

On the Robustness of Support Vector Machines against Adversarial Examples

Peter Langenberg and Emilio Balda and Arash Behboodi and Rudolf Mathar
Institute for Theoretical Information Technology
RWTH Aachen University, D-52056 Aachen, Germany

Abstract—In this paper, the robustness of Support Vector Machines (SVMs) against adversarial instances is considered in relation to the design parameters. After generating adversarial instances using convex programming, it is shown through extensive numerical analysis that the robustness is significantly affected by parameters which change the linearity of the models. Interestingly, robustness is only slightly sensitive to the parameter determining the margin between classes. It is shown that adversarial robustness not only depends on the geometric properties of the classifier but is also subject to the accuracy of the model. The results are discussed in the light of the so-called linearity hypothesis, regarding adversarial robustness of machine learning algorithms.

I. INTRODUCTION

Despite the success of machine learning algorithms and particularly Deep Neural Networks (DNNs) in various tasks [1], [2], [3], it is well known that they are vulnerable to so called adversarial perturbations [4], [5]. Adversarial perturbations are purposefully designed and added to input images. They mislead classifiers to decide for an incorrect class with high confidence [6], [7], [8]. It is peculiar for these perturbations that they are hardly visible to the human eye, and even if they can be spotted, it is difficult to infer how they change the classifier’s output. Given the sensitivity of safety critical systems to these examples, many researchers focused on understanding the nature of adversarial instances for machine learning applications [9], [10], [11], [12].

There are different ways of generating adversarial examples. In [5], adversarial perturbations are obtained such that the classifier output is changed to a target class by additional constraints on the minimum ℓ_2 -norm perturbation. A non-convex optimization problem, the solution is approximated using a box-constrained limited memory Broyden-Fletcher-Goldfarb-Shanno algorithm (L-BFGS). Another method consists of maximizing a certain function that is related to the classification error of the classifier. The algorithm Fast Gradient Sign Method (FastGrad) in [6] maximizes the output perturbation for the approximate linearized version of a DNN under some ℓ_∞ -norm constraint on the input perturbation. The linearization idea was used further for proposing other methods as in DeepFool [7]. The effectiveness of this method gave rise to the linearity hypothesis about the nature of adversarial instances. As stated in [6], the linearity hypothesis attributes the existence of adversarial images to the approximate linearity of classifiers. There are many discussions about the validity of this assumption as it can be seen in [12], [13]. There are other

theories focusing mostly on decision boundaries of classifiers and their analytic properties [14], [15].

There are some difficulties in understanding the nature of adversarial instances for DNNs. One particular reason is that the interpretation of these models is challenging and therefore it is difficult to see which design parameters matter most for robustness. Although the focus has been on robustness of DNNs, a similar study was initiated in [16] for SVMs, which are more convenient for studying the interrelation of robustness and design parameters.

In this work, we continue this line of work by considering robustness of SVMs against adversarial instances. We consider SVMs with polynomial and radial basis function (RBF) kernels and study their robustness as a function of parameters like the degree of polynomials, the RBF kernel exponent or the parameter implicitly determining the margin of SVM. The adversarial instances are generated in a similar fashion to [7], [6], and formulated as optimization problems.

By extensive numerical simulations it can be seen that parameters like the polynomial degree visibly affect the robustness while other parameters do not have a similar effect. Note that the attacks in this paper are based on linearization of the model, which implicitly means that robustness in this work is meant against certain classes of algorithms that are based on linearization. Hence, it is not surprising that non-linear kernels are more robust against these attacks since the linear approximation is not accurate. It can also be seen that the adversarial perturbations are easier to detect with the naked eye for non-linear SVMs.

However, the above conclusion should be pondered with care. First of all, the robustness cannot be considered in isolation from accuracy. In other words, lower accuracy classifiers can be more robust, as it can be seen for RBF kernels. Increasing the RBF kernel exponent is a way to make the classifier more non-linear. Although non-linearity improves robustness, accuracy finally drops by increasing the exponent. If the exponent becomes very large, even a small perturbation can change the class labels significantly.

After introducing the framework and formulation of the adversarial generation problem, we discuss our above conclusions in detail.

II. GENERATING ADVERSARIAL EXAMPLES FOR SVMs

In this section, we introduce the framework that is used to generate adversarial examples for SVMs as an adaptation of

the algorithms proposed in [7]. We use the short-hand notation $[A]$ to denote the set $[A] = \{1, 2, \dots, A\}$ for some integer $A \in \mathbb{N}$. Further, let $\mathbf{X} \in \mathbb{R}^D$ be a D -dimensional random variable and $Y \in [L]$ its corresponding label for $L > 1$. The goal of a classifier is to predict Y given \mathbf{X} . In order to tune such a classifier, a training set composed of M independent realizations of (\mathbf{X}, Y) , denoted by $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_M, y_M)\}$, is used.

We study the effect of hyper-parameter selection and non-linearity of the classifier function on the robustness of SVMs against adversarial examples. More precisely, we are interested to investigate the behavior of SVMs for two widely used kernels, namely the polynomial kernel

$$K_{\text{poly}}(\mathbf{x}_i, \mathbf{x}_j) = (\zeta \mathbf{x}_i^\top \mathbf{x}_j + 1)^d \text{ with } d \in \mathbb{N}, \zeta \in \mathbb{R}_+, \quad (1)$$

and the Radial Basis Function (RBF) kernel

$$K_{\text{rbf}}(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|_2^2) \text{ with } \gamma \in \mathbb{R}_+. \quad (2)$$

In a one-vs.-rest scheme, the proxy functions f_l are trained separately as binary "l-vs.-rest" classifiers. For this purpose binary l-vs.-rest labels are constructed as $y_i^{(l)} := \text{sign}(\mathbf{1}(y == l) - \frac{1}{2}) \in \{-1, 1\}$. Then, training a SVM amounts to solving the dual problem

$$\begin{aligned} \max_{\lambda^{(l)}} \quad & \sum_{i=1}^M \lambda_i^{(l)} - \frac{1}{2} \sum_{i,j} y_i^{(l)} y_j^{(l)} \lambda_i^{(l)} \lambda_j^{(l)} K(\mathbf{x}_i, \mathbf{x}_j) \\ \text{s. t.} \quad & 0 \leq \lambda_i^{(l)} \leq C \quad \forall i \in [M] \\ & \sum_{i=1}^M y_i^{(l)} \lambda_i^{(l)} = 0 \end{aligned} \quad (3)$$

for all $l \in [L]$. There is a vast existing literature on methods for efficiently solving this optimization problem. Moreover, the solution of this problem yields $\lambda_i^{(l)*}$, which are used for computing the SVM parameters as $\alpha_i^{(l)} = \lambda_i^{(l)} y_i^{(l)}$ and

$$\mathbf{w}_i^{(l)} = \begin{cases} \mathbf{x}_i, & \lambda_i^{(l)*} > 0 \\ \mathbf{0}, & \lambda_i^{(l)*} = 0 \end{cases} \text{ for all } i \in [M], l \in [L]. \quad (4)$$

Note that only the vectors $\mathbf{w}_i^{(l)}$ such that $\mathbf{w}_i^{(l)} \neq \mathbf{0}$ are needed to compute the parameters. These vectors are called support vectors.

In the same spirit as [7] we construct adversarial examples $\tilde{\mathbf{x}} \in \mathbb{R}^D$ as perturbed versions of \mathbf{x} . We consider additive adversarial perturbation $\mathbf{r} \in \mathbb{R}^D$, so that $\tilde{\mathbf{x}} = \mathbf{x} + \mathbf{r}$. For a given \mathbf{x} some classifier k is said to be fooled by the adversarial example if $k(\mathbf{x}) \neq k(\tilde{\mathbf{x}})$. The problem of finding a perturbation \mathbf{r} such that $k(\mathbf{x}) \neq k(\tilde{\mathbf{x}})$ can be written as determining some \mathbf{r} such that

$$\min_{l \neq k(\mathbf{x})} \{f_{k(\mathbf{x})}(\mathbf{x} + \mathbf{r}) - f_l(\mathbf{x} + \mathbf{r})\} < 0. \quad (5)$$

If there is no constraint on \mathbf{r} , fooling a classifier is trivial.

In the literature, adversarial examples are considered a threat to the stability of the system if they are indistinguishable from undisturbed inputs by a human observer. This is usually

modeled by constraining the ℓ_p norm of the perturbation for some $p \geq 1$. A popular choice is $p = \infty$, which corresponds to a box constraint on \mathbf{r} . Including this constraint in (5), the problem of finding optimal adversarial perturbations reads as

$$\begin{aligned} \min_{l \neq k(\mathbf{x})} \min_{\mathbf{r}} \quad & \theta_l(\mathbf{x} + \mathbf{r}) \\ \text{s. t.} \quad & \|\mathbf{r}\|_\infty \leq \varepsilon, \end{aligned} \quad (6)$$

where $\theta_l(\mathbf{x}) := f_{k(\mathbf{x})}(\mathbf{x}) - f_l(\mathbf{x})$.

Solving (6) is challenging since $\theta_l(\mathbf{x})$ is in general non-convex. Hence, we relax the problem by linearly approximating $\theta_l(\mathbf{x} + \mathbf{r}_l)$ by its first-order Taylor expansion, that is $\theta_l(\mathbf{x} + \mathbf{r}_l) \approx \theta_l(\mathbf{x}) + \mathbf{r}_l^\top \nabla \theta_l(\mathbf{x})$. This leads to the approximate version of generating adversarial examples by solving

$$\begin{aligned} \min_{l \neq k(\mathbf{x})} \min_{\mathbf{r}} \quad & \{\theta_l(\mathbf{x}) + \mathbf{r}^\top \nabla \theta_l(\mathbf{x})\} \\ \text{s. t.} \quad & \|\mathbf{r}\|_\infty \leq \varepsilon. \end{aligned} \quad (7)$$

For fixed $l \neq k(\mathbf{x})$ the problem

$$\mathbf{r}_l = \underset{\mathbf{r}}{\text{argmin}} \{\theta_l(\mathbf{x}) + \mathbf{r}^\top \nabla \theta_l(\mathbf{x})\} \text{ s. t. } \|\mathbf{r}\|_\infty \leq \varepsilon$$

has a closed form solution $\mathbf{r}_l = -\varepsilon \text{sign}(\nabla \theta_l(\mathbf{x}))$. Therefore, since $\theta_l(\mathbf{x}) + \mathbf{r}_l^\top \nabla \theta_l(\mathbf{x}) = \theta_l(\mathbf{x}) - \varepsilon \|\nabla \theta_l(\mathbf{x})\|_1$, the relaxed problem (7) has the closed-form solution

$$l^* = \underset{l \in [L]}{\text{argmin}} \theta_l(\mathbf{x}) - \varepsilon \|\nabla \theta_l(\mathbf{x})\|_1 \quad (8)$$

$$\mathbf{r}^* = -\varepsilon \text{sign}(\nabla \theta_{l^*}(\mathbf{x})). \quad (9)$$

In order to compute (9) the gradients ∇f_l for all $l \in [L]$ are needed since

$$\nabla \theta_l(\mathbf{x}) = \nabla f_{k(\mathbf{x})}(\mathbf{x}) - \nabla f_l(\mathbf{x}).$$

In case of the polynomial kernel the gradients are given by

$$\nabla f_l(\mathbf{x}) = d \zeta \sum_{i=1}^M \alpha_i^{(l)} (\zeta \mathbf{w}_i^{(l)\top} \mathbf{x} + 1)^{d-1} \mathbf{w}_i^{(l)}, \quad (10)$$

while for the RBF kernel we have

$$\nabla f_l(\mathbf{x}) = -2\gamma \sum_{i=1}^M \alpha_i^{(l)} (\mathbf{x} - \mathbf{w}_i^{(l)}) \exp(-\gamma \|\mathbf{x} - \mathbf{w}_i^{(l)}\|_2^2). \quad (11)$$

For continuous differentiable functions and sufficiently small ε the solution of the relaxed problem (7) with linear approximations will be close to the one of the original problem (6). In general, however, one may encounter highly non-linear functions or large values of ε , entailing inaccurate approximations of (6). This problem is addressed in [7] by iteratively solving approximate versions (6) in $T \in \mathbb{N}$ steps with constraint ε/T . The next anchor point for the linear approximation is the previous one shifted by the optimal perturbation vector of ℓ_∞ -norm less than ε/T . The final perturbation vector is obtained as the sum of all intermediate perturbations. Algorithm 1 gives the corresponding details. By this approach a neighborhood of the vector to be fooled is searched and approximate solutions of (6) even for large values of ε and highly non-linear classifiers can be obtained.

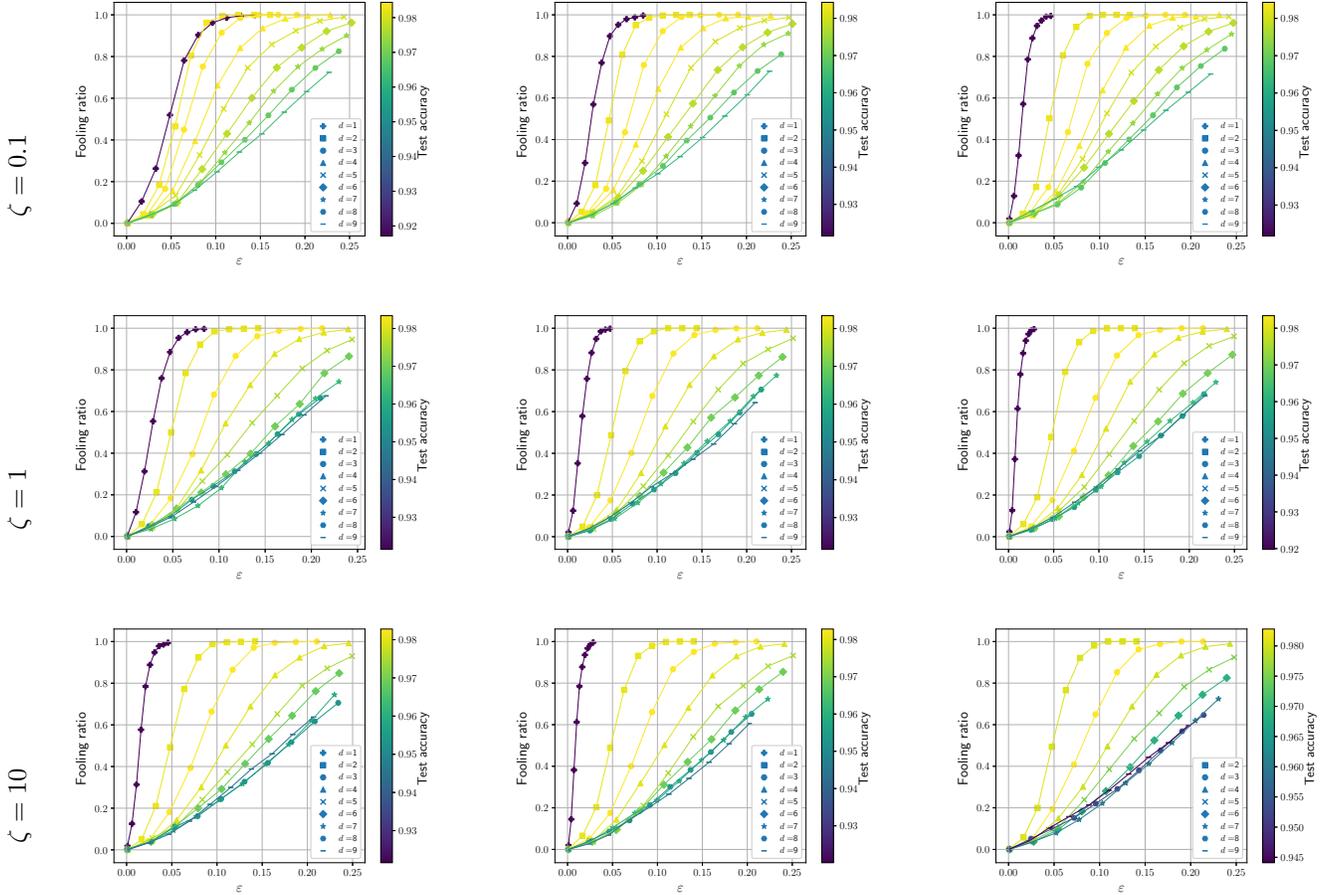
$C = 0.1$ $C = 1$ $C = 10$ 

Fig. 1. Fooling ratios for polynomial kernel SVMs with variation of $d \in \{1, 2, \dots, 9\}$, $C \in \{0.1, 1, 10\}$ and $\zeta \in \{0.1, 1, 10\}$. The test accuracy is indicated by the color bar.

Algorithm 1 Adversarial noise design through iterative approximations

input: $\mathbf{x}, f_1, \dots, f_L, T, \varepsilon$.

output: \mathbf{r}^* .

Initialize $\mathbf{x}^{(1)} \leftarrow \mathbf{x}$.

for $t = 1, \dots, T$ **do**

$l^* \leftarrow \underset{l \in [L]}{\operatorname{argmin}} \theta_l(\mathbf{x}^{(t)}) - \frac{\varepsilon}{T} \|\nabla \theta_l(\mathbf{x}^{(t)})\|_1$

$\mathbf{r}^{(t)*} \leftarrow -\frac{\varepsilon}{T} \operatorname{sign}(\nabla \theta_{l^*}(\mathbf{x}^{(t)}))$

$\mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)} + \mathbf{r}^{(t)*}$

end for

return: $\mathbf{r}^* \leftarrow \sum_{t=1}^T \mathbf{r}^{(t)*}$

III. EXPERIMENTAL RESULTS FOR THE MNIST DATA

In this section, we present our experimental results on the 10-class MNIST handwritten digit dataset [17]. This dataset contains 55 000 gray-scale images of dimension 28×28 for training and 10 000 for testing. Images are vectorized to 784-dimensional vectors. All classifiers are trained using the LibSVM [18] implementation.

The fooling ratio [7] is employed as an empirical measure of the sensitivity of classifiers against adversarial perturbations for a given upper bound $\|\mathbf{r}\|_\infty \leq \varepsilon$. Adversarial perturbations are computed using Algorithm 1.

The fooling ratio is defined as the percentage of correctly classified unperturbed images whose perturbed versions, given a certain ε , are incorrectly classified. This ratio is calculated on a subset of 1000 randomly chosen images from the test set.

We first study the impact of the polynomial kernel degree $d \in \{1, 2, \dots, 9\}$ on the fooling ratio. The kernel degree allows for curved decision boundaries such as quadratic shapes for $d = 2$ (parabolas, hyperbolas, ellipses) and more complex shapes for $d > 2$. Further, we study how the inverse regularization parameter C and the polynomial kernel parameter ζ affect the fooling ratio by computing results for $C \in \{0.1, 1, 10\}$ and $\zeta \in \{0.1, 1, 10\}$. The choice of C determines the trade-off between misclassification and simplicity of the decision boundary. For large C , a decision boundary with smaller margin will be obtained by the optimization if more training samples can be correctly classified. On the other hand, for

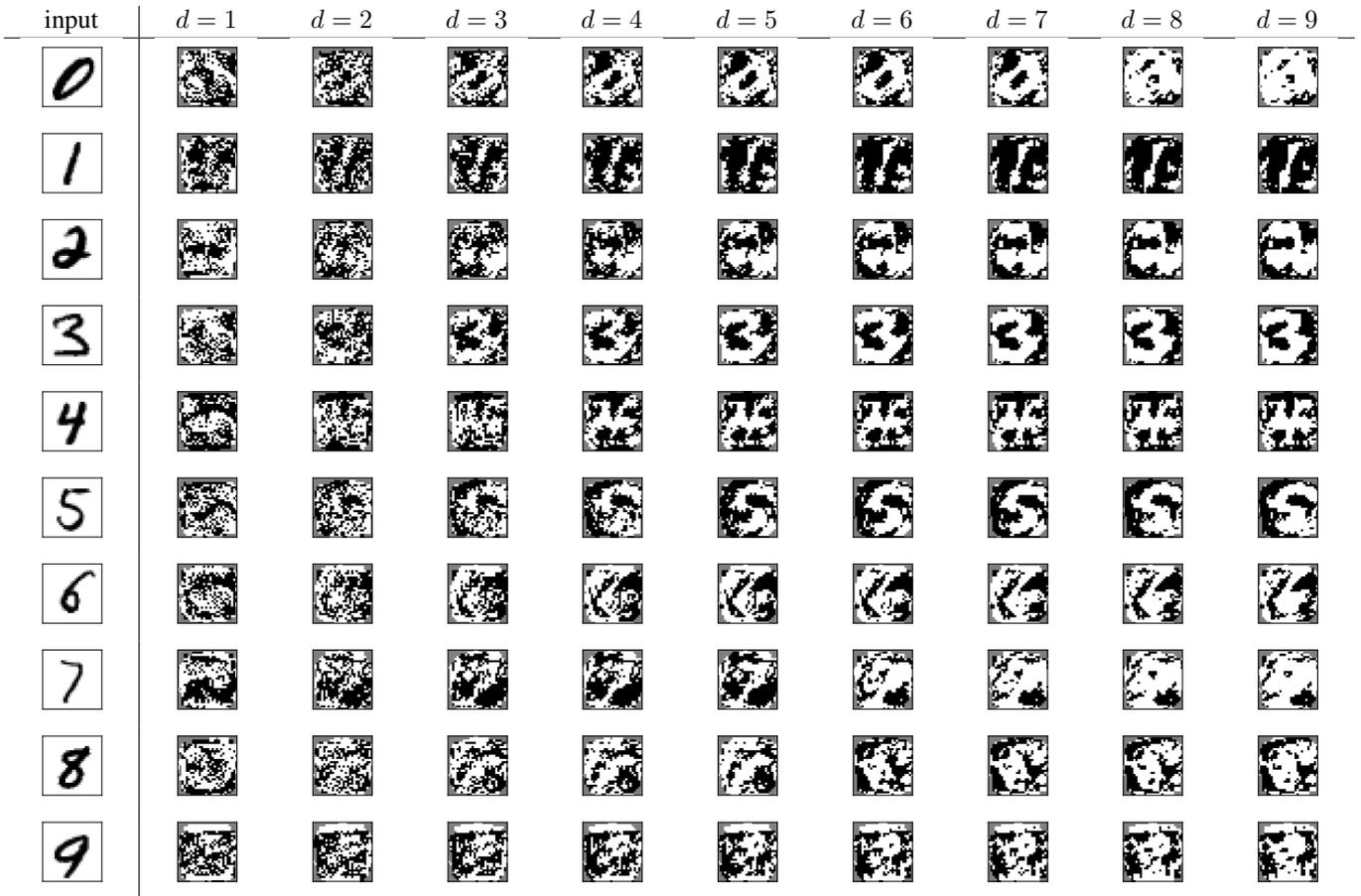


Fig. 2. Illustration of sample images and the corresponding adversarial noise computed for models with different values of d and fixed $\varepsilon = 0.25$. In the additive noise images, black pixels refer to $+\varepsilon$, white pixels to $-\varepsilon$, and gray pixels to 0.

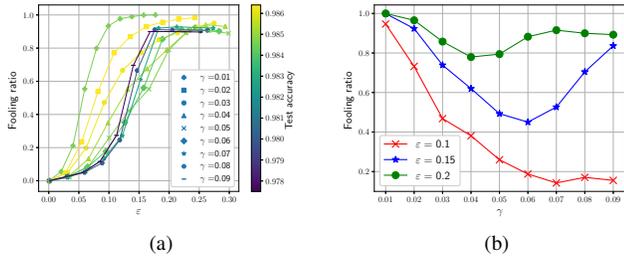


Fig. 3. Resulting fooling ratio for RBF kernel SVM with variation of $\gamma \in \{0.01, 0.02, \dots, 0.09\}$ and fixed $C = 10$.

small values of C , a larger margin and smoother decision boundary will be obtained by the optimization, at the cost of more misclassified training samples.

The results for the polynomial kernel are shown in Figure 1. The color bar indicates the test accuracy for each setup. Increasing $d \in \{1, 2, \dots, 6\}$ improves robustness indicated by a decrease of the fooling ratio. This observation is aligned with our expectation that increasing non-linearity of a decision boundary also increases the robustness against adversarial examples, which supports the linear hypothesis of [6]. In other

words, linear modeling in SVMs results in weak robustness against adversarial examples.

Nevertheless, it can be seen that the decrease in the fooling ratio becomes rather insignificant if $d \in \{7, 8, 9\}$ is further increased. This result shows that increasing the non-linearity of the model does not necessarily lead to higher robustness. We even observe that for some setups such as $(C, \zeta) \in \{(0.1, 10), (10, 10)\}$ the fooling ratio increases when increasing $d \in \{7, 8, 9\}$. Note that the increase of fooling ratio for $d \in \{7, 8, 9\}$ is most visible for $\zeta = 10$, while it is not visible for $\zeta = 0.1$. We conjecture that this effect may be partially caused by a substantial performance drop in terms of test accuracy of the classifier (compare test accuracies for $d \in \{7, 8, 9\}$ to those with $d < 7$).

Further, we see that C has the most impact on the fooling ratio for small d . In particular, in the linear and quadratic cases $d \in \{1, 2\}$ an increasing fooling ratio is observed as C increases. The impact of C on the fooling ratio becomes less significant as d becomes larger. It can be argued that for large d the classifier is already quite flexible so that an increase of C does not lead to fewer misclassifications in the training set and therefore does not lead to a smaller margin classifier with larger fooling ratio.

We observe a very similar impact of ζ . For small d ,

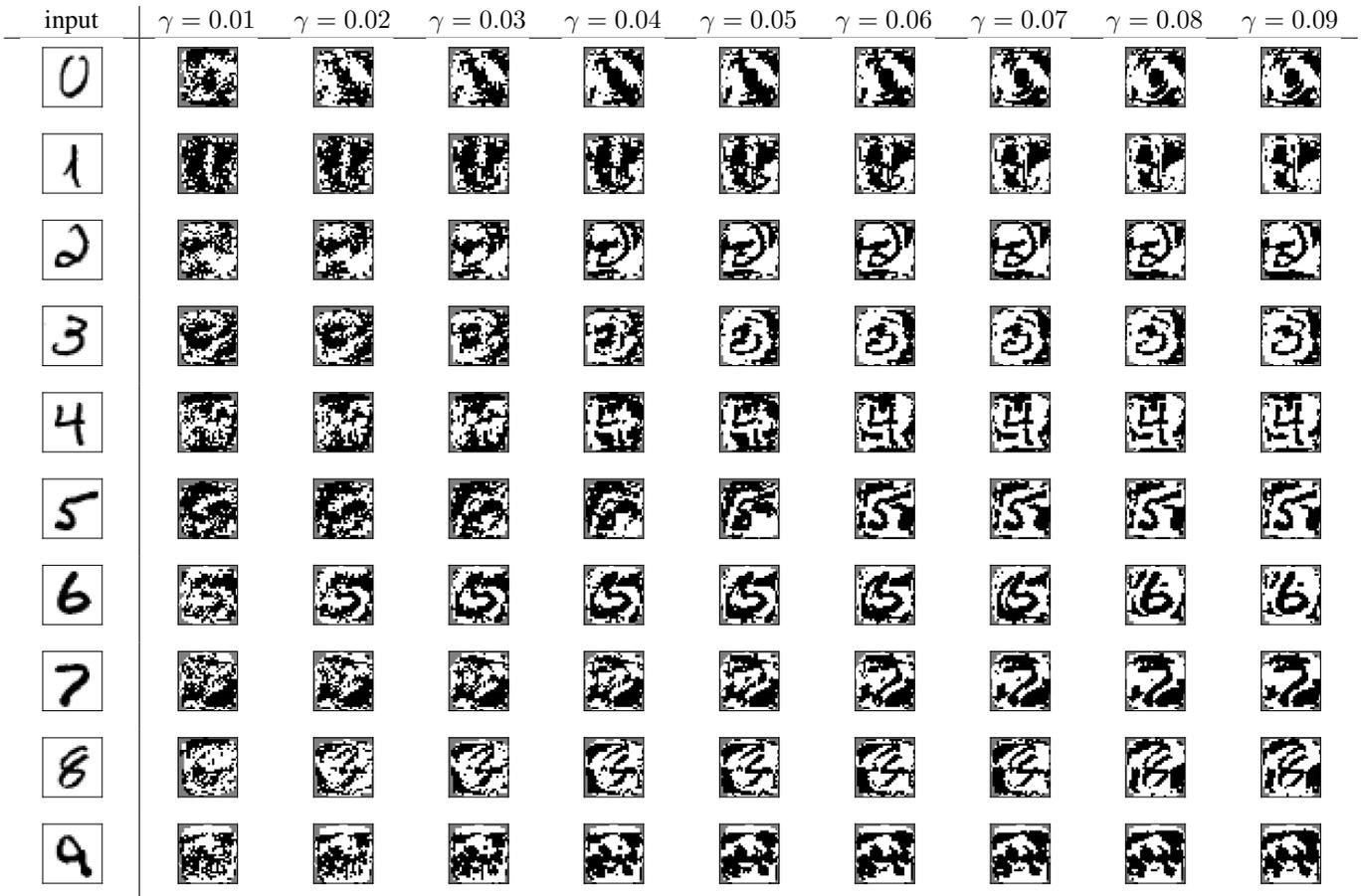


Fig. 4. Illustration of sample images and the corresponding adversarial noise for models with different values of γ and fixed $\varepsilon = 0.15$. In the additive noise pictures, black pixels refer to $+\varepsilon$, white pixels to $-\varepsilon$, and gray pixels to 0.

enlarging ζ substantially increases the fooling ratio while for larger d this effect is rather insignificant. We find that the least robust SVMs are obtained for small d , and large C and ζ . We hence conjecture that the classifier with $(C, \zeta, d) = (10, 10, 1)$ is least robust.

Our analysis suggests that C and ζ need to be chosen carefully for small d in order to achieve robustness while their impact is rather insignificant for rather non-linear classifiers as d gets larger. Figure 2 shows adversarial perturbations for different choices of d . It is interesting to see that clear patterns arise in the adversarial perturbations as models become more non-linear.

Figure 3 shows a similar study for the RBF kernel with varying $\gamma \in \{0.01, \dots, 0.09\}$ and fixed $C = 10$. Recall the inverse relation $\gamma = 1/(2\sigma^2)$ where σ denotes the standard deviation of the RBF kernel, i.e., enlarging γ decreases the kernel width. We observe that the fooling ratio initially decreases with increasing $\gamma \in \{0.01, 0.02, \dots, 0.05\}$, while moving backwards towards a larger fooling ratio when further increasing $\gamma \in \{0.06, \dots, 0.09\}$. Interestingly, the RBF kernel parameter γ seems to have a similar impact on the fooling ratio as d in case of the polynomial kernel.

We also visualize the adversarial noise for fixed $\varepsilon = 0.15$ and different choices of γ (see Figure 4), where the same

trend as in Figure 2 is observed. Again, very similar to what has been observed in the case of polynomial kernels, we see that the shape of the adversarial noise aligns to the shape of the unperturbed input as γ increases.

IV. CONCLUSIONS

In the literature, adversarial noise for artificial neural networks is extensively investigated. There is much less work on how support vector machines react to perturbations which are intentionally crafted for misleading classification. In the present paper, we first derive a mathematical framework for constructing adversarial noise, which cannot be detected by a human observer. To circumvent the hardness of the corresponding optimization problem, the objective function is approximated by its first order Taylor expansion. The relaxed problem allows for explicit solutions. On the basis of this framework extensive numerical tests are carried out, and visualized for the MNIST data set of handwritten digits. As a general tendency we observe that increasing non-linearity in the polynomial and radial basis function kernels provides improved robustness against adversarial attacks. An analytically quantified description of this relation is still missing and will be a topic of future research.

REFERENCES

- [1] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, Nov. 2012.
- [2] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.
- [3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778.
- [4] M. Barreno, B. Nelson, A. D. Joseph, and J. D. Tygar, "The security of machine learning," *Machine Learning*, vol. 81, no. 2, pp. 121–148, Nov. 2010.
- [5] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *International Conference on Learning Representations*, 2014, arXiv: 1312.6199.
- [6] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples," in *International Conference on Learning Representations*, Dec. 2014.
- [7] S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2574–2582, 2016.
- [8] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," *arXiv preprint arXiv:1610.08401*, 2016.
- [9] N. Akhtar and A. Mian, "Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey," *arXiv:1801.00553 [cs]*, Jan. 2018, arXiv: 1801.00553. [Online]. Available: <http://arxiv.org/abs/1801.00553>
- [10] B. Wang, J. Gao, and Y. Qi, "A Theoretical Framework for Robustness of (Deep) Classifiers against Adversarial Examples," in *International Conference on Learning Representations*, 2017, arXiv: 1612.00334.
- [11] A. Fawzi, O. Fawzi, and P. Frossard, "Fundamental limits on adversarial robustness," *Proceedings of ICML, Workshop on Deep Learning*, 2015.
- [12] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, P. Frossard, and S. Soatto, "Robustness of Classifiers to Universal Perturbations: A Geometric Perspective," in *International Conference on Learning Representations*, 2018.
- [13] T. Tanay and L. Griffin, "A Boundary Tilting Perspective on the Phenomenon of Adversarial Examples," *arXiv:1608.07690 [cs, stat]*, Aug. 2016, arXiv: 1608.07690. [Online]. Available: <http://arxiv.org/abs/1608.07690>
- [14] A. Fawzi, S.-M. Moosavi-Dezfooli, and P. Frossard, "Robustness of classifiers: from adversarial to random noise," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Curran Associates, Inc., 2016, pp. 1632–1640.
- [15] A. Fawzi, S. M. Moosavi-Dezfooli, and P. Frossard, "The Robustness of Deep Networks: A Geometrical Perspective," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 50–62, Nov. 2017.
- [16] A. Fawzi, O. Fawzi, and P. Frossard, "Analysis of classifiers' robustness to adversarial perturbations," *Machine Learning*, vol. 107, no. 3, pp. 481–508, Mar. 2018.
- [17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [18] C. C. Chang and C. J. Lin, "Libsvm: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.