Bridging the Gap: Advanced Tools for Side-Channel Leakage Estimation beyond Gaussian Templates and Histograms

Tobias Schneider[1], Amir Moradi[1], François-Xavier Standaert[2], and Tim Güneysu[3]

[1] Horst Görtz Institute for IT Security, Ruhr-Universität Bochum, Germany
[2] ICTEAM/ELEN/Crypto Group, Université catholique de Louvain, Belgium
[3] University of Bremen and DFKI, Germany

Abstract. The accuracy and the fast convergence of a leakage model are both essential components for the efficiency of side-channel analysis. Thus for efficient leakage estimation an evaluator is requested to pick a Probability Density Function (PDF) that constitutes the optimal trade-off between both aspects. In the case of parametric estimation, Gaussian templates are a common choice due to their fast convergence, given that the actual leakages follow a Gaussian distribution (as in the case of an unprotected device). In contrast, histograms and kernel-based estimations are examples for non-parametric estimation that are capable to capture any distribution (even that of a protected device) at a slower convergence rate.
With this work we aim to enlarge the statistical toolbox of a side-channel evaluator by introducing new PDF estimation tools that fill the gap between both extremes. Our tools are designed for parametric estimation and can efficiently characterize leakages up to the fourth statistical moment. We show that such an approach is superior to non-parametric estimators in contexts where key-dependent information in located in one of those moments of the leakage distribution. Furthermore, we successfully demonstrate how to apply our tools for the (worst-case) information-theoretic evaluation on masked implementations with up to four shares, both in a profiled and non-profiled attack scenario. We like to remark that this flexibility capturing information from different moments of the leakage PDF can provide very valuable feedback for hardware designers to their task to evaluate the individual and combined criticality of leakages in their (protected) implementations.

1 Introduction

Physical attacks are known to pose a major threat to the cryptographic components and security services in many embedded devices. An attacker obtaining side-channel leakages such as the power consumption or electromagnetic emissions from a cryptographic implementation can extract the secret cryptographic key by applying suitable statistical tools on the collected data. A number of

reports have demonstrated that such attacks are not just a theoretical concern but that also real-world devices can be compromised [19, 30, 40, 56]. As a consequence, the seminal Differential Power Analysis (DPA) paper by Kocher *et al.* [23] has been followed by a vast literature on solutions for a wide range of contexts to mitigate these attacks. For example, the inclusion of random delays [11], or shuffling [54] are a frequently used heuristic to improve the physical protection of software implementations. In contrast to this, re-keying strategies, formalized under the name of leakage-resilient cryptography, provide theoretical tools that enable reducing the security of multiple iterations to a single one (cf. [18] for an early result and [52] for a recent one). In this context, one of the most investigated and best understood protection against side-channel attacks is masking [8, 14, 44] that bridges theory and practice. Its underlying principle is to encode any sensitive variable in an implementation into d shares, and to perform the computations on these shares only. Given that the leakage of all the shares is independent and that the measurements are sufficiently noisy, it ensures that the smallest key-dependent (mixed) moment in the leakage distribution is d. Therefore, any adversary trying to extract information from a masked implementation should (ideally) estimate this (mixed) higher-order moment, a task of which the complexity increases exponentially in d.

A drawback with all these solutions is the significant performance overhead. As a result, the development of methodologies enabling a fair assessment of their security level has evolved in parallel with the development of countermeasures so that designers can discuss security and performance implications for their implementations on a sound basis [51]. Since side-channel analysis is essentially based on the comparison of key-dependent leakage models with actual measurements, these methodological developments have led to a central division between *profiled* and *non-profiled* evaluation tools and attacks [55]. In the first case, the adversary/evaluator is allowed to build an accurate (yet not perfect [17]) model for his target device that generally corresponds to an estimation of the leakage Probability Density Function (PDF)[4]. As depicted in the upper left part of Figure 1, Gaussian Template Attacks (TA) are the most common tool for this purpose [9]. In this (here: exhaustive) approach, one builds a Gaussian model for the leakage of every target intermediate value in the implementation. The main limitation of Gaussian templates is that they are bound to the analysis of the first two moments in a leakage distribution (i.e., unprotected implementations and masking with $d = 2$). According to the state-of-the-art, the canonical way to analyze higher-order masked implementations would be to switch to non-parametric PDF estimation, e.g., based on histograms and kernels. But this comes at the cost of two important drawbacks. First, these tools imply a more complex (hence measurement intensive) estimation problem. Second, they estimate all the statistical moments at once, meaning that one loses the detailed intuition that could be obtained from the separate examination of all moments. Alternatively, one could use the Moments-Correlating Profiled DPA (MCP-DPA)

[4] Profiled attacks can also be referred to when the adversary possesses a device with a biased randomness source (as masks).

Fig. 1. Summary of side-channel evaluation tools and attacks.

introduced in [33] that suffers from the complementary drawback. Namely, since MCP-DPA is essentially a "per moment" approach, the intuitions extracted now only correspond to moment taken separately, and it is unclear how one could extend these attacks towards the joint exploitation of multiple moments at the same time.

A comprehensive understanding of how the information leakage of a masked cryptographic implementation is spread among different statistical moments is essential to interpret the results of its security evaluation. That is, in general a $(d-1)$th-order secure implementation is defined as an implementation for which the smallest key-dependent moment in the leakage distribution is d, and this is ideally expected to occur for d shares. But in practice, it frequently happens that glitches (i.e., non-independent leakages) contradict this expectation, leading to informative moments of smaller orders than d, both in hardware and software case studies [10, 28]. Significant research efforts have been dedicated to the design of glitch-free implementations, e.g., based on multiparty computation [45] or threshold implementations [32, 34]. However, in the latter case the number of shares is larger than the claimed order. This, however, highlights the demand for the ability to determine the exact moment that actually leaks [3]. Simple leakage detection tests (e.g., t-test [47]) can be used for this, however they provide only limited information and merely show the existence of leakage (for a more detailed discussion of the limitations of t-test based leakage detection see [16]). Eventually, the recent results in [15] showed that by quantifying the informativeness of each statistical moment in a side-channel attack, one can extrapolate the

security level of an implementation in function of the noise in its measurements (i.e., a parameter that is typically easier to adapt for HW engineers).

Contribution. Based on this state-of-the art, our contribution is threefold.

First, we extend the evaluation toolbox for profiled side-channel analysis with three new PDF estimation tools, based on Exponentially Modified Gaussian (EMG) distributions, Pearson distribution system and Shifted Generalized Lognormal (SGL) distributions. As illustrated in the upper left part of Figure 1, they allow characterizing statistical moments up to the fourth one, which captures all most relevant masked implementations published so far.

Second, we show that these tools enable the computation of the information leakage in each statistical moment of a leakage distribution (up to the fourth one). We further illustrate that based on such computations, we can design efficient attacks that are able to exploit the information in all the leaking moments jointly, and that the efficiency of these attacks is proportional to the sum of the information provided by each moment.

Eventually, we observe that our tools also have applications in the context of non-profiled side-channel analysis, where the adversary assumes some a-priori model for his target implementation (e.g., typically Hamming weights, Hamming distances). In this context as well, one can divide existing solutions between "per moment" and "PDF-based" distinguishers (see the middle right part of Figure 1). Usual representatives of the first category include Correlation Power Analysis (CPA) [6] or its equivalents [27] for first-order moments, and higher-order DPA [39], Correlation-Enhanced Power Analysis Collision Attacks (CEPACA) [29] or Moments Correlating Collision-DPA (MCC-DPA) [33] for higher-order moments. The most common representative of the second category is Mutual Information Analysis (MIA) [21], which usually relies on (non-parametric) histograms or kernels [2], although any PDF estimation tool is in principle eligible[5]. We show that MIA based on the previously mentioned PDF estimation tools (EMG, Pearson, SGL) leads to interesting efficiency tradeoffs for implementations leaking in moments up to four.

The combination of these tools and methods are valuable inputs for the evaluation of the masking countermeasure, since they allow a more accurate understanding of its implementation weaknesses due to glitches (or any other physical default). Furthermore, they are not limited to analysis techniques and also lead to new attacks exploiting a (practically relevant) combination of moments. Eventually, we remark that our results raise relevant questions regarding the so-called simplifying distinguishers in the bottom of Figure 1. In this context, the adversary/evaluator does not build a model for every target intermediate value but for a combination of them (or of their bits). All the published simplifying distinguishers (e.g., linear regression in the profiled case [46], its on-the-fly extension [13] and stepwise regression [55] in the non-profiled case) mix a "per moment" approach [12] with simple (typically Gaussian) PDF estimations.

[5] Such as cumulants which are used in [24] to estimate the mutual information.

Hence, finding whether one could combine a simplifying distinguisher (that provides useful intuitions regarding the parts of the computations that leak more) with more complex PDF estimation tools as in this paper (that provide similarly useful intuitions regarding which moments are leaking) remains an interesting open problem.

2 Background

Generally, *density estimation* – as a well-studied field in statistics – refers to two major categories, namely non-parametric and parametric methods. Histograms and kernels are amongst the well-known non-parametric ones, which do not make any assumptions about the form of the distribution and use only the sampled data to estimate the distribution. A more detailed description of the two methods is provided in Appendix A. By contrast, Gaussian density estimation, which is the most popular parametric PDF estimator, assumes a symmetric form for the distribution, and characterizes it based on its (sample) mean and standard deviation only. As mentioned in the introduction, our focus in this paper is side-channel evaluation, which is commonly based on PDF estimation for building the leakage models. In this section, we shortly recall some frequently-applied PDF estimation techniques in the field of side-channel analysis. We only consider a univariate scenario, which is motivated by our experimental case study in Section 5, that is based on a threshold implementation in which all the shares are manipulated in parallel.

Notations. The parametric PDF estimators make use of statistical moments that we specify as follows. Let X be a (univariate) random variable. The dth-order raw statistical moments are defined as $\mathsf{E}(X^d)$, with $\mu = \mathsf{E}(X)$ the mean of the distribution and $\mathsf{E}(.)$ the expectation operator. The dth-order central moments are defined as $\mathsf{E}\left((X-\mu)^d\right)$, with $\sigma^2 = \mathsf{E}\left((X-\mu)^2\right)$ the variance of the distribution. The dth-order standardized moments are defined as $\mathsf{E}\left(\left(\frac{X-\mu}{\sigma}\right)^d\right)$, with $\gamma_1 = \mathsf{E}\left(\left(\frac{X-\mu}{\sigma}\right)^3\right)$ the skewness (a measure of the *asymmetry* of the distribution, also known as the first shape parameter), and $\beta_2 = \mathsf{E}\left(\left(\frac{X-\mu}{\sigma}\right)^4\right)$ the kurtosis (a measure of the *peakedness* of the distribution, also known as the second shape parameter). It is noteworthy that the central and standardized moments can be also derived from the raw moments. The corresponding expressions are given in Appendix B. Unless otherwise stated, for simplicity we denote first raw, second central, third (and fourth) standardized moments by first, second, third (and fourth) moments respectively.

Gaussian Density Estimation. In this case, it is assumed that the leakages follow a Gaussian (normal) distribution, and the PDF is given by:

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}},$$

with μ and σ the estimated mean and standard deviation of the samples. Since a Gaussian distribution considers only the first two moments, it generally leads to a more efficient estimation compared to the non-parametric histograms or kernels (as long as the actual distribution is close enough to a Gaussian one). In other words, if the higher ($>$ 2nd) statistical moments of the underlying distribution of the samples are negligible, Gaussian density estimation is going to be extremely efficient. Gaussian Templates and regression-based models are part of the widely-used tools exploiting such an assumption [17].

Gaussian Mixtures. We mention that yet another approach to PDF estimation for masked implementations would be to consider mixture distributions. As demonstrated in [53], this solution is especially efficient when the profiling phase assumes the knowledge of the shares. By contrast, it becomes heuristic – since based on the Expectation Maximization (EM) algorithm – if they are not [25], which will be our running scenario in this work. In particular, we will consider contexts where the different modes of the mixture distributions are well interleaved (i.e. when the noise is large enough for masking to enforce good security guarantees), which makes the EM algorithm hard(er) to apply and stands in contrast with contexts where the modes can be trivially identified by the adversary (for example see [31]). That is, our goal is to investigate simple(r) tools that apply to masking when it delivers its promises and are guaranteed to converge without any need to guess about the number of shares in the target device.

3 New Proposals

We now describe three alternative parametric distributions that can cover moments up to the fourth one. We discuss their advantages as well as the challenges one may face to set the parameters to use them.

3.1 Exponentially Modified Gaussian

Since the Gaussian distribution is symmetric, its skewness is always zero. The exponentially Modified Gaussian (EMG) is another parametric distribution which additionally includes this first shape parameter. The PDF of such a distribution, that covers the first three moments, is defined by [22]:

$$F(x) = \frac{\lambda_3}{2} e^{\frac{\lambda_3}{2}(2\lambda_1+\lambda_3\lambda_2^2-2x)} erfc\left(\frac{\lambda_1+\lambda_3\lambda_2^2-x}{\sqrt{2}\lambda_2}\right), \quad (1)$$

where λ_1, λ_2, λ_3 are the parameters of the distribution and $erfc(.)$ refers to the complementary error function defined as:

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2}\, dt.$$

By means of the sample mean μ, standard deviation σ and skewness γ_1 of the data, these three parameters can be estimated as:

$$\lambda_1 = \mu - \sigma \left(\frac{\gamma_1}{2}\right)^{1/3}, \quad \lambda_2^2 = \sigma^2 \left(1 - \left(\frac{\gamma_1}{2}\right)^{2/3}\right), \quad \lambda_3 = \frac{1}{\sigma \left(\frac{\gamma_1}{2}\right)^{1/3}}.$$

It should be noted that EMG does not cover symmetric distributions, i.e., $\gamma_1 = 0$. However, it usually causes no issue in practice (and in particular for side-channel attacks) as the estimated skewness is never exactly zero. Nevertheless, if the underlying skewness is zero, the estimated skewness might be very small. These cases can lead to numerical problems, which can be solved by using libraries for higher precision computations or switching to a distribution which covers zero skewness (Gaussian, Pearson). Besides, note that for a negative skewness $\gamma_1 < 0$, the distribution is parametrized with the absolute value $|\gamma_1|$, and then mirrored around the mean.

3.2 Pearson Distribution System

The Pearson distribution system is a collection of probability distributions that can be parametrized using the first four moments. In total twelve different distributions (cf. [35–37]) are defined in such a way that depending on the estimated moments one type is preferred, and the corresponding PDF estimation technique is applied. In our experiments we noticed that types I, IV and VI (which are presented in detail below) are the only necessary ones. For further descriptions of the other types, the interested reader is referred to the original articles [35–37]. In addition, we provide a brief discussion of the three types in Appendix C.

Cautionary Note. Distribution systems like Pearson's are in general very flexible as they allow characterizing a broad range of combinations of moments. However, they require the estimation of several PDFs, and may face stability problems at the transitions between the different types of distributions (which may occur, e.g., by increasing the number of side-channel samples). Hence, in these cases, it is preferable to rely on a single distribution.

3.3 Shifted Generalized Lognormal

In [26], Low introduced the Shifted Generalized Lognormal distribution (SGL). It can be parametrized with the first four moments and covers a large interval of possible combinations of skewness and kurtosis. Both of these properties are desirable in side-channel evaluations, and therefore this distribution can be

an interesting alternative to the Pearson's distribution system. The realm covered by the SGL is vast and we found it to be sufficient for all our practical experiments. This is illustrated by the plot of the distributions coverage given in Appendix D (which is similar to the aforementioned one given for Pearson's distribution system).

Concretely, the PDF of the SGL is given by:

$$F(x) = \frac{1}{2\lambda_3^{1/\lambda_3}\lambda_4\Gamma(1+1/\lambda_3)(x-\lambda_1)} e^{-\frac{1}{\lambda_3\lambda_4^{\lambda_3}}\left|ln\left(\frac{x-\lambda_1}{\lambda_2}\right)\right|^{\lambda_3}}, \tag{2}$$

for $\lambda_1 < x < \infty$, where λ_1, λ_2, λ_3, and λ_4 are the distribution parameters and $\Gamma(.)$ denotes the gamma function. These parameters can be estimated using the first four moments. For conciseness, we only give a brief overview of the resulting estimation problem in Appendix E, and refer the interested readers to [26].

3.4 Computational Complexity

The presented parametric methods have all different PDFs with different computation complexities. For SGL, the computation of the parameters from the first four moments takes considerably longer than for all other discussed distributions. To present some intuitions on the run time of the different PDFs, we performed experiments using 100 randomly generated sets of moments and run each PDF[6] 100 times for each of these sets. Then we computed the average over all 1000 executions of each PDF. The Gaussian distribution is used as a reference value and has an average of 0.0034 s on an Intel i5-4200M CPU. The averages increase with the number of moments considered in the distribution: 0.0082 s (EMG), 0.029 s (Pearson), 1.70 s (SGL).

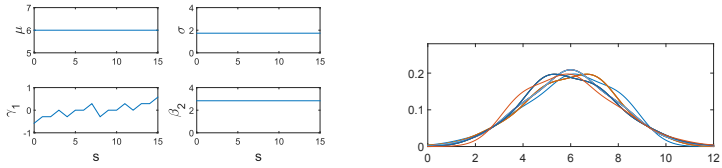
4 Simulated Experiments

In order to better understand the interest of the tools proposed in Section 3 in the context of side-channel analysis, we present a couple of simulated experiments. In the following we use mathematically-generated leakages derived from:

$$l = \mathrm{HW}(s \oplus c_1 \oplus c_2) + \mathrm{HW}(c_1) + \mathrm{HW}(c_2), \tag{3}$$

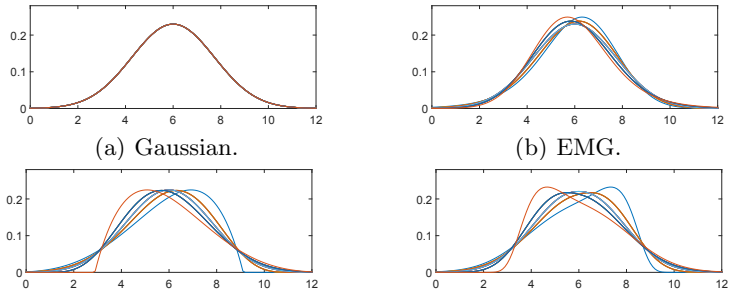
where HW(.) denotes the Hamming weight function, s a sensitive (secret) 4-bit variable, and c_1 and c_2 uniformly distributed random masks in $\{0,1\}^4$. Note that this example is related to any nibble-oriented cipher, e.g., PRESENT [4], and the basic evaluation procedure presented in this paper does not change for larger bit sizes. The only adjustment is the number of possible different classifications, i.e., 2^n instead of 2^4 for n-bit variables. In this simulation it is supposed that the

[6] We implemented three distributions in MATLAB and used the publicly available pearspdf [5].



(a) Estimated moments. (b) Kernel-based PDF.

Fig. 2. The estimated moments for each possible $s \in \{0,1\}^4$ (a) and kernel-estimated PDFs (b) for mathematically-generated leakages corresponding to a 2nd-order masking.



(a) Gaussian. (b) EMG.

(c) Pearson. (d) SGL.

Fig. 3. The estimated PDFs for mathematically-generated leakages corresponding to a 2nd-order masking, obtained with various parametric tools from Sections 2 and 3.

target is a hardware design where the shares are processed at the same time. This scenario essentially emulates a second-order Boolean masking scheme, where we only focus on the encoding of a single variable s in a noise-free situation. In this context, the first and second moments of the leakage distribution are expected to be independent of s. For each $s \in \{0,1\}^4$, we estimate the PDF using both non-parametric (kernels) and parametric (Gaussian, EMG, Pearson, SGL) tools. The first four moments for each s, plotted in Figure 2(a), reveal that there is indeed no dependency between s and the first two moments (i.e., they remain constant for all s). Hence, the only way that s can be distinguished is by observing the third moment. Since kernel-based density estimation considers all possible moments, it can be used to distinguish s as shown in Figure 2(b).

By contrast, the third moment is not used to parametrize the Gaussian distribution and thus each s results in the same distribution in this case (as per Figure 3(a)). This example shows why Gaussian density estimation cannot be used to analyze the leakages that reside in an order higher than two. Eventually, our newly proposed estimators consider moments up to the fourth one, and therefore they can be used to quantify the information leakage of our simulated masking experiment (this can be seen in the remaining part of Figure 3).

5 Practical Case Studies

To examine the application and efficiency of the above-mentioned solutions, we consider a threshold implementation of the PRESENT cipher [4] on an FPGA platform. More precisely, the target design is the *Profile 2* presented in [38] that follows a serialized architecture, i.e., using one instance of the S-box for the whole SLayer. Such a masked hardware implementation has been selected for the practical investigations due to its second- and third-order univariate leakages which allow us to examine our proposed tools. If we would have no leakage at order three and higher, examining the difference between our tools and Gaussian would not be possible.

In the target implementation, the data state is represented by $d = 3$ Boolean shares, and the SLayer is based on the 2-stage masked S-box described in [34]. In other words, each S-box on a 4-bit data is implemented in a pipeline fashion and needs two clock cycles to be computed. For more details on the design architecture we refer the interested reader to [33] and [38].

The leakage traces are collected from a Xilinx Virtex-II Pro FPGA embedded on SASEBO [1]. The sampling rate was set to 1 GS/s and the target FPGA clock was driven at a frequency of 3 MHz. Figure 12(a) (in Appendix F) shows an exemplary trace covering six clock cycles with respect to the full computation of 5 S-boxes on 5 key-whitened plaintext nibbles.

We collected $100,000,000$ traces to be used in our experiments. During the measurements, the PRNG that provides random data (masks) for the sharing of the plaintext was kept active. We also examined and confirmed the uniform distribution of the masks.

A former analysis of MCP-DPA by Moradi and Standaert in [33] on the same implementation revealed that the first pipeline stage of the target S-box exhibits the most informative leakages. The result of such an analysis is given in Appendix F for completeness (see the lower part of Figure 12). It confirms that no first-order leakage can be exploited from this implementation, whereas the second and third moments are indeed informative. It also suggests that second-order leakages are more informative than third ones. By contrast, and as exhaustively discussed in the introduction, two important questions remain open. First, can we quantify the informativeness of the different moments on a (somewhat) more formal basis? Second, and given that more than a single moment provides information, can we design an attack that jointly exploits these moments? (which is in contrast with MCP-DPA that only exploits moments one by one).

Both questions can be answered in the affirmative by the following discussion. In order to make our results comparable with [33], we focus on the same parts of the leakage traces. Namely, we analyze the most informative clock cycle in the S-box execution that corresponds to samples between 13.3 μs and 13.6 μs in Figure 12(a). Based on this case study, we show that the newly introduced PDF estimation tools are powerful ingredients for the information theoretic analysis of a threshold implementation. First, they are able to extract an amount of information from the traces comparable to a kernel density estimation. Second,

they are useful to estimate the informativeness of each moment, and to perform attacks based on the best combination of moments carrying significant information. Eventually, they can naturally and efficiently be embedded in PDF-based non-profiled attacks such as MIA.

5.1 Profiled Evaluations and Attacks

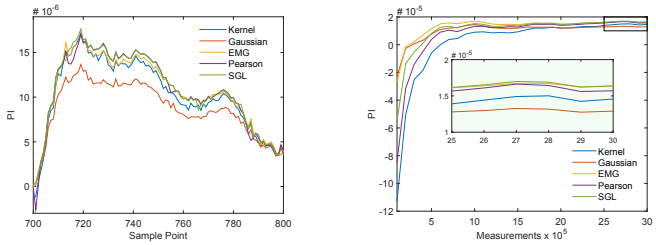
First, we examine the information leakage of the target device using an information theoretic approach. The idea to use Mutual Information (MI) as an evaluation metric was introduced in [51]. It was later refined in [41] to incorporate the fact that the leakage distribution is only estimated, which can potentially bias the estimation of the MI. The so-called Perceived Information (PI) is used to reflect this bias and can be computed as:

$$\hat{PI}(S;L) = H[S] - \sum_{s \in \mathcal{S}} Pr[s] \sum_{l \in \mathcal{L}} Pr_{\mathsf{chip}}[l|s] \cdot log_2 \hat{Pr}_{\mathsf{model}}[s|l], \tag{4}$$

where Pr_{chip} denotes the chip's true distribution (which is unknown but can be sampled) and $\hat{Pr}_{\mathsf{model}}$ refers to the adversary's estimated model (for which we have an analytical formula). Computing the PI essentially requires an estimated $\hat{Pr}_{\mathsf{model}}$, which is exactly what our PDF estimation tools provide. In our experiments, we followed the procedure presented in [17] for computing this metric. In particular, we used 10-fold cross-validation and report the mean of the resulting PI estimates. We start by looking at the information extracted using all the moments enabled by each PDF estimation tool. We then analyze (subsets of) these moments separately.

Combined Moments. In order to compare our proposed solutions (EMG, Pearson, SGL) with the established ones (kernels, Gaussian), we first compute the PI using all the covered moments. We estimate $\hat{Pr}_{\mathsf{model}}$ using the different estimators and compare the results. As previously mentioned, this experiment only covers 100 sample points corresponding to the power peak of the targeted clock cycle, i.e., between 13.3 μs and 13.4 μs in Figure 12(a). The 100,000,000 traces are divided into 10 sets. For each of the 10 runs we use one of these 10 sets (each with 10,000,000 measurements) as samples of the chip's true distributions, and the remaining 9 sets (90,000,000 measurements) to estimate the model distribution ($\hat{Pr}_{\mathsf{model}}$). Figure 4(a) contains the results.

At the first glance, it can be observed that the achieved PI using the Gaussian distribution to estimate $\hat{Pr}_{\mathsf{model}}$ is the lowest. This implies that not all available information is contained in the first two moments (that are the only ones captured by a Gaussian distribution). More interestingly, kernel-based density estimation is non-parametric and therefore is expected to provide the highest PI if its bandwidth is well adapted and enough samples are available. Yet, we observe that this is not exactly the case in our experiments. As depicted in Figure 4(b) (where we focus on the most informative sample 719), this is most likely due to an estimation issue (i.e., a lack of samples). As expected, the non-parametric



(a) 100 sample points of the power peak.

(b) At sample point 719.

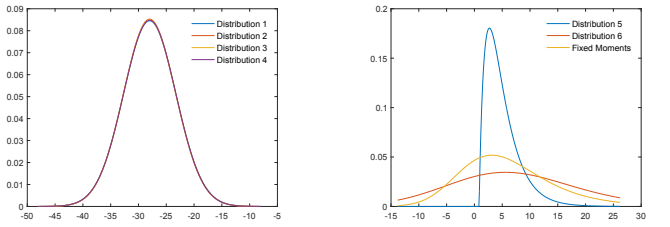
Fig. 4. Kernel-, Gaussian-, EMG-, Pearson- and SGL-based PI estimation with all covered moments (a) using 100,000,000 meas., (b) over the number of meas.

kernel density estimation is the slowest to converge in this case. This suggests an interesting feature of our new parametric tools. Namely, whereas Gaussian estimation is very fast but limited to the exploitation of two moments (hence leads to less efficient attacks, as will be discussed next), EMG-, Pearson- and SGL-based estimations combine a faster convergence than kernels with a similar informativeness.

Summarizing, we can conclude that PDFs covering the right combination of moments lead to the best tradeoff between a fast convergence towards a well estimated model, and a well-informative model once properly estimated (i.e., a model for which the PI should be close to the MI [17]). By contrast, the previous results do not allow to deduce about the relative informativeness of each moment (which could possibly be used to further speed up the model estimation and attacks), which motivates the following analysis.

Separate Moments. An interesting property of the parametric estimators is the ability to consider only selected moments instead of trying to characterize any possible moment (as in non-parametric estimations). Using the Gaussian distribution as an example, we can compute the information contained exclusively in the first two moments, as this distribution only considers the mean and variance. Similarly, it is also possible to compute the PI for the first three moments (with EMG distributions) and the first four moments (with Pearson's distribution system and SGL distributions). In the following, we present an approach that enables us to compute the PI both for each moment taken separately and for any combination of those.

For this purpose, and taking the case where we focus on a single moment, we simply have to set all but one of the moments to a fixed value. For example, suppose that we want to consider the information contained in the first moments of a Gaussian distribution only. We achieve this by considering a Gaussian model where the means are estimated as in the previous section, but the variances are set to a fixed value, which essentially removes any secret-dependent information they could carry from the templates through the second moments. Since chang-



(a) At point 719 (cf. Table 1). (b) Simulations (cf. Table 2).

Fig. 5. The PDFs of the six distributions from Table 1 and 2.

ing the variances affects the shape of the distributions, the fixed value can be chosen as the average of the variances (over the 16 templates) to minimize the distance between the original distributions and the ones with a fixed variance[7]. A similar technique actually works for any of our parametric estimators, and for any (combination of) moments.

As an illustration, let us first recall the influence of the first four statistical moments on the shape of the resulting distribution. The third moment, called skewness, measures the asymmetry of the distribution. Therefore, distributions with positive skewness tend to left while distributions with a negative skewness tend to the right. The fourth moment, called kurtosis, measures the "peakedness" (sharpness) of the distribution. As a consequence, the higher the kurtosis, the sharper is the distribution. As an illustration, Figure 9 in Appendix F depicts different distributions with varying third and fourth moments. Note that we consider only the first four moments in our analysis, hence we omitted definitions for moments of any further orders.

When we set specific higher-order moments (as in our approach) to specific values, we actually fix the width of the distributions (i.e., variance), or their right-left tendency (i.e., skewness) or their sharpness (i.e., kurtosis). Hence, information sitting in the corresponding moments does not contribute in the information-theoretic-based evaluation, e.g., mutual information. We like to emphasize that the estimated higher-order moments in real side-channel measurements (categorized, for example, based on the processed data) are very slightly different. Consider for example the PDFs of four exemplary distributions shown in Figure 5(a), taken from the most leaking point of the measurements of our case study (see Figure 4(a)). The first four moments of each distribution are given in Table 1. All moments of the four distributions are very similar to each other, e.g., the skewness of all these four distributions is only slightly different. Hence, fixing the skewness of all of them to a specific value (e.g., the average of all skewnesses given by 0.0064627) does not significantly change the shape of the distributions.

[7] Instead, one can also consider the variance of whole trace set. Here we need only a fixed value which is not too different from the variance of each template. Such an approach is not valid in case of Gaussian mixtures as stated in Section 2.

Table 1. The first four statistical moments of four distributions at sample point 719.

	Dist. 1	Dist. 2	Dist. 3	Dist. 4	Average
Mean	-27.9734310	-27.9811494	-27.9827913	-27.9782609	-27.9789082
Variance	22.3624316	21.9979663	22.2165081	22.2660171	22.2107308
Skewness	0.0075083	0.0053184	0.0131009	-0.0000767	0.0064627
Kurtosis	3.0177549	3.0202503	3.0219293	3.0183596	3.0195735

Table 2. The first four statistical moments of two simulated distributions.

	Dist. 5	Dist. 6	Average
Mean	4.9997939	7.400773	6.2002834
Variance	10.0032941	149.017440	79.5103671
Skewness	1.7063003	0.377136	1.0417184
Kurtosis	7.8417563	3.648649	5.7452030

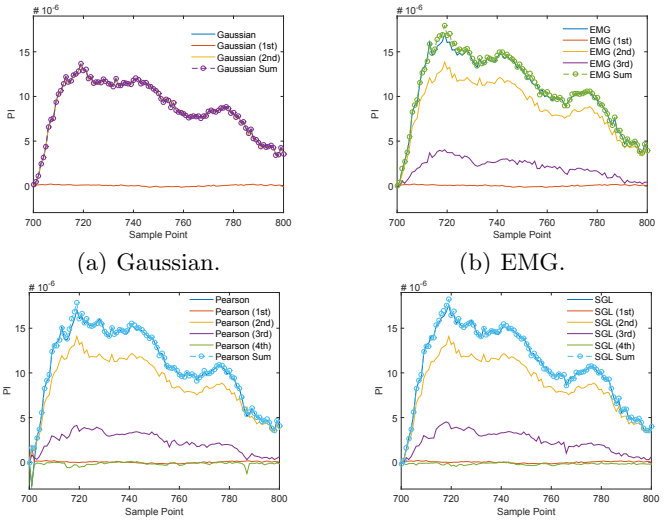
Here we consider four different cases:

1. All moments except the first are fixed to their average (evaluation through means).
2. All moments except the second are fixed to their average (evaluation through variances).
3. All moments except the third are fixed to their average (evaluation through skewnesses)
4. All moments except the fourth are fixed to their average (evaluation through kurtoses).

For each case, the shape of the resulting distributions is very close to the original shape in Figure 5(a). The resulting PDFs of the modified distributions for each case is provided in Appendix F.

It should be noted that in case of simulated data with significantly different moments for each distribution the resulting shapes of each distribution would be also dramatically different to each other. Therefore in this case, setting the corresponding moments to a fixed (average) value does not make the distributions to roughly follow the same shape. If such a huge difference between the moments of the (categorized) distributions exists in practice by any (rare) chance, the corresponding implementation is significantly vulnerable to certain attacks. Obviously, this makes the necessity of performing per-moment evaluations questionable. As an example, we show in Figure 5(b) two simulated distributions formed by the moments from Table 2. It is obvious that the shape of the distribution with fixed moments is considerably different than that of the original two distributions. In this case, a per-moment approach would not be easily possible with an information-theoretic evaluation tool.

We analyze this moment-based investigation based on the same case study as for the previous information theoretic analysis. Hence, we repeat the previous experiments (of Figure 4(a)) with the same parametric estimators (Gaussian, EMG, Pearson, SGL), but this time we consider each possible moment separately. The results are depicted in Figure 6 where the PI curves are categorized



(a) Gaussian. (b) EMG.

(c) Pearson. (d) SGL.

Fig. 6. PI estimates for the separate moments.

based on the employed estimator. Each part of the figure contains the PI curves obtained for each moment separately. For example, in Figure 6(a) the curve labeled *Gaussian (1st)* shows the PI achieved for the first moments (and the curve *Gaussian (2nd)* depicts the same for the second moments, etc.). Further, we included the PI curve of the combined moments (taken from Figure 4(a)) and the sum of the PI curves of the separate moments (e.g., *Gaussian Sum* as the sum of the PI curves of *Gaussian (1st)* and *Gaussian (2nd)*).

As expected, the first moment does not contain any exploitable information as the implementation is first-order secure. It is also noticeable that the chosen estimator does not affect the PI for the first moment. The second moment leads to the highest PI, and therefore is the most informative moment. As similarly indicated by MCP-DPA, the third moment is informative but not as much as the second one. Furthermore, using two estimators (Pearson, SGL) that also cover the fourth moment, we are not able to detect any significant information leakage in the fourth moment. Therefore, a combination of the second and third moments should suffice to capture most of the available information in the underlying measurements.

Most interestingly, we observe that the sum of the PI values obtained for the separate moments is actually close to the PI estimated with the combined moments. Although informal, this observation is particularly interesting in view of the recent results by Duc *et al.* in [15] where the PI values are connected with the success rate of a (worst-case) template attack using the same model. Indeed, since the sum of the PI values obtained per moment is essentially the same as the PI value obtained with the non-parametric kernel method, it implies that *in*

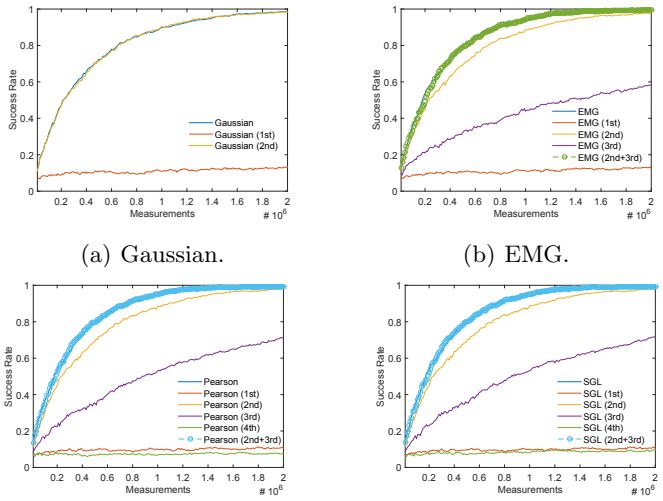
our case study, the separation between moments did not lead to any significant information loss. This suggests that a (simple and intuitive) moment-based side-channel evaluation could be well-founded, at least in certain contexts that would be worth formalizing. And very concretely, it also means that an attack exploiting out two informative (i.e., second and third) moments will be close to optimal in our case.

Profiled attacks. The results in [15] prove that (under sufficiently noisy leakages) the success rate of a profiled template attack is inversely proportional to the PI value estimated with the same model. In view of the previous discussions, it means that our proposed estimation tools (EMG, Pearson, SGL) should lead to more effective profiled attacks than their counterparts with Gaussian estimation (because of modeling errors) and kernels (because of assumption errors). Furthermore, the attacks exploiting the second moment should lead to a higher success rate than attacks exploiting the other three moments. Eventually, the best attack should exploit the combination of second and third moments. For completeness, we ran experiments to confirm these expectations. We built univariate templates (for the most informative sample point 719) from 90,000,000 measurements and, for each given number of measurements, repeated an attack 1000 times for different measurements (excluding those used for profiling) to compute a subkey recovery success rate. The results of this last experiment are depicted in Figure 7 and are well in line with theoretical predictions. In this respect, the most interesting curves are the ones corresponding to the combination of second and third moments, since they correspond to the best tradeoff between model complexity and attack efficiency, and could not have been reached with existing side-channel evaluation tools. (Additional curves are provided in Appendix F, Figure 11(a), including attacks exploiting kernel-based models that are as efficient, but as mentioned earlier, more expensive to estimate.)

5.2 Non-Profiled Attacks

In addition, we briefly discuss the application of our solutions in the non-profiled attack setting. For this purpose, we consider a univariate MIA, which is the standard representative for non-profiled attacks exploiting PDF estimation. As usual in this context, we cannot directly use a generic (i.e., identity) power model, since it would not be able to extract any key-dependent information [55] if the first encryption round is targeted[8]. Further, MIA needs a non-bijective model to be effective. Besides these constraints, our tools are easily applicable in this scenario. A more detailed discussion with experimental results is provided in Appendix G.

[8] Such an identity model is applicable to e.g., the Sbox output of the second encryption round [43].



(a) Gaussian. (b) EMG.

(c) Pearson. (d) SGL.

Fig. 7. Success rate of several univariate template attacks exploiting separate and combined moments, for the most informative sample point 719 in our traces.

5.3 Selection of Tools

We have discussed multiple parametric tools, each with its own advantages and disadvantages. Compared to the traditional non-parametric tools, they offer a higher flexibility and convergence. Therefore, they should be preferred if the number of samples is too small or a special case (e.g., only two moments) should be evaluated. The PDF of EMG can be computed very efficiently compared SGL and Pearson. However, it considers only the first three moments instead of four. The Pearson distribution system includes the kurtosis and its PDF is still relatively efficient compared to SGL. Nevertheless, it is made up of multiple different distributions which can be problematic in certain cases as pointed out in Section 3.2. Therefore, in scenarios where the computation time of the PDF can be ignored and the leakages are covered by SGL, it is the preferable tool.

However, the computation time is often a limiting factor and it can be significantly reduced in certain cases by choosing a more limited distribution which is still sufficient to capture all relevant leakage. If the type of implementation and leakage is known, this choice is easily possible. Gaussian (resp. EMG) is the preferred choice for leakage which is exclusive in the first two (resp. three) moments due to its very efficient PDF. Leakage in the fourth moment can be also efficiently captured with the Pearson distribution system, assuming that the aforementioned problems do not arise. If the type of masked implementation,i.e., the order of masking, is unknown, then this choice of distribution cannot be that easily made. SGL is then the best approach, if the distribution is inside the plane of existence of SGL. For the separate moments method, it is still possible to re-

duce the computation time by using some of the other distributions (Gaussian, EMG) for the moments of lower order.

6 Conclusions

This paper introduced a variety of PDF estimation tools to improve the evaluation of leaking devices, both in the profiled and non-profiled settings. Their main interest is their flexibility: our proposals can indeed capture information lying in different moments of the leakage PDF. As a result, we can easily analyze masked implementations and extract useful feedback to hardware designers, i.e. in terms of how much information is lying in every moment and how to combine it. This brings a concrete and more founded counterpart the recent evaluations of implementations with non-independent leakages in [15], where this quantity of information "per moment" is required. More generally, our findings provide efficient tradeoffs between the cost of profiling and the efficiency of the resulting attacks, since they allow adversaries and evaluators to build models that are tailored to their implementations. These results naturally open various interesting research challenges for future work. As mentioned in introduction, combining an analysis of moments as in this work with simplifying approaches to leakage modeling (e.g. based on linear regression) would be even more convenient to evaluators. Besides, investigating the "summing rule" of Section 5.1 more formally is certainly worth further efforts as well. Eventually, our current tools are limited to univariate leakages. While this was sufficient to analyze our hardware case study, it naturally suggests the extension to multivariate case studies as yet another important question. This is especially interesting given that even hardware designs with univariate d-order security may include a multivariate vulnerability for which less than d points are combined [42]. A starting point for this purpose would be to exploit some popular "combining" functions from the side-channel literature (which would allow us to exploit our univariate tools directly).

References

1. Side-channel Attack Standard Evaluation Board (SASEBO). Further information are available via `http://satoh.cs.uec.ac.jp/SAKURA/index.html`.
2. Lejla Batina, Benedikt Gierlichs, Emmanuel Prouff, Matthieu Rivain, François-Xavier Standaert, and Nicolas Veyrat-Charvillon. Mutual information analysis: a comprehensive study. *J. Cryptology*, 24(2):269–291, 2011.
3. Begül Bilgin, Benedikt Gierlichs, Svetla Nikova, Ventzislav Nikov, and Vincent Rijmen. A More Efficient AES Threshold Implementation. In *AFRICACRYPT 2014*, volume 8469 of *LNCS*, pages 267–284. Springer, 2014.
4. Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: An Ultra-Lightweight Block Cipher. In *CHES 2007*, volume 4727 of *LNCS*, pages 450–466. Springer, 2007.

5. Pierce Brady. pearspdf. `http://www.mathworks.com/matlabcentral/fileexchange/26516-pearspdf`.
6. Eric Brier, Christophe Clavier, and Francis Olivier. Correlation Power Analysis with a Leakage Model. In *CHES 2004*, volume 3156 of *LNCS*, pages 16–29. Springer, 2004.
7. Mathieu Carbone, Sébastien Tiran, Sébastien Ordas, Michel Agoyan, Yannick Teglia, Gilles R. Ducharme, and Philippe Maurine. On Adaptive Bandwidth Selection for Efficient MIA. In *COSADE 2014*, volume 8622 of *LNCS*, pages 82–97. Springer, 2014.
8. Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, and Pankaj Rohatgi. Towards Sound Approaches to Counteract Power-Analysis Attacks. In *CRYPTO 1999*, volume 1666 of *LNCS*, pages 398–412. Springer, 1999.
9. Suresh Chari, Josyula R. Rao, and Pankaj Rohatgi. Template Attacks. In *CHES 2002*, volume 2523 of *LNCS*, pages 13–28. Springer, 2002.
10. Jean-Sébastien Coron, Christophe Giraud, Emmanuel Prouff, Soline Renner, Matthieu Rivain, and Praveen Kumar Vadnala. Conversion of Security Proofs from One Leakage Model to Another: A New Issue. In *COSADE 2012*, volume 7275 of *LNCS*, pages 69–81. Springer, 2012.
11. Jean-Sébastien Coron and Ilya Kizhvatov. An Efficient Method for Random Delay Generation in Embedded Software. In *CHES 2009*, volume 5747 of *LNCS*, pages 156–170. Springer, 2009.
12. Guillaume Dabosville, Julien Doget, and Emmanuel Prouff. A New Second-Order Side Channel Attack Based on Linear Regression. *IEEE Trans. Computers*, 62(8):1629–1640, 2013.
13. Julien Doget, Emmanuel Prouff, Matthieu Rivain, and François-Xavier Standaert. Univariate side channel attacks and leakage modeling. *J. Cryptographic Engineering*, 1(2):123–144, 2011.
14. Alexandre Duc, Stefan Dziembowski, and Sebastian Faust. Unifying Leakage Models: From Probing Attacks to Noisy Leakage. In *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 423–440. Springer, 2014.
15. Alexandre Duc, Sebastian Faust, and François-Xavier Standaert. Making Masking Security Proofs Concrete - Or How to Evaluate the Security of Any Leaking Device. In *EUROCRYPT 2015*, volume 9056 of *LNCS*, pages 401–429. Springer, 2015.
16. François Durvaux and François-Xavier Standaert. From improved leakage detection to the detection of points of interests in leakage traces. In *EUROCRYPT (1)*, volume 9665 of *Lecture Notes in Computer Science*, pages 240–262. Springer, 2016.
17. François Durvaux, François-Xavier Standaert, and Nicolas Veyrat-Charvillon. How to Certify the Leakage of a Chip? In *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 459–476. Springer, 2014.
18. Stefan Dziembowski and Krzysztof Pietrzak. Leakage-Resilient Cryptography. In *Foundations of Computer Science 2008*, pages 293–302. IEEE Computer Society, 2008.
19. Thomas Eisenbarth, Timo Kasper, Amir Moradi, Christof Paar, Mahmoud Salmasizadeh, and Mohammad T. Manzuri Shalmani. On the Power of Power Analysis in the Real World: A Complete Break of the KeeLoq Code Hopping Scheme. In *CRYPTO 2008*, volume 5157 of *LNCS*, pages 203–220. Springer, 2008.
20. David Freedman and Persi Diaconis. On the histogram as a density estimator:L 2 theory. *Zeitschrift fr Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 57(4):453–476, 1981.
21. Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel. Mutual Information Analysis. In *CHES 2008*, volume 5154 of *LNCS*, pages 426–442. Springer, 2008.

22. Eli Grushka. Characterization of exponentially modified Gaussian peaks in chromatography. *Analytical Chemistry*, 44(11):1733–1738, 1972. PMID: 22324584.
23. Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential Power Analysis. In *CRYPTO 1999*, volume 1666 of *LNCS*, pages 388–397. Springer, 1999.
24. Thanh-Ha Le and Maël Berthier. Mutual Information Analysis under the View of Higher-Order Statistics. In *IWSEC 2010*, volume 6434 of *LNCS*, pages 285–300. Springer, 2010.
25. Kerstin Lemke-Rust and Christof Paar. Gaussian Mixture Models for Higher-Order Side Channel Analysis. In *CHES 2007*, volume 4727 of *LNCS*, pages 14–27. Springer, 2007.
26. Y.M. Low. A new distribution for fitting four moments and its applications to reliability analysis. *Structural Safety*, 42(0):12 – 25, 2013.
27. Stefan Mangard, Elisabeth Oswald, and François-Xavier Standaert. One for all - all for one: unifying standard differential power analysis attacks. *IET Information Security*, 5(2):100–110, 2011.
28. Stefan Mangard, Thomas Popp, and Berndt M. Gammel. Side-Channel Leakage of Masked CMOS Gates. In *CT-RSA 2005*, volume 3376 of *LNCS*, pages 351–365. Springer, 2005.
29. Amir Moradi. Statistical Tools Flavor Side-Channel Collision Attacks. In *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 428–445. Springer, 2012.
30. Amir Moradi, Alessandro Barenghi, Timo Kasper, and Christof Paar. On the vulnerability of FPGA bitstream encryption against power analysis attacks: extracting keys from xilinx Virtex-II FPGAs. In *Computer and Communications Security, CCS 2011*, pages 111–124. ACM, 2011.
31. Amir Moradi, Mario Kirschbaum, Thomas Eisenbarth, and Christof Paar. Masked Dual-Rail Precharge Logic Encounters State-of-the-Art Power Analysis Methods. *IEEE Trans. VLSI Syst.*, 20(9):1578–1589, 2012.
32. Amir Moradi, Axel Poschmann, San Ling, Christof Paar, and Huaxiong Wang. Pushing the Limits: A Very Compact and a Threshold Implementation of AES. In *EUROCRYPT 2011*, volume 6632 of *LNCS*, pages 69–88. Springer, 2011.
33. Amir Moradi and François-Xavier Standaert. Moments-correlating DPA. *IACR Cryptology ePrint Archive*, 2014:409, 2014.
34. Svetla Nikova, Vincent Rijmen, and Martin Schläffer. Secure Hardware Implementation of Nonlinear Functions in the Presence of Glitches. *J. Cryptology*, 24(2):292–321, 2011.
35. Karl Pearson. Contributions to the Mathematical Theory of Evolution. II. Skew Variation in Homogeneous Material. *Royal Society of London Philosophical Transactions Series A*, 186:343–414, 1895.
36. Karl Pearson. Mathematical Contributions to the Theory of Evolution. X. Supplement to a Memoir on Skew Variation. *Royal Society of London Philosophical Transactions Series A*, 197:443–459, 1901.
37. Karl Pearson. Mathematical Contributions to the Theory of Evolution. XIX. Second Supplement to a Memoir on Skew Variation. *Royal Society of London Philosophical Transactions Series A*, 216:429–457, 1916.
38. Axel Poschmann, Amir Moradi, Khoongming Khoo, Chu-Wee Lim, Huaxiong Wang, and San Ling. Side-Channel Resistant Crypto for Less than 2, 300 GE. *J. Cryptology*, 24(2):322–345, 2011.
39. Emmanuel Prouff, Matthieu Rivain, and Régis Bevan. Statistical Analysis of Second Order Differential Power Analysis. *IEEE Trans. Computers*, 58(6):799–811, 2009.

40. Josyula R. Rao, Pankaj Rohatgi, Helmut Scherzer, and Stephane Tinguely. Partitioning Attacks: Or How to Rapidly Clone Some GSM Cards. In *IEEE Symposium on Security and Privacy 2002*, pages 31–41. IEEE Computer Society, 2002.
41. Mathieu Renauld, François-Xavier Standaert, Nicolas Veyrat-Charvillon, Dina Kamel, and Denis Flandre. A Formal Study of Power Variability Issues and Side-Channel Attacks for Nanoscale Devices. In *EUROCRYPT 2011*, volume 6632 of *LNCS*, pages 109–128. Springer, 2011.
42. Oscar Reparaz, Begül Bilgin, Svetla Nikova, Benedikt Gierlichs, and Ingrid Verbauwhede. Consolidating Masking Schemes. In *CRYPTO (1)*, volume 9215 of *Lecture Notes in Computer Science*, pages 764–783. Springer, 2015.
43. Oscar Reparaz, Benedikt Gierlichs, and Ingrid Verbauwhede. Generic DPA Attacks: Curse or Blessing? In *COSADE 2014*, volume 8622 of *LNCS*, pages 98–111. Springer, 2014.
44. Matthieu Rivain and Emmanuel Prouff. Provably Secure Higher-Order Masking of AES. In *CHES 2010*, volume 6225 of *LNCS*, pages 413–427. Springer, 2010.
45. Thomas Roche and Emmanuel Prouff. Higher-order glitch free implementation of the AES using Secure Multi-Party Computation protocols - Extended version. *J. Cryptographic Engineering*, 2(2):111–127, 2012.
46. Werner Schindler, Kerstin Lemke, and Christof Paar. A Stochastic Model for Differential Side Channel Cryptanalysis. In *CHES 2005*, volume 3659 of *LNCS*, pages 30–46. Springer, 2005.
47. Tobias Schneider and Amir Moradi. Leakage Assessment Methodology - A Clear Roadmap for Side-Channel Evaluations. In *CHES*, volume 9293 of *Lecture Notes in Computer Science*, pages 495–513. Springer, 2015.
48. David W. Scott. On Optimal and Data-Based Histograms. *Biometrika*, 66(3):pp. 605–610, 1979.
49. Simon J. Sheather. Density Estimation. *Statist. Sci.*, 19(4):588–597, 11 2004.
50. B. W. Silverman. Density estimation for statistics and data analysis. 1986.
51. François-Xavier Standaert, Tal Malkin, and Moti Yung. A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks. In *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 443–461. Springer, 2009.
52. François-Xavier Standaert, Olivier Pereira, and Yu Yu. Leakage-Resilient Symmetric Cryptography under Empirically Verifiable Assumptions. In *CRYPTO 2013*, volume 8042 of *LNCS*, pages 335–352. Springer, 2013.
53. François-Xavier Standaert, Nicolas Veyrat-Charvillon, Elisabeth Oswald, Benedikt Gierlichs, Marcel Medwed, Markus Kasper, and Stefan Mangard. The World Is Not Enough: Another Look on Second-Order DPA. In *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 112–129. Springer, 2010.
54. Nicolas Veyrat-Charvillon, Marcel Medwed, Stéphanie Kerckhof, and François-Xavier Standaert. Shuffling against Side-Channel Attacks: A Comprehensive Study with Cautionary Note. In *ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 740–757. Springer, 2012.
55. Carolyn Whitnall, Elisabeth Oswald, and François-Xavier Standaert. The Myth of Generic DPA...and the Magic of Learning. In *CT-RSA 2014*, volume 8366 of *LNCS*, pages 183–205. Springer, 2014.
56. Yuanyuan Zhou, Yu Yu, François-Xavier Standaert, and Jean-Jacques Quisquater. On the Need of Physical Security for Small Embedded Devices: A Case Study with COMP128-1 Implementations in SIM Cards. In *Financial Cryptography 2013*, volume 7859 of *LNCS*, pages 230–238. Springer, 2013.

A Non-Parametric Density Estimation

A.1 Histograms

Amongst the most straightforward techniques to estimate a PDF, one can group the sampled data into (commonly equally-sized) bins. The probability for a given input x can then be given by $Pr[x] = \frac{bin(x)}{n}$, where $bin(x)$ returns the number of samples in the bin to which x belongs, and n indicates the total number of samples.

Although histograms are a non-parametric estimator, the width of the bins (respectively the number of bins) is an important parameter that can significantly influence the resulting PDF. For certain distributions (e.g., Gaussian), there are practical guidelines on how to select such parameters (e.g., Scott's rule [48] and Freedman-Diaconis rule [20]). But for distributions that strongly deviate from these assumed forms, the optimal choice of these parameters is unknown.

In our side-channel context, measurements usually correspond to 8-bit data, as the analogue to digital converters which sample the leakages (by means of an oscilloscope) typically have 8-bit effective length. Therefore, the histogram of side-channel leakages can most precisely be estimated with 256 bins. However, by using such a narrow bin width, the number of required samples to fill the bins increases and makes the estimation more data intensive. Hence, the number of bins is commonly selected with respect to the underlying hypothetical model used by the adversary, e.g., 9 in case of Hamming weights for an 8-bit intermediate value [2]. As histograms make no assumptions about the distribution, the side-channel leakages of all moments are encapsulated and can be exploited.

A.2 Kernels

The foundation of kernel-based density estimation is to approximate the PDF with a sum of so-called kernel functions. That is, for each sample point, a kernel function that is centered around this point (l_i) is added to the probability density function. The density for a given input x can then be estimated as:

$$F(x) = \frac{1}{n\,h} \sum_{i=0}^{n-1} K\left(\frac{x - l_i}{h}\right),$$

where h is the bandwidth and $K(.)$ the kernel function. In contrast with histograms, the kernel-based estimation builds a continuous function which can be integrated. This allows for a faster convergence (i.e., with less samples) to the real distribution compared to histograms. For the rest, both methods are similar: they are able to capture any moment of a distribution, but cannot differentiate between them.

Concretely, the kernel function should fulfill the property $\int_{-\infty}^{\infty} K(x)dx = 1$. Although there exist many different proposals for such functions (e.g., Gaussian and Epanechnikov, see [2]), the type of kernel has only little influence on the

resulting PDF [49], and the bandwidth h (also called "smoothing parameter") plays a more important role in the precision of the estimation. A common approach to choose the bandwidth is known as Silverman's rule [50], where h is selected as $c\,\sigma\,n^{-1/5}$, with σ the standard deviation of the samples, and the constant c selected based on the chosen kernel function. Recent results [7] showed that an adaptive procedure (i.e., dynamically altering h) can lead to the best success rates when the PDFs used in a MIA are estimated by a kernel function.

B Central & Standardized Moments from Raw Moments

To compute the parameters for SGL, it is necessary to derive μ, σ, γ_1, and β_2 from the first four raw moments M_1, M_2, M_3, M_4 (as $M_d = \mathsf{E}(X^d)$). This can be done as follows:

$$\mu = M_1, \tag{5}$$

$$\sigma^2 = M_2 - M_1^2, \tag{6}$$

$$\gamma_1 = \frac{M_3 - 3M_1M_2 + 2M_1^3}{(M_2 - M_1^2)^{\frac{3}{2}}} \tag{7}$$

$$\beta_2 = \frac{M_4 - 4M_1M_3 + 6M_1^2M_2 - 3M_1^4}{(M_2 - M_1^2)^2} \tag{8}$$

C Type I, IV and VI of Pearson Distribution System

To determine the type of distribution and find the parameters for the associated PDF, we first define b_0, b_1, b_2 as:

$$b_0 = \frac{\sigma(4\beta_2 - 3\beta_1)}{10\beta_2 - 12\beta_1 - 18}, \quad b_1 = \frac{\sqrt{\sigma}\gamma_1(\beta_2 + 3)}{10\beta_2 - 12\beta_1 - 18}, \quad b_2 = \frac{2\beta_2 - 3\beta_1 - 6}{10\beta_2 - 12\beta_1 - 18},$$

where $\beta_1 = \gamma_1^2$ (squared skewness) and β_2 denotes the kurtosis. Based on the estimated skewness and kurtosis, the most suited type is selected as follows. If $\kappa_2 = \frac{b_1^2}{4b_0b_2} < 0$, type I is chosen. Otherwise, if κ_2 is in the interval $]0, 1[$, type IV is preferred. In the last case ($\kappa_2 > 1$) type VI is used. (The remaining cases where $\kappa_2 = 0$ and $\kappa_2 = 1$ require different types of distribution but, as previously mentioned, were not encountered in our experiments and are therefore omitted in this section). A visual representation of these type of distributions in function of γ_1 and β_2 is given in Appendix D, Figure 8(b).

In order to estimate the type I and VI distributions, it is necessary to find the roots of the quadratic function:

$$f(x) = b_2x^2 + b_1x + b_0, \tag{9}$$

denoted as a_1 and a_2 in the following. The rest of the computations are type specific and briefly described in the following.

Type I. This distribution is a generalization of the beta distribution using four parameters. In this case, Equation (9) has two real roots with different signs. We assume without loss of generality that $a_1 < 0 < a_2$ and define:

$$m_1 = \frac{b_1 + a_1}{b_2(a_2 - a_1)}, \qquad m_2 = \frac{-b_1 - a_2}{b_2(a_2 - a_1)}.$$

The PDF is then defined as:

$$F(x) = \frac{\left(\frac{x-\mu-a_1\sqrt{\sigma}}{(a_2-a_1)\sqrt{\sigma}}\right)^{m_1}\left(1 - \frac{x-\mu-a_1\sqrt{\sigma}}{(a_2-a_1)\sqrt{\sigma}}\right)^{m_2}}{B(m_1+1, m_2+1)(a_2-a_1)\sqrt{\sigma}}, \tag{10}$$

where $B(.,.)$ refers to the beta function. Hence, x is bounded on both sides within $]a_1\sqrt{\sigma}+\mu, a_2\sqrt{\sigma}+\mu[$.

Type IV. In this case, we first compute the four parameters m_1, m_2, m_3, m_4 as:

$$m_1 = \frac{1}{2b_2},\ m_2 = \frac{2b_1(1-m_1)}{\sqrt{4b_0b_2 - b_1^2}},\ m_3 = \sqrt{\frac{(2m_1-2)^3 - (2m_1-2)^2}{(2m_1-2)^2 + m_2^2}},\ m_4 = \frac{m_3 m_2}{2m_1 - 2}.$$

Then the PDF can be estimated by:

$$F(x) = \frac{e^{-arctan(\frac{x-\mu-m_4\sigma}{m_3\sigma})m_2}}{B(m_1-\frac{1}{2},\frac{1}{2})m_3\sigma}\left|\frac{\Gamma(m_1 + \frac{m_2}{2}i)}{\Gamma(m_1)}\right|^2\left(1 + \left(\frac{x-\mu-m_4\sigma}{m_3\sigma}\right)^2\right)^{-m_1}, \tag{11}$$

where $\Gamma(.)$ denotes the gamma function. In contrast to types I and VI, this distribution is unbounded on both sides and supports x in the interval $[-\infty, +\infty]$.

Type VI. This distribution is related to the F distribution. In this case, Equation (9) has two real roots with the same sign, and we assume without loss of generality that $|a_1| \leq |a_2|$. For this distribution, we first compute m_1 and m_2 as:

$$m_1 = \frac{a_1 + b_1}{b_2(a_2 - a_1)}, \qquad m_2 = \frac{a_2 - a_1}{b_2(a_2 - a_1)}.$$

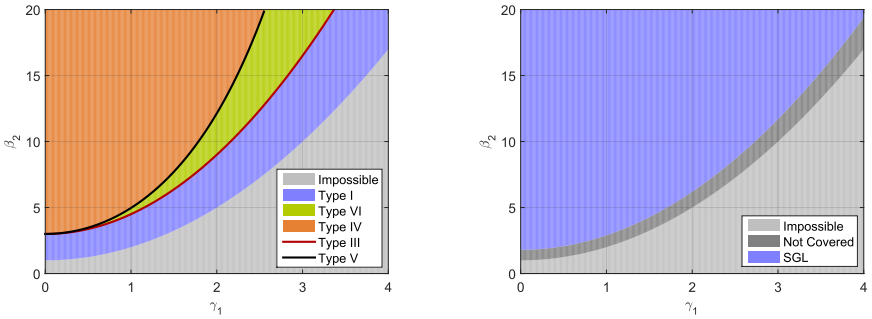
The PDF is then defined as:

$$F(x) = \frac{\left(\frac{x-\mu-a_1\sqrt{\sigma}}{(a_1-a_2)\sqrt{\sigma}}\right)^{m_1}\left(1 + \frac{x-\mu-a_1\sqrt{\sigma}}{(a_1-a_2)\sqrt{\sigma}}\right)^{-(m_1+m_2)}}{B\left(m_1+1, m_2-1\right)|a_1-a_2|\sqrt{\sigma}}. \tag{12}$$

Depending on the sign of the skewness, the covered range for x is either $]a_1\sqrt{\sigma}+\mu, +\infty]$ ($\gamma_1 > 0$) or $[-\infty, a_1\sqrt{\sigma}+\mu[$ ($\gamma_1 < 0$).

D Coverage of Pearson and SGL

In Figure 8(a), the coverage for the different types of Pearson distributions is illustrated. Type I is limited by the impossible region ($\beta_2 \leq \gamma^2 + 1$). Type III covers the border between type I and type VI (i.e., $2\beta_2 = 3\gamma^2 + 6$). Similarly, the border between type VI and type IV is covered by type V ($\gamma_1^2(\beta_2 + 3)^2 = 4(4\beta_2 - 3\gamma_1^2)(2\beta_2 - 3\gamma_1^2 - 6)$). Note that we did not consider these two border cases (type III and type V) in our experiments. Figure 8(b) shows a similar coverage area for SGL distributions. In both cases, the non-covered realm of these PDF estimators is marked in grey to allow straight comparisons.



(a) Pearson's distribution system. (b) SGL.

Fig. 8. Plane of existence of the different distributions.

E Estimation Problem of SGL

At the first step, we introduce a new variable Y defined as $Y = \frac{(X-\lambda_1)}{\lambda_2}$. The raw moments of Y can be computed using (λ_3, λ_4) as:

$$E[Y^k] = \frac{1}{\Gamma(1/\lambda_3)} \sum_{i=0}^{\infty} \frac{(k\lambda_4)^{2i}}{(2i)!} \lambda_3^{2i/\lambda_3} \Gamma\left(\frac{2i+1}{\lambda_3}\right). \tag{13}$$

From these raw moments, the mean μ_Y, variance σ_Y^2, skewness γ_Y, and kurtosis β_Y of Y can be derived (from the definitions given in Appendix B). Given the actual mean μ_X, variance σ_X^2, skewness γ_X, and kurtosis β_X of X, we strive to find a pair (λ_3, λ_4) such that $(\gamma_Y, \beta_Y) = (\gamma_X, \beta_X)$. In [26] it is suggested to use Newton's method to approximate a vector $\mathbf{u} = [\lambda_4 \lambda_3]^T$ with:

$$\mathbf{G}(\mathbf{u}) = \begin{bmatrix} \gamma_Y(\mathbf{u}) - \gamma_X \\ \beta_Y(\mathbf{u}) - \beta_X \end{bmatrix} = 0. \tag{14}$$

In each iteration, the vector $\mathbf{u}$ is updated using the relation:

$$\mathbf{u}_{j+1} = \mathbf{u}_j - \mathbf{J}^{-1}(\mathbf{u}_j)\mathbf{G}(\mathbf{u}_j), \tag{15}$$

where $\mathbf{J}(.)$ is the Jacobian matrix defined as:

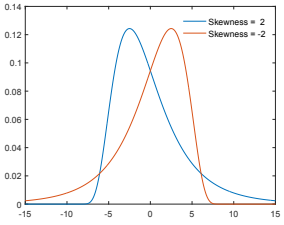
$$\mathbf{J}(\mathbf{u}_j) = \begin{bmatrix} \frac{\partial \gamma_Y(\mathbf{u}_j)}{\partial \lambda_4} & \frac{\partial \gamma_Y(\mathbf{u}_j)}{\partial \lambda_3} \\ \frac{\partial \beta_Y(\mathbf{u}_j)}{\partial \lambda_4} & \frac{\partial \beta_Y(\mathbf{u}_j)}{\partial \lambda_3} \end{bmatrix}. \tag{16}$$

Once λ_3 and λ_4 are fixed, the other parameters can easily be computed by:

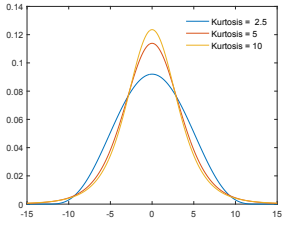
$$\lambda_2 = \frac{\sigma_X}{\sigma_Y}, \qquad \lambda_1 = \mu_X - \lambda_2 \mu_Y. \tag{17}$$

Similar to the EMG, the SGL only considers positive non-zero skewness and needs to be mirrored for a negative skewness. Besides, and compared to the EMG and Pearson's system, this procedure has a higher computational complexity which can become significant if a large number of PDFs have to be estimated. For example, this can be the case for non-profiled attacks such as MIA that require to compute PDFs for every possible subkey candidate. Indeed, our practical experiments employing SGL (presented in Section 5) required significant more time compared to the other considered estimators but remained tractable.

F Supporting Figures

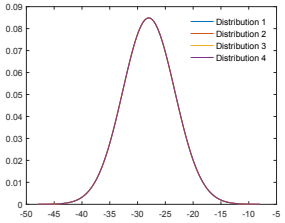


(a) Fixed kurtosis, varying skewness.

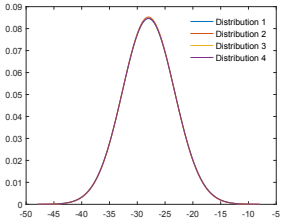


(b) Fixed skewness, varying kurtosis.

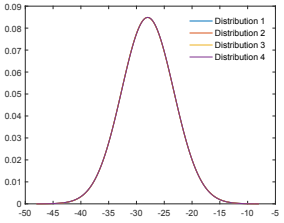
Fig. 9. Distributions with fixed mean and variance and varying skewness and kurtosis.



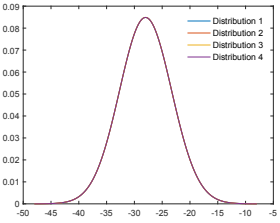
(a) Evaluation through means.



(b) Evaluation through variances.

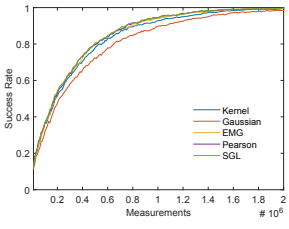


(c) Evaluation through skewnesses.

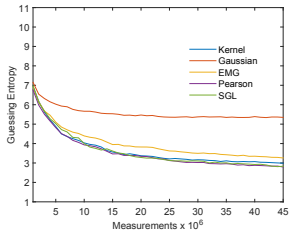


(d) Evaluation through kurtoses.

Fig. 10. The estimated PDFs of the four distributions from Table 1 with partly fixed moments according to the four evaluations cases.



(a) Profiled Template Attack.



(b) Non-profiled MIA Attack.

Fig. 11. Additional (profiled and non-profiled) attacks with kernel density estimation and comparison with other attacks exploiting all the moments at time sample 719.

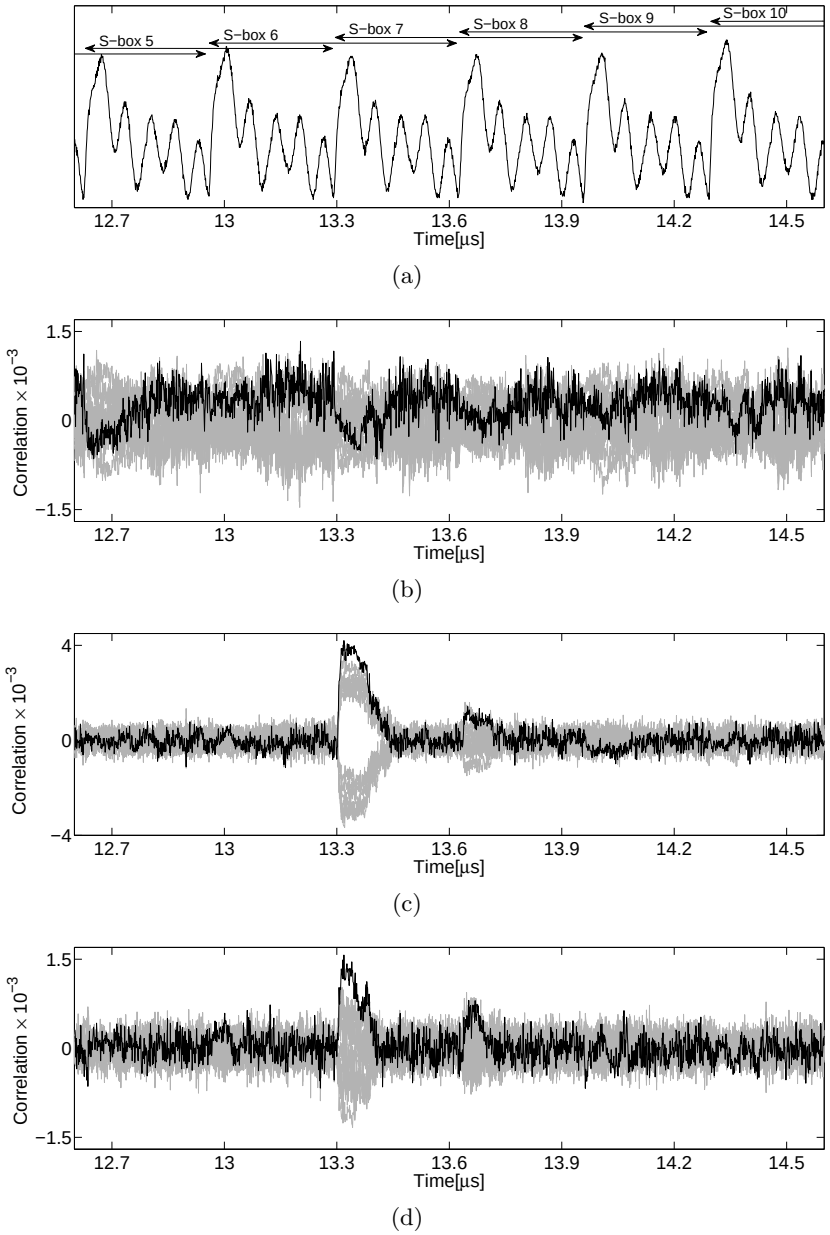
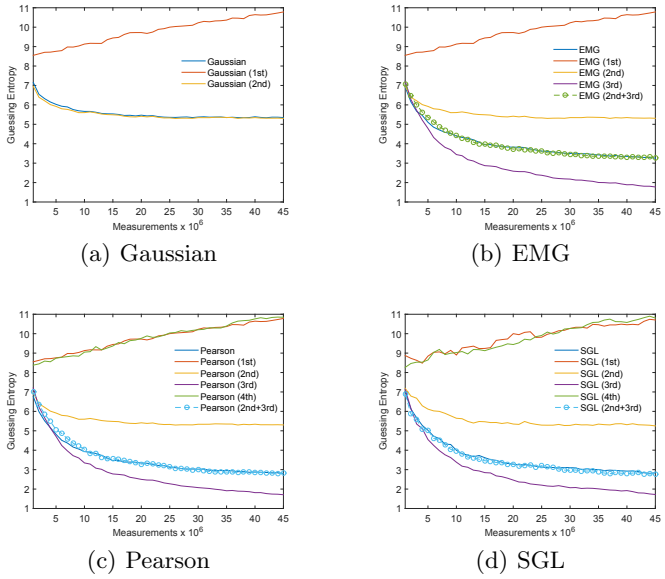


Fig. 12. (a) sample trace. (b) first-order, (c) second-order, and (d) third-order MCP-DPA results for different time samples in the leakage traces (taken from [33]).

G Mutual Information Analysis

After examining many different models,[9] we selected the three most significant bits of the S-box output as the best alternative.



(a) Gaussian (b) EMG

(c) Pearson (d) SGL

Fig. 13. Guessing entropy for MIA based on different estimation tools (at sample 719).

Using each density estimator (with various combinations of moments), we further ran 1000 MIA experiments for each given number of traces, and computed the guessing entropy as defined in [51]. The reason for not using the success rate again is that the convergence of the attacks is not guaranteed in this case (and actually not all the attacks converged). The results depicted in Figure 13 indicate that the estimators that capture more of the available moments generally perform better. Yet, the most interesting (and somewhat surprising) fact is that the most useful moment is now the third one rather than the second one. A similarly interesting observation is that the best attack is not the one combining all moments. This is not contradictory with the previous analysis, since such non-profiled attacks naturally deviate from the worst case predictions based on the profiled PI values. Indeed, in the case of MIA, the estimation of the model parameters is performed "on-the-fly", which implies that the best option is not to characterize the leakage the most carefully, but to reach a sufficiently precise estimation sufficiently quickly. Besides, our experiments also indicate that (non-profiled) models that are useful for certain moments (and as a matter of fact, certain time samples as well) may not be as good for others. This somehow

[9] Including HW, any single bit, pair and triple of bits of the S-box output.

joins the conclusions in [53] regarding the difficulty to interpret the result of non-profiled side-channel attacks in the context of masking. Additional curves are provided in Figure 11(b), including attacks exploiting kernel-based models that have an efficiency comparable to other attacks using all the moments, and hence are less efficient than the best attacks using only the third moment.