Robustness of Sketched Linear Classifiers to Adversarial Attacks

Ananth Mahadevan*
ananth.mahadevan@helsinki.fi
University of Helsinki
Helsinki, Finland

Arpit Merchant*
arpit.merchant@helsinki.fi
University of Helsinki
Helsinki, Finland

Yanhao Wang
yhwang@dase.ecnu.edu.cn
East China Normal University
Shanghai, China

Michael Mathioudakis
michael.mathioudakis@helsinki.fi
University of Helsinki
Helsinki, Finland

ABSTRACT

Linear classifiers are well-known to be vulnerable to adversarial attacks: they may predict incorrect labels for input data that are adversarially modified with small perturbations. However, this phenomenon has not been properly understood in the context of sketch-based linear classifiers, typically used in memory-constrained paradigms, which rely on random projections of the features for model compression. In this paper, we propose novel Fast-Gradient-Sign Method (FGSM) attacks for sketched classifiers in full, partial, and black-box information settings with regards to their internal parameters. We perform extensive experiments on the MNIST dataset to characterize their robustness as a function of perturbation budget. Our results suggest that, in the full-information setting, these classifiers are less accurate on unaltered input than their uncompressed counterparts but just as susceptible to adversarial attacks. But in more realistic partial and black-box information settings, sketching improves robustness while having lower memory footprint.

CCS CONCEPTS

• **Theory of computation → Sketching and sampling**; **Adversarial learning**.

KEYWORDS

Sketching; Robustness; Adversarial Machine Learning

ACM Reference Format:
Ananth Mahadevan, Arpit Merchant, Yanhao Wang, and Michael Mathioudakis. 2022. Robustness of Sketched Linear Classifiers to Adversarial Attacks. In *Proceedings of the 31st ACM Int'l Conference on Information and Knowledge Management (CIKM '22), Oct. 17–21, 2022, Atlanta, GA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3511808.3557687

1 INTRODUCTION

The resource-constrained learning paradigm aims to design classifiers that operate on high-dimensional feature spaces while having low memory footprint. It finds applications on small appliances like smart phones where memory-intensive language models for speech recognition [13] or computer vision models for facial recognition [11] are required to be trained locally with limited RAM. In this paradigm, sketching [7] is a useful tool for compression employed by classifiers to *(i)* reduce memory footprint by hashing features to lower dimensions and *(ii)* maintain accurate weight estimates of an equivalent (i.e., trained on the same data) uncompressed classifier. However, in many data mining problems such as frequency estimation [5], heavy hitters [6], and k-means clustering [12], sketches have been shown to be brittle to adversarial inputs [1, 2, 10]. Similarly, linear classifiers are also well-known to be susceptible to adversarial data [9, 16, 18]. That is, small perturbations of data points can alter sketch estimates or the predicted labels of an uncompressed linear classifier. Nonetheless, the sensitivity of sketched classifiers to adversarial data has not been well studied. Therefore, our goal is to analyze the extent to which sketching adds robustness to linear classifiers against small perturbations, if at all, while offering its original recovery guarantees.

To this end, we focus on a state-of-the-art sketching-based linear classifier, i.e., the Weight-Median Sketched Classifier [17] (that we refer to as WM-Sketch for short), as a compressed version of an online linear classifier (Learner). The WM-Sketch internally defines an initial random count sketch [5] and then updates it using gradient descent with a regularized loss function. At inference time, WM-Sketch predicts labels using either the count sketch directly or the weights in the original space of Learner recovered from the count sketch. As shown in [17], for appropriately chosen dimensions of the sketch matrix, the weights recovered from the sketched classifier closely approximate the weights of its uncompressed counterpart. This implies that adversarial perturbations designed for Learner may be adapted for WM-Sketch.

In general, attack algorithms for uncompressed learners can be broadly classified into three categories namely, *(i)* gradient-based attacks such as Fast Gradient-Sign Method [9] (FGSM), *(ii)* score-based attacks such as local-search method [14], and *(iii)* decision-based attacks such as boundary method [3]. We refer interested readers to the surveys by Chakraborty et al. [4] and Pitropakis et al. [15] for further details. In this paper, we focus on characterizing the robustness of WM-Sketch to FGSM-style adversarial attacks. This attack shifts the original feature vector by a small distance so as to position it on the opposite side of the decision boundary of the classifier being attacked. Generating adversarial examples in this manner requires a distance measure (e.g., L_∞-norm) that captures the size of the perturbation and complete knowledge of

*Both authors contributed equally to this research.

CIKM '22, October 17–21, 2022, Atlanta, GA, USA

ACM ISBN 978-1-4503-9236-5/22/10.
https://doi.org/10.1145/3511808.3557687

the classifier's internal parameters such as its weights, learning rate, etc. which might be a strong assumption in practice.

Our Contributions. For a target WM-Sketch, we define three adversaries with FGSM-style attack protocols. Depending on the knowledge each adversary has about the target, the protocol involves choosing a surrogate classifier, crafting a perturbed example based on the surrogate, and attacking the target with it. First, in the white-box setting, the adversary knows that the target is a WM-Sketch and has complete information about the target's internal parameters, including its count sketch matrix. Here, the surrogate is the target itself. Second, in the grey-box setting, the adversary knows that the target is a WM-Sketch and has complete information about all of WM-Sketch's internal parameters except for the count sketch matrix. Here, the adversary constructs a surrogate WM-Sketch classifier with a different sketch matrix. And third, in the black-box setting, the adversary does not know that the target is a WM-Sketch and hence has no access to the target's internal parameters. Thus, it constructs a Learner as a surrogate. In each case, the robustness of the target WM-Sketch is quantified by the accuracy on the adversarial input as a function of the perturbation budget. Our main contributions include:

- We analytically define FGSM-style attack protocols for WM-Sketch in the context of the three adversaries discussed above.
- We conduct experiments on MNIST data to empirically quantify the robustness of WM-Sketch against the three adversaries.
- In the white-box setting, we find that WM-Sketch is just as susceptible to adversarial attacks as an uncompressed classifier.
- In the grey-box and black-box settings, however, sketching provides WM-Sketch with improved robustness to adversarial attacks in addition to compression.

2 BACKGROUND: LINEAR CLASSIFIERS

Uncompressed Learner. Let $[n]$ denote the set $\{1, \cdots, n\}$. Denote Learner as an online linear classifier for binary classification. Let $(\mathcal{X}, \mathcal{Y})$ be a stream of data points, where $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ is a feature vector and $y \in \mathcal{Y} = \{-1, +1\}$ is the binary label for $\mathbf{x}$. A learning history of size T is a set of labeled data points $\mathcal{H} = \left\{\left(\mathbf{x}^t, y^t\right)\right\}_{t\in[T]}$. The parameters of Learner include (1) the hypothesis space $\mathcal{W} \subseteq \mathbb{R}^d$, where $\|\mathcal{W}\|_2 \leq D$, as well as the hypothesis (also referred to as weights) $\mathbf{w}^t \in \mathcal{W}$ at any time t and (2) a time-decaying learning rate $\eta^t > 0$. Then, a regularized loss function at time t for labeled data point $\left(\mathbf{x}^t, y^t\right)$ and hypothesis $\mathbf{w} \in \mathcal{W}$ is defined as

$$L^t\left(\mathbf{x}^t, y^t, \mathbf{w}\right) = l\left(y^t \cdot \left\langle \mathbf{w}, \mathbf{x}^t \right\rangle\right) + \frac{\lambda}{2}\|\mathbf{w}\|_2^2, \tag{1}$$

where $l(\cdot)$ is any convex, differentiable function and λ is a regularization parameter. Learner's gradient descent update for labeled point $\left(\mathbf{x}^t, y^t\right)$ is defined as

$$\mathbf{w}^{t+1} = \mathbf{w}^t - \eta^t \nabla L^t\left(\mathbf{x}^t, y^t, \mathbf{w}^t\right).$$

Denote $\mathbf{w}^*_{\mathcal{H}} = \arg\min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T_{\mathcal{H}}} L^t\left(\mathbf{x}^t, y^t, \mathbf{w}\right)$ as Learner's optimal hypothesis for history $\mathcal{H}$. At inference time, Learner's prediction for $\mathbf{x}^t$ is $\hat{y}^t = \text{sign}\left(\left\langle \mathbf{w}^t, \mathbf{x}^t \right\rangle\right)$. In this paper, we focus specifically on the logistic loss function $l(a) = -\log(\sigma(a))$, where $\sigma(a) = 1/(1+\exp(-a))$. Note that our analyses can also be generalized to other convex, differentiable loss functions.

Sketched Learner. Next, we consider the Weighted-Median Sketched Learner [17] (WM-Sketch), which maintains a count sketch [5] over history $\mathcal{H}$ and estimates Learner's optimal hypothesis $\mathbf{w}^*_{\mathcal{H}}$. WM-Sketch inherits the parameters η^t, $l(\cdot)$, and λ from Learner and its other parameters include (1) the size k and depth s of count sketch, (2) the hypothesis $\mathbf{z}^t \in \mathcal{Z} \subseteq \mathbb{R}^k$, where $\mathbf{z}^t$ is the count sketch at time t arranged as a vector, and (3) a scaled projection matrix $R = \left\{-1/\sqrt{s}, +1/\sqrt{s}\right\}^{k\times d}$ created using random hash functions $h_j : [d] \to [k/s]$ and $\sigma_j : [d] \to [-1, +1]$ for each $j \in [s]$. In this case, the regularized loss function at time t for $\left(\mathbf{x}^t, y^t\right)$ and hypothesis $\mathbf{z} \in \mathcal{Z}$ is defined using feature projection as

$$\hat{L}^t\left(\mathbf{x}^t, y^t, \mathbf{z}\right) = l\left(y^t \cdot \left\langle \mathbf{z}, R\mathbf{x}^t \right\rangle\right) + \frac{\lambda}{2}\|\mathbf{z}\|_2^2. \tag{2}$$

WM-Sketch's gradient descent update for $\left(\mathbf{x}^t, y^t\right)$ is given accordingly as $\mathbf{z}^{t+1} = \mathbf{z}^t - \eta^t \nabla \hat{L}^t\left(\mathbf{x}^t, y^t, \mathbf{z}^t\right)$. Furthermore, WM-Sketch's estimate $\hat{\mathbf{w}}^t_{\text{WM}}$ of $\mathbf{w}^t$ at time t is obtained as

$$\hat{\mathbf{w}}^t_i = \text{median}\left\{\sqrt{s}\alpha\sigma_j(i)\, \mathbf{z}^t_{j(k/s)+h_j(i)} \;:\; j \in [s]\right\},$$

where $\hat{\mathbf{w}}^t_i$ is the i-th weight of $\hat{\mathbf{w}}^t_{\text{WM}}$ and $\alpha = \left(1 - \eta^t\lambda\right)$ is a global scale parameter. Note that $\mathbf{z}^t$'s index $j(k/s) + h_j(i)$ represents the entry in Count-Sketch's j-th row and $h_j(i)$-th column. There are two ways for the prediction of $\mathbf{x}^t$ using the sketched learner: *(i) weight recovery*, i.e., $\hat{y}^t_{\text{WR}} = \text{sign}\left(\left\langle \hat{\mathbf{w}}^t_{\text{WM}}, \mathbf{x}^t \right\rangle\right)$, and *(ii) feature projection*, i.e., $\hat{y}^t_{\text{FP}} = \text{sign}\left(\left\langle \mathbf{z}^t, R\mathbf{x}^t \right\rangle\right)$. WM-Sketch's optimal hypothesis for history $\mathcal{H}$ of size $T_{\mathcal{H}}$ is $\mathbf{z}^*_{\mathcal{H}} = \arg\min_{\mathbf{z}\in\mathcal{Z}} \sum_{t=1}^{T_{\mathcal{H}}} \hat{L}^t\left(\mathbf{x}^t, y^t, \mathbf{z}\right)$. As shown in Theorem 1 of [17], $\|\mathbf{z}^*_{\mathcal{H}} - R\mathbf{w}^*_{\mathcal{H}}\|_2^2$ is small. Further, with high probability over the choice of R, the optimal hypothesis can be recovered from $\mathbf{z}^*_{\mathcal{H}}$ from Count-Sketch within an error ϵ, i.e. $\mathbb{E}_{\mathcal{H}}\left[\|\mathbf{w}^*_{\mathcal{H}} - \hat{\mathbf{w}}^*_{\text{WM}}\|_\infty\right] \leq \epsilon\|\mathbf{w}^*_{\mathcal{H}}\|_1$ (cf. Theorem 2 of [17]), where the expectation is over a random permutation of data points in the history $\mathcal{H}$. We use the feature projection $\hat{y}^t_{\text{FP}}$ in our experiments[1].

3 ADVERSARIAL ATTACKS

Our adversarial attacks are motivated by the question: *how does the performance of a given* WM-Sketch *worsen when attacked by adversaries operating in the (i) white-box (full), (ii) grey-box (partial), and (iii) black-box (zero) information settings?* We formally specify an adversary in Section 3.1 and three attack protocols based on the aforementioned information settings in Section 3.2.

3.1 Adversary

For a target WM-Sketch, an Adversary's parameters are:

- **Budget** (δ) denotes the maximum allowed perturbation of input point $\mathbf{x}$ to obtain a perturbed point $\tilde{\mathbf{x}}$, i.e., $\|\tilde{\mathbf{x}} - \mathbf{x}\| \leq \delta$.
- **Knowledge** denotes the available information about the target (also assume the access to the training history $\mathcal{H}$), namely,
  - *White-Box* (wb): target is a WM-Sketch, and access to all of the target's parameters, including the sketching matrix R as well as s, k, η^t, λ, etc.
  - *Grey-Box* (gb): target is a WM-Sketch, and access to all of the target's parameters, except the sketching matrix R.

[1] The original implementation of [17] uses $\hat{y}^t_{\text{WR}}$ for both training and prediction.

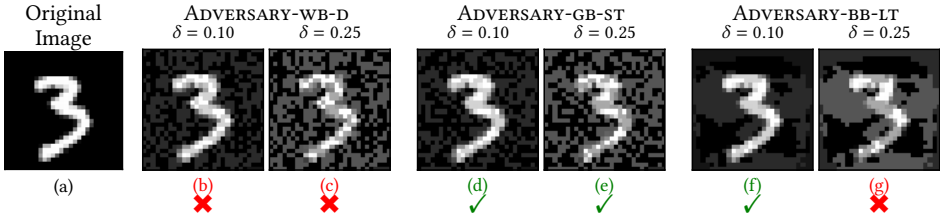

Figure 1: Illustration of adversarial attacks. The original image is as depicted in Figure (a). In Figures (b)-(c), (d)-(e), and (f)-(g), we show the perturbed images with budget $\delta \in \{0.1, 0.25\}$ by Adversary-wb-d, Adversary-gb-st, and Adversary-bb-lt targeting a WM-Sketch with $k = 16$ and $s = 2$, respectively. Here, "✓" and "✗" indicate if the prediction is correct or not.

– *Black-Box* (bb): access only to the target WM-Sketch's predicted label $\hat{y}_{\text{FP}}$ for $\mathbf{x}$.

The Adversary's goal is to induce misclassification by finding a perturbed vector $\tilde{\mathbf{x}}$ for a labeled data point $(\mathbf{x}, y)$ given its *budget* and *knowledge*. This implies that the target WM-Sketch predicts the correct label given $\mathbf{x}$ but an incorrect label given $\tilde{\mathbf{x}}$. That is, $\text{sign}\left(\langle \mathbf{z}^t, R\mathbf{x}^t \rangle\right) = y^t$, but $\text{sign}\left(\langle \mathbf{z}^t, R\tilde{\mathbf{x}}^t \rangle\right) \neq y^t$.

3.2 Attack Protocols

The Adversary's attack protocol consists of three steps namely, (i) choosing a surrogate classifier, (ii) crafting a perturbed point $\tilde{\mathbf{x}}$ for the surrogate, and (iii) attacking the target WM-Sketch with $\tilde{\mathbf{x}}$.

(i) Choosing a Surrogate. Based on the Adversary's *Knowledge* (including the training history $\mathcal{H}$), we have three surrogate options:

- *Direct* (d): the target WM-Sketch itself, in the white-box setting.
- *Sketch Transfer* (st): a WM-Sketch trained with the same parameters as the target except a different R, in the grey-box setting.
- *Learner Transfer* (lt): an independently trained Learner, in the black-box setting.

Formally, we study the following three adversaries: (a) Adversary-wb-d (*White-Box, Direct*); (b) Adversary-gb-st (*Grey-Box, Sketch Transfer*); and (c) Adversary-bb-lt (*Black-Box, Learner Transfer*).

(ii) Crafting a Perturbed Point. We focus on Fast-Gradient-Sign-Method (FGSM)-style attacks [9] for crafting the perturbed point for the surrogate. The key intuition behind this attack is to shift input $\mathbf{x}$ by a maximum distance δ such that it moves to the opposite side of the classifier's decision boundary to induce misclassification.

The L_∞-norm FGSM attack for a WM-Sketch parameterized by R for input $(\mathbf{x}, y)$ can be analytically derived as follows. Denote $\tilde{\mathbf{x}} = \mathbf{x}+\mathbf{r}$ as the perturbed poin. Note, the regularization term $\frac{\lambda}{2}\|\mathbf{z}\|_2^2$ is independent of $\tilde{\mathbf{x}}$ (cf. Equation 2). We construct the attack for a general convex, differentiable loss function $l(\cdot)$ and then instantiate specifically for logistic loss. Using first order approximation:

$$l\left(y \cdot \langle \mathbf{z}, R\tilde{\mathbf{x}} \rangle\right) = l\left(y \cdot \langle \mathbf{z}, R\mathbf{x} \rangle\right) + \nabla_{\tilde{\mathbf{x}}} l\left(y \cdot \langle \mathbf{z}, R\mathbf{x} \rangle\right)^\top \cdot \mathbf{r}$$

To compute $\tilde{\mathbf{x}}$, the smallest perturbation $\mathbf{r}$ within allowed budget δ which minimizes the loss is obtained as follows:

$$\begin{aligned} \arg\min_{\mathbf{r}} \quad & -\nabla_{\tilde{\mathbf{x}}} l\left(y \cdot \langle \mathbf{z}, R\mathbf{x} \rangle\right)^\top \cdot \mathbf{r} - l\left(y \cdot \langle \mathbf{z}, R\mathbf{x} \rangle\right) \\ \text{s.t.} \quad & \|\mathbf{r}\|_\infty \leq \delta \end{aligned} \tag{3}$$

Using Hölder's inequality with $\phi = -\nabla_{\tilde{\mathbf{x}}} l\left(y \cdot \langle \mathbf{z}, R\mathbf{x} \rangle\right)$, we have

$$\phi^\top \mathbf{r} \geq -\|\mathbf{r}\|_\infty \|\phi\|_1 \geq -\delta \|\phi\|_1$$

This lower bound for $\phi^\top \mathbf{r}$ is achieved when

$$\mathbf{r} = -\delta \cdot \text{sign}\left(\phi\right) = -\delta \cdot \text{sign}\left(-\nabla_{\tilde{\mathbf{x}}} l\left(y \cdot \langle \mathbf{z}, R\mathbf{x} \rangle\right)\right)$$

In the case of the logistic loss function, we have

$$\nabla_{\tilde{\mathbf{x}}} l\left(y \cdot \langle \mathbf{z}, R\mathbf{x} \rangle\right) = -\sigma\left(-y \cdot \langle \mathbf{z}, R\mathbf{x} \rangle\right) \cdot y R^\top \mathbf{z}$$

(iii) Attacking the Target. Putting them all together, we have $\mathbf{r} = -\delta \cdot \text{sign}\left(yR^\top \mathbf{z}\right)$. Then, Adversary-wb-d's perturbed point is:

$$\tilde{\mathbf{x}}_{\text{wb-d}} = \mathbf{x} - \delta \cdot \text{sign}\left(yR^\top \mathbf{z}\right) \tag{4}$$

Similarly, Adversary-gb-st perturbes an input point using a surrogate's projection matrix $\bar{R}$ and Count-Sketch $\bar{\mathbf{z}}$ instead of those of the target WM-Sketch as follows:

$$\tilde{\mathbf{x}}_{\text{gb-st}} = \mathbf{x} - \delta \cdot \text{sign}\left(y\bar{R}^\top \bar{\mathbf{z}}\right) \tag{5}$$

And finally, Adversary-bb-lt perturbs the input point $\mathbf{x}$ based on a Learner with weights $\mathbf{w}$ as follows:

$$\tilde{\mathbf{x}}_{\text{bb-lt}} = \mathbf{x} - \delta \cdot \text{sign}\left(y\mathbf{w}\right) \tag{6}$$

4 EXPERIMENTS

4.1 Experimental Setup

Dataset. We perform our experiments on the MNIST handwritten digits dataset [8] where the task is to identify a given image as the digit "3" or "7". Each image is a 784-dimensional feature vector. We choose 12,396 and 2,038 images randomly for training and testing, respectively. For consistency, we randomly permute the training points to create a fixed history $\mathcal{H}$ for all the classifiers.

Classifiers. We train an instance of Learner starting from an all-zero initial hypothesis with initial learning rate $\eta^0 = 0.1$ and regularization parameter $\lambda = 10^{-6}$. We set $l(\cdot)$ as the logistic loss function. We construct WM-Sketch classifiers of four different sizes $k \in \{8, 16, 32, 64\}$. Smaller sizes imply higher compression ratios yet larger errors in estimates. For each size k, we use four sketch depths $s \in \{1, 2, 4, 8\}$ and corresponding widths k/s. Furthermore, for each configuration of k and s, we create 20 unique R matrices with different random seeds. We report the results for WM-Sketch that uses the feature projection for training and prediction.

Adversaries. We vary δ in the range $\{0, 0.05, \cdots, 0.4\}$. For Adversary-wb-d, we perturb the test set by δ (cf. Equation 4). For Adversary-gb-st, we create one perturbed test set per δ for each of the 20 WM-Sketches parameterized by their R (cf. Equation 5) to attack the remaining 19 WM-Sketches. And for Adversary-bb-lt, we perturb the test set by δ according to Equation 6 using Learner

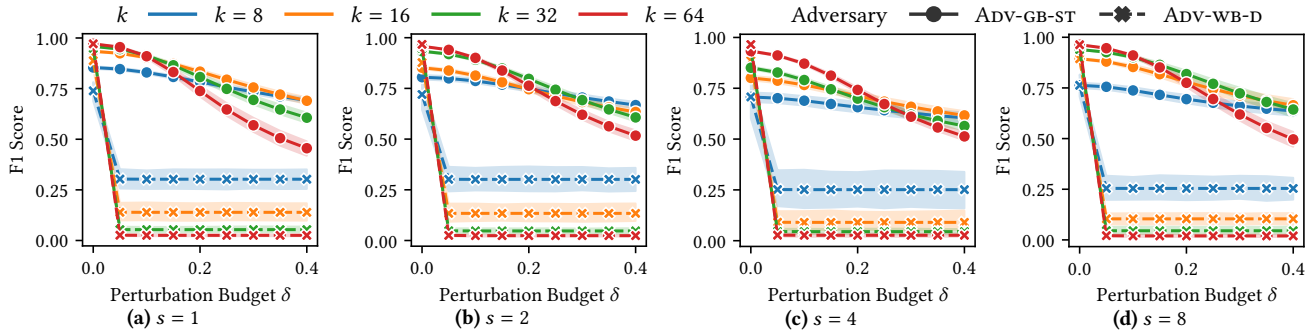

Figure 2: Accuracy of WM-Sketch with sizes $k \in \{8, 16, 32, 64\}$ for depths $s \in \{1, 2, 4, 8\}$ as a function of budget δ for the perturbed input constructed by Adversary-wb-d (dashed lines) and Adversary-gb-st (solid lines) in full and partial information settings. WM-Sketch is vulnerable against Adversary-wb-d, but compression adds robustness against Adversary-gb-st.

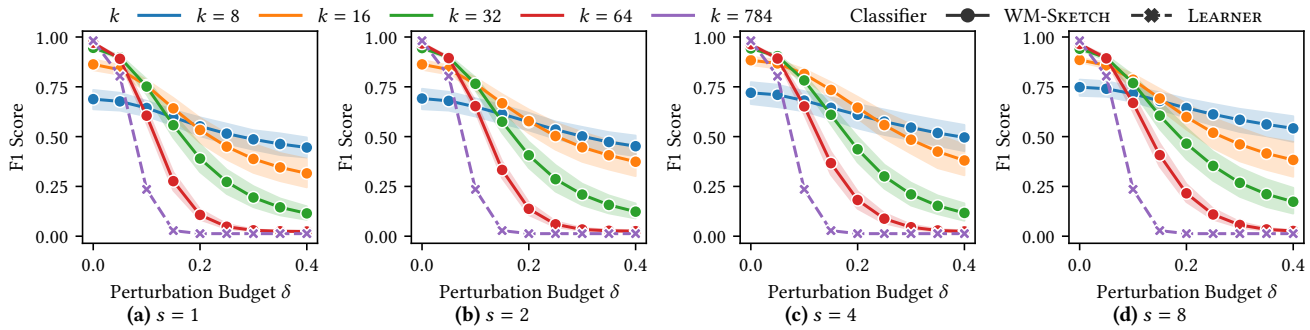

Figure 3: Accuracy of WM-Sketch classifiers with sizes $k \in \{8, 16, 32, 64\}$ for depths $s \in \{1, 2, 4, 8\}$ as well as Learner ($k = 784$) as a function of budget δ for the perturbed input constructed by Adversary-bb-lt. Higher levels of compression provide increased robustness against adversarial attacks but lower accuracy on unaltered input.

($\mathbf{w}$) as the surrogate. In Figure 1, we present an illustrated instance of the original and Adversary perturbed images.

Implementation. We adapt the original C++ implementation of WM-Sketch by Tai et al. [17]. Our attack protocols are implemented in Python 3.9. Our code is available here. All experiments were run on a Linux server with 32 CPU cores and 50GB RAM.

4.2 Results and Analyses

The dashed lines in Figure 2 present the F1-scores of WM-Sketches when the input data is perturbed by Adversary-wb-d (cf. Equation 4). The lines in different colors of the same sub-figure denote the results for WM-Sketches of different sizes (with the same depth s). While the results for WM-Sketches of different depths are plotted in different sub-figures. WM-Sketch achieves high accuracy on unperturbed data. However, even with small perturbation budget $\delta \geq 0.05$, its accuracy degrades significantly. This implies that WM-Sketch is just as vulnerable to FGSM-style attacks as Learner.

The solid lines in Figure 2 depict the F1-scores of WM-Sketches given input data perturbed by Adversary-gb-st (cf. Equation 5). Lines with error bands represent the average and variance of the F1-scores on perturbed points obtained from 19 surrogate WM-Sketches of the same k yet different R's. The largest WM-Sketch ($k = 64$) demonstrates the highest accuracy on unaltered input but the least robustness compared to smaller sketches as δ increases.

Figure 3 shows the robustness of WM-Sketches as a function of δ-perturbed input from Adversary-bb-lt (cf. Equation 6). The purple line confirms that Learner is misled by small perturbations while WM-Sketches are more robust. A key observation is that smaller compression levels (e.g., $k = 64$) lead to lower robustness but higher accuracy on unaltered input, and vice versa.

5 CONCLUSION

In this paper, we initiate the study of the robustness of sketched classifiers. Specifically focusing on FGSM-style attacks that we design for WM-Sketch [17], we find that (i) under full-information settings about WM-Sketch's internal parameters, it is as brittle as uncompressed classifiers to small targeted perturbations, but (ii) in more realistic partial and black-box information settings, its random count sketch matrix adds robustness to transfer attacks at the expense of classification accuracy on unaltered input. Our findings motivate further analysis of sketched classifiers to other attack paradigms and the design of countermeasures such as adversarial training, use of robust sketch alternatives, and ensemble methods to identify better tradeoffs between robustness and accuracy.

ACKNOWLEDGMENTS

Michael Mathioudakis is supported by University of Helsinki and Academy of Finland Projects MLDB (322046) and HPC-HD (347747).

REFERENCES

[1] Omri Ben-Eliezer, Rajesh Jayaram, David P. Woodruff, and Eylon Yogev. 2020. A Framework for Adversarially Robust Streaming Algorithms. In *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (PODS '20)*. ACM, New York, NY, USA, 63–80.

[2] Vladimir Braverman, Avinatan Hassidim, Yossi Matias, Mariano Schain, Sandeep Silwal, and Samson Zhou. 2021. Adversarial Robustness of Streaming Algorithms through Importance Sampling. In *Advances in Neural Information Processing Systems*, Vol. 34. Curran Associates, Inc., Red Hook, NY, USA, 3544–3557.

[3] Wieland Brendel, Jonas Rauber, and Matthias Bethge. 2018. Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 12 pages.

[4] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. 2018. Adversarial Attacks and Defences: A Survey. arXiv:1810.00069 [cs.LG]

[5] Moses Charikar, Kevin C. Chen, and Martin Farach-Colton. 2004. Finding frequent items in data streams. *Theor. Comput. Sci.* 312, 1 (2004), 3–15.

[6] Graham Cormode, Flip Korn, S. Muthukrishnan, and Divesh Srivastava. 2008. Finding Hierarchical Heavy Hitters in Streaming Data. *ACM Trans. Knowl. Discov. Data* 1, 4, Article 2 (2008), 48 pages.

[7] Graham Cormode and S. Muthukrishnan. 2005. An improved data stream summary: the count-min sketch and its applications. *J. Algorithms* 55, 1 (2005), 58–75.

[8] Li Deng. 2012. The MNIST Database of Handwritten Digit Images for Machine Learning Research. *IEEE Signal Process. Mag.* 29, 6 (2012), 141–142.

[9] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and Harnessing Adversarial Examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. OpenReview.net, 11 pages.

[10] Moritz Hardt and David P. Woodruff. 2013. How Robust Are Linear Sketches to Adaptive Inputs?. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing (STOC '13)*. ACM, New York, NY, USA, 121–130.

[11] Ashish Kapoor, Simon Baker, Sumit Basu, and Eric Horvitz. 2012. Memory constrained face recognition. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2539–2546.

[12] Stuart P. Lloyd. 1982. Least squares quantization in PCM. *IEEE Trans. Inf. Theory* 28, 2 (1982), 129–136.

[13] Ian McGraw, Rohit Prabhavalkar, Raziel Alvarez, Montse Gonzalez Arenas, Kanishka Rao, David Rybach, Ouais Alsharif, Hasim Sak, Alexander Gruenstein, Françoise Beaufays, and Carolina Parada. 2016. Personalized speech recognition on mobile devices. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 5955–5959.

[14] Nina Narodytska and Shiva Prasad Kasiviswanathan. 2016. Simple Black-Box Adversarial Perturbations for Deep Networks. arXiv:2112.07030 [cs.LG]

[15] Nikolaos Pitropakis, Emmanouil Panaousis, Thanassis Giannetsos, Eleftherios Anastasiadis, and George Loukas. 2019. A taxonomy and survey of attacks against machine learning. *Comput Sci. Rev.* 34 (2019), 100199.

[16] Jacob Steinhardt, Pang Wei Koh, and Percy Liang. 2017. Certified Defenses for Data Poisoning Attacks. In *Advances in Neural Information Processing Systems*, Vol. 30. Curran Associates, Inc., Red Hook, NY, USA, 3517–3529.

[17] Kai Sheng Tai, Vatsal Sharan, Peter Bailis, and Gregory Valiant. 2018. Sketching Linear Classifiers over Data Streams. In *Proceedings of the 2018 International Conference on Management of Data (SIGMOD '18)*. ACM, New York, NY, USA, 757–772.

[18] Peng Yang and Ping Li. 2021. Adversarial Kernel Sampling on Class-Imbalanced Data Streams. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management (CIKM '21)*. ACM, New York, NY, USA, 2352–2362.