IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 4 January 2024, accepted 5 February 2024, date of publication 13 February 2024, date of current version 28 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3365517

RESEARCH ARTICLE

Adversarial Robustness on Image Classification With k-Means

ROLLIN OMARI, JUNAE KIM, AND PAUL MONTAGUE
Defence Science and Technology Group, Australian Department of Defence, Edinburgh, SA 5111, Australia
Corresponding author: Rollin Omari (rollin.omari@defence.gov.au)

ABSTRACT Attacks and defences in adversarial machine learning literature have primarily focused on supervised learning. However, it remains an open question whether existing methods and strategies can be adapted to unsupervised learning approaches. In this paper we explore the challenges and strategies in attacking a k-means clustering algorithm and in enhancing its robustness against adversarial manipulations. We evaluate the vulnerability of clustering algorithms to adversarial attacks on two datasets (MNIST and Fashion-MNIST), emphasising the associated security risks. Our study investigates the impact of incremental attack strength on training, introduces the concept of transferability between supervised and unsupervised models, and highlights the sensitivity of unsupervised models to sample distributions. We additionally introduce and evaluate an adversarial training method that improves testing performance in adversarial scenarios, and we highlight the importance of various parameters in the proposed training method, such as continuous learning, centroid initialisation, and adversarial step-count. Overall, our study emphasises the vulnerability of unsupervised learning and clustering algorithms to adversarial attacks and provides insights into potential defence mechanisms.

INDEX TERMS Adversarial examples, adversarial machine learning, adversarial robustness, adversarial training, k-means clustering, unsupervised learning.

I. INTRODUCTION

In recent years, the field of machine learning has witnessed remarkable progress, with advancements particularly in unsupervised learning techniques, providing solutions to complex problems where unlabelled data is plentiful. However, this progress is now accompanied by a growing concern for reliability and adversarial robustness [1], [2]. As unsupervised learning becomes integral to various artificial intelligence applications, its robustness becomes synonymous with the reliability of the entire system. A failure to address adversarial vulnerabilities may lead to undesirable consequences, ranging from biased decision-making to compromised security [3]. Hence, understanding and addressing adversarial vulnerabilities in unsupervised learning is crucial, as it directly impacts the real-world applicability of such models [4].

Amongst the unsupervised techniques, clustering is potentially the most popular. The primary objective of clustering is to partition data such that similar samples are grouped together, while dissimilar ones are kept in separate clusters [5]. Machine learning literature contains a broad range of clustering algorithms and their applications, including but not limited to density-based (e.g., DBSCAN [6]), distribution-based (e.g., Gaussian mixture model [7]), centroid-based (e.g., k-means [8]) and hierarchical-based (e.g., BIRCH [9]) clustering algorithms.

The k-means clustering algorithm in particular, iteratively assigns each sample to the cluster with the closest center, relying on similarity measurements to update the cluster centres. Due to its simplicity and versatility, k-means is often used in the initial stages of data exploration and analysis. However, these traits also makes it highly vulnerable to adversarial attacks [4]. Namely, k-means relies on the Euclidean distance between samples and cluster centres to assign samples to clusters. Hence perturbed samples can disrupt the clustering process by pushing samples

The associate editor coordinating the review of this manuscript and approving it for publication was Tiago Cruz.

across cluster boundaries, leading to different clustering results [10].

Biggio et al. [11] were one of the first to consider adversarial attacks to clustering, where they described the obfuscation and poisoning attack settings, and provided results on single-linkage hierarchical clustering. They also considered evasion attacks against surrogate models in a limited-knowledge scenario [12]. Recently, Crussell and Kegelmeyer [13] proposed a poisoning attack specific to DBSCAN clustering, and Chhabra et al. [4] proposed a black-box attack for k-means clustering on a subset of the MNIST dataset, while Demontis et al. [14] provided a comprehensive evaluation on transferability and the factors contributing to the somewhat model-agnostic nature of adversarial examples.

To mitigate the threat of adversarial examples, a large variety of defence methods have been proposed, including adversarial training, which involves incorporating adversarial examples during the training process [15], [16], [17]; input transformations, which involve altering the input data via augmentation, smoothing or normalisation to improve model robustness [18], [19]; de-noising, which removes or reduces noise from input data with filtering techniques [20]; and certified defence, which provides bounds and guarantees for a model's output [21], [22]. However, these existing methods are heavily specialised towards supervised training. Among them, TRadeoff-inspired Adversarial DEfense via Surrogate-loss minimization (TRADES) [17] and Projected Gradient Descent Adversarial Training [16] are the most popular adversarial training methods. as they provide consistent improvements on robustness against various attacks.

In this paper, we take inspiration from these methods and introduce an adversarial training algorithm designed to enhance the robustness of a k-means clustering algorithm. Our method involves manipulating proportions of clean and perturbed samples in training data and iteratively training k-means in a continuous manner. An underlying intention of this method is to establish a much needed baseline for adversarial training for unsupervised algorithms. Our experimental results, conducted on widely recognised benchmark datasets, (i.e., MNIST [23] and Fashion-MNIST [24], see Fig. 1 for examples from each set), demonstrate the effectiveness of our simple adversarial learning algorithm. It significantly enhances the robustness of the clustering algorithm while also maintaining its overall performance. Importantly, since our method is directed towards manipulating training data distributions, it can be seamlessly integrated into various unsupervised learning frameworks to bolster their robustness.

In summary, the key contributions of this paper are threefold. Firstly, we introduce an *unsupervised* adversarial training method and demonstrate its effectiveness in enhancing the robustness of unsupervised models against adversarial attacks. Secondly, we apply and validate this training method with k-means clustering. We note the potential extension of this method to other unsupervised learning techniques. Finally, we highlight the effectiveness of transferability by utilising a supervised model in targeting an unsupervised model, with both trained on different datasets.

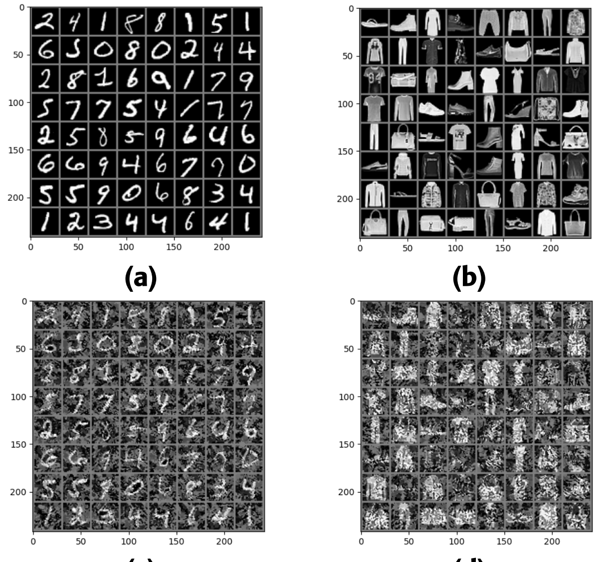

FIGURE 1. Collages of 8 × 8 randomly-picked images of handwritten digits and fashion items, respectively from the MNIST and Fashion-MNIST training datasets. Both datasets have a total of 70,000 samples with 60,000 images for training and 10,000 for testing. Collages (a) and (b) contain clean examples, while collages (c) and (d) contain adversarial examples of (a) and (b). For both (c) and (d), the adversarial examples are generated with I-FGSM.

II. BACKGROUND

In this section we introduce the machine learning concepts of adversarial examples and adversarial training. With the latter representing an effective defence strategy against the former.

A. ADVERSARIAL EXAMPLES

Given a standard clustering task, let x be an image and g be a clustering model. An adversarial example to g can be crafted through solving the following optimisation problem:

$$\min_{\delta} d(x, x+\delta) \text{ such that } g(x) \neq g(x+\delta), \quad (1)$$

where d measures similarity. This optimisation problem searches for a minimal perturbation δ that can change the class assignment for x or expected output of the model [25].

Depending on g's application, the adversarial example $x' = x + \delta$ can have devastating effects [3], [26], [27]. Moreover, in some instances, x' can be somewhat model-agnostic, such that when generated for model g, it can be effective in fooling another model f, which is either different in architecture, training dataset or both [14]. We exploit this exact phenomenon in generating adversarial examples for k-means clustering.

B. ADVERSARIAL TRAINING

Mitigating the effects of adversarial examples commonly involves adversarial training. This defence strategy utilises adversarial examples during training to improve a model's performance on similar examples during deployment. Existing adversarial training methods primarily focus on supervised learning approaches. This defence strategy can be formulated as a minimax optimisation problem [16]:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}}[\max_{\delta\in\Delta} L(x+\delta, y, \theta)], \tag{2}$$

where Δ is the perturbation set, $L(x, y, \theta)$ is the loss function of a neural network with parameters θ and (x, y) is the input-label pair taken from the distribution $\mathcal{D}$. This minimax problem is often solved by first crafting adversarial examples to solve the loss maximisation problem $\max_{\delta\in\Delta} L(x+\delta, y, \theta)$ and then optimising the model parameters θ using the generated adversarial examples.

III. METHODOLOGY

In this section, we detail our use of transferability. We also detail the specifics of our proposed adversarial training algorithm and its application to k-means clustering.

A. EXPLOITING TRANSFERABILITY

Generally, Eq. (2) cannot be directly applied to unsupervised algorithms due to its dependence on label y. However, this equation can be approximated by either replacing y with $g(x)$ or $f(x)$ such that $f(x) \approx g(x)$ and $\theta = \theta_f$, with f representing the surrogate or substitute model, g the target model (e.g., k-means) and $g(x)$ a cluster identifier. The input pair (x, y) then becomes $(x, g(x))$ in the absence of a substitute model, or conversely, $(x, f(x))$.

Having $f(x) \approx g(x)$ can be achieved by constructing f such that the outputs of f are similar in dimension to the outputs of g. Thereafter f can be trained on the outputs of g for a given input x. Typically in traditional adversarial training, f is a neural network designed to approximate the behaviour of the target model g.

In cases where labels are readily available for the datasets, it is hypothetically possible to use a pre-trained supervised model as a surrogate model f to approximate the loss function $L(x+\delta, y, \theta_f)$ for an unsupervised target model g. In such a situation, the concept of transferability may be leveraged to generate adversarial examples $x+\delta$ for model f and utilised to enhance the robustness of model g. This situation exploits the fact that adversarial examples crafted for one model can be effective against other models [14], and that most machine learning models rely on inductive bias [28], [29]. In this paper, we try to realise and demonstrate this hypothesis.

B. IMPROVING ROBUSTNESS

For the purposes of realising and demonstrating the hypothesis above, we use ground-truth label y in solving Eq. (2). Despite the presence of y, target model g is trained in an unsupervised manner and the ground-truth label is not required for the proposed adversarial training method to function.

To ensure that the target model g, (i.e., k-means), maintains competitive accuracies on both clean and adversarial data with the proposed training method, we manipulate the proportion η of clean and adversarial examples in the training dataset $\mathcal{D}'$. Furthermore, we increment training attack strength ϵ values at each step s with $\epsilon = s/\beta$, where β is the maximum number of steps or alternatively referred to as the adversarial step-count, and train k-means on both clean and adversarial examples in $\mathcal{D}'$. In each training step s we anchor clusters by initialising each step with the centroids from the previous step $\texttt{centroids}_{s-1}$. After training, the final centroids $\texttt{centroids}_{\beta}$ can be utilised appropriately in the required application. For more details, see Algorithm 1 below.

Algorithm 1 k-Means Adversarial Training

Input: training dataset $\mathcal{D}$, number of clusters k, proportion size η, surrogate model $f(\cdot)$, adversarial step-count β
Output: $k\texttt{-means.CENTROIDS}$

```
1:  initialise: centroids = k-means++(D, k)
2:  initialise: A = random.uniform(D, |D| × η)
3:  initialise: B = D − A
4:  k-means.TRAIN(D)
5:  for each s in [1, ..., β] do
6:      ε = s/β
7:      A′ = attack(A, f, ε)        ▷ apply I-FGSM
8:      D′ = A′ ∪ B
9:      k-means.TRAIN(D′)
10: end for
```

IV. METHODS AND RESOURCES

In this section, we detail the datasets and attacks utilised in our experiments. We also provide the necessary implementation details for replication.

A. DATASETS

To evaluate our method, we utilise the MNIST [23] and Fashion-MNIST [24] datasets. MNIST and Fashion-MNIST each contain a total of 70,000 samples, with 60,000 training and 10,000 testing. In both sets, each sample is a 28×28 monochrome 0 to 255 normalised image. MNIST contains handwritten digits, while Fashion-MNIST consists of various fashion products, (see Fig. 1). These datasets are primarily chosen due to their use in benchmarking, but also due to their simplicity, accessibility, popularity and resource-efficiency.

B. ADVERSARIAL ATTACKS

For attacks, we utilise the iterative Fast Gradient Sign Method (I-FGSM). This is one of many ℓ_∞ attacks that can be substituted into our training method. I-FGSM is an iterative version of the Fast Gradient Sign Method (FGSM), which operates by adjusting the input data in an attempt to maximise loss at each adversarial step. For more details on the attack algorithm, we refer the interested reader to [16].

C. IMPLEMENTATION DETAILS

For software, we use Python 3.9 and we implement our adversarial attacks with the Adversarial-Robustness-Toolbox [30]. The attack parameters for I-FGSM are set to default, except for the epsilon step-size α, which is set to $\epsilon/4$. We implement k-means with the scikit-learn Python library [31]. To determine the number of clusters k for k-means, we use the elbow method heuristic [32]. In all our experiments we use $k = 856$ and translate the clusters into classes with majority voting to calculate clustering accuracy relative to the ground-truth labels. For the surrogate model f, we use ResNet-18 [33] trained on the MNIST dataset. For adversarial training parameters, we use proportion size $\eta = 1/2$ and adversarial training step-count $\beta = 40$, unless stated otherwise. To evaluate the performance of the clustering algorithm under attack, we set $\epsilon = 1$ for attacks on testing data and report clustering accuracy. We repeat the experiments 30 times and report the average results. For hardware, we conduct our experiments on an Amazon Web Services cloud computer containing an NVIDIA Tesla T4 16GB 585MHz GPU and an Intel Xeon Platinum 8259CL 16 Cores 2.50GHz CPU.

V. RESULTS AND DISCUSSION

Here we evaluate the performance of k-means clustering under adversarial attacks and the effectiveness of our adversarial training algorithm. As demonstrated in Fig. 2, several trends and patterns emerge. Each shedding light on the impact of epsilon values on accuracy, the behaviour of accuracy with or without continuous learning, and the relationship between data distribution, robustness and clean performance.

One notable observation from Fig. 2 is the effectiveness of transferability. Transferability, in machine learning, refers to the ability of adversarial attacks generated for one model or dataset to successfully impact the performance of a different model or dataset. This concept has been studied extensively in previous work for evasion attacks, e.g., [34], [35], [36], [37], [38], and [39]. Notably, Biggio et al. [12] were the first to consider evasion attacks against surrogate models in a limited-knowledge scenario, while Goodfellow et al. [15], Tramer et al. [40], and Moosavi-Dezfooli et al. [41] were some of the first to make the observation that different models might learn intersecting decision boundaries in both benign and adversarial dimensions. In practical scenarios, adversarial attacks often exploit surrogate models due to limited access to the target model's architecture or loss function [37]. To our knowledge however, the work presented here is the first example of a supervised model being utilised as a surrogate model in targeting an unsupervised model and a surprising example of transferability working across two different datasets.

Namely, in all our experiments, the surrogate model f was a ResNet-18 model trained on the MNIST handwritten digits dataset. The model did not receive any further training or tuning. The generated attacks on Fashion-MNIST for model g (k-means) were generated with f, without any further training on Fashion-MNIST. That is, transferability was exploited explicitly in crafting adversarial examples using a supervised ResNet-18 model, which was trained on the MNIST dataset, to attack an unsupervised k-means clustering model on the Fashion-MNIST dataset. Considering this, we observe some very surprising results in Fig. 2(a-f). We observe that each row illustrates strikingly similar behaviour, where plots in the first column illustrate performance on MNIST and the second on Fashion-MNIST. We note that any observable differences in performance between MNIST and Fashion-MNIST (e.g., Fig. 2(c) and 2(d)) may be a result of MNIST being a simpler dataset than Fashion-MNIST. That is, any difference in performance may not be strictly due to the quality or

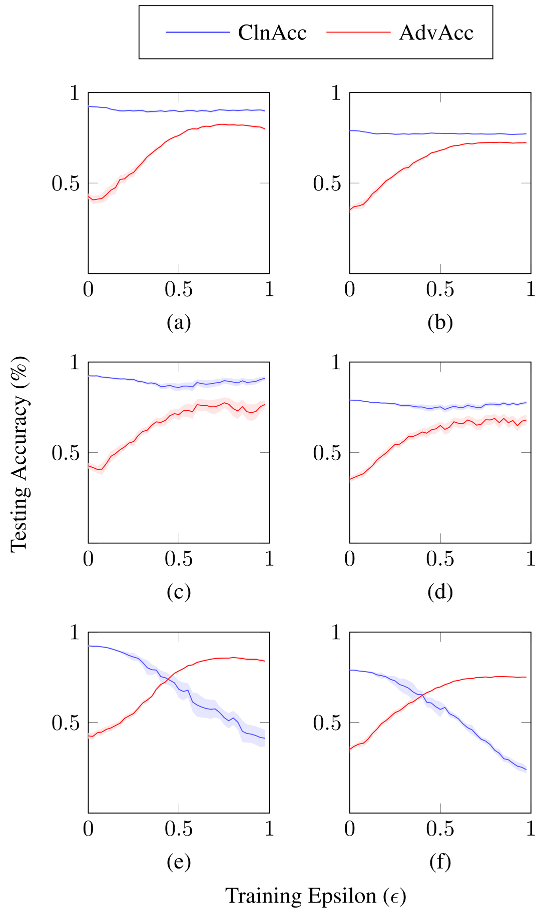

FIGURE 2. Clean (ClnAcc) and adversarial (AdvAcc) clustering accuracies, on MNIST and Fashion-MNIST. For all the provided plots we have I-FGSM as the attack and adversarial training algorithm, the attack strength (ϵ) used in training along the x-axis and the clustering accuracies (%) along the y-axis. In each plot, the solid lines represent the average results from 30 experiments, while shaded areas illustrate the error bars for a confidence level of 99%. In the first column we have results for MNIST and Fashion-MNIST in the second. In (a) and (b) we have the full implementation of the proposed adversarial training algorithm. In (c) to (f) we have parameter sensitivity results. In (c) and (d) we have k-means trained in a similar manner as to that in (a) and (b), however without the initialisation of centroids from previous steps, i.e., no continuous learning. In (e) and (f) we have fully perturbed training sets as opposed to half of the training sets, i.e., $\eta = 1$.

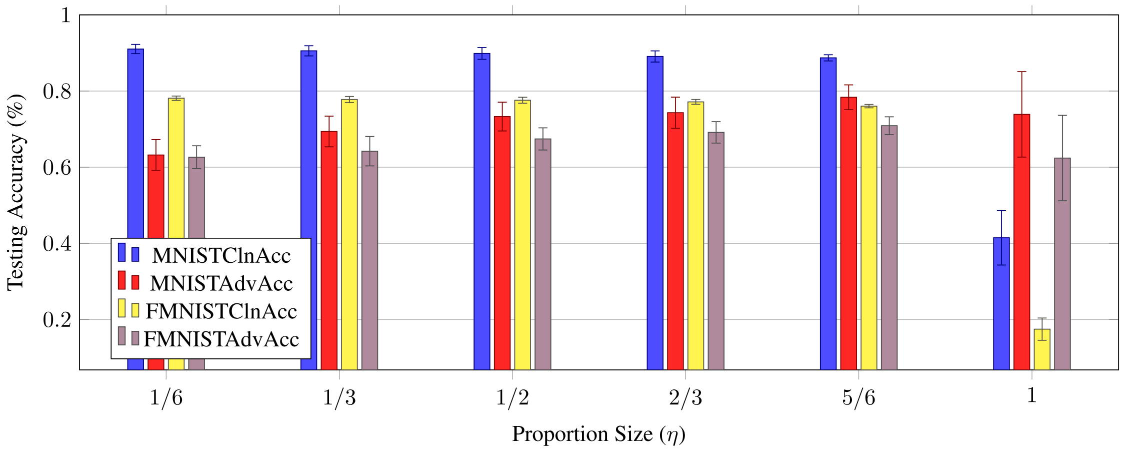

FIGURE 3. Parameter sensitivity results on various adversarial to clean proportions, without incremental training, i.e., different values for η and when adversarial step-count $\beta = 1$. Along the x-axis we have proportion size η, used in controlling the ration between clean and adversarial data. Along the y-axis we have the clustering accuracies (%). Each shaded bar illustrates average testing accuracies on MNIST and Fashion-MNIST after 30 experiments. Error bars are for a confidence level of 99%. I-FGSM is the attacking and defending algorithm. For all proportions, both training and testing attack strengths use $\epsilon = 1$.

efficacy of the adversarial examples generated with surrogate model f.

In Fig. 2(a) and 2(b) we also observe how epsilon values affect accuracy metrics. As ϵ increases, we notice a consistent decline in accuracy on clean test data, i.e., from approximately 92% to 90% for MNIST (Fig. 2(a)), and 79% to 78% for Fashion-MNIST (Fig. 2(b)). Conversely, accuracy on adversarial test data shows a gradual increase with each increase in epsilon, i.e., from 43% to 80% for MNIST, and 38% to 74% for Fashion-MNIST. This behaviour demonstrates a well-established trade-off between robustness against adversarial attacks and performance on clean data. As models are made more robust, their performance on clean data tends to degrade. However in our case, we only witness slight degradation.

Switching focus to our parameter sensitivity studies, in Fig. 2(c-f), we observe the effects of two key parameters of our adversarial training algorithm. Comparing Fig. 2(a) and 2(c) we observe the importance of initialising the centroids of each step $\texttt{centroids}_s$ with centroids from the previous step $\texttt{centroids}_{s-1}$, thereby anchoring learnt clusters and establishing continuous learning. The same observation can be made for Fashion-MNIST when comparing Fig. 2(b) and 2(d), where the continuous learning strategy clearly stabilises both clean and adversarial testing accuracy. In Fig. 2(e) and 2(f) we observe the importance of having an even proportion of clean and adversarial examples. In Fig. 2(e) and 2(f) we have $\eta = 1$ as opposed to $\eta = 1/2$ as in Fig. 2(a) and 2(b).

The importance of controlling clean and adversarial proportions with η is further highlighted in Fig. 3. Here we observe that a proportion of $\eta = 1/2$ or $\eta = 2/3$ adversarial examples results in competitive performance for both datasets, with the former having slightly better clean test accuracy (i.e., 89.9% vs. 89.1% for MNIST and 77.6% vs. 77.2% for Fashion-MNIST) and the latter having slightly better adversarial testing accuracy (i.e., 73.3% vs. 74.3% for MNIST and 67.4% vs. 69.1% for Fashion-MNIST). A proportion of $\eta = 5/6$ adversarial examples results in the best adversarial testing accuracy (78.4% for MNIST and 70.9% for Fashion-MNIST), however at the expense of clean testing accuracy (i.e., 88.7% for MNIST and 76.0% for Fashion-MNIST). For our purposes, the choice between $\eta = 1/2$, $\eta = 2/3$ and $\eta = 5/6$ considered both time efficiency and performance. $\eta = 1/2$ resulted in the least amount of time and training data required for acceptable performance, especially when statistically significant replications of each experiment had to be conducted.

Continuing with further observations from Fig. 3, we see that when $\eta = 1$, we have the worst clean (17.5%) and adversarial testing (62.4%) performance for Fashion-MNIST and acceptable performance for MNIST (i.e., 41.5% and 73.9%, respectively). For both datasets, we observe the greatest amount of variance when $\eta = 1$. We also coincidentally observe the importance of the step-count parameter β, i.e., incremental training. Namely when $\eta = 1$ and incremental training is absent, the adversarial testing accuracy is relatively acceptable for MNIST (i.e., 73.9%) and considerably low for Fashion-MNIST (62.4%). Conversely, when the entire training set is perturbed with an adversarial step-count of $\beta = 40$, i.e., when incremental training is available, the k-means algorithm attains an adversarial testing accuracy of 84% on MNIST and 75% on Fashion-MNIST, as shown in Fig. 2(e) and 2(f) respectively.

These results generally highlight the importance of continuous learning via centroid initialisation and emphasise the sensitivity of unsupervised models to sample distributions. For sensitivity in particular, unsupervised algorithms, unlike their supervised counterparts, operate without reliance on labelled data or environmental signals [42], [43]. Hence sample distributions determine equitable or non-equitable exposure, either enhancing or degrading an algorithm's performance in recognising and characterising underlying patterns [44].

Before concluding, we emphasise that the work presented here is a special case of our proposed adversarial training method. In a strict unsupervised scenario, the input pair (x, y) in Eq. 2 must be approximated. In the case where a label-independent attack exists (e.g., evolutionary attack [45] with modification) and target model g has an accessible loss function L, then the input pair can be approximated with $(x, g(x))$, where $g(x)$ is a cluster identifier. Otherwise in the case where g's loss function is not accessible or expressible, then a surrogate model f must be constructed to approximate g, such that $f(x) \approx g(x)$, and the input pair (x, y) can be replaced with $(x, f(x))$ and model parameters θ with θ_f. Additionally, as presented in this work, if ground-truth label y is available but an appropriate attack or loss function L for g is not readily available, then the model parameters θ are replaced with θ_f. If y and $g(x)$ have different dimensions, then a post-processing step such as majority voting or the Hungarian method can be utilised in resolving this assignment problem [46].

Due to the effectiveness of transferability, we consider it immaterial whether the substitute model f is a pre-trained model or not. Our focus on pre-trained models is a result of their ubiquity and their permissibility as vectors in generating and transferring adversarial attacks. The choice of datasets follows a similar vein, i.e., the MNIST and Fashion-MNIST datasets are accessible, popular and are often used for benchmarking. The two datasets also contain features that are representative of broader image datasets, enabling results generated from these datasets to be potentially applicable to practical scenarios involving image analysis and clustering (e.g., image segmentation and anomaly detection). Additionally, since MNIST and Fashion-MNIST have relatively limited feature spaces and homogeneous backgrounds compared to more complex image datasets, they have allowed us to study a scenario in which adversaries have limited ability in camouflaging manipulations. But more importantly, they have allowed us to establish one of the first baselines for unsupervised adversarial training.

Finally, the findings discussed here extend beyond image analysis and clustering, and have significant implications for the security and reliability of unsupervised learning and clustering applications in general. For instance, applications such as customer segmentation in targeted marketing for e-commerce platforms or disease sub-typing based on patient data in healthcare applications may face accuracy and reliability challenges amid adversarial modifications. Additional examples also include manufacturing and autonomous vehicles, where adversarial examples could lead to safety and quality control concerns if adversarial manipulations mislead the clustering process. It is due to these examples, and others, that we stress the importance of developing robust algorithms.

VI. CONCLUSION

Our evaluation of the performance of k-means clustering under adversarial attacks and the effectiveness of our adversarial training algorithm sheds light on several important factors. We have observed the impact of adversarial step-count β on accuracy, the trade-off between robustness against adversarial attacks and performance on clean data, and the significance of key parameters in our adversarial training algorithm.

One notable finding is the effectiveness of transferability, which highlights the potential of using supervised models as surrogates for target unsupervised models. This finding emphasises the importance of considering diverse scenarios in model training and evaluation, since real-world applications often involve models facing data distributions not encountered during training.

This paper also highlights the sensitivity of unsupervised models to sample distributions, emphasising the need for careful control of clean and adversarial example proportions. Our results underscore the importance of continuous learning and proper initialisation of centroids, which can significantly enhance both clean and adversarial testing accuracy. For practical implications, we consider that the insights generated in this study may find utility in deployed machine learning models, especially those utilising unsupervised learning or clustering algorithms.

Overall, our study emphasises the vulnerability of unsupervised learning and clustering algorithms to adversarial attacks and provides insights into potential defence mechanisms. Future research could explore dynamic defence mechanisms that adapt to the specific characteristics of the surrogate model and the target model. Future studies may also investigate the incorporation of domain knowledge to enhance model robustness, i.e., understanding the inherent characteristics of different datasets could inform the development of more resilient models. Finally, extending our method to diverse real-world applications, such as medical imaging or cyber-security, could provide a more comprehensive understanding of the practical implications and generalisability of our findings. We hope these avenues, and others, will contribute to the ongoing efforts to fortify machine learning models.

REFERENCES

[1] H. Xu, Y. Ma, H.-C. Liu, D. Deb, H. Liu, J.-L. Tang, and A. K. Jain, "Adversarial attacks and defenses in images, graphs and text: A review," *Int. J. Autom. Comput.*, vol. 17, no. 2, pp. 151–178, Apr. 2020.
[2] A. Chakraborty, M. Alam, V. Dey, A. Chattopadhyay, and D. Mukhopadhyay, "A survey on adversarial attacks and defences," *CAAI Trans. Intell. Technol.*, vol. 6, no. 1, pp. 25–45, Mar. 2021.
[3] N. Martins, J. M. Cruz, T. Cruz, and P. H. Abreu, "Adversarial machine learning applied to intrusion and malware scenarios: A systematic review," *IEEE Access*, vol. 8, pp. 35403–35419, 2020.
[4] A. Chhabra, A. Roy, and P. Mohapatra, "Suspicion-free adversarial attacks on clustering algorithms," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 4, pp. 3625–3632.
[5] R. Dubes and A. K. Jain, "Clustering methodologies in exploratory data analysis," *Adv. Comput.*, vol. 19, pp. 113–228, Jan. 1980.
[6] K. Khan, S. U. Rehman, K. Aziz, S. Fong, and S. Sarasvady, "DBSCAN: Past, present and future," in *Proc. 5th Int. Conf. Appl. Digit. Inf. Web Technol. (ICADIWT)*, Feb. 2014, pp. 232–238.
[7] D. A. Reynolds, "Gaussian mixture models," *Encyclopedia Biometrics*, vol. 741, nos. 659–663, Jul. 2009.

[8] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A K-means clustering algorithm," *J. Roy. Stat. Soc.*, vol. 28, no. 1, pp. 100–108, May 1979.

[9] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: A new data clustering algorithm and its applications," *Data Mining Knowl. Discovery*, vol. 1, no. 2, pp. 141–182, 1997.

[10] D. B. Skillicorn, "Adversarial knowledge discovery," *IEEE Intell. Syst.*, vol. 24, no. 6, pp. 54–61, Nov. 2009.

[11] B. Biggio, I. Pillai, S. R. Bulo, D. Ariu, M. Pelillo, and F. Roli, "Is data clustering in adversarial settings secure?" in *Proc. ACM Workshop Artif. Intell. Secur.*, Nov. 2013, pp. 87–98.

[12] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli, "Evasion attacks against machine learning at test time," in *Machine Learning and Knowledge Discovery in Databases*. Prague, Czech Republic. Springer, 2013, pp. 387–402.

[13] J. Crussell and P. Kegelmeyer, "Attacking dbscan for fun and profit," in *Proc. SIAM Int. Conf. Data Mining*, 2015, pp. 235–243.

[14] A. Demontis, M. Melis, M. Pintor, M. Jagielski, B. Biggio, A. Oprea, C. Nita-Rotaru, and F. Roli, "Why do adversarial attacks transfer? Explaining transferability of evasion and poisoning attacks," in *Proc. USENIX Security Symp.*, 2019, pp. 321–338.

[15] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2014, *arXiv:1412.6572*.

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

[17] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan, "Theoretically principled trade-off between robustness and accuracy," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7472–7482.

[18] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille, "Mitigating adversarial effects through randomization," 2017, *arXiv:1711.01991*.

[19] C. Guo, M. Rana, M. Cisse, and L. van der Maaten, "Countering adversarial images using input transformations," 2017, *arXiv:1711.00117*.

[20] A. Mustafa, S. H. Khan, M. Hayat, J. Shen, and L. Shao, "Image super-resolution as a defense against adversarial attacks," *IEEE Trans. Image Process.*, vol. 29, pp. 1711–1724, 2020.

[21] J. Cohen, E. Rosenfeld, and Z. Kolter, "Certified adversarial robustness via randomized smoothing," in *Proc. ICML*, 2019, pp. 1310–1320.

[22] A. Raghunathan, J. Steinhardt, and P. Liang, "Certified defenses against adversarial examples," 2018, *arXiv:1801.09344*.

[23] *The Mnist Database of Handwritten Digits*. Accessed: Oct. 19, 2023. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[24] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.

[25] N. Dalvi, P. Domingos, S. Sanghai, and D. Verma, "Adversarial classification," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, Aug. 2004, pp. 99–108.

[26] H.-Y. Lin and B. Biggio, "Adversarial machine learning: Attacks from laboratories to the real world," *Computer*, vol. 54, no. 5, pp. 56–60, May 2021.

[27] G. Apruzzese, H. S. Anderson, S. Dambra, D. Freeman, F. Pierazzi, and K. Roundy, "'Real attackers don't compute gradients': Bridging the gap between adversarial ML research and practice," in *Proc. IEEE Conf. Secure Trustworthy Mach. Learn. (SaTML)*, Feb. 2023, pp. 339–364.

[28] T. M. Mitchell, "The need for biases in learning generalizations," Dept. Comput. Sci., Lab. Comput. Sci. Res., Rutgers Univ. New Brunswick, NJ, USA, Tech. Rep., 1980, vol. 59.

[29] D. F. Gordon and M. Desjardins, "Evaluation and selection of biases in machine learning," *Mach. Learn.*, vol. 20, nos. 1–2, pp. 5–22, 1995.

[30] M.-I. Nicolae, M. Sinn, M. Ngoc Tran, B. Buesser, A. Rawat, M. Wistuba, V. Zantedeschi, N. Baracaldo, B. Chen, H. Ludwig, I. M. Molloy, and B. Edwards, "Adversarial robustness toolbox v1.0.0," 2018, *arXiv:1807.01069*.

[31] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.

[32] C. Yuan and H. Yang, "Research on K-value selection method of K-means clustering algorithm," *J*, vol. 2, no. 2, pp. 226–235, Jun. 2019.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, vol. 16, 2016, pp. 770–778.

[34] Y. Dong, F. Liao, T. Pang, X. Hu, and J. Zhu, "Boosting adversarial attacks with momentum," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 9185–9193.

[35] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," 2016, *arXiv:1611.02770*.

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

[37] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proc. ACM Asia Conf. Comput. Commun. Secur.*, Apr. 2017, pp. 506–519.

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

[39] Y. Wang and F. Farnia, "On the role of generalization in transferability of adversarial examples," in *Proc. 39th Conf. Uncertainty Artif. Intell.*, 2023, pp. 2259–2270.

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

[41] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1765–1773.

[42] E. M. Darling and J. G. Raudseps, "Non-parametric unsupervised learning with applications to image classification," *Pattern Recognit.*, vol. 2, no. 4, pp. 313–335, Dec. 1970.

[43] Z. Ghahramani, "Unsupervised learning," in *Summer School on Machine Learning*. Berlin, Germany: Springer, 2003, pp. 72–112.

[44] M. Qian and C. Zhai, "Robust unsupervised feature selection," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 1–7.

[45] J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 828–841, Oct. 2019.

[46] H. W. Kuhn, "The Hungarian method for the assignment problem," *Nav. Res. Logistics Quart.*, vol. 2, pp. 83–97, Mar. 1955.

ROLLIN OMARI received the B.M.S. and B.A.S. (Hons.) degrees from the University of Canberra, Canberra, Australia, in 2014 and 2015, respectively. He is currently pursuing the Ph.D. degree with the College of Engineering, Computing and Cybernetics, Australian National University, Canberra, Australia. He is a Researcher with the Defence Science Technology Group, Australian Department of Defence. His current research interests include machine learning, bio-inspired computing, and cyber security.

JUNAE KIM received the B.S. degree from Ewha Womans University, Seoul, South Korea, in 2000, the M.S. degree from the Pohang University of Science and Technology, Pohang, South Korea, in 2002, and the M.Phil. and Ph.D. degrees from Australian National University, Acton, Australia, in 2007 and 2011, respectively. She is currently a Researcher with the Defence Science and Technology Group, Australian Department of Defence. Her current research interests include computer vision and machine learning.

PAUL MONTAGUE received the Ph.D. degree in mathematical physics from the University of Cambridge. Following a research career in mathematical physics, he transitioned into computer security and developed security and cryptographic solutions for Motorola. He is currently a Researcher of computer security with the Defence Science and Technology Group, Australian Department of Defence. His research interests include the application of machine learning and data analytic techniques to cyber security problems and the robustness of machine learning.