Bernoulli at the Root of Horizontal Side Channel Attacks

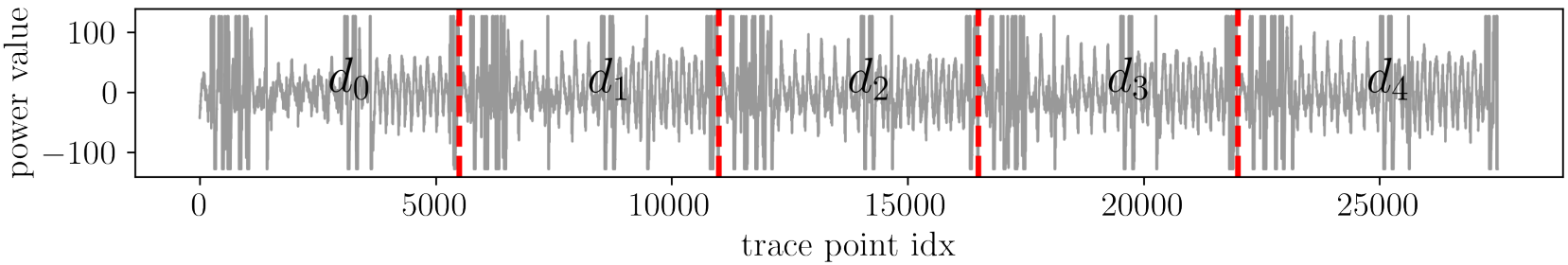
G. Cler
S. Ordas
P. Maurine

November 15, 2023

Horizontal Attacks

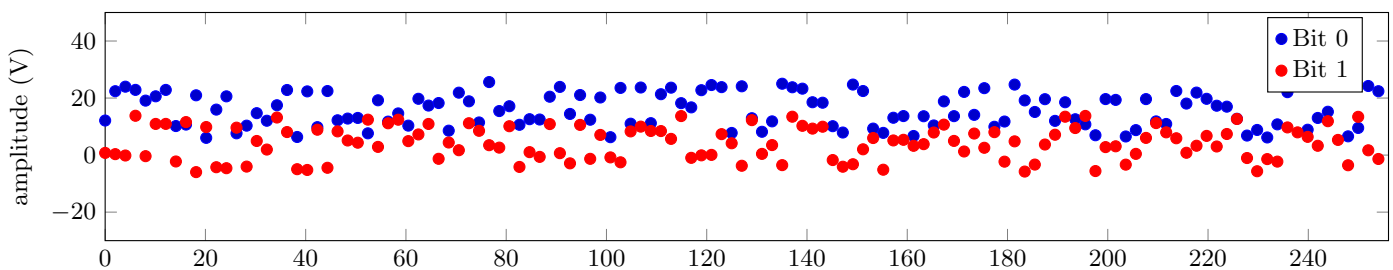
Horizontal attacks on regular implementations consists in:

1. Patterns cutting



2. PoI selection
   - Patterns compression, dimensionality reduction (PCA)
   - Univariate features selection (Perin 2014, ...)

3. Patterns discrimination

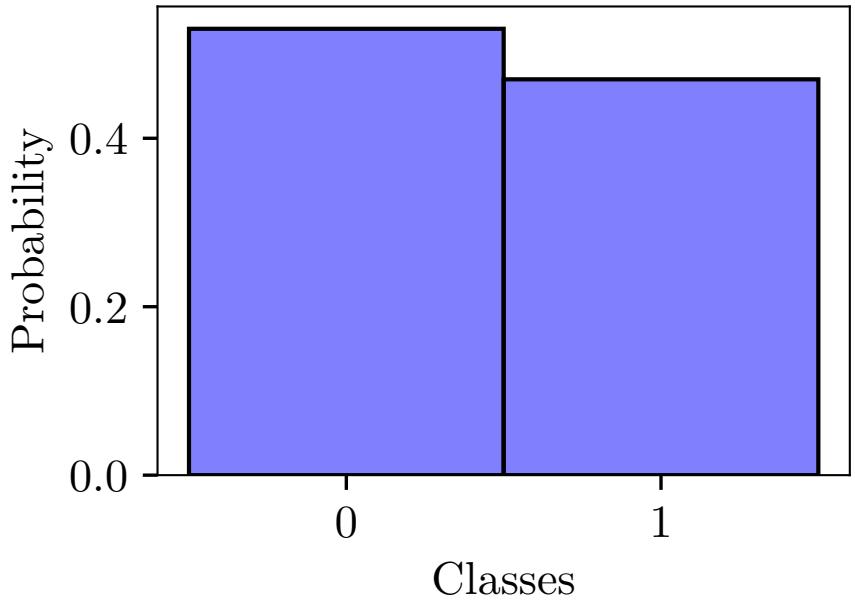


Objectives

This work aims to:

- Redefine a leakage model for univariate PoI analysis
- Select PoI based on a statistical test and gaussian mixture modeling
- Characterize the exploitability of identified PoI and rank them for the attack
- Be applicable on noisy targets

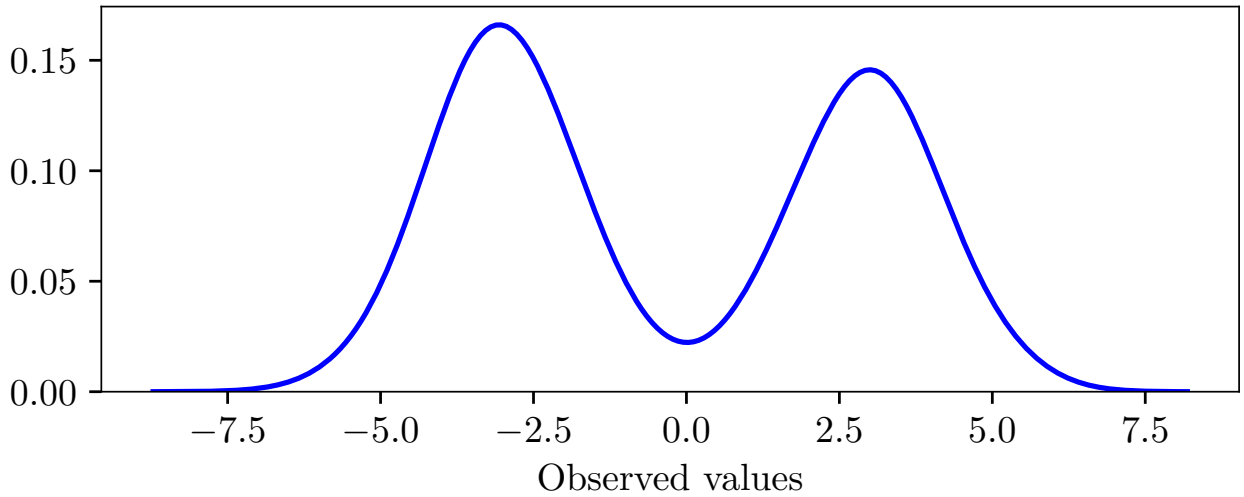
Secure Implementations and Bernoulli Process



Want to find PoI such that $\mathcal{Y} \sim B(\pi_0 = 0.5, n)$.

$$\begin{cases} H_0 : \mathcal{Y} \sim B(\pi_0 = 0.5, n) & \rightarrow \text{ possibly a PoI} \\ H_1 : \mathcal{Y} \nsim B(\pi_0 = 0.5, n) & \rightarrow \text{not a PoI} \end{cases}$$

Bernoulli Process to Gaussian Mixture



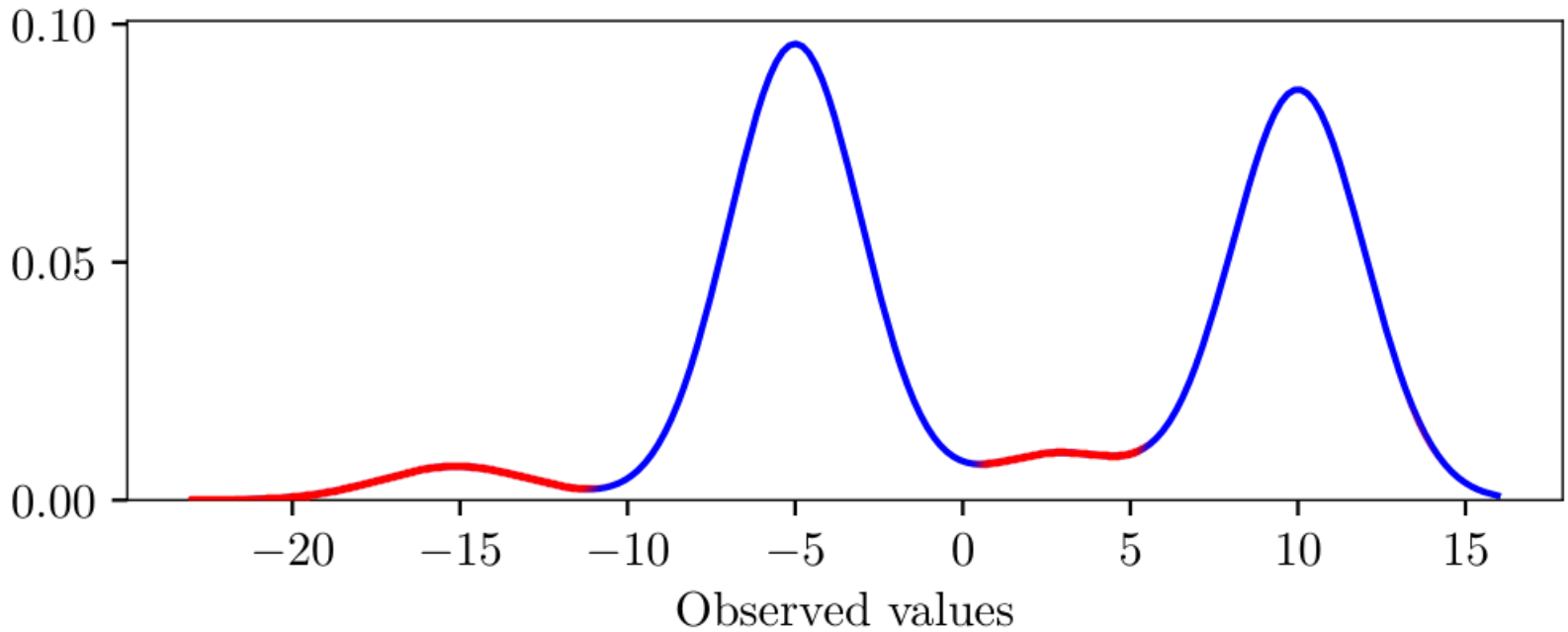
$$X \sim f(x, \boldsymbol{\theta}) = \sum_{i \in K} \pi_i \mathcal{N}(x | \mu_i, \sigma_i^2)$$

Identify underlying components such that:

$$\begin{cases} H_0 : \mathcal{Y} \sim B(\pi_0 = 0.5, n) & \rightarrow \text{possibly a PoI} \\ H_1 : \mathcal{Y} \nsim B(\pi_0 = 0.5, n) & \rightarrow \text{not a PoI} \end{cases}$$

Real Traces

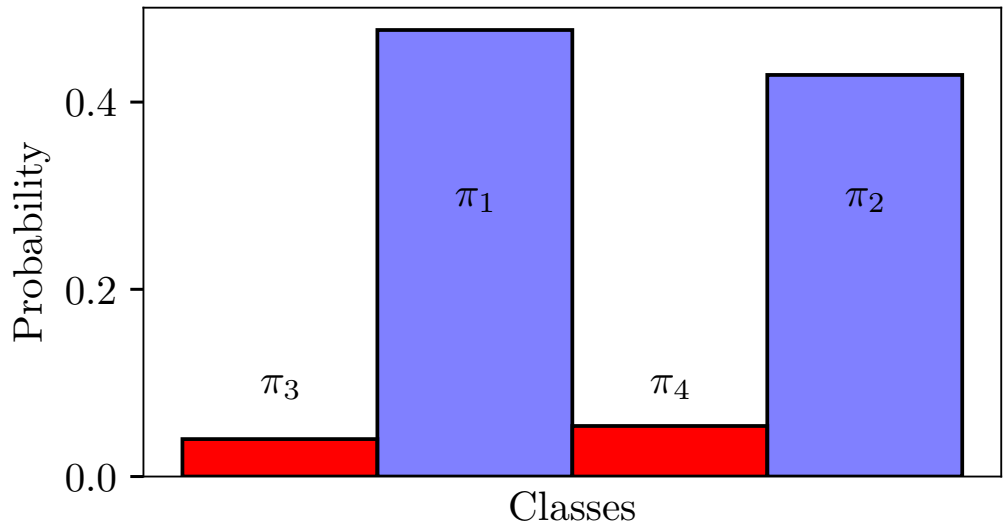
Traces acquired on modern products often show additional behaviors due to outliers or countermeasures.



$$X \sim f(x, \boldsymbol{\theta}) = \sum_{i \in K} \pi_i \mathcal{N}(x|\mu_i, \sigma_i^2) + \sum_{j \in L} \pi_j \mathcal{N}(x|\mu_j, \sigma_j^2)$$

Real Traces

Multinomial distribution $\mathcal{Y} \sim M(\pi, n)$ with two major components π_1 and π_2 should be considered instead of a Binomial one.

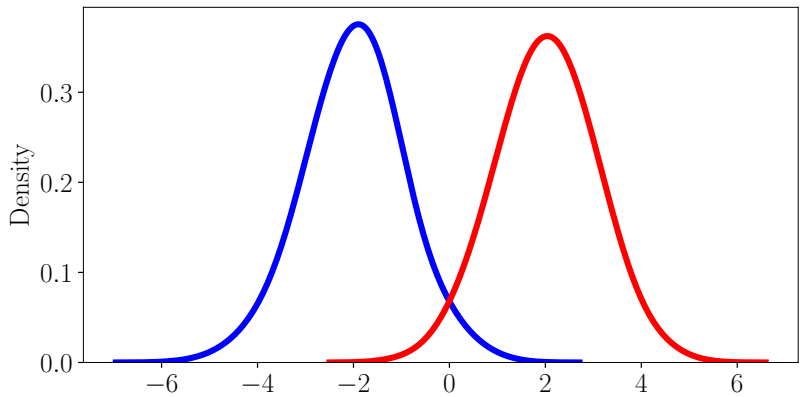


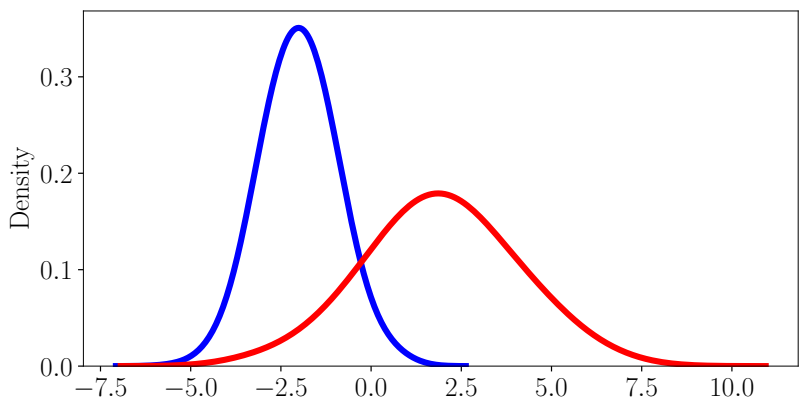
with $\pi_1 = \pi_2 = \frac{1-\sum_{i \in L} \pi_i}{2}$ and $\pi_1 + \pi_2 \geq 0.9$

$$\begin{cases} H_0 : \mathcal{Y} \sim M([\pi_1, \pi_2, \ldots, \pi_s], n) & \rightarrow \text{possibly a PoI} \\ H_1 : \mathcal{Y} \nsim M([\pi_1, \pi_2, \ldots, \pi_s], n) & \rightarrow \text{not a PoI} \end{cases}$$

Impact of Gaussian noise

Gaussian noise is expected to affect similarly all components of the mixture. Assess the variance disparity of components with F-test.

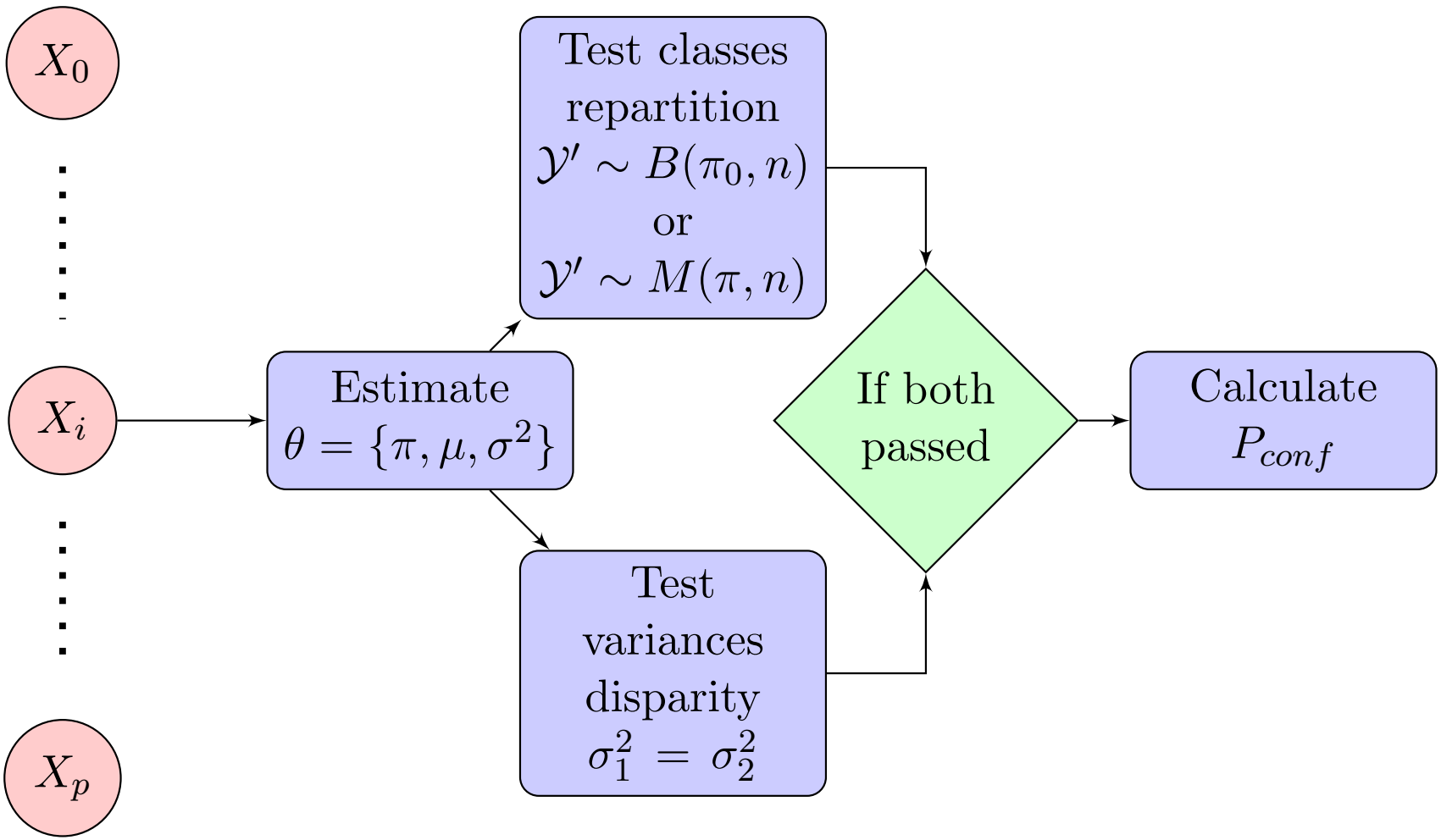




$$\begin{cases} H_0 : \sigma_1^2 = \sigma_2^2 & \rightarrow \text{possibly a PoI} \\ H_1 : \sigma_1^2 \neq \sigma_2^2 & \rightarrow \text{not a PoI} \end{cases}$$

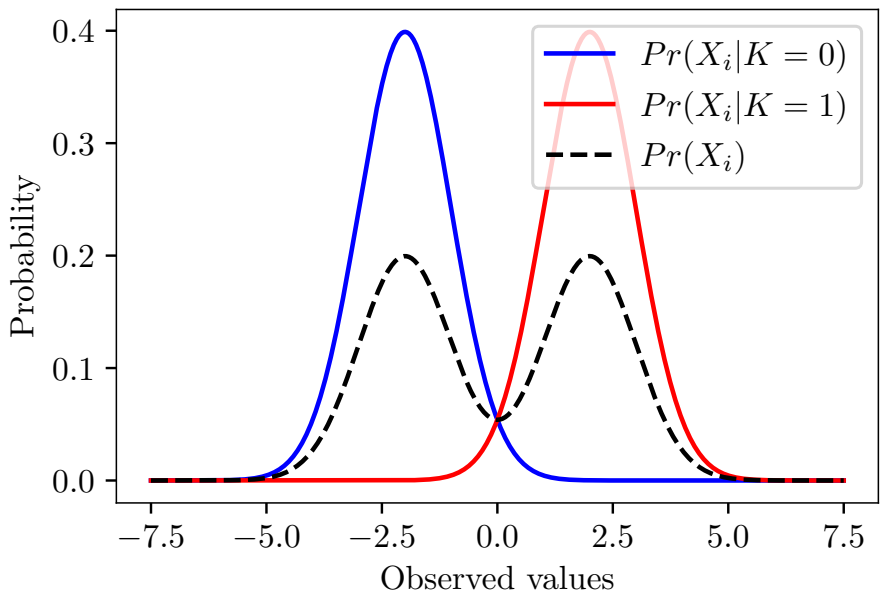
PoI selection procedure

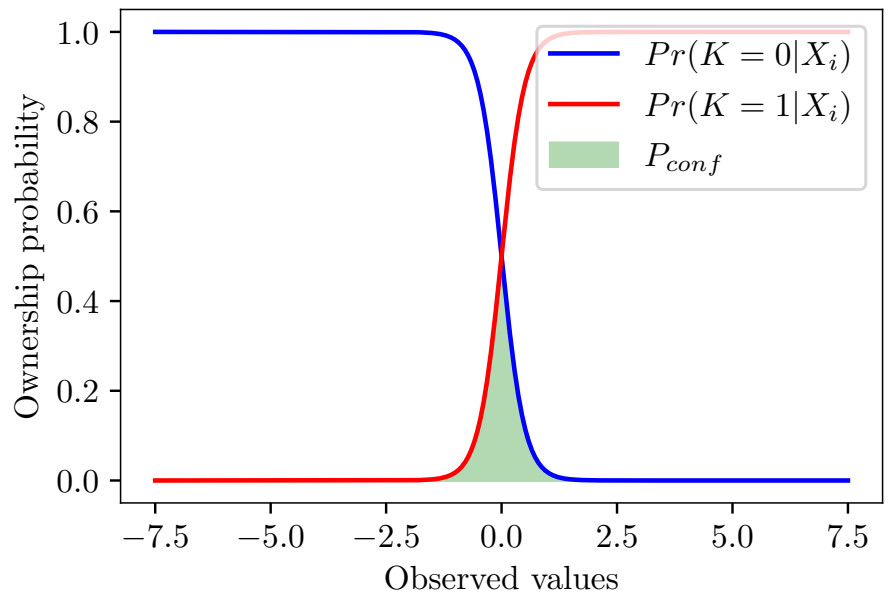
For each temporal point X_i accross all patterns:



Probability of confusion

Evaluate the exploitability of estimated dominant mixture components

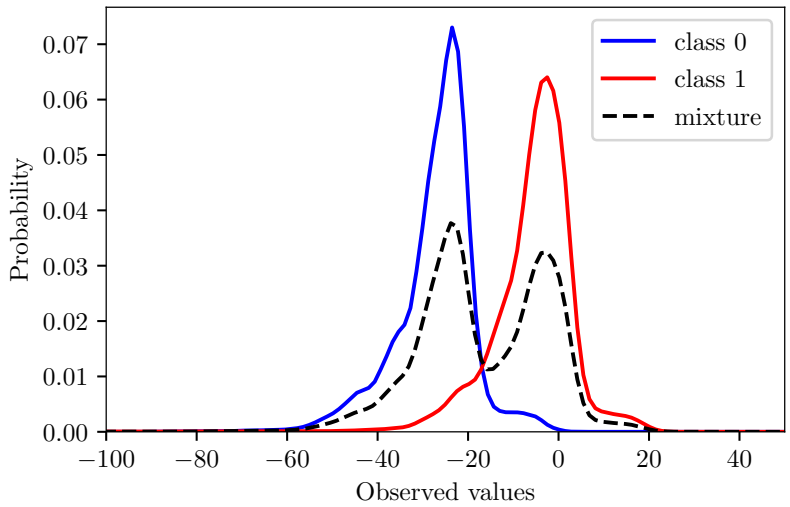


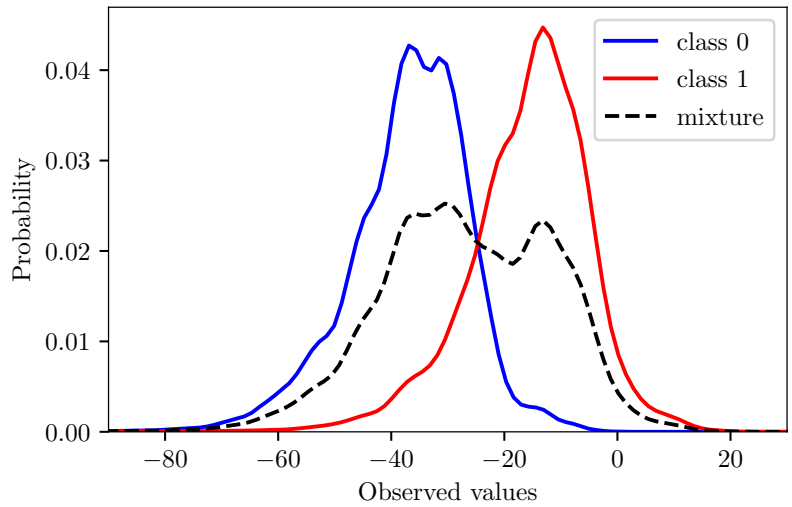


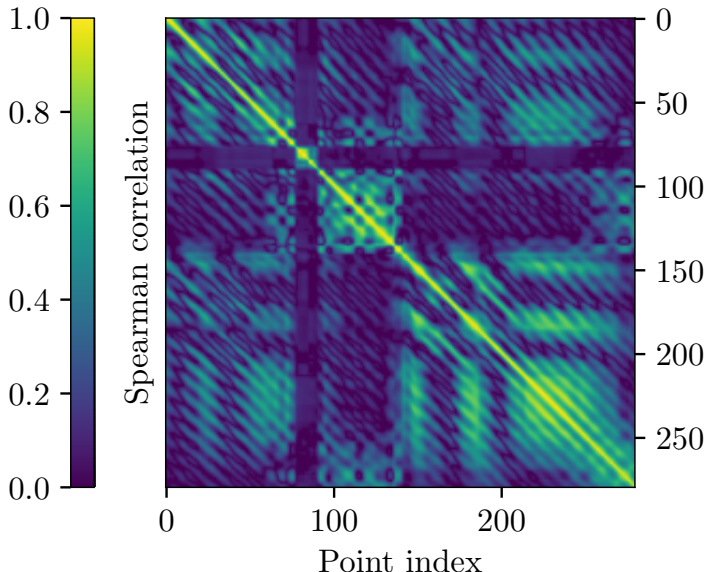
$$P_{conf} = \sum_{j \in L} \pi_j + \sum_{i \in K} \pi_i \int_{-\infty}^{\infty} \min_{i \in K} \{\Pr(i|X=x)\} \; dx$$

Widening the set of PoI

Neighboring points of identified PoI can carry same leakage information but not pass statistical testing. They can be seen as transformations of PoI.





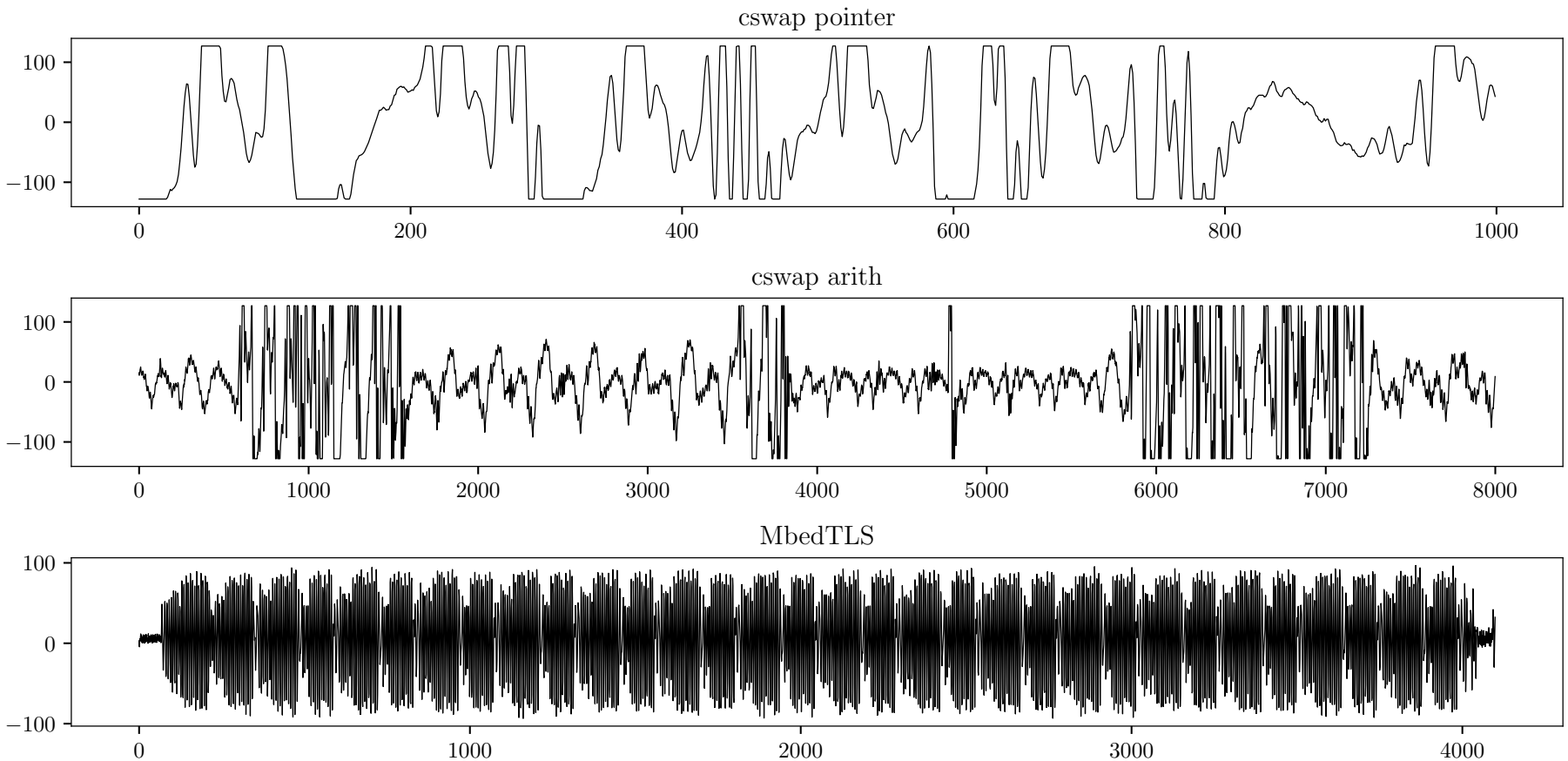


Apply Spearman correlation $\rho(X_i, X_j)$ to consider excluded neighbors with highly correlation as PoI.

Experimental Validation

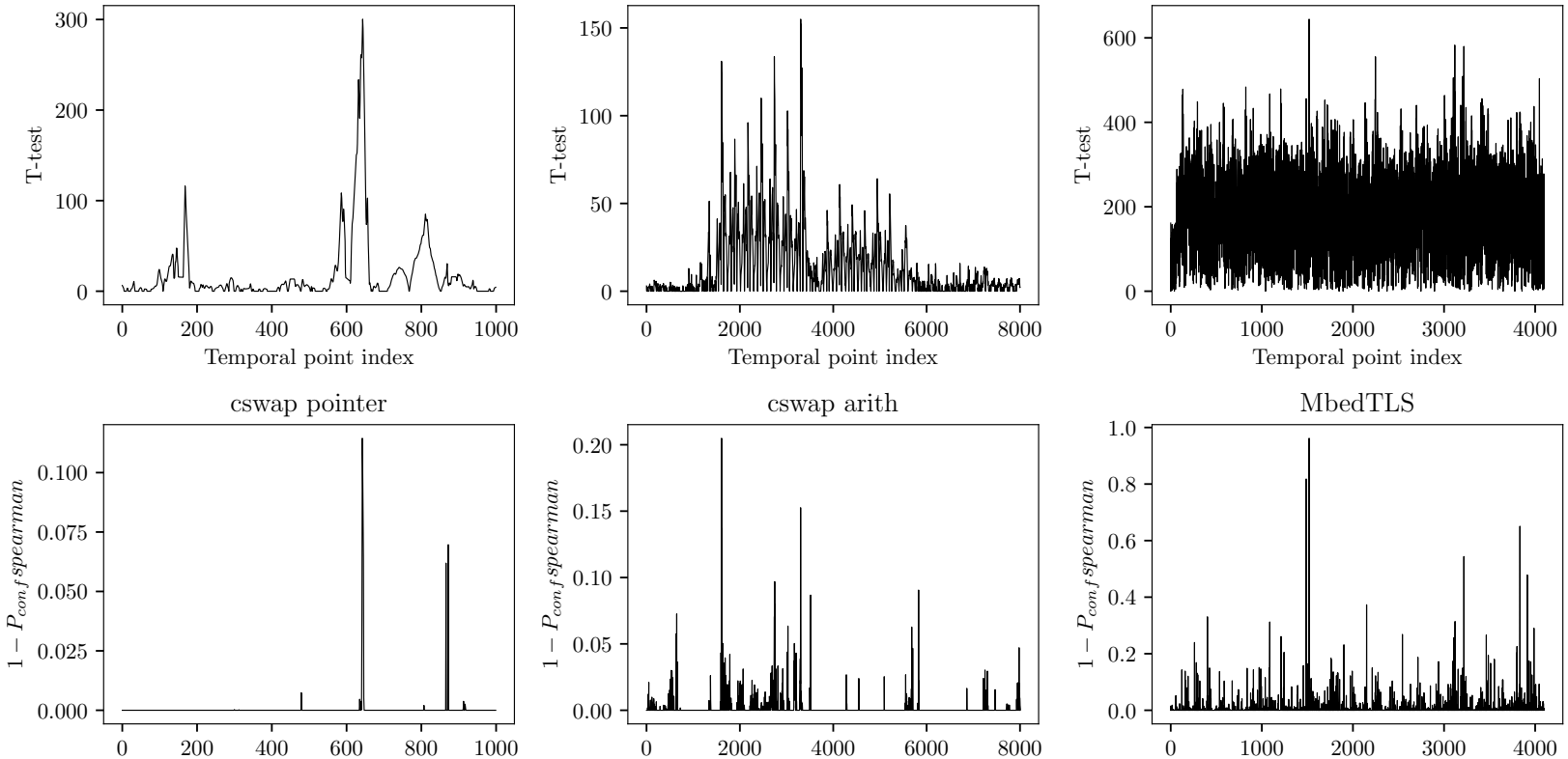
Methodology applied to several targets:

- Public datasets cswap arith and pointer From Perin 2021, ECSM Montgomery ladder on STM32F4, 255 patterns per trace
- MbedTLS, RSA S&M on LPC55S69, 2048 patterns per trace



Single pattern

PoI selection results



Number of PoI selected

	Pointer	Arith	MbedTLS
PoI selection	22	1019	861
PoI selection + Spearman	52	1583	1482

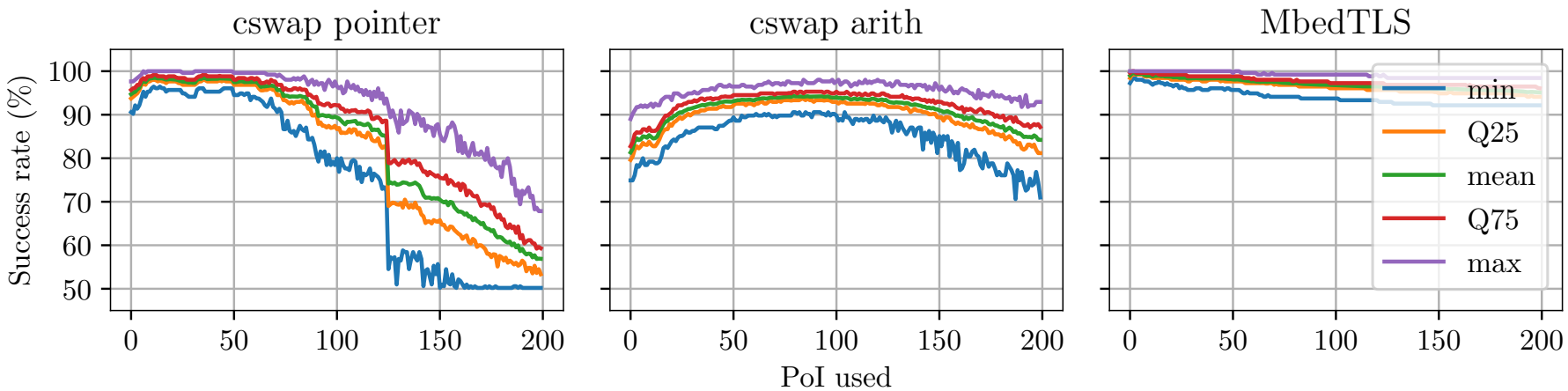
Attack procedure

Compare proposed approach with supervised PoI selection. For unbiased results, repeat and average 100 times:

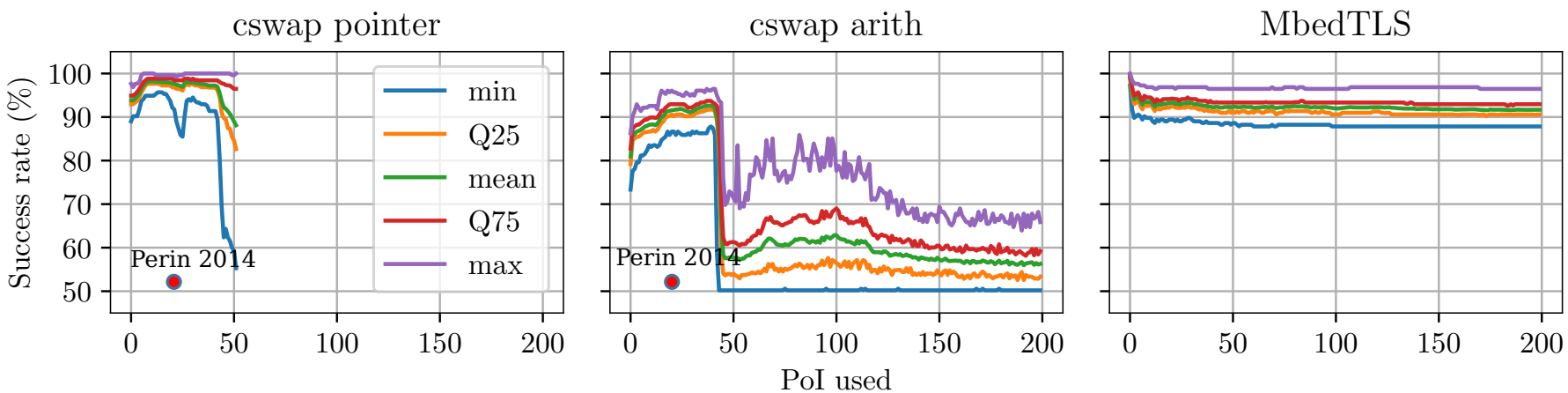
1. Select the $m \in \{1, \ldots, 200\}$ most leaking PoI
   - lowest P_{conf} values
   - highest supervised T-values
2. Apply Fuzzy clustering using m PoI on a single trace

Results

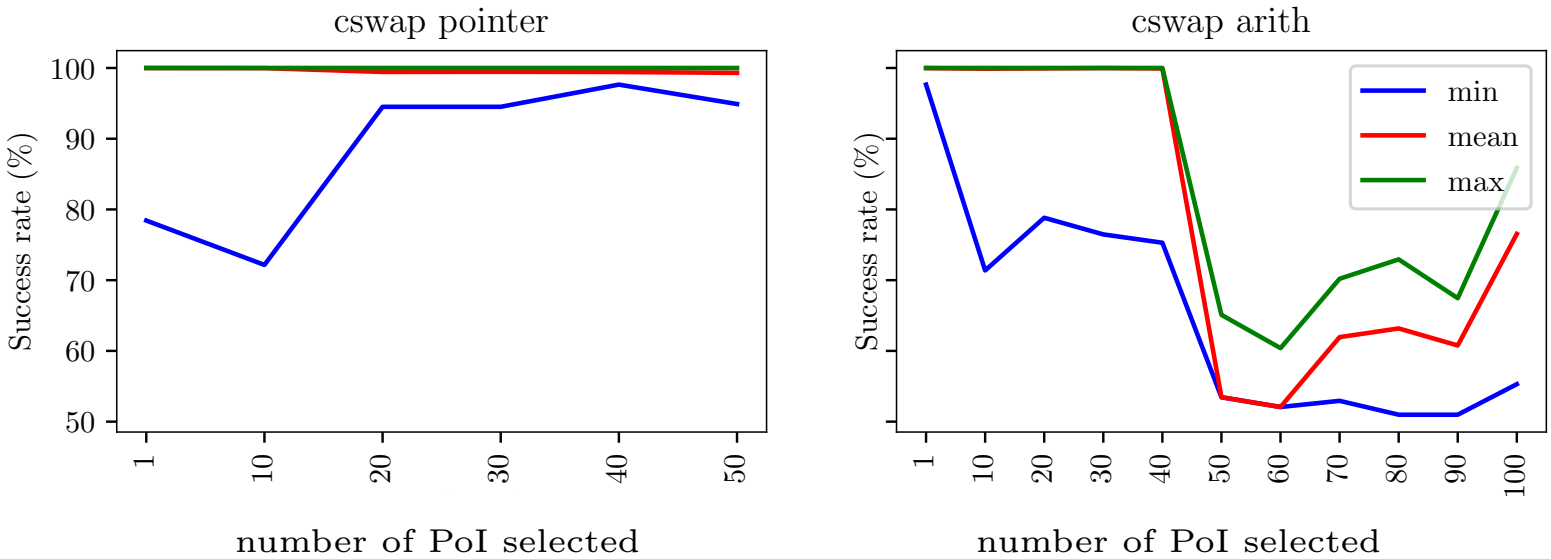
Supervised PoI selection (knowing labels)



Unsupervised PoI selection (Our, after application of spearman)



Application of Perin 2021 framework on selected PoI



Conclusion

Proposed approach allows to:

- Detect noisy Bernoulli distributed leakage for horizontal attacks
- Enlarge set of PoI by considering highly correlated neighbors
- Characterize PoI exploitability and propose a ranking metric for the attack
- Reduce gap between supervised and unsupervised attacks with a deficit of information