Proceedings of Machine Learning Research vol 98:1–33, 2019 30th International Conference on Algorithmic Learning Theory

A minimax near-optimal algorithm for adaptive rejection sampling

Juliette Achddou JULIETTE.ACHDOU@GMAIL.COM
Numberly (1000mercis Group)
Paris, France

Joseph Lam-Weil JOSEPH.LAM@OVGU.DE
Otto-von-Guericke University
Magdeburg, Germany

Alexandra Carpentier ALEXANDRA.CARPENTIER@OVGU.DE
Otto-von-Guericke University
Magdeburg, Germany

Gilles Blanchard GILLES.BLANCHARD@MATH.UNI-POTSDAM.DE
Potsdam University
Potsdam, Germany

Editors: Aurélien Garivier and Satyen Kale

Abstract

Rejection Sampling is a fundamental Monte-Carlo method. It is used to sample from distributions admitting a probability density function which can be evaluated exactly at any given point, albeit at a high computational cost. However, without proper tuning, this technique implies a high rejection rate. Several methods have been explored to cope with this problem, based on the principle of adaptively estimating the density by a simpler function, using the information of the previous samples. Most of them either rely on strong assumptions on the form of the density, or do not offer any theoretical performance guarantee. We give the first theoretical lower bound for the problem of adaptive rejection sampling and introduce a new algorithm which guarantees a near-optimal rejection rate in a minimax sense.

Keywords: Adaptive rejection sampling, Minimax rates, Monte-Carlo, Active learning.

1. Introduction

The breadth of applications requiring independent sampling from a probability distribution is sizable. Numerous classical statistical results, and in particular those involved in machine learning, rely on the independence assumption. For some densities, direct sampling may not be tractable, and the evaluation of the density at a given point may be costly. Rejection sampling (RS) is a well-known Monte-Carlo method for sampling from a density f on $\mathbb{R}^d$ when direct sampling is not tractable (see Von Neumann, 1951, Devroye, 1986). It assumes access to a density g, called the proposal density, and a positive constant M, called the rejection constant, such that f is upper-bounded by Mg, which is called the *envelope*. Sampling from g is assumed to be easy. At every step, the algorithm draws a proposal sample X from the density g and a point U from the uniform distribution on $[0, 1]$, and accepts X if U is smaller than the ratio of $f(X)$ and $Mg(X)$, otherwise it rejects X. The algorithm outputs all accepted samples, which can be proven to be independent and identically distributed samples from the density f. This is to be contrasted with Markov Chain Monte Carlo (MCMC) methods which produce a sequence of non dependent samples and therefore fulfill a different objective. Besides, the application of rejection sampling includes variational inference: Naesseth et al. (2016, 2017) generalize the reparametrization trick to distributions which can be generated by rejection sampling.

1.1. Adaptive rejection sampling

Rejection sampling has a very intuitive geometrical interpretation. Consider the variable $Z = (X, Mg(X)U)$, where X, M, g and U are defined as above. As shown in Figure 4 in the Supplementary Material, Z has a uniform distribution on the region under the graph of Mg, and the sample is accepted if it falls into the region under the graph of f. Conditional to acceptance, Z is then drawn uniformly from the area under the graph of f. Thus X is drawn from the distribution with density f. The acceptance probability is the ratio of the two areas, $1/M$. This means that the closer g is to f and M to 1, the more samples are accepted. The goal is hence to find a good envelope of f in order to obtain a number of rejected samples as small as possible. In the absence of prior knowledge on the target density f, the proposal is typically the uniform density on a set including the support of f (here assumed to be compact), and the rejection constant M is set as an upper bound on f. Consequently, this method leads to rejecting many samples in most cases and f is evaluated many times uselessly.

Adaptive rejection sampling is a variant motivated by the high number of rejected samples mentioned above. Given n, a *budget* of evaluations of f, the goal is to maximize $\hat{n}$, the number of output samples which have to be drawn independently from f. In other words, the ratio $\frac{n-\hat{n}}{n}$, also called *rejection rate*, is to be made as small as possible, like in standard rejection sampling. To achieve this maximal number of output samples, adaptive rejection sampling methods gradually improve the proposal function and the rejection constant by using the information given by the evaluations of f at the previous proposal samples. These samples are used to estimate and tightly bound f from above.

1.2. Literature review

Closely related works. A recent approach in Erraqabi et al. (2016), pliable rejection sampling (PRS), allows sampling from multivariate densities satisfying mild regularity assumptions. In particular the function f is of a given s-Hölder regularity. PRS is a two-step adaptive algorithm, based on the use of non-parametric kernel methods for estimating the target density. Assume that PRS is given a budget of n evaluations of the function f. For a density f defined on a compact domain, PRS first evaluates f on a number $N < n$ of points uniformly drawn in the domain of f. It uses these evaluations to produce an estimate of the density f using Kernel regression. Then it builds a proposal density using a high probability confidence bound on the estimate of f. The associated rejection constant is then the renormalization constant. The proposal density multiplied by the rejection constant is proven to be with high probability a correct envelope, i.e., an upper bound for f. PRS then applies rejection sampling $n - N$ times using such an envelope. This method provides

with high probability a *perfect sampler*, i.e., a sampler which outputs *i.i.d. samples from the density* f. It also comes with efficiency guarantees. Indeed in dimension d, if $s \leq 2$ ($s > 1$ means that f is $\mathcal{C}^{1,s-1}$) and for n large enough, PRS reaches an average rejection rate of the order of $(\log(nd)/n)^{\frac{s}{3s+d}}$. This means that it asymptotically accepts almost all the samples. However, there is no guarantee that this rate might not be improved using another algorithm. Indeed, no lower bound on the rejection rate over all algorithms is presented.
Another recent related sampling method is A* sampling (Maddison et al., 2014). It is close to the OS* algorithm from Dymetman et al. (2012) and relies on an extension of the Gumbel-max trick. The trick enables the sampling from a categorical distribution over classes $i \in [1, \ldots, n]$ with probability proportional to $\exp(\phi(i))$, where ϕ is an unnormalized mass. It uses the following property of the Gumbel distribution. Adding Gumbel noise to each of the $\phi(i)$'s and taking the argmax of the resulting variables returns i with a probability proportional to $\exp(\phi(i))$. Then, the authors generalize the notion of Gumbel-max trick to a continuous distribution. This method shows good empirical efficiency in the number of evaluations of the target density. However, the assumption that the density can be decomposed into a bounded function and a function, that is easy to integrate and sample from, is rarely true in practice.

Other related works. Gilks and Wild (1992) introduced ARS: a technique of adaptive rejection sampling for one-dimensional log-concave and differentiable densities whose derivative can be evaluated. ARS sequentially builds a tight envelope of the density by exploiting the concavity of $\log(f)$ in order to bound it from above. At each step, it samples a point from a proposal density. It evaluates f at this point, and updates the current envelope to a new one which is closer to f. The proposal density and the envelope thus converge towards f, while the rejection constant converges towards 1. The rejection rate is thereby improved. Gilks (1992) also developed an alternative to this ARS algorithm for the case where the density is not differentiable or the derivative can not be evaluated. The main difference with the former method is that the computation of the new proposal does not require any evaluation of the derivative. For this algorithm, as for the one presented in Gilks et al. (1995), the assumption that the density is log-concave represents a substantial constraint in practice. In particular, it restrains the use of ARS to unimodal densities.
An extension from Hörmann (1995) of ARS adapts it to T-concave densities, with T being a monotonically increasing transformation. However, this method still cannot be used with multimodal densities. In 1998, Evans and Swarz proposed a method applicable to multimodal densities presented in Evans and Swartz (1998) which extends the former one. It deals with T-transformed densities and spots the intervals where the transformed density is concave or convex. Then it applies an ARS-like method separately on each of these intervals. However it needs access to the inflection points, which is a strong requirement. A more general method in Görür and Teh (2011) consists of decomposing the log of the target density into a sum of a concave and convex functions. It deals with these two components separately. An obvious drawback of this technique is the necessity of the decomposition itself, which may be a difficult task. Similarly, Martino and Míguez (2011) deal with cases where the log-density can be expressed as a sum of composition of convex functions and of functions that are either convex or concave. This represents a relatively broad class of functions; other variants focusing on the computational cost of ARS have been explored in

Martino (2017); Martino and Louzada (2017).
For all the methods previously introduced, no theoretical efficiency guarantees are available.

A further attempt at improving simple rejection sampling resulted in Adaptive Rejection Metropolis Sampling (ARMS) (Gilks et al., 1995). ARMS extends ARS to cases where densities are no longer assumed to be log-concave. It builds a proposal function whose formula is close to the one in Gilks (1992). This time however, the proposal might not be an envelope, which would normally lead to oversampling in the regions where the proposal is smaller than the density. In ARMS, this is compensated with a Metropolis-Hastings control-step. One drawback of this method is that it outputs a Markov Chain, in which the samples are correlated. Moreover, the chain may be trapped in a single mode. Improved adaptive rejection Metropolis (Martino et al., 2012) modifies ARMS in order to ensure that the proposal density tends to the target density. In Meyer et al. (2008) an alternative is presented that uses polynomial interpolations as proposal functions. However, this method still yields correlated samples.

Markov Chain Monte Carlo (MCMC) methods (Metropolis and Ulam, 1949; Andrieu et al., 2003) represent a very popular set of generic approaches in order to sample from a distribution. Although they scale with dimension better than rejection sampling, they are not perfect samplers, as they do not produce i.i.d. samples, and can therefore not be applied to achieve our goals. Variants producing independent samples were proposed in Fill (1997); Propp and Wilson (1998). However, to the best of our knowledge, no theoretical studies on the rejection rate of these variants is available in the literature.

Importance sampling is a problem close to rejection sampling, and adaptive importance sampling algorithms are also available (see e.g., Oh and Berger, 1992; Cappé et al., 2008; Ryu and Boyd, 2014). Among them, the algorithm in Zhang (1996) sequentially estimates the target function, whose integral has to be computed using kernel regression, similarly to the approach of Erraqabi et al. (2016). A recent notable method regarding discrete importance sampling was introduced in Canévet et al. (2016). In Delyon and Portier (2018), adaptive importance sampling is shown to be efficient in terms of asymptotic variance.

1.3. Our contributions

The above mentioned sampling methods either do not provide i.i.d samples, or do not come with theoretical efficiency guarantees, apart from Erraqabi et al. (2016) or Zhang (1996); Delyon and Portier (2018) in importance sampling. In the present work, we propose the Nearest Neighbour Adaptive Rejection Sampling algorithm (NNARS), an adaptive rejection sampling technique which requires f to have s-Hölder regularity (see Assumption 3). Our contributions are threefold, since NNARS:

- is a *perfect sampler* for sampling from the density f.
- offers an *average rejection rate of order* $\log(n)^2 n^{s/d}$, if $s \leq 1$. This significantly improves the state of the art average rejection rate from Erraqabi et al. (2016) over s-Hölder densities, which is of order $(\log(nd)/n)^{\frac{s}{3s+d}}$.
- matches a *lower bound for the rejection rate* on the class of all adaptive rejection sampling algorithms and all s-Hölder densities. It gives an answer to the theoretical problem of quantifying the difficulty of adaptive rejection sampling in the minimax

sense. So NNARS offers a near-optimal average rejection rate, in the minimax sense over the class of Hölder densities.

NNARS follows a common approach to that of most adaptive rejection sampling methods. It relies on non-parametric estimation of f. It improves this estimation iteratively, and as the latter gets closer to f, the envelope also approaches f. Our improvements consist of designing an optimal envelope, and updating the envelope as we get more information at carefully chosen times. This leads to an average rejection rate for NNARS which is minimax near-optimal (up to a logarithmic term) over the class of Hölder densities. No adaptive rejection algorithm can perform significantly better on this class. The proof of the minimax lower bound is also new to the best of our knowledge.

The optimal envelope we construct is a very simple one. For every known point of the target density f, we use the regularity assumptions on f in order to construct an envelope which is piecewise constant. It stays constant in the neighborhood of every known point of f. Figure 1 depicts NNARS' first steps on a mixture of Gaussians in dimension 1.

In the second section of this paper, we set the problem formally and discuss the assumptions that we make. In the third section, we introduce the NNARS algorithm and provide a theoretical upper bound on its rejection rate. In the fourth section, we present a minimax lower bound for the problem of adaptive rejection sampling. In the fifth section, we discuss our method and detail the open questions regarding NNARS. In the sixth section, we present experimental results on both simulated and real data that compare our strategy with state of the art algorithms for adaptive rejection sampling. The implementation of the code of NNARS can be found on the following webpage: https://github.com/jlamweil/NNARS. Finally, the Supplementary Material contains the proofs of all the results presented in this paper.

2. Setting

Let f be a bounded density defined on $[0,1]^d$. The objective is to provide an algorithm which outputs as many i.i.d. samples drawn according to f as possible, with a fixed number n of evaluations of f. We call n the budget.

2.1. Description of the problem

The framework that we consider is *sequential and adaptive rejection sampling.*

Adaptive Rejection Sampling (ARS). Set $\mathcal{S} = \emptyset$ and let n be the budget. An ARS method sequentially performs n steps At each step $t \leq n$, the samples $\{X_1, \ldots, X_{t-1}\}$ collected until t, each in $[0,1]^d$, are known to the learner, as well as their images by f. The learner A chooses a positive constant M_t and a density g_t defined on $[0,1]^d$ that both depend on the previous samples and on the evaluations of f at these points $\{(X_1, f(X_1)), \ldots, (X_{t-1}, f(X_{t-1}))\}$. Then the learner A performs a rejection sampling step with the proposal and rejection constant (g_t, M_t), as depicted in Algorithm 1. It generates a point X_t from g and a variable U_t that is independent from every other variable and drawn uniformly from $[0,1]$. X_t is accepted as a sample from f if $U_t \leq \frac{f(X_t)}{M_t g_t(X_t)}$ and rejected otherwise. If it is accepted, the output is X_t, otherwise the output is $\emptyset$. Once the rejection sampling step is complete, the learner adds the output of this rejection sampling step to $\mathcal{S}$. The learner iterates until the budget n of evaluations of f has been spent.

Algorithm 1: Rejection Sampling Step with (f, g, M): **RSS**(f, g, M)

```
Input  : Target density $f$, proposal density $g$, rejection constant $M$.
Output: Either a sample $X$ from $f$, or nothing.
Sample $X \sim g$ and $U \sim \mathcal{U}_{[0,1]}$.
if $U \leq \frac{f(X)}{Mg(X)}$ then
    output $X$.
end
else
    output $\emptyset$.
end
```

Definition 1 (Class of Adaptive Rejection Sampling (ARS) Algorithms) *An algorithm A is an ARS algorithm if, given f and n, at each step $t \in \{1 \dots n\}$:*

- *A chooses a density g_t, and a positive constant M_t, depending on $\Big\{(X_1, f(X_1)), \dots, (X_{t-1}, f(X_{t-1}))\Big\}$.*
- *A performs a Rejection Sampling Step with (f, g_t, M_t).*

The objective of an ARS algorithm is to sample as many i.i.d. points according to f as possible.

Theorem 2 *Given access to a positive, bounded density f defined on $[0,1]^d$, any Adaptive Rejection Sampling algorithm (as described above) satisfies:*
if $\forall t \leq n$, $\forall x \in [0,1]^d$, $f(x) \leq M_t g_t(x)$, the output $\mathcal{S}$ contains i.i.d. samples drawn according to f.

Definition of the loss. Theorem 2 gives a sufficient condition under which an adaptive rejection sampling algorithm is a *perfect sampler*, that is, it outputs i.i.d. samples. Its proof is given in the Supplementary Material, see Appendix B.

If the learner is a perfect sampler at every step, we define the loss as $L_n = n - \#\mathcal{S}$, which corresponds to the number of rejected samples. Otherwise, we just set $L_n = n$. Finally, we note that the rejection rate is L_n/n.

Remark on the loss. Let $\mathcal{A}$ be the set of ARS algorithms defined in Definition 1. Note that for any algorithm $A \in \mathcal{A}$, the loss $L_n(A)$ can be interpreted as a *regret*. Indeed, a learner that can sample directly from f would not reject a single sample, and would hence achieve $L_n^* = 0$. So $L_n(A)$ is equal to the difference between $L_n(A)$ and L_n^*. Hence $L_n(A)$ is the cost of not knowing how to sample directly from f.

2.2. Assumptions

We make the following assumptions on f. They will be used by the algorithm and for the theoretical results.

Assumption 3

- *The function f is (s, H)-Hölder for some $0 < s \leq 1$ and $H \geq 0$, i.e., $\forall x, y \in [0,1]^d,\ |f(x) - f(y)| \leq H\|x-y\|_\infty^s$, where $\|u\|_\infty = \max_i |u_i|$;*
- *There exists $0 < c_f \leq 1$ such that $\forall x \in [0,1]^d,\ c_f < f(x)$.*

Let $\mathcal{F}_0 := \mathcal{F}_0(s, H, c_f, d)$ denote the set of functions satisfying Assumption 3 for given $0 < s \leq 1$, $H \geq 0$ and $0 < c_f \leq 1$.

Remarks. Here the domain of f is assumed to be $[0,1]^d$, but it could without loss of generality be relaxed to any hyperrectangle of $\mathbb{R}^d$. Besides, for any distribution with sub-Gaussian tails, this assumption is almost true. In practice, the diameter of the support is bounded by $O(\sqrt{\log n})$, where n is the number of evaluations, because of the vanishing tail property. The assumption of Hölder regularity is a usual regularity assumption in order to control for rates of convergence. It is also a mild one, considering that s can be chosen arbitrarily close to 0. Note however that we assume the knowledge of s and H for the NNARS algorithm. Since f is a Hölder regular density defined on the unit cube, we can obtain the following upper bound: $f(x) \leq 1 + H\ \forall x \in [0,1]^d$. As for the assumption involving the constant c_f, it is widespread in non-parametric density estimation. Besides, the algorithm will still produce exact independent samples from the target distribution without the latter assumption. It is important to note that f is chosen as a probability density for clarity, but it is not a required hypothesis. In the proofs, we study the general case when f is not assumed to be a probability density.

3. The NNARS Algorithm

The NNARS algorithm proceeds by constructing successive proposal functions g_t and rejection constants M_t that gradually approach f and 1, respectively. In turn, an increased number of evaluations of f should result in a more accurate estimate and thus in a better upper bound.

3.1. Description of the algorithm

The algorithm outlined in Algorithm 2 takes as inputs the budget n, and c_f, H, s as defined in Assumption 3. Let $\mathcal{S}$ denote its output. At each round k, the algorithm performs a number of RSS steps with specifically chosen g_k and M_k. We call χ_k the set of points generated at round k and of their images by f, whether they get accepted or rejected.

Initialization. The sets $\mathcal{S}$ and $\chi_k, k \in \mathbb{N}$ are initialized to $\emptyset$. g_1 is a uniform proposal on $[0,1]^d$. $M_1 = 1 + H$ is an upper bound on f and $N = N_1 = \lceil 2(10H)^{d/s} \log(n) c_f^{-1-d/s} \rceil$. For any function h defined on $[0,1]^d$, we set $I_h = \int_{[0,1]^d} h(x)dx$.

Loop. The algorithm proceeds in $K = \lceil \log_2(\frac{n}{N}) \rceil$ rounds, $\lceil \, \rceil$ is the ceiling function, and $\log_2$ is the logarithm in base 2.
Each round $k \in \{1, \ldots, K\}$ consists of the following steps.

1. Perform a Rejection Sampling Step $\text{RSS}(f, g_k, M_k)$ N_k times. Add the accepted samples to $\mathcal{S}$. All proposal samples as well as their images by f produced in the Rejection Sampling Step are stored in χ_k, whether they are rejected or not.

2. Build an estimate $\hat{f}_{\cup_{i \le k}\chi_i}$ of f based on the evaluations of f at all points stored in $\cup_{i \le k}\chi_i$, thanks to the Approximate Nearest Neighbor Estimator, referred to in Definition 4, applied to the set χ_k.

3. Compute the proposal with the formula:

$$g_{k+1}(x) = \frac{\hat{f}_{\cup_{i \le k}\chi_i}(x) + \hat{r}_{\cup_{i \le k}\chi_i}}{I_{\hat{f}_{\cup_{i \le k}\chi_i}} + \hat{r}_{\cup_{i \le k}\chi_i}}, \tag{1}$$

   and the rejection constant with the formula:

$$M_{k+1} = I_{\hat{f}_{\cup_{i \le k}\chi_i}} + \hat{r}_{\cup_{i \le k}\chi_i}, \tag{2}$$

   where $\hat{r}_{\cup_{i \le k}\chi_i}$ is defined in Equation (3) below, in Definition 4. Note that g_{k+1} and M_{k+1} are indexed here by the number of the round, unlike in the last section where the index was the current time.

4. If $k < K - 1$, set N_{k+1} as $2N_k = 2^k N$. Otherwise $N_K = n - (2^{K-1} - 1)N$.

Finally, the algorithm outputs $\mathcal{S}$, the set of accepted samples that have been collected.

Definition 4 (Approximate Nearest Neighbor Estimator applied to χ) *Let f be a positive density satisfying Assumption 3. We consider a set of $\tilde{N}$ points and their images by f, $\chi = \{(X_1, f(X_1)), \ldots, (X_{\tilde{N}}, f(X_{\tilde{N}}))\}$. Let us define a set of centers of cells constituting a uniform grid of $[0,1]^d$, namely*

$$\mathcal{C}_{\tilde{N}} = \left\{ 2^{-1}(\lfloor \tilde{N}^{\frac{1}{d}} \rfloor + 1)^{-1} u, u \in \{1, \ldots, 2(\lfloor \tilde{N}^{\frac{1}{d}} \rfloor + 1) - 1\}^d \right\}.$$

The cells are of side-length $1/(\lfloor N^{\frac{1}{d}} \rfloor + 1)$. For $x \in [0,1]^d$, write $C_{\tilde{N}}(x) = \arg\min_{u \in \mathcal{C}_{\tilde{N}}} \|x - u\|_\infty$.

We define the approximate nearest neighbor estimator as the piecewise-constant estimator $\hat{f}_\chi$ of f by $\forall x \in [0,1]^d$, $\hat{f}_\chi(x) = \hat{f}_\chi(C_{\tilde{N}}(x)) = f\left(X_{i(C_{\tilde{N}}(x))}\right)$, where $i(x) = \arg\min_{i \le \tilde{N}}(\|x - X_i\|_\infty)$.

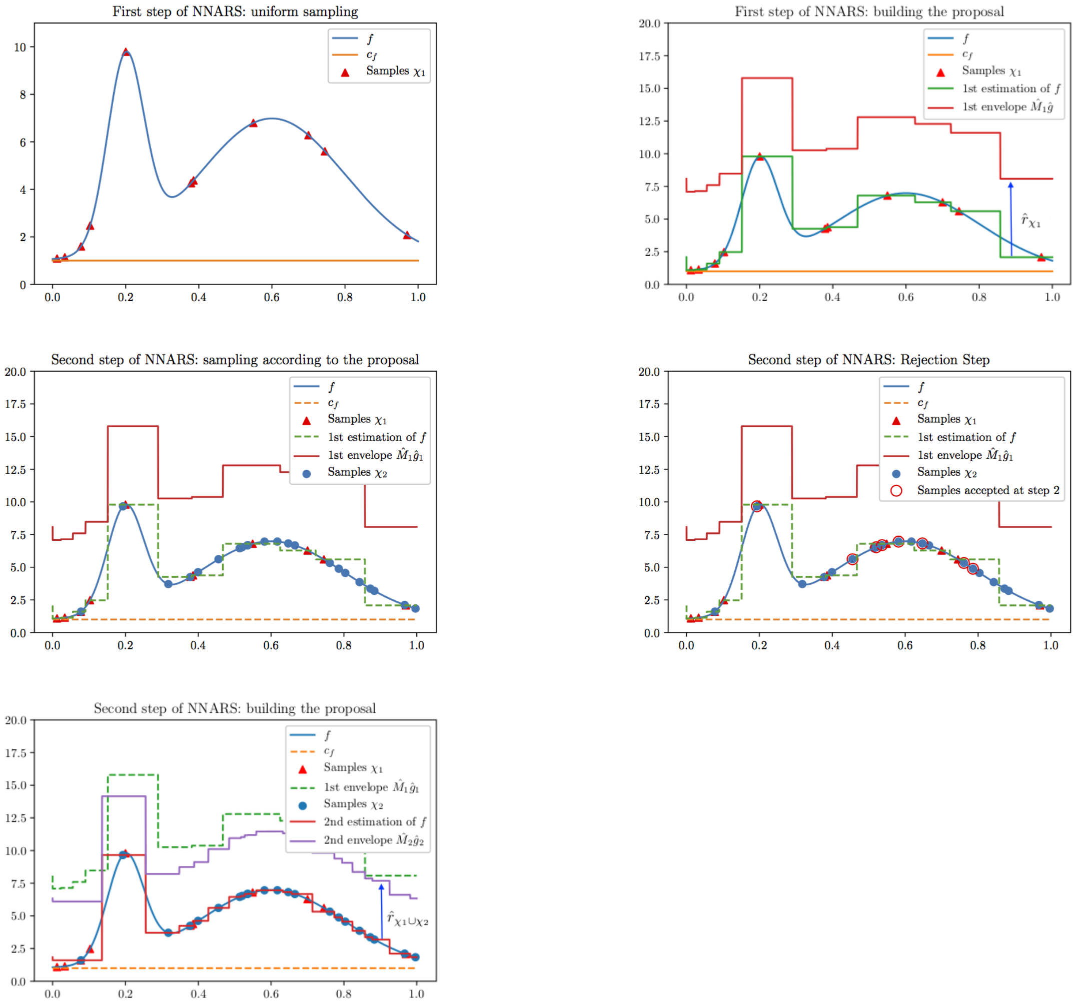
We also define a confidence term as

$$\hat{r}_\chi = H \left(\max_{u \in \mathcal{C}_{\tilde{N}}} \min_{i \le \tilde{N}} \|u - X_i\|_\infty + \frac{1}{2(\lfloor \tilde{N}^{\frac{1}{d}} \rfloor + 1)} \right)^s. \tag{3}$$

Remarks on the proposal densities and rejection constants At each step, the envelope is made up of evaluations of f summed with a positive constant which stands for a confidence term of the estimation. It provides an upper bound for f. Furthermore, the use of nearest neighbour estimation in a noiseless setting implies that this bound is optimal. Besides, the approximate construction of the estimator builds proposal densities which are simple to sample from.

As explained in Lemma 11 in the Supplementary Material, an important remark is that the proposal density g_k from Equation (1) multiplied by the rejection constant M_k from Equation (2) is an envelope of f. This means $M_k g_k \geq f$ for all $k \leq K$. So by Theorem 2, NNARS is a perfect sampler.

The algorithm is illustrated in Figure 1.


First step of NNARS: uniform sampling

First step of NNARS: building the proposal

Second step of NNARS: sampling according to the proposal

Second step of NNARS: Rejection Step

Second step of NNARS: building the proposal

Figure 1: NNARS' first steps on a mixture of Gaussians (to be read in natural reading direction)

```
Algorithm 2: Nearest Neighbor Adaptive Rejection Sampling
Input  : The budget $n$; the constants $H$, $s$ and $c_f$; the dimension $d$.
Output: The set $\mathcal{S}$ of i.i.d. samples from $f$.
Initialize $\mathcal{S} = \emptyset$, $\chi_k = \emptyset\ \forall k$. Set $N_1 = N$, $g_1 = \mathcal{U}_{[0,1]^d}$, $M_1 = 1 + H$.
for $k = 1$ to $K$ do
    for $i = 1$ to $N_k$ do
        Perform a Rejection Sampling Step RSS$(f, g_k, M_k)$.
        Add the output of RSS to $\mathcal{S}$.
        Add to $\chi_k$ both the sample from $g_k$ collected in the RSS, and its image by $f$.
    end
    Estimate $\hat{f}_{\cup_{i \le k}\chi_k}$ according to Definition 4.
    Compute $g_{k+1}$ and $M_{k+1}$ as in Equations (1) and (2).
    if $k < K - 1$ then
        set $N_{k+1} = 2N_k$.
    end
    else
        set $N_K = n - (2^{K-1} - 1)N$.
    end
end
```

Remark on sampling from the proposal densities g_k in NNARS. The number of rounds is of order $\lfloor \log(n) \rfloor$. The construction of the proposal in NNARS involves at each round k the storage of $|\cup_{i \le k} \chi_i| \propto 2^{k+1}\lfloor \log(n) \rfloor$ values. So the total number of values stored is upper bounded by the budget. At each round, each value corresponds to a hypercube of side-length $1/|\cup_{i \le k} \chi_i|^{1/d}$ splitting $[0,1]^d$ equally. Partitioning the space in this way allows us to efficiently assign a value to every $x \in [0,1]^d$, depending on which cell of the grid x belongs to. Besides, sampling from the proposal amounts to sampling from a multinomial convolved with a uniform distribution on a hypercube. In other words, a cell is chosen multinomially, then a point is sampled uniformly inside that cell, because the proposal is piecewise constant.

The process to sample according to g_k is the following: given $\cup_{i \le k}\chi_i$,

1. Each center of the cells from the grid $u \in \mathcal{C}_{|\cup_{i \le k}\chi_i|}$ is mapped to a value $g_k(u)$.

2. One of the centers $\tilde{C} \in \mathcal{C}_{|\cup_{i \le k}\chi_i|}$ is sampled with probability $g_k(\tilde{C})$.

3. The sample point is drawn according to the uniform distribution on the hypercube of center $\tilde{C}$ and side-length $1/|\cup_{i \le k} \chi_i|^{1/d}$.

3.2. Upper bound on the loss of NNARS

In this section, we present an upper bound for the expectation of the loss of the NNARS sampler. This bound holds under Assumption 3, that only requires n to be large enough in comparison with constants depending on d, s, c_f and H. Related conditions about the sample size are in most theoretical works on Rejection Sampling (Gilks and Wild, 1992, Meyer et al., 2008, Görür and Teh, 2011, Erraqabi et al., 2016).

Assumption 5 (Assumption on n)
Assume that $n \geq 8$ and $N/n \leq 1/(2K^2)$, i.e.,

$$n \geq \left\lceil 2(10H)^{d/s}\log(n)c_f^{-1-d/s}\right\rceil \frac{4\log(n)^2}{\log{(2)}^2} = O(\log(n)^3).$$

Theorem 6 *Let $0 < s \leq 1$, $H \geq 0$ and $c_f > 0$. If f satisfies Assumption 3 with (s, H, c_f) such that $f \in \mathcal{F}_0(s, H, c_f, d)$, then NNARS is a perfect sampler according to f.*

Besides if n satisfies Assumption 5, then

$$\begin{aligned}\mathbb{E}_f L_n(NNARS) \leq& \frac{20}{2^{1-s/d}-1}{c_f}^{-2}(1+\sqrt{2\log 3n})\log^{s/d}(5n)n^{1-s/d}\\ &+ \quad (25+40+2(10Hc_f^{-1})^{d/s})c_f^{-1}\log^2(n) = O(\log^2(n)n^{1-s/d}),\end{aligned}$$

where $\mathbb{E}_f L_n(NNARS)$ is the expected loss of NNARS on the problem defined by f. The expectation is taken over the randomness of the algorithm. This result is uniform over $\mathcal{F}_0(s, H, c_f, d)$.

The proof of this theorem is in the Supplementary Material, see Section C. The loss presented here divided by n is to be interpreted as an upper bound for the expected rejection rate obtained by the NNARS algorithm.

Sketch of the proof. The average number of rejected samples is $\sum_k N_k(1-1/M_k)$, since a sample is accepted at round k with probability $1-1/M_k$. In order to bound the average number of rejected samples, we bound M_k at each round k with high probability.

By Hölder regularity and the definition of $\hat{r}_{\cup_{i\leq k}\chi_i}$ in Equation (3) (in Definition 4), we always have $|\hat{f}_{\cup_{i\leq k}\chi_i} - f| \leq \hat{r}_{\cup_{i\leq k}\chi_i}$, as shown in the proof of Proposition 8. So $M_k = I_{\hat{f}_{\cup_{i\leq k-1}\chi_i}} + \hat{r}_{\cup_{i\leq k-1}\chi_i} \leq I_f + 2\hat{r}_{\cup_{i\leq k-1}\chi_i}$ with $I_f = 1$. Then, we consider the event $\mathcal{A}_{k,\delta} = \{\forall j \leq k,\ \hat{r}_{\cup_{i\leq j}} \leq C_0 H(\log(N_j/\delta)/N_j)^{s/d}\}$, where C_0 is a constant. Now, for each k, on $\mathcal{A}_{k-1,\delta}$, M_k is bounded from above, with a bound of the order of $(\log(N_{k-1}/\delta)/N_{k-1})^{s/d}$. So, on $\mathcal{A}_{K,\delta}$, the average number of rejected samples has an upper bound of the order of $\log(n)^2 n^{1-s/d}$, as presented in Theorem 6.

Now, we prove by induction that event $\mathcal{A}_{k,\delta}$ has high probability, as in the proof of Lemma 12. More precisely, $\mathcal{A}_{k,\delta}$ has probability larger than $1-2k\delta$. At every step k, we verify that g_k is positively lower bounded conditionally on $\mathcal{A}_{k-1,\delta}$. Hence, the probability of having drawn at least one point in each hypercube of the grid with centers $\mathcal{C}_{|\cup_{i\leq k}\chi_i|}$ is high, as shown in the proof of Proposition 9. So the distance from any center to its closest drawn point is upper bounded with high probability. And this implies that $\mathcal{A}_{k,\delta}$ has high probability if $\mathcal{A}_{k-1,\delta}$ has high probability, which gets the induction going. On the other hand, the number of rejected samples is always bounded by n on the small probability event where $\mathcal{A}_{K,\delta}$ does not hold. This concludes the proof.

■

4. Minimax Lower Bound on the Rejection Rate

It is now essential to get an idea of how much it is possible to reduce the loss obtained in Theorem 6. That is why we apply the framework of minimax optimality and complement the upper bound with a lower bound. The minimax lower bound on this problem is the infimum of the supremum of the loss of algorithm A on the problem defined by f; the infimum is taken over all adaptive rejection sampling algorithms A and the supremum over all functions f satisfying Assumption 3. It characterizes the difficulty of the rejection sampling problem. And it provides the best rejection rate that can possibly be achieved by such an algorithm in a worst-case sense over the class $\mathcal{F}_0(s, H, c_f, d)$.

Theorem 7 *For $0 < s \leq 1$, there exists a constant $N(s,d)$ that depends only on s, d and such that for any $n \geq N(s,d)$:*

$$\inf_{A \in \mathcal{A}} \sup_{f \in \mathcal{F}_0(s,1,1/2,d) \cap \{f : I_f = 1\}} \mathbb{E}_f(L_n(A)) \geq 3^{-1} 2^{-1-3s-2d} 5^{-s/d} n^{1-s/d} = O(n^{1-s/d}),$$

where $\mathbb{E}_f(L_n(A))$ is the expectation of the loss of A on the problem defined by f. It is taken over the randomness of the algorithm A.

The proof of this theorem is in Section D, but the following discussion outlines its main arguments.

Sketch of the proof in dimension 1. Consider the setup where firstly n points from f are chosen and evaluated. Secondly, n other points are sampled using rejection sampling with a proposal based only on the n first points. This is related to Definition 13. This setting is easier than that of adaptive rejection sampling, as proven in Lemma 15. Consequently a lower bound for this simpler problem also constitutes a lower bound for adptive rejection sampling. Now, $\mathcal{F}_0(1,1,1/2,1)$ corresponds to one-dimensional $(1,1)$-Hölder functions which are bounded from below by $1/2$. We consider a subset of $\mathcal{F}_0(1,1,1/2,1)$ satisfying Assumption 3. Set $V_n = \{\nu = (\nu_i)_{0 \leq i \leq 4n-1} \mid \nu_i \in \{-1,1\}, ; \sum_{i=0}^{4n-1} \nu_i = 0\}$.

Let us define the bump function $b : [0, 1/(4n)] \to \mathbb{R}^+$ such that for any $\nu \in V_n$:

$$b(x) = \begin{cases} x, & \text{for } x \leq 1/(8n). \\ 1/(4n) - x, & \text{otherwise.} \end{cases}$$

We will consider the following functions $f_\nu : [0,1] \to \mathbb{R}_+^*$ such that for any $\nu \in V_n$:

$$f_\nu(x) = 1 + \nu_i b(x - i/(4n)), \text{ if } i/(4n) \leq x \leq (i+1)/(4n),$$

We note that $f_\nu \in \mathcal{F}_0(1,1,1/2,1)$, for n large enough ensuring that $f_\nu \geq 1/2$.

An upward bump at position i corresponds to $\nu_i = 1$ and a downward bump to $\nu_i = -1$. The construction presented here is analog to the one in the proof of Lemma 17. The function f_ν is entirely determined by the knowledge of ν. It is only possible to determine a ν_i by evaluating f at some $x \in (i/(4n), (i+1)/(4n))$. So with a budget of n, we observe at most n of the $4n$ signs in ν. Among the unobserved ν_i, at least n are positive and n are negative, because $\sum_{i=0}^{4n-1} \nu_i = 0$. Now, we compute the loss. In the case when Mg is not an envelope, the loss simply is n. Now let us consider the case where Mg is an envelope.

The loss is $n(1 - 1/I_{Mg})$. Mg has to account for at least n upward bumps at unknown positions; and the available information is insufficient to distinguish between upward and downward bumps. This results in an envelope that is not tight for the negative ν_i with unknown positions. So a necessary loss is incurred at the downward bumps corresponding to those negative ν_i. This translates as $I_{Mg} - 1 \geq nc_s n^{-(1+s)}$, where c_s is a constant only dependent on s, with $s = 1$ in our case. Finally, we obtain a risk $n(1 - 1/I_{Mg})$ which is of order n^{1-s}, as seen in Lemma 16.

In a nutshell, we first made a setup with more available information than in the problem of adaptive rejection sampling, from Definition 1. Then we restricted the setting to some subspace of $\mathcal{F}_0(1, 1, 1/2, 1)$. This led to our obtaining of a lower bound on the risk for an easier setting. This implies we have displayed a lower bound for the problem of adaptive rejection sampling over $\mathcal{F}_0(1, 1, 1/2, 1)$, too. ■

This theorem gives a lower bound on the minimax risk of all possible adaptive rejection sampling algorithms. Up to a $\log(n)$ factor, NNARS is minimax-optimal and the rate in the lower bound is then the minimax rate of the problem. It is remarkable that this problem admits such a fast minimax rate; the same rate as a standard rejection sampling scheme with an envelope built using the knowledge of n evaluations of the target density (see Setting 13 in the Appendix).

5. Discussion

Theorem 7 asserts that NNARS provides a minimax near-optimal solution in expectation — up to a multiplicative factor of the order of $\log(n)^{s/d}$. This result holds for all adaptive rejection sampling algorithms and densities in $\mathcal{F}_0(s, H, c_f, d)$. To the best of our knowledge, this is the first time a lower bound is proved on adaptive rejection samplers; or that an adaptive rejection sampling algorithm that achieves near-optimal performance is presented. In order to ensure the theoretical rates mentioned in this work, the algorithm requires to know c_f, a positive lower bound for f, and the regularity constants of f: s, and H. Note that to achieve a near-optimal rejection rate, the precise knowledge of s is required. Indeed, replacing the exponent s by a smaller number will result in adding a confidence term $\hat{r}_{\cup_{i \leq k} \mathcal{X}_i}$ to the estimator which is too large. Finally, it will result in a higher rejection rate than if one had set s to the exact Hölder exponent of f. The assumption on c_f implies in particular that f does not vanish. However, as long as it remains positive, c_f can be chosen arbitrarily small, and n has to be taken large enough to ensure that c_f is approximately larger than $\frac{1}{\log \log n}$. When c_f is not available, asymptotically taking c_f of this order will offer a valid algorithm, which outputs independent samples drawn according to f. Moreover taking c_f of this order will still result in a minimax near-optimal rejection rate. Indeed it will approximately boil down to multiplying the rejection rate by a $\log \log n$. Similarly H can be taken of order $\log n$ without hindering the minimax near-optimality. Extending NNARS to non lower-bounded densities is still an open question.

The algorithm NNARS is a perfect sampler. Since our objective is to maximize the number of i.i.d. samples generated according to f, we cannot compare the algorithm with MCMC methods, which provide non-i.i.d. samples. In our setting, they have a loss of n. The same argument is valid for other adaptive rejection samplers that produce correlated samples, like e.g., Gilks et al. (1995); Martino et al. (2012); Meyer et al. (2008). Considering other perfect adaptive rejection samplers, like the ones in e.g., Gilks (1992); Martino and Míguez (2011); Hörmann (1995); Görür and Teh (2011), their assumptions differ in nature from ours. Instead of shape constraint assumptions, like log-concavity, which are often assumed in the quoted literature, we only assume Hölder regularity. Note that log-concavity implies Hölder regularity of order two almost everywhere. Moreover no theoretical results on the proportion of rejected samples are available for most samplers, except possibly asymptotic convergence to 0, which is induced by our result.

PRS (Erraqabi et al., 2016) is the only algorithm with a theoretical guarantee on the rate with the proportion of rejected samples decreasing to 0. But it is not optimal, as explained in Section 1. So the near-optimal rejection rate is a major asset of the NNARS algorithm compared to the PRS algorithm. Besides, PRS only provides an envelope with high probability, whereas NNARS provides it with probability 1 at any time. The improved performance of NNARS compared to PRS may be attributed to the use of an estimator more adapted to noiseless evaluations of f, and to the multiple updates of the proposal.

6. Experiments

Let us compare NNARS numerically with Simple Rejection Sampling (SRS), PRS (Erraqabi et al., 2016), OS* (Dymetman et al., 2012) and A* sampling (Maddison et al., 2014). The value of interest is the sampling rate corresponding to the proportion of samples produced with respect to the number of evaluations of f. This is equivalent to the acceptance rate in rejection sampling. Every result is an average over 10 runs with its standard deviation. The implementation of the code of NNARS can be found on the following webpage: https://github.com/jlamweil/NNARS.

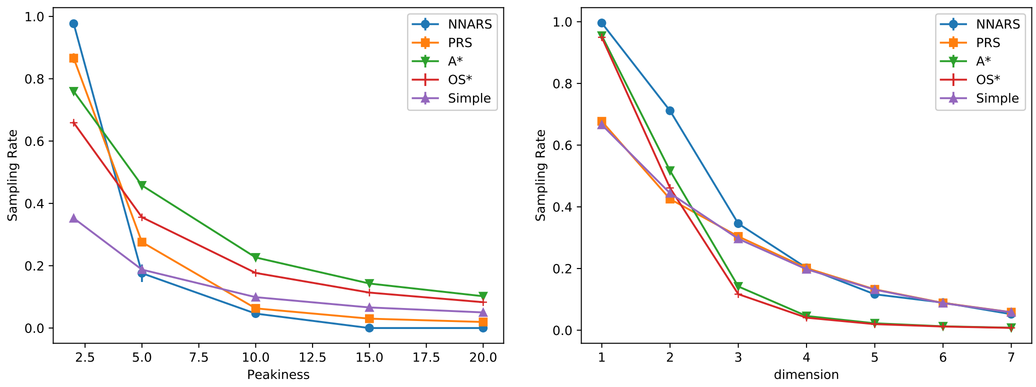
6.1. Presentation of the experiments

EXP1. We first consider the following target density from Maddison et al. (2014): $f(x) \propto e^{-x}/(1+x)^a$, where a is the peakiness parameter. Increasing a also increases the sampling difficulty. In Figure 2a, PRS and NNARS both give good results for low peakiness values, but their sampling rates fall drastically as the peakiness increases. So their results are similar to SRS after a peakiness of 5.0. On the other hand, the rates of A* and OS* sampling decrease more smoothly.

EXP2. For the next experiment, we are interested in how the method scale when the dimension increases and consider a distribution that is related to the one in Erraqabi et al. (2016): $f(x_1, \dots, x_d) \propto \prod_{i\in[|0,1|]^d} \left(2 + \sin\left(4\pi x_i - \frac{\pi}{2}\right)\right)$, where $(x_1, \dots, x_d) \in [0,1]^d$. In Figure 2b, we present the results for d between 1 and 7. NNARS scales the best in dimension. A* and OS* have the same behaviour, while PRS and SRS share very similar results. A* and OS* start with good sampling rates, which however decrease radically when the dimension increases.

EXP3. Then, we focus on how the efficiency scales with respect to the budget. The distribution tested is: $f(x) \propto \exp(\sin(x))$, with x in $[0,1]$. In Figure 3a, NNARS, A* and OS* give the best performance, reaching the asymptotic regime after 20,000 function evaluations. So NNARS is applicable in a reasonable number of evaluations. Coupled with the study of the evolution of the standard deviations in Figure 3b, we conclude that the results in the transition regime may vary, but the time to the asymptotic region is not initialization-sensitive.

EXP4. Finally, we show the efficiency of NNARS on non-synthetic data from the set in Cortez and Morais (2007). It consists of 517 observations of meteorological data used in order to predict forest fires in the north-eastern part of Portugal. The goal is to enlarge the data set. So we would like to sample artificial data points from a distribution which is close to the one which generated the data set. This target distribution is obtained in a non-parametric way, using the Epanechnikov kernel which creates a non-smooth f. We then apply samplers which do not use the decomposition of f described in Maddison et al. (2014). That is why A* and OS* sampling will not be applied. From the 13 dimensions of the dataset we work with those corresponding to Duff Moisture Code (DMC) and Drought Code (DC) and we get the sampling rates in Table 1. NNARS clearly offers the best performance.


(a) Sampling rate vs. Peakiness [Exp1]

(b) Sampling rate vs. Dimension [Exp2]

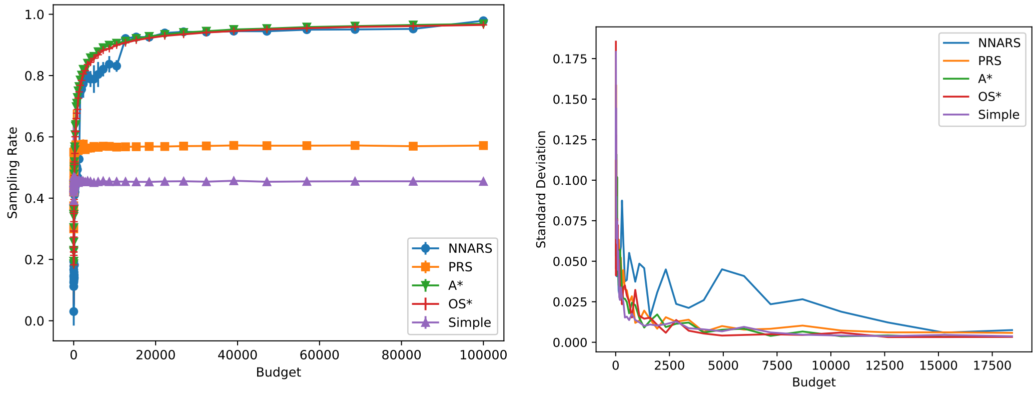
Figure 2: Empirical sampling rates for [Exp1] and [Exp2]

n=10^5, 2D	sampling rate
NNARS	$45.7\% \pm 0.1\%$
PRS	$16.0\% \pm 0.1\%$
SRS	$15.5\% \pm 0.1\%$

Table 1: Sampling rates for forest fires data [Exp4]

6.2. Synthesis on the numerical experiments

The essential features of NNARS have been brought to light in the experiments presented in Figures 2, 3 and using the non-synthetic data from Cortez and Morais (2007). In particular, Figure 3a gives the evidence that the algorithm reaches good sampling rates in a relatively small number of evaluations of the target distribution. Furthermore, Figure 2b illustrates the possibility of applying the algorithm in a multidimensional setting. In Figure 2a, we


(a) Sampling rate vs. Budget [Exp3]

(b) Standard deviation of the sampling rate vs. budget [Exp3]

Figure 3: Empirical sampling rates and their standard deviations for [Exp3]

observe that A* and OS* sampling benefit from the knowledge of the specific decomposition of f needed in Maddison et al. (2014). We highlight the fact that this assumption is not true in general. Besides, A* sampling requires relevant bounding and splitting strategies. We note that tuning NNARS only requires the choice of a few numerical hyperparameters. They might be chosen thanks to generic strategies like grid search. Finally, the application to forest fire data generation illustrates the great potential of NNARS for applications reaching beyond the scope of synthetic experiments.

7. Conclusion

In this work, we introduced an adaptive rejection sampling algorithm, which is a perfect sampler according to f. It offers a rejection rate of order $(\log(n)/n)^{s/d}$, if $s \leq 1$. This rejection rate is near-optimal, in the minimax sense over the class of s-Hölder smooth densities. Indeed, we provide the first lower bound for the adaptive rejection sampling problem, which provides a measure of the difficulty of the problem. Our algorithm matches this bound up to logarithmic terms.

In the experiments, we test our algorithm in the context of synthetic target densities and of a non-synthetic dataset. A first set of experiments shows that the behavior of the sampling rate of our algorithm is similar to that of state of the art methods, as the dimension and the budget increase. Two of the methods used in this set of experiments require the target density to allow a specific decomposition. Therefore, these methods are neglected for the experiment which aims at generating forest fire data. In this experiment, NNARS clearly performs better than its competitors.

The extension of the NNARS algorithm to non lower-bounded densities is still an open question, as well as the development of an optimal adaptive rejection sampler, when the density's derivative is Hölder regular instead. We leave these interesting open questions for future work.

Acknowledgements. The work of A. Carpentier is partially supported by the Deutsche Forschungsgemeinschaft (DFG) Emmy Noether grant MuSyAD (CA 1488/1-1), by the DFG - 314838170, GRK 2297 MathCoRe, by the DFG GRK 2433 DAEDALUS, by the DFG CRC 1294 'Data Assimilation', Project A03, and by the UFA-DFH through the French-German Doktorandenkolleg CDFA 01-18.

References

Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I Jordan. An introduction to mcmc for machine learning. *Machine learning*, 50(1-2):5–43, 2003.

Olivier Canévet, Cijo Jose, and François Fleuret. Importance sampling tree for large-scale empirical expectation. In *Proceedings of the International Conference on Machine Learning (ICML)*, number EPFL-CONF-218848, 2016.

Olivier Cappé, Randal Douc, Arnaud Guillin, Jean-Michel Marin, and Christian P Robert. Adaptive importance sampling in general mixture classes. *Statistics and Computing*, 18(4):447–459, 2008.

Paulo Cortez and Aníbal de Jesus Raimundo Morais. A data mining approach to predict forest fires using meteorological data. 2007.

Bernard Delyon and François Portier. Efficiency of adaptive importance sampling. *arXiv preprint arXiv:1806.00989*, 2018.

Luc Devroye. Sample-based non-uniform random variate generation. In *Proceedings of the 18th conference on Winter simulation*, pages 260–265. ACM, 1986.

Marc Dymetman, Guillaume Bouchard, and Simon Carter. The os* algorithm: a joint approach to exact optimization and sampling. *arXiv preprint arXiv:1207.0742*, 2012.

Akram Erraqabi, Michal Valko, Alexandra Carpentier, and Odalric Maillard. Pliable rejection sampling. In *International Conference on Machine Learning*, pages 2121–2129, 2016.

Michael Evans and Timothy Swartz. Random variable generation using concavity properties of transformed densities. *Journal of Computational and Graphical Statistics*, 7(4):514–528, 1998.

James Allen Fill. An interruptible algorithm for perfect sampling via markov chains. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 688–695. ACM, 1997.

Walter R Gilks. Derivative-free adaptive rejection sampling for gibbs sampling, bayesian statistics 4, 1992.

Walter R Gilks and Pascal Wild. Adaptive rejection sampling for gibbs sampling. *Applied Statistics*, pages 337–348, 1992.

Walter R Gilks, NG Best, and KKC Tan. Adaptive rejection metropolis sampling within gibbs sampling. *Applied Statistics*, pages 455–472, 1995.

Dilan Görür and Yee Whye Teh. Concave-convex adaptive rejection sampling. *Journal of Computational and Graphical Statistics*, 20(3):670–691, 2011.

Wolfgang Hörmann. A rejection technique for sampling from t-concave distributions. *ACM Transactions on Mathematical Software (TOMS)*, 21(2):182–193, 1995.

Chris J Maddison, Daniel Tarlow, and Tom Minka. A* sampling. In *Advances in Neural Information Processing Systems*, pages 3086–3094, 2014.

Luca Martino. Parsimonious adaptive rejection sampling. *Electronics Letters*, 53(16):1115–1117, 2017.

Luca Martino and Francisco Louzada. Adaptive rejection sampling with fixed number of nodes. *Communications in Statistics-Simulation and Computation*, pages 1–11, 2017.

Luca Martino and Joaquín Míguez. A generalization of the adaptive rejection sampling algorithm. *Statistics and Computing*, 21(4):633–647, 2011.

Luca Martino, Jesse Read, and David Luengo. Improved adaptive rejection metropolis sampling algorithms. *arXiv preprint arXiv:1205.5494*, 2012.

Nicholas Metropolis and Stanislaw Ulam. The monte carlo method. *Journal of the American statistical association*, 44(247):335–341, 1949.

Renate Meyer, Bo Cai, and François Perron. Adaptive rejection metropolis sampling using lagrange interpolation polynomials of degree 2. *Computational Statistics & Data Analysis*, 52(7):3408–3423, 2008.

Christian Naesseth, Francisco Ruiz, Scott Linderman, and David Blei. Reparameterization gradients through acceptance-rejection sampling algorithms. In *Artificial Intelligence and Statistics*, pages 489–498, 2017.

Christian A Naesseth, Francisco JR Ruiz, Scott W Linderman, and David M Blei. Rejection sampling variational inference. *arXiv preprint arXiv:1610.05683*, 2016.

Man-Suk Oh and James O Berger. Adaptive importance sampling in monte carlo integration. *Journal of Statistical Computation and Simulation*, 41(3-4):143–168, 1992.

James Propp and David Wilson. Coupling from the past: a user's guide. *Microsurveys in Discrete Probability*, 41:181–192, 1998.

Ernest K Ryu and Stephen P Boyd. Adaptive importance sampling via stochastic convex programming. *arXiv preprint arXiv:1412.4845*, 2014.

John Von Neumann. 13. various techniques used in connection with random digits. *Appl. Math Ser*, 12:36–38, 1951.

Ping Zhang. Nonparametric importance sampling. *Journal of the American Statistical Association*, 91(435):1245–1253, 1996.

Appendix A. Illustration of Rejection Sampling

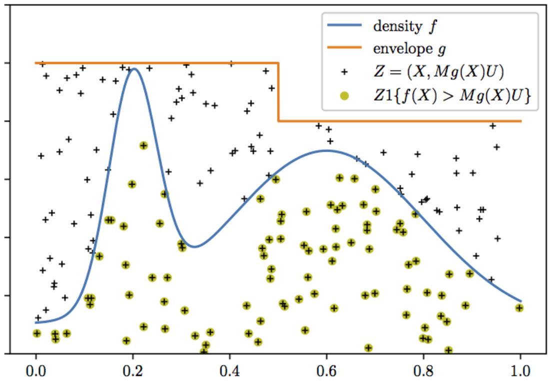


Figure 4: Geometrical interpretation of Rejection Sampling

In the following, we do not assume that f is a density. In fact ARS samplers could be given evaluations of the density multiplied by a positive constant. We prove in the sequel that as long as the resulting function satisfies Assumption 3, the upper bound presented in Theorem 6 holds in this case as well as in the case when f is a density. The lower bound is also proved without the assumption that f is a density.

Appendix B. Proof of Theorem 2

Let us assume that $\forall t \leq n, \forall x \in [0,1]^d, f(x) \leq M_t g_t(x)$. If X_t has been drawn at time t, and E_t denotes the event in which X_t is accepted, and $\tilde{\chi}_j$ denotes the set of the proposal samples drawn at time $j \leq n$ and of their images by f, then $\forall \Omega \subset [0,1]^d$ such that Ω is Lebesgue measureable, it holds:

$$\begin{aligned}
&\mathbb{P}_{X_t \sim g_t, U \sim \mathcal{U}_{[0,1]}}\Big(\{X_t \in \Omega\} \cap E_t \,\Big|\, \bigcup_{j<t} \chi_j\Big) \\
&\qquad\qquad\qquad = \mathbb{P}_{X_t \sim g_t, U \sim \mathcal{U}_{[0,1]}}\Big(X_t \in \Omega ; \frac{f(X_t)}{M_t g_t(X_t)} \geq U_t \,\Big|\, \bigcup_{j<t} \chi_j\Big) \\
&\qquad\qquad\qquad = \int_\Omega \frac{f(x)}{M_t g_t(x)} g_t(x) dx \\
&\qquad\qquad\qquad = \int_\Omega \frac{f(x)}{M_t} dx,
\end{aligned}$$

because U_t is independent from X_t conditionally to $\bigcup_{j<t}\chi_j$.
Hence, since $\mathbb{P}_{X_t\sim g_t, U\sim\mathcal{U}_{[0,1]}}(E_t) = I_f/M_t$, we have:

$$\mathbb{P}_{X_t\sim g_t, U\sim\mathcal{U}_{[0,1]}}\left(X_t\in\Omega \,\middle|\, E_t; \bigcup_{j<t}\chi_j\right) = \int_\Omega \frac{f(x)}{M_t}\left(\frac{M_t}{I_f}\right)dx$$
$$= \int_\Omega \frac{f(x)}{I_f}dx.$$

Thus $X_t|E_t$ is distributed according to f/I_f and is independent from the samples accepted before step t, since $X_t|E_t$ is independent from $\bigcup_{j<t}\chi_j$.

We have proved that the algorithm provides independent samples drawn according to the density f/I_f.

Appendix C. Proof of Theorem 6

C.1. Approximate Nearest Neighbor Estimator

In this subsection, we study the characteristics of the Approximate Nearest Neighbor Estimator. First, we prove a bound on the distance between the image of x by the Approximate Nearest Neighbor Estimator of f and $f(x)$, under the condition that f satisfies Assumption 3. More precisely, we prove that $\hat{f}(x)$ lies within a radius of $\hat{r}_\chi$ away from $f(x)$. Then we prove a high probability bound on the radius $\hat{r}_\chi$ under the same assumptions. This bound only depends on the probability, the number of samples, and constants of the problem. These propositions will be of use in the proof of Theorem 6.

Let $\tilde{N} > 0$, we write $C := C_{\tilde{N}}$ (as in Definition 4) for simplicity.

Proposition 8 *Let f be a positive function satisfying Assumption 3. Consider $\widetilde{N}$ points $\chi = \{(X_1, f(X_1)), \ldots, (X_{\widetilde{N}}, f(X_{\widetilde{N}}))\}$.*
If $\hat{f}_\chi$ is the Approximate Nearest Neighbor Estimate of f, as defined in Definition 4, then

$$\forall x \in [0,1]^d,\ |\hat{f}_\chi(x) - f(x)| \le \hat{r}_\chi,$$

where $\hat{r}_\chi$ is defined in Equation (3) (in Definition 4).

Proof of Proposition 8. We have that $\forall x \in [0,1]^d$,

$$\|x - X_{i(C(x))}\|_\infty \le \|x - C(x)\|_\infty + \|C(x) - X_{i(C(x))}\|_\infty,$$

where the set $\mathcal{C}_{\tilde{N}}$ and the function i are defined in Definition 4.
Now, $\|x - C(x)\|_\infty \le \frac{1}{2\widetilde{N}^{\frac{1}{d}}}$ and $\|C(x) - X_{i(C(x))}\|_\infty \le \max_{u\in\mathcal{C}_{\tilde{N}}}\|u - X_{i(u)}\|_\infty, \forall x \in [0,1]^d$ and where $\mathcal{C}_{\tilde{N}}$ is defined in Definition 4.
Thus $\forall x \in [0,1]^d$,

$$\|x - X_{i(C(x))}\|_\infty \le \frac{1}{2\widetilde{N}^{\frac{1}{d}}} + \max_{u\in\mathcal{C}_{\tilde{N}}}\|u - X_{i(u)}\|_\infty, \tag{4}$$

and from Assumption 3

$$\forall x \in [0,1]^d,\ |\hat{f}_\chi(x) - f(x)| \le \hat{r}_\chi.$$

■

Proposition 9 *Consider the same notations and assumptions as in Proposition 8. Let g be a density on $[0,1]^d$ such that*

$$\exists\, 1 \ge c > 0 \text{ such that } \forall x \in [0,1]^d,\ c < g(x),$$

and assume that the points X_i in χ are sampled in an i.i.d. fashion according to g. Defining $\delta_0 = \frac{1}{\widetilde{N}} \exp(-\widetilde{N})$, it holds for any $\delta > \delta_0$, that with probability larger than $1-\delta$:

$$\hat{r}_\chi \le 2^s r_{\widetilde{N},\delta,c}.$$

where we write $r_{\widetilde{N},\delta,c} = H \left(\frac{\log(\widetilde{N}/\delta)}{c\widetilde{N}} \right)^{\frac{s}{d}}$.

Proof of Proposition 9. Let ϵ be a positive number smaller than 1 such that ϵ^{-d} is an integer. We split $[0,1]^d$ in $\frac{1}{\epsilon^d}$ hypercubes of side-length ϵ and of centers in $\mathcal{C}_{\epsilon^{-d}}$. Let I be one of these hypercubes, we have $\mathbb{P}(X_1 \dots X_{\widetilde{N}} \notin I) \le (1 - c\epsilon^d)^{\widetilde{N}} \le \exp(-c\epsilon^d \widetilde{N})$.
So with probability larger than $1 - \exp(-c\epsilon^d\widetilde{N})$, at least one point has been drawn in I.
Thus $\forall x \in [0,1]^d$, with probability larger than $1 - \exp(-c\epsilon^d\widetilde{N})$, it holds:

$$\|x - X_{i(x)}\|_\infty \le \epsilon,$$

where $i(x) = \arg\min_{i\in\{1,\dots,N\}} (\|x - X_i\|_\infty)$.
Thus $\forall x \in [0,1]^d$, with probability larger than $1-\delta'$,

$$\|x - X_{i(x)}\|_\infty \le \left(\frac{\log(1/\delta')}{c\widetilde{N}} \right)^{\frac{1}{d}},$$

where $\delta' = \exp(-c\epsilon^d\widetilde{N})$ (observe $\delta' > \exp(-\widetilde{N})$).
Thus with probability larger than $1 - \frac{1}{\epsilon^d}\delta'$, it holds

$$\forall x \in [0,1]^d,\ \|x - X_{i(x)}\|_\infty \le \left(\frac{\log(1/\delta')}{c\widetilde{N}} \right)^{\frac{1}{d}}.$$

With probability larger than $1 - c\widetilde{N}\delta' > 1 - \frac{c\widetilde{N}}{\log(1/\delta')}\delta'$, it holds

$$\forall x \in [0,1]^d,\ \|x - X_{i(x)}\|_\infty \le \left(\frac{\log(1/\delta')}{c\widetilde{N}} \right)^{\frac{1}{d}}.$$

Hence, by letting $\delta = (c\widetilde{N})\delta'$, with probability larger than $1-\delta$,

$$\begin{aligned}\forall x \in [0,1]^d,\ \|x - X_{i(x)}\|_\infty &\leq \left(\frac{\log(c\widetilde{N}/\delta)}{c\widetilde{N}}\right)^{\frac{1}{d}}\\ &\leq \left(\frac{\log(\widetilde{N}/\delta)}{c\widetilde{N}}\right)^{\frac{1}{d}}.\end{aligned}$$

Thus $\forall \delta > c\widetilde{N}\exp(-\widetilde{N})$, with probability larger than $1-\delta$,

$$\forall x \in [0,1]^d,\ \|x - X_{i(x)}\|_\infty \leq \left(\frac{\log(\widetilde{N}/\delta)}{c\widetilde{N}}\right)^{\frac{1}{d}},$$

and in particular, with probability larger than $1-\delta$,

$$\max_{u \in \mathcal{C}_{\tilde{N}}} \|u - X_{i(u)}\|_\infty \leq \left(\frac{\log(\widetilde{N}/\delta)}{c\widetilde{N}}\right)^{\frac{1}{d}}.$$

Furthermore we have since $|\chi_{\tilde{N}}| = \tilde{N}$

$$\frac{1}{2\widetilde{N}^{\frac{1}{d}}} \leq \max_{x \in [0,1]^d} \|x - X_i(x)\|_\infty.$$

So we also have (since $c \leq 1$ and $\log(1/\delta) \geq 1$)

$$\frac{1}{2\widetilde{N}^{\frac{1}{d}}} \leq \left(\frac{\log(\widetilde{N}/\delta)}{c\widetilde{N}}\right)^{\frac{1}{d}}.$$

Finally, from Equation (4), with probability larger than $1-\delta$, $\forall x \in [0,1]^d$,

$$\begin{aligned}\|x - X_{i(C(x))}\|_\infty &\leq \frac{1}{2\widetilde{N}^{\frac{1}{d}}} + \max_{u \in \mathcal{C}_{\tilde{N}}} \|u - X_{i(u)}\|_\infty\\ &\leq 2\left(\frac{\log(\widetilde{N}/\delta)}{c\widetilde{N}}\right)^{\frac{1}{d}},\end{aligned}$$

and with probability larger than $1-\delta$,

$$\begin{aligned}\hat{r}_\chi &= H\left(\frac{1}{2\widetilde{N}^{\frac{1}{d}}} + \max_{u \in \mathcal{C}_{\tilde{N}}} \|u - X_{i(u)}\|_\infty\right)^s\\ &\leq H\left(2\left(\frac{\log(\widetilde{N}/\delta)}{c\widetilde{N}}\right)^{\frac{1}{d}}\right)^s = 2^s r_{\widetilde{N},\delta,c}.\end{aligned}$$

■

C.2. Proof of Theorem 6

Let us define the number of samples which are known to be independent and sampled according to f based on Theorem 2: $\hat{n} = \#\mathcal{S}\mathbf{1}\{\forall t \leq n : \ f \leq M_t g_t\}$. Then the loss of a learner can be written as $L_n = n - \hat{n}$. This is justified by considering two complementary events. In the first, the rejection sampling procedure is correct at all steps, that is to say all proposed envelopes bound f from above; and in the second, there exists a step where the procedure is not correct. In the first case, the sampler will output i.i.d. samples drawn from the density f. So the loss of the learner L_n is the number of samples rejected by the sampler. In the second case, the sampler is not trusted to produce correct samples. So the loss becomes n.

In this subsection, we prove Theorem 6 by first proving a high probability bound on $n - \hat{n}$. We prove this high probability bound thanks to Proposition 10, and Lemma 12. Proposition 10 claims that the algorithm provides independent samples drawn according to f/I_f, under Assumption 3. The proof of Proposition 10 uses Lemma 11 which states that $\forall x \in [0,1]^d, \ f(x) \leq M_k g_k(x)$, under the relevant assumptions. We define two events: one on which every proposal envelope until time $k+1$ is bounded from below by $\frac{6}{10}c_f$: $\mathcal{W}_{k+1}$, and the other one on which every confidence radius $\hat{r}_{\cup_{i\leq k}\chi_k}$ until time k is upper bounded by a quantity $r_{\widetilde{N},\delta,6c_f/10}$: $\mathcal{A}_{k,\delta}$, where δ is a confidence term (designed to be used in the high probability bound on $n-\hat{n}$). Lemma 12 states that the probability of the event $\mathcal{W}_{k+1}$ conditional to the event $\mathcal{A}_{k,\delta}$ is equal to 1, and that that the probability of the event $\mathcal{A}_{k,\delta}$ is larger than $1 - 2k\widetilde{\delta}$ when $\delta = N/nK$. The proof of Theorem 6 uses the fact that the number of rejected samples at step k on $\mathcal{A}_{k,\delta}$ is a sum of Bernoulli variables of parameter smaller than a known quantity that depends on $\widehat{\delta}$, and by applying the Bernstein inequality on this sum. The proof is then concluded by summing on k.

In this subsection, we write :

$$\hat{f}_k = \hat{f}_{\cup_{i\leq k}\chi_i}, \quad \text{and} \quad \hat{r}_k = \hat{r}_{\cup_{i\leq k}\chi_i},$$

to ensure the simplicity of notations. We also write:

$$r_{\widetilde{N},\delta} := r_{\widetilde{N},\delta,6c_f/10} = H\left(\frac{10\log(\widetilde{N}/\delta)}{6c_f\widetilde{N}}\right)^{\frac{s}{d}}.$$

Let us also define the events:

$$\begin{cases} \mathcal{W}_k = \{\forall j \leq k, \ \forall x \in [0,1]^d, \ g_j(x) > \frac{6}{10}c_f\}, \\ \mathcal{A}_{k,\delta} = \{\forall j \leq k, \ \hat{r}_j \leq 2^s r_{N_j,\delta}\}. \end{cases}$$

Proposition 10 *If Assumption 3 holds, the algorithm provides independent samples drawn according to the density f/I_f.*

Lemma 11 *Consider any $k \leq K$. Under the assumptions made in Proposition 10,*

$$\forall x \in [0,1]^d, \ f(x) \leq M_k g_k(x).$$

Proof of Lemma 11. g_1 is the uniform density and M_1 is taken as an upper bound on f. So $\forall x \in [0,1]^d$:

$$M_1 g_1(x) \geq f(x).$$

Let $k \in \{2, \dots, K\}$. From Proposition 8:

$$\forall x \in [0,1]^d,\ |\hat{f}_{k-1}(x) - f(x)| \leq \hat{r}_{k-1}.$$

Thus, $\forall x \in [0,1]^d$:

$$g_k(x) = \frac{\hat{f}_{k-1}(x) + \hat{r}_{k-1}}{I_{\hat{f}_{k-1}} + \hat{r}_{k-1}} \geq \frac{f(x)}{I_{\hat{f}_{k}-1} + \hat{r}_{k-1}} \geq \frac{f(x)}{M_k}.$$

Hence:

$$\forall x,\ M_k g_k(x) \geq f(x).$$

■

Proof of Proposition 10. We have that $\forall j \leq k$, $\forall x \in [0,1]^d$, $f(x) \leq M_k g_k(x)$. Theorem 2 proves that the algorithm provides independent samples drawn according to the density f/I_f. ■

Lemma 12 *Let $\tilde{\delta} = N/(nK)$. If Assumption 3 and 5 hold for n, then*

$$\begin{cases} \mathbb{P}(\mathcal{W}_1) = 1, \quad \mathbb{P}(\mathcal{W}_{k+1} | \mathcal{A}_{k,\widetilde{\delta}}) = 1, \\ \mathbb{P}(\mathcal{A}_{k,\widetilde{\delta}}) \geq 1 - 2k\widetilde{\delta}. \end{cases}$$

Proof of Lemma 12. Since $g_1(x) = 1$, $s \leq 1, c_f \leq 1$, the event $\mathcal{W}_1 = \{\forall x \in [0,1]^d,\ g_1(x) > \frac{6}{10} c_f\}$ has probability 1. Also by Proposition 8 and Proposition 9, the event $\mathcal{A}_{1,\widetilde{\delta}}$ has probability larger than $1 - \widetilde{\delta}$.
Consider now that the event $\mathcal{A}_{k,\widetilde{\delta}}$ holds for a given $k \leq K$. Then by Proposition 8 and Proposition 9, it holds that for all $j \leq k$ and for all $x \in [0,1]^d$

$$|\hat{f}_j(x) - f(x)| \leq 2^s r_{N_j,\widetilde{\delta}}.$$

This implies that

$$\begin{aligned} g_{j+1}(x) &\geq \frac{f(x)}{M_{j+1}} \geq \frac{f(x)}{I_f + 2^{s+1} r_{N_j,\widetilde{\delta}}} \\ &\geq \frac{f(x)/I_f}{1 + 2^{s+1} r_{N_j,\widetilde{\delta}}/I_f} \geq \frac{f(x)}{I_f}\left(1 - 2^{s+1}\frac{r_{N_j,\widetilde{\delta}}}{I_f}\right) \\ &\geq c_f\left(1 - 2^{s+1}\frac{r_{N_j,\widetilde{\delta}}}{I_f}\right). \end{aligned}$$

Hence,

$$g_{j+1}(x) \geq c_f\left(1 - \frac{2^{s+1} r_{N,\widetilde{\delta}}}{c_f}\right) \geq \frac{6}{10} c_f,$$

where we have used $r_{N_j,\widetilde{\delta}} \leq r_{N,\widetilde{\delta}} \leq c_f/10$ (see Assumption 5). So $\mathbb{P}(\mathcal{W}_{k+1}|\mathcal{A}_{k,\widetilde{\delta}}) = 1$ and we have proved the first part of the lemma.

Moreover, conditional to $\mathcal{A}_{k,\widetilde{\delta}}$ we have that $g_{k+1}(x) \geq \frac{6}{10} c_f$. Then we apply Proposition 8 and Proposition 9. With probability larger than $1 - \tilde{\delta}$ on χ_k only, and conditional to $\mathcal{A}_{k,\widetilde{\delta}}$, it holds that for all $x \in [0,1]^d$:

$$|\hat{f}_{k+1}(x) - f(x)| \leq 2^s r_{N_{k+1},\tilde{\delta}},$$

where we use that $\hat{r}_{k+1} \leq \hat{r}_{\chi_{k+1}}$. This implies that $\mathbb{P}(\mathcal{A}_{k+1,\widetilde{\delta}}|\mathcal{A}_{k,\widetilde{\delta}}) \geq 1 - \tilde{\delta}$, and so for any $k \leq K$

$$\mathbb{P}(\mathcal{A}_{k,\widetilde{\delta}}) \geq (1 - \tilde{\delta})^k.$$

This concludes the proof since $(1 - \widetilde{\delta})^k \geq 1 - 2k\tilde{\delta}$ for $\tilde{\delta} \leq 1/(2K)$. ■

Proof of Theorem 6. Let $\tilde{\delta} = N/(nK)$ and $\delta = K\tilde{\delta}$ and let $\hat{n}_k$ denote the number of accepted samples at round k.
From Lemma 11, we know that $\forall k \leq K$,

$$\forall x,\ M_k g_k(x) \geq f(x).$$

Hence, the samples accepted at step $k+1$ are independently sampled according to f/I_f, and $N_{k+1} - \hat{n}_k$, the number of rejected samples, is a sum of Bernoulli variables of parameter $1 - I_f/M_k$.
On $\mathcal{A}_{k,\widetilde{\delta}} \cap \mathcal{W}_k$,

$$\begin{aligned}\frac{I_f}{M_{k+1}} &\geq \frac{I_f}{I_f + 2^{s+1} r_{N_k,\widetilde{\delta}}}\\ &\geq 1 - \frac{2^{s+1} r_{N_k,\widetilde{\delta}}}{I_f}.\end{aligned}$$

Thus, $1 - \frac{I_f}{M_{k+1}} \leq \frac{2^{s+1} r_{N_k,\widetilde{\delta}}}{I_f}$.
On $\mathcal{A}_{K,\widetilde{\delta}} \cap \mathcal{W}_K$, according to the Bernstein inequality, $\forall k \leq K$ the event

$$\mathcal{V}_k = \left\{ N_{k+1} - \hat{n}_k - \left(1 - \frac{I_f}{M_{k+1}}\right) N_{k+1} \leq \sqrt{2N_{k+1}\left(1 - \frac{I_f}{M_{k+1}}\right)\log\left(\frac{1}{\widetilde{\delta}}\right)} + \log\left(\frac{1}{\widetilde{\delta}}\right) \right\}$$

has probability larger than $1 - \widetilde{\delta}$.
Hence on $\mathcal{A}_{K,\widetilde{\delta}} \cap \mathcal{W}_K$, $\bigcap_{k\in\{1,\dots,K\}} \mathcal{V}_k$ has probability $1 - K\widetilde{\delta}$.
Consequently, since $\mathcal{A}_{K,\widetilde{\delta}} \cap \mathcal{W}_K$ has probability larger than $1 - K\widetilde{\delta}$ according to Lemma 11,

$\bigcap_{k\in\{1,\dots,K\}} \mathcal{V}_k \cap \mathcal{A}_{K,\widetilde{\delta}} \cap \mathcal{W}_K$ has probability larger than $1-2K\widetilde{\delta}$.
On $\mathcal{V}_k \cap \mathcal{A}_{K,\widetilde{\delta}} \cap \mathcal{W}_K$,

$$N_{k+1} - \hat{n}_k - \frac{2^{s+1} r_{N_k,\widetilde{\delta}}}{I_f} N_{k+1} \leq \sqrt{2N_{k+1}\frac{2^s r_{N_k,\widetilde{\delta}}}{I_f}\log\left(\frac{1}{\widetilde{\delta}}\right)} + \log\left(\frac{1}{\widetilde{\delta}}\right)$$

(and we know from Proposition 10, that on $\mathcal{V}_k \cap \mathcal{A}_{K,\widetilde{\delta}} \cap \mathcal{W}_K$, we also have that the drawn samples are independently drawn according to f/I_f).
Hence on $\bigcap_k \mathcal{V}_k \cap \mathcal{A}_{K,\widetilde{\delta}} \cap \mathcal{W}_K$, which has probability larger than $1-2K\widetilde{\delta} := 1-2\delta$:

$$\sum_1^{K-1}\left(N_{k+1} - \hat{n}_k - \frac{2^{s+1} r_{N_k,\widetilde{\delta}}}{I_f} N_{k+1}\right) \leq \sum_1^{K-1}\left(\sqrt{2N_{k+1}\frac{2^{s+1} r_{N_k,\widetilde{\delta}}}{I_f}\log(\frac{1}{\widetilde{\delta}})}\right) + K\log\left(\frac{1}{\widetilde{\delta}}\right),$$

i.e:

$$n - \hat{n} \leq \underbrace{\frac{2^{s+1}}{I_f}\sum_1^{K-1}\left(r_{N_k,\widetilde{\delta}} N_{k+1}\right) + 4\sqrt{\frac{\log(\frac{1}{\widetilde{\delta}})}{I_f}}\sum_1^{K-1}\left(\sqrt{N_{k+1} r_{N_k,\widetilde{\delta}}}\right)}_{(1)} + \underbrace{K\log\left(\frac{1}{\widetilde{\delta}}\right)}_{(2)} + N.$$

Hence,

$$\begin{aligned}(2) &= K\log\left(\frac{K}{\delta}\right)\\ &= \log_2\left(\frac{n}{N}\right)\log\left(\frac{\log_2(n/N)}{\delta}\right),\end{aligned}$$

and if $\beta = \frac{2^{s+1}}{I_f} + 4\sqrt{\frac{\log(\frac{1}{\widetilde{\delta}})}{I_f}}$, and $\tilde{C} = H(10/(6c_f))^{s/d}$,

$$\begin{aligned}(1) &\leq \beta\Big[\sum_1^{K-1}\left(r_{N_k,\widetilde{\delta}} N_{k+1}\right) + K\Big]\\ &\leq \beta\sum_1^{K-1}\left(\widetilde{C}\log\left(\frac{2^k N}{\widetilde{\delta}}\right)^{s/d}(2^k N)^{1-s/d}\right) + K\beta.\end{aligned}$$

Now, assume $s/d < 1$, then

$$\begin{aligned}(1) &\leq \beta\widetilde{C}\left(\log\left(\frac{n}{\widetilde{\delta}}\right)\right)^{s/d} N^{1-s/d}\left(\frac{2^{(1-s/d)K}-1}{2^{(1-s/d)}-1}\right) + K\beta\\ &\leq \frac{\beta\widetilde{C}}{2^{(1-s/d)}-1}\left(\log\left(\frac{n}{\widetilde{\delta}}\right)\right)^{s/d} n^{1-s/d} + K\beta.\end{aligned}$$

And if $s = d = 1$, then

$$(1) \leq K\beta\widetilde{C}\left(\log\left(\frac{n}{\widetilde{\delta}}\right)\right) + K\beta.$$

We have proved that if the assumptions of Theorem 6 are satisfied, with probability $1-2\delta$,

$$n-\hat{n} \leq \left(\frac{2^{s+1}}{I_f}+4\sqrt{\frac{\log(\frac{1}{\tilde{\delta}})}{I_f}}\right) C_{s,d}\left(\log\left(\frac{n}{\tilde{\delta}}\right)\right)^{s/d} n^{1-s/d}$$

$$+ \quad \log_2\left(\frac{n}{N}\right)\log\left(\frac{\log_2(n/N)}{\delta}\right)+N+\left(\frac{2^{s+1}}{I_f}+4\sqrt{\frac{\log(\frac{1}{\tilde{\delta}})}{I_f}}\right)\log_2\left(\frac{n}{N}\right),$$

where $C_{s,d}$ is a constant dependent on s and d. Finally, the proof is finished following a few strings of inequalities and taking the expected value. The following reminders may help:
$d \geq 1$; $s \leq 1$; $c_f \leq 1$.
$\tilde{C} = H(10/(6c_f))^{s/d}$; $\delta = N/n = K\tilde{\delta}$; $K = \lceil \log_2(n/N) \rceil$ and $N = \lceil 2(10H)^{d/s}\log(n)c_f^{-1-d/s} \rceil$.
In particular, we have: $I_f \geq c_f$ and $1/\tilde{\delta} \leq 5n^2$.

■

Appendix D. Proof of Theorem 7.

D.1. Setting

Let us introduce two different settings:

Setting 13 (Class of Rejection Samplers with Access to Multiple Evaluations of the density (RSAME))
A sampler belongs to the RSAME class if it follows the following steps:

- *For each step $t \in \{1 \dots n\}$:*
  Choose a distribution $\mathcal{D}_t$ on $\mathbb{R}$, depending on $\big((Y_1, f(Y_1)) \dots (Y_{t-1}, f(Y_{t-1}))\big)$. Draw Y_t according to $\mathcal{D}_t$.
- *Choose a density g and a positive constant M depending on $\big((Y_1, f(Y_1)) \dots (Y_n, f(Y_n))\big)$, and sample Z by performing one Rejection Sampling Step(f, M, g).*

Objective : *The objective of a RSAME sampler is to sample one point according to a normalized version of f.*
Loss : *The loss of a RSAME sampler is defined as follows :*

$$L'_n = n(1-\mathbf{1}\{Z \text{ is accepted }\}\mathbf{1}\{f \leq Mg\}).$$

Strategy : *A strategy $\mathfrak{s}'$ consists of the choice of $\mathcal{D}_t$ depending on $\big((Y_1, f(Y_1)) \dots (Y_{t-1}, f(Y_{t-1}))\big)$, and of the choice of M, g depending on $\big((Y_1, f(Y_1)) \dots (Y_n, f(Y_n))\big)$. Denote $\mathfrak{S}'$ the set of strategies for this setting.*

Setting 14 (Class of Adaptive Rejection Samplers (ARS))
A sampler belongs to the ARS class if, at each step $t \in \{1 \dots n\}$: it

- *Chooses a density g_t, and a positive constant M_t, depending only on $\Big\{(X_1, f(X_1)), \ldots, (X_{t-1}, f(X_{t-1}))\Big\}$.*

- *Samples X_t by performing rejection sampling on the target function f using M_t and g_t as the rejection constant and the proposal. Store X_t in $\mathcal{S}$ if it is accepted.*

Objective : *The objective of an ARS sampler is to sample i.i.d. points according to a normalized version of f*
Loss : *The loss of an ARS sampler is defined as follows : $L_n = n - \#\mathcal{S}\mathbf{1}\{\forall t \leq n, f \leq M_t g_t\}$.*
Strategy : *A strategy $\mathfrak{s}$ consists of the choice of M_t, g_t depending on $\big((X_1, f(X_1)) \ldots (X_{t-1}, f(X_{t-1}))\big)$. Denote $\mathfrak{S}$ the set of strategies for this setting.*

For the class of samplers defined in Setting 14 (and similarly for Setting 13) we call value of the class the quantity $\inf_{\mathfrak{s}\in\mathfrak{S}} \sup_{f\in\mathcal{F}_0} \mathbb{E}^{(\mathfrak{s},f)}(L_n)$, where the symbol $\mathbb{E}^{(\mathfrak{s},f)}$ denotes the expectation with respect to all relevant random variables, when those are generated by a sampler of the relevant class, using function f and strategy $\mathfrak{s}$; and $\mathcal{F}_0$ denotes the set of functions satisfying Assumption 3.

D.2. Setting Comparison

Lemma 15 *The value of the class defined in Setting 13 is smaller than the value of the class defined in Setting 14:*

$$\inf_{\mathfrak{s}'\in\mathfrak{S}'} \sup_{f\in\mathcal{F}_0} \mathbb{E}^{(\mathfrak{s}',f)}(L'_n) \leq \inf_{\mathfrak{s}\in\mathfrak{S}} \sup_{f\in\mathcal{F}_0} \mathbb{E}^{(\mathfrak{s},f)}(L_n).$$

In other terms, Setting 13 is easier than Setting 14

Proof of Lemma 15. For any given strategy $\mathfrak{s}$ designed for Setting 14 that chooses (g_i, M_i) to generate X_i, consider the associated strategies $\mathfrak{s}'_1, \ldots, \mathfrak{s}'_n$ for Setting 13 consisting of:

1. Generating $Y_1, \ldots, Y_{n-1}$ from the same probability distributions as $X_1, \ldots, X_{n-1}$ generated for Setting 14 using strategy $\mathfrak{s}$; this is a valid choice since the distribution $\mathcal{D}_t$ of X_t only depends on $\big((X_1, f(X_1), \ldots, (X_{t-1}, f(X_{t-1}))\big)$.

2. Using (g_i, M_i), given by step i of strategy $\mathfrak{s}$ applied to $\big((Y_1, f(Y_1), \ldots, (Y_{i-1}, f(Y_{i-1}))\big)$, in order to sample Z by rejection sampling. It is still a valid choice, which actually discards the information of $\big((Y_i, f(Y_i), \ldots, (Y_{n-1}, f(Y_{n-1}))\big)$.

Then, we have for any $f \in \mathcal{F}_0$:

$$\begin{aligned}
\mathbb{E}^{(\mathfrak{s},f)}(L_n) &= n - \mathbb{E}^{(\mathfrak{s},f)}(\#\mathcal{S}\mathbf{1}\{\forall t \leq n, f \leq M_t g_t\}) \\
&= n - \mathbb{E}^{(\mathfrak{s},f)}\Big(\sum_{i=1}^{n} \mathbf{1}\{X_i \text{ is accepted }\}\mathbf{1}\{\forall t \leq n, f \leq M_t g_t\}\Big) \\
&\geq n - \mathbb{E}^{(\mathfrak{s},f)}\Big(\sum_{i=1}^{n} \mathbf{1}\{X_i \text{ is accepted }\}\mathbf{1}\{f \leq M_i g_i\}\Big) \\
&= \sum_{i=1}^{n} \mathbb{E}^{(\mathfrak{s}_i,f)}\Big(1 - \mathbf{1}\{X_i \text{ is accepted }\}\mathbf{1}\{f \leq M_i g_i\}\Big) \\
&= \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}^{(\mathfrak{s}_i,f)}(L'_n).
\end{aligned}$$

Hence, there exists at least one strategy amongst $\mathfrak{s}'_1, \dots, \mathfrak{s}'_n$ that reaches an expected loss in Setting 13 lower than that of strategy $\mathfrak{s}$ in Setting 14.

■

D.3. Lower Bound for Setting 13

Lemma 16

$$\inf_{\mathfrak{s}' \in \mathfrak{S}'} \sup_{f \in \mathcal{F}_0} \mathbb{E}_f(L'_n(\mathfrak{s}')) \geq 3^{-1}2^{-1-3s-2d}5^{-s/d}n^{1-s/d},$$

for n large enough.

The Theorem is a direct consequence of Lemmas 15 and 16. We use Lemma 17 to prove Lemma 16.

Lemma 17 *Let*

$$\begin{aligned}
\mathcal{F}'_1 = \Bigg\{ & f \text{ s.t. } \forall u = (k_1, \dots k_d) \in \{0, 1, \dots, a_{n,d} - 1\}^d, \\
& \text{either } \forall x \in H_u := \Big[\frac{k_1}{a_{n,d}}, \frac{k_1+1}{a_{n,d}}\Big] \dots \Big[\frac{k_d}{a_{n,d}}, \frac{k_d+1}{a_{n,d}}\Big], \\
& f(x) = \phi^+\Big(x - \frac{u}{a_{n,d}}\Big), \\
& \text{or } \forall x \in H_u, f(x) = \phi^-\Big(x - \frac{u}{a_{n,d}}\Big)\Bigg\},
\end{aligned} \tag{5}$$

where:

$$\begin{aligned}
& a_{n,d} = \min\{2p \in \mathbb{N}; 2p \geq (4n)^{\frac{1}{d}}\}, \\
& \phi^+(x) = 1 + (2a_{n,d})^{-s} - \Big\|x - \frac{1}{2a_{n,d}}\mathbf{I}\Big\|_\infty^s, \\
& (\textit{with } \mathbf{I} \textit{ denoting the unit vector}), \\
& \phi^-(x) = 2 - \phi^+(x).
\end{aligned}$$

Then any function in $\mathcal{F}_1'$ *is s-Hölder-smooth.*

Remark 18 *If* $d = 1$,

$$\mathcal{F}_1' = \Bigg\{ f \ s.t. \ \forall i \in \{0, 1, \dots, 4n-1\}, \\ \textit{either } \forall x \in H_i := \Big[\frac{i}{4n}, \frac{i+1}{4n}\Big], \ f(x) = \phi^+\Big(x - \frac{i}{4n}\Big), \\ \textit{or } \forall x \in H_i, f(x) = \phi^-\Big(x - \frac{i}{4n}\Big) \Bigg\}, \tag{6}$$

with $\forall x \in [0, 1/(4n)]$

$$\phi^+(x) = 1 + (8n)^{-s} - \Big|x - \frac{1}{8n}\Big|^s,$$

and

$$\phi^-(x) = 1 - (8n)^{-s} + \Big|x - \frac{1}{8n}\Big|^s.$$

Proof of Lemma 17. Let us first prove that ϕ^+ is s-smooth. $|\phi^+(x) - \phi^+(y)| = |\|y - \frac{1}{2a_{n,d}}\mathbf{I}\|_\infty^s - \|x - \frac{1}{2a_{n,d}}\mathbf{I}\|_\infty^s| \leq \|x - y\|_\infty^s$.
It is straightforward to see that ϕ^- is also s-smooth and that all $f \in \mathcal{F}_1'$ are also s-smooth. ■

Proof of Lemma 16. Let us consider Setting 13 on a subset of functions of $\mathcal{F}_0$. Let

$$\mathcal{F}_1 = \mathcal{F}_{int} \cap \mathcal{F}_1',$$

where $\mathcal{F}_1'$ is defined in Equation (5) and

$$\mathcal{F}_{int} = \left\{ f, \int_0^1 f = 1 \right\}.$$

And $\mathcal{F}_1$ is not empty since $a_{n,d}$ defined in equation (5) is even. Since $\mathcal{F}_1 \subset \mathcal{F}_0$ by application of Lemma 17,

$$\inf_{\mathfrak{s}' \in \mathfrak{S}'} \sup_{f \in \mathcal{F}_0} \mathbb{E}_f(L_n'(\mathfrak{s}')) \geq \inf_{\mathfrak{s}' \in \mathfrak{S}'} \sup_{f \in \mathcal{F}_1} \mathbb{E}_f(L_n'(\mathfrak{s}')).$$

We first note that

$$\inf_{\mathfrak{s}' \in \mathfrak{S}'} \sup_{f \in \mathcal{F}_0} \mathbb{E}(L_n'(\mathfrak{s}')) \geq \inf_{\mathfrak{s}' \in \mathfrak{S}'} \mathbb{E}_{f \sim \mathcal{D}_{\mathcal{F}_1}}(L_n'(\mathfrak{s}')),$$

where $\mathcal{D}_{\mathcal{F}_1}$ is the distribution such that for any F, $\mathbb{P}_{f \sim \mathcal{D}_{\mathcal{F}_1}}(f = F) = \frac{\mathbf{1}\{F \in \mathcal{F}_1\}}{\#\mathcal{F}_1}$. A hypercube will refer to a H_u as defined in Equation (5).

We also note that the choice of M, g where M is a multiplicative constant and g is a density is equivalent to the choice of a positive function G, where $G = Mg$, or $M = I_G$ and $g = \frac{G}{I_G}$.
Furthermore a strategy $\mathfrak{s}'$ for this setting is the combination of three strategies:

1. $\mathfrak{s}'_1$: The strategy to choose $Y_1 \dots Y_n$,
2. $\mathfrak{s}'_2$: The strategy to choose G.

For the first step, let us fix a strategy $\mathfrak{s}'_1$. Let f_1 be a realization of $\mathcal{D}_{\mathcal{F}_1}$. Then by application of strategy $\mathfrak{s}'_1$, $Y_1, \dots, Y_n$ are drawn. Then the evaluations $f_1(Y_1), \dots, f_1(Y_n)$ are obtained. Now let $u_1, \dots u_n$ be the indices such that $H_{u_1} \dots H_{u_n}$ are the hypercubes where $Y_1 \in H_{u_1}, \dots, Y_n \in H_{u_n}$.

Let us define the restricted set $\mathcal{F}_{1|f_1} = \{f \in \mathcal{F}_1 \text{ and } f(Y_1) = f_1(Y_1), \dots, f(Y_n) = f_1(Y_n)\}$. And we consider the distribution $\mathcal{D}_{\mathcal{F}_{1|f_1}}$ such that $\mathbb{P}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}(f = F) = \frac{\mathbf{1}\{F\in\mathcal{F}_1\}}{\#\mathcal{F}_1}$. In a second step, let us fix a strategy $\mathfrak{s}'_2$. This defines a distribution $\mathcal{D}_G$ corresponding to the choice of G. By the law of total expectation, we have:

$$\begin{aligned}\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_1}}(L'_n(\mathfrak{s}')) &= \mathbb{E}_{f_1\sim\mathcal{D}_{\mathcal{F}_1}}\mathbb{E}_{G\sim\mathcal{D}_G}\Big[\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\Big(L'_n(\mathfrak{s}')|(Y_1, f_1(Y_1)), \dots (Y_n, f_1(Y_n)), G\Big) \\ &\qquad |(Y_1, f_1(Y_1)), \dots (Y_n, f_1(Y_n))\Big] \\ &= \mathbb{E}_{f_1\sim\mathcal{D}_{\mathcal{F}_1}}\mathbb{E}_{G\sim\mathcal{D}_G}\Big[\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\Big(L'_n(\mathfrak{s}')|(Y_1, f(Y_1)), \dots (Y_n, f(Y_n)), G\Big) \\ &\qquad |(Y_1, f_1(Y_1)), \dots (Y_n, f_1(Y_n))\Big].\end{aligned}$$

We can write:

$$\begin{aligned}&\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\left(L'_n\left(\mathfrak{s}'\right)|(Y_1, f(Y_1)), \dots (Y_n, f(Y_n)), G\right) \\ &= \mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\Bigg[\mathbf{1}\left\{\exists u \notin \{u_1 \dots u_n\},\ \exists x \in H_u : G(x) < f(x)\right\} n \\ &\quad + \mathbf{1}\left\{\forall u \notin \{u_1 \dots u_n\},\ \forall x \in H_u : G(x) \geq f(x)\right\} n \left(1 - \frac{1}{1 + \|f - G\|_1}\right) \\ &\qquad \Bigg| \ (Y_1, f(Y_1)) \dots (Y_n, f(Y_n)), G\Bigg] \\ &\geq \mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\left(1 - \frac{1}{1 + \|f - G\|_1}\Big| G\right) n.\end{aligned}$$

Now, since for any $x \geq 0$,

$$1 - \frac{1}{1+x} \geq \frac{1}{2}(1 \wedge x),$$

we have

$$\begin{aligned}
&\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\left(L'_n\left(\mathfrak{s}'\right)|(Y_1,f(Y_1)),\dots(Y_n,f(Y_n)),G\right)\\
&\qquad\geq \frac{1}{2}\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\left(\|f-G\|_1\wedge 1\middle|G\right)n\\
&\qquad\geq \frac{1}{2}\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\left(\left\||f-G|\wedge 1\right\|_1\middle|G\right)n\\
&\qquad\geq \frac{1}{2}\left\|\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\left(|f-G|\wedge 1\middle|G\right)\right\|_1 n\\
&\qquad\geq \frac{1}{2}\left\|\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\left([|f-G|\wedge 1](1-\mathbf{1}\{\cup_{i=1}^n H_{u_i}\})\middle|G\right)\right\|_1 n.
\end{aligned}$$

For any $u\neq u_1\dots u_n$, $\forall x\in H_u$, since any realization from $\mathcal{D}_{\mathcal{F}_{1|f_1}}$ is in $\mathcal{F}_{int}$ almost surely,

$$\begin{aligned}
\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\left(|f(x)-G(x)|\wedge 1\right)\geq \frac{1}{3}\Bigg[&\left(\left|\phi^+\left(x-\frac{u}{a_{n,d}}\right)-G(x)\right|\wedge 1\right)\\
&+\left(\left|\phi^-\left(x-\frac{u}{a_{n,d}}\right)-G(x)\right|\wedge 1\right)\Bigg]
\end{aligned}$$

And, for any $u\notin\{u_1,\dots,u_n\}$, and for any $x\in H_u$:

$$\begin{aligned}
&\left(\left|\phi^+\left(x-\frac{u}{a_{n,d}}\right)-G(x)\right|\wedge 1\right)+\left(\left|\phi^-\left(x-\frac{u}{a_{n,d}}\right)-G(x)\right|\wedge 1\right)\\
&\qquad\geq\left(\left|\phi^+\left(x-\frac{u}{a_{n,d}}\right)-G(x)\right|\wedge 1\right)\vee\left(\left|\phi^-\left(x-\frac{u}{a_{n,d}}\right)-G(x)\right|\wedge 1\right)\\
&\qquad\geq\min_\theta\left[\left(\left|\phi^+\left(x-\frac{u}{a_{n,d}}\right)-\theta\right|\wedge 1\right)\vee\left(\left|\phi^-\left(x-\frac{u}{a_{n,d}}\right)-\theta\right|\wedge 1\right)\right]\\
&\qquad=\left(\left|\phi^+\left(x-\frac{u}{a_{n,d}}\right)-1\right|\wedge 1\right)\vee\left(\left|\phi^-\left(x-\frac{u}{a_{n,d}}\right)-1\right|\wedge 1\right).
\end{aligned}$$

And since $|\phi^+-1|=|\phi^--1|$, we end up with:

$$\begin{aligned}
&\left(\left|\phi^+\left(x-\frac{u}{a_{n,d}}\right)-G(x)\right|\wedge 1\right)+\left(\left|\phi^-\left(x-\frac{u}{a_{n,d}}\right)-G(x)\right|\wedge 1\right)\\
&\qquad\geq\left|\phi^+\left(x-\frac{u}{a_{n,d}}\right)-1\right|\wedge 1=\phi^+\left(x-\frac{u}{a_{n,d}}\right)-1.
\end{aligned}$$

So

$$\begin{aligned}
&\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_{1|f_1}}}\left(L'_n\left(\mathfrak{s}'\right)|(Y_1,f(Y_1)),\dots(Y_n,f(Y_n)),G\right)\\
&\qquad\geq\frac{1}{6}\sum_{u\neq u_1,\dots u_n}\int_{H_u}\left(\phi^+\left(x-\frac{u}{a_{n,d}}\right)-1\right)dx.
\end{aligned}$$

And

$$\sum_{u \neq u_1,\ldots u_n} \int_{H_u} \left(\phi^+\left(x - \frac{u}{a_{n,d}}\right) - 1\right) dx \geq (a_{n,d}^d - n) \int_{[0,1/a_{n,d}]^d} \big(\phi^+(x) - 1\big)\, dx$$

$$\geq (a_{n,d}^d - n) \int_{[1/(4a_{n,d}),3/(4a_{n,d})]^d} \big(\phi^+(x) - 1\big)\, dx.$$

Now, for any $x \in [1/(4a_{n,d}), 3/(4a_{n,d})]^d$, we have $\phi^+(x) - 1 \geq (4a_{n,d})^{-s}$.
Then

$$\sum_{u \neq u_1,\ldots u_n} \int_{H_u} \left(\phi^+\left(x - \frac{u}{a_{n,d}}\right) - 1\right) dx \geq (a_{n,d}^d - n)(4a_{n,d})^{-s}(2a_{n,d})^{-d}$$

$$\geq 2^{-3s-2d}5^{-s/d}n^{-s/d},$$

where the second inequality used the fact that $a_{n,d} \leq 2(5n)^{1/d}$.
Hence, there exists $N(s,d)$, such that for n larger than $N(s,d)$,

$$\mathbb{E}_{f\sim\mathcal{D}_{\mathcal{F}_1}}(L_n'(\mathfrak{s}')) \geq 3^{-1}2^{-1-3s-2d}5^{-s/d}n^{1-s/d}.$$

■