CS 475/675 Project Proposal

Xuan Wu, Jinhan Zhang, Heidi Zhang, Yunxiao Yang
xwu71, jzhan205, czhan105, yyang117

Abstract

Motivated by the complexity of practical data set. Dealing with robustness in machine learning has become increasingly important and interesting. Thus we choose to focus on robustness in machine learning in this project.

1 Project choice

Choose either a **methods** or **applications** project, and a subarea from the below table.

☐ **Applications**				
☐ Genomics data	☐ Healthcare data	☐ Text data	☐ Image data	☐ Finance data
☒ **Methods**				
☐ Fairness in ML	☐ Interpretable ML	☐ Graphical Models	☒ Robust ML	☐ Privacy in ML

2 Introduction

Robustness measures the stability of a machine learning model's test accuracy. A robust machine learning model can yield high accuracy on independent test sets in the presence of noisy inputs and perform reasonably well even large perturbations exist in data [7]. Robustness is gradually gaining importance in machine learning as models are being used in high-stake applications, such as autonomous vehicles, disease diagnosis, and surgical robots [5, 10, 11].

Current studies on the robustness of machine learning involve generating adversarial samples for training [9], enhancing a model's robustness by performing data transformation on training data [2] or designing more robust traning algorithms [12, 14], or modifying the model's architectures [8].

We will focus on enhancing ML model to withstand the adversarial attacks when performing image classification, mostly by designing more robust training procedures. The adversarial examples we plan to introduce will be generated by the Adversarial Robustness Toolbox (ART) [9]. The models on which we plan to test the robustness are SVM and CNN. Most of the current robust algorithms can be applied in both of them [2]. The input to our algorithm is an image of hand-written digit, and the output is the predicted digit.

3 Dataset and Features

We plan to develop and test a robust training algorithm on models trained on image classification with data sets attacked by ART [9]. The dataset we will focus on is the `digits dataset` from `sklearn.datasets` package, which is adapted from the test set of Optical Recognition of Handwritten Digits Data [1, 13].

The dataset contains 1797 instances of hand-written digits, where each instance is an image of size eight by eight (thus a total of 64 pixels). For each pixel, the valid range of value is an integer between 0 to 16. There are ten digits classes, and each class contains about 180 pictures; thus, the digit dataset is a well-balanced dataset. This dataset does not provide an underlying train-test split, so we will do the split by ourselves.

We would like to try PCA as a defense against adversarial examples. Note that the given dataset was already pre-processed with normalization and dimension reduction [1, 13], so we won't perform additional feature extractions, especially given the fact that the dimension of digit images is low enough (8x8 pixels).

4 Methods

For SVM, our hypothesis is still the hyperplane. The loss function is the hinge loss and optimization will be done by the QP solver implemented by sklearn. We will also use a simple convolutional neural network (CNN), where the loss function is the cross-entropy loss and we will use backpropagation and the Adam optimizer.

We will evaluate our models with adversarial examples generated by the Adversarial Robustness Toolbox (ART) [9]. More specifically, we will use the method called Poisoning Attack provided by ART [6, 3].

In order to combat adversarial examples, we first plan to identify the outliers in the data with a naive approach. For example, we can sample several test hyperplanes and identify $(1 + \epsilon)\gamma$ percents of outliers (providing there are γ percents of outliers). Then we apply the classical SVM algorithm on the inliers. We would also use data transformation and stability training approaches [2, 14].

Another general way to handle adversarial samples is that we employ a loss function (may or may not equal the loss function of the original problem) on each data point, then according to rate of adversarial samples γ, we guess the the set of adversarial samples when applying a hypothesis. We will explore ways to implement the guessing method. For example, in training SVM, given the current hyperplane H, we can remove the furthest γ fraction points to H and evaluate the cost function in the rest of data points. In conclusion, our algorithm will dynamically guess the adversary samples, update the training data set, and retrain the model, which adapts the idea of feedback learning [12].

One baseline of the task is the accuracy achieved by a classical SVM / CNN model without robustness approaches. We expect our algorithm to obtain higher accuracy. Another baseline is the accuracy of a classical SVM / CNN model run only on the benign examples. If our algorithms can obtain close accuracy, we can claim our algorithm is robust.

5 Deliverables

5.1 Must accomplish

1. Obtain the digits dataset; detect and eliminate outliers.
2. Perform data transformations (e.g. PCA) on training data [2].
3. Train SVM and evaluate it on the data attacked by adversarial examples generated by the ART toolbox; compare it with the baseline SVM.

5.2 Expect to accomplish

1. Go through the same process as above with Neural Network (in particular, CNN).
2. Develop a robust training algorithm adapted from the idea of feedback learning [12].
3. Evaluate the performance of the models against baseline.

5.3 Would like to accomplish

1. Use SVM kernel methods to handle adversaries [8].
2. Experiment with other stability training algorithms [14].
3. Experiment with the denoising and verification ensembles approach for CNN [4].

References

[1] Ethem Alpaydin and Cenk Kaynak. CASCADING CLASSIFIERS. page 7.

[2] Arjun Nitin Bhagoji, Daniel Cullina, Chawin Sitawarin, and Prateek Mittal. Enhancing Robustness of Machine Learning Systems via Data Transformations. *arXiv:1704.02654 [cs]*, November 2017. arXiv: 1704.02654.

[3] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines, 2013.

[4] Ka Ho Chow, Wenqi Wei, Yanzhao Wu, and Ling Liu. Denoising and verification cross-layer ensemble against black-box adversarial attacks. *CoRR*, abs/1908.07667, 2019.

[5] Thomas G. Dietterich. Steps Toward Robust Artificial Intelligence. *AI Magazine*, 38(3):3–24, October 2017.

[6] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain, 2019.

[7] Peter J. Huber and Elvezio Ronchetti. *Robust statistics*. Wiley series in probability and statistics. Wiley, Hoboken, N.J, 2nd ed edition, 2009. OCLC: ocn236325889.

[8] Yue Ma, Yiwei He, and Yingjie Tian. Online robust lagrangian support vector machine against adversarial attack. *Procedia Computer Science*, 139:173 – 181, 2018. 6th International Conference on Information Technology and Quantitative Management.

[9] Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian Molloy, and Ben Edwards. Adversarial robustness toolbox v1.2.0. *CoRR*, 1807.01069, 2018.

[10] Jonas Rauber, Wieland Brendel, and Matthias Bethge. Foolbox: A Python toolbox to benchmark the robustness of machine learning models. *arXiv:1707.04131 [cs, stat]*, March 2018. arXiv: 1707.04131.

[11] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, May 2019.

[12] Chang Song, Zuoguan Wang, and Hai Li. Feedback Learning for Improving the Robustness of Neural Networks. *arXiv:1909.05443 [cs, stat]*, September 2019. arXiv: 1909.05443.

[13] L. Xu, A. Krzyzak, and C.Y. Suen. Methods of combining multiple classifiers and their applications to handwriting recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(3):418–435, June 1992.

[14] Stephan Zheng, Yang Song, Thomas Leung, and Ian Goodfellow. Improving the Robustness of Deep Neural Networks via Stability Training. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4480–4488, Las Vegas, NV, USA, June 2016. IEEE.