# Advanced Topic A06: *Adversarial Attacks on ML Models*

Data Science with Python – BSc Advanced Lectures

Joerg Osterrieder


10 Minutes

# Motivation: Models That Fail on Purpose

**Imperceptible perturbations can fool state-of-the-art models**

- Goodfellow et al. (2014): add a tiny structured noise to an image and a 99% confident classifier changes its prediction entirely
- The perturbation is invisible to humans but exploits the geometry of the loss landscape
- In finance: adversarial examples threaten fraud detectors, credit scorers, and NLP-based compliance filters
- Understanding attacks is prerequisite to building robust models

**Adversarial robustness is a safety property, not an accuracy property**

# The Threat Model

**Clarifying who attacks what and how**

- **White-box attack:** attacker has full access to model weights and gradients
- **Black-box attack:** attacker can only query the model (transfer or score-based)
- **Targeted attack:** force the model to predict a specific wrong class
- **Untargeted attack:** any wrong prediction is sufficient
- **Perturbation budget** $\varepsilon$**:** maximum allowed change (measured in $\ell_\infty$, $\ell_2$, or $\ell_1$ norm)

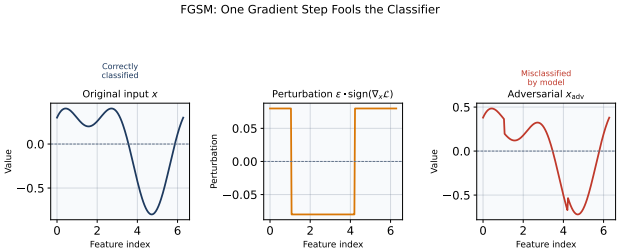
**Define the threat model before evaluating any defence – otherwise comparisons are meaningless**

# Fast Gradient Sign Method (FGSM)

**One gradient step in the direction that maximises the loss**

$$x_{\text{adv}} = x + \varepsilon \cdot \text{sign}\big(\nabla_x \mathcal{L}(\theta, x, y)\big)$$

- $\nabla_x \mathcal{L}$: gradient of the loss with respect to the *input* (not the weights)
- sign$(\cdot)$: takes $+1$ or $-1$ per pixel, so the perturbation is exactly $\varepsilon$ in $\ell_\infty$
- Single forward-backward pass: fast, cheap, and surprisingly effective
- The perturbation is structured (aligned with the decision boundary), not random




**FGSM: the simplest white-box attack; used as a baseline for all robustness evaluations**

# Projected Gradient Descent (PGD)

**FGSM iterated with projections back into the $\varepsilon$-ball**

$$x^{(t+1)} = \Pi_{B(x,\varepsilon)}\big(x^{(t)} + \alpha \cdot \operatorname{sign}(\nabla_x \mathcal{L})\big)$$

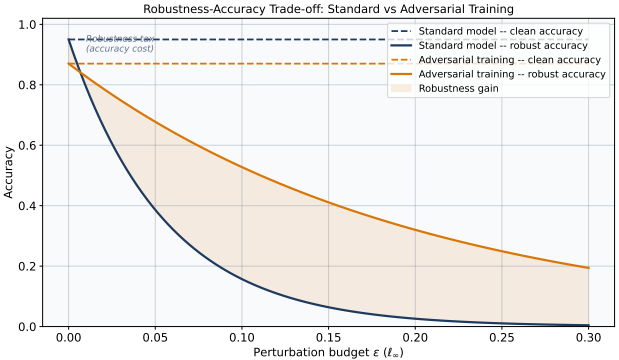
- $\Pi_{B(x,\varepsilon)}$: projection onto the $\ell_\infty$ ball of radius $\varepsilon$ around $x$
- Typically 10–40 steps with step size $\alpha \approx \varepsilon/5$
- Madry et al. (2018): PGD is a universal first-order adversary; defence against PGD implies defence against all gradient attacks
- More expensive than FGSM but finds much stronger perturbations

**PGD: the standard benchmark for adversarial robustness – if it passes PGD it is likely robust**

# Epsilon vs Accuracy Trade-off

**Every model faces a robustness-accuracy frontier**

- Standard (clean) accuracy drops monotonically as $\varepsilon$ increases for the adversary
- Adversarially trained models: lower clean accuracy but much higher robust accuracy at large $\varepsilon$
- The gap between clean and robust accuracy is the "robustness tax"
- No free lunch: you cannot have both maximum clean accuracy and maximum robustness with current architectures




**Choosing $\varepsilon$ is a risk management decision: higher $\varepsilon$ tolerance costs accuracy points**

# CW Attack and Beyond

**Optimisation-based attacks find minimal perturbations**

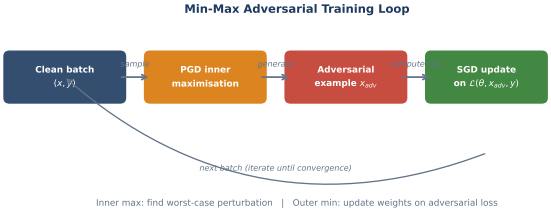
- Carlini-Wagner (CW): minimise $\|x_{\text{adv}} - x\|_2$ subject to the model predicting $t \neq y$
- Formulated as an unconstrained optimisation via a change of variable: $x = \frac{1}{2}(\tanh(w) + 1)$
- CW finds smaller perturbations than FGSM/PGD at the cost of more compute
- **AutoAttack** (2020): parameter-free ensemble of four attacks; now the standard for reporting adversarial robustness in papers
- **Square Attack**: score-based black-box attack; needs only model outputs

**AutoAttack is the community benchmark: always evaluate robust accuracy using AutoAttack**

# Adversarial Training

**Augmenting training data with adversarial examples**

- Madry et al. (2018): solve the min-max problem $\min_\theta \mathbb{E}_{(x,y)}\left[\max_{\delta:\|\delta\|_\infty \leq \varepsilon} \mathcal{L}(\theta, x+\delta, y)\right]$
- Inner maximisation: generate a PGD adversarial example during each training step
- Outer minimisation: standard SGD update on the adversarial loss
- Result: the decision boundary moves away from training points, improving robustness
- Cost: 5–10x longer training; slight drop in clean accuracy




**Adversarial training is currently the most effective certified defence strategy**

# Certifiable Defences

**Proving that no attack within the budget can fool the model**

- Empirical robustness (PGD, AutoAttack) does not prove robustness: a better attack may still exist
- **Randomised smoothing:** add Gaussian noise at inference; the smoothed classifier is certifiably robust within an $\ell_2$ radius $R$ (Cohen et al. 2019)
- **Interval bound propagation (IBP):** propagate input intervals through the network; if the worst-case output is still correct, the prediction is certified
- Certification is computationally expensive; current certificates are loose for large networks

**A certified model gives a mathematical guarantee, not just an empirical one**

# Transferability: Black-Box Attacks

**Adversarial examples transfer across models and architectures**

- Adversarial examples crafted on model A often fool model B (even with different weights)
- This enables black-box attacks: craft on a surrogate, attack the target
- Transferability is stronger between architecturally similar models
- In finance: an attacker queries a public API to build a surrogate model, then crafts adversarial transactions that evade the live fraud detector
- Defence: use ensembles with diverse architectures to reduce transfer rate

**Transferability: why black-box systems are not safe from white-box attack methodology**

# Adversarial Attacks on NLP and Finance Text

**Text-based models face discrete, semantics-preserving perturbations**

- Image attacks perturb pixels continuously; text attacks must work in discrete token space while preserving meaning and grammatical correctness
- **Word substitution attacks**: replace words with synonyms that fool the classifier but are semantically equivalent to humans (`TextFooler`, `BERT-Attack`)
- **Character-level attacks**: insert typos, unicode homoglyphs, or zero-width spaces that bypass tokenisers without changing visible text
- Finance NLP: compliance filters based on BERT can be fooled by synonym substitution in suspicious communications; regulators increasingly require adversarial testing of NLP-based surveillance systems
- Defence: certified smoothing via word-substitution certificates; ensembles trained on synonym-augmented data

**NLP adversarial attacks: synonym substitution preserves meaning for humans but changes model predictions**

# Physical-World and Patch Attacks

**Adversarial perturbations that survive printing and real-world capture**

- Digital attacks require direct pixel access; physical attacks must survive real-world distortions (lighting, angle, JPEG compression, camera noise)
- **Adversarial patches** (Brown et al. 2017): a small sticker placed anywhere in a scene fools a classifier at any distance; the patch is universal and context-independent
- **Stop sign attacks**: placing specific patches on traffic signs causes autonomous vehicles to misclassify them; safety-critical implication
- Physical attacks are optimised over a distribution of transformations (rotation, scale, brightness shifts) using expectation-over-transformation training
- Finance relevance: QR codes, cheque image processing, and document classification are all vulnerable to printed adversarial perturbations

**Physical adversarial attacks bridge the gap between digital robustness and real-world safety**

# Defences Beyond Adversarial Training

**Other strategies for building more robust models**

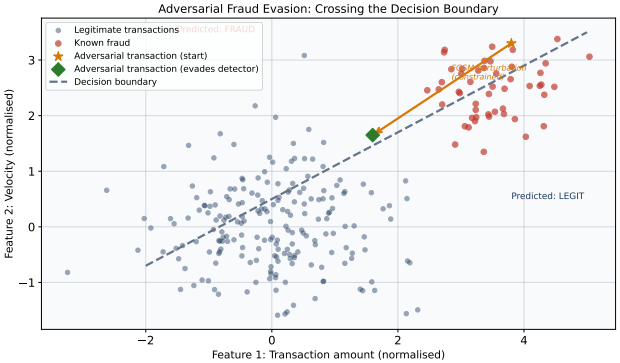
- **Input preprocessing:** JPEG compression, denoising, feature squeezing – cheap but easily circumvented by adaptive attacks
- **Certified smoothing:** mathematically proven robustness radius (see prior slide)
- **Ensemble diversity:** multiple models with varied architectures reduce transfer
- **Anomaly detection:** flag inputs that are far from the training distribution
- **Adversarial detection:** a binary classifier that identifies adversarial inputs before they reach the main model

**Never evaluate defences without an adaptive attack – non-adaptive evaluation is misleading**

# Finance Application: Adversarial Fraud Evasion

**Attackers craft transactions that evade fraud detection**

- A fraud ring learns the decision boundary of a bank's fraud classifier by probing it
- FGSM-style perturbations applied to transaction features: amount, time-of-day, merchant category
- Constraint: the perturbation must be feasible (real transactions, valid amounts)
- Finance adversarial attacks are constrained: features are correlated and bounded




**Adversarial robustness in fraud detection is an arms race – defences must be continuously updated**

# Practical Constraints in Finance

**Financial features are not images – perturbation constraints are tighter**

- Image attacks perturb raw pixels; financial attacks must respect feature semantics
- Amount cannot be negative; timestamps must be monotone; card numbers have check digits
- These "semantic constraints" severely limit the attack space
- Constrained PGD: project onto the feasible set after each gradient step
- Consequence: financial fraud models may be empirically harder to attack than unconstrained models, but domain-specific attacks remain a real threat

**Always define the feasible perturbation set with domain experts before evaluating robustness**

# Code Pointer

**Adversarial attacks in Python**

- **CleverHans**: TensorFlow library; implements FGSM, PGD, CW, and more
- **Foolbox**: PyTorch/TF/JAX; over 30 attacks; clean API (`attack = fb.attacks.FGSM(); advs, _, _ = attack(model, inputs, labels, epsilons=[0.1])`)
- **ART (Adversarial Robustness Toolbox)**: IBM; supports scikit-learn models too; useful for tabular financial data
- AutoAttack: `pip install autoattack`; single function call for benchmark evaluation
- Robustbench: leaderboard of certified robust models with download APIs

**For tabular data use ART; for neural networks use Foolbox or AutoAttack**

# Limitations and Open Problems

**The field has not solved adversarial robustness**

- Robust accuracy on CIFAR-10 at $\varepsilon = 8/255$: best model is around 71% vs 99% clean
- The robustness-accuracy trade-off may be fundamental rather than a current limitation
- Certified defences do not yet scale to large transformers or production-size networks
- Most evaluations use $\ell_\infty$ or $\ell_2$ balls; real attacks may use semantic spaces
- Open question: does adversarial robustness generalise across distribution shifts?

**Adversarial robustness research is ongoing: treat defences as mitigations, not solutions**

# Summary and Open Questions

**Three things to remember**

- FGSM and PGD exploit the gradient of the loss with respect to the input to construct imperceptible perturbations; adversarial training (min-max) is the most effective current defence
- Robustness and accuracy trade off: choosing the perturbation budget $\varepsilon$ is a risk management decision, not a technical one
- Finance adversarial attacks are constrained by feature semantics but remain a real threat; ART is the toolkit of choice for tabular models

**Open questions:**

1. Is the robustness-accuracy trade-off provably unavoidable?
2. Can certified defences scale to transformer-sized models?
3. How do we audit adversarial robustness in production fraud systems?

**Further reading: Goodfellow et al. (2015) FGSM; Madry et al. (2018) PGD; Carlini & Wagner (2017)**