

Query-Efficient Black-box Adversarial Attacks Guided by a Transfer-based Prior

Yinpeng Dong[†], Shuyu Cheng[†], Tianyu Pang, Hang Su, and Jun Zhu[‡], *Senior Member, IEEE*

Abstract—Adversarial attacks have been extensively studied in recent years since they can identify the vulnerability of deep learning models before deployed. In this paper, we consider the black-box adversarial setting, where the adversary needs to craft adversarial examples without access to the gradients of a target model. Previous methods attempted to approximate the true gradient either by using the transfer gradient of a surrogate white-box model or based on the feedback of model queries. However, the existing methods inevitably suffer from low attack success rates or poor query efficiency since it is difficult to estimate the gradient in a high-dimensional input space with limited information. To address these problems and improve black-box attacks, we propose two prior-guided random gradient-free (PRGF) algorithms based on biased sampling and gradient averaging, respectively. Our methods can take the advantage of a transfer-based prior given by the gradient of a surrogate model and the query information simultaneously. Through theoretical analyses, the transfer-based prior is appropriately integrated with model queries by an optimal coefficient in each method. Extensive experiments demonstrate that, in comparison with the alternative state-of-the-arts, both of our methods require much fewer queries to attack black-box models with higher success rates.

Index Terms—Adversarial examples, black-box attacks, zeroth-order optimization, query efficiency, transferability.

1 INTRODUCTION

DESPITE the significant success of deep learning models on various tasks [1], the security and reliability of these models have been challenged in the presence of adversarial examples [2], [3], [4], [5]. The maliciously crafted adversarial examples aim at causing misclassification of a target model by applying human imperceptible perturbations to natural examples. It has garnered increasing attention to study the generation of adversarial examples (i.e., adversarial attack), which is indispensable to discover the weaknesses of deep learning algorithms [3], [6], [7]. Adversarial attacks therefore serve as a surrogate to evaluate robustness [8], [9], [10], and consequently contribute to the design of more robust deep learning models [4], [9], [11].

Adversarial attacks are predominantly categorized into *white-box* attacks and *black-box* attacks according to different accessibility to the target model. Getting access to the model architecture, parameters and especially gradients, an adversary can adopt various gradient-based methods [4], [5], [8], [9] to generate adversarial examples under the white-box setting, such as the fast gradient sign method (FGSM) [4], projected gradient descent method (PGD) [9], etc. By contrast, under the more challenging black-box adversarial setting, the adversary has no or limited knowledge about the target model, and therefore needs to generate adversarial examples without any gradient information. In various real-world applications, the black-box setting is more practical than the white-box counterpart [12], [13].

Tremendous efforts have been made to develop black-box adversarial attacks [12], [13], [14], [15], [16], [17], [18], [19], [20]. A common idea of these techniques is to utilize an approximate gradient instead of the true but unknown gradient for generating adversarial examples. The approximate gradient can either stem from the gradient of a surrogate white-box model (termed as *transfer-based* attacks) or be numerically estimated by the zeroth-order optimization algorithms (termed as *query-based* attacks).

In transfer-based attacks, adversarial examples produced for a surrogate model are probable to remain adversarial for the target model due to the transferability [21], [22]. Recent methods have been introduced to improve the transferability by adopting a momentum optimizer [16] or performing input augmentations [7], [23]. However, the success rate of transfer-based attacks is still far from satisfactory. This is because that there lacks an adjustment procedure when the gradient of the surrogate model points to a non-adversarial region of the target model. In query-based attacks, the true gradient can be estimated by various methods, such as finite difference [15], [17], random gradient estimation [18] and natural evolution strategies [12]. Although these methods usually result in a higher attack success rate compared with the transfer-based attack methods [15], [17], they inevitably require a tremendous number of queries to perform a successful attack. The query inefficiency primarily comes from the under-utilization of priors, since the current methods are nearly optimal to estimate the gradient [19].

To overcome the aforementioned problems and improve black-box adversarial attacks, we propose two **prior-guided random gradient-free (PRGF)** algorithms based on biased sampling (BS) and gradient averaging (GA), respectively, which can utilize a transfer-based prior for query-efficient black-box attacks. The transfer-based prior originated from the gradient of a surrogate white-box model contains abun-

• [†]Y. Dong and S. Cheng contribute equally.
 • [‡]J. Zhu is the corresponding author.
 • The authors are with the Dept. of Comp. Sci. and Tech., Institute for AI, Tsinghua-Bosch Joint ML Center, THBI Lab, BNRist Center, Tsinghua University, Beijing 100084, China; Pazhou Lab, Guangzhou, 510330, China. Email: {dongyinpeng@mail, chengsy18@mails, pty17@mails, suhangss@mail, dcszj@mail}.tsinghua.edu.cn

dant prior knowledge of the true gradient. Despite the same goal, the two proposed methods utilize the transfer gradient in different ways. Specifically, our first method, abbreviated as PRGF-BS, provides a gradient estimate by querying the target model with random samples that are biased towards the transfer gradient and acquiring the corresponding loss values. Our second method, denoted as PRGF-GA, performs a weighted average of the transfer gradient and the gradient estimate provided by the ordinary random gradient-free (RGF) method [24], [25], [26]. Under the gradient estimation framework, we provide theoretical analyses on deriving the optimal coefficients of controlling the strength of the transfer gradient in both algorithms.

Furthermore, our methods are flexible to integrate other prior information. As a concrete example, we incorporate the commonly used *data-dependent prior* [19] into our algorithms along with the transfer-based prior. We also provide theoretical analyses on how to embrace both priors appropriately. Besides, we extend our methods to the scenario that multiple surrogate models are available, as studied in [16], [22], in which we can further boost the attack performance with a more effective transfer-based prior. Extensive experiments demonstrate that both of our methods significantly outperform the previous state-of-the-art methods in terms of black-box attack success rate and query efficiency, verifying the superiority of our algorithms for black-box attacks.

This paper substantially extends and improves the conference version [27]. We additionally propose a new PRGF algorithm based on gradient averaging and integrate it with the data-dependent prior. We also consider the scenario that multiple surrogate models are available and provide a subspace projection method to extract a more effective transfer-based prior. Besides, we conduct additional experiments by comparing more methods, using different surrogate models, and considering another dataset, to show the superiority of our methods. Overall, we make the following contributions:

- 1) We propose to improve black-box adversarial attacks by incorporating a transfer-based prior given by the gradient of a surrogate model. The transfer-based prior provides abundant prior information of the true gradient due to the adversarial transferability.
- 2) We develop two prior-guided random gradient-free (PRGF) algorithms to utilize the transfer-based prior, based on biased sampling and gradient averaging, respectively. Theoretical analyses derive the optimal coefficients of integrating the transfer-based prior.
- 3) We demonstrate the flexibility of our algorithms by incorporating the widely used data-dependent prior and considering multiple surrogate models.
- 4) We validate that the proposed methods can improve the success rate of black-box adversarial attacks and reduce the requisite numbers of queries significantly compared with the state-of-the-art methods.

The rest of the paper is organized as follows. Section 2 reviews the background and related work on black-box adversarial attacks. Section 3 introduces the gradient estimation framework. Section 4 and Section 5 present the proposed PRGF algorithms, and their extensions with data-dependent priors and multiple surrogate models. Section 6 presents empirical studies. Finally, Section 7 concludes.

2 BACKGROUND

2.1 Adversarial Setup

Given a classifier $C(x)$ and an input-label pair (x, y) where $x \in \mathbb{R}^D$, the goal of attacks is to generate an adversarial example x^{adv} that is misclassified by C while the distance between the adversarial example x^{adv} and the natural one x measured by the ℓ_p norm is smaller than a threshold ϵ as

$$C(x^{adv}) \neq y, \text{ s.t. } \|x^{adv} - x\|_p \leq \epsilon. \quad (1)$$

Note that formulation (1) corresponds to an untargeted attack. We present our framework and algorithms based on untargeted attacks for clarity, while the extension to targeted ones is straightforward.

An adversarial example can be generated by solving the constrained optimization problem as

$$x^{adv} = \arg \max_{x': \|x' - x\|_p \leq \epsilon} f(x', y), \quad (2)$$

where f is a loss function on top of the classifier $C(x)$, e.g., the cross-entropy loss. Several gradient-based methods [4], [5], [8], [9], [16] have been proposed to solve this optimization problem. The typical projected gradient descent method (PGD) [9] iteratively generates adversarial examples as

$$x_{t+1}^{adv} = \Pi_{\mathcal{B}_p(x, \epsilon)}(x_t^{adv} + \eta \cdot g_t), \quad (3)$$

where $\mathcal{B}_p(x, \epsilon) = \{x' : \|x' - x\|_p \leq \epsilon\}$ denotes the ℓ_p ball centered at x with radius ϵ , Π is the projection operation, η is the step size, and g_t is the normalized gradient under the ℓ_p norm, e.g., $g_t = \frac{\nabla_x f(x_t^{adv}, y)}{\|\nabla_x f(x_t^{adv}, y)\|_2}$ under the ℓ_2 norm, and $g_t = \text{sign}(\nabla_x f(x_t^{adv}, y))$ under the ℓ_∞ norm. Those methods such as PGD require full access to the gradients of the target model, which are known as white-box attacks.

2.2 Black-box Attacks

In contrast to white-box attacks, black-box attacks have no or limited knowledge about the target model, which can be challenging yet practical in various real-world applications. We can still adopt the PGD method to generate adversarial examples, except that the true gradient $\nabla_x f(x, y)$ is usually replaced by an approximate gradient. Black-box attacks can be roughly divided into transfer-based attacks and query-based attacks. Transfer-based attacks depend on the gradient of a surrogate white-box model to generate adversarial examples, which are probable to fool the black-box model due to the transferability [21], [22]. Some query-based attacks estimate the gradient by the zeroth-order optimization methods, when the loss values could be accessed through queries. Chen et al. [15] propose to estimate the gradient at each coordinate by the symmetric difference quotient [28] as

$$\hat{g}_i = \frac{f(x + \sigma e_i, y) - f(x - \sigma e_i, y)}{2\sigma} \approx \frac{\partial f(x, y)}{\partial x_i}, \quad (4)$$

where σ is a small constant and e_i is the i -th unit basis vector. Although query-efficient mechanisms have been developed [15], [17], the coordinate-wise gradient estimation inherently leads to the query complexity being proportional to the input dimension D , which is prohibitively large with a high-dimensional input space, e.g., $D \approx 270,000$ for ImageNet [29]. To improve query efficiency, the approximated

gradient \hat{g} can be obtained by the random gradient-free (RGF) method [24], [25], [26] as

$$\hat{g} = \frac{1}{q} \sum_{i=1}^q \hat{g}_i, \text{ with } \hat{g}_i = \frac{f(x + \sigma u_i, y) - f(x, y)}{\sigma} \cdot u_i, \quad (5)$$

where $\{u_i\}_{i=1}^q$ are the random vectors sampled independently from a pre-defined distribution \mathcal{P} on \mathbb{R}^D and σ is the parameter to control the sampling variance. It can be noted that $\hat{g}_i \rightarrow u_i^\top \nabla_x f(x, y) \cdot u_i$ when $\sigma \rightarrow 0$, which is nearly an unbiased estimator of the gradient when $\mathbb{E}[u_i u_i^\top] = \mathbf{I}$ [26]. In practice, the final gradient estimator \hat{g} is averaged over q random directions to reduce the variance. [12] relies on the natural evolution strategies (NES) [30] to estimate the gradient, which is another variant of Eq. (5). The difference is that [12] conducts the antithetic sampling over a Gaussian distribution. Ilyas et al. [19] prove that these methods are nearly optimal to estimate the gradient, but the query efficiency could be improved by incorporating informative priors. They identify the time- and data-dependent priors for black-box attacks. Different from those alternative methods, our adopted transfer-based prior is more effective as shown in the experiments. Moreover, the transfer-based prior can also be used simultaneously with other priors. We demonstrate the flexibility of our algorithms by incorporating the commonly used data-dependent prior as an example.

2.3 Attacks based on both Transferability and Queries

There are also several works that utilize both the transferability of adversarial examples and the model queries for black-box attacks. A local substitute model can be trained to mimic the black-box model with a synthetic dataset, in which the labels are given by the black-box model through queries [14], [21]. Then the black-box model can be evaded by the adversarial examples crafted for the substitute model based on the transferability. A meta-model [31] can reverse-engineer the black-box model and predict its attributes (e.g., architecture, optimization procedure, and training samples) through a sequence of model queries. Given the predicted attributes of the black-box model, the attacker can find similar surrogate models, which exhibit better transferability of the generated adversarial examples against the black-box model. All of these methods use queries to obtain knowledge of the black-box model, and train/find surrogate models to generate adversarial examples, with the purpose of improving the transferability. However, we do not optimize the surrogate model, but focus on utilizing the gradient(s) of a (multiple) fixed surrogate model(s) to obtain a more accurate gradient estimate.

Although a recent work [32] also uses the gradient of a surrogate model to improve the query efficiency of black-box attacks, it focuses on a different attack scenario, where the adversary can only acquire the hard-label outputs, but we consider the adversarial setting that the loss values can be accessed. Moreover, this method controls the strength of the transfer gradient by a preset hyperparameter, but we obtain its optimal value through theoretical analyses based on the gradient estimation framework. It is worth mentioning that a similar but independent work [33] also uses surrogate gradients to improve zeroth-order optimization, but they do not apply their method to black-box adversarial attacks.

3 GRADIENT ESTIMATION FRAMEWORK

Before we delve into the details of the proposed methods, we first introduce the gradient estimation framework in this section, which builds up the foundation of our theoretical analyses.

The key problem in black-box adversarial attacks is to estimate the gradient of a target model, which can then be used to carry out gradient-based attacks. The goal of this work is to estimate the gradient $\nabla_x f(x, y)$ of the black-box model f more accurately to improve black-box attacks. We denote the gradient $\nabla_x f(x, y)$ by $\nabla f(x)$ in the sequel for notation clarity. We assume that $\nabla f(x) \neq 0$ in this paper. The objective of gradient estimation is to find the best estimator that approximates the true gradient $\nabla f(x)$ by reaching the minimum value of the loss function as

$$\hat{g}^* = \arg \min_{\hat{g} \in \mathcal{G}} L(\hat{g}), \quad (6)$$

where \hat{g} is a gradient estimator given by any estimation algorithm, \mathcal{G} is the set of all possible gradient estimators, and $L(\hat{g})$ is a loss function to evaluate the performance of the estimator \hat{g} . Specifically, we let the loss function of the gradient estimator \hat{g} be

$$L(\hat{g}) = \min_{b \geq 0} \mathbb{E} \|\nabla f(x) - b\hat{g}\|_2^2, \quad (7)$$

where the expectation is taken over the randomness of the estimation algorithm to obtain \hat{g} . We define the loss $L(\hat{g})$ to be the minimum expected squared ℓ_2 distance between the true gradient $\nabla f(x)$ and the scaled estimator $b\hat{g}$. The previous work [18] considers the expected squared ℓ_2 distance $\mathbb{E} \|\nabla f(x) - \hat{g}\|_2^2$ as the loss function, which is similar to ours. However, the value of their adopted loss function will change with different magnitudes of the estimator \hat{g} (i.e., scaling \hat{g} can cause varying loss values). In the process of generating adversarial examples, the gradient is usually normalized [4], [5], [9], indicating that the direction of the gradient estimator, instead of the magnitude, will affect the performance of attacks. Thus, we incorporate a scaling factor b in Eq. (7) and minimize the error w.r.t. b , which can neglect the impact of the magnitude on the loss of the estimator \hat{g} .

4 METHODS

In this section, we present the two proposed **prior-guided random gradient-free (PRGF)** methods, which are variants of the ordinary random gradient-free (RGF) method. Recall that in RGF, the gradient is estimated through a set of random vectors $\{u_i\}_{i=1}^q$ as in Eq. (5) with q being the total number. Directly using RGF without prior information (i.e., sampling u_i from an uninformative distribution such as a uniform distribution) will result in poor query efficiency as demonstrated in our experiments. Therefore, we propose to improve the RGF estimator by utilizing the transfer-based prior, through either biased sampling or gradient averaging.

We denote the normalized transfer gradient of a surrogate model as v such that $\|v\|_2 = 1$, and the cosine similarity between the transfer gradient and the true gradient as

$$\alpha = v^\top \overline{\nabla f(x)}, \text{ with } \overline{\nabla f(x)} = \frac{\nabla f(x)}{\|\nabla f(x)\|_2}, \quad (8)$$

where $\overline{\nabla f(x)}$ is the ℓ_2 normalization of the true gradient $\nabla f(x)$.¹ We assume that $\alpha \geq 0$ without loss of generality, since we can reassigned $v \leftarrow -v$ when $\alpha < 0$.

We will introduce the two PRGF methods in Section 4.1 and Section 4.2, respectively. As the true value of the cosine similarity α is unknown, we develop a method to estimate it efficiently, which will be introduced in Section 4.3.

4.1 PRGF with Biased Sampling

Rather than sampling the random vectors $\{u_i\}_{i=1}^q$ from an uninformative distribution as the ordinary RGF method, our first proposed method samples the random vectors that are biased towards the transfer gradient v , to fully exploit the prior information. For the gradient estimator \hat{g} in Eq. (5), we further assume that the sampling distribution \mathcal{P} is defined on the unit hypersphere in the D -dimensional input space, such that the random vectors $\{u_i\}_{i=1}^q$ drawn from \mathcal{P} satisfy $\|u_i\|_2 = 1$. Then, we can calculate the loss of the gradient estimator \hat{g} in Eq. (5) by the following theorem.

Theorem 1. (Proof in Appendix A.1) If f is differentiable at x , the loss of the gradient estimator \hat{g} defined in Eq. (5) is

$$\lim_{\sigma \rightarrow 0} L(\hat{g}) = \|\nabla f(x)\|_2^2 - \frac{(\nabla f(x)^\top \mathbf{C} \nabla f(x))^2}{(1 - \frac{1}{q})\nabla f(x)^\top \mathbf{C}^2 \nabla f(x) + \frac{1}{q}\nabla f(x)^\top \mathbf{C} \nabla f(x)}, \quad (9)$$

where σ is the sampling variance, $\mathbf{C} = \mathbb{E}[u_i u_i^\top]$ with u_i being the random vector, $\|u_i\|_2 = 1$, and q is the number of random vectors as in Eq. (5).

It can be noted from Theorem 1 that we can minimize $L(\hat{g})$ by optimizing \mathbf{C} , i.e., we can obtain an optimal gradient estimator by appropriately sampling the random vectors u_i , yielding an query-efficient adversarial attack. Given the definition of \mathbf{C} , it needs to satisfy two constraints: (1) it should be positive semi-definite; (2) its trace should be 1 since $\text{Tr}(\mathbf{C}) = \mathbb{E}[\text{Tr}(u_i u_i^\top)] = \mathbb{E}[u_i^\top u_i] = 1$.

Specifically, \mathbf{C} can be decomposed as $\sum_{j=1}^D \lambda_j v_j v_j^\top$, in which $\{\lambda_j\}_{j=1}^D$ and $\{v_j\}_{j=1}^D$ are the non-negative eigenvalues and the orthonormal eigenvectors of \mathbf{C} , satisfying $\sum_{j=1}^D \lambda_j = 1$. In our method, we propose to sample u_i that are biased towards the transfer gradient v to exploit its prior information. So we specify an eigenvector of \mathbf{C} to be v , and let the corresponding eigenvalue be a tunable coefficient. For the other eigenvalues, we set them to be equal since we do not have any prior knowledge about the other eigenvectors. To this end, we let

$$\mathbf{C} = \lambda v v^\top + \frac{1 - \lambda}{D - 1} (\mathbf{I} - v v^\top), \quad (10)$$

where $\lambda \in [0, 1]$ controls the strength of the transfer gradient that the random vectors $\{u_i\}_{i=1}^q$ are biased towards. We can easily construct a random vector with unit length while satisfying Eq. (10) as (proof in Appendix A.2)

$$u_i = \sqrt{\lambda} \cdot v + \sqrt{1 - \lambda} \cdot (\mathbf{I} - v v^\top) \xi_i, \quad (11)$$

where ξ_i is sampled uniformly from the D -dimensional unit hypersphere. Hereby, the problem becomes optimizing λ

1. We use \bar{e} to denote the ℓ_2 normalization of a vector e in this paper.

Algorithm 1 Prior-guided random gradient-free algorithm based on biased sampling (PRGF-BS)

Input: The black-box model f ; input x and label y ; the normalized transfer gradient v ; sampling variance σ ; number of queries q ; input dimension D .

Output: Estimate of the gradient $\nabla f(x)$.

- 1: Estimate the cosine similarity $\alpha = v^\top \overline{\nabla f(x)}$ (detailed in Section 4.3);
- 2: Calculate λ^* according to Eq. (12) given α , q , and D ;
- 3: **if** $\lambda^* = 1$ **then**
- 4: **return** v ;
- 5: **end if**
- 6: $\hat{g} \leftarrow \mathbf{0}$;
- 7: **for** $i = 1$ to q **do**
- 8: Sample ξ_i from the uniform distribution on the D -dimensional unit hypersphere;
- 9: $u_i = \sqrt{\lambda^*} \cdot v + \sqrt{1 - \lambda^*} \cdot (\mathbf{I} - v v^\top) \xi_i$;
- 10: $\hat{g} \leftarrow \hat{g} + \frac{f(x + \sigma u_i, y) - f(x, y)}{\sigma} \cdot u_i$;
- 11: **end for**
- 12: **return** $\nabla f(x) \leftarrow \frac{1}{q} \hat{g}$.

that minimizes $L(\hat{g})$. Note that when $\lambda = \frac{1}{D}$ and $\mathbf{C} = \frac{1}{D} \mathbf{I}$, such that the random vectors are drawn from the uniform distribution on the hypersphere, our method degenerates into the ordinary RGF method. When $\lambda \in [0, \frac{1}{D})$, it indicates that the transfer gradient is worse than a random vector, so we are encouraged to search in other directions by using a small λ .

To find the optimal λ that leads to the minimum value of the loss $L(\hat{g})$, we plug Eq. (10) into Eq. (9), and obtain the closed-form solution as (proof in Appendix A.3)

$$\lambda^* = \begin{cases} 0 & \text{if } \alpha^2 \in [0, a_l] \\ \frac{(1 - \alpha^2)(\alpha^2(D + 2q - 2) - 1)}{2\alpha^2 D q - \alpha^4 D(D + 2q - 2) - 1} & \text{if } \alpha^2 \in (a_l, a_r) \\ 1 & \text{if } \alpha^2 \in [a_r, 1] \end{cases} \quad (12)$$

where $a_l = \frac{1}{D + 2q - 2}$ and $a_r = \frac{2q - 1}{D + 2q - 2}$ (recall that α is the cosine similarity defined in Eq. (8)).

Remark 1. It can be proven (in Appendix A.4) that λ^* is a monotonically increasing function of α^2 , and a monotonically decreasing function of q (when $\alpha^2 > \frac{1}{D}$). It indicates that a larger α or a smaller q (when the transfer gradient is not worse than a random vector) would result in a larger λ^* , which makes sense since we tend to rely on the transfer gradient more when (1) it approximates the true gradient better; (2) the number of queries is not enough to provide much gradient information.

We summarize the PRGF-BS algorithm in Algorithm 1. Note that when $\lambda^* = 1$, we do not need to sample q random vectors because they all equal to v , and we directly return the transfer gradient v as the estimate of $\nabla f(x)$ (Step 3-5), which can save many queries.

4.2 PRGF with Gradient Averaging

In this section, we propose an alternative method to incorporate the transfer gradient v based on gradient averaging. The motivation is as follows. We observe that the RGF estimator in Eq. (5) has the form $\hat{g} = \frac{1}{q} \sum_{i=1}^q \hat{g}_i$, where multiple rough estimates are averaged. Indeed, the transfer gradient itself can also be considered as an estimate of the true gradient.

Thus it is reasonable to perform a weighted average of the transfer gradient and the RGF estimator.

In particular, we first obtain the ordinary RGF estimator defined in Eq. (5) with the sampling distribution \mathcal{P} being the uniform distribution on the D -dimensional unit hypersphere, which is denoted as \hat{g}^U . Then we normalize \hat{g}^U and perform a weighted average of the normalized transfer gradient v and the normalized RGF estimator \bar{g}^U as

$$\hat{g} = \mu v + (1 - \mu) \bar{g}^U, \quad (13)$$

where $\mu \in [0, 1]$ is a balancing coefficient playing a similar role as λ in PRGF-BS.

Given the gradient estimator in Eq. (13), we also aim at deriving the optimal μ that minimizes the loss of the estimator $L(\hat{g})$. We let $\beta = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)^\top \nabla f(x)$ be the cosine similarity between $\frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)$ and the true gradient $\nabla f(x)$, where $\{u_i\}_{i=1}^q$ are sampled from the uniform distribution. As discussed in Section 2.2, the RGF estimator $\hat{g}^U \rightarrow \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)$ when $\sigma \rightarrow 0$, and consequently $\beta \rightarrow \bar{g}^U^\top \nabla f(x)$ as the cosine similarity between the ordinary RGF estimator and the true gradient. Recall that $\alpha = v^\top \nabla f(x)$ is the cosine similarity between the transfer gradient and the true gradient. Then we have the following theorem on the loss of the gradient estimator in Eq. (13).

Theorem 2. (Proof in Appendix A.5) If f is differentiable at x , the loss of the gradient estimator defined in Eq. (13) is

$$\lim_{\sigma \rightarrow 0} L(\hat{g}) = \|\nabla f(x)\|_2^2 - \frac{(\mu\alpha + (1 - \mu)\mathbb{E}[\beta])^2}{\mu^2 + (1 - \mu)^2 + 2\mu(1 - \mu)\alpha\mathbb{E}[\beta]} \|\nabla f(x)\|_2^2, \quad (14)$$

where σ is the sampling variance to get \hat{g}^U .

Theorem 2 indicates that we can achieve the minimum value of $L(\hat{g})$ by optimizing μ . We can calculate the closed-form solution of the optimal μ as (proof in Appendix A.6)

$$\mu^* = \frac{\alpha(1 - \mathbb{E}[\beta]^2)}{\alpha(1 - \mathbb{E}[\beta]^2) + (1 - \alpha^2)\mathbb{E}[\beta]}. \quad (15)$$

Remark 2. We can easily see that μ^* is a monotonically increasing function of α , as well as a monotonically decreasing function of $\mathbb{E}[\beta]$. As will shown in Eq. (16), a larger number of queries q for the RGF estimator can result in a larger $\mathbb{E}[\beta]$, such that μ^* is a monotonically decreasing function of q . These conclusions are consistent with our intuition as explained in Remark 1.

Although we have derived the optimal μ in Eq. (15), the true value of $\mathbb{E}[\beta]$ is still unknown. We find that $\mathbb{E}[\beta]$ cannot directly be calculated but can roughly be approximated as (proof in Appendix A.7)

$$\mathbb{E}[\beta] \approx \sqrt{\frac{q}{D + q - 1}}, \quad (16)$$

where D and q are the input dimension and the number of queries to get \hat{g}^U , respectively. Note that $\mathbb{E}[\beta]$ is irrelevant to the true gradient $\nabla f(x)$. We find such an approximation works well in practice.

It should be noted that we have $\mu^* < 1$, which means that we always need to take q queries to get \hat{g}^U . However,

Algorithm 2 Prior-guided random gradient-free algorithm based on gradient averaging (PRGF-GA)

Input: The black-box model f ; input x and label y ; the normalized transfer gradient v ; sampling variance σ ; number of queries q ; input dimension D ; threshold c .

Output: Estimate of the gradient $\nabla f(x)$.

- 1: Estimate the cosine similarity $\alpha = v^\top \nabla f(x)$ (detailed in Section 4.3);
- 2: Approximate $\mathbb{E}[\beta]$ by $\sqrt{\frac{q}{D+q-1}}$ as in Eq. (16);
- 3: Calculate μ^* according to Eq. (15) given α and $\mathbb{E}[\beta]$;
- 4: **if** $\mu^* \geq c$ **then**
- 5: **return** v ;
- 6: **end if**
- 7: $\hat{g}^U \leftarrow \mathbf{0}$;
- 8: **for** $i = 1$ to q **do**
- 9: Sample u_i from the uniform distribution on the D -dimensional unit hypersphere;
- 10: $\hat{g}^U \leftarrow \hat{g}^U + \frac{f(x + \sigma u_i, y) - f(x, y)}{\sigma} \cdot u_i$;
- 11: **end for**
- 12: **return** $\nabla f(x) \leftarrow \mu^* v + (1 - \mu^*) \bar{g}^U$.

when μ^* is close to 1, the improvement of using $\hat{g} = \mu^* v + (1 - \mu^*) \bar{g}^U$ instead of directly using v as the estimate is marginal. But the former requires q more queries than the latter. To save queries, we use the transfer gradient v as the estimate of $\nabla f(x)$ when it approximates $\nabla f(x)$ well. Thus we preset a threshold $c \in (0, 1)$ such that when $\mu^* \geq c$, we return v directly as the gradient estimate. We summarize the overall PRGF-GA algorithm in Algorithm 2.

Comparisons between PRGF-BS and PRGF-GA. Because the two proposed methods utilize the transfer-based prior in different ways, we are interested in the loss (in Eq. (7)) of the gradient estimators given by different methods, as well as the improvements over the ordinary RGF estimator and the transfer-based prior. To this end, we show the loss curves of gradient estimators given by RGF, transfer gradient, PRGF-BS, and PRGF-GA, respectively, w.r.t. different α , in Fig. 1. PRGF-GA can get a lower loss value than PRGF-BS with a given α , indicating that PRGF-GA can utilize the transfer-based prior better. This is also verified in the experiments.

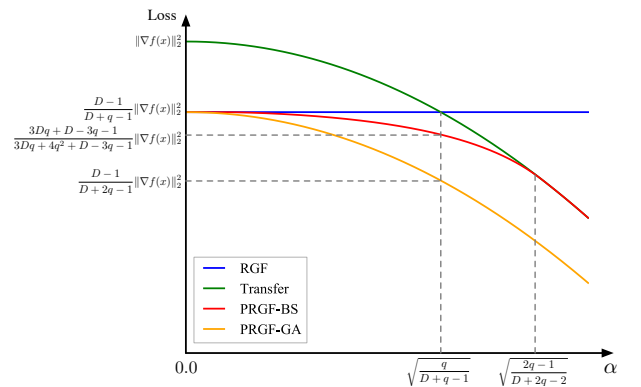


Fig. 1. The loss curves of the different gradient estimators w.r.t. α . The loss of the RGF estimator is $\frac{D-1}{D+q-1} \|\nabla f(x)\|_2^2$. The loss of the transfer gradient is $(1 - \alpha^2) \|\nabla f(x)\|_2^2$. The loss of the PRGF-BS and PRGF-GA estimators can be derived by plugging λ^* and μ^* into Eq. (9) and Eq. (14), respectively.

2. The actual implementation of PRGF-GA is slightly different from Algorithm 2, which will be explained in Appendix B.

4.3 Estimation of Cosine Similarity

To complete our algorithms, we need to estimate the cosine similarity $\alpha = v^\top \nabla f(x) = \frac{v^\top \nabla f(x)}{\|\nabla f(x)\|_2}$, where v is the normalized transfer gradient. Note that the inner product $v^\top \nabla f(x)$ can directly be estimated by the finite difference method as

$$v^\top \nabla f(x) \approx \frac{f(x + \sigma v, y) - f(x, y)}{\sigma}, \quad (17)$$

with a small σ . Hence, the problem is reduced to estimating the norm of the gradient $\|\nabla f(x)\|_2$.

The basic method of estimating $\|\nabla f(x)\|_2$ is to adopt a r -degree homogeneous function g of S variables, i.e., $g(az) = a^r g(z)$ where $a \in \mathbb{R}$ and $z \in \mathbb{R}^S$. Then we have

$$g(\mathbf{W}^\top \nabla f(x)) = \|\nabla f(x)\|_2^r \cdot g(\mathbf{W}^\top \nabla f(x)), \quad (18)$$

where $\mathbf{W} = [w_1, \dots, w_S]$ denotes the matrix consisting of the S random vectors $\{w_s\}_{s=1}^S$. Based on Eq. (18), the norm of the gradient $\|\nabla f(x)\|_2$ could be computed easily if both $g(\mathbf{W}^\top \nabla f(x))$ and $g(\mathbf{W}^\top \nabla f(x))$ can be obtained.

Suppose that we utilize S queries to estimate $\|\nabla f(x)\|_2$. We draw a set of S random vectors $\{w_s\}_{s=1}^S$ independently and uniformly from the D -dimensional unit hypersphere, and then estimate $w_s^\top \nabla f(x)$ based on Eq. (17). Given the estimated $w_s^\top \nabla f(x)$, we can obtain $g(\mathbf{W}^\top \nabla f(x))$ directly.

However, it is non-trivial to obtain the value of $w_s^\top \nabla f(x)$ as well as the function value $g(\mathbf{W}^\top \nabla f(x))$. Nevertheless, we note that the distribution of $w_s^\top \nabla f(x)$ is the same regardless of the direction of $\nabla f(x)$, thus we can compute the expectation of the function value $\mathbb{E}[g(\mathbf{W}^\top \nabla f(x))]$. Based on that, we use $\frac{g(\mathbf{W}^\top \nabla f(x))}{\mathbb{E}[g(\mathbf{W}^\top \nabla f(x))]}$ as an unbiased estimator of $\|\nabla f(x)\|_2^r$. In particular, we choose g as $g(z) = \frac{1}{S} \sum_{s=1}^S z_s^2$. Then $r = 2$, and we have

$$\begin{aligned} \mathbb{E}[g(\mathbf{W}^\top \nabla f(x))] &= \mathbb{E}[(w_1^\top \nabla f(x))^2] \\ &= \overline{\nabla f(x)}^\top \mathbb{E}[w_1 w_1^\top] \nabla f(x) = \frac{1}{D}. \end{aligned} \quad (19)$$

By plugging Eq. (19) into Eq. (18), we can obtain the estimate of the gradient norm as

$$\|\nabla f(x)\|_2 \approx \sqrt{\frac{D}{S} \sum_{s=1}^S \left(\frac{f(x + \sigma w_s, y) - f(x, y)}{\sigma} \right)^2}. \quad (20)$$

To save queries, we estimate the gradient norm periodically instead of in every iteration, since usually it does not change very fast in the optimization process.

5 EXTENSIONS

In this section, we extend our algorithms for incorporating the data-dependent prior and adopting multiple surrogate models to give the transfer-based prior.

5.1 Data-dependent Prior

The commonly used data-dependent prior [19] is proposed to reduce the query complexity, which suggests that we can utilize the structure of the inputs to reduce the input space dimension without sacrificing much accuracy of gradient estimation. The idea of reducing the input dimension has already been adopted in several works [15], [18], [32], [34],

which has shown promise for query-efficient black-box attacks. We observe that many works restrict the adversarial perturbations to lie in a linear subspace of the input space, which allows the application of our theoretical framework. Specifically, we focus on the data-dependent prior proposed in [19]. Below we introduce how to incorporate it into RGF, PRGF-BS, and PRGF-GA appropriately.

RGF. For the RGF gradient estimator in Eq. (5), to leverage the data-dependent prior, suppose that $u_i = \mathbf{V} \xi_i$, where $\mathbf{V} = [v_1, v_2, \dots, v_d]$ is a $D \times d$ matrix ($d < D$), $\{v_j\}_{j=1}^d$ is an orthonormal basis in the d -dimensional subspace of the input space, and ξ_i is a random vector sampled from the d -dimensional unit hypersphere. In [19], the random vector ξ_i drawn in \mathbb{R}^d is up-sampled to u_i in \mathbb{R}^D by the nearest neighbor algorithm. The orthonormal basis $\{v_j\}_{j=1}^d$ can be obtained by first up-sampling the standard basis in \mathbb{R}^d with the same method and then applying normalization. For the ordinary RGF method, ξ_i is sampled uniformly from the d -dimensional unit hypersphere, and $\mathbf{C} = \frac{1}{d} \sum_{j=1}^d v_j v_j^\top$ (recall that $\mathbf{C} = \mathbb{E}[u_i u_i^\top]$ as defined in Theorem 1).

PRGF-BS. For PRGF with biased sampling, we consider incorporating the data-dependent prior into the algorithm along with the transfer-based prior. Similar to Eq. (10), we let one eigenvector of \mathbf{C} be v to exploit the transfer-based prior, and the others are given by the orthonormal basis in the subspace to exploit the data-dependent prior, as

$$\mathbf{C} = \lambda v v^\top + \frac{1 - \lambda}{d} \sum_{j=1}^d v_j v_j^\top. \quad (21)$$

By plugging Eq. (21) into Eq. (9), we can similarly obtain the optimal λ as (proof in Appendix A.8)

$$\lambda^* = \begin{cases} 0 & \text{if } \alpha^2 \in [0, a_l] \\ \frac{A^2(A^2 - \alpha^2(d + 2q - 2))}{A^4 + \alpha^4 d^2 - 2A^2 \alpha^2(q + dq - 1)} & \text{if } \alpha^2 \in (a_l, a_r) \\ 1 & \text{if } \alpha^2 \in [a_r, 1] \end{cases} \quad (22)$$

where $A^2 = \sum_{j=1}^d (v_j^\top \nabla f(x))^2$, $a_l = \frac{A^2}{d + 2q - 2}$, and $a_r = \frac{A^2(2q-1)}{d}$. Note that A is unknown, which should also be estimated. We use a method similar to the one for estimating α , which is detailed in Appendix C.

The remaining problem is to construct a random vector u_i satisfying $\mathbb{E}[u_i u_i^\top] = \mathbf{C}$, with \mathbf{C} specified in Eq. (21). In general, this is difficult since v is not orthogonal to the subspace. To address this problem, we sample u_i in a way that $\mathbb{E}[u_i u_i^\top]$ is a good approximation of \mathbf{C} (explanation in Appendix A.9), which is similar to Eq. (11) as

$$u_i = \sqrt{\lambda} \cdot v + \sqrt{1 - \lambda} \cdot (\mathbf{I} - v v^\top) \mathbf{V} \xi_i, \quad (23)$$

where ξ_i is sampled uniformly from the d -dimensional unit hypersphere.

The PRGF-BS algorithm with the data-dependent prior is similar to Algorithm 1. We first estimate α and A , and then calculate λ^* by Eq. (22). If $\lambda^* = 1$, we use the transfer gradient v as the estimate. Otherwise, we sample q random vectors by Eq. (23) and get the gradient estimate by Eq. (5).

PRGF-GA. We similarly incorporate the data-dependent prior into the PRGF-GA algorithm. In this case, we first get an ordinary subspace RGF estimator \hat{g}^S instead of the ordinary RGF estimator, by sampling ξ_i uniformly from the

d -dimensional unit hypersphere and letting $u_i = \mathbf{V}\xi_i$. Then we normalize \hat{g}^S and obtain the averaged gradient estimator in a similar manner to Eq. (13) as

$$\hat{g} = \mu v + (1 - \mu)\hat{g}^S. \quad (24)$$

To derive the optimal μ that minimizes the loss $L(\hat{g})$, we define $\overline{\nabla f(x)}_T = (\sum_{j=1}^d v_j v_j^\top) \nabla f(x)$ as the projection of $\nabla f(x)$ onto the subspace corresponding to the data-dependent prior. We also need $A^2 = \sum_{j=1}^d (v_j^\top \nabla f(x))^2 = \|\overline{\nabla f(x)}_T\|^2$. We let $\beta = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)^\top \nabla f(x)$ be the cosine similarity between $\frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)$ and the true gradient $\nabla f(x)$, in which $\{u_i\}_{i=1}^q$ lie in the subspace. We have the following theorem on the loss of the gradient estimator in Eq. (24).

Theorem 3. (Proof in Appendix A.10) Let $\alpha_1 = v^\top \overline{\nabla f(x)}_T$. If f is differentiable at x and $A^2 > 0$, the loss of the gradient estimator define in Eq. (24) is

$$\lim_{\sigma \rightarrow 0} L(\hat{g}) = \|\nabla f(x)\|^2 - \frac{(\mu\alpha + (1 - \mu)\mathbb{E}[\beta])^2}{\mu^2 + (1 - \mu)^2 + 2\mu(1 - \mu)\frac{\alpha_1}{A^2}\mathbb{E}[\beta]} \|\nabla f(x)\|^2, \quad (25)$$

where σ is the sampling variance to get \hat{g}^S .

Based on Theorem 3, we calculate the optimal solution of μ by minimizing Eq. (25) as (proof in Appendix A.11)

$$\mu^* = \frac{A^2\alpha - \alpha_1\mathbb{E}[\beta]^2}{(A^2 - \alpha_1\mathbb{E}[\beta])(\alpha + \mathbb{E}[\beta])} \approx \frac{\alpha}{\alpha + \mathbb{E}[\beta]}. \quad (26)$$

The approximation works mainly because $A \gg \mathbb{E}[\beta]$ (since $\mathbb{E}[\beta] \approx A\sqrt{\frac{q}{d+q-1}}$ as shown Appendix A.11). Therefore, μ^* can be approximated without α_1 , such that we do not need to estimate α_1 .

The PRGF-GA algorithm with the data-dependent prior is similar to Algorithm 2. We first estimate α and A , approximate $\mathbb{E}[\beta]$ by $A\sqrt{\frac{q}{d+q-1}}$, and then calculate μ^* by Eq. (26). If $\mu^* \geq c$, we use the transfer gradient v as the estimate. Otherwise, we get the ordinary subspace RGF estimate \hat{g}^S with q queries, and then use $\hat{g} \leftarrow \mu^* v + (1 - \mu^*)\hat{g}^S$.

5.2 Multiple Surrogate Models

The idea of utilizing multiple surrogate models has been adopted in [16], [22] for improving transfer-based black-box attacks. They show that the adversarial examples generated for multiple models are more likely to fool other black-box models with the increased transferability. In our algorithms, we can also utilize multiple surrogate models to extract a more effective transfer-based prior, which can consequently enhance the attack performance.

Assume that we have M surrogate models. For an input x , we denote the gradients of these surrogate models at x as $\{g^{(m)}\}_{m=1}^M$, where the gradients are not normalized for now. A simple approach to obtain the transfer-based prior is averaging these gradients directly, as $v = \frac{1}{M} \sum_{m=1}^M g^{(m)}$. Despite the simplicity, this approach treats the gradients of surrogate models with equal importance and neglects the intrinsic similarity between different surrogate models and the target model. It has been observed that the adversarial

examples are more likely to transfer within the same family of model architectures [35], indicating that we could design an improved transfer-based prior by leveraging more useful surrogate models/gradients.

Specifically, we denote the M -dimensional subspace spanned by $\{g^{(m)}\}_{m=1}^M$ as \mathbf{G} . The best approximation of the true gradient $\nabla f(x)$ that lies in \mathbf{G} is the projection of $\nabla f(x)$ onto the subspace \mathbf{G} . Therefore, we first get an orthonormal basis of \mathbf{G} by the Gram-Schmidt orthonormalization method, denoted as $\{v^{(m)}\}_{m=1}^M$. Then the projection of $\nabla f(x)$ onto \mathbf{G} can be expressed as

$$\nabla f(x)_G = \sum_{m=1}^M \nabla f(x)^\top v^{(m)} \cdot v^{(m)}, \quad (27)$$

in which the inner product $\nabla f(x)^\top v^{(m)}$ can be approximated by the finite difference method as shown in Eq. (17). Hence, we let the transfer-based prior be $v = \nabla f(x)_G$. With v obtained by multiple surrogate gradients, we then perform PRGF-BS or PRGF-GA attacks with the same algorithms.

6 EXPERIMENTS

In this section, we present the empirical results to demonstrate the effectiveness of the proposed methods on attacking black-box image classifiers. We perform untargeted attacks under both the ℓ_2 and ℓ_∞ norms on the ImageNet [29] and CIFAR-10 [36] datasets. We show the results under the ℓ_2 norm in this section and leave the extra results under the ℓ_∞ norm in Appendix D. The results for both norms are consistent to verify the superiority of our methods. We also conduct experiments on defense models in Appendix E. We first specify the experimental setting in Section 6.1. Then we show the performance of gradient estimation in Section 6.2. We further compare the attack performance of the proposed algorithms with others on ImageNet in Section 6.3, and on CIFAR-10 in Section 6.4, respectively.

6.1 Experimental Settings

ImageNet [29]. We choose 1,000 images randomly from the ILSVRC 2012 validation set to perform evaluations. Those images are normalized to $[0, 1]$. We consider three black-box target models, which are Inception-v3 [37], VGG-16 [38], and ResNet-50 [39]. For most experiments, we use the ResNet-v2-152 model [40] as the surrogate model to provide the transfer gradient. We also study different surrogate models in Section 6.3.1. For the proposed PRGF-BS and PRGF-GA algorithms, we set the number of queries in each step of gradient estimation as $q = 50$ and the sampling variance as $\sigma = 0.0001 \cdot \sqrt{D}$. We let the attack loss function f in Eq. (2) be the cross-entropy loss. After we obtain the gradient estimate, we apply the PGD update rule as in Eq. (3) to generate the adversarial example with the estimated gradient. We set the perturbation size as $\epsilon = \sqrt{0.001 \cdot D}$ and the step size as $\eta = 2$ in PGD under the ℓ_2 norm, while set $\epsilon = 0.05$ and $\eta = 0.005$ under the ℓ_∞ norm. For PRGF-GA, there is a preset threshold c determining whether to directly return the transfer gradient, which is set as $c = \sqrt{2}/(\sqrt{2}+1)$.

CIFAR-10 [36]. We adopt all the 10,000 test images for evaluations, which are in $[0, 1]$. The black-box target models include ResNet-50 [39], DenseNet-121 [41], and SENet-18 [42]. We adopt a Wide ResNet model (WRN-34-10) [43]

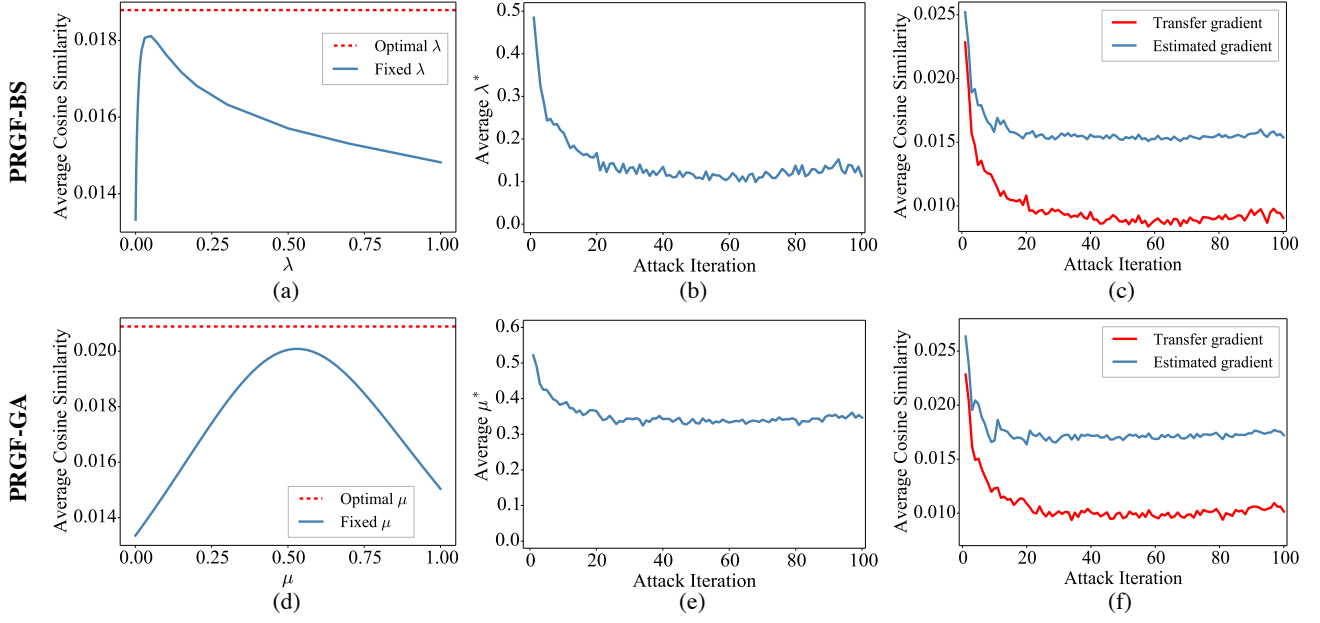


Fig. 2. (a) The average cosine similarity between the estimated gradient and the true gradient. The estimate is given by PRGF-BS with fixed λ and optimal λ , respectively. (b) The average λ^* in PRGF-BS across attack iterations. (c) The average cosine similarity between the transfer and the true gradients, and that between the estimated and the true gradients, across attack iterations in PRGF-BS. (d) The average cosine similarity between the estimated gradient and the true gradient. The estimate is given by PRGF-GA with fixed μ and optimal μ , respectively. (e) The average μ^* in PRGF-GA across attack iterations. (f) The average cosine similarity between the transfer and the true gradients, and that between the estimated and the true gradients, across attack iterations in PRGF-GA.

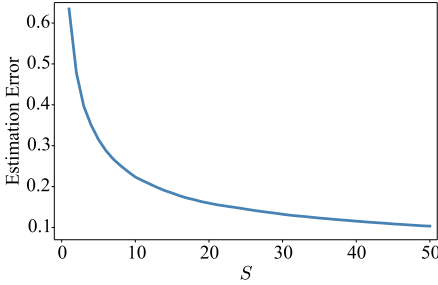


Fig. 3. The estimation error of gradient norm w.r.t. different queries S .

as the surrogate model. We set $q = 50$ and $\sigma = 0.001 \cdot \sqrt{D}$. The loss function f is the CW loss [8] since it performs better than the cross-entropy loss on CIFAR-10. The perturbation size is $\epsilon = 1.0$ under the ℓ_2 norm and $\epsilon = 8/255$ under the ℓ_∞ norm. The step size in PGD is $\eta = 0.25$ under the ℓ_2 norm and $\eta = 2/255$ under the ℓ_∞ norm. The threshold in PRGF-GA is also set as $c = \sqrt{2}/(\sqrt{2}+1)$.

Note that when counting the total number of queries in our methods, we include the additional queries of estimating the cosine similarity α .

6.2 Performance of Gradient Estimation

We now conduct several ablation studies to show the performance of gradient estimation. All experiments in this section are performed on the Inception-v3 [37] model on ImageNet.

Estimation of gradient norm. First, we demonstrate the performance of gradient norm estimation as introduced in Section 4.3. In general, the gradient norm (or cosine similarity) is easier to estimate than the true gradient since it's a scalar value. Fig. 3 illustrates the estimation error of the gradient norm, defined as the (normalized) RMSE —

$$\sqrt{\mathbb{E} \left(\frac{\|\widehat{\nabla f(x)}\|_2 - \|\nabla f(x)\|_2}{\|\nabla f(x)\|_2} \right)^2}, \text{ w.r.t. the number of queries } S,$$

where $\|\nabla f(x)\|_2$ is the true norm, $\|\widehat{\nabla f(x)}\|_2$ is the estimated one, and the expectation is taken over all images along the attack procedure. It can be obtained that dozens of queries are sufficient to reach a small estimation error of gradient norm. We choose $S = 10$ in the following experiments to reduce the number of queries while the estimation error is acceptable. The gradient norm is estimated every 10 attack iterations to further reduce the required queries, since usually its value is relatively stable in the optimization process.

Performance of gradient estimation. Second, we verify the effectiveness of the derived *optimal* λ in PRGF-BS and μ in PRGF-GA (i.e., λ^* in Eq. (12) and μ^* in Eq. (15)) for gradient estimation, compared with any fixed $\lambda, \mu \in [0, 1]$. To this end, we perform attacks against Inception-v3 using PRGF-BS with λ^* or PRGF-GA with μ^* , and at the same time calculate the cosine similarity between the estimated gradient and the true gradient. In both methods, λ^* and μ^* are calculated using the estimated α instead of its true value. Meanwhile, along the PGD updates, we also use fixed λ or μ to get gradient estimates, and calculate the corresponding cosine similarities. Note that λ^* and μ^* do not correspond to any fixed value, since they vary during iterations.

We show the average cosine similarities of different fixed values of λ in Fig. 2(a), and those of different fixed values of μ in Fig. 2(d). The first observation is that when a suitable value of λ (or μ) is chosen, the proposed PRGF-BS (or PRGF-GA) provides a better gradient estimate than both the ordinary RGF method with uniform distribution (when $\lambda = \frac{1}{D} \approx 0$ or $\mu = 0$) and the transfer gradient (when $\lambda = 1$ or $\mu = 1$). The second observation is that adopting λ^* (or μ^*) brings further improvement upon any fixed λ (or μ), demonstrating the effectiveness of our theoretical analyses.

Gradient estimation across attack iterations. Finally, we aim at examining the effectiveness of the transfer-based

TABLE 1

The experimental results of black-box attacks against Inception-v3, VGG-16, and ResNet-50 under the ℓ_2 norm on ImageNet. We report the attack success rate (ASR), and the average/median number of queries (AVG. Q/MED. Q) needed to generate an adversarial example over successful attacks. We mark the best results in **bold**. The subscript “D” denotes the methods with the data-dependent prior.

Methods	Inception-v3			VGG-16			ResNet-50		
	ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q
NES [12]	95.5%	1752	1071	98.7%	1103	816	98.4%	988	714
SPSA [44]	93.7%	1808	1122	98.1%	1290	1020	98.4%	1236	969
AutoZoom [18]	85.4%	2443	1847	96.3%	1589	949	94.8%	2065	1223
Bandits _T [19]	92.4%	1560	810	94.0%	584	225	96.2%	1076	446
Bandits _{TD} [19]	97.2%	874	352	94.9%	278	82	96.8%	512	195
\mathcal{N} ATTACK [45]	98.2%	1020	510	99.6%	593	357	99.5%	535	357
RGF	97.7%	1309	816	99.8%	749	561	99.6%	673	510
PRGF-BS ($\lambda = 0.05$)	97.4%	1047	561	99.7%	624	408	99.3%	511	306
PRGF-BS (λ^*)	98.1%	745	320	99.6%	331	182	99.6%	265	132
PRGF-GA ($\mu = 0.5$)	97.9%	958	572	99.8%	528	364	99.6%	485	312
PRGF-GA (μ^*)	97.9%	735	314	99.7%	320	184	99.5%	250	134
RGF _D	99.1%	910	561	100.0%	372	306	99.7%	429	306
PRGF-BS _D ($\lambda = 0.05$)	98.8%	728	408	99.9%	359	255	99.8%	379	255
PRGF-BS _D (λ^*)	99.1%	649	332	99.8%	250	180	99.6%	232	140
PRGF-GA _D ($\mu = 0.5$)	99.3%	734	416	100.0%	332	260	99.7%	340	260
PRGF-GA _D (μ^*)	99.2%	644	312	99.7%	239	184	99.7%	240	140

prior across attack iterations. We show the average λ^* and μ^* over all images w.r.t. attack iterations in Fig. 2(b) for PRGF-BS, and in Fig. 2(e) for PRGF-GA, respectively. The curves show that λ^* and μ^* decrease along the iterations. Besides, Fig. 2(c) and Fig. 2(f) show the average cosine similarity between the transfer and the true gradients, and that between the estimated and the true gradients w.r.t. attack iterations, in PRGF-BS and PRGF-GA. All of these results demonstrate that the transfer gradient is more useful at beginning, and becomes less useful along the iterations. However, the estimated gradient in either PRGF-BS or PRGF-GA can remain a higher cosine similarity with the true gradient, which facilitates the adversarial attacks consequently. The results also corroborate that we need to use the adaptive λ^* or μ^* in different attack iterations.

6.3 Results on ImageNet

In this section, we perform black-box adversarial attacks against three ImageNet models, including Inception-v3 [37], VGG-16 [38], and ResNet-50 [39]. Besides the two proposed PRGF-BS and PRGF-GA algorithms, we incorporate several baseline methods, including the ordinary RGF method with uniform sampling, the PRGF-BS method with the fixed $\lambda = 0.05$, and the PRGF-GA method with the fixed $\mu = 0.5$. Those fixed values are chosen according to Fig. 2(a) and Fig. 2(d), which can estimate the gradient more accurately. We set the number of queries as $q = 50$ for gradient estimation and the sampling variance as $\sigma = 0.0001 \cdot \sqrt{D}$, which are identical for all of these methods. We also incorporate the data-dependent prior into these methods for comparison (which are denoted by adding a subscript “D”). We set the dimension of the subspace as $d = 50 \times 50 \times 3$.

Besides, we compare the attack performance with various state-of-the-art attack methods, including the natural evolution strategies (NES) [12], SPSA [44], AutoZoom [18], bandit optimization methods (Bandits_T and Bandits_{TD}) [19], and \mathcal{N} ATTACK [45]. For all methods, we restrict the maximum number of queries for each image to be 10,000. We report a successful attack if a method can generate an adversarial example within 10,000 queries and the size of perturbation is smaller than the budget (i.e., $\epsilon = \sqrt{0.001 \cdot D}$).

Table 1 shows the results, where we report the success rate of black-box attacks and the average/median number of queries needed to generate an adversarial example over successful attacks. We have the following observations. First, compared with the state-of-the-art attacks, the proposed methods generally lead to higher attack success rates and require much fewer queries. Second, the transfer-based prior provides useful prior information for black-box attacks since PRGF based methods perform better than the ordinary RGF method. Third, using a fixed λ in PRGF-BS or a fixed μ in PRGF-GA cannot exceed the performance of using their optimal values, although they already lead to comparable performance with the state-of-the-art methods. Fourth, the results also prove that the data-dependent prior is orthogonal to the proposed transfer-based prior, since integrating the data-dependent prior leads to better results. Fifth, PRGF-GA requires slightly fewer queries than PRGF-BS in most cases, which are consistent with the loss curves in Fig. 1.

6.3.1 Different Surrogate Models

Here we conduct an ablation study to investigate the effectiveness of adopting different surrogate models. We use the ResNet-v2-152 model [40] as the surrogate model in the above experiments. We additionally consider Inception-v4 [46], ResNet-v2-152 + Inception-v4, and ResNet-v2-152 + Inception-v4 + Inception-ResNet-v2 [46] as the surrogate models. Note that the latter two include multiple surrogate models. We adopt the *subspace projection* method introduced in Section 5.2 to get the transfer-based prior when multiple surrogate models are available. Besides, we also compare this method with the *equal averaging* method that directly averages the gradients of multiple models (only in the case of using ResNet-v2-152 + Inception-v4).

We show the attack performance of PRGF-BS, PRGF-GA, PRGF-BS_D, and PRGF-GA_D with different surrogate models in Table 2. It is easy to see that adopting multiple surrogate models can significantly improve the attack success rates and reduce the number of queries. When using three surrogate models, the median number of queries is less than 100 for all target models, which validates the effectiveness of the transfer-based prior. Besides, it can be noted that the

TABLE 2

The experimental results of PRGF-BS and PRGF-GA attacks against Inception-v3, VGG-16, and ResNet-50 under the ℓ_2 norm on ImageNet using different surrogate models. We report the attack success rate (ASR), and the average/median number of queries (AVG. Q/MED. Q) needed to generate an adversarial example over successful attacks. We mark the best results in **bold**. The subscript “D” denotes the methods with the data-dependent prior.

Surrogate Model(s)	Methods	Inception-v3			VGG-16			ResNet-50		
		ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q
ResNet-v2-152	PRGF-BS	98.1%	745	320	99.6%	331	182	99.6%	265	132
	PRGF-GA	97.9%	735	314	99.7%	320	184	99.5%	250	134
	PRGF-BS _D	99.1%	649	332	99.8%	250	180	99.6%	232	140
	PRGF-GA _D	99.2%	644	312	99.7%	239	184	99.7%	240	140
Inception-v4	PRGF-BS	98.9%	673	252	99.9%	350	184	99.7%	386	234
	PRGF-GA	99.0%	622	242	99.9%	343	186	99.7%	350	192
	PRGF-BS _D	99.2%	569	252	99.9%	251	180	99.7%	298	224
	PRGF-GA _D	99.5%	595	248	100.0%	256	186	99.8%	298	194
ResNet-v2-152 + Inception-v4 (equal averaging)	PRGF-BS	98.2%	592	188	99.7%	290	132	99.8%	282	130
	PRGF-GA	98.7%	575	190	99.9%	283	134	99.7%	262	132
	PRGF-BS _D	99.2%	537	230	99.9%	219	140	99.7%	245	138
	PRGF-GA _D	99.1%	516	236	99.9%	219	136	99.7%	242	138
ResNet-v2-152 + Inception-v4 (subspace projection)	PRGF-BS	99.1%	348	94	99.9%	163	59	99.6%	151	70
	PRGF-GA	99.5%	342	96	100.0%	146	66	99.6%	135	70
	PRGF-BS _D	99.1%	412	156	99.9%	165	94	99.8%	182	105
	PRGF-GA _D	99.5%	404	152	100.0%	169	96	99.7%	164	96
ResNet-v2-152 + Inception-v4 + Inception-ResNet-v2 (subspace projection)	PRGF-BS	99.5%	198	50	100.0%	93	40	99.9%	103	40
	PRGF-GA	99.8%	191	50	100.0%	89	40	99.8%	96	40
	PRGF-BS _D	99.7%	296	95	99.9%	122	70	99.9%	135	75
	PRGF-GA _D	99.6%	267	97	100.0%	118	72	99.8%	126	77

TABLE 3

The experimental results of black-box attacks against ResNet-50, DenseNet-121, and SENet-18 under the ℓ_2 norm on CIFAR-10. We report the attack success rate (ASR) and the average/median number of queries (AVG. Q/MED. Q) needed to generate an adversarial example over successful attacks. We mark the best results in **bold**.

Methods	ResNet-50			DenseNet-121			SENet-18		
	ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q
NES [12]	99.7%	642	459	99.6%	631	459	99.8%	582	408
SPSA [44]	99.8%	785	561	99.7%	780	510	99.9%	718	459
Bandits _T [19]	100.0%	375	194	100.0%	356	174	100.0%	317	150
\mathcal{N} ATTACK [45]	100.0%	401	255	100.0%	404	255	100.0%	350	204
RGF	99.9%	460	357	99.9%	472	357	99.9%	423	306
PRGF-BS ($\lambda = 0.05$)	99.8%	290	204	99.9%	274	204	99.9%	262	153
PRGF-BS (λ^*)	99.9%	268	124	100.0%	220	124	100.0%	187	76
PRGF-GA ($\mu = 0.5$)	99.3%	306	204	99.7%	260	204	99.9%	243	153
PRGF-GA (μ^*)	99.9%	173	76	99.9%	168	76	99.9%	146	65

subspace projection method performs better than the equal averaging method, because the subspace projection method can obtain the transfer-based prior which approximates the true gradient best in the subspace.

Another observation from the results is that adopting a similar surrogate model of the target model can enhance the attack performance. In particular, ResNet-v2-152 is better than Inception-v4 as the surrogate model for attacking the ResNet-50 models. On the other hand, Inception-v4 is better than ResNet-v2-152 for attacking the Inception-v3 model. It is reasonable since the gradients of models within the same family of model architectures would be similar, which has been verified in [35] showing that the adversarial transferability is higher across similar model architectures.

Finally, we find that the data-dependent prior becomes less useful with a more powerful transfer-based prior obtained by multiple surrogate models. Specifically, PRGF-BS_D and PRGF-GA_D require more queries than PRGF-BS and PRGF-GA when using two or three surrogate models. The reason is as follows. For PRGF-BS and PRGF-GA without the data-dependent prior, it is more likely to obtain $\lambda^* = 1$ or $\mu^* = 1$ with the more effective transfer-based prior, such that we do not need to perform q queries to estimate the gradient. However, in PRGF-BS_D and PRGF-GA_D, λ^*

and μ^* are less probable to be 1 due to that sampling in the data-dependent subspace can also improve the gradient estimate, and therefore we need q more queries to get the estimate. Although the data-dependent prior helps to give a more accurate gradient estimate, the cost of q more queries degrades the efficiency of attacks.

6.4 Results on CIFAR-10

In this section, we show the results of black-box adversarial attacks on CIFAR-10. Similar to the experiments on ImageNet, we compare the performance of PRGF-BS and PRGF-GA with three baselines — RGF, PRGF-BS with the fixed $\lambda = 0.05$, and PRGF-GA with the fixed $\mu = 0.5$, as well as four other attacks — NES [12], SPSA [44], Bandits_T [19], and \mathcal{N} ATTACK [45]. Since the image resolution in CIFAR-10 is not very high (i.e., $32 \times 32 \times 3$), we do not adopt the data-dependent prior. We also restrict the maximum number of queries for each image to be 10,000. Note that hundreds of queries could be sufficient due to the lower input dimension of CIFAR-10, but we adopt the maximum 10,000 queries to make it consistent with the setting on ImageNet.

The black-box attack results of those methods against ResNet-50 [39], DenseNet-121 [41], and SENet-18 [42] are presented in Table 3. It can be seen that with the maximum

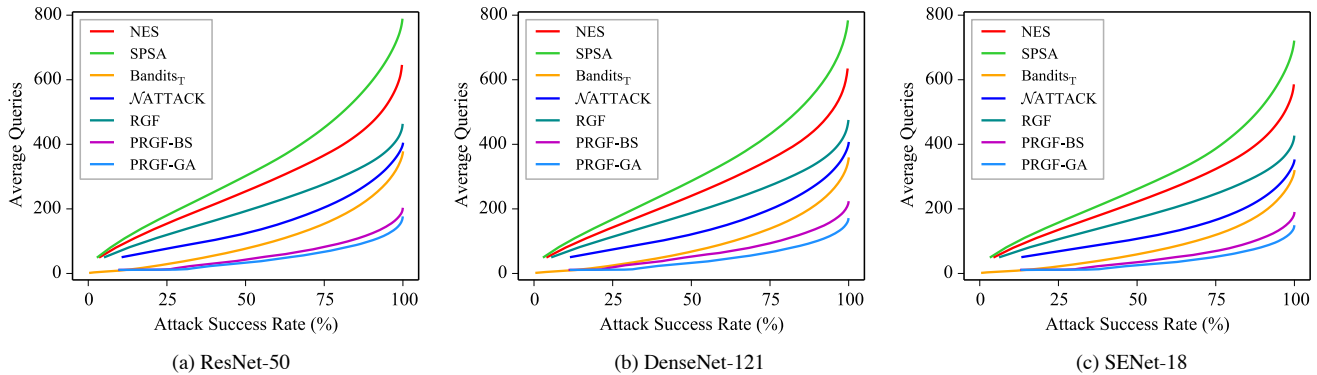


Fig. 4. The average number of queries for generating the adversarial examples that are successfully misclassified by the black-box model at any desired success rate on CIFAR-10.

number of 10,000 queries, all attack methods can achieve near 100% attack success rate. Nevertheless, our proposed methods (especially PRGF-GA) require much less queries to successfully generate adversarial examples, which demonstrates the query efficiency of our proposed methods.

Fig. 4 shows the average number of queries for successfully misleading the black-box model by reaching a desired success rate. For a given attack success rate, our methods require much less queries, indicating that they are much more query-efficient than other baseline methods.

7 CONCLUSION

In this paper, two prior-guided random gradient-free algorithms were proposed for improving black-box attacks. Our methods can utilize a transfer-based prior given by the gradient of a surrogate model through biased sampling and gradient averaging, respectively. We appropriately integrated the transfer-based prior with model queries by the derived optimal coefficient in both methods under the gradient estimation framework. Furthermore, we extended the proposed methods by incorporating the data-dependent prior and utilizing multiple surrogate models. The experimental results consistently demonstrate the effectiveness of our methods, which require much fewer queries to attack black-box models with higher success rates compared with various state-of-the-art attack methods. We released our codes at <https://github.com/thu-ml/Prior-Guided-RGF>.

ACKNOWLEDGMENTS

This work was supported by the National Key Research and Development Program of China (No. 2020AAA0104304), NSFC Projects (Nos. 61620106010, 62061136001, 61621136008, 62076147, U19B2034, U1811461, U19A2081), Beijing NSF Project (No. JQ19016), Beijing Academy of Artificial Intelligence (BAAI), Tsinghua-Huawei Joint Research Program, Tsinghua Institute for Guo Qiang, Tsinghua-OPPO Joint Research Center for Future Terminal Technology and Tsinghua-China Mobile Communications Group Co., Ltd. Joint Institute.

REFERENCES

- [1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- [2] B. Biggio, I. Corona, D. Maiorca, B. Nelson, P. Laskov, G. Giacinto, and F. Roli, "Evasion attacks against machine learning at test time," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 387–402, 2013.
- [3] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *International Conference on Learning Representations (ICLR)*, 2014.
- [4] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *International Conference on Learning Representations (ICLR)*, 2015.
- [5] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *arXiv preprint arXiv:1607.02533*, 2016.
- [6] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *International Conference on Machine Learning (ICML)*, pp. 274–283, 2018.
- [7] Y. Dong, T. Pang, H. Su, and J. Zhu, "Evading defenses to transferable adversarial examples by translation-invariant attacks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4312–4321, 2019.
- [8] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *IEEE Symposium on Security and Privacy*, pp. 39–57, 2017.
- [9] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *International Conference on Learning Representations (ICLR)*, 2018.
- [10] Y. Dong, Q.-A. Fu, X. Yang, T. Pang, H. Su, Z. Xiao, and J. Zhu, "Benchmarking adversarial robustness on image classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 321–331, 2020.
- [11] T. Pang, X. Yang, Y. Dong, H. Su, and J. Zhu, "Bag of tricks for adversarial training," in *International Conference on Learning Representations (ICLR)*, 2021.
- [12] A. Ilyas, L. Engstrom, A. Athalye, and J. Lin, "Black-box adversarial attacks with limited queries and information," in *International Conference on Machine Learning (ICML)*, pp. 2137–2146, 2018.
- [13] W. Brendel, J. Rauber, and M. Bethge, "Decision-based adversarial attacks: Reliable attacks against black-box machine learning models," in *International Conference on Learning Representations (ICLR)*, 2018.
- [14] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 506–519, 2017.
- [15] P. Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C. J. Hsieh, "Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models," in *ACM Workshop on Artificial Intelligence and Security (AISec)*, pp. 15–26, 2017.
- [16] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9185–9193, 2018.
- [17] A. Nitin Bhagoji, W. He, B. Li, and D. Song, "Practical black-box attacks on deep neural networks using efficient query mechanisms," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 154–169, 2018.
- [18] C.-C. Tu, P. Ting, P.-Y. Chen, S. Liu, H. Zhang, J. Yi, C.-J. Hsieh, and S.-M. Cheng, "Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks," in *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI)*, pp. 742–749, 2019.

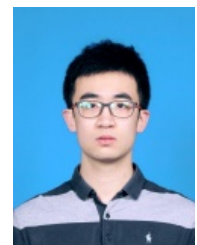
- [19] A. Ilyas, L. Engstrom, and A. Madry, "Prior convictions: Black-box adversarial attacks with bandits and priors," in *International Conference on Learning Representations (ICLR)*, 2019.
- [20] Y. Dong, H. Su, B. Wu, Z. Li, W. Liu, T. Zhang, and J. Zhu, "Efficient decision-based black-box adversarial attacks on face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7714–7722, 2019.
- [21] N. Papernot, P. McDaniel, and I. Goodfellow, "Transferability in machine learning: from phenomena to black-box attacks using adversarial samples," *arXiv preprint arXiv:1605.07277*, 2016.
- [22] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," in *International Conference on Learning Representations (ICLR)*, 2017.
- [23] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. L. Yuille, "Improving transferability of adversarial examples with input diversity," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2730–2739, 2019.
- [24] Y. Nesterov and V. Spokoiny, "Random gradient-free minimization of convex functions," *Foundations of Computational Mathematics*, vol. 17, no. 2, pp. 527–566, 2017.
- [25] S. Ghadimi and G. Lan, "Stochastic first-and zeroth-order methods for nonconvex stochastic programming," *SIAM Journal on Optimization*, vol. 23, no. 4, pp. 2341–2368, 2013.
- [26] J. C. Duchi, M. I. Jordan, M. J. Wainwright, and A. Wibisono, "Optimal rates for zero-order convex optimization: The power of two function evaluations," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2788–2806, 2015.
- [27] S. Cheng, Y. Dong, T. Pang, H. Su, and J. Zhu, "Improving black-box adversarial attacks with a transfer-based prior," in *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 10934–10944, 2019.
- [28] P. D. Lax and M. S. Terrell, *Calculus with applications*. Springer, 2014.
- [29] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
- [30] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters, and J. Schmidhuber, "Natural evolution strategies," *Journal of Machine Learning Research*, vol. 15, no. 27, pp. 949–980, 2014.
- [31] S. J. Oh, M. Augustin, B. Schiele, and M. Fritz, "Towards reverse-engineering black-box neural networks," in *International Conference on Learning Representations (ICLR)*, 2018.
- [32] T. Brunner, F. Diehl, M. T. Le, and A. Knoll, "Guessing smart: Biased sampling for efficient black-box adversarial attacks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 4958–4966, 2019.
- [33] N. Maheswaranathan, L. Metz, G. Tucker, D. Choi, and J. Sohl-Dickstein, "Guided evolutionary strategies: Augmenting random search with surrogate gradients," in *International Conference on Machine Learning (ICML)*, pp. 4264–4273, 2019.
- [34] C. Guo, J. S. Frank, and K. Q. Weinberger, "Low frequency adversarial perturbation," *arXiv preprint arXiv:1809.08758*, 2018.
- [35] D. Su, H. Zhang, H. Chen, J. Yi, P.-Y. Chen, and Y. Gao, "Is robustness the cost of accuracy?—a comprehensive study on the robustness of 18 deep image classification models," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 631–648, 2018.
- [36] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," tech. rep., University of Toronto, 2009.
- [37] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, 2016.
- [38] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations (ICLR)*, 2015.
- [39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.
- [40] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 630–645, 2016.
- [41] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4700–4708, 2017.
- [42] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7132–7141, 2018.
- [43] S. Zagoruyko and N. Komodakis, "Wide residual networks," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2016.
- [44] J. Uesato, B. O'Donoghue, A. v. d. Oord, and P. Kohli, "Adversarial risk and the dangers of evaluating against weak attacks," in *International Conference on Machine Learning (ICML)*, pp. 5025–5034, 2018.
- [45] Y. Li, L. Li, L. Wang, T. Zhang, and B. Gong, "Nattack: Learning the distributions of adversarial examples for an improved black-box attack on deep neural networks," in *International Conference on Machine Learning (ICML)*, pp. 3866–3876, 2019.
- [46] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI)*, pp. 4278–4284, 2017.



Yinpeng Dong received his BS degree from the Department of Computer Science and Technology in Tsinghua University. He is currently a PhD student in the Department of Computer Science and Technology in Tsinghua University. His research interests are primarily on the adversarial robustness of machine learning and deep learning. He received Microsoft Research Asia Fellowship and Baidu Fellowship.



Shuyu Cheng received his BS degree from the Department of Computer Science and Technology in Tsinghua University. He is currently a PhD student in the Department of Computer Science and Technology in Tsinghua University. His research interests are primarily on the adversarial robustness of machine learning and deep learning.



Tianyu Pang received his BS degree from the Department of Physics in Tsinghua University. He is currently a PhD student in the Department of Computer Science and Technology in Tsinghua University. His research interests are primarily on the adversarial robustness of machine learning and deep learning. He received Microsoft Research Asia Fellowship and Baidu Fellowship.



Hang Su is an associated professor in the Department of Computer Science and Technology at Tsinghua University. His research interests lie in the development of computer vision and machine learning algorithms for solving scientific and engineering problems arising from artificial learning and reasoning. He received "Young Investigator Award" from MICCAI2012, the "Best Paper Award" in AVSS2012, and "Platinum Best Paper Award" in ICME2018.



Jun Zhu received his BS and PhD degrees from the Department of Computer Science and Technology in Tsinghua University, where he is currently a professor. He was an adjunct faculty and postdoctoral fellow in the Machine Learning Department, Carnegie Mellon University. His research interest is primarily on developing machine learning methods to understand scientific and engineering data arising from various fields. He regularly serves as Area Chairs at prestigious conferences, including ICML, NeurIPS, ICLR, IJCAI and AAAI. He is a senior member of the IEEE, and was selected as "AI's 10 to Watch" by IEEE Intelligent Systems.

APPENDIX A

PROOFS

We provide the proofs in this section.

A.1 Proof of Theorem 1

Theorem 1. *If f is differentiable at x , the loss of the gradient estimator \hat{g} defined in Eq. (5) is*

$$\lim_{\sigma \rightarrow 0} L(\hat{g}) = \|\nabla f(x)\|_2^2 - \frac{(\nabla f(x)^\top \mathbf{C} \nabla f(x))^2}{(1 - \frac{1}{q})\nabla f(x)^\top \mathbf{C}^2 \nabla f(x) + \frac{1}{q}\nabla f(x)^\top \mathbf{C} \nabla f(x)},$$

where σ is the sampling variance, $\mathbf{C} = \mathbb{E}[u_i u_i^\top]$ with u_i being the random vector, $\|u_i\|_2 = 1$, and q is the number of random vectors as in Eq. (5).

Remark 3. Rigorously speaking, we assume $\nabla f(x)^\top \mathbf{C} \nabla f(x) \neq 0$ in the statement of the theorem (and also in the proof), since when $\nabla f(x)^\top \mathbf{C} \nabla f(x) = 0$, both the numerator and the denominator of the fraction above are zero. When $\nabla f(x)^\top \mathbf{C} \nabla f(x) = 0$, $u_i^\top \nabla f(x) = 0$ holds almost surely, which implies that $L(\hat{g}) = \|\nabla f(x)\|_2^2$ regardless of the value of σ . In fact, this case will not happen almost surely. In the setting of black-box attacks, we cannot even design a \mathbf{C} with trace 1 such that $\nabla f(x)^\top \mathbf{C} \nabla f(x) = 0$ since $\nabla f(x)$ is unknown.

Proof. First, we derive $L(\hat{g})$ based on the assumption that the single estimate \hat{g}_i in Eq. (5) is equal to $u_i^\top \nabla f(x) \cdot u_i$, which will hold when f is locally linear.

Lemma 1. *Assume that the single estimate \hat{g}_i in Eq. (5) is equal to $u_i^\top \nabla f(x) \cdot u_i$. We have*

$$L(\hat{g}) = \|\nabla f(x)\|_2^2 - \frac{(\nabla f(x)^\top \mathbf{C} \nabla f(x))^2}{(1 - \frac{1}{q})\nabla f(x)^\top \mathbf{C}^2 \nabla f(x) + \frac{1}{q}\nabla f(x)^\top \mathbf{C} \nabla f(x)}. \quad (\text{A.1})$$

Proof. First, we have

$$\mathbb{E}\|\nabla f(x) - b\hat{g}\|_2^2 = \|\nabla f(x)\|_2^2 - 2b\nabla f(x)^\top \mathbb{E}[\hat{g}] + b^2\mathbb{E}\|\hat{g}\|_2^2.$$

We have $\nabla f(x)^\top \mathbb{E}[\hat{g}] = \nabla f(x)^\top \mathbb{E}[\hat{g}_i] = \mathbb{E}[\nabla f(x)^\top u_i u_i^\top \nabla f(x)] = \mathbb{E}[(\nabla f(x)^\top u_i)^2] \geq 0$. Hence

$$L(\hat{g}) = \min_{b \geq 0} \mathbb{E}\|\nabla f(x) - b\hat{g}\|_2^2 = \min_b \mathbb{E}\|\nabla f(x) - b\hat{g}\|_2^2 = \|\nabla f(x)\|_2^2 - \frac{(\nabla f(x)^\top \mathbb{E}[\hat{g}])^2}{\mathbb{E}\|\hat{g}\|_2^2}. \quad (\text{A.2})$$

Since $\hat{g}_i = u_i^\top \nabla f(x) \cdot u_i$, and $u_i^\top u_i \equiv 1$, we have

$$\begin{aligned} \mathbb{E}[\hat{g}_i] &= \mathbf{C} \nabla f(x), \\ \mathbb{E}\|\hat{g}_i\|_2^2 &= \mathbb{E}[\hat{g}_i^\top \hat{g}_i] \\ &= \mathbb{E}[\nabla f(x)^\top u_i u_i^\top u_i u_i^\top \nabla f(x)] \\ &= \nabla f(x)^\top \mathbb{E}[u_i (u_i^\top u_i) u_i^\top] \nabla f(x) \\ &= \nabla f(x)^\top \mathbb{E}[u_i u_i^\top] \nabla f(x) \\ &= \nabla f(x)^\top \mathbf{C} \nabla f(x). \end{aligned}$$

Given $\mathbb{E}[\hat{g}_i]$ and $\mathbb{E}\|\hat{g}_i\|_2^2$, the corresponding moments of \hat{g} can be computed as

$$\mathbb{E}[\hat{g}] = \mathbb{E}[\hat{g}_i] = \mathbf{C} \nabla f(x), \quad (\text{A.3})$$

$$\begin{aligned} \mathbb{E}\|\hat{g}\|_2^2 &= \mathbb{E}\|\hat{g} - \mathbb{E}[\hat{g}]\|_2^2 + \|\mathbb{E}[\hat{g}]\|_2^2 \\ &= \frac{1}{q} \mathbb{E}\|\hat{g}_i - \mathbb{E}[\hat{g}_i]\|_2^2 + \|\mathbb{E}[\hat{g}_i]\|_2^2 \\ &= \frac{1}{q} \mathbb{E}\|\hat{g}_i\|_2^2 + (1 - \frac{1}{q}) \|\mathbb{E}[\hat{g}_i]\|_2^2 \\ &= (1 - \frac{1}{q}) \nabla f(x)^\top \mathbf{C}^2 \nabla f(x) + \frac{1}{q} \nabla f(x)^\top \mathbf{C} \nabla f(x). \end{aligned} \quad (\text{A.4})$$

Plug them into Eq. (A.2) and we complete the proof. \square

Next, we prove that if f is not locally linear, as long as it is differentiable at x , then by picking a sufficiently small σ , the loss tends to be that of the local linear approximation.

Lemma 2. *If f is differentiable at x , let L_0 denote the right-hand side of Eq. (A.1), then we have*

$$\lim_{\sigma \rightarrow 0} L(\hat{g}) = L_0.$$

Proof. Let $\hat{g}'_i = u_i^\top \nabla f(x) \cdot u_i$, $\hat{g}' = \frac{1}{q} \sum_{i=1}^q \hat{g}'_i$. Then $L_0 = L(\hat{g}')$. By Eq. (A.2), Eq. (A.3), and Eq. (A.4), it suffices to prove $\lim_{\sigma \rightarrow 0} \mathbb{E}[\hat{g}_i] = \mathbb{E}[\hat{g}'_i]$ and $\lim_{\sigma \rightarrow 0} \mathbb{E}\|\hat{g}_i\|_2^2 = \mathbb{E}\|\hat{g}'_i\|_2^2$.

For clarity, we redefine the notations. We omit the subscript i , make the dependence of \hat{g}_i on σ explicit (let \hat{g}_σ denote \hat{g}_i), and let \hat{g}_0 denote \hat{g}'_i . Then we omit the hat in \hat{g} . That is, let $g_0 \triangleq u^\top \nabla f(x) \cdot u$ and $g_\sigma \triangleq \frac{f(x+\sigma u) - f(x)}{\sigma} \cdot u$, where u is sampled uniformly from the unit hypersphere. Then we want to prove $\lim_{\sigma \rightarrow 0} \mathbb{E}[g_\sigma] = \mathbb{E}[g_0]$ and $\lim_{\sigma \rightarrow 0} \mathbb{E}\|g_\sigma\|_2^2 = \mathbb{E}\|g_0\|_2^2$.

Since f is differentiable at x , we have

$$\lim_{\sigma \rightarrow 0} \sup_{\|u\|_2=1} \left| \frac{f(x + \sigma u) - f(x)}{\sigma} - u^\top \nabla f(x) \right| = 0. \quad (\text{A.5})$$

Since $\|u\|_2 \equiv 1$, we have

$$\begin{aligned} \lim_{\sigma \rightarrow 0} \mathbb{E}\|g_\sigma - g_0\|_2 &\leq \lim_{\sigma \rightarrow 0} \sup_{\|u\|_2=1} \left| \frac{f(x + \sigma u) - f(x)}{\sigma} - u^\top \nabla f(x) \right| = 0, \\ \lim_{\sigma \rightarrow 0} \mathbb{E}\|g_\sigma - g_0\|_2^2 &\leq \lim_{\sigma \rightarrow 0} \sup_{\|u\|_2=1} \left| \frac{f(x + \sigma u) - f(x)}{\sigma} - u^\top \nabla f(x) \right|^2 = 0. \end{aligned}$$

By applying Jensen's inequality to convex function $\|\cdot\|_2$, we have $\|\mathbb{E}[g_\sigma] - \mathbb{E}[g_0]\|_2 \leq \mathbb{E}\|g_\sigma - g_0\|_2$. Since $\lim_{\sigma \rightarrow 0} \mathbb{E}\|g_\sigma - g_0\|_2 = 0$, and we have $\lim_{\sigma \rightarrow 0} \mathbb{E}[g_\sigma] = \mathbb{E}[g_0]$.

Since $|\|g_\sigma\|_2 - \|g_0\|_2| \leq \|g_\sigma - g_0\|_2$, $\lim_{\sigma \rightarrow 0} \mathbb{E}\|g_\sigma - g_0\|_2 = 0$ and $\lim_{\sigma \rightarrow 0} \mathbb{E}\|g_\sigma - g_0\|_2^2 = 0$, we have $\lim_{\sigma \rightarrow 0} \mathbb{E}|\|g_\sigma\|_2 - \|g_0\|_2| = 0$ and $\lim_{\sigma \rightarrow 0} \mathbb{E}(\|g_\sigma\|_2 - \|g_0\|_2)^2 = 0$. Also, we have $\|g_0\|_2 \leq \|\nabla f(x)\|_2$. Hence, we have

$$\begin{aligned} \lim_{\sigma \rightarrow 0} |\mathbb{E}\|g_\sigma\|_2^2 - \mathbb{E}\|g_0\|_2^2| &\leq \lim_{\sigma \rightarrow 0} \mathbb{E}|\|g_\sigma\|_2^2 - \|g_0\|_2^2| \\ &= \lim_{\sigma \rightarrow 0} \mathbb{E} \left[|\|g_\sigma\|_2 - \|g_0\|_2| (\|g_\sigma\|_2 + \|g_0\|_2) \right] \\ &\leq \lim_{\sigma \rightarrow 0} \mathbb{E} \left[(\|g_\sigma\|_2 - \|g_0\|_2)^2 + 2\|g_0\|_2 |\|g_\sigma\|_2 - \|g_0\|_2| \right] \\ &\leq \lim_{\sigma \rightarrow 0} \mathbb{E} \left[(\|g_\sigma\|_2 - \|g_0\|_2)^2 + 2\|\nabla f(x)\|_2 |\|g_\sigma\|_2 - \|g_0\|_2| \right] \\ &= 0. \end{aligned}$$

The proof is complete. □

By combining the two lemmas above, our proof for Theorem 1 is complete. □

A.2 Proof of Eq. (11)

Suppose that v is a fixed random vector and $\|v\|_2 = 1$. Let the D -dimensional random vector u be

$$u = \sqrt{\lambda} \cdot v + \sqrt{1 - \lambda} \cdot (\mathbf{I} - vv^\top)^\top \xi,$$

where ξ is sampled uniformly from the unit hypersphere. We need to prove that

$$\mathbf{C} \equiv \mathbb{E}[uu^\top] = \lambda vv^\top + \frac{1 - \lambda}{D - 1} (\mathbf{I} - vv^\top).$$

Proof. Let $r \triangleq \overline{(\mathbf{I} - vv^\top)}^\top \xi$. We choose an orthonormal basis $\{v_1, \dots, v_D\}$ of \mathbb{R}^D such that $v_1 = v$. Then ξ can be written as $\xi = \sum_{i=1}^D a_i v_i$, where $a = (a_1, \dots, a_D)^\top$ is sampled uniformly from the unit hypersphere. Hence $(\mathbf{I} - vv^\top)^\top \xi = \sum_{i=2}^D a_i v_i$, and $r = \frac{\sum_{i=2}^D a_i v_i}{\sqrt{\sum_{i=2}^D a_i^2}}$. Let $b_i = \frac{a_i}{\sqrt{\sum_{i=2}^D a_i^2}}$ for $i = 2, 3, \dots, D$, then $b = (b_2, b_3, \dots, b_D)^\top$ is sampled uniformly from the $(D - 1)$ -dimensional unit hypersphere, and $r = \sum_{i=2}^D b_i v_i$. Hence $\mathbb{E}[r] = 0$. To compute $\mathbb{E}[rr^\top]$, we need a lemma first.

Lemma 3. Suppose that d is a positive integer, $u = \sum_{i=1}^d a_i v_i$ where $a = (a_1, \dots, a_d)^\top$ is sampled uniformly from the d -dimensional unit hypersphere, then $\mathbb{E}[uu^\top] = \frac{1}{d} \sum_{i=1}^d v_i v_i^\top$.

Proof. $\mathbb{E}[uu^\top] = \mathbb{E}[(\sum_{i=1}^d a_i v_i)(\sum_{j=1}^d a_j v_j^\top)] = \sum_{i=1}^d \sum_{j=1}^d v_i v_j^\top \mathbb{E}[a_i a_j]$. By symmetry, we have $\mathbb{E}[a_i a_j] = 0$ when $i \neq j$, and $\mathbb{E}[a_i^2] = \mathbb{E}[a_j^2]$ for any i, j . Since $\sum_{i=1}^d a_i^2 = 1$, we have $\mathbb{E}[a_i^2] = \frac{1}{d}$ for any i . Hence $\mathbb{E}[uu^\top] = \frac{1}{d} \sum_{i=1}^d v_i v_i^\top$. □

Using the lemma, we have $\mathbb{E}[rr^\top] = \frac{1}{D-1} \sum_{i=2}^D v_i v_i^\top = \frac{1}{D-1} (\mathbf{I} - vv^\top)$. Since $\mathbb{E}[r] = 0$, we have $\mathbb{E}[vr^\top] = \mathbb{E}[rv^\top] = 0$. Hence, we have

$$\begin{aligned} \mathbb{E}[uu^\top] &= \mathbb{E}[(\sqrt{\lambda} \cdot v + \sqrt{1 - \lambda} \cdot r)(\sqrt{\lambda} \cdot v + \sqrt{1 - \lambda} \cdot r)^\top] \\ &= \lambda vv^\top + (1 - \lambda) \mathbb{E}[rr^\top] \\ &= \lambda vv^\top + \frac{1 - \lambda}{D - 1} (\mathbf{I} - vv^\top). \end{aligned}$$

The proof is complete. \square

Remark 4. The construction of the random vector u such that $\mathbb{E}[uu^\top] = \lambda vv^\top + \frac{1-\lambda}{D-1}(\mathbf{I} - vv^\top)$ is not unique. One can choose a different kind of distribution or simply take the negative of u while remaining $\mathbb{E}[uu^\top]$ invariant.

A.3 Proof of Eq. (12)

Let $\alpha = v^\top \nabla f(x)$. Suppose that $D \geq 2, q \geq 1$. After plugging Eq. (10) into Eq. (9), the optimal λ is given by

$$\lambda^* = \begin{cases} 0 & \text{if } \alpha^2 \leq \frac{1}{D+2q-2} \\ \frac{(1-\alpha^2)(\alpha^2(D+2q-2)-1)}{2\alpha^2 Dq - \alpha^4 D(D+2q-2) - 1} & \text{if } \frac{1}{D+2q-2} < \alpha^2 < \frac{2q-1}{D+2q-2} \\ 1 & \text{if } \alpha^2 \geq \frac{2q-1}{D+2q-2} \end{cases}. \quad (\text{A.6})$$

Proof. After plugging Eq. (10) into Eq. (9), we have

$$L(\lambda) = \|\nabla f(x)\|_2^2 \left(1 - \frac{(\lambda\alpha^2 + \frac{1-\lambda}{D-1}(1-\alpha^2))^2}{(1-\frac{1}{q})(\lambda^2\alpha^2 + (\frac{1-\lambda}{D-1})^2(1-\alpha^2)) + \frac{1}{q}(\lambda\alpha^2 + \frac{1-\lambda}{D-1}(1-\alpha^2))} \right).$$

To minimize $L(\lambda)$, we should maximize

$$F(\lambda) = \frac{(\lambda\alpha^2 + \frac{1-\lambda}{D-1}(1-\alpha^2))^2}{(1-\frac{1}{q})(\lambda^2\alpha^2 + (\frac{1-\lambda}{D-1})^2(1-\alpha^2)) + \frac{1}{q}(\lambda\alpha^2 + \frac{1-\lambda}{D-1}(1-\alpha^2))}. \quad (\text{A.7})$$

Note that $F(\lambda)$ is a quadratic rational function w.r.t. λ .

Since we optimize λ in a closed interval $[0, 1]$, checking $\lambda = 0$, $\lambda = 1$ and the stationary points (i.e., $F'(\lambda) = 0$) would suffice. By solving $F'(\lambda) = 0$, we have at most two solutions:

$$\begin{aligned} \lambda_1 &= \frac{(1-\alpha^2)(\alpha^2(D+2q-2)-1)}{2\alpha^2 Dq - \alpha^4 D(D+2q-2) - 1}, \\ \lambda_2 &= \frac{1-\alpha^2}{1-\alpha^2 D}, \end{aligned} \quad (\text{A.8})$$

where λ_1 or λ_2 is the solution if and only if the denominator is not 0. Given $\alpha^2 \leq 1$ and $D \geq 2$, $\lambda_2 \notin (0, 1)$, so we only need to consider λ_1 .

First, we figure out when $\lambda_1 \in (0, 1)$. We can verify that $\lambda_1 = 1$ when $\alpha^2 = 0$ and $\lambda_1 = 0$ when $\alpha^2 = 1$. Suppose that $\alpha^2 \in (0, 1)$. Let J denote the numerator in Eq. (A.8) and K denote the denominator. We have that when $\alpha^2 > \frac{1}{D+2q-2}$, $J > 0$; otherwise $J \leq 0$. We also have that when $\alpha^2 < \frac{2q-1}{D+2q-2}$, $J < K$; otherwise $J \geq K$. Note that $J/K \in (0, 1)$ if and only if $0 < J < K$ or $0 > J > K$. Hence, $\lambda_1 \in (0, 1)$ if and only if $\frac{1}{D+2q-2} < \alpha^2 < \frac{2q-1}{D+2q-2}$.

Case 1: $\lambda_1 \notin (0, 1)$. Then it suffices to compare $F(0)$ with $F(1)$. We have

$$F(0) = \frac{(1-\alpha^2)q}{D+q-2}, \quad F(1) = \alpha^2.$$

Hence, $F(0) \geq F(1)$ if and only if $\alpha^2 \leq \frac{q}{D+2q-2}$. It means that if $\alpha^2 \geq \frac{2q-1}{D+2q-2}$, then $\lambda^* = 1$; if $\alpha^2 \leq \frac{1}{D+2q-2}$, then $\lambda^* = 0$.

Case 2: $\lambda_1 \in (0, 1)$. After plugging Eq. (A.8) into Eq. (A.7), we have

$$F(\lambda_1) = \frac{4\alpha^2(1-\alpha^2)(q-1)q}{-1+2\alpha^2(D(2q-1)+2(q-1)^2)-\alpha^4(D+2q-2)^2}. \quad (\text{A.9})$$

Now we prove that $F(\lambda_1) \geq F(0)$ and $F(\lambda_1) \geq F(1)$. Since when $0 < \lambda < 1$, both the numerator and the denominator in Eq. (A.7) is positive, we have $F(\lambda) > 0, \forall \lambda \in (0, 1)$. Since the numerator in Eq. (A.9) is non-negative and $F(\lambda_1) > 0$, we know that the denominator in Eq. (A.9) is positive. Hence, we have

$$\begin{aligned} F(\lambda_1) - F(0) &= \frac{q(1-\alpha^2)(\alpha^2(D+2q-2)-1)^2}{(q+D-2)(-1+2\alpha^2(D(2q-1)+2(q-1)^2)-\alpha^4(D+2q-2)^2)} > 0; \\ F(\lambda_1) - F(1) &= \frac{\alpha^2(\alpha^2(D+2q-2)+1-2q)^2}{-1+2\alpha^2(D(2q-1)+2(q-1)^2)-\alpha^4(D+2q-2)^2} > 0. \end{aligned}$$

Hence in this case $\lambda^* = \lambda_1$.

The proof is complete. \square

A.4 Monotonicity of λ^*

We will prove that λ^* is a monotonically increasing function of α^2 , and a monotonically decreasing function of q (when $\alpha^2 > \frac{1}{D}$).

Proof. To find the monotonicity w.r.t. α^2 , note that $\lambda^* = 0$ if $\alpha^2 \leq \frac{1}{D+2q-2}$ and $\lambda^* = 1$ when $\alpha^2 \geq \frac{2q-1}{D+2q-2}$. When $\frac{1}{D+2q-2} < \alpha^2 < \frac{2q-1}{D+2q-2}$, we have

$$\begin{aligned}\lambda^* &= \frac{(1-\alpha^2)(\alpha^2(D+2q-2)-1)}{2\alpha^2 Dq - \alpha^4 D(D+2q-2) - 1} \\ &= \frac{\alpha^4(D+2q-2) - \alpha^2(D+2q-1) + 1}{\alpha^4 D(D+2q-2) - 2\alpha^2 Dq + 1} \\ &= \frac{1}{D} \left(1 - \frac{(\alpha^2 D - 1)(D-1)}{\alpha^4 D(D+2q-2) - 2\alpha^2 Dq + 1} \right) \\ &= \frac{1}{D} - \frac{D-1}{\alpha^2 D(D+2q-2) - (2Dq - D - 2q + 2) - 2\frac{(D-1)(q-1)}{\alpha^2 D-1}}.\end{aligned}\tag{A.10}$$

When $\alpha^2 < \frac{1}{D}$ or $\alpha^2 > \frac{1}{D}$, a larger α^2 leads to larger values of both $\alpha^2 D(D+2q-2)$ and $-2\frac{(D-1)(q-1)}{\alpha^2 D-1}$, and consequently leads to a larger λ^* . Meanwhile, by the argument in the proof of Eq. (12), when $\frac{1}{D+2q-2} < \alpha^2 < \frac{2q-1}{D+2q-2}$, the denominator of Eq. (A.8) is positive, hence $\alpha^4 D(D+2q-2) - 2\alpha^2 Dq + 1 < 0$. By Eq. (A.10), when $\alpha^2 < \frac{1}{D}$, $\lambda^* < \frac{1}{D}$; when $\alpha^2 = \frac{1}{D}$, $\lambda^* = \frac{1}{D}$; when $\alpha^2 > \frac{1}{D}$, $\lambda^* > \frac{1}{D}$. We conclude that λ^* is a monotonically increasing function of α^2 .

To find the monotonicity w.r.t. q when $\alpha^2 > \frac{1}{D}$, Eq. (12) tells us that when $q \leq \frac{\alpha^2(D-2)+1}{2(1-\alpha^2)}$, $\lambda^* = 1$; else, $0 < \lambda^* < 1$. In the latter case, we rewrite Eq. (A.10) as

$$\lambda^* = \frac{1}{D} \left(1 + \frac{(\alpha^2 D - 1)(D-1)}{2\alpha^2 D(1-\alpha^2)q - \alpha^4 D(D-2) - 1} \right).$$

We have $(\alpha^2 D - 1)(D-1) > 0$, and as explained before, the denominator is positive for any q such that $0 < \lambda^* < 1$. Hence, when $\alpha^2 > \frac{1}{D}$, λ^* is a monotonically decreasing function of q . \square

A.5 Proof of Theorem 2

Theorem 2. If f is differentiable at x , the loss of the gradient estimator defined in Eq. (13) is

$$\lim_{\sigma \rightarrow 0} L(\hat{g}) = \left(1 - \frac{(\mu\alpha + (1-\mu)\mathbb{E}[\beta])^2}{\mu^2 + (1-\mu)^2 + 2\mu(1-\mu)\alpha\mathbb{E}[\beta]} \right) \|\nabla f(x)\|_2^2,$$

where σ is the sampling variance to get \hat{g}^U .

Proof. As in Eq. (5), $\hat{g}^U = \frac{1}{q} \sum_{i=1}^q \hat{g}_i^U$ and $\hat{g}_i^U = \frac{f(x+\sigma u_i) - f(x)}{\sigma} \cdot u_i$, where u_i is sampled from the uniform distribution on the D -dimensional unit hypersphere. First, we derive $L(\hat{g})$ based on the assumption that \hat{g}_i^U is equal to $u_i^\top \nabla f(x) \cdot u_i$, which will hold when f is locally linear.

Lemma 4. Assume that $\hat{g}^U = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)$ (then $\beta = \overline{\hat{g}^U}^\top \overline{\nabla f(x)}$). We have

$$L(\hat{g}) = \left(1 - \frac{(\mu\alpha + (1-\mu)\mathbb{E}[\beta])^2}{\mu^2 + (1-\mu)^2 + 2\mu(1-\mu)\alpha\mathbb{E}[\beta]} \right) \|\nabla f(x)\|_2^2.$$

Proof. It can be verified³ that $\hat{g}^U = 0$ happens with probability 0, hence we only consider $\hat{g}^U \neq 0$, which does not affect our conclusion. Then $\overline{\hat{g}^U}$ is always well-defined. The distribution of \hat{g}^U is symmetric around the direction of $\nabla f(x)$, and so is the distribution of $\overline{\hat{g}^U}$. Hence we can suppose that $\mathbb{E}[\overline{\hat{g}^U}] = k \overline{\nabla f(x)}$. Since $\mathbb{E}[\beta] = \mathbb{E}[\overline{\hat{g}^U}]^\top \overline{\nabla f(x)} = k$, we have $\mathbb{E}[\overline{\hat{g}^U}] = \mathbb{E}[\beta] \overline{\nabla f(x)}$.

We have

$$\mathbb{E}[\overline{\hat{g}^U}]^\top \nabla f(x) = \mathbb{E}[\beta] \overline{\nabla f(x)}^\top \nabla f(x) = \mathbb{E}[\beta] \|\nabla f(x)\|_2,$$

and

$$v^\top \mathbb{E}[\overline{\hat{g}^U}] = v^\top \mathbb{E}[\beta] \overline{\nabla f(x)} = \alpha \mathbb{E}[\beta].$$

3. If $\hat{g}^U = 0$, $\nabla f(x)^\top \hat{g}^U = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x))^2 = 0$, hence $u_i^\top \nabla f(x) = 0$ for $i = 1, 2, \dots, q$, whose probability is 0.

Together with $v^\top \nabla f(x) = \alpha \|\nabla f(x)\|_2$ and $\|v\|_2 = 1$, we have

$$\begin{aligned}
& \mathbb{E} \|\nabla f(x) - b\hat{g}\|_2^2 \\
&= \mathbb{E} \|b\mu v + b(1-\mu)\bar{\hat{g}}^U - \nabla f(x)\|_2^2 \\
&= b^2\mu^2 + b^2(1-\mu)^2 + \|\nabla f(x)\|_2^2 + 2b^2\mu(1-\mu)v^\top \mathbb{E}[\bar{\hat{g}}^U] - 2b\mu\alpha\|\nabla f(x)\|_2 - 2b(1-\mu)\mathbb{E}[\bar{\hat{g}}^U]^\top \nabla f(x) \\
&= b^2\mu^2 + b^2(1-\mu)^2 + \|\nabla f(x)\|_2^2 + 2b^2\mu(1-\mu)\alpha\mathbb{E}[\beta] - 2b\mu\alpha\|\nabla f(x)\|_2 - 2b(1-\mu)\mathbb{E}[\beta]\|\nabla f(x)\|_2 \\
&= ((1-\mu)^2 + \mu^2 + 2\mu(1-\mu)\alpha\mathbb{E}[\beta])b^2 - 2(\alpha\mu + \mathbb{E}[\beta](1-\mu))\|\nabla f(x)\|_2 b + \|\nabla f(x)\|_2^2.
\end{aligned} \tag{A.11}$$

Since $\nabla f(x)^\top \hat{g}^U = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x))^2 \geq 0$, then $\beta \geq 0$, and hence $\mathbb{E}[\beta] \geq 0$. Then $(1-\mu)^2 + \mu^2 + 2\mu(1-\mu)\alpha\mathbb{E}[\beta] > 0$ and $\alpha\mu + \mathbb{E}[\beta](1-\mu) \geq 0$. Since $L(\hat{g}) = \min_{b \geq 0} \mathbb{E} \|\nabla f(x) - b\hat{g}\|_2^2$, by optimizing the objective w.r.t. b we complete the proof. \square

Next, we prove that if f is not locally linear, as long as it is differentiable at x , then by picking a sufficiently small σ , the loss tends to be that of the local linear approximation. Here, we redefine the notations as follows. We make the dependency of \hat{g}^U on σ explicit, i.e., we use \hat{g}_σ^U to denote it. Meanwhile, we define $\hat{g}_0^U \triangleq \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)$ as the RGF estimator under the local linear approximation. We define $\hat{g}_\sigma = \mu v + (1-\mu)\bar{\hat{g}}_\sigma^U$ and $\hat{g}_0 = \mu v + (1-\mu)\bar{\hat{g}}_0^U$. Then we have the following lemma.

Lemma 5. *If f is differentiable at x , then*

$$\lim_{\sigma \rightarrow 0} L(\hat{g}_\sigma) = L(\hat{g}_0)$$

Proof. By Eq. (A.11), it suffices to prove $\lim_{\sigma \rightarrow 0} \mathbb{E}[\bar{\hat{g}}_\sigma^U] = \mathbb{E}[\bar{\hat{g}}_0^U]$.

For any value of u_1, u_2, \dots, u_q , we have $\lim_{\sigma \rightarrow 0} \hat{g}_\sigma^U = \hat{g}_0^U$, i.e., \hat{g}_σ^U converges pointwise to \hat{g}_0^U . Recall that $\Pr(\hat{g}_0^U = 0) = 0$, so we can only consider $\hat{g}_0^U \neq 0$, which does not affect our conclusion. Since $\bar{x} = \frac{x}{\|x\|_2}$ is continuous everywhere in its domain, $\bar{\hat{g}}_\sigma^U$ converges pointwise to $\bar{\hat{g}}_0^U$. Since the family $\{\bar{\hat{g}}_\sigma^U\}$ is uniformly bounded, by dominated convergence theorem we have $\lim_{\sigma \rightarrow 0} \mathbb{E}[\bar{\hat{g}}_\sigma^U] = \mathbb{E}[\bar{\hat{g}}_0^U]$. \square

By combining the two lemmas above, our proof for Theorem 2 is complete. \square

A.6 Proof of Eq. (15)

Let \hat{g} be the PRGF-GA estimator with the balancing coefficient μ as defined in Eq. (13). Let $L(\mu) = \lim_{\sigma \rightarrow 0} L(\hat{g}) = \|\nabla f(x)\|_2^2 - \frac{(\mu\alpha + (1-\mu)\mathbb{E}[\beta])^2}{\mu^2 + (1-\mu)^2 + 2\mu(1-\mu)\alpha\mathbb{E}[\beta]} \|\nabla f(x)\|_2^2$. Then the optimal μ minimizing $L(\mu)$ is given by

$$\mu^* = \frac{\alpha(1 - \mathbb{E}[\beta]^2)}{\alpha(1 - \mathbb{E}[\beta]^2) + (1 - \alpha^2)\mathbb{E}[\beta]}.$$

Proof. To minimize $L(\mu)$, we should maximize

$$F(\mu) = \frac{(\mu\alpha + (1-\mu)\mathbb{E}[\beta])^2}{\mu^2 + (1-\mu)^2 + 2\mu(1-\mu)\alpha\mathbb{E}[\beta]}.$$

Note that $F(\mu)$ is a quadratic rational function w.r.t. μ .

Since we optimize μ in a closed interval $[0, 1]$, checking $\mu = 0$, $\mu = 1$ and the stationary points (i.e. $F'(\mu) = 0$) would suffice. By solving $F'(\mu) = 0$, we have two solutions:

$$\begin{aligned}
\mu_1 &= \frac{\alpha(1 - \mathbb{E}[\beta]^2)}{\alpha(1 - \mathbb{E}[\beta]^2) + (1 - \alpha^2)\mathbb{E}[\beta]}, \\
\mu_2 &= \frac{\mathbb{E}[\beta]}{\mathbb{E}[\beta] - \alpha},
\end{aligned}$$

where μ_2 is the solution only when $\alpha \neq \beta$. Then we have

$$\begin{aligned}
F(0) &= \mathbb{E}[\beta]^2, \\
F(1) &= \alpha^2, \\
F(\mu_1) &= \frac{\alpha^2 + \mathbb{E}[\beta]^2 - 2\alpha^2\mathbb{E}[\beta]^2}{1 - \alpha^2\mathbb{E}[\beta]^2}, \\
F(\mu_2) &= 0.
\end{aligned}$$

We have $F(0) \geq F(\mu_2)$, $F(1) \geq F(\mu_2)$, $F(\mu_1) - F(0) = \frac{\alpha^2(1 - \mathbb{E}[\beta]^2)^2}{1 - \alpha^2\mathbb{E}[\beta]^2} \geq 0$, $F(\mu_1) - F(1) = \frac{\mathbb{E}[\beta]^2(1 - \alpha^2)^2}{1 - \alpha^2\mathbb{E}[\beta]^2} \geq 0$. Therefore, the optimal solution of μ is $\mu^* = \mu_1$. \square

A.7 Proof of Eq. (16)

Let $\beta = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)^\top \nabla f(x)$, we need to prove

$$\mathbb{E}[\beta] \approx \sqrt{\frac{q}{D+q-1}},$$

where D and q are the input dimension and the number of queries to get \hat{g}_0^U , respectively.

Proof. We let $\hat{g}_0^U = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)$ as above. We can approximate $\mathbb{E}[\beta]$ by

$$\begin{aligned} \mathbb{E}[\beta] &= \mathbb{E}[\sqrt{\beta^2}] \\ &\approx \sqrt{\mathbb{E}[\beta^2]} \\ &= \sqrt{1 - \mathbb{E}[\min_b \|\nabla f(x) - b\hat{g}_0^U\|^2]} \\ &= \sqrt{1 - \frac{1}{\|\nabla f(x)\|_2^2} \mathbb{E}[\min_b \|\nabla f(x) - b\hat{g}_0^U\|^2]} \\ &\approx \sqrt{1 - \frac{1}{\|\nabla f(x)\|_2^2} \min_b \mathbb{E}[\|\nabla f(x) - b\hat{g}_0^U\|^2]} \\ &= \sqrt{1 - \frac{1}{\|\nabla f(x)\|_2^2} L(\hat{g}_0^U)^2}. \end{aligned}$$

Here, the first equality is because that $\nabla f(x)^\top \hat{g}_0^U = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x))^2 \geq 0$ and the second equality is because that we have $\min_b \|\nabla f(x) - b\hat{g}_0^U\|^2 = 1 - (\nabla f(x)^\top \hat{g}_0^U)^2 = 1 - \beta^2$. Intuitively, the two approximations work well because that the variances of β and $\|\hat{g}_0^U\|_2$ are relatively small.

Now we define $F(\hat{g}_0^U) = 1 - \frac{1}{\|\nabla f(x)\|_2^2} L(\hat{g}_0^U)^2$. Then we have $\mathbb{E}[\beta] \approx \sqrt{F(\hat{g}_0^U)}$. Note that when u_i is sampled from the uniform distribution on the unit hypersphere, $F(\hat{g}_0^U)$ is in fact $F(\frac{1}{D})$ in Eq. (A.7), since \hat{g}_0^U is an RGF estimator w.r.t. locally linear f , and $\mathbb{E}[u_i u_i^\top] = \frac{1}{D} \mathbf{I}$ which corresponds to $\lambda = \frac{1}{D}$ in Eq. (10). We can calculate $F(\frac{1}{D}) = \frac{q}{D+q-1}$. Hence, $\mathbb{E}[\beta] \approx \sqrt{\frac{q}{D+q-1}}$. \square

A.8 Proof of Eq. (22)

Let $\alpha = v^\top \nabla f(x)$, $A^2 = \sum_{j=1}^d (v_j^\top \nabla f(x))^2$. Suppose that $\alpha^2 \leq 1$, $d \geq 1$, $q \geq 1$. After plugging Eq. (21) into Eq. (9), the optimal λ is given by

$$\lambda^* = \begin{cases} 0 & \text{if } \alpha^2 \leq \frac{A^2}{d+2q-2} \\ \frac{A^2(A^2 - \alpha^2(d+2q-2))}{A^4 + \alpha^4 d^2 - 2A^2 \alpha^2(q+dq-1)} & \text{if } \frac{A^2}{d+2q-2} < \alpha^2 < \frac{A^2(2q-1)}{d} \\ 1 & \text{if } \alpha^2 \geq \frac{A^2(2q-1)}{d} \end{cases}.$$

Proof. The proof is very similar to that in Appendix A.3. After plugging Eq. (21) into Eq. (9), we have

$$L(\lambda) = \|\nabla f(x)\|_2^2 \left(1 - \frac{(\lambda \alpha^2 + \frac{1-\lambda}{d} A^2)^2}{(1 - \frac{1}{q})(\lambda^2 \alpha^2 + (\frac{1-\lambda}{d})^2 A^2) + \frac{1}{q}(\lambda \alpha^2 + \frac{1-\lambda}{d} A^2)} \right).$$

To minimize $L(\lambda)$, we should maximize

$$F(\lambda) = \frac{(\lambda \alpha^2 + \frac{1-\lambda}{d} A^2)^2}{(1 - \frac{1}{q})(\lambda^2 \alpha^2 + (\frac{1-\lambda}{d})^2 A^2) + \frac{1}{q}(\lambda \alpha^2 + \frac{1-\lambda}{d} A^2)}. \quad (\text{A.12})$$

Note that $F(\lambda)$ is a quadratic rational function w.r.t. λ .

Since we optimize λ in a closed interval $[0, 1]$, checking $\lambda = 0$, $\lambda = 1$ and the stationary points (i.e., $F'(\lambda) = 0$) would suffice. By solving $F'(\lambda) = 0$, we have at most two solutions:

$$\begin{aligned} \lambda_1 &= \frac{A^2(\alpha^2(d+2q-2) - A^2)}{2A^2 \alpha^2(dq+q-1) - \alpha^4 d^2 - A^4}, \\ \lambda_2 &= \frac{A^2}{A^2 - \alpha^2 d}, \end{aligned} \quad (\text{A.13})$$

where λ_1 or λ_2 is the solution if and only if the denominator is not 0. $\lambda_2 \notin (0, 1)$, so we only need to consider λ_1 .

First, we figure out when $\lambda_1 \in (0, 1)$. We can verify that $\lambda_1 = 1$ when $\alpha^2 = 0$ and $\lambda_1 = 0$ when $A^2 = 0$. Suppose $\alpha^2 \neq 0$ and $A^2 \neq 0$. Let J denote the numerator in Eq. (A.13) and K denote the denominator. We have that when $\alpha^2 > \frac{A^2}{d+2q-2}$, $J > 0$; otherwise $J \leq 0$. We also have that when $\alpha^2 < \frac{A^2(2q-1)}{d}$, $J < K$; otherwise $J \geq K$. Note that $J/K \in (0, 1)$ if and only if $0 < J < K$ or $0 > J > K$. Hence, $\lambda_1 \in (0, 1)$ if and only if $\frac{A^2}{d+2q-2} < \alpha^2 < \frac{A^2(2q-1)}{d}$.

Case 1: $\lambda_1 \notin (0, 1)$. Then it suffices to compare $F(0)$ and $F(1)$. We have

$$F(0) = \frac{A^2 q}{d+q-1}, F(1) = \alpha^2.$$

Hence, $F(0) \geq F(1)$ if and only if $\alpha^2 \leq \frac{A^2 q}{d+q-1}$. It means that if $\alpha^2 \geq \frac{A^2(2q-1)}{d}$, then $\lambda^* = 1$; if $\alpha^2 \leq \frac{A^2}{d+2q-2}$, then $\lambda^* = 0$.

Case 2: $\lambda_1 \in (0, 1)$. After plugging Eq. (A.13) into Eq. (A.12), we have

$$F(\lambda_1) = \frac{4A^2\alpha^2(A^2 + \alpha^2)(q-1)q}{2A^2\alpha^2(2q(d+q-1)-d) - \alpha^4 d^2 - A^4}. \quad (\text{A.14})$$

Now we prove that $F(\lambda_1) \geq F(0)$ and $F(\lambda_1) \geq F(1)$. Since when $0 < \lambda < 1$, both the numerator and the denominator in Eq. (A.12) is positive, we have $F(\lambda) > 0, \forall \lambda \in (0, 1)$. Since the numerator in Eq. (A.14) is non-negative, and $F(\lambda_1) > 0$, we know that the denominator in Eq. (A.14) is positive. Hence, we have

$$\begin{aligned} F(\lambda_1) - F(0) &= \frac{qA^2(\alpha^2(d+2q-2) - A^2)^2}{(q+d-1)(2A^2\alpha^2(2q(d+q-1)-d) - \alpha^4 d^2 - A^4)} > 0; \\ F(\lambda_1) - F(1) &= \frac{\alpha^2(\alpha^2 d + A^2(1-2q))^2}{2A^2\alpha^2(2q(d+q-1)-d) - \alpha^4 d^2 - A^4} > 0. \end{aligned}$$

Hence in this case $\lambda^* = \lambda_1$.

The proof is complete. \square

A.9 Explanation on Eq. (23)

We explain why the construction of u_i in Eq. (23) makes $\mathbb{E}[u_i u_i^\top]$ a good approximation of \mathbf{C} .

Recall the setting: In \mathbb{R}^D , we have a normalized transfer gradient v , and a specified d -dimensional subspace with $\{v_1, \dots, v_d\}$ as its orthonormal basis. Let $\mathbf{C} = \lambda v v^\top + \frac{1-\lambda}{d} \sum_{j=1}^d v_j v_j^\top$. Here we argue that if $u = \sqrt{\lambda} \cdot v + \sqrt{1-\lambda} \cdot (\mathbf{I} - v v^\top) \mathbf{V} \xi$, then $\mathbb{E}[u u^\top] \approx \mathbf{C}$.

Let $r \triangleq (\mathbf{I} - v v^\top) \mathbf{V} \xi$. The reason why $\mathbb{E}[u u^\top] \neq \mathbf{C}$ is that $\mathbb{E}[r r^\top] \neq \frac{1}{d} \sum_{j=1}^d v_j v_j^\top$ when v is not orthogonal to the subspace spanned by $\{v_1, \dots, v_d\}$. However, by symmetry, we still have $\mathbb{E}[r] = 0$. To get an expression of $\mathbb{E}[r r^\top]$, we let v_T denotes the projection of v onto the subspace, and let $v_1 = \overline{v_T}$ so that v_2, \dots, v_d are orthonormal to v_T (hence also orthonormal to v). We temporarily assume $v_T \neq v$ and $v_T \neq 0$. Now let $v'_1 = (\mathbf{I} - v v^\top) v_T = v_T - v^\top v_T \cdot v$, then $\{v'_1, v_2, \dots, v_d\}$ form an orthonormal basis of the subspace in which r lies, and v is orthogonal to this modified subspace. Now we have $\mathbb{E}[r r^\top] = \lambda_1 v'_1 v'_1{}^\top + \frac{1-\lambda_1}{d-1} \sum_{j=2}^d v_j v_j^\top$ where λ_1 is a number in $[0, \frac{1}{d}]$. Note that when $v = v_T$, although v'_1 cannot be defined, we have $\lambda_1 = 0$. When $v_T = 0$, we can just set $v'_1 = v_1$ and $\lambda_1 = \frac{1}{d}$. When d is large, λ_1 is small, so for approximation we can replace v'_1 with v_1 ; $|\lambda - \frac{1}{d}|$ is small, so for approximation we can set $\lambda_1 = \frac{1}{d}$. Then we have $\mathbb{E}[r r^\top] \approx \frac{1}{d} \sum_{j=1}^d v_j v_j^\top$. Since $\mathbb{E}[r] = 0$, we have $\mathbb{E}[u u^\top] = \lambda v v^\top + (1-\lambda) \mathbb{E}[r r^\top] \approx \lambda v v^\top + \frac{1-\lambda}{d} \sum_{j=1}^d v_j v_j^\top$.

Remark 5. To avoid approximation, one can choose the subspace as spanned by $\{v'_1, v_2, \dots, v_d\}$ instead of $\{v_1, v_2, \dots, v_d\}$ to ensure that v is orthogonal to the subspace. Then u can be sampled as

$$u = \sqrt{\lambda} \cdot v + \sqrt{1-\lambda} \cdot \overline{\mathbf{V}' \xi},$$

where $\mathbf{V}' = [v'_1, v_2, \dots, v_d]$ and ξ is sampled uniformly from the d -dimensional unit hypersphere. Note that here the optimal λ is calculated using $A'^2 = v_1^\top \nabla f(x) + \sum_{j=2}^d (v_j^\top \nabla f(x))^2$. However, in practice, it is not convenient to make the subspace dependent on v , and the computational complexity is high to construct an orthonormal basis with one vector (v'_1) specified.

A.10 Proof of Theorem 3

Theorem 3. Let $\alpha_1 = v^\top \nabla f(x)_T$. If f is differentiable at x and $A^2 > 0$, the loss of the gradient estimator define in Eq. (24) is

$$\lim_{\sigma \rightarrow 0} L(\hat{g}) = \left(1 - \frac{(\mu\alpha + (1-\mu)\mathbb{E}[\beta])^2}{\mu^2 + (1-\mu)^2 + 2\mu(1-\mu)\frac{\alpha_1}{A^2}\mathbb{E}[\beta]} \right) \|\nabla f(x)\|^2,$$

where σ is the sampling variance to get \hat{g}^S .

Proof. Similar to the proof of Theorem 2, we define $\hat{g}_0^S = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i) = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x)_T \cdot u_i)$, where $\nabla f(x)_T = \|\nabla f(x)\|_2 \overline{\nabla f(x)_T}$ denotes the projection of $\nabla f(x)$ onto the subspace. Then $\beta = \hat{g}_0^S \cdot \overline{\nabla f(x)} = \hat{g}_0^S \cdot \overline{\nabla f(x)_T}$. Since $A^2 > 0$, we have $\nabla f(x)_T \neq 0$. As described in Footnote 3, we can prove $\Pr(\hat{g}_0^S = 0) = 0$ similarly. Now we only

consider $\hat{g}_0^S \neq 0$. The distribution of \hat{g}_0^S is symmetric around the direction of $\nabla f(x)_T$, and so is the distribution of $\overline{\hat{g}_0^S}$. Hence we can suppose that $\mathbb{E}[\hat{g}_0^S] = k \nabla f(x)_T$. Since $\mathbb{E}[\beta] = \mathbb{E}[\hat{g}_0^S]^\top \nabla f(x)_T = k \|\nabla f(x)_T\|_2^2 = kA^2$, we have $\mathbb{E}[\hat{g}_0^S] = \frac{\mathbb{E}[\beta]}{A^2} \nabla f(x)_T$.

Note that

$$v^\top \mathbb{E}[\hat{g}_0^S] = v^\top \frac{\mathbb{E}[\beta]}{A^2} \nabla f(x)_T = \frac{\alpha_1}{A^2} \mathbb{E}[\beta].$$

The rest of the proof is the same as that of Theorem 2. \square

A.11 Proof of Eq. (26)

Let \hat{g} be the PRGF-GA estimator incorporating the data-dependent prior with the balancing coefficient μ as defined in Eq. (24). Let $L(\mu) = \lim_{\sigma \rightarrow 0} L(\hat{g}) = \|\nabla f(x)\|_2^2 - \frac{(\mu\alpha + (1-\mu)\mathbb{E}[\beta])^2}{\mu^2 + (1-\mu)^2 + 2\mu(1-\mu)\frac{\alpha_1}{A^2}\mathbb{E}[\beta]}\|\nabla f(x)\|_2^2$. Then the optimal μ minimizing $L(\mu)$ is given by

$$\mu^* = \frac{A^2\alpha - \alpha_1\mathbb{E}[\beta]^2}{(A^2 - \alpha_1\mathbb{E}[\beta])(\alpha + \mathbb{E}[\beta])}.$$

Proof. The proof is very similar to that in Appendix A.6. To minimize $L(\mu)$, we should maximize

$$F(\mu) = \frac{(\mu\alpha + (1-\mu)\mathbb{E}[\beta])^2}{\mu^2 + (1-\mu)^2 + 2\mu(1-\mu)\frac{\alpha_1}{A^2}\mathbb{E}[\beta]}.$$

Note that $F(\mu)$ is a quadratic rational function w.r.t. μ .

Since we optimize μ in a closed interval $[0, 1]$, checking $\mu = 0$, $\mu = 1$ and the stationary points (i.e. $F'(\mu) = 0$) would suffice. By solving $F'(\mu) = 0$, we have two solutions:

$$\begin{aligned} \mu_1 &= \frac{A^2\alpha - \alpha_1\mathbb{E}[\beta]^2}{(A^2 - \alpha_1\mathbb{E}[\beta])(\alpha + \mathbb{E}[\beta])}, \\ \mu_2 &= \frac{\mathbb{E}[\beta]}{\mathbb{E}[\beta] - \alpha}, \end{aligned}$$

where μ_2 is the solution only when $\alpha \neq \beta$. Then we have

$$\begin{aligned} F(0) &= \mathbb{E}[\beta]^2, \\ F(1) &= \alpha^2, \\ F(\mu_1) &= \frac{A^4(\alpha^2 + \mathbb{E}[\beta]^2) - 2A^2\alpha\alpha_1\mathbb{E}[\beta]^2}{A^4 - \alpha_1^2\mathbb{E}[\beta]^2}, \\ F(\mu_2) &= 0. \end{aligned}$$

We have $F(0) \geq F(\mu_2)$, $F(1) \geq F(\mu_2)$, $F(\mu_1) - F(0) = \frac{(A^2\alpha - \alpha_1\mathbb{E}[\beta]^2)^2}{A^4 - \alpha_1^2\mathbb{E}[\beta]^2} \geq 0$, $F(\mu_1) - F(1) = \frac{\mathbb{E}[\beta]^2(A^2 - \alpha\alpha_1)^2}{A^4 - \alpha_1^2\mathbb{E}[\beta]^2} \geq 0$. Therefore, the optimal solution of μ is $\mu^* = \mu_1$. \square

Let $\beta = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)^\top \nabla f(x)$, in which $\{u_i\}_{i=1}^q$ lie in the subspace, we further need to prove

$$\mathbb{E}[\beta] \approx A \sqrt{\frac{q}{d+q-1}},$$

where d is the subspace dimension, q is the number of queries to get \hat{g}^S , and $A^2 = \sum_{i=1}^d (v_i^\top \nabla f(x))^2$.

Proof. Similar to the proof in Appendix A.7, we approximate $\mathbb{E}[\beta]$ by $\sqrt{F(\hat{g}_0^S)}$, in which $F(\hat{g}_0^S) = 1 - \frac{1}{\|\nabla f(x)\|_2^2} L(\hat{g}_0^S)^2$, and $\hat{g}_0^S = \frac{1}{q} \sum_{i=1}^q (u_i^\top \nabla f(x) \cdot u_i)$. Note that when u_i is sampled from the uniform distribution on the unit hypersphere in the subspace, $F(\hat{g}_0^S)$ is in fact $F(0)$ in Eq. (A.12), since \hat{g}_0^S is an RGF estimator w.r.t. locally linear f , and $\mathbb{E}[u_i u_i^\top] = \frac{1}{d} \sum_{i=1}^d v_i v_i^\top$ which corresponds to $\lambda = 0$ in Eq. (21). We can calculate $F(0) = \frac{A^2 q}{d+q-1}$. Hence, $\mathbb{E}[\beta] \approx A \sqrt{\frac{q}{d+q-1}}$. \square

APPENDIX B

ACTUAL IMPLEMENTATION OF PRGF-GA

Note that in the PRGF-GA algorithm, the optimal coefficient μ^* in Eq. (15) is calculated by minimizing the loss $L(\hat{g})$ of the gradient estimator defined as $\hat{g} = \mu v + (1-\mu)\hat{g}^U$, where v is the normalized transfer gradient and \hat{g}^U is the ordinary RGF estimator. Since the loss $L(\hat{g})$ is a deterministic scalar whose computation requires taking expectation w.r.t. the randomness of \hat{g}^U , μ^* is a precomputed scalar which does not depend on the value of \hat{g}^U . However, since μ is not concerned with the estimation process to get \hat{g}^U , we can actually obtain the value of \hat{g}^U first and let μ depend on it, which could be beneficial when \hat{g}^U exhibits high variance.

Algorithm 3 Actual implementation of prior-guided random gradient-free algorithm based on gradient averaging (PRGF-GA)

Input: The black-box model f ; input x and label y ; the normalized transfer gradient v ; sampling variance σ ; number of queries q ; input dimension D ; threshold c .

Output: Estimate of the gradient $\nabla f(x)$.

- 1: Estimate the cosine similarity $\alpha = v^\top \nabla f(x)$ (detailed in Section 4.3);
- 2: Approximate $\mathbb{E}[\beta]$ by $\sqrt{\frac{q}{D+q-1}}$ as in Eq. (16);
- 3: Calculate μ^* according to Eq. (15) given α and $\mathbb{E}[\beta]$;
- 4: **if** $\mu^* \geq c$ **then**
- 5: **return** v ;
- 6: **end if**
- 7: $\hat{g}^U \leftarrow \mathbf{0}$;
- 8: **for** $i = 1$ to q **do**
- 9: Sample u_i from the uniform distribution on the D -dimensional unit hypersphere;
- 10: $\hat{g}^U \leftarrow \hat{g}^U + \frac{f(x + \sigma u_i, y) - f(x, y)}{\sigma} \cdot u_i$;
- 11: **end for**
- 12: Estimate $v^\top \nabla f(x)$ by $\frac{f(x + \sigma v, y) - f(x, y)}{\sigma}$; Estimate $\hat{g}^U^\top \nabla f(x)$ by $\frac{f(x + \sigma \hat{g}^U, y) - f(x, y)}{\sigma}$;
- 13: **return** $\nabla f(x) \leftarrow v^\top \nabla f(x) \cdot v + \hat{g}^U^\top \nabla f(x) \cdot \hat{g}^U$.

To this end, we need to calculate μ that leads to the best gradient estimator given the values of v and \hat{g}^U . We first assume that v and \hat{g}^U are almost orthogonal with high probability, which is true in a high dimensional input space. (Without this assumption, we could perform Gram–Schmidt orthonormalization.) The problem is to find a vector in the subspace spanned by v and \hat{g}^U that approximate the true gradient $\nabla f(x)$ best. This can be simply accomplished by projecting $\nabla f(x)$ onto the subspace, as

$$\hat{g} = v^\top \nabla f(x) \cdot v + \hat{g}^U^\top \nabla f(x) \cdot \hat{g}^U. \quad (\text{B.1})$$

Therefore, the optimal μ can be expressed as

$$\mu^* = \frac{v^\top \nabla f(x)}{v^\top \nabla f(x) + \hat{g}^U^\top \nabla f(x)}. \quad (\text{B.2})$$

$v^\top \nabla f(x)$ and $\hat{g}^U^\top \nabla f(x)$ can be estimated by the finite difference method shown in Eq. (17). We summarize the actual implementation of PRGF-GA in Algorithm 3.

APPENDIX C ESTIMATION OF A

Suppose that the subspace is spanned by a set of orthonormal vectors $\{v_1, \dots, v_d\}$. Now we want to estimate

$$A^2 = \sum_{j=1}^d (v_j^\top \nabla f(x))^2 = \frac{\sum_{j=1}^d (v_j^\top \nabla f(x))^2}{\|\nabla f(x)\|_2^2} = \frac{\|h(x)\|_2^2}{\|\nabla f(x)\|_2^2},$$

where $h(x) = \sum_{j=1}^d v_j^\top \nabla f(x) \cdot v_j$ is the projection of $\nabla f(x)$ to the subspace. We can estimate $\|\nabla f(x)\|_2^2$ using the method introduced in Section 4.3. Here, we introduce the method to estimate $\|h(x)\|_2^2$.

Let $w = \mathbf{V}\xi$ where $\mathbf{V} = [v_1, v_2, \dots, v_d]$ and ξ is a random vector uniformly sampled from the d -dimensional unit hypersphere. By Lemma 3, $\mathbb{E}[ww^\top] = \frac{1}{d} \sum_{j=1}^d v_j v_j^\top$. Suppose that we have S i.i.d. such samples of w denoted by w_1, \dots, w_S , and we let $\mathbf{W} = [w_1, \dots, w_S]$.

With $g(x_1, \dots, x_S) = \frac{1}{S} \sum_{s=1}^S x_s^2$, we have

$$g(\mathbf{W}^\top \nabla f(x)) = g(\mathbf{W}^\top h(x)) = \|h(x)\|_2^2 \cdot g(\mathbf{W}^\top \overline{h(x)}).$$

Hence $\frac{g(\mathbf{W}^\top \nabla f(x))}{\mathbb{E}[g(\mathbf{W}^\top \overline{h(x)})]}$ is an unbiased estimator of $\|h(x)\|_2^2$. Now, $\overline{h(x)}$ is in the subspace spanned by $\{v_1, \dots, v_d\}$, and w_1 is uniformly distributed on the unit hypersphere of this subspace. Hence $\mathbb{E}[(w_1^\top \overline{h(x)})^2]$ is independent of the direction of $\overline{h(x)}$ and can be computed. We have

$$\mathbb{E}[g(\mathbf{W}^\top \overline{h(x)})] = \mathbb{E}[(w_1^\top \overline{h(x)})^2] = \overline{h(x)}^\top \mathbb{E}[w_1 w_1^\top] \overline{h(x)} = \overline{h(x)}^\top \frac{1}{d} \sum_{i=1}^d v_i v_i^\top \overline{h(x)} = \frac{1}{d}.$$

Hence, we have the estimator $\|h(x)\|_2 \approx \sqrt{\frac{d}{S} \sum_{s=1}^S (w_s^\top \nabla f(x))^2}$, where $w_s = \mathbf{V}\xi_s$ and ξ_s is uniformly sampled from the unit hypersphere in \mathbb{R}^d . Finally we can get an estimate of A by $A = \frac{\|h(x)\|_2}{\|\nabla f(x)\|_2}$.

TABLE 4

The experimental results of black-box attacks against Inception-v3, VGG-16, and ResNet-50 under the ℓ_∞ norm on ImageNet. We report the attack success rate (ASR), and the average/median number of queries (AVG. Q/MED. Q) needed to generate an adversarial example over successful attacks. We mark the best results in **bold**.

Methods	Inception-v3			VGG-16			ResNet-50		
	ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q
NES [12]	87.5%	1887	1122	95.6%	1507	1020	96.5%	1433	969
SPSA [44]	93.6%	1766	1020	98.1%	1198	918	98.4%	1166	867
Bandits _T [19]	89.5%	1891	952	93.8%	585	175	95.2%	1199	458
Bandits _{TD} [19]	94.7%	1099	330	95.1%	288	46	96.5%	651	158
\mathcal{N} ATTACK [45]	98.3%	1101	612	99.7%	639	408	99.5%	588	408
RGF	94.4%	1565	816	98.8%	1064	714	99.4%	990	663
PRGF-BS ($\lambda = 0.05$)	92.7%	1409	714	97.5%	1031	612	98.3%	891	561
PRGF-BS (λ^*)	93.8%	979	414	98.5%	635	306	99.0%	507	236
PRGF-GA ($\mu = 0.5$)	94.9%	1263	624	98.9%	851	520	99.2%	758	468
PRGF-GA (μ^*)	94.8%	974	424	98.5%	560	298	99.3%	490	226
RGF _D	97.2%	1034	561	100.0%	502	383	99.7%	595	408
PRGF-BS _D ($\lambda = 0.05$)	97.7%	1005	510	99.9%	543	408	99.7%	598	408
PRGF-BS _D (λ^*)	97.3%	812	384	99.7%	370	262	99.6%	388	234
PRGF-GA _D ($\mu = 0.5$)	98.0%	898	468	100.0%	481	364	99.8%	504	364
PRGF-GA _D (μ^*)	98.4%	772	364	99.7%	374	246	99.6%	365	240

TABLE 5

The experimental results of black-box attacks against ResNet-50, DenseNet-121, and SENet-18 under the ℓ_∞ norm on CIFAR-10. We report the attack success rate (ASR) and the average/median number of queries (AVG. Q/MED. Q) needed to generate an adversarial example over successful attacks. We mark the best results (including ASR $\geq 99.9\%$) in **bold**.

Methods	ResNet-50			DenseNet-121			SENet-18		
	ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q	ASR	AVG. Q	MED. Q
NES [12]	93.9%	781	408	96.1%	742	408	95.8%	699	357
SPSA [44]	99.9%	627	408	99.8%	622	408	99.9%	571	357
Bandits _T [19]	100.0%	372	186	100.0%	345	156	100.0%	312	142
\mathcal{N} ATTACK [45]	100.0%	384	255	100.0%	383	255	100.0%	343	204
RGF	98.4%	524	306	99.0%	499	306	99.1%	470	255
PRGF-BS ($\lambda = 0.05$)	99.2%	331	153	99.7%	275	153	99.7%	261	153
PRGF-BS (λ^*)	99.6%	213	78	99.9%	206	113	99.9%	178	74
PRGF-GA ($\mu = 0.5$)	99.1%	310	153	99.7%	259	153	99.8%	229	153
PRGF-GA (μ^*)	99.6%	184	65	99.9%	156	65	99.9%	140	64

APPENDIX D

ADDITIONAL EXPERIMENTS

We show the experimental results of black-box adversarial attacks under the ℓ_∞ norm on ImageNet in Table 4, and on CIFAR-10 in Table 5.