Proceedings of Machine Learning Research vol 107:896–920, 2021 2nd Annual Conference on Mathematical and Scientific Machine Learning

Robust Certification for Laplace Learning on Geometric Graphs

Matthew Thorpe MATTHEW.THORPE-2@MANCHESTER.AC.UK
Department of Mathematics
University of Manchester, Manchester, UK, M13 9PR

Bao Wang WANGBAONJ@GMAIL.COM
Department of Mathematics
Scientific Computing and Imaging Institute
University of Utah, Salt Lake City, UT, 84112

Editors: Joan Bruna, Jan S Hesthaven, Lenka Zdeborova

Abstract

Graph Laplacian (GL)-based semi-supervised learning is one of the most used approaches for classifying nodes in a graph. Understanding and certifying the adversarial robustness of machine learning (ML) algorithms has attracted large amounts of attention from different research communities due to its crucial importance in many security-critical applied domains. There is great interest in the theoretical certification of adversarial robustness for popular ML algorithms. In this paper, we provide the first adversarial robust certification for the GL classifier. More precisely we quantitatively bound the difference in the classification accuracy of the GL classifier before and after an adversarial attack. Numerically, we validate our theoretical certification results and show that leveraging existing adversarial defenses for the k-nearest neighbor classifier can remarkably improve the robustness of the GL classifier.

Keywords: Graph Laplacian; Semi-supervised learning; Robust certification

1. Introduction

Let $\Omega_N := \{\boldsymbol{x}_i\}_{i=1}^N \subset \mathbb{R}^d$ be a set of feature vectors with a subset of $\Gamma_N := \{\boldsymbol{x}_i\}_{i \in Z_N \subset [N]}$ being labeled. If $i \in Z_N$ then $\boldsymbol{x}_i$ is labeled $\ell(\boldsymbol{x}_i) \in \mathbb{R}$ and we denote $\ell_N := \ell|_{\Gamma_N}$. The Graph Laplacian (GL) framework encodes the geometry of the feature vectors Ω_N by constructing an undirected graph, $G_N = (\Omega_N, \mathbf{W}_N)$, where Ω_N forms the nodes of the graph and $\mathbf{W}_N := (\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}})_{\boldsymbol{x},\boldsymbol{y} \in \Omega_N}$ is the set of edge weights with $\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}$ being the weight of the edge between $\boldsymbol{x}$ and $\boldsymbol{y}$. The graph Dirichlet energy is defined by

$$\mathcal{E}(u; \Omega_N) = \sum_{\boldsymbol{x},\boldsymbol{y} \in \Omega_N} \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} (u(\boldsymbol{x}) - u(\boldsymbol{y}))^2,$$

where u is a function defined on the nodes Ω_N of the graph. We can then predict the label for the unlabeled data by solving the following constrained energy minimization problem

$$\text{minimize } \mathcal{E}(u; \Omega_N) \text{ over } u : \Omega_N \to \mathbb{R} \text{ subject to } u(\boldsymbol{x}) = \ell_N(\boldsymbol{x}) \, \forall \boldsymbol{x} \in \Gamma_N. \tag{1}$$

Laplacian regression is the solution to (1). To go from regression to (binary) classification one thresholds u, e.g. if the classes are represented by $\{0, 1\}$ then the GL classifier predicts the label 1 if $u(\boldsymbol{x}) \geq 1/2$, and 0 otherwise. Note that the GL classifier classifies any unlabeled data leveraging

both labeled and unlabeled data. As a comparison, for any unlabeled $\boldsymbol{x}$, the k-nearest neighbor (kNN) classifier classifies $\boldsymbol{x}$ with the most common label amongst its labeled nearest neighbors.

The GL classifier has been successfully used for semi-supervised data classification (Wang et al., 2006; Zhou et al., 2004; Zhu et al., 2003), image processing (Buades et al., 2006; Gilboa and Osher, 2009; Shi et al., 2017), improving robustness and accuracy of deep neural nets (DNNs) (Wang et al., 2018a; Wang and Osher, 2019), etc. Direct application of GL classification with Gaussian (Belkin and Niyogi, 2004) or locally linear embedding weights (Roweis and Saul, 2000) for the above tasks may cause inference inconsistency in the low labeling ratio regime. To resolve this dilemma, many regularisation strategies have been developed to adapt GL to the ultra-low ratio of the labeled training data, e.g., scaling the weights (Shi et al., 2017, 2018) of the labeled data and the p-Laplacian (Calder, 2018; Rios et al., 2019; Zhou and Schölkopf, 2005).

Despite the tremendous success of machine learning (ML) algorithms, they are generally vulnerable to adversarial attacks (Szegedy et al., 2013). The adversarial vulnerability of ML algorithms raises concerns in applications to security-critical domains, such as: autonomous cars (Akhtar and Mian, 2018; Schneier, 2019), medical imaging (Finlayson et al., 2019), and national defense (Hoadley and Lucas, 2018). Many algorithms have been recently proposed to improve robustness of ML including adversarial training (Goodfellow et al., 2014; Madry et al., 2017), augmenting training data with unlabeled instances (Carmon et al., 2019), and noise injection (Wang et al., 2019).Nevertheless, there is a lack of theoretical understanding of adversarial issues of ML models. In this paper, we focus on *theoretical analysis of the conditions that guarantee adversarial robustness of the GL classifier for semi-supervised learning (SSL).*

1.1. Our Contribution

A classifier is said to be certifiably robust in classifying $\boldsymbol{x}$, if the classification result remains constant provided the perturbation on $\boldsymbol{x}$ is within a ball, e.g., ℓ_2-ball, of radius r. In this paper, we provide the first certification of the adversarial robustness of the GL classifier under the ℓ_2-norm. Our theory shows that within a certain adversarial attack regime, the GL classifier with $O(k)$ edges per node is intrinsically more robust than the kNN classifier. We show that to achieve certified robustness, the GL method needs significantly fewer nearest neighbors, with a small computational overhead. Our theoretical result resonates with the finding that unlabeled data can improve the robustness of ML algorithms (Carmon et al., 2019) and provides a feasible avenue to explain the observation that GL-based activation function remarkably improves DNNs' robustness (Wang and Osher, 2019). We summarize these high probability results in Table 1, where N and M are the total number of data and the number of unlabeled data respectively, k is the number of nearest neighbors involved in kNN and the approximate order of edges per node for the GL classifier, r is the maximum allowed adversarial perturbation measured in the ℓ_2-norm, and κ is the condition number of the matrix $\mathbf{W}_N$. We point out, however, that the results for the GL classifier in Table 1 are a special case and in particular one can reduce the number of neighbors k at the cost of reducing the probability (going from high probability bounds to low probability bounds). Note that if a constant fraction of the data is labeled i.e., $(N-M)/N$ is constant, then $k = \Omega(\log N)$ for the GL classifier. We will numerically verify these theoretical results with the existing benchmark experiments in Section 4. More detail on how we extracted these bounds from our theoretical results is given in Remark 3.

Table 1: High probability robustness guarantees and computational complexity of GL vs. kNN.

Classifier	k	Assumption on r	Computational Complexity	Reference
kNN	$\boldsymbol{\Omega(\sqrt{N \log N})}$	None	$O(kM \log(N-M))$	Wang et al. (2018b)
GL	$\boldsymbol{\Omega\left(\frac{N \log N}{N-M}\right)}$	$r \leq c\sqrt{\frac{N-M}{N}}\left(\frac{\log N}{N-M}\right)^{\frac{1}{d}}$	$O(kN \log N + Nk\sqrt{\kappa})$	**This Work**

1.2. Additional Related Works

The first theoretical characterisation of the number of nearest neighbors required for a robust kNN classifier appeared in (Wang et al., 2018b), where the authors also proposed a robust one nearest neighbor approach. We apply the robust characterisation used by Wang et al. (2018b) and develop a robust certification for the GL classifier in SSL.

To prove robustness, we connect with large data results and we mention several here. When the labeling rate is low Laplacian regularisation becomes degenerate and the label function u becomes nearly constant with sharp spikes at the labeled points (El Alaoui et al., 2016; Nadler et al., 2009; Slepčev and Thorpe, 2019). The degeneracy can be avoided by either using p-Dirichlet energies, with $p > d$ (Calder, 2019; El Alaoui et al., 2016; Slepčev and Thorpe, 2019), by increasing the label rate $(N-M)/N$ (Calder et al., 2020), or by reweighting the Laplacian in order to gain more regularity (Calder and Slepčev, 2019; Shi et al., 2018). Similar results hold for the game theoretic p-Laplacian (Calder, 2018, 2019). In addition, pointwise convergence of Laplacians has been considered several times, for example (Belkin and Niyogi, 2007; Calder, 2018; Calder et al., 2020; García Trillos et al., 2019; García Trillos and Slepčev, 2018; Hein et al., 2005; Singer, 2006).

1.3. Organization

We organize this paper as follows: In Section 2, we present the main theory on the certified robustness of the GL classifier. In Section 3, we analyze the computational complexity of the GL classifier. We verify the robustness of the GL classifier in different settings and compare it with the kNN classifier in Section 4. This paper ends with some concluding remarks in Section 5. Technical proofs and some more experimental details and results are provided in the appendix.

1.4. Notation

We denote vectors/matrices by lower/upper case bold face letters. Given two sequences $\{a_n\}$ and $\{b_n\}$, we write $a_n = O(b_n)$ if there exists a positive constant C such that $a_n \leq Cb_n$; and $a_n = \Omega(b_n)$ if for large enough n, b_n is at least ca_n for some constant c. Throughout $0 < c \leq C < +\infty$ will be arbitrary constants (independent of data realisations and all other parameters but possibly depending on dimension and the density of the data generating distribution) and may change value from line-to-line. We denote the set $\{1, 2, \cdots, N\}$ by $[N]$.

2. Main Theory

2.1. Preliminaries and Assumptions

To certify the robustness of the GL classifier, we make the following assumptions on the dataset:

(A1) $\Omega \subset \mathbb{R}^d$ is open connected and bounded with Lipschitz boundary;

(A2) $\boldsymbol{x}_i \overset{\text{iid}}{\sim} \mu \in \mathcal{P}(\Omega)$ where μ has density $\rho \in \mathrm{C}^2(\Omega)$ that is bounded below by a positive constant, i.e. $\inf_{\boldsymbol{x}\in\Omega} \rho(\boldsymbol{x}) =: \rho_{\min} > 0$;

(A3) $\mathbb{P}(\boldsymbol{x} \in \Gamma_N | \boldsymbol{x} \in \Omega_N) = \mathbb{P}(i \in Z_N | \boldsymbol{x}_i \in \Omega_n) = \beta$ and if $\boldsymbol{x} \in \Gamma_N$, then $\boldsymbol{x}$ is labeled as $\ell(\boldsymbol{x})$ for a Lipschitz function $\ell : \Omega \to \mathbb{R}$.

It will be convenient to define $\ell_N = \ell|_{\Gamma_N}$. Note that β is the probability of a data point being labeled and so (in the notation of Table 1) we can make the formal association $\beta \sim (N - M)/N$. For convenience, we introduce the following constrained graph Dirichlet energy functional:

$$\mathcal{E}_{\text{con}}(u; D_N) = \begin{cases} \mathcal{E}(u; \Omega_N) & \text{if } u(\boldsymbol{x}) = \ell_N(\boldsymbol{x}) \, \forall \boldsymbol{x} \in \Gamma_N \\ +\infty & \text{else.} \end{cases}$$

The Euler-Lagrange equation corresponding to minimizing $\mathcal{E}_{\text{con}}(\cdot; D_N)$ is

$$\begin{aligned} \mathcal{L}_N(u; \Omega_N)(\boldsymbol{x}) &= 0 && \text{for } \boldsymbol{x} \in \Omega_N \setminus \Gamma_N \\ u(\boldsymbol{x}) &= \ell_N(\boldsymbol{x}) && \text{for } \boldsymbol{x} \in \Gamma_N, \end{aligned}$$

where $\mathcal{L}_N(\cdot; \Omega_N)$ is the graph Laplacian defined by

$$\mathcal{L}_N(u; \Omega_N)(\boldsymbol{x}) = \sum_{\boldsymbol{y}\in\Omega_N} \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} \left(u(\boldsymbol{x}) - u(\boldsymbol{y})\right).$$

We have made explicit the dependence of the domain Ω_N on the functionals $\mathcal{E}$, $\mathcal{E}_{\text{con}}$ and the operator $\mathcal{L}_N$. Although this notation may feel cumbersome at this stage, it will aid clarity when we have two sets of data; the original dataset Ω_N and the (adversarially-) perturbed dataset $\hat{\Omega}_N$.

We will consider *Geometric Random* graphs. This construction involves weighting edges between all pairs of nodes as a function of the distance between nodes (and we say there is no edge between two nodes if the edge weight is zero). We use a parameter ε, which is often chosen relative to N, to control the length scale in the graph. This is summarised below:

(A4) $\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} = \mathbf{W}_{\varepsilon,\boldsymbol{x},\boldsymbol{y}}$ where $\mathbf{W}_{\varepsilon,\boldsymbol{x},\boldsymbol{y}} = \eta_\varepsilon(|\boldsymbol{x} - \boldsymbol{y}|)$ and $\eta_\varepsilon = \frac{1}{\varepsilon^d}\eta(\cdot/\varepsilon)$ and $\eta : [0, +\infty) \to [0, +\infty)$ is non-increasing, positive, $\eta(t) \geq 1$ for all $t \leq 1$ and $\eta(t) = 0$ for all $t \geq 2$. In addition, either η is Lipschitz continuous, or $\eta(t) = \mathbb{1}_{t\leq 1}$.

We note that whilst we use the geometric random graph construction in (A4) and we use the kNN graph in our experiments. The parameters k and ε are related as follows $k \sim N\varepsilon^d$ (cf Lemma 10). There are additional technical challenges when addressing the kNN constructions, however, we believe our results carry through to this setting (see also Remark 3 below).

The assumptions in (A4) allow us to bound the degrees of nodes and, letting $\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}$ be the adversarial perturbations of $\boldsymbol{x}, \boldsymbol{y}$, show that either (i) $\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}$ is always close to $\mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}}$ (when η is Lipschitz) or (ii) we can control the number of $\boldsymbol{x}, \boldsymbol{y}$ such that $\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}$ is not close to $\mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}}$ (when $\eta = \mathbb{1}_{\cdot\leq 1}$).

2.2. Robustness of Semi-Supervised Learning with Graph Laplacian

In this subsection, we give a theoretical bound of the following question: *how is the classification estimate affected if an adversary replaces the clean dataset $D_N = (\Omega_N, \ell_N)$ with a new, corrupted, dataset $\hat{D}_N = (\hat{\Omega}_N, \hat{\ell}_N)$?* Following Wang et al. (2018b), we assume that the adversary can corrupt

features by adding a small perturbation to the unlabeled data; the question of robustness under poisoning attacks (Dalvi et al., 2004; Lowd and Meek, 2005) is an interesting question we leave open. We assume the adversary can corrupt the unlabeled data by moving each point a maximum distance of r in ℓ_2-norm. That is, the adversary can replace the set Ω_N with a corrupted dataset $\hat{\Omega}_N$ by, for each $i = 1, \dots, N$, choosing $\hat{\boldsymbol{x}}_i \in B(\boldsymbol{x}_i, r)$ thus defining $\hat{\Omega}_N = \{\hat{\boldsymbol{x}}_i\}_{i=1}^N$. Here, and in the sequel, $\hat{\boldsymbol{x}}, \hat{\boldsymbol{x}}_i$ is understood to be a perturbation of $\boldsymbol{x}, \boldsymbol{x}_i$, respectively. Although the labels are not perturbed, the domain of the labeling function ℓ_N is, i.e. the perturbed domain is $\hat{\Gamma}_N = \{\hat{\boldsymbol{x}}_i\}_{\{i\,:\,x_i \in \Gamma_N\}}$, and so we define $\hat{\ell}_N : \hat{\Gamma}_N \to \mathbb{R}$ by $\hat{\ell}_N(\hat{\boldsymbol{x}}) = \ell_N(\boldsymbol{x})$ for all $\hat{\boldsymbol{x}} \in \hat{\Gamma}_N$. Note that $\hat{\ell}_N(\hat{\boldsymbol{x}}) = \ell_N(\boldsymbol{x})$ is precisely the condition that the adversary doesn't corrupt labels.

A *learning strategy* is a map from the dataset $D_N = (\Omega_N, \ell_N)$ to a function $u : \Omega_N \to \mathbb{R}$. For example, in the previous section we defined the learning strategy

$$D_N = (\Omega_N, \ell_N) \mapsto u(\cdot; D_N) := \operatorname{argmin}_{u:\Omega_N \to \mathbb{R}} \mathcal{E}_{\text{con}}(\cdot; D_N). \tag{2}$$

This is the learning strategy we will analyse.

Given a dataset $D_N = (\Omega_N, \ell_N)$ and a perturbation $\hat{D}_N = (\hat{\Omega}_N, \hat{\ell}_N)$ we will compare $u(\cdot; D_N)$ with $u(\cdot; \hat{D}_N)$ by $|u(\boldsymbol{x}; D_N) - u(\hat{\boldsymbol{x}}; \hat{D}_N)|$. The L^∞ distance between $u(\cdot; D_N)$ and $u(\cdot; \hat{D}_N)$ can be defined as $\max_{\boldsymbol{x} \in \Omega_N} |u(\boldsymbol{x}; D_N) - u(\hat{\boldsymbol{x}}; \hat{D}_N)|$.

We let $\delta > 0$ be a prescribed tolerance then the robustness radius is the smallest r such that it is possible to perturb $u(\cdot; D_N)$ by more than δ. More precisely, we define the δ-robustness radius below which is a modification of the robustness radius in (Wang et al., 2018b).

Definition 1 **δ-Robustness Radius.** *Let $D_N \mapsto u(\cdot; D_N)$ be a learning strategy. The δ-robustness radius $\mathcal{R}_\delta(\Omega', u, D_N)$ of u over a subset $\Omega' \subset \Omega$ given the data D_N is the smallest radius r such that $\sup_{\boldsymbol{x} \in \Omega_N} |u(\boldsymbol{x}; D_N) - u(\hat{\boldsymbol{x}}; \hat{D}_N)| > \delta$ where $|\hat{\boldsymbol{x}} - \boldsymbol{x}| < r$ for all $\boldsymbol{x} \in \Omega'$, i.e.*

$$\mathcal{R}_\delta(\Omega', u, D_N) = \inf_{r>0} \left\{ \forall \boldsymbol{x}_i \in \Omega_N \cap \Omega' \, \exists \hat{\boldsymbol{x}}_i \in B(\boldsymbol{x}_i, r) \text{ s.t. } \sup_{\boldsymbol{x} \in \Omega_N} |u(\boldsymbol{x}; D_N) - u(\hat{\boldsymbol{x}}; \hat{D}_N)| > \delta \right\}.$$

We prove δ-robustness over Ω' in order to avoid problems at the boundary $\partial\Omega$. In particular, we take Ω' such that $\operatorname{dist}(\Omega', \partial\Omega)$ is sufficiently large. We believe our arguments can be extended to the boundary but the techniques to do so are more involved and will involve estimates between the GL and its continuum analogue at the boundary. In particular, our proof uses a bound between the graph Laplacian and its continuum analogue, for which there are quantitative bounds away from the boundary, e.g. Singer (2006); Calder (2018). Near the boundary the bound between the graph Laplacian and its continuum counterpart deteriorates to $O(1)$, i.e. there are currently no established rates of convergence close to the boundary, see Calder et al. (2020).

Our main theoretical results are the following, the proofs can be found in Appendix A.

Theorem 2 **δ-Robustness of GL-based Regression.** *Under Assumptions (A1-A4) define u by (2). There exists constants $C_0 > 0$, $\varepsilon_0 > 0$, $C > c > 0$ such that if $\varepsilon \in (0, \varepsilon_0)$, $r \in (0, r_{\max})$ where $r_{\max} = c\sqrt{\beta}\varepsilon$, $\beta \in [\varepsilon^2, 1]$ and $\Omega' \subset \Omega$, with $\operatorname{dist}(\Omega', \partial\Omega) > C_0 \beta^{-\frac{1}{2}} \varepsilon \log\left(\beta^{\frac{1}{2}} \varepsilon^{-1}\right)$, then $\mathcal{R}_\delta(\Omega', u, D_N) \geq r$ with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$ where*

$$\delta = \frac{C\varepsilon}{\sqrt{\beta}} \log\left(\frac{\sqrt{\beta}}{\varepsilon}\right). \tag{3}$$

Remark 3 *The comparison with kNN given in Table 1 can be derived from the above theorem as follows. With probability at least* $1 - CNe^{-cN\varepsilon^d}$ *the number of neighbors in an* ε *connected graph scales as* $N\varepsilon^d$ *(cf Lemma 10); hence* $k \sim N\varepsilon^d$*. Now to achieve a high probability convergence rate we require that* $(N\beta\varepsilon^d)/(\log N)$ *is large, which gives a lower bound on* ε*. Choosing* ε *as small as possible then implies that* $k \gg (\log N)/\beta$*. Since* $\beta \sim (N-M)/N$ *then we arrive at the form of the bound stated in Table 1. Moreover, we believe the above theorem can be generalised to include the kNN graph construction:* $\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} = \mathbf{W}_{N,k,\boldsymbol{x},\boldsymbol{y}}$ *where* $\mathbf{W}_{N,k,\boldsymbol{x},\boldsymbol{y}} = \frac{N}{k}\mathbb{1}_{\boldsymbol{x}\sim_k\boldsymbol{y}}$ *and* $\mathbb{1}_{\boldsymbol{x}\sim_k\boldsymbol{y}} = 1$ *if* $\boldsymbol{x}$ *is a kNN of* $\boldsymbol{y}$ *(or vice versa) and* $\mathbb{1}_{\boldsymbol{x}\sim_k\boldsymbol{y}} = 0$ *otherwise. Formally, we conjecture that if one substitutes* $\varepsilon = \left(\frac{k}{N}\right)^{\frac{1}{d}}$ *then Theorem 2 continues to hold with kNN weights, i.e.* $\mathcal{R}_\delta(\Omega', u, D_N) \geq r$ *with probability at least* $1 - CNe^{-ck\beta}$ *where*

$$\delta = \frac{Ck^{\frac{1}{d}}}{N^{\frac{1}{d}}\sqrt{\beta}} \log\left(\frac{N^{\frac{1}{d}}\sqrt{\beta}}{k^{\frac{1}{d}}}\right).$$

Remark 4 *Theorem 2 shows the* δ*-robustness of GL-based regression up to* $r_{\max} = c\sqrt{\beta}\varepsilon$*. We can restate this in terms of the number of labels,* $N-M$*, by using the formal scaling* $\beta \sim (N-M)/N$*, so that* $r_{\max} = c\varepsilon\sqrt{(N-M)/N}$*. In particular, the number of labels increases the* δ*-robustness following a square-root law.*

Typically, one uses Laplacian regularisation for labeling by projecting the solution u of (1) onto the set of labels. For simplicity we consider the binary classification problem, that is we seek a function $v : \Omega_N \to \{0, 1\}$ where 0 and 1 are the two classes. As is common, we define

$$v(\boldsymbol{x}; D_N) = \begin{cases} 1 & \text{if } u(\boldsymbol{x}; D_N) \geq \frac{1}{2} \\ 0 & \text{else.} \end{cases}$$

Corollary 5 *Let* δ *be given by* (3)*. In addition to the assumptions of Theorem 2 we assume*

$$\sup_{\xi>0} \frac{1}{\xi} \mathrm{Vol}\left(\left\{\boldsymbol{x} : \frac{1}{2} - \xi \leq \ell(\boldsymbol{x}) \leq \frac{1}{2} + \xi\right\}\right) \leq A, \tag{4}$$

for some $A > 0$*, then there exists* $\Omega_\delta \subset \Omega$ *such that* $\mu(\Omega \setminus \Omega_\delta) \leq C\delta$ *and* $\mathcal{R}_0(\Omega_\delta, v, D_N) \geq r$ *with probability at least* $1 - CNe^{-cN\beta\varepsilon^d}$*.*

The classification decision boundary is $\{\boldsymbol{x} : u(\boldsymbol{x}) = \frac{1}{2}\}$, which (c.f. Theorem 6) is approximately the set $\{\boldsymbol{x} : \ell(\boldsymbol{x}) = \frac{1}{2}\}$. The additional assumption in equation (4) is in order to ensure that the set where ℓ is close to $\frac{1}{2}$ can be controlled. When $\ell(\boldsymbol{x})$ is sufficiently far from $\frac{1}{2}$ then we obtain $v(\boldsymbol{x}; D_N) = v(\hat{\boldsymbol{x}}; \hat{D}_N)$.

The proof of Theorem 2 and Corollary 5 is given in Appendix A, and relies on a quantitative bound between solutions of (1) and the true function ℓ. In particular, if the data points are close to being iid then we can use the result in Calder et al. (2020) to infer a high probability bound between $u(\cdot; D_N)$ and ℓ. Our proof shows that if $r < r_{\max}$ then we can consider the perturbed data points $\hat{\Omega}_N = \{\hat{\boldsymbol{x}}_i\}_{i=1}^N$ to be close to iid and hence apply the result to infer a high probability bound between $u(\cdot; \hat{D}_N)$ and ℓ. In fact, we can show the following result, and the proof is also given in Appendix A.

Table 2: CPU time and memory cost of GL vs. kNN classifiers for MNIST 1v7 classification (without attack), and the dataset is described in Subsection 4.1.

Classifier	kNN				GL			
Number of Nearest Neighbors (k)	5	10	15	20	5	10	15	20
Peak Memory (MB)	382	382	382	382	439	445	448	449
CPU Time (second)	2.55	2.62	2.70	2.89	2.99	3.27	3.89	5.07

Theorem 6 *Under the assumptions of Theorem 2 we have that* $\max_{\hat{\boldsymbol{x}}\in\hat{\Omega}_N} |u(\hat{\boldsymbol{x}}; \hat{D}_N) - \ell(\boldsymbol{x})| \leq \delta$ *with probability at least* $1 - CNe^{-cN\beta\varepsilon^d}$ *where* δ *is given by* (3).

Theorem 2 quantifies the robustness of GL regression. More precisely, it establishes an upper bound of the maximum allowed adversarial perturbation under which the GL solution, u, of (2) is close to the ground truth label function $\ell(\boldsymbol{x})$ with high probability. Furthermore, Theorem 6 implies that after an adversarial attack the solutions to GL regression remain close to the true solution with a quantifiable bound. In Corollary 5 we infer the robustness of GL classification.

3. Computational Complexity Analysis

We ignore the common pre-processing time for both kNN and GL. The computational complexity of kNN is dominated by nearest neighbor searching, and the total computational complexity of searching for the nearest neighbors for all unlabeled points is $O(kM\log(N-M))$ (Muja and Lowe, 2014). For the GL classifier, if we use the top $k(k \ll N)$-nearest neighbors, the total computational complexity for constructing the weight matrix would be $O(kN\log N)$. The additional computational complexity of GL comes from solving a sparse linear system of the size $N \times N$, which can be solved by using the conjugate gradient method in $O(Nk\sqrt{\kappa})$ time, with κ being the condition number of $\mathbf{W}_N$ (Shewchuk, 1994). Hence, the total computational complexity of the GL classifier is $O(kN\log N + Nk\sqrt{\kappa})$. Table 2 lists a comparison of CPU time and peak RAM consumption for MNIST 1v7 classification with different numbers of nearest neighbors (k) being used, and all the experiments are done on an Intel(R) Xeon(R) CPU E5-P2690 0 @ 2.90GHz. We provide the detailed experimental settings in Section 4. GL is slightly more computationally expensive than kNN, but for the most used k the computational overhead is not an obstacle. Moreover, GL classifier can achieve at least comparable results to kNN with a much smaller k.

4. Experiments

We consider performance of GL classifier and its enhanced variants in classifying different datasets under adversarial attacks to numerically validate: 1) our certification results in Table 1, 2) the efficacy of adversarial defenses, and 3) the advantages of the GL classifier over the kNN classifier. In all experiments below, we construct $\mathbf{W}_N$ in the same way as that used in Shi et al. (2017).

4.1. Datasets, Classifiers, & Attacks

We use the same benchmark datasets, adversarial attacks, and defenses as those used in Wang et al. (2018b), where the authors analyzed robustness of kNN. Below we give a brief description of these baselines.

Datasets. We consider three benchmarks: Halfmoon, MNIST 1v7, and Abalone (Dua and Graff, 2017). Halfmoon is a randomly generated 2D synthetic dataset in which we randomly generate 2000 points and 1000 points, with a standard deviation of 0.2, as the training and test set, respectively. For the MNIST 1v7, we randomly select 500 images for each of digit 1 and 7 to form the training set and the same size for the test set. For Abalone, we use 500/100 samples for training/test. We also generate a validation set for each of the three benchmarks with the same size as the test set.

Adversarial Attacks. We apply the same white-box (WB) and black-box (BB) attacks as that used in Wang et al. (2018b). In particular, we consider the following two WB attacks:

- **Direct attack (DA).** Given a perturbation of ℓ_2 magnitude r and the training dataset S (which might be an augmentation or pruning of D_N), the adversarial example of $\boldsymbol{x}$ is $\boldsymbol{x}_{\text{adv}} = \boldsymbol{x} + r(\boldsymbol{x} - \boldsymbol{x}')/|\boldsymbol{x} - \boldsymbol{x}'|$, where $\boldsymbol{x}'$ is the nearest neighbor of $\boldsymbol{x}$ in S that is labeled differently from $\boldsymbol{x}$. In our experiments, we vary the value of r to change the maximum ℓ_2 norm of the adversarial perturbation.

- **Kernel substitution attack (KSA).** KSA attacks a surrogate kernel classifier, trained on the same training set as that of GL classifier, using the fast gradient sign method with the target maximum ℓ_2 norm of the adversarial perturbation, see (Goodfellow et al., 2014).

In the BB attacks, we adopt three substitute classifiers: kernel classifier (Kernel), logistic regression (LR), and neural net (NN). The adversary trains these substitute models using the method of Papernot et al. (2017) and generate adversarial examples by attacking surrogate classifiers. We use the same setting as that used in Wang et al. (2018b) for both WB and BB attacks.

4.2. An Approximated Numerical Robust Certification for the GL Classifier

According to Table 1, GL classifier is robust when the number of nearest neighbors that used to construct $\mathbf{W}_N$ satisfies $k \geq C(N \log N)/(N - M)$ and $r \leq c\sqrt{(N - M)/N}\sqrt[d]{(\log N)/(N - M)}$, where c and C are two constants that are independent of N and M; we will numerical verify this in this subsection. In particular, we first perform a grid search with a small amount of labeled data, say one-fifth of the whole labeled data, for each benchmark to estimate the parameters c and C. Then, we apply the obtained c and C to compute the smallest number of nearest neighbors k and the largest perturbation r such that the GL classifier is guaranteed to be robust. Finally, we apply the above five attacks to the GL classifier with twenty different values of maximum perturbation uniformly sampled from $[0, r]$. For each attack value we do 20 independent runs. Table 3 shows the accuracies of GL classifier on three benchmarks and the theoretical values (assisted with grid search on small amount of labeled data) of k and r with different amount of labeled data $N - M$. These results show that for any benchmark with a given amount of labeled data and k that is used to construct $\mathbf{W}_N$, the GL classifier's accuracy is consistent under different adversarial attacks using different attack strengths, and these results confirm our theoretical robustness of GL classifier.

Table 3: Accuracies of GL classifier for Abalone, Halfmoon, and MNIST classification with different number of labeled data $N - M$. For different $N - M$, the GL classifier is robust when the adversarial attack does not exceed r provided the number of nearest neighbors that used to construct $\mathbf{W}_N$ is at least k. (Unit: %)

$N-M$	k	r	No Attack	WB-DA	WB-KSA	BB-LR	BB-Kernel	BB-NN
Abalone								
100	18	0.0186	64.7 ± 0.48	64.6 ± 0.53	64.2 ± 1.61	64.2 ± 0.66	64.5 ± 0.87	64.3 ± 0.43
200	14	0.0196	69.6 ± 0.72	69.2 ± 0.92	69.8 ± 0.85	69.7 ± 0.93	68.6 ± 1.55	68.9 ± 0.86
300	13	0.0198	67.5 ± 0.36	67.1 ± 0.51	67.1 ± 1.36	66.6 ± 0.39	68.1 ± 1.18	67.1 ± 0.82
400	13	0.0197	69.5 ± 0.87	69.0 ± 0.93	69.1 ± 1.50	69.8 ± 1.01	68.8 ± 1.72	69.4 ± 1.13
500	13	0.0196	70.2 ± 0.99	70.1 ± 1.09	69.8 ± 0.91	69.8 ± 0.82	69.4 ± 0.88	69.8 ± 1.21
Halfmoon								
400	22	0.0216	97.0 ± 0.08	96.8 ± 0.05	96.9 ± 0.12	96.6 ± 0.07	96.8 ± 0.09	96.6 ± 0.11
800	15	0.0195	96.6 ± 0.15	96.7 ± 0.17	96.4 ± 0.08	96.7 ± 0.08	96.5 ± 0.10	96.6 ± 0.14
1200	13	0.0177	96.8 ± 0.06	96.7 ± 0.11	96.8 ± 0.05	96.7 ± 0.11	97.0 ± 0.16	97.0 ± 0.06
1600	11	0.0165	96.7 ± 0.07	96.5 ± 0.09	96.7 ± 0.16	96.9 ± 0.08	96.7 ± 0.18	96.7 ± 0.08
2000	10	0.0156	96.6 ± 0.10	96.2 ± 0.07	96.6 ± 0.10	96.4 ± 0.15	96.5 ± 0.08	96.6 ± 0.16
MNIST								
200	19	0.407	98.9 ± 0.11	98.7 ± 0.12	98.8 ± 0.18	98.8 ± 0.09	98.8 ± 0.05	98.8 ± 0.11
400	11	0.532	98.7 ± 0.11	98.7 ± 0.15	98.6 ± 0.13	98.7 ± 0.14	98.6 ± 0.12	98.7 ± 0.05
600	9	0.609	98.8 ± 0.09	98.8 ± 0.11	98.9 ± 0.10	98.8 ± 0.14	98.9 ± 0.13	98.8 ± 0.11
800	7	0.663	99.0 ± 0.13	98.7 ± 0.10	99.0 ± 0.12	98.9 ± 0.10	98.8 ± 0.05	98.8 ± 0.12
1000	7	0.703	98.9 ± 0.13	98.9 ± 0.14	98.7 ± 0.11	98.7 ± 0.11	98.9 ± 0.13	98.9 ± 0.09

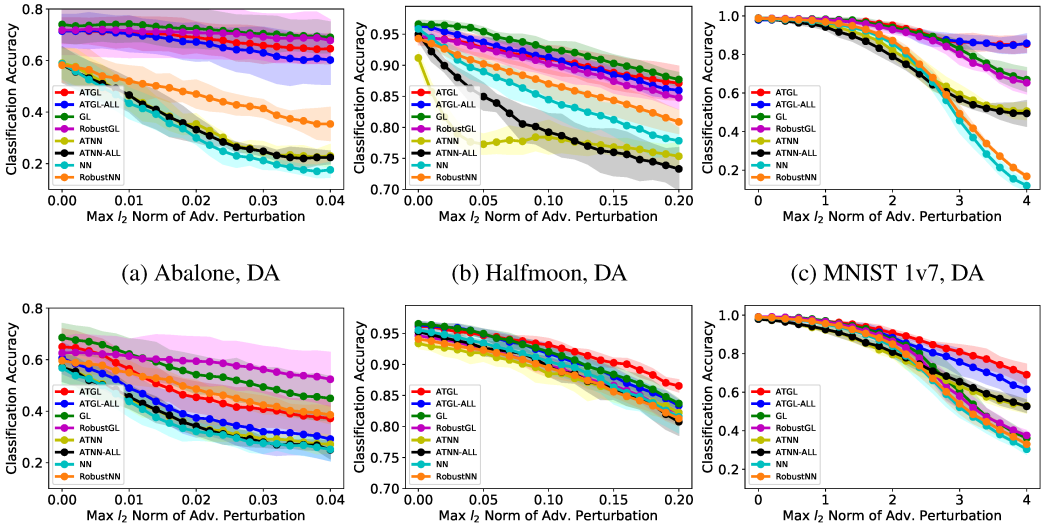
4.3. The Effects of Adversarial Defenses

Wang et al. (2018b) propose to enhance the robustness of kNN by i) augmenting the training set with $\boldsymbol{x}_{\text{adv}}$ generated from WB-DA, and resulting in the classifier ATNN; ii) augmenting the training data with adversarial examples crafted by all the above attacks and leads to the classifier ATNN-ALL; iii) pruning the training set such that the pruned training set is a-separated Wang et al. (2018b) and gives the RobustNN classifier. In the above kNN-based classifiers, if we replace kNN by GL, we get four more classifiers: GL, ATGL, ATGL-ALL, RobustGL. For GL-based classifiers we adapt the same setting as that used in (Wang et al., 2018b) for kNN-based classifiers, except the number of nearest neighbors, k, in constructing $\mathbf{W}_N$. We use the values of k in Table 3 for the whole labeled training set, i.e., 13, 10, and 7, respectively, for Abalone, Halfmoon, and MNIST.

Figure 1 shows the results (20 runs) of the above eight classifiers for three datasets classification under the WB DA and KSA attacks with different perturbations r. We provide the results of these classifiers under BB attacks in Appendix B. We vary r from 0 to 0.04, to 0.2, and to 4 for these three datasets classification, respectively. For these three datasets, GL-based classifiers are always more accurate than kNN-based classifiers with or without defense under different WB attacks. Furthermore, as r increases, the improvement becomes more significant. The training data pruning-based adversarial defense remarkably improves kNN-based classifiers in all cases (Fig. 1 (a), (b), (c), (d), and (f)) with the exception of the Halfmoon dataset under the KSA attack, and improves GL-based classifiers when classifying the Abalone dataset under the KSA attack with large r (Fig. 1 (d)). Similarly, both data augmentation methods can usually improve classifiers' robustness, especially for MNIST 1v7.

4.4. Robust Accuracy with Different Number of Training Data

Our theory indicates that for a given k, the robustness of GL classifier depends on the number of labeled data. In this subsection, we study the effects of the number of labeled data on the classifiers'



(a) Abalone, DA (b) Halfmoon, DA (c) MNIST 1v7, DA

(d) Abalone, KSA (e) Halfmoon, KSA (f) MNIST 1v7, KSA

Figure 1: Robust accuracies of GL vs. kNN classifiers for three datasets classification under WB attacks with different maximum perturbation measured in ℓ_2-norm. GL-based classifiers are consistently more accurate than kNN-based classifiers. (Best viewed on a computer screen.)

robustness. We present the results of classifying the Abalone dataset under WB attacks with $r = 0.01, 0.02$ and 0.04, respectively, in Fig. 2. The BB attacks are discussed in Appendix C. In general, under these three choices of r, the GL-based classifiers tend to be more robust as the number of training data increases.

5. Conclusions

In this paper, we gave the first rigorous analysis of the adversarial robustness of the Graph Laplacian (GL)-based semi-supervised learning algorithm under evasion attacks. Theoretically, we showed the sample limit that guarantees adversarial robustness of the GL classifier and its theoretical advantages over the k-nearest neighbor (kNN) classifier. We have also empirically shown that the robustness of the GL classifier can be remarkably improved by adversarial defenses and increasing the amount of training data. Our theoretical results echo the observation that unlabeled data can improve the robustness of machine learning algorithms (Carmon et al., 2019) and provides a potential explanation for the observation that GL-based activation function remarkably improves neural nets' robustness (Wang and Osher, 2019). Many interesting problems are remaining, for instance, how to develop a theory of robustness of GL classification under other types of the adversary, e.g., poisoning attacks?

The results in this paper are built on large data limits and there is an increasing body of literature that could provide the tools to analyse such problems as spectral clustering, e.g. (García Trillos et al., 2019; Singer, 2006), and Cheeger cuts, e.g. (Trillos et al., 2020). The main difficulty in applying these results is that one needs to be able to quantify how the "closeness" of the samples to being independent and identically distributed samples affects the labeling. In this work, the bound on the

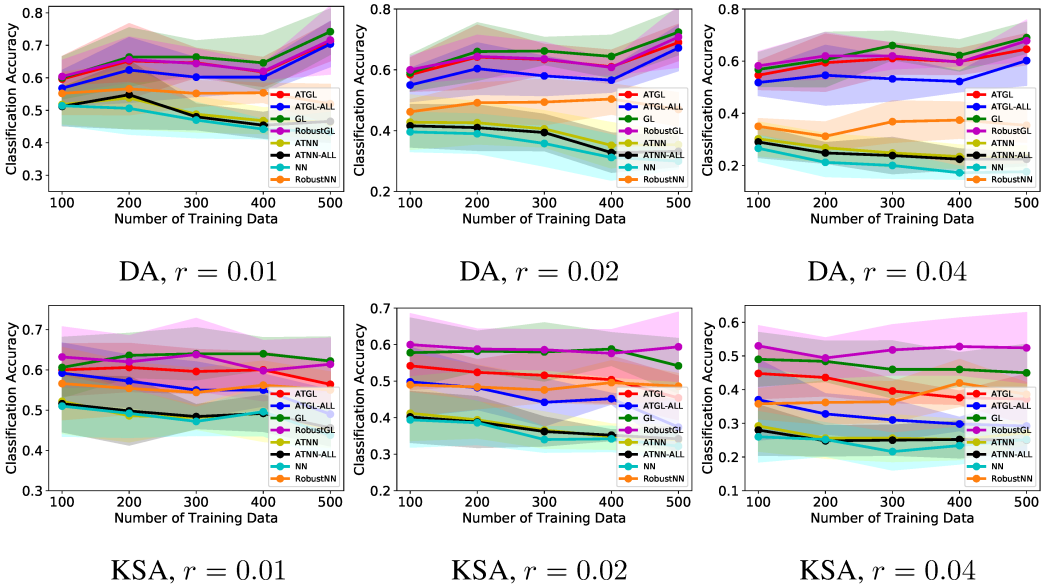


Figure 2: Robust accuracies of GL vs. kNN classifiers, trained by different number of training data, for classifying the Abalone under the WB attacks. As the number of training data increases, GL classifiers becomes more robust, while kNN classifiers are not. (Best viewed on a computer screen.)

maximum adversarial distance means that the perturbed data are still close to being independent and identically distributed and so the large data theory still applies. Extending these results to other settings is of great interest and, in particular, we expect that different methods will see different robustness scalings as we believe the regularity of the solutions will play a role.

Acknowledgments

This material is based on research sponsored by the NSF grant DMS-1924935 and DMS-1952339, and the DOE grant DE-SC0021142.

References

N. Akhtar and A. Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *arXiv preprint arXiv:1801.00553*, 2018.

M. Belkin and P. Niyogi. Semi-supervised learning on Riemannian manifolds. *Machine learning*, 56(1-3):209–239, 2004.

M. Belkin and P. Niyogi. Convergence of Laplacian eigenmaps. In *Advances in Neural Information Processing Systems*, page 129, 2007.

A. Buades, B. Coll, and J.-M. Morel. Neighborhood filters and PDE's. *Numerische Mathematik*, 105(1):1–34, 2006.

J. Calder. The game theoretic p-Laplacian and semi-supervised learning with few labels. *Nonlinearity*, 32(1), 2018.

J. Calder. Consistency of Lipschitz learning with infinite unlabeled data and finite labeled data. *SIAM Journal on Mathematics of Data Science*, 1(4):780–812, 2019.

J. Calder and D. Slepčev. Properly-weighted graph Laplacian for semi-supervised learning. *Applied Mathematics and Optimization: Special Issue on Optimization in Data Science*, 2019.

J. Calder, D. Slepčev, and M. Thorpe. Rates of convergence for Laplacian semi-supervised learning with low labelling rates. *arXiv preprint arXiv:2006.02765*, 2020.

Y. Carmon, A. Raghunathan, L. Schmidt, J. C. Duchi, and P. S. Liang. Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems*, pages 11190–11201, 2019.

N. Dalvi, P. Domingos, S. Sanghai, and D. Verma. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 99–108, 2004.

D. Dua and C. Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml/datasets/Abalone.

A. El Alaoui, X. Cheng, A. Ramdas, M. J. Wainwright, and M. I. Jordan. Asymptotic behavior of ℓ_p-based Laplacian regularization in semi-supervised learning. In *Conference on Learning Theory*, pages 879–906, 2016.

S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, and I. S. Kohane. Adversarial attacks on medical machine learning. *Science*, 363(6433):1287–1289, 2019.

N. García Trillos and D. Slepčev. A variational approach to the consistency of spectral clustering. *Applied and Computational Harmonic Analysis*, 45(2):239–281, 2018.

N. García Trillos, M. Gerlach, M. Hein, and D. Slepčev. Error estimates for spectral convergence of the graph Laplacian on random geometric graphs towards the Laplace–Beltrami operator. *Foundations of Computational Mathematics*, 2019.

G. Gilboa and S. Osher. Nonlocal operators with applications to image processing. *Multiscale Modeling & Simulation*, 7(3):1005–1028, 2009.

I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6275*, 2014.

M. Hein, J.-Y. Audibert, and U. Von Luxburg. From graphs to manifolds–weak and strong pointwise consistency of graph Laplacians. In *International Conference on Computational Learning Theory*, pages 470–485. Springer, 2005.

D. S. Hoadley and N. J. Lucas. Artificial intelligence and national security. Technical report, Congressional Research Service Reports, 2018.

D. Lowd and C. Meek. Good word attacks on statistical spam filters. In *Proceedings of the Second Conference on Email and Anti-Spam*, 2005.

A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

M. Muja and D. G. Lowe. Scalable nearest neighbor algorithms for high dimensional data. *IEEE transactions on pattern analysis and machine intelligence*, 36(11):2227–2240, 2014.

B. Nadler, N. Srebro, and X. Zhou. Statistical analysis of semi-supervised learning: The limit of infinite unlabelled data. In *Advances in Neural Information Processing Systems*, pages 1330–1338, 2009.

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*, pages 506–519, 2017.

M. F. Rios, J. Calder, and G. Lerman. Algorithms for lp-based semi-supervised learning on graphs. *arXiv preprint arXiv:1901.05031*, 2019.

S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.

B. Schneier. Adversarial machine learning against Tesla's autopilot. https://www.schneier.com/blog/archives/2019/04/adversarial_mac.html, 2019.

J. R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain, 1994.

Z. Shi, S. Osher, and W. Zhu. Weighted nonlocal Laplacian on interpolation from sparse data. *Journal of Scientific Computing*, 73(2-3):1164–1177, 2017.

Z. Shi, B. Wang, and S. J. Osher. Error estimation of weighted nonlocal Laplacian on random point cloud. *arXiv preprint arXiv:1809.08622*, 2018.

A. Singer. From graph to manifold Laplacian: The convergence rate. *Applied and Computational Harmonic Analysis*, 21(1):128–134, 2006.

D. Slepčev and M. Thorpe. Analysis of p-Laplacian regularization in semisupervised learning. *SIAM Journal on Mathematical Analysis*, 51(3):2085–2120, 2019.

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

Nicolas Garcia Trillos, Ryan Murray, and Matthew Thorpe. From graph cuts to isoperimetric inequalities: Convergence rates of cheeger cuts on data clouds. *arXiv preprint arXiv:2004.09304*, 2020.

B. Wang and S. J. Osher. Graph interpolating activation improves both natural and robust accuracies in data-efficient deep learning. *arXiv preprint arXiv:1907.06800*, 2019.

B. Wang, X. Luo, Z. Li, W. Zhu, Z. Shi, and S. Osher. Deep neural nets with interpolating function as output activation. In *Advances in Neural Information Processing Systems*, pages 743–753, 2018a.

B. Wang, Z. Shi, and S. Osher. Resnets ensemble via the feynman-kac formalism to improve natural and robust accuracies. In *Advances in Neural Information Processing Systems*, pages 1655–1665, 2019.

F. Wang, C. Zhang, H. C. Shen, and J. Wang. Semi-supervised classification using linear neighborhood propagation. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 160–167. IEEE, 2006.

Y. Wang, S. Jha, and K. Chaudhuri. Analyzing the robustness of nearest neighbors to adversarial examples. In *International Conference on Machine Learning*, pages 5133–5142, 2018b.

D. Zhou and B. Schölkopf. Regularization on discrete spaces. In *27th DAGM Conference on Pattern Recognition*, pages 361–368, 2005.

D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *Advances in neural information processing systems*, pages 321–328, 2004.

X. Zhu, Z. Ghahramani, and J. D. Lafferty. Semi-supervised learning using Gaussian fields and harmonic functions. In *Proceedings of the 20th International Conference on Machine learning*, pages 912–919, 2003.

Appendices

The appendices are structured as follows. In Section A, we proof the results in Section 2. In Section B, we numerically study the robustness of the GL-based classifier under the black-box attacks. In Section C, we empirically study the effects of the number of training data in the robustness of the GL-based classifiers and contrast it to the kNN-based classifiers. In Section D, we visualize the adversarial examples of GL-based classifiers under different adversarial attacks.

Appendix A. Proofs of the Main Results

We define the following quantities:

$$\begin{aligned} d_{N,\varepsilon}(\boldsymbol{x};\Omega_N) &= \sum_{\boldsymbol{y}\in\Omega_N} \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} = \sum_{\boldsymbol{y}\in\Omega_N} \eta_\varepsilon(|\boldsymbol{y}-\boldsymbol{x}|) \\ p_{N,\varepsilon}(\boldsymbol{x};\Gamma_N) &= \sum_{\boldsymbol{y}\in\Gamma_N} \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} = \sum_{\boldsymbol{y}\in\mathbf{\Gamma_N}} \eta_\varepsilon(|\boldsymbol{y}-\boldsymbol{x}|) \\ \mathcal{L}\varphi(\boldsymbol{x}) &= \frac{\sigma_\eta}{\rho(\boldsymbol{x})}\mathrm{div}(\rho^2\nabla\varphi)(\boldsymbol{x}) \end{aligned}$$

where $\sigma_\eta = \int_{B(0,2)} \eta(|\boldsymbol{z}|)|\boldsymbol{z}_1|^2\,\mathrm{d}\boldsymbol{z}$. We also define the constant $C_\eta = \int_{B(0,2)} \eta(|\boldsymbol{z}|)\,\mathrm{d}\boldsymbol{z}$. The value $d_{N,\varepsilon}(\boldsymbol{x};\Omega_N)$ is the degree of the node at $\boldsymbol{x}$ and $p_{N,\varepsilon}(\boldsymbol{x};\Omega_N)$ is degree of the node at $\boldsymbol{x}$ using only labeled data. We also define the "thickened" boundary $\partial_\varepsilon\Omega$ by

$$\partial_\varepsilon\Omega = \{\boldsymbol{x}\in\Omega \,:\, \mathrm{dist}(\boldsymbol{x},\partial\Omega) < \varepsilon\}\,.$$

We now recall Theorem 3.11 from Calder et al. (2020) which forms the basis for our proofs.

Theorem 7 *(Calder et al., 2020, Theorem 3.11) Let $\varepsilon \in (0,1)$ and $\beta \in [\varepsilon^2, 1]$, $\ell : \Omega \to \mathbb{R}$ be Lipschitz, $\rho \in \mathrm{C}^2(\Omega)$, Ω satisfy (A1), and $W_{\boldsymbol{x},\boldsymbol{y}}$ is constructed as in (A4). Assume there exists $C > c > 0$ such that $\Omega'_N = \{\boldsymbol{x}'_i\}_{i=1}^N \subset \Omega$ and $\Gamma'_N \subseteq \Omega'_N$ satisfy*

$$\left|d_{N,\varepsilon}(\boldsymbol{x};\Omega'_N) - C_\eta N\rho(\boldsymbol{x})\right| \le CN\sqrt{\beta} \tag{5}$$

$$d_{N,\varepsilon}(\boldsymbol{x};\Omega'_N) \ge cN \tag{6}$$

$$p_{N,\varepsilon}(\boldsymbol{x};\Gamma'_N) \ge cN\beta \tag{7}$$

$$\left|\frac{1}{N\varepsilon^2}\mathcal{L}_N(\varphi;\Omega'_N)(\boldsymbol{x}) - \mathcal{L}\varphi(\boldsymbol{x})\right| \le C\|\varphi\|_{\mathrm{C}^3(\bar\Omega)}\frac{\sqrt{\beta}}{\varepsilon} \qquad \forall\varphi\in\mathrm{C}^3(\bar\Omega) \tag{8}$$

for all $\boldsymbol{x}\in\Omega'_N\setminus\partial_{2\varepsilon}\Omega$. There exists $C_0 > 0$ and $\bar C$ such that if u' satisfies

$$\begin{aligned} \mathcal{L}_N(u';\Omega'_N) &= 0 && \forall\boldsymbol{x}\in\Omega'_N\setminus\Gamma'_N \\ u'(\boldsymbol{x}) &= \ell(\boldsymbol{x}) && \forall\boldsymbol{x}\in\Gamma'_N \end{aligned}$$

then

$$\max_{\boldsymbol{x}\in\Omega'_N\setminus\partial_\tau\Omega}\left|u'(\boldsymbol{x}) - \ell(\boldsymbol{x})\right| \le \frac{\bar C\varepsilon}{\sqrt{\beta}}\log\left(\frac{\sqrt{\beta}}{\varepsilon}\right)$$

where $\tau = \frac{C_0\varepsilon}{\sqrt{\beta}}\log\left(\frac{\sqrt{\beta}}{\varepsilon}\right)$.

The theorem is stated for any collection of data points $\Omega'_N = \{\boldsymbol{x}'_i\}_{i=1}^N$ and, under the assumptions, gives a quantitative bound between the GL solution and the label ℓ. The general idea behind the proof of Theorem 7 is to exploit the connection between solutions of the GL-based classifier with an appropriately defined random walk that terminates when it hits a labeled data point. In particular, the distance of the random walk after k steps from its starting position is like $\varepsilon\sqrt{k}$ (the size of each step is $\sim \varepsilon$ so this coincides with the usual random walk bounds in continuum domains) and the random walk will terminate after approximately $k \sim N/|\Gamma'_N|$ steps. By (7) the probability of a data point being labelled is a fraction of β. Putting all this together one gets that the error of the GL-based solution should be on the order of $\epsilon/\sqrt{\beta}$. A more careful treatment gives the extra logarithmic terms and makes precise the high probability bound, we refer to Calder et al. (2020) for details.

We will apply the theorem to the dataset Ω_N and the adversarially-perturbed domain $\hat{\Omega}_N$. In Calder et al. (2020) it is shown that the conditions (5-8) hold with high probability when the data points $\boldsymbol{x}_i$ are iid, we recall this result in Lemma 9.

To make notation easier we let

$$\begin{aligned} u &= \operatorname{argmin} \mathcal{E}_{\text{con}}(\cdot; D_N) & \hat{u} &= \operatorname{argmin} \mathcal{E}_{\text{con}}(\cdot; \hat{D}_N) \\ \mathcal{L}_N &= \mathcal{L}_N(\cdot; \Omega_N) & \hat{\mathcal{L}}_N &= \mathcal{L}_N(\cdot; \hat{\Omega}_N) \\ d_{N,\varepsilon} &= d_{N,\varepsilon}(\cdot; \Omega_N) & \hat{d}_{N,\varepsilon} &= d_{N,\varepsilon}(\cdot; \hat{\Omega}_N) \\ p_{N,\varepsilon} &= p_{N,\varepsilon}(\cdot; \Gamma_N) & \hat{p}_{N,\varepsilon} &= p_{N,\varepsilon}(\cdot; \hat{\Gamma}_N). \end{aligned}$$

We are unable to apply Theorem 7 directly to the adversarially-perturbed problem. This is because the function $\hat{\ell}_N$ is not Lipschitz continuous (and depends on N). To control the adversarially-perturbed problem we use stability of Laplace's equation. In particular, we let $\hat{w}$ satisfy

$$\begin{aligned} \hat{\mathcal{L}}_N \hat{w}(\hat{\boldsymbol{x}}) &= 0 & &\forall \hat{\boldsymbol{x}} \in \hat{\Omega}_N \setminus \hat{\Gamma}_N \\ \hat{w}(\hat{\boldsymbol{x}}) &= \ell(\hat{\boldsymbol{x}}) & &\forall \hat{\boldsymbol{x}} \in \hat{\Gamma}_N. \end{aligned}$$

We first show that $\hat{u}$ and $\hat{w}$ are close.

Lemma 8 *Assume the graph $\hat{G}_N$, which consists of nodes $\hat{\Omega}_N$ and edges between any two nodes $\hat{\boldsymbol{x}}$, $\hat{\boldsymbol{y}}$ for which $\mathbf{W}_{\varepsilon,\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}} > 0$, is connected. Let $\hat{\ell}_N(\hat{\boldsymbol{x}}) = \ell(\boldsymbol{x})$ where $|\hat{\boldsymbol{x}} - \boldsymbol{x}| \leq r$ for all $\boldsymbol{x} \in \Omega_N$ and suppose that ℓ is Lipschitz continuous. Then,*

$$\max_{\hat{\boldsymbol{x}} \in \hat{\Omega}_N} |\hat{u}(\hat{\boldsymbol{x}}) - \hat{w}(\hat{\boldsymbol{x}})| \leq \operatorname{Lip}(\ell) r.$$

Proof Let $\hat{v} = \hat{u} - \hat{w}$. Then $\hat{v}$ satisfies

$$\begin{aligned} \hat{\mathcal{L}}_N \hat{v}(\hat{\boldsymbol{x}}) &= 0 & &\forall \hat{\boldsymbol{x}} \in \hat{\Omega}_N \setminus \hat{\Gamma}_N \\ \hat{v}(\hat{\boldsymbol{x}}) &= \ell(\boldsymbol{x}) - \ell(\hat{\boldsymbol{x}}) & &\forall \hat{\boldsymbol{x}} \in \hat{\Gamma}_N. \end{aligned}$$

By the maximum principle (for example see (Calder, 2018, Theorem 3))

$$\max_{\hat{\boldsymbol{x}} \in \hat{\Omega}_N} \hat{v}(\hat{\boldsymbol{x}}) = \max_{\hat{\boldsymbol{x}} \in \hat{\Gamma}_N} \hat{v}(\hat{\boldsymbol{x}}) \leq \operatorname{Lip}(\ell) r.$$

Similarly, by the minimum principle we have $\min_{\hat{\boldsymbol{x}} \in \hat{\Omega}_N} \hat{v}(\hat{\boldsymbol{x}}) \geq -\text{Lip}(\ell) r$. Combining the two bounds we can conclude the result. ■

Next we recall that the conditions 7 hold in the unperturbed domain.

Lemma 9 *Let Assumptions (A1-A4) hold. There exists $C > c > 0$ such that if $\varepsilon \in (0,1)$ and $\beta \in [\varepsilon^2, 1]$ then*

$$\begin{aligned} |d_{N,\varepsilon}(\boldsymbol{x}) - C_\eta N \rho(\boldsymbol{x})| &\leq CN\sqrt{\beta} \\ d_{N,\varepsilon}(\boldsymbol{x}) &\geq cN \\ p_{N,\varepsilon}(\boldsymbol{x}) &\geq cN\beta \\ \left| \frac{1}{N\varepsilon^2} \mathcal{L}_N \varphi(\boldsymbol{x}) - \mathcal{L}\varphi(\boldsymbol{x}) \right| &\leq C \|\varphi\|_{\mathrm{C}^3(\bar{\Omega})} \frac{\sqrt{\beta}}{\varepsilon} \qquad \forall \varphi \in \mathrm{C}^3(\bar{\Omega}) \end{aligned}$$

for all $\boldsymbol{x} \in \Omega_N \setminus \partial_{2\varepsilon}\Omega$ with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$.

Proof The first inequality holds by choosing $\delta = \sqrt{\beta}$ in (Calder, 2018, Theorem 5) (and noting that in the proof one establishes the bound with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$). The second inequality holds by (Calder et al., 2020, Propositions 3.5 and 3.8). The third inequality holds by Remark 12 below. The fourth inequality holds by choosing $\delta = \frac{\sqrt{\beta}}{\varepsilon}$ in (Calder, 2018, Theorem 5). ■

To prove the bounds for the perturbed model we will use the following preliminary result.

Lemma 10 *Let $A_N \subset \Omega_N$ satisfy $\mathbb{P}(\boldsymbol{x} \in A_N) = \alpha \in [0,1]$. Then, there exists, $a > 0$ and $C > c > 0$ (independent of α) such that for all $\tau \in (0,1]$ and $0 < \vartheta \leq 1$*

$$\begin{aligned} \mathbb{P}\left((1-\vartheta - a\tau) C_\tau(\boldsymbol{x}) N\alpha \leq \#\left\{\boldsymbol{y} \in A_N \,:\, |\boldsymbol{x}-\boldsymbol{y}| \leq \tau\right\} \leq (1+\vartheta+a\tau) C_\tau(x) N\alpha, \, \forall \boldsymbol{x} \in \Omega_N\right) \\ \geq 1 - CNe^{-cN\alpha\tau^d\vartheta^2} \end{aligned}$$

where $C_\tau(\boldsymbol{x}) = \rho(\boldsymbol{x}) \text{Vol}(B(\boldsymbol{x},\tau) \cap \Omega)$. Moreover, there exists $\tau_0 > 0$, $C_2 > C_1 > 0$, such that, for all $\tau \in (0, \tau_0)$,

$$\mathbb{P}\left(C_1 N\alpha\tau^d \leq \#\left\{\boldsymbol{y} \in A_N \,:\, |\boldsymbol{x}-\boldsymbol{y}| \leq \tau\right\} \leq C_2 N\alpha\tau^d, \, \forall \boldsymbol{x} \in \Omega_N\right) \geq 1 - CNe^{-cN\alpha\tau^d}.$$

Remark 11 *We can apply the above lemma to lower bound the number of labeled data points in $B(\boldsymbol{x},\varepsilon)$. In particular, if $\boldsymbol{x} \in \Omega \setminus \partial_{2\varepsilon}\Omega$ then $C_\varepsilon(\boldsymbol{x}) = \rho(\boldsymbol{x})\text{Vol}(B(0,1))\varepsilon^d$, and if $\eta = \mathbb{1}_{\cdot \leq 1}$ then $C_\eta = \text{Vol}(B(0,1))$. We choose $\alpha = 1$ and $\tau = \varepsilon$ to infer*

$$d_{N,\varepsilon}(\boldsymbol{x}) = \sum_{\boldsymbol{y} \in \Omega_N} \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} = \frac{1}{\varepsilon^d} \#\left\{\boldsymbol{y} \in \Omega_N \,:\, |\boldsymbol{x}-\boldsymbol{y}| \leq \varepsilon\right\} \leq C_\eta N\rho(\boldsymbol{x}) + O\left((\vartheta+\varepsilon)N\right)$$

with probability at least $1 - CNe^{-cN\varepsilon^d\vartheta^2}$. Choosing $\vartheta = \sqrt{\beta}$ proves the first inequality in Lemma 9 for the special case $\eta = \mathbb{1}_{\cdot \leq 1}$.

Remark 12 *We can also use the above lemma to bound the number of labeled data points in $B(\boldsymbol{x}, \varepsilon)$. Let $\boldsymbol{x} \in \Omega \setminus \partial_{2\varepsilon}\Omega$, and choose $\alpha = \beta$. Applying the second bound in Lemma 10, and using that $\eta(t) \geq 1$ for all $t < 1$, we have*

$$p_{N,\varepsilon}(\boldsymbol{x}) = \sum_{\boldsymbol{y} \in \Omega_N} \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} \geq \frac{1}{\varepsilon^d} \#\left\{\boldsymbol{y} \in \Gamma_N \,:\, |\boldsymbol{x} - \boldsymbol{y}| \leq \varepsilon\right\} \geq C_1 N\beta$$

with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$. This proves the third inequality in Lemma 9.

Proof [Proof of Lemma 10] Fix $\boldsymbol{x} \in \Omega_N$ and $\tau > 0$. Let $\xi_{\boldsymbol{y}} = 1$ if $\boldsymbol{y} \in A_N$ and $|\boldsymbol{x} - \boldsymbol{y}| \leq \tau$, and $\xi_{\boldsymbol{y}} = 0$ otherwise. We can write

$$\#\left\{\boldsymbol{y} \in A_N \,:\, |\boldsymbol{x} - \boldsymbol{y}| \leq \tau\right\} = \sum_{\boldsymbol{y} \in \Omega_N} \xi_{\boldsymbol{y}}.$$

By Bernstein's inequality

$$\mathbb{P}\left(\sum_{\boldsymbol{y} \in \Omega_N} (\xi_{\boldsymbol{y}} - \mathbb{E}[\xi_{\boldsymbol{y}}]) \geq t\right) \leq \exp\left(-\frac{ct^2}{N\sigma^2 + t}\right)$$

for all $t > 0$ and where

$$\sigma^2 = \mathbb{E}\left(\xi_{\boldsymbol{y}} - \mathbb{E}[\xi_{\boldsymbol{y}}]\right)^2$$

(note that the right hand side is independent of $\boldsymbol{y}$). Using the lower bound on the density ρ of $\boldsymbol{x}$ we infer

$$\begin{aligned}
\mathbb{P}(\xi_{\boldsymbol{y}} = 1) &= \mathbb{P}\left(B\left(\boldsymbol{x}, \tau\right) \cap \Omega\right) \mathbb{P}(\boldsymbol{y} \in A_N) \\
&= \alpha \mathrm{Vol}\left(B(\boldsymbol{x}, \tau) \cap \Omega\right) \left(\rho(\boldsymbol{x}) + O(\tau)\right) \\
&= \alpha C_\tau(\boldsymbol{x}) \left(1 + O(\tau)\right).
\end{aligned}$$

So there exists $a > 0$ such that

$$\alpha C_\tau(\boldsymbol{x}) \left(1 - a\tau\right) \leq \mathbb{E}[\xi_{\boldsymbol{y}}] = \mathbb{P}(\xi_{\boldsymbol{y}} = 1) \leq \alpha C_\tau(\boldsymbol{x}) \left(1 + a\tau\right).$$

Hence,

$$\sigma^2 \leq \mathbb{E}[\xi_{\boldsymbol{y}}^2] = \mathbb{E}[\xi_{\boldsymbol{y}}] \leq C\alpha\tau^d$$

for some $C > 0$. We can then bound

$$\begin{aligned}
\mathbb{P}\left(\#\left\{\boldsymbol{y} \in A_N \,:\, |\boldsymbol{x} - \boldsymbol{y}| \leq \tau\right\} \leq t + C_\tau(\boldsymbol{x}) N\alpha(1 + a\tau)\right) & \\
\geq \mathbb{P}\left(\sum_{\boldsymbol{y} \in \Omega_N} (\xi_{\boldsymbol{y}} - \mathbb{E}[\xi_{\boldsymbol{y}}]) \leq t\right) & \\
\geq 1 - \exp\left(-\frac{ct^2}{N\sigma^2 + t}\right) & \\
\geq 1 - \exp\left(-\frac{ct^2}{CN\alpha\tau^d + t}\right). &
\end{aligned}$$

Similarly,

$$\begin{aligned}\mathbb{P}\left(\#\left\{\boldsymbol{y}\in A_N\,:\,|\boldsymbol{x}-\boldsymbol{y}|\le\tau\right\}\ge -t+C_\tau(x)N\alpha(1-a\tau)\right)&\\ \ge\mathbb{P}\left(\sum_{\boldsymbol{y}\in\Omega_N}\left(\xi_{\boldsymbol{y}}-\mathbb{E}[\xi_{\boldsymbol{y}}]\right)\ge -t\right)&\\ \ge 1-\exp\left(-\frac{ct^2}{N\sigma^2+t}\right)&\\ \ge 1-\exp\left(-\frac{ct^2}{CN\tau^d\alpha+t}\right).&\end{aligned}$$

Choosing $t=C_\tau(x)N\alpha\vartheta$ implies

$$\begin{aligned}\mathbb{P}\left((1-a\tau-\vartheta)C_\tau(x)N\alpha\le\#\left\{y\in A_N\,:\,|\boldsymbol{x}-\boldsymbol{y}|\le\tau\right\}\le(1+a\tau+\vartheta)C_\tau(\boldsymbol{x})N\alpha\right)&\\ \ge 1-2e^{-cN\alpha\tau^d\vartheta^2}&\end{aligned}$$

where we restrict $\vartheta\le 1$. Union bounding (Fréchet inequality for logical conjugation) implies the first result.

Choose $\tau_0>0$ sufficiently small so that $1-a\tau_0>0$ and set $\vartheta=\frac{1}{2}(1-a\tau_0)>0$. Noticing that we can find C_1, C_2 such that $0<C_1\le(1-a\tau-\vartheta)C(\boldsymbol{x})\le(1+a\tau+\vartheta)C(\boldsymbol{x})\le C_2$ we can conclude the second part of the lemma. ■

We are left to show the analogue of Lemma 9 for the perturbed quantities $\hat{d}_{N,\varepsilon}$, $\hat{p}_{N,\varepsilon}$ and $\hat{\mathcal{L}}_N$.

Lemma 13 *Under Assumptions (A1-A4) and assuming $|\boldsymbol{x}_i-\hat{\boldsymbol{x}}_i|\le r$ for all $i=1,\dots,N$ there exists $\varepsilon_0>0$, $C>c>0$ such that if $\varepsilon\in(0,\varepsilon_0)$, $\beta\in[\varepsilon^2,1]$ and $r\in(0,c\sqrt{\beta}\varepsilon]$ then*

$$\begin{aligned}\left|\hat{d}_{N,\varepsilon}(\hat{\boldsymbol{x}})-C_\eta N\rho(\hat{\boldsymbol{x}})\right|&\le CN\sqrt{\beta}\\ \hat{d}_{N,\varepsilon}(\hat{\boldsymbol{x}})&\ge cN\\ \hat{p}_{N,\varepsilon}(\hat{\boldsymbol{x}})&\ge cN\beta\\ \left|\frac{1}{N\varepsilon^2}\hat{\mathcal{L}}_N\varphi(\hat{\boldsymbol{x}})-\mathcal{L}\varphi(\hat{\boldsymbol{x}})\right|&\le C\|\varphi\|_{\mathrm{C}^3(\bar{\Omega})}\frac{\sqrt{\beta}}{\varepsilon}\qquad\qquad\forall\varphi\in\mathrm{C}^3(\bar{\Omega})\end{aligned}$$

for all $\hat{\boldsymbol{x}}\in\hat{\Omega}_N\setminus\partial_{2\varepsilon}\Omega$ with probability at least $1-CNe^{-cN\beta\varepsilon^d}$.

Proof Let $\operatorname{dist}(\hat{\boldsymbol{x}},\partial\Omega)>2\varepsilon$. We first consider a bound on the difference between weights $\left|\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}-\mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}}\right|$. If η is Lipschitz continuous we have,

$$\begin{aligned}\left|\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}-\mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}}\right|&=\frac{1}{\varepsilon^d}\left|\eta\left(\frac{\boldsymbol{x}-\boldsymbol{y}}{\varepsilon}\right)-\eta\left(\frac{\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}}{\varepsilon}\right)\right|\\ &\le\frac{\operatorname{Lip}(\eta)}{\varepsilon^{d+1}}\left||\boldsymbol{x}-\boldsymbol{y}|-|\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}|\right|\mathbb{1}_{|\boldsymbol{x}-\boldsymbol{y}|\le 2(\varepsilon+r)}\\ &\le\frac{2\operatorname{Lip}(\eta)r}{\varepsilon^{d+1}}\mathbb{1}_{|\boldsymbol{x}-\boldsymbol{y}|\le 2(\varepsilon+r)}.\end{aligned}$$

On the other hand, If $\eta(t) = \mathbb{1}_{t<1}$ then we have

$$\left|\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} - \mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}}\right| \leq \frac{1}{\varepsilon^d}\mathbb{1}_{\varepsilon-2r\leq|\boldsymbol{x}-\boldsymbol{y}|\leq\varepsilon+2r}.$$

Now, for the first inequality we have, when η is Lipschitz continuous,

$$\begin{aligned}\left|\hat{d}_{N,\varepsilon}(\hat{\boldsymbol{x}}) - d_{N,\varepsilon}(\boldsymbol{x})\right| &\leq \sum_{\boldsymbol{y}\in\Omega_N}\left|\mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}} - \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}\right| \\ &\leq \frac{Cr}{\varepsilon^{d+1}}\#\left\{\boldsymbol{y}\in\Omega_N \,:\, |\boldsymbol{x}-\boldsymbol{y}|\leq 2(\varepsilon+r)\right\} \\ &\leq \frac{CrN(\varepsilon+r)^d}{\varepsilon^{d+1}} \\ &\leq CN\sqrt{\beta}\end{aligned}$$

by Lemma 10 with probability at least $1 - CNe^{-cN\varepsilon^d}$. And when $\eta(t) = \mathbb{1}_{t<1}$,

$$\begin{aligned}\left|\hat{d}_{N,\varepsilon}(\hat{\boldsymbol{x}}) - d_{N,\varepsilon}(\boldsymbol{x})\right| &\leq \frac{1}{\varepsilon^d}\#\left\{\boldsymbol{y}\in\Omega_N \,:\, \varepsilon-2r\leq|\boldsymbol{x}-\boldsymbol{y}|\leq\varepsilon+2r\right\} \\ &\leq \frac{CN}{\varepsilon^d}\left[(1+\vartheta+a(\varepsilon+2r))(\varepsilon+2r)^d - (1-\vartheta-a(\varepsilon-2r))(\varepsilon-2r)^d\right] \\ &\leq CN(\sqrt{\beta}+\vartheta)\end{aligned}$$

by Lemma 10 with probability at least $1 - CNe^{-cN\varepsilon^d\vartheta^2}$, choosing $\vartheta = \sqrt{\beta}$ we have, in both cases,

$$\left|\hat{d}_{N,\varepsilon}(\hat{\boldsymbol{x}}) - d_{N,\varepsilon}(\boldsymbol{x})\right| \leq \sum_{\boldsymbol{y}\in\Omega_N}\left|\mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} - \mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}}\right| \leq CN\sqrt{\beta}$$

with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$. By the triangle inequality and Lemma 9

$$\left|\hat{d}_{N,\varepsilon}(\boldsymbol{x}) - C_\eta N\rho(\boldsymbol{x})\right| \leq \left|\hat{d}_{N,\varepsilon}(\boldsymbol{x}) - d_{N,\varepsilon}(\boldsymbol{x})\right| + |d_{N,\varepsilon}(\boldsymbol{x}) - C_\eta N\rho(\boldsymbol{x})| \leq CN\sqrt{\beta}$$

with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$.

By assuming that c is sufficiently small (where $r \leq c\beta\varepsilon$) we can make C in the above bound arbitrarily small. Hence we can assume that $C < C_\eta\rho_{\min}$ and therefore the second inequality holds.

Similarly, for the third inequality when η is Lipschitz,

$$\begin{aligned}|\hat{p}_{N,\varepsilon}(\hat{\boldsymbol{x}}) - p_{N,\varepsilon}(\boldsymbol{x})| &\leq \sum_{\boldsymbol{y}\in\Gamma_N}\left|\mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}} - \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}\right| \\ &\leq \frac{Cr}{\varepsilon^{d+1}}\#\left\{\boldsymbol{y}\in\Gamma_N \,:\, |\boldsymbol{x}-\boldsymbol{y}|\leq 2(\varepsilon+r)\right\} \\ &\leq \frac{CrN\beta(\varepsilon+r)^d}{\varepsilon^{d+1}} \\ &\leq CN\beta^{\frac{3}{2}}\end{aligned}$$

by Lemma 10 with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$. And, if $\eta = \mathbb{1}_{\cdot<1}$,

$$|\hat{p}_{N,\varepsilon}(\hat{\boldsymbol{x}}) - p_{N,\varepsilon}(\boldsymbol{x})| \leq \frac{1}{\varepsilon^d}\#\left\{\boldsymbol{y}\in\Gamma_N \,:\, \varepsilon-2r\leq|\boldsymbol{x}-\boldsymbol{y}|\leq\varepsilon+2r\right\} \leq CN\beta^{\frac{3}{2}}$$

by Lemma 10 with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$. Again, by assuming that c is sufficiently small (where $r \leq c\beta\varepsilon$) we can make C in the above bound arbitrarily small so that

$$\hat{p}_{N,\varepsilon}(\hat{\boldsymbol{x}}) \geq p_{N,\varepsilon}(\boldsymbol{x}) - |\hat{p}_{N,\varepsilon}(\hat{\boldsymbol{x}}) - p_{N,\varepsilon}(\boldsymbol{x})| \geq cN\beta - CN\beta^{\frac{3}{2}} \geq cN\beta$$

by Lemma 9 with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$.

For the final inequality we have

$$\begin{aligned}
\left|\hat{\mathcal{L}}_N\varphi(\hat{\boldsymbol{x}}) - \mathcal{L}_N\varphi(\boldsymbol{x})\right| &\leq \left|\sum_{\boldsymbol{y}\in\Omega_N} \left(\mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}} - \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}\right)\left(\varphi(\hat{\boldsymbol{x}}) - \varphi(\hat{\boldsymbol{y}})\right)\right| \\
&\qquad + \left|\sum_{\boldsymbol{y}\in\Omega_N} \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}\left(\varphi(\hat{\boldsymbol{x}}) - \varphi(\hat{\boldsymbol{y}}) - \varphi(\boldsymbol{x}) + \varphi(\boldsymbol{y})\right)\right| \\
&\leq \|\varphi\|_{\mathrm{C}^1(\overline{\Omega})}(\varepsilon + 2r) \sum_{\boldsymbol{y}\in\Omega_N} \left|\mathbf{W}_{\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}} - \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}}\right| \\
&\qquad + 2r\|\varphi\|_{\mathrm{C}^1(\overline{\Omega})} \sum_{\boldsymbol{y}\in\Omega_N} \mathbf{W}_{\boldsymbol{x},\boldsymbol{y}} \\
&\leq CN\|\varphi\|_{\mathrm{C}^1(\overline{\Omega})}(\varepsilon + 2r)\sqrt{\beta} + CN\|\varphi\|_{\mathrm{C}^1(\overline{\Omega})}\sqrt{\beta}\varepsilon \\
&\leq CN\|\varphi\|_{\mathrm{C}^1(\overline{\Omega})}\sqrt{\beta}\varepsilon
\end{aligned}$$

by Lemma 9 with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$. ■

It is now very easy to prove Theorem 2.

Proof [Proof of Theorem 2 and Theorem 6.] By Theorem 7, Lemmas 8, 9 and 13, and the Lipschitz condition on ℓ we have

$$\begin{aligned}
|u(\boldsymbol{x}) - \ell(\boldsymbol{x})| &\leq \frac{C\varepsilon}{\sqrt{\beta}} \log \frac{\sqrt{\beta}}{\varepsilon} \\
|\hat{w}(\hat{\boldsymbol{x}}) - \ell(\hat{\boldsymbol{x}})| &\leq \frac{C\varepsilon}{\sqrt{\beta}} \log \frac{\sqrt{\beta}}{\varepsilon} \\
|\hat{w}(\hat{\boldsymbol{x}}) - \hat{u}(\hat{\boldsymbol{x}})| &\leq C\sqrt{\beta}\varepsilon \\
|\ell(\hat{\boldsymbol{x}}) - \ell(\boldsymbol{x})| &\leq C\sqrt{\beta}\varepsilon
\end{aligned}$$

with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$. So by the triangle inequality

$$|u(\boldsymbol{x}) - \hat{u}(\hat{\boldsymbol{x}})| \leq \frac{C\varepsilon}{\sqrt{\beta}} \log \frac{\sqrt{\beta}}{\varepsilon}$$

and

$$|\ell(\boldsymbol{x}) - \hat{u}(\hat{\boldsymbol{x}})| \leq \frac{C\varepsilon}{\sqrt{\beta}} \log \frac{\sqrt{\beta}}{\varepsilon}$$

with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$. This implies that $\mathcal{R}_\delta(\Omega', u, D_N) \geq r$ with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$. ■

Corollary 5 is not difficult to show using the Theorem 2.

Proof [Proof of Corollary 5.] We work on the set of realisations of $\{\boldsymbol{x}_i\}_{i=1}^{\infty}$ such that the conclusions of Theorem 2 hold; i.e. the following statements hold with probability at least $1 - CNe^{-cN\beta\varepsilon^d}$. Let us define $\hat{v}(\hat{\boldsymbol{x}}) = v(\hat{\boldsymbol{x}}; \hat{D}_N)$ and

$$\Omega_\delta = \left\{ \boldsymbol{x} \,:\, \frac{1}{2} - 2\delta > \ell(\boldsymbol{x}) \text{ or } \ell(\boldsymbol{x}) \geq \frac{1}{2} + 2\delta \right\}.$$

By Assumption 4 we have that $\mathrm{Vol}(\Omega_\delta^c) \leq C\delta$. Moreover, for $\boldsymbol{x} \in \Omega_N$ such that $\ell(\boldsymbol{x}) \geq \frac{1}{2} + 2\delta$ we have

$$\hat{u}(\hat{\boldsymbol{x}}) \geq u(\boldsymbol{x}) - \delta \geq \ell(\boldsymbol{x}) - 2\delta \geq \frac{1}{2}$$

so $|v(\boldsymbol{x}) - \hat{v}(\hat{\boldsymbol{x}})| = 0$. And similarly, if $\boldsymbol{x} \in \Omega_N$ is such that $\ell(\boldsymbol{x}) < \frac{1}{2} - 2\delta$ we have

$$\hat{u}(\hat{\boldsymbol{x}}) \leq u(\boldsymbol{x}) + \delta \leq \ell(\boldsymbol{x}) + 2\delta < \frac{1}{2}$$

so $|v(\boldsymbol{x}) - \hat{v}(\hat{\boldsymbol{x}})| = 0$. Hence, for any $\boldsymbol{x} \in \Omega_N \cap \Omega_\delta$ we have $|v(\boldsymbol{x}) - \hat{v}(\hat{\boldsymbol{x}})| = 0$. Therefore, $\mathcal{R}_0(\Omega_\delta, v, D_N) \geq r$ as required. ■

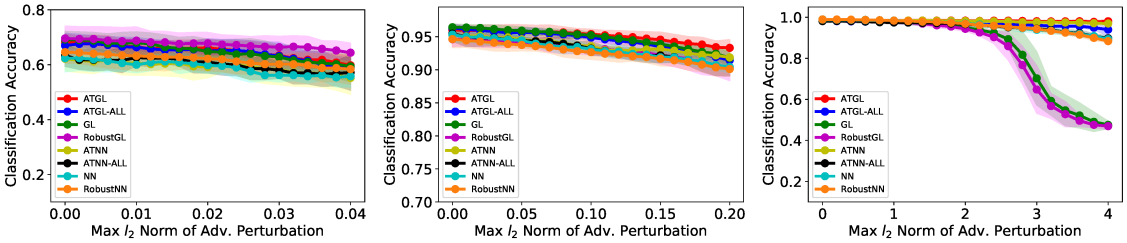
Appendix B. Robust Accuracy Under Different Black-Box Adversarial Attacks

In this section, we report the robustness results of the GL- and kNN-based classifiers under the three different BB attacks, where these BB attacks attack the substitute models: logistic regression (LR), neural net (NN), and kernel classifier (Kernel). For Abalone classification, GL-based classifiers performs better than the corresponding kNN-based classifiers; in particular, the RobustGL (GL classifier with the pruned dataset that satisfies the a-separation condition (Wang et al., 2018b)) outperforms all the other classifiers under three BB attacks with different maximum perturbation. Data augmentation with adversarial data can also enhance the classifiers' adversarial robustness. These adversarial defenses can also improve kNN-based classifiers for Abalone classification. For Halfmoon classification, GL-based classifiers again outperforms kNN-based classifiers consistently. Furthermore, with adversarial data augmentation, the classifiers' robustness can be significantly improved.

For MNIST 1v7 classification, GL-based classifiers are not always more robust than kNN-based classifiers. Nevertheless, ATGL or ATGL-ALL (GL with adversarial data augmentation) gives the most robust classification under different BB attacks.

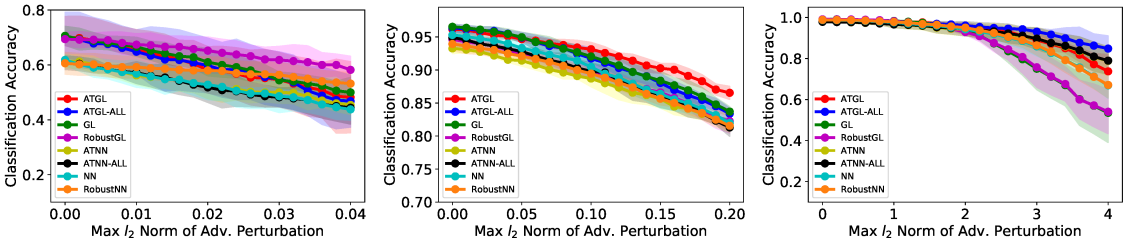
Appendix C. Robust Accuracy with Different Number of Training Data

We have numerically shown that as the number of training data increases, the robustness of GL-based classifiers on the Abalone dataset will increase under the WB attacks. In this section, we consider the effects of the number of training data on the classifiers' robustness under BB attacks. As shown in Fig. 6, under the BB attack with LR or Kernel as the substitute model, both GL- and kNN-based classifiers become more robust as the number of training data increases. As we increase the number of training data from 100 to 500, the models classification accuracy can increase more than 5% under the BB attacks with different maximum perturbations. For BB attacks with kNN



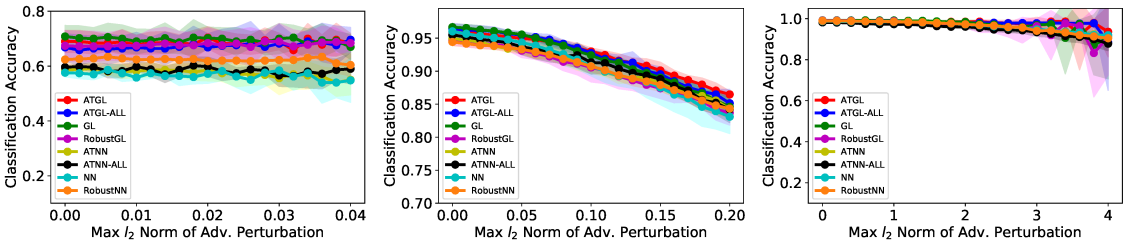
(a) Abalone (b) Halfmoon (c) MNIST 1v7

Figure 3: Robust accuracies of GL- vs. kNN-based classifiers for three different datasets classification under black-box attack using logistic regression as the substitute model with different maximum perturbation in l_2-norm. (Best viewed on a computer screen.)



(a) Abalone (b) Halfmoon (c) MNIST 1v7

Figure 4: Robust accuracies of GL vs. kNN classifiers for three different datasets classification under BB attack using kernel model as the substitute model with different maximum perturbation in l_2-norm. (Best viewed on a computer screen.)



(a) Abalone (b) Halfmoon (c) MNIST 1v7

Figure 5: Robust accuracies of GL vs. kNN classifiers for three different datasets classification under BB attack using neural net as the surrogate model with different maximum perturbation in l_2-norm. (Best viewed on a computer screen.)

as the surrogate model, the GL- and kNN- based classifiers accuracy under adversarial attacks

fluctuates, the average robust accuracy of the GL-based classifiers also has an upward trend as the number of training data increases.

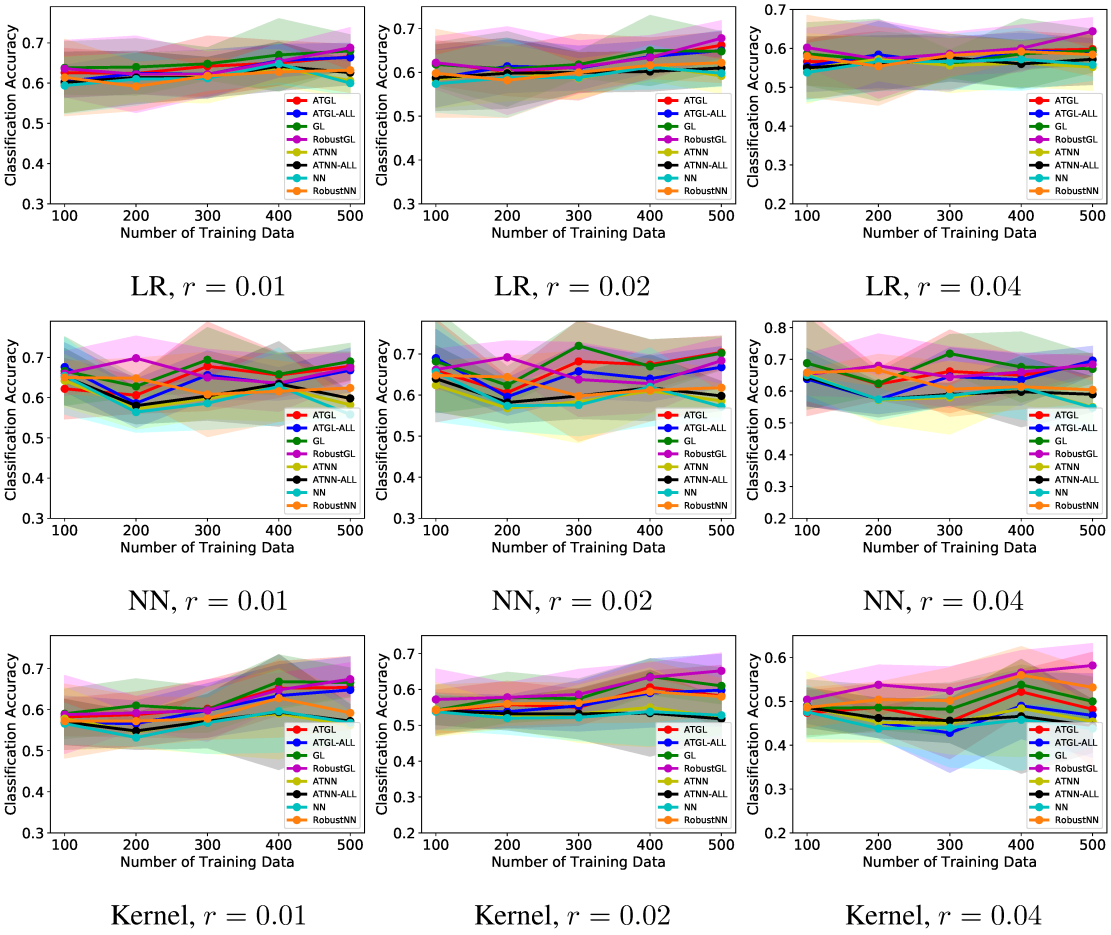


Figure 6: Robust accuracies of GL vs. kNN classifiers, trained with different numbers of training data, for classifying the Abalone dataset under the BB attacks. For both LR and Kernel surrogate models, both GL and kNN classifiers becomes more robust as the number of training data increases. When the NN is used as the substitute model, the accuracy fluctuates as the size of training data increases. (Best viewed on a computer screen.)

Appendix D. Visualizing the Adversarial Examples of GL-Based Classifiers

In this part, we visualize the adversarial examples of the MNIST 1v7 and the Halfmoon datasets under adversarial attacks with $r = 4$ and 0.2, respectively. For the MNIST 1v7, under the strong WB attacks ($r = 4$); in particular, the KSA attack, many adversarial digit 1's are hard for us to classify, and this explains there is a sharp accuracy drop when r is large (see Fig. 1 (f)). For the Halfmoon dataset, when we apply the strong WB-KSA attack, the adversarial examples enter the

opposite territory, which leads to misclassification. The GL-based classifiers leverage the global geometric information to improve the classification robustness in this scenario.

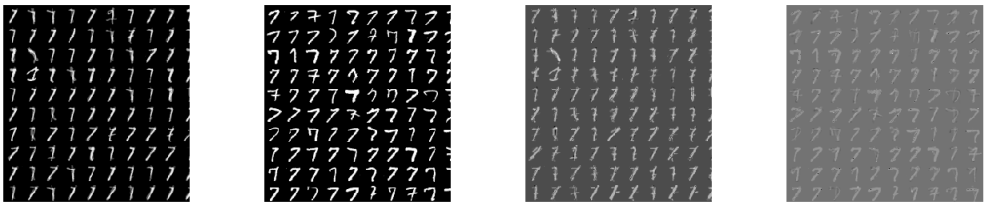


Figure 7: Adversarial images of the MNIST generated by WB attacks ($r = 4.0$). Adversarial images by both attacks are harder to classify than the clean images, and KSA attack changes the contrast of the images severely. Left (Right) two: adversarial images of digits 1 and 7 from DA (KSA).

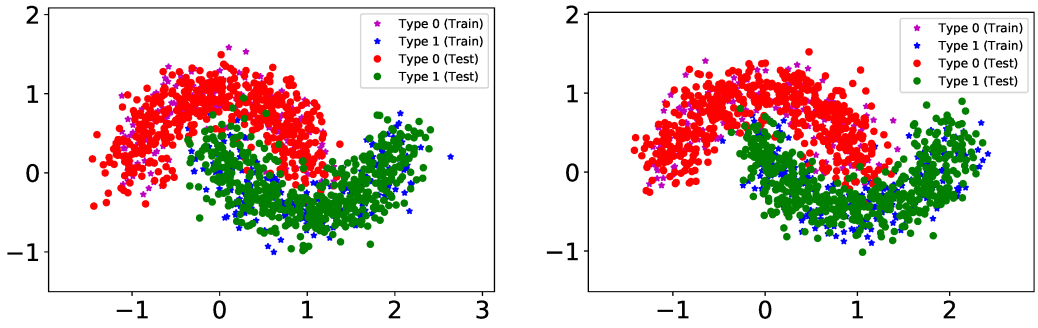
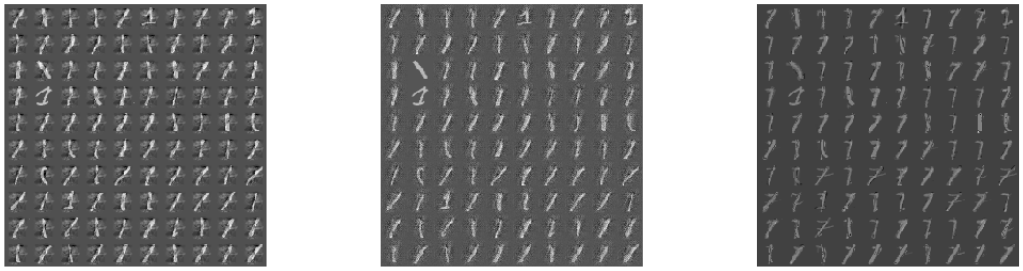


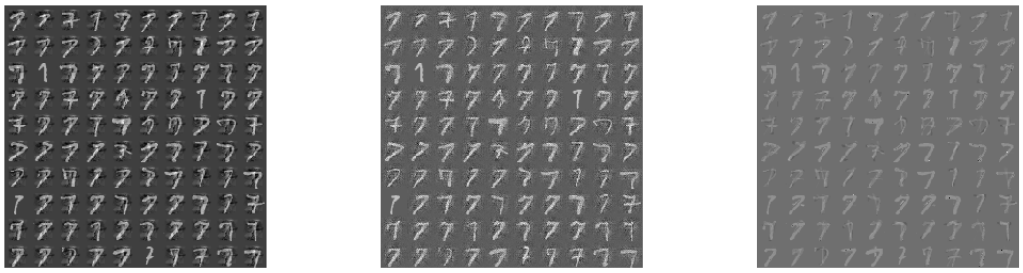
Figure 8: Plots of training and adversarial points of the test data for the Halfmoon dataset. The adversarial points are generated by WB attacks ($r = 0.2$), which have a significant overlap with the training data of different type. Left: DA; Right: KSA. (Best viewed on a computer screen.)

Figures 9 and 10 show the adversarial examples of the MNIST 1v7 and the Halfmoon datasets under the BB attacks with $r = 4$ and 0.2, respectively. For MNIST 1v7, under the BB attack with three different substitute models, namely, logistic regression (LR), neural net (NN), and the kernel classifier (Kernel), the contrast of the adversarial images has been changed dramatically from the original images. The adversarial images of digit 1 and digit 7 are much harder to classifier even for us compared with the clean images, especially those generated by BB attacks with LR or Kernel model substitution.

As shown in Fig. 10, for the Halfmoon dataset, with a maxium perturbation $r = 0.2$ the adversarial version of the test set of Type 0 will have a significant overlap with the training set of Type 1, and vice versa. This will degrade the classification accuracy GL and kNN classifiers severely, as shown in Figures 3-5.

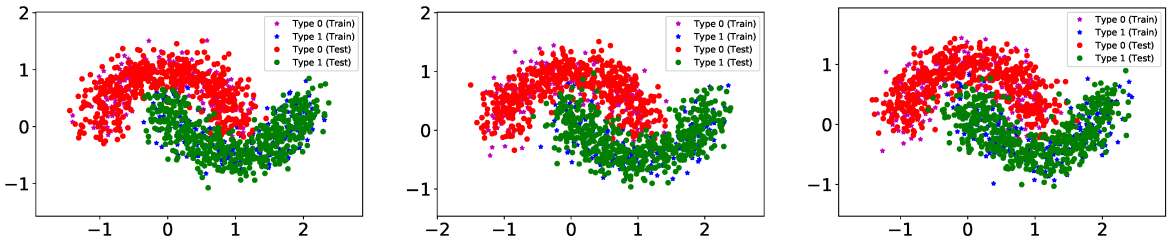


LR (Digit 1) NN (Digit 1) Kernel (Digit 1)



LR (Digit 7) NN (Digit 7) Kernel (Digit 7)

Figure 9: Adversarial images of the MNIST generated by BB attacks with $r = 4.0$. Both attacks make the adversarial images harder to classifier, and change the contrast of the adversarial remarkably. (Best viewed on a computer screen.)



LR NN Kernel

Figure 10: Plots of training and adversarial points of the test data for the Halfmoon dataset. The adversarial points are generated by BB attacks $r = 0.2$. The adversarial points have a significant overlap with the training data of different type. (Best viewed on a computer screen.)