On the Robustness of Clustering Algorithms to Adversarial Attacks

Zhang San

mc123456789@umac.mo

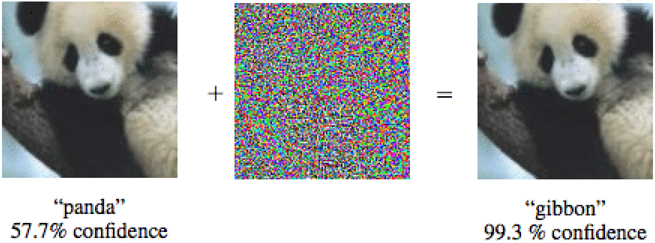
University of Macau Faculty of Science and Technology,
Department of Computer and Information Science

Master's Thesis Defence
10th July 2019

Supervisor: Marcello Pelillo

Adversarial Machine Learning

- Machine learning models provide good predictions in different domains
- They are completely data-driven and data contains noise by nature
- Sensitive to adversarial perturbations in the input data
- Several works mainly focused on supervised learning applications

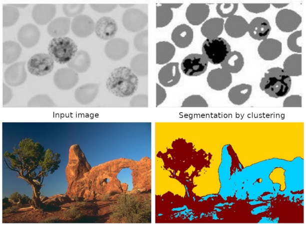


[1]

[1] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 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, 2015.

Clustering Applications

- Clustering is finding a wide range of applications, due to the absence of labeled data
- Ex: Image segmentation, face clustering, market, social or crime analysis, malicious software clustering, information processing, etc.



[2]

- Biggio et al. proposed a framework for fooling hierarchical clustering

[2] Nameirakpam Dhanachandra, Khumanthem Manglem, Yambem Jina Chanu, Image Segmentation Using K -means Clustering Algorithm and Subtractive Clustering Algorithm, Procedia Computer Science.

K-Means Clustering

- It splits n objects into K maximal cohesive groups by minimizing:

$$\underset{C}{\arg\min} \sum_{i=0}^{K} \left\{ \sum_{j \in \text{elements of } C_i \text{ cluster}} ||x_j - \mu_i||^2 \right\}$$

where μ_i is the centroid of cluster i

- It is a polynomial algorithm that guaranteed to converge in a finite number of steps

K-Means Algorithm

1. Initialize cluster centroids $\mu_1, \ldots, \mu_k,$.
2. Repeat until all points remain unchanged (convergence):
    1. Assign samples to clusters: $\forall i \in X \quad c^{(i)} = \arg\min_j ||x^{(i)} - \mu_j||^2$
    2. Update cluster centroids: $\forall j \in C \quad \mu_j = \frac{\sum_{i=1}^{m} 1\{c^{(i)}=j\} x^{(i)}}{\sum_{i=1}^{m} 1\{c^{(i)}=j\}}$

- Sensitive to cluster centroids initialization and can't handle non-convex set.

Spectral Clustering

- Uses properties of eigenvalues and eigenvectors for solving the clustering problem

Spectral Clustering Algorithm

1. Construct a similarity graph and compute the normalized graph Laplacian L_{sym}.
2. Embed data points in a low-dimensional space (spectral embedding), in which the clusters are more obvious, computing the k smallest eigenvectors $v_1, \dots, v_k$ of L_{sym}.
3. Let $V = [v_1, \dots, v_k] \in \mathbb{R}^{n \times k}$.
4. Form the matrix $U \in \mathbb{R}^{n \times k}$ from V by normalizing the row sums to have norm 1, that is:

$$u_{ij} = \frac{v_{ij}}{\left(\sum_k v_{ik}^2\right)^{1/2}}$$

5. For $i = 1, \dots, n$, let $y_i \in \mathbb{R}^k$ be the vector corresponding to the ith row of U.
6. Cluster the points y_i with $i = 1, \dots, n$ with the k-means algorithm into clusters $C_1, \dots, C_k$.

- Applying K-Means on the spectral embedding allows to cluster objects that are connected but not necessarily compact or clustered within convex boundaries.

Dominant Sets Clustering

- Graph-theory based approach supported by game theory
- Clusters correspond to dominant sets
- Implemented using deterministic game dynamics like discrete Replicator Dynamics
- Robust against noise, allows to find overlapped clusters, makes no assumptions on the similarity matrix and on the number of clusters

Extract a dominant set

```
1 distance = $\infty$
2 while distance $> \varepsilon$:
    1 $x_i(t+1) = x_i(t)\frac{(Ax(t))_i}{x(t)^T Ax(t)}$
    2 distance $= \texttt{norm}(x_i(t+1) - x_i(t))$
3 return $\sigma(x)$
```

Dominant Sets Algorithm

1. Extract all dominant sets or K of them from the input similarity matrix

Rows-based Adversarial Generator

- Fooling image segmentation systems using rows-based noise
- Optimization is done using a genetic algorithm

Rows-based Adversarial Generator Algorithm

1. Pick s most sensitive rows in input X
2. For each p-consecutive pixels in each row find the optimal adversarial noise ε^* to inject.
   - **Objective function**: distance or dissimilarity from the initial segmentation.
   - Maximum adversarial perturbation level Δ.
   - Generate a random perturbation ε to inject. Evaluate the fitness function and apply stochastic operators for improving ε.
3. Craft adversarial example: $X' \rightarrow X + \varepsilon^*$

Rows-based Adversarial Generator

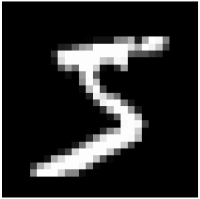


Figure: Input digit X from MNIST.

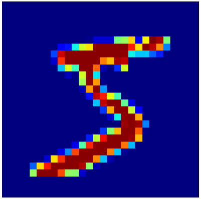


Figure: Jet visualization of X.

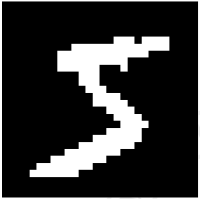


Figure: Spectral Clustering predicted segmentation.

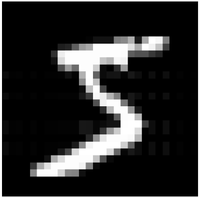


Figure: Adversarial digit X'.

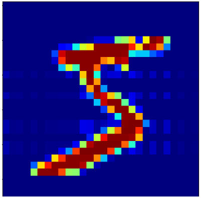


Figure: Jet visualization of X'.

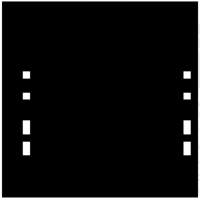


Figure: Spectral Clustering adversarial segmentation.

Pixel-wise Adversarial Generator

- Fooling image segmentation systems using pixel-wise noise
- Injected noise seems to be random (like gaussian or salt-and-pepper)
- Adversarial noise is targeted and attack sensitive regions

Pixel-wise Adversarial Generator Algorithm

1. Pick s most sensitive pixels in input X
2. For each p-consecutive sensitive pixels find the optimal adversarial noise ε^* to inject.
    - **Objective function**: distance or dissimilarity from the initial segmentation.
    - Maximum adversarial perturbation level Δ.
    - Generate a random perturbation ε to inject. Evaluate the fitness function and apply stochastic operators for improving ε.
3. Craft adversarial example: $X' \rightarrow X + \varepsilon^*$

Pixel-wise Adversarial Generator

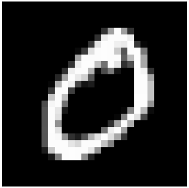


Figure: Input digit X from MNIST.

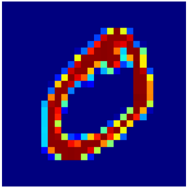


Figure: Jet visualization of X.

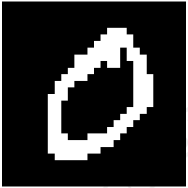


Figure: Dominant Sets predicted segmentation.

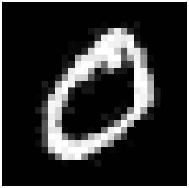


Figure: Adversarial digit X'.

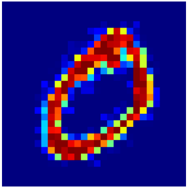


Figure: Jet visualization of X'.

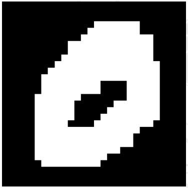


Figure: Dominant Sets adversarial segmentation.

Target Clustering Adversarial Generator

- Fooling feature-based data clustering systems
- The attacker aims to break partially the resulting clusters composition
- Sensitive or target samples are moved towards a target cluster

Rows-based Adversarial Generator Algorithm

1. Pick s most sensitive samples from input X
2. For each p-consecutive features find the optimal adversarial noise ε^* to inject for moving samples from a cluster towards a desired one.
    - **Objective function**: distance or dissimilarity from the initial cluster labeling.
    - Maximum adversarial perturbation level Δ.
    - Generate a random perturbation ε to inject. Evaluate the fitness function and apply stochastic operators for improving ε.
3. Craft adversarial example: $X' \rightarrow X + \varepsilon^*$

Experimental Setup

- Segmentation dataset: MNIST
- Clustering datasets: Synthetic, Yale Face, DIGITS
- No initialization effect is introduced (multiple iterations with different seeds)
- ARI[3] and ARI_h, as proposed by Milligan et al.,for analyzing clustering quality
- $||X - X'||_2$[4], as suggested by Biggio et al., for representing the attacker's capacity
- Gaussian Similarity is used for constructing the similarity matrix

$$A(i,j) = \exp\left(\frac{-||\psi(i) - \psi(j)||_2^2}{2\sigma^2}\right)$$

[3] Glenn Milligan and Martha Cooper. A study of the comparability of external criteria for hierarchical cluster-analysis. Multivariate Behavioral Research - MULTIVARIATE BEHAV RES, 21:441–458, 10 1986. doi: 10.1207/s15327906mbr2104 5.

[4] Battista Biggio, Ignazio Pillai, Samuel Rota Bulò, Davide Ariu, Marcello Pelillo, and Fabio Roli. Is data clustering in adversarial settings secure? In AISec'13, Proceedings of the 2013 ACM Workshop on Artificial Intelligence and Security, Co-located with CCS 2013, Berlin, Germany, November 4, 2013, pages 87–98, 2013.

Rows-based Results

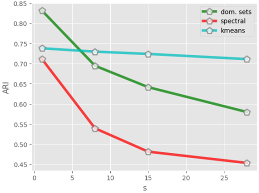


Figure: *ARI* over number of perturbed rows *s*.

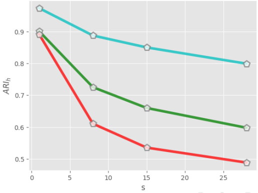


Figure: ARI_h over number of perturbed rows *s*.

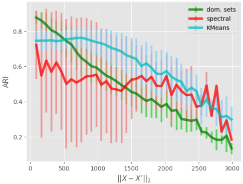


Figure: *ARI* over attacker's capacity $||X - X'||_2$.

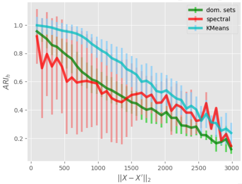


Figure: ARI_h over attacker's capacity $||X - X'||_2$.

- High sensitivity of Spectral Clustering to the number of attacked rows *s*
- Greater robustness provided by Dominant Sets and K-Means

Pixel-wise Results

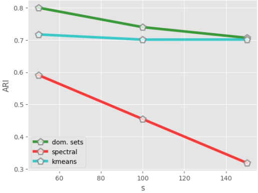


Figure: *ARI* over number of perturbed pixels *s*.

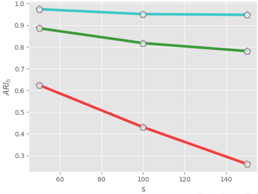


Figure: ARI_h over number of perturbed pixels *s*.

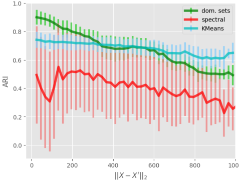


Figure: *ARI* over attacker's capacity $||X - X'||_2$.

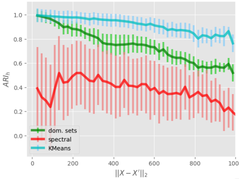


Figure: ARI_h over attacker's capacity $||X - X'||_2$.

- High sensitivity of Spectral Clustering to the number of attacked pixels *s*
- Greater robustness provided by Dominant Sets and K-Means

Target Clustering Results - Synthetic

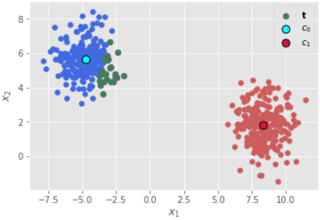


Figure: Synthetic dataset and target samples (green).

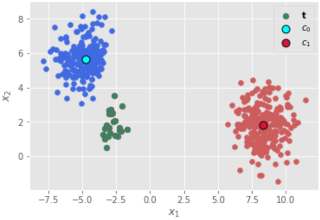


Figure: Adversarial samples X'.

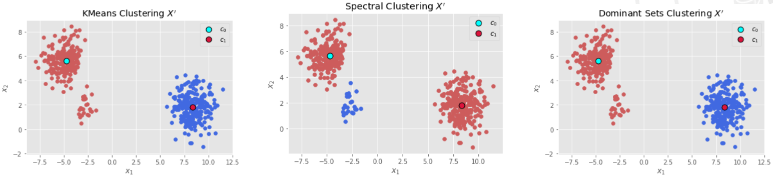


Figure: Data clustering obtained for X 0 using K-Means (left), Spectral (middle) and Dominant Sets (right) clustering.

Target Clustering Results - Synthetic

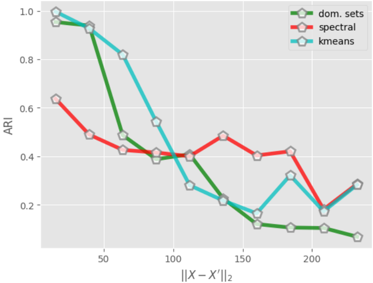


Figure: *ARI* over attacker's capacity $||X - X'||_2$.

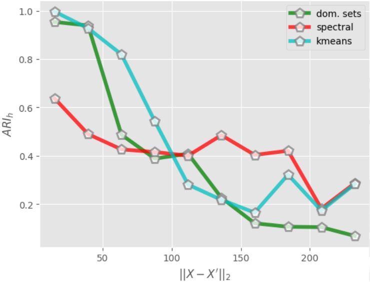


Figure: ARI_h over attacker's capacity $||X - X'||_2$.

- High sensitivity of Spectral Clustering to small perturbations
- Spectral embedding affected by adversarial noise
- Greater robustness provided by K-Means and Dominant Sets

Target Clustering Results - Yale Face

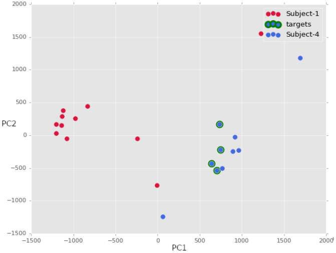


Figure: Yale Face subject-1 and subject-4 clusters with 2-PC projection. Target samples are highlighted in green.

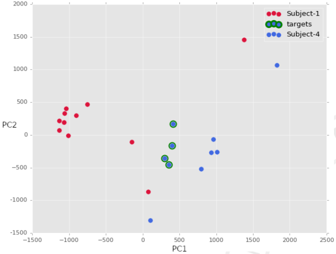


Figure: Adversarial Yale Face dataset X' with 2-PC projection.

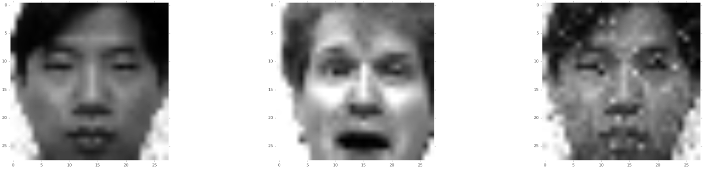


Figure: Sample in subject-4 on the right moved towards subject-1 cluster (sample on the middle). The crafted adversarial examples in shown on the right

Target Clustering Results - Yale Face

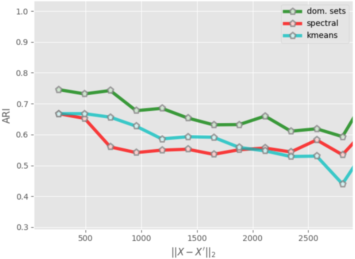


Figure: *ARI* over attacker's capacity $||X - X'||_2$.

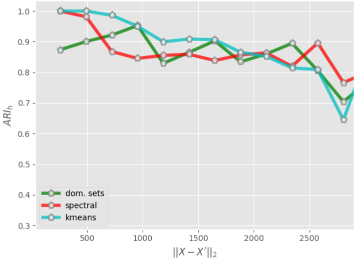


Figure: ARI_h over attacker's capacity $||X - X'||_2$.

- Even face clustering can be fooled
- The three algorithms react similarly in presence of adversarial noise

Conclusions and Future Work

Conclusions:

- Image segmentation and clustering algorithms are sensitive to adversarial noise. Defensive strategy are required.
- Dominant Sets seems to be robust even against adversarial noise
- Spectral Clustering seems to be strongly sensitive to small adversarial perturbations.

Future work:

- Speed up the adversarial algorithms using GPUs architectures
- Consider different similarity measures and embedding
- Generalize the adversarial target clustering to k clusters
- Evaluate an outlier detection analysis over crafted adversarial examples

Bibliography

- Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS 2018, Toronto, ON, Canada, October 15-19, 2018, pages 2154–2156, 2018.
- Battista Biggio, Ignazio Pillai, Samuel Rota Bulò, Davide Ariu, Marcello Pelillo, and Fabio Roli. Is data clustering in adversarial settings secure? In AISec'13, Proceedings of the 2013 ACM Workshop on Artificial Intelligence and Security, Co-located with CCS 2013, Berlin, Germany, November 4, 2013, pages 87–98, 2013.
- Massimiliano Pavan and Marcello Pelillo. Dominant sets and pairwise clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29:167–172, 2007.
- Ulrike von Luxburg. A tutorial on spectral clustering. Statistics and Computing, 17(4): 395–416, 2007.
- Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg.