# Not optimal but efficient: a distinguisher based on the Kruskal-Wallis test

Yan Yan[1], Elisabeth Oswald[1], and Arnab Roy[1,2]

[1] University of Klagenfurt
[2] University of Innsbruck

**Abstract.** Research about the theoretical properties of side channel distinguishers revealed the rules by which to maximise the probability of first order success ("optimal distinguishers") under different assumptions about the leakage model and noise distribution. Simultaneously, research into bounding first order success (as a function of the number of observations) has revealed universal bounds, which suggest that (even optimal) distinguishers are not able to reach theoretically possible success rates. Is this gap a proof artefact (aka the bounds are not tight) or does a distinguisher exist that is more trace efficient than the "optimal" one? We show that in the context of an unknown (and not linear) leakage model there is indeed a distinguisher that outperforms the "optimal" distinguisher in terms of trace efficiency: it is based on the Kruskal-Wallis test.

**Keywords:** Distinguisher · Side Channel

## 1 Introduction

To exploit the information contained in side channels we use distinguishers: these are key-guess dependent functions, which are applied to the side channel observations and some auxiliary input (plaintext or ciphertext information), that attribute scores to key guesses. Optimal distinguishers [HRG14] are distinguishing rules derived by the process of maximising the likelihood of ranking the key guess that corresponds to the true secret value first (via their respective scores). The mathematical setup to derive optimal distinguishers is agnostic to estimation and trace efficiency, and thus an optimal distinguisher is not per construction the most trace efficient one. However, the optimal distinguishing rules that were derived in [HRG14] outperformed (experimentally) other distinguishers, or when not, [HRG14] showed mathematical equivalence between an optimal distinguishing rule and a classical rule. For instance, the correlation distinguisher turned out to be equivalent to the optimal rule in the situation where the leakage function is known and the noise is Gaussian.

The situation in which an adversary is confronted with a new device that contains an unknown key is interesting because it corresponds to the "hardest challenge" for the adversary: they should recover the key with only information about the cryptographic implementation. Framing this in the context of

side channel distinguishers, this leads to a type of distinguisher that neither requires assumptions about the noise distribution nor information about the device leakage distribution. Previous research has looked at distinguishers such as mutual information [GBTP08], Spearman's rank correlation [BGL08], and the Kolmogorov-Smirnov (KS) test [WOM11] in this context — these papers pre-date the seminal paper [HRG14] that establishes how to derive an optimal distinguishing rule.

Relatively recently only it was argued that the mutual information can be recovered as the optimal distinguishing rule [dCGHR18] if no assumptions about the device leakage distribution can be made. They also show experimentally that mutual information is the most trace efficient distinguisher in this setting. Next, better bounds for the estimation of the first order success rate (i.e. the probability to rank the key guess that corresponds to the true secret key first based on distinguishing scores) were derived in [dCGRP19]. The idea here was to derive these bounds independently of any specific distinguisher, purely based on the mutual information between the observed leakage and the key. The bounds were then compared to the respective optimal distinguishing rule. It turned out that there is a considerable gap between the optimal distinguisher and the bounds. This begs the question: could there indeed be a distinguisher that is more trace efficient than the one recovered as the optimal distinguishing rule?

### 1.1 Our contributions

We find a more trace efficient distinguisher by switching to rank based statistics. Previous work has once touched on rank based statistics before (Spearman's rank correlation) but we seek out a method that works even if the relationship between the intermediate values and the device leakage is not monotonic: this leads us to explore the Kruskal-Wallis method. We show how to translate it to the side channel context **(the important trick here is to rank the traces itself prior to any partitioning)** and we demonstrate how to estimate the number of needed traces for statistical attack success. We extend the existing work here by **developing a lower bound for the number of needed traces**.

Following established practice we then provide experimental results that enable us to conclude also from a practical point of view that the anticipated theoretical advantages show in practice. We cover **a range of situations** where we explore different target functions and different device leakage functions. In terms of target functions, we use non-injective target functions (as required by the assumptions in [HRG14,dCGHR18]), and also injective target functions with the bit-dropping trick. For device leakage functions we cover functions that range from highly non-linear to linear. We investigate Gaussian and Laplacian noise. Our philosophy is to include settings from prevoius work and more. We also consider implementations based on shared out intermediate values. Experiments that vary all these factors are necessarily based on simulations. We also demonstrate that our observations translate to real device data by using traces from two AES implementations: one with and one without masking.

*Our research exhibits, for the first time, in the setting where no information about the device leakage distribution is available, a distinguishing rule that is more trace efficient than the optimal distinguishing rule (MI). Our research also shows for the first time that a purely rank based distinguisher is effective in the context of masking.*

We provide the necessary background about (rank based) distinguishers, and our notation in Sect 2. Then we introduce the Kruskal-Wallis method and turn it into a distinguisher (alongside the analysis for the number of needed traces from a statistical point of view) in Sect. 3. In Sect.4 we show and discuss the simulation results, and in Sect. 5 we show and discuss the results for the real traces. We conclude in Sect. 6.

## 2 Background

We try and use notation that is uncluttered whenever we refer to well established background, in particular, when it comes to known facts about distinguishers, and we "overload" variables so that they simultaneously refer to sets and random variables. For instance, we use $L$ to refer to the set of observed traces, which we also know to have a distribution.

### 2.1 Side channel attacks and notation

We assume that the side-channel leakage $L$ can be expressed as a sum of a key dependent function $M$ and some independent noise $\varepsilon$:

$$L = M(V_{k^*}) + \varepsilon.$$

The device leakage model $M$ is not known in practice. It is a function of $V$, an intermediate value, which depends on some input word $X$ and a fixed and unknown secret key word $k^*$. We assume that the noise follows a Gaussian distribution $\varepsilon \sim \mathcal{N}(0, \sigma)$.[3] The intermediate $V$ is derived by the keyed cryptographic function $f_{k^*}$:

$$V_{k^*} = f_{k^*}(X).$$

In a side-channel attack, the adversary is given a set of leakages $L$ and their corresponding inputs $X$[4]. To recover the correct (secret) key $k^*$ embedded within the device, the adversary first computes the (predicted) intermediates $V_k$ under all possible guesses of $k$, from the given input $X$. Then they compute the hypothetical leakage value $L_{\mathcal{H},k} = \mathcal{H}(V_k)$ by assuming a leakage function $\mathcal{H}$. In side-channel attacks that rely on a direct or proportional approximation of the device leakage, the quality of $\mathcal{H}$ determines the success or efficiency of the corresponding attacks. When no model is known, then $\mathcal{H}$ is simply the identity function.

[3] For readability we do not make input and key dependence explicit in the leakage $L$.
[4] Side-channel attacks are also possible by exploiting the output with $f_{k^*}^{-1}$.

A distinguisher $D$ is used to compute the distinguishing score $d_k$ from the predicted intermediates $V_k$ and the observed leakage $L$. In a successful side-channel attack, the correct key $k^*$ is determined as the maximum distinguishing score(s):

$$k^* = \arg\max_k d_k = \arg\max_k D(L_{\mathcal{H},k}, L)$$

It is important to bear in mind that distinguishers are based on estimators of statistical quantities, thus in the formulas below we indicate this fact by placing a hat above the respective quantity. Distinguishers may or may not be based on some either assumed or known properties of the observed leakage $L$. In statistical jargon, statistics that require assumptions about the distribution are called "parametric" and statistics that do not require assumptions about the distribution are called "non-parametric". In this paper we work on the assumption that we are in a "first contact" scenario where the adversary utilises no information about $L$ in their initial attack attempt: this hence requires them to use non-parametric statistics, thus a non-parametric distinguisher.

In all practical side-channel attacks, the targeted intermediate $V_k$ is normally a part of operands being processed by the device during the cryptographic algorithms, and the key $k$ is a chunk of the cryptographic key. The complete key recovery is done via performing multiple side-channel attacks on each of the key chunks (thus we use a divide and conquer strategy). Also observable leakage often is given as a real-valued vector: e.g. power traces consist of many measurement points. Distinguishers are either applied to individual trace points, or to specific subsets of trace points. Therefore, in our aim to keep the notation uncluttered, we do not include any variables for indices for trace points or the like. We implicitly understand that the distinguisher is applied to (many) trace points or sets of trace points individually.

### 2.2 Rank transformations

Many statistical techniques that do not require assumptions about the underlying distributions have been developed by working on ranked data. Suppose that we have a set of leakages $L$: there are several ways in which ranks can be assigned to the leakages in the set. The two most natural types of assigning ranks are the following:

**Type 1:** The entire set is ranked from smallest to largest (or vice versa), and the smallest leakage having rank 1, the second smallest having rank 2, etc.

**Type 2:** The set $L$ is partitioned according to some rule into subsets, then each subset is ranked independently of all other subset, by ordering the elements within a set (either from smallest to largest or vice versa).

Ties are resolved by assigning the average of the ranks that the ties would have received.

Any monotonic increasing function that is applied to the data does not change the ranking of the data. In our text we indicate that ranking takes place by applying the $\mathsf{rank}()$ function to the resp. variables. The type of ranking will be clear from the context.

### 2.3 Non-parametric side-channel distinguishers

For the sake of completeness we provide a very brief description of the non-parametric side-channel distinguishers that we use as comparisons with are new distinguisher.

**Difference of Means** The Difference of Means (DoM) [KJJ99] is often used as a baseline distinguisher, and it can be defined such that it makes minimal assumptions about the leakage distribution. For its' computation, the traces are divided into two groups $L_{V_k=0}$ and $L_{V_k=1}$ depending on whether a predicted single bit of a targeted intermediate is zero or one ($V_k = 0$ or $V_k = 1$). The distinguishing score is defined as the estimated difference of means (often one takes the absolute value)):

$$d_k = |\hat{\mathbb{E}}(L_{V_k=0}) - \hat{\mathbb{E}}(L_{V_k=1})|.$$

**Spearman's Rank Correlation** This is a non-parametric alternative to Pearson's correlation, and it was investigated in [BGL08] against an AES implementation. It was shown to be significantly more efficient (in terms of success rate) compared to Pearson's correlation-based attack [BCO04] (a.k.a. CPA). In this attack, the adversary computes the hypothetical leakage from $V_k$ by computing $L_{\mathcal{H},k}$ where $\mathcal{H}$ is guessed/assumed by the adversary. Then $L_{\mathcal{H},k}$ and $L$ are ranked and the (absolute value of the) correlation coefficient is estimated as follows

$$d_k = \left| \frac{\hat{Cov}(\mathsf{rank}(L), \mathsf{rank}(L_{\mathcal{H},k}))}{\hat{\sigma}_{\mathsf{rank}(L)} \hat{\sigma}_{\mathsf{rank}(L_{\mathcal{H},k})}} \right|.$$

Notice that although the adversay must "guess" a hypothetical leakage model, there is no requirement for the device leakage to follow a Gaussian distribution.

**Mutual Information** Mutual Information [GBTP08] analysis is a distinguishing method that can be used without the need for $\mathcal{H}$. The MI distinguishing score is computed by estimating the mutual information from a set of collected traces and the corresponding inputs or plaintexts:

$$d_k = \hat{I}(L, V_k) = \hat{H}(L) - \hat{H}(L|V_k)$$

where $\hat{H}$ and $\hat{I}$ denote the (estimated) Shannon's entropy and mutual information respectively. For estimating MI, different entropy estimation methods have been studied, but the most commonly applied and efficient method (over $\mathbb{R}$) is the so-called binning method that is used in the original proposal of MIA [GBTP08]. We also use this same estimation method in our experiments.

Note that MI requires that the target function $f_k$ is not a bijection as discussed in [WOS14,dCGHR18]. When MIA is applied to cryptographic target that is a bijection, then the bit dropping technique [RGV14] that simply chops off a selected number bits from the output, is used. Although it is not necessary to supply MI with a hypothetical leakage model $\mathcal{H}$ this is frequently done in the literature, in particular by selecting the Hamming weight as $\mathcal{H}$.

**Kolmogorov–Smirnov (KS)** The KS test-based distinguisher [WOM11] is suggested as an alternative to using MI. The distinguishing score (of a key) is defined as the average of KS distances between the leakage distribution of $L$ and leakage distributions of $L_{V_k}$ for each predicted intermediate $V_k$ i.e.

$$d_k = \hat{\mathbb{E}}_{V_k} \left( \sup_l |F_L(l) - F_{L_{V_k}}(l)| \right)$$

where $F_L(l)$ and $F_{L_{V_k}}(l)$ are the Cumulative Distribution Functions (CDFs) of $L$ and $L_{V_k}$ respectively. From a finite sample set $A$ the empirical CDF is computed by $F_A(x) = \frac{1}{n} \sum_{a \in A} I_{a \leq x}$ where $I$ is the indicator function and $|A| = n$.

## 3 The Kruskal-Wallis test as side-channel distinguisher

The Kruskal-Wallis test (KW) [KW52] is a non-parametric method for the *analysis of variance* (ANOVA): this means it does not require any distributional assumption about the leakage $L$. The KW test is based on the ranks of the observed data and it is often used to check whether (or not) multiple groups of samples are from the same distribution. In this section we explain how to construct a KW based distinguisher, and we discuss the salient properties of the resulting distinguisher.

### 3.1 The KW statistic as a distinguisher

In this section we describe how to compute the KW statistic in a side-channel setting, and we argue why it gives a sound side channel distinguisher. For a generic description of the KW statistic we refer the readers to appendix A.

Informally, the KW test statistic is derived by first ranking the observed data, and second by grouping the data according to the resp. (key dependent) intermediate values. Then the tests checks if the groups can be distinguished from another or not, by comparing the variances between the groups and within the groups.

More formally, let us assume that we have $N$ side channel leakages. We apply the type 1 rank transformation to the side channel leaks, and then work with the ranked data: $\mathsf{rank}(L)$. For each key guess $k$, the ranked data is grouped according to the respective intermediate $V_k$. Thus the set $R_k^i = \{\mathsf{rank}(L) | V_k = i\}$ contains the ranks of leakages where the intermediate $V_k$ equals $i$. Let $R_k^{i,j}$ refers to the $j$-th element in $R_k^i$. Suppose that we have $t$ groups and the size of group $R_k^i$ is $n^i$ and so $N = \sum_{i=1}^{t} n^i$.

Let us assume that the group $R_k^i$ has distribution $F^i$. The null hypothesis is that all the groups have the same distribution, and alternative hypotheses of KW test is that the groups can be distinguished:

$$\begin{aligned} H_0 &: F^0 = F^1 = \ldots = F^{t-1} \\ H_a &: F^i \neq F^j \quad \text{for some} \quad i, j \quad \text{s.t} \quad i \neq j. \end{aligned} \tag{1}$$

The average of the ranks in $R_i$ is given as:

$$\bar{R}_k^i = 1/n_i \sum_{j=1}^{n^i} R_k^{i,j}$$

and $\bar{R_k} = (N+1)/2$ the average of all $R_k^{i,j}$.
The KW test statistic is defined [KW52] as:

$$d_k = (N-1)\frac{\sum_{i=1}^{t} n_i(\bar{R_k^i} - \bar{R_k})^2}{\sum_{i=1}^{t}\sum_{j=1}^{n_i}(R_k^{i,j} - \bar{R_k})^2} \tag{2}$$

If the elements in $R_k^i$ are all from the same distribution, then all $\bar{R^i}_k$ are expected to be close to $\bar{R_k}$ and thus the statistic $d_k$ should be smaller, than when the elements in $R_k^i$ are from different distributions. Thus large values of the test statistic imply that we reject the null hypothesis of the KW test (i.e. we have enough data to conclude that there are meaningful groups). We can use this test statistic readily as a side channel distinguisher: the groups are given by the key dependent intermediate values $V_k$. Thus, for $k = k^*$ we have a meaningful grouping of the ranked leakages, and thus the test statistic is large. If $k \neq k^*$, then the ranked side channel leaks are randomly assigned to different groups, which will lead to a small test statistic. Consequently the value of $d_{k^*} >= d_k$ for $\forall k$, which implies that it is a sound side channel distinguisher.

### 3.2 Properties of the KW distinguisher

Side channel distinguisher are most useful if they can be applied in different settings, including higher order attacks. It is also beneficial to be able to derive sample size estimates. For some of the existing non-parametric, in particular in the case of MI, this is hard to achieved. We now explain what is possible for the KW distinguisher.

**Application to higher order attack scenarios.** In masked implementations, an intermediate value is represented as a tuple of shares. The leakage of a single share is uninformative, but a statistic that exploits the distribution of the entire tuple enables key recovery. The canonical way of applying distinguishers to masked implementations is via processing the observed leakage traces: a popular (processing) function is the multiplication of (mean-free) trace points [PRB09]. Such trace processing produces a new trace in which each point now is based on the joint leakage of multiple points (aka shares). Using the mean-free product to produce joint leakage is compatible with the Kruskal-Wallis distinguisher (if the mean-free product of two values is larger than the mean-free product of another two values then this property is preserved by ranking: it is a monotonically increasing function), and we show how well it performs in the experimental sections.

**Computational cost.** The KW test is often compared to the Wilcoxon-Whitney-Mann test (MWW) [MW47] with respect to computation costs, which is another rank based non-parametric test. The major difference between the two is that MWW is applied to paired data against two values, whereas KW is applied to multiple groups. The latter thus naturally fits with the side channel setting where the intermediate values fall naturally in multiple (independent) groups. Applying MWW in the side channel setting increases the computational cost. For example, in case of $t$ groups we need to apply MWW in the worst case $\binom{t}{2}$ times. Thus the KW test is a natural choice over the MWW test. We found that the computational cost of KW is of the same order as other generic distinguishers (MI, KS).

**Number of samples.** For the KW statistic, the theoretical analysis [FZZ11, Theorem 1] shows how to estimate the sample size. The main result necessary for estimating the sample size in a KW test is that under the alternative hypothesis the KW statistic follow a non-central $\chi^2$distribution. Let $\lambda_i = n_i/N \geq \lambda_0$ for all $i$ and a fixed $\lambda_0 > 0$. And let $\alpha$ be the confidence level and $\beta$ be the power of the test. Then the estimated sample size is given as

$$\widetilde{N} = \frac{\tau_{\alpha,\beta}}{12 \sum_{i=1}^{t} \lambda_i \left( \sum_{s \neq i} \lambda_s (\hat{p}_{is} - 1/2) \right)^2}. \tag{3}$$

For each pair $i, s$ s.t. $i \neq s$, the probability estimates $\hat{p}_{is}$ can be computed from the given data sample of size $N$ as follows

$$\hat{p}_{is} = \frac{1}{N_i N_k} \sum_{j=1}^{N_i} \sum_{\ell=1}^{N_s} \left( \mathcal{I}(X_{s\ell} < X_{ij}) + \mathcal{I}(X_{s\ell} = X_{ij})/2 \right)$$

where $\mathcal{I}$ is the indicator function, and $i, s \in \{1, 2, \ldots t\}$. Note that the second part of the above expression corresponds to the ties in ranking. In eq. (3) $\tau_{\alpha,\beta}$ is solution to $\mathbb{P}(\chi^2_{t-1}(\tau) > \chi^2_{t-1,1-\alpha}) = 1 - \beta$ for some fixed $\alpha, \beta$, and $\chi^2_{t-1,1-\alpha}$ is the $(1-\alpha)$ quantile of central $\chi^2$distribution with $t-1$ degrees of freedom.

The estimation of sample size following equation eq. (3) is biased and needs to be adjusted. As explained in [FZZ11], an adjusted estimator $\widehat{N}$ is defined as follows

$$\widehat{N} = \widetilde{N} \cdot \frac{\mathsf{median}\{\chi^2_{t-1}(\hat{\tau})\}}{\hat{\tau}} \tag{4}$$

where $\hat{\tau} = N \cdot 12 \sum_{i=1}^{t} \lambda_i \left( \sum_{s \neq i} \lambda_s (\hat{p}_{is} - 1/2) \right)^2$.

*Considering correct and incorrect key hypotheses.* The application of sample size estimation technique requires care in the context of side-channel key recovery attack. Recall that in a statistical (hypothesis) testing there are two types of errors namely

1. **Type I error** $\alpha$ where the null hypothesis $H_0$ is rejected when the hypothesis $H_0$ is true, and
2. **Type II error** $\beta$ where the null hypothesis $H_0$ is not rejected when the alternate hypothesis $H_a$ is true.

In a successful attack the null hypothesis should not be rejected for any $k$ where $k \neq k^*$ (thus we want $\alpha$ to be small). However, under the correct key guess $k = k^*$ the alternative hypothesis $H_a$ is true and we should not fail to reject $H_0$. Hence, $\beta$ should be small so that the power of the test $1 - \beta$ is large. In fact we wish to have a high power for both cases.

Thus we should perform the sample size estimation for both cases (correct and incorrect keys) and then take the maximum of these sample sizes as a conservative estimate. In statistical hypothesis testing typically it is ensured that the value of $\mathbb{P}(\text{Type I error}) \leq 0.1$ and $\mathbb{P}(\text{Type II error}) \leq 0.2$.

*Example 1.* In this example we show the sample size estimation for $N = 1000$ using simulated Hamming weight traces of AES Sbox where the Gaussian noise has $\sigma = 6$.

We choose $\alpha = 0.025$ (corresponding to the confidence level) and $\beta = 0.05$ (corresponding to the power of the test). First, using the technique as described above, we find the generic estimate of the sample size as per eq. (3). For applying the leakage estimation (or KW attack) we extract the 4 Least Significant Bits (LSB) from the output of the Sbox.

For this experiment the degrees of freedom of the $\chi^2$distributions is $16 - 1 = 15$ (the number of different groups are 16 corresponding to the 4-bit output values obtained). Note that $\tau_{\alpha,\beta}$ depends only on the degrees of freedom, $\alpha$ and $\beta$. In this case $\tau_{\alpha,\beta} = 1.8506$. We compute $\widetilde{N}$ for different key choices. Here we only show the computation for one key that corresponds to the maximum $\widetilde{N}$. The estimation process is carried out in the same way for other keys.

Estimating $\lambda_i$ and $\hat{p}_{is}$ from 1000 data points we obtain the $\widetilde{N} = \frac{1.8506}{.0041} \approx 451$. Since this is a biased estimate we obtain the adjusted estimate as

$$\widehat{N} = \widetilde{N} \cdot \frac{\mathsf{median}\{\chi^2_{t-1}(\hat{\tau})\}}{\hat{\tau}} = 451 \cdot \frac{\mathsf{median}\{\chi^2_{15}(4.1)\}}{4.1} \approx 2015. \tag{5}$$

*Remark 1.* For estimating the sample size in the context of side-channel attack, $\lambda_i$ can be estimated from the target cryptographic function (instead of estimating it from the data). Suppose, the target function is 8-bit Sbox, and say 4 bits of the output is chosen for the attack. In this case, for all $2^8$ input values, the number of elements $n_i$ in each $2^4$ groups can be computed.

*Example 2.* In this example we show the sample size estimation when traces are simulated from ARX function with a HW leakage model and Gaussian noise with $\sigma = 6$. We fix $N = 1000$ and follow the same process as in Example 1. Here we choose $\alpha = 0.001$ and $\beta = 0.1$.

We consider a key recovery attack (using KW statistic) which recovers 4-bit key chunk from each $k_1$ and $k_2$, in the usual divide and conquer process used for side-channel attack. The ARX function is defined as

$$A(x) = (x \oplus k_1) \boxplus (y \oplus k_2).$$

($\oplus$ denotes the bit-wise exclusive-or and $\boxplus$ the addition in $GF(2^{16})$). So, the degrees of freedom for the $\chi^2$distribution remains $16-1 = 15$. The biased sample size estimation gives $\widetilde{N} \approx 992$. After adjusting the bias as in Example 1 we get $\widehat{N} \approx 2212$.

**Corollary 1.** *The generic estimate $\widetilde{N}$ in eq.* (3) *(and bias adjusted estimate $\widehat{N}$ in eq.* (4)*) gives estimated lower bound on sample size.*

*Proof.* The sample estimate is derived from the fact that $\hat{\tau} \approx \tau_{\alpha,\beta}$. Recall that $\tau_{\alpha,\beta}$ is the solution to the equation

$$\mathbb{P}(\chi^2_{t-1}(\tau) > \chi^2_{t-1,1-\alpha}) = 1 - \beta.$$

for some fixed $\beta$. Now, if we obtain a $\hat{\tau}_1$ from the fixed sized data such that $\hat{\tau}_1 \geq \tau_{\alpha,\beta}$, then $\mathbb{P}(\chi^2_{t-1}(\tau_1) > \chi^2_{t-1,1-\alpha})$ will be more than $1-\beta$. This is favourable since we want to maximise the power of the test. Thus we have

$$\hat{\tau} \geq \tau_{\alpha,\beta} \quad \Longrightarrow \quad \widetilde{N}^* \geq \frac{\tau_{\alpha,\beta}}{12 \sum_{i=1}^{t} \lambda_i \left( \sum_{s \neq i} \lambda_s (\hat{p}_{is} - 1/2) \right)^2} = \widetilde{N}$$

The lower bound on the bias adjusted estimate $\widehat{N}^*$ follows from this.

## 4 Experiments based on simulated leakage

We now detail a range of experiments that are based on simulating side channel data. Experiments based on simulated data offer the advantage, over experiments based on data from devices, that we can efficiently vary implementation characteristics such as the leakage function, the cryptographic target function, and the signal to noise ratio. Therefore the inclusion of simulations is standard in research on distinguishers.

We display simulation outcomes in terms of the success rate as function of an increasing number of side channel observations. Our comparisons include the KW test, mutual information analysis (MI) with an identity leakage model, mutual information analysis with a Hamming weight leakage model (MI-HW), the Kolmogorov-Smirnov test and Spearman's test. We included MI-HW because of its wide use in the literature (and despite the obvious fact that it is no longer assumption free).

Before we discuss the outcomes, we provide an informal but detailed description of the choices for the cryptographic target functions $V_k$ as well as the device leakage functions $M$.

### 4.1 Simulation setup

Our choice of target functions $V_k$ is informed by best-practice: it is well known that properties of the target function impact on distinguishability and therefore we aimed to select a function that is known to be "poor" target, to challenge all distinguishers. Our selection observed a further requirement imposed by the use of MI (as main comparison) that MI is only a sound distinguisher for non-injective target functions (if a target function is injective, then MI cannot distinguish any key candidates)[HRG14], and the bit-dropping trick must be used. Therefore, we selected as a poor non-injective target function $V_{ni}$ the non-injective target function is the modular addition that is part of many ARX constructions, which is also the basis of modern permutation based ciphers such as SPARKLE:

$$V_{ni}(x_l, x_r, k_l, k_r) = (x_l \oplus k_l) \boxplus (x_r \oplus k_r)$$

where $x_l \| x_r \in \{0,1\}^{32}$ is a state element, and $k_l \| k_r \in \{0,1\}^{32}$ is the key, and $\boxplus$ is the addition modulo $2^{16}$.

We also experimented with a function that is known to be an excellent target function, namely the AES SubBytes operation, which is injective, and thus the bit-dropping trick must be applied. To aid the flow of this submission, we include the results of this in the appendix (they are aligned with the results for the injective target function).

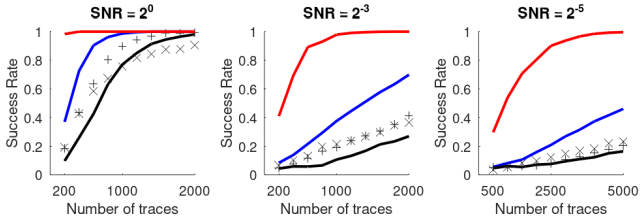
Our choice of leakage functions $M$ is also informed by best-practice: leakage functions are also well known to impact on distinguisher performance. Linear leakage functions help distinguishers that are based on distributional assumptions or simple hypothetical leakage models. Highly non-linear leakage functions are representative of of complex leakage originating in combinational logic ([LBS19] and [GMPO20]) are a motivating factor for studing "assumption free" distinguishers like MI, KS and KW.

In our experiments we thus use a range of device leakage functions, which are defined as follows. Let $y_i$ be the $i$th bit of $y$. Then we consider two linear device leakage functions (Hamming weight and Randomly weighted bits), and two non-linear leakage functions (Strongly non-linear and Binary), as follows:

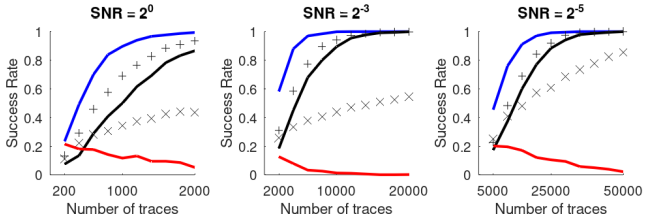
**Hamming weight:** $M(y) = \sum_{i=1}^{n} y_i$
**Randomly weighted bits:** $M(y) = \sum_{i=1}^{n} w_i y_i$ with $w \in [-1, 1]$
**Strongly non-linear:** $M(y) = S(y)$, with $S(y)$ defined to be the Present S-Box
**Binary:** $M(y) = \sum_i S(y)_i \pmod 2$, with $S(y)$ defined to be the Present S-Box

### 4.2 First order attack simulations

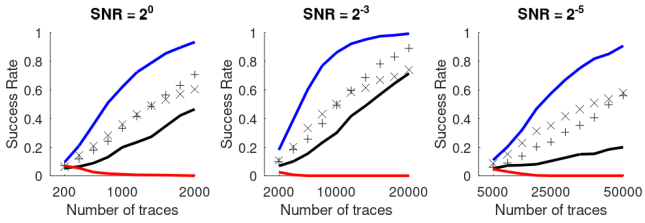
Figure 1a shows that the Spearman rank correlation has indeed a significant advantage (because it uses the correct hypothetical leakage model), compared to the other distinguishers. Note that the KW test-based attack outperforms the other generic distinguishers with a clear margin that is more significant in the lower SNRs.



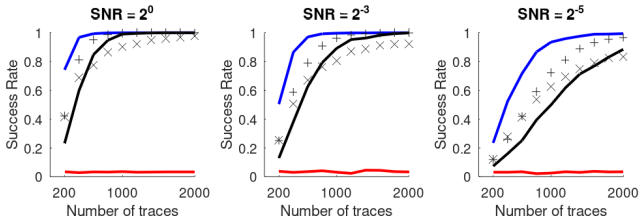

(a) HW leakage model




(b) Rrandomly weighted bits leakage model



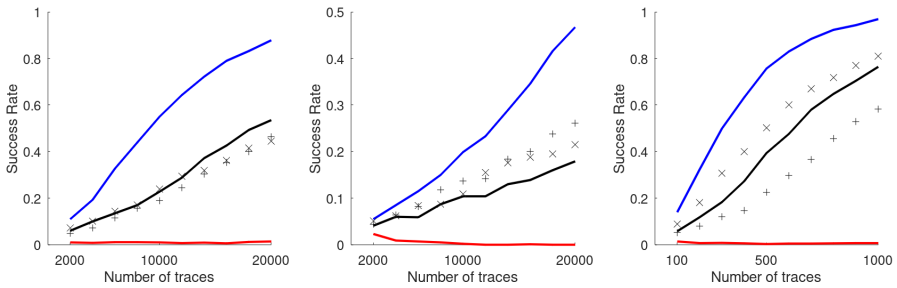

(c) Strongly non-linear (PRESENT S-Box) leakage model




(d) Binary leakage model

KW: MI: MI-HW:+ KS:× Spearman:

Fig. 1: Simulations for Modular Addition as a Target




(a) Hamming weight (b) Randomly weighted bits (c) Strongly non-linear

KW: MI: MI-HW:+ KS:× Spearman:

Fig. 2: 2-share Boolean masking of ARX with different leakage models

Figure 1b shows that the Spearman rank correlation fails: more traces reduce the success rate, which is a clear indication that that the "built in leakage model" is incompatible with $M$. This is a useful reminder that linear models are not necessarily compatible with a Hamming weight assumption. All model-free distinguishers succeed, and KW turns out to be the most trace efficient in all SNR settings. The MI and MI-HW distinguishers show similar performance while KS is the least trace efficient one among the successful attacks.

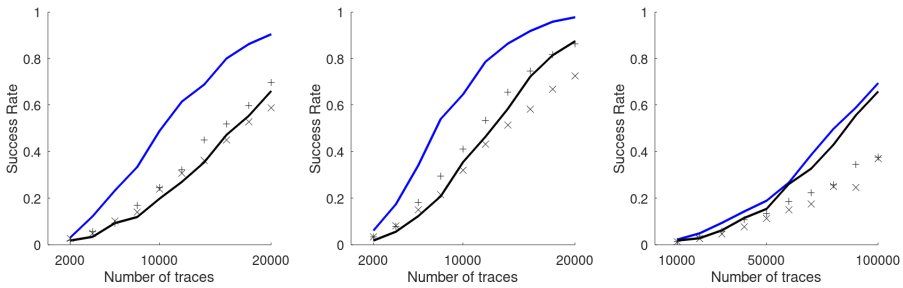
In the non-linear simulation (Figure 1c) We expect that Spearmans rank correlation will fail because the leakage model is not compatible. However MI with the same model works very well, alongside MI without model and KS. These three distinguishers show a very similar performance in all SNR settings. KW shows a clear margin to the other distinguishers, which is evidence that it is the preferable distinguisher in this setting.

The last simulation (Figure 1d) is a binary leakage model that represents an extreme case where the leakage is either 0 or 1 such that only a minimum resolution exists in the leakage values. In a high SNR setting, all assumption-free distinguishers recover the key. In low SNR seetings, the KW distinguisher show the quickest convergence to a high success rate, which is evidence that it is the preferable distinguisher in this setting.

### 4.3 Masked implementation

We further extend our simulations to a masked implementation by simulating the leakages of a 2-shares Boolean masking scheme using the same leakage models as before. To perform an attack we use the a well understood, and frequently adopted approach of combining the leakages from all independent shares via the centred product-combining function, [PRB09], which was also used in [BGP+11]. [5]

[5] It is worth noting that there exists no known optimal multivariate implementation for the above mentioned side-channel distinguishers [BGP+11,WOM11], because the




(a) 3 Shares SNR $2^2$ (b) 3 Shares SNR $2^3$ (c) 4 Shares SNR $2^3$

KW:— MI:— MI-HW:+ KS:×

Fig. 3: Higher order Boolean masking of AES with Hamming weight leakage

The results of the simulations for the 2-share Boolean masking scheme are shown in Figure 2a, Figure 2b and Figure 2c. For succinctness, we excluded the very low SNR settings of $2^{-3}$ and $2^{-5}$ (because the observations are the same as for the higher SNRs), and the results of binary leakage model (because all distinguishers failed in this setting). As is evident from the graphs, Spearman fails in all settings; among the successful attacks, KW turned out to be the most trace efficient distinguisher.

We then turn our attention to masking for the AES SubBytes operation, where Figures 3a-3c show that KW provides a clear advantage for low order masking.

## 5 Experiments based on Device Data

To complement our simulation results we also show experiments that were performed based on measurements from two processors. These processors are based on the ARM Cortex M0 and the ARM Cortex M3 architecture. We implemented the same target functions as before in the simulations.

To work with the masked implementation, we perform the same mean-free product combining pre-processing as in the simulations. Before showing the outcomes, we discuss the implementation characteristics in some more detail.

### 5.1 Implementation characteristics and experimental setup

Our simulated experiments ranged from unprotected implementations to implementations based on sharing out intermediate values. For implementations that are unprotected we only ensure functional correctness of our implementation. In the case of the non-injective target function, we utilise the modular addition in C and let the compiler translate this into Assbembly code. In the case of

outcomes are highly sensitive to various factors, including leakage models, noise levels and methods for pre-processing, etc.

the AES SubBytes implementation we use a simple table-based lookup. For the masked SubBytes implementation we use a custom Thumb-16 Assembly implementation of a two share ISW multiplication gadget. This implementation is specifically crafted to ensure that there are no first-order leaks.

Both processors are mounted in a special purpose measurement rig[6]. We have a state of the art scope and probe, but do not perform any filtering or de-noising before applying the distinguishers. The devices that we use are well characterised, and we know that they exhibit a range of leakage functions, which all have a strong linear component (thus they resemeble the two linear leakage functions that we considered in the simulations).

We apply the distinguishers to all trace points, and perform repeat experiments to determine the first order success rate. We then select the best point and plot the success rate graphs for this point only.

### 5.2 Experimental results

*Non-injective target function.* Figure 4a shows the results of repeat attacks on the modular addition on the M0. In the corresponding simulated experiments, we supplied Spearman with the Hamming weight leakage model and as a result it outperformed the other distinguishers when the device leakage model was also the Hamming weight. To demonstrate that Spearmans succeess in the Hamming weight simulation really was because we supplied it with the Hamming weight model, we now supply it with only 4 bits of the intermediate values. We give the same 4 bit intermediate values also to MI, MI-HW, KS and KW.

Lacking the correct leakage model, Spearman now completely fails. All other side-channel distinguishers successfully recover the key. KW shows again a better success rate than the competitors.

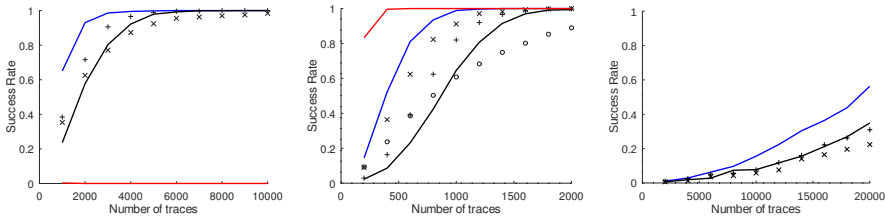
*Injective target function.* Figure 4b shows the results of repeat attacks on the SubBytes operation on the M0. Now we supply Spearman once more with the Hamming weight leakage model, which gives it a significant advantage over the other distinguishers (because the device features signifant linear leakage in all trace points).

KW is the most trace efficient distinguisher among the other distinguishers. DoM is the least efficient one which might due to the fact that DoM can only exploit a single bit leakage whereas other distinguishers exploit all 4 bit leakages.

*Masked implementation.* Figure 4c [7] shows a familiar picture: KW achieves a higher success rate by a clear margin over the other distinguishers. Spearman failed, so we did not include it anymore. The picture also shows that MI-HW no longer shows any advantage over MI, which one should expect given that

[6] We refrain to include more details at this point in order to maintain the anonymity of the submission.

[7] Spearman and DoM are excluded from Figure 4c as they failed against the masked implementation.




(a) 4LSB of the modular addition on M0

(b) 4LSB of AES SBox on M0

(c) 4LSB of Masked AES SBox on M3

KW: — MI: — MI-HW:+ KS:× Spearman: — DoM: ∘

Fig. 4: Experiments based on real device data

pre-processing is applied to the trace points as part of attacking the masking scheme.

## 6 Discussion and Conclusion

Of the distinguishers that we compared in this submission, Spearman and MI-HW are supplied with the Hamming weight leakage model. Theoretically, this gives them an advantage in situations where there is strong Hamming weight device leakage. We can see this advantage also experimentally: in all Hamming weight simulations, Spearman outperforms all other distinguishers, including MI-HW. This particular simulation showcases that iff the device leakage model is "simple" then there is no point in using MI, KS or KW.

In situations where the leakage model is unknown and HW based attack fail, they are the premise of our work, MI, KS, and KW are considerably better than Spearman (and MI-HW). When looking carefully at the experimental outcomes, then we can observe that the gap between the distinguishers decreases with lower SNR values. This behaviour is expected because of [MOS11], according to which they must, asymptotically speaking, get closer in terms of trace efficiency the lower the SNR.

All together our experiments provide strong evidence that MI is not the most trace efficient distinguisher setting where no leakage model is available, which is in contrast to [dCGHR18], who selected different distinguishers for comparison with MI.

Our results help clarify that "optimal distinguishers" are not necessarily the most trace efficient distinguishers, despite that in previous work they have always been identified as being more trace efficient (in their respective categories) than their "normal" counterparts.

## 7 Acknowledgment

Elisabeth Oswald and Yan Yan have been supported in part by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 725042).

## References


BCO04. Eric Brier, Christophe Clavier, and Francis Olivier. Correlation power analysis with a leakage model. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.

BGL08. Lejla Batina, Benedikt Gierlichs, and Kerstin Lemke-Rust. Comparative evaluation of rank correlation based DPA on an AES prototype chip. In Tzong-Chen Wu, Chin-Laung Lei, Vincent Rijmen, and Der-Tsai Lee, editors, *Information Security, 11th International Conference, ISC 2008, Taipei, Taiwan, September 15-18, 2008. Proceedings*, volume 5222 of *Lecture Notes in Computer Science*, pages 341–354. Springer, 2008.

BGP+11. Lejla Batina, Benedikt Gierlichs, Emmanuel Prouff, Matthieu Rivain, François-Xavier Standaert, and Nicolas Veyrat-Charvillon. Mutual information analysis: a comprehensive study. *J. Cryptol.*, 24(2):269–291, 2011.

dCGHR18. Eloi de Chérisey, Sylvain Guilley, Annelie Heuser, and Olivier Rioul. On the optimality and practicability of mutual information analysis in some scenarios. *Cryptogr. Commun.*, 10(1):101–121, 2018.

dCGRP19. Eloi de Chérisey, Sylvain Guilley, Olivier Rioul, and Pablo Piantanida. Best information is most successful mutual information and success rate in side-channel analysis. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2019(2):49–79, 2019.

FZZ11. Chunpeng Fan, Donghui Zhang, and Cun-Hui Zhang. On sample size of the kruskal-wallis test with application to a mouse peritoneal cavity study. *Biometrics*, 67 1:213–24, 2011.

GBTP08. Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel. Mutual information analysis. In Elisabeth Oswald and Pankaj Rohatgi, editors, *Cryptographic Hardware and Embedded Systems - CHES 2008, 10th International Workshop, Washington, D.C., USA, August 10-13, 2008. Proceedings*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442. Springer, 2008.

GMPO20. Si Gao, Ben Marshall, Dan Page, and Elisabeth Oswald. Share-slicing: Friend or foe? *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2020(1):152–174, 2020.

HRG14. 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.

KJJ99. Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 388–397. Springer, 1999.

KW52. William H. Kruskal and W. Allen Wallis. Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association*, 47(260):583–621, 1952.

LBS19. Itamar Levi, Davide Bellizia, and François-Xavier Standaert. Reducing a masked implementation's effective security order with setup manipulations and an explanation based on externally-amplified couplings. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2019(2):293–317, 2019.

MOS11. Stefan Mangard, Elisabeth Oswald, and François-Xavier Standaert. One for all - all for one: unifying standard differential power analysis attacks. *IET Inf. Secur.*, 5(2):100–110, 2011.

MW47. H. B. Mann and D. R. Whitney. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, 18(1):50 – 60, 1947.

PRB09. Emmanuel Prouff, Matthieu Rivain, and Régis Bevan. Statistical analysis of second order differential power analysis. *IEEE Trans. Computers*, 58(6):799–811, 2009.

RGV14. Oscar Reparaz, Benedikt Gierlichs, and Ingrid Verbauwhede. Generic DPA attacks: Curse or blessing? In Emmanuel Prouff, editor, *Constructive Side-Channel Analysis and Secure Design - 5th International Workshop, COSADE 2014, Paris, France, April 13-15, 2014. Revised Selected Papers*, volume 8622 of *Lecture Notes in Computer Science*, pages 98–111. Springer, 2014.

WOM11. Carolyn Whitnall, Elisabeth Oswald, and Luke Mather. An exploration of the kolmogorov-smirnov test as a competitor to mutual information analysis. In Emmanuel Prouff, editor, *Smart Card Research and Advanced Applications - 10th IFIP WG 8.8/11.2 International Conference, CARDIS 2011, Leuven, Belgium, September 14-16, 2011, Revised Selected Papers*, volume 7079 of *Lecture Notes in Computer Science*, pages 234–251. Springer, 2011.

WOS14. Carolyn Whitnall, Elisabeth Oswald, and François-Xavier Standaert. The myth of generic dpa...and the magic of learning. In Josh Benaloh, editor, *Topics in Cryptology - CT-RSA 2014 - The Cryptographer's Track at the RSA Conference 2014, San Francisco, CA, USA, February 25-28, 2014. Proceedings*, volume 8366 of *Lecture Notes in Computer Science*, pages 183–205. Springer, 2014.

## A The KW Statistic

Let $X_{ij}$ where $i = 1, \ldots, t$, $j = 1, \ldots, n_i$ be independent random samples collected from a population having $t$ groups and the sample size for group $i$ is $n_i$. Let us assume that the random variables $X_{ij}$ have distribution $F_i$. The generic null and alternative hypotheses of KW test are

$$\begin{aligned} &H_0 : F_1 = F_2 = \ldots = F_t \\ &H_a : F_i \neq F_j \quad \text{for some} \quad i, j \quad \text{s.t} \quad i \neq j. \end{aligned} \tag{6}$$

The observations are combined into one sample of size $N$ where

$$N = \sum_{i=1}^{t} n_i$$

This combined sample is ranked. Suppose, $R_{i,j}$ is the ranking of the $j$-th sample from the group $i$, $\bar{R}_i$ the average rank of all samples from group $i$:

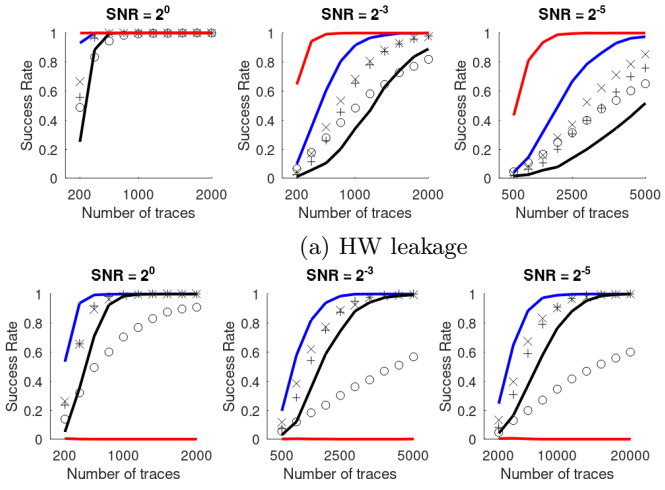
$$\bar{R}_i = {n_i}^{-1} \sum_{j=1}^{n_i} R_{i,j}$$

and $\bar{R} = (N+1)/2$ the average of all $R_{i,j}$.
The KW test statistic $H_{KW}$ is defined [KW52] as:

$$H_{KW} = (N-1)\frac{\sum_{i=1}^{t} n_i(\bar{R}_i - \bar{R})^2}{\sum_{i=1}^{t}\sum_{j=1}^{n_i}(R_{i,j} - \bar{R})^2} \tag{7}$$

In eq. (7) the denominator $\sum_{i=1}^{t} n_i(\bar{R}_i - \bar{R})^2$ describes the variation of ranks between groups, and the numerator $\sum_{i=1}^{t}\sum_{j=1}^{n_i}(R_{i,j} - \bar{R})^2$ describes the variation of ranks in the combined sample. Intuitively, if $X_{ij}$ are all sampled from the same distribution, then all $\bar{R}_i$ are expected to be close to $\bar{R}$ and thus the statistics $H_{KW}$ should be smaller, and vice versa. Large values of the test statistic results in rejecting the null hypothesis of the KW test.

## B Further experimental results




(a) HW leakage

(b) Randomly weighted bits leakage

KW: MI: MI-HW:+ KS:× Spearman: DoM: ∘

Fig. 5: Attacking the AES SubBytes target, dropping 4 most significant bits