

F-BLEAU: Fast Black-box Leakage Estimation

Giovanni Cherubin
EPFL
giovanni.cherubin@epfl.ch

Konstantinos Chatzikokolakis
University of Athens
kostasc@di.uoa.gr

Catuscia Palamidessi
INRIA, École Polytechnique
catuscia@lix.polytechnique.fr

Abstract—We consider the problem of measuring how much a system reveals about its secret inputs. We work in the black-box setting: we assume no prior knowledge of the system’s internals, and we run the system for choices of secrets and measure its leakage from the respective outputs. Our goal is to estimate the Bayes risk, from which one can derive some of the most popular leakage measures (e.g., min-entropy leakage).

The state-of-the-art method for estimating these leakage measures is the frequentist paradigm, which approximates the system’s internals by looking at the frequencies of its inputs and outputs. Unfortunately, this does not scale for systems with large output spaces, where it would require too many input-output examples. Consequently, it also cannot be applied to systems with continuous outputs (e.g., time side channels, network traffic).

In this paper, we exploit an analogy between Machine Learning (ML) and black-box leakage estimation to show that the Bayes risk of a system can be estimated by using a class of ML methods: the *universally consistent* learning rules; these rules can exploit patterns in the input-output examples to improve the estimates’ convergence, while retaining formal optimality guarantees. We focus on a set of them, the nearest neighbor rules; we show that they significantly reduce the number of black-box queries required for a precise estimation whenever nearby outputs tend to be produced by the same secret; furthermore, some of them can tackle systems with continuous outputs. We illustrate the applicability of these techniques on both synthetic and real-world data, and we compare them with the state-of-the-art tool, leakiEst, which is based on the frequentist approach.

I. INTRODUCTION

Measuring the information leakage of a system is one of the founding pillars of security. From side-channels to biases in random number generators, quantifying how much information a system leaks about its secret inputs is crucial for preventing adversaries from exploiting it; this has been the focus of intensive research efforts in the areas of privacy and of quantitative information flow (QIF). Most approaches in the literature are based on the white-box approach, which consists in calculating analytically the channel matrix of the system, constituted by the conditional probabilities of the outputs given the secrets, and then computing the desired leakage measures (for instance, mutual information [1], min-entropy leakage [2], or g -leakage [3]). However, while one typically has white-box access to the system they want to secure, determining a system’s leakage analytically is often impractical, due to the size or complexity of its internals, or to the presence of unknown factors. These obstacles led to investigate methods for measuring a system’s leakage in a black-box manner.

Until a decade ago, the most popular measure of leakage was Shannon mutual information (MI). However, in his semi-

nal paper [2] Smith showed that MI is not appropriate to represent a realistic attacker, and proposed a notion of leakage based on Rényi min-entropy (ME) instead. Consequently, in this paper we consider the general problem of estimating the Bayes risk of a system, which is the smallest error achievable by an adversary at predicting its secret inputs given the outputs. From the Bayes risk one can derive several leakage measures, including ME and the additive and multiplicative leakage [4]. These measures are considered by the QIF community among the most fundamental notions of leakage.

To the best of our knowledge, the only existing approach for the black-box estimation of the Bayes risk comes from a classical statistical technique, which refer to as the *frequentist* paradigm. The idea is to run the system repeatedly on chosen secret inputs, and then count the relative frequencies of the secrets and respective outputs so to estimate their joint probability distribution; from this distribution, it is then possible to compute estimates of the desired leakage measure. Leak-Watch [5] and leakiEst [6], two well-known tools for black-box leakage estimation, are applications of this principle.

Unfortunately, the frequentist approach does not always scale for real-world problems: as the number of possible input and output values of the channel matrix increases, the number of examples required for this method to converge becomes too large to gather. For example, LeakWatch requires a number of examples that is much larger than the product of the size of input and output space. For the same reason, this method cannot be used for systems with continuous outputs; indeed, it cannot even be formally constructed in such a case.

Our contribution

In this paper, we show that machine learning (ML) methods can provide the necessary scalability to black-box measurements, and yet maintain formal guarantees on their estimates. By observing a fundamental equivalence between ML and black-box leakage estimation, we show that any ML rule from a certain class (the *universally consistent* rules) can be used to estimate with arbitrary precision the leakage of a system. In particular, we study rules based on the nearest neighbor principle – namely, Nearest Neighbor (NN) and k_n -NN, which exploit a metric on the output space to achieve a considerably faster convergence than frequentist approaches. In Table I we summarize the number of examples necessary for the estimators to converge, for the various systems considered in the paper. We focus on nearest neighbor methods, among the existing universally consistent rules, because: i) they are

TABLE I
NUMBER OF EXAMPLES REQUIRED FOR CONVERGENCE OF THE ESTIMATES. “X” MEANS AN ESTIMATE DID NOT CONVERGE.

System	Dataset	frequentist	NN	k_n -NN
Random	100 secrets, 100 obs.	10 070	10 070	10 070
Geometric ($\nu = 0.1$)	100 secrets, 10K obs.	35 016	333	458
Geometric ($\nu = 0.02$)	100 secrets, 10K obs.	152 904	152 698	68 058
Geometric ($\nu = 2$)	10K secrets, 1K obs.	95 500	94 204	107 707
Multimodal Geometric ($\nu = 0.1$)	100 secrets, 10K obs.	44 715	568	754
Spiky (contrived example)	2 secrets, 10K obs.	22 908	29 863	62 325
Planar Geometric $\nu = 2$	Gowalla checkins in San Francisco area	X	X	19 948
Laplacian $\nu = 2$	"	N/A	X	19 961
Blahut-Arimoto $\nu = 2$	"	1 285	1 170	1 343

The proposed tool, F-BLEAU, is the combination of frequentist, NN, and k_n -NN estimates, as an alternative to the frequentist paradigm.

simple to reason about, and ii) we can identify the class of systems for which they will excel, which happens whenever the distribution is somehow regular with respect to a metric on the output (e.g., time side channels, traffic analysis, and most mechanisms used for privacy). Moreover, some of these methods can tackle directly systems with continuous output.

We evaluate these estimators on synthetic data, where we know the true distributions and we can determine exactly when the estimates converge. Furthermore, we use them for measuring the leakage in a real dataset of users’ locations, defended with three state-of-the-art mechanisms: two geo-indistinguishability mechanisms (planar geometric and planar Laplacian) [7], and the method by Oya et al. [8], which we refer to as the Blahut-Arimoto mechanism. Crucially, the planar Laplacian is real-valued, which k_n -NN methods can tackle out-of-the box, but the frequentist method cannot. Results in both synthetic and real-world data show our methods give a strong advantage whenever there is a notion of metric in the output that can be exploited. Finally, we compare our methods with leakiEst on the problem of estimating the leakage of European passports [6], [9], and on the location privacy data.

As a further evidence of their practicality, we use them in Appendix G to measure the leakage of a time side channel in a hardware implementation of finite field exponentiation.

No Free Lunch

A central takeaway of our work is that, while all the estimators we study (including the frequentist approach) are asymptotically optimal in the number of examples, none of them can guarantee on its finite sample performance; indeed, no estimator can. This is a consequence of the No Free Lunch theorem in ML [10], which informally states that all learning rules are equivalent among the possible distributions of data. This rules out the existence of an optimal estimator.

In practice, this means that we should always evaluate several estimators, and select the one that converged faster. Fortunately, our main finding (i.e., *any universally consistent ML rule is a leakage estimator*) adds a whole new class of estimators, which one can use in practical applications.

We therefore propose a tool, F-BLEAU (Fast Black-box Leakage Estimation AUtomatic), which computes nearest neighbor and frequentist estimates, and selects the one converging faster. We release it as Open Source software¹, and we hope in the future to extend it to support several more estimators based on UC ML rules.

Nearest Neighbor rules

Nearest neighbor rules excel whenever there is a notion of metric on the output space, and the output distributions is “regular” (in the sense that it does not change too abruptly between two neighboring points). We expect this to be the case for several real-world systems, such as: side channels whose output is time, an electromagnetic signal, or power consumption; for traffic analysis on network packets; and for geographic location data. Moreover, most mechanisms used in privacy and QIF use smooth noise distributions. Suitable applications may also come from recent attacks to ML models, such as model inversion [11] and membership inference [12].

Furthermore, we observe that even when there is no metric, or when the output distribution is irregular, (e.g., a system whose internal distribution has been randomly sampled), these rules are equivalent to the frequentist approach. Indeed, the only case we observe when they are misled is when the system is crafted so that the metric contributes *against* classification (e.g., see “Spiky” example in Table I).

II. RELATED WORK

Chatzikokolakis et al. [13] introduced methods for measuring the leakage of a deterministic program in a black-box manner; these methods worked by collecting a large number of inputs and respective outputs, and by estimating the underlying probability distribution accordingly; this is what we refer to as the frequentist paradigm. A fundamental development of their work by Boreale and Paolini [14] showed that, in the absence of significant a priori information about the output distribution, no estimator does better than the exhaustive enumeration of the input domain. In line with this work, section IV will show that, as a consequence of the No Free Lunch theorem in ML,

¹<https://github.com/gchers/fbleau>.

TABLE II
SYMBOLS TABLE.

Symbol	Description
$s \in \mathbb{S}$	A secret
$o \in \mathbb{O}$	An object/black-box output
$(s, o) \in \mathbb{S} \times \mathbb{O}$	An example
$(\pi, \mathcal{C}_{s,o})$	A system, given a set of priors π and channel matrix \mathcal{C}
μ	Distribution induced by a system on $\mathbb{S} \times \mathbb{O}$
$f : \mathbb{S} \mapsto \mathbb{O}$	A classifier
ℓ	Loss function w.r.t. which we evaluate a classifier
R^f	The expected misclassification error of a classifier f
R^*	Bayes risk

no leakage estimator can claim to converge faster than any other estimator for all distributions.

The best known tools for black-box estimation of leakage measures based on the Bayes risk (e.g., min-entropy) are leakiEst [6], [15] and LeakWatch [5], [16], both based on the frequentist paradigm. The former also allows a zero-leakage test for systems with continuous outputs. In section VIII we provide a comparison of leakiEst with our proposal.

Cherubin [17] used the guarantees of nearest neighbor learning rules for estimating the information leakage (in terms of the Bayes risk) of defenses against website fingerprinting attacks in a black-box manner.

Shannon mutual information (MI) is the main alternative to the Bayes risk-based notions of leakage in the QIF literature. Although there is a relation between MI and Bayes risk [18], the corresponding models of attackers are very different: the first corresponds to an attacker who can try infinitely many times to guess the secret, while the second has only one try at his disposal [2]. Consequently, MI and Bayes-risk measures, such as ME, can give very different results: Smith [2] shows two programs that have almost the same MI, but one has an ME several orders of magnitude larger than the other one; conversely, there are examples of two programs such that ME is 0 for both, while the MI is 0 in one case and strictly positive (several bits) in the other one.

In the black-box literature, MI is usually computed by using Kernel Density Estimation, which although only guarantees asymptotic optimality under smoothness assumptions on the distributions. On the other hand, the ML literature offered developments in this area: Belghazi et al. [19] proposed an MI lower bound estimator based on deep neural networks, and proved its consistency (i.e., it converges to the true MI value asymptotically). Similarly, other works constructed MI variational lower bounds [20], [21].

III. PRELIMINARIES

We define a system, and show that its leakage can be expressed in terms of the Bayes risk. We then introduce ML notions, which we will later use to estimate the Bayes risk.

A. Notation

We consider a system $(\pi, \mathcal{C}_{s,o})$, that associates a secret input s to an observation (or object) o in a possibly randomized way. The system is defined by a set of prior probabilities $\pi(s) :=$

$P(s)$, $s \in \mathbb{S}$, and a channel matrix \mathcal{C} of size $|\mathbb{S}| \times |\mathbb{O}|$, for which $\mathcal{C}_{s,o} := P(o|s)$ for $s \in \mathbb{S}$ and $o \in \mathbb{O}$. We call $\mathbb{S} \times \mathbb{O}$ the example space. We assume the system does not change over time; for us, \mathbb{S} is finite, and \mathbb{O} is finite unless otherwise stated.

B. Leakage Measures

The state-of-the-art in QIF is represented by the leakage measures based on *g-vulnerability*, a family whose most representative member is min-vulnerability [2], the complement of the Bayes risk. This paper is concerned with finding tight estimates of the Bayes risk, which can then be used to estimate the appropriate leakage measure.

a) *Bayes risk*: The Bayes risk, R^* , is the error of the optimal (idealized) classifier for the task of predicting a secret s given an observation o output by a system. It is defined with respect to a loss function $\ell : \mathbb{S} \times \mathbb{S} \mapsto \mathbb{R}_{\geq 0}$, where $\ell(s, s')$ is the risk of an adversary predicting s' for an observation o , when its actual secret is s . We focus on the 0-1 loss function, $\ell(s, s') := I(s \neq s')$, taking value 1 if $s \neq s'$, 0 otherwise. The Bayes risk of a system $(\pi, \mathcal{C}_{s,o})$ is defined as:

$$R^* := 1 - \sum_{o \in \mathbb{O}} \max_{s \in \mathbb{S}} \mathcal{C}_{s,o} \pi(s). \quad (1)$$

b) *Random guessing*: A baseline for evaluating a system is the error committed by an idealized adversary who knows priors but has no access to the channel, and who's best strategy is to always output the secret with the highest prior. We call the error of this adversary random guessing error:

$$R^\pi := 1 - \max_{s \in \mathbb{S}} \pi(s). \quad (2)$$

C. Black-box estimation of R^*

This paper is concerned with estimating the Bayes risk given n examples sampled from the joint distribution μ on $\mathbb{S} \times \mathbb{O}$ generated by $(\pi, \mathcal{C}_{s,o})$. By running the system n times on secrets $s_1, \dots, s_n \in \mathbb{S}$, chosen according to π , we generate a sequence of corresponding outputs o_1, \dots, o_n , thus forming a *training set*² of examples $\{(o_1, s_1), \dots, (o_n, s_n)\}$. From these data, we aim to make an estimate close to the real Bayes risk.

D. Learning Rules

We introduce ML rules (or, simply, *learning rules*), which are algorithms for selecting a classifier given a set of training examples. In this paper, we will use the error of some ML rules as an estimator of the Bayes risk.

Let $\mathcal{F} := \{f \mid f : \mathbb{O} \mapsto \mathbb{S}\}$ be a set of classifiers. A learning rule is a possibly randomized algorithm that, given a training set $\{(o_1, s_1), \dots, (o_n, s_n)\}$, returns a classifier $f \in \mathcal{F}$, with the goal of minimizing the expected loss $\mathbb{E} \ell(f(o), s)$ for a new example (o, s) sampled from μ [22]. In the case of the 0-1 loss function, the expected loss coincides with the *expected probability of error* (*expected error* for short), and if μ is

²In line with the ML literature, we call the training or test “set” what is technically a multiset; also, we loosely use the set notation “{ }” for both sets and multisets when the nature of the object is clear from the context.

generated by a system $(\pi, \mathcal{C}_{s,o})$, then the expected error of a classifier $f : \mathbb{O} \mapsto \mathbb{S}$ is:

$$R^f = 1 - \sum_{o \in \mathbb{O}} \mathcal{C}_{f(o),o} \pi(f(o)) \quad (3)$$

where $f(o)$ is the secret predicted for object o . If \mathbb{O} is infinite (and μ is continuous) the summation is replaced by an integral.

E. Frequentist estimate of R^*

The frequentist paradigm [13] for measuring the leakage of a channel consists in estimating the probabilities $\mathcal{C}_{s,o}$ by counting their frequency in the training data $(o_1, s_1), \dots, (o_n, s_n)$:

$$P(o|s) \approx \hat{\mathcal{C}}_{s,o} := \frac{|i : o_i = o, s_i = s|}{|i : s_i = s|}. \quad (4)$$

We can obtain the frequentist error from Equation 3:

$$R^{Freq} = 1 - \sum_o \mathcal{C}_{f^{Freq}(o),o} \pi(f^{Freq}(o)) \quad (5)$$

where f^{Freq} is the frequentist classifier, namely:

$$f^{Freq}(o) = \begin{cases} \operatorname{argmax}_s (\hat{\mathcal{C}}_{s,o} \hat{\pi}(s)) & \text{if } o \text{ in training data} \\ \operatorname{argmax}_s \hat{\pi}(s) & \text{otherwise,} \end{cases} \quad (6)$$

where $\hat{\pi}$ is estimated from the examples: $\hat{\pi}(s) = |i : s_i = s|/n$.

Consider a finite example space $\mathbb{S} \times \mathbb{O}$. Provided with enough examples, the frequentist approach always converges: clearly, $\hat{\mathcal{C}} \rightarrow \mathcal{C}$ as $n \rightarrow \infty$, because events' frequencies converge to their probabilities by the Law of Large Numbers.

However, there is a fundamental issue with this approach. Given a training set $\{(o_1, s_1), \dots, (o_n, s_n)\}$, the frequentist classifier can tell something meaningful (i.e., better than random guessing) for an object $o \in \mathbb{O}$, only as long as o appeared in the training set; but, for very large systems (e.g., those with a large object space), the probability of observing an example for each object becomes small, and the frequentist classifier approaches *random guessing*. We study this matter further in subsection VI-D and Appendix E.

IV. NO FREE LUNCH IN LEARNING

The frequentist approach performs well only for objects it has seen in the training data; in the next section, we will introduce estimators that aim to provide good predictions even for unseen objects. However, we shall first answer an important question: is there an estimator that is “optimal” for all systems?

A negative answer to this question is given by the so-called “No Free Lunch” (NFL) theorem by Wolpert [10]. The theorem is formulated for the expected loss of a learning rule on *unseen* objects (i.e., that were not in the training data), which is referred to as the off-training-set (OTS) loss.

Theorem 1 (No Free Lunch). *Let A_1 and A_2 be two learning rules, ℓ a cost function, and μ a distribution on $\mathbb{S} \times \mathbb{O}$. We indicate by $\mathbb{E}_i(\ell \mid \mu, n)$ the OTS loss of A_i given μ and n , where the expectation is computed over all the possible training sets of size n sampled from μ . Then, if we take the uniform average among all possible distributions μ , we have*

$$\mathbb{E}_1(\ell \mid \mu, n) = \mathbb{E}_2(\ell \mid \mu, n). \quad (7)$$

Intuitively, the NFL theorem says that, if all distributions (and, therefore, all channel matrices) are equally likely, then all learning algorithms are equivalent. Remarkably, this holds for *any* strategy, even if one of the rules is random guessing.

An important implication of this for our purposes is that for every two learning rules A and B there will always exist some system for which rule A converges faster than B , and vice versa there will be a system for which B outperforms A .

From the practical perspective of black-box security, this demonstrates that we should always test several estimators and select the one that converges faster. Fortunately, the connection between ML and black-box security we highlight in this paper results in the discovery of a whole class of new estimators.

V. MACHINE LEARNING ESTIMATES OF THE BAYES RISK

In this section, we define the notion of a *universally consistent* learning rule, and show that the error of a classifier selected according to such a rule can be used for estimating the Bayes risk. Then, we introduce various universally consistent rules based on the nearest neighbor principle.

Throughout the section, we use interchangeably a system $(\pi, \mathcal{C}_{s,o})$ and its corresponding joint distribution μ on $\mathbb{S} \times \mathbb{O}$. Note that there is a one-to-one correspondence between them.

A. Universally Consistent Rules

Consider a distribution μ and a learning rule A selecting a classifier $f_n \in \mathcal{F}$ according to n training examples sampled from μ . Intuitively, as the available training data increases, we would like the expected error of f_n for a new example (o, s) sampled from μ to be minimized (i.e., to get close the Bayes risk). The following definition captures this intuition.

Definition 1 (Consistent Learning Rule). *Let μ be a distribution on $\mathbb{S} \times \mathbb{O}$ and let A be a learning rule. Let $f_n \in \mathcal{F}$ be a classifier selected by A using n training examples sampled from μ . Let $(\pi, \mathcal{C}_{s,o})$ be the system corresponding to μ , and let R^{f_n} be the expected error of f_n , as defined by (3). We say that A is consistent if $R^{f_n} \rightarrow R^*$ as $n \rightarrow \infty$.*

The next definition strengthens this property, by asking the rule to be consistent for all distributions:

Definition 2 (Universally Consistent (UC) Learning Rule). *A learning rule is universally consistent if it is consistent for any distribution μ on $\mathbb{S} \times \mathbb{O}$.*

By this definition, the expected error of a classifier selected according to a universally consistent rule is also an estimator of the Bayes risk, since it converges to R^* as $n \rightarrow \infty$.

In the rest of this section we introduce Bayes risk estimates based on universally consistent nearest neighbor rules; they are summarized in Table III together with their guarantees.

B. NN estimate

The Nearest Neighbor (NN) is one of the simplest ML classifiers: given a training set and a new object o , it predicts the secret of its closest training observation (*nearest neighbor*). It is defined both for finite and infinite object spaces, although it is UC only in the first case.

TABLE III
ESTIMATES' GUARANTEES AS $n \rightarrow \infty$

Method	Guarantee	Space \mathbb{O}
frequentist	$\rightarrow R^*$	finite
NN	$\rightarrow R^*$	finite
k_n -NN	$\rightarrow R^*$	infinite, (d, \mathbb{O}) separable
NN Bound ⁺	$\leq R^*$	infinite, (d, \mathbb{O}) separable

⁺NN Bound is discussed in Appendix A.

We introduce a formulation of NN, which can be seen as an extension of the frequentist approach, that takes into account *ties* (i.e., neighbors that are equally close to the new object o), and which guarantees consistency when \mathbb{O} is finite.

Consider a training set $\{(o_1, s_1), \dots, (o_n, s_n)\}$, an object o , and a distance metric $d : \mathbb{O} \times \mathbb{O} \mapsto \mathbb{R}_{\geq 0}$. The NN classifier predicts a secret for o by taking a majority vote over the set of secrets whose objects have the smallest distance to o . Formally, let $I_{\min}(o) = \{i \mid d(o, o_i) = \min_{j=1 \dots n} d(o, o_j)\}$ and define:

$$NN(o) = s_{h(o)} \quad (8)$$

where

$$h(o) = \operatorname{argmax}_{i \in I_{\min}(o)} |\{j \in I_{\min}(o) \mid s_j = s_i\}|. \quad (9)$$

We show that NN is universally consistent for finite $\mathbb{S} \times \mathbb{O}$.

Theorem 2 (Universal consistency of NN). *Consider a distribution on $\mathbb{S} \times \mathbb{O}$, where \mathbb{S} and \mathbb{O} are finite. Let R_n^{NN} be the expected error of the NN classifier for a new observation o . As the number of training examples $n \rightarrow \infty$:*

$$R_n^{NN} \rightarrow R^*. \quad (10)$$

Sketch proof. For an observation o that appears in the training set, the NN classifier is equivalent to the frequentist approach. For a finite space $\mathbb{S} \times \mathbb{O}$, as $n \rightarrow \infty$, the probability that the training set contains all $o \in \mathbb{O}$ approaches 1. Thus, the NN rule is asymptotically (in n) equivalent to the frequentist approach, which means its error also converges to R^* . \square

C. k_n -NN estimate

Whilst NN guarantees universal consistency in finite example spaces, this does not hold for infinite \mathbb{O} . In this case, we can achieve universal consistency with the k -NN classifier, an extension of NN, for appropriate choices of the parameter k .

The k -NN classifier takes a majority vote among the secrets of its neighbors. Breaking ties in the k -NN definition requires more care than with NN. In the literature, this is generally done via strategies that add randomness or arbitrariness to the choice (e.g., if two neighbors have the same distance, select the one with the smallest index in the training data) [23]. We use a novel tie-breaking strategy, which takes into account ties, but gives more importance to the closest neighbors. In early experiments, we observed this strategy had a faster convergence than standard approaches.

Consider a training set $\{(o_1, s_1), \dots, (o_n, s_n)\}$, an object to predict o , and some metric $d : \mathbb{O} \times \mathbb{O} \mapsto \mathbb{R}_{\geq 0}$. Let $o_{(i)}$ denote the i -th closest object to o , and $s_{(i)}$ its respective secret. If ties do not occur after the k -th neighbor (i.e., if $d(o, o_{(k)}) \neq d(o, o_{(k+1)})$), then k -NN outputs the most frequent among the secrets of the first k neighbors:

$$k\text{-NN}(o) = s_{h(o)} \quad (11)$$

where

$$h(o) = \operatorname{argmax}_{i=1, \dots, k} |\{j \in I_{\min}(o) \mid s_{(j)} = s_{(i)}\}|. \quad (12)$$

If ties exist after the k -th neighbor, that is, for $k' \leq k < k''$:

$$d(o, o_{(k')}) = \dots = d(o, o_{(k)}) = \dots = d(o, o_{(k'')}) , \quad (13)$$

we proceed as follows. Let \hat{s} be the most frequent secret in $\{s_{(k')}, \dots, s_{(k'')}\}$; k -NN predicts the most frequent secret in the following multiset, truncated at the tail to have size k :

$$s_{(1)}, s_{(2)}, \dots, s_{(k'-1)}, \hat{s}, \hat{s}, \dots, \hat{s}.$$

We now define k_n -NN, a universally consistent learning rule that selects a k -NN classifier for a training set of n examples by choosing k as a function of n .

Definition 3 (k_n -NN rule). *Given a training set of n examples, the k_n -NN rule selects a k -NN classifier, where k is chosen such that $k_n \rightarrow \infty$ and $k_n/n \rightarrow 0$ as $n \rightarrow \infty$.*

Stone proved that k_n -NN is universally consistent [24]:

Theorem 3 (Universal consistency of the k_n -NN rule). *Consider a probability distribution μ on the example space $\mathbb{S} \times \mathbb{O}$, where μ has a density. Select a distance metric d such that (d, \mathbb{O}) is separable³. Then the expected error of the k_n -NN rule converges to R^* as $n \rightarrow \infty$.*

This holds for any distance metric. In our experiments, we will use the Euclidean distance, and we will evaluate k_n -NN rules for $k_n = \log n$ (natural logarithm) and $k_n = \log_{10} n$.

The ML literature is rich of UC rules and other useful tools for black-box security; we list some of them in Appendix A.

VI. EVALUATION ON SYNTHETIC DATA

In this section we evaluate our estimates on several synthetic systems for which the channel matrix is known. For each system, we sample n examples from its distribution, and then compute the estimate on the whole object space as in Equation 3; this is possible because \mathbb{O} is finite. Since for synthetic data we know the real Bayes risk, we can measure how many examples are required for the convergence of each estimate. We do this as follows: let R_n^f be an estimate of R^* , trained on a dataset of n examples. We say the estimate δ -converged to R^* after n examples if its relative change from R^* is smaller than δ :

$$\frac{|R_n^f - R^*|}{R^*} < \delta. \quad (14)$$

³A separable space is a space containing a countable dense subset; e.g., finite spaces and the space of q -dimensional vectors \mathbb{R}^q with Euclidean metric.

TABLE IV
SYNTHETIC SYSTEMS.

Name	Privacy parameter	$ \mathbb{S} $	$ \mathbb{O} $	R^*
Geometric	1.0	100	10K	~ 0
Geometric	0.1	100	10K	0.007
Geometric	0.01	100	10K	0.600
Geometric	0.2	100	1K	0.364
Geometric	0.02	100	10K	0.364
Geometric	0.002	100	100K	0.364
Geometric	2	100K	100K	0.238
Geometric	2	100K	10K	0.924
Multimodal	1.0	100	10K	0.450
Multimodal	0.1	100	10K	0.456
Multimodal	0.01	100	10K	0.797
Spiky	N/A	2	10K	0
Random	N/A	100	100	0.979

While relative change has the advantage of taking into account the magnitude of the compared values, it is not defined when the denominator is 0; therefore, when $R^* \approx 0$ (Table IV), we verify convergence with the absolute change:

$$|R_n^f - R^*| < \delta. \quad (15)$$

The systems used in our experiments are briefly discussed in this section, and summarized in Table IV; we detail them in Appendix B. A uniform prior is assumed in all cases.

A. Geometric systems

We first consider systems generated by adding geometric noise to the secret, one of the typical mechanisms used to implement differential privacy [25]. Their channel matrix has the following form:

$$\mathcal{C}_{s,o} = P(o | s) = \lambda \exp(-\nu |g(s) - o|), \quad (16)$$

where ν is a privacy parameter, λ a normalization factor, and g a function $\mathbb{S} \mapsto \mathbb{O}$; a detailed description of these systems is given in Appendix B.

We consider the following three parameters:

- the privacy parameter ν ,
- the ratio $|\mathbb{O}|/|\mathbb{S}|$, and
- the size of the secret space $|\mathbb{S}|$.

We vary each of these parameters one at a time, to isolate their effect on the convergence rate.

1) *Variation of the privacy parameter ν* : We fix $|\mathbb{S}| = 100$, $|\mathbb{O}| = 10K$, and we consider three cases $\nu = 1.0$, $\nu = 0.1$ and $\nu = 0.01$. The results for the estimation of the Bayes risk and the convergence rate are illustrated in Figure 1 and Table V respectively. In the table, results are reported for convergence level $\delta \in \{0.1, 0.05, 0.01, 0.005\}$; an “X” means a particular estimate did not converge within 500K examples; a missing row for a certain δ means no estimate converged.

The results indicate that the nearest neighbor methods have a much faster convergence than the standard frequentist approach, particularly when dealing with large systems. The reason is that geometric systems have a regular behavior with respect to the Euclidean metric, which can be exploited by NN and k_n -NN to make good predictions for unseen objects.

TABLE V
CONVERGENCE OF THE ESTIMATES WHEN VARYING ν , FIXED $|\mathbb{S}| \times |\mathbb{O}| = 100 \times 10K$

System	δ	Freq.	NN	k_n -NN	
				\log_{10}	\log
Geometric $\nu = 1.0$	0.1	1 994	267	396	679
	0.05	4 216	325	458	781
	0.01	19 828	425	633	899
	0.005	38 621	439	698	904
Geometric $\nu = 0.1$	0.1	18 110	269	396	673
	0.05	35 016	333	458	768
	0.01	127 206	439	633	899
	0.005	211 742	4 844	698	904
Geometric $\nu = 0.01$	0.1	105 453	103 357	99 852	34 243
	0.05	205 824	205 266	205 263	199 604

TABLE VI
CONVERGENCE OF THE ESTIMATES WHEN VARYING $|\mathbb{O}|/|\mathbb{S}|$.

System	δ	Freq.	NN	k_n -NN	
				\log_{10}	\log
Geometric 100x1K $\nu = 0.2$	0.1	8 679	8 707	7 108	2 505
	0.05	14 823	14 853	14 853	7 673
	0.01	51 694	60 796	60 796	60 796
	0.005	71 469	71 469	71 469	71 469
Geometric 100x10K $\nu = 0.02$	0.1	85 912	85 644	71 003	11 197
	0.05	152 904	152 698	151 153	68 058
Geometric 100x100K $\nu = 0.002$	0.1	X	X	413 974	2 967

2) *Variation of the ratio $|\mathbb{O}|/|\mathbb{S}|$* : Now we fix $|\mathbb{S}| = 100$, and we consider three cases $|\mathbb{O}|/|\mathbb{S}| = 10$, $|\mathbb{O}|/|\mathbb{S}| = 100$, and $|\mathbb{O}|/|\mathbb{S}| = 1K$. (Note that we want to keep the ratio ν/Δ_g fixed, see Appendix B; as a consequence ν has to vary: we set ν to 0.2, 0.02, and 0.002, respectively.) Results in Figure 2 and Table VI show how the nearest neighbor methods become much better than the frequentist approach as $|\mathbb{O}|/|\mathbb{S}|$ increases. This is because the larger the object space, the larger the number of unseen objects at the moment of classification, and the more the frequentist approach has to rely on random guessing. The nearest neighbor methods are not that much affected because they can rely on the proximity to outputs already classified.

3) *Case $|\mathbb{S}| \geq |\mathbb{O}|$* : We fix $\nu = 2$, and we consider two cases: $|\mathbb{S}| = |\mathbb{O}|$ and $|\mathbb{S}| > |\mathbb{O}|$. It should be noted that the formulation of geometric systems prohibits the number of secrets to exceed the number of outputs; for this reason, in the system $|\mathbb{S}| > |\mathbb{O}|$ some secrets are associated with the same distribution over the output space (Appendix B).

The results in Figure 3 and Table VII indicate that NN and frequentist are mostly equivalent: this is because they both need to observe at least one example for each secret. k_n -NN rules, on the other hand, show poor performances, due to the fact that they would need at least k_n examples for each secret. A natural extension of our work is to look at notions of metric also in the secret space for improving convergence.

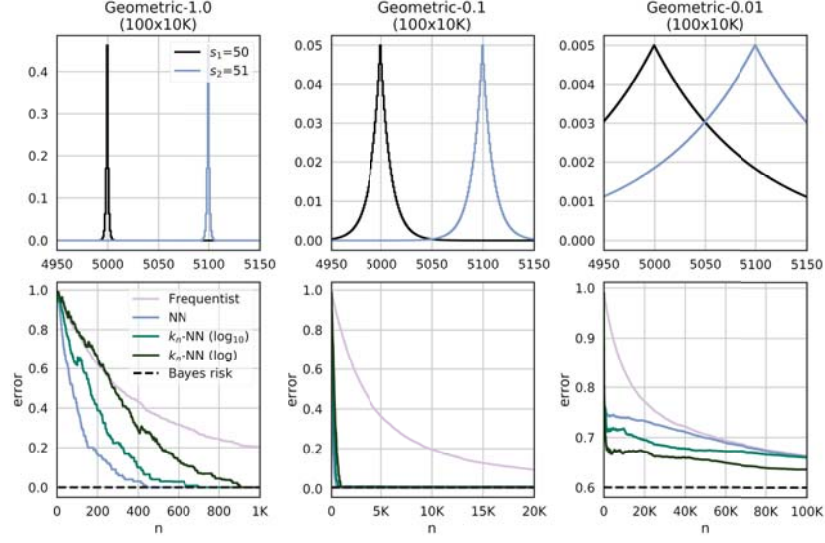


Fig. 1. Estimates' convergence for geometric systems when varying their privacy parameter ν . The respective distributions are shown in the top figure for two adjacent secrets $s_1 \sim s_2$.

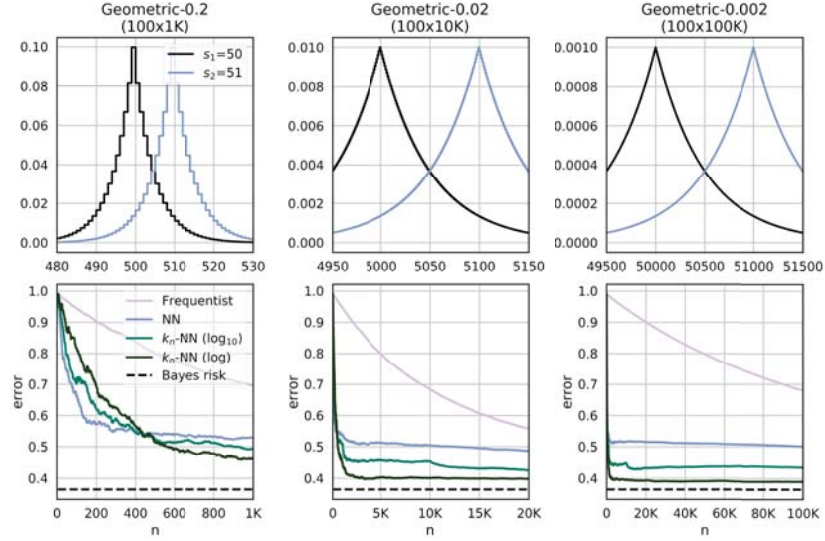


Fig. 2. Estimates' convergence for geometric systems when varying the ratio $|O|/|S|$. The respective distributions are shown in the top figure for two adjacent secrets $s_1 \sim s_2$.

B. Multimodal geometric system

We now evaluate the estimators on systems with a multimodal distribution. In particular, we create multimodal geometric systems by summing two geometric probability distributions, appropriately normalized and shifted by some parameter. We provide the details of this distribution in Appendix B.

1) *Evaluation:* Results are reported in Figure 4. As expected, we observe that nearest neighbor rules improve on the frequentist approach; the reason is that, even for multimodal distributions, there exists a metric on the outputs which they

can exploit. Detailed δ -convergence results are in Appendix C.

C. Spiky system: When k_n -NN rules fail

Nearest neighbor rules take advantage of the metric on the object space to improve their convergence considerably. However, as a consequence of the NFL theorem, there exist systems for which the frequentist approach outperforms NN and k_n -NN. Investigating the form of such systems is important to understand when these methods fail.

We craft one such system, the Spiky system, where the metric misleads predictions. The Spiky system is such that

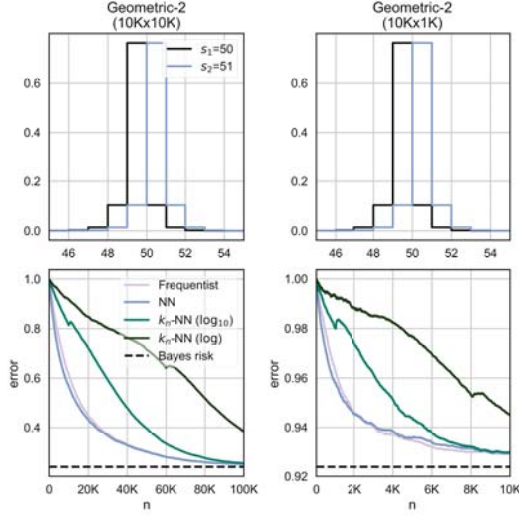


Fig. 3. Estimates' convergence for geometric systems when $|\mathcal{S}| \geq |\mathcal{O}|$. The distributions are shown in the top figure for two adjacent secrets $s_1 \sim s_2$. In the case $|\mathcal{S}| > |\mathcal{O}|$ (right) there are $10 = 10K/1K$ identical distributions that coincide on s_1 , and 10 identical distributions on s_2 .

TABLE VII
CONVERGENCE OF THE ESTIMATES WHEN $|\mathcal{S}| \geq |\mathcal{O}|$, $\nu = 2$.

System	δ	Freq.	NN	k_n -NN	
				\log_{10}	\log
Geometric 10Kx10K	0.1	74 501	73 085	88 296	140 618
	0.05	95 500	94 204	107 707	155 403
	0.01	137 099	137 348	144 846	192 014
	0.005	153 370	153 370	159 075	203 363
Geometric 10Kx1K	0.1	5	5	5	5
	0.05	721	514	2 309	5 977
	0.01	5 595	6 171	7 330	12 354
	0.005	10 770	10 797	11 037	14 575

neighboring points are associated with different secret. This means that NN and k_n -NN rules will tend to predict the wrong secret, until enough examples are available. We detail its construction in Appendix B.

a) Discussion: We conducted experiments for a Spiky system of size $|\mathcal{O}| = 10K$. Results in Figure 5 confirm the hypothesis: nearest neighbor rules are misled for this system.

Interestingly, while the NN estimate keeps decreasing as the number of examples n increases, there is a certain range of n 's where the k_n -NN estimates become worse than random guessing. Intuitively, this is because when n becomes larger than $|\mathcal{O}|$, all elements in \mathcal{O} tend to be covered by the examples. For every $i \in \mathcal{O}$ there are two neighbors, $i - 1$ and $i + 1$, that belong to the class opposite to the one of i , so if k is not too small with respect to n , it is likely that in the miset of the k closest neighbors of i , the number of $i - 1$'s and $i + 1$'s exceeds the number of i 's, which means that i will be misclassified. As n increases, however, the ratio between k and the number of i 's in the examples tends to decrease (because $k/n \rightarrow 0$ as $n \rightarrow \infty$), hence at some point we will have enough i 's to win

the majority vote in the k neighbors (i 's are considered before than $i - 1$'s and $i + 1$'s, by the nearest neighbor definition) so i will not be misclassified anymore.

Concerning the comparison between the NN and frequentist estimates, we can do it analytically. We start by computing the expected error of the NN method on the spiky system in terms of the number of training examples n . Let T^n be a training set of examples of size n . Given a new object i , let us consider the NN estimate $r_n(i)$ of r^* for i , i.e., the expected probability of error in the classification of i . This is the probability that the element o closest to i that appears in the training set has odd distance from i (i.e., $d(i, o) = 2\ell + 1$, for some natural number ℓ). Namely it is the probability that:

- i is not in the training data but either $i + 1$ or $i - 1$ are, or
- $i, i \pm 1, i \pm 2$ are not in the training data but either $i + 3$ or $i - 3$ are, or
- ...etc.

Hence we have:

$$r(i) = P(d(i, o) = 2\ell + 1) = \quad (17)$$

$$= P(i \notin T^n, i + 1 \in T^n) + P(i \notin T^n, i - 1 \in T^n) + \dots \quad (18)$$

$$= 2 \cdot \sum_{\ell=0}^{q/4-1} a^{4\ell+1} (1 - a), \quad (19)$$

where $a = (1 - 1/q)^n$ is the probability that an element $e \in \mathcal{O}$ does not occur in any of the n examples of the training set. (Thus $a^{4\ell+1}$ represents the probability that none of the elements $i, i \pm 1, i \pm 2, i \pm 2s$, with $\ell = 2s$, appear in the training set, and $1 - a$ represents the probability that the element $2s + 1$ (resp. $2s - 1$) appears in the training set.) By using the result of the geometric series

$$\sum_{t=0}^m a^t = \frac{1 - a^{m+1}}{1 - a}, \quad (20)$$

we obtain:

$$r_n(i) = 2a \frac{1 - a^q}{(1 + a^2)(1 + a)}. \quad (21)$$

Since we assume that the distribution on \mathcal{O} is uniform, we have $R_n^{NN} = r_n(i)$.

We want to study how the error estimate depends on the relative size of the training set with respect to the size of \mathcal{O} . Hence, let $x = n/q$. Then we have $a = (1 - 1/q)^{qx}$, which, for large q , becomes $a \approx e^{-x}$. Therefore:

$$R_x^{NN} \approx 2e^{-x} \frac{1 - e^{-qx}}{(1 + e^{-2x})(1 + e^{-x})}. \quad (22)$$

It is easy to see that $R_x^{NN} \rightarrow 1/2$ for $x \rightarrow 0$, and $R_x^{NN} \rightarrow 0$ for $x \rightarrow \infty$, as expected.

Consider now the frequentist estimate R_x^{Freq} . In this case, given an element $i \in \mathcal{O}$, the classification is done correctly if i appears in the training set. Otherwise, we do random guessing, which gives a correct or wrong classification with

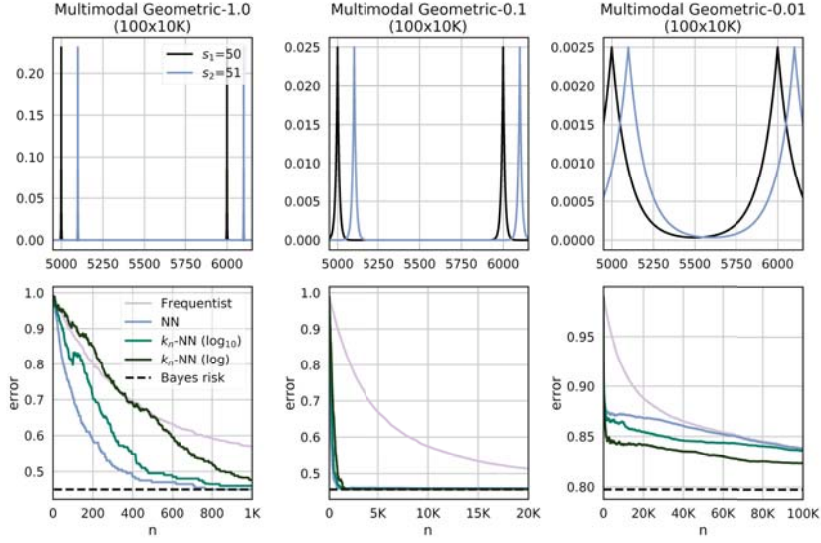


Fig. 4. Estimates' convergence for multimodal geometric systems when varying the privacy parameter ν . The distributions are shown in the top figure for two adjacent secrets $s_1 \sim s_2$.

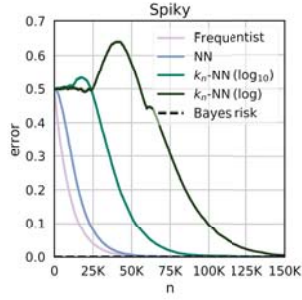


Fig. 5. Estimates' convergence for a Spiky system (2x10K).

equal probability. Only the latter case contributes to the probability of error, hence the error estimate is half the probability expectation that i does not belong to the training set:

$$R_x^{Freq} = \frac{1}{2} \left(1 - \frac{1}{q}\right)^n \approx \frac{1}{2} e^{-x} \quad (23)$$

Therefore, R_x^{NN} is always above R_x^{Freq} .

D. Random System

In the previous sections, we have seen cases when our methods greatly outperform the frequentist approach, and a contrived system example for which they fail. We now consider a system generated randomly to evaluate their performances for an “average” system.

a) System description: We construct the channel matrix of a Random system by drawing its elements from the uniform distribution, $C_{s,o} \leftarrow^s Uni(0, 1)$, and normalizing its rows.

TABLE VIII
RANDOM: EXAMPLES REQUIRED FOR δ -CONVERGENCE.

δ	Freq.	NN	k_n -NN	
			\log_{10}	\log
0.05	5	5	5	5
0.01	82	139	202	500
0.005	10 070	10 070	10 070	10 070

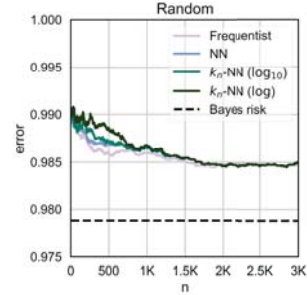


Fig. 6. Estimates' convergence for a Random system (100×100).

b) Evaluation: We consider a Random system with $|\mathbb{S}| = |\mathbb{O}| = 100$ and count the number of examples required for δ -convergence, for many δ 's. Table VIII reports the results.

The frequentist estimate is slightly better than NN and k_n -NN for $\delta = 0.01$. However, for stricter convergence requirements ($\delta = 0.001$), all the methods require the same (large) number of examples. Figure 6 shows that indeed the methods begin to converge similarly already after 1K examples.

c) Discussion: Results showed that nearest neighbor estimates require significantly fewer examples than the frequentist

approach when dealing with medium or large systems; however, they are generally equivalent to the frequentist approach in the case of small systems.

To better understand why this is the case, we provide a crude approximation of the frequentist Bayes risk estimate.

$$R_n^{Freq} \approx R^* \left(1 - \left(1 - \frac{1}{|\mathbb{O}|} \right)^n \right) + R^\pi \left(1 - \frac{1}{|\mathbb{O}|} \right)^n. \quad (24)$$

This approximation, derived and studied in Appendix E, makes the very strong assumption that all objects are equally likely, i.e.: $P(o) = \frac{1}{|\mathbb{O}|}$. However, this is enough to give us an insight on the performance of the frequentist approach: $\left(1 - \frac{1}{|\mathbb{O}|} \right)^n$ is the probability that some object does not appear within a training set of size n . This weighs the value of the frequentist estimate between the optimal R^* , used when the object appears in the training data, and random guessing R^π : while the estimate converges asymptotically to the Bayes risk, the probability of observing an object – often related to the size $|\mathbb{O}|$, has a major influence on its convergence rate.

VII. APPLICATION TO LOCATION PRIVACY

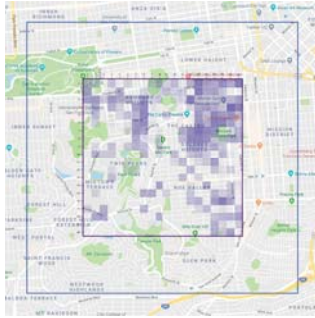


Fig. 7. Area of San Francisco considered for the experiments. The input locations correspond to the inner square, the output locations to the outer one. The colored cells represent the distribution of the Gowalla checkins.

We show that F-BLEAU can be successfully applied to estimate the degree of protection provided by mechanisms such as those used in location privacy. Since the purpose of this paper is to evaluate the precision of F-BLEAU, we consider basic mechanisms for which the Bayes risk can also be computed directly. Of course, the intended applications of F-BLEAU are mechanisms or situations where the Bayes risk *cannot* be computed directly, either because this is too complicated, or because of the presence of unknown factors. Examples abound; for instance, the availability of additional information, like the presence of points of interest (e.g., shops, churches), or geographical characteristics of the area (e.g., roads, lakes) can affect the Bayes risk in ways that are impossible to evaluate formally.

We will consider the planar Laplacian and the planar geometric, which are the typical mechanisms used to obtain geo-indistinguishability [7], and one of the optimal mechanisms proposed by Oya et al. [8] as a refinement of the optimal mechanism by Shokri et al. [26]. The construction of the last relies

on an algorithm that was independently proposed by Blahut and by Arimoto to solve the information theory problem of achieving an optimal trade-off between the minimization of the distortion rate and the mutual information [27]. From now on, we shall refer to this as the Blahut-Arimoto mechanism. Note that the Laplacian is a continuous mechanism (i.e., its outputs are on a continuous plane); the other two are discrete.

In these experiments we also deploy the method that F-BLEAU uses in practice to compute the estimate of the Bayes risk: we first split the data into a training set and a hold-out set; then, for an increasing number of examples $n = 1, 2, \dots$ we train the classifier on the first n examples on the training set, and then estimate its error on the hold-out set.

A. The Gowalla dataset

We consider real location data from the Gowalla dataset [28], [29], which contains users' checkins and their geographical location. We use a squared area in San Francisco, centered in the coordinates (37.755, -122.440), and extending for 1.5 Km in each direction. This input area corresponds to the inner (purple) square in Figure 7. We discretize the input using a grid of 20×20 cells of size 150×150 Sq m; the secret space \mathbb{S} of the system thus consists of 400 locations. The prior distribution on the secrets is derived from the Gowalla checkins, and it is represented in Figure 7 by the different color intensities on the input grid. The output area is represented in Figure 7 by the outer (blue) square. It extends 1050 m (7 cells) more than the input square on every side. We consider a larger area for the output because the planar Laplacian and Geometric naturally expand outside the input square.⁴ Since the planar Laplacian is continuous, its output domain \mathbb{O} is constituted by all the points of the outer square. As for the planar Geometric and the Blahut-Arimoto mechanisms, which are discrete, we divide the output square in a grid of 340×340 cells of size 15×15 Sq m; therefore, $|\mathbb{O}| = 340 \times 340 = 115,600$.

B. Defenses

The *planar Geometric* mechanism is characterized by a channel matrix $C_{s,o}$, representing the conditional probability of reporting the location o when the true location is s :

$$C_{s,o} = \lambda \exp \left(-\frac{\ln \nu}{100} d(s, o) \right), \quad (25)$$

where ν is a parameter controlling the level of noise, λ is a normalization factor, and d is the Euclidean distance.

The *planar Laplacian* is defined by the same equation, except that o belongs to a continuous domain, and the equation defines a probability density function.

As for the *Blahut-Arimoto*, it is obtained as the result of an iterative algorithm, whose definition can be found in [27].

⁴In fact these functions distribute the probability on the infinite plane, but on locations very distant from the origin the probability becomes negligible.

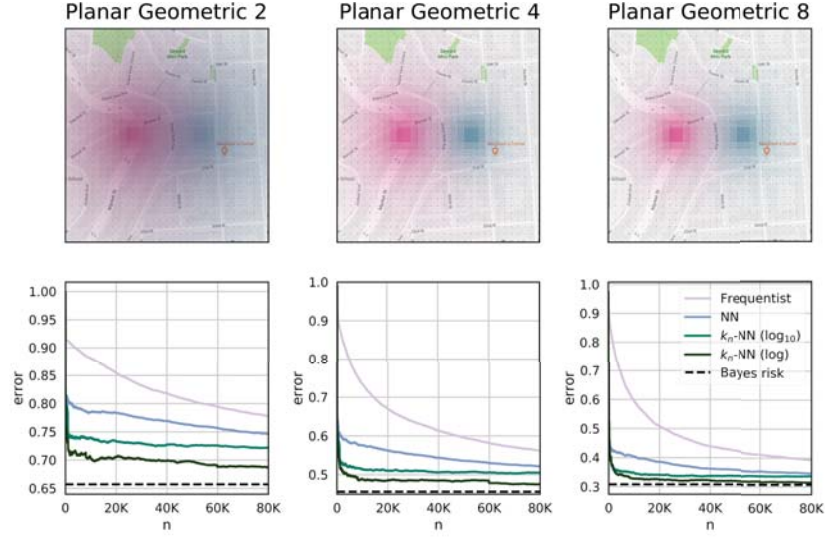


Fig. 8. Estimates' convergence speed for the planar Geometric defense applied to the Gowalla dataset, for $\nu = 2$, $\nu = 4$ and $\nu = 8$, respectively. Above each graph is represented the distribution of the geometric noise for two adjacent input cells.

C. Results

We evaluated the estimates' convergence as a function of the number of training examples n and for different values of the noise level: $\nu = \{2, 4, 8\}$. We randomly split the dataset (100K examples) into training (75%) and hold-out (25%) sets, and then evaluated the convergence of the estimators on an increasing number of training examples, 5, 6, ..., 75K.

Results for the geometric noise (Figure 8) indicate faster convergence when ν is higher (which means less noise and lower Bayes risk), in line with the results for the synthetic systems of the previous section. In all cases, the nearest neighbor methods outperform the frequentist one, as we expected given the presence of a large number of outputs. Table IX shows the number of examples required to achieve δ -convergence from the Bayes risk. The symbol "X" means we did not achieve a certain level of approximation with 75K examples.

The corresponding results for the Laplacian noise are shown in Figure 8 and in Table X. In this case, the frequentist approach is not applicable, but the k_n -NN rule can still approximate the Bayes risk for some approximation levels.

The case of the Blahut-Arimoto mechanism is quite different: surprisingly, the output probability concentrates on a small number of locations. For instance, in the case $\nu = 2$, with 100K sampled pairs we obtained only 19 different output locations (which reduced to 14 after we mapped them on the 20×20 grid). Thanks to the small number of actual outputs, all the methods converge very fast. The results are shown in Figure 10 and in Table XI.

VIII. COMPARISON WITH LEAKIEST

LeakWatch [5] and leakiEst [6] are the major existing black-box leakage measurement tools, both based on the frequentist approach. LeakWatch is an extension of leakiEst, which uses

TABLE IX
CONVERGENCE FOR THE PLANAR GEOMETRIC FOR VARIOUS ν .

ν	δ	frequentist	NN	k_n -NN	
				log 10	log
2	0.1	X	X	25 795	1 102
	0.05	X	X	X	55 480
4	0.1	X	X	36 735	2 820
	0.05	X	X	X	59 875
8	0.1	X	X	15 253	5 244
	0.05	X	X	X	19 948

TABLE X
CONVERGENCE FOR THE PLANAR LAPLACIAN FOR VARIOUS ν .

ν	δ	frequentist	NN	k_n -NN	
				log 10	log
2	0.1	N/A	X	X	259
4	0.1	N/A	X	X	4 008
8	0.1	N/A	X	X	6 135
	0.05	N/A	X	X	19 961

the latter as a subroutine, but leakiEst is more feature rich: both tools compute Shannon mutual information (MI) and min-entropy leakage (ME) on the finite-output case, but leakiEst can also perform tests in the continuous output case. We compare leakiEst with our methods, for a time side channel in the RFID chips of the European passports and for the Gowalla examples of the previous section.

LeakiEst performs two functions: i) a statistical test, detecting if there is evidence of leakage (here referred to as *leakage evidence test*), and ii) the estimation of ME (discrete) or MI (discrete and continuous output). The leakage evidence

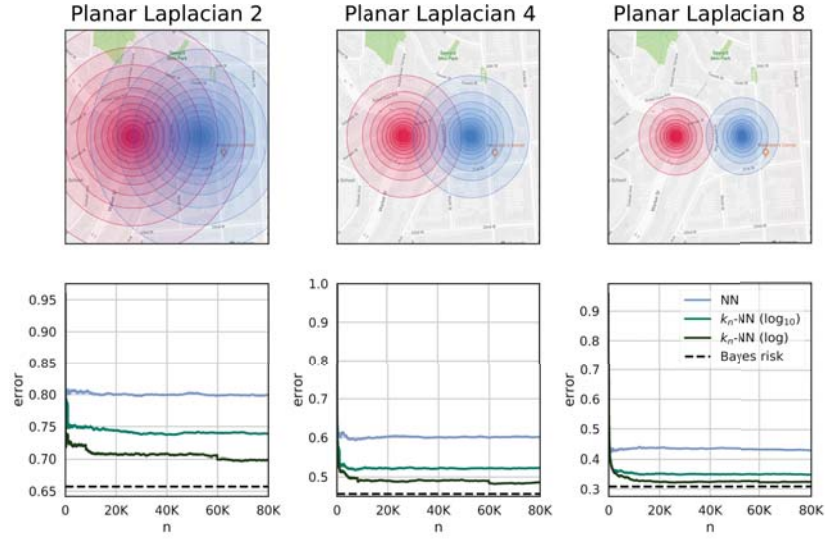


Fig. 9. Estimates' convergence speed for the planar Laplacian defense applied to the Gowalla dataset, for $\nu = 2$, $\nu = 4$ and $\nu = 8$, respectively. Above each graph is represented the distribution of the geometric noise for two adjacent input cells.

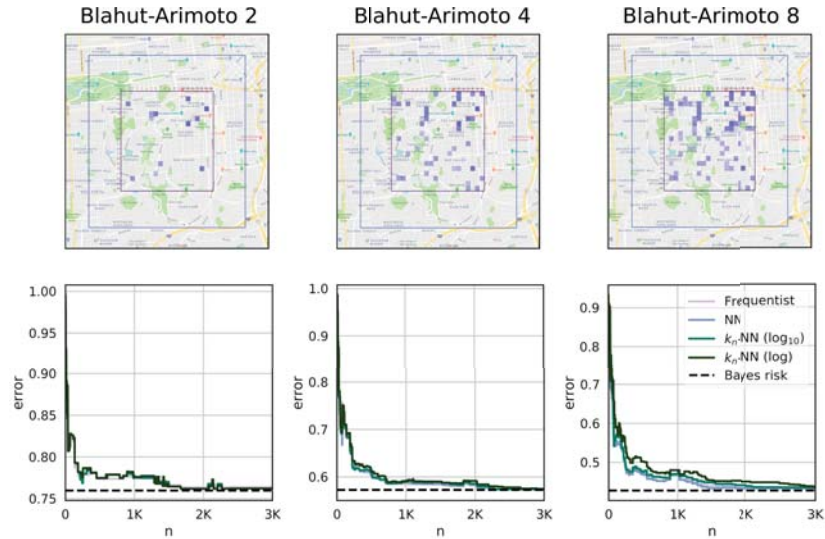


Fig. 10. Estimates' convergence speed for the Blahut-Arimoto defense applied to the Gowalla dataset, for $\nu = 2$, $\nu = 4$ and $\nu = 8$, respectively. Above each graph is represented the distribution of the output probability as produced by the mechanism. All the outputs with non-null probability turn out to be inside the input square. The outputs are points on the 340×340 grid, but here are mapped on the coarser 20×20 grid for the sake of visual clarity.

test generates a “no leakage” distribution via a bootstrapping variant, it estimates the leakage measure on it, and it compares this estimate with the measure computed on the original data: if its value is far from the former (w.r.t. some defined confidence level), then the tool declares there is evidence of leakage. The second function estimates the distribution with an appropriate method (frequentist, for finite outputs, Kernel Density Estimation, for continuous outputs).

A. Time side channel on e-Passports' RFID chips

Chothia et al. [9] discovered a side-channel attack in the way the protocols of various European countries' passports exchanged messages some years ago. (The protocols have been corrected since then.) The problem was that, upon receiving a message, the e-Passport would first check the Message Authentication Code (MAC), and only *afterwards* verify the nonce (so to assert the message was not replayed). Therefore, an attacker who previously intercepted a valid message from a legitimate session could replay the message and detect a

TABLE XI
CONVERGENCE FOR THE BLAHUT-ARIMOTO FOR VARIOUS ν .

ν	δ	frequentist	NN	k_n -NN	
				log 10	log
2	0.1	37	37	37	37
	0.05	135	135	135	135
	0.01	1 671	1 664	1 408	1 408
	0.005	6 179	6 179	1 671	1 671
4	0.1	220	220	220	257
	0.05	503	502	509	703
	0.01	2 029	1 986	2 055	2 404
	0.005	2 197	2 055	2 280	2 658
8	0.1	345	401	553	1 285
	0.05	1 285	1 170	1 343	1 679
	0.01	2 104	2 017	2 495	4 190
	0.005	2 231	2 231	3 881	6 121

TABLE XII
LEAKAGE OF EUROPEAN PASSPORTS

Passport	leakiEst: Leakage evidence	F-BLEAU: R^*
British	yes	0.383
German	no	0.490
Greek	no	0.462
Irish	yes	0.350

Random guessing baseline is $R^\pi = 0.5$.

difference between the response time of the victim’s passport and any other passport; this could be used to track the victim. As an initial solution, Chothia et al. [6] proposed to add padding to the response time, and they used leakiEst to look for any evidence of leakage after such a defense.

We compared F-BLEAU and leakiEst on the padded timing data [30]. The secret space contains answers to the binary question: “is this the same passport?”; the dataset is balanced, hence $R^\pi = 0.5$. We make this comparison on the basis that, if leakiEst detects no leakage, then the Bayes risk should be maximum: no leakage happens if and only if $R^* = R^\pi$. We compute ME from the Bayes risk as:

$$\text{ME} := -\log_2(1 - R^\pi) + \log_2(1 - R^*). \quad (26)$$

For F-BLEAU, we randomly split the data into training (75%) and hold-out sets, and then estimated R^* on the latter; we repeated this for 100 different random initialization seeds, and averaged the estimates. Results in Table XII show two cases where leakiEst did not find enough evidence of leakage, while F-BLEAU indicates non-negligible leakage. Note that, because F-BLEAU’s results are based on an actual classifier, they implicitly demonstrate there exists an attack that succeeds with accuracy 51% and 54%. We attribute this discrepancy between the tools to the fact that the dataset is small ($\approx 1\text{K}$ examples), and leakiEst may not find enough evidence to reject the hypothesis of “no leakage”; indeed, leakiEst sets a fairly high standard for this decision (95% confidence interval).

TABLE XIII
ESTIMATED LEAKAGE OF PRIVACY MECHANISMS ON GOWALLA DATA

Mechanism	ν	leakiEst		F-BLEAU: ME	True ME
		L.E.	ME		
B.-Arimoto	2	no*	1.481	1.479	1.501
	4	no*	2.305	2.310	2.304
	8	no*	2.738	2.746	2.738
Geometric	2	no	2.585	1.862	1.988
	4	no	2.859	2.591	2.638
	8	no	3.105	2.983	2.996
Mechanism	ν	leakiEst: L.E.	F-BLEAU: ME	True ME	
Laplacian	2	no	1.802	1.987	
	4	no	2.550	2.631	
	8	no	2.970	3.003	

L.E.: leakage evidence test

B. Gowalla dataset

We compare F-BLEAU and leakiEst on the location privacy mechanisms (section VII): Blahut-Arimoto, planar Geometric, and planar Laplacian. The main interest is to verify whether the advantage of F-BLEAU w.r.t. the frequentist approach, which we observed for large output spaces, translates into an advantage also w.r.t. leakiEst. For the first two mechanisms we also compare the ME estimates. For the Laplacian case (continuous), we only use leakiEst’s leakage evidence test.

We run F-BLEAU and leakiEst on the datasets as in section VII. Results in Table XIII show that, in the cases of planar Geometric and Laplacian distributions, leakiEst does not detect any leakage (the tool reports “Too small sample size”); furthermore, the ME estimates it provides for the planar Geometric distribution are far from their true values. F-BLEAU, however, is able to produce more reliable estimates.

The Blahut-Arimoto results are more interesting: because of the small number of actual outputs, the ME estimates of F-BLEAU and leakiEst perform equally well. However, even in this case leakiEst’s leakage evidence test reports “Too small sample size”. We think the reason is that leakiEst takes into account the declared size of the object space rather than the effective number of observed individual outputs; this problem should be easy to fix by inferring the output size from the examples (this is the meaning of the “*” in Table XIII).

IX. CONCLUSION AND FUTURE WORK

We showed that the black-box leakage of a system, measured until now with classical statistics paradigms (*frequentist* approach), can be effectively estimated via ML techniques. We considered a set of such techniques based on the nearest neighbor principle (i.e., close observations should be assigned the same secret), and evaluated them thoroughly on synthetic and real-world data. This allows to tackle problems that were impractical until now; furthermore, it sets a new paradigm in black-box security: thanks to an equivalence between ML and black-box leakage estimation, many results from the ML theory can be now imported into this practice (and vice versa).

Empirical evidence shows that the nearest neighbor techniques we introduce excel whenever there is a notion of metric they can exploit in the output space: whereas for unseen observations the frequentist approach needs to take a guess, nearest neighbor methods can use the information of neighboring observations. We also observe that whenever the output distribution is irregular, they are equivalent to the frequentist approach, but for maliciously crafted systems they can be misled. Even in those cases, however, we remark that asymptotically they are equivalent to the frequentist approach, thanks to their universal consistency property.

We also indicated that, as a consequence of the No Free Lunch (NFL) theorem in ML, no estimate can guarantee optimal convergence. We therefore proposed F-BLEAU, a combination of frequentist and nearest neighbor rules, which runs all these techniques on a system, and selects the estimate that converges faster. We expect this work will inspire researchers to explore new leakage estimators from the ML literature; in particular, we showed that any “universally consistent” ML rule can be used to estimate the leakage of a system. Future work may focus on other rules from which one can obtain universal consistency (e.g., Support Vector Machine (SVM) and neural networks); we discuss this further in Appendix A.

A crucial advantage of the ML formulation, as opposed to the classical approach, is that it gives immediate guarantees for systems with a continuous output space. Future work may extend this to systems with continuous secret space, which in ML terms would be formalized as regression (as opposed to the classification setting we considered here).

A current limitation of our methods is that they do not provide confidence intervals. We leave this as an open problem. We remark, however, that for continuous systems it is not possible to provide confidence intervals (or to prove convergence rates) under our weak assumptions [23]; this constraint applies to any leakage estimation method.

We reiterate, however, the great advantage of ML methods: they allow tackling systems for which until now we could not measure security, with a strongly reduced number of examples.

ACKNOWLEDGMENT

Giovanni Cherubin has been partially supported by an EcoCloud grant. The work of Konstantinos Chatzikokolakis and Catuscia Palamidessi was partially supported by the ANR project REPAS.

We are very thankful to Marco Stronati, who was initially involved in this project, and without whom the authors would have not started working together. We are grateful to Tom Chothia and Yusuke Kawamoto, for their help to understand leakEst. We also thank Fabrizio Biondi for useful discussion.

This work began as a research visit whose (secondary) goal was for some of the authors and Marco to climb in Fontainebleau, France. The trip to the magic land of Fontainebleau never happened – although climbing has been a fundamental aspect of the collaboration; the name F-BLEAU is to honor this unfulfilled dream. We hope one day the group will reunite, and finally climb there together.

REFERENCES

- [1] D. Clark, S. Hunt, and P. Malacaria, “Quantitative information flow, relations and polymorphic types,” *J. of Logic and Computation*, vol. 18, no. 2, pp. 181–199, 2005.
- [2] G. Smith, “On the foundations of quantitative information flow,” in *Proceedings of the 12th International Conference on Foundations of Software Science and Computation Structures (FOSSACS 2009)*, ser. LNCS, L. de Alfaro, Ed., vol. 5504. York, UK: Springer, 2009, pp. 288–302.
- [3] M. S. Alvim, K. Chatzikokolakis, C. Palamidessi, and G. Smith, “Measuring information leakage using generalized gain functions,” in *Proceedings of the 25th IEEE Computer Security Foundations Symposium (CSF)*, 2012, pp. 265–279. [Online]. Available: <http://hal.inria.fr/hal-00734044/en>
- [4] C. Braun, K. Chatzikokolakis, and C. Palamidessi, “Quantitative notions of leakage for one-try attacks,” in *Proceedings of the 25th Conf. on Mathematical Foundations of Programming Semantics*, ser. Electronic Notes in Theoretical Computer Science, vol. 249. Elsevier B.V., 2009, pp. 75–91. [Online]. Available: <http://hal.archives-ouvertes.fr/inria-00424852/en/>
- [5] T. Chothia, Y. Kawamoto, and C. Novakovic, “LeakWatch: Estimating information leakage from java programs,” in *Proc. of ESORICS 2014 Part II*, 2014, pp. 219–236.
- [6] —, “A tool for estimating information leakage,” in *International Conference on Computer Aided Verification (CAV)*. Springer, 2013, pp. 690–695.
- [7] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, “Geo-indistinguishability: Differential privacy for location-based systems,” in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM, 2013, pp. 901–914.
- [8] S. Oya, C. Troncoso, and F. Pérez-González, “Back to the drawing board: Revisiting the design of optimal location privacy-preserving mechanisms,” in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS ’17. ACM, 2017, pp. 1959–1972. [Online]. Available: <http://doi.acm.org/10.1145/3133956.3134004>
- [9] T. Chothia and V. Smirnov, “A traceability attack against e-passports,” in *International Conference on Financial Cryptography and Data Security*. Springer, 2010, pp. 20–34.
- [10] D. H. Wolpert, “The lack of a priori distinctions between learning algorithms,” *Neural computation*, vol. 8, no. 7, pp. 1341–1390, 1996.
- [11] M. Fredrikson, S. Jha, and T. Ristenpart, “Model inversion attacks that exploit confidence information and basic countermeasures,” in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 1322–1333.
- [12] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership inference attacks against machine learning models,” in *Security and Privacy (SP), 2017 IEEE Symposium on*. IEEE, 2017, pp. 3–18.
- [13] K. Chatzikokolakis, T. Chothia, and A. Guha, “Statistical measurement of information leakage,” *Tools and Algorithms for the Construction and Analysis of Systems*, pp. 390–404, 2010.
- [14] M. Boreale and M. Paolini, “On formally bounding information leakage by statistical estimation,” in *International Conference on Information Security*. Springer, 2014, pp. 216–236.
- [15] T. Chothia and A. Guha, “A statistical test for information leaks using continuous mutual information,” in *Proceedings of the 24th IEEE Computer Security Foundations Symposium, CSF 2011, Cernay-la-Ville, France, 27-29 June, 2011*. IEEE Computer Society, 2011, pp. 177–190. [Online]. Available: <https://doi.org/10.1109/CSF.2011.19>
- [16] T. Chothia, Y. Kawamoto, C. Novakovic, and D. Parker, “Probabilistic point-to-point information leakage,” in *Computer Security Foundations Symposium (CSF), 2013 IEEE 26th*. IEEE, 2013, pp. 193–205.
- [17] G. Cherubin, “Bayes, not naïve: Security bounds on website fingerprinting defenses,” *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 4, pp. 215–231, 2017.
- [18] N. Santhi and A. Vardy, “On an improvement over Rényi’s equivocation bound,” 2006, presented at the 44-th Annual Allerton Conference on Communication, Control, and Computing, September 2006. Available at <http://arxiv.org/abs/cs/0608087>.
- [19] I. Belghazi, S. Rajeswar, A. Baratin, R. D. Hjelm, and A. Courville, “Mine: mutual information neural estimation,” *arXiv preprint arXiv:1801.04062*, 2018.

- [20] X. Chen, Y. Duan, R. Houthoofd, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," in *Advances in neural information processing systems*, 2016, pp. 2172–2180.
- [21] J. Chen, L. Song, M. J. Wainwright, and M. I. Jordan, "Learning to explain: An information-theoretic perspective on model interpretation," *arXiv preprint arXiv:1802.07814*, 2018.
- [22] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.
- [23] L. Devroye, L. Györfi, and G. Lugosi, *A probabilistic theory of pattern recognition*. Springer Science & Business Media, 2013, vol. 31.
- [24] C. J. Stone, "Consistent nonparametric regression," *The annals of statistics*, pp. 595–620, 1977.
- [25] C. Dwork, "Differential privacy," in *33rd International Colloquium on Automata, Languages and Programming (ICALP 2006)*, ser. Lecture Notes in Computer Science, M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, Eds., vol. 4052. Springer, 2006, pp. 1–12. [Online]. Available: http://dx.doi.org/10.1007/11787006_1
- [26] R. Shokri, G. Theodorakopoulos, C. Troncoso, J.-P. Hubaux, and J.-Y. L. Boudec, "Protecting location privacy: optimal strategy against localization attacks," in *Proceedings of the 19th ACM Conference on Computer and Communications Security (CCS 2012)*, T. Yu, G. Danezis, and V. D. Gligor, Eds. ACM, 2012, pp. 617–627.
- [27] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. John Wiley & Sons, Inc., 2006.
- [28] "The gowalla dataset." [Online]. Available: <https://snap.stanford.edu/data/loc-gowalla.html>
- [29] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: User movement in location-based social networks," in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '11. New York, NY, USA: ACM, 2011, pp. 1082–1090. [Online]. Available: <http://doi.acm.org/10.1145/2020408.2020579>
- [30] Example: e-passport traceability. School of Computer Science - leakiest. [Online]. Available: www.cs.bham.ac.uk/research/projects/infotools/leakiest/examples/epassports.php
- [31] I. Steinwart, "Support vector machines are universally consistent," *Journal of Complexity*, vol. 18, no. 3, pp. 768–791, 2002.
- [32] T. Glasmachers, "Universal consistency of multi-class support vector classification," in *Advances in Neural Information Processing Systems*, 2010, pp. 739–747.
- [33] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21–27, 1967.
- [34] K. Fukunaga and D. M. Hummels, "Bayes error estimation using parzen and k-nn procedures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 5, pp. 634–643, 1987.
- [35] M. Backes and B. Köpf, "Formally bounding the side-channel leakage in unknown-message attacks," in *European Symposium on Research in Computer Security*. Springer, 2008, pp. 517–532.
- [36] P. C. Kocher, "Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems," in *Annual International Cryptology Conference*. Springer, 1996, pp. 104–113.
- [37] B. Köpf and D. Basin, "Timing-sensitive information flow analysis for synchronous systems," in *European Symposium on Research in Computer Security*. Springer, 2006, pp. 243–262.

APPENDIX A ADDITIONAL TOOLS FROM ML

The ML literature can offer several more tools to black-box security. We now enumerate additional UC rules, a lower bound of the Bayes risk, and a general way of obtaining estimates that converge from below.

The family of UC rules is fairly large. An overview of them is by Devroye et al. [23], who, in addition to nearest neighbor methods, report histogram rules and kinds of neural networks; these are UC under requirements on their parameters. Steinwart proved that Support Vector Machine (SVM) is also UC for some parameter choices, in the case $|\mathbb{S}| = 2$ [31]; to the

best of our knowledge, attempts to construct an SVM that is UC when $|\mathbb{S}| > 2$ have failed so far (e.g., [32])

In applications with strict security requirements, a (pessimistic) lower bound of the Bayes risk may be desirable. From a result by Cover and Hart one can derive a lower bound on the Bayes risk based on the NN error, R^{NN} [33]: as $n \rightarrow \infty$:

$$\frac{|\mathbb{S}| - 1}{|\mathbb{S}|} \left(1 - \sqrt{1 - \frac{|\mathbb{S}|}{|\mathbb{S}| - 1} R^{NN}} \right) \leq R^*. \quad (27)$$

This was used as the basis for measuring the black-box leakage of website fingerprinting defenses [17].

Finally, one may obtain estimators that converge to the Bayes risk in expectation *from below*, for example, by estimating the error of a k_n -NN rule on its training set [17], [34].

APPENDIX B DESCRIPTION OF THE SYNTHETIC SYSTEMS

A. Geometric system

Geometric systems are typical in differential privacy and are obtained by adding negative exponential noise to the result of a query. The reason is that the property of DP is expressed in terms of a factor between the probability of a reported answer, and that of its immediate neighbor. A similar construction holds for the geometric mechanism implementing geo-indistinguishability. In that case the noise is added to the real location to report an obfuscated location. Here we give an abstract definition of a geometric system, in terms of secrets (e.g., result of a query / real location) and observables (e.g., reported answer / reported location).

Let \mathbb{S} and \mathbb{O} be sets of consecutive natural numbers, with the standard notion of distance. Two numbers $s, s' \in \mathbb{S}$ are called *adjacent* if $s = s' + 1$ or $s' = s + 1$.

Let ν be a real non-negative number and consider a function $g : \mathbb{S} \mapsto \mathbb{O}$. After adding negative exponential noise to the output of g , the resulting geometric system is described by the following channel matrix:

$$C_{s,o} = P(o | s) = \lambda \exp(-\nu |g(s) - o|), \quad (28)$$

where λ is a normalizing factor. Note that the privacy level is defined by ν/Δ_g , where Δ_g is the sensitivity of g :

$$\Delta_g = \max_{s_1 \sim s_2 \in \mathbb{S}} (g(s_1) - g(s_2)), \quad (29)$$

where $s_1 \sim s_2$ means s_1 and s_2 are adjacent. Now let $\mathbb{S} = \{1, \dots, w\}$, $\mathbb{O} = \{1, \dots, w'\}$, we select g to be $g(s) = s \cdot w'/w$. We define

$$\lambda = \begin{cases} e^\nu / (e^\nu + 1) & \text{if } o = 1 \text{ or } o = w' \\ (e^\nu - 1) / (e^\nu + 1) & \text{otherwise,} \end{cases} \quad (30)$$

so to truncate the distribution at its boundaries.

This definition of Geometric system prohibits the case $|\mathbb{S}| > |\mathbb{O}|$. To consider such case, we generate a repeated geometric channel matrix, such that

$$C'_{s,o} = C_{s',o} \quad s' = s \bmod |\mathbb{O}|, \quad (31)$$

where $C_{s,o}$ is the geometric channel matrix described above.

B. Multimodal geometric system

We construct a multimodal distribution as the weighted sum of two geometric distributions, shifted by some shift parameter. Let $\mathcal{C}_{s,o}$ be a geometric channel matrix. The respective multimodal geometric channel, for shift parameter σ , is:

$$\mathcal{C}_{s,o}^M = w_1 \mathcal{C}_{s,o} + w_2 \mathcal{C}_{s+2\sigma,o}. \quad (32)$$

In experiments, we used $\sigma = 5$ and weights $w_1 = w_2 = 0.5$.

C. Spiky system

Consider an observation space constituted of q consecutive integer numbers $\mathbb{O} = \{0, \dots, q-1\}$, for some even positive integer q , and secrets space $|\mathbb{S}| = 2$. Assume that \mathbb{O} is a ring with the operations $+$ and $-$ defined as the sum and the difference modulo q , respectively, and consider the distance on \mathbb{O} defined as: $d(i, j) = |i - j|$. (Note that (\mathbb{O}, d) is a “circular” structure, that is, $d(q-1, 0) = 1$.) The Spiky system has uniform prior, and channel matrix constructed as follows:

$$\mathcal{C}_{s,o} = \begin{bmatrix} 2/q & 0 & 2/q & \dots & 0 \\ 0 & 2/q & 0 & \dots & 2/q \end{bmatrix}. \quad (33)$$

APPENDIX C

DETAILED CONVERGENCE RESULTS

Convergence for multimodal geometric systems, when varying ν and for fixed $|\mathbb{S}| \times |\mathbb{O}| = 100 \times 10K$.

System	δ	Freq.	NN	k_n -NN	
				\log_{10}	\log
Multimodal $\nu = 1.0$	0.1	3 008	369	478	897
	0.05	5 938	495	754	1 267
	0.01	26 634	765	1 166	1 487
	0.005	52 081	765	1 166	1 487
Multimodal $\nu = 0.1$	0.1	24 453	398	554	821
	0.05	44 715	568	754	1 175
	0.01	149 244	4 842	1 166	1 487
	0.005	226 947	79 712	1 166	1 487
Multimodal $\nu = 0.01$	0.1	27 489	753	900	381
	0.05	103 374	101 664	92 181	31 452

Detailed convergence results for a Spiky system of size $|\mathbb{S}| = 2$ and $|\mathbb{O}| = 10K$.

δ	Freq.	NN	k_n -NN	
			\log_{10}	\log
0.1	15 953	22 801	52 515	99 325
0.05	22 908	29 863	62 325	112 467
0.01	38 119	44 841	81 925	137 969
0.005	44 853	51 683	91 661	147 593

APPENDIX D

UNIFORM SYSTEM

We measured convergence of the methods for a uniform system; this system is constructed so that all secret-object examples are equally likely, that is $\mu(s, o) = \mu(s', o')$ for all $s, o \in \mathbb{S} \times \mathbb{O}$. The Bayes risk in this case is $R^* = 1 - 1/|\mathbb{S}|$.

Figure 11 shows that even in this case all rules are equivalent. Indeed, because the system leaks nothing about its

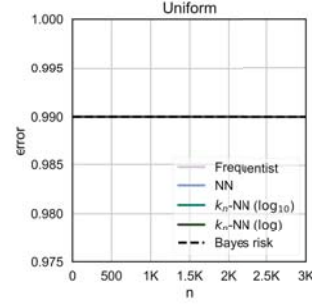


Fig. 11. Convergence for a Uniform system of size 100×100 .

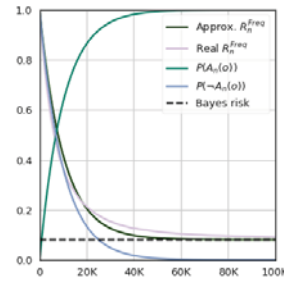


Fig. 12. Approximation of the frequentist estimate as n grows for $R^* \approx 0.08$, $|\mathbb{O}| = 10K$, and $|\mathbb{S}| = 1K$; the approximation is compared with the real frequentist estimate R_n^{Freq} .

secrets, all the estimators need to randomly guess; but because for this system the Bayes risk is identical to random guessing error ($R^* = 1 - 1/|\mathbb{S}| = R^\pi$), all the estimators converge immediately to its true value.

APPENDIX E

APPROXIMATION OF THE FREQUENTIST ESTIMATE

To better understand the behavior of the frequentist approach for observations that were not in the training data, we derive a crude approximation of this estimate in terms of the size of training data n . The approximation makes the following assumptions:

- 1) each observation $o \in \mathbb{O}$ is equally likely to appear in training data (i.e., $P(o) = 1 - \frac{1}{|\mathbb{O}|}$);
- 2) if an observation appears in the training data, the frequentist approach outputs the secret minimizing the Bayes risk;
- 3) the frequentist estimate knows the real priors π ;
- 4) if an observation does not appear in the training data, then the frequentist approach outputs the secret with the maximum prior probability.

The first two assumptions are very strong, and thus this is just an approximation of the real trend of such estimate. However, in practice it approximates well the real trend Figure 12.

Let $A_n(o)$ denote the event “observation o appears in a training set of n examples”; because of assumption 1),

$P(A_n(o)) = 1 - \left(1 - \frac{1}{|\mathcal{O}|}\right)^n$. The conditional Bayes risk estimated with a frequentist approach given n examples is:

$$\begin{aligned} r_n(o) &= r_n(o|A_n(o))P(A_n(o)) + r_n(o|\neg A_n(o))P(\neg A_n(o)) = \\ &= \left(1 - \max_{s \in \mathcal{S}} \frac{\hat{\mathcal{C}}_{s,o}\hat{\pi}(s)}{P(o)}\right)P(A_n(o)) + \\ &\quad + \left(1 - \max_{s \in \mathcal{S}} \hat{\pi}(s)\right)P(\neg A_n(o)) \approx \\ &\approx \left(1 - \max_{s \in \mathcal{S}} \frac{\mathcal{C}_{s,o}\pi(s)}{P(o)}\right)P(A_n(o)) + \\ &\quad + \left(1 - \max_{s \in \mathcal{S}} \pi(s)\right)P(\neg A_n(o)) \end{aligned}$$

Assumptions 2) and 3) were used in the last step. From this expression, we derive the frequentist estimate of R^* at step n :

$$\begin{aligned} R_n^{Freq} &= \mathbb{E}r_n = \\ &= \sum_{o \in \mathcal{O}} P(o) \left(1 - \max_{s \in \mathcal{S}} \frac{\mathcal{C}_{s,o}\pi(s)}{P(o)}\right)P(A_n(o)) + \\ &\quad + \sum_{o \in \mathcal{O}} P(o) \left(1 - \max_{s \in \mathcal{S}} \pi(s)\right)P(\neg A_n(o)) = \\ &= P(A_n(o)) \left(\sum_{o \in \mathcal{O}} P(o) - \sum_{o \in \mathcal{O}} \max_{s \in \mathcal{S}} \mathcal{C}_{s,o}\pi(s)\right) + \\ &\quad + P(\neg A_n(o)) \left(1 - \max_{s \in \mathcal{S}} \pi(s)\right) \sum_{o \in \mathcal{O}} P(o) = \\ &= P(A_n(o)) \left(1 - \sum_{o \in \mathcal{O}} \max_{s \in \mathcal{S}} \mathcal{C}_{s,o}\pi(s)\right) + \\ &\quad + P(\neg A_n(o)) \left(1 - \max_{s \in \mathcal{S}} \pi(s)\right) = \\ &= P(A_n(o))R^* + P(\neg A_n(o))R^\pi = \\ &= R^* \left(1 - \left(1 - \frac{1}{|\mathcal{O}|}\right)^n\right) + R^\pi \left(1 - \frac{1}{|\mathcal{O}|}\right)^n. \end{aligned}$$

Note that in the second step we used $P(A_n(o))$ as a constant, which is allowed by assumption 1).

The expression of R_n indicates that $P(A_n(o))$ weights between random guessing according to priors-based random guessing and the Bayes risk; when $P(A_n(o)) \geq P(\neg A_n(o))$, which happens for $n \geq -\frac{\log 2}{\log(1 - \frac{1}{|\mathcal{O}|})}$ the frequentist approach starts approximating using the actual Bayes risk (Figure 12).

APPENDIX F GOWALLA DETAILS

We report in Table XIV the real Bayes risk estimated analytically for the Gowalla dataset defended using the various mechanisms, and their respective utility.

APPENDIX G APPLICATION TO TIME SIDE CHANNEL

We use F-BLEAU to measure the leakage in the running time of the *square-and-multiply* exponentiation algorithm in the finite field \mathbb{F}_{2^w} ; exponentiation in \mathbb{F}_{2^w} is relevant, for example, for the implementation of the ElGamal cryptosystem.

TABLE XIV
TRUE BAYES RISK AND UTILITY FOR GOWALLA DATASET DEFENDED USING VARIOUS LOCATION PRIVACY MECHANISMS.

Mechanism	ν	R^*	Utility
Blahut-Arimoto	2	0.760	334.611
	4	0.571	160.839
	8	0.428	96.2724
Geometric	2	0.657	288.372
	4	0.456	144.233
	8	0.308	96.0195
Laplacian	2	0.657	288.66
	4	0.456	144.232
	8	0.308	96.212

TABLE XV
NUMBER OF UNIQUE SECRETS AND OBSERVATIONS FOR THE TIME SIDE CHANNEL TO FINITE FIELD EXPONENTIATION.

Operands' size	$ \mathcal{S} $	$ \mathcal{O} $
4 bits	2^4	34
6 bits	2^6	123
8 bits	2^8	233
10 bits	2^{10}	371
12 bits	2^{12}	541

We consider a hardware-equivalent implementation of the algorithm computing m^s in \mathbb{F}_{2^w} . We focus our analysis on the simplified scenario of a “one-observation” adversary, who makes exactly *one* measurement of the algorithm’s execution time o , and aims to predict the corresponding secret key s .

A similar analysis was done by Backes and Köpf [35] by using a leakage estimation method based on the frequentist approach. Their analysis also extended to a “many-observations adversary”, that is, an adversary who can make m observations (o_1, \dots, o_m) , all generated from the same secret s , and has to predict s accordingly.

A. Side channel description

Square-and-multiply is a fast algorithm for computing m^s in the finite field \mathbb{F}_{2^w} , where w here represents the bit size of the operands m and s . It works by performing a series of multiplications according to the binary representation of the exponent s , and its running time is proportional to the number of 1’s in s . This fact was noticed by Kocher [36], who suggested side channel attacks to the RSA cryptosystem based on time measurements.

B. Message blinding

We assume the system implements *message blinding*, a technique which hides to an adversary the value m for which m^s is computed. Blinding was suggested as a method for thwarting time side channels [36], which works as follows. Consider, for instance, decryption for the RSA cryptosystem: $m^d \pmod{N}$, for some decryption key d ; the system first computes $m \cdot r^e$, where e is the encryption key and r is

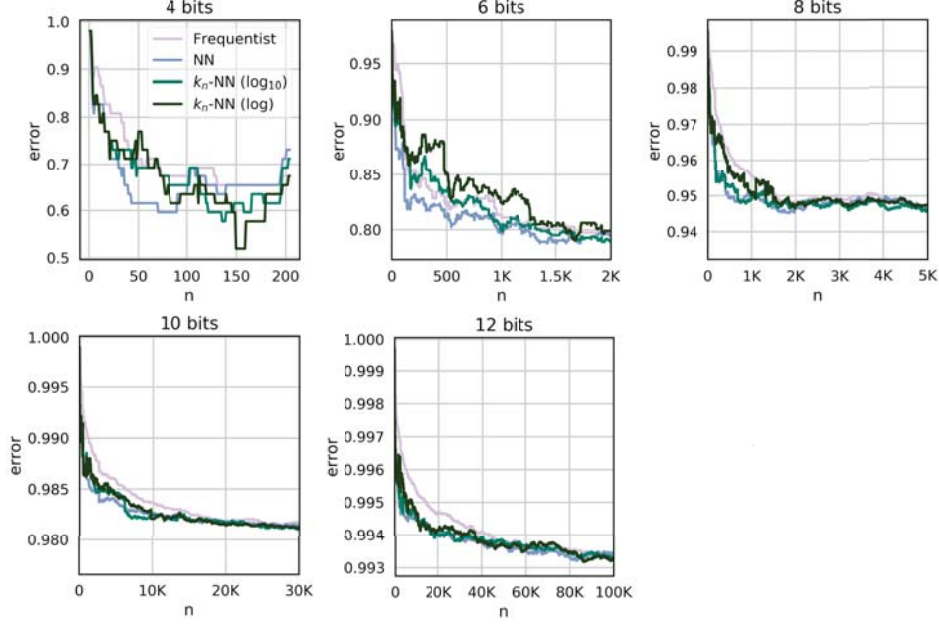


Fig. 13. Convergence of the estimates for the time side channel attack to the exponentiation algorithm as the bit size of the operands increases.

some random value; then it computes $(mr^e)^d$, and returns the decrypted message after dividing the result by r .

Message blinding has the advantage of hiding information to an adversary; however, it was shown that it is not enough for preventing time side channels (e.g., [35]).

C. Implementation and results

We consider a Gezel implementation of finite field exponentiation. Gezel is a description language for clocked hardware, equipped with a simulation environment whose executions preserve the corresponding circuit’s timing information. This means that the time measurements (i.e., clock cycles) we make reflect the corresponding circuit implementation [37].

We compare the performances of the frequentist and nearest neighbor approaches in terms of the number of black-box examples required for convergence. For each bit size $w \in \{4, 6, \dots, 12\}$, and for all the values $(m_i, s_i) \in \{0, \dots, 2^w - 1\}^2$, we run the exponentiation algorithm to compute m^s , and

measure its execution time o_i . As with our application to location privacy (section VII), we estimate the Bayes risk by training a classifier on a set of increasing examples n and by computing its error on a hold-out set. We set the size of the hold-out set to $\min(0.2 \cdot 2^{2w}, 250\,000)$.

Results in Figure 13 show that, while for small bit sizes the frequentist approach outperforms nearest neighbor rules, as w increases, the frequentist approach requires a much larger number of examples. Nevertheless, in these experiments we did not notice a substantial advantage in nearest neighbor rules, even though the output space is equipped with a notion of metric. Table XV helps interpreting this result: for larger bit sizes w of the exponentiation operands, the possible output values (i.e., clock cycles) only increase minimally; this confirms that, as noticed in our previous experiments, nearest neighbor and frequentist estimates tend to perform similarly for systems with small output space.