UNIVERSITÉ RENNES 2

HABILITATION À DIRIGER DES RECHERCHES

CONTRIBUTIONS
À L'ESTIMATION NON PARAMÉTRIQUE
ET
À LA SIMULATION D'ÉVÉNEMENTS RARES

par

Arnaud GUYADER

Soutenue le **13 décembre 2011** devant le jury composé de :

Gérard BIAU	Université Pierre et Marie Curie	Examinateur
Bernard DELYON	Université Rennes 1	Examinateur
Luc DEVROYE	McGill University	Rapporteur
Arnaud DOUCET	Oxford University	Rapporteur
Josselin GARNIER	Université Paris VII	Rapporteur
Nicolas HENGARTNER	Los Alamos National Laboratory	Examinateur
Eric MATZNER-LØBER	Université Rennes 2	Examinateur

Remerciements

En premier lieu, je tiens à remercier Luc Devroye, Arnaud Doucet et Josselin Garnier d'avoir accepté de rapporter sur cette habilitation en une période de l'année où, notoriété scientifique oblige, leurs agendas sont vraisemblablement aussi chargés que des cyclistes du Tour de France prêts à affronter la Grande Boucle.

Je suis également reconnaissant à Gérard Biau, le Woody Allen de la statistique (prolifique, hypocondriaque et clarinettiste), d'avoir pris sur son précieux temps pour participer au jury malgré le tsunami d'enseignements sous lequel il manque de sombrer en cette fin de premier semestre (*sic*). Dans une vie précédente, j'ai fait la connaissance de Bernard Delyon en tant que prête-nom comme directeur de thèse, il est devenu depuis un interlocuteur de choc pour débattre de toute nouvelle idée, que celle-ci concerne les probabilités, les statistiques ou autre, rien de ce qui est mathématique ne lui étant jamais complètement étranger. En espérant qu'il a réussi à prendre son avion (ou le suivant (ou encore le suivant)), je remercie également Nick Hengartner, *a.k.a.* le Keyser Söze de Los Alamos, d'avoir fait le déplacement depuis les États-Unis, quitte à avoir dû abandonner ses trois femmes pétillantes et ses deux chiens stupides. Enfin, au-delà du caractère anecdotique de ce titre, je suis ravi d'avoir comme directeur d'habilitation Eric Matzner-Løber, alias le Jason Bourne de Rennes 2, même si je désespère de parvenir un jour à cerner les méandres de son emploi du temps.

Côté INRIA, mes remerciements vont tout d'abord à François le Gland : faisant fi de mes lacunes sur ses thématiques à l'issue d'une thèse assez peu enthousiasmante, il n'a pas hésité un instant à m'associer à son équipe de recherche. C'est donc en grande partie grâce à lui que j'ai pu commencer à bricoler avec Frédéric Cérou, que ce soit en statistique non paramétrique ou en simulation d'événements rares. Ainsi avons-nous illustré, à notre niveau, les bienfaits des interactions et ce non uniquement pour les méthodes Monte-Carlo. À la périphérie de l'équipe ASPI, j'ai aussi eu la chance de côtoyer Pierre Del Moral (le Jackson Pollock des formules de Feynman-Kac), Tony Lelièvre (le Winston Wolfe de la dynamique moléculaire) et Florent Malrieu (l'inventeur du fameux Calcul de Malrieu), avec qui ce fut un réel plaisir de travailler.

Côté Rennes 2, je salue le dévouement sans faille de Jacques Bénasséni en tant que chef de département, un rôle dans lequel il excelle depuis bientôt trois ans. Lorsqu'on songe, ne serait-ce que dans l'histoire récente, aux hommes admirables qui ont vaillamment effectué plusieurs mandats pour le bien du plus grand nombre (Silvio Berlusconi, George W. Bush, John Edgar Hoover, Joseph Staline, Margaret Thatcher, etc.), on se demande franchement pourquoi il regimbe à l'idée de poursuivre sur cette belle lancée. Mes respects également à Dominique Dehay, qui parvient à rester étonnamment calme, objectif et honnête même lors des périodes de fortes houles qui secouent parfois le laboratoire. Enfin, un grand merci à Sébastien Bruneau, Marie-Laure Chatelet, Christine Menhour, Nelly Oger et Annabelle Proust-Granger qui accomplissent l'exploit quotidien de parvenir à travailler efficacement sur les logiciels administratifs les plus poussifs du cosmos.

Sur un plan plus personnel et restreint au seul périmètre rennais, je salue bien sûr les honorables membres du Binchsbury Group (Alexandre, Anne-Sophie, Christophe, Gaspard, Jérôme, Lau-

rent, Marie, Marine, Pierre-André), en remerciant tout particulièrement Elisa Barbolini (Audrey Hepburn, version *Vacanze Romane*) et Julie Josse (Audrey Hepburn itou, version *Breakfast at Tyffany's*) pour leur soutien logistique. Je ne saurais oublier Teddy Furon (le Michael Scofield du watermarking), François Husson (le Gaudí de *Statistiques avec R*), Nicolas Jégou (le Joe Strummer des Jackalope) et Ewa Kijak (la Maggie Fitzgerald du badminton).

Pour conclure, j'ai bien entendu une pensée pour mes parents, mes frères, ainsi que pour ma sœur Anne (la Debra Morgan de Spie Fondations).

Contents

Résumé

Ce document synthétise mes travaux de recherche depuis la fin de ma thèse. Les deux parties correspondent à des thèmes très largement indépendants. La première porte sur la statistique non paramétrique, plus précisément sur les questions relatives aux méthodes de type plus proches voisins, que ce soit en dimension finie ou infinie. La seconde concerne la simulation et l'estimation d'événements rares par des méthodes Monte-Carlo en interaction.

Méthodes de plus proches voisins

La méthode dite des plus proches voisins est l'une des techniques les plus classiques en estimation non paramétrique. Que ce soit en classification supervisée, en régression ou en estimation de la densité, l'idée est d'estimer le label Y, la fonction de régression r ou la densité f en un point $\mathbf{X}$ via une moyennisation sur ses voisins les plus proches dans l'échantillon d'apprentissage $\mathcal{D}_n = (\mathbf{X}_i, Y_i)_{1 \leq i \leq n}$.

Si les propriétés de cette règle très simple sont bien connues lorsque la variable explicative $\mathbf{X}$ est à valeurs dans $\mathbb{R}^d$, il n'en va pas de même lorsqu'elle vit dans un espace de dimension infinie. On parle alors de statistiques pour données fonctionnelles (courbes, etc.), domaine connaissant un intérêt croissant depuis une vingtaine d'années grâce aux puissances de calcul à disposition.

Dans le chapitre 1, basé sur l'article [31], nous montrons que la consistance universelle de la méthode des plus proches voisins, vraie en dimension finie, ne l'est plus en dimension infinie et donnons une condition suffisante de consistance dans ce cadre.

Ceci étant acquis, le chapitre 2 (référence [13]) prouve que sous des hypothèses raisonnables, les vitesses de convergence en dimension infinie sont typiquement logarithmiques, i.e. en $(\log n)^{-\alpha}$, et non plus en $n^{-\alpha}$ comme en dimension finie. Ce sont à notre connaissance les premières vitesses à avoir été exhibées dans un cadre fonctionnel.

Enfin, en dimension finie, le chapitre 3 montre la puissance des méthodes d'ensemble en prouvant qu'une règle aussi fruste que celle du plus proche voisin, lorsqu'elle est agrégée, permet d'atteindre des vitesses optimales de convergence. Cette section correspond à un résumé de l'article [12].

Méthodes Monte-Carlo pour les événements rares

Les méthodes Monte-Carlo ont pour principe général de recourir à la simulation pour estimer des quantités hors d'atteinte via des techniques analytiques classiques. Néanmoins, lorsque l'événement à estimer est de probabilité très faible, disons moins d'une chance sur un million, mais d'importance pratique cruciale, la procédure Monte-Carlo standard devient trop imprécise et demande donc à être affinée. Dans cette situation, on distingue généralement deux types d'approches : échantillonnage préférentiel d'un côté (*importance sampling*), méthodes multi-niveaux de l'autre (*multilevel*

splitting). C'est de cette seconde famille dont nous discutons ici.

Les contributions d'ordre méthodologique sont résumées dans le chapitre 4. Le premier article sur ce thème (voir [27]) porte sur les événements rares pour des processus stochastiques unidimensionnels. Dans ce cadre, nous avons proposé un nouvel algorithme, dit multi-niveaux adaptatif, et démontré son optimalité en terme de variance asymptotique d'estimation. La version de cet algorithme dans un cadre statique, typiquement un événement rare pour un vecteur aléatoire de grande dimension, a ensuite été proposée et étudiée dans [23]. Elle fait l'objet de la section 4.3 du présent document. Enfin, une variante permettant d'obtenir des résultats non asymptotiques est présentée en section 4.4. Elle est basée sur l'article [64].

L'algorithme de la section 4.3, bien que très général, a été initialement proposé pour répondre à des questions issues de la protection de données numériques par tatouage (*watermarking*) et par traçage de traître (*fingerprinting*). Il s'agit alors de déterminer très précisément la fiabilité d'un système en termes de probabilités de fausse alarme. Ces applications, présentées en sections 5.1 et 5.2, correspondent aux publications [23, 26, 30]. Les problèmes de dénombrement en grande dimension représentent un autre champ d'application de nos méthodes : en satisfiabilité, on veut par exemple déterminer les éventuelles solutions d'un très grand système d'équations booléennes. C'est ce qu'explique la section 5.3, basée sur l'article [29].

Dans un cadre non plus statique mais dynamique, la variante proposée en section 4.4 a été appliquée à la simulation de trajectoires réactives en dynamique moléculaire. L'enjeu est cette fois d'étudier aussi finement que possible le passage d'un état métastable à un autre. Le chapitre 6 expose succinctement ce champ applicatif, plus de détails étant disponibles dans l'article [28].

Enfin, l'annexe A présente un résultat non asymptotique sur l'approximation particulaire de modèles de Feynman-Kac non normalisés. Par rapport à la présentation générale faite dans l'article [24], nous insistons surtout ici sur son application en terme d'efficacité pour l'estimation d'événements rares par techniques multi-niveaux.

Part I

Nearest Neighbor Rules

Chapter 1

Consistency of the Functional k-Nearest Neighbor Rule

1.1 Introduction

In many experiments, scientists and practitioners often collect samples of curves and other functional observations. For instance, curves arise naturally as observations in the investigation of growth, in climate analysis, in food industry or in speech recognition; Ramsay and Silverman [96] discuss other examples. The aim of the present chapter is to investigate whether the classical nonparametric classification rule based on k-nearest neighbor (as discussed, for example, in Devroye, Györfi and Lugosi [46]) can be extended to classify functions.

Classical classification deals with predicting the unknown nature Y, called a *label*, of an observation $\mathbf{X}$ with values in $\mathbb{R}^d$ (see Boucheron, Bousquet and Lugosi [17] for a survey). Both $\mathbf{X}$ and Y are assumed to be random, and the distribution of $(\mathbf{X}, Y)$ just describes the frequency of encountering particular pairs in practice. We require for simplicity that the label only takes two values, say 0 and 1. Note that, in this framework, the label Y is random, and this casts the classification problem into a bounded regression problem.

The statistician creates a *classifier* $g : \mathbb{R}^d \to \{0, 1\}$ which represents his guess of the label of $\mathbf{X}$. An error occurs if $g(\mathbf{X}) \neq Y$, and the probability of error for a particular classifier g is

$$L(g) = \mathbb{P}\left(g(\mathbf{X}) \neq Y\right) .$$

It is easily seen that the *Bayes rule*

$$g^*(\mathbf{x}) = \begin{cases} 0 & \text{if } \mathbb{P}\left(Y = 0 | \mathbf{X} = \mathbf{x}\right) \geq \mathbb{P}\left(Y = 1 | \mathbf{X} = \mathbf{x}\right) \\ 1 & \text{otherwise,} \end{cases} \tag{1.1}$$

is the optimal decision, in the sense that, for any decision function $g : \mathbb{R}^d \to \{0, 1\}$,

$$L^* = \mathbb{P}\left(g^*(\mathbf{X}) \neq Y\right) \leq \mathbb{P}\left(g(\mathbf{X}) \neq Y\right) .$$

Unfortunately, the Bayes rule depends on the distribution of $(\mathbf{X}, Y)$, which is unknown to the statistician. The problem is thus to construct a reasonable classifier g_n based on independent observations $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$ with the same distribution as $(\mathbf{X}, Y)$.

Among the various ways to define such classifiers, one of the most simple and popular is probably the *$k-$nearest neighbor rule* given by

$$g_n(\mathbf{x}) = \begin{cases} 0 & \text{if } \sum_{i=1}^n w_i \mathbb{1}_{\{Y_i = 0\}} \geq \sum_{i=1}^n w_i \mathbb{1}_{\{Y_i = 1\}} \\ 1 & \text{otherwise,} \end{cases} \tag{1.2}$$

where $w_i = 1/k$ if $\mathbf{X}_i$ is amongst the k nearest neighbors of $\mathbf{x}$, and $w_i = 0$ elsewhere. This simple rule dates back to the fifties and the seminal papers of Fix and Hodges [57, 58]. For a complete and updated list of references, we refer the reader to the monograph by Devroye, Györfi and Lugosi [46], Chapters 5 and 11.

Now, if we are given any classification rule g_n based on the training data $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$, the best we can expect from the classification function g_n is to achieve the Bayes error probability $L^* = L(g^*)$. Generally, we cannot hope to obtain a function that exactly achieves the Bayes error probability, and we rather require that the error probability

$$L_n = \mathbb{P}\left(g_n(\mathbf{X}) \neq Y | (\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\right)$$

gets arbitrarily close to L^* with large probability. More precisely, a classification rule g_n is called *consistent* if

$$\mathbb{E}L_n = \mathbb{P}\left(g_n(\mathbf{X}) \neq Y\right) \to L^* \quad \text{as } n \to \infty.$$

A decision rule can be consistent for a certain class of distributions of $(\mathbf{X}, Y)$, but may not be consistent for others. On the other hand, it is clearly desirable to have a rule that gives good performance for all distributions. With this respect, a decision rule is called *universally consistent* if it is consistent for any distribution of the pair $(\mathbf{X}, Y)$. When $\mathbf{X}$ is $\mathbb{R}^d$-valued, equipped with any vector norm, it is known from Stone [110] that the conditions $k \to \infty$ and $k/n \to 0$ as $n \to \infty$ ensure that the $k-$nearest neighbor rule (1.2) is universally consistent.

In this chapter, we wish to investigate consistency properties of the $k-$nearest neighbor rule (1.2) in the setting of random functions, that is when $\mathbf{X}$ takes values in a metric space $(\mathcal{F}, d)$ instead of $\mathbb{R}^d$. In this general framework, the optimal decision remains the Bayes one $g^* : \mathcal{F} \to \{0, 1\}$ as in (1.1). Probably due to the difficulty of the problem, and despite nearly unlimited applications, the theoretical literature on regression and classification in infinite dimensional spaces is relatively recent. Key references on this topic are Rice and Silverman [97], Kneip and Gasser [77], Kulkarni and Posner [80], Ramsay and Silverman [96], Bosq [15], Ferraty and Vieu [56], Diabo-Niang and Rhomari [38], Hall, Poskitt and Presnell [67], Abraham, Cornillon, Matzner-Løber and Molinari [2], Antoniadis and Sapatinas [5], and Biau, Bunea and Wegkamp [11]. We also mention that Cover and Hart [36] consider classification of Banach space valued elements as well, but they do not establish consistency.

The classification rule (1.2) is fed with infinite-dimensional observations as inputs. In particular, it does not require any preliminary dimension reduction or model selection step. On the other hand, in the so-called "filtering approach", one first reduces the infinite dimension of the observations by considering only the first m coefficients of the data on an appropriate basis, and then perform finite dimensional classification. For more on this alternative approach, we refer the reader to Hall, Poskitt and Presnell [67], Abraham, Cornillon, Matzner-Løber and Molinari [2], Biau, Bunea and Wegkamp [11], and the references therein.

As a first contribution, we show in Section 1.2.1 that the universal consistency result valid for the rule (1.2) in the finite dimensional case when $\mathbb{R}^d$ is equipped with a vector norm, breaks down as soon as an arbitrary distance is allowed. More precisely, we are able to exhibit a distance on $[0, 1]$ and a distribution of $(\mathbf{X}, Y)$ such that the $k-$nearest neighbor rule (1.2) fails to be consistent. This negative finding makes it legitimate to put some restrictions both on the functional space and the distribution of $(\mathbf{X}, Y)$ in order to obtain the desired consistency property. Sufficient conditions of this sort are given in Section 1.2.2. Finally, these conditions are discussed in Section 1.3.

1.2 A Consistency Result

Let us first introduce a few notations for the abstract mathematical model. Let $\mathbf{X}$ be a random variable taking values in a metric space $(\mathcal{F}, d)$ and let Y be a random label with values 0 and 1. The distribution of the pair $(\mathbf{X}, Y)$ is completely specified by μ, the probability measure of $\mathbf{X}$ and by r, the regression function of Y on $\mathbf{X}$. That is, for any Borel-measurable set $A \subset \mathcal{F}$,

$$\mu(A) = \mathbb{P}(\mathbf{X} \in A)$$

and, for any $\mathbf{x} \in \mathcal{F}$, $r(\mathbf{x}) = \mathbb{P}(Y = 1 | \mathbf{X} = \mathbf{x})$.

1.2.1 Separability of the Metric Space

To generalize Stone's result, the first natural assumption is to require the separability of the metric space $(\mathcal{F}, d)$. The following example shows that this condition is necessary even in finite dimension.

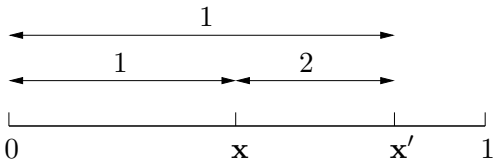

Figure 1.1: A pathological distance on $[0, 1]$.

Example: A pathological distance on $[0, 1]$
Let us define the distance d on $[0, 1]$ as follows (see also figure 1.1):

$$d(\mathbf{x}, \mathbf{x}') = \begin{cases} 0 & \text{if } \mathbf{x} = \mathbf{x}' \\ 1 & \text{if } \mathbf{x}\mathbf{x}' = 0 \text{ and } \mathbf{x} \neq \mathbf{x}' \\ 2 & \text{if } \mathbf{x}\mathbf{x}' \neq 0 \text{ and } \mathbf{x} \neq \mathbf{x}' \end{cases}$$

Since the triangle inequality holds, d is a distance on $[0, 1]$. But $([0, 1], d)$ is clearly not separable. Let us now define μ the probability distribution of $\mathbf{X}$ on $[0, 1]$ as follows: with probability one half, one picks the origin 0; with probability one half, one picks a point uniformly in $[0, 1]$. In other words, if $\lambda_{[0,1]}$ denotes the Lebesgue measure on $[0, 1]$ and δ_0 the Dirac measure at the origin:

$$\mu = \frac{1}{2}\delta_0 + \frac{1}{2}\lambda_{[0,1]}$$

The way to attribute a label Y to a point $\mathbf{x}$ in $[0, 1]$ is deterministic: if $\mathbf{x} = 0$ then $y = 0$, otherwise $y = 1$. As Y is a deterministic function of $\mathbf{X}$, the Bayes risk L^* is equal to zero. Nevertheless, it is intuitively clear that the asymptotic probability of error with the $k-$nearest neighbor rule does not converge to 0:

$$\lim_{n \to \infty} \mathbb{E}[L_n] = \frac{1}{2} > L^* = 0.$$

So the $k-$nearest neighbor classifier is not weakly consistent in this context, although we are in finite dimension.

In general metric spaces, the separability assumption is sufficient to have convergence of the nearest neighbor to the point of interest, as noticed by Cover and Hart [36]. As stated in the following lemma, this is also true for the k-th nearest neighbor, provided that k/n goes to zero when n goes

to infinity. First, a few notations are in order. Let $\mathcal{B}_{\mathbf{x},\delta}$ denote the open ball in $\mathcal{F}$ centered at $\mathbf{x}$ of radius δ. Then we let the support $\mathcal{S}(\mu)$ of the probability measure μ of $\mathbf{X}$ be defined as the collection of all $\mathbf{x}$ with $\mu(\mathcal{B}_{\mathbf{x},\delta}) > 0$ for all $\delta > 0$. Finally, we denote $\mathbf{X}_{(k)}(\mathbf{x})$ the k-th nearest neighbor of $\mathbf{x}$ among all $\mathbf{X}_1, \ldots, \mathbf{X}_n$.

Lemma 1 *If $\mathbf{x}$ is in the support of μ and $\lim_{n\to\infty} k/n = 0$, then $\lim_{n\to\infty} d(\mathbf{X}_{(k)}(\mathbf{x}), \mathbf{x}) = 0$ with probability one. If $\mathbf{X}$ is independent of the data and has probability measure μ, then with probability one*

$$\lim_{n\to\infty} d(\mathbf{X}_k(\mathbf{X}), \mathbf{X}) = 0.$$

We refer to Devroye, Györfi and Lugosi [46], Lemma 5.1, for the proof. The proof there is written in $\mathbb{R}^d$ equipped with the Euclidean norm, but the argument still works in any separable metric space. Consequently, from now on, we will assume that $(\mathcal{F}, d)$ is a separable metric space.

1.2.2 The Lebesgue-Besicovitch Condition

As we will see later, separability of the metric space is not a sufficient assumption for consistency of the $k-$nearest neighbor classifier. It is also necessary to put a regularity assumption on r with respect to μ. More precisely, we will require a differentiation hypothesis that will be called "the Lebesgue-Besicovitch property" (LB condition).

Assumption 1 (Lebesgue-Besicovitch (LB) property) *We say that the LB property is verified if for every $\varepsilon > 0$*

$$\lim_{\delta\to 0} \mu\left\{ \mathbf{x} \in \mathcal{F} : \frac{1}{\mu(\mathcal{B}_{\mathbf{x},\delta})} \int_{\mathcal{B}_{\mathbf{x},\delta}} |r(\mathbf{x}') - r(\mathbf{x})|\, d\mu(\mathbf{x}') > \varepsilon \right\} = 0.$$

Another formulation is the following convergence in μ-probability:

$$\frac{1}{\mu(\mathcal{B}_{\mathbf{x},\delta})} \int_{\mathcal{B}_{\mathbf{x},\delta}} |r - r(\mathbf{X})|\, d\mu \xrightarrow[\delta\to 0]{} 0 \qquad \text{in } \mu\text{-probability.}$$

We will discuss this condition in the final section. Let us now give the main result of this chapter.

Theorem 1 *If $(\mathcal{F}, d)$ is separable and if the LB property is fulfilled, then the $k-$nearest neighbor classifier is consistent*

$$\mathbb{E}L_n \xrightarrow[n\to\infty]{} L^*.$$

Remark. In finite dimension, Devroye already mentions in [45] that the LB property plays a key role for nearest neighbor estimates as well as for kernel estimates.

1.3 Discussion

1.3.1 Continuity of the Regression Function

If r is continuous on $(\mathcal{F}, d)$, then obviously the LB condition is fulfilled. However, intuitively, continuity is not necessary, since the key idea of nearest neighbor classification is the following: to guess the label Y of a new point $\mathbf{X}$, just average the labels Y_i for points $\mathbf{X}_i$ around $\mathbf{X}$. The continuous version which ensures the validity of this averaging method has an integral form: this

is exactly the LB property.

We will illustrate this point through an example where r is nowhere continuous, but where the k-nearest neighbor classifier is still consistent anyway. Before that, we formulate a stronger but more tractable assumption than the LB property, called the "μ-continuity property".

Assumption 2 (μ-continuity property) *We say that the μ-continuity property is verified if for every $\varepsilon > 0$, for μ almost every $\mathbf{x} \in \mathcal{F}$*

$$\lim_{\delta \to 0} \frac{\mu\left\{\mathbf{x}' \in \mathcal{F} : |r(\mathbf{x}') - r(\mathbf{x})| > \varepsilon \cap d(\mathbf{x}, \mathbf{x}') < \delta\right\}}{\mu\left\{\mathbf{x}' \in \mathcal{F} : d(\mathbf{x}, \mathbf{x}') < \delta\right\}} = 0.$$

This is a kind of continuity of r with respect to the measure μ, hence the name μ-continuity (see figure 1.2 for an illustration). Another equivalent definition is the following almost sure convergence:

$$\frac{1}{\mu(\mathcal{B}_{\mathbf{x},\delta})} \int_{\mathcal{B}_{\mathbf{x},\delta}} \mathbb{1}_{\{|r - r(\mathbf{X})| > \varepsilon\}} \, d\mu \xrightarrow[\delta \to 0]{} 0 \qquad \mu - a.s.$$

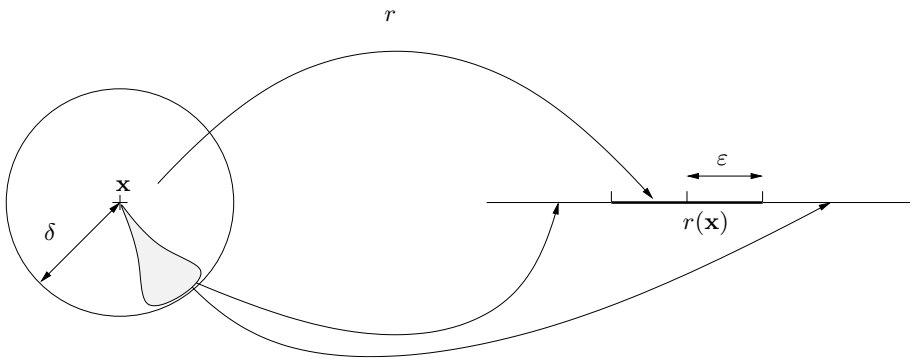

Figure 1.2: $\mu-$continuity: another way to see the Lebesgue-Besicovitch property.

Lemma 2 (μ-continuity $\Rightarrow$ Lebesgue-Besicovitch) *If the regression function r is μ-continuous, then the LB property is verified.*

One can in fact see that μ-continuity property is equivalent to almost sure convergence in the Besicovitch condition:

$$\frac{1}{\mu(\mathcal{B}_{\mathbf{x},\delta})} \int_{\mathcal{B}_{\mathbf{x},\delta}} |r - r(\mathbf{X})| \, d\mu \xrightarrow[\delta \to 0]{} 0 \qquad \mu - a.s.$$

As we will see in the following example, the $\mu-$continuity property 2 may be easier to check than the LB property.

Example: Trajectories of a Poisson process on $[0, 1]$
$\mathcal{F}$ is the space of all possible realizations of a Poisson process of intensity 1 between initial time 0 and final time 1. Its elements are denoted $\mathbf{x} = (x_t)_{0 \leq t \leq 1}$ or $\mathbf{x}' = (x'_t)_{0 \leq t \leq 1}$. The distance on $\mathcal{F}$ is derived from the L_1 norm:

$$d(\mathbf{x}, \mathbf{x}') = \|\mathbf{x} - \mathbf{x}'\|_1 = \int_0^1 |x_t - x'_t| dt$$

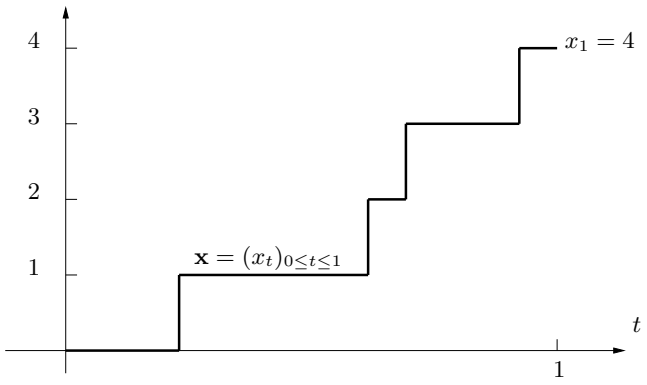

Figure 1.3: A trajectory $\mathbf{x} = (x_t)_{0 \leq t \leq 1}$ of a Poisson process.

The metric space $(\mathcal{F}, ||.||_1)$ is obviously separable: consider for example the trajectories that jump at rational times between time 0 and time 1. This is a countable set, and for every $\delta > 0$ and every $\mathbf{x} \in \mathcal{F}$, there exists such a trajectory in the ball $\mathcal{B}_{\mathbf{x},\delta}$.

Given a trajectory $\mathbf{x}$, its label is deterministic and depends only on its terminal point: if x_1 is even, then $y = 0$, otherwise $y = 1$. As a consequence, the Bayes risk L^* is null. Moreover, it is readily seen that r is nowhere continuous. Indeed, let us fix $\mathbf{x} \in \mathcal{F}$, $\delta \in (0, 1)$, and consider $\mathbf{x}' \in \mathcal{F}$ defined as follows (see figure 1.4):

$$x'(t) = \begin{cases} x(t) & \text{if } 0 \leq t \leq 1 - \delta \\ x(t) + 1 & \text{if } 1 - \delta < t \leq 1 \end{cases}$$

So $\mathbf{x}'$ is at distance δ from $\mathbf{x}$ but has not the same label as $\mathbf{x}$: since δ is arbitrary, this proves that r is not continuous at point $\mathbf{x}$. Since $\mathbf{x}$ is arbitrary, this proves that r is nowhere continuous.

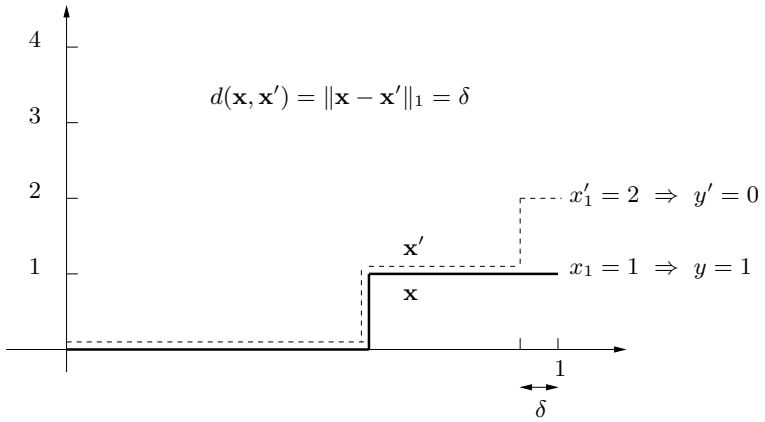

Figure 1.4: The trajectories $\mathbf{x}$ and $\mathbf{x}'$ do not have the same label.

However, we can prove that the k-nearest neighbor rule is consistent by checking that the LB property is fulfilled. In this aim, we make use of the more tractable μ-continuity property. Let us fix $\varepsilon > 0$ and $\mathbf{x} \in \mathcal{F}$. The aim is to show that

$$\lim_{\delta \to 0} \frac{\mu\left(\{|r(\mathbf{x}') - r(\mathbf{x})| > \varepsilon\} \cap \mathcal{B}_{\mathbf{x},\delta}\right)}{\mu(\mathcal{B}_{\mathbf{x},\delta})} = 0.$$

The following result provides some useful upper-bound.

Lemma 3 (The ratio of small balls) *If the trajectory* $\mathbf{x}$ *has no jump, then*

$$\frac{\mu\left(\{|r(\mathbf{x}')-r(\mathbf{x})|>\varepsilon\}\cap\mathcal{B}_{\mathbf{x},\delta}\right)}{\mu(\mathcal{B}_{\mathbf{x},\delta})}\leq\frac{e^{\delta}-1}{1+\delta}.$$

If the trajectory $\mathbf{x}$ *has* m *jumps, then*

$$\frac{\mu\left(\{|r(\mathbf{x}')-r(\mathbf{x})|>\varepsilon\}\cap\mathcal{B}_{\mathbf{x},\delta}\right)}{\mu(\mathcal{B}_{\mathbf{x},\delta})}\leq\frac{C_2(x)}{C_1(x)}\left[\frac{\sinh(\delta)}{\sinh(\frac{\delta}{m^2})}\right]^m\sinh(2\delta),$$

where the constants $C_1(x)$ *and* $C_2(x)$ *do not depend on* δ*.*

So, whatever the number of jumps of $\mathbf{x}$, it turns out that

$$\lim_{\delta\to0}\frac{\mu\left(\{|r(\mathbf{x}')-r(\mathbf{x})|>\varepsilon\}\cap\mathcal{B}_{\mathbf{x},\delta}\right)}{\mu(\mathcal{B}_{\mathbf{x},\delta})}=0.$$

As a consequence, the k-nearest neighbor rule is consistent, although r is nowhere continuous.

1.3.2 The Lebesgue-Besicovitch Condition in Infinite Dimension

In this section, we discuss the LB property. First of all, let us say a word about finite dimension. In this case, if $\mathbb{R}^d$ is equipped with a vector norm, the crucial result is the following one (see for instance [53], Chapter 1.7, pp 43-44).

Theorem 2 (Lebesgue-Besicovitch differentiation Theorem) *Let* μ *be a Radon measure on* $\mathbb{R}^d$ *and* $f\in L^p_{\text{loc}}(\mathbb{R}^d)$*, then*

$$\lim_{\delta\to0}\frac{1}{\mu(\mathcal{B}_{\mathbf{x},\delta})}\int_{\mathcal{B}_{\mathbf{x},\delta}}|f-f(\mathbf{x})|^p\,d\mu=0$$

for μ *almost every* $\mathbf{x}$*.*

In a classification context, μ is a probability measure on $\mathbb{R}^d$ and r is bounded by 1, so this result can be directly applied. Devroye already noted in [45] that this is another way to prove Stone's Theorem.

Corollary 1 (Stone's Theorem) *In* $(\mathbb{R}^d,||.||)$*, the* k*-nearest neighbor classifier is universally consistent.*

The LB condition also appears in recent papers on connected problems: Abraham, Biau and Cadre [1] use it for function classification with the kernel rule. Dabo-Niang and Rhomari [38] require it for nonparametric regression estimation in general metric spaces.

Now, concerning the LB property in infinite dimension, there have been several attempts to generalize this kind of result in general metric (separable) spaces. Interestingly, this topic has been investigated by several authors in geometric measure theory, see for example Federer [55], Preiss [93], and Preiss and Tišer [94]. Unlike the situation in $(\mathbb{R}^d,\|.\|)$, the LB property is no longer automatically fulfilled in infinite dimension.

In [93], Preiss introduces a rather technical notion, called the σ-finite dimensionality of a metric on a space. He shows that it is a sufficient condition for the LB property for all measures on a metric space. Without delving into the details of this notion, let us just mention that it is related

to the σ-finite dimensionality of the space. We can illustrate this idea on our example of Poisson trajectories on $[0,1]$.

Example. Fix $m \geq 0$ and denote $\mathcal{F}_m$ as all possible realizations of the Poisson process that have exactly m jumps. A process that has m jumps can be summarized by an m-dimensional vector of jump times. Then it is obvious that the metric space $(\mathcal{F}_m, \|.\|_1)$ is isometric to $([0,1]^m, \|.\|_1)$

$$(\mathcal{F}, \|.\|_1) = \bigcup_{m=0}^{+\infty} (\mathcal{F}_m, \|.\|_1) \sim \bigcup_{m=0}^{+\infty} ([0,1]^m, \|.\|_1),$$

and the σ-finite dimensionality is clear.

Let us focus now on the classical situation where $(\mathcal{F}, d)$ is a separable Hilbert space and μ a Gaussian measure. Let ν denote the centered and normalized Gaussian measure on $\mathbb{R}$, let (c_n) be a non-increasing sequence of positive numbers such that $\sum_{n=0}^{+\infty} c_n < +\infty$ and let $\ell_2(c)$ be the set of all sequences $\mathbf{x} = (x_n)$ such that

$$|\mathbf{x}|^2 = \sum_{n=0}^{+\infty} c_n x_n^2 < +\infty.$$

Then $\mu = \nu^{\otimes \mathbb{N}}$ is a σ-additive measure on the Hilbert space $\ell_2(c)$. In fact, each Gaussian measure can be represented in this way.

Even in this rather comfortable context, it turns out that one has to put conditions on the sequence (c_n) to get the Lebesgue-Besicovitch property. Namely, Preiss and Tišer [94] prove the following result: if there exists $q < 1$ such that

$$\forall n \in \mathbb{N} \qquad \frac{c_{n+1}}{c_n} < q,$$

then the LB property is true for every function $f \in L^1(\mu)$. Roughly speaking, if we see (c_n) as the sequence of variances of μ along the directions of the base vectors, it means that these variances have to decay exponentially fast: this is a very strong assumption.

Now let us see an example which shows that if the LB property is not satisfied, $k-$nearest neighbor classification might not work. This example is due to Preiss in [92].

Example: A problematic case for nearest neighbor classification
In [92], Preiss constructs a Gaussian measure μ on a separable Hilbert space $\mathcal{F}$ and a Borel set $M \subset \mathcal{F}$ with $\mu(M) < 1$ such that

$$\lim_{\delta \to 0} \frac{\mu(M \cap \mathcal{B}_{\mathbf{x},\delta})}{\mu(\mathcal{B}_{\mathbf{x},\delta})} = 1$$

for μ almost every $\mathbf{x} \in \mathcal{F}$.

Now suppose that $\mathbf{X}$ has law μ and that its label Y is deterministic, defined as $Y = \mathbb{1}_M(\mathbf{X})$. As usual the Bayes risk is then equal to 0. Nevertheless, in this situation, the k-nearest neighbor rule fails in classifying elements $\mathbf{x} \in \overline{M}$. Indeed, one can prove that

$$\varliminf_{n \to \infty} L_n^* \geq \frac{1}{2}\mu(\overline{M}) > 0 = L^*.$$

As a conclusion, let us mention that this result is not in contradiction with the one obtained by Biau, Bunea and Wegkamp in [11]. In this paper, they consider a random variable $\mathbf{X}$ taking

values in a separable Hilbert space $\mathcal{F}$, with label $Y \in \{0, 1\}$. They establish the universal weak consistency of a neighbor-*type* classifier, but not of the *classical* $k-$nearest neighbor classifier. Namely, they reduce the infinite dimension of $\mathcal{F}$ by considering only the first m coefficients of the Fourier series expansion of each $\mathbf{X}_i$, and then perform nearest neighbor classification in $\mathbb{R}^m$.

Chapter 2

Rates of Convergence of the Functional k-Nearest Neighbor Estimate

2.1 Introduction

Let $(\mathcal{F}, \|.\|)$ be a separable Banach space, and let $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ be independent $\mathcal{F} \times \mathbb{R}$-valued random variables with the same distribution as a generic pair $(\mathbf{X}, Y)$ such that $\mathbb{E}Y^2 < \infty$. In the regression function estimation problem, the goal is to estimate the regression function $r(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$ using the data $\mathcal{D}_n$. With this respect, we will say that a regression estimate $r_n(\mathbf{x})$ is consistent if $\mathbb{E}[r_n(\mathbf{X}) - r(\mathbf{X})]^2 \to 0$ as $n \to \infty$.

In the classical statistical setting, each observation $\mathbf{X}_i$ is supposed to be a collection of numerical measurements represented by a d-dimensional vector. Thus, to date, most of the results pertaining to regression estimation have been reported in the finite-dimensional case, where it is assumed that $\mathcal{F}$ is the standard Euclidean space $\mathbb{R}^d$. We refer the reader to the monograph of Györfi, Kohler, Krzyżak and Walk [66] for a comprehensive introduction to the subject and an overview of most standard methods and developments in $\mathbb{R}^d$.

However, in an increasing number of practical applications, input data items are in the form of random functions (speech recordings, multiple time series, images...) rather than standard vectors, and this casts the regression problem into the general class of functional data analysis. Here, "random functions" means that the variable $\mathbf{X}$ takes values in a space $\mathcal{F}$ of functions on a subset of $\mathbb{R}^d$, equipped with an appropriate norm. For example, $\mathcal{F}$ could be the Banach space of continuous real functions on $\mathcal{X} = [0, 1]^d$ with the norm

$$\|f\|_\infty = \sup_{x \in \mathcal{X}} |f(x)|,$$

but many other choices are possible. The challenge in this context is to infer the regression structure by exploiting the infinite-dimensional nature of the observations. The last few years have witnessed important developments in both the theory and practice of functional data analysis, and many traditional statistical tools have been adapted to handle functional inputs. The book of Ramsay and Silverman [96] provides a presentation of the area.

Interestingly, functional observations also arise naturally in the so-called kernel methods for general pattern analysis. These methods are based on the choice of a proper similarity measure, given by a positive definite kernel defined between pairs of objects of interest, to be used for inferring general types of relations. The key idea is to embed the observations at hand into a (typically infinite-dimensional) Hilbert space, called the feature space, and to compute inner products efficiently

directly from the original data items using the kernel function. For an exhaustive presentation of kernel methodologies and related algorithms, we refer the reader to Schölkopf and Smola [104], and Shawe-Taylor and Cristianini [106].

Motivated by this broad range of potential applications, we propose, in the present chapter, to investigate rates of convergence properties of the k_n-nearest neighbor (k_n-NN) regression estimate, assuming that the $\mathbf{X}_i$'s take values in a general separable Banach space $(\mathcal{F}, \|.\|)$, typically infinite-dimensional. Recall that, for $\mathbf{x}$ in $\mathcal{F}$, the k_n-NN estimate is defined by

$$r_n(\mathbf{x}) = \frac{1}{k_n} \sum_{i=1}^{k_n} Y_{(i,n)}(\mathbf{x}),$$

where $(\mathbf{X}_{(1,n)}(\mathbf{x}), Y_{(1,n)}(\mathbf{x})), \ldots, (\mathbf{X}_{(n,n)}(\mathbf{x}), Y_{(n,n)}(\mathbf{x}))$ denotes a reordering of the data according to the increasing values of $\|\mathbf{X}_i - \mathbf{x}\|$ (ties are broken in favor of smallest indices). This procedure is one of the oldest approaches to regression analysis, dating back to Fix and Hodges [57, 58]. It is among the most popular nonparametric methods, with over 900 research articles published on the method since 1981 alone. For implementation, it requires only a measure of distance in the sample space, hence its popularity as a starting-point for refinement, improvement and adaptation to new settings (see for example Devroye, Györfi and Lugosi [46], Chapter 19).

Stone [110] proved the striking result that the estimate r_n is universally consistent if $\mathcal{F} = \mathbb{R}^d$, provided $k_n \to \infty$ and $k_n/n \to 0$. Here, "universally consistent" means that the method is consistent for all distributions of $(\mathbf{X}, Y)$ with $\mathbb{E}Y^2 < \infty$ (universally consistent regression estimates can also be obtained by other local averaging methods as long as $\mathcal{F} = \mathbb{R}^d$, see e.g. [66]). As mentioned in the previous chapter, it turns out that the story is radically different in general spaces $\mathcal{F}$. In this respect, we have presented counterexamples indicating that the estimate r_n is not universally consistent for general $\mathcal{F}$, and that restrictions on $\mathcal{F}$ and the distribution of $(\mathbf{X}, Y)$ cannot be dispensed with.

In this chapter, we go one step further in the analysis and study the rates of convergence of $\mathbb{E}[r_n(\mathbf{X}) - r(\mathbf{X})]^2$ as $n \to \infty$, when $\mathbf{X}$ is allowed to take values in the separable Banach space $\mathcal{F}$. This important question has been first addressed by Kulkarni and Posner [80], who put forward the essential role played by the covering numbers of the support of the distribution of $\mathbf{X}$. Building upon the ideas in [80] and exploiting recent advances on compact embeddings of functional Banach spaces, we present explicit and general finite sample upper bounds on $\mathbb{E}[r_n(\mathbf{X}) - r(\mathbf{X})]^2$, and particularize our results to classical function spaces such as Sobolev spaces, Besov spaces and reproducing kernel Hilbert spaces.

2.2 Bias-Variance Tradeoff

Setting

$$\tilde{r}_n(\mathbf{x}) = \frac{1}{k_n} \sum_{i=1}^{k_n} r\left(\mathbf{X}_{(i,n)}(\mathbf{x})\right),$$

we start the analysis with the standard variance/bias decomposition (Györfi, Kohler, Krzyżak and Walk [66])

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 = \mathbb{E}\left[r_n(\mathbf{X}) - \tilde{r}_n(\mathbf{X})\right]^2 + \mathbb{E}\left[\tilde{r}_n(\mathbf{X}) - r(\mathbf{X})\right]^2. \tag{2.1}$$

The first term is a variance term, which can be upper bounded independently of the topological structure of the space $\mathcal{F}$. Proof of the next proposition can be found for example in [66], Chapter 6:

Proposition 1 *Suppose that, for all* $\mathbf{x} \in \mathcal{F}$,

$$\sigma^2(\mathbf{x}) = \mathrm{Var}[Y \,|\, \mathbf{X} = \mathbf{x}] \leq \sigma^2.$$

Then

$$\mathbb{E}\left[r_n(\mathbf{X}) - \tilde{r}_n(\mathbf{X})\right]^2 \leq \frac{\sigma^2}{k_n}.$$

The right-hand term in (2.1), which is a bias term, needs more careful attention. Let the symbol $\lfloor . \rfloor$ denote the integer part function. A quick inspection of the finite-dimensional proof (see [66], page 95) reveals the following result:

Proposition 2 *Suppose that, for all* $\mathbf{x}$ *and* $\mathbf{x}' \in (\mathcal{F}, \|.\|)$,

$$\left|r(\mathbf{x}) - r(\mathbf{x}')\right| \leq L\|\mathbf{x} - \mathbf{x}'\|, \tag{2.2}$$

for some positive constant L. *Then*

$$\mathbb{E}\left[\tilde{r}_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \leq L^2\,\mathbb{E}\|\mathbf{X}_{(1,\lfloor \frac{n}{k_n} \rfloor)}(\mathbf{X}) - \mathbf{X}\|^2.$$

Putting Proposition 1 and Proposition 2 together, we obtain

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \leq \frac{\sigma^2}{k_n} + L^2\,\mathbb{E}\|\mathbf{X}_{(1,\lfloor \frac{n}{k_n} \rfloor)}(\mathbf{X}) - \mathbf{X}\|^2. \tag{2.3}$$

Thus, in order to bound the rate of convergence of $\mathbb{E}[r_n(\mathbf{X}) - r(\mathbf{X})]^2$, we need to analyze the rate of convergence of *the* nearest neighbor distance in the Banach space $\mathcal{F}$. As noticed in Kulkarni and Posner [80], this task can be achieved via the use of covering numbers of totally bounded sets (Kolmogorov and Tihomirov [79]). Some recalls are in order. Let $\mathcal{B}_{\mathcal{F}}(\mathbf{x}, \varepsilon)$ denote the open ball in $\mathcal{F}$ centered at $\mathbf{x}$ of radius ε.

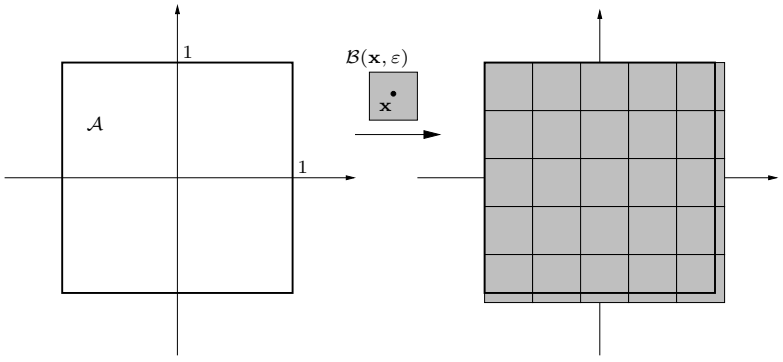

Figure 2.1: An ε-covering of $\mathcal{A} = (-1, 1)^2$ in $(\mathbb{R}^2, \|.\|_\infty)$.

Definition 1 *Let* $\mathcal{A}$ *be a subset of* $\mathcal{F}$. *The* ε*-covering number* $\mathcal{N}(\varepsilon)$ $[= \mathcal{N}(\varepsilon, \mathcal{A})]$ *is defined as the smallest number of open balls of radius* ε *that cover the set* $\mathcal{A}$. *That is*

$$\mathcal{N}(\varepsilon) = \inf\left\{ r \geq 1 \,:\, \exists\, \mathbf{x}_1, \ldots, \mathbf{x}_r \in \mathcal{F} \text{ such that } \mathcal{A} \subset \bigcup_{i=1}^{r} \mathcal{B}_{\mathcal{F}}(\mathbf{x}_i, \varepsilon) \right\}.$$

A set $\mathcal{A} \subset \mathcal{F}$ is said to be totally bounded if $\mathcal{N}(\varepsilon) < \infty$ for all $\varepsilon > 0$. In particular, any relatively compact set is totally bounded, and the converse assertion is true if the space $\mathcal{F}$ is complete. All totally bounded sets are bounded, and the converse assertion is satisfied when $\mathcal{F}$ is finite-dimensional. Figure 2.1 below illustrates this important concept in the finite-dimensional setting, with $(\mathcal{F}, \|.\|) = (\mathbb{R}^2, \|.\|_\infty)$ and $\mathcal{A} = (-1, 1)^2$.

As a function of ε, $\mathcal{N}(\varepsilon)$ is nonincreasing, piecewise-constant and right-continuous (see Figure 2.2 for an illustration). The following discrete function, called the metric covering radius, can be interpreted as a pseudo-inverse of the function $\mathcal{N}(\varepsilon)$.

Definition 2 *Let $\mathcal{A}$ be a subset of $\mathcal{F}$. The metric covering radius $\mathcal{N}^{-1}(r)$ $[= \mathcal{N}^{-1}(r, \mathcal{A})]$ is defined as the smallest radius such that there exist r open balls of this radius which cover the set $\mathcal{A}$. That is*

$$\mathcal{N}^{-1}(r) = \inf \left\{ \varepsilon > 0 \, : \exists \, \mathbf{x}_1, \ldots, \mathbf{x}_r \in \mathcal{F} \text{ such that } \mathcal{A} \subset \bigcup_{i=1}^{r} \mathcal{B}_{\mathcal{F}}(\mathbf{x}_i, \varepsilon) \right\}.$$

We note that $\mathcal{N}^{-1}(r)$ is a nonincreasing function of r (see Figure 2.2 for an illustration). Observe also that both $\mathcal{N}$ and $\mathcal{N}^{-1}$ are increasing with respect to the inclusion, that is $\mathcal{N}(\varepsilon, \mathcal{A}) \leq \mathcal{N}(\varepsilon, \mathcal{B})$ and $\mathcal{N}^{-1}(r, \mathcal{A}) \leq \mathcal{N}^{-1}(r, \mathcal{B})$ for $\mathcal{A} \subset \mathcal{B}$.

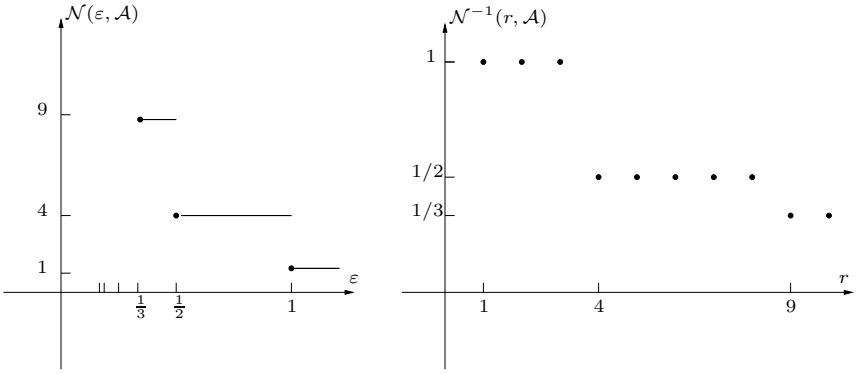

Figure 2.2: Covering numbers and covering radii of the set $\mathcal{A} = (-1, 1)^2$ in $(\mathbb{R}^2, \|.\|_\infty)$.

Finally, we let the support $\mathcal{S}(\mu)$ of the probability measure μ of $\mathbf{X}$ be defined as the collection of all $\mathbf{x}$ with $\mu(\mathcal{B}_{\mathcal{F}}(\mathbf{x}, \varepsilon)) > 0$ for all $\varepsilon > 0$. Throughout the paper, it will be assumed that $\mathcal{S}(\mu)$ is totally bounded. Observe then that $2\mathcal{N}^{-1}(1, \mathcal{S}(\mu))$ is an upper bound on the diameter of $\mathcal{S}(\mu)$.

Proposition 3 below bounds the convergence rate of the expected squared nearest neighbor distance in terms of the metric covering radii of $\mathcal{S}(\mu)$. This result sharpens the constant of Theorem 1, page 1032 in Kulkarni and Posner [80].

Proposition 3 *Let $\mathbf{X}_1, \ldots, \mathbf{X}_p$ be independent $\mathcal{F}$-valued random variables, distributed according to a common probability measure μ. Suppose that $\mathcal{S}(\mu)$ is a totally bounded subset of $(\mathcal{F}, \|.\|)$. Then*

$$\mathbb{E}\|\mathbf{X}_{(1,p)} - \mathbf{X}\|^2 \leq \frac{4}{p} \sum_{i=1}^{p} \left[\mathcal{N}^{-1}\left(i, \mathcal{S}(\mu)\right) \right]^2.$$

Example 2.2.1 *Take* $(\mathcal{F}, \|.\|) = (\mathbb{R}^d, \|.\|_\infty)$ *and suppose that* $\mathcal{S}(\mu) \subset \mathcal{A} = (-1,1)^d$. *Then a moment's thought shows that*

$$\mathcal{N}(\varepsilon, \mathcal{A}) = \left(\frac{1}{\varepsilon}\right)^d \mathbb{1}_{\{\varepsilon^{-1} \in \mathbb{N}\}} + \left(\left\lfloor \frac{1}{\varepsilon} \right\rfloor + 1\right)^d \mathbb{1}_{\{\varepsilon^{-1} \notin \mathbb{N}\}}. \tag{2.4}$$

In addition

$$\mathcal{N}^{-1}(i, \mathcal{A}) = i^{-\frac{1}{d}} \mathbb{1}_{\{i^{1/d} \in \mathbb{N}\}} + \left\lfloor i^{1/d} \right\rfloor^{-1} \mathbb{1}_{\{i^{1/d} \notin \mathbb{N}\}}.$$

Consequently, for $d \geq 3$, *by Proposition 3,*

$$\mathbb{E}\|\mathbf{X}_{(1,p)} - \mathbf{X}\|^2 \lesssim p^{-\frac{2}{d}},$$

where the notation $x \lesssim y$ *means* $x \leq Ay$ *for some positive constant* A. *Combining this result with inequality (2.3), we conclude that*

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \lesssim \frac{\sigma^2}{k_n} + L^2 \left\lfloor \frac{n}{k_n} \right\rfloor^{-\frac{2}{d}}.$$

Thus, for the choice $k_n \propto n^{\frac{2}{d+2}}$,

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \lesssim n^{-\frac{2}{d+2}}.$$

This shows that the nearest neighbor estimate is of optimal rate for the class of smooth distributions $(\mathbf{X}, Y)$ *such that* $\mathbf{X}$ *has compact support, the regression function* r *is Lipschitz with constant* L *and, for all* $\mathbf{x} \in \mathbb{R}^d$, $\mathrm{Var}[Y \,|\, \mathbf{X} = \mathbf{x}] \leq \sigma^2$ *(Ibragimov and Khasminskii [73] and Györfi, Kohler, Krzyżak and Walk [66], Chapter 3 and Theorem 6.2).*

Example 2.2.1 strongly relies on the fact that bounded subsets of $(\mathbb{R}^d, \|.\|_\infty)$ are in fact totally bounded, as expressed by identity (2.4). Indeed, as shown in Proposition 3, a key step in obtaining rates of convergence for the nearest neighbor regression estimate is the derivation of covering numbers for the support of the distribution μ of $\mathbf{X}$. Unfortunately, in infinite-dimensional spaces, closed balls are bounded but not totally bounded, so that $\mathcal{N}^{-1}(i, \mathcal{S}(\mu)) = \infty$ and Proposition 3 is useless.

To correct this situation, a possible route is to assume that the observations we are dealing with behave in fact more regularly than a generic element of the ambient space $\mathcal{F}$, thereby reducing the general complexity of $\mathcal{S}(\mu)$. To illustrate this idea, suppose for example that $\mathcal{F}$ is the space $\mathcal{C}([0,1])$ of continuous real functions on $[0,1]$ equipped with the supremum norm $\|.\|_\infty$. Then, guided by the experience and practical considerations, it may be fair to suppose that the random curves $\mathbf{X}_1, \ldots, \mathbf{X}_n$ are smooth enough, so that the support of their common distribution μ is in fact included and bounded in $\mathcal{D}^m([0,1])$, the space of m times differentiable functions with bounded derivatives, endowed with its canonical norm. Next, in this context, it can be proved that $\mathcal{N}^{-1}(i, \mathcal{S}(\mu)) < \infty$, and the show may go on. This example will be thoroughly discussed in the next section, together with other illustrations.

Thus, taking a general point of view, we will now suppose that the support of μ is bounded and included in a subspace $(\mathcal{G}, \|.\|_\mathcal{G})$ of $(\mathcal{F}, \|.\|)$, and that the embedding $(\mathcal{G}, \|.\|_\mathcal{G}) \hookrightarrow (\mathcal{F}, \|.\|)$ is compact. Here, "compact embedding" means that the unit ball (and thus, any ball) in $(\mathcal{G}, \|.\|_\mathcal{G})$ is totally bounded in $(\mathcal{F}, \|.\|)$. Put differently, balls in $\mathcal{G}$ (with respect to $\|.\|_\mathcal{G}$) become totally bounded as we see them as subsets of $\mathcal{F}$, endowed with the original metric $\|.\|$. The crux then is to identify covering numbers of balls in $\mathcal{G}$ with respect to the norm $\|.\|$. This will be the topic of the next section.

2.3 Compact Embeddings

As we are now working with two different spaces, to avoid notational confusion we will rather denote by $\|.\|_{\mathcal{F}}$ the original norm of $\mathcal{F}$. Thus, in our context, $(\mathcal{G}, \|.\|_{\mathcal{G}})$ is a separable Banach subspace of $(\mathcal{F}, \|.\|_{\mathcal{F}})$ and, to simplify notation a bit, we let in the sequel $\mathcal{B}_{\mathcal{G}}(R)$ be the open ball in $(\mathcal{G}, \|.\|_{\mathcal{G}})$ of radius $R > 0$ centered at the origin, that is

$$\mathcal{B}_{\mathcal{G}}(R) = \{\mathbf{x} \in \mathcal{G} : \ \|\mathbf{x}\|_{\mathcal{G}} < R\}.$$

Definition 3 *The embedding $I : (\mathcal{G}, \|.\|_{\mathcal{G}}) \hookrightarrow (\mathcal{F}, \|.\|_{\mathcal{F}})$ is called compact if $I(\mathcal{B}_{\mathcal{G}}(1))$ is totally bounded in $(\mathcal{F}, \|.\|_{\mathcal{F}})$.*

Note that this definition is equivalent to require that the closure $\overline{I(\mathcal{B})}$ is compact for any bounded set $\mathcal{B} \subset \mathcal{G}$. It turns out that many interesting Banach spaces can be embedded into a larger functional space. To convince the reader, four examples are discussed below.

Example 2.3.1 (Differentiable functions) *Let $\mathcal{X}$ be a compact domain in $\mathbb{R}^d$ with smooth boundary. For every $m \in \mathbb{N}$, let $\mathcal{D}^m(\mathcal{X})$ be the Banach space of m times differentiable functions with bounded partial derivatives, that is*

$$\mathcal{D}^m(\mathcal{X}) = \left\{ f : \mathcal{X} \to \mathbb{R}, \ \|f\|_{\mathcal{D}^m} = \sum_{|\alpha| \leq m} \|D^\alpha f\|_\infty < \infty \right\},$$

where the sum is taken over all multi-indices $\alpha = (\alpha_1, \ldots, \alpha_d)$ such that $|\alpha| = \alpha_1 + \cdots + \alpha_d \leq m$. Then the inclusion

$$I_m : (\mathcal{G}, \|.\|_{\mathcal{G}}) = (\mathcal{D}^m(\mathcal{X}), \|.\|_{\mathcal{D}^m}) \hookrightarrow (\mathcal{F}, \|.\|_{\mathcal{F}}) = (C(\mathcal{X}), \|.\|_\infty)$$

is a compact embedding. Moreover, for every $\varepsilon > 0$ and $R > 0$,

$$\ln \mathcal{N}\left(\varepsilon, \overline{I_m\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq \left(\frac{RC}{\varepsilon}\right)^{\frac{d}{m}},$$

for some positive constant C independent of ε and R (Kolmogorov and Tihomirov [79]). This implies, for $i \in \mathbb{N}^\star$ and $R > 0$,

$$\mathcal{N}^{-1}\left(i, \overline{I_m\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq RC\left(\ln(i+1)\right)^{-\frac{m}{d}}.$$

Example 2.3.2 (Sobolev spaces) *Let again $\mathcal{X}$ be a compact domain in $\mathbb{R}^d$ with smooth boundary. For every $s \in \mathbb{N}$ and $p \geq 1$, let $W^{s,p}(\mathcal{X})$ be the usual Sobolev space equipped with the norm*

$$\|f\|_{W^{s,p}} = \sum_{|\alpha| \leq m} \|D^\alpha f\|_p.$$

The Rellich-Kondrakov Theorem asserts that, for $s_1 > s_2$, the inclusion

$$I_{s_1,s_2} : (\mathcal{G}, \|.\|_{\mathcal{G}}) = (W^{s_1,p}(\mathcal{X}), \|.\|_{W^{s_1,p}}) \hookrightarrow (\mathcal{F}, \|.\|_{\mathcal{F}}) = (W^{s_2,p}(\mathcal{X}), \|.\|_{W^{s_2,p}})$$

is compact. It can be proved (see for example Edmunds and Triebel [52], page 105) that for every $\varepsilon > 0$ and $R > 0$,

$$\ln \mathcal{N}\left(\varepsilon, \overline{I_{s_1,s_2}\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq \left(\frac{RC}{\varepsilon}\right)^{\frac{d}{s_1 - s_2}},$$

for some positive constant C independent of ε and R. This implies, for $s_1 > s_2$, $i \in \mathbb{N}^\star$ and $R > 0$,

$$\mathcal{N}^{-1}\left(i, \overline{I_{s_1,s_2}\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq RC\left(\ln(i+1)\right)^{-\frac{s_1-s_2}{d}}.$$

This result can be extended to the more general context of Sobolev-type function spaces (Edmunds and Triebel [52]).

Example 2.3.3 (Besov spaces) *Let $\mathcal{X}$ be a compact domain in $\mathbb{R}^d$ with smooth boundary, and let $(B^s_{pq}(\mathcal{X}), \|.\|_{spq})$ be the Besov space on $\mathcal{X}$ (Edmunds and Triebel [52]). If $1 \leq p, q \leq \infty$ and $s > d/p$, then the inclusion*

$$I_s : (\mathcal{G}, \|.\|_{\mathcal{G}}) = \left(B^s_{pq}(\mathcal{X}), \|.\|_{spq}\right) \hookrightarrow (\mathcal{F}, \|.\|_{\mathcal{F}}) = (C(\mathcal{X}), \|.\|_\infty)$$

is compact. Besides, using a general result in [52], page 105, we have, for every $\varepsilon > 0$ and $R > 0$,

$$\ln \mathcal{N}\left(\varepsilon, \overline{I_s\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq \left(\frac{RC}{\varepsilon}\right)^{\frac{d}{s}},$$

and this gives raise to the bound

$$\mathcal{N}^{-1}\left(i, \overline{I_s\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq RC\left(\ln(i+1)\right)^{-\frac{s}{d}}.$$

As mentioned in [52], this inequality can be extended to compact embeddings of Besov-type function spaces.

Example 2.3.4 (Reproducing kernel Hilbert spaces) *Let $\mathcal{X}$ be a compact domain in $\mathbb{R}^d$, and let $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a Mercer kernel, i.e., K is continuous, symmetric and positive definite. Recall that we say that K is positive definite if for all finite sets $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subset \mathcal{X}$, the $m \times m$ matrix $[K(\mathbf{x}_i, \mathbf{x}_j)]_{1 \leq i,j \leq m}$ is positive definite. Typical examples of Mercer kernels are the Gaussian kernel $K(\mathbf{x}, \mathbf{x}') = \exp(-\|\mathbf{x} - \mathbf{x}'\|^2)$ and the kernel $K(\mathbf{x}, \mathbf{x}') = (c^2 + \|\mathbf{x} - \mathbf{x}'\|^2)^{-\alpha}$ with $\alpha > 0$.*

For $\mathbf{x} \in \mathcal{X}$, let $K_{\mathbf{x}} = K(\mathbf{x}, .)$. According to Moore-Aronszajn's Theorem (Aronszajn [7]), there exists a unique Hilbert space $\mathcal{H}_K$ of functions on $\mathcal{X}$ satisfying the following conditions:

(i) For all $\mathbf{x} \in \mathcal{X}$, $K_{\mathbf{x}} \in \mathcal{H}_K$;

(ii) The span of the set $\{K_{\mathbf{x}} = K(\mathbf{x}, .),\ \mathbf{x} \in \mathcal{X}\}$ is dense in $\mathcal{H}_K$;

(iii) For all $f \in \mathcal{H}_K$, $f(\mathbf{x}) = \langle K_{\mathbf{x}}, f \rangle$.

The Hilbert space $\mathcal{H}_K$ is said to be the reproducing kernel Hilbert space (for short, RKHS) associated with the kernel K. It can be shown that $\mathcal{H}_K$ consists of continuous functions and, provided K is a $\mathcal{C}^\infty$ Mercer kernel, that the inclusion

$$I_K : (\mathcal{G}, \|.\|_{\mathcal{G}}) = (\mathcal{H}_K, \|.\|_K) \hookrightarrow (\mathcal{F}, \|.\|_{\mathcal{F}}) = (C(\mathcal{X}), \|.\|_\infty)$$

is a compact embedding (Cucker and Smale [37], Theorem D). Moreover, as proved in [37], for all $h > d$, $\varepsilon > 0$ and $R > 0$,

$$\ln \mathcal{N}\left(\varepsilon, \overline{I_K\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq \left(\frac{RC}{\varepsilon}\right)^{\frac{2d}{h}},$$

where C is a positive constant independent of ε and R. This readily implies that for $h > d$, $i \in \mathbb{N}^\star$ and $R > 0$,

$$\mathcal{N}^{-1}\left(i, \overline{I_K\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq RC\left(\ln(i+1)\right)^{-\frac{h}{2d}}.$$

This result has been improved by Zhou [116], who studies convolution-type kernels on $[0,1]^d$*, i.e., kernels of form* $K(\mathbf{x},\mathbf{x}') = k(\mathbf{x}'-\mathbf{x})$*. Zhou provides estimates of* $\ln \mathcal{N}(\varepsilon, \overline{I_K(\mathcal{B}_{\mathcal{G}}(R))})$ *depending on the decay of* $\hat{k}$*, the Fourier transform of* k*. For example, when* $\hat{k}$ *decays exponentially, one has*

$$\ln \mathcal{N}\left(\varepsilon, \overline{I_K\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq C\left(\ln \frac{R}{\varepsilon}\right)^{d+1},$$

where C *depends only on the kernel and the dimension. This implies*

$$\mathcal{N}^{-1}\left(i, \overline{I_K\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right) \leq R \exp\left\{-\left(\frac{\ln(i+1)}{C}\right)^{\frac{1}{d+1}}\right\}.$$

This result can typically be applied to the Gaussian kernel.

Motivated by Examples 2.3.1-2.3.4 above, we shall impose the following set of assumptions on the distribution μ of $\mathbf{X}$:

- *A*1 There exists a subspace $(\mathcal{G}, \|.\|_{\mathcal{G}})$ of $(\mathcal{F}, \|.\|_{\mathcal{F}})$ such that the support $\mathcal{S}(\mu)$ is bounded in $(\mathcal{G}, \|.\|_{\mathcal{G}})$, that is $\mathcal{S}(\mu) \subset \mathcal{B}_{\mathcal{G}}(R)$ for some positive constant R.
- *A*2 There exists a compact embedding

$$I : (\mathcal{G}, \|.\|_{\mathcal{G}}) \hookrightarrow (\mathcal{F}, \|.\|_{\mathcal{F}}).$$

- *A*3 There exists a function $\phi :]0,\infty[\to]0,\infty[$ such that

$$\left[\mathcal{N}^{-1}\left(i, \overline{I\left(\mathcal{B}_{\mathcal{G}}(R)\right)}\right)\right]^2 \leq \phi\left(\ln(i+1)\right), \quad i \in \mathbb{N}^\star,$$

  where the covering number is taken with respect to $\|.\|_{\mathcal{F}}$, and ϕ satisfies the following conditions

  (*i*) ϕ is nonincreasing and $\lim_{t\to\infty} t\phi(\ln t) = \infty$;

  (*ii*) ϕ is differentiable on $]0,\infty[$ and $\dfrac{\phi'(u)}{\phi(u)} \to 0$ as $u \to \infty$;

  (*iii*) One has $\displaystyle\int_1^\infty \phi(\ln t)\,dt = \infty$.

The boundedness condition in assumption *A*1 is standard when establishing rates of convergence of nonparametric estimates, see e.g. Györfi, Kohler, Krzyżak and Walk [66]. As noticed in Theorem 7 of Kulkarni and Posner [80], this condition can be slightly relaxed, at the price of obtaining rates of convergence in probability.

Assumption *A*2 means that the balls in $\mathcal{G}$ (with respect to $\|.\|_{\mathcal{G}}$) are totally bounded as subsets of the space $(\mathcal{F}, \|.\|_{\mathcal{F}})$. This condition is not restrictive, and it is in particular satisfied by our leading Examples 2.3.1-2.3.4. From a practical perspective, we wish to emphasize that one usually has some latitude in choosing the space $\mathcal{G}$. This choice will typically be based on the regularity of the data (curves) to be processed. Roughly speaking, the smoother they are, the "smaller" the support of μ, and therefore the faster the convergence. On the other hand, we note that the Lipschitz condition in (2.2) needs to be valid in $(\mathcal{F}, \|.\|_{\mathcal{F}})$ — typically in $(\mathcal{C}(\mathcal{X}), \|.\|_\infty)$ — which is a stronger requirement than a Lipschitz condition in $(\mathcal{G}, \|.\|_{\mathcal{G}})$. To overcome this difficulty, we may decide to choose a "smaller" space $\mathcal{F}$, where the Lipschitz condition will be easier fulfilled. However, this operation may lead to slower rates of convergence, since they essentially depend

on the difference of regularity (on the difference of "size" in some sense) between $(\mathcal{F}, \|.\|_{\mathcal{F}})$ and $(\mathcal{G}, \|.\|_{\mathcal{G}})$, as enlightened by Example 2.3.2.

Finally, a quick inspection shows that the requirement $A3$ is verified for the presented examples 2.3.1-2.3.4. We will specify what is ϕ on each example in the following, but let us first give the main result of this chapter.

Theorem 3 *Suppose that assumptions A1-A3 are satisfied. Suppose in addition that, for all* $\mathbf{x}$ *and* $\mathbf{x}' \in \mathcal{F}$,

$$\sigma^2(\mathbf{x}) = \mathrm{Var}[Y \,|\, \mathbf{X} = \mathbf{x}] \leq \sigma^2$$

and

$$\left|r(\mathbf{x}) - r(\mathbf{x}')\right| \leq L\|\mathbf{x} - \mathbf{x}'\|_{\mathcal{F}},$$

for some positive constants σ^2 *and* L. *Then*

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \lesssim \frac{\sigma^2}{k_n} + L^2\phi\left(\ln\left\lfloor\frac{n}{k_n}\right\rfloor\right).$$

Theorem 3 can be illustrated in light of Examples 2.3.1-2.3.4. For differentiable functions (Example 2.3.1), we have $\phi(t) \propto t^{-2m/d}$, and the result reads

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \lesssim \frac{\sigma^2}{k_n} + L^2\left(\ln\left\lfloor\frac{n}{k_n}\right\rfloor\right)^{-\frac{2m}{d}}.$$

Therefore, with the choice $k_n \propto (\ln n)^{\frac{2m}{d}}$,

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \lesssim (\ln n)^{-\frac{2m}{d}}.$$

Similarly, in Sobolev spaces (Example 2.3.2), the choice $k_n \propto (\ln n)^{\frac{2(s_1-s_2)}{d}}$ leads to

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \lesssim (\ln n)^{-\frac{2(s_1-s_2)}{d}}.$$

In Besov spaces (Example 2.3.3), $\phi(t) \propto t^{-2s/d}$ and, with $k_n \propto (\ln n)^{\frac{2s}{d}}$, we obtain

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \lesssim (\ln n)^{-\frac{2s}{d}}.$$

Finally, in reproducing kernel Hilbert spaces (Example 2.3.4), attention shows that the choice $k_n \propto (\ln n)^{-\frac{h}{d}}$ results in

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \lesssim (\ln n)^{-\frac{h}{d}}.$$

For convolution-type kernels (Zhou [116]), the choice $k_n \propto \exp\left\{2\left(\frac{\ln n}{C}\right)^{\frac{1}{d+1}}\right\}$ implies

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \lesssim \exp\left\{-2\left(\frac{\ln n}{C}\right)^{\frac{1}{d+1}}\right\}.$$

The general finding here is that these rates of convergence are much slower than the traditional finite-dimensional rates (see Example 2.3.1). On the other hand, to the best of our knowledge, they are the first explicit available rates for the functional k_n-NN estimate. It is an open problem to know whether these rates are optimal over the smoothness classes we consider.

Chapter 3

On the Rate of Convergence of the Bagged Nearest Neighbor Estimate

3.1 Introduction

3.1.1 Bagging

Ensemble methods are popular machine learning algorithms which train multiple learners and combine their predictions. The success of ensemble algorithms on many benchmark data sets has raised considerable interest in understanding why such methods succeed and identifying circumstances in which they can be expected to produce good results. It is now well known that the generalization ability of an ensemble can be significantly better than that of a single predictor, and ensemble learning has therefore been a hot topic during the past years. For a comprehensive review of the domain, we refer the reader to Dietterich [47] and the references therein.

One of the first and simplest ways to combine predictors in order to improve their performance is bagging (**b**ootstrap **agg**regat**ing**), suggested by Breiman [18]. This ensemble method proceeds by generating subsamples from the original data set, constructing a predictor from each resample, and decide by combining. It is one of the most effective computationally intensive procedures to improve on unstable estimates or classifiers, especially for large, high dimensional data set problems where finding a good model in one step is impossible because of the complexity and scale of the problem. Bagging has attracted much attention and is frequently applied, although its statistical mechanisms are not yet fully understood and are still under active investigation. Recent theoretical contributions to bagging and related methodologies include those of Friedman and Hall [60], Bühlmann and Yu [21], Hall and Samworth [68], Buja and Stuetzle [22], and Biau and Devroye [14].

It turns out that Breiman's bagging principle has a simple application in the context of nearest neighbor methods. Nearest neighbor predictors are one of the oldest approaches to regression and classification (Fix and Hodges [57, 58], Cover and Hart [36], Cover [34, 35], Györfi [65], Venkatesh, Snapp and Psaltis [115], Psaltis, Snapp and Venkatesh [95]). Before we formalize the link between bagging and nearest neighbors, some definitions are in order. Throughout the paper, we suppose that we are given a sample $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ of i.i.d. $\mathbb{R}^d \times \mathbb{R}$-valued random variables with the same distribution as a generic pair $(\mathbf{X}, Y)$ satisfying $\mathbb{E}Y^2 < \infty$. The space $\mathbb{R}^d$ is equipped with the standard Euclidean metric. For fixed $\mathbf{x} \in \mathbb{R}^d$, our mission is to estimate the regression function $r(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$ using the data $\mathcal{D}_n$. With this respect, we say that a regression function estimate $r_n(\mathbf{x})$ is consistent if $\mathbb{E}[r_n(\mathbf{X}) - r(\mathbf{X})]^2 \to 0$ as $n \to \infty$. It is universally consistent if this property is true for all distributions of $(\mathbf{X}, Y)$ with $\mathbb{E}Y^2 < \infty$.

3.1.2 Bagging and Nearest Neighbors

Recall that the 1-nearest neighbor (1-NN) regression estimate sets $r_n(\mathbf{x}) = Y_{(1)}(\mathbf{x})$ where $Y_{(1)}(\mathbf{x})$ is the observation of the feature vector $\mathbf{X}_{(1)}(\mathbf{x})$ whose Euclidean distance to $\mathbf{x}$ is minimal among all $\mathbf{X}_1, \ldots, \mathbf{X}_n$. Ties are broken in favor of smallest indices. It is clearly not, in general, a consistent estimate (Devroye, Györfi and Lugosi [46], Chapter 5). However, by bagging, one may turn the 1-NN estimate into a consistent one, provided that the size of the resamples is sufficiently small.

We proceed as follows, via a randomized basic regression estimate r_{k_n} in which $1 \leq k_n \leq n$ is a parameter. The elementary predictor r_{k_n} is the 1-NN rule for a random subsample $\mathcal{S}_n$ drawn with (or without) replacement from $\{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$, with $\mathrm{Card}(\mathcal{S}_n) = k_n$. We apply bagging, that is, we repeat the random sampling an infinite number of times, and take the average of the individual outcomes. Thus, the bagged regression estimate $r_n^\star$ is defined by

$$r_n^\star(\mathbf{x}) = \mathbb{E}^\star \left[r_{k_n}(\mathbf{x}) \right],$$

where $\mathbb{E}^\star$ denotes expectation with respect to the resampling distribution, conditionally on the data set $\mathcal{D}_n$.

The following result, proved by Biau and Devroye [14], shows that for an appropriate choice of k_n, the bagged version of the 1-NN regression estimate is universally consistent:

Theorem 4 *If $k_n \to \infty$ and $k_n/n \to 0$, then $r_n^\star$ is universally consistent.*

In this theorem, the fact that resampling is done with or without replacement is irrelevant. Thus, by bagging, one may turn the crude 1-NN procedure into a consistent one, provided that the size of the resamples is sufficiently small. To understand the statistical forces driving Theorem 4, recall that if we let $V_1 \geq V_2 \geq \ldots \geq V_n \geq 0$ denote deterministic weights that sum to one, then the regression estimate

$$\sum_{i=1}^{n} V_i \, Y_{(i)}(\mathbf{x})$$

is called a weighted nearest neighbor regression estimate. It is known to be universally consistent provided $V_1 \to 0$ and $\sum_{i>\varepsilon n} V_i \to 0$ for all $\varepsilon > 0$ as $n \to \infty$ (Stone [110], Devroye [45], and Problems 11.7, 11.8 of Devroye, Györfi and Lugosi [46]). The crux to prove Theorem 4 is to observe that $r_n^\star$ is in fact a weighted nearest neighbor estimate with

$$V_i = \mathbb{P}(i\text{-th nearest neighbor of } \mathbf{x} \text{ is the 1-NN in a random selection}).$$

Then, a moment's thought shows that for the "with replacement" sampling

$$V_i = \left(1 - \frac{i-1}{n}\right)^{k_n} - \left(1 - \frac{i}{n}\right)^{k_n},$$

whereas for sampling "without replacement", V_i is hypergeometric:

$$V_i = \begin{cases} \dfrac{\binom{n-i}{k_n-1}}{\binom{n}{k_n}}, & i \leq n - k_n + 1 \\ 0, & i > n - k_n + 1. \end{cases}$$

The core of the proof of Theorem 4 is then to show that, in both cases, the weights V_i satisfy the conditions $V_1 \to 0$ and $\sum_{i>\varepsilon n} V_i \to 0$ for all $\varepsilon > 0$ as $n \to \infty$. These weights have been

independently exhibited by Steele [109], who also shows on practical examples that substantial reductions in prediction error are possible by bagging the 1-NN estimate. Note also that this new expression for the 1-NN bagged estimate makes any Monte Carlo approach unnecessary to evaluate the estimate. Indeed, up to now, this predictor was implemented by Monte Carlo, i.e., by repeating the random sampling T times, and taking the average of the individual outcomes. Formally, if $Z_t = r_{k_n}(\mathbf{x})$ is the prediction in the t-th round of bagging, the bagged regression estimate was approximately evaluated as

$$r_n^\star(\mathbf{x}) \approx \frac{1}{T}\sum_{t=1}^{T} Z_t,$$

where $Z_1, \ldots, Z_T$ are the outcomes in the individual rounds. Clearly, writing the 1-NN bagged estimate as an (exact) weighted nearest neighbor predictor makes such calculations useless.

On the other hand, the fact that the bagged 1-NN estimate reduces to a weighted nearest neighbor estimate may seem at first sight somehow disappointing. Indeed, we get the ordinary k_n-NN rule back by the choice

$$V_i = \begin{cases} 1/k_n & \text{if } i \leq k_n \\ 0 & \text{otherwise,} \end{cases}$$

and, with an appropriate choice of the sequence (k_n), this regression estimate is known to have optimal asymptotic properties (see Chapter 6 in Györfi, Kohler, Krzyżak and Walk [66] and the references therein). Thus, the question is: Why would one care about the bagged nearest neighbor rule then? The answer is twofold. First, bagging the 1-NN is a very popular technique for regression and classification in the machine learning community, and most — if not all — empirical studies report practical improvements over the traditional k_n-NN method. Secondly (and most importantly), analysing 1-NN bagging is part of a larger project trying to understand the driving forces behind the random forests estimates, which were defined by Breiman in [19]. In short, random forests are some of the most successful ensemble methods that exhibit performance on the level of boosting and support vector machines. These learning procedures typically involve a resampling step, which may be interpreted as a particular 1-NN bagged procedure based on the so-called "layered nearest neighbor" proximities (Lin and Jeon [85], Biau and Devroye [14]).

Thus, in the present chapter, we go one step further in bagging investigation and study the rate of convergence of $\mathbb{E}\left[r_n^\star(\mathbf{X}) - r(\mathbf{X})\right]^2$ to 0 as $n \to \infty$. We will start our analysis by stating a comprehensive theorem on the rate of convergence of general weighted nearest neighbor estimates (section 3.2.1). Then, this result will be particularized to 1-NN bagging, focusing on the "with replacement" case (section 3.2.2).

Throughout the document, we will be interested in rate of convergence results for the class $\mathcal{F}$ of $(1, C, \rho, \sigma^2)$-smooth distributions $(\mathbf{X}, Y)$ such that $\mathbf{X}$ has compact support with diameter 2ρ, the regression function r is Lipschitz with constant C and, for all $\mathbf{x} \in \mathbb{R}^d$, $\sigma^2(\mathbf{x}) = \mathrm{Var}[Y \,|\, \mathbf{X} = \mathbf{x}] \leq \sigma^2$. It is known (see for example Ibragimov and Khasminskii [72, 73, 74]) that for the class $\mathcal{F}$, the sequence $(n^{-\frac{2}{d+2}})$ is the optimal minimax rate of convergence. In particular,

$$\liminf_{n\to\infty} \inf_{r_n} \sup_{(\mathbf{X},Y)\in\mathcal{F}} \frac{\mathbb{E}[r_n(\mathbf{X}) - r(\mathbf{X})]^2}{\left((\rho C)^d \sigma^2\right)^{\frac{2}{d+2}} n^{-\frac{2}{d+2}}} \geq \Delta$$

for some positive constant Δ independent of C, ρ and σ^2. Here the infimum is taken over all estimates r_n, i.e., over all square integrable measurable functions of the data. As a striking result, we prove that for $d \geq 3$ and a suitable choice of the sequence (k_n), the estimate $r_n^\star$ is of optimum

rate for the class $\mathcal{F}$, that is

$$\limsup_{n\to\infty} \sup_{(\mathbf{X},Y)\in\mathcal{F}} \frac{\mathbb{E}[r_n^\star(\mathbf{X}) - r(\mathbf{X})]^2}{((\rho C)^d \sigma^2)^{\frac{2}{d+2}}\, n^{-\frac{2}{d+2}}} \leq \Lambda$$

for some positive Λ independent of C, ρ and σ^2. Since the parameter k_n of the estimate with the optimal rate of convergence depends on the unknown distribution of $(\mathbf{X}, Y)$, especially on the smoothness of the regression function, we present in section 3.2.3 adaptive (i.e., data-dependent) choices of k_n which preserve the minimax optimality of the estimate.

We wish to emphasize that all the results are obtained by letting the resampling size k_n grows with n in such a manner that $k_n \to \infty$ and $k_n/n \to 0$. These results are of interest because the majority of bagging experiments employ relatively large resample sizes. In fact, much of the evidence *against* the performance of bagged nearest neighbor methods is for full sample size resamples (see the discussion in Breiman [18], Paragraph 6.4), except the notable results of Hall and Samworth [68] and Steele [109], who also report encouraging numerical results in the context of regression and classification.

3.2 Rates of Convergence

3.2.1 Weighted Nearest Neighbor Estimates

As an appetizer, we start our analysis of the 1-NN bagged regression estimate from a larger point of view, by offering a general theorem on the rate of convergence of weighted nearest neighbor estimates, i.e., estimates of the form

$$r_n(\mathbf{x}) = \sum_{i=1}^{n} V_i\, Y_{(i)}(\mathbf{x})$$

with nonnegative weights satisfying the constraints $\sum_{i=1}^n V_i = 1$ and $V_1 \geq V_2 \geq \ldots \geq V_n \geq 0$. As in Chapter 2, we will make use of the well-known notions of covering numbers and covering radii which characterize the massiveness of a set (Kolmogorov and Tihomirov [79]). As put forward in Kulkarni and Posner [80], these quantities play a key role in the context of nearest neighbor analysis.

Throughout the paper, we will denote by $\mathcal{B}(\mathbf{x}, \varepsilon)$ the open Euclidean ball in $\mathbb{R}^d$ centered at $\mathbf{x}$ of radius ε. As usual in this context, μ will stand for the distribution of $\mathbf{X}$, which will be assumed to be a bounded random variable. Recall that the support $\mathcal{S}(\mu)$ of μ is defined as the collection of all $\mathbf{x}$ with $\mu(\mathcal{B}(\mathbf{x}, \varepsilon)) > 0$ for all $\varepsilon > 0$. Letting $\rho = \mathcal{N}^{-1}(1, \mathcal{S}(\mu))$, we observe that 2ρ is an upper bound on the diameter of $\mathcal{S}(\mu)$. We are now in a position to state the main result of this subsection. We let the symbol $\lfloor . \rfloor$ denote the integer part function.

Theorem 5 *Let $r_n(\mathbf{x}) = \sum_{i=1}^n V_i\, Y_{(i)}(\mathbf{x})$ be a weighted nearest neighbor estimate of $r(\mathbf{x})$. Suppose that $\mathbf{X}$ is bounded, and set $\rho = \mathcal{N}^{-1}(1, \mathcal{S}(\mu))$. Suppose in addition that, for all $\mathbf{x}$ and $\mathbf{x}' \in \mathbb{R}^d$,*

$$\sigma^2(\mathbf{x}) = \mathrm{Var}[Y \,|\, \mathbf{X} = \mathbf{x}] \leq \sigma^2$$

and

$$\big|r(\mathbf{x}) - r(\mathbf{x}')\big| \leq C\|\mathbf{x} - \mathbf{x}'\|,$$

for some positive constants σ^2 and C. Then, if $d \geq 3$,

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \leq \sigma^2 \sum_{i=1}^{n} V_i^2 + \frac{8\rho^2 C^2}{1 - 2/d} \sum_{i=1}^{n} V_i \left\lfloor \frac{n}{i} \right\rfloor^{-2/d}.$$

Sketch of the Proof of Theorem 5 Setting

$$\tilde{r}_n(\mathbf{x}) = \sum_{i=1}^{n} V_i \, r(\mathbf{X}_{(i)}(\mathbf{x})),$$

the proof of Theorem 5 will rely on the variance/bias decomposition

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 = \mathbb{E}\left[r_n(\mathbf{X}) - \tilde{r}_n(\mathbf{X})\right]^2 + \mathbb{E}\left[\tilde{r}_n(\mathbf{X}) - r(\mathbf{X})\right]^2. \tag{3.1}$$

The first term is easily bounded by noting that, for all $\mathbf{x} \in \mathbb{R}^d$,

$$\mathbb{E}\left[r_n(\mathbf{x}) - \tilde{r}_n(\mathbf{x})\right]^2 \leq \sigma^2 \sum_{i=1}^{n} V_i^2. \tag{3.2}$$

To analyse the bias term in (3.1), we will need the following result, which bounds the convergence rate of the expected i-th nearest neighbor squared distance in terms of the metric covering radii of the support of the distribution μ of $\mathbf{X}$. Proposition 4 is a generalization of Theorem 1, page 1032 in Kulkarni and Posner [80], which only reports results for the rate of convergence of *the* nearest neighbor. Therefore, this result is interesting by itself.

Proposition 4 *Suppose that* $\mathbf{X}$ *is bounded. Then*

$$\mathbb{E}\|\mathbf{X}_{(i)}(\mathbf{X}) - \mathbf{X}\|^2 \leq \frac{8i}{n} \sum_{j=1}^{\lfloor n/i \rfloor} \left[\mathcal{N}^{-1}\left(j, \mathcal{S}(\mu)\right)\right]^2.$$

For any bounded set $\mathcal{A}$ in the Euclidean d-space, the covering radius satisfies $\mathcal{N}^{-1}(r, \mathcal{A}) \leq \mathcal{N}^{-1}(1, \mathcal{A}) r^{-1/d}$ (see [79]). Hence the following corollary:

Corollary 2 *Suppose that* $\mathbf{X}$ *is bounded, and set* $\rho = \mathcal{N}^{-1}(1, \mathcal{S}(\mu))$. *Then if* $d \geq 3$,

$$\mathbb{E}\|\mathbf{X}_{(i)}(\mathbf{X}) - \mathbf{X}\|^2 \leq \frac{8\rho^2 \lfloor n/i \rfloor^{-\frac{2}{d}}}{1 - 2/d}.$$

Thus, to prove Theorem 5, it suffices to note from (3.1) and (3.2) that

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \leq \sigma^2 \sum_{i=1}^{n} V_i^2 + \mathbb{E}\left[\tilde{r}_n(\mathbf{X}) - r(\mathbf{X})\right]^2.$$

Next, applying Jensen's inequality and integrating with respect to the distribution of $\mathbf{X}$, we obtain

$$\mathbb{E}\left[\tilde{r}_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \leq C^2 \left[\sum_{i=1}^{n} V_i \, \mathbb{E}\|\mathbf{X}_{(i)}(\mathbf{X}) - \mathbf{X}\|^2\right],$$

and the conclusion follows by applying Corollary 2. ■

Theorem 5 offers a general result, which can be made more precise according to the weights definition. Taking for example

$$V_i = \begin{cases} 1/k_n & \text{if } i \leq k_n \\ 0 & \text{otherwise,} \end{cases}$$

we get the ordinary k_n-NN rule back. In this context, according to Theorem 5, for $d \geq 3$, there exists a sequence (k_n) with $k_n \propto n^{\frac{2}{d+2}}$ such that

$$\mathbb{E}\left[r_n(\mathbf{X}) - r(\mathbf{X})\right]^2 \leq \Lambda \left(\frac{(\rho C)^d \sigma^2}{n}\right)^{\frac{2}{d+2}},$$

for some positive constant Λ independent of ρ, C and σ^2. This is exactly Theorem 6.2, page 93 of Györfi, Kohler, Krzyżak and Walk [66], which states that the standard nearest neighbor estimate is of optimum rate for the class $\mathcal{F}$ of $(1, C, \rho, \sigma^2)$-smooth distributions $(\mathbf{X}, Y)$ such that $\mathbf{X}$ has compact support with covering radius ρ, the regression function r is Lipschitz with constant C and, for all $\mathbf{x} \in \mathbb{R}^d$, $\sigma^2(\mathbf{x}) = \mathrm{Var}[Y \,|\, \mathbf{X} = \mathbf{x}] \leq \sigma^2$.

The adaptation of Theorem 5 to the 1-NN bagged regression estimate needs more careful attention. This will be the topic of the next section.

3.2.2 Bagging

In the following, we will focus on bagging with replacement when the dimension d is equal to or greater than 3. Comparable results have been obtained for $d \leq 2$ and for bagging without replacement (see Biau, Cérou and Guyader [12]).

Bagging with replacement is sometimes called moon-bagging, standing for **m** o**u**t **o**f **n** **b**ootstrap **agg**regat**ing**. As seen in the introduction, in this case, the weighted nearest neighbor regression estimate takes the form

$$r_n^\star(\mathbf{x}) = \sum_{i=1}^{n} V_i \, Y_{(i)}(\mathbf{x}),$$

where

$$V_i = \left(1 - \frac{i-1}{n}\right)^{k_n} - \left(1 - \frac{i}{n}\right)^{k_n}.$$

Under the same assumptions as in Theorem 3.2.1, we obtain

Theorem 6 *If $d \geq 3$, there exists a sequence (k_n) with $k_n \propto n^{\frac{d}{d+2}}$ such that*

$$\mathbb{E}\left[r_n^\star(\mathbf{X}) - r(\mathbf{X})\right]^2 \leq \Lambda \left(\frac{(\rho C)^d \sigma^2}{n}\right)^{\frac{2}{d+2}},$$

for some positive constant Λ independent of ρ, C and σ^2.

Two important remarks are in order.

1. First, we note that, for a suitable choice of k_n, the bagged 1-NN estimate achieves both the minimax $n^{-2/(d+2)}$ rate and the optimal order of magnitude $((\rho C)^d \sigma^2)^{2/(d+2)}$ in the constant, for the class $\mathcal{F}$ of $(1, C, \rho, \sigma^2)$-smooth distributions $(\mathbf{X}, Y)$ such that $\mathbf{X}$ has compact support with covering radius ρ, the regression function r is Lipschitz with constant C and, for all $\mathbf{x} \in \mathbb{R}^d$, $\sigma^2(\mathbf{x}) = \mathrm{Var}[Y \,|\, \mathbf{X} = \mathbf{x}] \leq \sigma^2$. Seconds, the bound is valid for finite sample sizes, so that we are in fact able to approach the minimax lower bound not only asymptotically but even for finite sample sizes. On the other hand, the estimate with the optimal rate of convergence depends on the unknown distribution of $(\mathbf{X}, Y)$, and especially on the covering radius ρ and the smoothness of the regression function measured by the constant C. It is to correct this situation that we present adaptation results in section 3.2.3.

2. We have also obtained convergence rates in the cases $d = 1$ and $d = 2$ (see [12]). It turns out that, for $d = 1$, the obtained rate is not optimal, whereas it is optimal up to a log term for $d = 2$. This low-dimensional phenomenon is also known to hold for the traditional k_n-NN regression estimate, which does not achieve the optimal rates in dimensions 1 and 2 (see Problems 6.1 and 6.7 in [66], Chapter 3).

3.2.3 Adaptation

In the previous subsections, the parameter k_n of the estimate with the optimal rate of convergence for the class $\mathcal{F}$ depends on the unknown distribution of $(\mathbf{X}, Y)$, especially on the smoothness of the regression function measured by the Lipschitz constant C. In this subsection, we present a data-dependent way of choosing the resampling size k_n and show that, for bounded Y, the estimate with parameter chosen in such an adaptive way achieves the optimal rate of convergence (irrespectively of the resampling type). To this aim, we split the sample $\mathcal{D}_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ in two parts of size $\lfloor n/2 \rfloor$ and $n - \lfloor n/2 \rfloor$, respectively (assuming $n \geq 2$). The first half is denoted by $\mathcal{D}_n^\ell$ (learning set) and is used to construct the bagged 1-NN estimate $r^\star_{\lfloor n/2 \rfloor}(\mathbf{x}, \mathcal{D}_n^\ell) = r^\star_{k,\lfloor n/2 \rfloor}(\mathbf{x}, \mathcal{D}_n^\ell)$ (for the sake of clarity, we make the dependence of the estimate upon k explicit). The second half of the sample, denoted by $\mathcal{D}_n^t$ (testing set), is used to choose k by picking $\hat{k}_n \in K = \{1, \ldots, \lfloor n/2 \rfloor\}$ to minimize the empirical risk

$$\frac{1}{n - \lfloor n/2 \rfloor} \sum_{i=\lfloor n/2 \rfloor + 1}^{n} \left(Y_i - r^\star_{k,\lfloor n/2 \rfloor}(\mathbf{X}_i) \right)^2 .$$

Define the estimate

$$r_n^\star(\mathbf{x}) = r^\star_{\hat{k}_n,\lfloor n/2 \rfloor}(\mathbf{x}, \mathcal{D}_n^\ell),$$

and note that $r_n^\star$ depends on the entire data $\mathcal{D}_n$. If $|Y| \leq L < \infty$ almost surely, a straightforward adaptation of Theorem 7.1 in [66] shows that, for any $\delta > 0$,

$$\mathbb{E}[r_n^\star(\mathbf{X}) - r(\mathbf{X})]^2 \leq (1 + \delta) \inf_{k \in K} \mathbb{E}[r^\star_{k,\lfloor n/2 \rfloor}(\mathbf{X}) - r(\mathbf{X})]^2 + \Xi \frac{\ln n}{n},$$

for some positive constant Ξ depending only on L, d and δ. Thus we can conclude:

Theorem 7 *Suppose that $|Y| \leq L$ almost surely, and let $r_n^\star$ be the bagged 1-NN estimate with $k \in K = \{1, \ldots, \lfloor n/2 \rfloor\}$ chosen by data-splitting, irrespectively of the resampling type. Then $(\ln n)^{(d+2)/(2d)} n^{-1/2} \leq \rho C$ together with $d \geq 3$ implies, for $n \geq 2$,*

$$\mathbb{E}[r_n^\star(\mathbf{X}) - r(\mathbf{X})]^2 \leq (\Lambda + \mathrm{o}(1)) \left(\frac{(\rho C)^d}{n} \right)^{\frac{2}{d+2}},$$

for some positive constant Λ which depends only on L and d.

Thus, the expected error of the estimate obtained via data-splitting is bounded from above up to a constant by the corresponding minimax lower bound for the class $\mathcal{F}$ of regression functions, with the optimal dependence in C and ρ.

Part II

Rare Event Simulation and Estimation

Chapter 4

Methodology

4.1 Introduction

Monte Carlo approach is a common tool to estimate the expectation of any function of a random object when analytical or numerical methods are not available. Here, typically, we want to estimate precisely and in a reasonable time the small probability, say 10^{-9} or below, of an extreme event. Formally, suppose X is random vector in $\mathbb{R}^d$ with law μ that we can simulate, and Φ is a mapping from $\mathbb{R}^d$ to $\mathbb{R}$, also called a score function. Because of the complexity of the underlying process, we view Φ as black box, that is, we do not have an analytic expression for Φ but we can readily evaluate $\Phi(X)$ for any given instance X. Then given a threshold q which lies far out in the right hand tail of the distribution of $\Phi(X)$, we seek to estimate the very low probability $p = \mathbb{P}(\Phi(X) > q)$.

A Crude Monte Carlo (CMC) that uses an i.i.d. N-sample $X_1, \ldots, X_N$ to estimate p by the fraction $\hat{p}_{mc} = \#\{i : \ \Phi(X_i) > q\}/N$ is not practical when p is very small. Indeed, in order to obtain a reasonable precision of the estimate given by the relative variance $\mathrm{Var}(\hat{p}_{mc})/p^2$, which is equal to $(1-p)/(Np)$, one needs to select a sample size N of order p^{-1}. For instance, a random sample of one billion observations is needed to estimate a target probability of 10^{-9}.

Importance sampling, which draws samples according to π and weights each observation $X = x$ by $w(x) = d\mu(x)/d\pi(x)$ can decrease the variance of the estimated probability which in turn greatly reduces the need for such large sample sizes. We refer to Robert and Casella [98] for a discussion on variance reduction techniques in general and to Bucklew [20] for the application of importance sampling in the context of rare events estimation. Unfortunately, when Φ is a black box, these weights cannot be computed, and hence importance sampling is not available to us.

Multilevel splitting, also called *Importance splitting*, introduced by Kahn and Harris [76] and Rosenbluth and Rosenbluth [100], is another powerful algorithm for rare events simulations. The basic idea of multilevel splitting, adapted to our problem, is to fix a set of increasing levels $-\infty = L_0 < L_1 < L_2 < \cdots < L_n = q$, and to decompose the tail probability

$$\mathbb{P}(\Phi(X) > q) = \prod_{k=0}^{n-1} \mathbb{P}(\Phi(X) > L_{k+1} | \Phi(X) > L_k).$$

Each conditional probability $p_k = \mathbb{P}(\Phi(X) > L_{k+1} | \Phi(X) > L_k)$ is then estimated separately. We refer the reader to the paper by Glasserman, Heidelberger, Shahabuddin and Zajic [62] for an in-depth review of the multilevel splitting method and a detailed list of references. Two practical issues associated with the implementation of multilevel splitting are the need for computationally

efficient algorithms for estimating the successive conditional probabilities, and the optimal selection of the sequence of levels.

Recently Cérou, Del Moral, Le Gland and Lezaud [25] bridged multilevel splitting for Markovian processes and particle methods for Feynman-Kac models, thus introducing a rigorous mathematical framework for linking the sample used to estimate p_j to the one needed to estimate p_{j+1}. Within the context of Markovian processes, Cérou and Guyader [27] proposed an algorithm to adaptively select the levels in an optimal way.

To our knowledge, the first instance in which static rare event simulation using splitting was proposed is a paper by Au and Beck [8] (see also Au and Beck [9]). But these authors call it "Subset Simulation" and do not make any connection with multilevel splitting, which is why people in the rare event community do not mention this work afterwards. The next work where a reversible transition kernel was introduced to deal with such static rare events is due to Del Moral, Doucet and Jasra [41] (see also Johansen, Del Moral and Doucet [75]). However, these articles were written in a different framework, and thus do not deal with the practical details of our precise setting. In the present chapter, we detail a fixed and two adaptive multilevel algorithms. Given a fixed probability of success p_0 at each step, for example $p_0 = 0.75$, the first adaptive one consists in optimally placing the levels on the fly. The second adaptive algorithm goes one step further and minimizes the estimator variance, by taking $p_0 = 1 - 1/N$, where N is the number of particles.

Recently and independently, Botev and Kroese [16] proposed the same approach as in the first adaptive algorithm. These authors work on a similar algorithm, including the use of quantiles of the random variable $\Phi(X)$ on the swarm of particles in order to estimate the next level. The main difference is their two stage procedure (like in Garvels [61]): they first run the algorithm just to compute the levels, and then they restart from the beginning with these proposed levels. Actually we prove that by computing the levels on the fly, i.e. within the same run as the one to compute the rare event probability, we only pay a small bias on the estimate. Note also that [16] does not address the general construction of the transition kernels M_k, since the authors only tackle examples where they can derive a Gibbs sampler at each step. This is mainly possible because their function Φ is linear, which is a severe restriction.

Another related approach is the recent work on combinatorial counting of Rubinstein [101]. This article presents some optimizations for counting problems in which X has a uniform distribution over a discrete but very large state space. The author uses what he calls a cloning procedure, where the number of offspring is fixed (*i.e.* the same for all the particles in the sample) but adaptive to keep the number roughly constant, while removing redundant particles after the MCMC step. This is a main difference since we use a resampling with replacement procedure. But clearly results in [101] show that the adaptive procedure is well suited for SAT problems, or other hard finite set optimization problems. We would also like to mention that these last two papers [16][101] have demonstrated the performance of their algorithms via an extensive simulation study, to which we now lay out the mathematical foundations.

4.2 The Fixed-Levels Method

4.2.1 Assumptions and Ingredients

We assume that X is a random vector on $\mathbb{R}^d$ for some $d > 0$, and denote by μ its probability distribution on the underlying probability space $(\Omega, \mathcal{F}, \mathbb{P})$. We denote by A the rare set of interest,

and we assume that $A = \{x \in \mathbb{R}^d \text{ s.t. } \Phi(x) > q\}$ for some function $\Phi : \mathbb{R}^d \mapsto \mathbb{R}$ and some real number q. We also assume that we know how to draw i.i.d. samples from μ.

Our algorithm makes use of the following ingredients. An increasing sequence $\{L_0, \dots, L_n\}$ in $\overline{\mathbb{R}}$, with $L_0 = -\infty$ and $L_n = q$ defines a sequence of corresponding sets $A_k = \{x \in \mathbb{R}^d,\ \Phi(x) > L_k\}$. These sets are thus nested: $\mathbb{R}^d = A_0 \supset A_1 \supset \cdots \supset A_n = A$. We now need to choose sequence $\{L_0, \dots, L_n\}$ in such a way that $p_k = \mathbb{P}(X \in A_{k+1} | X \in A_k)$ is not too small. We also need to choose a Markov transition kernel K on $\mathbb{R}^d$ which is μ-symmetric, that is

$$\forall (x, y) \in \mathbb{R}^d \times \mathbb{R}^d,\ \mu(dx)K(x, dy) = \mu(dy)K(y, dx).$$

As a consequence, K has μ as an invariant measure.

As we will see in the sequel, the choice of the L_k's can be made adaptive and is thus not an issue. However, the choice of the kernel K is crucial. Even if any μ-symmetric kernel would eventually do the job, we need to carefully choose it to make the algorithm efficient, as discussed in section 4.4.4.

Consider now a Markov chain $(X_k)_{k\geq 0}$ defined by: $\mathcal{L}(X_0) = \mu$ and the inhomogeneous transitions kernels $M_k(x, dx') = \mathbb{P}(X_k \in dx' | X_{k-1} = x)$, with

$$M_k(x, dx') = \mathbb{1}_{A_k^c}(x)\, \delta_x(dx') + \mathbb{1}_{A_k}(x)(K(x, dx')\mathbb{1}_{A_k}(x') + K(x, A_k^c)\, \delta_x(dx')).$$

Starting from position x in A_k, moving a particle according to M_k is then twofold: firstly a new transition according to K is proposed, and secondly we accept this transition only if it stays in A_k, keeping the old position otherwise.

For $k \in \{0, \dots, n\}$, denote $\mu_k(dx) = \frac{1}{\mu(A_k)}\, \mathbb{1}_{A_k}(x)\, \mu(dx)$ the normalized restriction of μ on A_k, so that $\mu_k(A_{k+1}) = \mathbb{P}(X \in A_{k+1} | X \in A_k)$. At this point, we should also note that instead of a μ-symmetric kernel K to construct the M_k, one can use at level k, any kernel, if available, for which μ_k is invariant. In some applications this can be done directly through a Gibbs sampler (see for example [101]). We have chosen to adopt here a Metropolis-Hastings approach because it is somehow more general, and we will not particularly discuss this case. But from a practical point of view, if such a family of kernels M_k is readily available, then it is much advisable to use it. Anyway, the following stationarity property holds for μ and μ_k.

Proposition 5 *The measures μ and μ_k are both invariant by the transition kernel M_k.*

Now we may give a Feynman-Kac representation for μ_k. From a general point of view, a Feynman-Kac representation for μ_k is a formula of the form

$$\mu_k(\varphi) = \frac{\mathbb{E}[\varphi(X_k) \prod_{m=0}^{k-1} G_m(X_m)]}{\mathbb{E}[\prod_{m=0}^{k-1} G_m(X_m)]},$$

where the potentials G_m are positive functions, and $(X_k)_{k\geq 0}$ is a non homogeneous Markov chain with transitions M_k. If we know how to draw realizations of the Markov chain, then we can compute $\mu_k(\varphi)$ with a Monte Carlo approach. But Crude Monte Carlo is not efficient, because most of the realizations of the chain have small values for the product of the potentials.

However, in this form a much nicer Monte Carlo algorithm can be used. It mainly consists in keeping a cloud of particles (X_k^j), with time $0 \leq k \leq n$ and particle index $1 \leq j \leq N$. Then for each time step k, discard those with small potential G_k, and branch the others, with a rate proportional to $G_k(X_k^j)$. Then apply the Markov transition M_k to all the surviving particles, and

iterate on the time step.

This approach has given birth to a huge amount of literature, and is often referred to as *Sequential Importance Sampling (SIS)* or *Sequential Monte Carlo (SMC)*. See the monograph by Del Moral [40] for a theoretical overview and Doucet, de Freitas and Gordon [48] for examples of applications. In our context, the Feynman-Kac representation for μ_k has the following form.

Proposition 6 *For every test function φ, for $k \in \{0, \dots, n\}$, the Feynman-Kac representation is as follows*

$$\mu_k(\varphi) = \frac{\mathbb{E}[\varphi(X_k) \prod_{m=0}^{k-1} \mathbb{1}_{A_{m+1}}(X_m)]}{\mathbb{E}[\prod_{m=0}^{k-1} \mathbb{1}_{A_{m+1}}(X_m)]},$$

where $(X_k)_{k\geq 0}$ is a Markov chain given by the following conditions: $X_0 \sim \mu$ and the inhomogeneous transition kernels $(M_k)_{k\geq 1}$.

4.2.2 The Fixed-Levels Algorithm

Proposition 6 shows that the framework of Feynman-Kac formulae does apply, and thus this grants access to the approximation of the associated measures using an interacting particle method. Basically, at each iteration k, it consists in selecting the particles according to the potentials, here $\mathbb{1}_{A_{k+1}}$, and then in propagating the particles according to the transitions given by M_{k+1}. The approximation of the rare event probability stems from the following obvious property

$$p = \mathbb{P}(X \in A_n) = \prod_{k=0}^{n-1} \mathbb{P}(X \in A_{k+1} | X \in A_k) = \prod_{k=0}^{n-1} \mu_k(A_{k+1})$$

and finally

$$p = \prod_{k=0}^{n-1} \frac{\mathbb{E}[\mathbb{1}_{A_{k+1}}(X_k) \prod_{m=0}^{k-1} \mathbb{1}_{A_{m+1}}(X_m)]}{\mathbb{E}[\prod_{m=0}^{k-1} \mathbb{1}_{A_{m+1}}(X_m)]},$$

where the last equality comes from Proposition 6. We approximate at each stage the probability $p_k = \mu_k(A_{k+1}) = \mathbb{P}(X \in A_{k+1} | X \in A_k)$ by the proportion of the particles already in the next set, and the total probability is estimated as the product of those. This gives Algorithm 1 below.

Algorithm 1

Parameters

N the number of particles, the sequence $\{L_0, \dots, L_n\}$ of levels.

Initialization

Draw an i.i.d. N-sample $(X_0^j)_{1\leq j\leq N}$, of the law μ.

Iterations

for $k = 0$ to $n-1$ /* level number */

Let $I_k = \{j : X_k^j \in A_{k+1}\}$.

Let $\tilde{p}_k = \frac{|I_k|}{N}$.

for $j \in I_k$, let $\tilde{X}_{k+1}^j = X_k^j$

for $j \notin I_k$, let $\tilde{X}_{k+1}^j$ be a copy of X_k^ℓ where ℓ is chosen randomly in I_k with uniform probabilities.

for $j = 1$ to N /* particle index */

Draw a new particle $\hat{X}_{k+1}^j \sim K(\tilde{X}_{k+1}^j, .)$.

If $\hat{X}_{k+1}^j \in A_{k+1}$ then let $X_{k+1}^j = \hat{X}_{k+1}^j$, else $X_{k+1}^j = \tilde{X}_{k+1}^j$.

endfor

endfor

Output

Estimate the probability of the rare event by $\tilde{p} = \prod_{k=0}^{n-1} \tilde{p}_k$.

Remark: The last set of particles is a (non independent) sample that provides an approximation of the law μ_n of the rare event. The samples are not independent due to the splitting of successful particles.

4.2.3 Fluctuations Analysis

Del Moral [40] has extensively studied in a very general context the asymptotic behavior of the interacting particle model as the number N of particles goes to infinity. For example, it is well known that the estimate $\tilde{p}$ is unbiased. The next proposition presents a precise fluctuation result in our context of rare event analysis.

Proposition 7 *Let $\tilde{p}$ denote the estimate given by the fixed-levels algorithm, then*

$$\sqrt{N}\,\frac{\tilde{p}-p}{p} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0,\sigma^2),$$

with

$$\begin{aligned}\sigma^2 &= \sum_{k=0}^{n-1} \frac{1-p_k}{p_k} + \\ &\quad \sum_{k=1}^{n-1} \frac{1}{p_k} \mathbb{E}\left[\left(\frac{\mathbb{P}(X_{n-1} \in A_n | X_k)}{\mathbb{P}(X_{n-1} \in A_n | X_{k-1} \in A_k)} - 1 \right)^2 \middle| X_{k-1} \in A_k \right].\end{aligned}$$

This result does not correspond exactly to Algorithm 1. The difference is that the proposition assumes that the resampling is done using a multinomial procedure, which gives a higher variance than that of Algorithm 1. This does not make much difference for the following discussion, as the best possible variance is the same. In fact the variance is lower bounded

$$\sigma^2 \geq \sum_{k=0}^{n-1} \frac{1-p_k}{p_k},$$

with equality if and only if for all $k = 1, \ldots, n-1$, and knowing that $X_{k-1} \in A_k$, one has

$$\mathbb{P}(X_{n-1} \in A_n | X_k) \perp X_k.$$

This means that equality holds if, between step k and step $n-1$, the algorithm forgets the initial position X_k. In order to reach this goal, a possible route is to begin step k by applying an

infinite number of times (and not only one time as is the case in Algorithm 1) the transition kernel M_k with stationary distribution $\mu_k = \mathcal{L}(X|X \in A_k)$. We will discuss this point in section 4.4.4.

This will motivate us in the sequel to study an *idealized* version of the algorithm with at each step the possibility (never met in practice) to draw directly an i.i.d. sample of μ_k. As we will see from numerical results, the theoretical performance derived for this idealized version can almost be achieved by the actual algorithm at a reasonable cost. However, from now on, we *assume* that at each step k it is possible to draw an i.i.d. sample of the law of X conditionally on the event $\{X \in A_k\} = \{\Phi(X) > L_k\}$. Then the relative variance of the estimator reduces to

$$\sigma^2 = \sum_{k=0}^{n-1} \frac{1-p_k}{p_k}.$$

Thus, for a fixed value of p and a fixed number n of levels, this asymptotic variance would be minimal if $p_k \equiv p_0$ for all k. This is indeed a simple constrained optimization problem:

$$\underset{p_0,\ldots,p_{n-1}}{\operatorname{argmin}} \sum_{k=0}^{n-1} \frac{1-p_k}{p_k} \qquad \text{s.t.} \qquad \prod_{k=0}^{n-1} p_k = p.$$

In this case, the minimal asymptotic variance is simply $n\frac{1-p_0}{p_0}$, with $p_0 = p^{\frac{1}{n}}$. This optimal situation corresponds to the case where the levels are evenly spaced in terms of probability of success: as far as multilevel splitting for Markov processes is concerned, this point was also mentioned in Glasserman, Heidelberger, Shahabuddin and Zajic [62], Lagnoux [81], and Cérou, Del Moral, Le Gland, and Lezaud [25]. The following section addresses this crucial issue for the adaptive version of the algorithm. Before this, two remarks are in order.

Remarks:

1. If one's particular interest is the variance of $\tilde{p}$ rather than a convergence in distribution like the CLT-type result of Proposition 7, then we can turn to the recent non asymptotic results obtained in Cérou, Del Moral and Guyader [24, corollary 5.2] (see also appendix A). Under some regularity conditions mainly about the mixing property of the kernel K, there exist positive constants α_k, for $0 \leq k \leq n-1$, such that for all $N \geq N_0 = \sum_{k=0}^{n-1} \frac{\alpha_k}{p_k}$,

$$\mathbb{E}\left(\left[\frac{\tilde{p}-p}{p}\right]^2\right) \leq 4\frac{N_0}{N}.$$

   If we assume an i.i.d. sample at each step, then all the α_k's are all equal to 1, and $N_0 = \sum_{k=0}^{n-1} \frac{1}{p_k}$.

2. Finally, there is a maybe small, but non-zero, probability that the particle system dies at some stage. This may typically happen when two consecutive levels are too far apart, or when the number of particles is too small. A first solution to this problem is given in Le Gland and Oudjane [82]. The idea is to go on sampling new particles until a given number of them have reached the given level. The price to pay is a possibly very long computation time. A second solution is proposed in the next section.

4.3 A First Adaptive Method

4.3.1 The Algorithm

As we may not have a great insight about the law μ and/or the mapping Φ, typically when Φ is a black box, the choice of the levels $L_1, \ldots, L_{n-1}$ might prove to be quite problematic. We propose

from now on to adaptively choose the level sets, ensuring not only that the particle system never dies but also that the asymptotic variance of the estimate $\tilde{p}$ is minimized.

The method is very easy to implement. We choose a prescribed success rate p_0 between two consecutive levels. In practice, $p_0 = 0.75$ works well. At step k, the algorithm sorts the particles X_k^j according to their scores $\Phi(X_k^j)$. Then it sets the next level to the $(1-p_0)$ empirical quantile $\tilde{L}_{k+1}$, which means that a proportion p_0 of the particles scores are above it. Starting from this sample of $p_0 N$ particles which are (ideally) independently and identically distributed according to the law $\mathcal{L}(X|\Phi(X) > \tilde{L}_{k+1})$, an i.i.d. sample of size N is drawn with the same distribution, and the rest of the algorithm is unchanged.

The algorithm then stops when some $\tilde{L}_{\tilde{n}_0+1} > q$, and the probability is estimated by $\tilde{p} = \tilde{r}_0 \; p_0^{\tilde{n}_0}$, where $\tilde{r}_0$ denotes the number of particles in the last iteration being above level q. The number $\tilde{n}_0$ of steps is random, but if N is large enough, then we can prove that, outside an event of exponentially small probability, $\tilde{n}_0$ is actually fixed by the ratio of the logarithms

$$n_0 = \left\lfloor \frac{\log \mathbb{P}(X \in A)}{\log p_0} \right\rfloor = \left\lfloor \frac{\log p}{\log p_0} \right\rfloor .$$

As mentioned above, this variant enforces evenly spaced levels in terms of probability of success, and therefore a minimal asymptotic variance for the estimate $\tilde{p}$ of p. The pseudo-code for the adaptive (idealized) algorithm is given in Algorithm 2 below.

Algorithm 2

Parameters

N the number of particles, the number $N_0 < N$ of succeeding particles, and let $p_0 = N_0/N$.

Initialization

Draw an i.i.d. N-sample $(X_0^j)_{1 \le j \le N}$ of the law μ.

Compute $\tilde{L}_1$, the $(1 - p_0)$ quantile of $\Phi(X_0^j), j = 1, \ldots, N$.

$k = 1$;

Iterations

while $\tilde{L}_k \le q$ do

Starting from an i.i.d. $p_0 N$-sample with law $\mathcal{L}(X|\Phi(X) > \tilde{L}_k)$, draw an i.i.d. N-sample $(X_k^j)_{1 \le j \le N}$ with the same law.

Compute $\tilde{L}_{k+1}$, the $(1 - p_0)$ quantile of $\Phi(X_k^j), j = 1, \ldots, N$.

$k = k + 1$;

endwhile

Let N_L the number of particles X_{k-1}^j, $j = 1, \ldots, N$, such that $\Phi(X_{k-1}^j) > q$.

Output

Estimate the probability of the rare event by $\tilde{p} = \dfrac{N_L}{N} \; p_0^{k-1}$.

Remarks:

1. In this algorithm, the step drawing an N-sample starting from a $p_0 N$-sample is of course the trickiest one, that is why we call it the idealized algorithm. Once again, the analytical study of this idealized version in the next subsection assumes it can be done perfectly, although this will never be met in practice. In section 4.4.4, we propose a way to implement it in practice, at least approximately, and chapter 5 shows its practical efficiency on several examples.

2. The cost of adaptive levels is a higher complexity by a factor $\log N$, due to the quick sort, and a slight loss of accuracy due to a bias. Yet, Proposition 8 below proves that this bias becomes negligible compared to the standard deviation as N increases and provides an explicit formula, which allows to correct this bias and to derive confidence intervals. Experimental results of chapter 5 illustrate this.

4.3.2 Bias and Variance

The assumption of a continuous cumulative distribution function (cdf) F of $\Phi(X)$ is now required to derive the properties of the adaptive algorithm. Let us write the rare event probability as

$$p = r_0\, p_0^{n_0}, \text{ with } n_0 = \left\lfloor \frac{\log p}{\log p_0} \right\rfloor \text{ and } r_0 = p\, p_0^{-n_0},$$

so that $r_0 \in (p_0, 1]$. In the same way we write $\tilde{p} = \tilde{r}_0\, p_0^{\tilde{n}_0}$, with $\tilde{n}_0$ the number of steps before the algorithm stops. A first theorem shows a CLT-type convergence.

Theorem 8 *If F is continuous, then we have*

$$\sqrt{N}\, \frac{\tilde{p} - p}{p} \xrightarrow[N\to+\infty]{\mathcal{D}} \mathcal{N}(0, \sigma^2),$$

where

$$\sigma^2 = n_0 \frac{1 - p_0}{p_0} + \frac{1 - r_0}{r_0}.$$

Unlike the fixed-levels version of the algorithm, the adaptive version is biased. Nevertheless, the next result shows that the bias is of order $1/N$, and is thus negligible compared to the standard deviation given in Theorem 8 for the idealized algorithm.

Proposition 8 *If F is continuous, then we have*

$$N\, \frac{\mathbb{E}[\tilde{p}] - p}{p} \xrightarrow[N\to+\infty]{} n_0 \frac{1 - p_0}{p_0}.$$

Thus the bias is positive and of order $\frac{1}{N}$ when N goes to infinity

$$\mathbb{E}[\tilde{p}] - p \sim \frac{p}{N} \frac{n_0(1 - p_0)}{p_0}.$$

Putting all things together, we can write the following asymptotic expansion

$$\tilde{p} = p\left(1 + \frac{1}{\sqrt{N}} \sqrt{n_0 \frac{1 - p_0}{p_0} + \frac{1 - r_0}{r_0}}\; Z + \frac{1}{N} n_0 \frac{1 - p_0}{p_0} + o_{\mathbb{P}}\left(\frac{1}{N}\right)\right),$$

where Z is a standard Gaussian variable.

Finally, it is worth mentioning that the bias is always positive, giving a slightly overvalued estimate. As rare event analysis usually deals with catastrophic events, it is not a bad thing that the real value be a bit lower than the provided estimate. Moreover, if one wants to correct it, the explicit formula of Proposition 8 allows to do so.

4.4 A Second Adaptive Method

4.4.1 Introduction

The analysis of the statistical properties of $\tilde{p}$, the tail probability estimate of p presented in the previous section, reveals that when the number of particles N tends to infinity, the expectation and variance are respectively

$$\mathbb{E}[\tilde{p}] = p + \mathcal{O}(N^{-1}) \quad \text{and} \quad \mathrm{Var}(\tilde{p}) = \frac{p^2}{N}\left(n_0 \frac{1-p_0}{p_0} + \frac{1-r_0}{r_0}\right) + o(N^{-1}),$$

where

$$n_0 = \left\lfloor \frac{\log p}{\log p_0} \right\rfloor \quad \text{and} \quad r_0 = p\, p_0^{-n_0}.$$

so that $r_0 \in (p_0, 1]$. Since the function $\psi : p_0 \mapsto (1-p_0)/(-p_0 \log p_0)$ is nonincreasing on $(0,1)$ (see figure 4.1), one can deduce that the larger p_0, the lower the variance, with

$$\lim_{p_0 \to 1^-} \mathrm{Var}(\tilde{p}) = \frac{p^2 \times -\log p}{N}.$$

Hence the idea to choose p_0 as large as possible. However, with an adaptive method such as the one of the previous section, the largest possible value is clearly $p_0 = 1 - 1/N$. This is the main idea of this second adaptive method.

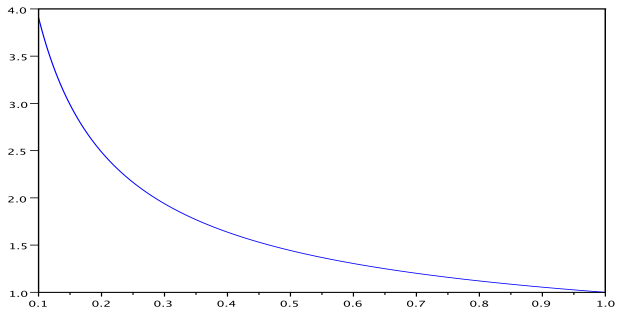

Figure 4.1: Function $\psi : p_0 \mapsto (1-p_0)/(-p_0 \log p_0)$.

Indeed, this section presents a refinement of Algorithm 2: at each iteration j, define the new level L_j as the minimum of $\Phi(\cdot)$ evaluated on the N particles, remove the particle that achieves the minimum, and use the Metropolis-Hastings algorithm to rebranch the removed particle according to the conditional distribution of X knowing that $\{\Phi(X) > L_j\}$. This is a crucial step of the algorithm. Ideally, we would like to exactly sample from the conditional distribution $\mathcal{L}(X|\Phi(X) > L_j)$. In practice, as in the previous section, this is impossible. Nevertheless, we will analyze our algorithm under that very strong assumption. As previously, to avoid any misunderstanding, we will call this the *idealized* algorithm. Even if it does not completely match with the algorithm used in practice, it gives us an insight about the optimal performance this latter could reach. In particular, we will show that our idealized algorithm improves the current state-of-the-art algorithms.

Unfortunately, the techniques used to prove Theorem 8 and Proposition 8 cannot be applied to this new algorithm. Instead, the analysis of this idealized algorithm uses a novel technique that

exploits Poisson processes to obtain an exact description of the statistical properties of the estimate for a finite number of particles N. The analysis holds for both the problem of estimating the tail probability for a given quantile and the problem of estimating the quantile given a specified tail probability. To our knowledge, the application of multilevel splitting techniques to quantile estimation is new. Furthermore, the idealized approach enables us to produce non-asymptotic confidence intervals for the estimated quantities with respect to the number of particles.

Finally, we would like to stress that our methodology fits nicely within the modern computational Bayesian paradigm, since it provides a novel tool for computing extreme quantiles of posterior distributions of univariate functions of the parameters. In a different context, it indeed bears a resemblance to the "Nested Sampling" approach as initially and independently proposed by Skilling [107, 108] and recently analysed by Chopin and Robert [33].

4.4.2 Algorithm

First of all, we present the new adaptive algorithm in a unified context, that means for the estimation of a tail probability as well as for the estimation of an extreme quantile, depending on the stopping rule.

- Start with an i.i.d. sample $(X_1, X_2, \ldots, X_N)$ from μ and initialize $L_0 = -\infty$ and
$$X_1^1 = X_1, \ldots, X_N^1 = X_N.$$
- For $m = 1, 2, \ldots$, set
$$L_m = \min(\Phi(X_1^m), \ldots, \Phi(X_N^m)),$$
and define for all $i = 1, 2, \ldots, N$:
$$X_i^{m+1} = \begin{cases} X_i^m & \text{if } \Phi(X_i^m) > L_m \\ X^\star \sim \mathcal{L}(X|\Phi(X) > L_m) & \text{if } \Phi(X_i^m) = L_m, \end{cases}$$
where $X^\star$ is independent of $\{X_1^m, \ldots, X_N^m\}$.
- Stopping rules:
  (1) **To estimate a tail probability p given a quantile q,**
  continue until $m = M$ where $M = \max\{m : L_m \leq q\}$ and set
  $$\hat{p} = \left(1 - \frac{1}{N}\right)^M.$$
  We will show that M is a Poisson distributed random variable.
  (2) **To estimate a quantile q given a tail probability p,**
  continue until iteration
  $$m = \left\lceil \frac{\log(p)}{\log(1 - N^{-1})} \right\rceil,$$
  and set $\hat{q} = L_m$. Note that this time, the number of iterations is deterministic.

Remark: Once again, simulating exactly according to $\mathcal{L}(X|\Phi(X) > L_m)$ is impossible in general and we propose in section 4.4.4 to do so approximately using Markov Chain Monte Carlo techniques. However, for the theoretical analysis, we will consider only the case where that simulation could be done perfectly, and we call it the idealized algorithm.

4.4.3 Statistical Results on the Idealized Algorithm

Suppose that the distribution μ of X and the mapping Φ are such that the univariate random variable $Y = \Phi(X)$ has cumulative distribution function F for which, as in the previous section, we only assume continuity. This is the only assumption we make about the distribution of X and the transformation Φ, unless stated otherwise. We denote the survival function and the integrated hazard function of Y by $S(y) = 1 - F(y)$, and $\Lambda(y) = -\log S(y)$, respectively. The main result in this section describes the joint distribution of the levels $L_1, L_2, L_3, \ldots$ generated by our algorithm.

Theorem 9 *The random variables $\Lambda(L_1), \Lambda(L_2), \Lambda(L_3), \ldots$ are distributed as the successive arrival times of a Poisson process with rate N, that is,*

$$\Lambda(L_m) \stackrel{d}{=} \frac{1}{N} \sum_{j=1}^{m} E_j,$$

where $E_1, \ldots, E_m$, are i.i.d. Exponential(1).

Estimating a Tail Probability

Consider the problem of estimating the tail probability $p = \mathbb{P}(\Phi(X) > q)$ for a given quantile q. Applying the results of Theorem 9 to stopping rule number 1, we obtain the following corollary:

Corollary 3 *The random variable $M = \max\{m : L_m \leq q\}$ is distributed according to a Poisson law with parameter $-N \log p$.*

It follows from this corollary that $\mathbb{E}[M] = \mathrm{Var}(M) = -N \log p$. Furthermore, the classical approximation of the Poisson distribution by a Gaussian law $\mathcal{N}(-N \log p, -N \log p)$ is of course valid in our context since N is assumed to be large (at least 100) and p small.

Since we discard exactly one particle among N at each step of the algorithm, a natural estimator for the tail probability p is indeed

$$\hat{p} = \left(1 - \frac{1}{N}\right)^M$$

and the following proposition describes its distribution.

Proposition 9 *The estimator $\hat{p}$ for the tail probability p is a discrete random variable taking values in*

$$\mathcal{S} = \left\{1, \left(1 - \frac{1}{N}\right), \left(1 - \frac{1}{N}\right)^2, \ldots\right\},$$

with probability

$$\mathbb{P}\left[\hat{p} = \left(1 - \frac{1}{N}\right)^m\right] = \frac{p^N (-N \log p)^m}{m!}, \quad m = 0, 1, 2, \ldots.$$

It follows that $\hat{p}$ is an unbiased estimator of p with variance

$$\mathrm{Var}(\hat{p}) = p^2 \left(p^{-\frac{1}{N}} - 1\right).$$

Comparing our estimator with the one obtained through Crude Monte Carlo (CMC) is instructive. Recall that the CMC estimate for the tail probability is given by

$$\hat{p}_{mc} = \frac{\hat{N}_{mc}}{N} = \frac{\#\{i \in \{1, \ldots, N\} : \Phi(X_i) > q\}}{N},$$

where N is the size of the CMC sample. The random variable $\hat{N}_{mc}$ has a Binomial distribution with parameters (N, p), and hence $\hat{p}_{mc}$ is an unbiased estimator with relative variance

$$\frac{\mathrm{Var}(\hat{p}_{mc})}{p^2} = \frac{1-p}{Np} \approx \frac{1}{Np}.$$

The last approximation assumes that p is small and hence $1-p \approx 1$. Thus the sample size N has to be at least as large as $1/p$ in order to get a reasonable precision. Compare the latter with the relative variance of our estimator $\hat{p}$

$$\frac{\mathrm{Var}(\hat{p})}{p^2} = \left(p^{-\frac{1}{N}} - 1\right) \approx \frac{-\log p}{N},$$

when p is very small and/or N is large. This proves that, for the same precision in terms of variance of the estimator, CMC requires about $(-p \log p)^{-1}$ more particles than the method presented in this paper. However the CMC estimator has a lower complexity than our algorithm, this point will be discussed in Section 4.4.5.

For now, we can use Proposition 9 to derive confidence intervals for p. Let α be a fixed number between 0 and 1, e.g. $\alpha = 0.05$, and denote by $Z_{1-\alpha/2}$ the quantile of order $1-\alpha/2$ of the standard Gaussian distribution.

Proposition 10 *Let us denote*

$$\hat{p}_{\pm} = \hat{p}\ \exp\left(\pm \frac{Z_{1-\alpha/2}}{\sqrt{N}} \sqrt{-\log \hat{p} + \frac{Z^2_{1-\alpha/2}}{4N}} - \frac{Z^2_{1-\alpha/2}}{2N}\right),$$

then $I_{1-\alpha}(p) = [\hat{p}_-, \hat{p}_+]$ *is a* $100(1-\alpha)\%$ *confidence interval for* p.

For example, if $\alpha = 0.05$, then $Z_{1-\alpha/2} \approx 2$, and neglecting the terms of order $1/N$ gives the following 95% confidence interval for p

$$\hat{p}\ \exp\left(-2\sqrt{\frac{-\log \hat{p}}{N}}\right) \leq p \leq \hat{p}\ \exp\left(+2\sqrt{\frac{-\log \hat{p}}{N}}\right). \tag{4.1}$$

The asymmetry of this confidence interval around $\hat{p}$ arises from the distribution of $\hat{p}$ around its mean p. Indeed, since M is approximately Gaussian, $\hat{p}$ is approximately log-Gaussian. We will illustrate this result in section 5.1.

Estimating a Large Quantile

Consider now the problem of estimating the quantile q for a given p such that $\mathbb{P}(\Phi(X) > q) = p$. Using stopping rule number 2 described in Section 4.4.2, a natural estimator for the quantile q is

$$\hat{q} = L_m,$$

where $m = \left\lceil \frac{\log p}{\log(1-N^{-1})} \right\rceil$. Given sufficient smoothness of the distribution at the quantile q, we obtain an asymptotic normality result for our estimator.

Proposition 11 *If cdf* F *is differentiable at point* q*, with density* $f(q) \neq 0$*, then*

$$\sqrt{N}(\hat{q} - q) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}\left(0, \frac{-p^2 \log p}{f(q)^2}\right).$$

The CMC estimator for q is defined as $\hat{q}_{mc} = Y_{(\lfloor (1-p)N \rfloor)}$, where $Y_{(1)} \leq \cdots \leq Y_{(N)}$ are the order statistics of $\Phi(X_1), \ldots, \Phi(X_N)$ and $\lfloor y \rfloor$ stands for the integer part of y. It satisfies the following CLT type result (see for example Schervish [103], Theorem 7.25)

$$\sqrt{N}(\hat{q}_{mc} - q) \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}\left(0, \frac{p(1-p)}{f(q)^2}\right).$$

This proves again that in order to achieve the same precision in terms of variance of the estimator, CMC requires about $(-p \log p)^{-1}$ more particles than the estimator proposed here. The following proposition describes the bias of our estimator. As already noticed in van Zwet [114] for the CMC estimator, the estimation of the bias requires further assumptions.

Proposition 12 *If F^{-1} is twice differentiable on $(0,1)$ with continuous second derivative on $(0,1)$, if $(F^{-1})'(t) > 0$ for $t \in (0,1)$, and if there exist non-negative numbers a and b such that $F^{-1}(t)t^a(1-t)^b$ is bounded for $t \in (0,1)$, then the bias of $\hat{q}$ is bounded from below by*

$$\lim_{N\to\infty} N(\mathbb{E}[\hat{q}] - q) \geq \left(\log p - \frac{pf'(q)}{2f(q)^2}(-2 - \log p)\right)\frac{p}{f(q)},$$

and bounded from above by

$$\lim_{N\to\infty} N(\mathbb{E}[\hat{q}] - q) \leq \left(1 + \log p - \frac{pf'(q)}{2f(q)^2}(2 - \log p)\right)\frac{p}{f(q)}.$$

Remarks:

1. As usual the bias is in $\mathcal{O}(1/N)$ where as standard deviation is in $\mathcal{O}(1/\sqrt{N})$, so that only this latter is worth of attention when N is large enough.

2. In these inequalities, it is assumed that $f'(q) < 0$. Suitably modified upper and lower bounds are readily obtained when $f'(q) > 0$. We chose to present the results for $f'(q) < 0$, as that assumption is more likely to hold in practice.

3. The assumptions to get expressions for the bias and the variance are the same as in CMC. For this estimator, it is known from the theory of order statistics (see for example van Zwet [114], Lemma 3.2.2, or Arnold, Balakrishnan and Nagaraja [6], p.128) that:

$$\mathbb{E}[\hat{q}_{mc}] = q - \frac{1}{N} \cdot \frac{p(1-p)f'(q)}{2f(q)^3} + o(1/N).$$

The obtained expression for the asymptotic variance in Proposition 11 proves that $\hat{q}$ is much more precise than the CMC estimator $\hat{q}_{mc}$, but is of limited practical use as it requires the knowledge of $f(q)$. Nonetheless, exploiting the connection with Poisson processes allows us to derive non asymptotic confidence intervals for q without having to estimate the density at the quantile q. Indeed, fix $\alpha \in (0,1)$, denote by $Z_{1-\alpha/2}$ the quantile of order $1-\alpha/2$ of the standard Gaussian distribution, and define

$$\begin{aligned} m_- &= \left\lfloor -N\log p - Z_{1-\alpha/2}\sqrt{-N\log p} \right\rfloor \\ m^+ &= \left\lceil -N\log p + Z_{1-\alpha/2}\sqrt{-N\log p} \right\rceil \end{aligned}$$

and consider L_{m_-}, L_{m^+} the associate levels. The following proposition provides a $1-\alpha$ confidence interval for q.

Proposition 13 *If the cdf F is continuous, then a $100(1-\alpha)\%$ confidence interval for the quantile q is $I_{1-\alpha}(q) = [L_{m_-}, L_{m^+}]$.*

Remarks:

1. The computational price to pay to obtain the confidence interval is the cost of running the algorithm until $m = m^+$ in order to get the upper confidence bound L_{m^+}. This requires the algorithm to run around $Z_{1-\alpha/2}\sqrt{-N \log p}$ additional steps.

2. Compared to Proposition 11, the great interest of this property lies in the fact that it does not require any estimation of the probability density function f. This result will also be illustrated in section 5.1.

4.4.4 Practical Implementation

This section explains how to generate a random variable $X^\star$ from the conditional distribution $\mathcal{L}(X|\Phi(X) > L_m)$ that is needed at each step in the first adaptive algorithm as well as in the second adaptive algorithm. Let us recall that μ denotes the law of X. As mentioned above, to draw $X^\star$, we run a Monte Carlo Markov Chain with a suitable μ-symmetric and one-step μ-irreducible kernel K. That is: K satisfies the detailed balance property with μ; and from any initial point x, the Radon-Nikodym derivative $dK(x, dx')/d\mu(dx')$ is strictly positive. Either, one knows such a kernel K or otherwise could use a Metropolis-Hasting kernel K based on a one-step μ-irreducible instrumental kernel $Q(x, dx')$ (see for example Robert and Casella [98]).

Toy Example: Let us suppose that X has a standard Gaussian distribution on $\mathbb{R}$. Then let us present two ways to get such a transition kernel K:

1. Direct construction: fix $\sigma > 0$ and denote K the transition kernel defined by

$$K(x, dx') = \sqrt{\frac{1+\sigma^2}{2\pi\sigma^2}} \exp\left(-\frac{1+\sigma^2}{2\sigma^2}\left(x' - \frac{x}{\sqrt{1+\sigma^2}}\right)^2\right) \lambda(dx'),$$

   where λ stands for Lebesgue measure on $\mathbb{R}^d$. Denoting W a Gaussian standard variable, the transition $X \rightsquigarrow X'$ proposed by K is simply $X' = (X + \sigma W)/\sqrt{1+\sigma^2}$.

2. Metropolis-Hastings kernel: fix $\sigma > 0$ and denote Q the transition kernel defined by

$$Q(x, dx') = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x'-x)^2}{2\sigma^2}\right) \lambda(dx') = q(x, x')\lambda(dx').$$

   Denoting W a Gaussian standard variable, the transition $X \rightsquigarrow X'$ proposed by K is $X' = X + \sigma W$. Then, starting from Q, the transition kernel K constructed by Metropolis-Hastings is μ-symmetric and one-step μ-irreducible.

Application to the Adaptive Algorithms

Consider that a μ-symmetric and one-step μ-irreducible transition kernel K is available. For all $m = 1, 2, \ldots$, knowing $L_1 = \ell_1, L_2 = \ell_2, \ldots$, consider the sets

$$A_m = \{x \in \mathbb{R}^d \text{ s.t. } \Phi(x) > \ell_m\},$$

and let us call μ_m the normalized restriction of μ on A_m

$$\mu_m(dx) = \frac{1}{\mu(A_m)} \mathbb{1}_{A_m}(x)\, \mu(dx).$$

We define also the transition kernel K_m by

$$K_m(x, dx') = \mathbb{1}_{A_m^c}(x)\,\delta_x(dx') + \mathbb{1}_{A_m}(x)(K(x, dx')\mathbb{1}_{A_m}(x') + K(x, A_m^c)\,\delta_x(dx')).$$

The idea behind the definition of K_m is very simple: starting from x in A_m, the kernel K proposes a transition $x \rightsquigarrow x'$. Then, if $\Phi(x') > \ell_m$, the transition is accepted, else it is rejected and x stays at the same place.

With these notations, it easy to see that the probability measure μ_m is invariant by the transition kernel K_m. Moreover, K_m is also Harris recurrent, and we have for any initial distribution ν such that $\nu(A_m) = 1$

$$\|\nu K_m^n - \mu_m\| \xrightarrow[n\to+\infty]{} 0, \tag{4.2}$$

where $\|\cdot\|$ is the total variation norm. In our context, let us fix $m = 1$, so that the algorithm begins with an i.i.d. sample $(X_1, X_2, \ldots, X_N)$ from μ, and initialize

$$X_1^1 = X_1, \ldots, X_N^1 = X_N.$$

In order to simplify notations, suppose that

$$\Phi(X_1^1) < \ldots < \Phi(X_N^1),$$

so that $L_1 = \Phi(X_1^1)$ and

$$X_2^2 = X_2^1, \ldots, X_N^2 = X_N^1.$$

Knowing $L_1 = \ell_1$, the sample $(X_2^2, \ldots, X_N^2)$ is i.i.d. with distribution μ_1. Now pick at random an integer i between 2 and N and set $X_0^\star = X_i^2$. Thus $X_0^\star$ is also distributed according to μ_1, but is not independent from $\{X_2^2, X_3^2, \ldots, X_N^2\}$. In order to get independence, apply iteratively the transition kernel K_1 to $X_0^\star$. Knowing $X_i^2 = x_i^2$, one has $\delta_{x_i^2}(A_1) = 1$ since by construction $\Phi(x_i^2) > \ell_1$. As a consequence, the result given by equation (4.2) may be applied

$$\left\| \int \delta_{x_i^2} K_1^n - \mu_1 \right\| \xrightarrow[n\to+\infty]{} 0.$$

Thus, after "enough" applications of the kernel K_1, $X_0^\star$ has mutated into a new particle $X^\star$ that is distributed according to μ_1 and is now "almost" independent from the initial position X_i^2. Denoting by $X_1^2 = X^\star$, we have constructed a sample $(X_1^2, \ldots, X_N^2)$ of i.i.d. random variables with common distribution $\mathcal{L}(X|\Phi(X) > \ell_1)$. The principle of the algorithm is to iteratively apply this simple idea.

Remarks:

1. One would theoretically have to iterate K_m an infinite number of times to get independence at each step and to match perfectly with the theoretical analysis of the idealized algorithm in section 4.4.3. This is of course unrealistic, and in practice it is applied only a finite number of times, denoted T. In the watermarking example of section 5.1, we have applied it $T = 20$ times at each step and this led to an excellent agreement between the idealized and empirical results. However, this is certainly due to the fact that this is an extremely regular situation, and we admit that one can undoubtedly find cases where things do not happen so nicely.

2. The second remark is about the choice of the transition kernel K. To fix ideas, let us come back to the toy example where X has a standard Gaussian distribution on $\mathbb{R}$, i.e., $\mu = \mathcal{N}(0, 1)$, and $\Phi(x) = x$. Two μ-symmetric kernels have been proposed. Both require to choose the value of a standard deviation parameter σ. The value of σ has in fact a great impact on the efficiency of the algorithm. Indeed, if σ is too small, then almost all of the T

proposed transitions will be accepted, but since each transition corresponds (in expectation) to a small move, it will require a large T to forget the initial position. On the other side, if σ is too large, then almost all of the T proposed transitions will be rejected, but each transition corresponds (in expectation) to a huge move, so that it will require a rather low T to forget its initial position. Consequently, a trade-off has to be found for the "mixing" parameter σ. As a rule of thumb, it seems reasonable to count the proportion of accepted transitions at each step, and if this proportion is below a certain rate (say for example 20%) then one may reduce σ (say for example by a factor of 10%). This adaptive tuning is possible since K has the desired properties with respect to μ for *any* value of σ. In this respect, we would like to mention that there is a huge amount of literature on appropriate scaling of random walk Metropolis algorithms, dating back at least to Roberts, Gelman and Gilks [99].

3. Keeping the notations of the previous remark, one could think that, as the algorithm goes on and concentrates on regions with smaller and smaller probabilities, one would have to reduce the mixing parameter σ with increasing iteration. In fact, and as will be illustrated in section 5.1, this is not the case when dimension d is large enough: in such a situation, a region with very small probability may indeed be very large. For our purpose, one could call this phenomenon the "blessing of dimensionality", in opposition to the statistical "curse of dimensionality".

Pseudo-Code for Estimating p

We give now the pseudo-code version of the algorithm for the tail probability estimation when q is given.

Algorithm 3

Parameters

The number N of particles, the quantile q, the number T of proposed transitions, a μ-reversible kernel transition K.

Initialization

$m = 1$.

Draw an i.i.d. N-sample $(X_1^m, \ldots, X_N^m)$ of the law μ.

Sort the vector $(\Phi(X_1^m), \ldots, \Phi(X_N^m))$.

Denote $(X_1^m, \ldots, X_N^m)$ the sorted sample according to Φ and $L_1 = \Phi(X_1^m)$.

Iterations

while $L_m < q$

Pick an integer R randomly between 2 and N.
Let $X_1^{m+1} = X_R^m$.
for $t = 1 : T$
From X_1^{m+1}, draw a new particle $X^\star \sim K(X_1^{m+1}, .)$.
If $\Phi(X^\star) > L_m$, then let $X_1^{m+1} = X^\star$.
endfor

Let $(X_2^{m+1}, \ldots, X_N^{m+1}) = (X_2^m, \ldots, X_N^m)$.

Put $\Phi(X_1^{m+1})$ at the right place in the sorted vector $(\Phi(X_2^{m+1}), \ldots, \Phi(X_N^{m+1}))$ (dichotomic search).

Denote $(X_1^{m+1}, \ldots, X_N^{m+1})$ the sorted sample according to Φ and $L_{m+1} = \Phi(X_1^{m+1})$.

$m = m + 1$.

endwhile

Output

$\hat{p} = \left(1 - \frac{1}{N}\right)^{m-1}$.

Remarks:

1. For the pseudo-code of the first adaptive method (Algorithm 2), one has to apply the same procedure "for $t = 1 : T$ (...) endfor" to all the $(1-p_0)N$ particles that need to be resampled, and not only to one particle as is the case in Algorithm 3 above.

2. As mentioned in section 4.4.2, if we want to estimate a large quantile instead of a tail probability, then we just have to replace the loop "while $L_m < q$ (...) endwhile" with the loop "for $m = 1 : \left\lceil \frac{\log(p)}{\log(1-N^{-1})} \right\rceil$ (...) endfor". We call it **Algorithm 4**.

4.4.5 Complexity, Efficiency and Asymptotic Efficiency

In this section, we mix the theoretical results of the idealized algorithm derived in section 4.4.3 and the computational complexity of Algorithm 3 above. Once again, we do acknowledge that one might not find this analysis totally convincing. However it gives us an insight about our method regarding the crucial issues of complexity and efficiency.

The expected computational complexity C_N of Algorithm 3 is $\mathcal{O}(N \log N \log p^{-1})$ since it requires:

- A sorting of the initial sample, whose cost is (in expectation) in $\mathcal{O}(N \log N)$ via a quicksort algorithm;
- Around $\mathbb{E}[M] = -N \log p$ steps (where $p = \mathbb{P}(\Phi(X) > q)$), whose cost is decomposed in:
  - T proposed kernel transitions,
  - the dichotomic search and the insertion of the new particle at the right place in the ordered sample, whose cost is in $\mathcal{O}(\log N)$ via a min-heap algorithm (see for example Knuth [78]).

By comparison, the algorithm complexity of CMC is N. The complexity of Algorithm 2, where at each iteration, instead of killing and branching the smallest particle, we are branching a proportion $(1 - p_0)$ (typically $p_0 = 3/4$) is also in $\mathcal{O}(N \log N \log p^{-1})$.

We noticed in section 4.4.3 that our estimator $\hat{p}$ of p has a smaller variance than $\hat{p}_{mc}$ but a larger computational complexity. To take into account both computational complexity and variance, Hammersley and Handscomb have proposed to define the efficiency of a Monte Carlo process as "inversely proportional to the product of the sampling variance and the amount of labour expended in obtaining this estimate" [69]. So Algorithm 3 is a bit more efficient than Algorithm 2 because the variance of $\hat{p}$ is a bit smaller than the variance of $\tilde{p}$ while sharing similar computational costs.

Specifically, the proposed estimator $\hat{p}$ is computationally more efficient than the CMC estimator $\hat{p}_{mc}$ whenever

$$\mathrm{Var}(\hat{p}) \times C_N \quad \leq \quad \mathrm{Var}(\hat{p}_{mc}) \times C_{mc},$$

that is

$$-\frac{p^2 \log p}{N} \cdot (-kN \log N \log p) \leq \frac{p(1-p)}{N} \cdot N \quad \Leftrightarrow \quad k \log N \quad \leq \quad \frac{1-p}{p(\log p)^2}.$$

That inequality is satisfied when p goes to zero since the right-hand side goes then to infinity. For example, let us fix $N = 200$ and $k = 10$, then one can check numerically that the condition

$$10 \log(200) \quad \leq \quad \frac{1-p}{p(\log p)^2}$$

is true as soon as $p \leq 1.0 \times 10^{-4}$. The take-home message here is that our adaptive method for estimating tail probabilities is useful only for rare events. If the probability is not that rare, then one might simply apply a Crude Monte Carlo method.

Our calculations on $\hat{p}$ enable us to derive another efficiency result for rare event probability estimation based on the asymptotic behavior of the relative variance of the estimator when the rare event probability p goes to 0. Here we will focus only on the asymptotic efficiency, as discussed in Glynn and Whitt [63]. Recall that an estimator $\hat{p}$ for the tail probability p is said to reach asymptotic efficiency if for N fixed

$$\lim_{p \to 0} \frac{\log\left(\mathrm{Var}(\hat{p}) \times C(\hat{p})\right)}{2 \log p} = 1.$$

Jensen's inequality shows that for any unbiased estimator

$$\limsup_{p \to 0} \frac{\log\left(\mathrm{Var}(\hat{p}) \times C(\hat{p})\right)}{2 \log p} \leq 1.$$

For example, the CMC method does not reach asymptotic efficiency since

$$\frac{\log\left(\mathrm{Var}(\hat{p}_{mc}) \times C(\hat{p}_{mc})\right)}{2 \log p} = \frac{\log p + \log(1-p)}{2 \log p} \xrightarrow[p \to 0]{} \frac{1}{2}.$$

Thanks to Proposition 9, we get for the proposed estimator

$$\frac{\log\left(\mathrm{Var}(\hat{p}) \times C(\hat{p})\right)}{2 \log p} = 1 + \frac{\log\left(p^{-\frac{1}{N}} - 1\right) + \log(kN \log N \log p^{-1})}{2 \log p} \xrightarrow[p \to 0]{} 1 - \frac{1}{2N}.$$

Consequently, since the number N of particles is supposed to be large, the proposed method almost reaches asymptotic efficiency.

Chapter 5

Applications in a Static Context

5.1 Watermarking

Digital watermarking is a set of techniques for embedding information in digital files, such as audio files, images, or video files. Ideally, this embedding should minimally distort the original, be robust to corruption, and be hard to remove. Digital watermarking with these properties enable ownership attribution of digital media that is essential for digital rights management. For example, watermarking is used for copy protection by optical disk players to prevent and deter unauthorized copying of digital media by refusing to record any watermarked content (see Digital Rights Management site for DVD copy [71] and Figure 5.1).

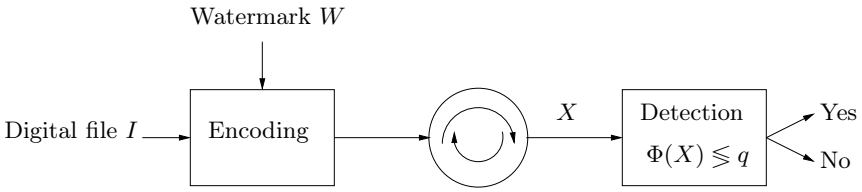

Figure 5.1: Schematic representation of digital watermarking.

The probability of false alarm is the probability that the detector considers an original piece of content (which has not been watermarked) as protected. The movie that a user shot during his holidays could be rejected by his storage device. This absolutely non user-friendly behavior really scares consumer electronics manufacturers, and should thus be very small. In 1997, the standards group for DVD copyright protection called for technologies capable of producing at most one false alarm in 400 hours of operations. As the detection rate was one decision per ten seconds, this implied a probability of false alarm of about 7×10^{-6}. Since 2001, consumer electronics manufacturers claim no error in "316,890 years", or equivalently a false positive probability of 1×10^{-12}. A fundamental problem in developing and evaluating watermarking for digital rights management is to estimate the probability of false positive by the watermarking detection scheme.

Formally, suppose that selecting a "random" (i.e., unwatermarked) digital file is equivalent to drawing a random element X from a distribution μ on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$. For that vector X, let $\Phi(X)$ be a score that is large when a watermark is detected, i.e., the device considers the file as watermarked if $\Phi(X) > q$, where q is a fixed given threshold. Because of the complexity of many decoding schemes, we view Φ as a black box, that is, we do not have an analytic expression for Φ but we can readily evaluate $\Phi(X)$ for any given instance X. Then given a threshold q, we seek to es-

timate the probability of false alarm, defined as the tail probability $p = \mathbb{P}(\Phi(X) > q)$ when $X \sim \mu$.

Consequently, we are exactly in the abstract context of the previous chapter. Here, we apply our algorithm to a well-known watermarking detector for which there exists a closed form expression for the probability of false alarm. This allows us to benchmark our method. For this purpose, we have selected the absolute value of the normalized correlation as the score function Φ (see for example Merhav and Sabbag [87]), so that X is deemed watermarked whenever

$$\Phi(X) = \frac{|X^T u|}{\|X\|} > q,$$

where u is a secret but fixed unit vector, and X is a $d-$dimensional random vector with an unknown isotropic distribution. Given a threshold value q we would like to find the tail probability p.

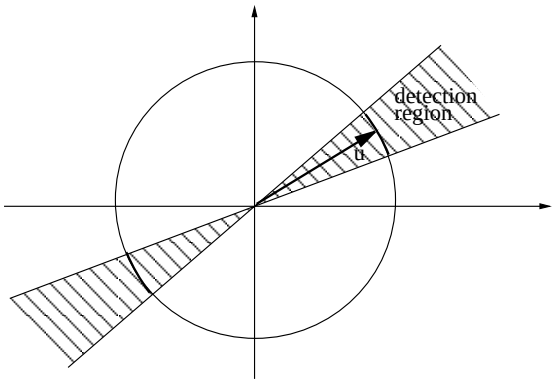

Figure 5.2: Detection region for zero-bit watermarking.

A geometrical interpretation shows that the acceptance region is a two-sheet hypercone (see figure 5.2) whose axis is given by u and whose angle is $\theta = \cos^{-1}(q)$ (with $0 < \theta < \pi/2$). Since X has an isotropic distribution, $X/\|X\|$ has the uniform law on the unit sphere in dimension d, so that any isotropic distribution makes the job to evaluate p. In the following, we propose to choose a standard Gaussian distribution: $X \sim \mathcal{N}(0, I_d)$. This allows us to derive explicit expressions for the probability of false positive detections to benchmark our algorithm. The following lemma describes the distribution of $\Phi(X)$.

Lemma 4 *Let us denote F the cdf of the random variable $Y = \Phi(X)$, G the cdf of a random variable following a Fisher-Snedecor distribution with $(1, d-1)$ degrees of freedom, f and g their respective pdf. Then for all q in $(0,1)$, we have*

$$p = \mathbb{P}(\Phi(X) > q) = 1 - F(q) = 1 - G\left(\frac{(d-1)q^2}{1-q^2}\right),$$

from which it follows that

$$f(q) = \frac{2(d-1)q}{(1-q^2)^2} \cdot g\left(\frac{(d-1)q^2}{1-q^2}\right).$$

In our simulations, we chose the following transition kernel for Gaussian random vectors on $\mathbb{R}^d$: Given a current location x, we propose the new position

$$X' = \frac{x + \sigma W}{\sqrt{1+\sigma^2}},$$

where W is a $\mathcal{N}(0, I_d)$ $\mathbb{R}^d$-valued random vector and σ a positive number. In the simulations, the dimension is $d = 20$, the number of proposed kernel transitions is $T = 20$, the numbers of particles are successively $N = 100, 200, 500, 1000, 5000$, and for each N we have run the algorithm 100 times in order to get boxplots, empirical relative standard deviations and confidence intervals. The choice $\sigma = 0.3$ has experimentally been proved to be a good trade-off for the "mixing" parameter.

Remark: The fact that we do not have to tune σ on the fly might seem quite surprising at first sight. Indeed, one could think that we should reduce it adaptively since we progressively focus on smaller and smaller hypercones. Anyway, since $d = 20$, the square of the distance between a particle and the origin is distributed according to a chi-square distribution χ^2_{20}, which is concentrated around its mean (i.e., 20). Thus, roughly speaking, the particles are concentrated around the hypersphere centered at the origin and with radius $\sqrt{20}$. Thus, if $\theta = \cos^{-1}(0.95)$, then even at the end of the algorithm the distance between the axis of the hypercone and its boundary is around 1.5: this is five times larger than the standard deviation $\sigma = 0.3$ of the Gaussian moves and explains that the rate of rejection does not dramatically increase with the iterations of the algorithm.

5.1.1 Estimation of p

For our illustrative example, we fix $q = 0.95$ and apply Lemma 4 to conclude that the probability of interest is approximately equal to $p = 4.704 \times 10^{-11}$. Estimating such a small probability by running a CMC algorithm is of course out of question. For this purpose, we have applied Algorithm 3 of the previous chapter. Figure 5.3 summarizes the results through boxplots for our method. As the number of particles increases, the distribution of the estimator concentrates around p.

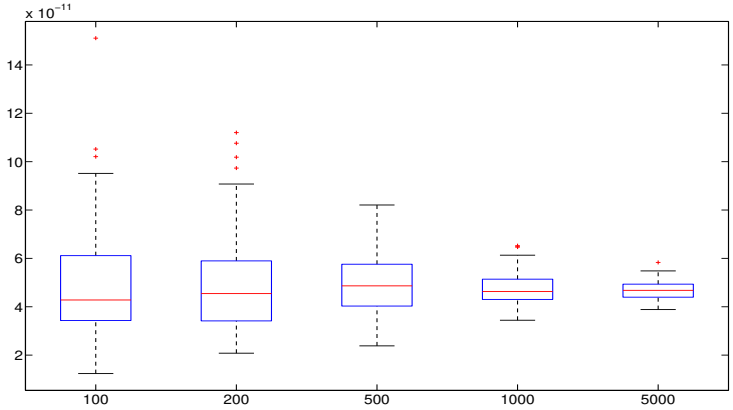

Figure 5.3: Boxplots for the estimation of p obtained with 100 simulations for $N = 100$ to $N = 5,000$ particles (Algorithm 3).

Figure 5.4 shows in log-log scales the theoretical and empirical relative standard deviations: the theoretical one is known thanks to Proposition 9, replacing p with the numerical value 4.704×10^{-11}, whereas the empirical one was estimated through 100 successive simulations. Let us recall that the theoretical relative standard deviations is namely

$$\frac{\sqrt{\mathrm{Var}(\hat{p})}}{p} = \sqrt{p^{-\frac{1}{N}} - 1} \approx \sqrt{\frac{-\log p}{N}},$$

the last approximation being valid when N is large enough, hence the slope equal to -0.5 on the

right hand of figure 5.4. One can notice the great coincidence between theory and practice on this example, that means between the idealized algorithm and its practical implementation.

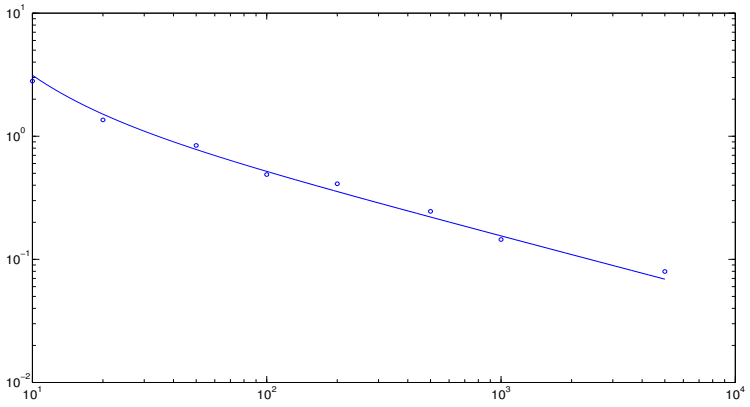

Figure 5.4: Theoretical and empirical relative standard deviations with 100 simulations for $N = 10$ to $N = 5,000$ particles (Algorithm 3).

To highlight the main differences between Algorithm 2 and Algorithm 3, we have also run Algorithm 2 on the same example and with exactly the same parameters, that means: $d = 20$, $T = 20$, $\sigma = 0.3$. When the proportion of particles surviving from one step to the next is fixed to p_0, Theorem 8 ensures that

$$\sqrt{N}\,\frac{\tilde{p} - p}{p} \xrightarrow[N \to +\infty]{\mathcal{D}} \mathcal{N}(0, \sigma^2),$$

where

$$\sigma^2 = n_0 \frac{1 - p_0}{p_0} + \frac{1 - r_0}{r_0} \quad \text{with } n_0 = \left\lfloor \frac{\log p}{\log p_0} \right\rfloor \text{ and } r_0 = p\, p_0^{-n_0}.$$

Taking $p_0 = 0.75$ for example, it follows that $n_0 = 82$ and $r_0 \approx 0.83$. In this case, the resulting relative standard deviation of the estimator $\tilde{p}$ is only slightly larger than the standard deviation of the estimator $\hat{p}$ since

$$\sqrt{n_0 \frac{1 - p_0}{p_0} + \frac{1 - r_0}{r_0}} \approx 1.66 \gtrapprox 1.58 \approx \sqrt{-\log p}.$$

One consequence is that Algorithm 3 requires a bit fewer particles to compute estimators for the tail probability with similar standard errors. But that is not the most interesting point. More important, Algorithm 3 gives the exact variance for as few as $N = 10$ particles, whereas the asymptotic variance of Algorithm 2 is reached only for $N \geq 500$. This is illustrated in Figures 5.4 and 5.5 that graph the estimated standard deviation as a function of the number of particle (dots) for both methods, and compares it with the theoretical lower bound (line).

As a consequence, Algorithm 3 enables us to draw confidence intervals even with a low number of particles, which, on this specific example, is only possible for $N \geq 500$ with Algorithm 2. In this respect, for $N = 100$ particles, figure 5.6 illustrates the 95% confidence intervals obtained in equation (4.1), i.e. with Algorithm 3.

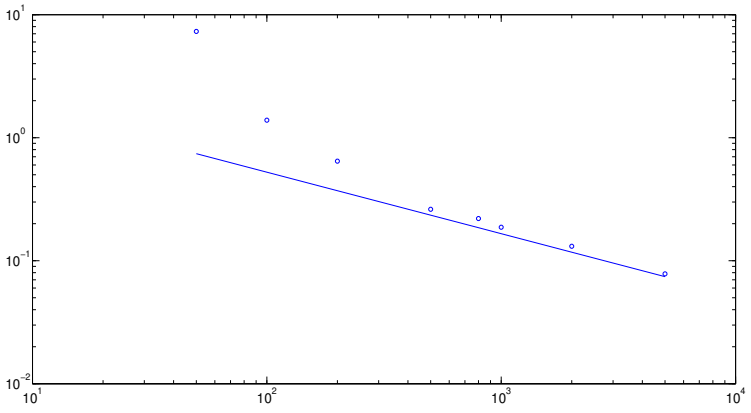

Figure 5.5: Theoretical and empirical relative standard deviations with 100 simulations for $N = 50$ to $N = 5,000$ particles (Algorithm 2).

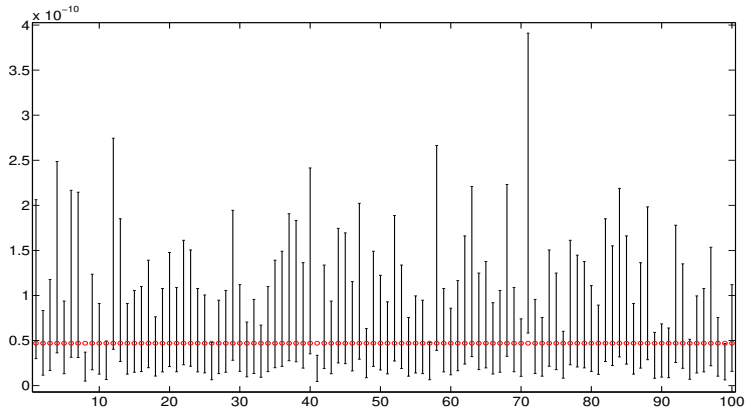

Figure 5.6: 95% confidence intervals for $p = 4.704 \cdot 10^{-11}$ with 100 simulations and $N = 100$ particles (Algorithm 3).

5.1.2 Estimation of q

Conversely, suppose that we fix $p = 4.704 \times 10^{-11}$ and seek to use Algorithm 4 to estimate its associated tail quantile. We know that the theoretical value is $q = 0.95$. Figure 5.7 summarizes the results through boxplots.

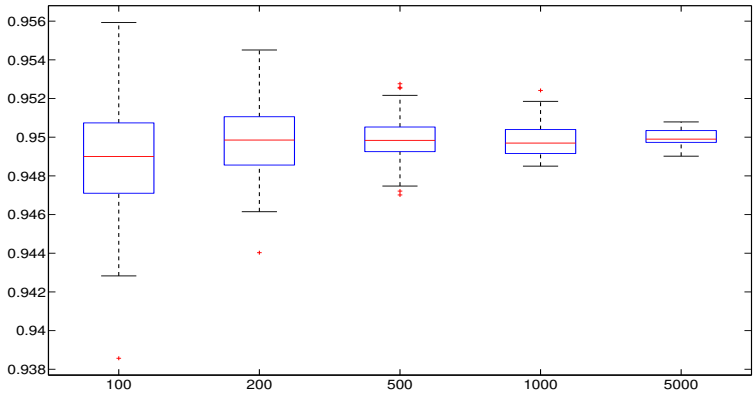

Figure 5.7: Boxplots for the estimation of q obtained with 100 simulations for $N = 100$ to $N = 5,000$ particles (Algorithm 4).

Figure 5.8 shows in log-log scales empirical and theoretical relative standard deviations: these last ones are known thanks to Proposition 11, replacing p with the numerical value 4.704×10^{-11} and $f(q)$ by the second formula of Lemma 4. The empirical standard deviation was estimated through 100 successive simulations of the algorithm.

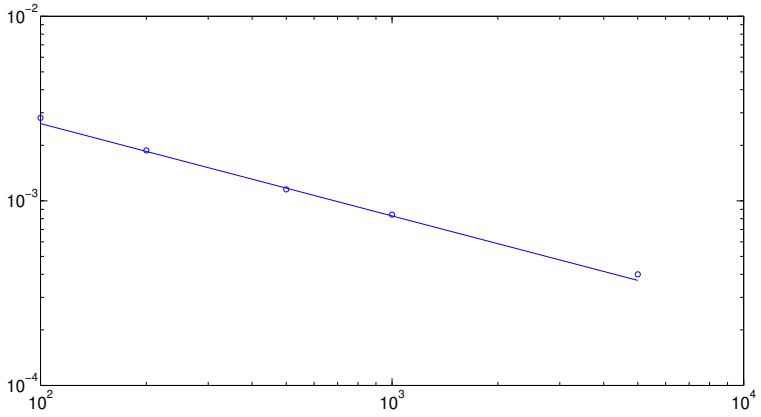

Figure 5.8: Theoretical and empirical relative standard deviations with 100 simulations for $N = 100$ to $N = 5,000$ particles (Algorithm 4).

Once again, one can notice the great coincidence between theory and practice on this example. Finally, figure 5.9 illustrates the 95% confidence intervals obtained in Proposition 13 for $N = 100$ particles.

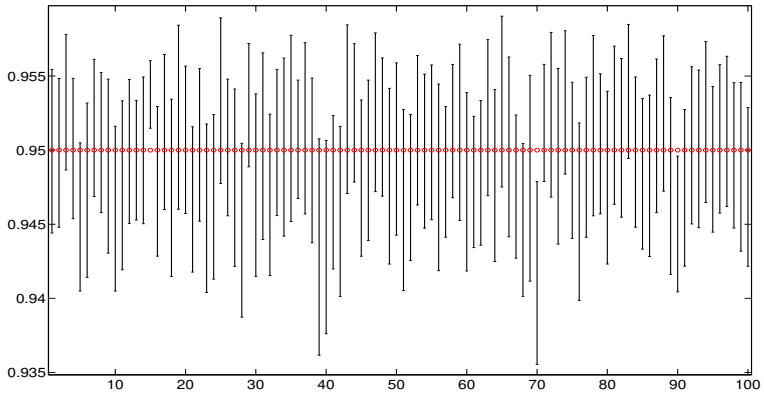

Figure 5.9: 95% confidence intervals for $q = 0.95$ with 100 simulations and $N = 100$ particles.

5.2 Fingerprinting

In this application, users' identifiers are embedded in a purchased content. When this content is found in an illegal place (e.g. a P2P network), the right holders decode the hidden message, find a serial number, and thus they can trace the traitor, i.e. the customer who has illegally broadcast his copy. However, the task is not that simple because dishonest users might collude. For security reason, anti-collusion codes have to be employed. Yet, these solutions (also called weak traceability codes, see for example Barg, Blakley and Kabatiansky [10]) have a non-zero probability of error, defined as the probability of accusing an innocent. This probability should be, of course, extremely low, but it is also a very sensitive parameter: in terms of the number of bits to be hidden in the host content, anti-collusion codes get longer as the probability of error decreases.

Consequently, fingerprint designers have to strike a trade-off, which is hard to conceive when only a rough estimation of the probability of error is known. The major issue for fingerprinting algorithms is the fact that embedding large sequences implies also assessing reliability on a huge amount of data which may be practically unachievable without using rare event analysis.

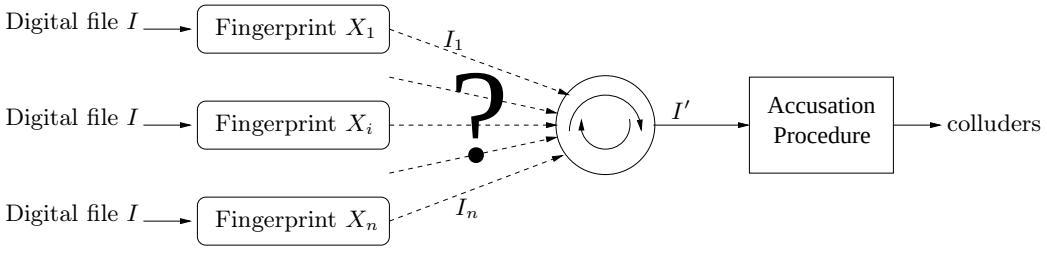

Figure 5.10: Schematic representation of a fingerprinting process.

In other words, fingerprinting is the application where a content server gives personal copies of the same content to n different buyers. c of them are dishonest users, called colluders, who mix their copies to yield a pirated content. A binary fingerprinting code is a set of n different m bit sequences $\{X_i\}_{1 \leq i \leq n}$. Each sequence identifying a user has to be hidden in the personal copy with a watermarking technique. When a pirated copy is found, the server retrieves a m bit sequence

and accuses some users or nobody (see figure 5.10). There are two kinds of errors: accusing an innocent (a false alarm) and accusing none of the colluders (a false negative). The designers of the fingerprinting code must assess the minimum length of the code so that the probabilities of error are below some significance levels: $P_{fa} < \epsilon_1$ and $P_{fn} < \epsilon_2$.

One of the best fingerprinting codes is a probabilistic code proposed by Tardos [111], where $m = O(c^2 \log \frac{1}{\epsilon_1})$. Before Tardos' work, the existence of such a short code was only theoretically proved. Tardos is the first to exhibit a construction which is, moreover, surprisingly simple. The main point of interest for us is that the accusation is based on the calculus of scores and their comparison to a threshold (see figure 5.11 for a schematic picture). Consequently, this fingerprinting code is very well suited with respect to our algorithm.

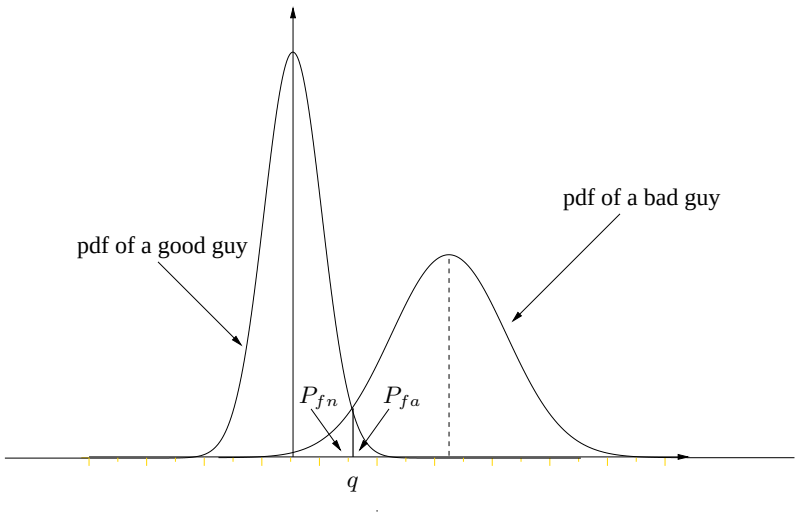

Figure 5.11: Accusation strategy of Tardos fingerprinting code.

5.2.1 New Accusation Strategy

In Tardos probabilistic fingerprinting code, the accusation is focused: The detection decides whether a given user is guilty or not. It calculates his score from the code sequence of the user and the sequence y recovered in the pirated copy. The user is deemed guilty when his score is higher than a threshold q. The size of the collusion c, the probabilities ϵ_1 and ϵ_2 are the inputs of the code. The outputs are the code length m and the value of the threshold q.

We think that this approach is not adapted in practice. We believe that the length of the code sequence to be embedded in content is not tunable but fixed by the payload of the watermarking technique and the length of the content. It is clear that the longer the sequence, the better the accusation process. But, in practice, there is certainly a wide range in the length of the sequences to be embedded due to a wide diversity of contents. In the same way, it might be complicated to derive the right value of the threshold for different sequence lengths.

We propose a different approach. Once we have recovered the sequence y in the pirated copy, we calculate all the scores of the users to which the content has been delivered and accuse the most likely guilty users, i.e. the ones with the highest scores. In the sequel, consider that user j is accused because he has the biggest score. There is no longer need of a threshold. However, we

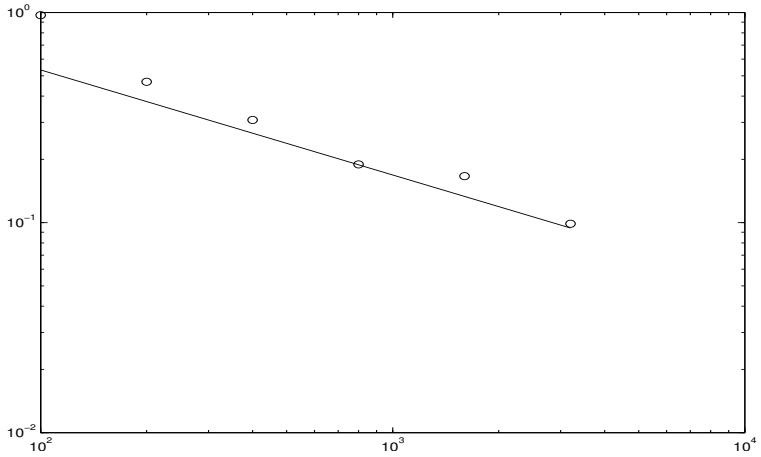

Figure 5.12: Theoretical and empirical relative standard deviations with 100 simulations for an example of Tardos code.

cannot guaranty a probability ϵ_1. To be fair, the output of our accusation process is the index j of the most likely guilty user associated with the probability of making an error, i.e. the probability that an innocent gets a larger score than the one of user j.

5.2.2 Accusing an Innocent

We are interested here in embedding an identifier in each copy of a purchased content. Then a pirate copy, which is the result of a collusion, is found on the web, and we want to decide whether or not it can be originated from a certain user. The rare event will be to consider an innocent as guilty.

The embedded message, called a fingerprint, consists of a sequence of bits $X = (X_1, \ldots, X_m)$, where each X_i is independent from the others, and drawn from a Bernoulli's $\mathcal{B}(p_i)$. The p_i's are themselves i.i.d. random variables, drawn from an arcsine distribution on $[0, 1]$. Then we find a pirate copy with fingerprint $y = (y_1, \ldots, y_m) \in \{0, 1\}^m$. Then for each user, with generic fingerprint $X = (X_1, \ldots, X_m)$, we compute his score $\Phi(X)$

$$\Phi(X) = \sum_{i=1}^{m} y_i \; g_i(X_i),$$

where the functions g_i's are defined as follows:

$$g_i(X_i) = \sqrt{\frac{1-p_i}{p_i}} \mathbb{1}_{\{X_i=1\}} - \sqrt{\frac{p_i}{1-p_i}} \mathbb{1}_{\{X_i=0\}}$$

This approach was proposed by Tardos in [111]. We refer the interested reader to Cérou, Furon and Guyader [26] to understand why the choices of the arcsine distribution for the p_i's and of these specific forms for the g_i's are optimal in some sense.

To apply our algorithm, we need to choose the kernel K. As the X_i's are independent, we randomly choose r indices $\{j_1, \ldots, j_r\} \in \{1, \ldots, m\}$, with r being a fixed parameter. Then for each j_ℓ, we draw a new X'_{j_ℓ} independently from the Bernoulli distribution $\mathcal{B}(p_{j_\ell})$.

For such codes, we first present the equivalent of Figure 5.5 on Figure 5.12. We consider the probability of accusing an innocent using a code of length $m = 200$. In this first experiment, the

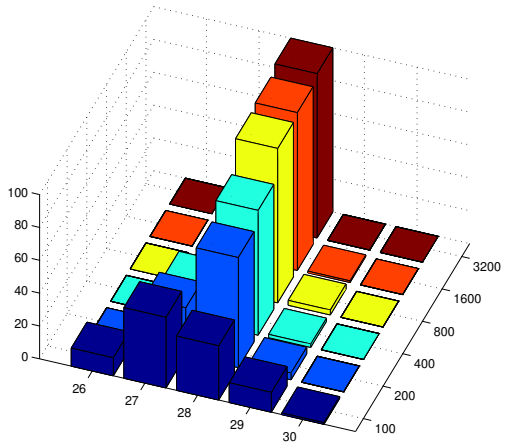

Figure 5.13: Distribution of the number of steps.

pirate fingerprint y is fixed and is an input of the algorithm. Algorithm 2 was run with $p_0 = 1/2$. As in the watermarking example of section 5.1, the transition kernel K was applied $T = 20$ times, with $r = 20$. As we do not have any other estimates on the rare event probability, we just plugged the mean of the estimates given by the runs of our algorithm with the largest number of particles ($N = 3,200$) in the theoretical variance given by Theorem 8. This best estimate on the probability of the rare event is 2.6×10^{-9}. Again, we see that the performance of the algorithm is close to that of the idealized version. Figure 5.13 shows the distribution of the number of steps as a function of the number of particles. We can see that for 800 particles and more, the number of steps can be seen practically as deterministic.

These results illustrate the efficiency of our algorithm on a discrete problem. It is indeed noticeable that the coincidence between theory and practice remains true even if in this case the continuity assumption on the cdf F of $\Phi(X)$ of section 4.3.2 is clearly not fulfilled: $\Phi(X)$ is here a discrete random variable. However, one can argue that in this precise setting $\Phi(X)$ can take a huge number of values, namely 2^m, so that it can almost be considered as continuous. When this is not possible, we can sometimes adapt our algorithm by embedding the discrete problem at hand in a continuous space. We will present this idea in section 5.3 and illustrate it on a counting problem.

5.2.3 Accusing None of the Colluders

Using our adaptive algorithm, we made some additional numerical experiments on such codes. More precisely, we can easily estimate the probability of false detection (false positive) for some code length m, and collusion size c. The collusion strategy is to randomly pick up the symbols of pirated copy among the c colluders' sequences. We can also estimate the probability of not accusing someone guilty (false negative). The results for $m = 200$, and $c = 2, 3, 4$ are shown on Figure 5.14. The parabolic form of the curves (in logarithmic scale on the y axis) stems from the definition of $\Phi(X)$ as a sum of i.i.d. random variables and the CLT phenomenon. However, from these curves, one can then decide how to set the threshold q to minimize the total error.

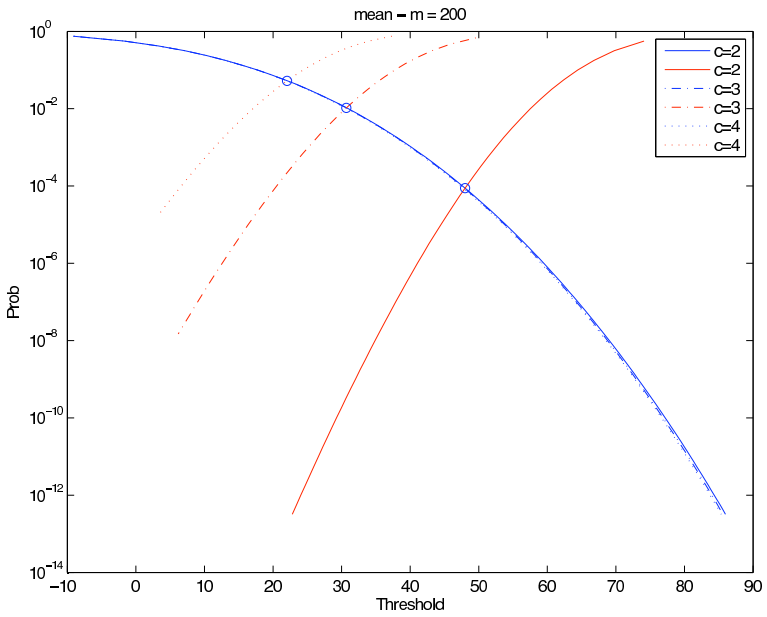

Figure 5.14: Mappings of the false positive probability (blue) and false negative probability (red) against the threshold. The score of a particle is the mean of the c colluders scores.

5.3 Counting

The goal of this work is to propose a novel and original way, called the *smoothed splitting method* (SSM), for counting on discrete sets associated with NP-hard discrete combinatorial problems and in particular counting the number of satisfiability assignments. The main idea of the SSM is to transform a combinatorial counting problem, so a discrete issue, into a continuous one using a type of "smoothing" of discrete indicator functions. Then we are in a position to apply a quite standard multilevel splitting method, as described for example in section 4.3, to this continuous integration problem.

Before proceeding with SSM we present the splitting method for counting, following Rubinstein [101, 102]. The main idea of the splitting method for counting is to design a sequential sampling plan, with a view of decomposing a "difficult" counting problem defined on some set $\mathcal{X}^*$ into a number of "easy" ones associated with a sequence of related nested sets $\mathcal{X}_0, \mathcal{X}_1, \ldots, \mathcal{X}_T$ and such that $\mathcal{X}_T = \mathcal{X}^*$. Similar to *randomized algorithms* [89, 90], splitting algorithms explore the connection between counting and sampling problems and in particular the reduction from approximate counting of a discrete set to approximate sampling of elements of this set.

A typical splitting algorithm comprises the following steps (see also figure 5.15):

1. Formulate the counting problem as that of estimating the cardinality $|\mathcal{X}^*|$ of some set $\mathcal{X}^*$.

2. Find a sequence of sets $\mathcal{X} = \mathcal{X}_0, \mathcal{X}_1, \ldots, \mathcal{X}_T$ such that $\mathcal{X}_0 \supset \mathcal{X}_1 \supset \cdots \supset \mathcal{X}_T = \mathcal{X}^*$, and $|\mathcal{X}| = |\mathcal{X}_0|$ is known.

3. Write $|\mathcal{X}^*| = |\mathcal{X}_T|$ as

$$|\mathcal{X}^*| = |\mathcal{X}_0| \prod_{t=1}^{T} \frac{|\mathcal{X}_t|}{|\mathcal{X}_{t-1}|} = |\mathcal{X}_0| p, \tag{5.1}$$

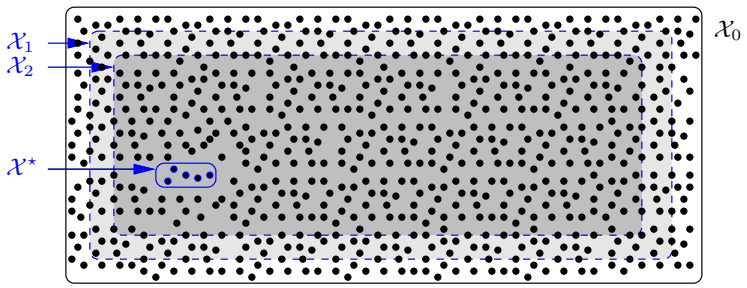

Figure 5.15: Nested sets $\mathcal{X}_0 \supset \mathcal{X}_1 \supset \cdots \supset \mathcal{X}_T = \mathcal{X}^*$ for counting.

where $p = \prod_{t=1}^{T} \frac{|\mathcal{X}_t|}{|\mathcal{X}_{t-1}|}$. Note that p is typically very small, like $p = 10^{-100}$, while each ratio

$$p_t = \frac{|\mathcal{X}_t|}{|\mathcal{X}_{t-1}|}$$

should not be small, like $p_t = 10^{-1}$ or bigger. Clearly, estimating p directly while sampling in $\mathcal{X}_0$ is meaningless, but estimating each p_t separately seems to be a good alternative.

4. Develop an efficient estimator $\widetilde{p}_t$ for each p_t and estimate $|\mathcal{X}^*|$ by

$$\widetilde{|\mathcal{X}^*|} = |\mathcal{X}_0|\,\widetilde{p} = |\mathcal{X}_0| \prod_{t=1}^{T} \widetilde{p}_t,$$

where $\widetilde{p} = \prod_{t=1}^{T} \widetilde{p}_t$ is an estimator of $p = \prod_{t=1}^{T} \frac{|\mathcal{X}_t|}{|\mathcal{X}_{t-1}|}$.

It is readily seen that in order to obtain a meaningful estimator of $|\mathcal{X}^*|$, we have to resolve the following two major problems:

(i) Put the NP-hard counting problem into the framework (5.1) by making sure that $\mathcal{X}_0 \supset \mathcal{X}_1 \supset \cdots \supset \mathcal{X}_T = \mathcal{X}^*$ and each p_t is not a rare event probability.

(ii) Obtain a low variance estimator $\widetilde{p}_t$ of each $p_t = |\mathcal{X}_t|/|\mathcal{X}_{t-1}|$.

In Section 5.3.1, we briefly recall the SAT problem, which we will focus on in order to present our new method. In Section 5.3.2, we show how to resolve problems (i) and (ii) for the SAT problem by using the smoothed splitting method (SSM), which presents an enhanced version of the splitting method of Rubinstein [101, 102]. Some remarks and comments are gathered in Section 5.3.3. We report the reader to the article [29] for the presentation of numerical results.

5.3.1 Presentation of the SAT Problem

The most common SAT problem comprises the following two components:

- A set of n Boolean variables $\{x_1, \ldots, x_n\}$, representing statements that can either be `TRUE` (=1) or `FALSE` (=0). The negation (the logical `NOT`) of a variable x is denoted by $\overline{x}$. For example, $\overline{\texttt{TRUE}} = \texttt{FALSE}$. A variable or its negation is called a *literal*.

- A set of m distinct *clauses* $\{S_1, S_2, \ldots, S_m\}$ of the form $S_j = z_{j_1} \vee z_{j_2} \vee \cdots \vee z_{j_q}$, where the z's are literals and the $\vee$ denotes the logical `OR` operator. For example, $0 \vee 1 = 1$.

The binary vector $\mathbf{x} = (x_1, \ldots, x_n)$ is called a *truth assignment*, or simply an *assignment*. Thus, $x_i = 1$ assigns truth to x_i and $x_i = 0$ assigns truth to $\overline{x}_i$, for each $i = 1, \ldots, n$. The simplest SAT problem can now be formulated as: find a truth assignment $\mathbf{x}$ such that *all* clauses are true.

Denoting the logical `AND` operator by $\wedge$, we can represent the above SAT problem via a single formula as

$$F = S_1 \wedge S_2 \wedge \cdots \wedge S_m,$$

where the S_j's consist of literals connected with only $\vee$ operators. The SAT formula is then said to be in conjunctive normal form (CNF). The problem of deciding whether there exists a valid assignment, and, indeed, providing such a vector, is called the *SAT-assignment* problem.

Toy Example: Let us consider the following toy SAT problem with two clauses and two variables: $(x_1 \vee x_2) \wedge (\bar{x}_1 \vee x_2)$. It is straightforward by considering all the four possible assignments, that this formula is satisfiable, with two valid assignments $x_1 = 1, x_2 = 1$ and $x_1 = 0, x_2 = 1$. If now we consider the three clauses formula $(x_1 \vee x_2) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_2)$, then it is clearly unsatisfiable.

Note that the SAT-assignment problem can be modeled via rare events with p given by

$$p = \mathbb{P}\left(\sum_{j=1}^{m} C_j(\mathbf{X}) = m \right), \tag{5.2}$$

where $\mathbf{X}$ has a uniform distribution on the finite set $\{0,1\}^n$, and

$$C_j(\mathbf{x}) = \max_{1 \le k \le n} \{0,\ (2\,x_k - 1)\,a_{jk}\}.$$

Here $C_j(\mathbf{x}) = 1$ if clause S_j is `TRUE` with truth assignment $\mathbf{x}$ and $C_j(\mathbf{x}) = 0$ if it is `FALSE`, $A = (a_{jk})$ is an $m \times n$ given clause matrix that indicates if the literal corresponds to the variable $(+1)$, its negation (-1), or that neither appears in the clause (0). If for example $x_k = 0$ and $a_{jk} = -1$, then the literal $\overline{x}_j$ is `TRUE`. The entire clause is `TRUE` if it contains at least one true literal. In other words, p in (5.2) is the probability that a uniformly generated SAT assignment $\mathbf{X}$ is valid, that is, all clauses are satisfied. Denoting

$$S(\mathbf{X}) = \min_{1 \le j \le m} C_j(\mathbf{X}),$$

we want to estimate $p = \mathbb{P}(S(\mathbf{X}) = 1)$, which is typically very small.

5.3.2 Smoothed Splitting Method

Before presenting the SSM algorithm we shall discuss its main features having in mind a SAT problem. To proceed, recall that the main idea of SSM is to work within a continuous space rather than a discrete one. As a result this involves a continuous random vector $\mathbf{Y}$ instead of the discrete random vector $\mathbf{X}$ with i.i.d. components $X_1, \ldots, X_n$ with law $\mathrm{Ber}(p = 1/2)$. For example for a SAT problem, one needs to adopt the following steps:

1. Choose a random vector $\mathbf{Y}$ of the same size as $\mathbf{X}$, such that the components $Y_1, \ldots, Y_n$ are i.i.d. uniformly distributed on the interval $(0,1)$. Clearly the Bernoulli components $X_1, \ldots, X_n$ can be written as $X_1 = \mathbb{1}_{\{Y_1 > 1/2\}}, \ldots, X_n = \mathbb{1}_{\{Y_n > 1/2\}}$.

2. Instead of the former $0-1$ variables x or $\bar{x}$ we will use for each clause a family of functions from $(0,1)$ to $(0,1)$. In particular, for each occurrence of x or $\bar{x}$, we consider two functions,

say $g_\varepsilon(y)$ and $h_\varepsilon(y) = g_\varepsilon(1-y)$ indexed by $\varepsilon \geq 0$. These functions need to be increasing in ε, which means that

$$0 < \varepsilon \leq \varepsilon' \;\Rightarrow\; g_\varepsilon(y) \leq g_{\varepsilon'}(y),\ \forall y \in (0,1).$$

and for $\varepsilon = 0$, $g_0(y) = \mathbb{1}_{\{y>1/2\}}$, $h_0(y) = g_0(1-y) = \mathbb{1}_{\{y\leq 1/2\}}$. Possible choices of $g_\varepsilon(y)$ are:

$$g_\varepsilon(y) = (2y)^{1/\varepsilon}\mathbb{1}_{\{0<y<\frac{1}{2}\}} + \mathbb{1}_{\{y>\frac{1}{2}\}} \tag{5.3}$$

or

$$g_\varepsilon(y) = \mathbb{1}_{\{\frac{1}{2}-\varepsilon<y<\frac{1}{2}\}}\left(\frac{y}{\varepsilon} + 1 - \frac{1}{2\varepsilon}\right) + \mathbb{1}_{\{y>\frac{1}{2}\}}. \tag{5.4}$$

or (see figure 5.16)

$$g_\varepsilon(y) = \mathbb{1}_{[1/2-\varepsilon,1]}(y). \tag{5.5}$$

3. For each clause C_j, we consider the approximate ε-clause $C_{j\varepsilon}$, where we replace x by $g_\varepsilon(y)$, $\bar{x}$ by $h_\varepsilon(y)$, and $\vee$ by $+$. Note also that the statement "$C_j(\mathbf{x})$ is true", i.e. $C_j(\mathbf{x}) = 1$, is replaced in the new notations by "$C_{j\varepsilon}(\mathbf{y}) \geq 1$". As a consequence, denoting

$$S_\varepsilon(\mathbf{y}) = \min_{1\leq j\leq m} C_{j\varepsilon}(\mathbf{y}),$$

the event $\{S(\mathbf{X}) = 1\}$ is replaced by the event $\{S_\varepsilon(\mathbf{Y}) \geq 1\}$.

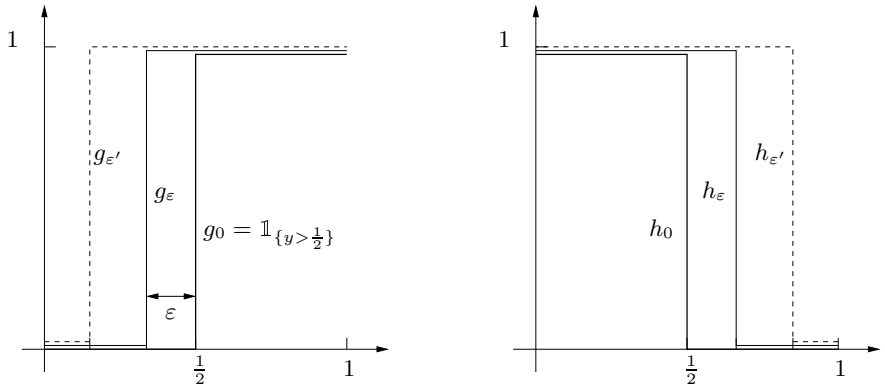

Figure 5.16: The families of functions g_ε and h_ε of (5.5).

4. **Nested sets**. For each $\varepsilon \geq 0$, consider the subset (or event) B_ε of $(0,1)^n$ defined as

$$B_\varepsilon = \{\mathbf{y} \in (0,1)^n :\ \forall j \in \{1,\ldots,m\}, C_{j\varepsilon}(\mathbf{y}) \geq 1\} = \{\mathbf{y} \in (0,1)^n :\ S_\varepsilon(\mathbf{y}) \geq 1\}.$$

Then it is clear from the above that for $\varepsilon_1 \geq \varepsilon_2 \geq 0$, we have the inclusions $B_0 \subset B_{\varepsilon_2} \subset B_{\varepsilon_1}$. Note that B_0 is the event for which *all* the *original* clauses are satisfied and B_ε is an event on which *all* the *approximate* ε-clauses are satisfied. Note also that $\varepsilon_t,\ t = 1,\ldots,T$, should be a decreasing sequence, with T being the number of nested sets, and $\varepsilon_T = 0$. In our SSM algorithm below, we shall choose the sequence $\varepsilon_t,\ t = 1,\ldots,T$, adaptively.

The SSM Algorithm

We are now in a position to describe the smoothed splitting algorithm. Algorithm 2' below is an adaptation of Algorithm 2 in this context, but adapting Algorithm 3 would make the job as well. Just note that the score function here is defined for any particle $\mathbf{y} \in (0,1)^n$ as

$$\Phi(\mathbf{y}) = \max_{1\leq j\leq m} \inf\{\varepsilon : C_{j\varepsilon}(\mathbf{y}) \geq 1\}.$$

Thus, given $\mathbf{y}$, the derivation of $\Phi(\mathbf{y})$ just uses the pseudo-inverses of g_ε and h_ε.

Algorithm 2'

Parameters

N the number of particles, the number $N_0 < N$ of succeeding particles, and let $p_0 = N_0/N$.

Initialization

Draw an i.i.d. N-sample $(\mathbf{Y}_0^j)_{1\leq j\leq N}$ of the law $\mathcal{U}((0,1)^n)$.

Compute $\tilde{\varepsilon}_1$, the $(1-p_0)$ quantile of $\Phi(\mathbf{Y}_0^j), j = 1,\ldots,N$.

$k = 1$;

Iterations

while $\tilde{\varepsilon}_k > 0$ do

Starting from an i.i.d. p_0N-sample with uniform law on B_{ε_k} draw an i.i.d. N-sample $(\mathbf{Y}_k^j)_{1\leq j\leq N}$ with the same law (see Gibbs sampling below).

Compute $\tilde{\varepsilon}_{k+1}$, the $(1-p_0)$ quantile of $\Phi(\mathbf{Y}_k^j), j = 1,\ldots,N$.

$k = k+1$;

endwhile

Let $\tilde{r}_0$ the proportion of particles $\mathbf{Y}_{k-1}^j$, $j = 1,\ldots,N$, such that $\Phi(\mathbf{Y}_{k-1}^j) = 0$.

Output

Estimate the probability of the rare event by $\tilde{p} = \tilde{r}_0\, p_0^{k-1}$.

At the end of the line, we obtain a set of $\tilde{r}_0 N$ non necessarily different solutions of the original SAT problem by a simple rounding operation: for each $\mathbf{Y} = (Y_1,\ldots,Y_n)$ such that $\Phi(\mathbf{Y}) = 0$, $\mathbf{X} = (\mathbb{1}_{\{Y_1>1/2\}},\ldots,\mathbb{1}_{\{Y_n>1/2\}})$ is an assignment such that all the clauses are true.

Gibbs Sampler

Starting from $\mathbf{Y} = (Y_1,\ldots,Y_n)$, which is uniformly distributed on

$$B_\varepsilon = \{\mathbf{y}\in(0,1)^n :\ \forall j\in\{1,\ldots,m\}, C_{j\varepsilon}(\mathbf{y})\geq 1\} = \{\mathbf{y}\in(0,1)^n :\ S_\varepsilon(\mathbf{y})\geq 1\},$$

a possible way to generate $\widetilde{\mathbf{Y}}$ with the same law as $\mathbf{Y}$ is to use the following general systematic Gibbs sampler (g is the target distribution.):

1. Draw $\widetilde{Y}_1$ from the conditional pdf $g(y_1|y_2,\ldots,y_n)$.
2. Draw $\widetilde{Y}_k$ from the conditional pdf $g(y_k|\widetilde{y}_1,\ldots,\widetilde{y}_{k-1},y_{k+1},\ldots,y_n)$, $2\leq k\leq n-1$.
3. Draw $\widetilde{Y}_n$ from the conditional pdf $g(y_n|\widetilde{y}_1,\ldots,\widetilde{y}_{n-1})$.

In our case, g is the uniform distribution on B_ε, and the conditional distribution of the kth component given the others is simply the uniform distribution on some interval (r, R) given as explained on the following toy example.

Toy Example: Let us consider a small example with four variables and two clauses: $(X_1 \vee X_2) \wedge (\bar{X}_1 \vee X_3 \vee \bar{X}_4)$. For a given $\varepsilon > 0$, this gives the two ε-clauses:

$$\begin{aligned} g_\varepsilon(Y_1) + g_\varepsilon(Y_2) &\geq 1 \\ h_\varepsilon(Y_1) + g_\varepsilon(Y_3) + h_\varepsilon(Y_4) &\geq 1. \end{aligned}$$

Let us say we want the distribution of Y_1 given Y_2, Y_3, Y_4. If we want the first one to be satisfied, we need $g_\varepsilon(Y_1) \geq 1 - g_\varepsilon(Y_2)$, that is $Y_1 \geq g_\varepsilon^{-1}(1 - g_\varepsilon(Y_2)) = r$. Similarly, the second clause gives $h_\varepsilon(Y_1) \geq 1 - g_\varepsilon(Y_3) - h_\varepsilon(Y_4)$, and because h_ε is decreasing, $Y_1 \leq h_\varepsilon^{-1}(1 - g_\varepsilon(Y_3) - h_\varepsilon(Y_4)) = 1 - g_\varepsilon^{-1}(1 - g_\varepsilon(Y_3) - h_\varepsilon(Y_4)) = R$. Thus the conditional distribution of Y_1 is uniform on the interval (r, R). The generalization is straightforward.

It is readily seen that $r < R$ and $\widetilde{\mathbf{Y}} = (\widetilde{Y}_1, \ldots, \widetilde{Y}_n)$ has the same distribution as $\mathbf{Y}$. This is so since the initial point $\mathbf{Y} = (Y_1, \ldots, Y_n)$ belongs to and is uniformly distributed in B_ε. Note that our simulation results in Cérou, Guyader, Rubinstein and Vaisman [29] clearly indicate that one round of the Gibbs sampler suffices for good experimental results. Nonetheless, if one wants the new vector $\widetilde{\mathbf{Y}}$ to be independent of its initial position $\mathbf{Y}$, then in theory the Gibbs sampler would have to be applied an infinite number of times. As before, we could call it the *idealized* SSM. In this case, the idealized SSM estimator inherits the variance and bias results from Theorem 8 and Proposition 8 of section 4.3.

5.3.3 Remarks and Comments

Estimate of the Rare Event Cardinality

The previous discussion focused on the estimation of the rare event probability, which in turn provides an estimate of the actual number of solutions to the original SAT problem by taking $\widetilde{|\mathcal{X}^*|} = 2^n \, \tilde{p}$. In fact, the number of solutions may be small and thus can be determined by counting the different instances in the last sample of the algorithm. This estimator will be denoted by $\widetilde{|\mathcal{X}^*_{dir}|}$. Clearly $\widetilde{|\mathcal{X}^*_{dir}|}$ underestimates the true number of solutions $|\mathcal{X}^*|$, but at the same time it has a smaller (empirical) variance than $\widetilde{|\mathcal{X}^*|}$. Even if we do not know its mathematical properties, this estimate can be useful. Indeed, it may be interesting for practical purposes to know the set (and the number) of all the different solutions that have been found for the original SAT problem.

Mixing Properties

Our purpose here is to explain why the Gibbs sampler used at each step of the algorithm is irreducible and globally reaching and hence has good mixing properties. For the sake of clarity, we will focus first on g_ε as per (5.5). With this function, for a given ε, we can split the region explored by the Gibbs sampler in several small (sub) hypercubes or hyperrectangles, as shown schematically in Figure 5.17. To each vertex of the whole hypercube $(0, 1)^n$ that represents a solution of the original SAT problem, corresponds a sub-hypercube of edge length $1/2 + \varepsilon$, including the central point with coordinates $(1/2, \ldots, 1/2)$. And around this point, we have a sub-hypercube of edge length 2ε, which is common to all those elements.

For the other parts of the domain, which do not correspond to a solution, things become a bit more complicated. It is a union of ε-thin "fingers" extending outwards in several directions (a

subspace). The corresponding sub-domain being explored depends on the minimum number of variables that need to be taken in $(1/2-\varepsilon, 1/2+\varepsilon)$ in order to satisfy all the ε-clauses. The domain is then a rectangle of length $1/2+\varepsilon$ on the "free" variables, and of length 2ε in the other directions, that is on the $(1/2-\varepsilon, 1/2+\varepsilon)$ constrained variables. Again, all those rectangles include the small central sub-hypercube. The union of all these sub-hypercubes/rectangles is the domain currently explored by the Gibbs sampler. The geometry of the whole domain is then quite complex.

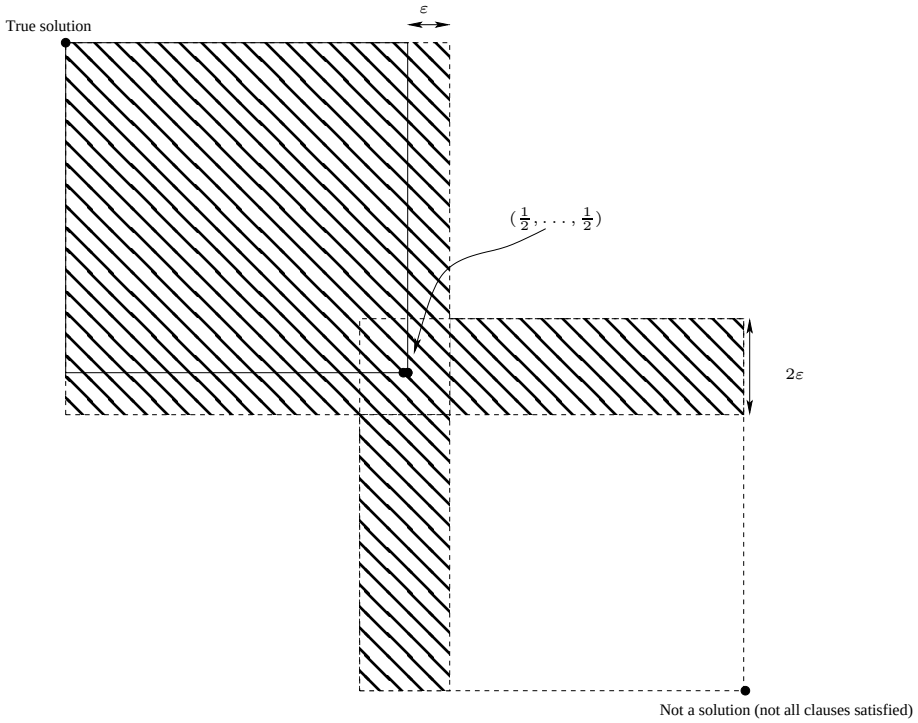

Figure 5.17: Partial mixing of the Gibbs sampler.

It is clear that starting with any one of these sub-hypercubes/rectangles we can reach any other point within it in one iteration of the Gibbs sampler. Moreover, as long as the Markov chain stays within the same sub-hypercube/rectangle, any other point is accessed with uniform probability. This means that the mixing properties of our Gibbs sampler are the best possible as long as *we are restricted to one sub-hypercube*. Actually this suffices to make the algorithm work.

To see this consider the particles at the beginning: after the first cloning, they are all in the central sub-hypercube with a high probability. Then at each step of the algorithm, the above mentioned "fingers" become thinner and thinner. Moreover, given that a replica is in one of these, it has a higher probability of being discarded when the "finger" is thin, that is the number of variables constrained to be in $(1/2-\varepsilon, 1/2+\varepsilon)$ is large. On the other hand, the replicas that satisfy a large number of clauses are favored. In the end, all the replicas hopefully find their way to a sub-hypercube corresponding to a true solution of the original SAT problem.

For g_ε as per (5.3) or (5.4), the same picture mostly holds, but the mixing properties within each sub-hypercube are not that easy to analyze. This is somehow compensated by an ability to deal with the inter-variable relations: the geometry of the domain explored around the centre point reflects these constraints, and thus has a much more complicated shape. These g_ε functions work in practice better than (5.5).

Walksat Algorithms

We truly acknowledge that our algorithm is in fact much slower than the best available stochastic algorithms to solve SAT problems, for example *Walksat* proposed by Selman, Kautz and Cohen [105]. Nonetheless, we would like to emphasize that:

1. our algorithm is more general, which means that *mutatis mutandis*, our method can be applied to other counting problems;
2. it finds almost all the solutions (when there are many) at once;
3. moreover, when there are so many solutions that we cannot find all of them in one run, we can still have an estimate of the total number.

Chapter 6

Application to Molecular Dynamics

6.1 Introduction

The aim of molecular dynamics computations is to evaluate macroscopic quantities from models at the microscopic scale. These can be:

1. thermodynamics quantities (stress, heat capacity, free energy), which imply averages of some observable with respect to an equilibrium measure;
2. dynamical quantities (diffusion coefficients, viscosity, transition rates), which imply averages over trajectories at equilibrium;

Molecular dynamics computations have many applications in various fields (biology, physics, chemistry, materials sciences, etc.), but they consume today a lot of CPU time. Formally, a molecular dynamics model is specified through a potential function V, which associates an energy $V(x_1, \ldots, x_d)$ to a configuration $(x_1, \ldots, x_d)$. Let us consider, to fix ideas, overdamped Langevin dynamics:

$$dX_t = -\nabla V(X_t)\, dt + \sqrt{2\beta^{-1}} dW_t, \tag{6.1}$$

where $\beta = 1/(k_B T)$. The equilibrium canonical measure is

$$d\mu = Z^{-1} \exp(-\beta V(x))\, dx$$

where $Z = \displaystyle\int_{\mathbb{R}^d} \exp(-\beta V(x))\, dx$ is the partition function. The equilibrium trajectories are those obtained with initial conditions X_0 distributed according to μ, and which satisfy (6.1). The difficulty when computing thermodynamics and/or dynamical quantities is due to the fact that V has several wells (called metastable states) around which the process X_t stays for a long time, especially at low temperature. The metastability of X_t implies that the convergence to equilibrium is very slow. Figure 6.1 presents a 2d schematic picture where X_t^1 is a slow variable of the system.

In this context, a very challenging problem in molecular dynamics is to compute reactive paths, namely trajectories of the system leaving a given metastable state, say A, and ending in another one, say B, without going back to A in the meantime (see figure 6.2 and references [70, 88]). The difficulty comes from the fact that a dynamics at equilibrium typically remains for a very long time around a metastable state before hoping to another one. In other words, most of the trajectories leaving A will go back to A, rather than reaching B. There exist many methods to sample the canonical equilibrium measure and compute equilibrium thermodynamics quantities like for example the free energy [32, 84], but it is much more difficult to compute dynamical quantities at equilibrium along reactive paths, like transport coefficients and transition rates.

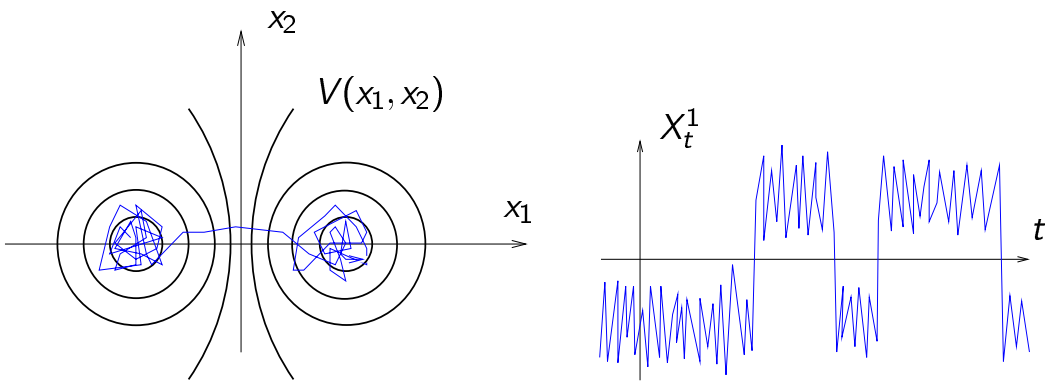

Figure 6.1: A 2d schematic picture where X_t^1 is a slow variable of the system.

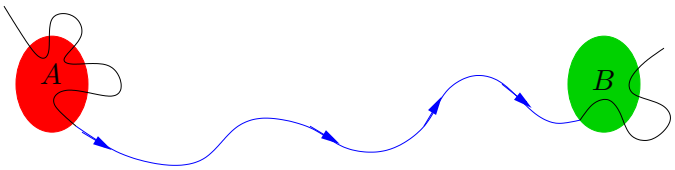

Figure 6.2: A reactive trajectory (in blue) between metastable states A and B.

In the following, we assume that a reaction coordinate (or an order parameter) is known, which in some sense, indexes transitions from A to B. In other words, a reaction coordinate is simply meant to be a smooth one-dimensional function $\xi : \mathbb{R}^d \to \mathbb{R}$ such that (see figure 6.3)

$$|\nabla \xi| \neq 0, \ A \subset \{x \in \mathbb{R}^d, \xi(x) < z_{\min}\} \text{ and } B \subset \{x \in \mathbb{R}^d, \xi(x) > z_{\max}\}, \tag{6.2}$$

where $z_{\min} < z_{\max}$ are two given real numbers. Let us denote

$$\Sigma_z = \{x \in \mathbb{R}^d, \xi(x) = z\}$$

the submanifold of configurations at a fixed value z of the reaction coordinate. For the algorithm we propose to give reliable results, one needs at least $\Sigma_{z_{\min}}$ (resp. $\Sigma_{z_{\max}}$) to be "sufficiently close" to A (resp. B). More precisely, we require that most trajectories starting from the submanifold $\Sigma_{z_{\min}}$ (resp. $\Sigma_{z_{\max}}$) end in A (resp. in B). Below, we will show on a simple two-dimensional example that a reaction coordinate as crude as the distance to a reference configuration in A may yield correct results.

In our context, the idea is to perform an iterative process on many replicas of trajectories which start from the metastable region A, and end either in A or in B, and to kill progressively the trajectories which have not reached high values along ξ. At the end of the day, an equilibrium ensemble of trajectories starting from A and ending in B are obtained. Compared to a brute force algorithm, the computational cost is typically reduced by a factor 1 000 (see Section 6.3 for more details). The details of the algorithm are provided in Section 6.2.

One of the differences between the algorithm we propose and the transition interface sampling method [112, 113], the forward flux sampling method [3, 4], or the milestoning method [54, 86] whose aim is also to compute reactive trajectories through paths ensembles, is that we do not need to decide *a priori* of a given discrete set of values $z_{\min} = z_0 < z_1 < z_2 < \ldots < z_n = z_{\max}$ through

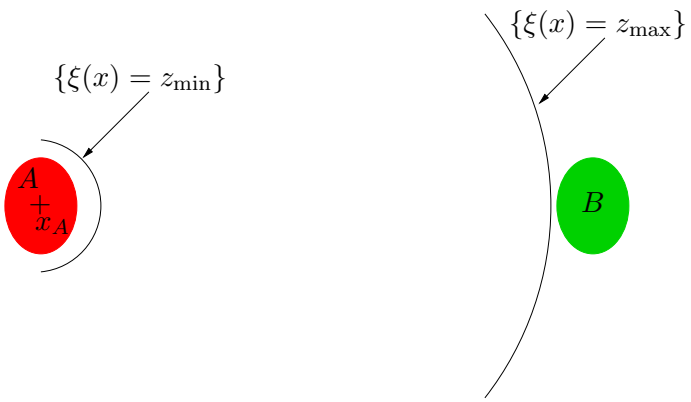

Figure 6.3: An example of reaction coordinate: $\xi(x) = \|x - x_A\|$.

which the trajectories will go. In some sense, these are *adaptively* selected by the algorithm, with typically fine discretizations in regions with high gradients of the potential energy, before saddle points, and coarser discretizations in flat regions. Other techniques to sample reactive trajectories include the string method [49, 50, 51], see also the review paper [44].

The main interests of the algorithm we propose are: (i) It does not require fine tuning of any numerical parameter, nor *a priori* discretization of the reaction coordinate values; (ii) It can be applied to any Markovian stochastic dynamics (overdamped Langevin, Langevin, Hybrid Monte Carlo, etc.) and is easy to implement starting from a standard molecular dynamics code; (iii) It seems to be reliable even for very simple choices of reaction coordinates satisfying (6.2), at least in our simple test cases. In particular, we will consider below a situation where the reaction coordinate does not describe all the metastabilities, namely a situation where, conditionally to a given value of ξ, the canonical measure is multimodal (or, equivalently, the potential energy exhibits wells separated by high barriers along some submanifolds Σ_z). This is actually a generic situation in practice, encountered in particular when multiple pathways link the two metastable states A and B (see Section 6.3.2 for a numerical illustration). Of course, the dependency of the whole procedure efficiency on the choice of the reaction coordinate for more complicated test cases remains to be investigated.

6.2 Computing Reactive Trajectories: the Algorithm

6.2.1 Reactive Trajectories

With the same notations as above, we propose an algorithm to build an ensemble of N reactive trajectories, using a reaction coordinate ξ which satisfies (6.2). In the numerical experiments of the paper [28], we have tested various reaction coordinates, but, to fix ideas, we will focus here on the specific case $\xi(x) = \|x - x_A\|$ where $x_A \in A$ denotes a reference configuration in A, and $\|\cdot\|$ is the Euclidean norm. The following algorithm can be seen as a dynamical version of Algorithm 3 of section 4.4: here the reaction coordinate ξ plays the same role as the score function Φ and the gradient dynamics replaces the Metropolis-Hastings procedure.

6.2.2 Details of the Algorithm

The algorithm starts with an initialization procedure which consists in generating an ensemble of N equilibrium trajectories $(X_t^n)_{0\le t\le \tau^n}$, which leave A, end either in A (the most likely) or in B, conditionally to the fact that $\sup_{t\ge 0}\xi(X_t^n)\ge z_{\min}$. For $n\in\{1,\dots,N\}$, let us denote these trajectories $(X_t^{1,n})$ and the associated stopping times $\tau^{1,n}=\tau^n$. Now, the adaptive multilevel splitting (AMS) algorithm goes as follows (see Figure 6.4 for a schematic representation): Iterate on $k\ge 1$,

1. Compute the largest reaction coordinate value attained for each path
$$z^{k,n}=\sup_{0\le t\le\tau^{k,n}}\xi(X_t^{k,n}).$$

2. Order the values $(z^{k,n})_{1\le n\le N}$
$$z^{k,\varepsilon^k(1)}\le z^{k,\varepsilon^k(2)}\le\ldots\le z^{k,\varepsilon^k(N)},$$
where ε^k is a permutation over $\{1,\dots,N\}$. To simplify the notation, let us denote
$$n^k=\varepsilon^k(1)=\operatorname*{argmin}_{n\in\{1,\dots,N\}} z^{k,n},$$
the index which realizes the smallest value $z^{k,\varepsilon^k(1)}$, and let us denote q^k the (empirical) quantile of order $1/N$, namely this smallest value
$$q^k=z^{k,\varepsilon^k(1)}=z^{k,n^k}=\min_{n\in\{1,\dots,N\}} z^{k,n}.$$

3. Kill the trajectory $(X_t^{k,n^k})_{0\le t\le\tau^{k,n^k}}$, and consider trajectories for iteration $k+1$ as follows:
   - For all $n\neq n^k$, the n-th trajectory is unchanged: $\tau^{k+1,n}=\tau^{k,n}$ and $(X_t^{k+1,n})_{0\le t\le\tau^{k+1,n}}=(X_t^{k,n})_{0\le t\le\tau^{k,n}}$;
   - Generate a new n^k-th trajectory in three steps:
     (i) Choose at random $i_k\in\{1,\dots,n^k-1,n^k+1,\dots,N\}$.
     (ii) Set $X_t^{k+1,n^k}=X_t^{k,i_k}$ for all $t\in(0,\sigma_k)$ where
     $$\sigma_k=\inf\{t\ge 0,\xi(X_t^{k,i_k})\ge q^k\}.$$
     (iii) Generate the end of the trajectory $(X_t^{k+1,n^k})_{t\ge\sigma_k}$ according to (6.1) (with $W_t=W_t^{n_k}$) until the stopping time
     $$\tau^{k+1,n_k}=\inf\{t\ge\sigma_k,X_t^{k+1,n^k}\in A\cup B\}.$$

4. Go back to 4 (with k being $k+1$), until $q^k\ge z_{\max}$. More precisely, the number of iterations is defined as
$$k_{\max}=\sup\{k\ge 1,\ q^k\le z_{\max}\}.$$

At iteration k of the algorithm, one obtains N equilibrium trajectories $(X_t^{k,n})_{0\le t\le\tau^{k,n}}$, which leave A, end either in A or in B, conditionally to the fact that $\sup_{0\le t\le\tau^{k,n}}\xi(X_t^n)\ge q^k$.
At the end of the algorithm, all the trajectories cross the submanifold $\Sigma_{q^{k_{\max}}}$. Since $k_{\max}$ is the last iteration index for which the quantile q^k is smaller than $z_{\max}$ and since $\Sigma_{z_{\max}}$ is assumed to be "close to" B, most of them end in B. The final step to retain only reactive trajectories is:

8. We retain only the trajectories which indeed end in B to perform statistics on reactive trajectories. We denote r the proportion of such trajectories among the ones obtained at the final iteration $k_{\max}$. Thus, most of the time, r is equal to 1.

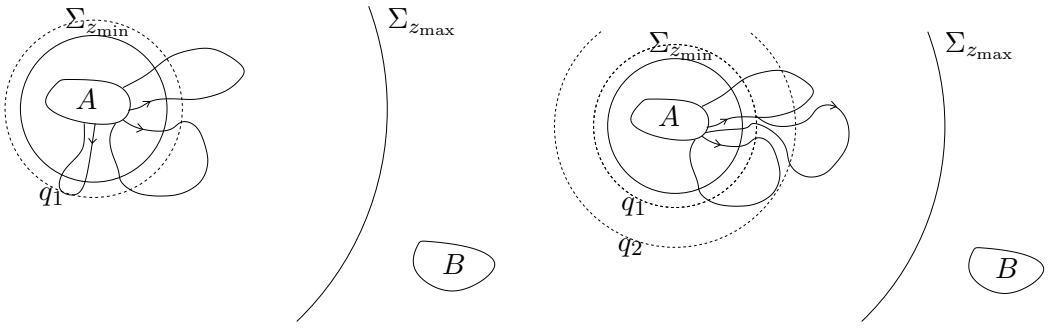

Figure 6.4: Schematic representation of the algorithm, with $N = 3$ paths: on the left, the path which goes the less far in the reaction coordinate direction is killed, and, on the right, a new path is generated, starting from a previous one at the $1/N$ empirical quantile value.

6.2.3 Discussion of the Algorithm

As for Algorithm 3, note that at the end of the day

$$\hat{\alpha}_N = r\left(1 - \frac{1}{N}\right)^{k_{\max}} \tag{6.3}$$

gives an estimate of the probability p of "observing a reactive trajectory". Specifically, this is an estimate of the probability, starting from $\Sigma_{z_{\min}}$ with the equilibrium distribution generated by the initialization procedure, to observe a trajectory which touches B before A. This probability actually depends on the choice of $\Sigma_{z_{\min}}$, while the law of the reactive trajectories generated by the algorithm does not.

A difficult mostly open question is to compute the asymptotic variance of an estimator associated to a given observable over reactive trajectories (say the time length of the trajectory) and then to try to optimize this estimator with respect to the chosen reaction coordinate (commonly called *importance function* [39] in the context of statistics). It can be shown that in terms of the asymptotic variance of $\hat{\alpha}_N$, the optimal reaction coordinate is the so-called *committor function* [70, 51] q which satisfies:

$$\begin{cases} -\nabla V \cdot \nabla q + \beta^{-1}\Delta q = 0 \text{ in } \mathbb{R}^d \setminus (\overline{A} \cup \overline{B}), \\ q = 0 \text{ on } \partial A \text{ and } q = 1 \text{ on } \partial B. \end{cases} \tag{6.4}$$

The function q can be interpreted in terms of the process X_t^x solution to (6.1) with initial condition $X_0^x = x$, as:

$$q(x) = \mathbb{P}\left(X_t^x \text{ reaches } B \text{ before } A\right). \tag{6.5}$$

We can check numerically on some simple test cases that the variance of the results seems to be smaller if ξ is chosen close to q (see [28]). But one interesting feature of the method is that it does not need to be the case to give reliable results in terms of reactive trajectories. This is a crucial point since it is impossible to solve numerically the partial differential equation (6.4) except in very low dimension.

6.3 Computing Reactive Trajectories: Numerical Illustrations

6.3.1 A One-Dimensional Case

In this section, we consider a one-dimensional situation and overdamped Langevin trajectories (6.1), with V being the double-well potential (see figure 6.5)

$$V(x) = x^4 - 2x^2.$$

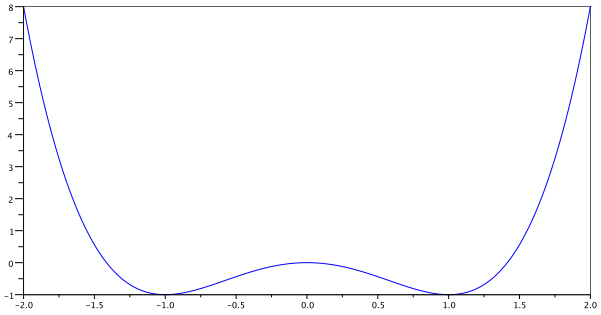

Figure 6.5: A double-well potential.

This potential has two global minima at ± 1 and one local maximum at 0. In this simple one dimensional setting, we set as metastable states $A = \{-1\}$ and $B = \{+1\}$, and the reaction coordinate is taken to be simply

$$\xi(x) = x.$$

For the numerical experiments, we take $z_{\max} = -z_{\min} = 0.9$. The aim of this section is mainly to validate the adaptive multilevel splitting (AMS) algorithm by comparing the results to those obtained by direct numerical simulation (DNS[1]), namely a simple Monte Carlo algorithm without any selection procedure, and then to have an idea of the computational gain. We will see that DNS can only be used for small values of β (typically $\beta \leq 10$ in this setting).

Distribution of the Time Lengths of Reactive Paths. To validate the algorithm, we compute an histogram of the distribution of the time length (duration) of a reactive path. On Figure 6.6, we compare the results obtained with DNS and our algorithm: the agreement is excellent. The interest of our algorithm is that it is possible to compute this distribution for very small temperatures (large values of β), for which a DNS cannot be used.

Computational Time. Let us now compare the computational time required to simulate an ensemble of reactive paths. In Table 6.1, we give CPU times for various values of β, using DNS (when possible) or our algorithm. The DNS time simulation rapidly explodes when β increases. For $\beta = 15$ and $N = 10^5$, the ratio between the CPU time of a DNS and the CPU time of our algorithm is of the order of 1 000.

Variance of the Estimators $\hat{\alpha}_N$. To complete the discussion on computational time, we also compare in Table 6.1 the relative variances of the estimators $\hat{\alpha}_N$ of the probability p, for DNS and for our algorithm. The relative variance is defined as the variance divided by the mean square. With the notations of this table, the relative variance of the DNS estimator for $\hat{\alpha}_N$ is estimated by $(1 - \hat{\alpha}_N)/N$. With our algorithm, we have seen in section 4.4 that this relative variance can be estimated by $-\log(\hat{\alpha}_N)/N$. This explains why in the four last rows of Table 6.1 (where $N = 10^5$), the relative variance of our estimator increases very slowly (in fact, logarithmically) when the probability of interest decreases to zero. To take into account both computational time and variance, Hammersley and Handscomb [69] propose to define the efficiency of a Monte Carlo process as "inversely proportional to the product of the sampling variance and the amount of labour expended in obtaining this estimate" (see also section 4.4.5). Using this definition of efficiency, for $\beta = 15$ and $N = 10^5$, our algorithm is about 800 times more efficient than DNS.

[1] also called CMC (Crude Monte Carlo).

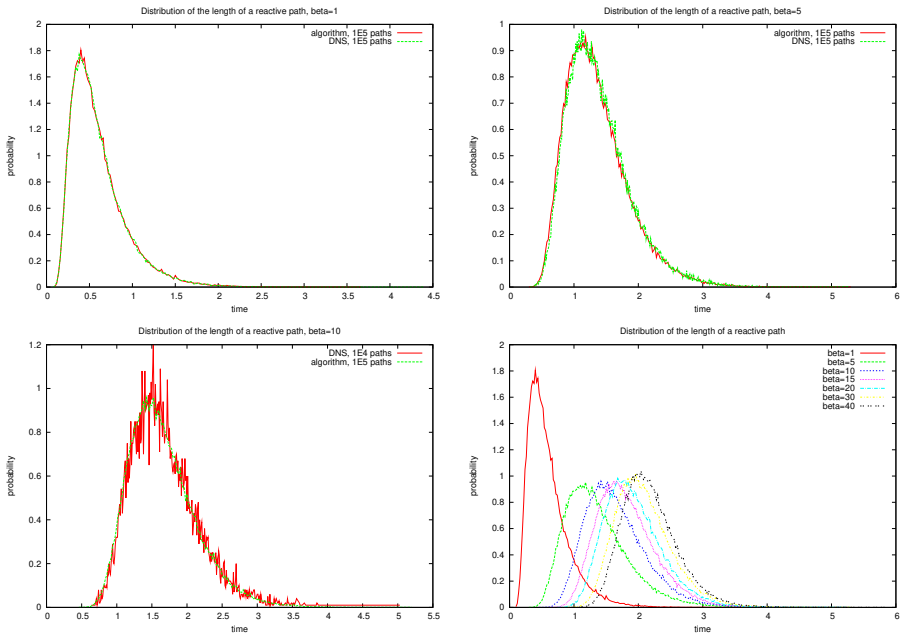

Figure 6.6: Distribution of the time lengths of reactive paths. In the first three figures we compare results computed with DNS and our algorithm. In the last figure, distributions of the lengths are given for various values of β: $\beta = 1, 5, 10, 15, 20, 30, 40$.

β	N	$\hat{\alpha}_N$	DNS time	AMS time	DNS RV	AMS RV
1	10^4	$1.03\,10^{-1}$	$2s$	$2s$	$9\,10^{-5}$	$2\,10^{-4}$
1	10^5	$1.01\,10^{-1}$	$21s$	1 min 19 s	$9\,10^{-6}$	$2\,10^{-5}$
10	10^4	$2.04\,10^{-5}$	140 min 05 s	5 s	10^{-4}	10^{-3}
10	10^5	$1.98\,10^{-5}$	1400 min $\star$	5 min 22 s	10^{-5}	10^{-4}
15	10^5	$1.78\,10^{-7}$	92000 min $\star$	7 min 52 s	10^{-5}	$1.5\,10^{-4}$
20	10^5	$1.33\,10^{-9}$		8 min 36 s		$2\,10^{-4}$
40	10^5	$5.82\,10^{-18}$		10 min 09 s		$4\,10^{-4}$

Table 6.1: Probability $\hat{\alpha}_N$ (see (6.3)), computational time and relative variance (RV) for the estimators of p, for different values of β and number of paths N. DNS CPU time with $\star$ is an extrapolated time deduced from a small number of iterations.

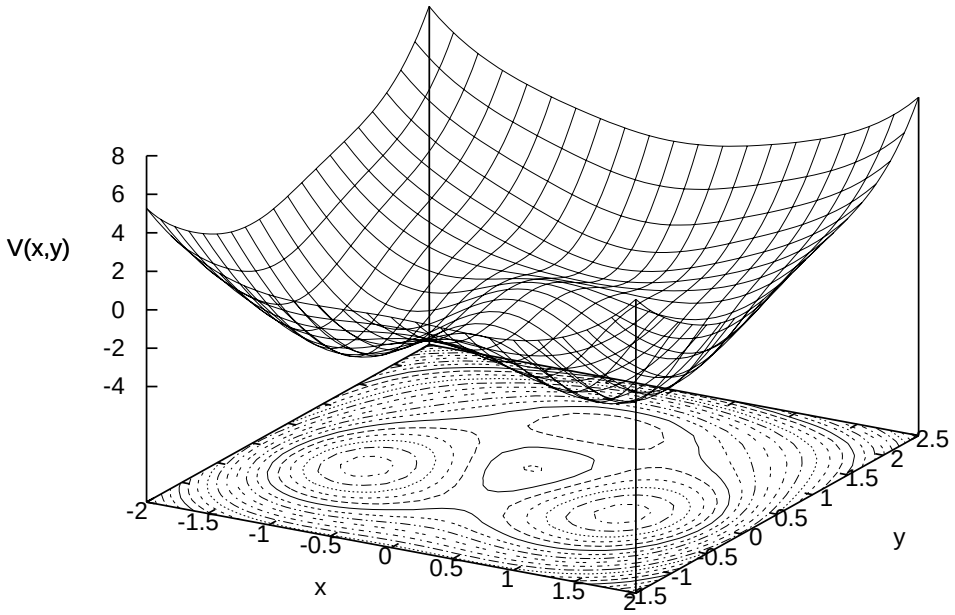

Figure 6.7: The potential V.

6.3.2 A Two-Dimensional Case with Two Channels

In this section, we apply the algorithm to a two-dimensional situation, again with overdamped Langevin trajectories (6.1). The potential V we consider is taken from [88, 91]:

$$\begin{aligned} V(x,y) = 3\mathrm{e}^{-x^2-(y-\frac{1}{3})^2} - 3\mathrm{e}^{-x^2-(y-\frac{5}{3})^2} - 5\mathrm{e}^{-(x-1)^2-y^2} \\ - 5\mathrm{e}^{-(x+1)^2-y^2} + 0.2x^4 + 0.2\left(y-\frac{1}{3}\right)^4. \end{aligned} \tag{6.6}$$

This potential (see Figure 6.7) has two deep minima approximately at $H_\pm = (\pm 1, 0)$, a shallow minimum approximately at $M = (0, 1.5)$ and three saddle points approximately at $U_\pm = (\pm 0.6, 1.1)$ and $L = (0, -0.4)$. In the notation of our algorithm above, A (resp. B) denotes a neighborhood of H_- (resp. H_+). The two metastable regions A and B can thus be connected by two channels: The upper channel around the points (H_-, U_-, M, U_+, H_+) and the lower channel around the points (H_-, L, H_+).

From large deviation theory (see for example Freidlin and Wentzell [59]), it is known that in the small temperature limit ($\beta \to \infty$) the reactive trajectories which will be favored will go through the upper channel, since the energy barrier is lower there. On the other hand, at higher temperature, the lower channel is also very likely, since the trajectories going through the upper channel have to pass two saddle points (two narrow corridors) to reach B instead of only one saddle point for the lower channel. The trajectories going through the upper channel are thus less favored in this regime, since the lower channel is more "direct". This temperature-dependent switching effect is well-known [88, 91].

For the numerical experiments, we use the two following values for the temperature [88]: $\beta = 6.67$ (low temperature), which is such that most of the trajectories go through the upper channel, and $\beta = 1.67$ (high temperature), which is such that most of the trajectories go through the lower channel. The region A (resp. B) is defined as the Euclidean ball $\mathcal{B}(H_-, 0.05)$ (resp. $\mathcal{B}(H_+, 0.05)$), and the reaction coordinate is the Euclidean distance to H_-: $\xi(x,y) = \|(x,y) - H_-\|$. Note that in such a low dimension, it is possible to solve numerically the PDE (6.4) and obtain the shape of the

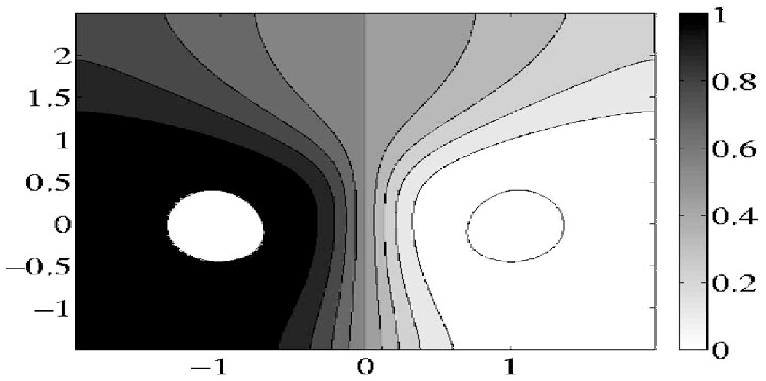

Figure 6.8: Level sets of $1-q$ for $\beta = 1.67$, with q the committor function as in (6.4).

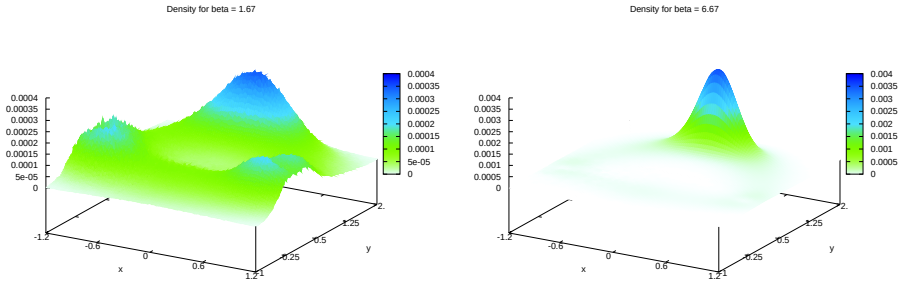

Figure 6.9: Density ρ for different choices of β.

committor function q (see figure 6.8 as given in [88] for the level sets of $1-q$). This will prove useful to compare our results with the ones obtained in [88] thanks to this optimal reaction coordinate.

To discretize the dynamics (6.1), we use an Euler scheme with a time-step size $\Delta t = 10^{-2}$. We take $z_{\min} = 0.05$ and $z_{\max} = 1.5$. We first plot the probability density ρ of positions, conditionally on being on a reactive trajectory. The discretization of this density uses a regular grid of size 100×100 with constant x and y intervals. Figure 6.9 gives the estimation of the density ρ for $N = 10^5$ and for the two temperature values $\beta = 1.67,\ 6.67$.

An important quantity computed from reactive paths is the flux of reactive trajectories, which is defined, up to a multiplicative constant, as [70, 88]: for any domain $\mathcal{D} \in \mathbb{R}^d \setminus (\overline{A} \cup \overline{B})$,

$$\int_{\mathcal{D}} \mathrm{div} J = \mathbb{P}(\text{a reactive trajectory enters } \mathcal{D}) - \mathbb{P}(\text{a reactive trajectory leaves } \mathcal{D}).$$

On Figure 6.10, we plot the flux J at the two temperature values, using a grid of size 20×20 with constant x and y intervals. It is clear from these figures that at low temperature ($\beta = 6.67$), the upper channel is favored, while at higher temperature ($\beta = 1.67$), the lower channel is more likely.

On Figure 6.11, a few reactive paths are plotted. To quantify the fact that the upper or the lower channel is preferentially used by reactive paths, let us consider $X^y_{\sigma_0}$ which is the y-value of the reactive path at the first time σ_0 such that the x-value of the process X_t is equal to 0. We consider that the reactive path goes through the upper (resp. the lower) channel if $X^y_{\sigma_0}$ is larger than 0.75 (resp. smaller than 0.25). For $\beta = 6.67$, the proportion of paths such that $0.25 \leq X^y_{\sigma_0} \leq 0.75$ is 0.28% and the paths going through the upper (resp. the lower) channel is 62.55% (resp. 37.17%).

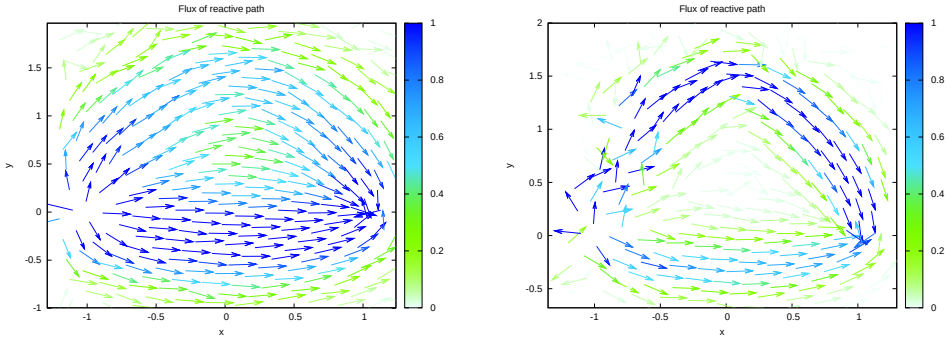

Figure 6.10: Flux of reactive trajectories, at inverse temperature $\beta = 1.67$ on the left, and $\beta = 6.67$ on the right. The color indicates the norm of the flux.

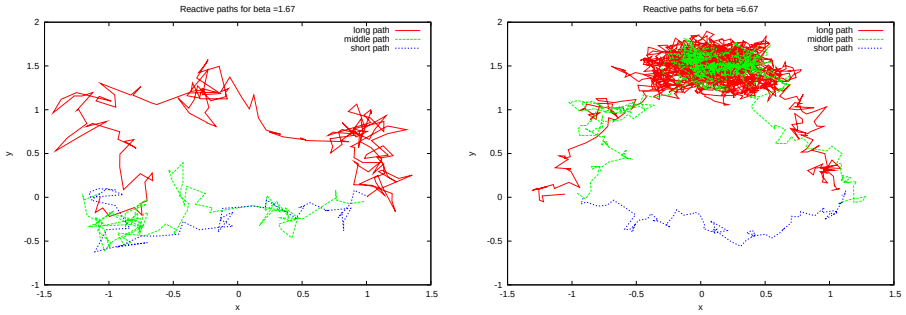

Figure 6.11: A few reactive paths for $\beta = 1.67$ (left), $\beta = 6.67$ (right).

For $\beta = 1.67$, the proportion of paths such that $0.25 \leq X^y_{\sigma_0} \leq 0.75$ is 11.26% and the paths going through the upper (resp. the lower) channel is 31.46% (resp. 57.28%).

Finally, we plot on Figure 6.12, the histogram of the time lengths of reactive trajectories, at the two temperatures. We observe two modes in this distribution when $\beta = 6.67$, corresponding to the two channels. These two modes overlap when $\beta = 1.67$. We would like to stress that all these results are in agreement with those obtained by Metzner, Schütte and Vanden-Eijden [88] using a different numerical method.

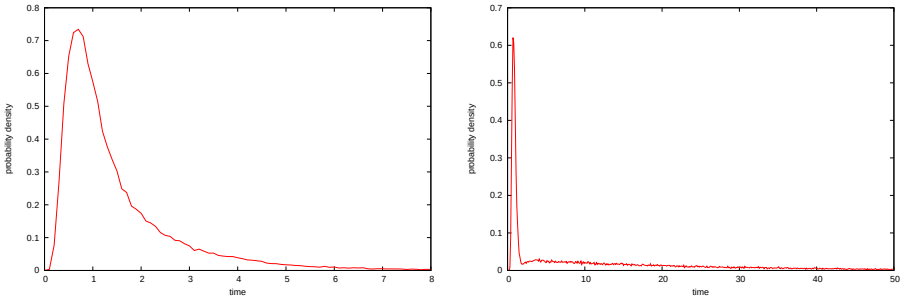

Figure 6.12: Distribution of the time lengths of reactive paths for $\beta = 1.67$ (left), $\beta = 6.67$ (right).

6.4 Conclusion and Perspectives

We have presented a multiple replica algorithm to generate an ensemble of reactive trajectories. We have illustrated its efficiency and accuracy on two test cases. In the paper [28], we have also proposed an estimator of the transition times. Future works are of course required in order to test the interest of such an approach for larger systems. We also would like to mention two possible extensions of the approach. First, in a case where only the initial metastable state A is known, one could think of using this algorithm to force the system to leave A (without knowing the metastable states around *a priori*), by using as a reaction coordinate the distance to a reference configuration x_A in A (like ξ above). The paths would then be generated until they go back to A, or they reach a given fixed final time T. This would generate equilibrium trajectories of time length T, conditionally to reach a certain distance from x_A. It should be an efficient procedure to explore the energy landscape at fixed positive temperature. Second, it would be interesting to test an adaptive procedure which, at the end of the algorithm, approximates the committor function thanks to the reactive trajectories, and then uses this approximation as a reaction coordinate, iteratively. In particular, note that the isocommittors are also isolines of the function $\exp(\beta V)\rho$, where ρ is the density along reactive paths which seems to be accurately obtained by our algorithm. This should produce better and better results as the approximations of the isocommittors get more and more refined.

Appendix A

The Application of a General Formula to Rare Event Analysis

A.1 Introduction

In this appendix, we present a non-asymptotic theorem for interacting particle approximations of unnormalized Feynman-Kac models. We provide an original stochastic analysis based on Feynman-Kac semigroup techniques combined with recently developed coalescent tree-based functional representations of particle block distributions. We present some regularity conditions under which the $\mathbb{L}_2$-relative error of these weighted particle measures grows linearly with respect to the time horizon yielding what seems to be the first results of this type for this class of unnormalized models. We also illustrate these results in the context of rare event analysis.

The field of Feynman-Kac path integrals and their particle interpretations are one of the most active contact points between probability, theoretical chemistry, quantum physics, and engineering sciences, including rare event analysis, and advanced signal processing. For a rather thorough discussion, the interested reader is recommended to consult the pair of books [40, 48], and the references therein. During the last two decades, the asymptotic analysis of these interacting particle models has been developed in various directions, including propagation of chaos analysis, $\mathbb{L}_p$-mean error estimates, central limit type theorems, and large deviation principles. Nevertheless, we emphasize that most of the non-asymptotic results developed in the literature are concerned with empirical particle measures and normalized Feynman-Kac probability distributions. Thus, they do not apply to engineering or physical problems involving the computation of unnormalized Feynman-Kac models including rare event particle simulation and partition functions estimation in statistical mechanics.

Loosely speaking, unnormalized Feynman-Kac measures represent the distribution of the paths of a Markov process, weighted by the product of a given collection of non-negative potential functions. The total masses of these measures are also called the normalizing constants. For instance, for set indicator potential functions the total mass of these functional represents the probability that the reference Markov chain stays in that set for a given number of time steps. We already mention that the particle approximations of these unnormalized measures are defined in terms of weighted products of empirical potential functions. The length of these products is directly related to the time horizon. The refined analysis of these unnormalized particle approximations requires to control the degeneracy of these weighted products in terms of the time parameter.

The main objective of this appendix is to present non-asymptotic $\mathbb{L}_2$-estimates for these weighted

particle measures. As shown in [25, 81] in the context of rare events, this result is sharp in the sense that the asymptotic variance of the relative errors grows linearly with respect to the time horizon. We design an original stochastic analysis that combines refined Feynman-Kac semigroup techniques with the recently developed algebraic tree-based functional representations of particle block distributions obtained by Del Moral, Patras and Rubenthaler in [43]. However, in the following, we will not go into the details of this analysis, nor into the description of the tree-based functional representation. For this purpose, we refer the interested reader to the article by Cérou, Del Moral and Guyader [24].

The rest of this appendix is organized as follows: In a preliminary section, section A.2, we provide a mathematical description of the Feynman-Kac models and their probabilistic particle interpretations. The advantage of the general Feynman-Kac model presented here is that it unifies the theoretical analysis of a variety of genetic type algorithms currently used in Bayesian statistics, biology, particle physics, and engineering sciences. It is clearly out of the scope of this chapter to present a detailed review of these particle approximation models. Section A.3 is devoted to the analysis of the total mass of unnormalized Feynman-Kac semigroups. We provide some regularity conditions under which the relative variation of these quantities depends only linearly on the time horizon of these semigroups. In section A.4, we state and prove the main results of the present chapter. We examine non-homogeneous models including degenerate potential functions that may vanish on some state space regions. In the final section, section A.5, we outline the preceding results in terms of efficiency for rare event probability estimation. Roughly speaking, we want to control the relative variance of our estimator when the event of interest is getting more and more rare. Our main result enables us to derive an efficiency result for rare event probability estimation, the first of its kind concerning the Interacting Particle System (IPS) approach applied to rare events.

A.2 Description of the Models and Statement of Some Results

We begin this section with a brief review of some of the standard notation. We denote respectively by $\mathcal{M}(E)$, $\mathcal{P}(E)$, and $\mathcal{B}_b(E)$, the set of bounded and signed measures, the subset of all probability measures on some measurable space $(E,\mathcal{E})$, and the Banach space of all bounded and measurable functions f on E equipped with the uniform norm $\|f\| = \sup_{x\in E} |f(x)|$. We denote by $\mu(f) = \int \mu(dx)\ f(x)$, the Lebesgue integral of a function $f \in \mathcal{B}_b(E)$, with respect to a measure $\mu \in \mathcal{M}(E)$. We slightly abuse the notation, and sometimes denote by $\mu(A) = \mu(1_A)$ the measure of a measurable subset $A \in \mathcal{E}$.

Recall that a bounded integral operator M from a measurable space E into itself, is an operator $f \mapsto M(f)$ from $\mathcal{B}_b(E)$ into itself such that the functions $M(f)(x) = \int_F M(x,dy)\ f(y)$ are measurable and bounded, for any $f \in \mathcal{B}_b(E)$. A bounded integral operator M from a measurable space $(E,\mathcal{E})$ into itself also generates a dual operator $\mu \mapsto \mu M$ from $\mathcal{M}(E)$ into $\mathcal{M}(E)$ defined by $(\mu M)(f) = \mu(M(f))$. Given a pair (M_1, M_2) of bounded integral operators we denote by M_1M_2 the composition of the operators given by the following formula $(M_1M_2)(x,dz) = \int M_1(x,dy)M_2(y,dz)$. We also set $M^m = M^{m-1}M = MM^{m-1}$ the m composition transition. Finally, the tensor product operator $M^{\otimes 2}$ is the bounded integral operator defined for every function $f \in \mathcal{B}_b(E\times E)$ by

$$M^{\otimes 2}(f)(x,x') = \int_{E\times E} M(x,dy)M(x',dy')f(y,y').$$

We consider a collection of bounded potential functions G_n on the state space E, a distribution η_0 on E, and a collection of Markov transitions $M_n(x,dy)$ from E into itself. We associate to these

objects the Feynman-Kac measures, defined for any $f \in \mathcal{B}_b(E)$ by the formulae

$$\eta_n(f) = \gamma_n(f)/\gamma_n(1) \quad \text{with} \quad \gamma_n(f) = \mathbb{E}[f(X_n) \prod_{0\leq k<n} G_k(X_k)]. \tag{A.1}$$

In (A.1), $(X_n)_{n\geq 0}$ represents a Markov chain with initial distribution η_0, and elementary transitions $(M_n)_{n>0}$. To simplify the presentation, we shall suppose that the potential functions G_n take values in $[0,1]$ and for any $n \geq 0$ we have $\eta_n(G_n) > 0$. By the Markov property and the multiplicative structure of (A.1), it is easily checked that the flow $(\eta_n)_{n\geq 0}$ satisfies the following equation

$$\eta_{n+1} = \Phi_n(\eta_n) = \Psi_{G_n}(\eta_n) M_{n+1}, \tag{A.2}$$

with the Boltzmann-Gibbs transformation Ψ_{G_n} defined as follows:

$$\Psi_{G_n}(\eta_n)(dx) = \frac{1}{\eta_n(G_n)}\ G_n(x)\ \eta_n(dx).$$

We also readily check the following multiplicative formula

$$\gamma_n(1) = \prod_{0\leq p<n} \eta_p(G_p). \tag{A.3}$$

The particle approximation of the flow (A.2) depends on the choice of the McKean interpretation model. These probabilistic interpretations consist of a chosen collection of Markov transitions K_{n+1,η_n}, indexed by the set of probability measures η_n on E, and satisfying the compatibility condition $\Phi_n(\eta_n) = \eta_n K_{n+1,\eta_n}$ (see for instance [40], definition 2.5.4 p.75). The choice of these Markov transitions is far from being unique. By (A.2), we find that

$$\forall n \geq 0 \quad \forall \alpha \in [0,1] \qquad \eta_{n+1} = \eta_n K^{(\alpha)}_{n+1,\eta_n} \tag{A.4}$$

with the McKean transition $K^{(\alpha)}_{n+1,\eta_n} = S_{\alpha G_n,\eta_n} M_{n+1}$ and the selection type transition

$$S_{\alpha G_n,\eta_n}(x,dy) = \alpha G_n(x)\ \delta_x(dy) + (1-\alpha G_n(x))\ \Psi_{G_n}(\eta_n)(dy).$$

Definition 4 *The mean field particle interpretation of the evolution equation (A.4) is the E^N-valued Markov chain $X^{(N)}_n = \left(X^{(N,i)}_n\right)_{1\leq i\leq N}$ with elementary transitions*

$$\mathbb{P}\left(X^{(N)}_{n+1} \in dx_{n+1} \mid X^{(N)}_n\right) = \prod_{i=1}^{N} K^{(\alpha)}_{n+1,\eta^N_n}(X^{(N,i)}_n, dx^i_{n+1}),$$

where η^N_n stands for the occupation measure $\eta^N_n = \frac{1}{N}\sum_{i=1}^N \delta_{X^{(N,i)}_n}$ of the N-uple $X^{(N)}_n$ at time n. The initial configuration $X^{(N)}_0 = \left(X^{(N,i)}_0\right)_{1\leq i\leq N}$ consists of N independent and identically distributed random variables with distribution η_0.

In our context, it is worth mentioning that the elementary transitions of the chain $X^{(N)}_n \rightsquigarrow X^{(N)}_{n+1}$ are decomposed into two separate mechanisms:

- Firstly, the current state $X^{(N,i)}_n$ of each individual with label $i \in \{1,\ldots,N\}$ performs an acceptance-rejection type transition $X^{(N,i)}_n \rightsquigarrow \widehat{X}^{(N,i)}_n$ according to Markov transition $S_{\alpha G_n,\eta^N_n}$. In other words with a probability $\alpha G_n(X^{(N,i)}_n)$ the particle remains in the same site and we set $\widehat{X}^{(N,i)}_n = X^{(N,i)}_n$. Otherwise it jumps to a new location randomly chosen according to the Boltzmann-Gibbs distribution

$$\Psi_{G_n}(\eta^N_n) = \sum_{j=1}^{N} \frac{G_n(X^{(N,j)}_n)}{\sum_{k=1}^N G_n(X^{(N,k)}_n)}\ \delta_{X^{(N,j)}_n}.$$

- After the acceptance-rejection stage, the selected individuals $\widehat{X}_n^{(N,i)}$ evolve independently to a new site $X_{n+1}^{(N,i)}$ randomly chosen with distribution $M_{n+1}(\widehat{X}_n^{(N,i)}, dx)$.

Remarks:

1. For $\alpha = 0$, we find that $K_{n+1,\eta_n}^{(0)}(x, dy) = \Phi_{n+1}(\eta_n)(dy)$. In this situation, $X_n^{(N)}$ evolves as a simple genetic algorithm with mutation transitions M_n, and selection fitness functions G_n. Theorem 10 below is only valid in this specific situation, but all other results in this chapter are true for any $\alpha \in [0, 1]$. In particular, when the G_n's are indicator functions, we always consider $\alpha = 1$ in practice (see also section 4.3).

2. Besides the fact that $\eta_n(G_n) > 0$, it is important to mention that the empirical quantities $\eta_n^N(G_n)$ may vanish at a given random time. The formal definition of this time τ^N is

$$\tau^N = \inf\left\{n \geq 0 \ : \ \eta_n^N(G_n) = 0\right\}.$$

   At time τ^N, the particle algorithm stops and from that time the particle approximation measures are defined as the null measures (see for instance chapter 7, section 7.2.2 in [40]): $\forall n > \tau^N, \eta_n^N = 0$.

Mimicking the multiplicative formula (A.3), we also consider the following N-particle approximation of the unnormalized Feynman-Kac measures.

Definition 5 *The N-particle approximation measures γ_n^N associated with the unnormalized Feynman-Kac models γ_n introduced in (A.1) are defined for any $f \in \mathcal{B}_b(E)$ by the following formulae:*

$$\gamma_n^N(f) = \gamma_n^N(1) \times \eta_n^N(f) \quad \text{with} \quad \gamma_n^N(1) = \prod_{0 \leq p < n} \eta_p^N(G_p).$$

As an aside, we observe that $\gamma_n^N = 0$ for any time $n > \tau^N$. It is well known that the particle measures γ_n^N are unbiased estimates of the unnormalized Feynman-Kac measures γ_n (see for instance [40], Theorem 7.4.2 p.239). That is we have that

$$\forall f \in \mathcal{B}_b(E) \qquad \mathbb{E}\left(\gamma_n^N(f)\right) = \gamma_n(f).$$

It is obviously out of the scope of this chapter to present a full asymptotic analysis of these particle models. We refer the interested reader to the book [40] and the series of articles [41, 42, 75] and the references therein. For instance, it is well known that the particle occupation measures converge to the desired Feynman-Kac measures as the size of the population tends to infinity. That is, we have with various precision estimates, and as N tends to infinity, the weak convergence results $\lim_{N\to\infty} \eta_n^N = \eta_n$ and $\lim_{N\to\infty} \gamma_n^N = \gamma_n$.

To give a flavor of our results, we present in this section non-asymptotic variance estimates only for time homogeneous models $(G_n, M_n) = (G, M)$. The next result is a representation/decomposition formula for the normalizing constant $\gamma_n^N(1)$.

Theorem 10 *For the simple genetic algorithm corresponding to the choice $\alpha = 0$, we have an explicit decomposition formula of the following form*

$$\forall N > 1 \qquad \mathbb{E}\left(\gamma_n^N(1)^2\right) = \gamma_n(1)^2 \left(1 + \left(1 - \frac{1}{N}\right)^{(n+1)} \sum_{s=1}^{n+1} \frac{1}{(N-1)^s} \, v_n(s)\right)$$

for some finite constants $v_n(s)$ explicitly described in terms of Feynman-Kac type coalescent tree based expansions and whose values do not depend on the precision parameter N.

Remark: The analysis of particle models with an acceptance parameter $\alpha \in]0,1]$ is much more involved. In particular, we have no explicit representation formulae for the second moment of $\gamma_n^N(1)$ but only some $\mathbb{L}_2$-mean error bounds between $\gamma_n^N(1)$ and its limiting value $\gamma_n(1)$.

The quantities $v_n(s)$ are expressed in terms of coalescent tree based expansions of path integrals associated with the semigroup $Q_{p,n}$ associated with the Feynman-Kac distribution flow $\gamma_n = \gamma_p Q_{p,n}$, with $0 \leq p \leq n$. It is clear that the preceding representation for the variance is only as good as our information about $v_n(s)$. Theorem 11 and its extension to non-homogeneous models, Theorem 12, provide some precise conditions under which $v_n(s)$ can be upper-bounded.

We also would like to emphasize that Theorem 10 holds true under no additional assumptions on the model. In particular, it is valid for Feynman-Kac models associated with non-homogeneous potential functions G_n and Markov transitions M_n, including the example of rare event estimation of section A.5. The next theorem is of a different flavor since it holds true for any $\alpha \in [0,1]$, but only under very strict conditions on pair (G, M). Its extension to non-homogeneous Feynman-Kac models is presented in section A.4 (see Theorem 12 and Corollary 4).

Theorem 11 *Suppose that the pair of potential-transitions (G, M) are chosen so that*

$$\forall (x, x') \in E^2 \quad G(x) \leq \delta\ G(x') \quad \text{and} \quad M^m(x, dy) \leq \beta\ M^m(x', dy) \tag{A.5}$$

for some $m \geq 1$ and some parameters $(\delta, \beta) \in [1, \infty[^2$. In this situation, for any $n \geq 0$ and any $N > (n+1)\beta\delta^m$ we have

$$\mathbb{E}\left[\left(\gamma_n^N(1) - \gamma_n(1)\right)^2\right] \leq \gamma_n(1)^2 \left(\frac{4}{N}\ (n+1)\ \beta\ \delta^m\right). \tag{A.6}$$

As the quantities $v_n(s)$ discussed above, the variance estimates (A.6) involve the analysis of coalescent tree based integrals expressed in terms of the semigroup $Q_{p,n}$. We already mention that the regularity condition (A.5) is mainly used to obtain a uniform control of the total mass mapping $x \mapsto Q_{p,n}(1)(x)$.

However, the first part of assumption (A.5) is clearly not satisfied for potential functions G_n that can be equal to zero, which is typically the case with indicator type potential functions. Consequently, the analysis of non-homogeneous models associated with indicator type potential functions (as is the case in rare event analysis) will be developed using a time non-homogeneous version of condition (A.5). In this situation, the upper bound corresponding to (A.6) can be expressed in terms of a sum over n quantities that depend on the oscillations of the potential functions G_n and on the mixing properties of the Markov transitions M_n (see for instance Corollary 4). More precisely, we can replace in condition (A.5) the triplet (E, G, M) by the triplet $(\widehat{E}_n, \widehat{G}_n, \widehat{M}_n)$ defined as

$$\widehat{E}_n = \widehat{G}_n^{-1}(]0,1]) \qquad \widehat{G}_n(x) = M_n(G_n)(x) \quad \text{and} \quad \widehat{M}_n(x, dy) = M_n(x, dy)G_n(y)/M_n(G)(x)$$

To illustrate these ideas, the application to rare events will be given in section A.5.

A.3 Regularity Properties of Feynman-Kac Semigroups

This section is concerned with some regularity properties of the Feynman-Kac semigroups involved in the coalescent tree based functional expansions for non-homogeneous models. To describe precisely these new conditions, we need to introduce another round of notations.

Definition 6 *We denote by A_n the support of the potential functions G_n, that is*

$$A_n = \{x \in E \ : \ G_n(x) > 0\}.$$

We let $(\widehat{\gamma}_n, \widehat{\eta}_n)$ be the updated Feynman-Kac measures on the set A_n given by

$$\forall n \geq 0 \qquad \widehat{\gamma}_n(dx) = \gamma_n(dx)\ G_n(x) \quad \textit{and} \quad \widehat{\eta}_n(dx) = \frac{1}{\eta_n(G_n)}\ G_n(x)\ \eta_n(dx).$$

We let $\left(\widehat{G}_n, \widehat{M}_n\right)$ be the pair of potential functions and Markov transitions given by :

$$\forall x \in A_n \qquad \widehat{G}_n(x) = M_{n+1}(G_{n+1})(x) \quad \text{and} \quad \forall x \in A_{n-1} \qquad \widehat{M}_n(x, dy) = \frac{M_n(x, dy)G_n(y)}{M_n(G_n)(x)}$$

Notice that the updated Feynman-Kac measures $(\widehat{\gamma}_n, \widehat{\eta}_n)$ can be rewritten in terms of $\left(\widehat{G}_n, \widehat{M}_n\right)$ with the following change of reference measure formula

$$\widehat{\eta}_n(f) = \frac{\widehat{\gamma}_n(f)}{\widehat{\gamma}_n(1)} \quad \text{with} \quad \widehat{\gamma}_n(f) = \eta_0(G_0)\ \mathbb{E}\left(f\left(\widehat{X}_n\right) \prod_{0 \leq p < n} \widehat{G}_p(\widehat{X}_p)\right) \tag{A.7}$$

In the above display, $\widehat{X}_n$ stands for a non-homogeneous Markov chain with initial distribution $\widehat{\eta}_0$ and elementary Markov transitions $\widehat{M}_n$ from A_{n-1} into A_n. We are now in position to describe these new conditions.

Condition $(\widehat{H})_m$:

- $(\widehat{G})$ The potential functions $\widehat{G}_n$ satisfy the following conditions

$$\forall n \geq 0 \qquad \widehat{\delta}_n = \sup_{(x,y) \in A_n^2} \frac{\widehat{G}_n(x)}{\widehat{G}_n(y)} < \infty$$

- $(\widehat{M})_m$ There exists some integer $m \geq 1$ and some sequence of numbers $\widehat{\beta}_p^{(m)} \in [1, \infty[$ such that for any $p \geq 0$ and any $(x, x') \in A_p^2$ we have

$$\widehat{M}_{p,p+m}(x, dy) \leq \widehat{\beta}_p^{(m)}\ \widehat{M}_{p,p+m}(x', dy) \quad \text{with} \quad \widehat{M}_{p,p+m} = \widehat{M}_{p+1}\widehat{M}_{p+2} \ldots \widehat{M}_{p+m}$$

Using the change of measure formula (A.7) we observe that the semigroup of the updated measures $\widehat{\gamma}_n$ is given by

$$\widehat{Q}_{p,n} = \widehat{Q}_{p+1}\widehat{Q}_{p+2} \ldots \widehat{Q}_n \quad \text{with} \quad \widehat{Q}_n(x, dy) = \widehat{G}_{n-1}(x)\ \widehat{M}_n(x, dy).$$

In other words, $\widehat{Q}_{p,n}$ is defined as the semigroup $Q_{p,n}$ by replacing the pair of objects (G_n, M_n) by the quantities $(\widehat{G}_n, \widehat{M}_n)$. With these notations, we can prove that for any $p \geq 0$

$$\sup_{(x,y) \in A_p^2} \frac{\widehat{Q}_{p,n}(1)(x)}{\widehat{Q}_{p,n}(1)(y)} \leq \widehat{\delta}_p^{(m)}\ \widehat{\beta}_p^{(m)} \quad \text{with} \quad \widehat{\delta}_p^{(m)} = \prod_{p \leq q < p+m} \widehat{\delta}_q$$

as soon as the regularity condition $(\widehat{H})_m$ is met for some parameters $(m, \widehat{\delta}_n, \widehat{\beta}_p^{(m)})$.

A.4 Non-Asymptotic $\mathbb{L}_2$-Estimates

This section is concerned with the statement of the main results of this appendix.

Theorem 12 *We suppose condition $(\widehat{H})_m$ is met for some parameters $(m, \widehat{\delta}_n, \widehat{\beta}_p^{(m)})$. In addition, we assume that the potential functions G_n satisfy the following conditions*

$$\forall n \geq 0 \qquad \widetilde{\delta}_n = \sup_{(x,y)\in A_n^2} \frac{G_n(x)}{G_n(y)} < +\infty \tag{A.8}$$

Then, for any non-negative function $F \in \mathcal{B}_b(E^2)$ with $\|F\| \leq 1$, and any $N > 1$ we have

$$\mathbb{E}\left((\gamma_n^N)^{\otimes 2}(F)\right) \leq \gamma_n(1)^2 \times \prod_{s=0}^{n} \left(1 + \frac{1}{N-1} \frac{\widetilde{\delta}_s\ \widehat{\delta}_s^{(m)}\ \widehat{\beta}_s^{(m)}}{\eta_s(A_s)}\right)$$

We conclude this section with a simple consequence of the above estimates. Notice that, in a homogeneous context, Theorem 11 is a direct consequence of the following corollary.

Corollary 4 *When conditions (A.8) and $(\widehat{H})_m$ are met for some $(m, \widehat{\delta}_n, \widehat{\beta}_p^{(m)})$, we have the non-asymptotic estimates*

$$N > \sum_{s=0}^{n} \frac{\widetilde{\delta}_s\ \widehat{\delta}_s^{(m)}\ \widehat{\beta}_s^{(m)}}{\eta_s(A_s)} \Longrightarrow \mathbb{E}\left(\left[\frac{\gamma_n^N(1)}{\gamma_n(1)} - 1\right]^2\right) \leq \frac{4}{N} \sum_{s=0}^{n} \frac{\widetilde{\delta}_s\ \widehat{\delta}_s^{(m)}\ \widehat{\beta}_s^{(m)}}{\eta_s(A_s)}.$$

A.5 Application to Rare Events

In this section, we want to outline the use of our main result in terms of efficiency for rare event probability estimation. By rare event we mean an event whose probability is too small to be accurately estimated by a simple Monte Carlo procedure in a reasonable time. Practically, this is the case if this probability is less than, say 10^{-9}. In this case, the normalizing constant $\gamma_n(1)$ is the probability, to be estimated, of the rare event under consideration.

One of the most used model for rare event is the following. Let $Z = \{Z_t\,,\, t \geq 0\}$ be a continuous–time strong Markov process taking values in some Polish state space S. For a given target Borel set $A \subset S$ we define the hitting time

$$T_A = \inf\{t \geq 0 \,:\, Z_t \in A\},$$

as the first time when the process Z hits A. In many applications, the set A is the (super) level set of a scalar measurable function ϕ defined on S, i.e.

$$A = \{z \in S \,:\, \phi(z) \geq \lambda_A\}.$$

It may happen that most of the realizations of X never reach the set A. As a consequence, the corresponding rare event probabilities are extremely difficult to analyze. In particular one would like to estimate the quantity

$$\mathbb{P}(T_A \leq T),$$

where T is a $\mathbb{P}$–almost surely finite stopping time, for instance the hitting time of a recurrent Borel set $R \subset S$, i.e. $T = T_R$ with

$$T_R = \inf\{t \geq 0 \,:\, Z_t \in R\} \quad \text{and} \quad \mathbb{P}(T_R < \infty) = 1.$$

In practice the process Z, before entering into the desired set A, passes through a decreasing sequence of closed sets

$$A = A_{n^\star} \subset A_{n^\star - 1} \subset \cdots \subset A_2 \subset A_1 \subset A_0.$$

The parameter $n^\star$ and the sequence of level sets depend on the problem at hand. We can easily fit this problem in the Feynman-Kac model presented in section A.2 simply by setting

$$\forall 1 \leq n \leq n^\star \quad X_n = Z_{T_n \wedge T}$$

where, with a slight abuse of notation, T_n stands for the first time T_{A_n} the process Z reaches A_n, that is

$$T_n = \inf\{t \geq 0 \ : \ Z_t \in A_n\},$$

with the convention $\inf \emptyset = \infty$. The potential functions G_n on S are defined by

$$G_n(x) = \mathbb{1}_{A_n}(x).$$

In this notation, we have $T_A = T_{n^\star}$ and for every $n \leq n^\star$

$$\gamma_n(1) = \mathbb{P}(T_n \leq T) \quad \text{and} \quad \eta_n = \mathcal{L}(X_n \mid T_n \leq T). \tag{A.9}$$

For more details on these excursion valued Feynman-Kac models, we refer the reader to Cérou, Del Moral, Le Gland and Lezaud [25]. As we will show now, our main result enables us to derive an efficiency result for rare event probability estimation, the first of its kind concerning the Interacting Particle System (IPS) approach applied to rare events.

Basically, efficiency results are about asymptotics when the rare event probability goes to 0: we want to control the relative variance of our estimator when the event of interest is getting more and more unlikely. In the context of *importance sampling*, a discussion about various efficiency (or robustness) properties may be found in L'Ecuyer, Blanchet, Tuffin and Glynn [83]. Among all those, we will focus here on logarithmic efficiency, a topic that was already mentioned in section 4.4.5.

Returning to the framework presented above, we further assume that we have a family of rare sets A^ε indexed by $\varepsilon \geq 0$, of the form

$$A^\varepsilon = \{z \in S \text{ s.t. } \phi(z) > \lambda_\varepsilon\},$$

for some real valued function ϕ. Denote as usual

$$T_{A^\varepsilon} = \inf\{t \geq 0 \ : \ Z_t \in A^\varepsilon\} \quad \text{and} \quad T_R = \inf\{t \geq 0 \ : \ Z_t \in R\}.$$

Assume further that we have for some fixed $\theta > 0$

$$\mathbb{P}(T_{A^\varepsilon} < T_R) = e^{-\theta/\varepsilon},$$

which is typical of behavior driven by a large deviation principle. We further assume that we are given a non-increasing sequence of level sets

$$A^\varepsilon = A_{n_\varepsilon} \subset A_{n_\varepsilon - 1} \subset \cdots \subset A_2 \subset A_1 \subset A_0$$

with a real valued function ψ so that

$$A_n = \{z \in S \ : \ \psi(z) > L_n\}.$$

In the above displayed formula $(L_n)_{1\leq n\leq n_\varepsilon}$ stands for a non-decreasing sequence of real numbers, with some fixed time horizon n_ε that may depend on the parameter ϵ, and so that $A_{n_\varepsilon} = A^\varepsilon$. In the rare event literature, such a function ψ is called an *importance function*. In this notation, by (A.9) the rare event probability of interest is given by

$$\gamma_{n_\varepsilon}(1) = \mathbb{P}(T_{n_\varepsilon} \leq T_R) = e^{-\theta/\varepsilon}$$

Then we say that our estimator $\gamma^N_{n_\varepsilon}(1)$ has the logarithmic efficiency property if we have

$$\lim_{\varepsilon\to 0} \frac{\log \mathbb{E}\left[\left(\gamma^N_{n_\varepsilon}(1)\right)^2\right]}{2\log \gamma_{n_\varepsilon}(1)} = 1.$$

Next, we discuss the regularity conditions $(\widehat{G})$ and $(\widehat{M})_m$ introduced on page 88. Firstly, we observe that the parameters $\widetilde{\delta}_n$ introduced in (A.8) are simply given by $\widetilde{\delta}_n = 1$. We check this claim using the fact that the potential functions G_n are the indicator functions on excursion subsets ending at the level sets A_n. Secondly, the assumption $(\widehat{M})_m$ is clearly a mixing type property. In this context $\widehat{M}_n(x_{n-1}, dx_n)$ is the elementary transition probability of an excursion $\widehat{X}_n$ starting at A_{n-1} (at the terminal state of an excursion x_{n-1} ending at A_{n-1}) and ending at the next level set A_n.

Example: We can illustrate condition $(\widehat{M})_m$ for the simple random walk on the one dimensional lattice $S = \mathbb{Z}$ starting at the origin, with the decreasing sequence of level sets $A_n = [n, \infty[$. In this context, we readily find that $(\widehat{M})_m$ is satisfied with $m = 1$ and $\widehat{\beta}^{(1)}_n = 1$, for every $n \geq 1$. More generally, in the simple setting of one dimension (i.e. the random process Z lives in $\mathbb{R}$), we always have $\widehat{\beta}^{(m)}_n = 1$ for all n.

Now we discuss the regularity condition $(\widehat{G})$. We observe that

$$\widehat{G}_n(x_n) = M_{n+1}\left(G_{n+1}\right)(x_n)$$

is the probability of reaching the set A_{n+1}, starting from the terminal value of a random excursion x_n ending at A_n. The less this quantity depends on x_n, the lower is the variance, as it is already well known for the asymptotic variance (as seen in [25]). So a good choice of the sets A_n is such that they are close to level sets for the probability of reaching the rare event.

From Corollary 4, we see that $\eta_n(A_n)$ is another quantity of interest. In this situation, we recall that A_n is the set of all random excursions ending at the level A_n and η_n is the distribution of the n-th excursion X_n of the process Z_t given the fact that it has reached the level A_{n-1} at time T_{n-1}. Thus, $\eta_n(A_n)$ is the probability of reaching level A_n, knowing that the trajectory has reached A_{n-1}. It is well known already (see [25, 81]) that we need to have these quantities $\eta_n(A_n)$ as close to each other as possible (the best would be equal). So not only do we need to have an importance function close to the optimal one, but also to have the sets A_n evenly spaced in terms of hitting probabilities.

We would like to stress here that the issue of constructing a good importance function is far from trivial. It has been nicely addressed in Dean and Dupuis [39] in the case of *importance splitting* techniques, which are close to the IPS approach. Their choice of importance function allows them to prove the asymptotically optimal efficiency of the importance splitting with their choice of the importance function.

From now on, we assume that we know how to construct a good importance function, in such a way that for all n, $\widehat{\delta}^{(m)}_n < \delta$ for some δ, and we know how to construct the level sets A_n so that

$$\mathbb{P}\left(T_n < T_R \mid T_{n-1} < T_R\right) = \eta_n(A_n) \approx p > 0$$

for some $p \in [0,1]$. A practical way for doing this has been proposed by Cérou and Guyader in [27]. We also suppose that the Markov process $\widehat{X}_n$ is sufficiently mixing, so that $\widehat{\beta}_n^{(m)} < \beta$, for some β. In this situation, using the fact that $\eta_n(A_n) \approx p > 0$, we get that the number n_ε of steps needed to get to the rare event is of order $-\frac{\theta}{\varepsilon \log p}$. Using Theorem 12, we see that

$$\mathbb{E}[(\gamma_{n_\varepsilon}^N(1))^2] \leq \gamma_{n_\varepsilon}(1)^2 \left(1 + \frac{\delta\beta}{(N-1)p}\right)^{-\frac{\theta}{\varepsilon \log p}},$$

Using the fact that $\log \gamma_{n_\varepsilon}(1) = -\theta/\varepsilon$, we get the lower bound

$$1 + \frac{1}{2\log p} \log\left(1 + \frac{1}{N-1}\frac{\delta\beta}{p}\right) \leq \frac{\log \mathbb{E}[(\gamma_{n_\varepsilon}^N(1))^2]}{2\log \gamma_{n_\varepsilon}(1)}.$$

Now, using Jensen's inequality and the fact that the estimator $\gamma_{n_\varepsilon}^N(1)$ is unbiased, we have the upper bound

$$\frac{\log \mathbb{E}[(\gamma_{n_\varepsilon}^N(1))^2]}{2\log \gamma_{n_\varepsilon}(1)} \leq 1.$$

Putting all things together, we get the asymptotic logarithmic efficiency at any (slow) rate, in the sense that

$$\lim_{N\uparrow\infty} \frac{\log \mathbb{E}[(\gamma_{n_\varepsilon}^N(1))^2]}{2\log \gamma_{n_\varepsilon}(1)} = 1.$$

Bibliography

[1] C. Abraham, G. Biau, and B. Cadre. On the kernel rule for function classification. *Ann. Inst. Statist. Math.*, 58(3):619–633, 2006.

[2] C. Abraham, P.-A. Cornillon, E. Matzner-Løber, and N. Molinari. Unsupervised curve clustering using B-splines. *Scand. J. Statist.*, 30(3):581–595, 2003.

[3] R.J. Allen, C. Valeriani, and P.R. ten Wolde. Forward flux sampling for rare event simulations. *J. Phys.-Condens. Mat.*, 21(463102), 2009.

[4] R.J. Allen, P.B. Warren, and P.R. ten Wolde. Sampling rare switching events in biochemical networks. *Phys. Rev. Lett.*, 94(1), 2005.

[5] A. Antoniadis and T. Sapatinas. Wavelet methods for continuous-time prediction using Hilbert-valued autoregressive processes. *J. Multivariate Anal.*, 87(1):133–158, 2003.

[6] B.C. Arnold, N. Balakrishnan, and H.N. Nagaraja. *A first course in order statistics*. Wiley Series in Probability and Mathematical Statistics: Probability and Mathematical Statistics. John Wiley & Sons Inc., New York, 1992.

[7] N. Aronszajn. Theory of reproducing kernels. *Trans. Amer. Math. Soc.*, 68:337–404, 1950.

[8] S.K. Au and J.L. Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4):263–277, 2001.

[9] S.K. Au and J.L. Beck. Subset simulation and its application to seismic risk based on dynamic analysis. *Journal of Engineering Mechanics*, 129(8):901–917, 2003.

[10] A. Barg, G.R. Blakley, and G.A. Kabatiansky. Digital fingerprinting codes: problem statements, constructions, identification of traitors. *IEEE Trans. on Signal Processing*, 51(4):960–980, April 2003.

[11] G. Biau, F. Bunea, and M.H. Wegkamp. Functional classification in Hilbert spaces. *IEEE Trans. Inform. Theory*, 51(6):2163–2172, 2005.

[12] G. Biau, F. Cérou, and A. Guyader. On the rate of convergence of the bagged nearest neighbor estimate. *J. Mach. Learn. Res.*, 11:687–712, 2010.

[13] G. Biau, F. Cérou, and A. Guyader. Rates of convergence of the functional k-nearest neighbor estimate. *IEEE Trans. Inform. Theory*, 56(4):2034–2040, 2010.

[14] G. Biau and L. Devroye. On the layered nearest neighbour estimate, the bagged nearest neighbour estimate and the random forest method in regression and classification. *J. Multivariate Anal.*, 101(10):2499–2518, 2010.

[15] D. Bosq. *Linear processes in function spaces*, volume 149 of *Lecture Notes in Statistics*. Springer-Verlag, New York, 2000.

[16] Z.I. Botev and D.P. Kroese. An efficient algorithm for rare-event probability estimation, combinatorial optimization, and counting. *Methodology and Computing in Applied Probability*, 10(4):471–505, 2008.

[17] S. Boucheron, O. Bousquet, and G. Lugosi. Theory of classification: a survey of some recent advances. *ESAIM Probab. Stat.*, 9:323–375, 2005.

[18] L. Breiman. Bagging predictors. *Mach. Learn.*, 24:123–140, 1996.

[19] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[20] J.A. Bucklew. *Introduction to rare event simulation*. Springer Series in Statistics. Springer-Verlag, New York, 2004.

[21] P. Bühlmann and B. Yu. Analyzing bagging. *Ann. Statist.*, 30(4):927–961, 2002.

[22] A. Buja and W. Stuetzle. Observations on bagging. *Statist. Sinica*, 16(2):323–351, 2006.

[23] F. Cérou, P. Del Moral, T. Furon, and A. Guyader. Sequential Monte Carlo for rare event estimation. *Statistics and Computing*, 2011.

[24] F. Cérou, P. Del Moral, and A. Guyader. A non asymptotic theorem for unnormalized Feynman-Kac particle models. *Ann. Inst. Henri Poincaré Probab. Stat.*, 47(3):629–649, 2011.

[25] F. Cérou, P. Del Moral, F. Le Gland, and P. Lezaud. Genetic genealogical models in rare event analysis. *ALEA Lat. Am. J. Probab. Math. Stat.*, 1:181–203, 2006.

[26] F. Cérou, T. Furon, and A. Guyader. On the design and optimization of Tardos probabilistic fingerprinting codes. In *Information Hiding*, volume 5284 of *Lecture Notes in Computer Science*, pages 341–356. Springer Berlin / Heidelberg, 2008.

[27] F. Cérou and A. Guyader. Adaptive multilevel splitting for rare event analysis. *Stoch. Anal. Appl.*, 25(2):417–443, 2007.

[28] F. Cérou, A. Guyader, T. Lelièvre, and D. Pommier. A multiple replica approach to simulate reactive trajectories. *The Journal of Chemical Physics*, 134(5):054108, 2011.

[29] F. Cérou, A. Guyader, R. Rubinstein, and R. Vaisman. On the use of smoothing to improve the performance of the splitting method. *Stochastic Models*, 2011.

[30] Frédéric Cérou, Teddy Furon, and Arnaud Guyader. Experimental assessment of the reliability for watermarking and fingerprinting schemes. *EURASIP J. Inf. Secur.*, 2008:6:1–6:12, January 2008.

[31] Frédéric Cérou and Arnaud Guyader. Nearest neighbor classification in infinite dimension. *ESAIM Probab. Stat.*, 10:340–355 (electronic), 2006.

[32] C. Chipot and A. Pohorille, editors. *Free Energy Calculations*, volume 86 of *Springer Series in Chemical Physics*. Springer, 2007.

[33] N. Chopin and C.P. Robert. Properties of nested sampling. *Biometrika*, 97(3):741–755, 2010.

[34] T.M. Cover. Estimation by the nearest neighbor rule. *IEEE Transactions on Information Theory*, 14(1):50–55, 1968.

[35] T.M. Cover. Rates of convergence for nearest neighbor procedures. In *Proceedings of the Hawaii International Conference on Systems Sciences*, pages 413–415, Honolulu, 1968.

[36] T.M. Cover and P.E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, IT-13(1):21–27, 1967.

[37] F. Cucker and S. Smale. On the mathematical foundations of learning. *Bull. Amer. Math. Soc. (N.S.)*, 39(1):1–49, 2002.

[38] S. Dabo-Niang and N. Rhomari. Kernel regression estimation in a Banach space. *J. Statist. Plann. Inference*, 139(4):1421–1434, 2009.

[39] T. Dean and P. Dupuis. Splitting for rare event simulation: a large deviation approach to design and analysis. *Stochastic Process. Appl.*, 119(2):562–587, 2009.

[40] P. Del Moral. *Feynman-Kac formulae, Genealogical and interacting particle systems with applications*. Probability and its Applications. Springer-Verlag, New York, 2004.

[41] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *J. R. Stat. Soc. Ser. B*, 68(3):411–436, 2006.

[42] P. Del Moral, A. Doucet, and G.W. Peters. Sharp propagation of chaos estimates for Feynman-Kac particle models. *Teor. Veroyatn. Primen.*, 51(3):552–582, 2006.

[43] P. Del Moral, F. Patras, and S. Rubenthaler. Tree based functional expansions for Feynman-Kac particle models. *Ann. Appl. Probab.*, 19(2):778–825, 2009.

[44] C. Dellago and P.G. Bolhuis. Transition Path Sampling and Other Advanced Simulation Techniques for Rare Events. In *Advances computer simulation approaches for soft matter sciences I II*, volume 221 of *Advances in polymer science*, pages 167–233. Springer, 2009.

[45] L. Devroye. On the almost everywhere convergence of nonparametric regression function estimates. *Ann. Statist.*, 9(6):1310–1319, 1981.

[46] L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*, volume 31 of *Applications of Mathematics*. Springer-Verlag, New York, 1996.

[47] T.G. Dietterich. Ensemble methods in machine learning. In *Proceedings of the First International Workshop on Multiple Classifier Systems*, MCS '00, pages 1–15, London, UK, 2000.

[48] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Statistics for Engineering and Information Science. Springer, New York, 2001.

[49] W. E, W. Ren, and E. Vanden-Eijnden. String method for the study of rare events. *Phys. Rev. B*, 66(5), 2002.

[50] W. E, W. Ren, and E. Vanden-Eijnden. Finite temperature string method for the study of rare events. *J. Phys. Chem. B*, 109(14):6688–6693, 2005.

[51] W. E and E. Vanden-Eijnden. Metastability, conformation dynamics, and transition pathways in complex systems. In *Multiscale modelling and simulation*, volume 39 of *Lect. Notes Comput. Sci. Eng.*, pages 35–68. Springer, Berlin, 2004.

[52] D.E. Edmunds and H. Triebel. *Function spaces, entropy numbers, differential operators*, volume 120 of *Cambridge Tracts in Mathematics*. Cambridge University Press, Cambridge, 1996.

[53] L.C. Evans and R.F. Gariepy. *Measure theory and fine properties of functions*. Studies in Advanced Mathematics. CRC Press, Boca Raton, FL, 1992.

[54] A.K. Faradjian and R. Elber. Computing time scales from reaction coordinates by milestoning. *J. Chem. Phys.*, 120(23):10880–10889, 2004.

[55] H. Federer. *Geometric measure theory*. Die Grundlehren der mathematischen Wissenschaften, Band 153. Springer-Verlag, New York, 1969.

[56] F. Ferraty and P. Vieu. *Nonparametric functional data analysis*. Springer Series in Statistics. Springer, New York, 2006.

[57] E. Fix and J.L. Hodges. Discriminatory analysis, non-parametric discrimination: consistency properties. Technical report, USAF Scholl of aviation and medicine, Randolph Field, 1951.

[58] E. Fix and J.L. Hodges. Discriminatory analysis: Small sample performance. Technical report, USAF Scholl of aviation and medicine, Randolph Field, 1952.

[59] M.I. Freidlin and A.D. Wentzell. *Random Perturbations of Dynamical Systems*. Springer-Verlag, 1984.

[60] J.H. Friedman and P. Hall. On bagging and nonlinear estimation. *J. Statist. Plann. Inference*, 137(3):669–683, 2007.

[61] M.J.J. Garvels. *The splitting method in rare event simulation*. Thesis, University of Twente, 2000.

[62] P. Glasserman, P. Heidelberger, P. Shahabuddin, and T. Zajic. Multilevel splitting for estimating rare event probabilities. *Oper. Res.*, 47(4):585–600, 1999.

[63] P.W. Glynn and W. Whitt. The asymptotic efficiency of simulation estimators. *Oper. Res.*, 40(3):505–520, 1992.

[64] A. Guyader, N. Hengartner, and E. Matzner-Løber. Simulation and estimation of extreme quantiles and extreme probabilities. *Applied Mathematics and Optimization*, 64:171–196, 2011. 10.1007/s00245-011-9135-z.

[65] L. Györfi. On the rate of convergence of nearest neighbor rules. *IEEE Trans. Inform. Theory*, 24(4):509–512, 1978.

[66] L. Györfi, M. Kohler, A. Krzyżak, and H. Walk. *A Distribution-Free Theory of Nonparametric Regression*. Springer Series in Statistics. Springer-Verlag, New York, 2002.

[67] P. Hall, D.S. Poskitt, and B. Presnell. A functional data-analytic approach to signal discrimination. *Technometrics*, 43(1):1–9, 2001.

[68] P. Hall and R.J. Samworth. Properties of bagged nearest neighbour classifiers. *Journal Of The Royal Statistical Society Series B*, 67(3):363–379, 2005.

[69] J.M. Hammersley and D.C. Handscomb. *Monte Carlo methods*. Methuen & Co. Ltd., London, 1965.

[70] G. Hummer. From transition paths to transition states and rate coefficients. *J. Chem. Phys.*, 120(2):516–523, 2004.

[71] IBM, 2001. `www.trl.ibm.com/projects/RightsManagement/datahiding/dhvgx_e.htm`.

[72] I.A. Ibragimov and R.Z. Khasminskii. Nonparametric regression estimation. *Dokl. Akad. Nauk SSSR*, 252(4):780–784, 1980.

[73] I.A. Ibragimov and R.Z. Khasminskii. *Statistical estimation*, volume 16 of *Applications of Mathematics*. Springer-Verlag, New York, 1981.

[74] I.A. Ibragimov and R.Z. Khasminskii. Bounds for the quality of nonparametric estimation of regression. *Teor. Veroyatnost. i Primenen.*, 27(1):81–94, 1982.

[75] A.M. Johansen, P. Del Moral, and A. Doucet. Sequential Monte Carlo samplers for rare events. In *Proceedings of the 6th International Workshop on Rare Event Estimation*, pages 256–267, 2006.

[76] H. Kahn and T.E. Harris. Estimation of particle transmission by random sampling. *National Bureau of Standards Appl. Math. Series*, 12:27–30, 1951.

[77] A. Kneip and T. Gasser. Statistical tools to analyze data representing a sample of curves. *Ann. Statist.*, 20(3):1266–1305, 1992.

[78] D.E. Knuth. *The art of computer programming. Volume 3*. Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont., 1973.

[79] A.N. Kolmogorov and V.M. Tihomirov. ε-entropy and ε-capacity of sets in functional space. *Amer. Math. Soc. Transl. (2)*, 17:277–364, 1961.

[80] S.R. Kulkarni and S.E. Posner. Rates of convergence of nearest neighbor estimation under arbitrary sampling. *IEEE Trans. Inform. Theory*, 41(4):1028–1039, 1995.

[81] A. Lagnoux. Rare event simulation. *Probability in the Engineering and Informational Sciences*, 20(1):45–66, 2006.

[82] F. Le Gland and N. Oudjane. A sequential particle algorithm that keeps the particle system alive. In *Stochastic hybrid systems*, volume 337 of *Lecture Notes in Control and Inform. Sci.*, pages 351–389. Springer, Berlin, 2006.

[83] P. L'Ecuyer, J. Blanchet, B. Tuffin, and P.W. Glynn. Asymptotic robustness of estimators in rare-event simulation. *ACM Transactions on Modeling and Computer Simulation*, 18(3):1269–1283, 2008.

[84] T. Lelièvre, M. Rousset, and G. Stoltz. *Free energy computations: A mathematical perspective*. Imperial College Press, 2010.

[85] Y. Lin and Y. Jeon. Random forests and adaptive nearest neighbors. *J. Amer. Statist. Assoc.*, 101(474):578–590, 2006.

[86] L. Maragliano, E. Vanden-Eijnden, and B. Roux. Free energy and kinetics of conformational transitions from Voronoi tessellated milestoning with restraining potentials. *J. Chem. Theory Comput.*, 5(10):2589–2594, 2009.

[87] N. Merhav and E. Sabbag. Optimal watermarking embedding and detection strategies under limited detection resources. *IEEE Trans. on Inf. Theory*, 54(1):255–274, 2008.

[88] P. Metzner, C. Schütte, and E. Vanden-Eijnden. Illustration of transition path theory on a collection of simple examples. *J. Chem. Phys.*, 125(8):084110, 2006.

[89] M. Mitzenmacher and E. Upfal. *Probability and computing*. Cambridge University Press, Cambridge, 2005.

[90] R. Motwani and P. Raghavan. *Randomized algorithms*. Cambridge University Press, Cambridge, 1995.

[91] S. Park, M.K. Sener, D. Lu, and K. Schulten. Reaction paths based on mean first-passage times. *J. Chem. Phys.*, 119(3):1313–1319, 2003.

[92] D. Preiss. Gaussian measures and the density theorem. *Comment. Math. Univ. Carolin.*, 22(1):181–193, 1981.

[93] D. Preiss. Dimension of metrics and differentiation of measures. In *General topology and its relations to modern analysis and algebra, V (Prague, 1981)*, volume 3 of *Sigma Ser. Pure Math.*, pages 565–568. Heldermann, Berlin, 1983.

[94] D. Preiss and J. Tišer. Differentiation of measures on Hilbert spaces. In *Measure theory, Oberwolfach 1981*, volume 945 of *Lecture Notes in Math.*, pages 194–207. Springer, Berlin, 1982.

[95] D. Psaltis, R.R. Snapp, and S.S. Venkatesh. On the finite sample performance of the nearest neighbor classifier. *IEEE Transactions on Information Theory*, 40(3):820–837, 1994.

[96] J. Ramsay and B.W. Silverman. *Functional Data Analysis*. Springer, 1997.

[97] J.A. Rice and B.W. Silverman. Estimating the mean and covariance structure nonparametrically when the data are curves. *J. Roy. Statist. Soc. Ser. B*, 53(1):233–243, 1991.

[98] C.P. Robert and G. Casella. *Monte Carlo statistical methods*. Springer Texts in Statistics. Springer-Verlag, New York, second edition, 2004.

[99] G.O. Roberts, A. Gelman, and W.R. Gilks. Weak convergence and optimal scaling of random walk Metropolis algorithms. *Ann. Appl. Probab.*, 7(1):110–120, 1997.

[100] M.N. Rosenbluth and A.W. Rosenbluth. Monte Carlo calculation of the average extension of molecular chains. *Journal of Chemical Physics*, 23(2):356–359, 1955.

[101] R. Rubinstein. The Gibbs cloner for combinatorial optimization, counting and sampling. *Methodol. Comput. Appl. Probab.*, 11(4):491–549, 2009.

[102] R. Rubinstein. Randomized algorithms with splitting: why the classic randomized algorithms do not work and how to make them work. *Methodol. Comput. Appl. Probab.*, 12(1):1–50, 2010.

[103] M.J. Schervish. *Theory of statistics*. Springer Series in Statistics. Springer-Verlag, New York, 1995.

[104] B. Schölkopf and A.J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[105] B. Selman, H. Kautz, and B. Cohen. Local search strategies for satisfiability testing. In *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 521–532. AMS, 1995.

[106] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[107] J. Skilling. Nested sampling for general Bayesian computation. *Bayesian Anal.*, 1(4):833–859, 2006.

[108] J. Skilling. Nested sampling for Bayesian computations. In *Bayesian statistics 8*, Oxford Sci. Publ., pages 491–524. Oxford Univ. Press, Oxford, 2007.

[109] B.M. Steele. Exact bootstrap k-nearest neighbor learners. *Mach. Learn.*, 74:235–255, March 2009.

[110] C.J. Stone. Consistent nonparametric regression. *Ann. Statist.*, 5(4):595–645, 1977.

[111] G. Tardos. Optimal probabilistic fingerprint codes. In *Proc. of the 35th annual ACM symposium on theory of computing*, pages 116–125, San Diego, 2003.

[112] T.S. van Erp and P.G. Bolhuis. Elaborating transition interface sampling methods. *J. Comp. Phys.*, 205(1):157–181, 2005.

[113] T.S. van Erp, D. Moroni, and P.G. Bolhuis. A novel path sampling method for the calculation of rate constants. *J. Chem. Phys.*, 118(17):7762–7774, 2003.

[114] W.R. van Zwet. *Convex transformations of random variables*, volume 7 of *Mathematical Centre Tracts*. Mathematisch Centrum, Amsterdam, 1964.

[115] S.S. Venkatesh, R.R. Snapp, and D. Psaltis. Bellman strikes again! the growth rate of sample complexity with dimension for the nearest neighbor classifier. In *Proceedings of the fifth annual workshop on Computational learning theory*, COLT '92, pages 93–102, 1992.

[116] D.-X. Zhou. The covering number in learning theory. *J. Complexity*, 18(3):739–767, 2002.