The Adversarial Attack and Detection under the Fisher Information Metric

Chenxiao Zhao,[*1] **P. Thomas Fletcher,**[2] **Mixue Yu,**[1] **Yaxin Peng,**[3,4] **Guixu Zhang,**[1] **Chaomin Shen**[†1,4]
[1]Department of Computer Science, East China Normal University, Shanghai, China
[2]Department of Electrical and Computer Engineering, and Department of Computer Science, University of Virginia, Virginia, USA
[3]Department of Mathematics, Shanghai University, Shanghai, China
[4]Westlake Institute for Brain-Like Science and Technology, Zhejiang, China

Abstract

Many deep learning models are vulnerable to the adversarial attack, i.e., imperceptible but intentionally-designed perturbations to the input can cause incorrect output of the networks. In this paper, using information geometry, we provide a reasonable explanation for the vulnerability of deep learning models. By considering the data space as a non-linear space with the Fisher information metric induced from a neural network, we first propose an adversarial attack algorithm termed one-step spectral attack (OSSA). The method is described by a constrained quadratic form of the Fisher information matrix, where the optimal adversarial perturbation is given by the first eigenvector, and the vulnerability is reflected by the eigenvalues. The larger an eigenvalue is, the more vulnerable the model is to be attacked by the corresponding eigenvector. Taking advantage of the property, we also propose an adversarial detection method with the eigenvalues serving as characteristics. Both our attack and detection algorithms are numerically optimized to work efficiently on large datasets. Our evaluations show superior performance compared with other methods, implying that the Fisher information is a promising approach to investigate the adversarial attacks and defenses.

1 Introduction

Deep learning models have achieved substantial achievements on various of computer vision tasks. Recent studies suggest that, however, even though a well-trained neural network generalizes well on the test set, it is still vulnerable to adversarial attacks (Szegedy et al. 2013). For image classification tasks, the perturbations applied to the images can be imperceptible for human perception, meanwhile misclassified by networks with a high rate. Moreover, empirical evidence has shown that the adversarial examples have the ability to transfer among different deep learning models. The adversarial examples generated from one model can often fool other models which have totally different structure and parameters (Papernot, McDaniel, and Goodfellow 2016), thus making the malicious black-box attack possible. Many deep learning applications, e.g. the automated vehicles and face authentication system, have low error-tolerance rate and are sensitive to the attacks. The existence of adversarial examples has raised severe challenges for deep learning models in security-critical computer vision applications.

Understanding the mechanism of adversarial examples is a fundamental problem for defending against the attacks. Many explanations have been proposed from different facets. (Szegedy et al. 2013) first observes the existence of adversarial examples, and suggests it is due to the excessive non-linearity of the neural networks. On the contrary, (Goodfellow, Shlens, and Szegedy 2014) suggests that the vulnerability results from the models being too linear. Despite its contradiction to the general impression, the explanation is supported by numbers of experimental results (Krotov and Hopfield 2017; Tabacof and Valle 2016; Tanay and Griffin 2016; Tramèr et al. 2017). On the other hand, by approximating the vertical direction of the decision boundary in the sample space, (Moosavidezfooli, Fawzi, and Frossard 2016) proposes to find the closest adversarial examples to the input with an iterative algorithm. (Moosavidezfooli et al. 2017b) further studies the existence of universal perturbations in state-of-the-art deep neural networks. They suggest the phenomenon is resulted from the high curvature regions on the decision boundary (Moosavidezfooli et al. 2017a).

These works have built both intuitive and theoretical understanding of the adversarial examples under the Euclidean metric. However, studying adversarial examples by the Euclidean metric has its limitations. Intrinsically, for neural networks, the adversarial attacking is about the correlation between the input space and the output space. Due to the complexity of the networks, it is hard to explain why small perturbation in the input space can result in large variation in the output space. Many previous attack methods presume the input space is flat, thus the gradient with respect to the input gives the fastest changing direction in the output space. However, if we regard the model output as the likelihood of the discrete distribution, and regard the model input as the pullback of the output, a meaningful distance measure for the likelihood will not be linear, making the sample space a manifold measured by a non-linear Riemannian metric. This motivates us to adopt the **Fisher information matrix (FIM)** of the input as a metric tensor to measure the vulnerability of deep learning models.

The significance of introducing the Fisher information metric is three folds. First, the FIM is the Hessian matrix of the

[*]Email: 51174506043@stu.ecnu.edu.cn
[†]Corresponding author. Email: cmshen@cs.ecnu.edu.cn

Kullback-Leibler (KL) divergence, which is a meaningful metric for probability distributions. Second, the FIM is symmetrical and positive semi-definite, making the optimization on the matrix easy and efficient. Third, the FIM is invariant to reparameterization as long as the likelihood does not change. This is particularly important for bypassing the influence of irrelevant variables (e.g. different network structures), and identifying the true cause for the vulnerability of deep learning models.

Based on these insights, we propose a novel algorithm to attack the neural networks. In our algorithm, the optimization is described by a constrained quadratic form of the FIM, where the optimal adversarial perturbation is given by the eigenvector, and the eigenvalues reflect the local vulnerability. Compared with previous attacking methods, our algorithm can efficiently characterize multiple adversarial subspaces with the eigenvalues. In order to overcome the difficulty in computational complexity, we then introduce some numerical tricks to make the optimization work on large datasets. We also give a detailed proof for the optimality of the adversarial perturbations under certain technical conditions, showing that the adversarial perturbations obtained by our method will not be "compressed" during the mapping of networks, which has contributed to the vulnerability of deep learning models.

Furthermore, we perform binary search for the least adversarial perturbation that can fool the networks, so as to verify the eigenvalues' ability to characterize the local vulnerability: the larger the eigenvalues are, the more vulnerable the model is to be attacked by the perturbation of corresponding eigenvectors. Hence we adopt the eigenvalues of the FIM as features, and train an auxiliary classifier to detect the adversarial attacks with the eigenvalues. We perform extensive empirical evaluations, demonstrating that the eigenvalues are of good distinguishability for defending against many state-of-the-art attacks.

Our main contributions in this paper are summarized as follows:

- We propose a novel algorithm to attack deep neural networks based on information geometry. The algorithm can characterize multiple adversarial subspaces in the neighborhood of a given sample, and achieves high fooling ratio under various conditions.
- We propose to adopt the eigenvalues of the FIM as features to detect the adversarial attacks. Our analysis shows the classifiers with the eigenvalues being their features are robust to various state-of-the-art attacks.
- We provide a novel geometrical interpretation for the deep learning vulnerability. The theoretical results confirm the optimality of our attack method, and serve as a basis for characterizing the vulnerability of deep learning models.

2 Preliminaries

Fisher information The Fisher information is initially proposed to measure the variance of the likelihood estimation given by a statistical model. Then the idea was extended by introducing differential geometry to statistics (Amari and Nagaoka 2007). By considering the FIM of the exponential family distributions as the Riemannian metric tensor, Chenstov further proves that the FIM as a Riemannian measure is the only invariant measure for distributions. Specifically, let $p(x|\boldsymbol{z})$ be a likelihood function given by a statistical model, where $\boldsymbol{z}$ is the model parameter, the Fisher information of $\boldsymbol{z}$ has the following equivalent forms:

$$\begin{aligned}\boldsymbol{G}_{\boldsymbol{z}} &= \mathbb{E}_{x|\boldsymbol{z}}[(\nabla_{\boldsymbol{z}} \log p(x|\boldsymbol{z}))(\nabla_{\boldsymbol{z}} \log p(x|\boldsymbol{z}))^T] \\ &= \mathbb{D}_{x|\boldsymbol{z}}[\nabla_{\boldsymbol{z}} \log p(x|\boldsymbol{z})] \\ &= -\mathbb{E}_{x|\boldsymbol{z}}[\nabla^2_{\boldsymbol{z}} \log p(x|\boldsymbol{z})], \end{aligned} \tag{1}$$

where $\mathbb{D}_{x|\boldsymbol{z}}[.]$ denotes the variance under distribution $p(x|\boldsymbol{z})$.

When the FIM is adopted as a Riemannian metric tensor, it enables a connection between statistics and differential geometry. It is proved that the manifold composed of exponential family distributions is flat under the e-connection, and the manifold of mixture distributions is flat under the m-connection (Amari and Nagaoka 2007). The significance is that the metric only depends on the distribution of the model output, i.e., the FIM is invariant to model reparameterization, as long as the distribution is not changed. For example, (Amari 1999) shows the steepest direction in the statistical manifold is given by the natural gradient, which is invariant to reparameterization and saturation-free.

Adversarial attacks Many methods are proposed to generate adversarial examples. The fast gradient method (FGM) and the one-step target class method (OTCM) are two basic methods that simply adopt the gradient w.r.t. the input as the adversarial perturbation (Kurakin, Goodfellow, and Bengio 2016b). The basic iterative method (BIM) performs an iterative FGM update for the input samples with less modifications (Kurakin, Goodfellow, and Bengio 2016a), which is a more powerful generalization of the ones-step attacks. Several attack strategies, including the optimization based attack (Liu et al. 2016) and the C&W attack (Carlini and Wagner 2017c), are proposed to craft the adversarial examples via optimization. The adversarial examples of C&W attack are proved to be highly transferable between different models, and can almost completely defeat the defensive distillation mechanism (Papernot et al. 2015).

Adversarial defenses The defense against the adversarial examples can be generally divided into the following categories. The adversarial training takes the adversarial examples as part of the training data, so as to regularize the models and enhance the robustness (Miyato et al. 2015; Sinha, Namkoong, and Duchi 2017). (Katz et al. 2017) proposes to verify the model robustness based on the satisfiability modulo theory. The adversarial detecting approaches add an auxiliary classifier to distinguish the adversarial examples (Metzen et al. 2017). Many detection measurements, including kernel density estimation, Bayesian uncertainty (Feinman et al. 2017), Jensen Shannon divergence (Meng and Chen 2017), local intrinsic dimensionality (Ma et al. 2018), have been introduced to detect the existence of adversarial attacks. Despite the success of the above defenses in detecting many attacks, (Carlini. and Wagner 2017a;

Carlini and Wagner 2017b) suggest these mechanisms can be bypassed with some modifications of the objective functions.

3 The adversarial attack under the Fisher information metric

3.1 Proposed algorithm

In this section, we formalize the optimization of the adversarial perturbations as a constrained quadratic form of the FIM. As mentioned in the previous section, for classification tasks, the output of the network can be considered as the likelihood of a discrete distribution. In information theory, a meaningful metric for different probability distributions is not linear. Therefore, we start by using the KL divergence to measure the variation of the likelihood distributions.

Consider a deep neural network with its likelihood distribution denoted as $p(y|\boldsymbol{x};\boldsymbol{\theta})$, where $\boldsymbol{x}$ is the input sample, and $\boldsymbol{\theta}$ is the model weights. Since the model weights are fixed after training, and $\boldsymbol{x}$ is the only changeable parameter when attacking, we omit the model parameters $\boldsymbol{\theta}$ in the conditional distribution, and regard $\boldsymbol{x}$ as the model parameter. What the attackers are likely to do is to find a subtle perturbation $\boldsymbol{\eta}$, such that the probability $p(y|\boldsymbol{x}+\boldsymbol{\eta})$ varies from the correct to the wrong output. Hence we adopt the KL divergence to measure the variation of the probability $p(y|\boldsymbol{x})$. The optimization objective can be formulated as follows:

$$\max_{\boldsymbol{\eta}} D_{KL}(p(y|\boldsymbol{x})||p(y|\boldsymbol{x}+\boldsymbol{\eta})) \qquad \text{s.t. } \|\boldsymbol{\eta}\|_2^2 = \epsilon, \quad (2)$$

where ϵ is a small parameter to limit the size of the perturbation under the Euclidean metric. Previous literature has shown that the adversarial examples generally exist in large and continuous regions (Goodfellow, Shlens, and Szegedy 2014). such that the models can always be fooled with small perturbation. Let us assume the perturbation $\|\boldsymbol{\eta}\|$ is sufficiently small, such that the log-likelihood $\log p(y|\boldsymbol{x}+\boldsymbol{\eta})$ can be decomposed using the second-order Taylor expansion. This yields a simple quadratic form of the FIM:

$$\begin{aligned} D_{KL}(p(y|\boldsymbol{x})\|p(y|\boldsymbol{x}+\boldsymbol{\eta})) &= \mathbb{E}_{y|\boldsymbol{x}}[\log \frac{p(y|\boldsymbol{x})}{p(y|\boldsymbol{x}+\boldsymbol{\eta})}] \\ &\approx \frac{1}{2}\boldsymbol{\eta}^T \boldsymbol{G}_{\boldsymbol{x}} \boldsymbol{\eta}, \end{aligned} \quad (3)$$

where $\boldsymbol{G}_{\boldsymbol{x}} = \mathbb{E}_{y|\boldsymbol{x}}[(\nabla_{\boldsymbol{x}} \log p(y|\boldsymbol{x}))(\nabla_{\boldsymbol{x}} \log p(y|\boldsymbol{x}))^T]$ is the Fisher information of $\boldsymbol{x}$. Note that the FIM here is not the same as that in (Miyato et al. 2015). Since the expectation is over the observed empirical distribution $p(y|\boldsymbol{x})$, let p_i be the probability of $p(y|\boldsymbol{x})$ when y takes the i-th class, and let $\mathcal{J}(y,\boldsymbol{x}) = -\log p(y|\boldsymbol{x})$ be the loss function of the network, the matrix can be explicitly calculated by

$$\boldsymbol{G}_{\boldsymbol{x}} = \sum_i p_i [\nabla_{\boldsymbol{x}} \mathcal{J}(y_i,\boldsymbol{x})][\nabla_{\boldsymbol{x}} \mathcal{J}(y_i,\boldsymbol{x})]^T. \quad (4)$$

Hence we have a variant form of the objective function, which is given by:

$$\max_{\boldsymbol{\eta}} \boldsymbol{\eta}^T \boldsymbol{G}_{\boldsymbol{x}} \boldsymbol{\eta} \quad \text{s.t. } \|\boldsymbol{\eta}\|_2^2 = \epsilon,\ \mathcal{J}(y,\boldsymbol{x}+\boldsymbol{\eta}) > \mathcal{J}(y,\boldsymbol{x}). \quad (5)$$

Setting the derivative of the Lagrangian w.r.t. $\boldsymbol{\eta}$ to 0 yields $\boldsymbol{G}_{\boldsymbol{x}}\boldsymbol{\eta} = \lambda\boldsymbol{\eta}$. In general, the optimization can be solved by applying eigen-decomposition for $\boldsymbol{G}_{\boldsymbol{x}}$, and assigning the eigenvector with the greatest eigenvalue to $\boldsymbol{\eta}$. Note that eigenvector gives a straight line, not a direction, i.e., multiplying $\boldsymbol{\eta}$ by -1 does not change the value of the quadratic form. Therefore, we add an additional constraint $\mathcal{J}(y,\boldsymbol{x}+\boldsymbol{\eta}) > \mathcal{J}(y,\boldsymbol{x})$ here, guaranteeing that the adversarial examples obtained by $\boldsymbol{x}+\boldsymbol{\eta}$ will always attain higher loss than the normal samples.

The significance of our method is as follows. If we consider $D_{KL}(p(y|\boldsymbol{x})||p(y|\boldsymbol{x}+\boldsymbol{\eta}))$ as a function of $\boldsymbol{\eta}$, the Fisher information is exactly the Hessian matrix of the infinitesimal KL divergence. This implies that the vulnerability of deep learning models can be described by the principal curvature of KL divergence. Therefore, given an input sample $\boldsymbol{x}$, the eigenvalues of the FIM represent the robustness in the subspaces of corresponding eigenvectors. The larger the eigenvalues are, the more vulnerable the model is to be attacked by the adversarial perturbations in the subspaces of corresponding eigenvectors. This allows us to efficiently characterize the local robustness using the eigenvalues of the FIM.

3.2 Optimization strategies

As mentioned before, the simplest approach to solve the objective function (5) is to calculate the greatest eigenvector of $\boldsymbol{G}_{\boldsymbol{x}}$. However, such optimization can be impractical for large datasets. One main obstacle is that $\boldsymbol{G}_{\boldsymbol{x}}$ is computed explicitly. When the image size is large, the exact eigen-decomposition of $\boldsymbol{G}_{\boldsymbol{x}}$ becomes inefficient and memory consuming. In order to reduce the computational complexity, the critical part is to avoid the access to the explicit form of $\boldsymbol{G}_{\boldsymbol{x}}$. This can be achieved by computing $\boldsymbol{G}_{\boldsymbol{x}}\boldsymbol{\eta}$ alternatively. Let $\boldsymbol{g}_y = \nabla_{\boldsymbol{x}}\mathcal{J}(y,\boldsymbol{x})$ be the gradient of the class y loss w.r.t. the input $\boldsymbol{x}$. Since the FIM has the form $\boldsymbol{G}_{\boldsymbol{x}} = \mathbb{E}_{y|\boldsymbol{x}}[\boldsymbol{g}_y \boldsymbol{g}_y^T]$, by putting $\boldsymbol{\eta}$ into the expectation we obtain

$$\boldsymbol{G}_{\boldsymbol{x}}\boldsymbol{\eta} = \mathbb{E}_{y|\boldsymbol{x}}[(\boldsymbol{g}_y^T\boldsymbol{\eta})\boldsymbol{g}_y]. \quad (6)$$

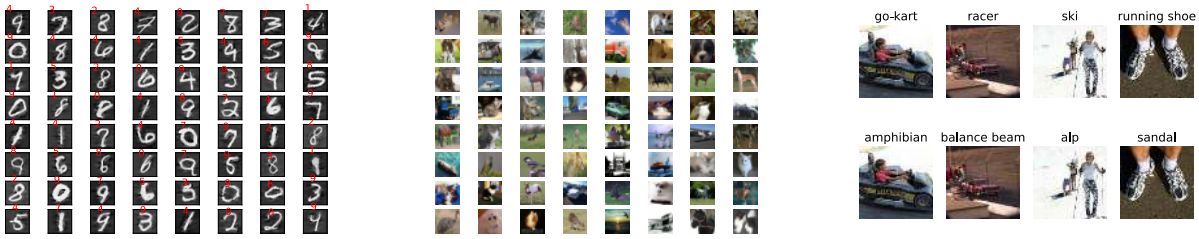
This allows us to calculate the inner product first, so as to avoid dealing with $\boldsymbol{G}_{\boldsymbol{x}}$ explicitly. After $\boldsymbol{\eta}$ converges, the greatest eigenvalue has the form $\mathbb{E}_{y|\boldsymbol{x}}[(\boldsymbol{g}_y^T\boldsymbol{\eta})^2]$.

Specifically, when computing the greatest eigenvector of $\boldsymbol{G}_{\boldsymbol{x}}$, a naive approach with the power iteration can be adopted to accelerate the eigen-decomposition. In Step k, the power iteration is described by the recurrence equation $\boldsymbol{\eta}_{k+1} = \frac{\boldsymbol{G}_{\boldsymbol{x}}\boldsymbol{\eta}_k}{\|\boldsymbol{G}_{\boldsymbol{x}}\boldsymbol{\eta}_k\|}$. The iteration thus becomes

$$\boldsymbol{\eta}_{k+1} = \frac{\mathbb{E}_{y|\boldsymbol{x}}[(\boldsymbol{g}_y^T\boldsymbol{\eta}_k)\boldsymbol{g}_y]}{\|\mathbb{E}_{y|\boldsymbol{x}}[(\boldsymbol{g}_y^T\boldsymbol{\eta}_k)\boldsymbol{g}_y]\|}. \quad (7)$$

Similar approach can be adopted when computing the top m eigenvalues and eigenvectors. The Lanczos algorithm, which also does not require the direct access to $\boldsymbol{G}_{\boldsymbol{x}}$, is an efficient eigen-decomposition algorithm for Hermitian matrices (Calvetti, Reichel, and Sorensen 1994). The algorithm is particularly fast for sparse matrices. Since $\boldsymbol{G}_{\boldsymbol{x}}$ is the pullback of a lower dimensional probability space, this guarantees the efficiency of our implementation.

Additionally, the expectation term in the exact computation of $\boldsymbol{G}_{\boldsymbol{x}}$ requires to sum over the support of $p(y|\boldsymbol{x})$, which



(a) MNIST (b) CIFAR-10 (c) ILSVRC-2012

Figure 1: Visualization for the adversarial examples crafted with our method (Best viewed with zoom-in). All the adversarial examples are obtained via one-step update for the original images. (a) The model prediction is marked in red numbers. (b) All the images here can successfully fool a 14-layer network trained on CIFAR-10. (c) The top row shows the original samples, while the second row is the adversarial examples. The model prediction is labeled in the top of the images.

Table 1: The comparison for the computation time of OSSA using different numerical methods

time (seconds)	Eigen-decomposition	Lanczos	Alias+Lanczos	Power iteration	Alias+Power iteration
CIFAR-10	1.49 ± 0.12	0.30 ± 0.02	0.15 ± 0.04	0.28 ± 0.03	0.25 ± 0.01
ILSVRC-2012	intractable	58.63 ± 3.50	7.23 ± 0.12	47.79 ± 3.15	3.15 ± 0.62

is still inefficient for the datasets with large number of categories. In practice, the estimation of the integral can be simplified by the Monte Carlo sampling from $p(y|\boldsymbol{x})$ with less iterations. The sampling iterations are set to be approximately $1/5$ number of the classes. Despite the simplicity, we empirically find the effectiveness is not degraded by the Monte Carlo approximation. The randomized sampling is performed using the alias method (Marsaglia, Tsang, and Wang 2004), which can efficiently sample from high dimensional discrete distribution with $O(1)$ time complexity.

Algorithm 1: One Step Spectral Attack (OSSA) Implemented with power iteration+alias sampling

Input: input sample $\boldsymbol{x}$, corresponding labels y, a deep learning model with the output $p(y|\boldsymbol{x})$ and the loss $\mathcal{J}(y, \boldsymbol{x})$.

Output: the perturbation $\boldsymbol{\eta}$, the greatest eigenvalue λ^*.

Initialize $\boldsymbol{\eta}$ as an random vector with unit norm;
Initialize the alias table with $p(y|\boldsymbol{x})$;
while $\boldsymbol{\eta}$ *not converged* **do**
  Update $\boldsymbol{\eta} \leftarrow \mathbb{E}_{y|\boldsymbol{x}}[(\boldsymbol{g}_y^T \boldsymbol{\eta})\boldsymbol{g}_y]$ using alias sampling;
  Normalize $\boldsymbol{\eta} \leftarrow \frac{\boldsymbol{\eta}}{\|\boldsymbol{\eta}\|_2}$;
end
The greatest eigenvalue $\lambda^* \leftarrow \mathbb{E}_{y|\boldsymbol{x}}[(\boldsymbol{g}_y^T \boldsymbol{\eta})^2]$;
if $\mathcal{J}(\boldsymbol{x} + \boldsymbol{\eta}) \leq \mathcal{J}(\boldsymbol{x})$ **then**
  $\boldsymbol{\eta} \leftarrow -\boldsymbol{\eta}$;
end

In our experiments, we only use the randomization trick for ILSVRC-2012. Table 1 shows the comparison for the time consumption of the aforementioned methods. To summarize, the algorithm procedure of the alias method+power iteration implementation is shown in Algorithm 1. In Figure 1, we also illustrate some visualization of the adversarial examples crafted with our method.

3.3 Geometrical interpretation

Characterizing the vulnerability of neural networks is an important question for studying adversarial examples. Under the Euclidean metric, (Sinha, Namkoong, and Duchi 2017) has suggested that identifying the worst case perturbation in ReLU networks is NP-hard. In this subsection, we give an explanation for the vulnerability of deep learning from a different aspect. Our aim is to prove that under the Fisher information metric, the perturbation obtained by our algorithm will not be "compressed" through the mapping of networks, which has contributed to the vulnerability of deep learning.

Geometrically, let $\boldsymbol{z} = f(\boldsymbol{x})$ be the mapping through the neural network, where $\boldsymbol{z} \in (0, 1)^k$ is the continuous output vector of the softmax layer, with $p(y|\boldsymbol{z}) = \prod_i \boldsymbol{z}_i^{y_i}$ being a discrete distribution. We can conclude the FIM $\boldsymbol{G}_{\boldsymbol{z}} = \mathbb{E}_{y|\boldsymbol{z}}[(\nabla_{\boldsymbol{z}} \log p(y|\boldsymbol{z}))(\nabla_{\boldsymbol{z}} \log p(y|\boldsymbol{z}))^T]$ is a non-singular diagonal matrix. The aforementioned Fisher information $\boldsymbol{G}_{\boldsymbol{x}}$ is thus interpreted as a Riemannian metric tensor induced from $\boldsymbol{G}_{\boldsymbol{z}}$. The corresponding relationship is

$$\boldsymbol{\eta}^T \boldsymbol{G}_{\boldsymbol{x}} \boldsymbol{\eta} = \boldsymbol{\eta}^T \boldsymbol{J}_f^T \boldsymbol{G}_{\boldsymbol{z}} \boldsymbol{J}_f \boldsymbol{\eta}, \tag{8}$$

where $\boldsymbol{J}_f$ is the Jacobian of $\boldsymbol{z}$ w.r.t. $\boldsymbol{x}$. Note that for most neural networks, the dimensionality of $\boldsymbol{z}$ is much less than that of $\boldsymbol{x}$. making $\boldsymbol{J}_f \boldsymbol{\eta}$ a mapping for $\boldsymbol{\eta}$ from high dimensional data space to low dimensional probability space. This means f is a surjective mapping and $\boldsymbol{G}_{\boldsymbol{x}}$ is a degenerative metric tensor. Therefore, the geodesic distance in the probability space is always no larger than the corresponding distance in the data space. Using the inequality, we can define the concept of optimal adversarial perturbation, formulated as follows.

Definition 1. *Let $\mathcal{N}$ and $\mathcal{M}$ be two Riemannian manifolds with the FIMs $\boldsymbol{G}_{\boldsymbol{z}}$ and $\boldsymbol{G}_{\boldsymbol{x}}$ being their metric tensor respec-*

tively. Let $f : \mathcal{M} \to \mathcal{N}$ be the mapping of the neural network. For $\boldsymbol{x} \in \mathcal{M}$, an adversarial perturbation $\boldsymbol{\eta} \in T_{\boldsymbol{x}}\mathcal{M}$ is **optimal** *if $f(\boldsymbol{x})$ is an isometry for the geodesic determined by the exponential mapping $\mathrm{Exp}_{\boldsymbol{x}}(\boldsymbol{\eta}) : T_{\boldsymbol{x}}\mathcal{M} \to \mathcal{M}$.*

Definition 2. *Let $f : \mathcal{M} \to \mathcal{N}$ be a smooth mapping, and $f_* : T_{\boldsymbol{x}}\mathcal{M} \to T_{f(\boldsymbol{x})}\mathcal{N}$ be the derivative of f. The mapping f is a* **submersion** *if f is surjective and f_* is surjective, and a* **Riemannian submersion** *if f_* is an isometry on the horizontal bundle $H = ker\ f_*^{\perp}$.*

The definitions show the optimal perturbations span only on the horizontal bundles. Thus the conclusion is as follows.

Theorem 1. *Let $\boldsymbol{J}_f$ be the Jacobian field of $f(\boldsymbol{x})$ w.r.t. $\boldsymbol{x}$. If $\boldsymbol{J}_f\boldsymbol{J}_f^T$ is non-singular, and $f : \mathcal{M} \to \mathcal{N}$ is a smooth mapping, then a sufficiently small perturbation $\boldsymbol{\eta} \in T_{\boldsymbol{x}}\mathcal{M}$ obtained by Algorithm 1 is optimal.*

Proof. For the neural network f, we define $V_{\boldsymbol{x}} \subset T_{\boldsymbol{x}}\mathcal{M}$, the vertical subspace at a point $\boldsymbol{x} \in \mathcal{M}$, as the kernel of the FIM $\boldsymbol{G}_{\boldsymbol{x}}$. In our algorithm, we always apply the greatest eigenvector in the FIM $\boldsymbol{G}_{\boldsymbol{x}}$ as the adversarial perturbation. Given a smooth network f, because $\boldsymbol{J}_f\boldsymbol{J}_f^T$ is always non-singular, the first eigenvalue in the FIM is always larger than zero, which corresponds to the non-degenerative direction. Therefore the adversarial perturbation $\boldsymbol{\eta} \in T_{\boldsymbol{x}}\mathcal{M}$ obtained is always in the horizontal bundle, i.e., $\boldsymbol{\eta} \in H$. By definition, $f_* : T_{\boldsymbol{x}}\mathcal{M} \to T_{f(\boldsymbol{x})}\mathcal{N}$ will be an isometry for the horizontal sub-bundles. Then f will also be an isometry for the geodesic determined by $\mathrm{Exp}_{\boldsymbol{x}}(\boldsymbol{\eta}) : T_{\boldsymbol{x}}\mathcal{M} \to \mathcal{M}$. □

In a broader sense, the theorem confirms the validity of our proposed approach, and serves as a basis for characterizing the vulnerability of deep learning models. Note that the theorem is concluded without any assumption for the network structures. The optimality can thus be interpreted as a generalization of the excessive linearity explanation (Goodfellow, Shlens, and Szegedy 2014). The statement shows that the linearity may not be a sufficient condition for the vulnerability of neural networks. Using our algorithm, similar phenomenon can be reproduced in a network with smooth activations, e.g. the exponential linear unit (Clevert, Unterthiner, and Hochreiter 2015).

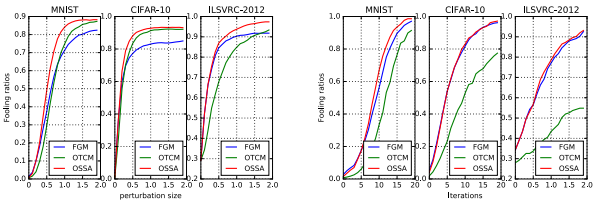
3.4 Experimental evaluation

In this section, by presenting experimental evaluations for the properties of the adversarial attacks, we show the ability of our attack method to fool deep learning models, and characterize the adversarial subspaces. The experiments are performed on three standard benchmark datasets MNIST, CIFAR-10 (Krizhevsky and Hinton 2009), and ILSVRC-2012 (Russakovsky et al. 2015). The pixel values in the images are constrained in the interval [0.0, 1.0]. We adopt three different networks for the three datasets respectively: LeNet-5, VGG, and ResNet-152 (He et al. 2015). The VGG network adopted here is a shallow variant of the VGG-16 network (Simonyan and Zisserman 2014), where the layers from conv4-1 to conv5-3 are removed to reduce redundancy. We use the pre-trained ResNet-152 model integrated in TensorFlow. In our experiments, all the adversarial perturbations are evaluated with ℓ_2 norm.

Table 2: The fooling rates and the mean of least ℓ_2 perturbation norms under two one-step attack strategies

	MNIST		CIFAR		ILSVRC	
Attacks	mean	rate %	mean	rate %	mean	rate %
FGM	2.11	94.98	1.11	95.21	0.48	100
OSSA	1.80	95.68	1.06	97.85	0.47	100

White-box attack In the first experiment, we perform comparisons for the ability of our method to fool the deep learning models. The comparison is made between two one-step attack methods, namely FGM and OTCM, and their iterative variants. The target class in OTCM is randomly chosen from the set of incorrect classes. Similar to the relationship between FGM and BIM, by computing the first eigenvector of FIM in each step, it is a natural idea to perform our attack strategy iteratively. For the iterative attack, we set the perturbation size $\epsilon = 0.05$. We only use the samples in the test set (validation set for ILSVRC-2012) to craft the adversarial examples.



(a) One-step attack (b) Iterative attack

Figure 2: (a) The models' misclassification rates increase with the perturbation size. (b) The models' misclassification rates increase with the number of iterations. When performing the iterative attacks, we set perturbation size to 0.05, 0.025, 0.0125 for the three datasets respectively.

The results are illustrated in Figure 2. Observe that our proposed method quickly attains a high fooling rate with smaller values of $\|\boldsymbol{\eta}\|$. For one-step attacks, our method achieves 90% fooling ratio with the ℓ_2 perturbation norms 2.1, 1.4, and 0.7 on MNIST, CIFAR-10, and ILSVRC-2012 respectively. This implies that the eigenvectors as adversarial perturbations is a better characterization for model robustness than the gradients. Another evidence is shown in Table 2, where we conduct binary search to find the mean of the least perturbation norm on three datasets. Our approach achieves higher fooling ratios than the gradient-based method, with a smaller mean of the least perturbation norm. Both of the results are consistent with our conclusion in previous sections.

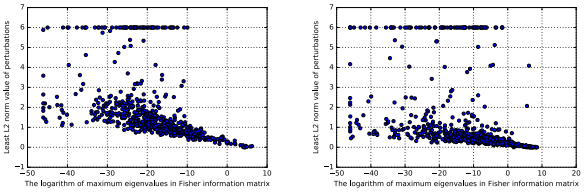
Black-box attack In the real world, the black-box attack is more common than the white-box attacks. It is thus important to analyze the transferability between different models. In this experiment, we show the ability of our attack approach to transfer across different models, particularly the models regularized with adversarial training. The experiment is performed on MNIST, with four different networks: LeNet,

Table 3: The cross-model fooling ratios on MNIST using OSSA.

Fooling rates	Crafted from			
Tested on	LeNet	VGG	LeNet-adv	VGG-adv
LeNet	100.0	62.02	88.49	82.20
VGG	53.64	100.0	76.93	74.45
LeNet-adv	27.92	17.83	100.0	90.04
VGG-adv	15.06	29.24	94.15	100.0

VGG, and their adversarial training variants, which is referred to as LeNet-adv and VGG-adv here. For the two variant networks, we replace all the ReLU activations with ELUs, and train the network with adversarial training using FGM. All of the above networks achieve more than 99% accuracy on the test set of MNIST. To make the comparison fair, we set $\epsilon = 2.0$ for all the tested attack methods.

The results of this experiment are shown in Table 3. The cross-model fooling ratios are obviously asymmetric between different models. Specifically, the adversarial training plays an important role for defending against the attack. The models without adversarial training produce 22.51% error rate in average on the the models with adversarial training, while the reversed case is 80.52% in average. Surprisingly, the adversarial examples crafted from the models with adversarial training yield high fooling ratios on the two normal networks. Whereas a heuristic interpretation is that the perturbations obtained by OSSA correspond to only one subspace, making the adversarial training less specific for our attack strategy, the reason of the phenomenon requires further investigation.



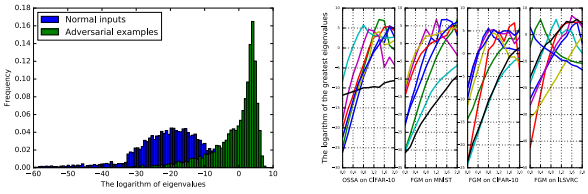
(a) MNIST (b) CIFAR-10

Figure 3: Using 800 random samples, the scatter illustrates the relationship between the least ℓ_2 perturbation norm and the maximum eigenvalues. The least ℓ_2 perturbation size is obtained via binary search in the interval [0.0, 6.0]. The horizontal axis is shown in logarithm.

Characterizing multiple adversarial subspaces As shown in the previous sections, the eigenvalues in FIM can be applied to measure the local robustness. We thus perform experiments to verify the correlation between the model robustness and the eigenvalues. In Figure 3, we show the scatter of 800 randomly selected samples in the validation set of MNIST and CIFAR-10. The horizontal axis is the logarithm of the eigenvalues, and the vertical axis is the least adversarial perturbation size, i.e., the least value of $\|\boldsymbol{\eta}\|_2$ to fool the network. The value is obtained via binary search between the interval $[0.0, 6.0]$. Most adversarial examples can successfully fool the model in this range. The result shows an obvious correlation between the eigenvalues and the model vulnerability: the least perturbations linearly decrease with the exponential increasing of eigenvalues.

A reasonable interpretation is that the eigenvalues reflect the size of the perturbations under the Fisher information metric. According to our optimality analysis, large eigenvalues can result in isometrical variation for the output likelihood, which is more likely to fool the model with less perturbation size. This property is crucial for our following discussion, the adversarial detection, where we take advantage of the distinguishability of the eigenvalues to detect the adversarial attacks.

4 The adversarial detection under the Fisher information metric



(a) Eigenvalues' distributions (b) Increment of eigenvalues

Figure 4: Some empirical evidence for the distinguishability of the eigenvalues. (a) The histograms for the distribution of largest eigenvalues. The statistic is performed on all the samples in the test set of MNIST. (b) The increment of the eigenvalues along the direction of the adversarial perturbations. The samples are randomly sampled from MNIST, CIFAR-10, and ILSVRC-2012.

As shown in the previous sections, given an input sample $\boldsymbol{x}$, the eigenvalues in FIM can well describe its local robustness. In this section, we show how the eigenvalues in FIM can serve as features to detect the adversarial attacks. Specifically, the detection is achieved by training an auxiliary classifier to recognize the adversarial examples, with the eigenvalues serving as the features for the detector. Motivated by (Fawzi, D. Moosavi, and Frossard 2016), besides the normal original samples and the adversarial inputs, we also craft some noisy samples to augment the detection. Since the networks are supposed to be robust to some random noise applied to the input, the set of negative samples should contain both the normal samples and noisy samples, while the set of positive samples contain the adversarial examples.

In the left of Figure 4, we show a histogram of the eigenvalues distribution. We adopt the FGM to generate adversarial examples for the samples from MNIST, and evaluate their greatest eigenvalues in FIM. The histogram shows that the distributions of the eigenvalues for normal samples and adversarial examples are different in magnitude. The eigenvalues of the latter are densely distributed in larger domain, while the distribution of the former is approximately an Gaussian distribution with smaller mean. Although there is overlapping part for the supports of the two distributions, the separabil-

Table 4: The AUC scores of detecting adversarial attacks using random forest. The best are marked with **bold font.**

	MNIST					CIFAR-10				
AUC (%)	FGM	OTCM	Opt	BIM	OSSA	FGM	OTCM	Opt	BIM	OSSA
KD	78.12	95.46	95.15	98.61	84.24	64.92	92.13	91.35	98.70	88.89
BU	32.37	91.55	71.30	25.46	74.21	70.40	91.93	91.39	97.32	87.44
KD+BU	82.43	95.78	95.35	98.81	85.97	76.40	**94.45**	93.77	98.90	93.54
Ours	**96.11**	**98.47**	**95.67**	**99.10**	**93.13**	**80.18**	93.68	**99.45**	**99.43**	**98.01**

ity for the adversarial examples can be largely enhanced by adding more eigenvalues as features. In the right of Figure 4, using our proposed OSSA, we illustrate some examples of the eigenvalues increasing along the direction of the adversarial perturbations. As we predicted, the eigenvalues increase with the increasing of the perturbation size, showing that the adversarial examples have higher eigenvalues in FIM compared with the normal samples.

The next question is which machine learning classifier should be adopted for the detection. In out experiments, we empirically find the models are more likely to attain high variance instead of high bias. The naive Bayes classifier with Gaussian likelihood, and the random forest classifier yields the best performance among various models. The success of the former demonstrates that the geometry structure in each subspace is relatively independent. As for the random forest classifier, we empirically find that varying the parameters (e.g. the tree depth, the value of ϵ, etc.) does not significantly affect the AUC scores. We also find the tree depth not to exceed 5, and more than 20 trees in the random forest yields good performance. These results imply that our detection with Fisher information enjoys low variance.

In Table 4, we adopt the AUC score to evaluate the performance of our random forest classifier under different attacks. The comparison is made between our approach and two characteristics described in (Feinman et al. 2017), namely the kernel density estimation (KD) and the Bayesian uncertainty (BU). In our experiments, only the top 20 eigenvalues are extracted as the features for classification. Observe that the detector achieves desirable performance in recognizing the adversarial examples. The eigenvalues as features outperform KD and BU on both datasets. In addition, our detector is particularly good at recognizing OSSA adversarial examples. The AUC scores are 7.16% and 4.47% higher than the combination of the other two characteristics.

In the real world, we cannot presume all the attacks strategies are known before we train the detector. It is thus important for the features to have sufficient generalization ability. In Table 5, we show the AUC scores of the detector trained on only one type of adversarial examples. Observe that most of our results exceed 90% of AUC scores, indicating the adversarial examples generated by various methods share similar geometric properties under the Fisher information metric. Interestingly, the detector trained on OSSA obtain the worst generalization ability among all methods. We regard this is due to the geometrical optimality of our method. According to our analysis in the previous section, the adversarial examples of OSSA may distribute densely in more limited subspaces, resulting in less diversity for generalization.

Table 5: The generalization ability for detecting adversarial attacks on MNIST with random forest classifier

AUC (%)	Tested on				
Trained on	FGM	OTCM	Opt	BIM	OSSA
FGM	94.31	91.92	90.78	91.87	92.13
OTCM	98.55	98.96	98.26	97.78	98.57
Opt	95.18	95.30	96.90	97.15	96.11
BIM	98.10	96.00	97.09	98.57	96.35
OSSA	91.17	91.47	89.77	89.47	89.67

5 Conclusion

In this paper, we have studied the adversarial attacks and detections using information geometry, and proposed a method unifying the adversarial attack and detection. For the attacks, we show that under the Fisher information metric, the optimal adversarial perturbation is the isometry between the input space and the output space, which can be obtained by solving a constrained quadratic form of the FIM. For the detection, we observe the eigenvalues of FIM can well describe the local vulnerability of a model. This property allows us to build machine learning classifiers to detect the adversarial attacks with the eigenvalues. Experimental results have shown promising robustness on the adversarial detection.

Addressing the adversarial attacks issue is typically difficult. One of the great challenges is the lack of theoretical tools to describe and analyze the deep learning models. We are confident that the Riemannian geometry is a promising approach to leverage better understanding for the vulnerability of deep learning.

In this paper, we only focus on the classification tasks, where the likelihood of the model is a discrete distribution. Besides classification, there are many other tasks which can be formulated as statistical problems, e.g. Gaussian distribution for regression tasks. Therefore, investigating the adversarial attacks and defenses on other tasks will be an interesting future direction.

Acknowledgement

This work is supported by the National Science Foundation of China (Nos. 11771276, 11471208, 61731009 and 61273298), Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University, China, and the Science and Technology Commission of Shanghai Municipality (No. 14DZ2260800).

References

Amari, S., and Nagaoka, H. 2007. *Methods of Information Geometry*. Providence, RI: American Mathematical Society.

Amari, S. 1999. Natural gradient works efficiently in learning. *Neural Computation* 10(2):251–276.

Calvetti, D.; Reichel, L.; and Sorensen, D. C. 1994. An implicit restarted Lanczos method for large symmetric eigenvalue problems. *Electronic Transactions on Numerical Analysis* 2:1–21.

Carlini., N., and Wagner, D. 2017a. Adversarial examples are not easily detected: Bypassing ten detection methods. *ArXiv preprint arXiv: 1705.07263*.

Carlini, N., and Wagner, D. 2017b. Magnet and "Efficient defenses against adversarial attacks" are not robust to adversarial examples. *ArXiv preprint arXiv: 1711.08478*.

Carlini, N., and Wagner, D. A. 2017c. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, 39–57.

Clevert, D.; Unterthiner, T.; and Hochreiter, S. 2015. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*.

Fawzi, A.; D. Moosavi, M. S.; and Frossard, P. 2016. Robustness of classifiers: From adversarial to random noise. In *Advances in Neural Information Processing Systems*. 1632–1640.

Feinman, R.; Curtin, R. R.; Shintre, S.; and Gardner, A. B. 2017. Detecting adversarial samples from artifacts. *ArXiv preprints arXiv:1703.00410*.

Goodfellow, I.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *ArXiv preprints arXiv:1412.6572*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*.

Katz, G.; Barrett, C.; Dill, D. L.; Julian, K.; and Kochenderfer, M. J. 2017. Reluplex: An efficient SMT solver for verifying deep neural networks. In *International Conference on Computer-Aided Verification*, 97–117.

Krizhevsky, A., and Hinton, G. 2009. Learning multiple layers of features from tiny images. Technical report, University of Toronto.

Krotov, D., and Hopfield, J. J. 2017. Dense associative memory is robust to adversarial inputs. *arXiv preprint arXiv:1701.00939*.

Kurakin, A.; Goodfellow, I.; and Bengio, S. 2016a. Adversarial examples in the physical world. *ArXiv preprints arXiv:1607.02533*.

Kurakin, A.; Goodfellow, I.; and Bengio, S. 2016b. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*.

Liu, Y.; Chen, X.; Liu, C.; and Song, D. 2016. Delving into transferable adversarial examples and black-box attacks. *ArXiv preprints arXiv:1611.02770*.

Ma, X.; Li, B.; Wang, Y.; Erfani, S. M.; Wijewickrema, S.; Schoenebeck, G.; Song, D.; Houle, M. E.; and Bailey, J. 2018. Characterizing adversarial subspaces using local intrinsic dimensionality. *ArXiv preprints arXiv:1801.02613*.

Marsaglia, G.; Tsang, W. W.; and Wang, J. 2004. Fast generation of discrete random variables. *Journal of Statistical Software* 11(3):17–24.

Meng, D., and Chen, H. 2017. Magnet: A two-pronged defense against adversarial examples. *ArXiv preprints arXiv:1705.09064*.

Metzen, J.; Genewein, T.; Fischer, V.; and Bischoff, B. 2017. On detecting adversarial perturbations. *ArXiv preprints arXiv:1702.04267*.

Miyato, T.; Maeda, S.-i.; Koyama, M.; Nakae, K.; and Ishii, S. 2015. Distributional smoothing with virtual adversarial training. *ArXiv preprints arXiv:1507.00677*.

Moosavidezfooli, S. M.; Fawzi, A.; Fawzi, O.; Frossard, P.; and Soatto, S. 2017a. Analysis of universal adversarial perturbations. *arXiv preprint arXiv:1705.09554*.

Moosavidezfooli, S. M.; Fawzi, A.; Fawzi, O.; and Frossard, P. 2017b. Universal adversarial perturbations. In *IEEE Conference on Computer Vision and Pattern Recognition*, 86–94.

Moosavidezfooli, S. M.; Fawzi, A.; and Frossard, P. 2016. Deepfool: A simple and accurate method to fool deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2574–2582.

Papernot, N.; McDaniel, P.; Wu, X.; Jha, S.; and Swami, A. 2015. Distillation as a defense to adversarial perturbations against deep neural networks. *ArXiv preprints arXiv:1511.04508*.

Papernot, N.; McDaniel, P.; and Goodfellow, I. 2016. Transferability in machine learning: From phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A. C.; and Li, F. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115(3):211–252.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sinha, A.; Namkoong, H.; and Duchi, J. 2017. Certifying some distributional robustness with principled adversarial training. *ArXiv preprints arXiv:1710.10571*.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *ArXiv preprints arXiv:1312.6199*.

Tabacof, P., and Valle, E. 2016. Exploring the space of adversarial images. In *International Joint Conference on Neural Networks*, 426–433.

Tanay, T., and Griffin, L. 2016. A boundary tilting persepective on the phenomenon of adversarial examples. *ArXiv preprints arXiv:1608.07690*.

Tramèr, F.; Papernot, N.; Goodfellow, I.; Boneh, D.; and McDaniel, P. 2017. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*.