D|$  for each dataset. The results show that the GaME framework clearly improves upon the robustness of the uniform probability distribution ensemble while maintaining a high level of clean accuracy. The results additionally show that the clean and robust accuracy both trend in a positive direction as one increases  $n$ .

---

We provide more detailed, numerical results for these figures in the supplementary material. Additionally, we provide studies of the effects of  $N$  (the sample number) on  $\text{GaME}_n$  generated ensembles along with an analysis of the computational cost of the framework.

## 8 CONCLUSION

The field of adversarial machine learning has begun to cycle between new defenses, followed by new specific attacks that break those defenses, followed by even more defenses. In this paper, we seek to go beyond this cat and mouse dynamic. We consider adversarial defense transferability, multi-model attacks and a game theoretic framework for compositional adversarial machine learning.

In terms of specific contributions, we develop two new white-box attacks, the Momentum Iterative Method over Expectation (MIME) for attacking single randomized defenses and Auto Expectation Self-Attention Gradient Attack (AE-SAGA) for dealing with a combination of randomized and non-randomized defenses. We are the first to show the transferability of adversarial examples generated by MIM, APGD, MIME and AE-SAGA on state-of-the-art defenses like FAT, BaRT and TiT.

Lastly, and most importantly, we develop a game theoretic framework for determining the optimal attack and defense strategy. Any newly proposed defense or attack can be easily integrated into our framework. Using a set of state-of-the-art attacks and defenses we demonstrate that our game theoretic framework can create a compositional defense that achieve a 13.5% increase in robustness on CIFAR-10 and a 33.88% increase in robustness on Tiny ImageNet using a multi-defense, mixed Nash strategy (as opposed to using the best single defense). Both compositional defenses for CIFAR-10 and Tiny-ImageNet also come with higher a clean accuracy than the most robust single defenses.

---

## REFERENCES

- Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4190–4197, 2020.
- Alexandre Araujo, Laurent Meunier, Rafael Pinot, and Benjamin Negrevergne. Advocating for multiple defense strategies against adversarial examples. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 165–177. Springer, 2020.
- Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International conference on machine learning*, pp. 284–293. PMLR, 2018.
- Maria-Florina Balcan, Rattana Pukdee, Pradeep Ravikumar, and Hongyang Zhang. Nash equilibria and pitfalls of adversarial training in adversarial robustness games. *arXiv preprint arXiv:2210.12606*, 2022.
- Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pp. 3–14, 2017.
- Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pp. 2206–2216. PMLR, 2020.
- Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.
- Wei Fang, Zhaofei Yu, Yanqi Chen, Tiejun Huang, Timothée Masquelier, and Yonghong Tian. Deep residual learning in spiking neural networks. *Advances in Neural Information Processing Systems*, 34:21056–21069, 2021.
- Samuel G Finlayson, John D Bowers, Joichi Ito, Jonathan L Zittrain, Andrew L Beam, and Isaac S Kohane. Adversarial attacks on medical machine learning. *Science*, 363(6433):1287–1289, 2019.
- Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.
- Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In *European conference on computer vision*, pp. 491–507. Springer, 2020.
- Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). URL <http://www.cs.toronto.edu/~kriz/cifar.html>.
- Trung Le, Anh Tuan Bui, Le Minh Tri Tue, He Zhao, Paul Montague, Quan Tran, and Dinh Phung. On global-view based defense via adversarial attack and defense risk guaranteed bounds. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 11438–11460. PMLR, 28–30 Mar 2022. URL <https://proceedings.mlr.press/v151/le22c.html>.
- Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.
- Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770*, 2016.
- Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

- 
- Kaleel Mahmood, Deniz Gurevin, Marten van Dijk, and Phuoung Ha Nguyen. Beware the black-box: On the robustness of recent defenses to adversarial examples. *Entropy*, 23(10):1359, 2021a.
- Kaleel Mahmood, Rigel Mahmood, and Marten Van Dijk. On the robustness of vision transformers to adversarial examples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7838–7847, 2021b.
- Kaleel Mahmood, Phuoung Ha Nguyen, Lam M. Nguyen, Thanh Nguyen, and Marten Van Dijk. Besting the black-box: Barrier zones for adversarial example defense. *IEEE Access*, 10:1451–1474, 2022. doi: 10.1109/ACCESS.2021.3138966.
- Pratyush Maini, Eric Wong, and Zico Kolter. Adversarial robustness against the union of multiple perturbation models. In *International Conference on Machine Learning*, pp. 6640–6650. PMLR, 2020.
- Laurent Meunier, Meyer Scetbon, Rafael B Pinot, Jamal Atif, and Yann Chevaleyre. Mixed nash equilibria in the adversarial examples game. In *International Conference on Machine Learning*, pp. 7677–7687. PMLR, 2021.
- John Nash. Non-cooperative games. *Annals of Mathematics*, 54(2):286–295, 1951. ISSN 0003486X. URL <http://www.jstor.org/stable/1969529>.
- Ambar Pal and René Vidal. A game theoretic analysis of additive adversarial attacks and defenses. *Advances in Neural Information Processing Systems*, 33:1345–1355, 2020.
- Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. In *International Conference on Machine Learning*, pp. 4970–4979. PMLR, 2019.
- Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016.
- Rafael Pinot, Raphael Ettetdgui, Geovani Rizk, Yann Chevaleyre, and Jamal Atif. Randomization matters how to defend against strong adversarial attacks. In *International Conference on Machine Learning*, pp. 7717–7727. PMLR, 2020.
- Adnan Qayyum, Muhammad Usama, Junaid Qadir, and Ala Al-Fuqaha. Securing connected & autonomous vehicles: Challenges posed by adversarial machine learning and the way forward. *IEEE Communications Surveys & Tutorials*, 22(2):998–1026, 2020. doi: 10.1109/COMST.2020.2975048.
- Edward Raff, Jared Sylvester, Steven Forsyth, and Mark McLean. Barrage of random transforms for adversarially robust defense. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6521–6530, 2019. doi: 10.1109/CVPR.2019.00669.
- Nitin Rathi and Kaushik Roy. Diet-snn: A low-latency spiking neural network with direct input encoding and leakage and threshold optimization. *IEEE Transactions on Neural Networks and Learning Systems*, 2021a.
- Nitin Rathi and Kaushik Roy. Diet-snn: A low-latency spiking neural network with direct input encoding and leakage and threshold optimization. *IEEE Transactions on Neural Networks and Learning Systems*, 2021b.
- Sailik Sengupta, Tathagata Chakraborti, and Subbarao Kambhampati. Mtdeep: boosting the security of deep neural nets against adversarial attacks with moving target defense. In *Workshops at the thirty-second AAAI conference on artificial intelligence*, 2018.
- Chawin Sitawarin, Zachary J Golan-Strieb, and David Wagner. Demystifying the adversarial robustness of random transformation defenses. In *International Conference on Machine Learning*, pp. 20232–20252. PMLR, 2022.
- Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *Advances in neural information processing systems*, (33), 2020.

- 
- Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations*, 2019.
- Chang Xiao and Changxi Zheng. One man’s trash is another man’s treasure: Resisting adversarial examples by adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 412–421, 2020.
- Nuo Xu, Kaleel Mahmood, Haowen Fang, Ethan Rathbun, Caiwen Ding, and Wujie Wen. Securing the spike: On the transferability and security of spiking neural networks to adversarial examples. *arXiv preprint arXiv:2209.03358*, 2022.
- Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pp. 7472–7482. PMLR, 2019.
- Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. Attacks which do not kill training make adversarial learning stronger. In *International conference on machine learning*, pp. 11278–11287. PMLR, 2020.

## 9 SUPPLEMENTARY MATERIAL

In the supplementary material we provide additional studies and results that further explore the results found in the main body of the paper. In Subsection 9.1 we provide numerical results and further analysis for the transferability experiments found in 4. In Subsection 6 we provide a more explicit attack setup for the experiments we performed throughout the paper. In Subsection 9.2 we provide further studies and analysis upon the GaME framework including comparisons to uniform probability distribution ensemble defenses, and a study on the effect of  $n$  on a  $\text{GaME}_n$  defense. In Subsection 9.3 we provide a study upon the computational cost of the  $\text{GaME}_n$  framework. In Subsection 9.4 we provide a study the effect of the number of samples,  $N$ , used to approximate the game matrix upon the robust accuracy of a  $\text{GaME}_n$  ensemble. In Subsection 9.5 we provide explicit forms for the linear programs one must solve to create a  $\text{GaME}_n$  defense or attack ensemble. In Subsection 9.6 we provide a brief study upon the effectiveness of AE-SAGA when attacking 3 defenses simultaneously. Lastly, in Subsection 9.7 we provide an approximate form for the update formula of MIME seen in Equation 5.

### 9.1 TRANSFERABILITY EXPERIMENTS

Transferability Between Defenses (CIFAR-10)									
Attacked	B1	B5	B10	RF	VF	ST	SB	BVT	VRT
B1	-	45.80%	67.20%	99.60%	99.20%	84.10%	93.10%	95.90%	89.90%
B5	4.30%	-	44.40%	98.70%	97.90%	71.30%	90.50%	89.70%	89.00%
B10	18.60%	40.00%	-	83.20%	90.80%	66.50%	77.50%	82.50%	70.20%
RF	91.80%	88.40%	82.20%	-	87.90%	68.60%	69.80%	80.20%	66.60%
VF	52.40%	55.70%	55.70%	86.00%	-	63.00%	62.90%	21.70%	74.00%
ST	91.50%	90.50%	89.40%	98.90%	98.90%	-	91.80%	91.80%	91.30%
SB	92.20%	89.10%	86.30%	93.90%	95.20%	64.90%	-	82.90%	84.70%
BVT	60.40%	69.90%	75.10%	98.20%	95.60%	78.10%	90.00%	-	90.80%
VRT	86.80%	83.40%	89.80%	82.30%	87.20%	76.00%	71.00%	78.80%	-

Table 4: Full numerical results for the transfer study shown pictorially in Figure 1

Previously we gave a pictorial representation of the transferability results for CIFAR-10. Above we include the full results from our transferability study in Table 4. Each of the defense names have been abbreviated in the interest of space: Bn is BaRT-n, RF is the FAT trained ResNet-164, VF is the FAT trained ViT-L, ST is the transfer SNN, SB is the Back-Prop SNN, VRT is the Vgg + ResNet TiT defense, and BVT is the ViT+BiT TiT defense.

We first chose 1000 class-wise balanced, clean images from the testing set of CIFAR-10. We additionally constrained these 1000 samples to be those which are correctly identified by every model in the table. For random transform defenses, each image is classified correctly with a probability of at least 98%. We then attacked each defense with APGD, every randomized defense with MIME, and every non-randomized defense with MIM. For APGD, MIM, and MIME the parameters can be seen in Subsection 6. From each defense we chose the adversarial samples generated by the attack with the highest attack success rate to be used in the transfer study.

**Analysis of Results** From the transferability table it becomes clear that there is generally very low transferability between each pair of defenses which use different classifier architectures. For instance, attacks generated on the BiT based BaRT models have a very low level of transferability with the ViT based FAT-ViT and TiT defense using BiT+ViT. In contrast to this, defenses which share architectures, unsurprisingly, have relatively high levels of transferability. One notable example of this is the attacks generated on the FAT-ViT models when transferred to the TiT defense using BiT-ViT. There are some exceptions to this idea however, as the BaRT models seem to be highly susceptible to attacks generated on the ViT based defenses such as those generated on the FAT trained ViT-L.

## 9.2 FULL EXPERIMENTAL RESULTS FOR GAME

n	G-Soft $r^*$	G-Soft Robust	G-Soft Clean	G-Hard $r^*$	G-Hard Robust	G-Hard Clean
1	56.00%	56.00%	85.96%	57.20%	54.00%	85.80%
2	57.00%	57.00%	86.80%	60.00%	56.00%	91.50%
3	61.30%	61.50%	91.80%	61.50%	61.00%	93.00%
4	61.70%	60.00%	93.70%	62.10%	62.50%	95.80%
5	61.80%	60.00%	94.10%	62.30%	63.50%	96.20%
6	61.84%	61.50%	94.40%	62.30%	62.30%	96.10%
7	61.84%	61.50%	94.20%	62.30%	61.00%	96.10%

n	U-Hard Robust	U-Hard Clean	U-Soft Robust	U-Soft Clean
1	49.50%	89.30%	50.00%	89.00%
2	47.50%	89.00%	49.00%	91.00%
3	50.50%	93.50%	53.00%	93.70%
4	55.50%	94.60%	50.00%	95.00%
5	53.50%	95.00%	56.00%	95.70%
6	51.50%	96.00%	56.00%	96.00%
7	55.00%	96.00%	55.00%	96.60%

Table 5: (CIFAR-10) Full numerical results for Figure 3. Here G-Hard and G-Soft refer to GaME generated ensembles utilizing  $F_h$  and  $F_s$  as their voting functions respectively. Similarly, U-Hard and U-Soft represent ensemble defenses with uniform probability distributions utilizing  $F_h$  and  $F_s$  as their voting functions respectively

n	G-Soft $r^*$	G-Soft Robust	G-Soft Clean	G-Hard $r^*$	G-Hard Robust	G-Hard Clean
1	36.90%	42.00%	72.00%	36.90%	39.50%	71.52%
2	41.75%	42.00%	72.60%	34.00%	40.00%	65.60%
3	42.10%	43.00%	73.00%	34.70%	38.50%	63.50%
4	32.00%	28.00%	75.80%	33.00%	35.50%	71.00%

n	U-Hard Robust	U-Hard Clean	U-Soft Robust	U-Soft Clean
1	21.50%	56.50%	24.00%	56.50%
2	23.00%	53.00%	27.50%	67.70%
3	29.00%	64.40%	29.50%	72.81%
4	32.50%	0.704	30.50%	76.11%

Table 6: (Tiny ImageNet) Full numerical results for Figure 3. Here the notation is consistent with Table 5



Full GaME <sub>1</sub> Table (CIFAR-10)							
	B1	B5	RF	VF	ST	SB	BVT
APGD(RF)	90.75%	86.75%	50.25%	81.00%	59.13%	58.13%	74.88%
APGD(SB)	90.25%	89.75%	79.63%	88.88%	55.25%	1.50%	79.50%
APGD(ST)	91.25%	88.88%	84.13%	93.50%	0.00%	81.13%	88.12%
APGD(VF)	54.37%	54.75%	72.50%	29.50%	55.37%	57.00%	21.88%
MIME(B1)	3.13%	44.62%	84.38%	94.00%	74.88%	82.75%	92.87%
MIME(B5)	4.62%	16.38%	83.50%	92.37%	63.50%	81.37%	87.00%
MIME(VBT)	58.13%	68.87%	83.75%	91.50%	69.87%	80.75%	9.75%
S(B1,B5)	5.00%	37.87%	83.87%	93.87%	74.25%	83.25%	90.87%
S(B1,RF)	11.25%	64.38%	59.13%	89.00%	69.37%	75.25%	87.50%
S(B1,SB)	8.87%	58.37%	80.25%	90.12%	61.25%	11.13%	85.62%
S(B1,ST)	14.75%	71.50%	83.00%	93.25%	0.13%	81.50%	89.12%
S(B1,VBT)	6.25%	55.37%	84.13%	93.50%	70.75%	83.37%	35.50%
S(B1,VF)	5.00%	51.38%	80.00%	71.75%	68.37%	76.00%	76.88%
S(B5,RF)	17.62%	36.13%	59.38%	88.50%	65.00%	73.62%	85.50%
S(B5,SB)	16.38%	29.00%	80.00%	90.63%	55.37%	9.88%	82.00%
S(B5,ST)	58.75%	59.25%	83.25%	93.50%	0.25%	82.75%	87.12%
S(B5,VBT)	15.63%	34.38%	83.87%	93.37%	69.50%	82.50%	32.75%
S(B5,VF)	12.88%	30.50%	79.75%	68.87%	65.75%	75.62%	70.63%
S(RF,SB)	88.50%	84.88%	55.75%	82.37%	54.00%	3.00%	76.38%
S(RF,ST)	92.13%	90.12%	70.00%	91.37%	0.13%	76.12%	83.75%
S(RF,VBT)	70.50%	76.88%	62.38%	83.37%	61.25%	69.50%	6.12%
S(RF,VF)	63.00%	61.00%	57.13%	39.25%	53.87%	56.00%	31.87%
S(SB,ST)	92.75%	90.50%	82.50%	92.50%	0.00%	33.87%	86.75%
S(SB,VBT)	78.00%	78.63%	80.00%	88.75%	53.75%	6.00%	15.00%
S(ST,VBT)	89.88%	88.62%	83.87%	93.63%	0.37%	81.37%	23.75%
S(VF,SB)	83.00%	82.37%	77.50%	75.62%	54.13%	0.88%	68.50%
S(VF,ST)	86.38%	83.50%	82.50%	86.50%	0.63%	80.13%	78.75%
S(VF,VBT)	54.75%	62.00%	76.25%	55.13%	60.12%	68.50%	3.00%

Table 7: Here we present the full results from GaME<sub>1</sub> performed on CIFAR-10. This extensive study shows us that, for any attack, there is going to be a defense that can counter it, and vice versa. For instance, MIME(B5) is extremely effective against BaRT-1 and BaRT-5, yet it fails to fool the ViT-FAT model 92% of the time. Here we use S(A,B) to denote the AE-SAGA attack run against defenses A and B

Defender Mixed Nash Strategy (CIFAR-10, GaME <sub>1</sub> )							
Defense	B1	B5	RF	VF	ST	SB	BVT
$\lambda^D$	0	.16	.75	.02	0	.07	0
Min Robust	3%	16%	50%	29%	0%	1%	3%
Defender Mixed Nash Strategy (Tiny ImageNet, GaME <sub>1</sub> )							
Defense	B1	B5	RF	VF	ST	SB	BVT
$\lambda^D$	0.152	0.443	-	0.243	-	-	0.162
Min Robust	4.75%	10.62%	-	3.87%	-	-	1.87%

Table 8: GaME<sub>1</sub> Defender results for CIFAR-10 and Tiny-ImageNet. Full numerical results corresponding to the robust accuracy for the individual attacks (APGD, MIM, MIME and AE-SAGA) and the clean accuracy of different defense are given in our supplementary material.

Attacker Mixed Nash Strategy (CIFAR-10, GaME <sub>1</sub> )				
Attack	S(B5,RF)	APGD(RF)	S(RF,VF)	S(RF,SB)
$\lambda^A$	0.265	0.051	0.618	0.066
Max Robust	89%	91%	63%	89%
Attacker Mixed Nash Strategy (Tiny ImageNet, GaME <sub>1</sub> )				
Attack	S(B1,BVT)	S(B1,VF)	MIME(B5)	S(B5,BVT)
$\lambda^A$	0.224	0.167	0.021	0.588
Max Robust	36%	56%	62%	38%

Table 9:  $GaME_1$  Attacker results for CIFAR-10 and Tiny-ImageNet. Attacks which have a probability of 0 in the mixed strategy are not represented here.

Robust Accuracy of Ensemble Generated By GaME <sub>1</sub> (CIFAR-10)						
APGD(RF)	APGD(SB)	APGD(ST)	APGD(VF)	MIME(B1)	MIME(B5)	MIME(BVT)
57.00%	80.50%	89.50%	66.50%	83.50%	72.50%	84.50%
S(B5,SB)	S(B5,ST)	S(B5,BVT)	S(B5,VF)	S(RF,SB)	S(RF,ST)	S(RF,BVT)
74.00%	80.00%	76.00%	73.00%	58.00%	76.00%	60.50%
S(B1,B5)	S(B1,RF)	S(B1,SB)	S(B1,ST)	S(B1,BVT)	S(B1,VF)	S(B5,RF)
74.50%	58.50%	76.50%	83.00%	81.50%	74.00%	57.00%
S(RF,VF)	S(SB,ST)	S(SB,BVT)	S(ST,BVT)	S(VF,SB)	S(VF,ST)	S(VF,BVT)
59.00%	82.50%	81.50%	86.00%	76.00%	83.00%	77.00%

Table 10: Here we list the robust accuracy of our ensemble generated by  $GaME_1$  when evaluated on CIFAR-10 attack samples that were not used in the formulation of the game’s linear program. If the attacker plays a best response strategy to the ensemble like APGD(RF), they can expect to lower the ensemble’s robust accuracy to 57%. This is very close to the value  $r^* = .573$  for the expected, guaranteed robustness of the ensemble. We can expect that with a larger sample number, i.e. generating more than 1000 adversarial examples per attack, that these values will get closer due to the law of large numbers.

Robust Accuracy of Ensemble With Uniform Weights in GaME <sub>1</sub> (CIFAR-10)						
APGD(RF)	APGD(SB)	APGD(ST)	APGD(VF)	MIME(B1)	MIME(B5)	MIME(VBT)
71.00%	69.50%	72.00%	51.50%	70.00%	59.00%	68.00%
S(B5,SB)	S(B5,ST)	S(B5,VBT)	S(B5,VF)	S(RF,SB)	S(RF,ST)	S(RF,VBT)
50.50%	66.00%	56.50%	56.00%	60.50%	76.50%	63.50%
S(B1,B5)	S(B1,RF)	S(B1,SB)	S(B1,ST)	S(B1,VBT)	S(B1,VF)	S(B5,RF)
65.00%	63.00%	55.00%	64.00%	57.50%	60.50%	57.50%
S(RF,VF)	S(SB,ST)	S(SB,VBT)	S(ST,VBT)	S(VF,SB)	S(VF,ST)	S(VF,VBT)
49.50%	61.50%	60.00%	69.50%	64.50%	74.00%	49.00%

Table 11: Here we show the robust accuracy for an ensemble defense using single model predictions and a uniform weighting for all of the defenses in its ensemble.

Full $GaME_1$ Table (Tiny ImageNet)				
Attack	B1	B5	VF	BVT
MIME(B1)	4.75%	34.63%	31.13%	66.00%
MIME(B5)	20.75%	10.62%	29.88%	62.38%
APGD(VF)	41.13%	36.00%	9.00%	5.75%
MIME(BVT)	51.75%	49.25%	29.63%	1.87%
S(B1,B5)	5.37%	20.62%	30.88%	66.50%
S(B1,VF)	4.88%	31.25%	5.87%	56.75%
S(B1,BVT)	5.12%	36.13%	29.50%	8.00%
S(B5,BVT)	38.37%	22.75%	30.88%	11.25%
S(B5,VF)	37.87%	19.63%	8.62%	63.12%
S(VF,BVT)	47.88%	44.87%	3.87%	3.00%

Table 12: Here we provide the full utility matrix for the defender when creating a  $GaME_1$  defense on Tiny ImageNet.

Robust Accuracy of Ensemble Generated By $GaME_1$ (Tiny ImageNet)				
MIME(B1)	MIME(B5)	APGD(VF)	MIME(BVT)	S(B1,B5)
35.00%	26.50%	44.00%	47.00%	34.50%
S(B1,VF)	S(B1,BVT)	S(B5,VF)	S(B5,BVT)	S(VF,BVT)
27.00%	28.00%	31.00%	20.50%	32.50%

Table 13: Here we list the robust accuracy of our ensemble generated by  $GaME_1$  when evaluated on Tiny ImageNet attack samples that were not used in the formulation of the game’s linear program.

### 9.3 STUDY OF COMPUTATIONAL COST

As one increases the value of  $n$  in a  $GaME_n$  defense, the number of possible choices for the defender grows in accordance to the binomial coefficient,  $\binom{|D|}{n}$ . Thus, here we will provide a brief study on the effect of  $n$  on the computational complexity of creating a  $GaME_n$  defense.

The computation time for forming the game-matrix largely depends on the time needed to compute the predictions of each of the defenses for each of the attacks, as seen in Figure 5. We can save a large amount of time by running each set of samples,  $s_i$ , through each defense  $d \in D$  once, receiving output  $y_{i,d}$  for each sample, defense pair. To get the robust accuracy of  $U \subset D$  when evaluating samples  $s_i$  we can substitute  $y_{i,d}$  for  $d(s_i)$  in the computation of  $F_h(s_i, U)$  or  $F_s(s_i, U)$  Equation 19 Equation 20. This means that we do not need to perform a number of model evaluations that scales with  $n$ .

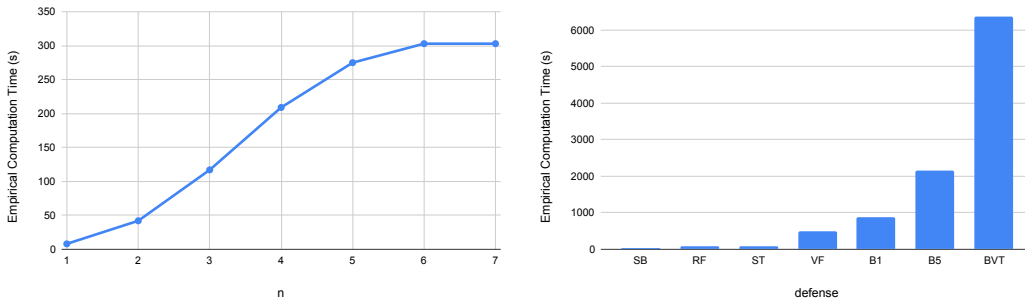


Figure 4: (CIFAR-10) Computational cost of creating the game matrix and solving the associated linear program for a  $\text{GaME}_n$  ensemble as a function of  $n$ . Model prediction time is not considered in the calculation as it does not depend on  $n$ .

Figure 5: (CIFAR-10) Computational cost of evaluating every single-model defense in the study on all 22,400 adversarial samples generated for creating the game matrix of a  $\text{GaME}_n$  ensemble. The total computation time amounts to 2.9 hours of which 3% of the time, 5 minutes, is spent evaluating each defense subset, as described above, and solving the game matrix for  $\text{GaME}_7$ .

The computational cost for evaluating the BaRT defense are high since a series of random transformations must be applied to the images before a prediction can be made. These transformations run on CPU and are run in parallel on a per-sample basis in order to speed up computation time. The computational cost for evaluating the BiT-ViT Trash is Treasure defense is the highest since it requires running a 13 step PGD attack against a large BiT model for each sample before it is given to the main ViT classifier.

These experiments were run on a computer with the following specifications: Intel Core i9-10900K CPU @ 3.70GHz, Nvidia RTX3080 12Gb, and 64Gb RAM.

#### 9.4 STUDY ON THE EFFECTS OF SAMPLE NUMBER FOR GAME

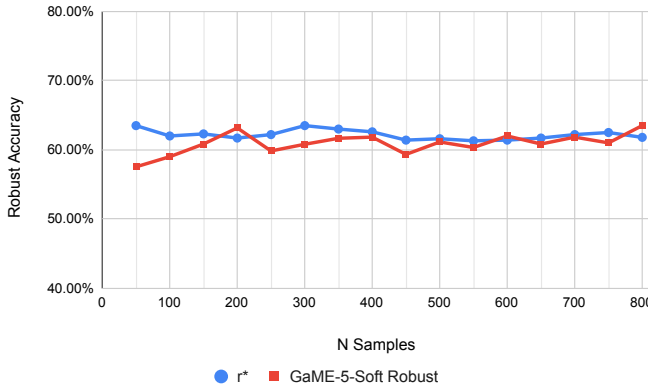


Figure 6: (CIFAR-10) Study on the effect of  $N$  on the expected robust accuracy,  $r^*$ , in blue, and the empirical robust accuracy, in red. Results are averaged over 3 trials and measured with respect to 200 samples from each attack in the study, as described in Section 6.

Here we provide a brief study on the effects of  $N$  on the effectiveness of a GaME defense. We choose  $\text{GaME}_5$  with voting function  $F_s$  since it had the highest robust accuracy in our experiments. For most values of  $N$  we analyzed the difference between  $r^*$  and the empirical robustness was very small, in fact the average difference over all  $N$  was only 1.73%. The results show that one can

achieve reasonable results with  $N$  as low as 150 on CIFAR-10. This implies that one can create a GaME ensemble defense without the large computational cost of running dozens of attacks against their defenses. However, it is likely that this will not hold for more complicated data sets with a larger number of classes such as ImageNet or even CIFAR-100.

### 9.5 LINEAR PROGRAMS FOR SOLVING GAME

Here we present the explicit linear program for solving GaME as the attacker. Let  $O_A = (\lambda_{a_1}^A, \dots, \lambda_{a_{|A|}}^A, r^*)$  be the row vector containing the elements of  $\lambda^A$  and  $r^*$ . Additionally let  $\hat{0}$  denote the zero vector. The attacker must then solve the following linear program:

$$\text{Subject to: } \begin{pmatrix} \max ( 0 \quad \cdots \quad 0 \quad 1 ) O_A^T \\ -r_{d_1, a_1} \quad -r_{d_1, a_2} \quad \cdots \quad 1 \\ -r_{d_2, a_1} \quad -r_{d_2, a_2} \quad \cdots \quad 1 \\ \vdots \quad \vdots \quad \ddots \quad \vdots \\ 1 \quad 1 \quad \cdots \quad 0 \end{pmatrix} O_A^T \leq \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \quad (22)$$

and:  $O_A \geq \hat{0}$

Here we show the explicit linear program for solving GaME as the defender. For convenience let  $O_D = (\lambda_{d_1}^D, \dots, \lambda_{d_{|D|}}^D, r^*)$  be the row vector containing the elements of  $\lambda^D$  and  $r^*$ . The defender must solve the following linear program:

$$\text{Subject to: } \begin{pmatrix} \max ( 0 \quad \cdots \quad 0 \quad 1 ) O_D^T \\ -r_{d_1, a_1} \quad -r_{d_2, a_1} \quad \cdots \quad 1 \\ -r_{d_1, a_2} \quad -r_{d_2, a_2} \quad \cdots \quad 1 \\ \vdots \quad \vdots \quad \ddots \quad \vdots \\ 1 \quad 1 \quad \cdots \quad 0 \end{pmatrix} O_D^T \leq \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \quad (23)$$

and:  $O_D \geq \hat{0}$

Due to the nature of the dual problem in linear programming, solving this problem will result in the same value for  $r^*$  as was found in the primal problem presented in section 4.

### 9.6 AE-SAGA AGAINST 3 OR MORE DEFENSES

For our implementation of GaME<sub>n</sub> we only performed AE-SAGA attacks against 2 model ensembles, this was for two reasons: AE-SAGA can have high computational cost, attacking size 3 ensembles increases the total number of experiments needed exponentially, and AE-SAGA does not scale well to more than 2 defenses. We provide an example of this below:

Attack	RF	VRT	SB	FR+VRT+SB
S(FR,VRT,SB)	.633	.316	.455	.546

### 9.7 APPROXIMATING MOMENTUM ITERATIVE METHOD OVER EXPECTATION ATTACK

In practice  $g^{(i)}$  is approximated using  $N$  Monte Carlo samples per input  $x$ :

$$g^{(i)} \approx \gamma g^{(i-1)} + \left( \frac{1}{N} \sum_{j=0}^N \frac{\partial L}{\partial t_j(x_{adv}^{(i)})} \right) \quad (24)$$