EEMCS - SCS

EVOLUTIONARY ALGORITHMS FOR CYBERSECURITY

GUEST LECTURE – APRIL 3, 2025

LUCA MARIOT

UNIVERSITY OF TWENTE.

SUMMARY

- Intro to Cybersecurity
- Cryptography & EA
- Network Intrusion Detection & EA

UNIVERSITY
OF TWENTE.

INTRO TO CYBERSECURITY

UNIVERSITY
OF TWENTE.

THE CIA TRIAD OF SECURITY

When building a secure system, we focus on three aspects [1]:

Confidentiality

Data are protected from unauthorized access/use

Integrity

Data have not been altered by unauthorized parties

Availability

Data are available to authorized parties when needed

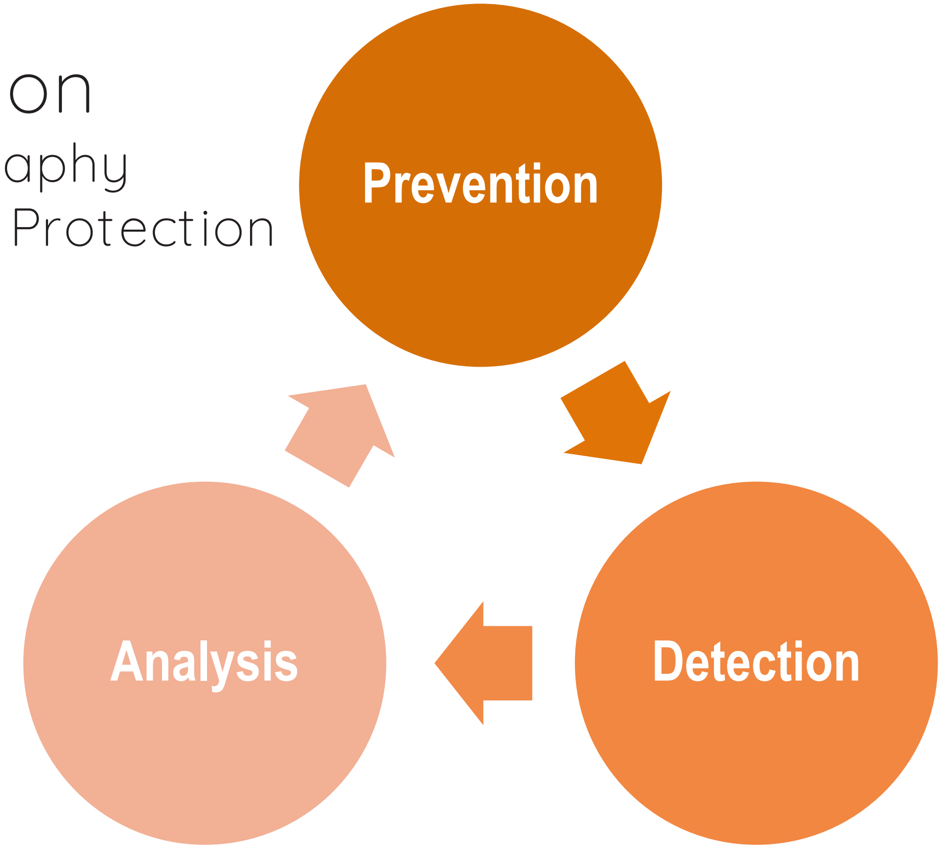
THREAT MODELING

- Threat: potential violation of a security goal
- Attack: intentional violation of a security goal
- Security: protection from attacks vs. cost

Security is economics!

ACHIEVING SECURITY

Prevention
- Cryptography
- Intrusion Protection



Detection
- Intrusion Detection
- Malware Detection

Analysis
- Forensics
- Incident Response

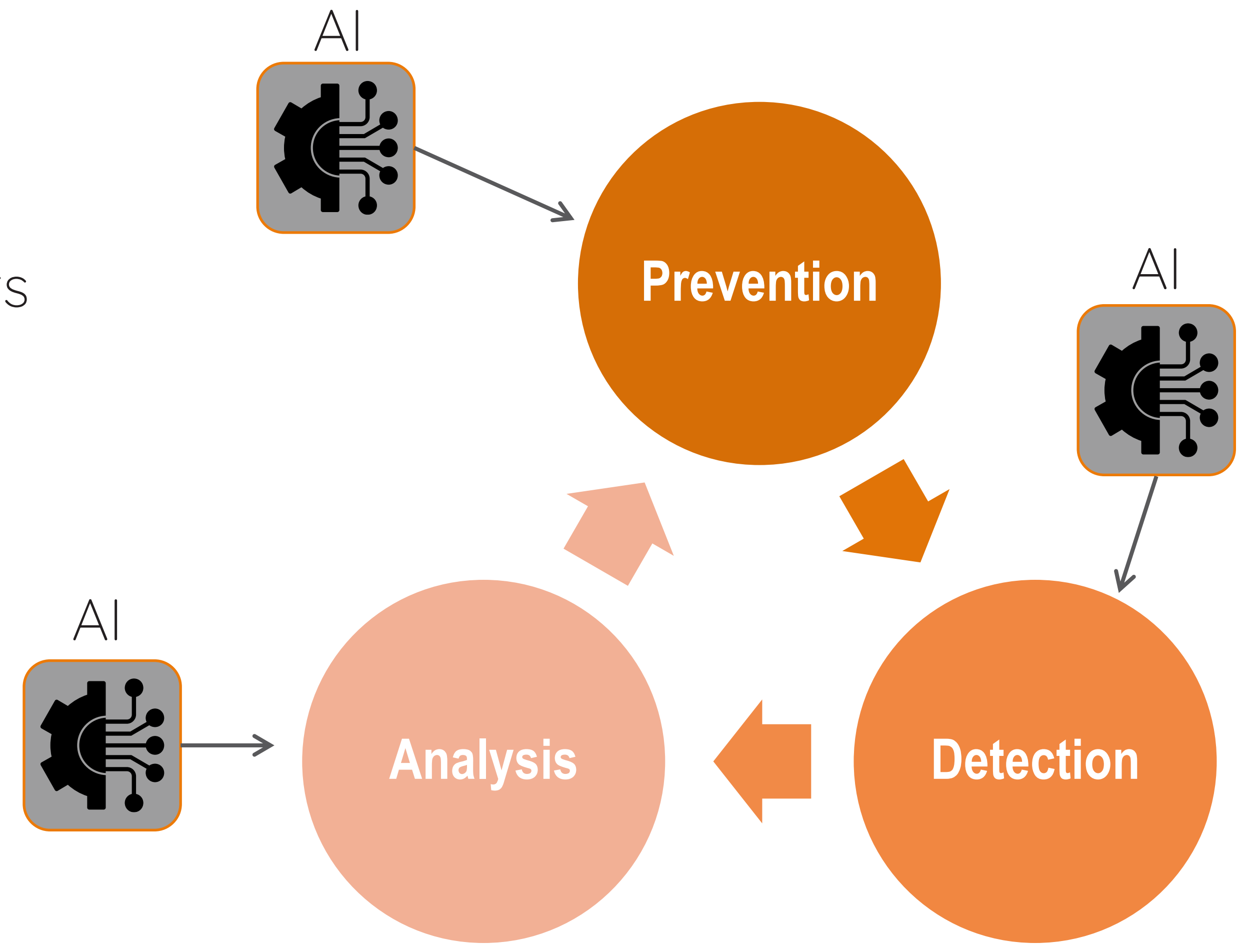
AI FOR SECURITY

Challenges:

- New vulnerabilities, new attack vectors
- Reducing human intervention

Approach: “Smarter” Security

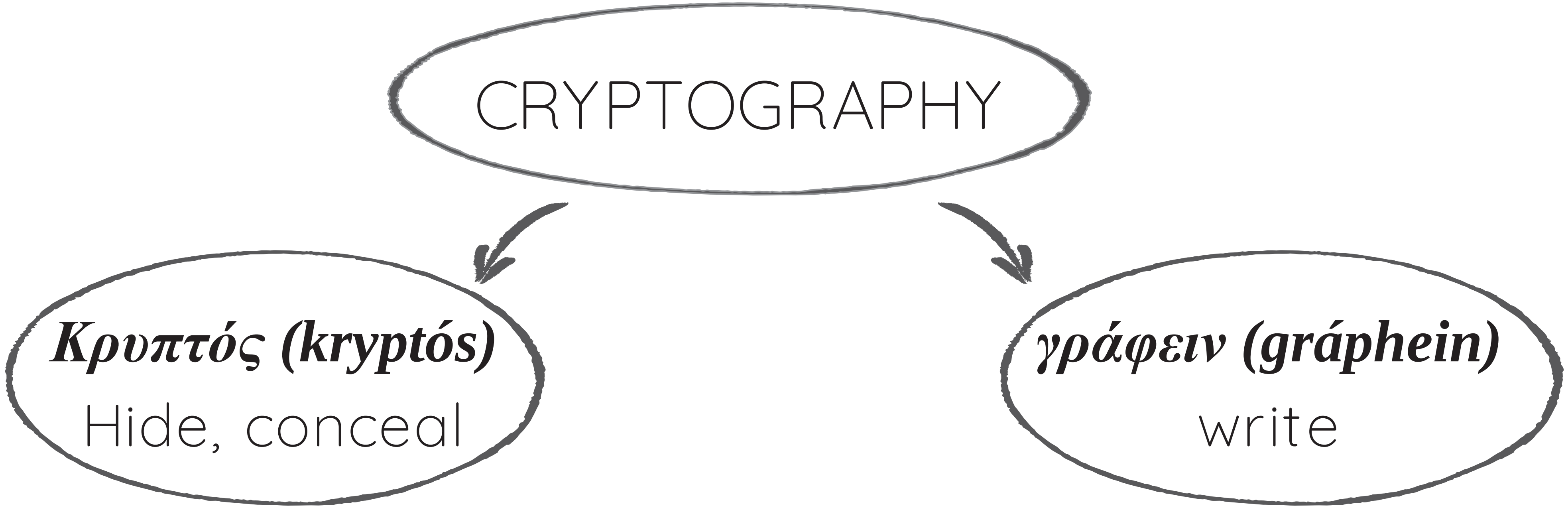
- Automate the design process
- Use AI techniques (e.g. Evolutionary Computing) for the automation



UNIVERSITY OF TWENTE.

CRYPTOGRAPHY

WHAT IS CRYPTOGRAPHY?



Historically: the art of hiding the meaning of messages, with the goal of protecting their confidentiality

THE CIA TRIAD IN CRYPTOGRAPHY

In cryptography, the "A" is usually replaced by Authenticity:

Confidentiality

Data are protected from unauthorized access/use

Integrity

Data have not been altered by unauthorized parties

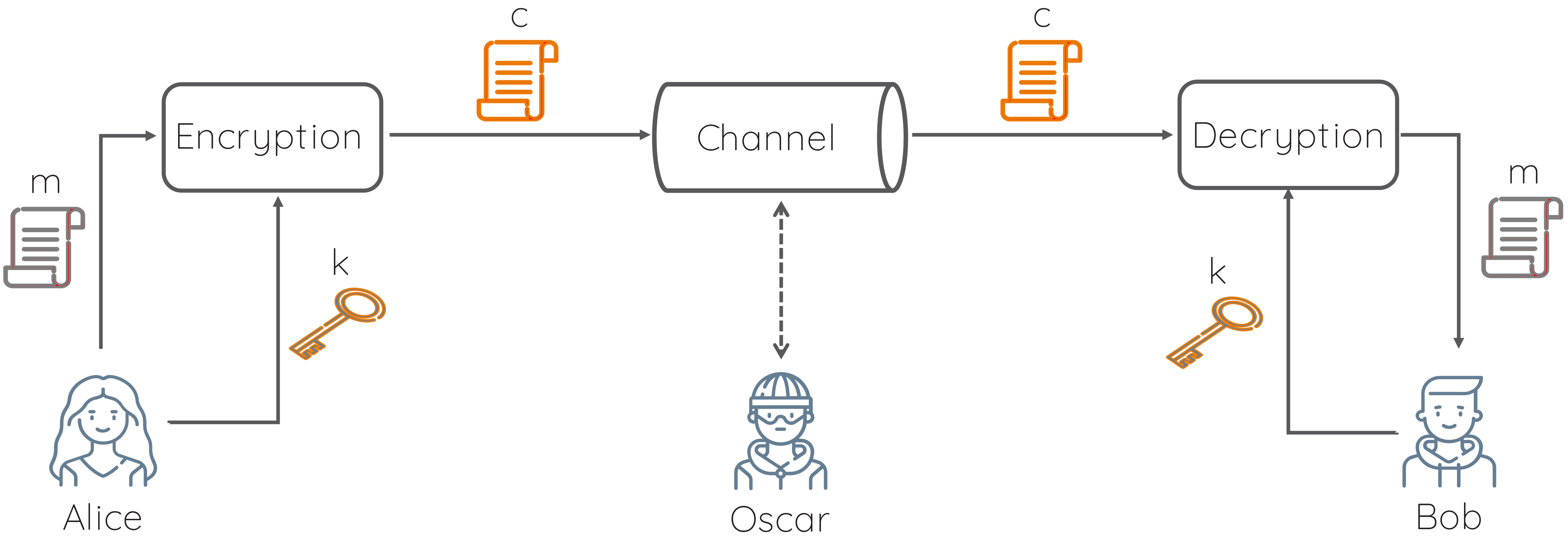
Authenticity

Data have been created by the intended party

Confidentiality → Encryption schemes

Integrity, Authenticity → Authentication codes, Digital signatures

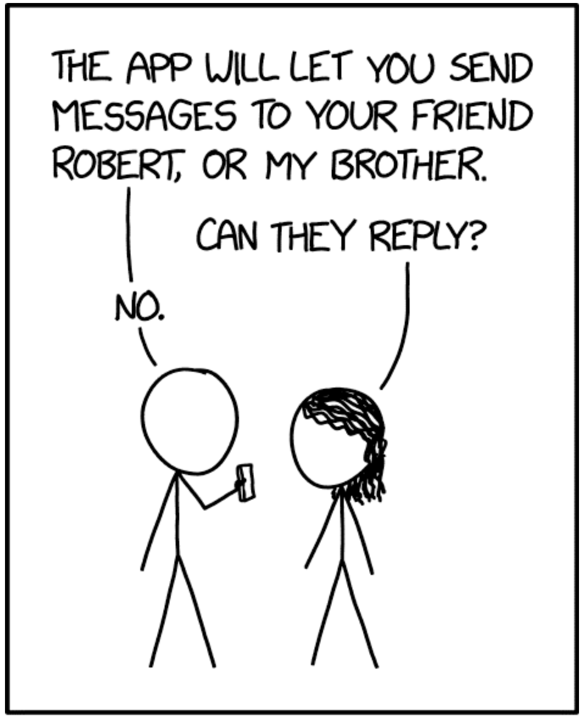
SYMMETRIC-KEY ENCRYPTION SCENARIO



m: plaintext message   k: encryption/decryption key   c: ciphertext message

SYMMETRIC-KEY ENCRYPTION SCENARIO

- The same key k is used both for encryption *and* decryption [6]
- The scheme is "secure" as long as Oscar does not know k
- Requires sharing k *before* communicating
- Here, we just assume Alice and Bob shared k somehow



MY NEW SECURE TEXTING APP ONLY ALLOWS PEOPLE NAMED ALICE TO SEND MESSAGES TO PEOPLE NAMED BOB.

KERCHOFFS'S PRINCIPLE (1883)

- The encryption scheme is known to the attacker [7]
- Security relies *only* on the secrecy of the encryption key

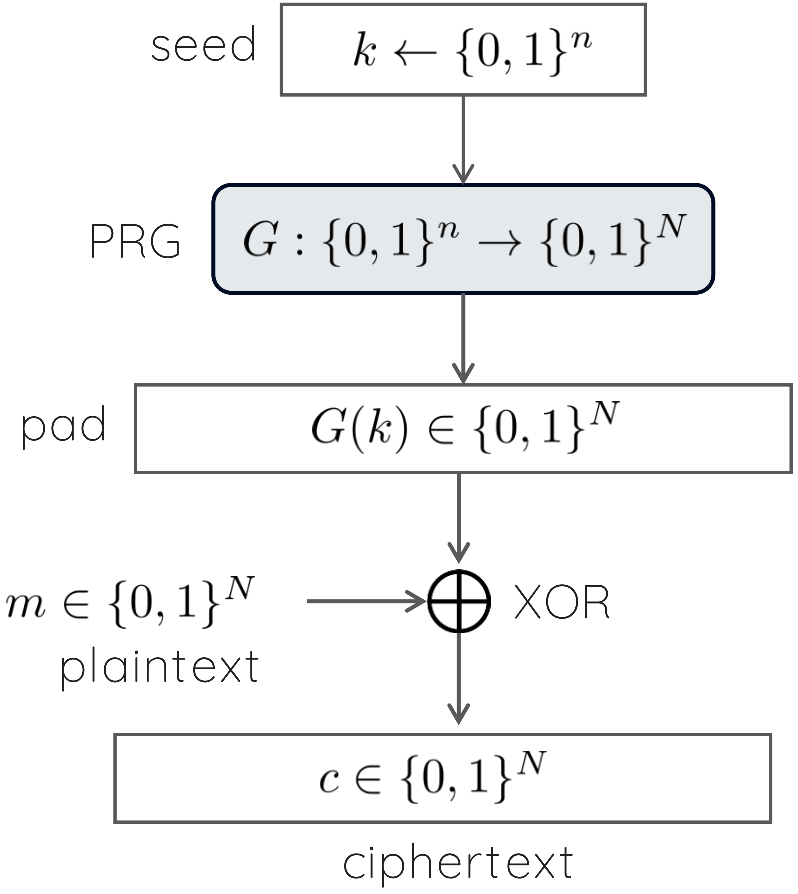
"The encryption/decryption system must not be kept secret, and can be stolen by the enemy without causing any problem."

UNIVERSITY OF TWENTE.

STREAM CIPHERS - ENCRYPTION

- Idea: Alice encode all messages as stream of bits, $m \in \{0,1\}^N$
- A Pseudo Random Generator (PRG) is used to generate a pad $p \in \{0,1\}^N$ of the same length of the message [6]
- The seed of the PRG is the key $k \leftarrow \{0,1\}^n$
- Encryption: Bitwise XOR between message and pad

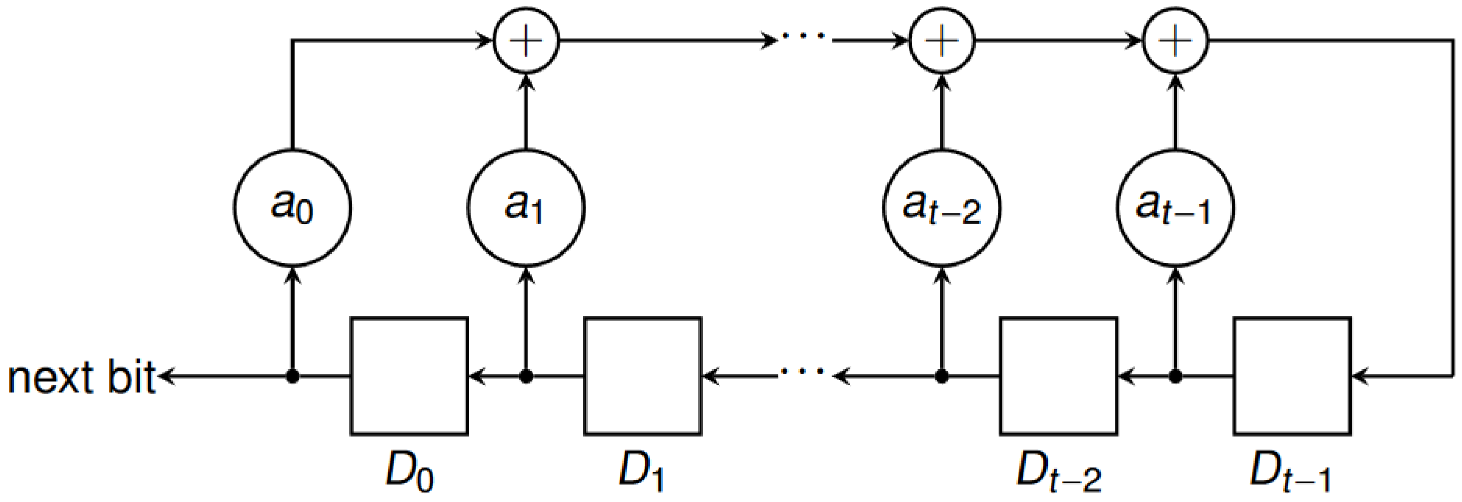
$$c_i := m_i \oplus p_i, \text{ for all } i \in \{1, \cdots, N\}$$



LINEAR FEEDBACK SHIFT REGISTERS (LFSR)

- A device computing a *linear recurring sequence* (LRS)

$$z_i = a_0 \cdot z_{i-t} \oplus a_1 \cdot z_{i-t+1} \oplus \cdots \oplus a_{t-1} \cdot z_{i-1} \qquad a_j, z_j \in \{0, 1\}$$



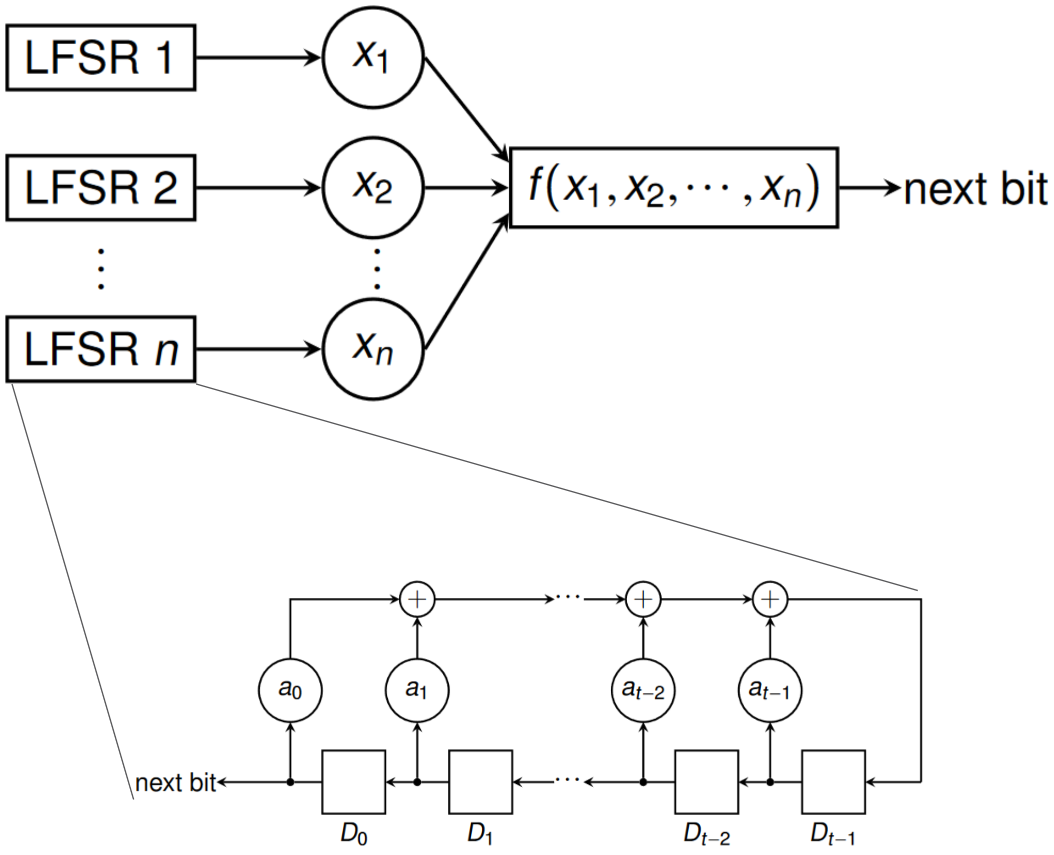
- Problem: very weak as a cryptographic PRG [6]

IMPROVING LFSR – COMBINER MODEL FOR PRG

- Idea: use n LFSRs instead of one [4]
- LFSRs outputs *combined* using a Boolean function:

$$f : \{0,1\}^n \rightarrow \{0,1\}$$

- Security of the PRG: cryptographic properties of $f : \{0,1\}^n \rightarrow \{0,1\}$



UNIVERSITY OF TWENTE.

BOOLEAN FUNCTIONS

- A mapping $f : \{0,1\}^n \rightarrow \{0,1\}$ represented by a *truth table*

(x_1, x_2, x_3)	000	001	010	011	100	101	110	111
$f(x_1, x_2, x_3)$	0	1	1	0	1	0	1	0

- The function must satisfy some properties to resist specific attacks [4]:
  - Balancedness (equal number of 0s and 1s)
  - High Nonlinearity (Hamming distance from linear functions)
  - High algebraic degree, etc. …

EVOLUTIONARY ALGORITHMS FOR BOOLEAN FUNCTIONS

UNIVERSITY
OF TWENTE.

OPTIMIZATION PROBLEM

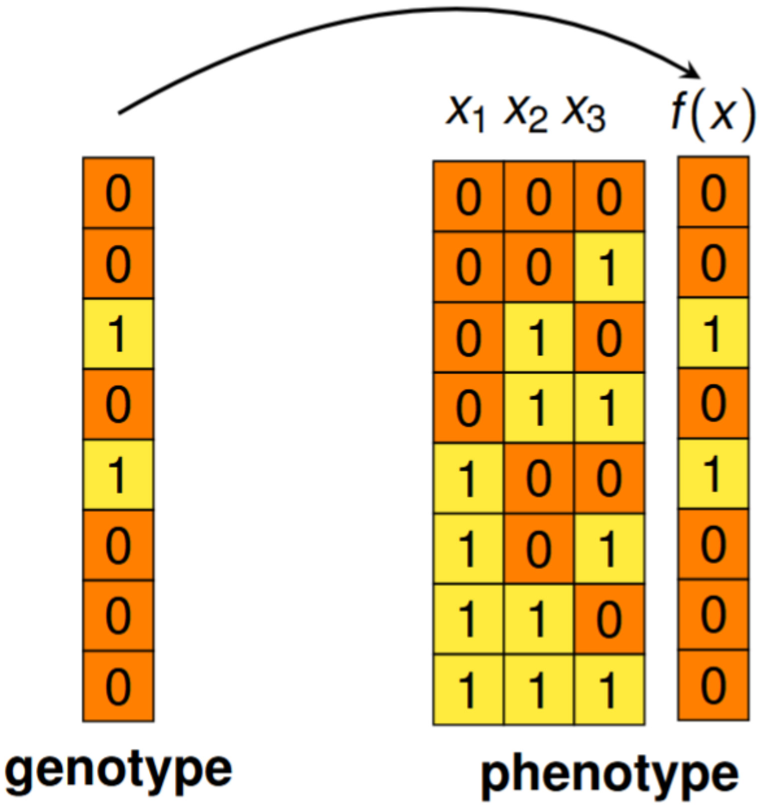
- Given $n \in \mathbb{N}$, how do we fill the table so that f is balanced and highly nonlinear?

(x_1, x_2, x_3)	000	001	010	011	100	101	110	111
$f(x_1, x_2, x_3)$	?	?	?	?	?	?	?	?

$$f^* = \text{argmax}_{f:\{0,1\}^n \to \{0,1\}} (BAL(f) + NL(f))$$

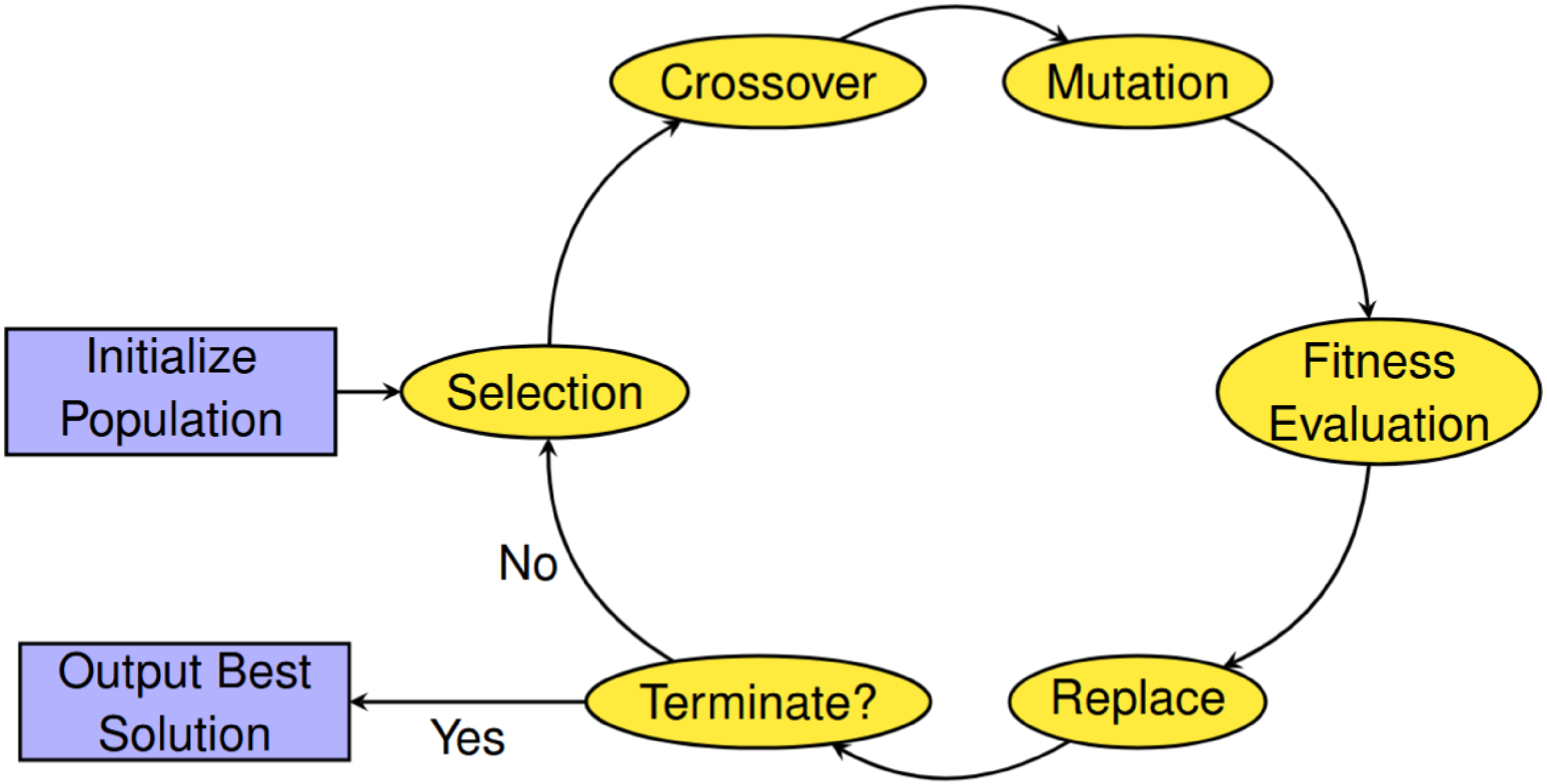
- The truth table has size 2^n so there are 2^{2^n} combinations
- For concrete security, we need $n > 13$
- But exhaustive search is already unfeasible for $n > 5$!

EVOLUTIONARY ALGORITHMS (EA)



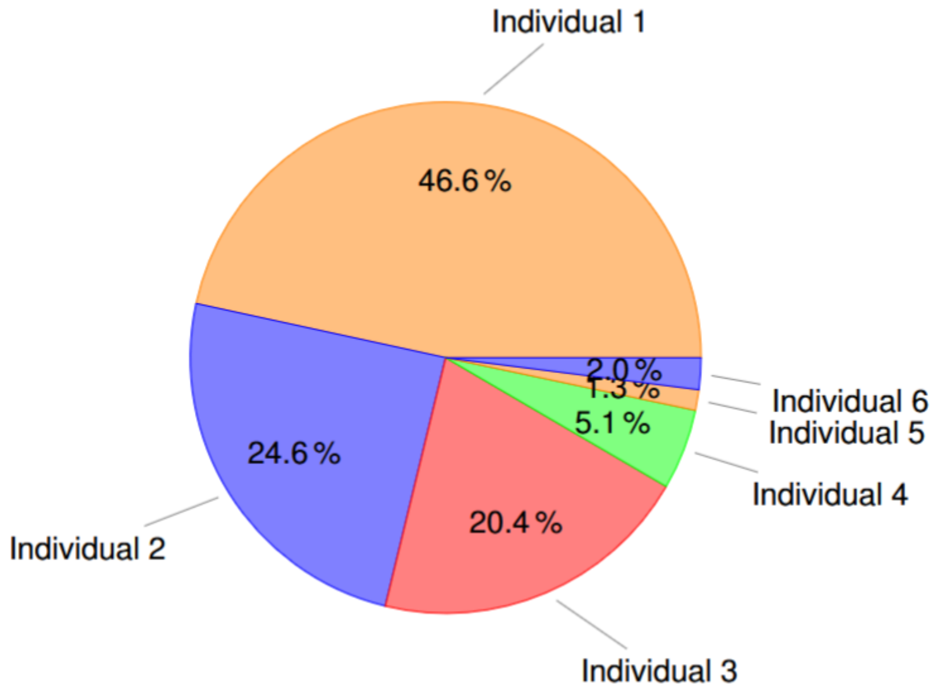
- Optimization algorithms loosely based on evolutionary principles
- Genetic Algorithms (GA): introduced by John Holland (1975)
- GA genotype: fixed-length bitstrings
- phenotype: truth table of f [5]

THE EA LOOP



UNIVERSITY OF TWENTE.

SELECTION



- Roulette-Wheel: selection probability proportional to individual's fitness

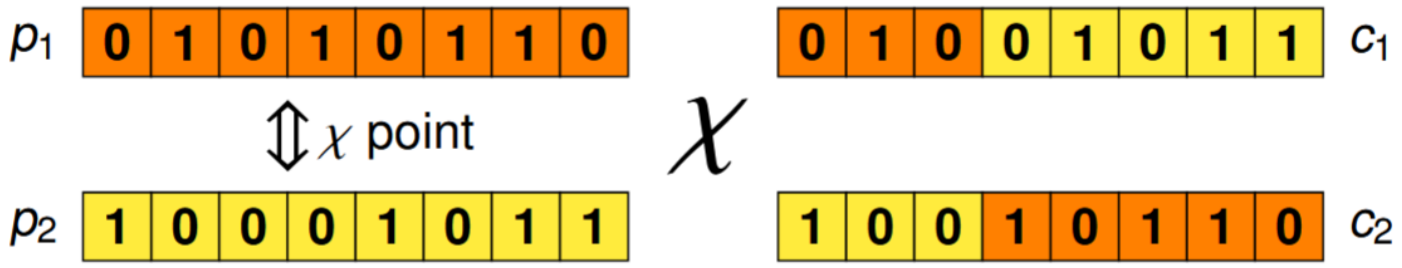
- Tournament: select the fittest individual from a random sample of t individuals

UNIVERSITY OF TWENTE.

CROSSOVER

- Idea: recombine the genes of two parents (Exploitation)

GA Example: One-Point Crossover

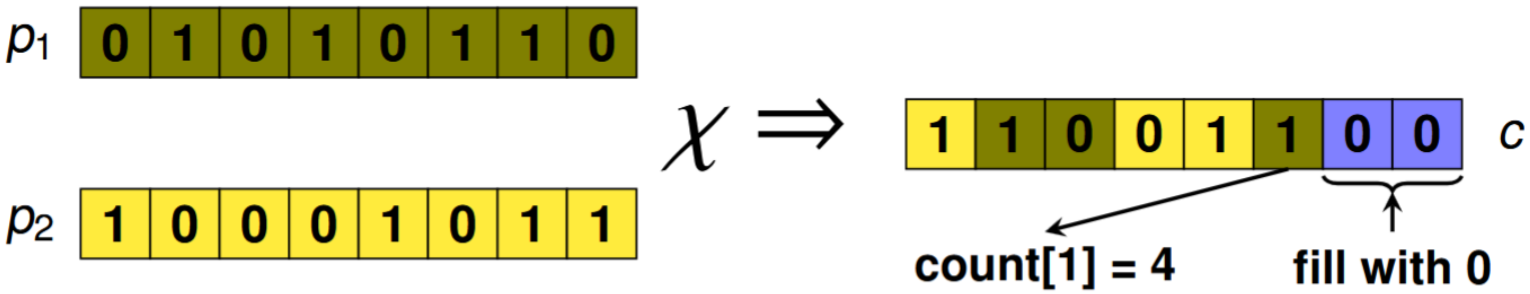


- Problem: how dow we ensure balancedness for cryptography?
- We could optimize it in the fitness function…

BALANCED CROSSOVER

- Idea: use counters to keep track of the numbers of 1s in the child [9, 13]

GA Example: Counter-based Crossover



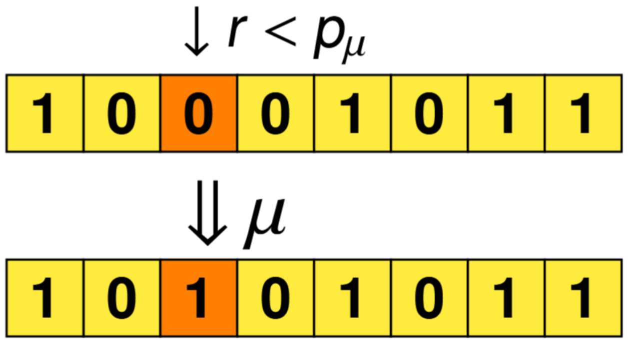
- If we start from balanced parents, we get balanced children

UNIVERSITY OF TWENTE.

MUTATION

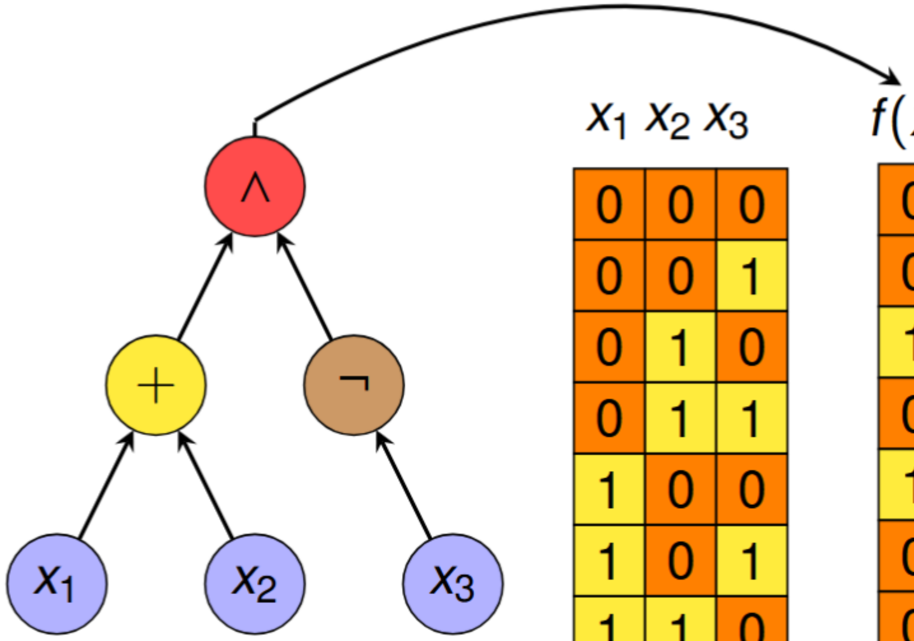
- Idea: introduce new "genetic material" in the offspring (Exploration)

GA Example: Bit-flip mutation



- For balancedness: randomly swap some bits instead of flipping them

GENETIC PROGRAMMING (GP)



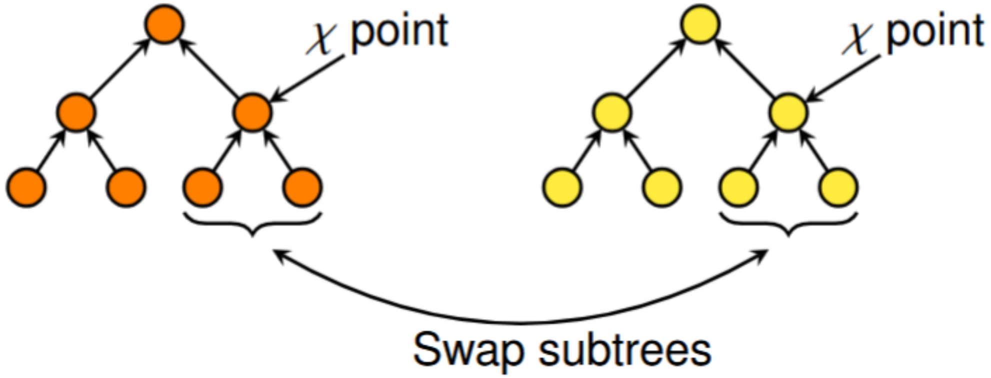
x_1	x_2	x_3	$f(x)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

- Idea: evolve computer programs to solve specific tasks
- GP Genotype: a syntactic tree
  - Leaf nodes: input variables
  - Internal nodes: operators (e.g. AND, OR, NOT, XOR, …)
- Phenotype: evaluate the tree for all possible assignments of the leaf nodes

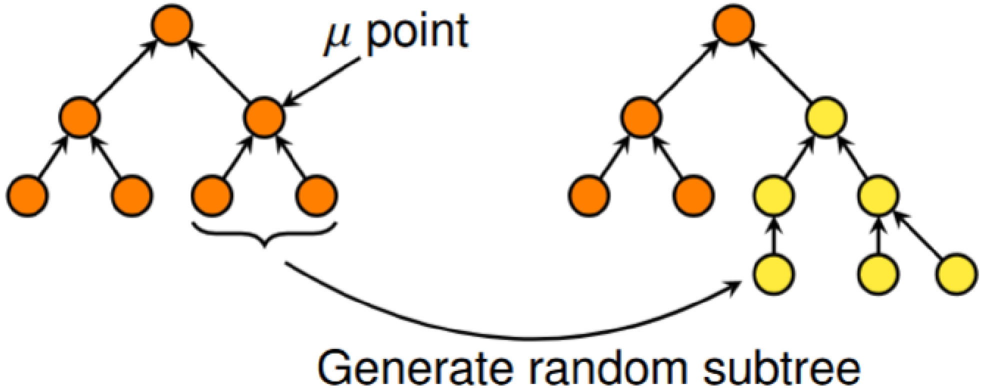
UNIVERSITY OF TWENTE.

GP CROSSOVER & MUTATION

Subtree Crossover



Subtree Mutation



- In general: not possible to preserve the balancedness of the Boolean function
- Nevertheless: GP usually fares better than (balanced) GA [10, 12, 14, 15]
- Other approaches: use GP to combine existing functions with high nonlinearity [3, 10]

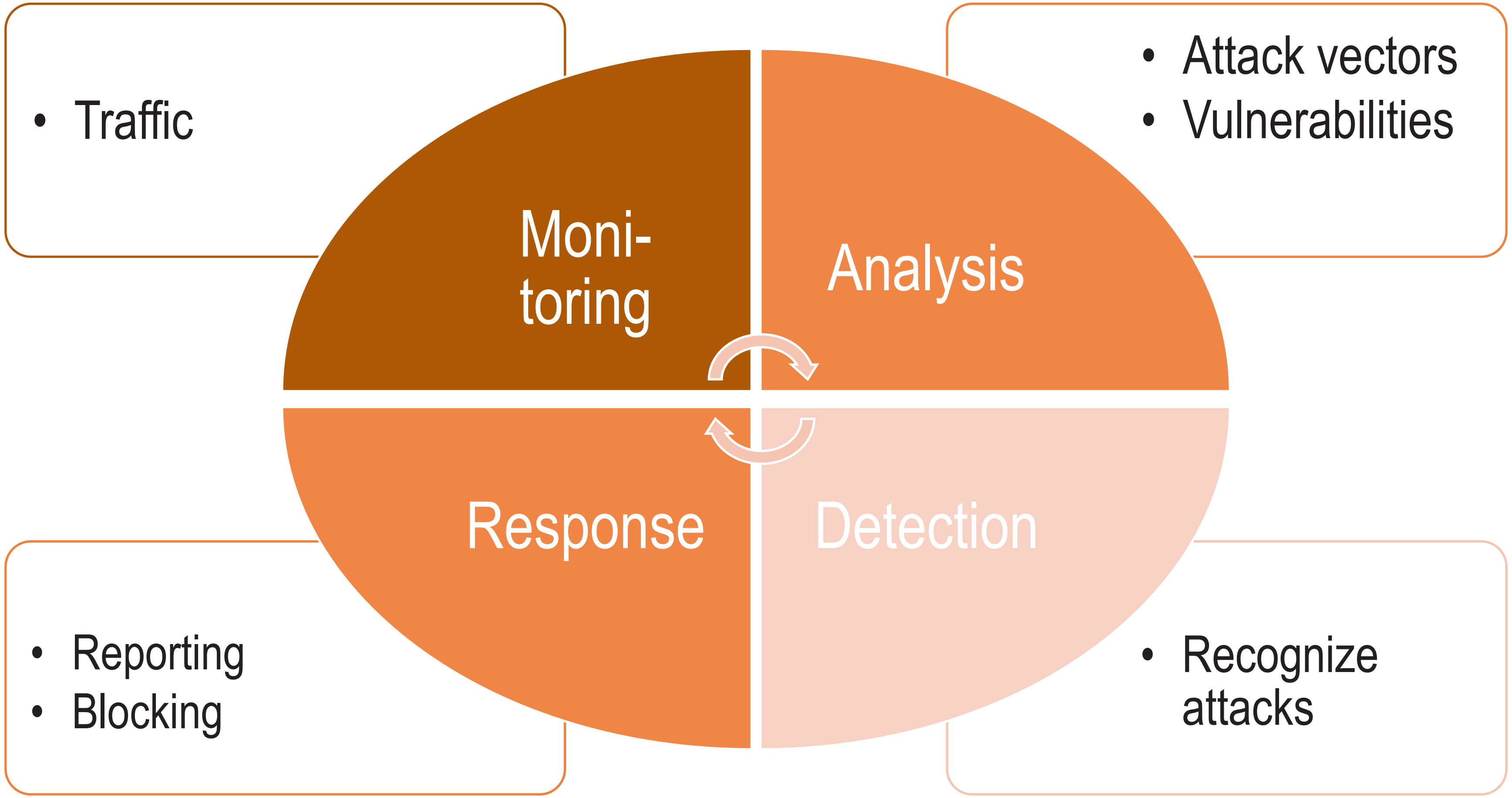
NETWORK INTRUSION DETECTION

UNIVERSITY
OF TWENTE.

INTRUSION DETECTION

- Intrusion Detection System: Monitoring for attacks
- Attack: attempt to compromise confidentiality, integrity, or availability
- Several types of IDS:
  - Monitor source: network, application, server, …
  - Analysis type: signatures, rules, machine learning, …

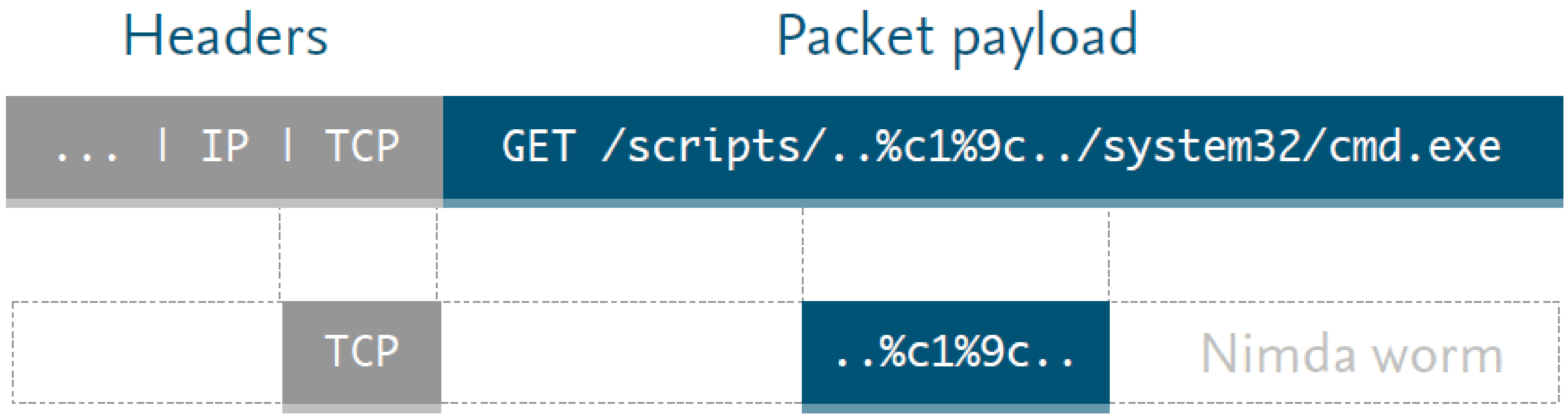
NETWORK INTRUSION DETECTION (NIDS)



UNIVERSITY
OF TWENTE.

SIGNATURE-BASED NIDS

- Traditional type of NIDS: signature-based [16]

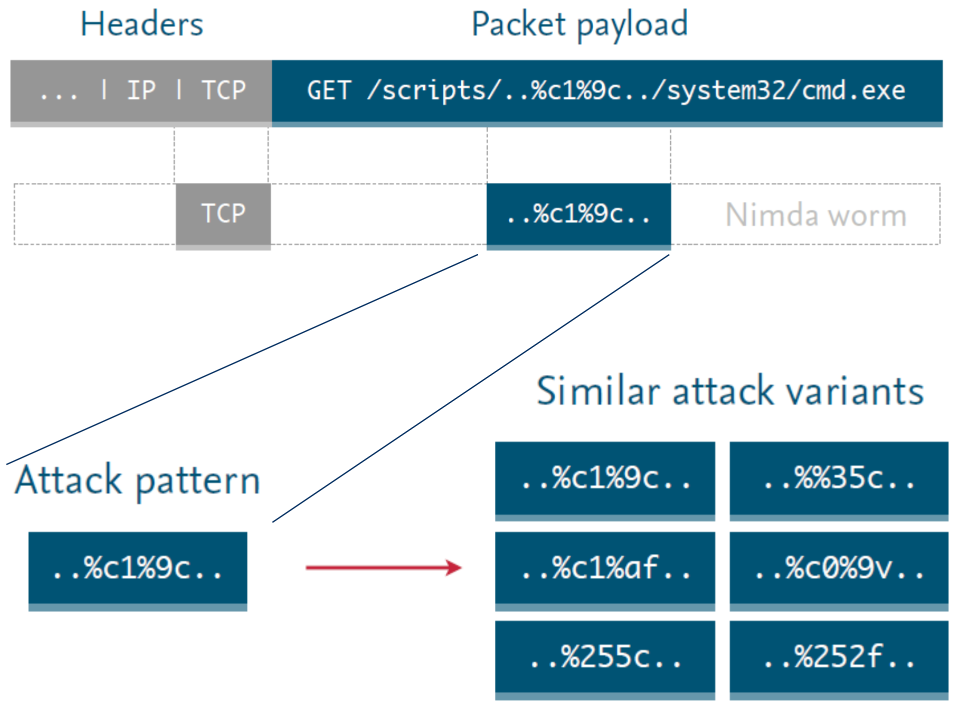


- Leverages pattern detection of known attacks (e.g., with Regular Expressions)

SIGNATURE-BASED NIDS

Common problems:

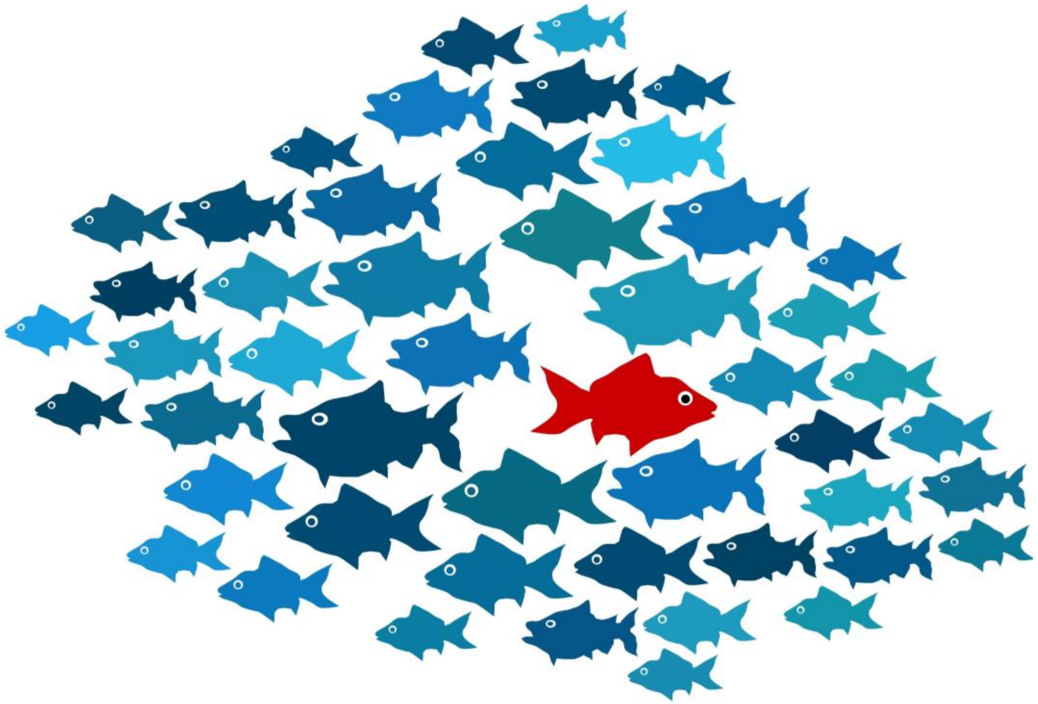
- Signatures from known attacks required
- Signature updates required
- Scalability issues, complexity and attack frequency
- Unable to detect unknown attacks



ANOMALY-BASED NIDS

- Idea: focus on benign traffic only [16]
- Learn a model of “normal” network traffic
- Assumptions:
  - Mainly benign training data
  - Unknown attacks differ from benign data
- Requires careful feature engineering

Easy example: spot the anomalous fish



ANOMALY-BASED NIDS

Another example: who is not Homer?

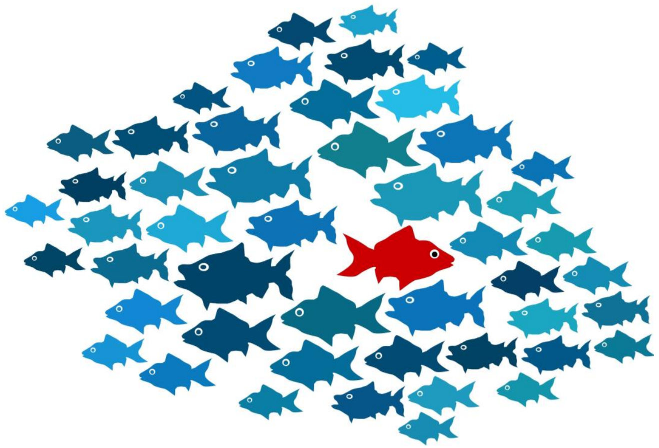
Features:

1. Size
2. Accessories
3. ...

UNIVERSITY OF TWENTE.

ANOMALY-BASED NIDS

- Risk: detection of irrelevant anomalies as attacks (false positives)
- Choice of features is crucial
- Attacks are deviations from normality
- Various techniques: SVM, KDE, evolutionary algorithms…

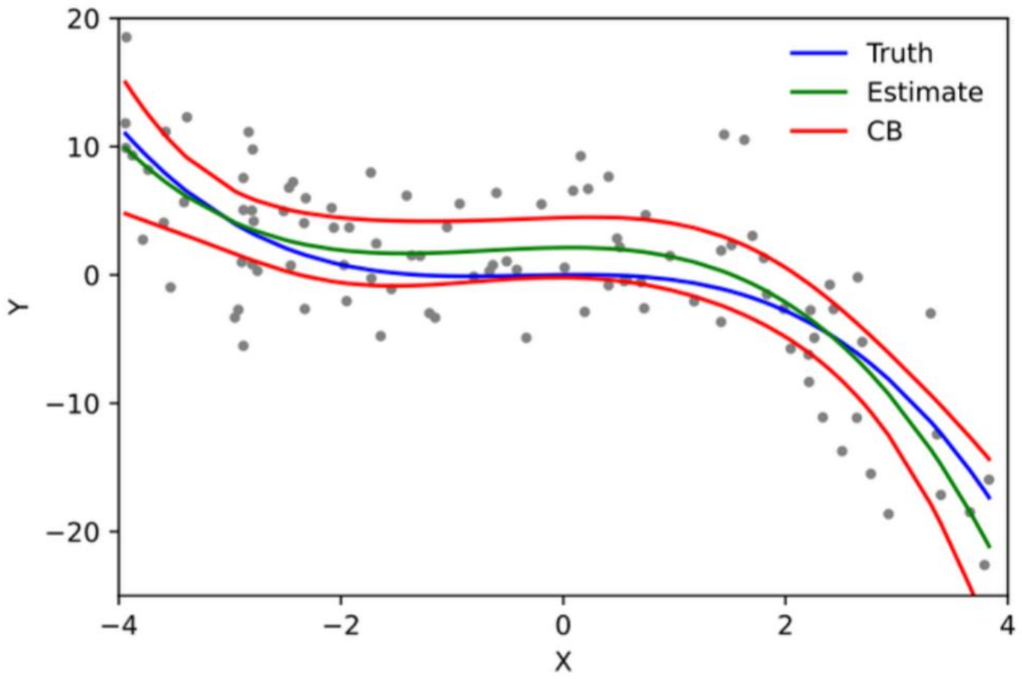


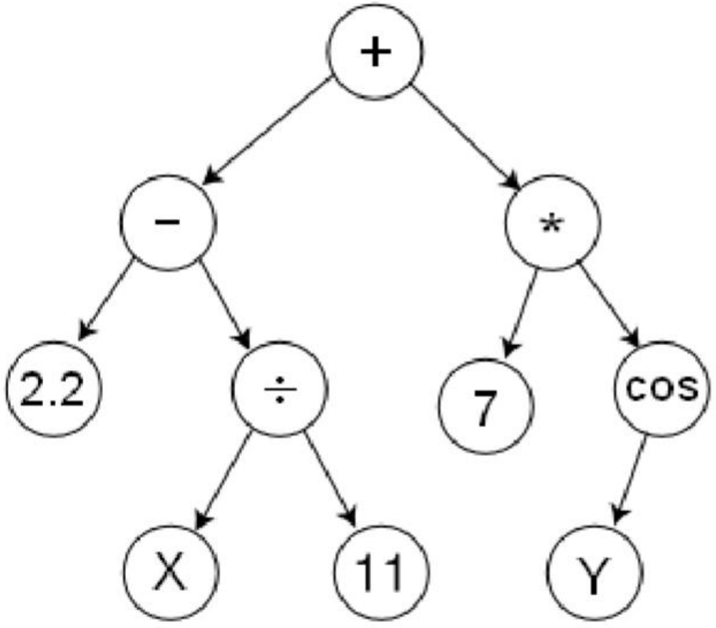
ANOMALY-BASED NIDS WITH GENETIC PROGRAMMING

UNIVERSITY OF TWENTE.

SYMBOLIC REGRESSION WITH GP

- Problem: use GP to find a symbolic expression that minimizes the errors in approximating a given set of data points





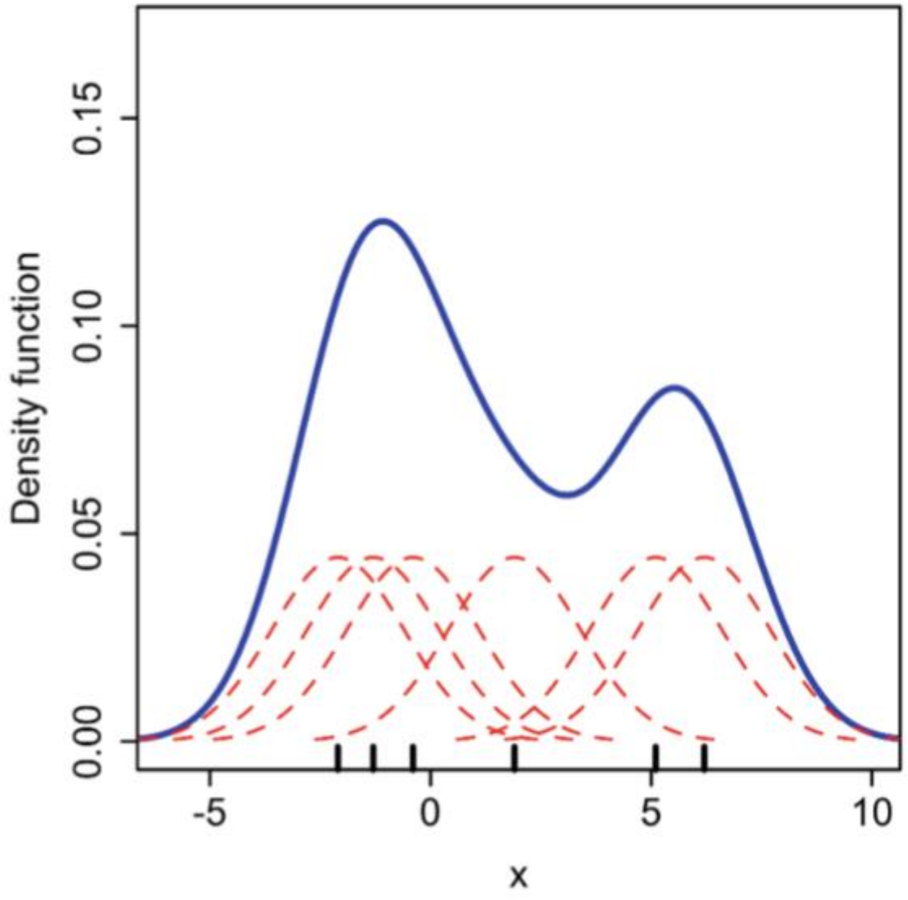
$$\left(2.2 - \left(\frac{X}{11}\right)\right) + \left(7 * \cos(Y)\right)$$

KERNEL DENSITY ESTIMATION FOR NIDS

- Problem: given a dataset of "normal" network packets, identify anomalies
- Possible approach: use kernel density estimation (KDE) [2]
- Given training points $x_1, \cdots, x_n \in \mathbb{R}^d$, estimate the density at x as:

$$\hat{p}(x) = \frac{1}{n} \sum_{i=1}^{n} K_h(x - x_i)$$

where K_h is a kernel function (e.g., a Gaussian distribution



UNIVERSITY OF TWENTE.

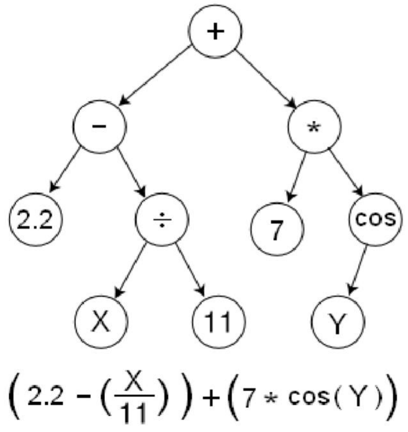
GENETIC PROGRAMMING FOR NIDS

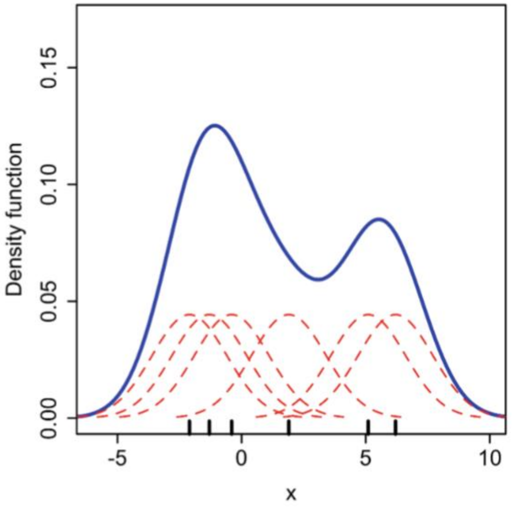
- KDE is expensive to compute at query time
- Need to compute $\hat{p}(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i)$ for every new network packet
- Idea: use GP to learn a surrogate f that approximates well the density [2]:

$$f(x_i) \approx d(x_i)$$

- Symbolic regression problem:

$$fit(f) = \sqrt{\frac{\sum_{i=1}^{n}(f(x_i) - d(x_i))^2}{n}}$$



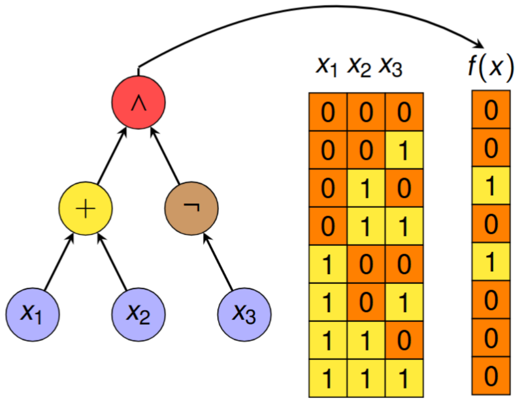


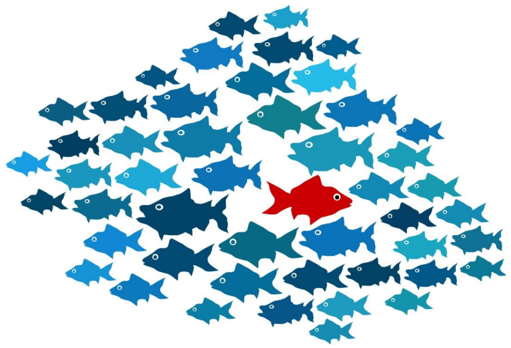
SUMMARY

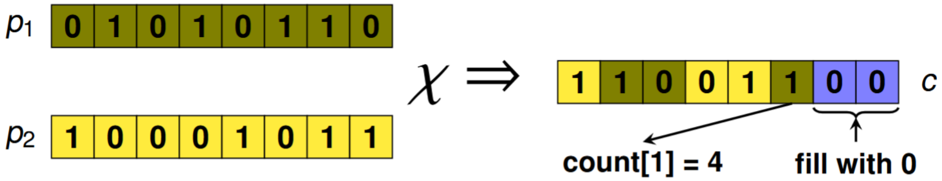
UNIVERSITY
OF TWENTE.

CONCLUDING THOUGHTS

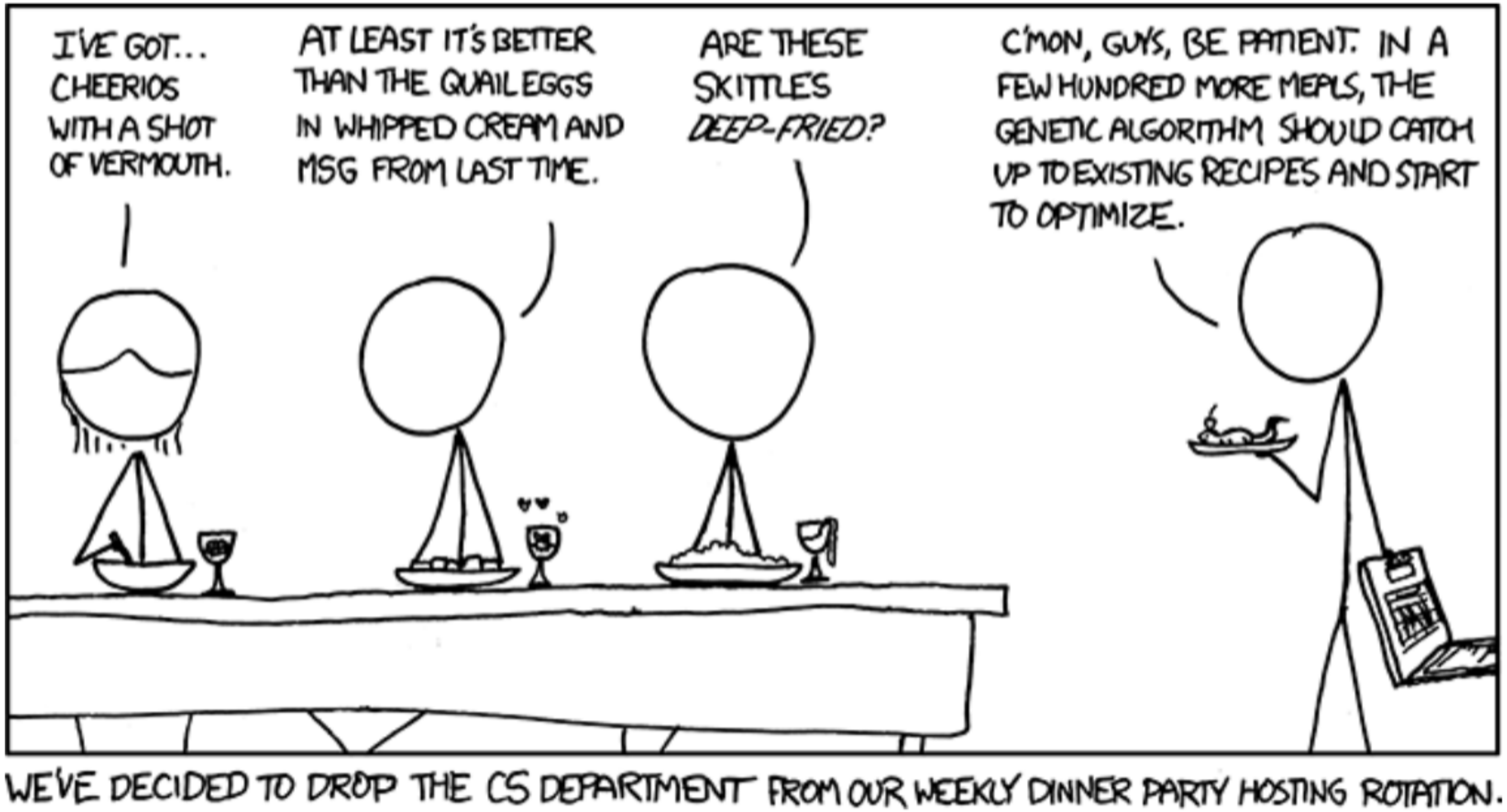
- EA are quite versatile to solve different problems in cybersecurity
- Examples seen here:
  - Cryptography
  - Network Intrusion Detection
- Many other applications!
  - Side-channel analysis [17]
  - Adversarial learning [18]
  - …







THANKS!
QUESTIONS?



UNIVERSITY
OF TWENTE.

REFERENCES

1. R. Anderson. Security Engineering. Third Edition, Wiley, (2020)
2. V. L. Cao, M. Nicolau, J. McDermott: One-Class Classification for Anomaly Detection with Kernel Density Estimation and Genetic Programming. Proceedings of EuroGP 2016, pp. 3–18 (2016)
3. C. Carlet, M. Djurasevic, D. Jakobovic, L. Mariot, S. Picek: Evolving constructions for balanced, highly nonlinear boolean functions. Proc. of GECCO 2022, pp. 1147–1155 (2022)
4. C. Carlet: Boolean functions for cryptography and coding theory. Cambridge University Press (2021)
5. M. Djurasevic, D. Jakobovic, L. Mariot, S..Picek: A survey of metaheuristic algorithms for the design of cryptographic Boolean functions. Cryptogr. Commun. 15(6): 1171-1197 (2023)
6. J. Katz, Y. Lindell: Introduction to Modern Cryptography. Third Edition, CRC Press (2021)
7. A. Kerckhoff, "La cryptographie militaire," Journal des sciences militaires, vol. IX, no. 5–38, Janvier 1883
8. J. R. Koza, M. A. Keane, J. P. Rice: Performance improvement of machine learning via automatic discovery of facilitating functions as applied to a problem of symbolic system identification. Proc. of ICNN 1993, pp. 191–198 (1993)
9. L. Manzoni, L. Mariot, E. Tuba: Balanced crossover operators in Genetic Algorithms. Swarm Evol. Comput. 54: 100646 (2020)
10. L. Mariot, M. Saletta, A. Leporati, L. Manzoni: Heuristic search of (semi-)bent functions based on cellular automata. Nat. Comput. 21(3): 377-391 (2022)
11. L. Mariot, D. Jakobovic, A. Leporati, S. Picek: Hyper-bent Boolean Functions and Evolutionary Algorithms. Proc. of EuroGP 2019, pp. 262-277 (2019)
12. L. Mariot, S. Picek, A. Leporati, D. Jakobovic: Cellular automata based S-boxes. Cryptogr. Commun. 11(1): 41-62 (2019)
13. W. Millan, J. Clark, E. Dawson: Heuristic Design of Cryptographically Strong Balanced Boolean Functions. Proc. of EUROCRYPT 1998, pp. 489–499 (1998)
14. S. Picek, K. Knezevic, L. Mariot, D. Jakobovic, A. Leporati: Evolving Bent Quaternary Functions. Proc. of CEC 2018, pp. 1–8 (2018)
15. S. Picek, D. Jakobovic, J. F. Miller, L. Batina, M. Cupic: Cryptographic Boolean functions: One output, many design criteria. Appl. Soft Comput. 40: 635-653 (2016)
16. K. Rieck: Machine Learning for Application-Layer Intrusion Detection. PhD Thesis, Technischen Universität Berlin (2009)
17. F. Schijlen, L. Wu, L. Mariot: NASCTY: Neuroevolution to Attack Side-channel Leakages Yielding Convolutional Neural Networks. Mathematics 11(12): 2616 (2024)
18. J. Su, D. Vasconcellos Vargas, S. Kouichi: One-pixel attack for fooling deep neural networks. IEEE Trans. Evol. Comp. 23(5): 828-841 (2019)