On the Power of Abstention and Data-Driven Decision Making for Adversarial Robustness

Anonymous authors
Paper under double-blind review

Abstract

We formally define a feature-space attack where the adversary can perturb datapoints by arbitrary amounts but in restricted directions. By restricting the attack to a small random subspace, our model provides a clean abstraction for non-Lipschitz networks which map small input movements to large feature movements. We prove that classifiers with the ability to abstain are provably more powerful than those that cannot in this setting. Specifically, we show that no matter how well-behaved the natural data is, any classifier that cannot abstain will be defeated by such an adversary. However, by allowing abstention, we give a parameterized algorithm with provably good performance against such an adversary when classes are reasonably well-separated in feature space and the dimension of the feature space is high. We further use a data-driven method to set our algorithm parameters to optimize over the accuracy vs. abstention trade-off with strong theoretical guarantees. Our theory has direct applications to the technique of contrastive learning, where we empirically demonstrate the ability of our algorithms to obtain high robust accuracy with only small amounts of abstention in both supervised and self-supervised settings. Our results provide a first formal abstention-based gap, and a first provable optimization for the induced trade-off in an adversarial defense setting.

1 Introduction

A substantial body of work has shown that deep networks can be highly susceptible to adversarial attacks, in which minor changes to the input lead to incorrect, even bizarre classifications (Nguyen et al., 2015; Moosavi-Dezfooli et al., 2016; Su et al., 2019; Brendel et al., 2018; Shamir et al., 2019). Much of this work has considered ℓ_p-norm adversarial examples, but there has also been recent interest in exploring adversarial models beyond bounded ℓ_p-norm (Brown et al., 2018; Engstrom et al., 2017; Gilmer et al., 2018; Xiao et al., 2018; Alaifari et al., 2019). What these results have in common is that changes that either are imperceptible or should be irrelevant to the classification task can lead to drastically different network behavior.

One reason for this vulnerability to adversarial attack is the non-Lipschitzness property of typical neural networks: small but adversarial movements in the input space can often produce large perturbations in the feature space. In this work, we consider the question of whether non-Lipschitz networks are intrinsically vulnerable, or if they could still be made robust to adversarial attack, in an abstract but (we believe) instructive adversarial model. In particular, suppose an adversary, by making an imperceptible change to an input x, can cause its representation $F(x)$ in feature space (the penultimate layer of the network) to move by an arbitrary amount: will such an adversary always win? Clearly if the adversary can modify $F(x)$ by an arbitrary amount in an arbitrary direction, then yes. But what if the adversary can modify $F(x)$ by an arbitrary amount but only in a *random* direction (which it cannot control)? In this case, we show an interesting dichotomy: if the classifier must output a classification on any input it is given, then yes the adversary will still win, no matter how well-separated the classes are in feature space and no matter what decision surface the classifier uses. However, if the classifier is allowed to abstain, then it can defeat such an adversary so long as natural data of different classes are reasonably well-separated in feature space. Our results hold for generalizations of these models as well, such as adversaries that can modify feature representations in random low-dimensional subspaces, or directions that are not completely random. More broadly, our results provide a theoretical explanation for the importance of allowing abstaining, or selective classification, in the presence of adversarial attack.

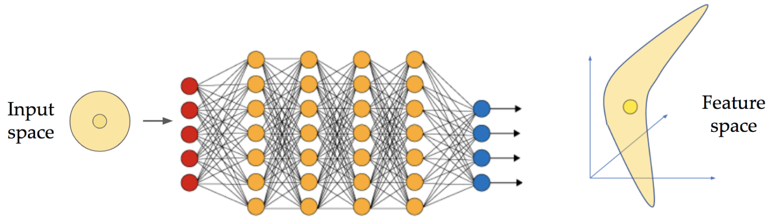

Figure 1: Illustration of a non-Lipschitz feature mapping using a deep network.

Apart from providing a useful abstraction for non-Lipschitz feature embeddings, our model may be viewed as capturing an interesting class of real attacks. There are various global properties of an image, such as brightness, contrast, or rotation angle whose change might be "perceptible but not relevant" to classification tasks. Our model could also be viewed as an abstraction of attacks of that nature. Feature space attacks of other forms, where one can perturb abstract features denoting styles, including interpretable styles such as vivid colors and sharp outlines and uninterpretable ones, have also been empirically studied in (Xu et al., 2020; Ganeshan & Babu, 2019).

An interesting property of our model is that it is critical to be able to refuse to predict: any algorithm which always predicts a class label—therefore without an ability to abstain—is guaranteed to perform poorly. This provides a first formal hardness result about abstention in adversarial defense, and also a first provable negative result in feature-space attacks. We therefore allow the algorithm to output "don't know" for some examples, which, as a by-product of our algorithm, serves as a detection mechanism for adversarial examples. It also results in an interesting trade-off between robustness and accuracy: by controlling how frequently we refuse to predict, we are able to trade (robust) precision off against recall. We also provide results for how to provably optimize for such a trade-off using a data-driven algorithm. Our strong theoretical advances are backed by empirical evidence in the context of contrastive learning (He et al., 2020; Chen et al., 2020; Khosla et al., 2020).

1.1 Our contributions

Our work tackles the problem of defending against adversarial perturbations in a *random feature subspace*, and advances the theory and practice of robust machine learning in multiple ways.

- We introduce a formal model that captures feature-space attacks and the effect of non-Lipschitzness of deep networks which can magnify input perturbations.
- We begin our analysis with a hardness result concerning defending against adversary without the option of "don't know". We show that *all* classifiers that partition the feature space into two or more classes—thus without an ability to abstain—are provably vulnerable to adversarial examples for at least one class of examples with nearly half probability.
- We explore the power of abstention option: a variant of nearest-neighbor classifier with the ability to abstain is provably robust against adversarial attacks, even in the presence of outliers in the training data set. We characterize the conditions under which the algorithm does not output "don't know" too often.
- We leverage and extend dispersion techniques from data-driven decision making, and present a novel data-driven method for learning data-specific optimal hyperparameters in our defense algorithms to simultaneously obtain high robust accuracy and low abstention rates. Unlike typical hyperparameter tuning, our approach provably converges to a global optimum.
- Experimentally, we show that our proposed algorithm achieves *certified* adversarial robustness on representations learned by supervised and self-supervised contrastive learning. Our method significantly outperforms algorithms without the ability to abstain.

2 Related Work

Adversarial robustness with abstention options. Classification with abstention option (a.k.a. selective classification (Geifman & El-Yaniv, 2017)) is a relatively less explored direction in the adversarial machine learning. Hosseini et al. (2017) augmented the output class set with a NULL label and trained the classifier to reject the adversarial examples by classifying them as NULL; Stutz et al. (2020) and Laidlaw & Feizi (2019) obtained robustness by rejecting low-confidence adversarial examples according to confidence thresholding or predictions on the perturbations of adversarial examples. Another related line of research to our method is the detection of adversarial examples (Grosse et al., 2017; Li & Li, 2017; Carlini & Wagner, 2017; Ma et al., 2018; Meng & Chen, 2017; Metzen et al., 2017; Bhagoji et al., 2018; Xu et al., 2017; Hu et al., 2019). However, theoretical understanding behind the empirical success of adversarial defenses with an abstention option remains elusive.

Data-driven decision making. Data-driven algorithm selection refers to choosing a good algorithm from a parameterized family of algorithms for given data. It is known as "hyperparameter tuning" to machine learning practitioners and typically involves a "grid search", "random search" (Bergstra & Bengio (2012)) or gradient-based search, with no guarantees of convergence to a global optimum. It was formally introduced to the theory of computing community by Gupta & Roughgarden (2017) as a learning paradigm, and was further extended in (Balcan et al., 2017). The key idea is to model the problem of identifying a good algorithm from data as a statistical learning problem. The technique has found useful application in providing provably better algorithms for several domains including clustering, mechanism design, and mixed integer programs, and providing guarantees like differential privacy and adaptive online learning (Balcan et al., 2018a;b; 2020). For learning in an adversarial setting, we provide the first demonstration of the effectiveness of data-driven algorithm selection in a defense method to optimize over the accuracy-abstention trade-off with strong theoretical guarantees.

3 Preliminaries

Notation. We will use *bold lower-case* letters such as $\boldsymbol{x}$ and $\boldsymbol{y}$ to represent vectors, *lower-case* letters such as x and y to represent scalars, and *calligraphy capital* letters such as $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{D}$ to represent distributions. Specifically, we denote by $\boldsymbol{x} \in \mathcal{X}$ the sample instance, and by $y \in \mathcal{Y}$ the label, where $\mathcal{X} \subseteq \mathbb{R}^{n_1}$ and $\mathcal{Y}$ indicate the image and label spaces, respectively. Denote by $F : \mathcal{X} \to \mathbb{R}^{n_2}$ the *feature embedding* which maps an instance to a high-dimensional vector in the latent space $F(\mathcal{X})$. It can be parameterized, e.g., by deep neural networks. We will frequently use $\mathbf{v} \in \mathbb{R}^{n_2}$ to represent an adversarial perturbation in the feature space. Denote by $\mathsf{dist}(\cdot, \cdot)$ the distance between any two vectors in the image or feature space. Examples of distances include $\mathsf{dist}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|$—the one induced by vector norm. We use $\mathbb{B}(\boldsymbol{x}, \tau)$ to represent a neighborhood of $\boldsymbol{x}$: $\{\boldsymbol{x}' : \mathsf{dist}(\boldsymbol{x}, \boldsymbol{x}') \leq \tau\}$ in the image or feature space. We will frequently denote by $\mathcal{D}_{\mathcal{X}}$ the distribution of instances in the input space, by $\mathcal{D}_{\mathcal{X}|y}$ the distribution of instances in the input space conditioned on the class y, by $\mathcal{D}_{F(\mathcal{X})}$ the distribution of features, and by $\mathcal{D}_{F(\mathcal{X})|y}$ the distribution of features conditioned on the class y.

3.1 Random Feature Subspace Threat model

In principle, the adversarial example for a given labeled data $(\boldsymbol{x}, y)$ is a data point $\boldsymbol{x}'$ that causes a classifier to output a different label on $\boldsymbol{x}'$ than the true label y. Probably one of the most popular adversarial examples is the norm-bounded perturbation in the input space. Despite a large literature devoted to defending against norm-bounded adversary by improving the Lipschitzness of neural network as a function mapping from input space to feature space (Zhang et al., 2019; Yang et al., 2020), it is typically not true that small perturbation in the input space necessarily implies small modification in the feature space. In this paper, we study a threat model where an adversary can modify the data by a large amount in the feature space. Note that because this large modification in feature space is assumed to come from a small perturbation in input space, we always assume that the *true correct label y is the same for x' as for x*. Our model highlights the power of abstention in the adversarial learning: there is a provable separation when we have and do not have an abstention option under our threat model.

Our threat model. In the setting of (robust) representation learning, we are given a set of training instances $\boldsymbol{x}_1, ..., \boldsymbol{x}_m \in \mathcal{X}$. Let $\boldsymbol{x}$ be an n_1-dimensional test input for classification. The input is embedded into a high n_2-dimensional feature space using a deep neural network F. We predict the class of $\boldsymbol{x}$ by a prediction function on $F(\boldsymbol{x})$ which can potentially output "don't know". The adversary may corrupt $F(\boldsymbol{x})$ such that the modified feature vector is restricted in a random n_3-dimensional affine subspace denoted by $\mathcal{S} + \{F(\boldsymbol{x})\}$, while the perturbation magnitude might be arbitrarily large. The adversary is given access to everything including F, $\boldsymbol{x}$, $\mathcal{S}$ and the true label of $\boldsymbol{x}$. Throughout the paper, we will refer *adversary* and *adversarial example* to this threat model.

Algorithm 1 ROBUSTCLASSIFIER(τ, σ)

```
Input: A test feature $F(\boldsymbol{x})$ (potentially an adversarial example), a set of training features $F(\boldsymbol{x}_i)$ and their labels $y_i$, $i \in [m]$, a threshold parameter $\tau$, a separation parameter $\sigma$.
Preprocessing: Delete training examples $F(\boldsymbol{x}_i)$ if $\min_{j\in[m], y_i \neq y_j} \mathsf{dist}(F(\boldsymbol{x}_i), F(\boldsymbol{x}_j)) < \sigma$
Output: A predicted label of $F(\boldsymbol{x})$, or "don't know".
if $\min_{i\in[m]} \mathsf{dist}(F(\boldsymbol{x}), F(\boldsymbol{x}_i)) < \tau$ then
    Return $y_{\arg\min_{i\in[m]} \mathsf{dist}(F(\boldsymbol{x}), F(\boldsymbol{x}_i))}$
else
    Return "don't know"
```

3.2 A META-ALGORITHM FOR INFERENCE-TIME ROBUSTNESS

Given a test data $\boldsymbol{x}$, let r denote the shortest distance between $F(\boldsymbol{x})$ and any training embedding $F(\boldsymbol{x}_i)$ of different labels. Throughout the paper, we consider the prediction rule that we classify an unseen (and potentially adversarially modified) example with the class of its nearest training example provided that the distance between them is at most τ; otherwise the algorithm outputs "don't know" (see Algorithm 1 and Figure 2). The adversary is able to corrupt $F(\boldsymbol{x})$ by a carefully-crafted perturbation along a random direction, i.e., $F(\boldsymbol{x}) + \mathbf{v}$, where $\mathbf{v}$ is an adversarial vector of arbitrary length in a random n_3-dimensional subspace of $\mathbb{R}^{n_2}$. The parameter τ trades the success rate off against the abstention rate; when $\tau \to \infty$, our algorithm is equivalent to the nearest-neighbor algorithm. We also preprocess to remove outliers and points too close to them.

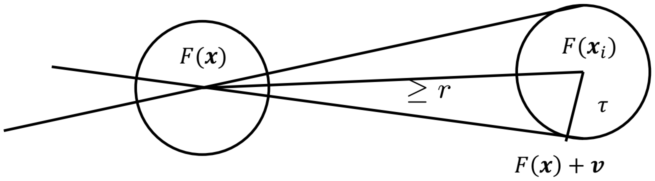

Figure 2: Adversarial misclassification for nearest-neighbor predictor.

4 NEGATIVE RESULTS WITHOUT AN ABILITY TO ABSTAIN

Several negative results are known for defending against adversarial examples beyond norm-bounded settings. For example, Shamir et al. (2019) provably show existence of targeted adversarial examples with small hamming distance in the input space to their clean examples. For feature-space attacks, several empirical negative results are known (Xu et al., 2020; Ganeshan & Babu, 2019). We present a hardness result concerning defenses without an ability to abstain, and prove that such defenses are inevitably doomed against our feature-space attacks.

Theorem 4.1. *For any classifier that partitions $\mathbb{R}^{n_2}$ into two or more classes, any data distribution $\mathcal{D}$, any $\delta > 0$ and any feature embedding F, there must exist at least one class y^*, such that for at least a $1-\delta$ probability mass of examples $\boldsymbol{x}$ from class y^* (i.e., $\boldsymbol{x}$ is drawn from $\mathcal{D}_{\mathcal{X}|y^*}$), for a random unit-length vector $\mathbf{v}$, with probability at least $1/2-\delta$ for some $\delta_0 > 0$, $F(\boldsymbol{x}) + \delta_0\mathbf{v}$ is not labeled y^* by the classifier. In other words, there must be at least one class y^* such that for at least $1-\delta$ probability mass of points $\boldsymbol{x}$ of class y^*, the adversary wins with probability at least $1/2-\delta$.*

Proof. Without loss of generality, we assume that the feature embedding F is an identity mapping. Define r_δ to be a radius such that for every class y, at least a $1-\delta$ probability mass of examples $\boldsymbol{x}$ of class y lie within distance r_δ of the origin. Let $R = r_\delta\sqrt{n_2}/\delta$. R is defined to be large enough such that if we take a ball of radius R and move it by a distance r_δ, at least a $1-\delta$ fraction of the volume of the new ball is inside the intersection with the old ball. Now, let $\mathcal{B}$ be the ball of radius R centered at the origin. Let $\mathsf{vol}(\mathcal{B})$ denote the volume of $\mathcal{B}$ and let $\mathsf{vol}_y(\mathcal{B})$ denote the volume of the subset of $\mathcal{B}$ that is assigned label y by the classifier. Let y^* be any label such that $\mathsf{vol}_{y^*}(\mathcal{B})/\mathsf{vol}(\mathcal{B}) \leq 1/2$. Such a class y^* exists because we do not have the option to output "don't know". Now by the definition of y^*, a point $\mathbf{z}$ picked uniformly at random from $\mathcal{B}$ has probability at least $1/2$ of being classified differently from y^*. This implies that, by the definition of R, if $\boldsymbol{x}$ is within distance r_δ of the origin, then a point $\mathbf{z}_x$ that is picked uniformly at random in the ball $\mathcal{B}_x$ of radius R centered at $\boldsymbol{x}$ has probability at least $1/2 - \delta$ of being classified differently from y^*. This immediately implies that if we choose a random unit-length vector $\mathbf{v}$, then with probability at least $1/2 - \delta$, there exists $\delta_0 > 0$ such that $\boldsymbol{x} + \delta_0 \mathbf{v}$ is classified differently from y^*, since we can think of choosing $\mathbf{v}$ by first sampling $\mathbf{z}_x$ from $\mathcal{B}_x$ and then defining $\mathbf{v} = (\mathbf{z}_x - \boldsymbol{x})/\|\mathbf{z}_x - \boldsymbol{x}\|_2$. So, the theorem follows from the fact that, by the definition of r_δ, at least $1-\delta$ probability mass of examples $\boldsymbol{x}$ from class y^* are within distance r_δ of the origin. □

We remark that our lower bound applies to any classifier and exploits the fact that a classifier without abstention must label the entire feature space. For a simple linear decision boundary (center of Figure 3), a perturbation in any direction (except parallel to the boundary) can cross the boundary with an appropriate magnitude. The left and right figures show that if we try to 'bend' the decision boundary to 'protect' one of the classes, the other class is still vulnerable. Our argument formalizes and generalizes this intuition, and shows that there must be at least one vulnerable class irrespective of how you may try to shape the class boundaries, where the adversary succeeds in a large fraction of directions.

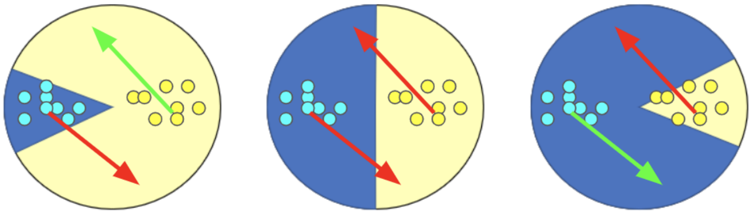

Figure 3: A simple example to illustrate Theorem 4.1.

Theorem 4.1 implies that *all* classifiers that partitions $\mathbb{R}^{n_2}$ into two or more classes—thus without an ability to abstain—are vulnerable to adversarial examples for at least one class of data with nearly half probability. Despite much effort has been devoted to empirically investigating the power of "don't know" in the adversarial robustness, theoretical understanding behind the empirical success of these methods remains elusive. To the best of our knowledge, our work is the first result that provably demonstrates the power of "don't know" in the algorithmic design of adversarially robust classifiers.

5 Positive Results with An Ability to Abstain

Theorem 4.1 gives a hardness result of robust classification without abtention. In this section, we explore the power of abstaining and show classifiers with an ability to abstain are provably robust.

Given a test instance $\boldsymbol{x} \sim \mathcal{D}_\mathcal{X}$, recall that r denotes the shortest distance between $F(\boldsymbol{x}) \in \mathbb{R}^{n_2}$ and any training embedding $F(\boldsymbol{x}_i) \in \mathbb{R}^{n_2}$ with a different label. The adversary is allowed to corrupt $F(\boldsymbol{x})$ with an arbitrarily large perturbation in a uniform-distributed subspace S of dimension n_3. Consider the prediction rule that we classify the unseen example $F(\boldsymbol{x}) \in \mathbb{R}^{n_2}$ with the class of its nearest training example provided that the distance between them is at most τ; otherwise the algorithm outputs "don't know" (see Algorithm 1 when $\sigma = 0$). Denote by $\mathcal{E}_{\text{adv}}^{\boldsymbol{x}}(f) := \mathbb{E}_{S\sim\mathcal{S}} \mathbf{1}\{\exists \mathbf{e} \in S + F(\boldsymbol{x}) \subseteq \mathbb{R}^{n_2}$ s.t. $f(\mathbf{e}) \neq \boldsymbol{y}$ and $f(\mathbf{e})$ does not abstain$\}$ the robust error of a given classifier f for classifying instance $\boldsymbol{x}$. Our analysis leads to the following positive results on this algorithm.

Theorem 5.1. *Let $\boldsymbol{x} \sim \mathcal{D}_\mathcal{X}$ be a test instance, m be the number of training examples and r be the shortest distance between $F(\boldsymbol{x})$ and $F(\boldsymbol{x}_i)$ where $\boldsymbol{x}_i$ is a training point from a different class. Suppose $\tau = o\left(r\sqrt{1-\frac{n_3}{n_2}}\right)$. The robust error of Algorithm 1, $\mathcal{E}_{\text{adv}}^{\boldsymbol{x}}(\textsc{RobustClassifier}(\tau, 0))$, is at most $m\left(\frac{c\tau}{r\sqrt{1-\frac{n_3}{n_2}}}\right)^{n_2-n_3} + mc_0^{n_2-n_3}$, where $c > 0$ and $0 < c_0 < 1$ are absolute constants.*

Proof Sketch. We begin our analysis with the case of $n_3 = 1$. Suppose we have a training example $\boldsymbol{x}'$ of another class, and suppose $F(\boldsymbol{x})$ and $F(\boldsymbol{x}')$ are at distance D in the feature space. Because $\tau = o(D)$, the probability that the adversary can move $F(\boldsymbol{x})$ to within distance τ of $F(\boldsymbol{x}')$ should be roughly the ratio of the surface area of a sphere of radius τ to the surface area of a sphere of radius D, which is at most $\left(\mathcal{O}\left(\frac{\tau}{D}\right)\right)^{n_2-1} \leq \left(\mathcal{O}\left(\frac{\tau}{r}\right)\right)^{n_2-1}$. The analysis for the general case of n_3 follows from a pealing argument: note that the random subspace in which the adversary vector is restricted to lie can be constructed by first sampling a vector $\mathbf{v}_1$ uniformly at random from a unit sphere in the ambient space $\mathbb{R}^{n_2}$ centered at 0; fixing $\mathbf{v}_1$, we then sample a vector $\mathbf{v}_2$ uniformly at random from a unit sphere in the null space of $\mathsf{span}\{\mathbf{v}_1\}$; we repeat this procedure n_3 times and let $\mathsf{span}\{\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_{n_3}\}$ be the desired adversarial subspace. For each step of construction, we apply the same argument as that of $n_3 = 1$ with $D = \Omega\left(r\sqrt{\frac{n_2-i}{n_2}}\right)$ by a high probability, if we project $F(\boldsymbol{x})$ and $F(\boldsymbol{x}')$ to a random subspace of dimension $n_2 - i$. Finally, a union bound over m training points completes the proof. □

Trade-off between success probability and abstention rate. Theorem 5.1 captures the trade-off between the success probability of an algorithm and the abstention rate: a smaller value of τ increases the success probability of the algorithm, while it also encourages Algorithm 1 to output "don't know" more often. A related line of research to this observation is the trade-off between robustness and accuracy: Zhang et al. (2019); Tsipras et al. (2019) showed that there might be no predictor in the hypothesis class that has low natural and robust errors; even such a predictor exists for the well-separated data (Yang et al., 2020), Raghunathan et al. (2020) showed that the natural error could increase by adversarial training if we only have finite number of data. To connect the two trade-offs, we note that a high success probability of ROBUSTCLASSIFIER$(\tau, 0)$ in Algorithm 1 tends to avoid the algorithm from predicting wrong labels for adversarial examples, while the associated high abstention rate encourages the algorithm to output "don't know" even for natural examples, thus leading to a trivial non-accurate classifier.

5.1 A MORE GENERAL ADVERSARY WITH BOUNDED DENSITY

We extend our results to a more general class of adversaries, which have a bounded distribution over the space of linear subspaces of a fixed dimension n_3 and the adversary can perturb a test feature vector arbitrarily in the sampled adversarial subspace.

Theorem 5.2. *Consider the setting of Theorem 5.1, with an adversary having a κ-bounded distribution over the space of linear subspaces of a fixed dimension n_3 for perturbing the test point. If $\mathbf{E}(\tau, r)$ denotes the bound on error rate in Theorem 5.1 for* ROBUSTCLASSIFIER$(\tau, 0)$ *in Algorithm 1, then the error bound of the same algorithm against the κ-bounded adversary is $\mathcal{O}(\kappa\mathbf{E}(\tau, r))$.*

5.2 OUTLIER REMOVAL AND IMPROVED UPPER BOUND

The upper bounds above assume that the data is well-separated in the feature space. For noisy data and good-but-not-perfect embeddings, the condition may not hold. In Theorem E.1 (in Appendix E) we show that we obtain almost the same upper bound on failure probability under weaker assumptions by exploiting the noise removal threshold σ.

5.3 CONTROLLING ABSTENTION RATE ON NATURAL DATA

We show that we can control the frequency of outputting "don't know", when the data are nicely distributed according to the following generative assumption. Intuitively, it says that for every label class one can cover most of the distribution of the class with (potentially overlapping) balls of a fixed radius, each having a small lower bound on the density contained. This holds for well-clustered datasets (as is typical for feature data) for a sufficiently large radius.

Assumption 1. *We assume that at least $1 - \delta$ fraction of mass of the marginal distribution $\mathcal{D}_{F(\mathcal{X})|y}$ over $\mathbb{R}^{n_2}$ can be covered by N balls $\mathbb{B}_1$, $\mathbb{B}_2$, ... $\mathbb{B}_N$ of radius $\tau/2$ and of mass $\Pr_{\mathcal{D}_{F(\mathcal{X})}}[\mathbb{B}_k] \geq \frac{C_0}{m}\left(n_2 \log m + \log \frac{4N}{\beta}\right)$, where $C_0 > 0$ is an absolute constant and $\delta, \beta \in (0, 1)$.*

Our analysis leads to the following guarantee on the abstention rate.

Theorem 5.3. *Suppose that $F(\boldsymbol{x}_1), ..., F(\boldsymbol{x}_m)$ are m training instances i.i.d. sampled from marginal distribution $\mathcal{D}_{F(\mathcal{X})}$. Under Assumption 1, with probability at least $1 - \beta/4$ over the sampling, we have $\Pr(\cup_{i=1}^m \mathbb{B}(F(\boldsymbol{x}_i), \tau)) \geq 1 - \delta$.*

Theorem 5.3 implies that when $\Pr[\mathbb{B}_k] \geq \frac{\beta}{N}$ and $m = \Omega(\frac{n_2 N}{\beta} \log \frac{n_2 N}{\beta})$, with probability at least $1 - \beta/4$ over the sampling, we have $\Pr(\cup_{i=1}^m \mathbb{B}(F(\boldsymbol{x}_i), \tau)) \geq 1 - \delta$. Therefore, with high probability, the algorithm will output "don't know" only for an δ fraction of natural data.

6 Learning Data-Specific Optimal Thresholds

Given an embedding function F and a classifier f_τ which outputs either a predicted class if the nearest neighbor is within distance τ of a test point or abstains from predicting, we want to evaluate the performance of f_τ on a test set $\mathcal{T}$ against an adversary which can perturb a test feature vector in a random subspace $S \sim \mathcal{S}$. To this end, we define $\mathcal{E}_{\text{adv}}(\tau) := \mathbb{E}_{S\sim\mathcal{S}} \frac{1}{|\mathcal{T}|} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{T}} \mathbf{1}\{\exists \mathbf{e} \in S + F(\boldsymbol{x}) \subseteq \mathbb{R}^{n_2}$ s.t. $f(\mathbf{e}) \neq \boldsymbol{y}$ and $f_\tau(\mathbf{e})$ does not abstain$\}$ as the robust error on the test set $\mathcal{T}$, and $\mathcal{D}_{\text{nat}}(\tau) := \frac{1}{|\mathcal{T}|} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{T}} \mathbf{1}\{f_\tau(F(\boldsymbol{x}))$ abstains$\}$ as the abstention rate on the natural data. $\mathcal{E}_{\text{adv}}(\tau)$ and $\mathcal{D}_{\text{nat}}(\tau)$ are monotonic in τ. The robust error $\mathcal{E}_{\text{adv}}(\tau)$ is optimal at $\tau = 0$, while we abstain from prediction all the time (i.e., $\mathcal{D}_{\text{nat}}(\tau) = 1$). A simple approach is to fix an upper limit d^* on $\mathcal{D}_{\text{nat}}(\tau)$, which corresponds to the maximum abstention rate on natural data under our budget. Then it is straightforward to search for the optimal τ^* such that $\mathcal{D}_{\text{nat}}(\tau^*) \approx d^*$ by using nearest neighbor distances of test points. For $\tau < \tau^*$ we have a higher abstention rate, and when $\tau > \tau^*$ we have a higher robust error rate. A potential problem with this approach is that $\mathcal{D}_{\text{nat}}(\tau)$ is non-Lipschitz, so small variation in τ can possibly make the abstention rate significantly higher than d^*.

An alternative objective which captures the trade-off between abstention rate and accuracy is defined as $g(\tau) := \mathcal{E}_{\text{adv}}(\tau) + c\mathcal{D}_{\text{nat}}(\tau)$, where c is a positive constant. If, for example, we are willing to take a one percent increase of the abstention rate for a two percent drop in the error rate, we could set c to be $\frac{1}{2}$. We can optimize $g(\tau)$ in a data-driven fashion and obtain theoretical guarantee on the convergence to a global optimum. In the following, we consider the case where the test examples appear in an online fashion in small batches of size b, and we set the threshold τ adaptively by a low-regret algorithm. We note in Corollary 6.3, using online-to-batch conversion, that our results imply a uniform convergence bound for objective $g(\tau)$ in the supervised setting. Details of proofs in this section can be found in Appendix H.

The significance of data-driven design in this setting is underlined by the following two observations. Firstly, as noted above, optimization for τ is difficult due to the non-Lipschitzness nature of $\mathcal{D}_{\text{nat}}(\tau)$ and the intractability of characterizing the objective function $g(\tau)$ exactly due to $\mathcal{E}_{\text{adv}}(\tau)$. Secondly, the optimal value of τ can be a complex function of the data geometry and sampling rate. We illustrate this by exact computation of optimal τ for a simple intuitive setting: consider a binary classification problem where the features lie uniformly on two one-dimensional manifolds embedded in two-dimensions (i.e., $n_2 = 2$, see Figure 4). Assume that the adversary perturbs in a uniformly random direction ($n_3 = 1$). For this setting, in Appendix J we show that

Theorem 6.1. *Let $\tau^* := \arg\max_{\tau\in\mathbb{R}^+} g(\tau)$ and $\beta = \frac{2\pi c r}{D}$. For the setting considered above, if we further assume $D = o(r)$ and $m = \omega(\log\beta)$, then there is a unique value of τ^* in $[0, D/2)$. Furthermore, we have $\tau^* = \Theta\left(\frac{D\log(\beta m)}{m}\right)$ if $m > \frac{1}{\beta}$; otherwise, $\tau^* = 0$.*

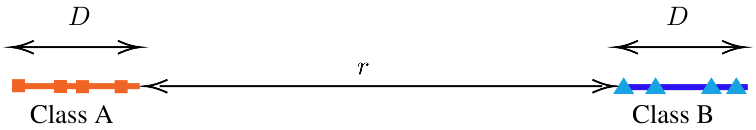

Figure 4: A simple intuitive example where we compute the optimal value of the abstention threshold exactly. Classes A and B are both distributed uniformly on one-dimensional segments of length D, embedded collinear and at distance r in $\mathbb{R}^2$.

The remaining section summarizes our main theoretical results.

Theorem 6.2. *Assume τ is $o\left(\min\{m^{-1/n_2}, r\}\right)$, and the data distribution is continuous, κ-bounded, positive and has bounded partial derivatives. If τ is set using a continuous version of the multiplicative updates algorithm (Algorithm 2 in Appendix H, Balcan et al. (2018a)), then with probability at least $1-\delta$, the expected regret in T rounds is bounded by $O\left(\sqrt{n_2 T \log\left(\frac{\kappa R T m b}{\delta r^{n_2 - n_3}}\right)}\right)$, where R is a bound* *on the largest distance between any two training points, b is the batch size, and r is the smallest distance between points of different labels.*

Corollary 6.3. *Suppose we run the online algorithm of Theorem 6.2 on a validation set of size T, and use a randomized threshold $\hat{\tau}$ on the test set drawn from a uniform distribution over the thresholds $\tau_1, \ldots, \tau_T$ used in online learning. If the threshold which maximizes $g(\tau)$ is τ^*, then with probability greater than $1-\delta$, we have $|\mathbb{E}[g(\hat{\tau})] - g(\tau^*)| \le O\left(\sqrt{\frac{n_2}{T}\log\left(\frac{\kappa RTmb}{\delta r^{n_2-n_3}}\right)}\right)$.*

Remark 1. *The results can be generalized to a bounded density adversary (Corollary H.3).*

Remark 2. *The above analysis can be extended to the problem of optimizing over σ by formulating the objective as function of two parameters, $g(\tau,\sigma) := \mathcal{E}_{\mathrm{adv}}(\tau,\sigma) + c\mathcal{D}_{\mathrm{nat}}(\tau,\sigma)$ within a range $\sigma \in [r,s]$. For fixed τ, both $\mathcal{E}_{\mathrm{adv}}(\tau,\sigma)$ and $\mathcal{D}_{\mathrm{nat}}(\tau,\sigma)$ are piece-wise constant and monotonic. The proof of Lipschitzness of the pieces can be adapted easily to the case of $\sigma \ge r$ (Lemma H.2). Discontinuities in $\mathcal{E}_{\mathrm{adv}}(\tau,\cdot)$ and $\mathcal{D}_{\mathrm{nat}}(\tau,\cdot)$ can be bounded using the upper bound s for σ (Lemma H.4). Finally, the number of discontinuities in $g(\tau,\sigma)$ in a ball of radius w can be upper bounded by a product of the number of discontinuities in $g(\tau,\cdot)$ and $g(\cdot,\sigma)$ in intervals of width w.*

7 Experiments on Contrastive Learning

Theorem 5.1 sheds light on algorithmic designs of robust learning of feature embedding F. In order to preserve robustness against adversarial examples regarding a given test point $\boldsymbol{x}$, in the feature space the theorem suggests minimizing τ—the closest distance between $F(\boldsymbol{x})$ and any training feature $F(\boldsymbol{x}_i)$ of the same label, and maximizing r—the closest distance between $F(\boldsymbol{x})$ and any training feature $F(\boldsymbol{x}_i)$ of different labels. This is conceptually consistent with the spirit of the nearest-neighbor algorithm, a.k.a. contrastive learning when we replace the *max* operator with the *softmax* operator for differentiable training:

$$\min_F -\frac{1}{m}\sum_{i\in[m]}\log\left(\frac{\sum_{j\in[m],j\neq i,y_i=y_j} e^{-\frac{\|F(\boldsymbol{x}_i)-F(\boldsymbol{x}_j)\|^2}{T}}}{\sum_{k\in[m],k\neq i} e^{-\frac{\|F(\boldsymbol{x}_i)-F(\boldsymbol{x}_k)\|^2}{T}}}\right), \tag{1}$$

where $T > 0$ is the temperature parameter. Loss (1) is also known as the soft-nearest-neighbor loss in the context of supervised learning (Frosst et al., 2019), or the InfoNCE loss in the setting of self-supervised learning (He et al., 2020).

7.1 Certified adversarial robustness against exact computation of attacks

We verify the robustness of Algorithm 1 when the representations are learned by contrastive learning. Given a embedding function F and a classifier f which outputs either a predicted class or abstains from predicting, recall that we define the natural and robust errors, respectively, as $\mathcal{E}_{\mathrm{nat}}(f) := \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\mathbf{1}\{f(F(\boldsymbol{x})) \neq \boldsymbol{y}$ and $f(F(\boldsymbol{x}))$ does not abstain$\}$, and $\mathcal{E}_{\mathrm{adv}}(f) := \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D},S\sim\mathcal{S}}\mathbf{1}\{\exists \mathbf{e} \in S + F(\boldsymbol{x}) \subseteq \mathbb{R}^{n_2}$ s.t. $f(\mathbf{e}) \neq \boldsymbol{y}$ and $f(\mathbf{e})$ does not abstain$\}$, where $S \sim \mathcal{S}$ is a random adversarial subspace of $\mathbb{R}^{n_2}$ with dimension n_3. $\mathcal{D}_{\mathrm{nat}}(f) := \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}\mathbf{1}\{f(F(\boldsymbol{x}))$ abstains$\}$ is the abstention rate on the natural examples. Note that the robust error is always at least as large as the natural error.

Self-supervised contrastive learning setup. Our experimental setup follows that of SimCLR (Chen et al., 2020). We use the ResNet-18 architecture (He et al., 2016) for representation learning with a two-layer projection head of width 128. The dimension of the representations is 512. We set batch size 512, temperature $T = 0.5$, and initial learning rate 0.5 which is followed by cosine learning rate decay. We sequentially apply four simple augmentations: random cropping followed by resize back to the original size, random flipping, random color distortions, and randomly converting image to grayscale with a probability of 0.2. In the linear evaluation protocol, we set batch size 512 and learning rate 1.0 to learn a linear classifier in the feature space by empirical risk minimization.

Supervised contrastive learning setup. Our experimental setup follows that of Khosla et al. (2020). We use the ResNet-18 architecture for representation learning with a two-layer projection head of width 128. The dimension of the representations is 512. We set batch size 512, temperature $T = 0.1$, and initial learning rate 0.5 which is followed by cosine learning rate decay. We sequentially apply four simple augmentations: random cropping followed by resize back to the original size, random

Table 1: Natural error $\mathcal{E}_{\text{nat}}$ and robust error $\mathcal{E}_{\text{adv}}$ on the CIFAR-10 dataset when $n_3 = 1$ and the 512-dimensional representations are learned by contrastive learning, where $\mathcal{D}_{\text{nat}}$ represents the fraction of each algorithm's output of "don't know" on the natural data. We report values for $\sigma \approx \tau$ as they tend to give a good abstention-error trade-off w.r.t. σ.

	Contrastive	Linear Protocol		Ours ($\tau = 3.0$)			Ours ($\tau = 2.0$)		
		$\mathcal{E}_{\text{nat}}$	$\mathcal{E}_{\text{adv}}$	$\mathcal{E}_{\text{nat}}$	$\mathcal{E}_{\text{adv}}$	$\mathcal{D}_{\text{nat}}$	$\mathcal{E}_{\text{nat}}$	$\mathcal{E}_{\text{adv}}$	$\mathcal{D}_{\text{nat}}$
($\sigma = 0$)	Self-supervised	8.9%	100.0%	15.4%	40.7%	2.2%	14.3%	26.2%	28.7%
	Supervised	5.6%	100.0%	5.7%	60.5%	0.0%	5.7%	33.4%	0.0%
($\sigma = 0.9\tau$)	Self-supervised	8.9%	100.0%	7.2%	9.4%	12.9%	10.0%	17.7%	29.9%
	Supervised	5.6%	100.0%	6.2%	18.9%	0.0%	5.6%	22.0%	0.1%
($\sigma = \tau$)	Self-supervised	8.9%	100.0%	1.1%	1.2%	33.4%	2.1%	3.1%	49.9%
	Supervised	5.6%	100.0%	1.9%	2.8%	10.6%	4.1%	4.8%	3.3%

flipping, random color distortions, and randomly converting image to grayscale with a probability of 0.2. In the linear evaluation protocol, we set batch size 512 and learning rate 5.0 to learn a linear classifier in the feature space by empirical risk minimization.

In both self-supervised and supervised setups, we compare the robustness of the linear protocol with that of our defense protocol in Algorithm 1 under exact computation of adversarial examples using a convex optimization program in n_3 dimensions and m constraints. Algorithm 4 in the appendix provides an efficient implementation of the attack.

Experimental results. We summarize our results in Table 1. Comparing with a linear protocol, our algorithms have much lower robust error. Note that even if abstention is added based on distance from the linear boundary, sufficiently large perturbations will ensure the adversary can always succeed. For an approximate adversary which can be efficiently implemented for large n_3, see Appendix L.2.

7.2 Robustness-abstention trade-off

The threshold parameter τ captures the trade-off between the robust accuracy $\mathcal{A}_{\text{adv}} := 1 - \mathcal{E}_{\text{adv}}$ and the abstention rate $\mathcal{D}_{\text{nat}}$ on the natural data. We report both metrics for different values of τ for supervised and self-supervised constrastive learning. The supervised setting enjoys higher adversarial accuracy and a smaller abstention rate for fixed τ's due to the use of extra label information. We plot $\mathcal{A}_{\text{adv}}$ against $\mathcal{D}_{\text{nat}}$ for Algorithm 1 as hyperparameters vary. For small τ, both accuracy and abstention rate approach 1.0. As the threshold increases, the abstention rate decreases rapidly and our algorithm enjoys good accuracy even with small abstention rates. For $\tau \to \infty$ (i.e. the nearest neighbor search), the abstention rate on the natural data $\mathcal{D}_{\text{nat}}$ is 0% but the robust accuracy is also roughly 0%. Increasing σ (for small σ) gives us higher robust accuracy for the same abstention rate. Too large σ may also lead to degraded performance.

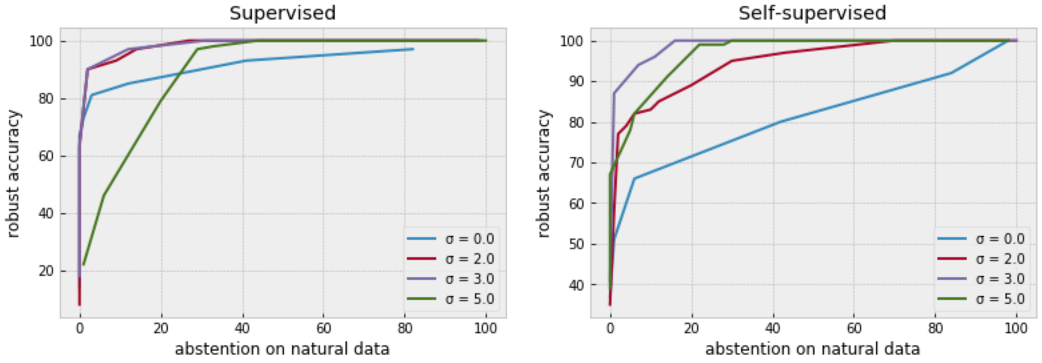

Figure 5: Adversarial accuracy (i.e., rate of adversary failure) vs. abstention rate as threshold τ varies for $n_3 = 1$ and different outlier removal thresholds σ.

REFERENCES

Rima Alaifari, Giovanni S Alberti, and Tandri Gauksson. ADef: an iterative algorithm to construct adversarial deformations. In *International Conference on Learning Representations*, 2019.

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*, pp. 15535–15545, 2019.

Maria-Florina Balcan, Vaishnavh Nagarajan, Ellen Vitercik, and Colin White. Learning-theoretic foundations of algorithm configuration for combinatorial partitioning problems. In *Annual Conference on Learning Theory*, pp. 213–274, 2017.

Maria-Florina Balcan, Travis Dick, and Ellen Vitercik. Dispersion for data-driven algorithm design, online learning, and private optimization. In *Annual Symposium on Foundations of Computer Science*, pp. 603–614. IEEE, 2018a.

Maria-Florina Balcan, Tuomas Sandholm, and Ellen Vitercik. A general theory of sample complexity for multi-item profit maximization. In *ACM Conference on Economics and Computation*, pp. 173–174, 2018b.

Maria-Florina Balcan, Travis Dick, and Dravyansh Sharma. Learning piecewise Lipschitz functions in changing environments. In *International Conference on Artificial Intelligence and Statistics*, pp. 3567–3577, 2020.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*, 13(1):281–305, 2012.

Arjun Nitin Bhagoji, Daniel Cullina, Chawin Sitawarin, and Prateek Mittal. Enhancing robustness of machine learning systems via data transformations. In *Annual Conference on Information Sciences and Systems*, pp. 1–5, 2018.

Avrim Blum, Travis Dick, Naren Manoj, and Hongyang Zhang. Random smoothing might be unable to certify ℓ_∞ robustness for high-dimensional images. *Journal of Machine Learning Research*, 2020.

Wieland Brendel, Jonas Rauber, Alexey Kurakin, Nicolas Papernot, Behar Veliqi, Marcel Salathé, Sharada P Mohanty, and Matthias Bethge. Adversarial vision challenge. *arXiv preprint arXiv:1808.01976*, 2018.

Tom B Brown, Nicholas Carlini, Chiyuan Zhang, Catherine Olsson, Paul Christiano, and Ian Goodfellow. Unrestricted adversarial examples. *arXiv preprint arXiv:1809.08352*, 2018.

Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *ACM Workshop on Artificial Intelligence and Security*, pp. 3–14, 2017.

Kamalika Chaudhuri and Sanjoy Dasgupta. Rates of convergence for the cluster tree. In *Advances in Neural Information Processing Systems*, pp. 343–351, 2010.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, 2020.

Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. A rotation and a translation suffice: Fooling CNNs with simple transformations. *arXiv preprint arXiv:1712.02779*, 2017.

Dafydd Evans, Antonia J Jones, and Wolfgang M Schmidt. Asymptotic moments of near–neighbour distance distributions. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 458(2028):2839–2849, 2002.

Nicholas Frosst, Nicolas Papernot, and Geoffrey Hinton. Analyzing and improving representations with the soft nearest neighbor loss. In *International Conference on Machine Learning*, 2019.

Aditya Ganeshan and R Venkatesh Babu. FDA: Feature disruptive attack. In *IEEE International Conference on Computer Vision*, pp. 8069–8079, 2019.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in neural information processing systems*, pp. 4878–4887, 2017.

Justin Gilmer, Ryan P Adams, Ian Goodfellow, David Andersen, and George E Dahl. Motivating the rules of the game for adversarial example research. *arXiv preprint arXiv:1807.06732*, 2018.

Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel. On the (statistical) detection of adversarial examples. *arXiv preprint arXiv:1702.06280*, 2017.

Rishi Gupta and Tim Roughgarden. A PAC approach to application-specific algorithm selection. *SIAM Journal on Computing*, 46(3):992–1017, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Olivier J Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*, 2019.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

Hossein Hosseini, Yize Chen, Sreeram Kannan, Baosen Zhang, and Radha Poovendran. Blocking transferability of adversarial examples in black-box learning systems. *arXiv preprint arXiv:1703.04318*, 2017.

Shengyuan Hu, Tao Yu, Chuan Guo, Wei-Lun Chao, and Kilian Q Weinberger. A new defense against adversarial images: Turning a weakness into a strength. In *Advances in Neural Information Processing Systems*, pp. 1635–1646, 2019.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020.

Cassidy Laidlaw and Soheil Feizi. Playing it safe: Adversarial robustness with an abstain option. *arXiv preprint arXiv:1911.11253*, 2019.

Xin Li and Fuxin Li. Adversarial examples detection in deep networks with convolutional filter statistics. In *IEEE International Conference on Computer Vision*, pp. 5764–5772, 2017.

Xingjun Ma, Bo Li, Yisen Wang, Sarah M Erfani, Sudanthi Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E Houle, and James Bailey. Characterizing adversarial subspaces using local intrinsic dimensionality. In *International Conference on Learning Representations*, 2018.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Dongyu Meng and Hao Chen. MagNet: a two-pronged defense against adversarial examples. In *ACM SIGSAC conference on computer and communications security*, pp. 135–147, 2017.

Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On detecting adversarial perturbations. In *International Conference on Learning Representations*, 2017.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2574–2582, 2016.

Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 427–436, 2015.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Aditi Raghunathan, Sang Michael Xie, Fanny Yang, John Duchi, and Percy Liang. Understanding and mitigating the tradeoff between robustness and accuracy. In *International Conference on Machine Learning*, 2020.

Adi Shamir, Itay Safran, Eyal Ronen, and Orr Dunkelman. A simple explanation for the existence of adversarial examples with small hamming distance. *arXiv preprint arXiv:1901.10861*, 2019.

David Stutz, Matthias Hein, and Bernt Schiele. Confidence-calibrated adversarial training: Generalizing to unseen attacks. In *International Conference on Machine Learning*, 2020.

Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European Conference on Computer Vision*, 2019.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019.

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3733–3742, 2018.

Chaowei Xiao, Jun-Yan Zhu, Bo Li, Warren He, Mingyan Liu, and Dawn Song. Spatially transformed adversarial examples. In *International Conference on Learning Representations*, 2018.

Qiuling Xu, Guanhong Tao, Siyuan Cheng, Lin Tan, and Xiangyu Zhang. Towards feature space adversarial attack. *arXiv preprint arXiv:2004.12385*, 2020.

Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. In *Network and Distributed Systems Security Symposium*, 2017.

Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Ruslan Salakhutdinov, and Kamalika Chaudhuri. Adversarial robustness through local Lipschitzness. In *Advances in neural information processing systems*, 2020.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P Xing, Laurent El Ghaoui, and Michael I Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, 2019.

Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *IEEE International Conference on Computer Vision*, pp. 6002–6012, 2019.

A Additional Related Works

Adversarial examples beyond norm-bounded models. Adversarial example for a given labeled data $(\boldsymbol{x}, y)$ is a data point $\boldsymbol{x}'$ that causes a classifier to output a wrong label on $\boldsymbol{x}'$. The adversarial examples typically come in the form of *restricted attacks* such as ϵ-bounded perturbations (Szegedy et al., 2013; Madry et al., 2018; Zhang et al., 2019; Blum et al., 2020), or *unrestricted attacks* such as adversarial rotations, translations, and deformations (Brown et al., 2018; Engstrom et al., 2017; Gilmer et al., 2018; Xiao et al., 2018; Alaifari et al., 2019). This work is on the latter setting, where we assume that the attacker is allowed to corrupt the representation of a test instance *arbitrarily* in a random subspace.

Feature-space attacks. Different from most existing attacks that directly perturb input pixels, there are a few prior works that focus on perturbing abstract features as ours. More specifically, the subspaces of features typically characterize styles, which include interpretable styles such as vivid colors and sharp outlines, and uninterpretable ones (Xu et al., 2020). Ganeshan & Babu (2019) proposed *feature disruptive attack* by generating image perturbation that disrupts features at each layer of the network and causes deep-features to be highly corrupt. They showed that the attacks generate strong adversaries for image classification, even in the presence of various defense measures. Despite a large amount of empirical works on adversarial feature-space attacks, many fundamental questions remain open, such as developing a *provable* defense against feature-space attacks.

Contrastive learning. Contrastive learning has received significant attention due to the recent popularity of self-supervised learning: many recent studies (Wu et al., 2018; Oord et al., 2018; Hjelm et al., 2018; Zhuang et al., 2019; Hénaff et al., 2019; Tian et al., 2019; Bachman et al., 2019) present promising results of unsupervised representation learning against their supervised counterparts. Representative self-supervised contrastive learning includes MoCo(v2) (He et al., 2020) and SimCLR (Chen et al., 2020). In ImageNet classification task, both methods almost match the accuracy of their supervised counterparts; in 7 detection/segmentation tasks on PASCAL VOC, COCO, and other datasets, MoCo (He et al., 2020) can outperform its supervised pre-training counterpart sometimes by large margins. A more recent work of Khosla et al. (2020) proposed *supervised contrastive learning*. The intuition behind supervised contrastive learning is to use soft k-nearest-neighbor classifier to replace the linear classifier and cross-entropy loss in the traditional design of network architecture and obtains improved performance than the cross-entropy training.

B Proof of Theorem 5.1

Theorem 5.1 (Restated)**.** *Given a test instance $\boldsymbol{x}$ and let m be the number of training data. Suppose that $\tau = o\left(r\sqrt{1-\frac{n_3}{n_2}}\right)$. The robust error $\mathcal{E}_{\text{adv}}^{\boldsymbol{x}}(f)$ of* RobustClassifier$(\tau, 0)$ *in Algorithm 1 for classifying $\boldsymbol{x}$ is at most $m\left(\frac{c\tau}{r\sqrt{1-\frac{n_3}{n_2}}}\right)^{n_2-n_3} + mc_0^{n_2-n_3}$, where $c > 0$ and $0 < c_0 < 1$ are absolute constants.*

Proof. We begin our analysis with the case of $n_3 = 1$. Suppose we have a training example $\boldsymbol{x}'$ of another class, and suppose $F(\boldsymbol{x})$ and $F(\boldsymbol{x}')$ are at distance D in the feature space. That is, $\mathsf{dist}(F(\boldsymbol{x}), F(\boldsymbol{x}')) = D$. Because $\tau = o(D)$, the probability that the adversary can move $F(\boldsymbol{x})$ to within distance τ of $F(\boldsymbol{x}')$ should be roughly the ratio of the surface area of a sphere of radius τ to the surface area of a sphere of radius D, which is at most

$$\left(\mathcal{O}\left(\frac{\tau}{D}\right)\right)^{n_2-1} \leq \left(\mathcal{O}\left(\frac{\tau}{r}\right)\right)^{n_2-1}$$

if the feature space is n_2-dimensional.

To analyze the case when the adversary subspace is n_3-dimensional, we need the following Random Projection Theorem.

Lemma B.1 (Random Projection Theorem[1]). *Let $\mathbf{z}$ be a fixed unit length vector in d-dimensional space and $\widetilde{\mathbf{z}}$ be the projection of $\mathbf{z}$ onto a random k-dimensional subspace. For $0 < \delta < 1$,*

$$\Pr\left[\left|\|\widetilde{\mathbf{z}}\|_2^2 - \frac{k}{d}\right| \geq \delta\frac{k}{d}\right] \leq e^{-\frac{k\delta^2}{4}}.$$

Without loss of generality, we assume $F(\boldsymbol{x}) = \mathbf{0}$ in $\mathbb{R}^{n_2}$. We use the peeling argument. Note that the random subspace in which the adversary vector is restricted to lie can be constructed by the following sampling scheme: we first sample a vector $\mathbf{v}_1$ uniformly at random from a unit sphere in the ambient space $\mathbb{R}^{n_2}$ centered at $\mathbf{0}$; fixing $\mathbf{v}_1$, we then sample a vector $\mathbf{v}_2$ uniformly at random from a unit sphere in the null space of $\mathsf{span}\{\mathbf{v}_1\}$; we repeat this procedure n_3 times and let $\mathsf{span}\{\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_{n_3}\}$ be the desired subspace. Note that the sampling scheme satisfies the random adversary model. For the fixed null space $\mathsf{null}(\mathsf{span}\{\mathbf{v}_1, ..., \mathbf{v}_i\})$ of dimension $n_2 - i$, according to the analysis of the case $n_3 = 1$, the failure probability of the algorithm over x' conditioning on D_i is at most $(\mathcal{O}(\tau/D_i))^{n_2-i-1}$, where D_i represents the distance of $F(\boldsymbol{x})$ and $F(\boldsymbol{x}')$ when they are projected to $\mathsf{null}(\mathsf{span}\{\mathbf{v}_1, ..., \mathbf{v}_i\})$. We also note that $\mathsf{null}(\mathsf{span}\{\mathbf{v}_1, ..., \mathbf{v}_i\})$ is a random subspace of dimension $n_2 - i$. Thus by Lemma B.1 (with constant δ), we have $D_i \geq Cr\sqrt{\frac{n_2-i}{n_2}}$ with probability at least $1 - e^{-c'(n_2-i)}$, where $C, c' > 0$ are absolute constants. Therefore, by the union bound over the choice of n_3 null spaces and the failure probability of the event $D_i \geq Cr\sqrt{\frac{n_2-i}{n_2}}$, the failure probability of the algorithm over x' is at most

$$\sum_{i=1}^{n_3} e^{-c'(n_2-i)} + \sum_{i=1}^{n_3}\left(\mathcal{O}\left(\frac{\tau}{Cr\sqrt{\frac{n_2-i}{n_2}}}\right)\right)^{n_2-i} \leq c_0^{n_2-n_3} + \left(\frac{c\tau}{r\sqrt{\frac{n_2-n_3}{n_2}}}\right)^{n_2-n_3}.$$

By the union bound over all m training data $\boldsymbol{x}'$'s completes the proof. □

C An Asymptotically Improved Bound of Theorem 5.1

Theorem C.1. *If $\tau = o(r)$, the robust error $\mathcal{E}^{\boldsymbol{x}}_{\text{adv}}(f)$ of* RobustClassifier$(\tau, 0)$ *in Algorithm 1 for classifying $\boldsymbol{x}$ is at most $\mathcal{O}\left(\frac{m}{n_2-n_3}\left(\frac{\tau}{r}\right)^{n_2-n_3}\frac{1}{B(n_3/2,(n_2-n_3)/2)}\right)$, where $B(\cdot,\cdot)$ is the Beta function.*

Remark 3. *Theorem C.1 is an asymptotic improvement over Theorem 5.1 for fixed n_3 and large n_2.*

Proof. Let $\boldsymbol{x}$ be the origin. By rotational symmetry, we assume WLOG that the random n_3-dimensional space R is given by $x_{n_3+1} = x_{n_3+2} = \cdots = x_{n_2} = 0$, and $\boldsymbol{x}'$ is the uniformly random unit vector $(z_1, \ldots, z_{n_2})$. Indeed, for a fixed direction from x, the set of subspaces for which the projection of x' lies along that direction is constrained by one vector each in the range space and kernel space, and is therefore in bijection to the set of subspaces associated with another fixed direction.

The adversary can win if and only if $\boldsymbol{x}\boldsymbol{x}'$ makes an angle $\theta \in \left[\frac{\pi}{2} - \phi, \frac{\pi}{2}\right]$ with the closest vector in R, i.e. with $(z_1, \ldots, z_{n_3}, 0, \ldots, 0)$, where $\phi = \arcsin\frac{\tau}{r}$. This is equivalent to

$$\frac{\pi}{2} - \phi \leq \arccos\frac{\sum_{i=1}^{n_3} z_i^2}{\sqrt{\sum_{i=1}^{n_3} z_i^2}\sqrt{\sum_{i=1}^{n_2} z_i^2}} \leq \frac{\pi}{2} \qquad \text{or,}$$

$$\sum_{i=1}^{n_3} z_i^2 \leq \sin^2\phi$$

i.e. distance from the orthogonal space $x_1 = x_2 = \cdots = x_{n_3} = 0$ is at most $\sin\phi = \frac{\tau}{r}$. Applying Lemma C.2, together with a union bound for the number of training points, gives the result. □

[1] http://www.cs.cornell.edu/courses/cs4850/2010sp/Scribe%20Notes/Lecture05.pdf

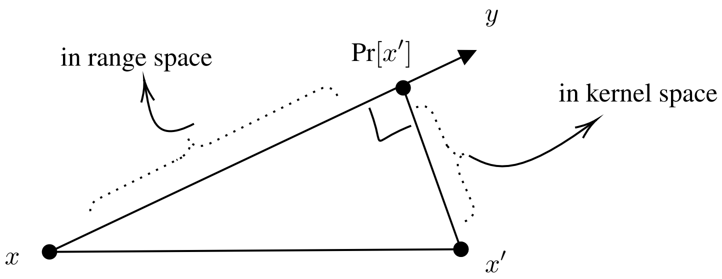

Figure 6: Rotational symmetry of adversarial subspaces. Let $\boldsymbol{y}$ be a random direction from test point $\boldsymbol{x}$, and $\Pr[\boldsymbol{x}']$ be the projection of training point $\boldsymbol{x}'$ on to $\boldsymbol{xy}$. For any adversarial space with $\Pr[\boldsymbol{x}']$ as the projection of $\boldsymbol{x}'$ on the space, we must have $\boldsymbol{xy}$ in the range space and $\boldsymbol{x}'\Pr[\boldsymbol{x}']$ in the null space.

Lemma C.2. *The fraction of points on the surface of the unit $(n-1)$-sphere at a distance at most small $\varepsilon = o(1)$ from a fixed $(n-k)$-hyperplane through its center is at most $\frac{2\varepsilon^k}{k}\frac{A(k-1)A(n-k-1)}{A(n-1)}$, where $A(m)$ is the surface-area of the unit m-sphere embedded in $m+1$ dimensions.*

Proof. Let the fixed hyperplane be $x_1 = x_2 = \cdots = x_k = 0$. We change the coordinates to a product of spherical coordinates (ρ is the distance from the hyperplane, r is the orthogonal component of the radius vector).

$$x_j = \begin{cases} \rho S_{j-1}\cos\phi_j & \text{if } j < k \\ \rho S_{j-1} & \text{if } j = k \\ r T_{j-k-1}\cos\alpha_{j-k} & \text{if } k < j < n \\ r T_{j-k-1} & \text{if } j = n \end{cases}$$

where $S_l = \prod_{i=1}^{l}\sin\phi_i, T_l = \prod_{i=1}^{l}\sin\alpha_i$. The desired surface area is easier to compute in the new coordinate system.

The new coordinates are $(y_1, \dots, y_n) = (\rho, \phi_1, \phi_2, \dots, \phi_{k-1}, r, \alpha_1, \dots, \alpha_{n-k-1})$, and let $z = \sqrt{r^2+\rho^2} = \sqrt{\sum_{i=1}^{n} x_i^2}$ denote the usual radial spherical coordinate. Volume element in this new coordinate system is given by

$$dV = |\det(J)|\, d\rho\, d\phi_1 \dots d\phi_{k-1} dr\, d\alpha_1 \dots d\alpha_{n-k-1}$$

where J is the Jacobian matrix, $J_{ij} = \frac{\partial x_i}{\partial y_j}$. We can write

$$J = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix}$$

where $A_{ij} = \frac{\partial x_i}{\partial y_j}$ for $1 \le i,j \le k$ and $B_{ij} = \frac{\partial x_{i+k}}{\partial y_{j+k}}$ for $1 \le i,j \le n-k$.

By Leibniz formula for determinants, it is easy to see

$$\begin{aligned} \det(J) &= \det(A)\cdot\det(B) \\ &= \rho^{k-1}\left(\prod_{i=1}^{k-2}\sin^{k-i-1}\phi_i\right)\cdot r^{n-k-1}\left(\prod_{i=1}^{n-k-2}\sin^{n-k-i-1}\alpha_i\right) \\ &= \rho^{k-1}r^{n-k-1}\left(\prod_{i=1}^{k-2}\sin^{k-i-1}\phi_i\right)\left(\prod_{i=1}^{n-k-2}\sin^{n-k-i-1}\alpha_i\right) \end{aligned}$$

Now the surface element is given by

$$dS = \frac{1}{z^{n-1}}\frac{dV}{dz} = \frac{1}{z^{n-1}}\left(\frac{dV}{dr}\frac{\partial r}{\partial z} + \frac{dV}{d\rho}\frac{\partial \rho}{\partial z}\right) = \frac{1}{rz^{n-2}}\frac{dV}{dr} + \frac{1}{\rho z^{n-2}}\frac{dV}{d\rho}$$

Plugging in our computation for dV,

$$dS = \left(\frac{\rho^{k-1}r^{n-k-2}}{z^{n-2}}\,d\rho + \frac{\rho^{k-2}r^{n-k-1}}{z^{n-2}}\,dr\right)\left(\prod_{i=1}^{k-2}\sin^{k-i-1}\phi_i d\phi_i\right)\left(\prod_{i=1}^{n-k-2}\sin^{n-k-i-1}\alpha_i d\alpha_1\right)$$

We care about $z = 1$ and $\rho \leq \varepsilon$ (or $r \geq \sqrt{1-\varepsilon^2}$). Notice

$$\int_{\sqrt{1-\varepsilon^2}}^{1} \frac{\rho^{k-2}r^{n-k-1}}{z^{n-2}}\,dr = \int_{\varepsilon}^{0} \rho^{k-2}r^{n-k-1}\frac{-\rho d\rho}{r} = \int_{0}^{\varepsilon} \rho^{k-1}r^{n-k-2}d\rho$$

Thus, using the surface element in the new coordinates and integrating, we get

$$\text{Area of } \varepsilon\text{-close points} = A(k-1)A(n-k-1)\cdot 2\int_0^{\varepsilon}\rho^{k-1}r^{n-k-2}d\rho \leq A(k-1)A(n-k-1)\cdot\frac{2\varepsilon^k}{k}$$

which gives the desired fraction. □

D Proof of Theorem 5.2

Theorem 5.2. *Consider the setting of Theorem 5.1, with an adversary having a κ-bounded distribution over the space of linear subspaces of a fixed dimension n_3 for perturbing the test point. If $\mathbf{E}(\tau, r)$ denotes the bound on error rate in Theorem 5.1 for* RobustClassifier$(\tau, 0)$ *in Algorithm 1, then the error bound of the same algorithm against the κ-bounded adversary is $\mathcal{O}(\kappa\mathbf{E}(\tau, r))$.*

Proof Sketch. To argue upper bounds on failure probability, we consider the set of adversarial subspaces which can allow the adversary to perturb the test point x close to a training point x'. Let $\mathcal{S}(x', \tau)$ denote the subset of linear subspaces of dimension n_3 such that for any $S \in \mathcal{S}(x', \tau)$ there exists $v \in S$ with $x + v \in \mathcal{B}(x', \tau)$. Note that from Section C, we can upper bound the fraction of the total probability space occupied by $\mathcal{S}(x', \tau)$ by $\frac{1}{m}\mathbf{E}(\tau, r)$, where constants in n_2, n_3 have been suppressed. If we show that $\mathcal{S}(x', \tau)$ is a measurable set, we can use the κ-boundedness of the adversary distribution to claim that the failure probability for misclassifying as x' is upper bounded by $\kappa\mathsf{vol}(\mathcal{S})\frac{1}{m}\mathbf{E}(\tau, r) = \mathcal{O}\left(\frac{\kappa}{m}\mathbf{E}(\tau, r)\right)$, since the volume of the complete adversarial space $\mathcal{S}$ is a constant in n_2, n_3. In Lemma D.1 below, we make the stronger claim that $\mathcal{S}(x', \tau)$ is convex. We can then use a union bound on the training points to get a bound on the total failure probability as $\mathcal{O}\left(\kappa\mathbf{E}(\tau, r)\right)$. □

Lemma D.1. *Let $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^{n_2}, \tau \in \mathbb{R}^+$ and $\mathcal{S}(x', \tau)$ denote the subset of linear subspaces of dimension n_3 such that for any $S \in \mathcal{S}(x', \tau)$ there exists $v \in S$ with $x + v \in \mathcal{B}(x', \tau)$. The set $\mathcal{S}(x', \tau)$ is convex.*

Proof. Let $S, S' \in \mathcal{S}(x', \tau)$. Then we have $v \in S, v' \in S'$ such that $x + v, x + v' \in \mathcal{B}(x', \tau)$. Let $S^* = \alpha S + (1-\alpha)S'$, $\alpha \in [0, 1]$. Pick $v^* = \alpha v + (1-\alpha)v' \in S^*$. $x + v^*$ must lie in $\mathcal{B}(x', \tau)$ by convexity of $\mathcal{B}(x', \tau)$. □

E Error Upper Bound with Outlier Removal

We will need the following assumption.

Assumption 2. *We assume that at least $1 - \delta$ fraction of mass of the marginal distribution $\mathcal{D}_{F(\mathcal{X})|y}$ over $\mathbb{R}^{n_2}$ can be covered by N balls $\mathbb{B}_1$, $\mathbb{B}_2$, ... $\mathbb{B}_N$ of radius τ, such that when the δ probability mass is excluded, each ball has density at least $\Pr_{\mathcal{D}_{F(\mathcal{X})}}[\mathbb{B}_k] \geq \frac{C_0}{m}\left(n_2 \log m + \log\frac{4N}{\beta}\right)$ (where $C_0 > 0$ is an absolute constant and $\delta, \beta \in (0, 1)$), all non-excluded points in each ball have the same label (call this the ball's label) and two balls with distinct labels are at least $\sigma > 2\tau$ apart.*

Theorem E.1. *If $\tau = o(\sigma)$ and $m = \Omega(\frac{n_2 N}{\beta}\log\frac{n_2 N}{\beta})$, the failure probability of Algorithm 1 is at most $\tilde{\mathcal{O}}\left(N\left(\frac{2\tau}{\sigma - 2\tau}\right)^{n_2 - n_3} + \delta + \beta\right)$, where the soft-O notation suppresses constants in n_2, n_3, and N, β, δ as in Assumption 2.*

Proof. We claim that the distance of a test point to any training point with a different label is at least $\sigma - 2\tau$ with high probability. To do this we use a covering argument to argue the test point x is within τ of some training point x^* of the same label.

Let $N(\tau, \sigma)$ be the number of balls of radius τ needed to cover at least $1 - \delta$ probability mass of the data, such that when the δ probability mass is excluded, each ball has density at least $\beta/N(\tau, \sigma)$, all non-excluded points in each ball have the same label and two balls with distinct labels are at least $\sigma > 2\tau$ apart (Assumption 2). Then with probability $1 - \delta$, the test point x will be within a ball $\mathbb{B}$ of its own label, and for $m = \Omega(\frac{n_2 N}{\beta} \log \frac{n_2 N}{\beta})$ with probability at least $1 - \beta/4$ over the sampling (by Lemma F.1), a training point x^* within $\mathbb{B}$ would have been sampled. Thus $\mathsf{dist}(x, x^*) \leq 2\tau$. Since any training point x' of a different label for x^* satisfies $\mathsf{dist}(x^*, x') > \sigma$, by triangle inequality, we have $\mathsf{dist}(x, x') > \sigma - 2\tau$. We can now apply the upper bound of Algorithm 1, together with a union bound over the failure modes, to get the desired bound on the failure probability. □

F Proof of Theorem 5.3

Theorem 5.3 (Restated)**.** *Suppose that $F(\boldsymbol{x}_1), ..., F(\boldsymbol{x}_m)$ are m training instances i.i.d. sampled from marginal distribution $\mathcal{D}_{F(\mathcal{X})}$. Under Assumption 1, with probability at least $1 - \beta/4$ over the sampling, we have $\Pr(\cup_{i=1}^m \mathbb{B}(F(\boldsymbol{x}_i), \tau)) \geq 1 - \delta$.*

Proof. The proofs of Theorem 5.3 are built upon the following lemma from Chaudhuri & Dasgupta (2010).

Lemma F.1 (Lemma 16, Chaudhuri & Dasgupta (2010))**.** *Suppose $\{\mathbf{z}_i\}_{i=1}^m \subseteq \mathbb{R}^{n_2}$ is a sample of m points drawn independently at random from a distribution $\mathcal{D}_\mathcal{Z}$ over $\mathcal{Z}$. There is a universal constant $C_0 > 0$ such that for any $\delta > 0$, with probability at least $1 - \delta/4$, for any ball $\mathbb{B} \subset \mathbb{R}^{n_2}$, if*

$$\Pr[\mathbb{B}] \geq \frac{C_0}{m}\left(n_2 \log m + \log \frac{4}{\delta}\right),$$

then there is at least a point $\mathbf{z}_i$ such that $\mathbf{z}_i \in \mathbb{B}$.

We are now ready to prove Theorem 5.3. By Lemma F.1, for a fixed ball $\mathbb{B}_k$ in Assumption 1, there is at least a sample $F(\boldsymbol{x}_i)$ such that $F(\boldsymbol{x}_i) \in \mathbb{B}_k$ with probability at least $1 - \beta/(4N)$. Therefore, with probability at least $1 - \beta/4$ (by the union bound over N balls), for all $k \in [N]$ there is at least a sample $F(\boldsymbol{x}_{i_k}) \in \{F(\boldsymbol{x}_1), F(\boldsymbol{x}_2), ..., F(\boldsymbol{x}_m)\}$ such that $F(\boldsymbol{x}_{i_k}) \in \mathbb{B}_k$. This implies $\cup_{i=1}^m \mathbb{B}(F(\boldsymbol{x}_i), \tau) \supseteq \cup_{k=1}^N \mathbb{B}_k$, since $\mathbb{B}_k$ is a ball of radius $\tau/2$. So with probability at least $1 - \beta/4$ over the sampling, we have $\Pr[\cup_{i=1}^m \mathbb{B}(F(\boldsymbol{x}_i), \tau)] \geq \Pr[\cup_{k=1}^N \mathbb{B}_k] \geq 1 - \delta$. □

G An Alternative Proof of Outputting "Don't Know" on Minority of Inputs

Definition 1 (Doubling dimension)**.** *A measure $\mathcal{D}_{F(\mathcal{X})}$ with support $F(\mathcal{X})$ is said to have a doubling dimension d, if for all points $F(\boldsymbol{x}) \in F(\mathcal{X})$ and all radius $\tau > 0$, we have $\mathcal{D}_{F(\mathcal{X})}(\mathcal{B}(F(\boldsymbol{x}), 2\tau)) \leq 2^d \mathcal{D}_{F(\mathcal{X})}(\mathcal{B}(F(\boldsymbol{x}), \tau))$.*

Lemma G.1. *Suppose that the measure $\mathcal{D}_{F(\mathcal{X})}$ has a doubling dimension d. Let D be the diameter of $F(\mathcal{X})$. Then for any point $F(\boldsymbol{x}) \in F(\mathcal{X})$ and any radius of the form $\tau = D/2^T$ for certain $T \in \mathbb{N}$, we have $\mathcal{D}_{F(\mathcal{X})}(\mathcal{B}(F(\boldsymbol{x}), \tau)) \geq (\tau/D)^d$.*

Proof. Since D is the diameter of $F(\mathcal{X})$, we have $\mathcal{D}_{F(\mathcal{X})}(\mathcal{B}(F(\boldsymbol{x}), D)) = 1$. Therefore, we have

$$\begin{aligned}\mathcal{D}_{F(\mathcal{X})}(\mathcal{B}(F(\boldsymbol{x}), \tau)) &= \mathcal{D}_{F(\mathcal{X})}(\mathcal{B}(F(\boldsymbol{x}), D/2^T)) \\ &\geq 2^{-d} \cdot \mathcal{D}_{F(\mathcal{X})}(\mathcal{B}(F(\boldsymbol{x}), D/2^{T-1})) \\ &\geq \cdots \\ &\geq 2^{-Td} \cdot \mathcal{D}_{F(\mathcal{X})}(\mathcal{B}(F(\boldsymbol{x}), D)) \\ &= 2^{-Td} \\ &= (\tau/D)^d.\end{aligned}$$

□

Lemma G.2 (Relating doubling dimension to covering number)**.** *Given any radius τ of the form $\tau = D/2^T$ for certain $T \in \mathbb{N}$, there is a covering of $F(\mathcal{X})$ using balls of radius τ of size no more than $(2D/\tau)^d$.*

Proof. We construct the covering balls of $F(\mathcal{X})$ as follows: when there is a point $F(\boldsymbol{x}) \in F(\mathcal{X})$ which is not contained in any current covering ball of radius τ, we add the ball $\mathcal{B}(F(\boldsymbol{x}), \tau)$ to the cover. We follow this procedure until every point in $F(\mathcal{X})$ is covered by some covering balls. Denote by $\mathcal{C}$ the set of centers for the balls in the cover.

We now show that this procedure stops after adding at most $(2D/\tau)^d$ balls to the cover. We note that by our construction, the centers of the covering are at least distance τ from each other, implying that the collection of $\mathcal{B}(F(\boldsymbol{x}), \tau/2)$ for $F(\boldsymbol{x}) \in \mathcal{C}$ are disjoint. This yields

$$\begin{aligned}1 &\geq \mathcal{D}_{F(\mathcal{X})}\left(\cup_{F(\boldsymbol{x})\in\mathcal{C}} \mathcal{B}(F(\boldsymbol{x}), \tau/2)\right) \\ &= \sum_{F(\boldsymbol{x})\in\mathcal{C}} \mathcal{D}_{F(\mathcal{X})}(\mathcal{B}(F(\boldsymbol{x}), \tau/2)) \quad \text{(since } \mathcal{B}(F(\boldsymbol{x}), \tau/2) \text{ are disjoint)} \\ &\geq \sum_{F(\boldsymbol{x})\in\mathcal{C}} \left(\frac{\tau}{2D}\right)^d \quad \text{(by Lemma G.1)} \\ &= |\mathcal{C}| \left(\frac{\tau}{2D}\right)^d.\end{aligned}$$

So we have $|\mathcal{C}| \leq (2D/\tau)^d$. □

Theorem G.3. *Suppose that the measure $\mathcal{D}_{F(\mathcal{X})|y}$ in the feature space has a doubling dimension d. Let D be the diameter of $F(\mathcal{X})$. For any $\tau > 0$ and any $\delta > 0$, if we draw an i.i.d. sample of size $m \geq \frac{2D}{\tau}(d \log \frac{4D}{\tau} + \log \frac{1}{\delta})$, then with probability at least $1 - \delta$ over the sampling of S, we have $\sup_{\boldsymbol{x}\in\mathcal{X}} d(F(\boldsymbol{x}), NN_S(F(\boldsymbol{x}))) \leq \tau$.*

Proof. Lemma G.2 implies that there exists a covering of $F(\mathcal{X})$ of size $(4D/\tau)^d$ which consists of balls of radius $\tau/2$. For any ball B in the cover, the probability that there is no sample point landing in ball B is

$$\begin{aligned}(1 - \mathcal{D}_{F(\mathcal{X})}(B))^m &\leq \left(1 - \frac{\tau}{2D}\right)^m \quad \text{(by Lemma G.1)} \\ &\leq \exp\left(-\frac{m\tau}{2D}\right).\end{aligned}$$

Thus by the union bound over all the balls in the cover, the probability of the event E that there is at least one ball B in the cover which does not contain any sample points is

$$\begin{aligned}\Pr[E] &\leq |\mathcal{C}| \exp\left(-\frac{m\tau}{2D}\right) \quad \text{(by the union bound)} \\ &\leq \left(\frac{4D}{\tau}\right)^d \exp\left(-\frac{m\tau}{2D}\right) \quad \text{(by Lemma G.2)} \\ &\leq \delta. \quad \left(\text{since } m \geq \frac{2D}{\tau}\left(d \log \frac{4D}{\tau} + \log \frac{1}{\delta}\right)\right)\end{aligned}$$

□

H Data-specific Optimal Thresholds

We begin with a useful lemma, which allows us to focus on small τ. For this section we will assume fixed n_2, n_3 for a simpler exposition, asymptotics in the the dimensions may be readily obtained by following our proofs.

Lemma H.1. *Let ϕ be a distribution defined on a compact convex subset C of $\mathbb{R}^n$ which is continuous and strictly positive on C, and has bounded partial derivatives throughout C. If m samples $B = \{\beta_1, \ldots, \beta_m\}$ are drawn from ϕ, for any β_i the probability that the distance d_i to its nearest neighbor in B is not $O(m^{-1/n})$ is $o(1)$.*

Proof. We compute asymptotic moments of nearest neighbor distance distribution using Evans et al. (2002) together with a concentration inequality to complete the proof. Indeed, the asymptotic mean nearest neighbor distance is shown to be $O(m^{-1/n})$, and the variance is $O(m^{-2/n})$. By Chebyshev's inequality, the probability that d_i is outside $\omega(1)$ standard deviations is $o(1)$. □

We can apply Lemma H.1 to (compact convex subsets of the support of) the distribution $F(\boldsymbol{x})$ as $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}$. Essentially it implies that for a large enough training sample, we expect most of the change in abstention rate occur for small values of τ. This crucially allows us to show that $\mathcal{E}_{\text{adv}}(\tau)$ as well as $\mathcal{D}_{\text{nat}}(\tau)$ (and therefore $g(\tau)$) are *nice* enough to be optimized. We will now state a theorem showing τ can be learned online to optimize $g(\tau)$.

We will need the following lemmas about $\mathcal{E}_{\text{adv}}(\tau)$ and $\mathcal{D}_{\text{nat}}(\tau)$ respectively.

Lemma H.2. *If τ is $o\left(\min\{m^{-1/n_2}, r\}\right)$, $\mathcal{E}_{\text{adv}}(\tau)$ is $O\left(m^{\frac{n_3+1}{n_2}}/r^{n_2-n_3}\right)$-Lipshcitz.*

Proof. Consider the probability that the adversary is able to succeed in misclassifying a test point x as a fixed training point x' (of different label) only when the threshold increases from τ to $\tau+d\tau$. WLOG, let x be the origin and the adversarial subspace S be given by $x_{n_3+1} = x_{n_3+2} = \cdots = x_{n_2} = 0$, and x' is the uniformly random unit vector $(z_1, \ldots, z_{n_2})$. The adversary can win if and only if xx' makes an angle $\theta \in \left[\frac{\pi}{2} - \phi, \frac{\pi}{2}\right]$ with the closest vector in S, i.e. $(z_1, \ldots, z_{n_3}, 0, \ldots, 0)$, where $\phi = \arcsin \frac{\tau}{r}$. This is equivalent to

$$\frac{\pi}{2} - \phi \leq \arccos \frac{\sum_{i=1}^{n_3} z_i^2}{\sqrt{\sum_{i=1}^{n_3} z_i^2}\sqrt{\sum_{i=1}^{n_2} z_i^2}} \leq \frac{\pi}{2} \qquad \text{or,}$$

$$\sum_{i=1}^{n_3} z_i^2 \leq \sin^2 \phi$$

i.e., the distance Δ from the orthogonal space $x_1 = x_2 = \cdots = x_{n_3} = 0$ is at most $\sin \phi = \frac{\tau}{r}$. Therefore a threshold change of τ to $\tau + d\tau$ corresponds to $\Delta \in \left(\frac{\tau}{r}, \frac{\tau+d\tau}{r}\right)$. We observe from the proof of Lemma C.2 that

$$\begin{aligned}\Pr\left[\Delta \in \left(\frac{\tau}{r}, \frac{\tau + d\tau}{r}\right)\right] &= C(n_2, n_3) \cdot \int_{\tau/r}^{(\tau+d\tau)/r} \rho^{n_2-n_3-1}\left(\sqrt{1-\rho^2}\right)^{n_3-2} d\rho \\ &\leq C(n_2, n_3) \cdot \frac{\tau^{n_2-n_3-1} d\tau}{r^{n_2-n_3}},\end{aligned}$$

where $C(n_2, n_3) = 2A(n_3 - 1)A(n_2 - n_3 - 1)$ is a constant for fixed dimensions n_2, n_3. Using a union bound we conclude,

$$\mathcal{E}_{\text{adv}}(\tau + d\tau) - \mathcal{E}_{\text{adv}}(\tau) \leq mC(n_2, n_3)\frac{\tau^{n_2-n_3-1} d\tau}{r^{n_2-n_3}}$$

The slope bound increases with τ, substituting $\tau = O\left(m^{-1/n_2}\right)$ gives the desired bound on Lipschitzness. □

Corollary H.3. *For a $\tilde{\kappa}$-bounded adversary, $\mathcal{E}_{\text{adv}}(\tau)$ is $O\left(\tilde{\kappa} m^{\frac{n_3+1}{n_2}}/r^{n_2-n_3}\right)$-Lipshcitz, if τ is $o\left(\min\{m^{-1/n_2}, r\}\right)$.*

Lemma H.4. *Suppose that the data distribution satisfies the assumptions in Lemma H.1, and further is κ-bounded. The expected number of discontinuities in $\mathcal{D}_{\mathrm{nat}}(\tau)$ in any interval of width w for $\tau = o\left(m^{-1/n_2}\right)$ is $O(\kappa m^{1/n_2}|T|w)$.*

Proof. Note that the discontinuities of $\mathcal{D}_{\mathrm{nat}}(\tau)$ in an interval $(\tau, \tau + w)$ corresponds to points $(\boldsymbol{x}, \boldsymbol{y}) \in T$ such that nearest neighbor distance of $\boldsymbol{x}$ is in that interval.

$$\begin{aligned}
E[\text{number of discontinuities in } (\tau, \tau + w)] &= |T| \Pr[\text{nearest neighbor of a test point } \in (\tau, \tau + w)] \\
&\leq |T| \Pr[\text{some neighbor of a test point } \in (\tau, \tau + w)] \\
&\leq \kappa m |T| \mathsf{vol}(\text{spherical shell of radius } \tau \text{ and width } w) \\
&= \kappa m |T| O(\tau^{n_2 - 1} w) \\
&= O(\kappa m^{1/n_2} |T| w)
\end{aligned}$$

□

For the full proof of Theorem 6.2, we will need the definition of dispersion, and a low-regret bound for dispersed functions.

Definition 2 (Dispersion, Balcan et al. (2018a)). *Let $u_1, \ldots, u_T : C \to [0, 1]$ be a collection of functions where u_i is piecewise Lipschitz over a partition P_i of C. We say that P_i splits a set A if A intersects with at least two sets in P_i. The collection of functions is (w, k)-dispersed if every ball of radius w is split by at most k of the partitions $P_1, \ldots, P_T$.*

Typically we would use discontinuities of u_i's as the partitions P_i's. Intuitively a sequence of functions is dispersed if the discontinuities do not concentrate in a small region of the domain space over time. If the sequence of functions is dispersed, we can bound the regret of a simple exponential forecaster algorithm (Algorithm 2) by the following theorem.

Theorem H.5 (Balcan et al. (2018a)). *Let $u_1, \ldots, u_T : C \to [0, 1]$ be any sequence of piecewise L-Lipschitz functions that are (w, k)-dispersed. Suppose $C \subset \mathbb{R}^d$ is contained in a ball of radius R and $B(\rho^*, w) \subset C$, where $\rho^* = \arg\max_{\rho \in C} \sum_{i=1}^{T} u_i(\rho)$. The exponentially weighted forecaster with $\lambda = \sqrt{d \ln(R/w)/T}$ has expected regret bounded by $O\left(\sqrt{Td \log(R/w)} + k + TLw\right)$.*

Algorithm 2 Exponential Forecaster Algorithm

```
Input: step size parameter $\lambda \in (0, 1]$.
Output: thresholds $\tau_t$ for times $t = 1, 2, \ldots, T$.
Set $w_1(\rho) = 1$ for all $\rho \in C$
for $t = 1, 2, \ldots, T$ do
    $W_t := \int_C w_t(\rho) d\rho$
    Sample $\rho$ with probability proportional to $w_t(\rho)$, i.e. with probability $p_t(\rho) = \frac{w_t(\rho)}{W_t}$
    Observe $u_t(\cdot)$
    For each $\rho \in C,$ set $w_{t+1}(\rho) = e^{\lambda u_t(\rho)} w_t(\rho)$
```

We now restate and prove our main theorem.

Theorem 6.2. *Assume τ is $o\left(\min\{m^{-1/n_2}, r\}\right)$, and the data distribution is continuous, κ-bounded, positive and has bounded partial derivatives. If τ is set using a continuous version of the multiplicative updates algorithm (Algorithm 2 in Appendix H, Balcan et al. (2018a)), then with probability at least $1 - \delta$, the expected regret in T rounds is bounded by $O\left(\sqrt{n_2 T \log\left(\frac{\kappa R T m b}{\delta r^{n_2 - n_3}}\right)}\right)$, where R is a bound on the largest distance between any two training points, b is the batch size, and r is the smallest distance between points of different labels.*

Proof. Assume the test data arrives in T batches of size b. We apply Algorithm 2 to set threshold τ for each batch. The utility function is set as $g(\tau)$ with the batch as the test set over which the error and abstention rates are computed.

Algorithm 3 Robust classifier in the feature space with point-specific threshold $\tau_i^{\mathcal{A}}$ of "don't know"

```
**Input:** A test feature $F(\boldsymbol{x})$ (potentially an adversarial example), a set $\mathcal{A}$ of training features $F(\boldsymbol{x}_i^{\mathcal{A}})$ and their labels $y_i^{\mathcal{A}}$, $i \in [m_{\mathcal{A}}]$, a set $\mathcal{B}$ of training features $F(\boldsymbol{x}_i^{\mathcal{B}})$ and their labels $y_i^{\mathcal{B}}$, $i \in [m_{\mathcal{B}}]$.
**Output:** A predicted label of $F(\boldsymbol{x})$, or "don't know".
$\tau_i^{\mathcal{A}} \leftarrow \min_{j:\, y_i^{\mathcal{A}} \neq y_j^{\mathcal{B}}} \mathsf{dist}(F(\boldsymbol{x}_i^{\mathcal{A}}), F(\boldsymbol{x}_j^{\mathcal{B}}))$ for all $i \in [m_{\mathcal{A}}]$
$i_{\min} \leftarrow \arg\min_{i \in [m]} \mathsf{dist}(F(\boldsymbol{x}), F(\boldsymbol{x}_i^{\mathcal{A}}))$
**if** $\mathsf{dist}(F(\boldsymbol{x}), F(\boldsymbol{x}_{i_{\min}}^{\mathcal{A}})) < \tau_{i_{\min}}^{\mathcal{A}}$ **then**
    Return $y_{i_{\min}}^{\mathcal{A}}$
**else**
    Return "don't know"
```

By Lemma H.2, we know that $\mathcal{A}_{\text{adv}}(\tau)$ is L-Lipschitz if τ is $o\left(\min\{m^{-1/n_2}, r\}\right)$, where $L = O\left(\frac{m^{\frac{n_3+1}{n_2}}}{r^{n_2-n_3}}\right)$. Since $\mathcal{D}_{\text{nat}}(\tau)$ is piecewise constant, this implies $g(\tau)$ is also L-Lipschitz.

By Lemma H.4, for batch size b, $\mathcal{D}_{\text{nat}}(\tau)$ has $O(\kappa m^2|T|w)$ in any interval of width w. To apply Theorem H.5, we need a concentration bound. By Markov's inequality, with probability at least $1-\delta$, $\mathcal{D}_{\text{nat}}(\tau)$ is $(w, \frac{\kappa m^{1/n_2} bw}{\delta})$-dispersed for any w. Since $\mathcal{E}_{\text{adv}}(\tau)$ is Lipschitz, $g(\tau)$ is also $(w, \frac{\kappa m^{1/n_2} bw}{\delta})$-dispersed.

We can now apply Theorem H.5 with $w = \frac{\delta}{\kappa m^{1/n_2} bL\sqrt{T}}$ to conclude the desired regret bound. □

I Estimating point-specific threshold of "don't know"

Algorithm 3 gives an alternative to our algorithm where instead of using a fixed threshold for each point, we use a variable point-specific threshold learned from the data.

For this algorithm, we have the following guarantee.

Theorem I.1. *Suppose that the sets $\mathcal{A}$ and $\mathcal{B}$ are two independent samples from $F(\mathcal{X})$ of size $m_{\mathcal{A}}$ and $m_{\mathcal{B}}$, respectively. Let $m_{\mathcal{B}} = \frac{m_{\mathcal{A}}}{\epsilon\delta}$. Then with probability at least $1-\delta$ over the draw of $\mathcal{A}$, for a new sample $F(\boldsymbol{x}')$, the probability that "$F(\boldsymbol{x}')$ is closer to $F(\boldsymbol{x}^{\mathcal{A}})$ than any point in $\mathcal{B}$ of different labels than $F(\boldsymbol{x}^{\mathcal{A}})$, and $F(\boldsymbol{x}')$ has a different label than $F(\boldsymbol{x}^{\mathcal{A}})$" is at most ϵ, where the probability is taken over the draw of $F(\boldsymbol{x}')$ and the draw of $\mathcal{B}$.*

Proof. Fixing the draw of set $\mathcal{A}$, we can think of picking a random set $\mathcal{S}$ of size $m_{\mathcal{B}}+1$ and randomly choosing one of the points in it to be $F(\boldsymbol{x}')$ and the rest to be $\mathcal{B}$. Assuming $\mathcal{S}$ has at least one point in it of a different label than $F(\boldsymbol{x}^{\mathcal{A}})$, then there is exactly a $\frac{1}{m_{\mathcal{B}}+1}$ probability that we choose $F(\boldsymbol{x}')$ to be the closest point in $\mathcal{S}$ to $F(\boldsymbol{x}^{\mathcal{A}})$ of a different label than $F(\boldsymbol{x}^{\mathcal{A}})$; if $\mathcal{S}$ has all points of the same label as $\boldsymbol{x}$, then the probability is 0. Now we can apply the union bound over all $F(\boldsymbol{x}^{\mathcal{A}})$ in $\mathcal{A}$ to get a total probability of failure at most $\frac{m_{\mathcal{A}}}{m_{\mathcal{B}}+1} < \epsilon\delta$.

The above analysis gives an expected failure probability over the draw of set $\mathcal{A}$. Applying the Markov inequality gives a high-probability bound. □

J A Simple Intuitive Example with exact calculation demonstrating significance of data-driven decision making

We will do an exact computation of the optimal value for the threshold τ in our Algorithm 1 RobustClassifier$(\tau, 0)$ to demonstrate its data dependence, and underline the significance of data-driven parameter setting as examined in Section 6. The optimality will be with respect to the objective function $g(\tau) := \mathcal{A}_{\text{adv}}(\tau) - c\mathcal{D}_{\text{nat}}(\tau)$, where $\mathcal{A}_{\text{adv}}(\tau) := 1 - \mathcal{E}_{\text{adv}}(\tau)$ is the robust accuracy and $\mathcal{D}_{\text{nat}}(\tau)$ is the abstention rate. Consider a simple instance of binary classification of data distribution in two uniformly distributed one-dimensional clusters of diameter D each, arranged collinearly and at distance r apart, as depicted in Figure J. Further assume that our training set consists of $2m$ examples, m from each class. Even in this toy setting, we are able to show that the optimal threshold varies with data-specific factors.

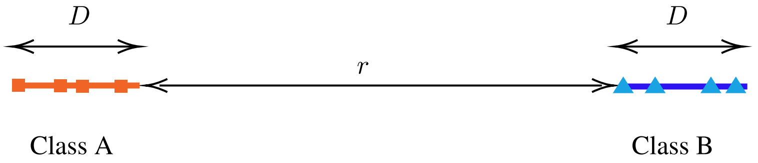

Figure 7: A simple example where we compute the optimal value of the abstention threshold exactly. Classes A and B are both distributed respectively on segments of length D, embedded collinear and at distance r in $\mathbb{R}^2$.

Formal setting: We set the feature and adversary dimensions as $n_2 = 2, n_3 = 1$. Examples of class A are all located on the segment $S_A = [(0,0),(D,0)]$, similarly instances of class B are located on $S_B = [(D+r,0),(2D+r,0)]$ (where $[\mathbf{a},\mathbf{b}] := \{\alpha\mathbf{a} + (1-\alpha)\mathbf{b} \mid \alpha \in [0,1]\}$). The data distribution returns an even number of samples, $2m$, with $m > 0$ points each drawn uniformly from S_A and S_B.

For this setting, we show that the optimal value of the threshold is a complex function of both the geometry (D, r) and the sampling rate (m).

Theorem J.1. *Let $\tau^* := \arg\max_{\tau\in\mathbb{R}^+} g(\tau)$. For the setting considered above, if we further assume $D = o(r)$ and $m = \omega\left(\log\left(\frac{2\pi cr}{D}\right)\right)$, then there is a unique value of τ^* in $[0, D/2)$. Further,*

$$\tau^* = \begin{cases} \Theta\left(\frac{D\log((\pi crm)/D)}{m}\right) & \text{if } \frac{1}{m} < \frac{\pi cr}{D} \\ 0 & \frac{\pi cr}{D} \leq \frac{1}{m} \end{cases}$$

Proof. We compute accuracy $\mathcal{A}_{\text{adv}}(\tau)$ and abstention rate $\mathcal{D}_{\text{nat}}(\tau)$ as functions of τ. Even with $D = o(r)$, the exact computation of the robust accuracy as a simple closed form is difficult without further assuming $\tau = o(r)$ as well. Fortunately, by Lemma J.2, we only need to consider $\tau \leq D$. For this case, indeed $\tau = o(r)$. We compute the abstention and accuracy rates in Lemmas J.3 and J.4, respectively. This gives us, for $\tau \leq D$,

$$\begin{aligned} g(\tau) = 1 - \frac{\tau}{\pi r}\left(1 - \frac{m+3}{m+1}\cdot\Theta\left(\frac{D}{r}\right)\right) - \Theta\left(\left(\frac{\tau}{r}\right)^3\right) \\ - \frac{c}{m+1}\left[2\left(1-\frac{\tau}{D}\right)^{m+1} + (m-1)\mathbb{I}_{\tau\leq D/2}\left(1-\frac{2\tau}{D}\right)^{m+1}\right] \end{aligned}$$

For $\tau \leq D/2$,

$$\begin{aligned} g'(\tau) = -\frac{1}{\pi r}\left(1 - \frac{m+3}{m+1}\cdot\Theta\left(\frac{D}{r}\right)\right) - \Theta\left(\frac{1}{r}\left(\frac{\tau}{r}\right)^2\right) \\ + \frac{2c}{D}\left[\left(1-\frac{\tau}{D}\right)^m + (m-1)\left(1-\frac{2\tau}{D}\right)^m\right] \end{aligned}$$

We need to consider two cases.

Case 1. $\frac{\pi cr}{D} \leq \frac{1}{m}$

In this case $g'(0) = \frac{-1}{\pi r} + \frac{2cm}{D} \leq 0$. Since $g''(\tau) \leq 0$, so we must have the only maximum at $\tau = 0$.

Otherwise, $g'(0) = \frac{-1}{\pi r} + \frac{2cm}{D} > 0$. Also $g'(D/2) = \frac{-1}{\pi r} + \frac{2c}{D2^m} < 0$ since $m > \log\left(\frac{2\pi cr}{D}\right)$. But $g''(\tau) \leq 0$, so we must have a unique local maximum in $(0, D/2)$, which is the global maximum.

Further, define y as $\tau = \frac{D}{m}\log y$. Now if $y = 2^{o(m)}$, we have $\frac{\tau}{D} = o(1)$, or

$$\left(1-\frac{\tau}{D}\right)^m = \exp\left(m\log\left(1-\frac{\tau}{D}\right)\right) = y^{-1-o(1)}$$

Case 2. $\frac{1}{m} < \frac{\pi c r}{D}$

If $y > 1$, for $y = \frac{2\pi c r m}{D}$,

$$\begin{aligned} g'(\tau) &= \frac{-1}{\pi r} + \frac{2c}{D}\left[\left(\frac{D}{2\pi c r m}\right)^{1+o(1)} + (m-1)\left(\frac{D}{2\pi c r m}\right)^{2+o(1)}\right] \\ &< \frac{-1}{\pi r} + \frac{2c}{D}\left[\left(\frac{D}{2\pi c r m}\right)^{1} + (m-1)\left(\frac{D}{2\pi c r m}\right)^{1}\right] \\ &= \frac{-1}{\pi r} + \frac{2c}{D}\left[\frac{D}{2\pi c r}\right] = 0 \end{aligned}$$

and for $y = \left(\frac{2\pi c r(m-1)}{D}\right)^{1/4}$,

$$\begin{aligned} g'(\tau) &= \frac{-1}{\pi r} + \frac{2c}{D}\left[\left(\frac{D}{2\pi c r(m-1)}\right)^{\frac{1}{4}+o(1)} + (m-1)\left(\frac{D}{2\pi c r m}\right)^{\frac{1}{2}+o(1)}\right] \\ &> \frac{-1}{\pi r} + \frac{2c}{D}\left[\left(\frac{D}{2\pi c r(m-1)}\right)^{1} + (m-1)\left(\frac{D}{2\pi c r(m-1)}\right)^{1}\right] \\ &= \frac{1}{\pi r(m-1)} > 0 \end{aligned}$$

Together, we get that $\tau^* = \Theta\left(\frac{D\log((\pi c r m)/D)}{m}\right)$ in this case. □

Lemma J.2. *In the setting of Theorem J.1, $g(\tau)$ is monotonically decreasing for $\tau > D$.*

Proof. Note that $\mathcal{D}_{\text{nat}}(\tau) = 0$ for $\tau > D$ as long as $m > 0$, since any test point of a class must be within D of every training point of that class. Hence, it suffices to note that $\mathcal{A}_{\text{adv}}(\tau)$ is monotonically decreasing in τ (increasing the threshold can only increase the ability of the adversary to successfully perturb to the opposite class). □

Lemma J.3. *In the setting of Theorem J.1, the abstention rate is given by*

$$\mathcal{D}_{\text{nat}}(\tau) = \frac{1}{m+1}\left[2\mathbb{I}_{\tau \le D}\left(1 - \frac{\tau}{D}\right)^{m+1} + (m-1)\mathbb{I}_{\tau \le D/2}\left(1 - \frac{2\tau}{D}\right)^{m+1}\right]$$

Proof. Note that for $\tau \ge D$, if $m > 0$, we never abstain on any test point. So we will assume $\tau \le D$ in the following. Consider a test point $\boldsymbol{x} = (x, 0)$ sampled from class A (class B is symmetric, so the overall abstention rate is the same is that of points drawn from class A). Let $\text{nbd}_{\boldsymbol{x}}(\tau)$ denote the intersection of a ball of radius τ around x with S_A. For x to be classified as 'don't know', we must have no training point sampled from $\text{nbd}_{\boldsymbol{x}}(\tau)$. This happens with probability $\left(1 - \frac{|\text{nbd}_{\boldsymbol{x}}(\tau)|}{D}\right)^m$, where $|\text{nbd}_{\boldsymbol{x}}(\tau)|$ is the size of $\text{nbd}_{\boldsymbol{x}}(\tau)$ and is given by

$$|\text{nbd}_{\boldsymbol{x}}(\tau)| = \begin{cases} \max\{x+\tau, D\} & x < \tau \\ \max\{2\tau, D\} & \tau \le x \le D - \tau \\ \max\{D - x + \tau, D\} & x > D - \tau \end{cases}$$

Averaging over the distribution of test points $\boldsymbol{x}$, we get

$$\begin{aligned} \mathcal{D}_{\text{nat}}(\tau) &= \frac{1}{D}\int_0^D \left(1 - \frac{|\text{nbd}_{\boldsymbol{x}}(\tau)|}{D}\right)^m dx \\ &= \frac{1}{m+1}\left[2\left(1 - \frac{\tau}{D}\right)^{m+1} + (m-1)\mathbb{I}_{\tau \le D/2}\left(1 - \frac{2\tau}{D}\right)^{m+1}\right] \end{aligned}$$

□

Lemma J.4. *In the setting of Theorem J.1, the robust accuracy rate for $\tau \leq D$ is given by*

$$\mathcal{A}_{\text{adv}}(\tau) = 1 - \frac{\tau}{\pi r}\left(1 - \frac{m+3}{m+1} \cdot \Theta\left(\frac{D}{r}\right)\right) - \Theta\left(\left(\frac{\tau}{r}\right)^3\right)$$

Proof. Consider a test point $\boldsymbol{x} = (x, 0)$ from S_A. Let $\boldsymbol{y} = (y, 0)$ denote the nearest point in S_B. In the given geometry, it is easy to see that if $\boldsymbol{x}$ can be perturbed into the τ neighborhood of some point $\boldsymbol{y'} \in S_B$ when moved along a fixed direction, then it must be possible to perturb it into the τ neighborhood of $\boldsymbol{y}$ (Figure J). Therefore it suffices to consider directions where perturbation to the τ-ball around $\boldsymbol{y}$ is possible.

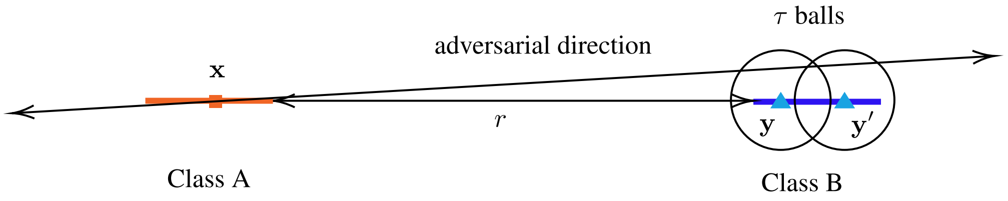

Figure 8: It suffices to consider the nearest point of the opposite class for adversarial perturbation.

Therefore the probability of adversary's success for $\boldsymbol{x}$, given $\boldsymbol{y}$ is the nearest point of the opposite class, is

$$\text{err}_{\boldsymbol{x}|\boldsymbol{y}}(\tau) = \frac{1}{\pi}\arcsin\left(\frac{\tau}{y - x}\right) = \frac{1}{\pi}\arcsin\left(\frac{\tau}{r + d}\right)$$

where $d = y - x - r \in [0, 2D]$. Now since $\tau \leq D = o(r)$, we have

$$\text{err}_{\boldsymbol{x}|\boldsymbol{y}}(\tau) = \frac{\tau}{\pi(r + d)} + \Theta\left(\left(\frac{\tau}{r}\right)^3\right) = \frac{\tau}{\pi r}\left(1 - \Theta\left(\frac{d}{r}\right)\right) + \Theta\left(\left(\frac{\tau}{r}\right)^3\right)$$

We can now compute the average error using the probability distributions for $\boldsymbol{x}$ and $\boldsymbol{y}$, $\boldsymbol{x}$ is a uniform distribution over S_A, while $\boldsymbol{y}$ is a nearest-neighbor distribution.

$$p(x) = \frac{1}{D}, \quad p(y) = \frac{m}{D}\left(1 - \frac{y - r - D}{D}\right)^{m-1}$$

The average value of d is

$$\bar{d} = \int_0^D \int_0^D (y' + x')\frac{m}{D}\left(1 - \frac{y'}{D}\right)^{m-1} dy' \frac{dx'}{D} = \frac{D(m+3)}{2(m+1)}$$

Using this to compute the average of $\text{err}_{\boldsymbol{x}|\boldsymbol{y}}(\tau)$ gives the result. □

K Algorithm for the Exact Computation of Attacks for our Algorithm under our Threat Model

Overview of Algorithm 4: If the point $\boldsymbol{u}_i$ closest to the training point $\boldsymbol{x}_i$ of different label than test point $\boldsymbol{x}$ in the adversarial subspace $\mathcal{S}$ (slight abuse of notation to refer to $\boldsymbol{x} + \mathcal{S}$ as $\mathcal{S}$) is closer to $\boldsymbol{x}_i$ than any training point $\boldsymbol{w}_j$ with the same label as $\boldsymbol{x}$ and within the threshold τ of $\boldsymbol{x}_i$, it will be misclassified as $\boldsymbol{x}_i$ (or potentially another point of an incorrect label). If however $\boldsymbol{u}_i$ is closer to some $\boldsymbol{w}_j$, we look at the points closer to $\boldsymbol{x}_i$ than all $\boldsymbol{w}_j$ in the subspace $\mathcal{S}$, and consider the closest point $\mathbf{z}_i$ to $\boldsymbol{x}_i$ (if it is within threshold τ) which should be misclassified. This can be computed using a convex optimization program (Line 14 of Algorithm 4) in n_3 dimensions. We claim it is sufficient to look at these two points for each training example $\boldsymbol{x}_i$.

Proof of correctness: To argue correctness of Algorithm 5, suppose an adversary wins by perturbing to some point $\mathbf{v}$. Then $\mathbf{v}$ must be closer to some point $\boldsymbol{x}_i$ than all $\boldsymbol{w}_j \in C$ (the set of training points with same label as $\boldsymbol{x}$) and within τ of $\boldsymbol{x}_i$. If $\boldsymbol{u}_i$ is closer to $\boldsymbol{x}_i$ than all $\boldsymbol{w}_j \in C$ then, it must be at least as close as $\mathbf{v}$ (since $\mathbf{v}$ is in the adversarial subspace $\mathcal{S}$) and therefore within τ of $\boldsymbol{x}_i$.

Algorithm 4 Exact computation of attacks under threat model 3.1 against Algorithm 1

```
Input: A randomly-sampled adversarial subspace $\mathcal{S}$ of dimension $n_3$, a test feature $F(\boldsymbol{x})$ and its label $y$, a set of training features $F(\boldsymbol{x}_i)$ and their labels $y_i$, $i \in [m]$, a threshold parameter $\tau$.
Output: A misclassified adversarial feature $F(\boldsymbol{x}) + \mathbf{v}$, $\mathbf{v} \in \mathcal{S}$ if it exists; otherwise, output “no adversarial example”.
$F_{\text{center}}(\boldsymbol{x}_i) \leftarrow F(\boldsymbol{x}_i) - F(\boldsymbol{x})$ for $i \in [m]$
for $i = 1, ..., m$ do
  if $y_i \neq y$ then
    $\boldsymbol{u}_i = \arg\min_{\boldsymbol{u}\in\mathcal{S}} d(\boldsymbol{u}, F_{\text{center}}(\boldsymbol{x}_i))$        (candidate adversarial perturbation)
    $C \leftarrow \{\boldsymbol{x}_j \mid y_j = y\}$
    if $\exists \boldsymbol{w} \in C \mid \mathsf{dist}(\boldsymbol{u}_i, F_{\text{center}}(\boldsymbol{w})) < \mathsf{dist}(\boldsymbol{u}_i, F_{\text{center}}(\boldsymbol{x}_i))$ then
      $H_j \leftarrow \{\mathbf{z} \mid \mathsf{dist}(F_{\text{center}}(\boldsymbol{x}_i), \mathbf{z}) \leq \mathsf{dist}(\boldsymbol{w}_j, \mathbf{z}), \boldsymbol{w}_j \in C\}$
      $H \leftarrow \cap_i H_i$
      $A \leftarrow H \cap \mathcal{S}$
      if $A = \{\}$ then
        continue
      $\mathbf{z}_i = \arg\min_{\mathbf{z}\in A} \mathsf{dist}(\mathbf{z}, F_{\text{center}}(\boldsymbol{x}_i))$        (candidate adversarial perturbation)
    else
      $\mathbf{z}_i \leftarrow \boldsymbol{u}_i$
    if $\mathsf{dist}(\mathbf{z}_i, F_{\text{center}}(\boldsymbol{x}_i)) < \tau$ then
      Output $F(\boldsymbol{x}) + \mathbf{z}_i$
Output “no adversarial example”
```

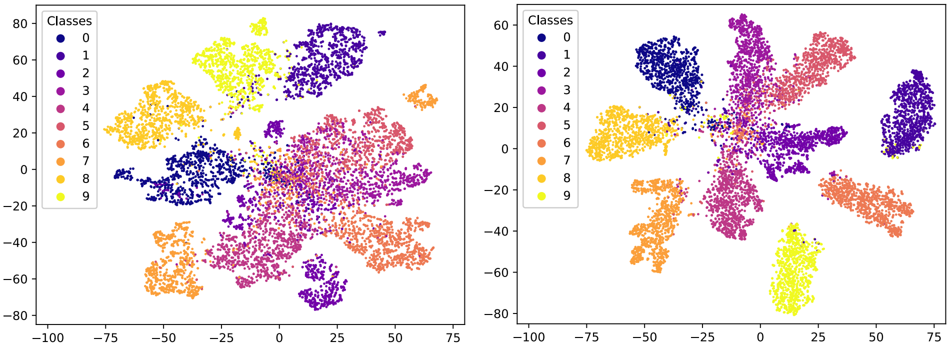

Figure 9: Two-dimensional t-SNE visualization of 512-dimensional embedding by contrastive learning on the CIFAR10 test dataset. **Left Figure:** Self-supervised contrastive learning. **Right Figure:** Supervised contrastive learning.

Otherwise there is some $\boldsymbol{w}_j$ closer to $\boldsymbol{u}_i$ than $\boldsymbol{x}_i$. Let H be the convex polytope of points closer to $\boldsymbol{x}_i$ than $\boldsymbol{w}_j$'s in C. Consider the intersection A of H with $\mathcal{S}$. All points in A are misclassified by our algorithm, if within the threshold τ. $\mathbf{v}$ must lie within A since it is closer to $\boldsymbol{x}_i$. $\boldsymbol{u}_i$ must lie outside of A in this case. If $\mathbf{v}$ is within τ of $\boldsymbol{x}_i$, so is $\boldsymbol{u}_i$ and therefore also the line joining the two. If this line intersects A at point $\mathbf{v}$, then $\mathbf{v}$ is a valid adversarial point and so is point closest to $\boldsymbol{x}_i$ in A. This proves completeness of the algorithm, soundness is more straightforward to verify.

L Additional Experiments

L.1 Visualization of representations of contrastive learning

Figure 9 shows the two-dimensional t-SNE visualization of 10,000 features by minimizing loss (1) on the CIFAR10 test dataset. It shows that $\tau_x \ll r_x$ for most of data, where $\tau_x := \min_{i:y=y_i} \mathsf{dist}(F(\boldsymbol{x}), F(\boldsymbol{x}_i))$, $r_x := \min_{i:y\neq y_i} \mathsf{dist}(F(\boldsymbol{x}), F(\boldsymbol{x}_i))$, and $\{\boldsymbol{x}_i\}_{i=1}^m$ are a set of training example with label y_i.

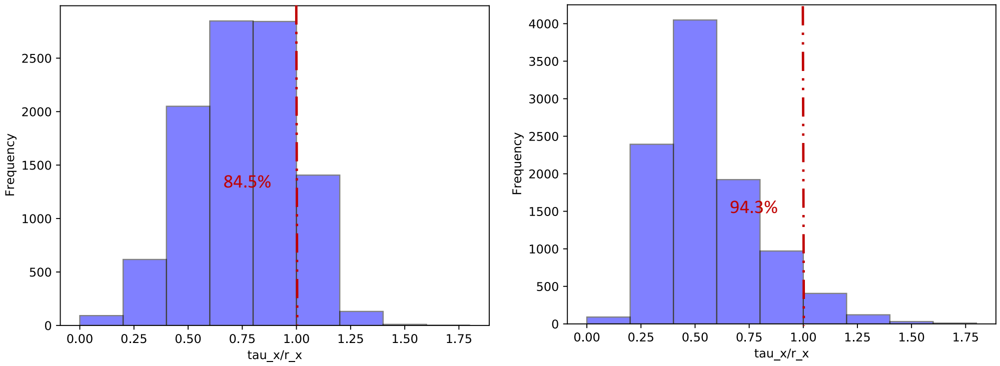

Figure 10: Frequency of τ_x/r_x by contrastive learning on the CIFAR10 dataset, where τ_x represents the closest distance between the test embedding and any training embedding of the same label, and r_x stands for the closest distance between the test embedding and any training embedding of different labels. **Left Figure:** Self-supervised contrastive learning. **Right Figure:** Supervised contrastive learning.

To have a closer look at τ_x vs. r_x, we plot the frequency of τ_x/r_x in Figure 10. For self-supervised contrastive learning, there are 84.5% data which has τ_x/r_x smaller than 1.0, while for supervised setting, there are 94.3% data which has τ_x/r_x smaller than 1.0.

L.2 Approximating robust accuracy for large n_3

The experiments in Section 7 consider an adversary which is difficult to compute in practice for large adversarial space, i.e. large n_3. In this section we present a 'greedy' adversary (Algorithm 5) which provides a good approximation to the exact adversary for small τ, which can be easily run even for large n_3: we can generate the adversarial examples of $F(\boldsymbol{x})$ by projecting each training feature onto the affine subspace $F(\boldsymbol{x}) + S$ and pick the one with the closest distance to $F(\boldsymbol{x})$. We denote the accuracy against this algorithm as $\hat{\mathcal{A}}_{\text{adv}}$. The averaged results of multiple runs are in Table 2: we report the *natural accuracy* ($\mathcal{A}_{\text{nat}} = 1 - \mathcal{E}_{\text{nat}}$), the *adversarial accuracy*, and the *abstention rate*, where the *abstention rate* represents the fraction of algorithm's output of "don't know" among the misclassified data by the nearest-neighbor classifier.

Algorithm 5 Approximate computation of attacks under threat model 3.1 against Algorithm 1

```
**Input:** A randomly-sampled adversarial subspace $S$ of dimension $n_3$, a test feature $F(\boldsymbol{x})$ and its label $y$, a set of training features $F(\boldsymbol{x}_i)$ and their labels $y_i$, $i \in [m]$, a threshold parameter $\tau$.
**Output:** A misclassified adversarial feature $F(\boldsymbol{x}) + \mathbf{v}$, $\mathbf{v} \in S$ if it exists; otherwise, output "no adversarial example".
$F_{\text{center}}(\boldsymbol{x}_i) \leftarrow F(\boldsymbol{x}_i) - F(\boldsymbol{x})$ for $i \in [m]$
// The $F_{\text{proj}}(\boldsymbol{x}_i)$'s in the next step are candidate adversarial examples
Project $F_{\text{center}}(\boldsymbol{x}_i)$, $i \in [m]$ onto $S$ and obtain $F_{\text{proj}}(\boldsymbol{x}_i)$ for $i \in [m]$
**for** $i = 1, ..., m$ **do**
    Run the nearest-neighbor algorithm to predict the label of $F_{\text{proj}}(\boldsymbol{x}_i)$ with the training set $\{(F_{\text{center}}(\boldsymbol{x}_j), y_j) : j = 1, ..., m\}$
    **if** the output of the nearest-neighbor algorithm is NOT $y$ **and** the closest distance is smaller than $\tau$ **then**
        Output $F(\boldsymbol{x}) + F_{\text{proj}}(\boldsymbol{x}_i)$
        Terminate the algorithm
Output "no adversarial example"
```

We observe that as the dimension of adversarial subspaces n_3 increases, the adversarial accuracy $\hat{\mathcal{A}}_{\text{adv}}$ decreases while the abstention rate tends to increase, which verifies an intrinsic trade-off between robustness and abstention rate. Recall that our algorithm abstains if and only if the closest distance between the given test feature and any training feature is larger than a threshold τ. As the threshold

Table 2: Natural accuracy $\mathcal{A}_{\text{nat}}$ and adversarial accuracy $\hat{\mathcal{A}}_{\text{adv}}$ on the CIFAR-10 dataset when the 512-dimensional representations are learned by contrastive learning, where *abstain* represents the fraction of each algorithm's output of "don't know" among the misclassified data by ours ($\tau \to \infty$, a.k.a. the nearest-neighbor classifier).

	Contrastive	Linear Protocol		Ours ($\tau \to \infty$)		Ours ($\tau = 1.0$)		Ours ($\tau = 0.8$)	
		$\mathcal{A}_{\text{nat}}$	$\hat{\mathcal{A}}_{\text{adv}}$	$\mathcal{A}_{\text{nat}}$	$\hat{\mathcal{A}}_{\text{adv}}$	$\mathcal{A}_{\text{nat}}$/abstain	$\hat{\mathcal{A}}_{\text{adv}}$/abstain	$\mathcal{A}_{\text{nat}}$/abstain	$\hat{\mathcal{A}}_{\text{adv}}$/abstain
$n_3 = 1$	Self-supervised	91.1%	0.0%	84.5%	81.5%	99.3%/95.5%	99.2%/95.7%	100.0%/100.0%	100.0%/100.0%
	Supervised	94.4%	0.0%	94.3%	93.5%	95.0%/12.3%	94.5%/15.4%	97.7%/59.6%	97.7%/64.6%
$n_3 = 25$	Self-supervised	91.1%	0.0%	84.5%	65.1%	99.3%/95.5%	98.8%/96.6%	100.0%/100.0%	100.0%/100.0%
	Supervised	94.4%	0.0%	94.3%	84.5%	95.0%/12.3%	91.6%/45.8%	97.7%/59.6%	96.8%/79.4%
$n_3 = 50$	Self-supervised	91.1%	0.0%	84.5%	56.3%	99.3%/95.5%	98.3%/96.1%	100.0%/100.0%	100.0%/100.0%
	Supervised	94.4%	0.0%	94.3%	71.7%	95.0%/12.3%	89.7%/63.6%	97.7%/59.6%	95.5%/84.1%
$n_3 = 100$	Self-supervised	91.1%	0.0%	84.5%	31.1%	99.3%/95.5%	96.7%/95.2%	100.0%/100.0%	99.7%/99.6%
	Supervised	94.4%	0.0%	94.3%	35.0%	95.0%/12.3%	86.3%/78.9%	97.7%/59.6%	93.0%/89.2%
$n_3 = 200$	Self-supervised	91.1%	0.0%	84.5%	1.2%	99.3%/95.5%	91.1%/91.0%	100.0%/100.0%	98.6%/98.6%
	Supervised	94.4%	0.0%	94.3%	0.7%	95.0%/12.3%	74.7%/74.5%	97.7%/59.6%	85.8%/85.7%

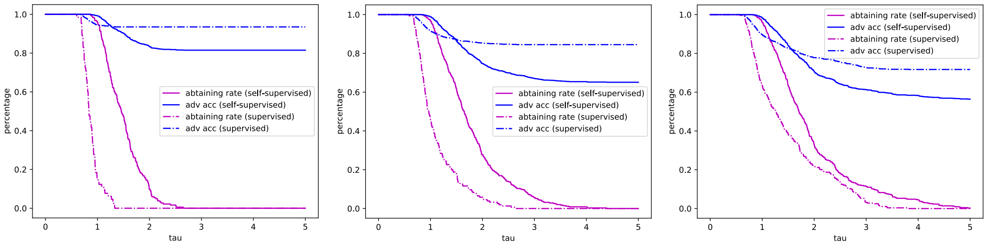

Figure 11: Sensitivity of model success rate (estimated by $\hat{\mathcal{A}}_{\text{adv}}$) and abstention rate on the parameter τ, where *abstain* represents the fraction of algorithm's output of "don't know" among the misclassified data by ours ($\tau \to \infty$, a.k.a. the nearest-neighbor classifier). **Left Figure:** $n_3 = 1$. **Middle Figure:** $n_3 = 25$. **Right Figure:** $n_3 = 50$.

parameter τ decreases, the adversarial accuracy $\hat{\mathcal{A}}_{\text{adv}}$ increases while the algorithm abstains from predicting the class of more data.

L.2.1 Sensitivity of threshold parameter τ

The threshold parameter τ is an important hyperparameter in our proposed method. It captures the trade-off between the accuracy and the abstention rate. We show how the threshold parameter affects the performance of our robust classifiers by numerical experiments on the CIFAR-10 dataset. We first train a embedding function F by following the setups in Section 7.1. We then fix F and run our evaluation protocol by varying τ from 0.0 to 5.0 with step size 0.001. We summarize our results in Figure 11 which plots the adversarial accuracy $\hat{\mathcal{A}}_{\text{adv}}$ and the abstention rate for three representative dimension of adversarial subspace. Compared with self-supervised contrastive learning (the solid line), supervised contrastive learning (the dashed line) enjoys higher adversarial accuracy (the blue curve) and smaller abstention rate (the red curve) for fixed τ's due to the use of extra label information. For both setups, the adversarial accuracy is not very sensitive to the choice of τ.