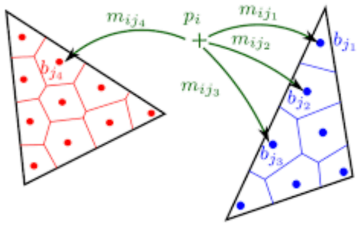
AGENCE NATIONALE DE LA RECHERCHE

ANR

REPAS

RELIABLE AND PRIVACY-AWARE SOFTWARE SYSTEMS



Deliverable D.5.b

PROGRESS REPORT
F-BLEAU: A tool for estimating information leakage

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, and it 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 numerous studies on measuring a system's leakage in a black-box manner.

Historically, black-box leakage estimation methods have been based on classical statistical techniques and they follow what we refer to as the *frequentist* paradigm: the idea is to let the system run repeatedly and count the relative frequencies of the inputs (the secrets) and the respective outputs, with the goal of estimating their joint probability distribution. From this distribution, it is then possible to derive the conditional probabilities, and then proceed as usual to compute the desired leakage measures. LeakWatch [4] and leakiEst [5], two well-known tools for black-box leakage estimation, are examples of application of this principle.

Unfortunately, the frequentist approach does not always scale to real-world problems: as the number of possible input and output values of the channel matrix increases, the amount of examples required for this method to converge becomes too large to gather and handle. For example, LeakWatch requires a number of examples that is much larger than the product of the size of input and output space. This means that, if their respective sizes are in the order of a thousands (i.e., a 10 bits input and a 10 bits output), the number of examples needed is of the order of several millions. For the same reason, these methods cannot tackle systems with continuous outputs, at least in the cases of min-entropy leakage and g-leakage – as a matter of fact, the frequentist approach cannot even be constructed formally in such case.

Our contribution

In this paper, we show that machine learning (ML) methods can provide the necessary scalability to black-box measurements, yet maintaining formal guarantees on their estimates. By pointing out a fundamental equivalence between ML and the black-box measurement estimation, we show that any ML rule from a certain class (the *universally consistent* rules) can be used to estimate the leakage of a channel. In particular, we consider nearest neighbor-based learning rules – 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 methods 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 easy to understand, which allows determining their strengths and weaknesses (i.e., systems for which they excel or perform badly), ii) we are able to formulate them as an extension of the well-understood frequentist approach; in particular, we define the NN rule so that it is equivalent to frequentist when predicting the secret of previously observed outputs, but which improves on it for unseen outputs (in which case the frequentist has to random guess). Notably, our methods are also directly applicable for measuring the leakage of systems with continuous output.

We dub this set of techniques F-BLEAU (Fast Black-box LEAkage Upper-bounds), which computes nearest neighbor and frequentist estimates, and selects the one converging faster; we will release the code of F-BLEAU as Open Source.

We evaluate our methods on synthetic data, where we know the true distributions and we can determine exactly when the estimates converge. Furthermore, we apply these techniques for measuring the leakage in a real dataset of users' checkins (`Gowalla` [6], [7]), defended under three state-of-the-art mechanisms: two geo-indistinguishability mechanisms (planar geometric and planar laplacian) [8] and a method by Oya et al. [9], which is a refinement of the optimal mechanism by Shokri et al. [10], 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 frequentist approaches cannot. Finally, we compare F-BLEAU with leakiEst on the problem of estimating the leakage of European passports [5], [11], and on the location privacy mechanisms, showing that the same advantage in terms

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.	**66 300**	**66 300**	**66 300**
Geometric ($\nu = 0.1$)	100 secrets, 10K obs.	127 201	**434**	693
Geometric ($\nu = 0.5$)	1K secrets, 100K obs.	403 135	**4 083**	7 018
Geometric ($\nu = 0.02$)	100 secrets, 10K obs.	85 907	85 639	**11 192**
Spiky (contrived example)	2 secrets, 10K obs.	**50 341**	55 964	101 737
Planar Geometric $\nu = 2$	`Gowalla` checkins in San Francisco area	X	X	**1 102**
Laplacian $\nu = 2$	"	N/A	X	**259**
Blahut-Arimoto $\nu = 2$	"	**37**	**37**	**37**

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

of convergence rate that F-BLEAU provides with respect to the frequentist approach also translate into an advantage with respect to the real tool leakiEst.

As a further evidence of the practicality of F-BLEAU, in Appendix C we give an example of application to measure the leakage of a time side channel in a hardware implementation of finite field exponentiation.

In summary, our paper demonstrates that ML methods can be successfully applied to black-box leakage estimation, and they generally either offer an an advantage over the frequentist approach, or they are equivalent to it – except for particularly malicious channel matrices (Table I). Furthermore, as a consequence of the NFL theorem in ML, we point out that in practice one should always evaluate more than one estimator, and then choose the best performing one, as there exist no optimal estimator across all systems. This paper provides the basis for future research on leakage estimators, and suggests an entire class of methods (universally consistent learning rules) on which they can develop.

II. RELATED WORK

Chatzikokolakis et al. [12] 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 estimating the underlying probability distribution accordingly. This is what we here refer to as the frequentist paradigm. A fundamental development of their work by Boreale and Paolini [13] showed that, in absence of significant a priori information about the output distribution, no estimator does better than the exhaustive enumeration of the input domain. In particular, Boreale and Paolini showed that it is difficult to obtain tight upper bounds under relaxed assumptions; on the other hand, when one has some control over the input distribution, they constructed an estimator that with high probability gives lower bounds irrespectively of the underlying distribution, and tight upper bounds if the input distribution induces a "close to uniform" output distribution. In line with this work, subsection III-F will show that, as a consequence of the No Free Lunch theorem in ML, no leakage estimator can claim to converge faster than any other estimator on all distributions.

One of the best known tools that have been developed on the basis of the frequentist paradigm is LeakWatch [4], [5]. Subsequently, Chothia et al. proposed leakiEst [14], [15], an extension able to cope also with continuous output. In Section VII we will compare leakiEst with our proposal in terms of convergence rate and computational efficiency.

Cherubin [16] used guarantees of nearest neighbor learning rules for measuring in a black-box manner the smallest error (Bayes risk) of a one-try adversary performing website fingerprinting attacks; however, this work was limited to a small set of techniques, and to the specific traffic analysis problem. In section IV we will make evident the connection between ML and QIF, and we will apply similar techniques to the more general problem of measuring the leakage of a system.

III. PRELIMINARIES

We define a system, and formulate its leakage in terms of the Bayes risk. We further introduce ML notions, which we will use in the next sections to estimate the Bayes risk.

A. Notation

We consider a system $(\pi, \mathcal{C}_{s,o})$, that associates to a secret input s 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. In this paper, $\mathbb{S}$ is finite, and $\mathbb{O}$ is finite unless otherwise stated.

B. Measuring Leakage

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 will be concerned with finding tight estimates of the Bayes risk, which can then be used to compute 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 for an adversary to predict 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.

Consider a system, $(\pi, \mathcal{C}_{s,o})$. The conditional Bayes risk $r^*(o)$ given observation $o \in \mathbb{O}$ is the error minimizing the risk on the prediction s for the 0-1 loss function:

$$r^*(o) := 1 - \max_{s \in \mathbb{S}} P(s|o) \quad .$$

By taking the expectation of $r^*(o)$ over the distribution on $\mathbb{O}$ we obtain the Bayes risk, $R^* := \mathbb{E}\, r^*$. If $(\pi, \mathcal{C}_{s,o})$ are known, the Bayes risk is computed as follows:

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

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; the best strategy, in this case, is to always output the secret with the highest prior. We refer to this as the random guessing error, defined as:

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

c) Leakage measures: From an estimate of Bayes risk and random guessing error we can construct several leakage measures. In this paper, we will min-entropy leakage ME, defined as:

$$\mathrm{ME} := -\log_2(1 - R^\pi) + \log_2(1 - R^*)\,.$$

In section VII, we will compare F-BLEAU with LeakWatch and leakiEst on the basis of ME.

C. Black-box estimation of R^*

This paper is concerned with estimating the Bayes risk given n examples sampled from the joint distribution $\mu(\mathbb{S} \times \mathbb{O})$ generated by $(\pi, \mathcal{C}_{s,o})$. By running the system n times on secrets $s_1, \ldots, s_n \in \mathbb{S}$, chosen according to π, we generate a sequence of corresponding outputs $o_1, \ldots, o_n$, thus forming a *training set*[1] of examples $\{(o_1, s_1), ..., (o_n, s_n)\}$. From these data, we aim to make an estimate close to the real Bayes risk.

D. Learning Rules

ML rules (or, simply, *learning rules*) are algorithms for selecting a classifier given a set of training examples.

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 $\{(o_1, s_1), ..., (o_n, s_n)\}$, 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 $\mu(\mathbb{S} \times \mathbb{O})$ [17]. In case of the 0-1 loss function, the expected loss coincides with the *expected probability of error* (*expected error* for short), and if $\mu(\mathbb{S} \times \mathbb{O})$ is generated by a system $(\pi, \mathcal{C}_{s,o})$, then the expected error of a classifier $f : \mathbb{O} \mapsto \mathbb{S}$ is:

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

[1]In line with the ML practice, we call 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.

where $f(o)$ is the predicted secret for observation o. If $\mathbb{O}$ is infinite (and μ is continuous) the summation is replaced by an integral. In section IV we will show that a class of learning rules can be used to estimate a system's leakage.

E. Frequentist estimate of R^*

The frequentist paradigm [12] for measuring the leakage of a channel consists in estimating the probabilities $\mathcal{C}_{s,o}$ by counting their frequency in the training data $(o_1, s_1), ..., (o_n, s_n)$:

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

We can obtain the frequentist error from Equation 1:

$$R^{Freq} = 1 - \sum_o C_{f^{Freq}(o),o}\, \pi(f^{Freq}(o))$$

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

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

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, $\hat{\mathcal{C}} \to \mathcal{C}$ as $n \to \infty$, 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 $\{(o_1, s_1), ..., (o_n, s_n)\}$, a 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., large object space), the probability of observing and example for each object within the training set becomes small, and the frequentist classifier approaches *random guessing*. We study this matter further in section A.

F. No Free Lunch In Learning

To measure the leakage of a black-box with large $\mathbb{O} \times \mathbb{S}$, we need a classifier that makes good predictions also for objects $o \in \mathbb{O}$ that are not contained in the training set. This intuition motivates the techniques used in the next section, where we will suggest Bayes risk estimates based on nearest neighbor rules; these rules predict, for an unseen object o, the secret of its closest observation in the training set.1

However, we will first state a fundamental impossibility result on the existence of an "optimal" learning rule. One may wonder whether there is a learning rule that outperforms all the others at classifying objects (including, of course, those that were not observed in the training set). More precisely, a rule that, trained on a finite number n of examples coming from a distribution $\mu(\mathbb{S} \times \mathbb{O})$, produces a classifier whose expected error is the smallest, for all distributions μ over $\mathbb{S} \times \mathbb{O}$. Unfortunately, the answer is negative, as shown by the following simplified and weak version of the NFL theorem [18].

Theorem 1 (No Free Lunch [18])**.** *Let A and B be two learning rules that, given a training set of examples*

$\{(o_1, s_1), ..., (o_n, s_n)\}$ *sampled from a joint distribution* μ *on* $\mathbb{S} \times \mathbb{O}$*, produce classifiers* $f_n^A, f_n^B : \mathbb{O} \mapsto \mathbb{S}$ *respectively. Then, there is always a distribution* μ *such that* $\mathbb{E}_\mu R^{f_n^A} < \mathbb{E}_\mu R^{f_n^B}$*, and vice versa, there is always a distribution* μ' *such that* $\mathbb{E}_{\mu'} R^{f_n^B} < \mathbb{E}_{\mu'} R^{f_n^A}$.

Remarkably, this holds for *any* strategy, even if the learning rule is random guessing. This tells us that, under the relaxed assumption that μ can be any distribution, the very best we can do to select between A and B is to evaluate them empirically.

IV. Machine Learning Estimates of R^*

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

Throughout the section, we use interchangeably a system $(\pi, \mathcal{C}_{s,o})$ or 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 $\mu(\mathbb{S} \times \mathbb{O})$ and a learning rule A selecting a classifier $f_n \in \mathcal{F}$ using n training examples sampled from μ. Intuitively, we would like the expected error of f_n with respect to a new example (o, s) sampled from μ to approximate the Bayes risk of the corresponding system as the training set increases in size. 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* (1)*. We say that* A *is consistent if* $R^{f_n} \to R^*$ *as* $n \to \infty$*.*

The next definition captures the property for a learning rule of being *intrinsically* consistent, i.e., not thanks to a particular distribution, but for all of them:

Definition 2 (Universal Consistent 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 error of a classifier selected according to a universally consistent rule is an estimate of the Bayes risk, which converges to R^* as $n \to \infty$.

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

B. NN estimate

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*). However, it is generally defined for infinite object spaces $\mathbb{O}$, where it does not guarantee universal consistency.

TABLE II
Estimates's guarantees as $n \to \infty$

Method	Guarantee	Space $\mathbb{O}$
Frequentist	$\to R^*$	finite
NN	$\to R^*$	finite
k_n-NN	$\to R^*$	infinite, $(d, \mathbb{O})$ separable

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 $\{(o_1, s_1), ..., (o_n, s_n)\}$, 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 $I_{min}(o) = \{i \mid d(o, o_i) = \min_{j=1...n} d(o, o_j)\}$ and define:

$$NN(o) = s_{h(o)} \quad \text{where} \quad h(o) = \operatorname*{argmax}_{i \in I_{min}(o)} |\{j \mid s_j = s_i\}| .$$

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

Theorem 2 (Universal consistency of NN)**.** *Consider a distribution on* $\mathbb{O} \times \mathbb{S}$*, where* $\mathbb{O}$ *and* $\mathbb{S}$ *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* $n \to \infty$*:*

$$R_n^{NN} \to R^*.$$

Proof. 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 $n \to \infty$, 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^*. □

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 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) [19]. 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 $\{(o_1, s_1), ..., (o_n, s_n)\}$, 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:

$$k\text{-}NN(o) = s_{h(o)} \quad \text{where} \quad h(o) = \underset{i=1,\dots,k}{\operatorname{argmax}} \left|\{j \mid s_{(j)} = s_{(i)}\}\right| .$$

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

$$d(o, o_{(k')}) = ... = d(o, o_{(k)}) = ... = d(o, o_{(k'')}) ,$$

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

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

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 $k_n \to \infty$ and $k_n/n \to 0$ as $n \to \infty$.*

Stone proved that the k_n-NN rule is universally consistent:

Theorem 3 (Universal consistency of the k_n-NN rule [20])**.** *Consider an example space $\mathbb{S} \times \mathbb{O}$ and a probability distribution $\mu(\mathbb{S} \times \mathbb{O})$, where μ has a density. Select a distance metric d such that $(d, \mathbb{O})$ is separable*[2]*. Then the expected error of the k_n-NN rule converges to R^* as $n \to \infty$.*

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

V. Evaluation on Synthetic Data

We evaluate the estimates on discrete synthetic systems defined for various distributions on the channel matrix. We sample n examples from a system's distribution, and then compute the estimate on the whole object space as in Equation 1; 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:

$$\frac{\left|R_n^f - R^*\right|}{R^*} < \delta \quad .$$

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 III), we verify convergence with the absolute change:

$$\left|R_n^f - R^*\right| < \delta \quad .$$

[2]A separable space is a space containing a countable dense subset; finite spaces are separable, and so is the space of q-dimensional vectors, $\mathbb{R}^q$, with Euclidean metric.

TABLE III
Synthetic systems.

Name	$\lvert\mathbb{S}\rvert$	$\lvert\mathbb{O}\rvert$	R^*
Random 100-100	100	100	0.979
Geometric 1.0 100x10K	100	10K	~ 0
Geometric 0.1 100x10K	100	10K	0.007
Geometric 0.01 100x10K	100	10K	0.600
Geometric 0.5 100x10K	100	10K	~ 0
Geometric 0.5 1Kx100K	1K	100K	~ 0
Geometric 0.5 10Kx1M	10K	1M	~ 0
Geometric 0.2 100x1K	100	1K	0.364
Geometric 0.02 100x10K	100	10K	0.364
Geometric 0.002 100x100K	100	100K	0.364

Table III summarizes the systems we evaluate in our experiments; the rest of this section describes each into details. In this section, we assume uniform priors for all the systems.

A. Geometric systems

We first consider systems generated by Geometric noise functions, which are one of the typical mechanisms used to implement differential privacy [21]. We consider different parameters, to illustrate the effect of their variation on the convergence of the k-NN methods and the frequentist one.

1) System description: 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 $g : \mathbb{S} \mapsto \mathbb{O}$. The channel matrix of the Geometric system is:

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

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:

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

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

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

so to truncate the distribution at its boundaries.
We will now consider the following three parameters:

- the privacy level ν/Δ_g, which here is equal to $\nu|\mathbb{S}|/|\mathbb{O}|$,
- the size of the secret space $|\mathbb{S}|$, and
- the ratio $|\mathbb{O}|/|\mathbb{S}|$.

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

2) Variation of the privacy level: We fix $|\mathbb{S}| = 100$, $|\mathbb{O}| = 10$K, 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 IV respectively. In the tables, results are reported for δ convergence levels $\{0.1, 0.01, 0.001\}$; an "X" means a particular estimate did not converge within 500K examples, a missing row for a certain δ means no estimate converged.

TABLE IV
CONVERGENCE OF THE ESTIMATES WHEN VARYING ν.

System	δ	Freq.	NN	k_n-NN $\log_{10}$	k_n-NN log
Geometric 100x10K ν = 1.0	0.1	1 989	**262**	391	674
	0.01	19 823	**420**	628	894
	0.001	198 057	**434**	693	899
Geometric 100x10K ν = 0.1	0.1	18 105	**264**	391	668
	0.01	127 201	**434**	628	894
	0.001	X	X	10 727	**900**
Geometric 100x10K ν = 0.01	0.1	105 448	103 352	99 847	**34 238**

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.

3) Variation of the input size: Here we fix $\nu = 0.5$, $|\mathbb{O}|/|\mathbb{S}| = 100$, and we consider three cases $|\mathbb{S}| = 100$, $|\mathbb{S}| = 1$K, and $|\mathbb{S}| = 10$K. The results are in Figure 2 and Table V. They confirm what was logical to expect, namely that if we scale the number of inputs of a factor c and all the other parameters remain the same, then the results (the number of examples necessary to get the same estimation) are scaled by the same factor c, for all the methods. Although not surprising, it reassures us on the correctness of our procedures.

TABLE V
CONVERGENCE OF THE ESTIMATES WHEN VARYING $|\mathbb{S}|$.

System	δ	Freq.	NN	k_n-NN $\log_{10}$	k_n-NN log
Geometric 100x10K ν = 0.5	0.1	3 926	**264**	391	678
	0.01	38 181	**434**	628	894
	0.001	371 823	**434**	693	899
Geometric 1Kx100K ν = 0.5	0.1	39 461	**2 191**	4 570	7 287
	0.01	403 135	**4 083**	7 018	11 337
	0.001	X	**5 329**	8 427	14 133
Geometric 10Kx1M ν = 0.5	0.1	X	**22 929**	51 705	92 740
	0.01	X	**46 712**	82 211	X
	0.001	X	**66 610**	X	X

4) Variation of the ratio $|\mathbb{O}|/|\mathbb{S}|$: Now we fix $|\mathbb{S}| = 100$, $\nu|\mathbb{O}|/|\mathbb{S}| = 2$, and we consider three cases $|\mathbb{O}|/|\mathbb{S}| = 10$, $|\mathbb{O}|/|\mathbb{S}| = 100$, and $|\mathbb{O}|/|\mathbb{S}| = 1$K. (Note that as a consequence also ν has to vary: we have to set ν to 0.2, 0.02, and 0.002, respectively.) Results in Figure 3 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 is the object space, the larger is 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.

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

System	δ	Freq.	NN	k_n-NN $\log_{10}$	k_n-NN log
Geometric 100x1K ν = 0.2	0.1	8 674	8 702	7 103	**2 500**
	0.01	**51 689**	60 791	60 791	60 791
	0.001	**180 659**	**180 659**	**180 659**	**180 659**
Geometric 100x10K ν = 0.02	0.1	85 907	85 639	70 998	**11 192**
Geometric 100x100K ν = 0.002	0.1	X	X	413 969	**2 962**

B. Spiky system: When k_n-NN rules fail

kNN rules can take advantage of the metric on the object space to improve their convergence considerably. However, there are systems for which the Frequentist outperforms kNN. While this does not come as a surprise, given the NFL theorem (Theorem 1), investigating the form of such systems is important to understand when these methods fail.

a) System description: We construct an example of such systems, which we call the Spiky system. Consider an observation space constituted of q consecutive integer numbers $\mathbb{O} = \{0, ..., q-1\}$, 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, i.e., $d(q-1, 0) = 1$.) The Spiky system has uniform prior, and

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

This system is crafted so that most neighbors of an observable are more likely to be associated with the wrong secret. This means that NN and k_n-NN rules will tend to predict the wrong secret, until enough examples are available.

b) Discussion: We conducted experiments for a Spiky system of size $|\mathbb{O}| = 10K$. Results in Figure 4 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 $|\mathbb{O}|$, all elements in $\mathbb{O}$ tend to be covered by the examples. For every $i \in \mathbb{O}$ 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 multiset 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 $k/n \to 0$ as $n \to \infty$), 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 nearest neighbor definition) so i will not be misclassified anymore.

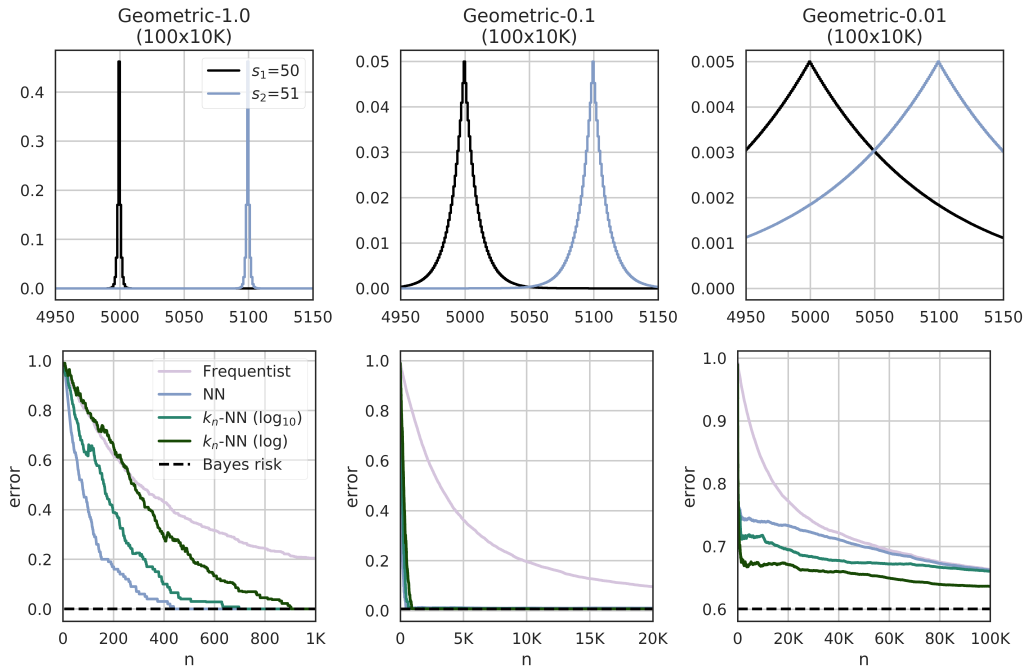


Fig. 1. Estimates' convergence for Geometric systems when varying their noise parameter. The respective distributions are shown in the top figure for two adjacent secrets $s_1 \sim s_2$.

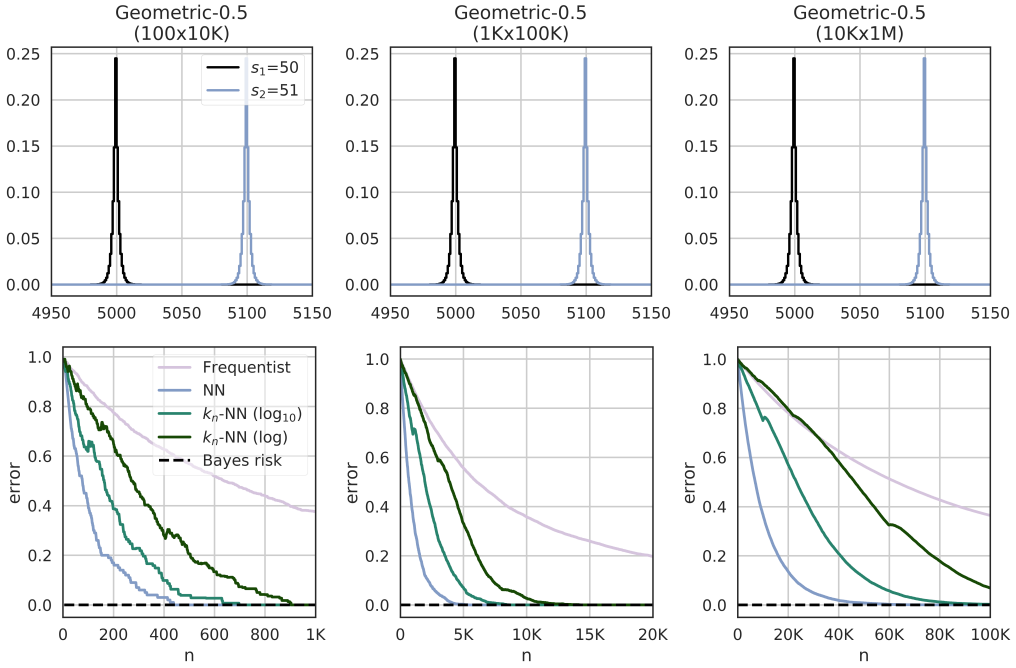


Fig. 2. Estimates' convergence for Geometric systems when varying the number of secrets. The respective distributions are shown in the top figure for two adjacent secrets $s_1 \sim s_2$.

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 is at odd distance from i (i.e., $d(i,o) = 2\ell + 1$, for some natural number ℓ). Namely it is the probability that:

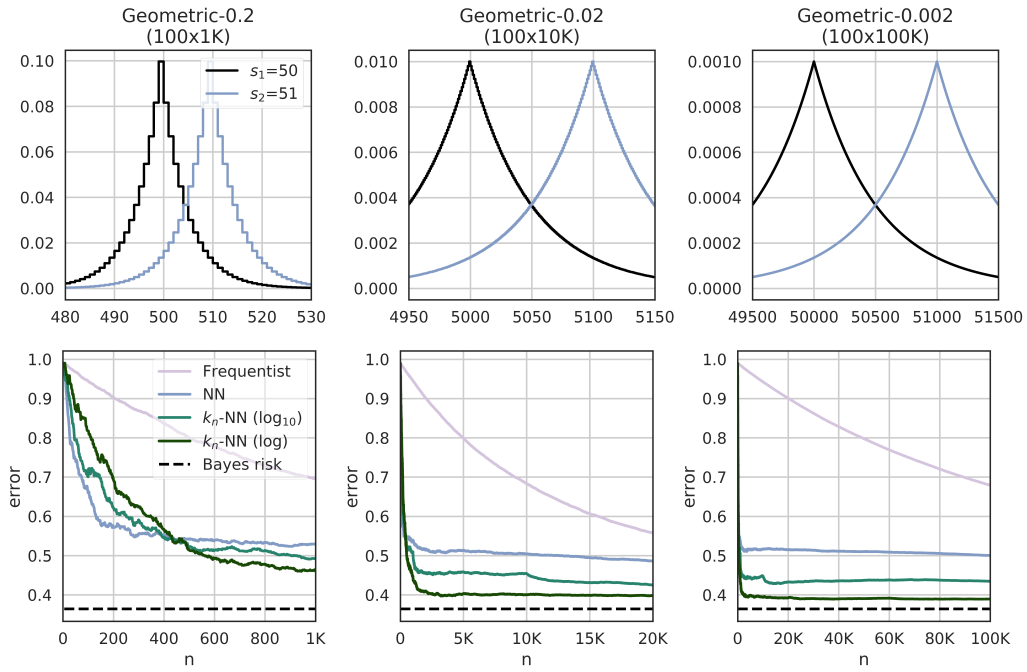


Fig. 3. Estimates' convergence for Geometric systems when varying the ratio $|\mathbb{O}|/|\mathbb{S}|$. The respective distributions are shown in the top figure for two adjacent secrets $s_1 \sim s_2$.

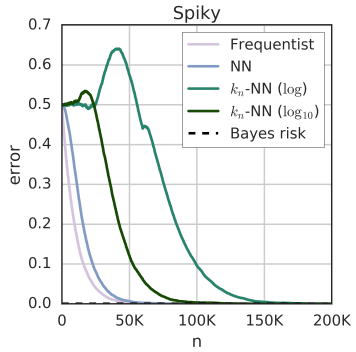


Fig. 4. Estimates' convergence for a Spiky system (2x10K).

- i is not in training data but either $i+1$ or $i-1$ are, or
- $i, i\pm1, i\pm2$ are not in training data but either $i+3$ or $i-3$ are, or
- ...etc.

Hence we have:

$$\begin{aligned} r(i) &= P(d(i,o) = 2l+1) = \\ &= P(i \notin T^n, i+1 \in T^n) + P(i \notin T^n, i-1 \in T^n) + \dots \\ &= 2 \cdot \sum_{\ell=0}^{q/4-1} a^{4\ell+1}(1-a), \end{aligned}$$

where $a = (1 - 1/q)^n$ is the probability that an element $e \in \mathbb{O}$ 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 $i, i\pm1, i\pm2, i\pm2s$, 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

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

we obtain:

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

Since we assume that the distribution on $\mathbb{O}$ 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 $\mathbb{O}$. Hence, let $x = n/q$. Then we have $a = (1 - 1/q)^{qx}$, which, for large q, becomes $a \approx e^{-x}$. Therefore:

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

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

Consider now the frequentist estimate R_x^{Freq}. In this case, given an element $i \in \mathbb{O}$, 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 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:

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

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

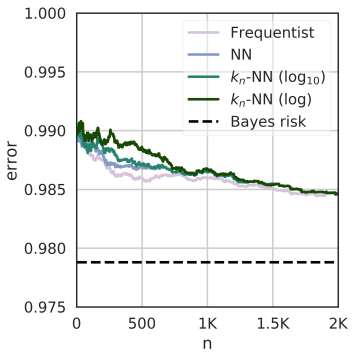


Fig. 5. Estimates' convergence for a Random system (100×100).

C. 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: The channel matrix of a Random system is produced by drawing its elements from the uniform distribution, $\mathcal{C}_{s,o} \xleftarrow{\$} Uni(0,1)$, and normalizing its rows appropriately so that $\sum_{o\in\mathbb{O}} P(o|s) = 1 \quad \forall s \in \mathbb{S}$.

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 VII reports the results.

TABLE VII
RANDOM: EXAMPLES REQUIRED FOR δ-CONVERGENCE.

			k_n-NN	
δ	Freq.	NN	$\log_{10}$	log
0.01	**77**	134	197	495
0.001	668 124	668 124	668 124	668 124

The frequentist estimate is slightly better than kNN for $\delta = 0.01$. However, for stricter convergence requirements ($\delta = 0.001$), all the methods require the same (large) number of examples. Figure 5 show 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 derive a crude approximation of the frequentist Bayes risk estimate.

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

This approximation, derived and studied in Appendix A, makes the very strong assumption that all objects are equally likely to be sampled from $\mu(s,o)$, i.e.: $P(o) = \frac{1}{|\mathbb{O}|}$. However, it is enough to give us an insight on the performance of 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 probability 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. This estimate converges to the Bayes risk asymptotically. However, the probability of observing an object, and thus the size of the object space, is the principal factor influencing its convergence rate.

VI. Application to Location Privacy

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, so that we can use it for comparison. 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, that are the typical mechanisms used to obtain geo-indistinguishability [8], and one of the optimal mechanisms proposed by Oya et al. [9] as a refinement of the optimal mechanism by Shokri et al. [10]. In particular, we will use the mechanism that achieves an optimal trade-off between privacy (measured as residual entropy) and utility loss (measured as expected distance between the true location and the obfuscated one). The construction of such a mechanism relies on an algorithm that was independently proposed by Blahut and by Arimoto to solve a problem in information theory, namely that of achieving an optimal trade-off between the minimization of the distortion rate and the minimization of the mutual information [22]. From now on, we shall refer to this as the Blahut-Arimoto mechanism. Note that the Laplacian is a continuous mechanism, i.e., it outputs obfuscated locations on the 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 $n = 1, 2, ...$ 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 will consider real location data from the `Gowalla` dataset [6], [7], which contains users' checkins and their location in terms of latitude and longitude. We use data from a squared area in San Francisco centered in the point of latlon coordinates (37.755, -122.440), and extending for 1.5 Km in each direction. This input area corresponds to the inner

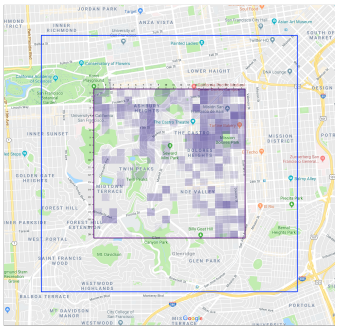


Fig. 6. Area of San Francisco considered for the experiments. The input locations corresponds to the inner square, the output locations to the outer one. The colored cells represent the distribution of the Gowalla checkins.

(purple) square in Figure 6. 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 consists thus of 400 locations. The prior distribution on the secrets (input distribution) is derived from the `Gowalla` checkins, and it is represented in Figure 6 by the different color intensities on the cells of the input grid.

The output area is represented in Figure 6 by the outer (blue) square. It spawns 1050 m (7 cells) more than the input square on every side. The reason we consider a larger area for the output is that the planar Laplace and the planar Geometric naturally expand outside the input square.[3] 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 350×340 cells of size 15×15 Sq m. The size of $\mathbb{O}$ for these mechanisms is therefore $340 \times 340 = 115,600$ cells.

B. Defenses

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

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

where ν is a parameter controlling the level of noise, λ is a normalization factor, and $d(s,o)$ is the Euclidean distance between s and o.

The conditional probability of 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 [22].

C. Results

We have evaluated the estimation's convergence to the Bayes risk as a function of the number of training examples n and for different values of the level of noise: $\nu = \{2, 4, 8\}$.

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

TABLE VIII
CONVERGENCE FOR THE PLANAR GEOMETRIC FOR VARIOUS ν.

				k_n-NN	
ν	δ	Freq.	NN	log 10	log
2	0.1	X	X	26 809	**1 102**
	0.05	X	X	X	**54 914**
4	0.1	X	X	35 942	**2 820**
	0.05	X	X	X	**45 032**
8	0.1	X	X	13 236	**5 249**
	0.05	X	X	X	**19 948**

TABLE IX
CONVERGENCE FOR THE PLANAR LAPLACIAN FOR VARIOUS ν.

				k_n-NN	
ν	δ	Freq.	NN	log 10	log
2	0.1	X	X	X	**259**
4	0.1	X	X	X	**4 008**
8	0.1	X	X	X	**6 135**
	0.05	X	X	X	**19 961**

For the geometric noise the results are shown in Figure 7. As we can see convergence is faster when ν is higher (which means less noise and lower Bayes risk), in line with the results for the syntetic systems of previous section. In all cases, the k-NN methods outperform the frequentist one, as we expected given the presence of a large number of outputs. Table VIII shows the number of examples required to achieve distance δ from the Bayes risk. The presence of the symbol X means that we did not achieve the required level of approximation with 80K examples.

The corresponding results for the Laplacian noise are shown in Figure 7 and in Table IX. In this case we have not shown the frequentist approach, since it does not make sense in the continuous case (the estimate remains on value 1).

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 9 and in Table X.

VII. COMPARISON WITH LEAKWATCH AND LEAKIEST

LeakWatch [4] and leakiEst [5] are the major existing black-box leakage measurement tools. Both are based on the frequentist approach. In this section we compare F-BLEAU with leakiEst, which is an evolution of LeakWatch, and it is more complete: they both compute Shannon mutual information (MI) and min-entropy leakage (ME) on the finite-output case; leakiEst computes also MI in the infinite-output case under some continuity conditions. We perform this comparison for a time side channel on the RFID on European passports, and on the `Gowalla` examples of the previous section.

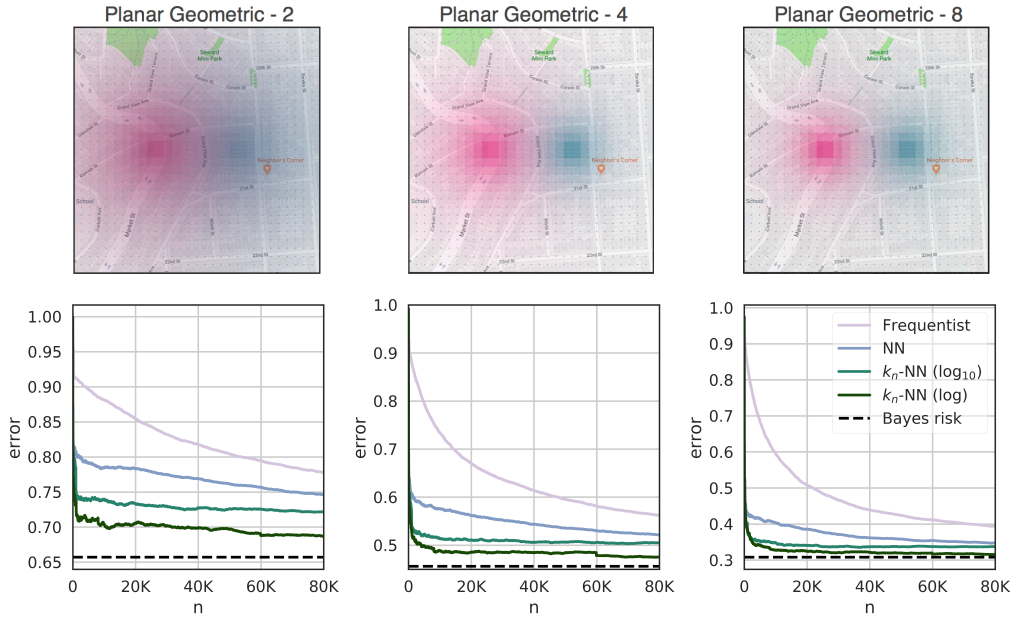


Fig. 7. Estimates' convergence speed for the planar Geometric defense applied to the `Gowalla` dataset, for $\nu = 2$, $\nu = 4$ and $\nu = 8$, respectively. On top of each graph it is represented the distribution of the geometric noise for two adjacent input cells.

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

				k_n-NN	
ν	δ	Freq.	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	5 724	**1 671**	**1 671**
	0.001	X	X	X	X
4	0.1	**220**	**220**	**220**	257
	0.05	503	**502**	509	703
	0.01	**2 029**	**2 029**	2 055	2 404
	0.005	2 197	**2 055**	2 280	2 658
	0.001	X	**2 404**	2 830	3 481
8	0.1	**345**	398	553	1 285
	0.05	1 285	**1 211**	1 343	1 679
	0.01	2 104	**2 017**	2 495	4 190
	0.005	**2 231**	**2 231**	3 433	6 121
	0.001	**3 881**	**3 881**	6 079	7 724

LeakiEst gives two results: i) evidence / no evidence of leakage, and ii) leakage estimation. They are accompanied by confidence indications, and it is possible that leakiEst reports no evidence of leakage, and still a non-zero leakage estimation.

A. Time side channel on e-Passports' RFID

Chothia et al. [11] discovered a side-channel attack in the way the protocols of various European countries exchanged message 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 difference between the response time of the victim's passport and any other passport, that could be used to track the victim. To avoid such attack, Chothia et al. [5] proposed to add padding to the response time and they used LeakiEst to show that after such defense there was no evidence of leakage anymore.

TABLE XI
ESTIMATED LEAKAGE OF EUROPEAN PASSPORTS

Passport	leakiEst: Evidence of leak? (MI)	F-BLEAU: $\mathcal{BL}$
British	yes (0.053)	0.757
German	**no** (0.152)	0.978
Greek	**no** (0.034)	0.938
Irish	yes (0.421)	0.698

We compared F-BLEAU and leakiEst on the data with time padding applied [23], available on the leakiEst web page. The secret space $\mathbb{S}$ contains answers to the binary question: "is this the same passport?"; the dataset is balanced, hence $R^\pi = 0.5$.

On continuous data leakiEst only deals with MI, which is not directly comparable to leakage measures derivable from R^*. However, we can base our comparison on leakiEst's no-leak test: indeed, MI is 0 if and only if $R^* = R^\pi$.

For F-BLEAU, we randomly splitted the data into training (75%) and hold-out set, and then estimated R^* on the latter; we repeated this for 10 different random initialization seeds, and averaged the estimates. Table XI reports the results: there are two cases where leakiEst did not find enough evidence of

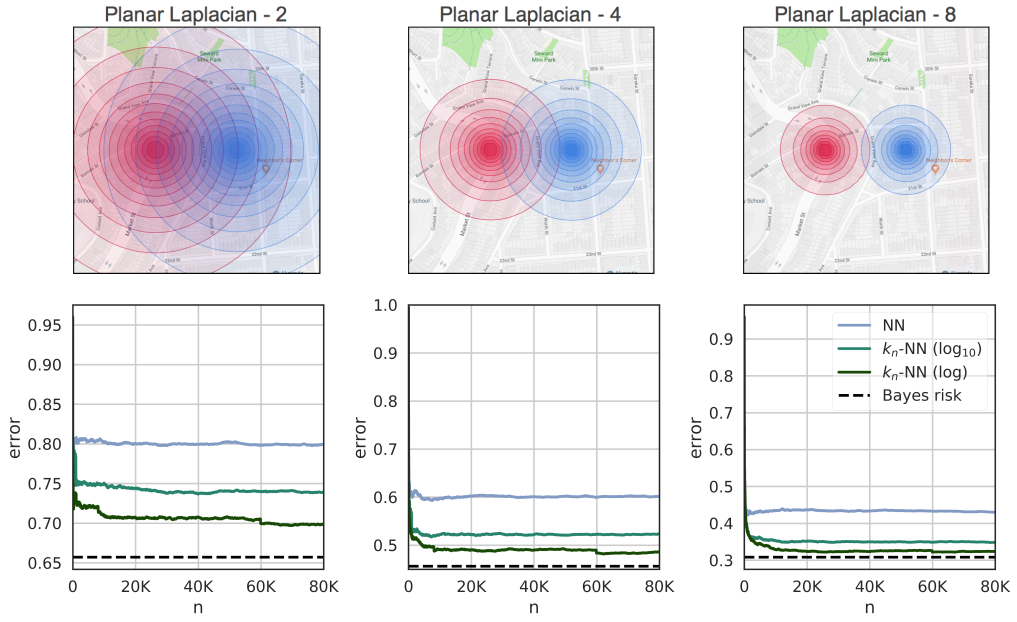


Fig. 8. Estimates' convergence speed for the planar Laplacian defense applied to the `Gowalla` dataset, for $\nu = 2$, $\nu = 4$ and $\nu = 8$, respectively. On top of each graph it is represented the distribution of the geometric noise for two adjacent input cells.

TABLE XII
ESTIMATED LEAKAGE OF PRIVACY MECHANISMS ON GOWALLA DATA

Mechanism	ν	leakiEst: Conf? (ME)	F-BLEAU: ME	True 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: Conf? (MI)	F-BLEAU: ME	True ME
Laplacian	2	**no** (1.150)	1.802	1.987
	4	**no** (1.911)	2.550	2.631
	8	**no** (2.401)	2.970	3.003

leakage, while F-BLEAU shows the leakage is non-negligible: In the case of the German passport, a Bayes error of 0.48 corresponds to a probability of 0.52 to detect the victim's passport, and for the Greek passport, a Bayes error of 0.47 corresponds to a probability of 0.53.

B. `Gowalla` dataset

We compare F-BLEAU with leakiEst on the location privacy mechanisms examined in section VI: Blahut-Arimoto, planar Geometric, and planar Laplacian. The main interest if to verify whether the advantage of F-BLEAU w.r.t. the frequentist approach observed in case of large output sets translates into an advantage also w.r.t. leakiEst. For the first two mechanisms we also compare the estimated values of ME. For the latter this is not possible because the Laplacian is continuous, hence leakiEst can only estimate MI.

We run F-BLEAU and leakiEst on the defended datasets, comprising of $n = 100$K examples. The results are reported in Table XII, where "Conf?" indicate whether leakiEst considers having achieved the intended level of confidence, or not. The values between parentheses indicate the leakage estimate that leakiEst reports anyway. On the planar Geometric leakiEst reports "Too small sample size", and indeed its estimate of ME is quite distant from the true ME. F-BLEAU, on the contrary, provides a quite tight bound (recall that F-BLEAU provides a lower bound of the true ME). The situation is similar for the planar Laplacian.

On the Blahut-Arimoto the situation is more interesting: because of the small number of actual outputs, F-BLEAU and the frequentist approach perform equally well (cfr. section VI), hence we were expecting a similar outcome from leakiEst. This was not the case: on the Blahut-Arimoto leakiEst still reports "Too small sample size". However, we think this is because leakiEst takes into account the number of outputs declared, instead of the actual number generated with the examples. Indeed, its ME estimate is close to ours. Hence this problem should be easy to fix simply by inferring the output size from the examples (this is the meaning of the "*" in the leakiEst column in Table XII).

VIII. CONCLUSION AND FUTURE WORK

We showed that the black-box leakage of a system, measured until now with classical statistics and information theory paradigms, can be effectively estimated via ML techniques. We proposed 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 systems and real-world data. This allows to tackle problems

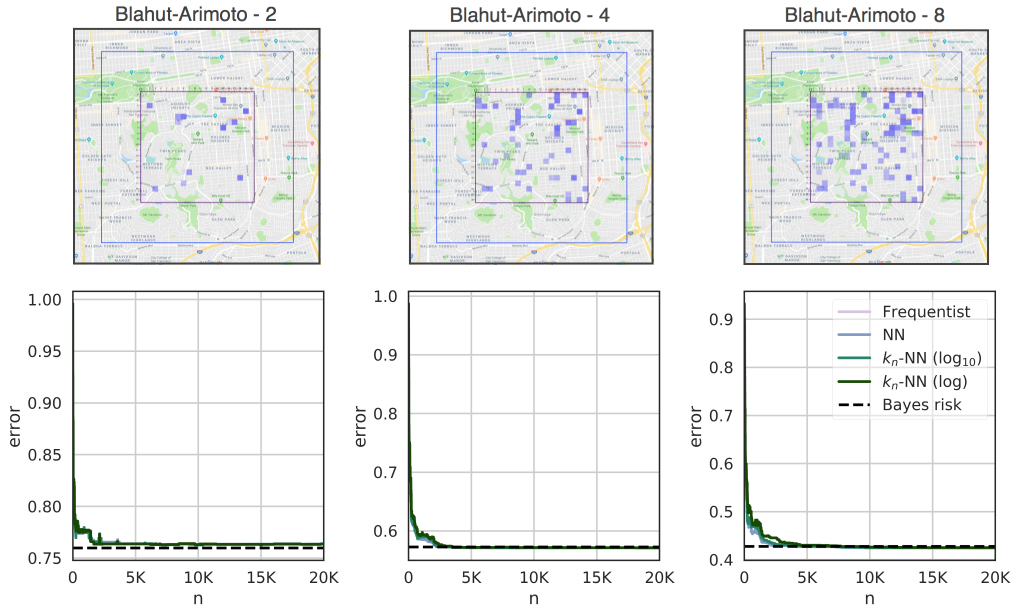


Fig. 9. Estimates' convergence speed for the planar Laplacian defense applied to the `Gowalla` dataset, for $\nu = 2$, $\nu = 4$ and $\nu = 8$, respectively. On top of each graph it is represented the distribution of the output probability as produced by the mechanism. All the outputs with non-null probability turn out to be inside the input square. The Blahut-Arimoto noise for two adjacent input cells distributes in on the outputs with non-null probability in a way similar to the laplacians. The outputs are originally points on the 340×340 output grid, but here are mapped on the coarser 20×20 grid for the sake of visual clarity.

that were impossible until now, and it sets a paradigm change in the QIF literature: thanks to the natural equivalence we discovered between ML and black-box leakage estimation, many results from the ML theory can be now imported into QIF (and vice versa).

Empirical evidence shows that, in general, our nearest neighbor techniques either are equivalent or they (often substantially) outperform the standard frequentist approach in terms of the number of examples required for convergence. In particular, they excel whenever there is a notion of metric in the output space: when the frequentist approach needs to make a prediction for an unseen observation, it has to guess the secret according to priors; nearest neighbor methods can exploit the information of neighboring observations.

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 corresponding to the largest leakage.

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).

A fundamental advantage of the ML formulation, as opposed to the standard approach, is that it gives immediate guarantees for systems with 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 question. We remark, however, that for continuous systems it will not be possible to define confidence intervals (or to prove convergence rates) under our weak assumptions [19]; this constraint applies to any leakage estimation method.

We reiterate, although, 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.

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 *International Conference on Foundations of Software Science and Computational Structures*. 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] 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.

[5] ——, "A tool for estimating information leakage," in *International Conference on Computer Aided Verification*. Springer, 2013, pp. 690–695.

[6] "The gowalla dataset." [Online]. Available: https://snap.stanford.edu/data/loc-gowalla.html

[7] 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

[8] 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.
[9] 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
[10] 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.
[11] 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.
[12] 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.
[13] M. Boreale and M. Paolini, "On formally bounding information leakage by statistical estimation," in *International Conference on Information Security*. Springer, 2014, pp. 216–236.
[14] 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
[15] 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.
[16] G. Cherubin, "Bayes, not naïve: Security bounds on website fingerprinting defenses," *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 4, pp. 215–231, 2017.
[17] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.
[18] D. H. Wolpert, "The lack of a priori distinctions between learning algorithms," *Neural computation*, vol. 8, no. 7, pp. 1341–1390, 1996.
[19] L. Devroye, L. Györfi, and G. Lugosi, *A probabilistic theory of pattern recognition*. Springer Science & Business Media, 2013, vol. 31.
[20] C. J. Stone, "Consistent nonparametric regression," *The annals of statistics*, pp. 595–620, 1977.
[21] 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
[22] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. John Wiley & Sons, Inc., 2006.
[23] Example: e-passport traceability. School of Computer Science - leakiEst. [Online]. Available: www.cs.bham.ac.uk/research/projects/infotools/leakiest/examples/epassports.php
[24] 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.
[25] 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.
[26] 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
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:

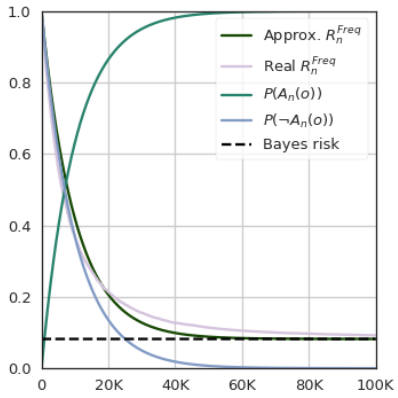


Fig. 10. Approximation of the frequentist estimate as n grows for $R^* \approx 0.08$, $|O| = 10K$, and $|S| = 1K$; the approximation is compared with the real frequentist estimate R_n^{Freq}.

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 appears 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 10.

Let $A_n(o)$ denote the event "observation o appears in a training set of n examples"; because of assumption 1), $P(A_n(o)) = 1 - \left(1 - \frac{1}{|\mathbb{O}|}\right)^n$. The conditional Bayes risk estimated with a frequentist approach given n examples is:

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

Assumptions 2) and 3) were used in the last step. From this

expression, we derive the frequentist estimate of R^* t step n:

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

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 $n \geq -\frac{\log 2}{\log\left(1-\frac{1}{|\mathbb{O}|}\right)}$ the frequentist approach starts approximating using the actual Bayes risk (Fig. 10).

APPENDIX B
GOWALLA DETAILS

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

TABLE XIII
TRUE BAYES RISK AND UTILITY FOR GOWALLA DATASET DEFENDED UNDER 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

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

System (dataset)		$\lvert\mathbb{S}\rvert$	$\lvert\mathbb{O}\rvert$
Time side channel	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 [24] 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 $(o_1, ..., o_m)$, 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 [25], 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 [25], which works as follows. Consider, for instance, decryption for the RSA cryptosystem: $m^d(modN)$, for some decryption key d; the system first computes $m \cdot r^e$, where e is the encryption key and r is 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., [24]).

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 [26].

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 $w \in \{4, 6, .., 12\}$, and for all the values $(m_i, s_i) \in \{0, ..., 2^w - 1\}^2$, we run the exponentiation algorithm to compute m^s, and measure its execution time o_i. As with our application to location privacy (section VI), 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 11 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 XIV 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.

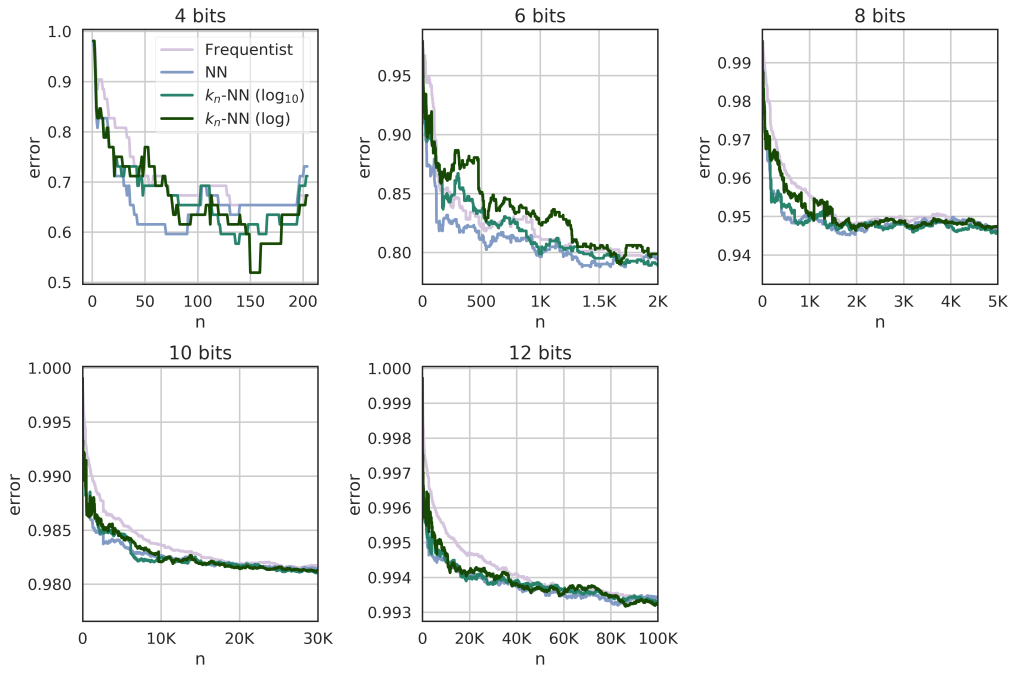


Fig. 11. Convergence of the estimates for the time side channel attack to the exponentiation algorithm as the bit size of the operands increases.