Technical Note: Adversarial ML

Kyle A. Simpson

4th November 2021

Adversarial machine learning is a family of techniques and approaches which aim to trick a machine learning model to produce an incorrect output for a label (Papernot *et al.*, 2018). These attacks typically assume a white-box attacker (i.e., one who has direct read access to the ML model), who is able to use their knowledge of the ML model to subtly alter an input to produce this mislabelling. Typically, an attack is formalised as an optimisation problem in terms of the underlying model (which can be solved) via a stochastic optimiser like *Adam* (Kingma & Ba, 2014). The constraint to be minimised is some distance metric in $\ell_{\{0,1,2,\ldots,\infty\}}$ between the altered data and its original.

An attacker's goal, however, may be either to induce *any* incorrect label or a specific one; where the former can usually be achieved with smaller change to the input. Typically, these alterations aim to be so subtle as to be unnoticeable to a human operator. These adversarial examples typically occur very close to the decision hyperplane; applying too much noise can either accidentally 'push' the data into a classification the attacker did not desire, or it may become humanly perceptible.

In reality, this problem has been known to security experts for a much longer time under the moniker of *evasion attacks* (Barreno *et al.*, 2006). The context for these evasions includes cases as simple as spam filter avoidance, and as complex as self-modifying and virtualisation-aware malware (Copty *et al.*, 2018). The term does not purely cover ML-based approaches in this context.

Defence Defences against these techniques are regularly proposed (Cao & Gong, 2017; Papernot *et al.*, 2016; Smutz & Stavrou, 2016; Zhang *et al.*, 2020), then defeated in short order (Carlini & Wagner, 2017; Tramèr *et al.*, 2020). Broadly speaking, these have included ensembles of classifiers, ensembles of examples (e.g., by adding noise to inputs), and distilling vulnerable models into smaller representations. The manner in which the model is used or learns doesn't offer any defence, as *reinforcement learning* models remain vulnerable (Huang *et al.*, 2017). Most of this research concerns neural network-based approaches, but historical works have examined classical ML under the same lens.

Recent work (Tramèr *et al.*, 2020) suggests an inherent balancing act between sensitivity/invariance-based attacks—in that defence against one creates a vulnerability against the other. Sensitivity attacks are what we usually consider in this

family (a small change which doesn't impact the input's true label), while invariance attacks use a change which *would* change the true label, but is performed in such a way that the model still outputs the old label. The defence in question would be against attacks within some ℓ_p norm ball (i.e., similar pixel/state similarity)—with the findings suggesting that a 'robust' neural network is even more sensitive than an undefended one.

Examples in networks Smutz and Stavrou (2016) examined an ensemble-based defence on top of the PDFrate (PDF malware) and Drebin (Android executables) malware detection systems. Both of these platforms had well-established adversarial attacks (Maiorca *et al.*, 2013; Srndic & Laskov, 2014), built around the constraint that core exploit functionality must be preserved.

Furthermore, it should be noted that the *white-box* requirement can be discarded in network-facing or observable models. Jagielski *et al.* (2020) and Tramèr *et al.* (2016) have shown that visibility of input-output pairs can allow neural network parameters to be reverse-engineered, and that attacks computed on these surrogates transfer to the target.

I examined some recent work (Alhajjar *et al.*, 2020) which claims to use *genetic algorithms*, *Monte Carlo methods*, and *generative adversarial networks* to help samples evade a NIDS. Sadly, it seems to perturb the output of other feature extractors, which isn't particularly interesting. I list it here purely for completeness.

At this point, I'm having trouble finding anything targetting data-driven network techniques–I feel like there should be a rich baseline here? For instance, you'd think it might be possible to trick ML-driven AQM to punish legitimate flows or similar (i.e., novel DDoS rather than just ML evasion). Indeed, others (Pierazzi *et al.*, 2020) are noticing that most research remains in *feature-space* (i.e., techniques and mathematics), rather than *problem-space*—and most of these are in malware.

References

Alhajjar, E., Maxwell, P. & Bastian, N. D. (2020). Adversarial machine learning in network intrusion detection systems. *CoRR*, *abs/2004.11898*. https://arxiv.org/abs/2004.11898

Barreno, M., Nelson, B., Sears, R., Joseph, A. D. & Tygar, J. D. (2006). Can machine learning be secure? (F. Lin, D. Lee, B. P. Lin, S. Shieh & S. Jajodia, Eds.). *Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security, ASIACCS 2006, Taipei, Taiwan, March 21-24, 2006*, 16–25. https://doi.org/10.1145/1128817.1128824

Cao, X. & Gong, N. Z. (2017). Mitigating evasion attacks to deep neural networks via region-based classification. *Proceedings of the 33rd Annual Computer Security Applications Conference, Orlando, FL, USA, December 4-8, 2017*, 278–287. https://doi.org/10.1145/3134600.3134606

Carlini, N. & Wagner, D. A. (2017). Towards evaluating the robustness of neural networks. *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, 39–57. https://doi.org/10.1109/SP.2017.49

Copty, F., Danos, M., Edelstein, O., Eisner, C., Murik, D. & Zeltser, B. (2018). Accurate malware detection by extreme abstraction. *Proceedings of the 34th Annual Computer Security Applications Conference, ACSAC 2018, San Juan, PR, USA, December 03-07, 2018*, 101–111. https://doi.org/10.1145/3274694.3274700

Huang, S. H., Papernot, N., Goodfellow, I. J., Duan, Y. & Abbeel, P. (2017). Adversarial attacks on neural network policies. *CoRR, abs/1702.02284*. http://arxiv.org/abs/1702.02284

Jagielski, M., Carlini, N., Berthelot, D., Kurakin, A. & Papernot, N. (2020). High accuracy and high fidelity extraction of neural networks. In S. Capkun & F. Roesner (Eds.), *29th USENIX security symposium, USENIX security 2020, august 12-14, 2020* (pp. 1345–1362). USENIX Association. https://www.usenix.org/conference/usenixsecurity20/presentation/jagielski

Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. *CoRR, abs/1412.6980*. http://arxiv.org/abs/1412.6980

Maiorca, D., Corona, I. & Giacinto, G. (2013). Looking at the bag is not enough to find the bomb: An evasion of structural methods for malicious PDF files detection. In K. Chen, Q. Xie, W. Qiu, N. Li & W. Tzeng (Eds.), *8th ACM symposium on information, computer and communications security, ASIA CCS '13, hangzhou, china - may 08 - 10, 2013* (pp. 119–130). ACM. https://doi.org/10.1145/2484313.2484327

Papernot, N., McDaniel, P. D., Sinha, A. & Wellman, M. P. (2018). Sok: Security and privacy in machine learning. *2018 IEEE European Symposium on Security and Privacy, EuroS&P 2018, London, United Kingdom, April 24-26, 2018*, 399–414. https://doi.org/10.1109/EuroSP.2018.00035

Papernot, N., McDaniel, P. D., Wu, X., Jha, S. & Swami, A. (2016). Distillation as a defense to adversarial perturbations against deep neural networks. *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016*, 582–597. https://doi.org/10.1109/SP.2016.41

Pierazzi, F., Pendlebury, F., Cortellazzi, J. & Cavallaro, L. (2020). Intriguing properties of adversarial ML attacks in the problem space. *2020 IEEE Symposium on Security and Privacy, SP 2020, San Francisco, CA, USA, May 18-21, 2020*, 1332–1349. https://doi.org/10.1109/SP40000.2020.00073

Smutz, C. & Stavrou, A. (2016). When a tree falls: Using diversity in ensemble classifiers to identify evasion in malware detectors. *23rd Annual Network and Distributed System Security Symposium, NDSS 2016, San Diego, California, USA, February 21-24, 2016*. http://wp.internetsociety.org/ndss/wp-content/uploads/sites/25/2017/09/when-tree-falls-using-diversity-ensemble-classifiers-identify-evasion-malware-detectors.pdf

Srndic, N. & Laskov, P. (2014). Practical evasion of a learning-based classifier: A case study. *2014 IEEE Symposium on Security and Privacy, SP 2014, Berkeley, CA, USA, May 18-21, 2014*, 197–211. https://doi.org/10.1109/SP.2014.20

Tramèr, F., Behrmann, J., Carlini, N., Papernot, N. & Jacobsen, J. (2020). Fundamental tradeoffs between invariance and sensitivity to adversarial perturbations. *CoRR*, *abs/2002.04599*. https://arxiv.org/abs/2002.04599

Tramèr, F., Zhang, F., Juels, A., Reiter, M. K. & Ristenpart, T. (2016). Stealing machine learning models via prediction apis (T. Holz & S. Savage, Eds.). *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016.*, 601–618. https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/tramer

Zhang, H., Chen, H., Xiao, C., Gowal, S., Stanforth, R., Li, B., Boning, D. S. & Hsieh, C. (2020). Towards stable and efficient training of verifiably robust neural networks. *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. https://openreview.net/forum?id=Skxuk1rFwB