

TITLE: Cryptography and Machine Learning

LECTURER: Alon Rosen

WHEN AND WHERE: Tuesdays 16:00–19:00, TAU, Shenkar 222.

PREREQUISITE: Algorithms, probability, linear algebra, complexity, and mathematical maturity. Cryptography is helpful but not required.

TEACHING MATERIAL:

- Understanding Machine Learning: From Theory to Algorithms. Shalev-Shwartz and Ben-David.
- Pseudorandom Functions: Three Decades Later. Bogdanov and Rosen.
- A Decade of Lattice Cryptography. Peikert.
- Cryptography and ML: Foundations and Frontiers. Goldwasser and Vaikuntanathan, MIT 6.S976, Spring 2026.
- Great Algorithms. Bogdanov, CSI 4103/5138, University of Ottawa.
- Interaction with and tutoring by language models is encouraged, **except for homework**.

HOMEWORK AND GRADING: Three problem sets (30%), class (or office hours) participation (10%), and a final project w/ a written report (60%).

DESCRIPTION: This course explores the interface between cryptography and machine learning. While modern ML is driven by rapidly evolving architectures and empirical advances, we step back and study principles underlying trust, robustness, and reliability. Drawing on cryptographic thinking, we examine formal models of adversarial behavior and ask what can be guaranteed about learning systems in the presence of computationally bounded adversaries. We aim at a technology-independent perspective.

The course combines conceptual foundations with recent research, moving beyond heuristic defenses toward systems whose security and reliability can be defined and, where possible, proven. Topics include:

- Hardness of learning
- Adversarial robustness
- Backdoors
- Watermarking
- Model extraction
- Cryptography for ML
- Verifiable inference and training
- Provenance
- Low logit rank, sketching, sandbagging, and some mathematics of AI safety

Later topics are yet to be shown to have direct connection to cryptography, but there is hope.

OBJECTIVES AND LEARNING OUTCOMES: Students will learn to:

- Model (some) machine learning in a technologically independent manner.
- Formulate precise security definitions for learning: watermarking, backdoors, verifiability, privacy.
- Apply cryptographic tools (pseudorandomness, signatures, proofs) to machine learning.
- Prove hardness of learning and estimation problems by reduction from standard assumptions.
- Be ready to conduct research at the cryptography/machine learning frontier.

PROGRAM (subject to change): 13 lectures of 3 hours each. Topics are ordered so that no tool is used before it is taught, and difficulty increases over the semester.

Lecture 1: Cryptography and learning

- Introduction and course overview
- Pseudorandom functions and hardness of PAC learning
- Circuit complexity; neural nets as threshold circuits
- Learning parity with noise (LPN)

Lecture 2: Discrete vs. continuous assumptions

- LWE; worst-case hardness; generating fresh samples
- Continuous LWE and Gaussian pancakes
- Statistical vs. computational gaps

Lecture 3: Adversarial robustness in classification

- Accuracy does not imply robustness
- Gradient-based attacks and adversarial training
- Certified and cryptographic robustness
- Invariance attacks; adaptively robust random projections

Lecture 4: Backdoors

- Spectral signatures
- Undetectable backdoors from signatures and hCLWE
- Undetectability vs. power asymmetry; invariance backdoors in deep networks
- Removal without detection

Lecture 5: Hardness of score estimation

- Diffusion; sampling reduces to score estimation
- Learning the score; hardness from hCLWE

Lecture 6: Watermarking

- Reliable detection; the red–green trick
- Undetectable watermarks via keyed Gumbel sampling
- Robust watermarks from pseudorandom codes
- Watermarking diffusion models

Lecture 7: Model extraction and low logit rank

- Functional and distributional extraction
- Outputs as equations; ReLU boundaries reveal hidden neurons
- Distillation: training a surrogate on queried outputs
- The extended logit matrix; low rank in trained language models
- Model extraction from unrelated queries
- Efficient learning from logit queries

Lecture 8: Cryptography for ML

- Encryption and homomorphic encryption
- Secure two-party computation
- Private vector search: PIR and encrypted matrix–vector products (EMVP)
- Private neural-network inference

Lecture 9: Verifiable inference and training

- Interactive proofs: randomized verification, sumcheck, layered circuits
- Succinct arguments; zero knowledge
- Provenance: authentication, proofs of provenance, homomorphic signatures, malleable proofs

Lecture 10: Sketching learning algorithms

- The data deletion problem
- Forward-mode automatic differentiation; sketching circuits via higher-order derivatives

Lecture 11: AI safety

- Sandboxed evaluation of untrusted models
- Sandbagging: strategic underperformance on evaluations; detection via noise injection
- Heuristic arguments; the computational no-coincidence conjecture
- Open problems

Lectures 12–13: Project presentations

- Final project presentations
- Groups of 4–5 students; about 20 minutes per group