Consistent, Efficient and Leakage-Model Free Mutual Information Estimation

Arnab Roy, Aakash Chowdhury, and Elisabeth Oswald

University of Klagenfurt, Austria
{arnab.roy, aakash.chowdhury, elisabeth.oswald}@aau.at

Abstract. The mutual information between the observable device leakage and the unknown key is a key metric in the context of side channel attacks, evaluations, and countermeasures. Estimating this mutual information has been a problem and was addressed in several recent contributions. We explain why previous work has ended up in a "catch-22" and we show how to avoid this situation by using a leakage model free estimation approach based on a recently discovered, consistent mutual information estimator. Our work demonstrates that mutual information estimation in the side channel setting can be done extremely efficiently (even in a multivariate setting), with strong mathematical guarantees, without the need for an explicit device leakage model, discretisation, or assumptions about the nature of the device leakage.

1 Introduction

Side channel theory and practice require metrics to describe how "secure" or "vulnerable" a device, or an implementation is with respect to side channel attacks. Thus suitable metrics are required in the context of measuring the quality of attacks, the impact of countermeasures, and also the quality of an evaluation.

For instance, the quality of a side channel evaluation depends on how close the evaluation process, representing the "worst case adversary"comes to the "ideal adversary" [2]: the worst case adversary utilises an estimated leakage model as part of their attack strategy; the ideal adversary knows the actual leakage model of the device.

Previous works relating to evaluations [5] and attacks [10], but also contributions describing a general framework for considering side channel adversaries [15], of security proofs [7]) all use the mutual information as their metric of choice. Clearly, the concept of the mutual information (MI) between (a function of) the pair of (plaintext input, secret key) and the observable leakage $I(L;(X,K))$ is at the heart of side channel theory and practice.

1.1 Significance of MI estimation

The MI between (a function of) the key (and input) and the observed traces depends on the joint distribution between these variables, which is unknown in practice. Thus *the MI can only be estimated* from the available data (i.e. leakage traces that are obtained from some device holding a secret key). In the side channel literature, the estimation of the MI has so far always been linked to entropy estimation via the usual "2H" or "3H" approaches (we review this in the technical sections of this article), which boils down to estimating the joint (or conditional) distribution between the variables in question.

Constructing consistent estimators for joint (or conditional) densities of *arbitrary* distributions is a well studied problem in the context of entropy and MI estimators. This estimation problem is notoriously hard (when considering arbitrary distributions), and [12] shows how badly behaved well known variations of "plug-in" estimators can be. Clearly this is an unsatisfactory situation for the use of MI as a metric within the side channel community, because the MI is at the heart of theory that is highly connected to side channel practice. Further more, in the context of leakage certification ([4], a process by which we want to judge the quality of a profiling adversary) this leads to a catch-22: estimating the MI itself is a fraught process, and yet we hope to judge the profiling model quality by an estimated MI.

1.2 Achievements of Previous Work

Previous research by [5] and the follow-on works [3, 4] make some progress here and provide mathematical reasoning for various MI related estimates, resulting in bounds for their estimators.

Specifically, [5] propose that to evaluate MI estimates, it is important to distinguish between assumption errors (they relate to the leakage function captured by the pdf estimation approach) and estimation errors (they relate to the amount of data available for pdf estimation). They put forward the observation that a divergence between the MI and a related metric called perceived information (PI) reflects assumption errors. Later on [4] experiment with using statistical moments to demonstrate assumption errors, and introduce the notion of hypothetical information (HI). Most recently, [3] show that the expected empirical MI, called eHI, is not only an upper bound for the MI, it also converges towards the true MI. They also show that the expected PI (ePI) is a lower bound for the MI. These statements hold assuming that they key is uniformly distributed and that the noise-free leakage of the device is discrete and deterministic. Such a result is great, and it fits with the known fact that for specific combinations of variables (either both are discrete, or both are continuous) provably consistent estimators exist [1]. For the general case, where we work with mixture distributions of discrete and continuous variables, the situation is more complex.

1.3 The Remaining Gap and Our Contributions

The side channel setting (without simplifying assumptions such as made in previous work) is exactly such a general case for mutual information estimation: the device leakage is a function of some (probabilistic) physical process interacting with discrete variables (input and key) and independent noise (i.e. the physical processes that do not interact wit the input and key). Different parts/components of a device exhibit different leakage functions, and measurement setups both discretize measurements and smooth them via post-processing. Thus depending on various factors we may encounter leakage functions that are either discrete and deterministic in key and inputs (e.g. an data transfer instruction on a microprocessor leaks the Hamming weight) or that are continuous and probabilistic in key and inputs (e.g. an arithmetic instruction invokes a complex multiplier or other arithmetic circuit). This mix of leakage function can even happen when considering an implementation on a single device.

We offer several contributions:

- We clearly explain the relationships between different mutual information quantities of interest, and highlight the critical role that discretisation plays in the context of mutual information estimation. In particular we explain that besides assumption errors and estimation errors, the use of a biased estimator is a cause for concern in practical mutual information estimation.
- We prove that the mutual information between the observed leakage and the key (as well as input) can be computed without the need for density estimation or discretisation by using a novel estimator that was recently introduced [6].
- We provide practical evidence in the channel context that the application of the estimator [6] is data efficient, computationally efficient, **and that it can also be used elegantly in a multivariate scenario.**

Our results thus provide an approach for mutual information estimation which:

- uses a consistent estimator for all leakage functions that can be observed in practice (discrete, continuous, deterministic, probabilistic),
- naturally extends to the multivariate setting, and
- has the same asymptotic convergence rate as existing estimators.

We explain our notation and provide a review of basic mathematics of mutual information and mutual information estimation in Sect.2. We describe why it is important to cater for a variety of leakage functions, including ones that are probabilistic in the key and inputs in Sect.3; we also explain the relationships between mutual information quantities of interest and why discretisation in the process of mutual information estimation must be avoided. In Sect. 6.1 we provide the mathematical reasoning for the fact that under mild assumptions about the noise distribution, it is possible to compute the mutual information between the observed leakage and the key without the need of an estimated leakage model. Before providing experimental evidence for the efficiency and soundness of our proposed mutual information estimation process in Sect. 6, we discuss implementation aspects of the estimator in Sect. 5.

2 Notation and Background

Following convention, we represent random variables with upper case letters, and their realisations with the corresponding lower case letters. We abuse notation and treat random variables and their corresponding sets synonimously. For two functions g and h, $g \circ h$ denotes the composition of the functions.

We denote the probability density function (pdf) and cumulative distribution function (CDF) of a continuous random variable with f and F respectively. For a discrete random variable, p will denote its probability mass function (pmf); for an arbitrary event we use $\mathbb{P}$ to denote its probability. Whenever necessary, in a pdf, CDF or pmf we will make the corresponding random variable explicit in the subscript (e.g. f_X or F_X). For any random variable X, $\mathbb{E}(X)$ and resp. $\mathbb{E}_X$ denote the expectation. For a real valued variable x, $[x]$ denotes the integral part of the value.

We refer to an estimated quantity by using the sample size n in the subscript, e.g. I_n refers to a mutual information estimate obtained from a sample with size n. The same notation is used to denote estimated pmf or pdf, e.g. $f_{X,n}$ or $p_{X,n}$ denotes the estimated pdf or pmf corresponding to a random variable X. The indicator function for x corresponding to a random variable X, is denoted as $\mathbb{I}_{X=x}$. We will use $\mathcal{N}(\mu, \sigma)$ to denote the Gaussian/normal distribution with mean μ and standard deviation σ. log and ln denote logarithm with base 2 and base e respectively.

In general, for side-channel setting we will use R to denote the random variable corresponding to the device noise.

2.1 Mutual Information

For general random variables X, Y the mutual information can be defined via the Radon-Nikodym derivative:

$$I(X;Y) = \int_{X \times Y} \log \frac{dP_{XY}}{dP_X P_Y} dP_{XY}.$$

Another common definition for MI, which is more commonly used in the side channel literature, is as a reduction in entropy, leading to the so-called "2H" and "3H" expressions:

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) && (1) \\ &= H(X) + H(Y) - H(X,Y) && (2) \end{aligned}$$

The conditional entropy term is defined as $H(X|Y) = \sum_y p(Y = y) H(X|Y = y)$ when Y is discrete random variable, and as $H(X|Y) = \int_y f_Y(y) H(X|y)$ when Y is continuous random variable.

As explained before, in the side channel setting, we are interested in the mutual information between the key and the observed leakage. The distribution of the key is typically known (consequently the entropy term corresponding to it can be directly calculated), but the distribution of the leakage, as well as the joint or conditional distribution (between key and leakage) is not known. Thus in side channel practice we have to inevitably estimate the mutual information (and if using the 2H/3H formulas, the resp. entropy terms). Because side channel observations can have many points (especially in the context of power and electro-magnetic emanation we are observing leakage traces), it is practically not desireable (and sometimes even feasible) to derive the precise statistical characterisation of each point in a leakage trace. As a consequence, we do not wish to make any distributional assumptions about the leakage (as well as joint/conditional leakage distributions).

2.2 Non-Parametric Mutual Information Estimation

MI estimators need to cope with different constellations of random variables (discrete or continuous). Consequently there exist three cases:

- two continuous random variables (referred to as cont. MI)
- two discrete random variables (referred to as discrete MI)
- a continuous and a discrete random variable (referred to as mixed MI)

The case of considering two discrete univariate random variables (i.e. discrete MI) has been solved in the statistical literature [1], which we will discuss below.

There are two principal ways to estimate the mutual information between two variables. The first option is to estimate the conditional/joint density and use either the 2H or the 3H formula to compute an MI estimate. The second option is to select an estimator that does not require the explicit estimation of the densities but use a nearest neighbour based estimator instead.

There exist a wide range of results for the first option, and recently a number of new ideas for the second option have emerged as well. The crucial property of any estimator is how well it "approximates" the true MI. This property is called the convergence of the estimator. It is defined over a sequence of the estimator in question, whereby the sequence is given over an increasing number of samples n.

The strongest notion of convergence of a sequence of estimators $\{\theta_n : n \in \mathbb{N}\}$ of θ is that the estimator *converges almost surely* (asymptotically, short a.s.) if

$$\mathbb{P}\left(\left\{\omega : \lim_{n\to\infty} \theta_n(\omega) = \theta(\omega)\right\}\right) = 1.$$

This is also written as $\theta_n \xrightarrow{a.s.} \theta$, and one often uses the term *strong consistency* in this context. Another strong notion of convergence is that of convergence in squared mean: $\lim_{n\to\infty} \mathbb{E}\left[\theta_n - \theta\right]^2 = 0$, or, equivalently, if $\lim_{n\to\infty} \mathbb{E}\left[\hat{\theta}_n - \theta\right] = 0$ and $\lim_{n\to\infty} Var(\hat{\theta}_n) = 0$. The weaker form of convergence is in probability which holds if for all $\varepsilon > 0$ $\lim_{n\to\infty} \mathbb{P}\left(\{\omega : |\theta_n(\omega) - \theta(\omega)| > \varepsilon\}\right) = 0$. This is also written as $\theta_n \xrightarrow{\mathbb{P}} \theta$.

MI Based on Density Estimation The most widely applied technique of MI estimation in side-channel analysis is via entropy estimation e.g. via the 2H formula:

$$I_n(X, Y) = H_n(X) - H_n(X|Y). \tag{3}$$

In general the estimation of $H(X|Y)$ i.e. computing $H_n(X|Y)$ requires estimating $H(X|y)$ as well as pdf or pmf of Y. In the context of side-channel analysis one random variable, namely Y is discrete, it assumes finitely many values, and is typically uniformly distributed. Thus, it suffices to estimate only $H(X|Y = y)$ for each possible value $y \in Y$. Evidently, the convergence of the MI estimator based on the 2H approach depends fully on the convergence of the entropy estimator.

There are several relevant plug-in estimators (coping with general distributions) for entropy that connect to the current practice in side-channel analysis and leakage certification, namely - the *integral estimate* and *resubstitution estimate*. Györfi and van Mulen [8] showed that integral estimator is strongly consistent if the conditional distribution satisfies specific conditions. The resubstitution estimator only provides mean-square consistency Hall and Morton [9] (again under certain conditions regarding the distribution). Both estimators do not generalise to the multivariate setting (either their efficiency drops significantly or the convergence guarantee does not extend to the multivariate setting).

The plug-in estimator produces the best results in terms of consistency when the random variables are both discrete. For the plug-in pmf estimate p_n, the corresponding entropy and MI estimates are defined as

$$H_n = \sum_x p_n(x) \log p_n(x), \quad I_n(X, Y) = \sum_{x,y} p_n(x, y) \log_2 \frac{p_n(x, y)}{p_{X,n}(x) p_{Y,n}(y)} \tag{4}$$

where $p_n(x, y) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}_{X_i = x, Y_i = y}$. Antos and Kontoyiannis [1] showed that if $H(X, Y)$ is finite then the plug-in estimator of MI as in Equation (4) is both strongly consistent and mean square consistent i.e. $I_n \xrightarrow{a.s.} I$ and $\lim_{n \to \infty} \mathbb{E}[(I_n - I)^2] = 0$.

Bronchain et al. [3] put forward the notions of ePI and eHI as MI estimators. These quantities are based on the 2H entropy estimate for MI. Instead of using one of known estimation techniques for entropy, they first compute an *empirical pdf*, denoted by $\tilde{e_n}(x|y)$ (based on discretised leakage x, and a key dependent variable y) in the form of a histogram. With the empirical pdf, they then define:

$$\mathrm{eHI}_n(X; Y) = H(Y) + \sum_{y \in Y} p_Y(y) \cdot \sum_{x \in X} \tilde{e}_n(x|y) \log_2 \tilde{e}(y|x) \tag{5}$$

$$\mathrm{ePI}_n(X; Y) = H(Y) + \sum_{y \in Y} p_Y(y) \cdot \sum_{x \in X} p_n(x|y) \log_2 \tilde{e}(y|x) \tag{6}$$

Previous papers [3–5]) assume that the noise distribution is Gaussian (e.g. $R \sim \mathcal{N}(\mu, \sigma)$), and the device leakage function is deterministic and discrete. Because the eHI and ePI are defined within the side channel setting, only one of the random variables is continuous. With this assumption, Bronchain et al. [3] show that the eHI converges to the $I(L;Y)$ (in their notation $Y = \mathcal{C}(X, K)$) and is a bound for $I(L;Y)$, thus the eHI is a biased estimator.

Nearest Neighbour Estimator for MI Motivated by the need for a non-parametric MI estimator that applies even to high-dimensional/multivariate problems, [11] introduced the idea of using a t nearest neighbour (short t-NN) based estimator (also known as KSG estimator in the wider statistical literature). Recently, a generalization of the KSG estimator was proposed by Gao, Krishnan, Oh and Vishwanath [6] (that we will refer to as GKOV estimator), which is applicable to a mixture of continuous and discrete random variables. The GKOV estimator is defined as

$$I_n(X;Y) = \frac{1}{n}\sum_{i=1}^{n} \hat{I}_i = \sum_{i=1}^{n}(\psi(\tilde{t}_i) + \log n - \log(n_{x,i}+1) - \log(n_{y,i}+1)). \quad (7)$$

Here, $\psi(u)$ is the digamma function $\psi(u) = \frac{d}{du}\log \Gamma(u) \approx \ln u - \frac{1}{2u}$. The details of how to compute the quantities $n_{x,i}, n_{y_i}$ and $\tilde{t}_i$ can be found in algorithm 1.

The following theorem guarantees the mean square convergence of the GKOV estimator (defined in equation 7).

Theorem 1 (Convergence of MI estimator [6]). *Assume that*

- *t is chosen to be a function of the sample size n s.t. $t_n \to \infty$ and $t_n \log n/n \to 0$ as $n \to \infty$*
- *The set of discrete points $\{(x,y) : \mathbb{P}_{XY}(x,y,0) > 0\}$ is finite where*

$$\mathbb{P}_{XY}(x,y,r) = \mathbb{P}_{XY}(\{(a,b) \in X \times Y : \|a-x\| \le r, \|b-x\| \le r\}$$

- $\int_{X\times Y} \left|\log \frac{d\mathbb{P}_{XY}}{d\mathbb{P}_X d\mathbb{P}_Y}\right| d\mathbb{P}_{XY} < \infty$.

Algorithm 1 $I(X;Y)$ estimation for mixed r.v.s (X,Y) [6]

```
Require: ${x_i, y_i}_{i=1}^n$ and $t_n = t$
for $i = 1, \ldots, n$ do
    $d_{i,xy} = t$th smallest distance from${d_{ij} = \max\{\|x_j - x_i\|, \|y_j - y_i\|\} : i \neq j}$
    if $d_{i,xy} = 0$ then
        $\tilde{d_i} = |\{j : d_{ij} = 0\}|$
    else
        $\tilde{d_i} = t$
    end if
    $n_{x,i} = |\{j : \|x_j - x_i\| \le d_{i,xy}\}|$
    $n_{x,i} = |\{j : \|y_j - y_i\| \le d_{i,xy}\}|$
    $\alpha_i = \psi(\tilde{d_i}) - \log(n_{x,i}+1) - \log(n_{y,i}+1)$
end for
return $\frac{1}{n}\sum_i \alpha_i + \log(n)$
```

Then, $\lim_{n\to\infty} \mathbb{E}(I_n) = I(X,Y)$*. Additionally, if we assume* $(t_n \log n)^2/n \to 0$ *as* $n \to \infty$*, then* $\lim_{n\to\infty} Var(I_n) = 0$*.*

With a suitable choice of t_n the GKOV estimator has the same convergence rate as existing pmf/pdf based mutual information estimators.

3 MI in the Side Channel Setting

In the side channel setting we work with random variables that represent inputs/intermediates/outputs of cryptographic processes and leakage observations: we use $x \in X$ for the input, which is mapped according to the cryptographic process via the application of some (cryptographic) target function(s) $\mathcal{C}$ and an (unknown) key $k^* \ \in K$ to an intermediate $y \in Y$. Implementations process cryptographic keys in "chunks", thus K is typically of a small size.

An adversary is assumed to be able to observe inputs/outputs $x \in X$ of the device and the side channel leakage trace $l \in L$ that corresponds to the execution of a cryptographic algorithm using the input and the key k^* that is embedded in the device. A side channel trace is a vector of leakage points. Each point corresponds to the physical processes that happen inside the device (at that point in time/step in the execution) and some independent noise R.

An important detail is that the observed leakage L may be either a deterministic or a probabilistic function of multiple variables. The secret key $k^* \in K$ and the input X interact via the target function $\mathcal{C}$ (a step in the computation of the cryptographic algorithm) and leak via the (unknown) leakage function $\mathcal{L}$. The leakage function for a specific step in the execution of an algorithm can be simple (e.g. Hamming weight for a bus transfer), in which case it can be modelled as a discrete deterministic random variable:

$$L = \mathcal{L}(\mathcal{C}(X,K)) + R.$$

But for many steps in an execution the leakage function depends on some complex interaction between many components in the device, and is influenced by probabilistic processes (due to glitches, cross talk, couplings, etc.). It is also possible that the measurement setup itself impacts on the leakage, and as a result, leakage functions are often continuous probabilistic random variables. We model such a probabilistic leakage function by some unknown internal randomness S. Note that S can be discrete or continuous, and it is different from R. Unlike R, the random variable S is not independent of (X,K) i.e. the leakage density function is $f(x,k,s)$. For a target device we can then model the observed leakage as

$$L = \mathcal{L}(S, \mathcal{C}(X,K)) + R.$$

Previous work has exclusively considered deterministic leakage functions, but we consider deterministic and probabilistic leakage functions in our work. This is important because we do want a mutual information estimation process to be strongly consistent *for all observable leakage functions*, even if we don't understand their true nature.

Whenever the probabilistic nature of the leakage is not relevant, i.e. a statement holds irrespective of S and thus irrespective of whether $\mathcal{L}$ is discrete and deterministic or continuous and probabilistic, we drop S in the text for readability.

3.1 Relationships between MI Quantities

The mutual information between the observed traces and the pair (input, key) measures how much information about the key is contained in the observable trace. We call this I^{max}:

$$I^{\mathsf{max}} = I((X, K); L). \tag{8}$$

The cryptographic target function $\mathcal{C}$ maps the key and input value to an intermediate value. Consequently, because of the data processing inequality we know that

$$I^{c} = I(\mathcal{C}(X, K), L) \leq I^{\mathsf{max}}. \tag{9}$$

Equality holds if and only if $\mathcal{C}$ is one-to-one. In an attack, a worst-case adversary would utilise a power model. In the best case, this model would be exactly the device leakage mode. If the device leakage is independent of $\mathcal{C}$ then we can utilise the data processing inequality again:

$$I^{b} = I(\mathcal{L} \circ \mathcal{C}(X, K), L) \leq I^{c} \leq I^{\mathsf{max}}. \tag{10}$$

Consequently, the mutual information I^{max} is indeed an upper bound to other MI quantities of interest: it represents the ability of an ideal adversary. Finally, we briefly consider the so-called worst case adversary [2]. The worst-case adversary is a profiling adversary [2] utilising a profiling model $\mathcal{L}_{wca}$. Thus the mutual information $I^{\mathsf{wca}} = I(\mathcal{L}_{wca} \circ \mathcal{C}(X, K), L)$ characterises the worst-case adversary, in the sense that if $|I^{\mathsf{wca}} - I^{b}| < |I^{\mathsf{wca}'} - I^{b}|$ then the model $\mathcal{L}_{wca}$ is closer to $\mathcal{L}$ when measured by the MI than $\mathcal{L}_{wca'}$.

3.2 The Curse of Discretisation

The discretization of a random variable is often used in the context of estimating parameters from real data, particularly, when the observed random variable is continuous and can assume values in $(-\infty, \infty)$. Discretization divides the range of a continuous random variable X into possibly an infinite number of intervals. The quality of the discretization is clearly extremely important for any statistical process that follows on from it: e.g. the number of intervals needs to be appropriate, but also the width of intervals, which may have to depend on the shape of the underlying continuous density.

Consequently, also the consistency of an entropy estimate of a quantized random variable depends strongly on the method used for quantization. If a mutual information estimate is defined via the 2H formula, then also this estimate highly depends on the discretisation. For example, when X is a continuous real valued random variable and $[X]$ defines the corresponding quantised random variable, Rényi showed [14] early on that the entropy of the sequence $[qX]$ satisfies

$$\lim_{q \to \infty} (H([qX]) - \log_2 q) = H(X)$$

where $1/q$ is the step-size. Clearly this early result demonstrates that the entropy of the discretised random variable is *larger* than the entropy of the real valued random variable, especially if q is not large. In the computation of the mutual information with the 2H formula, the difference between two entropies can potentially be smaller or larger than the true mutual information. Which inevitably, and irrespective of the number of available observations, leads to *biased* estimators for the mutual information. The paper by Paninski [12] demonstrates that even bias corrected estimators are still surprisingly biased in some situations. Bronchain et al.'s work is exactly the type of discrete estimator that Paninski discussed, and their eHI (5) estimator is exactly such a biased estimator. Bronchain et al. use the eHI as an upper bound for the true mutual information: eHI $\geq I(L;(X,K))$. Such a bound is indeed useful in the absence of a consistent estimator of $I(L;Y)$. However, when a consistent estimator of $I(L;Y)$ is available one can simply use it to estimate the mutual information.

Crucially, in the context of leakage certification, where model quality is judged by a mutual information estimate, using a biased estimator implies that convergence to the true mutual information is simply impossible. This is neither an issue with the profiling model (i.e. this is not an assumption error) nor is it an estimation error [4], it is simply a property of the estimator that is based on discretisation. Thus to judge the model quality for a worst case adversary $I^{\mathsf{wca}} = \mathcal{L}_{wca} \circ \mathcal{C}(X,K)$ it is important to avoid discretisation so as to avoid arriving at a biased estimate.

4 Model Free MI Estimation

In this section we show that $I^b(\mathcal{L} \circ \mathcal{C}(X,K), L) = I^{\mathsf{max}}$ under mild conditions on $\mathcal{L}$ (and for bijective $\mathcal{C}$): the best MI (the one that corresponds to the knowledge of the true leakage function, and the target intermediate value) is equal to the MI between the discrete inputs (key and data) and the observed leakage. This equality implies that in many practical cases the "best" MI can be calculated without the need for estimating the distribution of the device leakage $\mathcal{L}$.

4.1 Determining the Conditional Distribution $L|\mathcal{L}$

The observed leakage from a traget device is modeled as $L = Z + R$, where Z can be either:

$$\begin{aligned} Z &= \mathcal{L}(\mathcal{C}(X,K)), \text{deterministic in} (X,K) \\ Z &= \mathcal{L}(S, \mathcal{C}(X,K)), \text{probabilistic in} (X,K). \end{aligned}$$

Note that S can be discrete or continuous, and it is different from R. Unlike R, the r.v. S is not independent of (X, K). Independent of the nature of Z, we can show that the the distribution of $L|Z$ is fully determined by the distribution of R, simply because R is independent of Z. The formal argument for a continuous Z is slightly more complex.

Z is deterministic and discrete. The pdf $f_{L|Z}$ of the conditional variable $L|Z$ can easily be derived:

$$\begin{aligned}\mathbb{P}(L \leq \ell | Z = z) &= F_R(\ell - z) \\ &= \int_\ell dF_R(\ell - z) = \int_\ell f_R(\ell - z) d\ell \end{aligned} \tag{11}$$

For example, when $R \sim \mathcal{N}(0, \sigma)$ which is an assumption often made in SCA, $L|Z = z \sim \mathcal{N}(z, \sigma)$.

Z is continuous and probabilistic. The conditional pdf is given as

$$f_{L|Z}(l|z) = \frac{f_{LZ}(l, z)}{f_Z(z)}.$$

To derive the joint distribution f_{LU}, we use a standard trick and define the transformation $L = Z + R, U = Z$ and obtain

$$f_{LU}(l, u) = \frac{f_{RZ}(r, z)}{\left|\frac{\partial(l,u)}{\partial(r,z)}\right|} = f_R(r) f_Z(z) = f_R(l - z) f_Z(z) \implies f(l, z) = f_R(l - z) f_Z(z) \tag{12}$$

since the Jacobian $\left|\frac{\partial(l,u)}{\partial(r,z)}\right| = 1$, and $f_{RZ}(r, z) = f_R(r) f_Z(z)$ due to the independence of R and Z. Thus from equation 12 (and recalling that $Z = U$) we obtain

$$f_{L|Z}(l|z) = f_R(l - z)$$

4.2 Computing the MI Does Not Require Any Model

Now we can prove that under suitable noise distributions (and assuming that $\mathcal{C}$ is bijective) $I^b = I^{\mathsf{max}}$. The MIs $I(L; Y)$ where $Y = \mathcal{C}(K, X)$, and $I(L; Z)$ where $Z = \mathcal{L}(Y)$ can be written as

$$I(L; Y) = H(L) - H(L|Y) \tag{13}$$

$$I(L; Z) = H(L) - H(L|Z) \tag{14}$$

Proposition 1. *Suppose R follows a distribution $\mathcal{D}$ with the location and scaling parameters μ and σ (> 0) respectively i.e. $R \sim \mathcal{D}(\mu, \sigma)$, and $H(R) = \varphi(\sigma)$ where φ is a function determined by the form of f_R. Then $I(L; (X, K, S)) = I(L; \mathcal{L}(S, \mathcal{C}(X, K)))$ where S is the r.v. quantifying the internal randomness of the target device.*

Proof. We again consider the two cases: first $\mathcal{L}$ being deterministic, and second $\mathcal{L}$ being probabilistic.

First, we assume that Z is discrete and deterministic $Z = \mathcal{L} \circ \mathcal{C}(X, K)$:

$$\begin{aligned} I(L; Z) &= H(L) - H(L|Z) = H(L) - \sum_z p_Z(z) H(L|Z = z) \qquad (15) \\ &= H(L) - \sum_z p_Z(z) \mathbb{E}_{L|z}(-\log(f_R(l - z)) \\ &= H(L) - \sum_z p_Z(z) \varphi(\sigma) = H(L) - \varphi(\sigma) \end{aligned}$$

When Z is continuous (but still deterministic), the sum (over z) is replaced by $\int_z$ and the pmf $p(z)$ is replaced with pdf $f(z)$ in Equation (15). More precisely, we have

$$\begin{aligned} I(L; Z) &= H(L) - \int_z f(z) \mathbb{E}(-\log(f_{L|Z}(l|z)) dz \qquad (16) \\ &= H(L) - \int_z f(z) \mathbb{E}(-\log(f_R(l - z)) dz = H(L) - \varphi(\sigma). \end{aligned}$$

Now we consider the term $H(L|(X, K, S))$ (in $I(L; (X, K, S))$) which is defined as

$$H(L|(X, K, S)) = \sum_{x,k} \int_s f(x, k, s) \mathbb{E}[-\log f(l|(x, k, s))] ds \qquad (17)$$

As before, using the transformation of random variables (and the corresponding Jacobian) we have

$$f(l|(x, k, s)) = f_R(l - v) \qquad (18)$$

where $\mathcal{L}(s, \mathcal{C}(x, k)) = v$ and $\mathcal{L}$ is continuously differentiable function.

Thus we get

$$\begin{aligned} I(L; (X, K, S)) &= H(L) - H(L|(X, K, S)) \\ &= H(L) - \sum_{x,k} \int_s f(x, k, s) \mathbb{E}[-\log f(l|(x, k, s))] ds \qquad (19) \\ &= H(L) - \sum_{x,k} \int_s f(x, k, s) \mathbb{E}[-\log(f(l - v)] ds \\ &= H(L) - \varphi(\sigma) \sum_{x,k} \int_s f(x, k, s) ds = H(L) - \varphi(\sigma) \end{aligned}$$

The integral $\int_s$ in Equation (19) is replaced by $\sum_s$ when S (and consequently Z) is discrete. □

Remark 1. We emphasize that the entropy assumption in Proposition 1 encompasses many distributions e.g. Normal distribution, Laplace distribution, Cauchy distribution etc. The distributions characterized by this assumption are also most commonly used in side-channel analysis experiments in literature e.g. [10]. Proposition 1 can easily be extended for a distribution of R characterized with only scaling parameter $\sigma > 0$, which is done in the following proposition.

Proposition 2. *Suppose, R follows a distribution $\mathcal{D}$ with scaling parameter $\sigma > 0$. Then $I(L; Z) = I(L; (X, K, S))$.*

Proof. The proof follows the same way as in proof of Proposition 1.

Proposition 2 covers the general exponential distribution of R (with pdf $f(x) = \sigma e^{-\sigma x}$ when $x \geq 0$ and $f(x) = 0$ otherwise).

Note that the r.v. S in Proposition 1 and Proposition 2 was introduced to define the output distribution of $\mathcal{L}$. In practice it is implicit to a device and is not required in the process of leakage certification (or MI estimation). Thus, we have the following result.

Theorem 2 (Model Independent MI Estimation). *If the entropy of the noise distribution (of a device) is location independent then $I(L; \mathcal{L} \circ \mathcal{C}(X, K))$ can be computed (or estimated) without any hypothetical (leakage) model, by simply computing (or estimating) $I(L; (X, K))$.*

Proof. Follows from the proof of Proposition 1 and Proposition 2. □

A special case of Proposition 1 is when noise distribution is Gaussian, and we use this special case to provide a supporting concrete example.

4.3 Model Free MI Estimation under Hamming Weight Leakage

We illustrate the result of Theorem 2 with the commonly used device leakage function - Hamming weight (HW), where $\mathcal{L}(u) = \mathsf{HW}(u)$. Suppose, we use a m-bit cryptographic Sbox as the $\mathcal{C}$ i.e. $Y = \mathcal{C}(x, k) = \text{Sbox}(x \oplus k)$ and the underlying noise distribution follows $\mathcal{N}(0, \sigma)$. Then, L will follow a mixture of Gaussian with pdf

$$\sum_{i=0}^{m} \frac{1}{2^m} \binom{m}{i} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-i)^2}{2\sigma^2}}.$$

For any given value $j = \mathcal{C}(x \oplus k)$, $L|(Y = j)$ has pdf $\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\ell - h_j)^2}{2\sigma^2}}$ where $j = 0, 1, \ldots, m$ and $h_j = \mathsf{HW}(j)$. So, the entropy of $H(L|Y = j) = \frac{1}{2}\log(2\pi e\sigma^2)$. Hence,

$$I(L; Y) = H(L) - \sum_j \mathbb{P}(Y = j) H(L|Y = j) = H(L) - \frac{1}{2}\log(2\pi e\sigma^2).$$

On the other hand $H(L|\mathsf{HW}(Y)=i)$ also has pdf $f(\ell|i) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\ell-i)^2}{2\sigma^2}}$. Thus,

$$\begin{aligned}\sum_i \mathbb{P}(\mathsf{HW}(Y)=i)\mathbb{E}_{L|i}[-\log f(\ell|i)] &= \frac{1}{2}\log(2\pi e\sigma^2)\sum_{i=0}^{m}\frac{1}{2^m}\binom{m}{i} \\ &= \frac{1}{2}\log(2\pi e\sigma^2)\end{aligned}$$

Hence, we will have

$$I(L;\mathsf{HW}\circ\mathcal{C}(X,K)) = H(L) - \sum_i \mathbb{P}(\mathsf{HW}(Y)=i)\mathbb{E}_{L|i}[-\log f(\ell|i)] = I(L;\mathcal{C}(X,K)).$$

Remark 2. The above example is in fact a special case of the analysis provided in [13]. Our result (Theorem 2) is much more general than the earlier work of [13], because it can be applied to any noise distribution whose entropy is location independent. In Section 6.1 we provide further experimental evidence by simulating the above example for Gaussian and Laplacian noise distributions.

4.4 Multivariate Analysis

No assumptions were necessary in the previous section regarding the dimensionality of L, thus our analysis naturally applies also to situations where multiple trace points are considered jointly. Working with joint distributions is also of interest when considering countermeasures, such as masking, where intermediate values are split up into multiple shares.

Consider a simple two share setting, where for an intermediate value we just have one additional random variable, denoted by M. From the data processing inequality we know that

$$I(L;(X,K,M)) \geq I(L;\mathcal{C}(X,K,M)) \geq I(L;\mathcal{L}\circ\mathcal{C}(X,K,M)). \tag{20}$$

Because the mask is part of the target function, and is chosen independently of all other quantities, it does not change our analysis from before.

It is important to bear in mind that $I(L;(X,K,M)) \geq I(L;(X,K))$. Consequently an evaluator, who may have access to the masks, can accurately compute the mutual information between the observed (multivariate) leakage and the variables X, K, M. This mutual information quantity again represents a best case. In practice leakage from shares is often exploited by adversaries via a further processing of the traces, which can only lead to a smaller mutual information value (based on the data processing inequality).

5 MI Estimation Using t_n-NN Method

The recently proposed GKOV estimator [6] is a strongly consistent for general mixtures, that is, it can be used for the estimation of the MI between any combination (including discrete/continuous) of random variables. Their estimator is based on the nearest neighbour principle, and it does not require any explicit density estimation. In contrast to previous nearest neighbour estimators, the t (from t-NN estimator) value in the GKOV estimator is now a function of the sample size n (thus denoted as t_n), rather than a constant.

5.1 Fast implementation of Alg.1

In practice, the $I_n(L, Y)$ is computed for $L \in \mathbb{R}^m$ where $m \geq 1$. Depending on the scenario that is considered in an evaluation, the mutual information estimate is calculated for each point in a leakage trace independently of all other points (univariate setting), or over multiple points (multivariate setting).

The computational cost for estimating the mutual information in a multivariate setting using a histogram method (pdf estimation method) is high and a known problem. For finding a "good" binning strategy one may need to compute I_n for range of values of the tuple $(b_1, b_2, \ldots, b_m) \in \mathbb{Z}^m$, where b_i denotes the number of bins along each dimension. This naturally increases the cost of estimating the mutual information using a histogram method.

In contrast the t_n–NN estimator by Gao et al. does elegantly generalise to multiple points because it's only configuration parameter is the function t_n (based on the sample size). This is a significant advantage over previous t–NN estimators. The only remaining computational challenge is measuring the distance of all sample points L_j from the sample point L_i where $j \neq i$ for each i. A number of efficient algorithms for finding nearest neighbours are part of common machine learning libraries in both C/C++ and Python.

For our C++ implementation of MI estimation, we used the popular machine learning library mlpack. The library offers several in-built distance metrics including the option of providing a custom distance metric. From the available options of efficient nearest neighbour search algorithms we used VPTree and BallTree. Note that the search algorithm may depend on the choice of distance metric. For example, the ℓ_∞ metric is not compatible with the KDTree search algorithm. This is not a limitation of mlpack but a consequence of the mathematical requirements of a specific search algorithm.

For calculating distances of each sample point from all other points which is necessary beyond the NN search, we have used OpenMP to parallelize the computation. Note that the OpenMP library can also be used by mlpack if it is available on the system. A particular observation on this part of our experiment is that for multidimensional leakage, computing the ℓ_∞ norm with an unrolled loop is more efficient than using the looped version or the mlpack library function for the same. For example, with the dimension $m = 2$, computing the ℓ_∞ norm as

```
max( abs(data(i,0)-data(j,0)), abs(data(i,1)-data(j,1)) );
```

is more efficient than using the library function

```
arma::norm(data.row(i)-data.row(j), "inf");
```

For all experiments we have used an Intel(R) Core(TM) i7-8700 CPU 3.20GHz system having 6 CPU cores and Ubuntu operating system.

5.2 Establishing Practical Choices for t_n

The parameter t_n, which is a function of the number of side channel observations n, can be chosen by observing the convergence of the sequences $t_n \log n/n$ and $(t_n \log n)^2/n$ (in theorem 1). In our experiments we selected t_n equal to $\log n$ and $\log_{10}^2 n$. Figures 1 & 2 show the convergence behaviour of the t_n–NN estimator (Alg. 1) in different situations. To create these plots, we performed a number of simulations. Each simulation fixes a simple, practically relevant, device leakage function $\mathcal{L}$ (Figure 1 considers Hamming weight for two plots, a weighted linear combination of the bits of the Sbox output for the other plot. While Figure 2 have taken a non-linear leakage model of Sbox output for different noise distributions), an algorithmic target $\mathcal{C}$ which is always the AES Sbox, and some defined noise (Gaussian or Laplace). Thus each simulation randomly creates leakage samples, and then uses Alg. 1 with two different choices for t_n to estimate the MI. Each simulation is repeatedly executed, thus the y-axis represents averages of calculated mutual information estimates. Because we fully specify the leakage function and the noise distribution, we can also compute the true mutual information via numerical integration. Figures 1 & 2 thus show both the behaviour of the estimator w.r.t. the choice of t_n ($\log n$ performs better overall, and in particular for lower n) and simultaneously, it shows that it converges to the true mutual information. In the remaining practical experiments, we will set $t_n = \log n$.

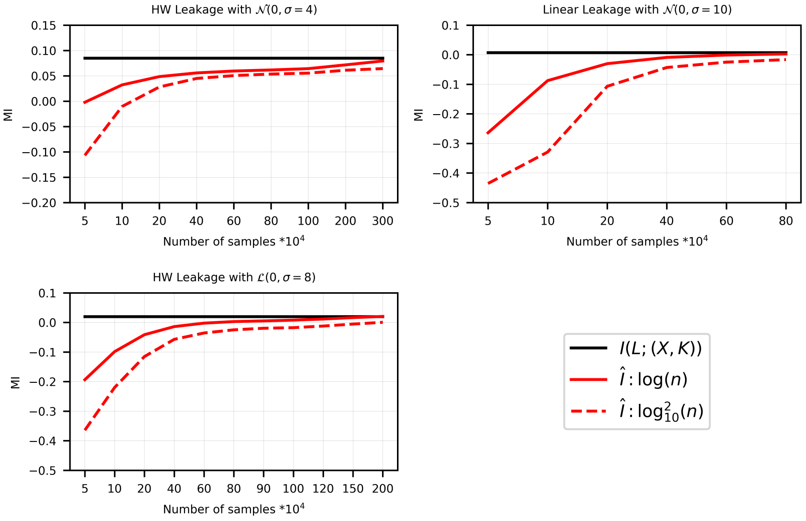

Fig. 1. Experiment: convergence rate of t_n–NN estimator for different choices of t_n for cases related to linear leakage models. Here $\hat{I}$ denotes the estimated MI, and $\mathcal{L}$ and $\mathcal{N}$ denotes Laplace and Normal distribution respectively.

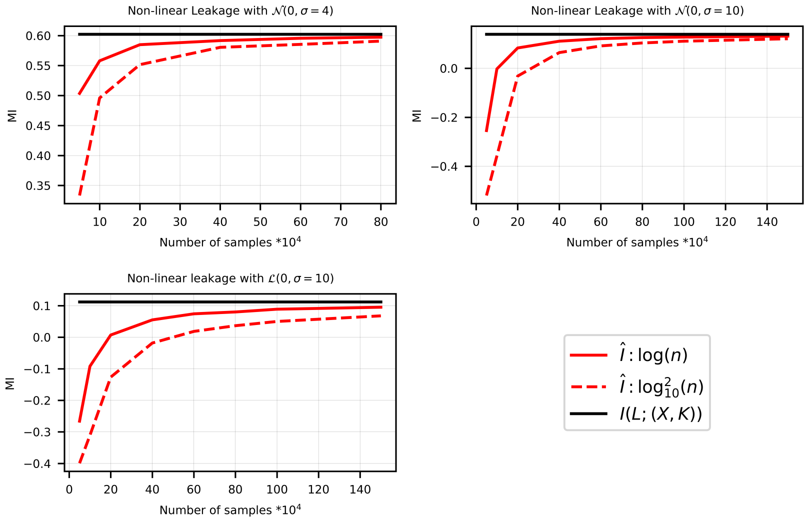

Fig. 2. Experiment: convergence rate of t_n–NN estimator for different choices of t_n related to nonlinear leakage model. Here $\hat{I}$ denotes the estimated MI, and $\mathcal{L}$ and $\mathcal{N}$ denotes Laplace and Normal distribution respectively.

A practically relevant observation is that, unlike a plug-in (histogram) estimator that requires data dependent parameter tuning, the choice of the parameter t_n can be pre-determined based only on the sample size n. Furthermore the choice of t_n only affects the rate of convergence, i.e. the efficiency of the estimation unlike histogram based estimators, where a wrong choice can lead to bias.

6 Experimental Evaluation and Comparison

We now provide a range of experiments, which are based on highly controlled experiments with simulated leakage, as well as experiments were we sampled real leakage traces from a micro-processor. The purpose of these experiments are to provide a comparison with the t_n-NN estimator that we propose as a better tool for mutual information estimation, with the commonly used alternatives from the side channel literature.

6.1 Experiments Based on Simulated Leakage

Like in the simulations before, we fix a cryptographic target function (the AES SubBytes operation), as well as some simple but practically relevant leakage models (Hamming weight , a linear combination of bits and a nonlinear combination of bits), alongside some noise distributions for the experiments in the

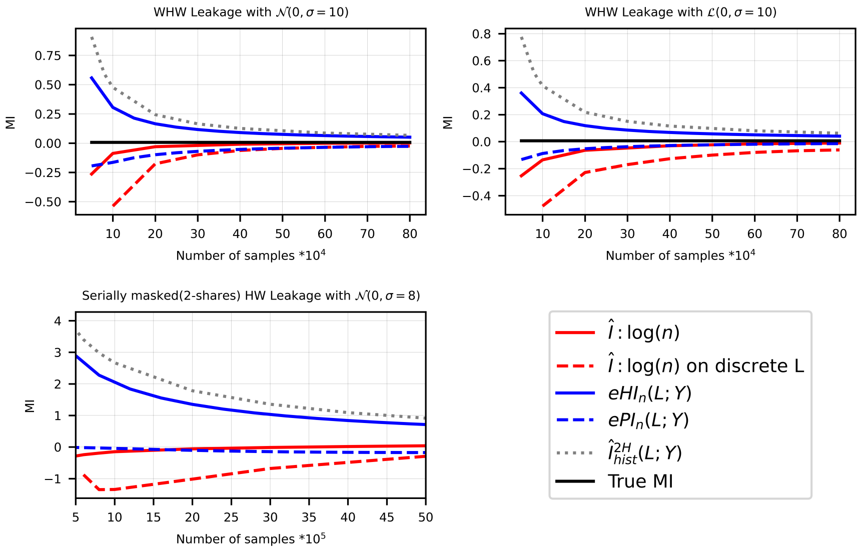

Fig. 3. Experiment: comparison of estimators for the simulated linear leakage AES SubBytes. Here $\hat{I}$ denotes the estimated MI, and $\mathcal{L}$ and $\mathcal{N}$ denotes Laplace and Normal distribution respectively.

following. We show two experiments that correspond to a "naive Sbox", i.e. no masking is considered, and one experiment where the Sbox inputs and outputs are represented by two shares. The mutual information in the latter case is then defined bi-variate (i.e. in the respective mutual information quantities L is a vector consisting of two leakage points). All plots are based on repeatedly running an experiment, and thus the y-axis represents an average mutual information value.

The calculation of the t_n–NN estimator, as well as the classical histogram-based plugin estimator, is based on our own implementation (we described this before). For the computations of the eHI and the ePI, we use the Python scripts that were provided by the authors of [3]. We show the mutual information estimates for the t_n–NN estimator for the continuous leakage (as it is produced by our simulation), as well as for the discretised leakage. The discretised leakage is also the input for the eHI and ePI estimates: we ensured that the discretisation for the t_n–NN estimator produces the same distribution as the discretisation that is used as part of the eHI and ePI computation. Because in a simulated setting with simple leakage functions we know all the involved distributions, we are able to plot the true mutual information (it is just above zero), which we derive via numerical integration.

The experiments clearly demonstrate that the t_n–NN estimator quickly converges to the true mutual information value (even when dealing with the discre-

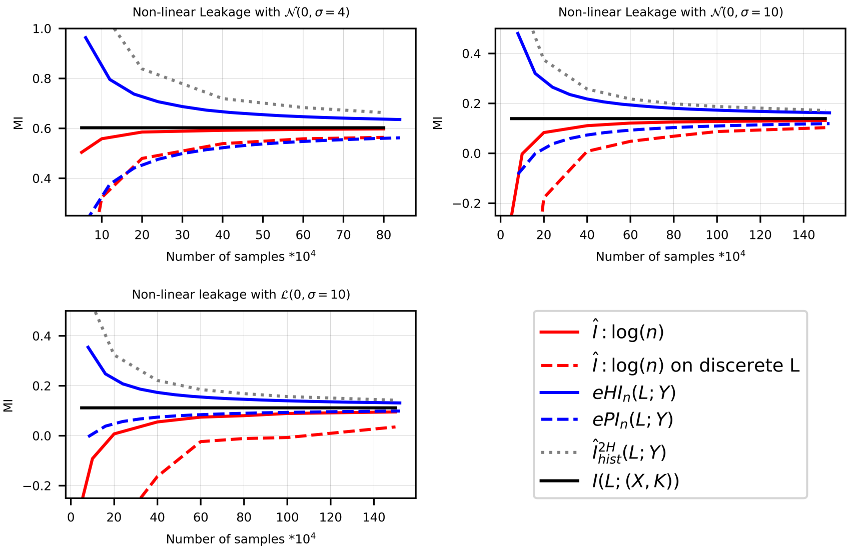

Fig. 4. Experiment: comparison of estimators for the simulated nonlinear leakage AES SubBytes. Here $\hat{I}$ denotes the estimated MI, and $\mathcal{L}$ and $\mathcal{N}$ denotes Laplace and Normal distribution respectively.

tised form of the leakage), whereas the eHI, as explained in the previous work is a biased upper bound. The bias of eHI in our experiments appears to be small in the univariate experiments, but larger in the bi-variate experiment. The convergence of the histogram-based plug-in estimator is particularly bad, and it also appears to show bias. When considering all three experiments with respect to the bias of the (non t_n–NN) estimators, it should be clear that the problem is not just that there is bias, but that the bias can have different magnitudes depending on the leakage characteristics and the dimensionality of the leakage. This means that in practice we cannot know, unless we fully understand the leakage and how it interacts with our chosen estimator, what kind of bias we should expect. In practice, we want to apply this type of mutual information estimation for a trace with hundreds of thousands of trace points, where each of them corresponds to slightly different leakage functions. Consequently, mutual information "traces" cannot be soundly interpreted unless an unbiased estimator has been applied.

6.2 Experiments Based on Real Leakage

We complement our experiments based on simulations with two experiments based on real power traces. The traces were acquired from a setup based on an ARM Cortex M3 micro-processor. We take measurements directly from the supply voltage of the core, without any further processing, thus they are suitable for the eHI, ePI, as well as the t_n–NN estimator.

The traces are sampled from running parts of an AES implementation (written in Thumb Assembly language). The first experiment samples the leakage from a key addition i.e. the computation $x \oplus k$, followed by a table lookup operation, which represent the execution of AES SubBytes $S(x \oplus k)$. The second experiment relates to the same basic operations, but now each operand is split up into two shares, thus representing a simple two share masked implementation. We capture here only the salient table look-up operation.

Figures 5 and 6 shows the results of running four different mutual information estimators: a histogram based estimator based on [8], eHI, ePI, and the t_n–NN estimator. Both plots show in fact averages of mutual information estimates: we ran each estimator multiple times.

Both experiments show how dramatically biased the straightforward histogram based mutual information estimate turns out. The ePI and eHI estimates are consistently larger than the t_n–NN estimate, which is expected as they are also known to be positively biased. The plot for the unmasked implementation, which contains also contains the key addition, also nicely demonstrates that for a bijective target function $\mathcal{C}$, we see indeed that $I^c = I^{\max}$ because the eHI and the t_n–NN estimator track each other nicely.

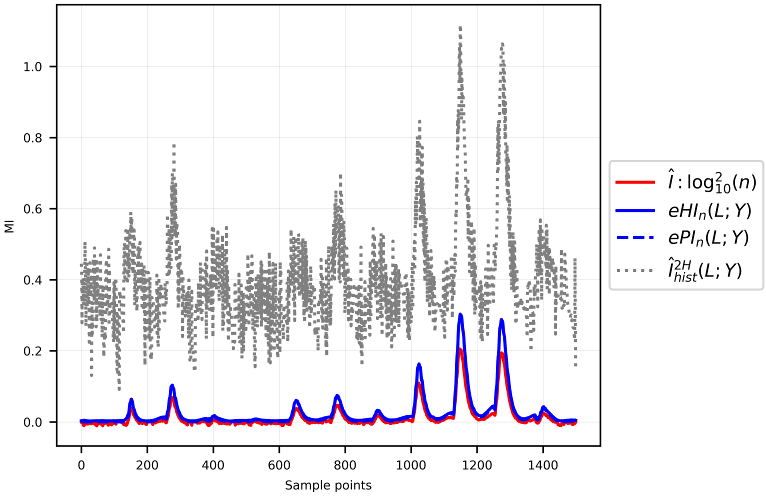

Fig. 5. Experiment: comparison of estimators for the leakage of a naive AES SubBytes implementation. Here $\hat{I}$ denotes the estimated MI.

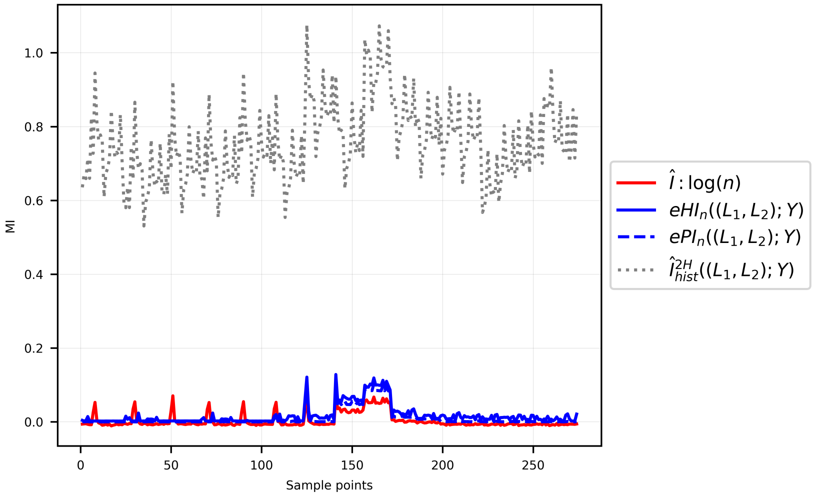

Fig. 6. Experiment: comparison of estimators for the leakage of an AES implementations based on two shares

References

1. András Antos and Ioannis Kontoyiannis. Convergence properties of functional estimates for discrete distributions. *Random Structures & Algorithms*, 19:163 – 193, 10 2001.
2. Melissa Azouaoui, Davide Bellizia, Ileana Buhan, Nicolas Debande, Sébastien Duval, Christophe Giraud, Éliane Jaulmes, François Koeune, Elisabeth Oswald, François-Xavier Standaert, and Carolyn Whitnall. A systematic appraisal of side channel evaluation strategies. In Thyla van der Merwe, Chris J. Mitchell, and Maryam Mehrnezhad, editors, *Security Standardisation Research - 6th International Conference, SSR 2020, London, UK, November 30 - December 1, 2020, Proceedings*, volume 12529 of *Lecture Notes in Computer Science*, pages 46–66. Springer, 2020.
3. Olivier Bronchain, Julien M. Hendrickx, Clément Massart, Alex Olshevsky, and François-Xavier Standaert. Leakage certification revisited: Bounding model errors in side-channel security evaluations. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part I*, volume 11692 of *Lecture Notes in Computer Science*, pages 713–737. Springer, 2019.
4. François Durvaux, François-Xavier Standaert, and Santos Merino Del Pozo. Towards Easy Leakage Certification. In Benedikt Gierlichs and Axel Y. Poschmann, editors, *Cryptographic Hardware and Embedded Systems – CHES 2016*, pages 40–60, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.
5. François Durvaux, François-Xavier Standaert, and Nicolas Veyrat-Charvillon. How to certify the leakage of a chip? In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, volume 8441 of *Lecture Notes in Computer Science*, pages 459–476. Springer, 2014.
6. Weihao Gao, Sreeram Kannan, Sewoong Oh, and Pramod Viswanath. Estimating mutual information for discrete-continuous mixtures. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5988–5999, Red Hook, NY, USA, 2017. Curran Associates Inc.
7. Vincent Grosso and François-Xavier Standaert. Masking proofs are tight and how to exploit it in security evaluations. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology - EUROCRYPT 2018*, volume 10821, pages 385–412. Springer, 2018.
8. László Györfi and Edward C. van der Meulen. Density-free convergence properties of various estimators of entropy. *Computational Statistics and Data Analysis*, 5(4):425–436, 1987.
9. Peter Hall and Sally Morton. On the estimation of entropy. *Annals of the Institute of Statistical Mathematics*, 45:69–88, 02 1993.
10. Annelie Heuser, Olivier Rioul, and Sylvain Guilley. Good is not good enough - deriving optimal distinguishers from communication theory. In Lejla Batina and Matthew Robshaw, editors, *Cryptographic Hardware and Embedded Systems - CHES 2014 - 16th International Workshop, Busan, South Korea, September 23-26, 2014. Proceedings*, volume 8731 of *Lecture Notes in Computer Science*, pages 55–74. Springer, 2014.

11. Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Phys Rev E Stat Nonlin Soft Matter Phys*, 69:pp. 066138, 07 2004.
12. Liam Paninski. Estimation of Entropy and Mutual Information. *Neural Computation*, 15(6):1191–1253, 2003.
13. Emmanuel Prouff and Matthieu Rivain. Theoretical and practical aspects of mutual information-based side channel analysis. *Int. J. Appl. Cryptogr.*, 2(2):121–138, 2010.
14. Alfréd Rényi. On the dimension and entropy of probability distributions. *Acta Mathematica Academiae Scientiarum Hungarica*, 10:193–215, 1959.
15. François-Xavier Standaert, Tal Malkin, and Moti Yung. A unified framework for the analysis of side-channel key recovery attacks. In Antoine Joux, editor, *Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings*, volume 5479 of *Lecture Notes in Computer Science*, pages 443–461. Springer, 2009.