High dimensional importance sampling through projections on a low dimensional subspace

Maxime El Masri

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Institut Supérieur de l'Aéronautique et de l'Espace

---

Présentée et soutenue par

Maxime EL MASRI

Le 16 mars 2022

Echantillonnage préférentiel en grande dimension via des projections dans un sous-espace de petite dimension

---

Ecole doctorale : **EDMITT - Ecole Doctorale Mathématiques, Informatique et Télécommunications de Toulouse**

Spécialité : **Mathématiques et Applications**

Unité de recherche :
ISAE-ONERA MOIS MOdélisation et Ingénierie des Systèmes

Thèse dirigée par
Florian SIMATOS et Jérôme MORIO

Jury

M. Jean-Michel MARIN, Rapporteur
M. Bruno TUFFIN, Rapporteur
M. Jean-Marc BOURINET, Examinateur
M. Olivier ZAHM, Examinateur
M. Florian SIMATOS, Directeur de thèse
M. Jérôme MORIO, Co-directeur de thèse
Mme Gersende FORT, Présidente

Remerciements

Mes premiers remerciements vont évidemment à mes directeurs de thèse : Jérôme Morio et Florian Simatos. Merci de votre implication, de votre soutien permanent, et de vos conseils qui m'ont amené à progresser pendant ces trois (et même quatre) années. Votre passion et votre intérêt pour mes travaux m'ont aussi permis de garder la motivation malgré les difficultés, et j'en suis extrêmement reconnaissant.

Je tiens ensuite à remercier les membres du jury qui ont accepté d'examiner ou de rapporter ma thèse. Merci à Jean-Michel Marin et Bruno Tuffin pour leurs rapports que j'ai eu plaisir à lire. Merci à Gersende Fort d'avoir présidé le jury de thèse, ainsi qu'à Jean-Marc Bourinet et Olivier Zahm d'avoir accepté d'en être examinateurs. Merci à tous pour vos commentaires bienveillants et vos questions intéressantes lors de la soutenance.

Il y a de nombreuses autres personnes qui m'ont accompagné durant cette aventure et je tiens également à leur témoigner ma reconnaissance. Il est possible que j'en oublie et je m'en excuse, mais tous ceux que j'ai côtoyés ou juste croisés à l'ONERA, l'ISAE, ou ailleurs pendant ces quatre dernières années, peuvent se sentir concernés par ces remerciements. Les deux premiers que j'ai envie de citer ici sont les deux doctorants qui ont débuté leur thèse au même moment que moi : Gaspard et Alexandre. Gaspard, nous avons travaillé dans le même bureau à l'ONERA presque du début à la fin de nos thèses et nous avons pu partager nos difficultés et nos bons moments. Merci pour les discussions de maths, de sport, de politique, et surtout les débriefs des parties de tarot ! Alexandre, merci pour les longues pauses à l'ISAE à partager nos doutes au sujet de la thèse, à refaire le monde et à disserter sur notre passion commune pour le tennis (ah, ce Wimbledon 2019...). Je remercie ensuite tous les autres doctorants/post-docs/stagiaires/apprentis, pour la bonne ambiance et les nombreuses parties de tarot. Les anciens d'abord, Pierre (merci de m'avoir fait découvrir le squash), Gabriel (merci pour les quelques parties de foot), Rémy P, Morgane, Tiphaine (ravi d'avoir profité de ton expertise sur le biathlon, et surtout sur Martin Fourcade), Vincent, Julien M, Rémi P (mon premier co-bureau)... et ceux qui ont encore quelques mois ou années à trimer, Marco (sans rancune pour l'Euro 2020), Marie, Charles, Julien D, Inès, Florent, Vincenzo, Paul, Rémy C, Luiz, ainsi que les habitants du 2ème étage (mais qu'on accepte quand même au 1er) Camille, Sam, Anouck, et j'en oublie sans doute. Merci aussi à tous les permanents du département DTIS de l'ONERA et du DISC à l'ISAE avec qui j'ai pu échanger et qui permettent aussi d'avoir une bonne ambiance au sein de ces deux laboratoires.

Je tiens aussi à remercier tous mes amis qui m'ont permis de m'évader et de penser à autre chose durant cette thèse, lors de restos, soirées jeux, ou voyages. Merci donc à mes vieux amis de lycée, que j'ai toujours plaisir à retrouver après toutes ces années, Damien L (merci pour ces soirées jeux vidéos qu'on a passées ...et à toutes celles qui sont à venir !), Stéphane, Jean-Baptiste, Paul, et Félix, ainsi qu'à ceux rencontrés lors de mes années universitaires, Jean-Marc, Eva, Florian, Julie, Damien B, Anne, Hugo, Floriane, Valentin, Maylis, Baptiste. Et bien entendu, merci à Laura de me soutenir (et de me supporter !) sans relâche chaque jour qui passe.

J'adresse également mes remerciements à tous mes nouveaux collègues de l'IMT que je côtoie depuis quelques semaines. J'en profite aussi pour remercier mon club de tennis, l'US Ramonville, et mes coéquipiers du challenge Hermet pour cette fantastique épopée qui nous a menés en finale du tournoi 2021, et qui m'ont permis de garder un équilibre entre travail et activités sportives.

Pour finir, je remercie toute ma famille, en particulier mes parents, mon frère et mes sœurs, sur qui je peux toujours compter.

Table des matières

Introduction

Contexte Dans de nombreux domaines scientifiques, on s'intéresse à la performance d'un système définie par l'espérance mathématique d'une variable aléatoire. Cette variable est souvent le résultat d'un code de simulation potentiellement couteux en temps de calcul, qui est défini par une fonction de performance dépendant de variables extérieures aléatoires et décrivant le comportement du système. L'estimation de l'espérance peut alors être primordiale pour prévoir une éventuelle défaillance du système et éviter un potentiel accident ou simplement pour améliorer sa performance.

Un premier exemple provient de la théorie du signal, où l'on s'intéresse au problème de la localisation d'une cible par un réseau de plusieurs capteurs dont les mesures sont entachées d'une erreur (voir [Bugallo et al., 2017] et [Ihler et al., 2005]). Le but est de retrouver la position de la cible étant donné les mesures observées des différents capteurs. Cette quantité est calculée par une espérance (qui minimise l'erreur quadratique moyenne entre la position réelle et les observations) que l'on cherche alors à estimer. Dans ce cas, les variables extérieures sont les erreurs de mesure sur les observations commises par chaque capteur.

Lorsque l'on s'intéresse à la fiabilité d'un système, la quantité à estimer est une probabilité de défaillance. Un exemple en aéronautique est donné par l'estimation de la probabilité de retombée extrême d'un drone en cas de panne en vol (voir par exemple [Morio et al., 2021]). Dans ce cas, la fonction de performance est un code simulant la trajectoire du drone jusqu'à l'impact au sol et mesurant la distance au point de départ. Cette fonction dépend de plusieurs paramètres aléatoires agissant sur la trajectoire du véhicule (comme la vitesse du vent, la hauteur de vol...). La probabilité que le véhicule dépasse une certaine distance est alors importante à estimer afin de déterminer une zone de sécurité autour du drone.

Il existe différentes méthodes pour estimer ces espérances, notamment l'échantillonnage préférentiel qui est un sujet de recherche particulièrement actif ces dernières années et qui est l'objet d'étude principal de cette thèse. L'échantillonnage préférentiel ("*Importance Sampling*", IS) est une méthode stochastique, basée sur la méthode de Monte-Carlo. [Kahn, 1950] est un des premiers à l'utiliser pour l'estimation d'une probabilité d'événement rare en physique des particules. Le principe est de générer un échantillon selon une loi de probabilité auxiliaire, au lieu de la loi initiale comme dans la méthode de Monte-Carlo, avant de calculer un estimateur faisant intervenir des poids d'importance. Une loi auxiliaire bien choisie permet de réduire la variance de l'estimateur classique de Monte-Carlo, et entraine ainsi une diminution du nombre d'appels au code de calcul, ce dernier pouvant être très couteux. De nombreux algorithmes basés sur l'échantillonnage préférentiel ont alors vu le jour, en particulier dans les domaines de la fiabilité des systèmes et de l'inférence bayésienne, la plupart reposant sur la mise à jour de densités de probabilité paramétriques, et donc des paramètres définissant ces densités, de manière itérative.

Position du problème et objectifs de la thèse Les algorithmes d'échantillonnage préférentiel proposés dans la littérature ont montré leur efficacité lorsque la dimension du problème est petite, ou autrement dit lorsque le nombre de variables extérieures en entrée du code de simulation est assez faible. Dans l'échantillonnage préférentiel paramétrique le nombre de variables d'entrée est généralement lié au nombre de paramètres estimés dans l'algorithme. Cependant, la performance de ces méthodes se dégrade dès lors que la dimension augmente. En effet, dans ce cas, les algorithmes d'IS, et l'échantillonnage préférentiel en général, deviennent imprécis et peuvent entrainer de grandes erreurs dans l'estimation finale de l'espérance. Cette inefficacité est notamment due aux approximations effectuées sur chaque dimension qui entrainent une erreur d'estimation d'autant plus grande que la dimension est élevée.

Des auteurs ont tenté de donner des premières solutions pour améliorer l'efficacité de l'IS en grande dimension. [Rubinstein and Glynn, 2009] proposent par exemple de sélectionner les paramètres les plus influents pour n'en mettre à jour qu'un petit nombre à chaque étape de leur algorithme. La diminution du nombre de paramètres estimés entraine bien une réduction de l'erreur d'estimation de l'espérance mais la méthode est inefficace lorsque toutes les variables sont influentes et n'est adaptée qu'à un nombre restreint de problèmes. [Wang and Song, 2016] et [Papaioannou et al., 2019a] suggèrent quant à eux d'utiliser des densités auxiliaires plus efficaces en grande dimension et impliquant l'estimation d'un nombre réduit de paramètres. Ces méthodes améliorent la précision de l'estimation comparées aux algorithmes d'IS classiques mais peuvent malgré tout nécessiter un important budget de simulation. D'autres techniques, comme celles proposées dans [Koblents and Míguez, 2015] ou [El-Laham et al., 2018], préconisent de transformer les poids d'IS apparaissant dans l'estimation des paramètres afin d'éviter les valeurs aberrantes (i.e. particulièrement éloignées de la plupart des autres valeurs) et réduire leur variance, mais elles restent relativement inefficaces pour des dimensions dépassant quelques dizaines. Enfin, [Uribe et al., 2021] proposent de construire un sous-espace de petite dimension dans lequel mettre à jour les paramètres après avoir appliqué une projection dans ce sous-espace. Cette méthode est particulièrement efficace et précise pour de grandes dimensions mais la construction du sous-espace nécessite l'évaluation du gradient de la fonction de performance, qui peut lui aussi être très couteux à estimer ou même ne pas être disponible.

L'objectif principal de la thèse est alors de proposer de nouvelles méthodes visant à améliorer la précision de l'estimation par échantillonnage préférentiel en grande dimension, en réduisant le nombre de paramètres estimés, tout en gardant un budget de simulation raisonnable. Pour cela, nous privilégions l'utilisation d'une projection dans un sous-espace, sans faire d'hypothèse de régularité sur la fonction de performance et donc sans évaluation du gradient. Enfin, les techniques de projection étudiées sont couplées à un algorithme d'échantillonnage préférentiel adaptatif afin d'estimer une espérance en grande dimension avec un faible budget de simulation.

Plan de la thèse

Ce manuscrit comprend cinq chapitres, les deux premiers étant des chapitres d'état de l'art autour des méthodes d'échantillonnage préférentiel, et les trois suivants présentent les principales contributions de cette thèse pour améliorer l'IS en grande dimension.

Le chapitre 1 définit le problème d'estimation d'une espérance et introduit les notations et hypothèses utilisées dans tout le manuscrit. La question de la résolution de ce type de problèmes est abordée avec la présentation de différentes méthodes d'estimation, en mettant l'accent sur l'échantillonnage préférentiel paramétrique (en particulier dans le cadre gaussien). Nous évoquons ensuite deux domaines de recherche dans lesquels l'IS est largement appliqué. Le premier est celui de la fiabilité des systèmes, dans lequel il est souvent nécessaire d'utiliser l'IS pour estimer des

probabilités d'événements rares. Le second domaine d'application est l'inférence bayésienne où l'on cherche à estimer une ou plusieurs espérances dépendant d'une loi a posteriori étant donné des observations. Cette loi n'étant connue qu'à une constante près, l'échantillonnage préférentiel auto-normalisé est alors utile pour estimer ces espérances à l'aide d'une loi auxiliaire. Plusieurs algorithmes d'IS adaptatifs spécifiques aux deux domaines sont décrits.

Dans le chapitre 2, l'inefficacité de l'échantillonnage préférentiel paramétrique en grande dimension est étudiée, en commençant par l'illustration de la dégradation de l'algorithme d'entropie croisée, utilisé en fiabilité. Nous décrivons ensuite deux phénomènes permettant d'expliquer cette dégradation. Le phénomène de dégénérescence des poids dans les algorithmes adaptatifs d'échantillonnage préférentiel est abordé ainsi que le problème de l'estimation de matrices de covariance de grande taille que nous illustrons par la dégradation de la divergence de Kullback-Leibler. Enfin, nous proposons un bref état de l'art des techniques proposées dans la littérature pour remédier à la défaillance de l'échantillonnage préférentiel en grande dimension.

Le chapitre 3 a pour but d'étudier l'effet d'une projection des paramètres dans un sous-espace sur la précision de l'estimation par échantillonnage préférentiel. Un calcul impliquant la divergence de Kullback-Leibler dans un cas simple montre que celle-ci peut être réduite en choisissant une projection pertinente pour estimer les paramètres. Des simulations numériques sont ensuite réalisées pour montrer que l'estimation de la matrice de covariance dans un sous-espace de petite dimension entraine souvent l'amélioration des résultats d'estimation d'intégrales, même lorsque la projection dans le sous-espace est choisie de manière naïve.

Le chapitre 4 est consacré à la recherche de directions de projection pertinentes pour l'estimation de la matrice de covariance, afin d'améliorer les résultats d'estimation observés dans le chapitre 3 avec des projections naïves. La première direction influente identifiée donne une projection dans un sous-espace de dimension 1 dans lequel la variance diminue, notamment dans le cadre des événements rares. Des directions optimales sont ensuite déterminées par minimisation de la divergence Kullback-Leibler. Ces deux contributions sont ensuite testées dans différents exemples d'estimation dans un cadre théorique.

Pour appliquer les techniques de projection du chapitre 4, le dernier chapitre (5) vise à obtenir le couplage d'un algorithme adaptatif d'échantillonnage préférentiel avec ces projections. Plusieurs algorithmes sont mis en place pour l'estimation d'une probabilité d'événement rare. Les algorithmes en question proposent ainsi un couplage de l'algorithme d'entropie croisée avec chacune des deux méthodes de projection du chapitre 4. Ces approches sont ensuite confrontées à différents cas-tests analytiques.

Chapitre 1

Méthodes d'échantillonnage et d'estimation

Sommaire

1.1 Contexte et définition du problème

1.1.1 Modèle boite noire entrée-sortie et estimation d'une intégrale

Dans de nombreux domaines scientifiques, on s'intéresse à la modélisation d'un système complexe et d'une quantité d'intérêt associée, décrite par une application du type :

$$\begin{array}{rccc} \phi: & \mathcal{X} \subset \mathbb{R}^n & \longrightarrow & \mathcal{Y} \subset \mathbb{R} \\ & \mathbf{x} & \mapsto & y = \phi(\mathbf{x}). \end{array}$$

Cette fonction, appelée **fonction d'intérêt**, représente un code de calcul, potentiellement couteux, simulant l'évolution du système en question, que ce soit en physique [Kahn, 1950], en ingénierie [Melchers, 1989], ou en finance [Glasserman, 2004] par exemple. L'application ϕ est une fonction déterministe qui prend en entrée un vecteur $\mathbf{x} = (x_1, \ldots, x_n)^\top$, représentant les paramètres extérieurs agissant sur le système et retourne une sortie $y = \phi(\mathbf{x})$ qui est un nombre réel. Elle est aussi considérée comme une **boite noire**, ce qui signifie en pratique qu'elle n'a pas nécessairement d'expression analytique et que l'utilisateur n'a accès qu'à la sortie y du code, étant donnée une entrée $\mathbf{x}$. De plus, une telle application est souvent couteuse à évaluer (un seul appel à ϕ pouvant prendre plusieurs heures), et il est donc préférable de minimiser le nombre total d'appels à ϕ. Pour cette raison, dans tout le manuscrit on désigne par **budget de simulation**, le nombre d'appels à la fonction ϕ, toutes les autres applications étant considérées comme peu couteuses comparées à un appel à ϕ.

Par ailleurs, les entrées d'un tel système sont considérées comme étant aléatoires, du fait de l'incertitude des paramètres due à d'éventuelles erreurs de mesure, des approximations numériques ou à des aléas provenant de phénomènes naturels. Ces paramètres d'entrée sont ainsi représentés par un vecteur aléatoire $\mathbf{X}$, à valeurs dans $\mathbb{R}^n$. Dans cette thèse, on suppose que la loi de probabilité de $\mathbf{X}$ est absolument continue par rapport à la mesure de Lebesgue et possède une densité, notée f, définie sur $\mathbb{R}^n$. La sortie est alors elle aussi aléatoire et modélisée par une variable aléatoire réelle $Y = \phi(\mathbf{X})$, supposée de variance finie. Différentes quantités d'intérêt peuvent alors être étudiées afin de comprendre le comportement stochastique de Y : les moments de la variable Y, une probabilité de dépassement de seuil de Y, ou même la recherche complète de la fonction de répartition ou de la densité de probabilité de Y.

Dans cette thèse, on s'intéressera en particulier à l'estimation de l'espérance de la sortie et de probabilités de dépassement de seuil. Dans toute la suite, on cherche ainsi à estimer l'intégrale :

$$E = \mathbb{E}_f(\phi(\mathbf{X})) = \int_{\mathbb{R}^n} \phi(\mathbf{x}) f(\mathbf{x}) \mathrm{d}\mathbf{x}, \tag{1.1}$$

où ϕ est supposée être à valeurs dans $\mathbb{R}_+$. La fonction ϕ étant couteuse à évaluer, on s'attachera à garder un budget de simulation modéré pour réaliser l'estimation. La dimension du problème

d'estimation est la dimension de l'espace des entrées, notée n. De plus, on suppose que l'on dispose de suffisamment d'informations (voir la section 1.1.2) sur la densité de probabilité f, de sorte que l'on puisse toujours se ramener à une densité gaussienne standard, sauf mention explicite du contraire. Cette hypothèse couramment adoptée (notamment en fiabilité des systèmes, comme évoqué en section 1.3) est justifiée dans le paragraphe suivant 1.1.2.

1.1.2 Transformations isoprobabilistes

Pour estimer une espérance, il est pratique de se ramener à l'espace normal standard, c'est-à-dire se ramener au cas où f est la densité de la loi normale $\mathcal{N}(\mathbf{0}, I_n)$, de vecteur moyenne $\mathbf{0} \in \mathbb{R}^n$ et de covariance I_n, la matrice identité de taille n. Rappelons que cette densité est définie, pour tout $\mathbf{x} \in \mathbb{R}^n$, par :

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}} \exp\left(-\frac{1}{2}\|\mathbf{x}\|^2\right), \tag{1.2}$$

où $\|\mathbf{x}\|$ est la norme euclidienne sur $\mathbb{R}^n$, du vecteur $\mathbf{x}$. Pour réaliser ce changement, il existe des transformations isoprobabilistes permettant de transformer un vecteur aléatoire $\mathbf{X}$ en un nouveau vecteur $\mathbf{U} \sim \mathcal{N}(\mathbf{0}, I_n)$. Ces transformations sont définies par un difféomorphisme

$$T : \mathbf{x} \in \mathbb{R}^n \to \mathbf{u} \in \mathbb{R}^n$$

tel que $\mathbf{U} = T(\mathbf{X})$ suive une loi normale, $\mathbf{U} \sim \mathcal{N}(\mathbf{0}, I_n)$. Ainsi, quitte à effectuer le changement de variables $\mathbf{x} = T^{-1}(\mathbf{u})$ dans l'intégrale (1.1), et à changer ϕ en $\phi \circ T^{-1}$, on peut supposer que f est la densité gaussienne standard $\mathcal{N}(\mathbf{0}, I_n)$. Suivant les informations dont on dispose sur la loi de $\mathbf{X}$, on peut appliquer différents types de transformations. La transformation de Nataf, introduite par [Nataf, 1962], puis par [Liu and Der Kiureghian, 1986] dans le cadre fiabiliste, nécessite la connaissance des lois marginales du vecteur $\mathbf{X}$ (autrement dit les lois des $X_1, \ldots, X_n$) ainsi que les coefficients de corrélations entre chaque variable. Chaque composante X_i est transformée en une variable gaussienne, grâce notamment aux fonctions de répartition des X_i et de la gaussienne univariée, puis le vecteur normal obtenu est réduit en utilisant la matrice des corrélations. Le second type d'applications permettant de se ramener à l'espace normal standard est la transformation de Rosenblatt (développée par [Rosenblatt, 1952] puis [Hohenbichler and Rackwitz, 1981] en fiabilité), applicable dès que la loi jointe du vecteur $\mathbf{X}$ est connue, en utilisant les lois conditionnelles en chaque variable. Ces deux applications ne seront pas détaillées dans ce manuscrit mais le lecteur intéressé peut se référer à [Bourinet, 2018] dans lequel elles sont décrites.

Travailler avec un vecteur gaussien centré réduit présente plusieurs avantages. En effet, la mise en œuvre des calculs théoriques et des simulations numériques est plus aisée avec des vecteurs gaussiens, grâce notamment à l'existence de nombreuses formules explicites, et au fait que toutes les composantes du vecteur sont indépendantes et normalisées (c'est-à-dire de variances toutes égales à 1). C'est pourquoi, dans toute la suite de ce manuscrit, lors de l'estimation d'une espérance, on considèrera que f est la densité de la loi $\mathcal{N}(\mathbf{0}, I_n)$, sauf mention explicite du contraire.

1.2 Méthodes stochastiques d'estimation d'une espérance

1.2.1 La méthode de Monte-Carlo

Dans cette partie, nous abordons des méthodes classiques d'estimation de l'intégrale E (1.1). Les méthodes déterministes pour calculer numériquement une intégrale (typiquement les méthodes

de quadrature) sont efficaces en petite dimension et convergent plus rapidement en dimension 1 que la méthode de Monte-Carlo décrite ci-dessous (voir la remarque 1.2.1). En revanche, elles sont contraignantes car elles nécessitent des hypothèses de régularité sur ϕ et deviennent extrêmement couteuses lorsque la dimension augmente (voir par exemple [Hinrichs et al., 2014] ou [Dimov, 2008]). En effet, ce phénomène appelé le "fléau de la dimension" ("*curse of dimensionality*") entraine de grandes imprécisions dans l'approximation de l'intégrale. C'est pourquoi il est préférable d'utiliser des méthodes stochastiques, moins contraignantes et nécessitant souvent un budget de simulation plus faible, comme la méthode de Monte-Carlo.

Cette technique consiste à approcher l'intégrale E en calculant une moyenne empirique à partir d'un échantillon tiré aléatoirement selon la loi de densité f. Ainsi l'estimateur de Monte-Carlo de l'espérance E s'écrit :

$$\hat{E}_N^{\mathrm{MC}} = \frac{1}{N}\sum_{i=1}^{N}\phi(\mathbf{X}_i), \tag{1.3}$$

où $\mathbf{X}_1, \ldots, \mathbf{X}_N$ est un échantillon de N réalisations indépendantes et identiquement distribuées (échantillon i.i.d.) du vecteur aléatoire $\mathbf{X}$ de loi de densité f. Cet estimateur est sans biais ($\mathbb{E}(\hat{E}_N^{\mathrm{MC}}) = E$), par linéarité de l'espérance, et la loi forte des grands nombres implique qu'il est consistant, c'est-à-dire que $\hat{E}_N^{\mathrm{MC}}$ tend presque sûrement vers E lorsque la taille de l'échantillon N tend vers $+\infty$. L'avantage de cet estimateur, notamment par rapport aux méthodes de quadrature évoquées ci-dessus, est que son erreur n'est pas affectée par l'augmentation de la dimension. En effet, sa variance est égale à :

$$\begin{aligned}\mathrm{Var}_f\left(\hat{E}_N^{\mathrm{MC}}\right) &= \frac{1}{N^2}\sum_{i=1}^{N}\mathrm{Var}_f\left(\phi(\mathbf{X}_i)\right)\\ &= \frac{1}{N}\mathrm{Var}_f(\phi(\mathbf{X})).\end{aligned}$$

Si $\mathrm{Var}_f(\phi(\mathbf{X}))$ prend de grandes valeurs, il faut alors accroitre la taille de l'échantillon N pour avoir une précision raisonnable, ce qui peut être problématique pour estimer l'intégrale lorsque ϕ est couteuse. C'est pourquoi des méthodes alternatives permettant de réduire la variance, en gardant une faible taille d'échantillon N, ont été développées. Dans cette thèse, nous nous concentrons sur la méthode d'échantillonnage préférentiel, abordée dans la section 1.2.2.

Remarque 1.2.1. Le théorème central limite assure la convergence en loi de la méthode de Monte-Carlo à une vitesse de $1/\sqrt{N}$:

$$\sqrt{N}\frac{\hat{E}_N^{\mathrm{MC}} - E}{\sqrt{\mathrm{Var}_f(\phi(\mathbf{X}))}} \underset{N\to+\infty}{\longrightarrow} \mathcal{N}(0,1).$$

Notons que les approches déterministes sont en général plus rapides en dimension 1, avec par exemple des vitesses de convergence en $1/N^2$ pour la méthode des trapèzes et $1/N^4$ pour la méthode de Simpson, si ϕ est suffisamment régulière. Mais la méthode de Monte-Carlo ne nécessite pas d'hypothèse de régularité de ϕ et son erreur n'est pas affectée par la dimension contrairement aux méthodes déterministes.

Il est néanmoins possible de réduire cette vitesse de convergence, et donc la variance de l'estimateur, avec les méthodes de type Quasi-Monte-Carlo (QMC) qui offrent des vitesses de convergence

de l'ordre de $\ln(N)^n/N$ (voir par exemple [Niederreiter, 1992] ou [L'Ecuyer and Lemieux, 2002]). En se ramenant à une loi uniforme sur $[0,1]^n$, ces approches consistent à choisir une suite, dite à discrépance faible, à la place d'un échantillon aléatoire dans la méthode de Monte-Carlo. Une telle suite permet d'obtenir un échantillon mieux réparti sur $[0,1]^n$ qu'en tirant aléatoirement et d'avoir une convergence plus rapide. Néanmoins, les techniques QMC exigent des conditions de régularité sur ϕ pouvant être restrictives, et sont moins robustes à la dimension que la méthode de Monte-Carlo.

1.2.2 L'échantillonnage préférentiel pour réduire la variance de l'estimateur

L'échantillonnage préférentiel (*Importance Sampling*, IS) a été introduit (à l'origine par [Kahn, 1950]) dans le but de réduire la variance de l'estimateur de Monte-Carlo. Il consiste à échantillonner selon une loi auxiliaire qui, si elle est bien choisie, permet de diminuer la variance.

Soit g la densité de la loi auxiliaire, telle que $g(\mathbf{x}) = 0 \Rightarrow f(\mathbf{x}) = 0$, pour tout $\mathbf{x} \in \mathbb{R}^n$ (ou autrement dit telle que la loi initiale de densité f soit absolument continue par rapport à la loi de g). L'intégrale E peut être réécrite de la manière suivante :

$$E = \int_{\mathbb{R}^n} \phi(\mathbf{x})L(\mathbf{x})g(\mathbf{x})\mathrm{d}\mathbf{x} = \mathbb{E}_g\left(\phi(\mathbf{X})L(\mathbf{X})\right),$$

où $L(\mathbf{x}) = \dfrac{f(\mathbf{x})}{g(\mathbf{x})}$ est appelé rapport de vraisemblance ("*likelihood ratio*") ou poids d'importance. L'estimateur d'échantillonnage préférentiel associé est alors défini par :

$$\hat{E}_N = \frac{1}{N}\sum_{i=1}^{N} \phi(\mathbf{X}_i)L(\mathbf{X}_i), \tag{1.4}$$

où $\mathbf{X}_1, \ldots, \mathbf{X}_N$ est un échantillon i.i.d. selon la loi de densité g. Cet estimateur est sans biais et est consistant d'après la loi des grands nombres. La précision de $\hat{E}_N$ dépend fortement du choix de g. On peut déjà noter que pour une densité g ayant une queue de distribution plus légère que celle de f (ou autrement dit si g prend des valeurs très petites devant celles de f loin de leurs valeurs centrales), la variance de l'estimateur peut exploser et tendre vers l'infini. La variance de $\hat{E}_N$ vaut $\mathrm{Var}_g\left(\hat{E}_N\right) = \mathrm{Var}_g\left(\phi(\mathbf{X})L(\mathbf{X})\right)/N$, donc pour la minimiser (à N fixé), il faut chercher g qui minimise $\mathrm{Var}_g\left(\phi(\mathbf{X})L(\mathbf{X})\right)$. La solution de ce problème de minimisation est donnée (dans [Robert and Casella, 2004]) par :

$$g^*(\mathbf{x}) = \frac{|\phi(\mathbf{x})|f(\mathbf{x})}{\int|\phi(\mathbf{u})|f(\mathbf{u})\mathrm{d}\mathbf{u}},$$

et lorsque ϕ est positive, comme on l'a supposé, on a :

$$g^*(\mathbf{x}) = \frac{\phi(\mathbf{x})f(\mathbf{x})}{E}. \tag{1.5}$$

En effet, la variance de $\hat{E}_N$ est nulle lorsque la densité auxiliaire d'échantillonnage vaut g^*. Dans le cas $\phi \geq 0$, que l'on considère dans toute la suite, g^* dépend de la quantité recherchée E et n'est donc pas directement utilisable. Le but est donc de trouver une densité se rapprochant de la densité optimale g^*, et avec laquelle il est facile d'échantillonner. Cet objectif est souvent atteint progressivement au travers d'algorithmes adaptatifs d'échantillonnage préférentiel.

1.2.3 Algorithme générique d'échantillonnage préférentiel adaptatif

Nous avons vu que l'efficacité de l'échantillonnage préférentiel reposait sur le choix de la densité auxiliaire d'échantillonnage. En effet, celle-ci doit être proche de la densité théorique optimale (1.5) minimisant la variance de l'estimateur (1.4), pour que l'estimation par IS soit suffisamment précise. Pour atteindre la densité théorique cible, inconnue en pratique, des algorithmes adaptatifs d'échantillonnage préférentiel ("*Adaptive Importance Sampling*", AIS) ont été développés. Ils permettent de trouver une densité proche de la loi cible en passant par plusieurs densités intermédiaires mises à jour de manière itérative. La procédure générale des algorithmes d'AIS est divisée en trois étapes principales [Bugallo et al., 2017] :

1. génération des échantillons selon une, ou plusieurs, densités auxiliaires connues ("*sampling*")
2. calcul des poids associés à chaque échantillon ("*weighting*")
3. mise à jour des nouvelles densités d'échantillonnage ("*adapting*")

Ces trois étapes sont ensuite répétées jusqu'à ce qu'un critère d'arrêt soit vérifié ou qu'un nombre maximal d'itérations soit atteint. En partant d'une densité auxiliaire choisie arbitrairement, la procédure retourne un ensemble d'échantillons pondérés, exploitables pour estimer une espérance.

De nombreux algorithmes, basés sur ces trois étapes, ont été proposés dans la littérature, chacun adaptant de manière différente une ou plusieurs des trois phases. Nous présentons quelques-unes des méthodes d'AIS les plus répandues dans les sections 1.3.3 et 1.4.2. Dans la partie suivante, nous détaillons la méthode paramétrique générale, dans laquelle la densité optimale est approchée par une densité appartenant à une famille paramétrique.

1.2.4 Approximation paramétrique de la densité optimale d'échantillonnage préférentiel

Pour réduire la variance de l'estimateur d'échantillonnage préférentiel, il est important de bien choisir la densité auxiliaire d'IS. Nous avons vu dans la section 1.2.2 que la densité optimale d'échantillonnage préférentiel g^* (1.5) n'était pas connue en pratique. Nous expliquons dans les paragraphes suivants comment trouver une approximation de g^* par une densité paramétrique.

1.2.4.1 Cas général

Une méthode classique pour approcher la densité optimale g^* est de considérer une famille paramétrique de densités $\mathcal{G} = \{g_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \Theta\}$ (avec $\Theta \subset \mathbb{R}^m$), et de chercher une densité proche de g^* à l'intérieur de cette famille. Un choix naturel de paramètre est le paramètre $\boldsymbol{\theta}^*$ minimisant la variance de l'estimateur $\hat{E}_N$ (1.4) avec $g = g_{\boldsymbol{\theta}}$ comme densité auxiliaire :

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}\in\Theta}{\arg\min}\ \mathrm{Var}_{g_{\boldsymbol{\theta}}}\left(\hat{E}_N\right) = \underset{\boldsymbol{\theta}\in\Theta}{\arg\min}\ \mathrm{Var}_{g_{\boldsymbol{\theta}}}\left(\phi(\mathbf{X})L_{\boldsymbol{\theta}}(\mathbf{X})\right) \tag{1.6}$$

où $L_{\boldsymbol{\theta}} = f/g_{\boldsymbol{\theta}}$. Mais généralement ce problème n'admet pas de solutions analytiques et doit être résolu par des méthodes numériques [Rubinstein and Kroese, 2017]. C'est pourquoi, il est courant de minimiser une divergence entre g^* et la famille $\mathcal{G}$, comme la divergence de Kullback-Leibler très largement utilisée dans la littérature.

La divergence de Kullback-Leibler (KL) [Kullback and Leibler, 1951] entre deux densités f et g, telles que f soit absolument continue par rapport à g, est définie par :

$$D(f,g) = \mathbb{E}_f\left(\ln\left(\frac{f(\mathbf{X})}{g(\mathbf{X})}\right)\right) = \int f(\mathbf{x}) \ln\left(\frac{f(\mathbf{x})}{g(\mathbf{x})}\right) \mathrm{d}\mathbf{x}. \tag{1.7}$$

Cette quantité est toujours positive et est nulle si et seulement si $f = g$ presque partout. La divergence KL est parfois appelée abusivement "distance" de Kullback-Leibler car elle sert à mesurer l'écart entre les deux densités, même si ce n'est pas une distance puisqu'elle n'est pas symétrique. Par ailleurs, [Au and Beck, 2003] montrent que la divergence KL est liée à la variance de l'estimateur $\hat{E}_N$ par la relation :

$$\frac{N\mathrm{Var}_g(\hat{E}_N)}{E^2} \geq \exp(D(g^*, g)) - 1,$$

pour une densité auxiliaire g. Cela signifie qu'une grande divergence KL entraine une grande variance de l'estimateur. De plus, [Chatterjee and Diaconis, 2018] parviennent à borner l'erreur (en norme L_1), $\mathbb{E}_g(|\hat{E}_N - E|)$, en fonction de $\exp(D(g^*, g))$, et suggèrent que la taille d'échantillon N, doit être environ égale à $N \approx \exp(D(g^*, g))$ pour avoir une estimation précise. Ces résultats justifient le choix de la divergence de Kullback-Leibler pour mesurer la distance entre deux densités, et trouver une densité auxiliaire approchant la densité cible g^*.

Ainsi, le problème de minimisation de la variance (1.6) peut être remplacé par le problème de minimisation de la divergence KL pour trouver le paramètre optimal :

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}\in\Theta}{\arg\min}\ D(g^*, g_{\boldsymbol{\theta}}). \tag{1.8}$$

Comme $D(g^*, g_{\boldsymbol{\theta}}) = \mathbb{E}_{g^*}(\ln(g^*(\mathbf{X}))) - \mathbb{E}_{g^*}(\ln(g_{\boldsymbol{\theta}}(\mathbf{X})))$, en isolant le terme dépendant de $\boldsymbol{\theta}$, le problème (1.8) revient à maximiser l'entropie croisée ("*Cross-Entropy*", CE) : $\mathbb{E}_{g^*}(\ln(g_{\boldsymbol{\theta}}(\mathbf{X})))$. Ce problème d'optimisation est le problème de maximisation de l'entropie croisée et s'écrit :

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}\in\Theta}{\arg\max}\ \mathbb{E}_f\left(\ln\left(g_{\boldsymbol{\theta}}(\mathbf{X})\right)\phi(\mathbf{X})\right), \tag{1.9}$$

où on a utilisé l'expression de $g^* = \phi f/E$. L'avantage de cette approche, comparée à la minimisation de la variance (1.6), est qu'elle admet des solutions analytiques lorsque $g_{\boldsymbol{\theta}}$ appartient à la famille exponentielle. Dans la suite, nous allons expliciter le cas de la famille gaussienne, qui fait partie de la famille exponentielle, paramétrée par $\boldsymbol{\theta} = (\mathbf{m}, \Sigma)$, avec $\mathbf{m} \in \mathbb{R}^n$ le vecteur moyenne et $\Sigma \in \mathcal{S}_n^+$ la matrice de covariance appartenant à l'ensemble des matrices symétriques définies positives noté $\mathcal{S}_n^+$.

1.2.4.2 Cas Gaussien unimodal

Rappelons tout d'abord que la densité initiale f est supposée être la densité gaussienne standard (1.2). Dans ce cas, un choix courant de famille paramétrique de densités auxiliaires est la famille gaussienne $\mathcal{G} = \{g_{\mathbf{m},\Sigma}, \mathbf{m} \in \mathbb{R}^n, \Sigma \in \mathcal{S}_n^+\}$, qui contient f, de sorte que f et $g_{\mathbf{m},\Sigma}$ aient la même queue de distribution.

Notons que ce choix n'est pas adapté lorsque la densité g^* visée comporte plusieurs modes. En effet, si g^* est multimodale, une densité gaussienne pourrait n'identifier qu'un seul mode et "oublier" les autres, ce qui entrainerait une estimation biaisée de l'espérance. Pour cette raison, dans cette thèse nous nous concentrons uniquement sur des problèmes unimodaux. Concernant les

cas multimodaux, des mélanges de densités (gaussiennes ou autres) peuvent être utilisés, comme évoqué dans la remarque 1.3.1.

Rappelons l'expression de la densité gaussienne paramétrée par $\mathbf{m}$ et Σ, pour tout $\mathbf{x} \in \mathbb{R}^n$:

$$g_{\mathbf{m},\Sigma}(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}\det(\Sigma)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{m})^\top \Sigma^{-1}(\mathbf{x}-\mathbf{m})\right).$$

Avec cette famille, le problème de maximisation de l'entropie croisée (1.9) admet comme solution (voir [Rubinstein and Kroese, 2011] et [Kroese et al., 2013]) :

$$\mathbf{m}^* = \mathbb{E}_{g^*}(\mathbf{X}) \quad \text{et} \quad \Sigma^* = \mathbb{E}_{g^*}((\mathbf{X}-\mathbf{m}^*)(\mathbf{X}-\mathbf{m}^*)^\top). \tag{1.10}$$

En pratique, ces deux paramètres ne peuvent pas être calculés explicitement, puisqu'ils dépendent de g^*. En revanche, si on dispose d'un échantillon i.i.d. $\mathbf{X}_1, \ldots, \mathbf{X}_N$ généré selon g^*, on peut les estimer par :

$$\hat{\mathbf{m}}^* = \frac{1}{N}\sum_{i=1}^{N} \mathbf{X}_i \quad \text{et} \quad \hat{\Sigma}^* = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{X}_i-\mathbf{m}^*)(\mathbf{X}_i-\mathbf{m}^*)^\top. \tag{1.11}$$

En effet, il est possible d'échantillonner selon g^*, avec un algorithme de Monte-Carlo par Chaines de Markov ("*Markov Chain Monte Carlo*", MCMC) par exemple ou la méthode d'acceptation-rejet, pour ensuite estimer les paramètres (1.11). Nous évoquons ces méthodes d'échantillonnage dans la section 1.5. Cependant, elles peuvent nécessiter un budget de simulation important. Une autre possibilité, permettant de réduire le budget, est d'estimer ces paramètres par échantillonnage préférentiel, à l'aide d'une densité auxiliaire, obtenue de manière adaptative comme évoqué dans le paragraphe 1.2.3.

Enfin, une fois les paramètres estimés, l'intégrale E est approchée par $\hat{E}_N$ (1.4) avec $g = g_{\hat{\mathbf{m}}^*,\hat{\Sigma}^*}$ comme densité auxiliaire.

Remarque 1.2.2. L'échantillonnage préférentiel non-paramétrique ("*Non-parametric Importance Sampling*", NIS) introduit par [Zhang, 1996] peut également être employé pour approcher g^*. Le principe est de construire un estimateur à l'aide d'une estimation par noyau pondérée, à partir d'un échantillon i.i.d. $\mathbf{X}_1, \ldots, \mathbf{X}_N$ selon une loi auxiliaire g_0. L'estimateur de g^* s'écrit ainsi :

$$\hat{g}_N(\mathbf{x}) = \frac{1}{\det(H)^{1/2}} \sum_{i=1}^{N} \bar{w}_i K\left(H^{-1/2}(\mathbf{x}-\mathbf{X}_i)\right)$$

où $K : \mathbb{R}^n \to \mathbb{R}$ est un noyau que nous considèrerons gaussien (égal à la densité de la loi normale standard $\mathcal{N}(\mathbf{0}, I_n)$), H est une matrice symétrique définie positive, appelée matrice des fenêtres, et $\bar{w}_i$ sont les poids normalisés associés aux $\mathbf{X}_i$. Ceux-ci sont égaux, pour $i = 1, \ldots N$, à :

$$\bar{w}_i = \frac{w_i}{\sum_{k=1}^{N} w_k} \quad \text{avec} \quad w_i = \frac{\phi(\mathbf{X}_i) f(\mathbf{X}_i)}{g_0(\mathbf{X}_i)}.$$

La matrice des fenêtres H est un paramètre de lissage qu'il faut calibrer pour obtenir une estimation suffisamment lisse tout en restant proche de la densité cible. L'intégrale E peut alors être estimée par la formule (1.4), avec $\hat{g}_N$ comme densité auxiliaire d'IS.

Cette méthode est efficace et suffisamment flexible pour approcher précisément la densité cible lorsque celle-ci est particulièrement complexe, notamment si elle est constituée de plusieurs modes. Cependant, elle est inefficace dès que la dimension devient supérieure à 10, car un grand nombre

d'échantillon est nécessaire pour que l'estimation soit suffisamment précise. Le cout de calcul peut alors devenir très élevé et c'est pourquoi les techniques paramétriques sont souvent préférées.

Notons aussi qu'il existe des méthodes semi-paramétriques, comme l'algorithme MCIS ("Markov Chain Importance Sampling") développé dans [Botev et al., 2013] où la densité auxiliaire est supposée être le produit de n densités unidimensionnelles, lesquelles sont apprises grâce à une étape de MCMC pour approcher les marginales de g^*. Néanmoins, dans le cadre de cette thèse, nous ne nous intéresserons qu'aux approches paramétriques.

Dans les deux sections qui suivent, nous présentons l'échantillonnage préférentiel dans le contexte de la fiabilité et de l'inférence bayésienne. L'IS est en effet un sujet de recherche important et très actif dans ces deux disciplines.

1.3 Estimation en fiabilité

Dans le domaine de la fiabilité des systèmes complexes, on s'intéresse à l'estimation d'une probabilité de défaillance du système considéré, souvent définie comme la probabilité d'un dépassement de seuil d'une fonction de performance (voir par exemple [Ditlevsen and Madsen, 1996]). Dans ce cas, la fonction ϕ est égale à la fonction indicatrice $\phi(\mathbf{x}) = \mathbb{I}_{\{\varphi(\mathbf{x})\geq 0\}}$, qui vaut 1 lorsque $\varphi(\mathbf{x}) \geq 0$, et 0 sinon, et où φ est une fonction de $\mathbb{R}^n$ dans $\mathbb{R}$, qui fait office de boite noire. La quantité recherchée s'écrit alors

$$E = \mathbb{E}_f(\mathbb{I}_{\{\varphi(\mathbf{X})\geq 0\}}) = \mathbb{P}_f(\varphi(\mathbf{X}) \geq 0),$$

qui est souvent une **probabilité d'événement rare** (c'est-à-dire une faible probabilité de l'ordre de 10^{-3} ou moins). La fonction φ est alors appelée **fonction d'état limite** ou **fonction de performance**, l'**état limite** étant défini par l'ensemble $\{\mathbf{x} \in \mathbb{R}^n \mid \varphi(\mathbf{x}) = 0\}$, et l'ensemble $\{\mathbf{x} \in \mathbb{R}^n \mid \varphi(\mathbf{x}) \geq 0\}$ est le **domaine de défaillance** du système.

Dans cette section, nous présentons quelques méthodes utilisées en fiabilité pour estimer ce type de probabilités, notamment l'algorithme d'entropie croisée (1.3.3) sur lequel nous reviendrons particulièrement tout au long de cette thèse.

1.3.1 L'inefficacité de la méthode de Monte-Carlo

Nous avons déjà évoqué dans la partie 1.2.1, que l'estimateur de Monte-Carlo n'était pas toujours performant, car il pouvait avoir une grande variance. C'est le cas en fiabilité lorsque E est une probabilité d'événement rare. Dans ce cas, la variance de l'estimateur de Monte-Carlo (1.3) vaut :

$$\begin{aligned}
\mathrm{Var}_f\left(\hat{E}_N^{\mathrm{MC}}\right) &= \frac{1}{N}\mathrm{Var}_f\left(\mathbb{I}_{\{\varphi(\mathbf{X})\geq 0\}}\right) \\
&= \frac{E(1-E)}{N} \\
&\approx \frac{E}{N}
\end{aligned}$$

puisque la variable $\mathbb{I}_{\{\varphi(\mathbf{X})\geq 0\}}$ suit une loi de Bernoulli de paramètre E, qui prend de très faibles valeurs ($E \ll 1$). Ainsi, le coefficient de variation (ou erreur relative) de $\hat{E}_N^{\mathrm{MC}}$ est égal à

$$\frac{\sqrt{\mathrm{Var}_f(\hat{E}_N^{\mathrm{MC}})}}{\mathbb{E}_f(\hat{E}_N^{\mathrm{MC}})} = \frac{\sqrt{1-E}}{\sqrt{NE}} \approx \frac{1}{\sqrt{NE}},$$

donc pour avoir une erreur d'environ 10%, on doit avoir $N \approx 100/E$. Par exemple, si $E = 10^{-4}$, il faut alors générer et évaluer $N = 10^6$ échantillons, ce qui peut entrainer un temps de calcul excessivement long.

Autrement dit, la méthode de Monte-Carlo est inefficace dans ce contexte car une faible proportion des échantillons selon f tombe dans le domaine de défaillance, ce qui induit une grande erreur d'estimation. Différentes méthodes ont été développées pour avoir une plus grande précision, en générant un grand nombre d'échantillons dans la région de défaillance, tout en gardant un budget de simulation raisonnable. L'échantillonnage préférentiel en fait partie et nous décrivons deux algorithmes d'AIS à la fin de cette section. D'autres techniques sont présentées succinctement dans les paragraphes suivants.

1.3.2 Méthodes alternatives à la méthode de Monte-Carlo

1.3.2.1 FORM/SORM

Les méthodes FORM ("*First Order Reliability Methods*") et SORM ("*Second Order Reliability Methods*") [Madsen et al., 2006] servent à estimer une probabilité de défaillance en effectuant une approximation linéaire (pour FORM) ou quadratique (SORM) de la fonction de performance φ en un point appelé point de conception. Ces deux techniques supposent d'abord que le vecteur aléatoire $\mathbf{X}$ est gaussien centré réduit (autrement dit f est la densité de la loi $\mathcal{N}(\mathbf{0}, I_n)$), quitte à utiliser les transformations isoprobabilistes évoquées au début de ce chapitre (1.1.2). Ensuite, le point de conception est défini comme le point de défaillance le plus probable (ou MPFP, "*Most Probable Failure Point*"), c'est-à-dire celui qui minimise la distance entre l'origine et l'hypersurface $\{\varphi(\mathbf{x}) = 0\}$:

$$\mathbf{x}^* = \underset{\mathbf{x}\in\mathbb{R}^n}{\arg\min}\|\mathbf{x}\|, \quad \text{tel que } \varphi(\mathbf{x}) = 0.$$

Ce problème peut être résolu à l'aide d'algorithmes d'optimisation, décrits par exemple dans [Liu and Der Kiureghian, 1991].

Dans le cas de FORM, une approximation linéaire de la fonction d'état limite φ est calculée au point $\mathbf{x}^*$, en utilisant le développement de Taylor à l'ordre 1 (et donc le gradient de φ). Cette approximation permet finalement d'estimer la probabilité par : $\hat{E}^{\mathrm{FORM}} = F_{\mathcal{N}}(-\|\mathbf{x}^*\|)$, où $F_{\mathcal{N}}$ est la fonction de répartition de la loi $\mathcal{N}(0,1)$.

La méthode SORM peut permettre d'améliorer la précision de FORM (notamment lorsque φ est fortement non linéaire) en approchant la fonction à l'ordre 2, ce qui nécessite alors le calcul de la matrice Hessienne de φ. Une estimation de la probabilité de défaillance est alors donnée par : $\hat{E}^{\mathrm{SORM}} = F_{\mathcal{N}}(-\|\mathbf{x}^*\|)\prod_{j=1}^{n-1}(1-\kappa_j\|\mathbf{x}^*\|)^{-1/2}$, où les κ_j sont des coefficients réels de courbure de la fonction φ. Il existe d'autres formules d'estimation de la probabilité par SORM, mais nous ne les évoquerons pas ici. Pour plus de détails sur les deux méthodes, nous renvoyons à [Bourinet, 2018] et [Der Kiureghian et al., 2005].

Les techniques FORM et SORM ont l'avantage d'être peu couteuses et simples à mettre en œuvre. Elles nécessitent néanmoins des hypothèses de régularité sur φ pouvant être restrictives et sont peu efficaces lorsque celle-ci est fortement non linéaire ou lorsque la dimension est élevée. De plus, aucun contrôle de l'erreur d'estimation n'est disponible.

1.3.2.2 Échantillonnage préférentiel basé sur le point de conception

Le point de conception $\mathbf{x}^*$ déterminé dans les méthodes FORM/SORM peut aussi être utilisé pour trouver une densité auxiliaire pour l'échantillonnage préférentiel. Sous l'hypothèse où f est la densité $\mathcal{N}(\mathbf{0}, I_n)$, [Melchers, 1989] suggère pour cela d'estimer la probabilité avec un échantillon généré selon la loi $\mathcal{N}(\mathbf{x}^*, I_n)$. L'estimateur est alors simplement l'estimateur d'IS $\hat{E}_N$ (1.4) calculé avec un échantillon i.i.d. $\mathbf{X}_1, \dots \mathbf{X}_N$ selon la densité auxiliaire $g = g_{\mathbf{x}^*, I_n}$. Cet échantillon a l'avantage d'être autour du domaine de défaillance (par définition de $\mathbf{x}^*$) et on peut s'attendre à ce qu'un assez grand nombre de points tombent effectivement dans ce domaine, ce qui permet d'avoir une précision raisonnable. Une limite de cette méthode survient néanmoins lorsque le point de conception n'est pas unique, dans les cas multimodaux notamment. La recherche de ce point, par la résolution d'un problème d'optimisation sous contrainte non linéaire, peut aussi être une autre difficulté lorsque la dimension augmente.

1.3.2.3 "*Subset Simulation*"

Une autre méthode classique et efficace dans le domaine de la fiabilité est celle de "*Subset Simulation*" [Au and Beck, 2001], également appelé "*Adaptive Splitting*" [Cérou and Guyader, 2007]. Le principe est de réécrire la probabilité de défaillance comme un produit de m probabilités en faisant intervenir une suite d'événements emboités contenant le domaine de défaillance. En notant $\{\varphi(\mathbf{X}) \geq 0\} = \mathcal{F}_m \subset \cdots \subset \mathcal{F}_1$, la probabilité est égale à

$$E = \mathbb{P}_f(\mathcal{F}_m) = \prod_{j=2}^{m} \mathbb{P}_f(\mathcal{F}_j | \mathcal{F}_{j-1}) \mathbb{P}_f(\mathcal{F}_1).$$

Le problème de départ se ramène donc à l'estimation de m probabilités conditionnelles plus grandes que E et ainsi plus simples à estimer. La principale difficulté réside dans la simulation des échantillons selon les lois conditionnelles $f(\cdot|\mathcal{F}_j)$. Cette étape est généralement réalisée grâce à un algorithme MCMC. Par ailleurs, les événements intermédiaires sont construits de manière adaptative à l'aide de seuils $s_1 < \ldots < s_m = 0$ tels que $\mathcal{F}_j = \{\varphi(\mathbf{X}) \geq s_j\}$ et de sorte que $\mathbb{P}_f(\mathcal{F}_1) = \mathbb{P}_f(\mathcal{F}_j|\mathcal{F}_{j-1}) = p_0$, pour tout $j = 2 \dots m-1$, où p_0 est une valeur préalablement choisie (usuellement $p_0 = 0.1$).

Ainsi, à la première étape, on tire un échantillon selon f, avec lequel on détermine le seuil s_1 tel que $\mathbb{P}_f(\mathcal{F}_1) = p_0$. Dans les étapes $j = 2, \dots, m-1$, on génère un échantillon selon $f(\cdot|\mathcal{F}_{j-1})$ à l'aide d'un algorithme MCMC, et on détermine le seuil s_j de sorte que $\mathbb{P}_f(\mathcal{F}_j|\mathcal{F}_{j-1}) = p_0$. La dernière probabilité $\mathbb{P}_f(\mathcal{F}_m|\mathcal{F}_{m-1})$ est alors estimée par $N^{-1}\sum_{i=1}^{N} \mathbb{I}_{\{\varphi(\mathbf{X}_i)\geq 0\}}$, où les $\mathbf{X}_i$ sont générés selon la loi $f(\cdot|\mathcal{F}_{m-1}) = f\mathbb{I}_{\{\varphi(\cdot)\geq s_{m-1}\}}/\mathbb{P}_f(\mathcal{F}_{m-1})$.

L'algorithme de "*Subset Simulation*" permet alors de générer de manière adaptative un échantillon selon la densité conditionnelle $f(\cdot|\mathcal{F}_{m-1})$, en partant d'un échantillon selon f, et en s'appuyant sur un algorithme MCMC (Metropolis-Hastings modifié dans [Au and Beck, 2001]). Finalement, l'estimation de E est donnée par :

$$\hat{E}_N^{SS} = p_0^{m-1} \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}_{\{\varphi(\mathbf{X}_i)\geq 0\}},$$

où $\mathbf{X}_1, \ldots, \mathbf{X}_N$ est un échantillon généré selon $f(\cdot|\mathcal{F}_{m-1})$. Cette technique est particulièrement efficace sur des problèmes non linéaires ou en grande dimension mais le temps et le cout de calcul peuvent parfois être élevés, du fait de l'utilisation d'un algorithme MCMC.

Nous avons résumé les méthodes principales d'estimation de probabilités de défaillance. Le lecteur intéressé peut se référer à [Morio et al., 2014] pour une présentation de techniques alternatives. Dans la suite, nous nous concentrons sur les algorithmes d'AIS utilisés en fiabilité.

1.3.3 Algorithmes adaptatifs d'échantillonnage préférentiel dans le contexte des événements rares

Dans cette section nous décrivons deux algorithmes d'échantillonnage préférentiel adaptatif pour estimer une probabilité d'événement rare. Dans ce cas, la densité optimale d'échantillonnage préférentiel g^* (1.5), décrite dans la section 1.2.2, est :

$$g^*(\mathbf{x}) = \frac{f(\mathbf{x})\mathbb{I}_{\{\varphi(\mathbf{x})\geq 0\}}}{E}.$$

Cette densité est égale à la densité f conditionnée à l'événement rare $\{\varphi(\mathbf{x}) \geq 0\}$. Pour approcher g^*, il faut donc pouvoir générer suffisamment d'échantillons dans ce domaine de défaillance. Or sous la loi initiale f, on sait qu'il y a une très faible proportion (égale à E) des échantillons qui seront dans la région de défaillance. C'est pourquoi, certains algorithmes proposent des méthodes adaptatives permettant de générer des échantillons s'approchant de la défaillance progressivement.

C'est le cas de l'algorithme d'entropie croisée ("*Cross-Entropy*", CE) [Rubinstein and Kroese, 2011], et d'une version améliorée ("*Improved CE*", iCE) suggérée par [Papaioannou et al., 2019a], que nous présentons ci-dessous.

1.3.3.1 L'algorithme d'entropie croisée

L'algorithme d'entropie croisée est une méthode paramétrique d'estimation de probabilités d'événements rares par échantillonnage préférentiel. Il permet d'estimer de manière adaptative les paramètres optimaux d'IS, maximisant l'entropie croisée (1.9). Nous décrivons le principe de la méthode pour des densités gaussiennes $g_{\mathbf{m},\Sigma}$.

On rappelle que les paramètres optimaux sont $\mathbf{m}^*$ et Σ^* (définis en (1.10)) et peuvent s'écrire comme suit :

$$\mathbf{m}^* = \mathbb{E}_f(\mathbf{X}|\varphi(\mathbf{X}) \geq 0) \quad \text{et} \quad \Sigma^* = \mathbb{E}_f((\mathbf{X}-\mathbf{m}^*)(\mathbf{X}-\mathbf{m}^*)^\top|\varphi(\mathbf{X}) \geq 0).$$

Si on dispose d'un échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_N$ selon une loi auxiliaire g donnée, les paramètres peuvent être estimés par

$$\hat{\mathbf{m}}^* = \sum_{i=1}^{N} \mathbf{X}_i \bar{w}_i \quad \text{et} \quad \hat{\Sigma}^* = \sum_{i=1}^{N} (\mathbf{X}_i - \hat{\mathbf{m}}^*)(\mathbf{X}_i - \hat{\mathbf{m}}^*)^\top \bar{w}_i \tag{1.12}$$

où $w_i = \mathbb{I}_{\{\varphi(\mathbf{X}_i)\geq 0\}} f(\mathbf{X}_i)/g(\mathbf{X}_i)$ correspond au poids associé à l'échantillon $\mathbf{X}_i$, et $\bar{w}_i = w_i/\sum_k w_k$. Comme on l'a déjà dit, la difficulté est de générer suffisamment d'échantillons tels que $\varphi(\mathbf{X}_i) \geq 0$, pour que les indicatrices ne soient pas toutes nulles. L'algorithme CE y parvient en créant une suite de seuils intermédiaires $\gamma_0 < \gamma_1 < \cdots \leq 0$, de sorte que les événements $\{\varphi(\mathbf{x}) \geq \gamma_t\}$ soient

moins rares mais tendent vers $\{\varphi(\mathbf{x}) \geq 0\}$ lorsque le nombre d'itérations t grandit. Une suite de paramètres intermédiaires, $(\mathbf{m}_t, \Sigma_t)$, est également construite (avec les formules (1.12)), chacun dépendant du seuil γ_t, et se rapprochant de $(\mathbf{m}^*, \Sigma^*)$. Les seuils γ_t sont construits de sorte qu'une proportion $\rho \in]0,1[$, préalablement choisie, des échantillons tirés selon $g_{\mathbf{m}_t,\Sigma_t}$ soit dans le domaine $\{\varphi(\mathbf{x}) \geq \gamma_t\}$.

Ainsi, étant donné un couple de paramètres $(\mathbf{m}_t, \Sigma_t)$ à l'instant t, l'algorithme CE repose sur la répétition des deux étapes suivantes :

- Générer un échantillon i.i.d. $\mathbf{X}_1, \ldots, \mathbf{X}_N$ selon $g_{\mathbf{m}_t,\Sigma_t}$
- Définir γ_t comme le ρ-quantile des $\varphi(\mathbf{X}_1), \ldots, \varphi(\mathbf{X}_N)$, et estimer les nouveaux paramètres :

$$\mathbf{m}_{t+1} = \sum_{i=1}^{N} \mathbf{X}_i \bar{w}_i \quad \text{et} \quad \Sigma_{t+1} = \sum_{i=1}^{N} (\mathbf{X}_i - \mathbf{m}_{t+1})(\mathbf{X}_i - \mathbf{m}_{t+1})^\top \bar{w}_i$$

  avec $w_i = \mathbb{I}_{\{\varphi(\mathbf{X}_i) \geq \gamma_t\}} f(\mathbf{X}_i)/g_{\mathbf{m}_t,\Sigma_t}(\mathbf{X}_i)$, et $\bar{w}_i = w_i / \sum_k w_k$.

L'algorithme est initialisé avec $\mathbf{m}_0$, Σ_0, choisis arbitrairement et s'arrête dès qu'un seuil γ_τ est supérieur à 0. Avec les derniers paramètres estimés, on peut enfin calculer l'estimation $\hat{E}_N$ (1.4) avec $g = g_{\mathbf{m}_\tau,\Sigma_\tau}$ comme densité auxiliaire. Le détail de la méthode CE est donné dans l'algorithme 1, où les paramètres initiaux sont fixés à $\mathbf{m}_0 = \mathbf{0}$, $\Sigma_0 = I_n$, pour coïncider avec la loi initiale f. Cet algorithme est souvent appelé algorithme CE "multi-niveau" ("*multilevel*" CE) du fait de la construction des seuils intermédiaires permettant d'approcher progressivement les paramètres optimaux. Le choix du paramètre ρ joue un rôle important dans la vitesse de convergence et la précision de l'algorithme. Plus il est petit, et plus le critère d'arrêt ($\gamma_t \geq 0$) est atteint rapidement. Mais un ρ trop faible peut entrainer une estimation imprécise puisque seulement ρN échantillons sont utilisés dans l'estimation des paramètres à chaque itération (il y a ρN indicatrices non nulles). Un choix classique et efficace, recommandé par les auteurs [Rubinstein and Kroese, 2011], est de prendre ρ entre 0.01 et 0.1.

Algorithme 1 : Entropie Croisée (*Cross Entropy*, CE, [Rubinstein and Kroese, 2011]).

Données : Paramètre $\rho \in]0,1[$, taille de l'échantillon N par itération
Résultat : Estimation $\hat{E}_N$ de la probabilité de défaillance $E = \mathbb{P}_f(\varphi(\mathbf{X}) \geq 0)$

1 Initialisation : poser $t = 0$, $\mathbf{m}_t = \mathbf{0}$ et $\Sigma_t = I_n$;
2 Générer $\mathbf{X}_1, \ldots, \mathbf{X}_N$ indépendamment selon $g_{\mathbf{m}_t,\Sigma_t}$;
3 Évaluer $q_i = \varphi(\mathbf{X}_i)$ pour tout $i = 1, \ldots N$;
4 Ranger les échantillons dans l'ordre croissant : $q_{(1)} \leq \cdots \leq q_{(N)}$, et poser $\gamma_t = q_{(\lfloor (1-\rho)N \rfloor)}$;
5 Calculer les poids $\bar{w}_i = w_i / \sum_{j=1}^{n} w_j$ avec $w_i = \mathbb{I}_{\{q_i \geq \gamma_t\}} L_t(\mathbf{X}_i)$ et $L_t = f/g_{\mathbf{m}_t,\Sigma_t}$;
6 **tant que** $\gamma_t < 0$ **faire**
7 | Estimer $\mathbf{m}_{t+1} = \sum_{i=1}^{N} \bar{w}_i \mathbf{X}_i$ et $\Sigma_{t+1} = \sum_{i=1}^{N} \bar{w}_i (\mathbf{X}_i - \mathbf{m}_{t+1})(\mathbf{X}_i - \mathbf{m}_{t+1})^\top$;
8 | Incrémenter t : $t \longleftarrow t + 1$;
9 | Répéter les étapes 2, 3, 4 et 5 ;
10 **fin**
11 Estimer $\hat{E}_N = \dfrac{1}{N} \sum_{i=1}^{N} \mathbb{I}_{\{q_i \geq 0\}} L_t(\mathbf{X}_i)$.

Pour mieux utiliser les N échantillons à chaque itération, [Papaioannou et al., 2019a] propose une amélioration de la CE que nous décrivons dans la partie suivante.

Remarque 1.3.1. Si l'algorithme CE (1) a montré son efficacité dans de nombreux exemples (voir [Bourinet, 2018]), il connait des difficultés lorsque la région de défaillance (et donc la densité g^*) est multimodale, car une seule densité est mise à jour au cours de l'algorithme. Pour remédier à ce problème, [Kurtz and Song, 2013] et [Geyer et al., 2019] ont adapté la méthode avec des mélanges de densités (gaussiennes), capables d'identifier plusieurs modes. Plus généralement, il est possible d'appliquer l'échantillonnage préférentiel paramétrique avec des mélanges de plusieurs densités pour traiter des cas multimodaux. Dans cette thèse nous nous concentrons sur des problèmes unimodaux et ne considérons donc pas ces approches. L'extension des méthodes proposées dans ce manuscrit à des problèmes multimodaux constitue une piste intéressante qui est discutée en conclusion.

Remarque 1.3.2. Il existe aussi des méthodes d'AIS non-paramétriques pour estimer des probabilités d'événements rares, qui s'appuient sur l'échantillonnage préférentiel non-paramétrique de [Zhang, 1996], évoqué dans la remarque 1.2.2. Le lecteur intéressé peut se référer aux algorithmes NAIS ("*Non-parametric Adaptive Importance Sampling*") développés dans [Kim et al., 2000] et [Morio, 2012]. Ces méthodes reprennent l'idée d'adapter des seuils successivement, comme dans l'algorithme CE, tout en approchant la densité optimale g^* grâce à des estimations par noyaux gaussiens.

1.3.3.2 Une amélioration de l'algorithme d'entropie croisée

La méthode iCE développée par [Papaioannou et al., 2019a] reprend les mêmes étapes que la CE mais effectue les estimations des paramètres en remplaçant la fonction indicatrice $\mathbb{I}_{\{\varphi(\cdot)\geq 0\}}$ par une approximation lisse, $\psi(\cdot,\sigma)$ vérifiant $\lim_{\sigma\to 0}\psi(\mathbf{x},\sigma) = \mathbb{I}_{\{\varphi(\mathbf{x})\geq 0\}}$. Cette modification permet de prendre en compte les N échantillons à chaque étape de l'algorithme, contre ρN dans la CE (avec $\rho \in]0,1[$ potentiellement petit), et donc d'augmenter la précision des estimations intermédiaires. L'approximation choisie par les auteurs est la suivante : $\psi(\cdot,\sigma) = F_{\mathcal{N}}(\varphi(\cdot)/\sigma)$, où $F_{\mathcal{N}}$ est la fonction de répartition de la loi normale centrée réduite, $\mathcal{N}(0,1)$, et $\sigma > 0$ est un paramètre de lissage. C'est un choix classique et efficace que nous garderons dans tout le manuscrit, mais le lecteur intéressé pourra trouver d'autres approximations suggérées dans la littérature, par exemple dans [Katafygiotis and Zuev, 2007] et [Uribe et al., 2021]. Avec cette approximation, l'estimation des paramètres gaussiens est donnée par les mêmes formules (1.12) que pour la CE, mais où les poids deviennent :

$$w_i = \frac{\psi(\mathbf{X}_i,\sigma)f(\mathbf{X}_i)}{g(\mathbf{X}_i)}$$

pour $\mathbf{X}_1,\dots,\mathbf{X}_N$ un échantillon i.i.d. généré selon une loi de densité g, et un paramètre de lissage $\sigma > 0$. Le caractère séquentiel de l'algorithme iCE repose alors sur la définition d'une suite décroissante $\sigma_0 > \sigma_1 > \cdots > 0$ à la place de la suite des γ_t dans la méthode CE. Les paramètres σ_t sont mis à jour successivement de sorte que le coefficient de variation des poids se rapproche d'une valeur $\delta > 0$, choisie au préalable. Autrement dit, étant donné des paramètres σ_t, $\mathbf{m}_t$, Σ_t à l'instant t, une nouvelle valeur σ_{t+1} est définie en résolvant le problème :

$$\sigma_{t+1} = \underset{\sigma\in]0,\sigma_t[}{\arg\min}\left(\hat{\delta}_t(\sigma) - \delta\right)^2$$

où $\hat{\delta}_t(\sigma)$ est le coefficient de variation des $F_{\mathcal{N}}(\varphi(\mathbf{X}_i)/\sigma)L_t(\mathbf{X}_i)$, avec $L_t = f/g_{\mathbf{m}_t,\Sigma_t}$, $\mathbf{X}_i \sim g_{\mathbf{m}_t,\Sigma_t}$ et $\delta > 0$ fixé. La résolution de ce problème d'optimisation en dimension 1 ne nécessite pas d'évaluation supplémentaire de la fonction boite noire φ et n'augmente donc pas le budget de simulation.

Les nouveaux paramètres $\mathbf{m}_{t+1}$ et Σ_{t+1} sont ensuite mis à jour comme indiqué dans l'algorithme 2, décrivant le déroulement de la méthode iCE. Le programme s'arrête lorsque le coefficient de variation empirique des $\mathbb{I}_{\{\varphi(\mathbf{X}_i)\geq 0\}}/F_{\mathcal{N}}(\varphi(\mathbf{X}_i)/\sigma_t)$ devient inférieur à δ, pour un échantillon $\mathbf{X}_i$ donné à un instant t. Ce critère d'arrêt permet de s'assurer que la fonction $F_{\mathcal{N}}(\varphi(\cdot)/\sigma)$ est suffisamment proche de l'indicatrice.

Algorithme 2 : iCE (*improved CE method* [Papaioannou et al., 2019a]).

Données : Paramètre δ, taille de l'échantillon N par itération
Résultat : Estimation $\hat{E}_N$ de la probabilité de défaillance $E = \mathbb{P}_f(\varphi(\mathbf{X}) \geq 0)$

1 Initialisation : poser $t = 0$, $\mathbf{m}_t = 0$, $\Sigma_t = I_n$ et $\sigma_t = \infty$;
2 Générer $\mathbf{X}_1, \ldots, \mathbf{X}_N$ indépendamment selon $g_{\mathbf{m}_t,\Sigma_t}$;
3 Évaluer $q_i = \varphi(\mathbf{X}_i)$ pour tout $i = 1, \ldots N$;
4 Calculer $\widehat{\mathrm{cv}}$ le coefficient de variation empirique des $\mathbb{I}_{\{q_i\geq 0\}}/F_{\mathcal{N}}(q_i/\sigma_t)$;
5 **tant que** $\widehat{\mathrm{cv}} \geq \delta$ **faire**
6 Calculer $\sigma_{t+1} = \arg\min \left(\hat{\delta}_t(\sigma) - \delta\right)^2$ où le minimum est évalué sur $\sigma \in]0, \sigma_t[$ et $\hat{\delta}_t(\sigma)$ est le coefficient de variation des $F_{\mathcal{N}}(q_i/\sigma)L_t(\mathbf{X}_i)$ avec $L_t = f/g_{\mathbf{m}_t,\Sigma_t}$;
7 Calculer les poids $\bar{w}_i = w_i/\sum_j w_j$ avec $w_i = F(q_i/\sigma_{t+1})L_t(X_i)$;
8 Estimer $\mathbf{m}_{t+1} = \sum_{i=1}^N \bar{w}_i \mathbf{X}_i$ and $\Sigma_{t+1} = \sum_{i=1}^N \bar{w}_i(\mathbf{X}_i - \mathbf{m}_{t+1})(\mathbf{X}_i - \mathbf{m}_{t+1})^\top$;
9 Incrémenter t : $t \longleftarrow t + 1$;
10 Répéter les étapes 2, 3 et 4 ;
11 **fin**
12 Estimer $\hat{E}_N = \dfrac{1}{N}\displaystyle\sum_{i=1}^N \mathbb{I}_{\{q_i\geq 0\}} L_t(\mathbf{X}_i)$.

Les performances de l'algorithme iCE (2) dépendent du choix du paramètre δ. Si celui-ci est trop petit, alors un grand nombre d'itérations sera nécessaire avant que le critère d'arrêt ne soit satisfait, ce qui entrainerait une augmentation du budget de simulation. En revanche, si δ est trop élevé, la fonction $\psi(\cdot, \sigma)$ peut donner une approximation imprécise de l'indicatrice et les estimations réalisées seront elles-mêmes imprécises. Les auteurs de [Papaioannou et al., 2019a] suggèrent que $\delta = 1.5$ est un bon compromis pour avoir une estimation finale précise et garder un budget de simulation raisonnable. Avec ce choix, ils montrent sur différents exemples que la variance de l'estimation de la probabilité est nettement réduite avec l'algorithme iCE, par rapport à CE avec un même budget.

1.4 Estimation d'intégrales en inférence bayésienne

L'inférence bayésienne est un autre domaine dans lequel l'échantillonnage préférentiel est très utilisé pour estimer une intégrale et où de nombreux algorithmes d'AIS ont été proposés (voir [Bugallo et al., 2017]). Dans ce contexte, le but est d'estimer l'espérance E où la densité f représente la loi a posteriori liée à des observations $\mathbf{y} \in \mathbb{R}^k$, et pouvant être exprimée comme suit :

$$f(\mathbf{x}) = h(\mathbf{x}|\mathbf{y}) = \frac{l(\mathbf{y}|\mathbf{x})h_0(\mathbf{x})}{c(\mathbf{y})} \propto l(\mathbf{y}|\mathbf{x})h_0(\mathbf{x}) = \tilde{f}(\mathbf{x}),$$

où $h(\mathbf{x}|\mathbf{y})$ est la densité a posteriori sachant les observations $\mathbf{y}$, $l(\mathbf{y}|\mathbf{x})$ la vraisemblance, $h_0(\mathbf{x})$ la densité a priori, et $c(\mathbf{y}) = \int l(\mathbf{y}|\mathbf{x})h_0(\mathbf{x})\mathrm{d}\mathbf{x}$ la constante de normalisation pouvant être difficile à

calculer. Ce type de problèmes ne permet pas en général de se ramener à une densité $\mathcal{N}(\mathbf{0}, I_n)$, du fait de la complexité ou du manque d'information sur la loi de f, et l'objectif premier est d'apprendre à échantillonner selon f (ou une autre densité bien choisie) avant d'estimer E. Dans la suite, on considère donc que $f = \tilde{f}/c$ est une densité connue à une constante de normalisation près, où $c = \int_{\mathbb{R}^n} \tilde{f}(\mathbf{x})\mathrm{d}\mathbf{x} \in \mathbb{R}$ est cette constante, que l'on peut chercher à estimer également.

1.4.1 L'échantillonnage préférentiel auto-normalisé

L'échantillonnage préférentiel peut encore être utilisé pour estimer une espérance dans le cas où on ne connait pas la constante de normalisation c de la densité $f = \tilde{f}/c$. L'intégrale E doit alors être approchée par l'estimateur d'échantillonnage préférentiel auto-normalisé :

$$\tilde{E}_N = \frac{1}{N\hat{c}_N} \sum_{i=1}^{N} \phi(\mathbf{X}_i)\tilde{L}(\mathbf{X}_i), \tag{1.13}$$

où $\mathbf{X}_1, \ldots, \mathbf{X}_N$ est un échantillon i.i.d. tiré selon une loi auxiliaire g, $\tilde{L}(\mathbf{X}_i) = \tilde{f}(\mathbf{X}_i)/g(\mathbf{X}_i)$ correspond au poids (non normalisé), et $\hat{c}_N = (1/N)\sum_{i=1}^{N} \tilde{L}(\mathbf{X}_i)$ est l'estimation de la constante de normalisation c. L'estimateur $\tilde{E}_N$ est consistant et, contrairement à $\hat{E}_N$ (1.4), il n'est pas sans biais mais asymptotiquement sans biais. Le théorème central limite, avec l'aide du lemme de Slutsky, nous donne la convergence en loi suivante :

$$\sqrt{N}(\tilde{E}_N - E) = \frac{\frac{1}{\sqrt{N}} \sum_{i=1}^{N} \tilde{L}(\mathbf{X}_i)(\phi(\mathbf{X}_i) - E)}{\hat{c}_N} \underset{N \to +\infty}{\longrightarrow} \mathcal{N}\left(0, \tilde{\sigma}^2_{\mathrm{IS}}\right),$$

où $\tilde{\sigma}^2_{\mathrm{IS}} = \mathrm{Var}_g\left(\tilde{L}(\mathbf{X})(\phi(\mathbf{X}) - E)\right)/c^2$, en utilisant le fait que $\hat{c}_N$ converge presque sûrement vers c. La densité auxiliaire g minimisant la variance asymptotique $\tilde{\sigma}^2_{\mathrm{IS}}$ est alors la densité proportionnelle à $|\phi - E|f$ [Hesterberg, 1988]. Mais cette quantité est inconnue et inexploitable facilement étant donné qu'elle dépend de l'espérance E recherchée.

La stratégie couramment adoptée dans le contexte bayésien, où l'on peut s'intéresser à l'estimation de plusieurs espérances sous la loi f, est alors de rechercher une densité auxiliaire qui minimise la variance des poids, ou de manière équivalente la variance de l'estimateur $\hat{c}_N$ (voir [Bugallo et al., 2017], [Doucet et al., 2009]). En effet, la variance de $\hat{c}_N$ est :

$$\mathrm{Var}_g(\hat{c}_N) = \frac{1}{N}\mathrm{Var}_g\left(\tilde{L}(\mathbf{X})\right).$$

Celle-ci est évidemment nulle lorsque $g = f$ (dans ce cas $\tilde{L} \equiv c$), ce qui signifie que la densité auxiliaire recherchée doit approcher la densité initiale f. Le problème posé revient alors à un problème d'échantillonnage selon une loi cible, dont la densité est connue à un facteur de normalisation près. Ce cas est en réalité similaire à celui de la partie 1.2.2, où l'on cherche à échantillonner selon la densité optimale $g^* \propto \phi f$, avec f connue, pour estimer l'espérance E. Lorsqu'on souhaite échantillonner selon $f \propto \tilde{f}$, on cherche la densité d'IS optimale pour estimer la constante de normalisation $c = \int \tilde{f}$. Le problème d'échantillonnage selon f est ainsi vu comme un problème d'estimation de l'intégrale c.

Notons enfin que lorsqu'on estime $\tilde{E}_N$ avec un échantillon pondéré $(\mathbf{X}_1, \bar{w}_1), \ldots, (\mathbf{X}_N, \bar{w}_N)$, où $\bar{w}_i = \tilde{L}(\mathbf{X}_i)/(N\hat{c}_N)$ sont les poids normalisés, la densité f est approchée par :

$$\hat{f}_N(\mathbf{x}) = \sum_{i=1}^{N} \bar{w}_i \delta_{\mathbf{X}_i}(\mathbf{x}), \tag{1.14}$$

où $\delta_{\mathbf{u}}$ est la distribution de Dirac centrée en $\mathbf{u} \in \mathbb{R}^n$. L'estimateur $\tilde{E}_N$ est ainsi égal à :

$$\tilde{E}_N = \int_{\mathbb{R}^n} \phi(\mathbf{x}) \hat{f}_N(\mathbf{x}) \mathrm{d}\mathbf{x}.$$

Lorsqu'on souhaite échantillonner selon f, on cherche donc à obtenir un échantillon pondéré $(\mathbf{X}_1, \bar{w}_1), \ldots, (\mathbf{X}_N, \bar{w}_N)$ pour construire $\hat{f}_N$ (1.14) et estimer une intégrale ou bien estimer la constante $\hat{c}_N$.

Nous revenons sur cette méthode d'échantillonnage dans la section 1.5, où nous présentons également deux autres techniques d'échantillonnage utilisées dans la thèse. Dans le paragraphe suivant, nous décrivons certains algorithmes d'AIS principalement développés pour les problèmes d'inférences bayésiennes.

1.4.2 Algorithmes d'échantillonnage préférentiel adaptatif dans le cadre bayésien

Les méthodes présentées dans cette section sont des approches paramétriques mettant à jour un vecteur moyenne $\mathbf{m}$, et dans certains cas, une matrice de covariance Σ associés aux lois auxiliaires d'échantillonnage préférentiel. Nous résumons les algorithmes en considérant que les densités auxiliaires sont gaussiennes ($g_{\mathbf{m},\Sigma}$), mais ils sont évidemment applicables avec d'autres types de lois.

1.4.2.1 L'algorithme "*Population Monte Carlo*" et ses variantes

Un premier algorithme d'AIS populaire, ayant engendré plusieurs variantes par la suite, est l'algorithme PMC, ou "*Population Monte Carlo*", développé dans [Cappé et al., 2004]. Le principe de cet algorithme est d'utiliser N densités auxiliaires (paramétriques), $g_i = g_{\mathbf{m}_i,\Sigma}$, pour générer N échantillons $\mathbf{X}_i$ (Σ est fixée et sera omise dans les notations). Chaque $\mathbf{X}_i$ est ensuite associé à un poids $w_i = \frac{\hat{f}(\mathbf{X}_i)}{g_i(\mathbf{X}_i)}$, qui est normalisé en $\bar{w}_i$, de sorte que $\sum_i \bar{w}_i = 1$. On procède ensuite à un ré-échantillonnage de N échantillons avec remise sur les $(\mathbf{X}_1, \bar{w}_1), \ldots, (\mathbf{X}_N, \bar{w}_N)$ de sorte à éliminer les échantillons qui ont un faible poids. Les $\tilde{\mathbf{X}}_i$ ainsi générés dans l'étape de ré-échantillonnage déterminent les nouvelles moyennes $\mathbf{m}_i = \tilde{\mathbf{X}}_i$, utilisées dans l'itération suivante. L'algorithme est initialisé avec N paramètres, en fixant le nombre T d'itérations, et retourne l'estimation de la densité cible : $\hat{f}_N(\mathbf{x}) = \sum_{i=1}^N \bar{w}_i \delta_{\mathbf{X}_i}(\mathbf{x})$, permettant d'estimer une intégrale, ou bien l'estimation de la constante de normalisation $\hat{c}_N = \frac{1}{N} \sum_{i=1}^N w_i$. L'algorithme 3 résume les différentes étapes de la méthode PMC de base.

L'algorithme PMC est simple à mettre en œuvre mais son efficacité dépend beaucoup du choix des paramètres initiaux. De plus, le fait que chaque échantillon soit tiré selon une loi différente peut entrainer une grande variance des poids et fait apparaitre le phénomène de dégénérescence des poids, dans lequel les poids normalisés $\bar{w}_i$ sont presque tous nuls sauf un qui est proche de 1 (voir section 2.1.2). Cela implique alors une mauvaise estimation finale en terme de variance de l'estimateur.

Une première amélioration de cet algorithme, appelée "Mixture PMC" (M-PMC) a été proposée dans [Cappé et al., 2008] en utilisant des mélanges de J densités, $\sum_{j=1}^J \alpha_j g_{\mathbf{m}_j}$, où $\sum_{j=1}^J \alpha_j = 1$, au lieu des densités uniques. De plus, les paramètres α_j et $\mathbf{m}_j$ sont mis à jour en minimisant la divergence de Kullback-Leibler avec la densité cible. Les matrices de covariance Σ_j peuvent également être mises à jour dans cette version de l'algorithme, et des formules analytiques sont disponibles

Algorithme 3 : PMC, "*Population Monte Carlo*" [Cappé et al., 2004]

Données : Taille de l'échantillon N, nombre d'itérations T, paramètres initiaux $\mathbf{m}_1^0, \ldots, \mathbf{m}_N^0$

Résultat : Estimation de la densité f par $\hat{f}_N(\mathbf{x}) = \sum_{i=1}^N \bar{w}_i \delta_{\mathbf{X}_i}(\mathbf{x})$ ou de la constante de normalisation $\hat{c}_N = \frac{1}{N}\sum_{i=1}^N w_i$.

1 Initialisation : $g_1 = g_{\mathbf{m}_1^0}, \ldots, g_N = g_{\mathbf{m}_N^0}$;
2 **pour** $t = 1 \ldots T$ **faire**
3 Générer $\mathbf{X}_1 \sim g_1, \ldots, \mathbf{X}_N \sim g_N$;
4 Pour tout $i = 1, \ldots, N$, calculer les poids des $\mathbf{X}_i$: $w_i = \tilde{f}(\mathbf{X}_i)/g_i(\mathbf{X}_i)$, puis les normaliser $\bar{w}_i = w_i / \sum_{k=1}^N w_k$;
5 **si** $t<T$ **alors**
6 Rééchantillonner N valeurs avec remise sur l'ensemble $\{(\mathbf{X}_1, \bar{w}_1), \ldots, (\mathbf{X}_N, \bar{w}_N)\}$, en prenant en compte les poids $\bar{w}_i$, pour former un nouvel échantillon $\tilde{\mathbf{X}}_1, \ldots, \tilde{\mathbf{X}}_N$;
7 Mettre à jour les paramètres $\mathbf{m}_i^t = \tilde{\mathbf{X}}_i$ et les densités auxiliaires $g_i = g_{\mathbf{m}_i^t}$;
8 **fin**
9 **fin**
10 Estimer $\hat{f}_N(\mathbf{x}) = \sum_{i=1}^N \bar{w}_i \delta_{\mathbf{X}_i}(\mathbf{x})$ ou $\hat{c}_N = \frac{1}{N}\sum_{i=1}^N w_i$.

pour l'estimation des paramètres, notamment dans le cadre gaussien. Ainsi, les N échantillons $\mathbf{X}_i$ sont générés selon le mélange $\sum_{j=1}^J \alpha_j g_{\mathbf{m}_j}$, et les poids sont égaux à $w_i = \tilde{f}(\mathbf{X}_i)/\sum_{j=1}^J \alpha_j g_{\mathbf{m}_j}(\mathbf{X}_i)$. Les paramètres sont ensuite mis à jour à l'aide de ces échantillons pondérés. Contrairement à PMC, il n'y a pas d'étape de ré-échantillonnage dans M-PMC. Cet algorithme est plus robuste que PMC et permet de réduire la variance de l'estimateur.

Plus récemment, la méthode "*Deterministic Mixture*" PMC (DM-PMC) [Elvira et al., 2016], reprend le même schéma que l'algorithme PMC d'origine mais en considérant J densités pour générer N échantillons selon chacune des densités : $\mathbf{X}_{i,j} \sim g_{\mathbf{m}_j}$ pour $i = 1, \ldots, N$ et $j = 1, \ldots, J$. Les poids associés sont alors égaux à $w_{i,j} = \tilde{f}(\mathbf{X}_{i,j})J/\sum_{k=1}^J g_{\mathbf{m}_k}(\mathbf{X}_{i,j})$. Un ré-échantillonnage de N valeurs avec remise est ensuite effectué sur les $\mathbf{X}_{i,j}$ munis de leurs poids normalisés $\bar{w}_{i,j}$. [Elvira et al., 2016] montrent théoriquement que cette modification offre une plus grande stabilité et une variance de l'estimateur inférieure à celle de PMC.

1.4.2.2 L'algorithme "*Adaptive Multiple Importance Sampling*"

L'algorithme AMIS ("*Adaptive Multiple Importance Sampling*") développé dans [Cornuet et al., 2012] propose de prendre en compte tous les échantillons et les densités auxiliaires des précédentes itérations dans le calcul des poids. À chaque étape t, il y a de plus une mise à jour des poids des itérations $l = 0$ à $t - 1$ avec les échantillons nouvellement générés. Les paramètres (moyenne et covariance) sont mis à jour en minimisant la divergence de Kullback-Leibler, avec les échantillons pondérés. L'algorithme 4 présente le déroulement de la méthode AMIS, avec des densités auxiliaires gaussiennes, pour lesquelles il existe des formules analytiques pour l'estimation des paramètres.

Le recyclage de tous les échantillons générés durant l'algorithme permet d'estimer les paramètres avec un plus grand nombre de valeurs et ainsi de gagner en précision, par rapport à un algorithme d'AIS classique. La variance de l'estimateur final est également réduite grâce à ce

Algorithme 4 : AMIS : "*Adaptive Multiple Importance Sampling*" [Cornuet et al., 2012]

Données : Taille de l'échantillon N, nombre d'itérations T, paramètres initiaux $\mathbf{m}_0$, Σ_0

Résultat : Estimation de la densité f par $\hat{f}_N(\mathbf{x}) = \sum_{l=0}^{T} \sum_{i=1}^{N} \bar{w}_i^l \delta_{\mathbf{X}_i^{(l)}}(\mathbf{x})$, ou de la constante de normalisation $\hat{c}_N = \frac{1}{(T+1)N} \sum_{l=0}^{T} \sum_{i=1}^{N} w_i^l$

1 **pour** $t = 0 \dots T$ **faire**

2 Générer $\mathbf{X}_1^{(t)}, \dots, \mathbf{X}_N^{(t)}$ indépendamment selon $g_{\mathbf{m}_t,\Sigma_t}$;

3 Pour tout $i = 1, \dots, N$, calculer le mélange de densités : $S_i^t = \sum_{l=0}^{t} g_{\mathbf{m}_l,\Sigma_l}\left(\mathbf{X}_i^{(t)}\right)$;

4 Calculer les poids des $\mathbf{X}_i^{(t)}$: $w_i^t = \tilde{f}\left(\mathbf{X}_i^{(t)}\right) \Big/ \left(\dfrac{S_i^t}{t+1}\right)$;

5 Pour tout $l = 0, \dots, t-1$ et $i = 1, \dots, N$ mettre à jour les poids des précédentes itérations : $S_i^l \leftarrow S_i^l + g_{\mathbf{m}_t,\Sigma_t}(\mathbf{X}_i^{(l)})$ et $w_i^l \leftarrow \tilde{f}\left(\mathbf{X}_i^{(l)}\right) \Big/ \left(\dfrac{S_i^l}{t+1}\right)$;

6 Estimer $\mathbf{m}_{t+1} = \sum_{l=0}^{t} \sum_{i=1}^{N} \bar{w}_i^l \mathbf{X}_i^{(l)}$ et $\Sigma_{t+1} = \sum_{l=0}^{t} \sum_{i=1}^{N} \bar{w}_i^l (\mathbf{X}_i^{(l)} - \mathbf{m}_{t+1})(\mathbf{X}_i^{(l)} - \mathbf{m}_{t+1})^\top$ où $\bar{w}_i^l = w_i^l / \sum_{l=0}^{t} \sum_{i=1}^{N} w_i^l$;

7 **fin**

8 Estimer $\hat{f}_N(\mathbf{x}) = \sum_{l=0}^{T} \sum_{i=1}^{N} \bar{w}_i^l \delta_{\mathbf{X}_i^{(l)}}(\mathbf{x})$ ou $\hat{c}_N = \frac{1}{(T+1)N} \sum_{l=0}^{T} \sum_{i=1}^{N} w_i^l$.

procédé. Cependant, le fait de ne mettre à jour qu'une seule densité auxiliaire rend l'algorithme AMIS moins efficace pour traiter des problèmes multimodaux. Pour y remédier, les auteurs font remarquer qu'il est possible d'utiliser un mélange de densités plutôt qu'une unique densité. Notons enfin qu'une modification de l'algorithme a été apportée par [Marin et al., 2012], où les paramètres sont estimés sans prendre en compte les échantillons des itérations précédentes et seule l'estimation finale bénéficie du recyclage de tous les échantillons. Cette modification a été introduite principalement pour démontrer des résultats de convergence de l'estimateur de l'intégrale.

On termine cette partie en citant la méthode "*Adaptive Population Importance Sampling*" (APIS) [Martino et al., 2015], qui s'appuie également sur le procédé de recyclage des échantillons proposé dans AMIS. L'algorithme reprend ensuite le même schéma que PMC standard en prenant en compte les échantillons de toutes les itérations. Une amélioration a ensuite été proposée avec "*Gradient APIS*" (GAPIS) [Elvira et al., 2015], où les paramètres (moyennes et covariances) sont mis à jour à l'aide du gradient et de la matrice hessienne de la fonction $\tilde{f}$ (proportionnelle à la densité cible f). Ce dernier algorithme est particulièrement robuste mais la nécessité du calcul du gradient est son principal défaut, du fait du cout supplémentaire que cela peut engendrer.

1.5 Méthodes d'échantillonnage selon une loi cible

Pour estimer les paramètres optimaux (1.11), il est nécessaire d'avoir un échantillon de loi de densité g^*. Dans les paragraphes qui suivent nous décrivons des méthodes utilisées pour obtenir un échantillon selon une loi cible.

1.5.1 La méthode du rejet

La méthode du rejet permet de simuler une variable aléatoire selon une loi cible, que nous appelons toujours g^*, à partir d'une loi auxiliaire h avec laquelle on sait échantillonner facilement. Pour cela, on suppose qu'il existe une constante $b > 1$, telle que $g^*(\mathbf{x}) \leq b \cdot h(\mathbf{x})$. Étant donné U une variable uniforme sur $[0, 1]$, et $\mathbf{Z}$ une variable de loi h, toutes deux indépendantes, on peut ainsi montrer (voir [Robert and Casella, 2004]) que la loi de g^* est la loi conditionnelle de $\mathbf{Z}$ sachant l'événement $\Omega = \{(U, \mathbf{Z}), b \cdot Uh(\mathbf{Z}) \leq g^*(\mathbf{Z})\}$. Ainsi, pour tirer une variable aléatoire selon g^*, on applique l'algorithme suivant :

Algorithme 5 : Méthode du rejet

1 Générer $\mathbf{Z}$ selon h ;
2 Générer $U \sim \mathcal{U}([0, 1])$ une variable uniforme sur $[0, 1]$;
3 **tant que** $b \cdot Uh(\mathbf{Z}) > g^*(\mathbf{Z})$ **faire**
4 | Répéter les étapes 1 et 2 ;
5 **fin**
6 Poser $\mathbf{X} = \mathbf{Z}$.

Dans le cas où $g^* = \phi f/E$ est la densité optimale d'IS (1.5), l'algorithme de rejet peut s'appliquer simplement dès que l'on connait un majorant, $\phi_{\max}$, de la fonction ϕ (c'est le cas par exemple lorsque ϕ est une indicatrice, comme dans la section 1.3). On peut alors poser $b = \phi_{\max}/E$, $h = f$, et la condition de l'algorithme se réécrit $U \cdot \phi_{\max} > \phi(\mathbf{Z})$.

Le nombre d'itérations T de cet algorithme est aléatoire et suit une loi géométrique de paramètre $1/b$, dont l'espérance vaut b. La méthode est donc efficace et peu couteuse si la constante b est suffisamment petite. C'est précisément la difficulté de cet algorithme, à savoir qu'il n'est pas évident de trouver une densité h de sorte que b soit assez petite, d'autant plus lorsque ϕ n'admet pas d'expression analytique.

1.5.2 Monte-Carlo par chaines de Markov

Les méthodes de Monte-Carlo par chaines de Markov ("*Markov Chain Monte Carlo*", MCMC) permettent de générer un échantillon selon une approximation d'une densité cible (g^* dans notre cas), sans nécessairement connaitre sa constante de normalisation. Elles consistent à construire une chaine de Markov dont la loi stationnaire est la loi cible g^*. Nous commençons par décrire le principe de l'algorithme de Metropolis-Hastings (MH), [Metropolis et al., 1953], [Hastings, 1970], qui est une des méthodes MCMC les plus populaires. Il repose principalement sur l'itération de deux étapes : une étape d'exploration de l'espace, et une étape d'acceptation-rejet. La phase d'exploration consiste à générer un nouvel échantillon à l'aide du noyau de transition de la chaine de Markov, étant donné l'état précédent de la chaine. Ce nouvel échantillon est accepté avec une certaine probabilité, la probabilité d'acceptation, dans la phase d'acceptation-rejet. Considérons une densité notée h définissant un noyau de transition de la chaine de Markov, c'est-à-dire qui permet de passer d'un état de la chaine au suivant. Étant donné un état $\mathbf{X}_i$ de la chaine, les deux phases de l'algorithme MH sont les suivantes :

- Exploration : on génère $\mathbf{Y}_i$ selon $h(\mathbf{u}|\mathbf{X}_i)$.

- Acceptation-rejet : on pose $\mathbf{X}_{i+1} = \mathbf{Y}_i$ avec probabilité $\alpha(\mathbf{X}_i, \mathbf{Y}_i) = \min\left(1, \dfrac{g^*(\mathbf{Y}_i)h(\mathbf{X}_i|\mathbf{Y}_i)}{g^*(\mathbf{X}_i)h(\mathbf{Y}_i|\mathbf{X}_i)}\right)$, et $\mathbf{X}_{i+1} = \mathbf{X}_i$ avec probabilité $1 - \alpha(\mathbf{X}_i, \mathbf{Y}_i)$.

La chaine est initialisée à $\mathbf{X}_0$, tiré selon une loi choisie arbitrairement, avant d'itérer les deux étapes décrites précédemment un certain nombre de fois. La loi des $\mathbf{X}_i$ ainsi générés converge vers la loi de densité g^*, dès que le noyau de transition vérifie les hypothèses d'irréductibilité et d'apériodicité requises dans le théorème ergodique [Roberts and Smith, 1994]. Un choix classique de densités de transition h correspond à une densité vérifiant la propriété de symétrie, $h(\mathbf{u}|\mathbf{x}) = h(\mathbf{x}|\mathbf{u})$, permettant de simplifier la probabilité d'acceptation ($\alpha(\mathbf{x}, \mathbf{y}) = \min(1, g^*(\mathbf{y})/g^*(\mathbf{x}))$). Un exemple courant est de générer $\mathbf{Y}_i$ selon la loi normale centrée en $\mathbf{X}_i$, $\mathcal{N}(\mathbf{X}_i, I_n)$, ou bien la loi uniforme sur l'hypercube centré en $\mathbf{X}_i$ (voir [Au and Beck, 2001], [Bourinet, 2018]). Des choix alternatifs sont discutés dans [Chib and Greenberg, 1995], notamment en prenant des densités proches de g^*, ou en échantillonnant indépendamment de l'état précédent ($\mathbf{X}_i$), l'objectif étant de rendre l'algorithme simple à mettre en œuvre et rapide à converger.

Les différents états de la chaine de Markov forment alors un échantillon de loi approchant g^*, d'après le théorème ergodique, et pouvant servir à estimer les espérances (1.10) avec l'estimateur de Monte-Carlo (1.3). En général, les premiers états de la chaine sont retirés pour effectuer l'estimation puisqu'il faut un certain temps, appelé "temps de chauffe", avant de converger vers la loi cible. L'estimation de la moyenne (1.10) s'écrit alors

$$\frac{1}{N}\sum_{i=N_0+1}^{N+N_0} \mathbf{X}_i,$$

où les $\mathbf{X}_i$, pour $i = N_0 + 1, \ldots, N + N_0$, suivent une loi approchant g^*.

Une limite de Metropolis-Hastings est son inefficacité en grande dimension car la probabilité d'acceptation s'approche de 0 lorsque la dimension augmente. [Au and Beck, 2001] proposent un algorithme de Metropolis-Hastings modifié où les échantillons sont générés composantes par composantes selon des lois univariées, ce qui le rend plus robuste en grande dimension. Une autre méthode MCMC pouvant être plus efficace en grande dimension est l'échantillonneur de Gibbs [Geman and Geman, 1984], cas particulier de l'algorithme MH, qui génère l'échantillon composante par composante selon les lois conditionnelles de la loi cible, supposées connues. Mais de manière générale, toutes les techniques MCMC peuvent engendrer un temps et un cout de calcul important, notamment car elles nécessitent un "temps de chauffe" avant de converger vers la loi cible, et il n'est pas aisé de savoir à l'avance quand la loi de la chaine est suffisamment proche de la distribution cible.

1.5.3 L'échantillonnage préférentiel pour générer un échantillon selon une loi cible

L'échantillonnage préférentiel est une méthode d'estimation, comme évoqué dans la section 1.2.2, mais il peut aussi être vu comme une méthode d'échantillonnage pour simuler selon une loi donnée. En effet, on a vu (section 1.4.1) que $f \propto \tilde{f}$ pouvait être approchée par :

$$\hat{f}_N(\mathbf{x}) = \sum_{i=1}^{N} \bar{w}_i \delta_{\mathbf{X}_i}(\mathbf{x}), \tag{1.15}$$

où $\bar{w}_i = w_i / \sum_{j=1}^{N} w_j$ sont les poids normalisés, $w_i = \tilde{f}(\mathbf{X}_i)/g(\mathbf{X}_i)$, et $\mathbf{X}_1, \ldots, \mathbf{X}_N$ est un échantillon i.i.d. généré selon une loi auxiliaire donnée g. Ainsi, l'échantillon pondéré $(\mathbf{X}_1, \bar{w}_1), \ldots, (\mathbf{X}_N, \bar{w}_N)$

suit la loi $\hat{f}_N$ qui tend vers la loi cible f. La précision de cette estimation repose bien entendu sur la densité auxiliaire g. La densité qui minimise la variance des poids étant exactement f (voir section 1.4.1), un choix de densité efficace serait de prendre g suffisamment proche de f, suivant les informations que l'on a sur la loi cible. En général, on obtient un échantillon selon $\hat{f}_N$ grâce aux algorithmes d'AIS précédemment décrits, en mettant à jour les densités auxiliaires successivement afin d'estimer une intégrale.

Remarque 1.5.1. Il est également possible d'approcher g^* par la même méthode, en prenant $w_i = \varphi(\mathbf{X}_i)f(\mathbf{X}_i)/g(\mathbf{X}_i)$. Cependant, il faut noter qu'un échantillon généré selon g^* ne sert pas à estimer directement l'espérance E. En effet, pour pouvoir calculer $\hat{E}_N$ (1.4) ou $\tilde{E}_N$(1.13), il faut connaitre l'expression de la densité g^*, qui dépend elle-même de l'inconnue E et n'est donc pas exploitable.

Dans le chapitre suivant, nous nous intéressons à la performance des méthodes d'échantillonnage préférentiel paramétrique en grande dimension. En effet, tous les algorithmes d'AIS présentés dans ce chapitre connaissent des difficultés et deviennent imprécis lorsque la dimension des paramètres augmentent. Ces problèmes dûs à la dimension apparaissent plus généralement dans la méthode d'échantillonnage préférentiel, lors de l'estimation des paramètres. C'est ce que nous tentons d'expliquer dans la suite. Nous étudions particulièrement les deux algorithmes (CE et iCE) présentés dans cette partie, en choisissant toujours la famille gaussienne comme famille de densités auxiliaires.

Chapitre 2

Échantillonnage préférentiel en grande dimension

Sommaire

L'échantillonnage préférentiel est couramment utilisé pour estimer une espérance dans différents domaines, au travers d'algorithmes adaptatifs. Si les nombreuses méthodes d'AIS développées (dont quelques-unes sont décrites dans le chapitre 1) sont performantes lorsque la dimension de l'espace des paramètres est assez petite (inférieure à 10 ou 15 environ), leur efficacité se dégrade dès que la dimension augmente et que le budget de simulation est limité. Dans ce chapitre, nous commençons par illustrer cette dégradation sur des exemples simples, avant d'évoquer deux raisons principales pouvant expliquer l'inefficacité de l'échantillonnage préférentiel en grande dimension. Dans un second temps, nous décrivons plusieurs méthodes mises en place ces dernières années pour améliorer l'échantillonnage préférentiel en grande dimension, en expliquant leurs avantages et leurs limites. En revanche, les techniques de projection permettant de réduire la dimension seront étudiées dans les chapitres suivants.

2.1 Dégradation de l'échantillonnage préférentiel en grande dimension

2.1.1 Dégradation d'un algorithme d'échantillonnage préférentiel adaptatif

Dans cette partie, nous commençons par montrer l'inefficacité en grande dimension de l'algorithme d'entropie croisée classique, CE (1), et celui sous sa forme "améliorée", iCE (2), deux algorithmes d'AIS décrits dans le chapitre 1. On cherche à estimer la probabilité d'événement rare $E = \mathbb{P}_f(\varphi_1(\mathbf{X}) \geq 0)$ avec φ_1 la fonction suivante :

$$\varphi_1 : \mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n \mapsto \sum_{j=1}^{n} x_j - 3\sqrt{n}. \tag{2.1}$$

Cette application est un cas-test classique pour tester l'efficacité d'un algorithme en fiabilité, notamment lorsque la dimension varie (voir par exemple [Engelund and Rackwitz, 1993]).

La valeur théorique de la probabilité E est indépendante de la dimension n et est égale à $E = \mathbb{P}(\mathbf{U} \geq 3) \simeq 1.35 \times 10^{-3}$, où $\mathbf{U}$ suit la loi normale standard sur $\mathbb{R}$. L'espérance E est estimée à l'aide des algorithmes CE et iCE en faisant varier la dimension n de 5 à 60. On rappelle que la famille auxiliaire d'échantillonnage choisie, $\mathcal{G}$, est la famille Gaussienne $\{g_{\mathbf{m},\Sigma}\}$, avec $\mathbf{m} \in \mathbb{R}^n$ la moyenne et $\Sigma \in \mathcal{S}_n^+$ la matrice de covariance. Les valeurs optimales (i.e. celles minimisant la divergence de Kullback-Leibler avec la densité d'IS optimale g^* (1.5)) de la moyenne $\mathbf{m}^*$ et de la covariance Σ^* données par les formules (1.10), peuvent être calculées explicitement dans cet exemple : $\mathbf{m}^* = m^* \cdot \mathbf{1}_n$, avec $m^* = \dfrac{e^{-9/2}}{E\sqrt{2\pi}}$ et $\mathbf{1}_n = \frac{1}{\sqrt{n}}(1, \ldots, 1)^\top \in \mathbb{R}^n$, et $\Sigma^* = (v^* - 1)\mathbf{1}_n\mathbf{1}_n^\top + I_n$, où $v^* = 3m^* - (m^*)^2 + 1$ (voir la démonstration en annexe A.1).

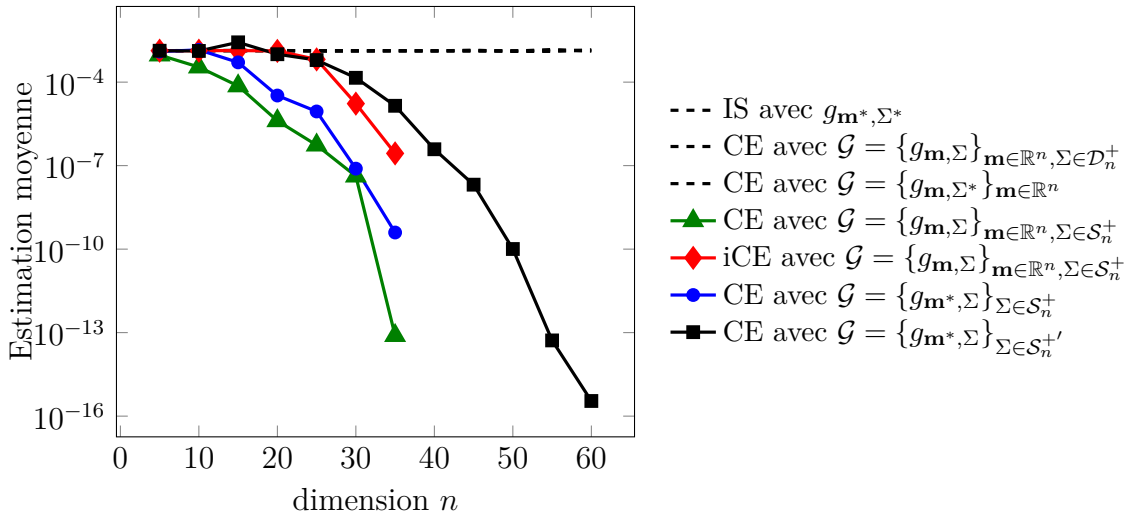


Figure 2.1 – Évolution de l'estimation de la probabilité $E = \mathbb{P}_f(\varphi_1(\mathbf{X}) \geq 0)$ en fonction de la dimension par différents algorithmes CE et iCE, avec φ_1 définie en 2.1. Les paramètres utilisés sont $\rho = 0.1$ pour la CE, $\delta = 1.5$ pour iCE, et $N = 1000$ dans les deux méthodes. Chaque valeur est une moyenne sur 100 estimations indépendantes de la probabilité.

Le but est d'estimer la probabilité E à l'aide des algorithmes CE et iCE en faisant varier la dimension n. Les paramètres utilisés sont $\rho = 0.1$ pour la CE, $\delta = 1.5$ pour iCE, et $N = 1000$

dans les deux méthodes. L'évolution de l'estimation moyenne de E en fonction de la dimension est représentée figure 2.1. Les algorithmes CE et iCE de base (avec $\mathcal{G} = \{g_{\mathbf{m},\Sigma}\}_{\mathbf{m}\in\mathbb{R}^n,\Sigma\in\mathcal{S}_n^+}$) sont représentés par les triangles verts et losanges rouges respectivement. Lorsque la dimension augmente, l'estimation devient de moins en moins précise pour ces deux méthodes. Pour la CE, l'erreur d'estimation est déjà significative à partir de la dimension $n \approx 15$, alors qu'iCE est assez précis jusqu'à $n \approx 25$, avant de se dégrader fortement. Cette dégradation est due aux erreurs d'estimation des paramètres, comme le montrent les autres courbes de la figure 2.1. En effet, CE et iCE approchent la densité optimale g^* d'IS par une densité auxiliaire de la famille $\{g_{\mathbf{m},\Sigma}\}_{\mathbf{m}\in\mathbb{R}^n,\Sigma\in\mathcal{S}_n^+}$, en utilisant des techniques de simulation séquentielle pour obtenir des estimateurs, $\hat{\mathbf{m}}^*$ et $\hat{\Sigma}^*$ (1.12), des paramètres optimaux $\mathbf{m}^*$ et Σ^*. On a ainsi n paramètres mis à jour successivement pour estimer $\mathbf{m}^*$ et $n(n+1)/2$ pour estimer Σ^* (car Σ^* est symétrique de taille n), ce qui donne $n(n+3)/2$ paramètres à estimer au total. Ce nombre grandit de manière quadratique avec la dimension n, et l'accumulation de toutes les erreurs commises dans chaque dimension peut entraîner une grande imprécision dans l'estimation finale. En effet, si l'on estime la probabilité par échantillonnage préférentiel avec $g_{\mathbf{m}^*,\Sigma^*}$ (en pointillés sur la figure 2.1), c'est-à-dire avec les valeurs théoriques des paramètres optimaux, l'estimation reste très précise quelle que soit la dimension, ce qui indique que l'erreur provient des approximations $\hat{\mathbf{m}}^*$ et $\hat{\Sigma}^*$. Les quatre autres courbes sont des cas intermédiaires entre l'IS avec $g_{\mathbf{m}^*,\Sigma^*}$ comme densité auxiliaire, où aucun paramètre n'est estimé, et la CE avec $\{g_{\mathbf{m},\Sigma}\}_{\mathbf{m}\in\mathbb{R}^n,\Sigma\in\mathcal{S}_n^+}$, où $n(n+3)/2$ paramètres sont estimés. Dans les cas où la famille est $\{g_{\mathbf{m},\Sigma^*}\}_{\mathbf{m}\in\mathbb{R}^n}$ ou $\{g_{\mathbf{m},\Sigma}\}_{\mathbf{m}\in\mathbb{R}^n,\Sigma\in\mathcal{D}_n^+}$, on estime n et $2n$ paramètres respectivement ($\mathcal{D}_n^+$ étant l'ensemble des matrices diagonales où tous les coefficients diagonaux sont strictement positifs), et l'estimation finale est à nouveau très précise (courbe en pointillés). Dans le cas $\{g_{\mathbf{m}^*,\Sigma}\}_{\Sigma\in\mathcal{S}_n^{+\prime}}$ (carrés noirs), on a fixé la moyenne à sa valeur théorique, et $\mathcal{S}_n^{+\prime}$ représente l'ensemble des matrices de $\mathcal{S}_n^+$ où l'on estime 3/4 de la matrice, le dernier quart prenant les valeurs exactes de Σ^*. On a ainsi $3n(n+1)/8$ paramètres à estimer et on peut observer une meilleure performance que dans le cas où on estime toute la matrice de covariance, c'est-à-dire $n(n+1)/2$ coefficients (cercles bleus), lui-même légèrement meilleur que la CE de base (triangles verts) où $n(n+3)/2$ paramètres sont mis à jour.

Finalement, la figure 2.1 montre que la performance de la CE dépend du nombre de paramètres à estimer, autrement dit moins il y a de paramètres et plus la précision de l'algorithme augmente. Comme la plupart des paramètres sont issus de la matrice de covariance ($\approx n^2/2$), cela soulève la question de son estimation en grande dimension, qui est un problème existant en dehors du cadre de l'échantillonnage préférentiel. Nous évoquons ce problème dans la section 2.1.3 et montrons son influence sur la dégradation de la divergence de Kullback-Leibler. Par ailleurs, l'algorithme CE, comme les autres algorithmes AIS, subissent le phénomène de dégénérescence des poids en grande dimension. Notre figure ne permet pas de mesurer ce phénomène mais la dégénérescence des poids peut entrainer la défaillance des méthodes d'échantillonnage préférentiel comme nous l'évoquons dans la partie 2.1.2.

2.1.2 Le phénomène de dégénérescence des poids en grande dimension

Une raison souvent mise en avant pour expliquer la dégradation de l'échantillonnage préférentiel, et des algorithmes AIS, en grande dimension est la dégénérescence des poids ("*weight degeneracy*"). En effet, il arrive que la majorité des poids normalisés s'effondre rapidement vers 0, en particulier lorsque la dimension augmente (voir [Cappé et al., 2004], [Koblents and Míguez, 2015], [Rubinstein and Glynn, 2009]). Ce phénomène est illustré sur la figure 2.2 qui représente la valeur des poids

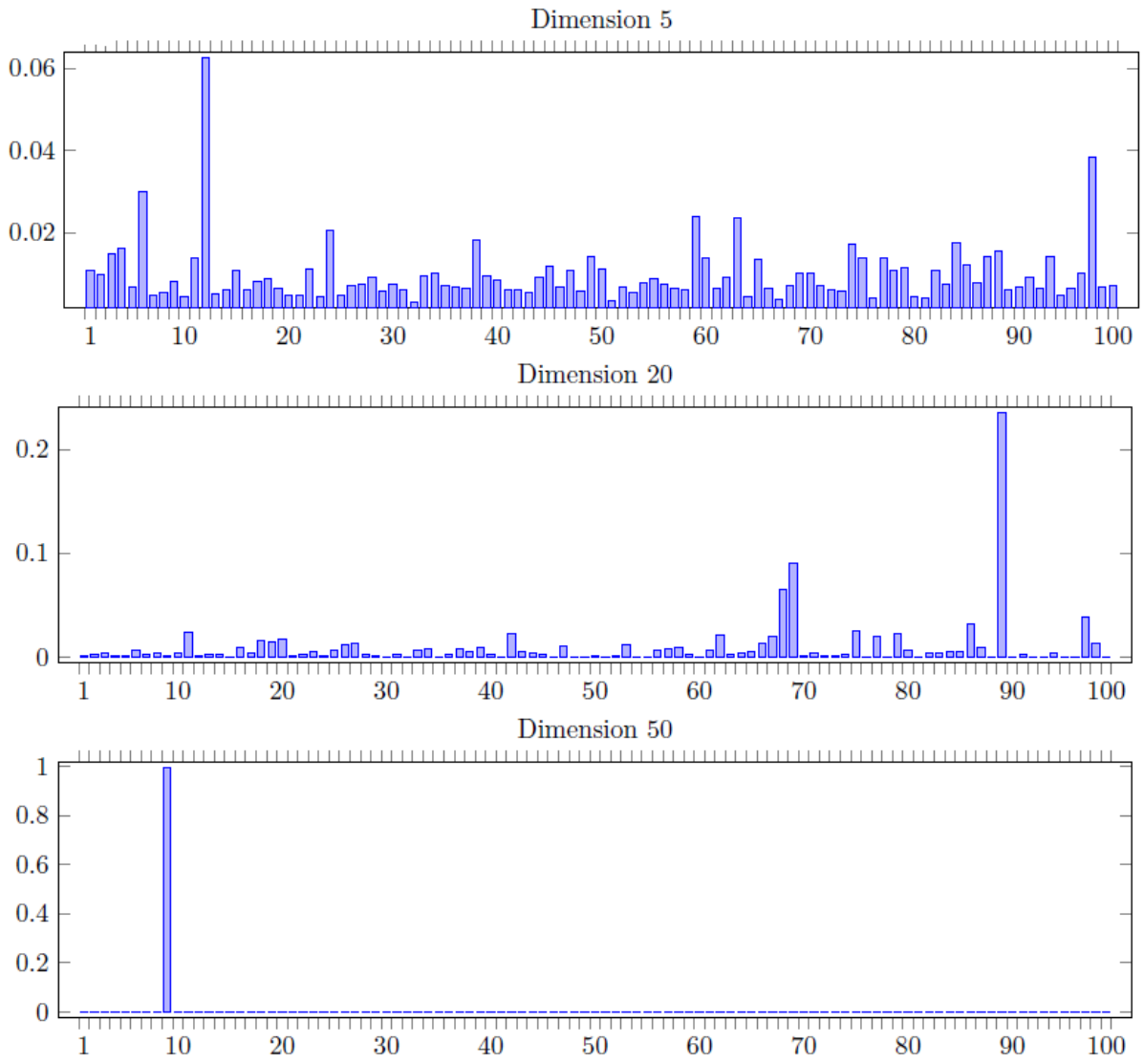


Figure 2.2 – Valeurs des poids normalisés lors d'une itération de l'algorithme CE en dimension $n = 5$, 20, et 50, avec une taille d'échantillon par itération $N = 1000$, et le paramètre $\rho = 0.1$.

normalisés $\bar{w}_i$ (définis dans l'algorithme 1) à une même itération de l'algorithme CE dans l'exemple considéré au paragraphe précédent. Les poids représentés sont les poids non nuls (associés aux échantillons situés dans la région de défaillance intermédiaire) lors d'une réalisation de l'algorithme CE, en dimension $n = 5$, 20, et 50. Les valeurs de ces poids sont assez proches en dimension 5 (entre 0.005 et 0.06 environ). Lorsque la dimension augmente, un échantillon semble dominer tous les autres, et c'est particulièrement flagrant pour $n = 50$, où l'un des poids vaut presque 1, alors que tous les autres sont quasiment nuls (de l'ordre de 10^{-5} ou moins). Cela revient donc à mettre à jour les paramètres suivants avec un échantillon de taille 1, ce qui explique la dégradation de l'algorithme CE. Ce phénomène a été plusieurs fois observé dans la littérature.

C'est le cas par exemple dans le cadre des filtres particulaires avec [Bengtsson et al., 2008]. Les méthodes de filtres particulaires [Doucet et al., 2009], aussi appelées méthodes de Monte-Carlo séquentielles [Cérou et al., 2012], sont des techniques d'estimation d'espérances en inférence bayésienne, où l'on souhaite échantillonner selon une loi a posteriori étant donné des observations.

Plus précisément, le but est d'estimer une espérance du type :

$$\mathbb{E}(\phi(\mathbf{X})|\mathbf{Y}) = \int \phi(\mathbf{x}) \frac{l(\mathbf{Y}|\mathbf{x})h_0(\mathbf{x})}{c(\mathbf{Y})} \mathrm{d}\mathbf{x},$$

où h_0 est la densité a priori, $l(\mathbf{Y}|\mathbf{x})$ la vraisemblance des observations $\mathbf{Y}$ étant donné $\mathbf{x}$, et $c(\mathbf{Y})$ est la constante de normalisation (égale à $\int l(\mathbf{Y}|\mathbf{x}')h_0(\mathbf{x}')\mathrm{d}\mathbf{x}'$) de la densité a posteriori $l(\mathbf{Y}|\mathbf{x})h_0(\mathbf{x}) = h(\mathbf{x}|\mathbf{Y})$. L'espérance peut alors être estimée par $\sum_{i=1}^N \phi(\mathbf{X}_i)\bar{w}_i$, avec $\bar{w}_i = l(\mathbf{Y}|\mathbf{X}_i)/\sum_j l(\mathbf{Y}|\mathbf{X}_j)$, et $\mathbf{X}_i$ des échantillons générés selon h_0. En grande dimension [Bengtsson et al., 2008] montrent dans le cas du filtrage particulaire que $\max \bar{w}_i \approx 1$ et que tous les autres poids sont presque nuls, si la taille de l'échantillon n'est pas suffisante, comme observé sur le graphique 2.2 en dimension 50 dans le cas de la CE. Par ailleurs, ils prouvent, dans le cas gaussien, que si $\ln(N)/n$ tend vers $+\infty$ lorsque la taille de l'échantillon N et la dimension n tendent vers $+\infty$, alors l'estimateur de l'espérance considéré est consistant. Autrement dit, pour éviter l'effondrement des poids et espérer une estimation précise, la taille de l'échantillon doit croître de façon exponentielle avec la dimension, ce qui n'est pas réalisable en pratique puisque cela entrainerait un budget de simulation démesuré.

Dans un contexte fiabiliste, pour l'estimation de probabilités d'événement rare, [Katafygiotis and Zuev, 2008] adoptent un point de vue géométrique pour montrer, lorsque les densités sont gaussiennes, que les rapports de vraisemblance prennent des valeurs très faibles en grande dimension, ce qui entraine une sous-estimation de la probabilité. En effet, les auteurs s'appuient sur le fait que la norme d'un vecteur aléatoire gaussien standard $\mathbf{X}$ en dimension n, suit une loi s'approchant d'une distribution gaussienne $\mathcal{N}(\sqrt{n}, 1/2)$ lorsque n tend vers l'infini. Des considérations géométriques leur permettent ensuite de montrer que les rapports de vraisemblance ($L(\mathbf{X}_i) = f(\mathbf{X}_i)/g_{\mathbf{m},I_n}(\mathbf{X}_i)$, avec $\mathbf{X}_i \sim g_{\mathbf{m},I_n}$) sont de l'ordre de $e^{-n/2}$ si n est grand. Ainsi, plus la dimension est grande, plus ces poids (non normalisés) sont excessivement petits et plus la probabilité, estimée par $N^{-1}\sum_{i=1}^N \mathbb{I}_{\{\varphi(\mathbf{X}_i)\geq 0\}} L(\mathbf{X}_i)$, est potentiellement sous-évaluée. Ce phénomène, différent de celui décrit dans [Bengtsson et al., 2008], permet également d'expliquer l'inefficacité de l'échantillonnage préférentiel en grande dimension.

Plus généralement, la dégénérescence des poids d'échantillonnage préférentiel en grande dimension est liée au fait que la densité auxiliaire peut être très éloignée de la densité initiale (comme illustré dans la section 2.1.3, avec l'augmentation de la divergence de Kullback-Leibler). Lorsque celle-ci est gaussienne standard, [Au and Beck, 2003] préconisent de prendre une densité gaussienne auxiliaire avec une matrice de covariance proche de l'identité afin d'éviter que le coefficient de variation des poids ne tende vers l'infini. Des méthodes ont été proposées dans la littérature pour éviter ce problème, et nous en décrirons quelques-unes dans la section 2.2.

2.1.3 Estimation d'une matrice de covariance de grande dimension avec un échantillon de petite taille et dégradation de la divergence de Kullback-Leibler

Une autre raison pouvant expliquer la dégradation de l'échantillonnage préférentiel en grande dimension est l'inefficacité de l'estimateur empirique de la matrice de covariance (1.11) lorsque l'échantillon est de petite taille. [Ledoit and Wolf, 2004] rappellent par exemple que si la dimension n est plus grande que la taille de l'échantillon observé N, la matrice empirique est un très mauvais estimateur de la covariance (en particulier, elle n'est même pas inversible, son rang étant au maximum égal à N). Dans le cas où $n \approx N$, avec N légèrement supérieur à n, la matrice est

encore mal conditionnée et induit une grande erreur d'estimation. Ainsi, si la taille de l'échantillon est insuffisante par rapport à la dimension de la matrice estimée, celle-ci peut très mal approcher la matrice de covariance théorique. Dans l'échantillonnage préférentiel, cela peut impliquer une approximation imprécise de la loi optimale d'IS g^* et de ce fait une mauvaise estimation de l'intégrale. Pour obtenir un estimateur plus précis et bien conditionné de la matrice de covariance en grande dimension, des méthodes de "contraction" ("*shrinkage*") des paramètres sont souvent utilisées dans la littérature. Nous en évoquons quelques-unes dans la section suivante.

Pour illustrer l'impact de la dégradation de la matrice de covariance estimée sur la qualité de l'estimation, nous représentons l'évolution de la divergence de Kullback-Leibler en fonction de la dimension. En effet, nous avons vu qu'une grande divergence KL entrainait une grande variance de l'estimateur d'échantillonnage préférentiel (comme souligné dans la section 1.2.4 et [Au and Beck, 2003], [Chatterjee and Diaconis, 2018]).

Nous considérons le problème d'échantillonnage selon une loi cible (que nous appellerons g^* dans un premier temps). Il peut s'agir de la densité optimale d'IS (définie en (1.5)) ou d'une distribution quelconque à partir de laquelle on veut générer un échantillon. Dans cette section, on appellera toujours g^* la densité que l'on souhaite approcher, et la famille auxiliaire d'échantillonnage préférentiel est la famille de densités gaussiennes $\{g_{\mathbf{m},\Sigma} : \mathbf{m} \in \mathbb{R}^n, \Sigma \in \mathcal{S}_n^+\}$. On cherche alors à minimiser la divergence de Kullback-Leibler entre g^* et la famille gaussienne : $D(g^*, g_{\mathbf{m},\Sigma}) = \mathbb{E}_{g^*}\left(\ln\left(\frac{g^*(\mathbf{X})}{g_{\mathbf{m},\Sigma}(\mathbf{X})}\right)\right)$. Les paramètres optimaux obtenus sont : $\mathbf{m}^* = \mathbb{E}_{g^*}(\mathbf{X})$ et $\Sigma^* = \mathrm{Var}_{g^*}(\mathbf{X})$ (1.10). Mais en pratique ceux-ci ne sont pas connus et sont estimés par $\hat{\mathbf{m}}^*$ et $\hat{\Sigma}^*$ (définis en (1.11)) qui sont imprécis lorsque la dimension est grande, en particulier $\hat{\Sigma}^*$, ce qui implique une augmentation de la divergence KL avec la dimension.

Pour observer cet accroissement, simplifions d'abord l'expression de la divergence de Kullback-Leibler. Minimiser $D(g^*, g_{\mathbf{m},\Sigma})$ revient à minimiser la quantité :

$$\tilde{D}(\mathbf{m}, \Sigma) = \ln(\det \Sigma) + \mathbb{E}_{g^*}\left((\mathbf{X} - \mathbf{m})^\top \Sigma^{-1} (\mathbf{X} - \mathbf{m})\right).$$

Comme les erreurs d'estimation proviennent majoritairement de la matrice de covariance, nous nous concentrons sur celle-ci et nous fixons $\mathbf{m}$ à sa valeur optimale $\mathbf{m}^*$ qui minimise la divergence. On définit alors :

$$\begin{aligned} D'(\Sigma) &= \ln\det\Sigma + \mathbb{E}_{g^*}\left((\mathbf{X} - \mathbf{m}^*)^\top \Sigma^{-1} (\mathbf{X} - \mathbf{m}^*)\right) \\ &= \ln\det\Sigma + \mathbb{E}_{g^*}\left(\mathrm{tr}\left((\mathbf{X} - \mathbf{m}^*)(\mathbf{X} - \mathbf{m}^*)^\top \Sigma^{-1}\right)\right) \end{aligned}$$

en utilisant l'identité $a^\top b = \mathrm{tr}(ab^\top)$ pour deux vecteurs colonnes a et b de $\mathbb{R}^n$, où "tr" représente la trace d'une matrice carrée. Enfin, comme l'espérance et la trace commutent, par linéarité, et comme $\Sigma^* = \mathbb{E}_{g^*}\left((\mathbf{X} - \mathbf{m}^*)(\mathbf{X} - \mathbf{m}^*)^\top\right)$, on a :

$$D'(\Sigma) = \ln\det\Sigma + \mathrm{tr}(\Sigma^*\Sigma^{-1}). \tag{2.2}$$

Nous allons observer la dégradation de cette divergence de Kullback-Leibler "partielle" D' sur un exemple d'échantillonnage selon la loi "*banana shape*", ou loi "en forme de banane". C'est un cas-test classique d'échantillonnage utilisé pour tester des algorithmes d'AIS (voir par exemple [Cornuet et al., 2012], [Elvira et al., 2019], [Martino et al., 2017b]). Pour générer une variable aléatoire $\mathbf{X}$, à valeur dans $\mathbb{R}^n$, suivant la loi "*banana shape*", on tire une variable gaussienne de dimension n, $\mathbf{U} \sim \mathcal{N}(0, C)$, dont la matrice de covariance est $C = \mathrm{diag}(\sigma^2, 1, \ldots, 1)$ et on transforme la seconde coordonnée U_2 en $X_2 = U_2 - b(U_1^2 - \sigma^2)$, où σ et b sont des constantes réelles fixées. Un échantillon

de cette loi en dimension 2 est tracé en annexe A.2. On peut montrer aisément que la moyenne de cette variable $\mathbf{X}$ est nulle, sa covariance est donnée par la matrice $\mathrm{diag}(\sigma^2, 1 + 2b^2\sigma^4, 1, \ldots, 1)$ et sa densité, notée π, est égale à (voir [Cornuet et al., 2012]) :

$$\pi(x_1, x_2, \ldots, x_n) = g_{0,C}(x_1, x_2 + b(x_1^2 - \sigma^2), x_3, \ldots, x_n) \tag{2.3}$$

où on rappelle que $g_{0,C}$ est la densité de la loi $\mathcal{N}(0, C)$. La fonction π est donc la densité à approcher (autrement dit g^* est remplacée par π dans les calculs précédents, ce qui revient à chercher la densité optimale d'IS pour estimer la constante de normalisation de π : $\int \pi = 1$, voir section 1.4.1). Les valeurs des constantes sont fixées à $b = 0.03$ et $\sigma = 10$ dans les applications numériques, et les paramètres optimaux d'IS valent alors : $\mathbf{m}^* = \mathbf{0}$ pour la moyenne et $\Sigma^* = \mathrm{diag}(100, 19, 1, \ldots, 1)$ pour la covariance. Notons que la famille gaussienne n'est pas la plus efficace pour approcher la loi "*banana shape*", un mélange de plusieurs densités gaussiennes serait plus performant, mais une densité gaussienne donne déjà des résultats satisfaisants comme on le verra par la suite.

La figure 2.3 représente ainsi l'évolution de la divergence KL partielle D' en fonction de la dimension, lorsqu'on prend la matrice optimale Σ^* (cercles bleus), ou son estimation empirique $\hat{\Sigma}^*$ (carrés rouges), réalisée en générant un échantillon de loi π de taille $N = 200$.

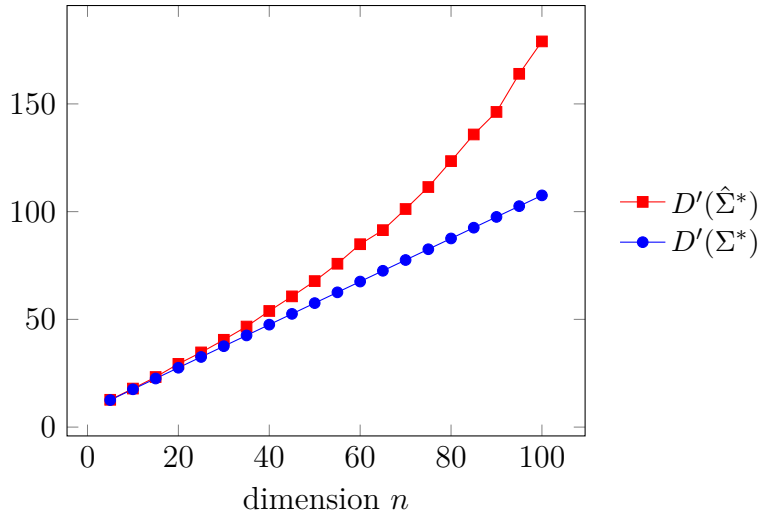


Figure 2.3 – Évolution de la divergence KL partielle D', entre la densité de la loi "*banana shape*" et la densité gaussienne $g_{\mathbf{m}^*,\Sigma}$, en fonction de la dimension, lorsque Σ est la matrice optimale Σ^* (cercles bleus), ou la matrice empirique $\hat{\Sigma}^*$ (carrés rouges).

La divergence augmente bien plus rapidement avec la dimension en prenant la matrice $\hat{\Sigma}^*$ au lieu de Σ^*, ce qui signifie que la densité auxiliaire $g_{\mathbf{m}^*,\hat{\Sigma}^*}$ est une moins bonne approximation de la densité cible π que $g_{\mathbf{m}^*,\Sigma^*}$. Cet accroissement est notamment dû aux erreurs d'estimation sur chaque coefficient de la matrice et sont d'autant plus nombreuses que la dimension est grande. Lorsqu'on veut estimer une intégrale, cela peut alors impliquer une estimation moins précise, puisqu'une grande divergence KL entraîne une grande variance de l'estimateur. Nous verrons effectivement dans les chapitres suivants l'impact d'une telle augmentation sur l'erreur d'estimation.

Dans cette thèse, nous nous concentrons sur l'estimation de la matrice de covariance en cherchant à réduire le nombre total de paramètres estimés dans cette matrice. Les idées développées pour y parvenir sont abordées dans les chapitres suivants. La section qui suit résume différentes stratégies utilisées dans la littérature pour éviter ou atténuer les problèmes rencontrés en grande dimension dans les algorithmes d'AIS.

2.2 Techniques permettant d'améliorer les algorithmes d'échantillonnage préférentiel en grande dimension

2.2.1 Transformation des poids pour estimer les paramètres d'échantillonnage préférentiel

Une première stratégie pour renforcer l'efficacité de l'échantillonnage préférentiel en grande dimension est de transformer les poids afin d'exclure les valeurs aberrantes et d'éviter qu'ils aient une trop grande variance. Ainsi, [Koblents and Míguez, 2015] proposent une amélioration de l'algorithme PMC (1.4.2), nommée *Nonlinear* PMC (NPMC), en remplaçant les poids (non normalisés) w_i par $\tilde{w}_i = \mathcal{T}(w_i)$, où $\mathcal{T} : \mathbb{R}_+ \to \mathbb{R}_+$ est une fonction (non linéaire), choisie de sorte à réduire leur variance. [El-Laham et al., 2018] modifient aussi les poids en particulier pour estimer la matrice de covariance dans un algorithme d'AIS ("*Covariance* AIS").

Deux types de transformations sont couramment utilisées : les fonctions de "*clipping*" visant à diminuer la valeur des poids les plus grands, et les fonctions de "*tempering*" qui atténuent les variations des w_i. Le "*clipping*" (aussi appelé "*Truncated Importance Sampling*", [Ionides, 2008]) consiste d'abord à classer les poids w_i dans l'ordre décroissant, $w_{(1)} \geq \cdots \geq w_{(N)}$, et à choisir une valeur seuil $w_{\max} = w_{(N_0)}$ ($N_0 < N$) qu'on ne pourra pas dépasser. La fonction $\mathcal{T}$ à appliquer, dépendant des poids, est alors définie par $\mathcal{T}(w_i) = \min(w_i, w_{max})$. Les N_0 plus grands poids prennent alors la valeur maximale $w_{\max}$. [Koblents and Míguez, 2015] suggèrent par exemple de modifier un dixième des poids (i.e. $N_0/N = 1/10$), ce choix donnant de bons résultats sur de nombreux exemples. [El-Laham et al., 2018] ajoutent qu'il faudrait de plus choisir N_0 supérieur à la dimension n, ce qui peut être contraignant si N n'est pas assez grand.

Les fonctions de "*tempering*" sont de la forme $\mathcal{T}(w_i) = (w_i)^{\alpha_t}$, où α_t est un réel compris entre 0 et 1, pouvant dépendre de l'itération t à laquelle les poids sont calculés. [Koblents and Míguez, 2015] conseillent en effet d'adapter α_t à chaque itération de façon à ce qu'il prenne de petites valeurs lors des premières itérations et qu'il se rapproche de 1 par la suite (par exemple $\alpha_t = (1 + e^{-t})^{-1}$). [El-Laham et al., 2018] proposent quant à eux d'adapter α_t à chaque itération en fonction de la "taille d'échantillon efficace" ou "effective" ("*effective sample size*", ESS, [Kong et al., 1994], [Martino et al., 2017a]). L'ESS donne une indication sur le nombre d'échantillons effectivement "utiles" dans l'estimation et est généralement définie par $\hat{\eta} = (\sum_{i=1}^N w_i)^2/(\sum_{i=1}^N w_i^2)$. Ainsi, si les N échantillons ont un poids équivalent ($w_i = 1/N$), alors $\hat{\eta} = N$, mais si tous les poids sont nuls sauf un alors $\hat{\eta} = 1$. [El-Laham et al., 2018] suggèrent de choisir le coefficient α_t afin que $\hat{\eta}$ soit proche d'un entier N_0 préalablement fixé.

Cependant toutes ces méthodes ne restent efficaces qu'en dimension modérément grande (de la dizaine à quelques dizaines) et souffrent encore de la dégénérescence des poids et de la mauvaise estimation des paramètres lorsque la dimension est trop élevée.

Une dernière stratégie que l'on peut évoquer, est appliquée dans [Chan and Kroese, 2012], et consiste à générer directement un échantillon selon la loi optimale d'IS, g^*, à l'aide d'une méthode MCMC, afin d'estimer les paramètres optimaux sans rapports de vraisemblance. La dégénérescence des poids est donc évitée dans l'estimation des paramètres, mais ceux-ci sont toujours estimés en grande dimension, ce qui peut entrainer une estimation imprécise de l'espérance. De plus, il peut être difficile de trouver un noyau de transition performant dans un algorithme MCMC, celui-ci pouvant aussi être couteux en budget et en temps de calcul lorsque la dimension est élevée.

2.2.2 Méthodes de contraction des paramètres pour estimer une matrice de covariance de grande taille

[El-Laham et al., 2019] ont développé un algorithme d'AIS qui combine les techniques de transformations des poids avec des méthodes de "contraction" ("*shrinkage*") de la matrice de covariance afin d'améliorer la performance en grande dimension. Les techniques de contraction des paramètres consistent à calculer une combinaison de plusieurs estimateurs. Par exemple, dans l'article [El-Laham et al., 2019], la matrice de covariance à l'étape t de l'algorithme d'AIS, est estimée par :

$$\Sigma_t = (1-\beta_t)\Sigma_{t-1} + \beta_t(1-\eta_t)\hat{\Sigma}_t + \beta_t\eta_t\tilde{\Sigma}_t,$$

où Σ_{t-1} est la matrice de l'étape précédente, $\hat{\Sigma}_t$ est la matrice empirique estimée à l'étape t, $\tilde{\Sigma}_t$ est une estimation de la covariance avec les poids transformés (avec la méthode de *clipping* ou de *tempering*, suggérées dans [El-Laham et al., 2018] et évoquées dans la section précédente), et β_t, η_t sont des coefficients (entre 0 et 1) adaptés à chaque étape de l'algorithme. Cet ajustement permet aux auteurs d'améliorer la performance de l'AIS classique sur des exemples de dimension modérée (≈ 10) mais aucun résultat n'existe pour des dimensions de plusieurs dizaines ou centaines.

Plus généralement et indépendamment du cadre de l'échantillonnage préférentiel, les méthodes de contraction sont utilisées dans la littérature pour construire un estimateur de la covariance plus robuste et plus précis que l'estimateur empirique lorsque la dimension n est supérieure ou du même ordre que la taille d'échantillon N. Ces approches considèrent des estimateurs de la forme $(1-\mu)u(\Sigma)+\mu I_n$, avec $u(\Sigma)$ un estimateur de la matrice de covariance Σ d'un vecteur centré $\mathbf{X}$ et $\mu \in [0,1]$ un réel. Elles reposent sur le choix de l'estimateur $u(\Sigma)$ et surtout sur l'optimisation du coefficient μ. Par exemple, [Ledoit and Wolf, 2004] proposent un choix optimal de μ en considérant l'estimateur empirique de la matrice de covariance, c'est-à-dire :

$$\tilde{\Sigma} = (1-\mu)\hat{\Sigma} + \mu I_n,$$

où $\hat{\Sigma} = \sum_{i=1}^{N} \mathbf{X}_i\mathbf{X}_i^\top/N$ est l'estimateur empirique de la covariance de $\mathbf{X}$ et $\mu \in [0,1]$. Les auteurs donnent le choix optimal du paramètre μ en minimisant l'erreur quadratique moyenne entre $\tilde{\Sigma}$ et Σ. L'estimateur $\tilde{\Sigma}$ améliore significativement l'estimation de Σ par rapport à $\hat{\Sigma}$.

D'autres travaux proposent de considérer l'estimateur de Tyler [Chen et al., 2011], à la place de l'estimateur empirique, pour des distributions elliptiques, c'est-à-dire :

$$u(\Sigma) = \frac{\mathrm{tr}(\Sigma)}{N}\sum_{i=1}^{N}\frac{\mathbf{X}_i\mathbf{X}_i^\top}{\mathbf{X}_i^\top\Sigma^{-1}\mathbf{X}_i},$$

où $\mathbf{X}_i = \tilde{\mathbf{X}}_i/\|\tilde{\mathbf{X}}_i\|$, et $\tilde{\mathbf{X}}_i$ sont des échantillons i.i.d. centrés suivant une distribution elliptique dont la covariance est la matrice Σ que l'on cherche à estimer. L'estimateur de Tyler permet d'atteindre la matrice Σ de manière itérative par un algorithme de point fixe. Un choix optimal du coefficient μ est également obtenu en minimisant l'erreur quadratique moyenne. [Chen et al., 2011] montrent sur plusieurs cas-tests que leur estimateur $\tilde{\Sigma}$ est plus efficace que celui de [Ledoit and Wolf, 2004]. Une généralisation de ces travaux à d'autres estimateurs est proposée dans [Ashurbekova et al., 2020].

Ces approches permettent d'améliorer la précision de l'estimation des matrices de covariance en grande dimension, mais sont difficilement applicables, voire inefficaces dans le contexte de l'échantillonnage préférentiel car elles ne permettent pas d'éviter la dégénérescence des poids d'importance.

2.2.3 Réduction du nombre de paramètres à estimer

Les exemples de la section 2.1.3 montrent que plus le nombre de paramètres à estimer est grand et plus l'estimation finale de l'espérance est dégradée. C'est pourquoi plusieurs articles proposent de réduire le nombre de coefficients estimés dans les algorithmes d'AIS, en particulier dans l'algorithme CE.

2.2.3.1 Sélection des variables par la méthode de "*Screening*"

[Rubinstein and Glynn, 2009] ont mis en place une procédure de sélection de variables ("*screening method*") combinée avec la CE, afin de construire l'algorithme CE-SCR (*CE-Screening algorithm*). Pour cela ils considèrent le problème d'estimation de l'espérance $E = \mathbb{E}_f(\mathbb{I}_{\{\varphi(\mathbf{X})\geq 0\}})$, avec f une densité de probabilité de $\mathbb{R}^n$ appartenant à la famille exponentielle, paramétrée par le vecteur $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_n) \in \mathbb{R}^n$ et dont les composantes sont indépendantes, c'est à dire s'écrivant sous la forme $f(\mathbf{x}) = g(\mathbf{x}; \boldsymbol{\nu}) = \prod_{j=1}^n g_j(x_j; \nu_j)$ (par exemple f peut être la densité gaussienne paramétrée par la moyenne et de covariance fixée à l'identité). Les rapports de vraisemblance s'écrivent alors

$$L(\mathbf{x}) = \frac{g(\mathbf{x}, \boldsymbol{\nu})}{g(\mathbf{x}, \boldsymbol{\theta})} = \prod_{j=1}^n \frac{g_j(x_j, \nu_j)}{g_j(x_j, \theta_j)},$$

où les densités auxiliaires sont supposées appartenir à la même famille que la densité initiale et où $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n) \in \mathbb{R}^n$. De plus, la fonction φ est supposée croissante en chacune de ses variables de sorte que le paramètre optimal $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^n} D(g^*, g(\cdot; \boldsymbol{\theta}))$ soit tel que pour tout $j = 1, \ldots, n$, on ait $\theta_j^* \geq \nu_j$. L'idée principale de la méthode de "*screening*" est alors d'identifier et de sélectionner les paramètres importants, c'est-à-dire ceux qui vont le plus contribuer à minimiser la divergence de Kullback-Leibler. Pour ce faire, l'ensemble des paramètres est d'abord séparé en deux sous-ensembles $\boldsymbol{\theta} = (\boldsymbol{\theta}^{(b)}, \boldsymbol{\theta}^{(nb)})$, un ensemble restreint de paramètres influents $\boldsymbol{\theta}^{(b)}$ (appelés "*bottleneck parameters*" dans l'article) et l'ensemble de tous les autres paramètres $\boldsymbol{\theta}^{(nb)}$ ("*nonbottleneck*") considérés comme non influents. L'objectif est ensuite de ne mettre à jour que les paramètres $\boldsymbol{\theta}^{(b)}$ et garder $\boldsymbol{\theta}^{(nb)} = \boldsymbol{\nu}^{(nb)}$ afin que les rapports de vraisemblance ne soient plus que le produit d'un faible nombre de facteurs. Pour sélectionner les paramètres, les auteurs suggèrent de calculer

$$\Delta_j = \frac{\hat{\theta}_{0j} - \nu_j}{\nu_j}, \text{ pour tout } j = 1, \ldots, n$$

où $\hat{\boldsymbol{\theta}}_0$ est le paramètre estimé après une première itération de l'algorithme CE (si certains ν_j sont nuls, on peut prendre $\boldsymbol{\Delta} = \hat{\boldsymbol{\theta}}_0 - \boldsymbol{\nu}$). Si Δ_j est inférieur à une constante Δ^* préalablement choisie (par exemple $\Delta^* = 0.1$), alors on pose $\hat{\theta}_{0j} = \nu_j$, sinon $\hat{\theta}_{0j}$ est considéré comme un élément influent et on construit ainsi l'ensemble des paramètres influents $\boldsymbol{\theta}^{(b)}$, ce procédé pouvant être répété plusieurs fois. Une fois l'ensemble des éléments influents identifiés, on effectue la suite de l'algorithme CE en ne mettant à jour que les paramètres de cet ensemble.

L'avantage de CE-SCR est, d'une part, que l'on peut espérer estimer un petit nombre, disons $r < n$, de paramètres (sélectionnés dans la phase de *screening*), et d'autre part, que les poids sont alors égaux au produit de seulement r facteurs, ce qui peut permettre d'éviter la dégénérescence. Néanmoins, cette méthode est inefficace dans les cas où toutes ou une grande partie des variables sont influentes (par exemple avec la fonction somme des coordonnées φ_1 (2.1)). De plus, elle ne permet pas de mettre à jour la matrice de covariance (notamment dans le cadre gaussien) et supposer φ croissante en chacune de ses variables peut être contraignant et peut restreindre le nombre de cas-tests sur lesquels CE-SCR est efficace.

2.2.3.2 Utilisation de densités auxiliaires plus efficaces en grande dimension

Une autre stratégie pour améliorer les performances de l'échantillonnage préférentiel en grande dimension est proposée par [Wang and Song, 2016]. Les auteurs suggèrent d'utiliser les densités de von Mises-Fisher comme famille auxiliaire dans l'algorithme CE, à la place des densités gaussiennes qui sont moins efficaces en grande dimension pour détecter le domaine de défaillance. En effet, ils considèrent le problème d'estimation d'une probabilité de défaillance dans l'espace normal standard de dimension n (i.e. f est la densité $\mathcal{N}(\mathbf{0}, I_n)$). En se basant sur les résultats de l'article [Katafygiotis and Zuev, 2008], ils utilisent le fait que la norme d'un échantillon gaussien standard se rapproche de la loi $\mathcal{N}(\sqrt{n}, 1/2)$ quand n grandit, ce qui signifie que la majorité des échantillons se situent autour de l'hypersphère de dimension $n-1$ et de rayon $\sqrt{n}$ (dans un "anneau d'importance" ou "*importance ring*"). Pour générer un vecteur gaussien en grande dimension, sa norme étant d'environ $\sqrt{n}$, il suffit de connaitre sa direction, qui peut être générée par la densité de von Mises-Fisher (vMF) :

$$g_{\text{vMF}}(\bar{\mathbf{x}}; \boldsymbol{\mu}, \kappa) = c_n(\kappa) \exp(\kappa \boldsymbol{\mu}^\top \bar{\mathbf{x}}),$$

où $\bar{\mathbf{x}}$ appartient à l'hypersphère, $\mathbb{S}^{n-1} \subset \mathbb{R}^n$, de dimension $n-1$ et de rayon 1, $\boldsymbol{\mu} \in \mathbb{R}^n$ est la direction moyenne ($\|\boldsymbol{\mu}\| = 1$) et $\kappa > 0$ est un paramètre de concentration autour de μ (plus κ est grand, plus les échantillons sont concentrés). Le coefficient $c_n(\kappa)$ est une constante de normalisation dépendant de κ et n. La mise à jour des paramètres de la densité vMF nécessite uniquement l'estimation de n coefficients, contre $n(n+3)/2$ pour la loi gaussienne. Une fois qu'on dispose d'un vecteur $\bar{\mathbf{X}}$ selon la loi vMF, [Wang and Song, 2016] suggèrent de tirer une variable ξ de loi $\chi(n)$, à n degrés de liberté (qui est la loi de la norme d'un vecteur gaussien). Le vecteur $\mathbf{X} = \xi\bar{\mathbf{X}}$ s'approche alors d'un vecteur gaussien. Dans le cas multimodal, [Wang and Song, 2016] proposent d'utiliser un mélange de lois de von Mises-Fisher (vMFM) et testent ensuite un algorithme adaptatif, CE-AIS-vMFM, pour estimer des probabilités de défaillance. Cette méthode donne des résultats précis pour des dimensions supérieures à 100, mais nécessite un budget de simulation assez élevé et n'est pas efficace en petite dimension. Par ailleurs, il n'existe pas de formule analytique pour mettre à jour le paramètre κ, et il faut donc résoudre un problème d'optimisation à l'aide de méthodes numériques pour l'estimer ou bien utiliser une approximation.

Une autre version de cet algorithme a été développée dans [Papaioannou et al., 2019a], afin d'améliorer ses performances en petite dimension. En effet, approcher la norme d'un vecteur gaussien par une loi du $\chi(n)$ est pertinent pour n suffisamment grand. Les auteurs suggèrent alors de remplacer la loi du $\chi(n)$ par la loi de Nakagami de densité :

$$g_N(x; p, \omega) = \frac{2p^p}{\Gamma(p)\omega^p} x^{2p-1} \exp\left(-\frac{p}{\omega}x^2\right),$$

pour tout $x > 0$, et où $p \geq 0.5$ est un paramètre de forme, $\omega > 0$ un paramètre de propagation, et Γ est la fonction Gamma. Cette modification améliore l'algorithme pour des dimensions petites ou modérées, avec un nombre de paramètres estimés équivalents ($n+2$ au lieu de n dans [Wang and Song, 2016]).

Ainsi, pour améliorer significativement l'échantillonnage préférentiel en grande dimension, plusieurs articles utilisent des techniques réduisant le nombre de paramètres estimés, soit en sélectionnant des variables, soit en modifiant les densités auxiliaires d'échantillonnage. Alors que les méthodes suggérant de transformer les poids semblent limitées à des dimensions de quelques dizaines, les approches qui réduisent la dimension des paramètres permettent de réaliser des estimations assez précises dans des dimensions supérieures. Cependant, CE-SCR n'est efficace que

sur un nombre restreint de cas-tests, et les techniques basées sur la famille de densités de von Mises-Fisher peuvent nécessiter un grand budget de simulation.

Une autre approche pour réduire la dimension des paramètres est d'utiliser une projection dans un sous-espace de petite dimension. C'est la méthode employée dans [Uribe et al., 2021] pour améliorer l'algorithme CE en grande dimension, en construisant un sous-espace identifiant la structure de petite dimension du problème. Nous détaillons l'algorithme (iCEred, "*improved Cross-Entropy method with failure-informed dimension reduction*") présenté dans cet article, et la projection utilisée (issue de [Zahm et al., 2018]) dans les chapitres suivants, dans lesquels nous nous concentrons sur le couplage des techniques de projection avec l'échantillonnage préférentiel. En particulier, l'objectif des prochains chapitres est d'utiliser une projection afin de réduire le nombre de paramètres estimés dans la matrice de covariance qui contribue majoritairement à la dégradation de l'IS.

Chapitre 3

Estimation des paramètres en petite dimension à l'aide d'une projection

Sommaire

Pour éviter la dégradation de l'estimation par échantillonnage préférentiel, plusieurs approches décrites dans le chapitre 2 ont suggéré de réduire le nombre total de paramètres à estimer afin de diminuer le nombre d'erreurs d'estimation. Cependant, les méthodes développées connaissent des difficultés pour effectivement réduire le nombre de paramètres ou demandent un budget de

simulation élevé. Plus récemment, [Zahm et al., 2018] ont proposé une projection dans un sous-espace de petite dimension, pour la résolution de problèmes inverses bayésiens, qui a été reprise dans [Uribe et al., 2021] pour améliorer l'algorithme CE (1). Cette technique donne des résultats d'une grande précision et c'est pourquoi, dans ce chapitre, nous proposons d'étudier l'influence de la projection des paramètres dans un sous-espace de petite dimension. Nous commençons par analyser l'effet d'une projection sur la divergence de Kullback-Leibler, dans un cadre simple. Nous montrerons ensuite sur des simulations qu'utiliser une projection, même naïve ou non optimale, permet souvent d'améliorer les résultats d'estimation d'une espérance. Enfin, nous présenterons la projection proposée dans [Uribe et al., 2021] et [Zahm et al., 2018], afin d'évaluer son efficacité et ses limites sur quelques exemples numériques.

3.1 Étude de l'influence d'une projection sur la précision de l'estimation

3.1.1 Réduction potentielle de la divergence de Kullback-Leibler à l'aide d'une projection

La divergence de Kullback-Leibler est liée à l'erreur d'estimation, comme évoqué dans le chapitre 1 (section 1.2.4), et s'assurer que la divergence ne prend pas de grandes valeurs peut permettre d'améliorer la précision de l'estimation. Nous proposons de montrer dans un cas très simple (cas gaussien avec covariance fixée à l'identité) comment une projection des paramètres dans un sous-espace de petite dimension peut éviter une forte augmentation de la divergence KL.

On souhaite approcher la densité optimale d'échantillonnage préférentiel $g^* = \dfrac{\phi f}{E}$ (1.5) pour estimer E, par une densité gaussienne $g_{\mathbf{m}}(\mathbf{x}) = g_{\mathbf{m},I_n}(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}} \exp\left(-\|\mathbf{x}-\mathbf{m}\|^2/2\right)$ (en rappelant également que $f = g_{\mathbf{0},I_n}$). La divergence KL (à minimiser) entre ces deux densités s'écrit :

$$\begin{aligned}
D(g^*, g_{\mathbf{m}}) &= \mathbb{E}_{g^*}\left(\ln\left(\frac{g^*(\mathbf{X})}{g_{\mathbf{m}}(\mathbf{X})}\right)\right) \\
&= \mathbb{E}_{g^*}\left(\ln\left(\frac{\phi(\mathbf{X})\exp\left(-\|\mathbf{X}\|^2/2\right)}{E\exp\left(-\|\mathbf{X}-\mathbf{m}\|^2/2\right)}\right)\right) \\
&= \frac{1}{2}\mathbb{E}_{g^*}(\|\mathbf{X}-\mathbf{m}\|^2) - \frac{1}{2}\mathbb{E}_{g^*}(\|\mathbf{X}\|^2) + \mathbb{E}_{g^*}(\ln(\phi(\mathbf{X}))) - \ln(E).
\end{aligned}$$

Après avoir développé le terme au carré, et rappelé que $\mathbf{m}^* = \mathbb{E}_{g^*}(\mathbf{X})$, on obtient :

$$D(g^*, g_{\mathbf{m}}) = \frac{1}{2}\|\mathbf{m}^* - \mathbf{m}\|^2 + D^*, \tag{3.1}$$

où on a posé $D^* = -\dfrac{1}{2}\|\mathbf{m}^*\|^2 + \mathbb{E}_{g^*}(\ln(\phi(\mathbf{X}))) - \ln(E)$, la divergence minimale (pour $\mathbf{m} = \mathbf{m}^*$).

En pratique, $\mathbf{m}^*$ est approché par $\hat{\mathbf{m}}^*$ (1.11) et :

$$D(g^*, g_{\hat{\mathbf{m}}^*}) = \frac{1}{2N}\|\boldsymbol{\varepsilon}\|^2 + D^*,$$

avec $\boldsymbol{\varepsilon} = (\hat{\mathbf{m}}^* - \mathbf{m}^*)\sqrt{N}$ un vecteur aléatoire de $\mathbb{R}^n$ représentant l'erreur d'estimation de $\mathbf{m}^*$, et N la taille de l'échantillon utilisée pour estimer $\hat{\mathbf{m}}^*$.

En gardant N fixe, l'augmentation de la dimension peut entrainer celle de l'erreur $\|\boldsymbol{\varepsilon}\|$ et donc de la divergence KL. En effet, comme $\|\boldsymbol{\varepsilon}\|^2 = \sum_{j=1}^n \varepsilon_j^2$, cette somme de termes positifs risque d'augmenter si n grandit. Regardons alors comment une projection pourrait nous aider à réduire cette erreur. Considérons un sous-espace $\mathcal{Y} \subset \mathbb{R}^n$ de petite dimension, dans lequel on cherche à minimiser la divergence KL, ou de manière équivalente $\|\mathbf{m} - \mathbf{m}^*\|$:

$$\mathbf{m}^*_{\mathcal{Y}} = \underset{\mathbf{m}\in\mathcal{Y}}{\arg\min}\ D(g^*, g_{\mathbf{m}}) = \underset{\mathbf{m}\in\mathcal{Y}}{\arg\min}\ \|\mathbf{m} - \mathbf{m}^*\|.$$

Lorsque $\mathcal{Y} = \mathbb{R}^n$, on a bien $\mathbf{m}^*_{\mathbb{R}^n} = \mathbf{m}^*$. Sinon, le théorème de projection dans un sous-espace de $\mathbb{R}^n$ donne : $\mathbf{m}^*_{\mathcal{Y}} = \Pi_{\mathcal{Y}}(\mathbf{m}^*)$, avec $\Pi_{\mathcal{Y}} : \mathbb{R}^n \to \mathbb{R}^n$ la projection ortogonale sur $\mathcal{Y}$. En pratique, comme $\mathbf{m}^*$ est inconnu, on utilise une estimation de $\mathbf{m}^*_{\mathcal{Y}}$: $\hat{\mathbf{m}}_{\mathcal{Y}} = \Pi_{\mathcal{Y}}(\hat{\mathbf{m}}^*)$, où $\hat{\mathbf{m}}^*$ est l'estimation de $\mathbf{m}^*$. Notons alors que pour $\mathbf{m} \in \mathcal{Y}$, $\mathbf{m} - \Pi_{\mathcal{Y}}(\mathbf{m}^*)$ est orthogonal à $\mathbf{m}^* - \Pi_{\mathcal{Y}}(\mathbf{m}^*)$ et donc :

$$\|\mathbf{m} - \mathbf{m}^*\|^2 = \|\mathbf{m} - \Pi_{\mathcal{Y}}(\mathbf{m}^*)\|^2 + \|\Pi_{\mathcal{Y}}(\mathbf{m}^*) - \mathbf{m}^*\|^2 = \|\mathbf{m} - \Pi_{\mathcal{Y}}(\mathbf{m}^*)\|^2 + \|\mathbf{m}^*\|^2 - \|\Pi_{\mathcal{Y}}(\mathbf{m}^*)\|^2.$$

Comme $\hat{\mathbf{m}}_{\mathcal{Y}} = \Pi_{\mathcal{Y}}(\hat{\mathbf{m}}^*) \in \mathcal{Y}$, et en utilisant (3.1), on déduit :

$$D(g^*, g_{\hat{\mathbf{m}}_{\mathcal{Y}}}) = \frac{1}{2}\|\Pi_{\mathcal{Y}}(\hat{\mathbf{m}}^* - \mathbf{m}^*)\|^2 - \frac{1}{2}\|\Pi_{\mathcal{Y}}(\mathbf{m}^*)\|^2 + \mathbb{E}_{g^*}(\ln(\phi(\mathbf{X}))) - \ln(E). \tag{3.2}$$

La différence entre les deux divergences donne enfin :

$$2D(g^*, g_{\hat{\mathbf{m}}_{\mathcal{Y}}}) - 2D(g^*, g_{\hat{\mathbf{m}}^*}) = \|\mathbf{m}^*\|^2 - \|\Pi_{\mathcal{Y}}(\mathbf{m}^*)\|^2 - \frac{1}{N}\left(\|\boldsymbol{\varepsilon}\|^2 - \|\Pi_{\mathcal{Y}}(\boldsymbol{\varepsilon})\|^2\right). \tag{3.3}$$

Comme $\|\boldsymbol{\varepsilon}\|^2 \geq \|\Pi_{\mathcal{Y}}(\boldsymbol{\varepsilon})\|^2$, il semble donc possible de rendre la quantité (3.3) négative en faisant en sorte que la différence $\|\mathbf{m}^*\|^2 - \|\Pi_{\mathcal{Y}}(\mathbf{m}^*)\|^2$ (qui est positive) ne soit pas trop grande. Autrement dit, si la projection choisie permet d'avoir d'une part $\|\Pi_{\mathcal{Y}}(\boldsymbol{\varepsilon})\|^2$ très petit devant $\|\boldsymbol{\varepsilon}\|^2$ et, d'autre part, $\|\mathbf{m}^*\|^2 \approx \|\Pi_{\mathcal{Y}}(\mathbf{m}^*)\|^2$, on peut alors espérer diminuer la divergence KL en projetant les paramètres. Dans la suite, nous allons reprendre cette idée de projection des paramètres pour tenter de diminuer la divergence de Kullback-Leibler et améliorer la précision de l'estimation.

3.1.2 Approximation de la matrice de covariance optimale

L'estimation des paramètres gaussiens $\mathbf{m}^*$ et Σ^* (1.10) en grande dimension induit de nombreuses erreurs qui dégradent l'estimation finale. C'est particulièrement le cas pour la matrice de covariance dont le nombre de coefficients à estimer $(n(n+1)/2)$ croît de manière quadratique avec la dimension n. Nous proposons donc de nous concentrer sur cette matrice et de n'en estimer qu'un petit nombre de coefficients.

Pour cela, on suggère d'estimer la matrice :

$$\Sigma_k = \sum_{i=1}^{k}(v_i - 1)\mathbf{d}_i\mathbf{d}_i^\top + I_n \tag{3.4}$$

où les $\mathbf{d}_i$ sont k vecteurs orthonormés, indiquant des directions influentes pour l'estimation de Σ^*, et $v_i > 0$ est la variance dans la direction de $\mathbf{d}_i$, c'est-à-dire $v_i = \mathbf{d}_i^\top \Sigma_k \mathbf{d}_i$. Ainsi, en complétant la famille $(\mathbf{d}_1, \ldots, \mathbf{d}_k)$ en une base orthonormée avec $(\mathbf{d}_{k+1}, \ldots, \mathbf{d}_n)$, il est facile de vérifier que Σ_k est la matrice de covariance du vecteur gaussien :

$$\mathbf{Y} = \sum_{i=1}^{k} v_i^{1/2} Y_i \mathbf{d}_i + \sum_{i=k+1}^{n} Y_i \mathbf{d}_i$$

où les Y_i sont des variables gaussiennes standards ($\mathcal{N}(0,1)$) indépendantes. On peut également montrer que Σ_k s'écrit sous la forme :

$$\Sigma_k = (R, R_\perp) \begin{pmatrix} V_k & 0 \\ 0 & I_{n-k} \end{pmatrix} \begin{pmatrix} R^\top \\ R_\perp^\top \end{pmatrix} \tag{3.5}$$

où $V_k = \mathrm{diag}(v_1, \ldots, v_k)$ est la matrice diagonale de taille k que l'on cherche à estimer, $R^\top = (\mathbf{d}_1, \ldots, \mathbf{d}_k)^\top \in \mathbb{R}^{k\times n}$ représente la matrice de projection dans un sous-espace influent de dimension k, et $R_\perp^\top = (\mathbf{d}_{k+1}, \ldots, \mathbf{d}_n)^\top$ la matrice de projection dans le sous-espace de dimension $n-k$ engendré par les directions supposées non influentes pour estimer Σ^*. Le but est donc d'estimer les k paramètres de variances $v_1, \ldots, v_k$ au lieu des $n(n+1)/2$ de la matrice $\hat{\Sigma}^*$ (1.11), et ainsi réduire le nombre d'erreurs d'estimation sans augmenter le budget. Nous verrons dans la prochaine section qu'un choix "naïf" des directions de projection $\mathbf{d}_i$ permet déjà de réduire significativement l'erreur d'estimation de E dans la plupart des cas.

3.1.3 Simulations numériques

Pour justifier l'utilisation de la matrice "projetée" Σ_k plutôt que la matrice empirique $\hat{\Sigma}^*$, commençons par tester la précision de l'estimation lorsqu'on choisit les directions de projection $\mathbf{d}_i$ de manière naïve. Pour cela nous allons d'abord tirer ces directions aléatoirement. Ensuite, nous testons la projection sur les directions canoniques, ce qui revient à modéliser Σ_k comme une matrice diagonale. Nous allons observer numériquement comment l'estimation de l'espérance E par échantillonnage préférentiel est affectée par ces deux choix de directions. Pour cela, on propose de suivre l'algorithme 6, où tous les $\mathbf{d}_i$, $i = 1 \ldots n$, sont fixés préalablement (et donc $k = n$).

Algorithme 6 : Estimation de E par IS gaussien où la matrice de covariance est de la forme Σ_k (3.4) avec $k = n$ et les $\mathbf{d}_i$ sont fixés à l'avance

Données : Tailles des échantillons N et M, vecteurs $\mathbf{d}_1, \ldots, \mathbf{d}_n$
Résultat : Estimation $\hat{E}_N$ de l'integrale E

1. Générer un échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_M$ selon g^* pour estimer $\mathbf{m}^*$ et Σ^* par $\hat{\mathbf{m}}^*$ et $\hat{\Sigma}^*$ (1.11) ;
2. Calculer les $\hat{v}_i = \mathbf{d}_i^\top \hat{\Sigma}^* \mathbf{d}_i$, pour $i = 1 \ldots n$, et la matrice $\hat{\Sigma}_k$ définie en (3.4), avec $v_i = \hat{v}_i$. ;
3. Générer un nouvel échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_N$ selon $g_{\hat{\mathbf{m}}^*, \hat{\Sigma}_k}$ et estimer E par $\hat{E}_N$ (1.4).

Pour réaliser les simulations selon ce procédé, on suppose que l'on sait générer un échantillon selon la loi optimale g^*, le but étant d'estimer les paramètres $\mathbf{m}^*$ et Σ^* directement (par les formules (1.11)) en évitant d'éventuels problèmes de convergence d'un algorithme adaptatif et l'influence des poids d'importance. En effet, un échantillon tiré selon g^* est généralement obtenu de manière itérative par des algorithmes d'AIS (comme ceux décrits dans le chapitre 1) ou par des méthodes MCMC (voir par exemple [Chan and Kroese, 2012], [Grace et al., 2014]). Dans les exemples de fiabilité, les échantillons selon g^* sont générés par une simple méthode de rejet, avec un budget de simulation potentiellement grand, mais le but premier étant d'étudier l'efficacité des projections sur les $\mathbf{d}_i$, on ne prendra pas en compte ce budget. Dans les autres exemples considérés, g^* correspond à une loi simple avec laquelle on sait échantillonner directement.

Pour générer les vecteurs $\mathbf{d}_i$ aléatoirement, on tire n vecteurs uniformément sur $[-1;1]^n$, et on orthonormalise la famille, (en utilisant par exemple le procédé d'orthonormalisation de Gram-Schmidt) pour en faire une base orthonormée de $\mathbb{R}^n$. D'autre part, pour projeter dans les directions

canoniques, on prendra $\mathbf{d}_i = \mathbf{e}_i$, pour $i = 1 \dots n$, où les $\mathbf{e}_i$ sont les vecteurs de la base canonique de $\mathbb{R}^n$ (avec tous les coefficients nuls sauf le i-ème qui est égal à 1), ce qui revient à estimer uniquement les coefficients diagonaux de la matrice Σ^*. On compare ensuite l'estimation $\hat{E}_N$ obtenue par échantillonnage préférentiel avec $g_{\hat{\mathbf{m}}^*,\Sigma}$ comme densité auxiliaire, et la divergence de Kullback-Leibler entre g^* et $g_{\mathbf{m}^*,\Sigma}$, où Σ est une des matrices suivantes :

- Σ^* la matrice optimale théorique, calculée de manière exacte lorsqu'elle est connue, ou estimée par Monte-Carlo avec un budget de simulation très important.
- $\hat{\Sigma}^*$ (1.11) l'estimation empirique de Σ^* obtenue avec un échantillon selon g^* de taille M.
- Σ^*_{rand} la matrice de la forme Σ_k (3.4), obtenue en projetant Σ^* dans n directions $\mathbf{d}_i$ aléatoires, autrement dit avec $v_i = \mathbf{d}_i^\top \Sigma^* \mathbf{d}_i$, pour $i = 1...n$.
- $\hat{\Sigma}^*_{\text{rand}}$ la matrice de la forme Σ_k (3.4), obtenue en projetant $\hat{\Sigma}^*$ dans n directions $\mathbf{d}_i$ aléatoires, autrement dit avec $v_i = \mathbf{d}_i^\top \hat{\Sigma}^* \mathbf{d}_i$, pour $i = 1...n$.
- Σ^*_{diag} la matrice diagonale issue de Σ^*, autrement dit la matrice Σ_k avec $v_i = \mathbf{e}_i^\top \Sigma^* \mathbf{e}_i$, pour $i = 1...n$.
- $\hat{\Sigma}^*_{\text{diag}}$ la matrice diagonale issue de $\hat{\Sigma}^*$, autrement dit la matrice Σ_k avec $v_i = \mathbf{e}_i^\top \hat{\Sigma}^* \mathbf{e}_i$, pour $i = 1...n$.

La matrice empirique $\hat{\Sigma}^*$ correspond à la situation que l'on cherche à améliorer pour s'approcher du cas (gaussien) optimal donné par Σ^*. Les matrices Σ^*_{rand} et Σ^*_{diag} servent à tester la qualité des projections. En effet, si la projection était optimale on aurait des résultats identiques à ceux de la matrice Σ^*. Enfin, les matrices $\hat{\Sigma}^*_{\text{rand}}$ et $\hat{\Sigma}^*_{\text{diag}}$ permettent d'évaluer l'éventuelle amélioration apportée par les projections comparé à $\hat{\Sigma}^*$.

Les simulations sont réalisées sur 4 cas-tests, et les résultats sont regroupés dans des tableaux où l'on fait apparaître la valeur moyenne de la divergence KL partielle (D' définie en (2.2)), l'erreur relative par rapport à la valeur optimale ($D'(\Sigma^*)$), l'estimation moyenne de $\hat{E}_N$, et le coefficient de variation correspondant, et ce pour différentes dimensions. La valeur de la divergence KL indiquée dans les tableaux ($D'(\Sigma)$) est une valeur moyenne sur 50 répétitions indépendantes de l'algorithme 6, où les tailles d'échantillons sont fixées à $M = 500$, et $N = 2000$, sauf mention contraire. L'erreur relative de la divergence KL, pour une matrice Σ, est donnée par :

$$\frac{D'(\Sigma) - D'(\Sigma^*)}{D'(\Sigma^*)}, \tag{3.6}$$

qui est une quantité positive car $D'(\Sigma^*)$ est la divergence KL partielle minimale. L'estimation moyenne et le coefficient de variation sont définis respectivement par :

$$\frac{1}{50}\sum_{i=1}^{50} \hat{E}_N^{(i)} \quad \text{et} \quad \frac{1}{E}\sqrt{\frac{1}{50}\sum_{i=1}^{50}\left(\hat{E}_N^{(i)} - E\right)^2}, \tag{3.7}$$

où $\hat{E}_N^{(1)}, \dots, \hat{E}_N^{(50)}$ sont 50 estimations indépendantes de E, dont la valeur exacte est estimée par Monte-Carlo avec un budget très important lorsqu'elle n'est pas connue théoriquement. De plus, les vecteurs choisis aléatoirement sont différents à chaque répétition. Les 4 exemples choisis sont deux exemples jouets d'estimation de probabilité d'événement rare (où $\phi = \mathbb{I}_{\{\varphi(\cdot)\geq 0\}}$), et deux

exemples jouets où l'on cherche à échantillonner selon une loi cible pour estimer une espérance (ici la constante de normalisation de la densité).

Notons enfin que les deux méthodes de projection ont déjà été utilisées dans la littérature pour diminuer le nombre de paramètres. Estimer uniquement la diagonale de Σ^* est suggéré par exemple par [Bourinet, 2018] dans l'algorithme d'entropie croisée. D'autre part, l'article [Wang et al., 2013] propose de générer une matrice de projection aléatoire pour la résolution de problèmes en optimisation bayésienne (méthode REMBO : *Random EMbeddings for Bayesian Optimisation*).

3.1.3.1 Exemple dans le cas événement rare : somme de variables indépendantes

Pour commencer, reprenons la fonction φ_1 définie en 2.1 (section 2.1.1) par

$$\varphi_1 : \mathbf{x} = (x_1, \dots, x_n) \in \mathbb{R}^n \mapsto \sum_{j=1}^{n} x_j - 3\sqrt{n}.$$

On rappelle que la moyenne et la covariance optimales sont données par : $\mathbf{m}^* = m^* \cdot \mathbf{1}_n$, avec $m^* = \dfrac{e^{-9/2}}{E\sqrt{2\pi}}$, et $\mathbf{1}_n = \frac{1}{\sqrt{n}}(1, \dots, 1)^\top \in \mathbb{R}^n$, et $\Sigma^* = (v^* - 1)\mathbf{1}_n\mathbf{1}_n^\top + I_n$, où $v^* = 3m^* - (m^*)^2 + 1$.

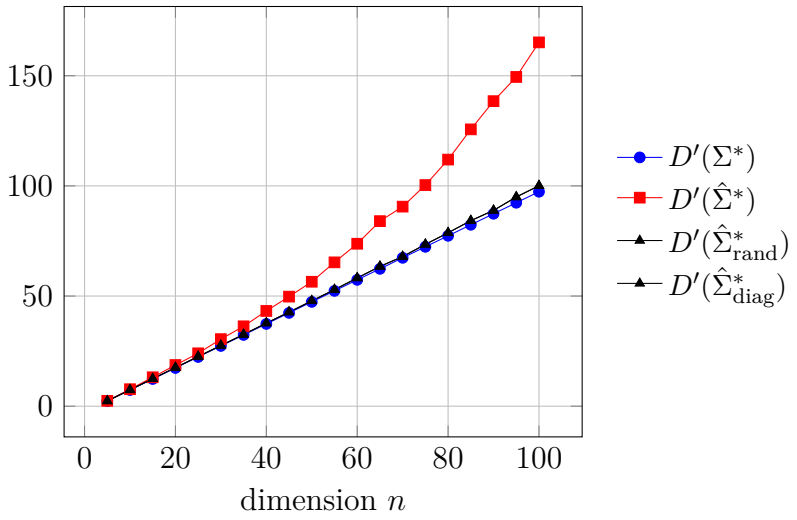


Figure 3.1 – Évolution de la divergence KL "partielle" D' (2.2), entre la densité g^* et la famille $\{g_{\mathbf{m}^*,\Sigma}\}$, en fonction de la dimension, avec la matrice optimale Σ^* (cercles bleus), la matrice empirique $\hat{\Sigma}^*$ (carrés rouges), la matrice empirique diagonale $\hat{\Sigma}^*_{\text{diag}}$, et la matrice empirique projetée aléatoirement $\hat{\Sigma}^*_{\text{rand}}$ (triangles noirs), lorsque $\phi = \mathbb{I}_{\{\varphi \geq 0\}}$ avec $\varphi = \varphi_1$ (2.1).

Observons d'abord l'évolution de la divergence de Kullback-Leibler partielle D' (2.2) avec la dimension lorsqu'on considère les matrices $\hat{\Sigma}^*_{\text{diag}}$ et $\hat{\Sigma}^*_{\text{rand}}$. Les courbes correspondantes sont représentées figure 3.1. Notons que la matrice $\hat{\Sigma}^*$ est ici estimée avec un échantillon selon g^* de taille $M = 200$. La divergence est bien plus faible pour les matrices "projetées" (triangles noirs, les deux courbes étant confondues) que pour la matrice empirique (carrés rouges) à partir de la dimension 50 environ, et reste proche de la divergence optimale (cercles bleus). Cette diminution de la divergence KL signifie que les densités d'échantillonnage $g_{\hat{\mathbf{m}}^*,\hat{\Sigma}^*_{\text{diag}}}$ et $g_{\hat{\mathbf{m}}^*,\hat{\Sigma}^*_{\text{rand}}}$ approchent mieux la densité cible g^*, que $g_{\hat{\mathbf{m}}^*,\hat{\Sigma}^*}$, ce qui implique également une meilleure estimation de la probabilité, comme on peut le voir dans le tableau 3.1, regroupant les résultats obtenus selon la procédure expliquée au début de la section 3.1.3.

Ce tableau montre en effet la forte dégradation de l'estimation lorsqu'on utilise la matrice empirique $\hat{\Sigma}^*$ (coefficient de variation de 9.2% en dimension 40, 79% en dimension 100), alors que la matrice optimale obtient toujours des résultats très précis (coefficient de variation toujours autour de 2%). Les matrices Σ^*_{diag} et Σ^*_{rand} donnent des résultats proches de ceux donnés par Σ^* (divergence à moins de 5% de l'optimum en dimension 40, et moins de 2% en dimension 100 ; coefficient de variation toujours inférieur à 5%), ce qui signifie que les directions de projection conviennent. De plus, les matrices $\hat{\Sigma}^*_{\text{diag}}$ et $\hat{\Sigma}^*_{\text{rand}}$, calculées à partir de $\hat{\Sigma}^*$, gardent une divergence KL proche de la valeur optimale pour toutes les dimensions (moins de 5%) et l'estimation est assez précise (coefficient de variation entre 4 et 6%). L'amélioration est donc significative, dans ce cas simple, même avec ces choix naïfs de projection. Cela montre qu'il peut être utile de projeter pour diminuer le nombre de paramètres à estimer et ainsi réduire le nombre d'erreurs d'estimation, même sans connaitre des directions optimales. Ici, on passe de $n(n+1)/2$ paramètres pour estimer $\hat{\Sigma}^*$, à n pour $\hat{\Sigma}^*_{\text{rand}}$ et $\hat{\Sigma}^*_{\text{diag}}$ (par exemple pour $n = 100$, on passe de 5050 à 100 paramètres) ce qui est déjà une réduction conséquente.

		Σ^*	$\hat{\Sigma}^*$	Σ^*_{diag}	$\hat{\Sigma}^*_{\text{diag}}$	Σ^*_{rand}	$\hat{\Sigma}^*_{\text{rand}}$
$n = 40$	$D'(\Sigma)$	37.4	39.3	39.1	39.1	39.0	39.1
	Erreur relative (%)	0	5.1	4.6	4.5	4.5	4.7
$E = 1.35 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.34	1.35	1.37	1.34	1.34	1.36
	Coefficient de variation (%)	2.0	9.2	4.1	4.9	4.0	5.0
$n = 70$	$D'(\Sigma)$	67.4	73.7	69.1	69.2	69.1	69.2
	Erreur relative (%)	0	9.4	2.5	2.8	2.5	2.7
$E = 1.35 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.35	1.26	1.34	1.36	1.35	1.34
	Coefficient de variation (%)	2.2	37	4.6	4.9	4.0	4.0
$n = 100$	$D'(\Sigma)$	97.4	111.8	99.1	99.3	99.1	99.3
	Erreur relative (%)	0	14.8	1.8	2.0	1.8	2.0
$E = 1.35 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.34	0.87	1.36	1.36	1.36	1.35
	Coefficient de variation (%)	1.9	79	3.0	5.5	4.3	5.9

Tableau 3.1 – Comparaison numérique de la divergence D' et de l'estimation de E pour différentes matrices de covariance lorsque $\phi = \mathbb{I}_{\varphi \geq 0}$ avec $\varphi = \varphi_1$ la somme des variables indépendantes (2.1).

Néanmoins, ces choix de directions de projection ne prennent pas en compte l'information donnée par la fonction φ, ou la forme de la matrice optimale, et ne permettent donc pas d'avoir des estimations finales très précises dans de nombreux exemples. Dans ce premier exemple, la matrice optimale Σ^* est assez proche de la matrice identité en grande dimension, ce qui peut expliquer le bon comportement des matrices $\hat{\Sigma}^*_{\text{diag}}$ mais également $\hat{\Sigma}^*_{\text{rand}}$. En effet, en projetant aléatoirement dans des directions potentiellement peu influentes, les coefficients $\hat{v}_i$ de l'algorithme 6 sont relativement proches de 1 (ou autrement dit des coefficients de variance initiaux, la matrice initiale étant égale à I_n), ce qui implique que $\hat{\Sigma}^*_{\text{rand}}$ est aussi proche de I_n. Les exemples suivants montrent que ces deux manières de projeter peuvent facilement être mises en défaut.

3.1.3.2 Exemple dans le cas événement rare : un polynôme de degré 2

Pour ce deuxième exemple, nous restons dans l'estimation d'une probabilité d'événement rare avec la fonction suivante :

$$\varphi_2(\mathbf{x}) = x_1 - 2(x_1 - x_2)^2 - 3(x_1 - x_3)^2 - 1. \tag{3.8}$$

Le graphe de cette fonction est tracé figure 3.2 en 3 dimensions, et correspond à l'état limite $\{\varphi_2(\mathbf{x}) = 0\}$.

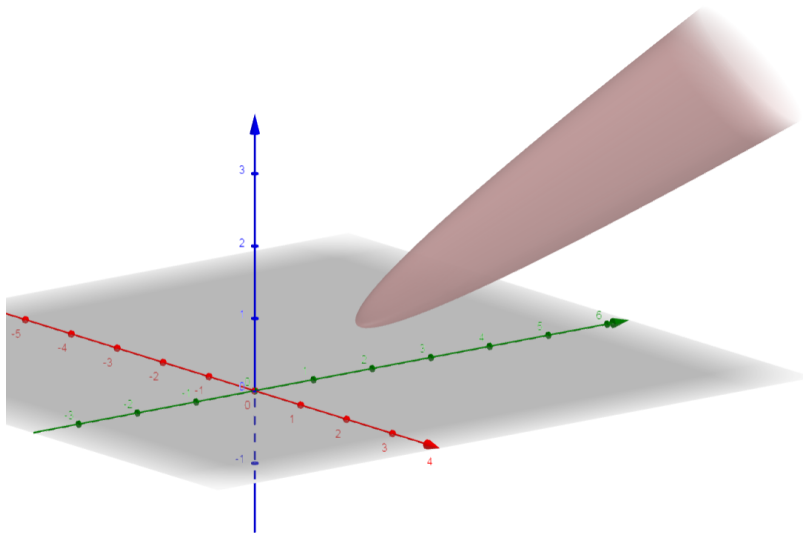


Figure 3.2 – Paraboloïde en 3 dimensions correspondant à l'état limite $\{\varphi_2(\mathbf{x}) = 0\}$. L'axe des x_1 est en rouge, celui des x_2 en vert, et des x_3 en bleu.

Dans ce cas, la valeur de référence de la probabilité E vaut environ $1.23 \cdot 10^{-3}$ pour toute dimension $n \geq 3$ (estimée par Monte-Carlo avec un budget très important). On peut montrer que la moyenne optimale est de la forme $\mathbf{m}^* = (m_1, m_2, m_3, 0 \ldots, 0)$ puisque φ_2 ne dépend que des trois premières variables, et ses coefficients valent environ $m_1 \approx 1.44$, $m_2 \approx 1.36$, et $m_3 \approx 1.39$. La matrice de covariance optimale, quant à elle, s'écrit $\Sigma^* = \begin{pmatrix} A & 0 \\ 0 & I_{n-3} \end{pmatrix}$, avec $A \in \mathcal{S}_3^+$ (dont les coefficients diagonaux sont approximativement estimés, par Monte-Carlo, à $A_{11} \approx 0.077$, $A_{22} \approx 0.094$, $A_{33} \approx 0.088$, et les coefficients de covariance $A_{12} \approx 0.060$, $A_{13} \approx 0.065$, $A_{23} \approx 0.049$). Cette matrice n'est donc pas diagonale, et les premiers coefficients diagonaux sont bien inférieurs à 1 (contrairement à l'exemple précédent où ils se rapprochent de 1 lorsque la dimension augmente).

L'évolution de la divergence de Kullback-Leibler partielle (D') en fonction de la dimension est tracée figure 3.3. Comme précédemment, la divergence pour $\hat{\Sigma}^*_{\text{rand}}$ (triangles noirs), et $\hat{\Sigma}^*_{\text{diag}}$ (losanges verts) est plus faible que pour la matrice empirique (carrés rouges) à partir de la dimension 50. Cela se traduit par une meilleure estimation de la probabilité, comme on peut l'observer dans le tableau 3.2. De plus, la divergence D' est légèrement plus grande pour $\hat{\Sigma}^*_{\text{rand}}$ que pour $\hat{\Sigma}^*_{\text{diag}}$ dans toutes les dimensions, les deux restant néanmoins assez proches de la valeur minimale $D'(\Sigma^*)$ (cercles bleus).

Le tableau 3.2 regroupe les résultats de la divergence D' et d'estimation de E pour les dimensions 30, 70, et 100. On peut remarquer qu'en dimension 30 les matrices Σ^*_{rand} et $\hat{\Sigma}^*_{\text{rand}}$ donnent des estimations moins précises que la matrice empirique (coefficient de variation de 18.3% et 18.9%

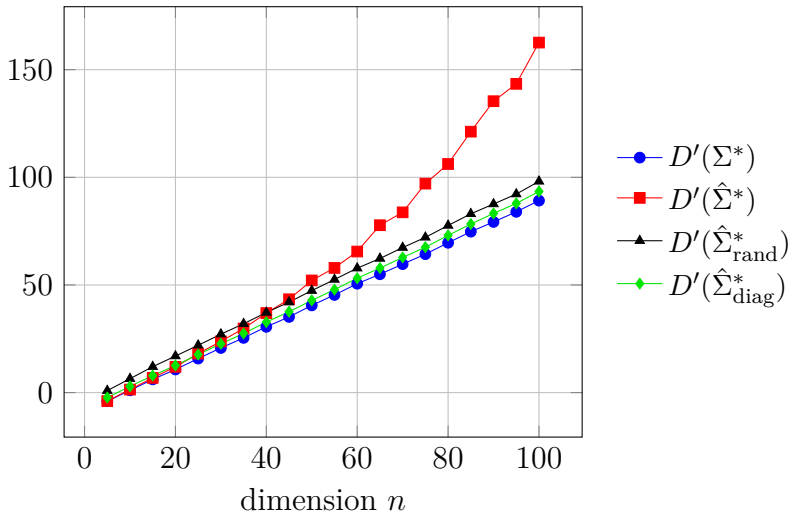


Figure 3.3 – Évolution de la divergence KL "partielle" D', entre la densité g^* et la famille $\{g_{\mathbf{m}^*,\Sigma}\}$, en fonction de la dimension, avec la matrice optimale Σ^* (cercles bleus), la matrice empirique $\hat{\Sigma}^*$ (carrés rouges), la matrice empirique diagonale $\hat{\Sigma}^*_{\text{diag}}$ (losanges verts), et la matrice empirique projetée aléatoirement $\hat{\Sigma}^*_{\text{rand}}$ (triangles noirs), lorsque $\phi = \mathbb{I}_{\{\varphi \geq 0\}}$ avec $\varphi = \varphi_2$ (3.8).

respectivement contre 9.2% pour $\hat{\Sigma}^*$), ce qui est lié à une plus grande erreur relative sur la divergence D' (29% pour les matrices projetées aléatoirement contre 4.6% pour $\hat{\Sigma}^*$). Les directions de projection semblent donc avoir une plus grande influence ici, ce qui n'est pas surprenant étant donné que la fonction φ_2 ne dépend que des trois premières coordonnées, il vaut ainsi mieux projeter dans le sous-espace engendré par $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$. En estimant la variance dans des directions aléatoires et potentiellement différentes de $\mathbf{e}_1$, $\mathbf{e}_2$, ou $\mathbf{e}_3$, les matrices Σ^*_{rand} et $\hat{\Sigma}^*_{\text{rand}}$ sont proches de la matrice identité, comme expliqué dans l'exemple précédent. Elles ne permettent donc pas de détecter les faibles valeurs de variance prises par les trois premières variables. Cependant, dans les dimensions supérieures, la matrice $\hat{\Sigma}^*$ devient bien moins précise (coefficient de variation entre 43 et 220% et estimation qui s'éloigne de la valeur attendue) que Σ^*_{rand} et $\hat{\Sigma}^*_{\text{rand}}$ qui gardent une précision raisonnable (estimation proche de la valeur de référence et coefficient de variation entre 19 et 24%). Si l'on regarde les résultats donnés par les matrices diagonales, on remarque que la divergence D' reste proche de la valeur optimale dans toutes les dimensions, comme évoqué figure 3.3, et de ce fait, que l'estimation reste assez précise (coefficient de variation toujours inférieur à 9% pour Σ^*_{diag}, et inférieur à 10% pour $\hat{\Sigma}^*_{\text{diag}}$). Ainsi, malgré la présence de coefficients de covariance non nuls dans Σ^*, ceux-ci sont peu nombreux et prennent de faibles valeurs, ce qui explique que l'estimation de E ne soit pas trop dégradée en ne les prenant pas en compte. Avec Σ^*, le coefficient de variation varie entre 3.4 et 6.2%, alors qu'avec Σ^*_{diag}, il reste entre 6.5 et 8.6%. On a donc bien une perte de précision mais celle-ci est minime et l'amélioration donnée par $\hat{\Sigma}^*_{\text{diag}}$ par rapport à $\hat{\Sigma}^*$ est significative et justifie de laisser les coefficients extra-diagonaux de côté pour estimer la matrice de covariance. Néanmoins, nous verrons dans l'exemple 3.1.3.4 que considérer uniquement les coefficients diagonaux peut être insuffisant. Finalement, lorsque la dimension est grande, ce cas suggère que les matrices "projetées" sont toujours plus efficaces que la matrice empirique, comme dans l'exemple précédent.

		Σ^*	$\hat{\Sigma}^*$	Σ^*_{diag}	$\hat{\Sigma}^*_{\text{diag}}$	Σ^*_{rand}	$\hat{\Sigma}^*_{\text{rand}}$
$n = 30$	$D'(\Sigma)$	21.0	22.0	22.7	22.7	27.1	27.1
	Erreur relative (%)	0	4.6	7.9	8.1	29	29
$E = 1.23 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.23	1.24	1.22	1.22	1.21	1.21
	Coefficient de variation (%)	6.2	9.2	6.9	9.6	18.3	18.9
$n = 70$	$D'(\Sigma)$	61.0	67.5	62.7	62.8	67.2	67.3
	Erreur relative (%)	0	10.6	2.7	3.0	10.1	10.3
$E = 1.23 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.23	1.16	1.21	1.23	1.24	1.23
	Coefficient de variation (%)	3.4	43	6.5	9.3	17.8	19
$n = 100$	$D'(\Sigma)$	91.0	106.1	92.7	92.9	97.2	97.5
	Erreur relative (%)	0	16.6	1.8	2.1	6.8	7.1
$E = 1.23 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.24	1.08	1.21	1.20	1.22	1.25
	Coefficient de variation (%)	6.2	220	8.6	8.4	19	24

Tableau 3.2 – Comparaison numérique de la divergence D' et de l'estimation de E pour différentes matrices de covariance lorsque $\phi = \mathbb{I}_{\varphi \geq 0}$ avec $\varphi = \varphi_2$ une fonction polynomiale de degré 2 (3.8).

3.1.3.3 Exemple pour l'échantillonnage selon la loi "*banana shape*"

On reprend ici l'exemple jouet de la section 2.1.3, où l'on cherche à échantillonner selon la loi en forme de banane, de densité π (2.3). L'intégrale que l'on évaluera par échantillonnage préférentiel ici sera la constante de normalisation de la densité $E = \int \pi(x)dx$, dont la valeur théorique est 1. La densité optimale d'IS g^* pour l'estimation de $\int \pi(x)dx$ est exactement π. On rappelle que les paramètres gaussiens optimaux sont $\mathbf{m}^* = \mathbf{0} \in \mathbb{R}^n$ et $\Sigma^* = \text{diag}(100, 19, 1, \ldots, 1) \in \mathcal{S}_n^+$.

La divergence de Kullback-Leibler partielle (D') et les résultats d'estimation de E sont regroupés dans le tableau 3.3. On peut voir, comme précédemment, que la divergence de $\hat{\Sigma}^*$ s'éloigne de la valeur minimale ($D'(\Sigma^*)$) quand la dimension grandit, alors que celle de $\hat{\Sigma}^*_{\text{diag}}$ reste très proche. Cela se traduit par une estimation finale très précise pour $\hat{\Sigma}^*_{\text{diag}}$, le coefficient de variation étant au maximum égal à 8.7%, alors qu'avec la matrice empirique, le coefficient de variation explose (plus de 900% en dimension 100). En revanche, pour Σ^*_{rand} et $\hat{\Sigma}^*_{\text{rand}}$ on remarque que la divergence est toujours éloignée de la valeur optimale, l'erreur relative valant jusqu'à 82% en dimension 40. De ce fait, le coefficient de variation de $\hat{E}_N$ est supérieur à 95% dans toutes les dimensions, pour ces deux matrices. L'inefficacité des matrices construites à partir des projections aléatoires s'explique en partie par la mauvaise estimation des deux premiers coefficients diagonaux. En effet, ces deux quantités prennent en théorie de grandes valeurs ($\Sigma^*_{11} = 100$ et $\Sigma^*_{22} = 19$) et contiennent l'essentiel de l'information de la matrice alors que Σ^*_{rand} et $\hat{\Sigma}^*_{\text{rand}}$ ont souvent des coefficients diagonaux assez petits (un exemple de matrices Σ^*_{rand} et $\hat{\Sigma}^*_{\text{rand}}$ donne en dimension 40, $\Sigma^*_{\text{rand},11} \approx 11$, $\Sigma^*_{\text{rand},22} \approx 5$, et $\hat{\Sigma}^*_{\text{rand},11} \approx 12$, $\hat{\Sigma}^*_{\text{rand},22} \approx 5$; en dimension 100, $\Sigma^*_{\text{rand},11} \approx 2$, $\Sigma^*_{\text{rand},22} \approx 1.6$, et $\hat{\Sigma}^*_{\text{rand},11} \approx 1.9$, $\hat{\Sigma}^*_{\text{rand},22} \approx 1.5$).

Ainsi, comme dans l'exemple précédent, projeter de manière aléatoire ne permet pas de bien approcher la loi cible π et donc de bien estimer l'intégrale E. En revanche, estimer uniquement la diagonale de la matrice de covariance donne de très bons résultats, dans toutes les dimensions considérées, étant donné que la matrice optimale Σ^* est elle-même diagonale. L'exemple suivant est un dernier cas jouet montrant les limites des deux choix de projection (aléatoire, et sur les vecteurs canoniques pour obtenir une matrice diagonale).

		$\Sigma^* = \Sigma^*_{\text{diag}}$	$\hat{\Sigma}^*$	$\hat{\Sigma}^*_{\text{diag}}$	Σ^*_{rand}	$\hat{\Sigma}^*_{\text{rand}}$
$n = 40$	$D'(\Sigma)$	47.6	49.5	47.7	86.2	86.4
	Erreur relative (%)	0	4.1	0.2	81	82
$E = 1$	Estimation moyenne	0.999	1.011	0.994	0.125	0.245
	Coefficient de variation (%)	9.9	38	8.7	94	110
$n = 70$	$D'(\Sigma)$	77.6	84.0	77.7	128.1	128.3
	Erreur relative (%)	0	8.3	0.2	65	65
$E = 1$	Estimation moyenne	0.982	0.968	0.981	0.232	0.416
	Coefficient de variation (%)	6.8	76	6.7	160	230
$n = 100$	$D'(\Sigma)$	107.6	122.1	107.8	164.6	164.9
	Erreur relative (%)	0	14	0.2	53	53
$E = 1$	Estimation moyenne	0.989	1.564	0.979	0.108	0.122
	Coefficient de variation (%)	6.0	940	7.9	97	120

Tableau 3.3 – Comparaison numérique de la divergence D' et de l'estimation de $E = \int \pi(x)dx$ pour différentes matrices de covariance, où g^* est égale la densité de la loi "*banana shape*" π (2.3).

3.1.3.4 Exemple pour l'échantillonnage selon une loi gaussienne centrée

Dans ce dernier exemple, on considère la loi normale centrée de dimension n, $\mathcal{N}(0, V_0)$, où $V_0 = (1/2)(I_n + \mathbf{1}_n\mathbf{1}_n^\top)$ est la matrice avec des 1 sur la diagonale, tous les autres coefficients étant égaux à $\frac{1}{2}$, et on veut approcher cette loi par échantillonnage préférentiel. Autrement dit, on a $\mathbf{m}^* = \mathbf{0}$ et $\Sigma^* = V_0$. On va calculer la divergence de Kullback-Leibler partielle $D'(\Sigma)$ pour les différentes matrices proposées, et estimer $E = \int g_{0,V_0}(x)dx (= 1)$ par échantillonnage préférentiel, toujours selon la méthode décrite par l'algorithme 6. Remarquons que dans ce cas, on a $g^* = g_{0,V_0}$ et donc que la divergence de Kullback-Leibler $D(g^*, g_{\mathbf{m}^*,\Sigma^*})$ vaut exactement 0, ce qui n'est pas le cas de la divergence partielle $D'(\Sigma^*)$, comme on le voit dans le tableau 3.4. Notons également qu'on ne peut pas estimer E directement par échantillonnage préférentiel avec la loi $g_{\mathbf{m}^*,\Sigma^*}$ puisque le rapport de vraisemblance vaut exactement 1 (ce qui revient à calculer : $\frac{1}{N}\sum_{i=1}^N 1 = 1$ quel que soit l'échantillon). C'est pourquoi les cases correspondantes dans le tableau sont laissées vides.

On lit ainsi dans le tableau 3.4 que toutes les matrices avec une projection donnent une mauvaise approximation de la densité cible, et donc une mauvaise estimation de l'espérance. En dimension 40 et 70, la matrice empirique est même nettement plus efficace que toutes les autres. En effet, la divergence D' de $\hat{\Sigma}^*$ a une erreur relative de 12% en dimension 40 et 25% en dimension 70, alors que les matrices Σ^*_{diag}, Σ^*_{rand}, $\hat{\Sigma}^*_{\text{diag}}$, et $\hat{\Sigma}^*_{\text{rand}}$ ont toutes une erreur supérieure à 100%. De même, les estimations sont très imprécises pour ces matrices, au vu de leurs coefficients de variation (supérieurs à 43% en dimension 40 et à 79% en dimension 70) et la valeur de l'espérance est souvent sous-estimée. La matrice $\hat{\Sigma}^*$ donne une estimation proche de la valeur théorique (1.016 en dimension 40, 0.918 en dimension 70) et un coefficient de variation nettement inférieur à celui des autres matrices (8.6% en dimension 40 et 31% en dimension 70). En dimension 100, la matrice de covariance empirique devient moins précise (coefficient de variation de 120%) mais reste toujours un peu plus performante que les matrices avec projection.

Ainsi, cet exemple met clairement en défaut les deux techniques d'estimation de la covariance proposées. On comprend bien qu'estimer uniquement la diagonale de Σ^*, et laisser tous les autres coefficients à zéro, ne donne pas des résultats performants ici, étant donné que tous les coefficients de covariance sont non nuls. De même, les matrices utilisant une projection aléatoire, Σ^*_{rand} et $\hat{\Sigma}^*_{\text{rand}}$,

		Σ^*	$\hat{\Sigma}^*$	Σ^*_{diag}	$\hat{\Sigma}^*_{\text{diag}}$	Σ^*_{rand}	$\hat{\Sigma}^*_{\text{rand}}$
$n = 40$	$D'(\Sigma)$	16.0	17.9	40.0	40.1	33.6	33.7
	Erreur relative (%)	0	12	150	150	110	110
$E = 1$	Estimation moyenne	/	1.016	0.556	0.762	0.719	0.770
	Coefficient de variation (%)	/	8.6	57	83	43	55
$n = 70$	$D'(\Sigma)$	25.74	32.19	70.0	70.14	61.77	61.91
	Erreur relative (%)	0	25	170	170	140	140
$E = 1$	Estimation moyenne	/	0.918	1.35	0.557	0.368	0.434
	Coefficient de variation (%)	/	31	740	150	79	120
$n = 100$	$D'(\Sigma)$	35.30	49.83	100.0	100.21	85.01	85.22
	Erreur relative (%)	0	41	180	180	140	140
$E = 1$	Estimation moyenne	/	0.816	0.155	0.218	0.164	0.185
	Coefficient de variation (%)	/	120	90	97	89	89

Tableau 3.4 – Comparaison numérique de la divergence D' et de l'estimation de E pour différentes matrices de covariance lorsque g^* est la densité gaussienne centrée g_{0,V_0}.

sont également proches de la matrice identité (avec des termes bien plus petits que 1/2 hors de la diagonale, comme on peut le voir figure 3.4), et ne permet donc pas d'avoir des résultats précis pour les mêmes raisons.

3.1.3.5 Conclusion

Les 4 exemples ont montré que l'estimation par échantillonnage préférentiel était souvent plus précise en grande dimension avec les matrices $\hat{\Sigma}^*_{\text{diag}}$ et $\hat{\Sigma}^*_{\text{rand}}$ qu'avec $\hat{\Sigma}^*$. Estimer un petit nombre de paramètres de covariance dans des directions préalablement choisies diminue la divergence de Kullback-Leibler et améliore donc l'estimation finale de l'espérance. Lorsque la matrice ciblée, Σ^*, est proche de l'identité (exemple 3.1.3.1), les 2 méthodes donnent des résultats similaires et très satisfaisants. Cependant pour des matrices plus complexes, prendre des directions aléatoires atteint vite ses limites (voir 3.1.3.2 et 3.1.3.3), et ne permet pas d'avoir une estimation très précise. En estimant uniquement la diagonale de la matrice optimale, l'exemple 3.1.3.4 a montré que les résultats n'étaient pas toujours performants, si la matrice optimale comportait beaucoup de termes de covariance non nuls qu'il ne fallait pas négliger. Ainsi, si estimer la variance dans un petit nombre de directions permet de diminuer les erreurs d'estimation, le choix de ces directions peut avoir une grande influence sur la qualité de l'approximation de la loi cible et sur la précision de l'estimation finale de l'espérance.

Dans la suite de ce manuscrit, on va chercher des directions particulières de projection, en prenant en compte l'information donnée par la fonction d'intérêt (ϕ) ou directement par la matrice optimale Σ^*. Dans la section suivante, on décrit une méthode proposée dans la littérature pour réduire la dimension dans des problèmes inverses bayésiens [Zahm et al., 2018], et repris dans le cadre de l'estimation de probabilités d'événements rares pour améliorer l'algorithme d'entropie croisée [Uribe et al., 2021]. Cette méthode consiste à identifier un sous-espace de petite dimension dans lequel projeter les paramètres d'échantillonnage préférentiel et repose notamment sur un calcul du gradient de la fonction ϕ (ou d'une approximation).

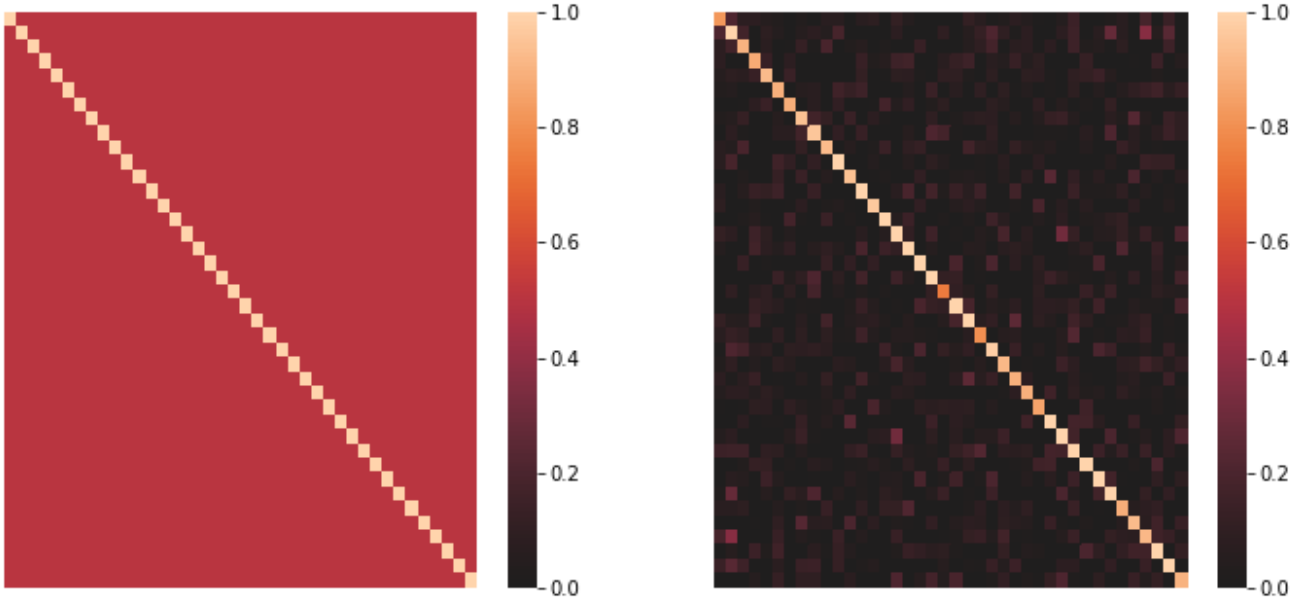


Figure 3.4 – Matrice $\Sigma^* = V_0$ (à gauche) et une réalisation de Σ^*_{rand} (à droite) en dimension 40

3.2 Une méthode de projection basée sur le gradient de la fonction d'intérêt

3.2.1 Projection sur un sous-espace de petite dimension déduit du gradient de la fonction d'intérêt

[Zahm et al., 2018] ont développé une technique de réduction de dimension ("*Certified Dimension Reduction*") pour la résolution de problèmes inverses bayésiens, en exploitant l'information du gradient du logarithme de la fonction d'intérêt. Cette méthode consiste à projeter l'espace des paramètres dans un sous-espace (nommé "*Failure-Informed Subspace*" - FIS - dans [Uribe et al., 2021], dans le contexte des probabilités de défaillance) qui identifie une structure de petite dimension du problème. Ainsi, les paramètres (dans notre cas, $\mathbf{m}^*$ et Σ^*) sont mis à jour en dimension réduite.

Nous allons décrire cette méthode dans le cadre de ce manuscrit, où l'on veut estimer l'espérance $E = \mathbb{E}_f(\phi(\mathbf{X}))$ par échantillonnage préférentiel. Dans cette section, on considère que la fonction d'intérêt ϕ est continument différentiable (ou pouvant être approchée par une fonction suffisamment régulière) et vérifie $\mathbb{E}_f(\|\nabla \ln \phi(\mathbf{X})\|^2) < +\infty$.

L'idée est d'approcher $g^* = \phi f / E$ par une densité g^*_k n'agissant que sur un sous-espace de petite dimension. En supposant que l'on connaisse un projecteur, $P_k \in \mathbb{R}^{n\times n}$, dans un sous-espace de dimension $k < n$ (en reprenant les notations de la partie 3.1.2, on a en fait $P_k = RR^\top$), [Uribe et al., 2021] suggèrent de définir cette approximation par $g^*_k(\mathbf{x}) \propto \mathbb{E}_f(\phi(\mathbf{X})|\mathbf{x}_k) f(\mathbf{x})$, où $\mathbf{x}_k = P_k\mathbf{x}$ est la projection du vecteur $\mathbf{x} \in \mathbb{R}^n$. D'après les travaux de [Zahm et al., 2018], l'espérance conditionnelle $\mathbb{E}_f(\phi(\mathbf{X})|\mathbf{x}_k)$ est en fait la meilleure approximation de ϕ selon la divergence de Kullback-Leibler, sur toutes les fonctions mesurables sur le sous-espace de dimension k (défini par P_k). Dans le cas où f est la densité $\mathcal{N}(\mathbf{0}, I_n)$, la divergence KL entre g^* et g^*_k peut alors être

majorée par la borne suivante :

$$D(g^*, g_k^*) \leq \frac{1}{2}\mathrm{tr}\left((I_n - P_k)H(I_n - P_k)\right)$$

où H est la matrice définie par :

$$H = \int_{\mathbb{R}^n} \nabla \ln \phi(\mathbf{x}) \nabla \ln \phi(\mathbf{x})^\top g^*(\mathbf{x}) d\mathbf{x} = \mathbb{E}_{g^*}\left(\nabla \ln \phi(\mathbf{X}) \nabla \ln \phi(\mathbf{X})^\top\right). \tag{3.9}$$

Ainsi, [Zahm et al., 2018] montrent que le projecteur minimisant cette borne est donné par :

$$P_k^* = \sum_{j=1}^{k} \mathbf{d}_j \mathbf{d}_j^\top, \tag{3.10}$$

où les $\mathbf{d}_j$ sont les vecteurs propres de la matrice H correspondants aux k plus grandes valeurs propres, celles-ci étant rangées dans l'ordre décroissant $\lambda_1 \geq \ldots \geq \lambda_n$. De plus, le minimum de cette borne vaut : $\mathrm{tr}\left((I_n - P_k^*)H(I_n - P_k^*)\right) = \sum_{j=k+1}^{n} \lambda_j$. L'approximation peut alors être contrôlée par un seuil de tolérance préalablement choisi ε :

$$D(g^*, g_k^*) \leq \frac{1}{2} \sum_{j=k+1}^{n} \lambda_j \leq \varepsilon, \tag{3.11}$$

et on peut sélectionner la dimension k de l'espace réduit comme le plus petit entier k' tel que $\sum_{j=k'+1}^{n} \lambda_j \leq \varepsilon$.

Dans le cas où $\phi = \mathbb{I}_{\{\varphi \geq 0\}}$, [Uribe et al., 2021] reprennent l'idée de l'algorithme iCE (2) et proposent de remplacer l'indicatrice par une approximation lisse, (par exemple $\psi(\cdot, \sigma) = F_{\mathcal{N}}(\varphi(\cdot)/\sigma)$, voir section 1.3.3.2). En supposant φ continument différentiable, et $\mathbb{E}_f(\|\nabla \ln \psi(\mathbf{X}, \sigma)\|^2) < +\infty$, pour tout $\sigma > 0$, on peut définir la matrice H (3.9) et le projecteur P_k^* (3.10), comme précédemment. Dans [Uribe et al., 2021], le sous-espace de petite dimension engendré par les vecteurs propres $\mathbf{d}_1, \ldots, \mathbf{d}_k$ de H est appelé ***Failure-Informed Subspace*** (FIS). Le sous-espace orthogonal au FIS est engendré par les $n - k$ vecteurs propres restants et est appelé ***Complementary Subspace*** (CS). Le terme "*Failure-Informed Subspace*" est particulièrement adapté à l'estimation des probabilités de défaillance, mais nous garderons ce nom même lorsqu'on estime une espérance en général.

Nous proposons maintenant de tester l'efficacité de cette projection sur quelques exemples, en reprenant la procédure 6. On suppose donc que l'on sait échantillonner selon la loi g^* comme dans la section précédente 3.1.3. La matrice H (3.9) est alors estimée par :

$$\hat{H} = \frac{1}{M} \sum_{i=1}^{M} \left[\nabla \ln \phi(\mathbf{X}_i)\right] \left[\nabla \ln \phi(\mathbf{X}_i)\right]^\top, \tag{3.12}$$

avec $\mathbf{X}_1, \ldots, \mathbf{X}_M$ générés indépendamment selon g^*. Lorsqu'on se place dans le cadre des probabilités d'événements rares, ϕ est remplacée par l'approximation de l'indicatrice $\psi(\cdot, \sigma)$ dans la formule (3.12) ($\sigma > 0$ fixé). Enfin, seule la matrice de covariance sera projetée (il est aussi possible de mettre à jour la moyenne avec cette méthode, comme dans [Uribe et al., 2021]) car comme nous l'avons déjà précisé, la majorité des paramètres estimés provient de cette matrice en grande dimension. Ainsi, l'estimation de E s'effectue grâce à l'algorithme 7.

La prochaine section présente des résultats numériques obtenus avec cet algorithme, et permet d'évaluer la performance de la projection sur le FIS.

Algorithme 7 : Algorithme d'estimation de E à l'aide de la projection sur le FIS

Données : Taille des échantillons M et N, paramètre ε

Résultat : Estimation $\hat{E}_N$ de E

1 Générer un échantillon $\mathbf{X}_1, \dots, \mathbf{X}_M$ selon g^* pour estimer $\mathbf{m}^*$ et Σ^* par $\hat{\mathbf{m}}^*$ et $\hat{\Sigma}^*$;

2 Évaluer les $\nabla \ln \phi(\mathbf{X}_i)$ pour estimer la matrice $\hat{H}$ (3.12), et calculer ses valeurs propres $\lambda_1 \geq \dots \geq \lambda_n$ et ses vecteurs propres $\mathbf{d}_1, \dots, \mathbf{d}_n$;

3 Déterminer le plus petit entier k tel que $\sum_{j=k+1}^{n} \lambda_j \leq \epsilon$;

4 Calculer les $\hat{v}_i = \mathbf{d}_i^\top \hat{\Sigma}^* \mathbf{d}_i$, pour $i = 1 \dots k$, et la matrice $\hat{\Sigma}_k$ définie en (3.4), avec $v_i = \hat{v}_i$;

5 Générer un nouvel échantillon $\mathbf{X}_1, \dots, \mathbf{X}_N$ selon $g_{\hat{\mathbf{m}}, \hat{\Sigma}_k}$ et estimer $\hat{E}_N$ (1.4).

3.2.2 Simulations numériques

De la même manière que dans la partie 3.1.3, les tableaux présentés dans la suite font apparaitre la divergence de Kullback-Leibler partielle D', l'erreur relative associée, l'estimation moyenne, et le coefficient de variation pour les matrices Σ^*, $\hat{\Sigma}^*$, et $\hat{\Sigma}^*_{\text{FIS}}$. Cette dernière est la matrice obtenue par l'algorithme 7, où les directions de projection sont celles du sous-espace "FIS". Les valeurs sont calculées sur 50 répétitions indépendantes, les tailles d'échantillon sont toujours fixées à $M = 500$ et $N = 2000$ et le paramètre ε à 0.01. Notons que dans les cas considérés, le gradient de $\ln \phi$ est calculable analytiquement et nous utilisons donc son expression exacte pour son évaluation. Cela implique que le FIS estimé est très proche (voire égal) du FIS théorique. De plus, en projetant Σ^* dans le FIS (au lieu de $\hat{\Sigma}^*$), la matrice Σ^*_{FIS} obtenue serait identique ou très proche de Σ^*, de sorte que les résultats d'estimation seraient indiscernables, et c'est pourquoi Σ^*_{FIS} n'est pas représentée dans les tableaux suivants.

3.2.2.1 Exemple jouet dans le cas événement rare : somme de variables indépendantes

Le premier exemple est l'estimation de la probabilité d'événement rare avec la fonction d'état limite φ_1 (2.1) et $\Sigma^* = (v^* - 1)\mathbf{1}_n\mathbf{1}_n^\top + I_n$ (où $\mathbf{1}_n = \frac{1}{\sqrt{n}}(1, \dots, 1)^\top$). On rappelle que dans ce cas, la fonction indicatrice $\mathbb{I}_{\{\varphi_1 \geq 0\}}$ est remplacée par l'approximation lisse $\psi_1(\cdot, \sigma) = F_{\mathcal{N}}(\varphi_1/\sigma)$ (avec $\sigma = 0.5$ dans les simulations). Le gradient de φ_1 valant $\nabla\varphi_1 = (1, \dots, 1)^\top$, on a

$$\nabla \ln \psi_1(\mathbf{x}, \sigma) = \frac{f(\varphi_1(\mathbf{x}))/\sigma)}{\sigma F_{\mathcal{N}}(\varphi_1(\mathbf{x})/\sigma)} \, (1, \dots, 1)^\top$$

pour tout $\mathbf{x} \in \mathbb{R}^n$ et la matrice H est alors égale à

$$H = \mathbb{E}_{g^*}\left(\frac{f(\varphi_1(\mathbf{X})/\sigma)^2}{\sigma^2 F_{\mathcal{N}}(\varphi_1(\mathbf{X})/\sigma)^2}\right) n\mathbf{1}_n\mathbf{1}_n^\top.$$

Les valeurs propres de H sont toutes nulles sauf une, qui vaut $n\mathbb{E}_{g^*}\left(\frac{f(\varphi_1(\mathbf{X})/\sigma)^2}{\sigma^2 F_{\mathcal{N}}(\varphi_1(\mathbf{X})/\sigma)^2}\right) > 0$, et qui est associée au vecteur propre $\mathbf{1}_n$. En théorie, le FIS est donc le sous-espace de dimension 1 engendré par $\mathbf{1}_n$. De plus, le gradient de φ_1 étant ici constant, et donc indépendant des échantillons générés, la matrice estimée $\hat{H}$, donnée par

$$\hat{H} = n\mathbf{1}_n\mathbf{1}_n^\top \sum_{i=1}^{N} \frac{f(\varphi_1(\mathbf{X}_i)/\sigma)^2}{\sigma^2 F_{\mathcal{N}}(\varphi_1(\mathbf{X}_i)/\sigma)^2}, \quad \mathbf{X}_i \underset{i.i.d.}{\sim} g^*$$

		Σ^*	$\hat{\Sigma}^*$	$\hat{\Sigma}^*_{\text{FIS}}$
$n = 40$	$D'(\Sigma)$	37.4	39.3	37.4
	Erreur relative (%)	0	5.1	0.02
$E = 1.35 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.34	1.35	1.34
	Coefficient de variation (%)	2.0	9.2	2.7
$n = 70$	$D'(\Sigma)$	67.4	73.7	67.4
	Erreur relative (%)	0	9.4	0.01
$E = 1.35 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.35	1.26	1.34
	Coefficient de variation (%)	2.2	37	2.4
$n = 100$	$D'(\Sigma)$	97.4	111.8	97.4
	Erreur relative (%)	0	14.8	0.01
$E = 1.35 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.34	0.87	1.35
	Coefficient de variation (%)	1.9	79	2.3

Tableau 3.5 – Comparaison numérique de la divergence D' et de l'estimation de E pour les matrices Σ^*, $\hat{\Sigma}^*$, $\hat{\Sigma}^*_{\text{FIS}}$ lorsque $\phi = \mathbb{I}_{\varphi \geq 0}$ avec $\varphi = \varphi_1$ la somme des variables indépendantes (2.1).

est toujours proportionnelle à $\mathbf{1}_n \mathbf{1}_n^\top$ et permet de construire un FIS engendré pas $\mathbf{1}_n$. La matrice $\hat{\Sigma}^*_{\text{FIS}}$ vaut alors $(\hat{v}_1 - 1)\mathbf{1}_n \mathbf{1}_n^\top + I_n$, avec $\hat{v}_1 = \mathbf{1}_n^\top \hat{\Sigma}^* \mathbf{1}_n$, et est exactement de la même forme que Σ^*, la seule différence étant l'estimation du paramètre $\hat{v}_1$. Ce n'est donc pas surprenant de voir que les résultats donnés par cette matrice dans le tableau 3.5, sont presque identiques à ceux de Σ^* et largement meilleurs que pour $\hat{\Sigma}^*$. En effet, la divergence D' de $\hat{\Sigma}^*_{\text{FIS}}$ est toujours à moins de 0.02% de l'optimum et le coefficient de variation de l'estimation de E reste entre 2.3 et 2.7% dans toutes les dimensions (contre 1.9 à 2.2% pour Σ^*). L'amélioration par rapport à $\hat{\Sigma}^*$ est donc significative.

Cependant, cette très grande précision est notamment due au fait que le gradient est constant, et que la direction de projection est exacte. Regardons alors ce qu'il se passe lorsque la fonction d'état limite n'est pas linéaire comme dans l'exemple suivant.

3.2.2.2 Exemple jouet dans le cas événement rare : un polynôme de degré 2

Reprenons la fonction polynomiale φ_2 (3.8). La fonction indicatrice, présente dans g^*, est approchée par $\psi_2(\cdot, \sigma) = F_{\mathcal{N}}(\varphi_2/\sigma)$ (avec $\sigma = 0.5$ dans les simulations), et on cherche à évaluer le gradient $\nabla \ln \psi_2(\cdot, \sigma)$ pour déterminer la matrice H (3.9). Comme le gradient de φ_2 est égal à

$$\nabla \varphi_2(\mathbf{x}) = (1 - 10x_1 + 4x_2 + 6x_3, 4x_1 - 4x_2, 6x_1 - 6x_3, 0, \ldots, 0)^\top,$$

la matrice H est de la forme $\begin{pmatrix} B & 0 \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{n \times n}$, avec B une matrice symétrique définie positive de taille 3. Seules trois valeurs propres de H sont donc strictement positives et les vecteurs propres associés sont des combinaisons linéaires des trois premières variables. Le FIS est ainsi engendré par ces trois vecteurs propres. En pratique, on estime $\hat{H}$ (3.12) à l'aide d'un échantillon tiré selon g^* (de taille $M = 500$ dans les expériences), mais les trois directions trouvées sont très proches des directions exactes (ou ici, estimées avec une taille $M > 10^4$) et le FIS provenant de $\hat{H}$ donne des résultats indiscernables du FIS "exact". Nous suggérons que cela vient du fait que $\nabla \varphi_2$ ne dépend que de trois variables et que peu d'échantillons sont alors nécessaires pour estimer précisément $\hat{H}$, qui revient à estimer une matrice de taille 3.

		Σ^*	$\hat{\Sigma}^*$	$\hat{\Sigma}^*_{\text{FIS}}$
$n = 30$	$D'(\Sigma)$	21.0	22.0	21.0
	Erreur relative (%)	0	4.6	0.07
$E = 1.23 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.23	1.24	1.22
	Coefficient de variation (%)	6.2	9.2	6.0
$n = 70$	$D'(\Sigma)$	61.0	67.5	61.0
	Erreur relative (%)	0	10.6	0.03
$E = 1.23 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.23	1.16	1.24
	Coefficient de variation (%)	3.4	43	4.3
$n = 100$	$D'(\Sigma)$	91.0	106.1	91.0
	Erreur relative (%)	0	16.6	0.02
$E = 1.23 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.24	1.08	1.23
	Coefficient de variation (%)	6.2	220	6.0

Tableau 3.6 – Comparaison numérique de la divergence D' et de l'estimation de E pour les matrices Σ^*, $\hat{\Sigma}^*$, $\hat{\Sigma}^*_{\text{FIS}}$ lorsque $\phi = \mathbb{I}_{\varphi \geq 0}$ avec $\varphi = \varphi_2$ une fonction polynomiale de degré 2 (3.8).

Les résultats fournis par les matrices Σ^*, $\hat{\Sigma}^*$ et $\hat{\Sigma}^*_{\text{FIS}}$ sont présentés dans le tableau 3.6. On peut observer, comme dans l'exemple précédent, que $\hat{\Sigma}^*_{\text{FIS}}$ est aussi précise que Σ^*. En effet, leurs divergences D' et leurs coefficients de variation sont quasiment identiques dans toutes les dimensions alors que la matrice $\hat{\Sigma}^*$ devient très imprécise pour $n = 70$ et $n = 100$. Cette grande efficacité vient d'abord de la forme de $\hat{\Sigma}^*_{\text{FIS}}$ qui est la même que celle de la matrice optimale. Ensuite, la fonction φ_2 (et son gradient) ne dépendant que des trois premières variables, l'estimation de $\hat{H}$ est très précise car seulement six coefficients sont estimés pour l'obtenir. Ainsi, les directions de projection sont bien choisies et les matrices $\hat{\Sigma}^*_{\text{FIS}}$ et Σ^* sont presque égales.

3.2.2.3 Exemple d'estimation d'une espérance : paiement d'une option asiatique

Le dernier cas-test considéré est une application en mathématique financière, tirée de l'article [Kawai, 2018], où l'on cherche à estimer l'espérance d'une variable aléatoire, $E = \mathbb{E}_f(\phi_3(\mathbf{X}))$, représentant le paiement d'une option asiatique discrétisée, sous le modèle Black-Scholes. La fonction d'intérêt ϕ_3 est la suivante :

$$\phi_3 : \mathbf{x} = (x_1, \ldots, x_n) \mapsto e^{-rT} \left[\frac{S_0}{n} \sum_{i=1}^{n} \beta_i(\mathbf{x}) - K \right]_+ \tag{3.13}$$

où $[y]_+ = \max(y, 0)$, pour y un nombre réel, et pour tout $i = 1 \ldots n$:

$$\beta_i(\mathbf{x}) = \exp\left(\sum_{j=1}^{i} \left(r - \frac{\sigma^2}{2} \right) \frac{T}{n} + \sigma \sqrt{\frac{T}{n}} x_j \right).$$

Lors des simulations numériques, les constantes sont fixées comme dans [Kawai, 2018], où les auteurs testent la fonction en dimension $n = 16$: $S_0 = 50$, $r = 0.05$, $T = 0.5$, $\sigma = 0.1$, $K = 55$. Dans cette partie, on évaluera aussi l'estimation de E en dimension 40 et 100, dont les valeurs de références sont indiquées tableau 3.7. La matrice Σ^* n'est pas connue analytiquement et est estimée par Monte-Carlo avec un échantillon de très grande taille. La fonction ϕ_3 est différentiable

sur $\mathbb{R}^n$ privé de l'ensemble (de mesure nulle) $\mathcal{E} = \{\sum_{i=1}^n \beta_i(\mathbf{x}) = nK/S_0\}$. Son gradient (au sens faible) est alors donné, pour tout $\mathbf{x}$ n'appartenant pas à $\mathcal{E}$, par

$$\nabla\phi_3(\mathbf{x}) = e^{-rT}\frac{S_0}{n}\sigma\sqrt{\frac{T}{n}}\left(\sum_{i=1}^n \beta_i(\mathbf{x}), \sum_{i=2}^n \beta_i(\mathbf{x}), \ldots, \beta_n(\mathbf{x})\right)^\top \mathbb{I}_{\sum_{i=1}^n \beta_i(\mathbf{x}) > nK/S_0}.$$

La matrice $\hat{H}$ (3.12) obtenue à l'aide de ce gradient fournit une seule direction de projection quelle que soit la dimension. Une nouvelle fois, cette direction est égale (à moins de 10^{-3} près) à la direction donnée par la matrice H (estimée avec un échantillon de très grande taille). Cette dernière est représentée en dimension 16 sur la figure 3.5 (à gauche). Ses valeurs propres sont toutes presque nulles ($< 10^{-3}$) sauf une qui vaut environ entre 50 et 130 suivant la dimension, c'est pourquoi une seule direction de projection est sélectionnée. Les coordonnées du vecteur propre associé à la plus grande valeur propre sont indiquées à droite de la figure 3.5, toujours pour la dimension 16. Les résultats de simulation de $\hat{\Sigma}^*_{\text{FIS}}$ sont présentés tableau 3.7 avec ceux des matrices optimale et empirique.

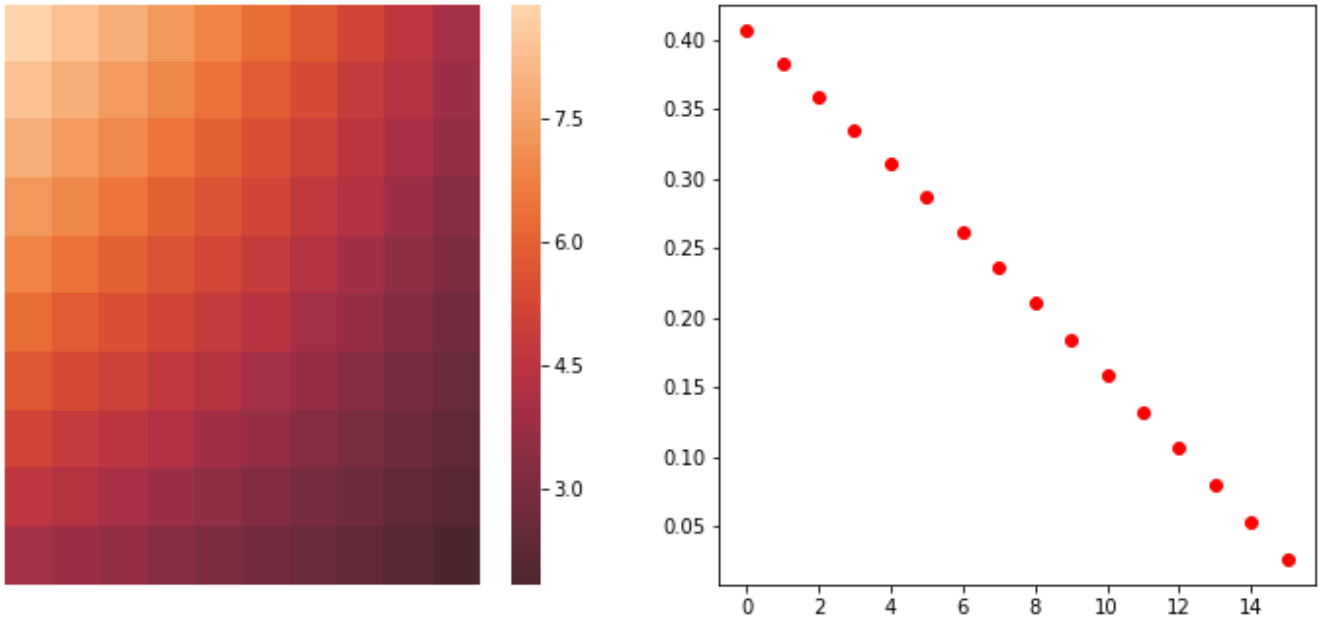


Figure 3.5 – Représentation de la matrice H (3.9) (à gauche) pour la fonction ϕ_3 en dimension $n = 16$, et les coordonnées de son vecteur propre (à droite) correspondant à la plus grande valeur propre.

Si l'estimation avec $\hat{\Sigma}^*$ est assez précise en dimension 16 (coefficient de variation de 4%), elle ne cesse de se dégrader en dimension 40, puis 100 (coefficients de variation de 27% et 91% respectivement). À l'inverse, la matrice $\hat{\Sigma}^*_{\text{FIS}}$ est performante dans toutes les dimensions, avec une divergence D' toujours à moins de 1% de la valeur optimale et un coefficient de variation autour de 2.5%. Ces résultats restent proches de ceux donnés par la matrice optimale Σ^*.

Ainsi, dans tous les exemples considérés, la projection sur le FIS permet d'approcher très précisément la matrice optimale et entraine une faible erreur d'estimation en grande dimension. Les résultats donnés par la matrice $\hat{\Sigma}^*_{\text{FIS}}$ sont même similaires à ceux de Σ^*.

		Σ^*	$\hat{\Sigma}^*$	$\hat{\Sigma}^*_{\text{FIS}}$
$n = 16$	$D'(\Sigma)$	14.2	15.1	14.3
	Erreur relative (%)	0	5.9	0.4
$E = 2.45 \cdot 10^{-2}$	Estimation moyenne ($\times 10^{-2}$)	2.45	2.46	2.45
	Coefficient de variation (%)	1.6	4.0	2.7
$n = 40$	$D'(\Sigma)$	38.1	43.6	38.3
	Erreur relative (%)	0	14.3	0.4
$E = 2.04 \cdot 10^{-2}$	Estimation moyenne ($\times 10^{-2}$)	2.03	1.86	2.04
	Coefficient de variation (%)	1.5	27	2.4
$n = 100$	$D'(\Sigma)$	97.5	137.8	98.4
	Erreur relative (%)	0	41	0.9
$E = 1.87 \cdot 10^{-2}$	Estimation moyenne ($\times 10^{-2}$)	1.88	0.231	1.88
	Coefficient de variation (%)	3.1	91	2.6

Tableau 3.7 – Comparaison numérique de la divergence D' et de l'estimation de E pour les matrices Σ^*, $\hat{\Sigma}^*$, $\hat{\Sigma}^*_{\text{FIS}}$ lorsque $\phi = \phi_3$ la fonction représentant le paiement d'une option asiatique (3.13).

Remarque 3.2.1. Dans le même esprit que cette approche, [Zahm et al., 2018] suggèrent également d'utiliser d'autres techniques de réduction de dimension basées sur le gradient et notamment la méthode des "*Active Subspaces*" ou sous-espaces actifs (développée par [Constantine, 2015]). La construction du sous-espace actif d'une fonction ϕ repose sur le calcul des vecteurs propres de la matrice :

$$C = \int_{\mathbb{R}^n} \nabla\phi(\mathbf{x})\nabla\phi(\mathbf{x})^\top f(\mathbf{x})d\mathbf{x},$$

où la densité de probabilité est f alors que c'est la densité optimale g^* pour H. [Zahm et al., 2018] proposent ainsi de calculer l'*Active Subspace* de la fonction "$\ln\phi$" afin de réduire la dimension du problème. Cette méthode permet en effet d'améliorer l'estimation, cependant elle ne garantit pas la minimisation, ou simplement le contrôle de la divergence de Kullback-Leibler, contrairement à l'approche présentée ici.

3.3 Conclusion

Dans ce chapitre, nous avons montré que projeter les paramètres dans un sous-espace de petite dimension pouvait diminuer la divergence de Kullback-Leibler et donc l'erreur d'estimation par échantillonnage préférentiel. Les simulations numériques ont montré que choisir des directions de projection aléatoires ou canoniques dans lesquelles estimer la matrice de covariance apportait souvent une amélioration de l'estimation de l'espérance par rapport à la matrice empirique. Il semble donc que même sans connaître les directions optimales de projection, projeter pour réduire la dimension permet d'améliorer la précision de l'estimation. Malgré tout, trouver des directions adaptées à chaque cas-test permettrait d'encore améliorer les performances d'estimation par échantillonnage préférentiel. [Zahm et al., 2018] et [Uribe et al., 2021] proposent par exemple des directions de projection assurant le contrôle de la divergence KL, en construisant le *Failure-Informed Subspace*. Les résultats de simulations obtenus à l'aide du FIS sont très performants mais celui-ci repose sur la connaissance du gradient, qui peut être très couteux à évaluer et limite l'application de cette technique aux fonctions suffisamment régulières. L'objectif du chapitre suivant est donc

de proposer une méthode de projection sans hypothèse de différentiabilité, en prenant en compte l'information des paramètres (gaussiens) optimaux d'échantillonnage préférentiel.

Chapitre 4

Identification de directions de projection pour estimer la matrice de covariance

Sommaire

Projeter les paramètres dans un sous-espace, ou les estimer uniquement dans un petit nombre de directions, semble donner de meilleurs résultats d'estimation et diminuer la divergence de Kullback-Leibler qu'en estimant la totalité des paramètres, comme le montrent les simulations de la section 3.1.3 du chapitre 3. Cependant, le choix des directions de projection peut aussi avoir une grande influence sur la qualité de l'estimation et il est préférable que ce choix ne soit pas fait de manière totalement aléatoire ou arbitraire, mais qu'il prenne en compte les données du problème disponibles. Un exemple évoqué dans le chapitre 3 est la méthode de projection développée dans [Zahm et al., 2018] puis [Uribe et al., 2021] où les projections trouvées proviennent d'une majoration de la divergence KL et reposent sur l'estimation d'une matrice construite à partir du gradient de la fonction d'intérêt. Cette approche est très performante mais elle n'est pas toujours applicable sachant que le gradient n'est pas toujours disponible, peut être couteux à évaluer, ou

même ne pas exister. L'objectif de ce chapitre est donc d'abord de déterminer des directions de projection, déduites directement des paramètres et ne nécessitant pas d'hypothèse de différentiabilité, puis de tester numériquement l'efficacité de ces directions sur différents cas-tests analytiques.

4.1 Identification de directions de projection pour estimer la matrice de covariance optimale sans utiliser le gradient

4.1.1 Définition du cadre numérique

Pour commencer, on reprend les notations de la section 3.1.2, où on cherche à estimer une intégrale E, en grande dimension, par échantillonnage préférentiel. Les paramètres gaussiens minimisant la divergence de Kullback-Leibler avec g^* (la densité IS optimale théorique) sont notés $\mathbf{m}^*$ et Σ^*, et on souhaite estimer la matrice $\Sigma_k = \sum_{i=1}^{k}(v_i - 1)\mathbf{d}_i\mathbf{d}_i^\top + I_n$ (3.4), à la place de Σ^*. La matrice Σ_k nécessite l'estimation des coefficients de Σ^* uniquement dans certaines directions. La question du choix de ces directions $\mathbf{d}_i$, et des coefficients v_i se pose alors. Nous allons donc chercher des directions influentes permettant d'approcher Σ^* et les tester sur différents cas d'estimation d'une espérance.

De manière analogue à l'algorithme 6 de la section 3.1.3, on propose ici de suivre la procédure 8.

Algorithme 8 : Estimation de E avec la matrice Σ_k où les $\hat{\mathbf{d}}_i$ sont déduits des paramètres

Données : Tailles des échantillons N et M
Résultat : Estimation de $\hat{E}_N$ de l'integrale E

1. Générer un échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_M$ selon g^* pour estimer $\mathbf{m}^*$ et Σ^* par $\hat{\mathbf{m}}^*$ et $\hat{\Sigma}^*$ (1.11) ;
2. Sélectionner k vecteurs $\hat{\mathbf{d}}_1, \ldots, \hat{\mathbf{d}}_k$ déduits des paramètres estimés ;
3. Calculer les $\hat{v}_i = \hat{\mathbf{d}}_i^\top \hat{\Sigma}^* \hat{\mathbf{d}}_i$, pour $i = 1 \ldots k$, et la matrice $\hat{\Sigma}_k$ définie en (3.4), avec $v_i = \hat{v}_i$ et $\mathbf{d}_i = \hat{\mathbf{d}}_i$. ;
4. Générer un nouvel échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_N$ selon $g_{\hat{\mathbf{m}}^*, \hat{\Sigma}_k}$ et estimer $\hat{E}_N$ (1.4).

Comme précédemment, on suppose qu'on sait échantillonner selon g^* pour estimer les paramètres optimaux, le but étant avant tout de tester l'efficacité des directions de projection (voir justification section 3.1.3). Le point 2 de l'algorithme 8 est développé dans les sections suivantes, où l'on définira des directions influentes pour l'estimation de la variance.

4.1.2 Identification d'une première direction influente : la moyenne optimale

Dans la première idée de projection que nous suggérons, nous nous plaçons dans le cadre d'estimations de probabilités d'événements rares. De plus, nous supposerons que la moyenne optimale $\mathbf{m}^*$ est non nulle, ajouté à l'hypothèse d'unimodalité déjà évoquée au début du manuscrit.

Une bonne direction dans laquelle estimer des paramètres de variance est une direction où la variance est significativement différente de 1. En effet, si la projection de la matrice de covariance Σ^* dans une certaine direction vaut 1, cela signifie que la variance est identique à celle de la densité

initiale dans cette direction et donc qu'elle n'a pas besoin d'être estimée. Dans le cas particulier des événements rares, nous avons remarqué dans divers exemples que la variance diminuait dans la direction donnée par le vecteur $\mathbf{m}^*$. Pour illustrer ce phénomène, regardons sur un exemple en deux dimensions, avec la fonction somme des coordonnées (2.1). La figure 4.1 représente un échantillon selon la loi d'origine f (gaussienne standard) et un échantillon selon la loi gaussienne optimale $g_{\mathbf{m}^*,\Sigma^*}$. Ce dernier est situé autour de la limite de la zone de défaillance, avec une faible variance dans la direction du vecteur $\mathbf{m}^*$ (représenté par la flèche noire), alors que dans la direction orthogonale à $\mathbf{m}^*$ la variance a peu changé par rapport à la variance initiale. C'est pourquoi nous suggérons de mettre à jour la variance dans la direction donnée par $\mathbf{m}^*$.

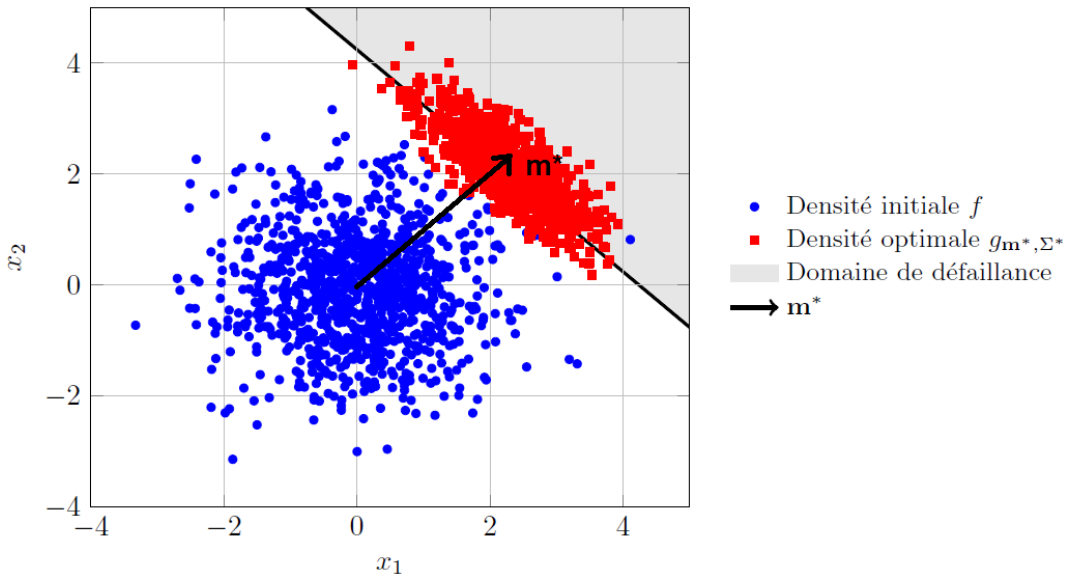


Figure 4.1 – Échantillon généré selon f (cercles bleus), et échantillon généré selon $g_{\mathbf{m}^*,\Sigma^*}$ (carrés rouges) pour estimer la probabilité $\mathbb{P}_f(\varphi_1(\mathbf{X}) \geq 0)$, avec φ_1 la fonction somme des coordonnées (2.1), en dimension $n = 2$. Le domaine de défaillance est situé au-dessus de la droite (d'équation $x_2 = 3\sqrt{2} - x_1$) et la flèche représente le vecteur $\mathbf{m}^*$.

Ce choix est justifié par la propriété de queue légère de la loi normale. En effet, dans le cadre des événements rares, rappelons que la loi optimale d'échantillonnage préférentiel (de densité g^*) est la loi de $\mathbf{X}$ sachant $\varphi(\mathbf{X}) \geq 0$, avec $\mathbf{X}$ supposé de loi normale centrée réduite. Si l'on considère le cas simple de dimension 1, avec X une variable aléatoire $\mathcal{N}(0,1)$ et S un réel positif fixé, la variable conditionnelle $X \mid X \geq S$ a une variance qui tend vers 0 lorsque S tend vers l'infini. On peut en effet montrer que cette variance vaut approximativement $1/S^2$ pour S grand. Les points défaillants étant loin de l'origine, ils ont une faible variance (du fait de la queue légère de la gaussienne), et comme ils s'éloignent de l'origine dans la direction de $\mathbf{m}^*$, leur variance décroit en particulier dans cette direction.

Ainsi, pour estimer une probabilité d'événement rare, une première idée de projection pour l'estimation de la matrice de covariance est donnée par la direction du vecteur moyenne $\mathbf{m}^*$. Autrement dit, on suggère ici d'évaluer la matrice Σ_k (3.4) avec $k = 1$ et $\mathbf{d}_1 = \mathbf{m}^*/\|\mathbf{m}^*\|$. On testera l'efficacité de cette technique en section 4.2 grâce à l'algorithme 8 avec $\hat{\mathbf{d}}_1 = \hat{\mathbf{m}}^*/\|\hat{\mathbf{m}}^*\|$ à l'étape 2. On verra également qu'elle peut être appliquée plus généralement à l'estimation d'une espérance, tant que $\mathbf{m}^* \neq \mathbf{0}$.

Cependant, si les résultats numériques 4.2 montrent une nette amélioration de l'estimation, la direction de projection proposée n'est pas optimale. On peut le voir par exemple pour l'estimation de la probabilité $\mathbb{P}_f(\varphi_4(\mathbf{X}) \geq 0)$ avec

$$\varphi_4 : \mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n \mapsto x_1 - 25x_2^2 - 3 \tag{4.1}$$

où $\mathbf{m}^*$ est colinéaire à $\mathbf{e}_1$ alors qu'il faudrait diminuer la variance selon $\mathbf{e}_2$ avant tout. Les échantillons selon f et selon $g_{\mathbf{m}^*,\Sigma^*}$ sont représentés figure 4.2. Le domaine de défaillance étant l'intérieur d'une étroite parabole, la variance des échantillons doit être suffisamment faible selon l'axe x_2 pour qu'il y ait assez de points dans la zone de défaillance. On remarque toutefois que dans la direction de $\mathbf{m}^*$ la variance a aussi diminué par rapport à la variance initiale, pour les raisons évoquées précédemment.

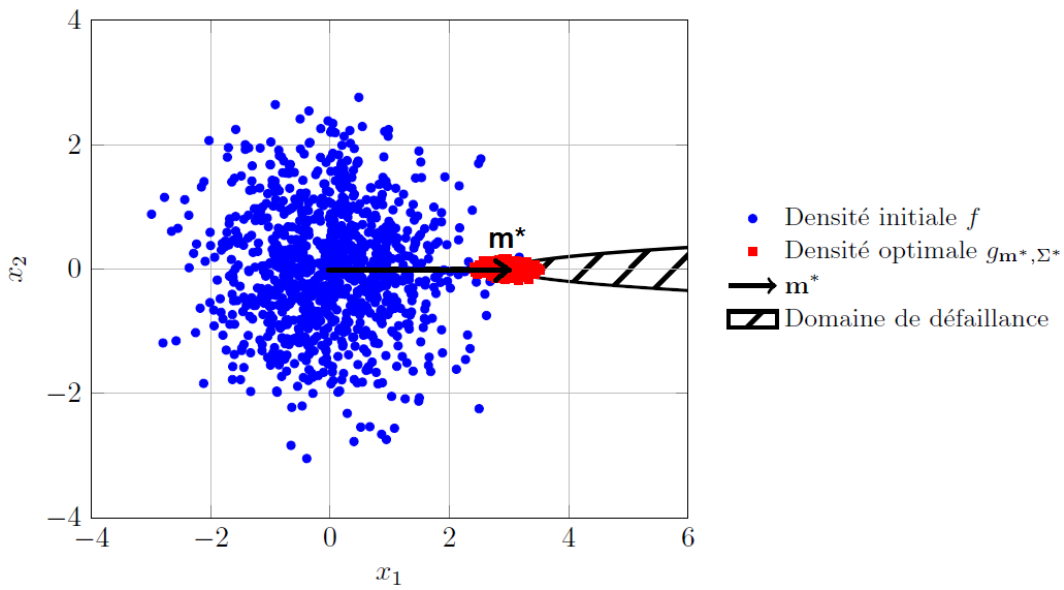


Figure 4.2 – Échantillon généré selon f (cercles bleus), et échantillon généré selon $g_{\mathbf{m}^*,\Sigma^*}$ (carrés rouges) pour estimer la probabilité $\mathbb{P}_f(\varphi_4(\mathbf{X}) \geq 0)$, avec φ_4 la fonction définie en 4.1, en dimension $n = 2$. La région de défaillance est l'intérieur de la parabole (d'équation $\varphi_4(\mathbf{x}) = 0$) et la flèche représente le vecteur $\mathbf{m}^*$.

Pour ce type de fonction, la projection selon $\mathbf{m}^*$ n'est donc pas optimale, et il serait intéressant de pouvoir projeter en plus d'une dimension. Dans la partie suivante, on va donc proposer des directions de projection optimales pour la matrice Σ_k, qui de plus, ne nécessitent pas de supposer $\mathbf{m}^* \neq \mathbf{0}$.

4.1.3 Identification des directions optimales par minimisation de la divergence de Kullback-Leibler

En voulant approcher Σ^* par Σ_k (3.4), on modifie la famille paramétrique considérée lorsqu'on veut minimiser la divergence de Kullback-Leibler entre g^* et $g_{\mathbf{m}^*,\Sigma}$. À l'origine, la covariance de la loi normale est mise à jour dans l'espace des matrices symétriques définies positives $\mathcal{S}_n^+$, alors que les matrices du type Σ_k appartiennent au sous-ensemble des matrices de la forme :

$$\mathcal{L}_{n,k} = \left\{ \sum_{i=1}^{k} (\alpha_i - 1) \frac{\mathbf{d}_i \mathbf{d}_i^\top}{\|\mathbf{d}_i\|^2} + I_n : \alpha_1, \ldots, \alpha_k > 0 \text{ et les } \mathbf{d}_i \in \mathbb{R}^n \text{ sont orthogonaux} \right\}.$$

L'espace dans lequel on recherche la matrice optimale est donc réduit, et le nombre de paramètres à estimer passe de $n(n+1)/2$ (dans $\mathcal{S}_n^+$) à $k(n+1)$ (dans $\mathcal{L}_{n,k}$). Dans un premier temps, on considère l'entier k comme fixé préalablement, mais on décrira par la suite une manière de le déterminer en fonction des paramètres. On cherche alors à résoudre le problème de minimisation suivant :

$$\Sigma_k^* = \arg\min \{ D(g^*, g_{\mathbf{m}^*,\Sigma}) : \Sigma \in \mathcal{L}_{n,k} \} \tag{4.2}$$

en sachant que la moyenne optimale est $\mathbf{m}^*$ (estimée par $\hat{\mathbf{m}}^*$). L'expression analytique de la matrice Σ_k^* est présentée dans le théorème 4.1.1, qui donne les directions optimales de projection de la matrice de covariance. L'énoncé de ce résultat nécessite la définition de la fonction ℓ suivante, représentée figure 4.3 :

$$\ell : x \in \mathbb{R}_+^* \mapsto \ln(x) - x + 1. \tag{4.3}$$

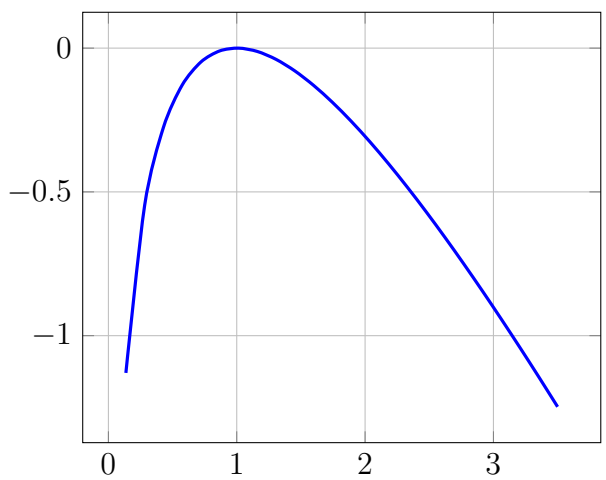


Figure 4.3 – Graphe de la fonction $\ell(x) = \ln(x) - x + 1$ (4.3).

Théorème 4.1.1. Soient $\lambda_1^*, \ldots, \lambda_n^*$ les valeurs propres de la matrice Σ^* telles que $\ell(\lambda_1^*) \leq \ldots \leq \ell(\lambda_n^*)$, et $\mathbf{d}_i^*$ les vecteurs propres associés. Alors pour $1 \leq k \leq n$, la solution Σ_k^* de (4.2) est donnée par

$$\Sigma_k^* = I_n + \sum_{i=1}^{k} (\lambda_i^* - 1) \frac{\mathbf{d}_i^* (\mathbf{d}_i^*)^\top}{\|\mathbf{d}_i^*\|^2}. \tag{4.4}$$

Démonstration. Comme on l'a déjà évoqué dans le chapitre 2, le problème (4.2) est équivalent au problème de minimisation de $D'(\Sigma)$ pour $\Sigma \in \mathcal{L}_{n,k}$, où $D'(\Sigma) = \ln\det\Sigma + \operatorname{tr}(\Sigma^*\Sigma^{-1})$ est défini en (2.2). Dans le reste de la preuve, on considère D' comme une fonction de $\mathbf{v} = (v_1, \dots, v_k) \in]0, \infty[^k$ et $\mathbf{d} = (\mathbf{d}_1, \dots \mathbf{d}_k)$, matrice de vecteurs orthogonaux, où $\Sigma = \sum_{i=1}^k (v_i - 1)\mathbf{d}_i\mathbf{d}_i^\top / \|\mathbf{d}_i\|^2 + I_n$.

- 1ère *étape : calcul de* $D'(\Sigma)$. Le but est de montrer que :

$$D'(\Sigma) = D'(\mathbf{v}, \mathbf{d}) = \sum_{i=1}^k \left[\ln(v_i) + \left(\frac{1}{v_i} - 1\right)\Psi(\mathbf{d}_i)\right] + \operatorname{tr}(\Sigma^*) \tag{4.5}$$

où $\Psi(\mathbf{x}) = \mathbf{x}^\top\Sigma^*\mathbf{x}/\|\mathbf{x}\|^2$ (Ψ est appelé le quotient de Rayleigh de la matrice Σ^*). Pour cela, notons d'abord que $\Sigma = Q\Delta Q^\top$, avec $\Delta = \operatorname{diag}(v_1, \dots, v_k, 1, \dots, 1)$ une matrice diagonale et Q une matrice orthogonale dont les k premières colonnes sont les $\mathbf{d}_i/\|\mathbf{d}_i\|$, $i = 1 \dots k$. En effet, on a $\Sigma\mathbf{d}_i = v_i\mathbf{d}_i$, et donc $\mathbf{d}_i$ est un vecteur propre associé à la valeur propre v_i. De plus, pour tout $\mathbf{d}_\perp$ dans l'espace orthogonal de $\operatorname{Vect}(\mathbf{d})$, on a $\Sigma\mathbf{d}_\perp = \mathbf{d}_\perp$, si bien que 1 est valeur propre de multiplicité $n - k$ (ou plus si d'autres v_i valent 1).

On en déduit que $\det(\Sigma) = v_1 \cdots v_k$, et on a la première moitié de l'égalité. D'autre part,

$$\operatorname{tr}\left(\Sigma^*\Sigma^{-1}\right) = \operatorname{tr}\left(\Sigma^* Q\Delta^{-1}Q^\top\right) = \operatorname{tr}\left(\Delta^{-1}Q^\top\Sigma^* Q\right).$$

Puisque les premières colonnes de Q sont les $\mathbf{d}_i/\|\mathbf{d}_i\|$, les premiers coefficients diagonaux de $Q^\top\Sigma^* Q$ valent $\Psi(\mathbf{d}_i)$. Ainsi, si $\mathbf{d}_{k+1}/\|\mathbf{d}_{k+1}\|, \dots, \mathbf{d}_n/\|\mathbf{d}_n\|$ complètent les $\mathbf{d}_i/\|\mathbf{d}_i\|$, $i = 1 \dots k$, en une base orthonormale, on obtient

$$\operatorname{tr}\left(\Delta^{-1}Q^\top\Sigma^* Q\right) = \sum_{i=1}^k \frac{1}{v_i}\Psi(\mathbf{d}_i) + \sum_{i=k+1}^n \Psi(\mathbf{d}_i) = \sum_{i=1}^k \left(\frac{1}{v_i} - 1\right)\Psi(\mathbf{d}_i) + \sum_{i=1}^n \Psi(\mathbf{d}_i).$$

Finalement, comme la dernière somme $\sum_{i=1}^n \Psi(\mathbf{d}_i)$ est égale à $\operatorname{tr}(Q^\top\Sigma^* Q) = \operatorname{tr}(\Sigma^*)$, on retombe bien sur l'égalité (4.5).

- 2ème *étape : minimisation.* La dérivée de (4.5) par rapport à v_i est :

$$\frac{\partial D'}{\partial v_i}(\mathbf{v}, \mathbf{d}) = \frac{1}{v_i} - \frac{1}{v_i^2}\Psi(\mathbf{d}_i) = \frac{1}{v_i^2}\left(v_i - \Psi(\mathbf{d}_i)\right).$$

Donc à $\mathbf{d}$ fixé, D' est décroissante en v_i pour $v_i < \Psi(\mathbf{d}_i)$ puis croissante pour $v_i > \Psi(\mathbf{d}_i)$, ce qui montre que D' est minimale en $v_i = \Psi(\mathbf{d}_i)$. Ainsi, en posant $\mathbf{v}^* = (\Psi(\mathbf{d}_1), \dots, \Psi(\mathbf{d}_k))$ on obtient

$$D'(\mathbf{v}^*, \mathbf{d}) = \sum_{i=1}^k \left[\ln(\Psi(\mathbf{d}_i)) + 1 - \Psi(\mathbf{d}_i)\right] + \operatorname{tr}(\Sigma^*) = \sum_{i=1}^k \ell(\Psi(\mathbf{d}_i)) + \operatorname{tr}(\Sigma^*). \tag{4.6}$$

Finalement, pour minimiser D', il faut minimiser la fonction ℓ. Cette fonction étant d'abord croissante puis décroissante, sa plus petite valeur est atteinte pour le $\mathbf{d}_i$ qui soit minimise, soit maximise Ψ (suivant la valeur qui minimise ℓ). D'après la caractérisation variationnelle des valeurs propres (ou le théorème de Courant-Fischer), les solutions $\mathbf{d}_i^*$ de ce problème sont exactement les vecteurs propres de Σ^*, et les $\Psi(\mathbf{d}_i^*)$ sont les valeurs propres associées, qui est le résultat attendu. □

Le théorème 4.1.1 nous dit par exemple, pour $k = 1$, que $\Sigma_1^* = I_n + (\lambda_1^* - 1)\mathbf{d}_1^*(\mathbf{d}_1^*)^\top / \|\mathbf{d}_1^*\|^2$ où λ_1^* est la valeur propre de Σ^* minimisant ℓ (la plus petite, ou la plus grande), et $\mathbf{d}_1^*$ est le vecteur propre associé.

En pratique, étant donné $\hat{\Sigma}^*$, le théorème suggère alors de calculer ses valeurs propres $\hat{\lambda}_i^*$, de les ranger de sorte que $\ell(\hat{\lambda}_1^*) \leq \cdots \leq \ell(\hat{\lambda}_n^*)$ puis de choisir les k premières valeurs propres, et les vecteurs propres associés $\hat{\mathbf{d}}_i^*$, pour construire la matrice $\hat{\Sigma}_k^* = \sum_{i=1}^k (\hat{\lambda}_i^* - 1)\hat{\mathbf{d}}_i^*(\hat{\mathbf{d}}_i^*)^\top / \|\hat{\mathbf{d}}_i^*\|^2 + I_n$.

Reste à savoir comment choisir le nombre de dimensions retenues, k. S'il est trop proche de n, la matrice $\hat{\Sigma}_k^*$ ressemblera à la matrice $\hat{\Sigma}^*$ qui est souvent mal estimée et que l'on veut justement améliorer. À l'inverse, toujours prendre $k = 1$ semble trop restrictif et des directions importantes peuvent être oubliées. Pour le déterminer, on propose une méthode basée sur la valeur de la divergence KL. Étant données les valeurs propres $\hat{\lambda}_i^*$, telles que $\ell(\hat{\lambda}_1^*) \leq \cdots \leq \ell(\hat{\lambda}_n^*)$, on cherche l'écart maximal dans la suite $(\ell(\hat{\lambda}_1^*), \ldots, \ell(\hat{\lambda}_n^*))$. Le nombre k ainsi choisi permet d'avoir $\sum_{i=1}^k \ell(\hat{\lambda}_i^*)$ proche de $\sum_{i=1}^n \ell(\hat{\lambda}_i^*)$, qui est égal au minimum de la divergence KL, à une constante près. Cette méthode est décrite dans l'algorithme 9.

Algorithme 9 : Choix du nombre de dimensions k

Données : n nombres positifs $\lambda_1, \ldots, \lambda_n$ tels que $\ell(\lambda_1) \leq \cdots \leq \ell(\lambda_n)$
Résultat : Nombre de dimensions sélectionnées k

1 Calculer les écarts $\delta_i = \ell(\lambda_{i+1}) - \ell(\lambda_i)$ pour $i = 1 \ldots n-1$;
2 Déterminer $k = \arg\max \delta_i$, l'indice du maximum des δ_i.

On obtient alors l'algorithme 10 qui est une réécriture de l'algorithme 8 en prenant en compte la méthode d'estimation de Σ_k^* suggérée par le théorème 4.1.1, et le choix de la dimension donné par l'algorithme 9.

Algorithme 10 : Algorithme suggéré par le Théorème 4.1.1.

Données : Tailles des échantillones N et M
Résultat : Estimation $\hat{E}_N$ de l'integrale E

1 Générer un échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_M$ de $\mathbb{R}^n$ indépendamment selon g^* ;
2 Estimer $\hat{\mathbf{m}}^*$ et $\hat{\Sigma}^*$ (1.11) avec cet échantillon ;
3 Calculer les éléments propres $(\hat{\lambda}_i^*, \hat{\mathbf{d}}_i^*)$ de $\hat{\Sigma}^*$, et les ranger dans l'ordre suivant : $\ell(\hat{\lambda}_1^*) \leq \cdots \leq \ell(\hat{\lambda}_n^*)$;
4 Calculer la matrice $\hat{\Sigma}_k^* = \sum_{i=1}^k (\hat{\lambda}_i^* - 1)\hat{\mathbf{d}}_i^*(\hat{\mathbf{d}}_i^*)^\top + I_n$ avec k obtenu par l'Algorithme 9 avec en entrée $(\hat{\lambda}_1^*, \ldots, \hat{\lambda}_n^*)$;
5 Générer un nouvel échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_N$ indépendamment selon $g_{\hat{\mathbf{m}}^*, \hat{\Sigma}_k^*}$;
6 Estimer $\hat{E}_N = \frac{1}{N} \sum_{i=1}^N \phi(\mathbf{X}_i) \frac{f(\mathbf{X}_i)}{\hat{g}(\mathbf{X}_i)}$ où $\hat{g} = g_{\hat{\mathbf{m}}^*, \hat{\Sigma}_k^*}$.

Cette méthode sera testée section 4.2 et comparée à celle décrite section 4.1.2 sur divers exemples, afin de mesurer l'efficacité de ces directions de projection.

4.2 Applications numériques

Cette partie est consacrée aux simulations numériques réalisées pour tester la performance des directions de projection proposées en 4.1.2 avec $\mathbf{m}^*$, et 4.1.3 avec les vecteurs propres de Σ^*. Pour

cela, nous allons comparer l'estimation $\hat{E}_N$ et la divergence de Kullback-Leibler, comme dans la section 3.1.3, pour les matrices Σ^* et $\hat{\Sigma}^*$, ainsi que les quatre autres matrices "projetées", de la forme $\sum_{i=1}^k (v_i - 1)\mathbf{d}_i\mathbf{d}_i^\top + I_n$ avec $v_i = \mathbf{d}_i^\top \hat{\Sigma}^* \mathbf{d}_i$, et définies comme suit :

- $\hat{\Sigma}^*_{\mathbf{d}}$ obtenue en choisissant $\mathbf{d}_i = \mathbf{d}_i^*$, vecteurs propres de Σ^*, supposés connus exactement.
- $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ obtenue en choisissant $\mathbf{d}_i = \hat{\mathbf{d}}_i^*$, vecteurs propres de $\hat{\Sigma}^*$ (estimations des $\mathbf{d}_i^*$).
- $\hat{\Sigma}^*_{\mathbf{m}}$ obtenue en choisissant $\mathbf{d}_1 = \mathbf{m}^*/\|\mathbf{m}^*\|$, supposé connu exactement.
- $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$ obtenue en choisissant $\mathbf{d}_1 = \hat{\mathbf{m}}^*/\|\hat{\mathbf{m}}^*\|$, où $\hat{\mathbf{m}}^*$ est l'estimation de $\mathbf{m}^*$.

Ces quatre matrices sont obtenues en projetant sur un des quatre choix suivants : $\mathbf{d}_i^*$ (du théorème 4.1.1) calculés de manière exacte, $\hat{\mathbf{d}}_i^*$ estimations des $\mathbf{d}_i^*$, $\mathbf{m}^*$ (comme suggéré dans la section 4.1.2) calculé de manière exacte, et $\hat{\mathbf{m}}^*$, estimation de $\mathbf{m}^*$. Les quatre façons de projeter sont résumées dans le tableau 4.1.

Direction	$\mathbf{d}_i^*$	$\mathbf{m}^*$
Exacte	$\hat{\Sigma}^*_{\mathbf{d}}$	$\hat{\Sigma}^*_{\mathbf{m}}$
Estimée	$\hat{\Sigma}^*_{\hat{\mathbf{d}}} = \hat{\Sigma}^*_k$	$\hat{\Sigma}^*_{\hat{\mathbf{m}}}$

Tableau 4.1 – Les quatre matrices de covariance estimées à l'aide d'une projection et de la forme $\sum_{i=1}^k (v_i - 1)\mathbf{d}_i\mathbf{d}_i^\top + I_n$ avec $v_i = \mathbf{d}_i^\top \hat{\Sigma}^* \mathbf{d}_i$. Les directions de projection considérées sont : $\mathbf{m}^*/\|\mathbf{m}^*\|$ calculée de manière exacte ou estimée, et les vecteurs $\mathbf{d}_i^*$ du théorème 4.1.1 (exacts ou estimés).

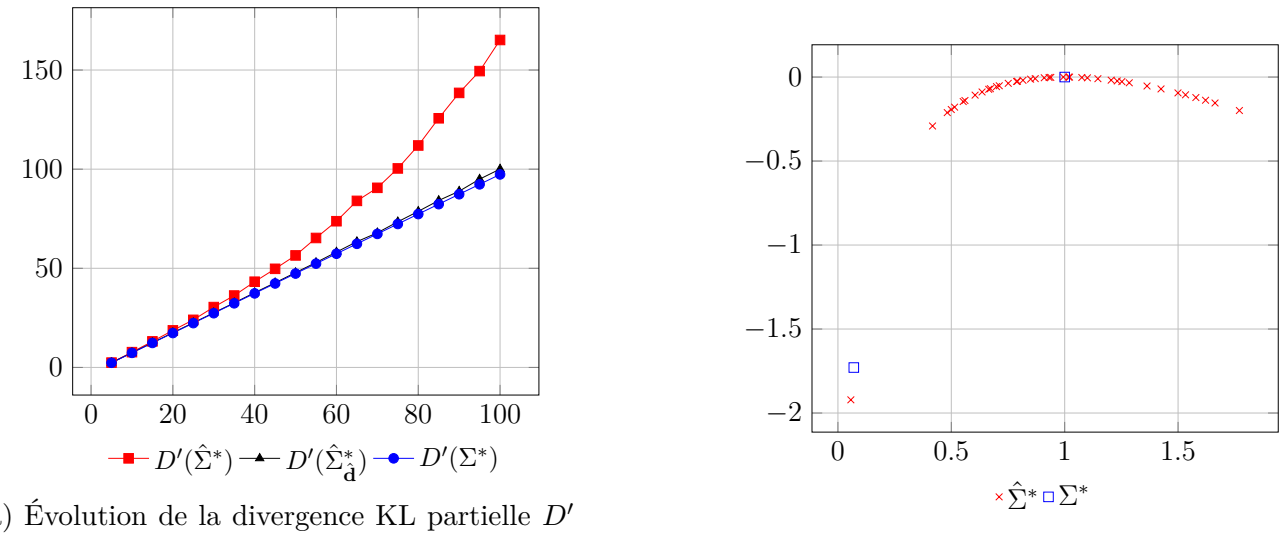
Pour $\hat{\Sigma}^*_{\mathbf{d}}$ et $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$, le nombre de directions k est déterminé par l'algorithme 9, et pour $\hat{\Sigma}^*_{\mathbf{m}}$ et $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$, on a toujours $k = 1$. Lorsque $\mathbf{m}^*$ et Σ^* (et donc ses vecteurs propres $\mathbf{d}_i^*$) ne peuvent pas être calculés analytiquement, ils sont estimés par Monte-Carlo avec un budget très important. Les matrices $\hat{\Sigma}^*_{\mathbf{d}}$ et $\hat{\Sigma}^*_{\mathbf{m}}$ estimées à partir des directions théoriques exactes ne sont pas disponibles en pratique et servent avant tout à évaluer l'efficacité de ces directions pour estimer l'intégrale. Les matrices $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ et $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$ permettent de tester l'efficacité des directions approchées et de vérifier l'influence de l'erreur d'estimation de ces directions. Notons enfin que la matrice $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ est en fait exactement la matrice $\hat{\Sigma}^*_k$ donnée par l'algorithme 10. Ce changement de notation est effectué pour rester cohérent avec les notations des autres matrices du tableau 4.1.

L'efficacité des six matrices est comparée dans les cinq exemples suivants, à travers les valeurs moyennes de la divergence KL partielle D' (2.2), et l'estimation de l'intégrale $\hat{E}_N$, regroupées dans les tableaux 4.2 à 4.6. Toutes les valeurs indiquées sont des moyennes calculées sur 50 réalisations des algorithmes 8 et 10 (suivant la matrice considérée), et les tailles d'échantillon sont fixées à $M = 500$, et $N = 2000$ (sauf mention explicite du contraire).

4.2.1 Exemple jouet dans le cas événement rare : somme de variables indépendantes

Le premier exemple considéré est l'estimation de la probabilité d'événement rare avec la fonction d'état limite φ_1 définie en 2.1. Les paramètres optimaux théoriques sont $\mathbf{m}^* = m^*\mathbf{1}_n$ et $\Sigma^* = (v^* - 1)\mathbf{1}_n\mathbf{1}_n^\top + I_n$, où les valeurs exactes de m^* et v^* sont données dans la partie 2.1.1.

La matrice optimale est déjà de la forme Σ_k^* (4.4), et ses valeurs propres sont $v^* < 1$ et 1 (de multiplicité $n-1$). La valeur propre v^* est associée au vecteur propre $\mathbf{d}_1^* = \mathbf{1}_n$, et minimise la fonction ℓ (voir figure 4.4b), ce qui suggère de prendre théoriquement $k = 1$. Par ailleurs, $\mathbf{m}^*$ étant colinéaire à $\mathbf{1}_n$, on a $\mathbf{d}_1^* = \mathbf{m}^*/\|\mathbf{m}^*\|$ et donc $\hat{\Sigma}_{\mathbf{d}}^* = \hat{\Sigma}_{\mathbf{m}}^*$. La matrice $\hat{\Sigma}_{\hat{\mathbf{d}}}^*$ est donnée par l'algorithme 10, où k vaut toujours 1 en pratique (ce qu'on peut déduire de la figure 4.4b en dimension 40), et $\hat{\mathbf{d}}_1^*$ est le vecteur propre associé à la valeur propre de $\hat{\Sigma}^*$ minimisant ℓ. Enfin, la matrice $\hat{\Sigma}_{\hat{\mathbf{m}}}^*$ vaut $(\hat{v}-1)\frac{\hat{\mathbf{m}}^*(\hat{\mathbf{m}}^*)^\top}{\|\hat{\mathbf{m}}^*\|^2} + I_n$, avec $\hat{v} = \frac{(\hat{\mathbf{m}}^*)^\top \hat{\Sigma}^* \hat{\mathbf{m}}^*}{\|\hat{\mathbf{m}}^*\|^2}$ et $\hat{\mathbf{m}}^*$ l'estimation de la moyenne optimale $\mathbf{m}^*$. Les résultats obtenus avec ces matrices sont donnés tableau 4.2 et figure 4.4.



(a) Évolution de la divergence KL partielle D' en fonction de la dimension, pour la matrice de covariance optimale Σ^* (cercles bleus), la covariance empirique $\hat{\Sigma}^*$ (carrés rouges), et la matrice avec projection $\hat{\Sigma}_{\hat{\mathbf{d}}}^*(= \hat{\Sigma}_k^*)$ (triangles noirs).

(b) Images des valeurs propres $\ell(\lambda_i)$ des matrices Σ^* (carrés bleus) et $\hat{\Sigma}^*$ (croix rouges) en dimension $n = 40$.

Figure 4.4 – Évolution de la divergence KL partielle et images des valeurs propres (par ℓ) pour l'estimation de la probabilité d'événement rare avec la fonction φ_1 (2.1).

La figure 4.4a représente l'évolution de la divergence D' en fonction de la dimension (allant de 5 à 100), pour les matrices Σ^*, $\hat{\Sigma}^*$ (estimée avec un échantillon selon g^* de taille $M = 200$), et $\hat{\Sigma}_{\hat{\mathbf{d}}}^*(= \hat{\Sigma}_k^*)$ (estimée à partir de $\hat{\Sigma}^*$ par l'algorithme 10). Notons que la divergence de la matrice $\hat{\Sigma}_{\hat{\mathbf{m}}}^*$ n'apparait pas car elle est en fait confondue avec celle de $\hat{\Sigma}_k^*$. Pour cette dernière, la quantité D' reste très proche de la valeur optimale quelle que soit la dimension, alors que la divergence augmente fortement avec la matrice empirique $\hat{\Sigma}^*$, comme on l'a déjà évoqué dans les chapitres précédents. Ce graphique suggère donc que l'on aura une meilleure estimation de la probabilité avec $\hat{\Sigma}_k^*$ qu'avec $\hat{\Sigma}^*$. La figure 4.4b donne une raison à cette amélioration. En effet, les valeurs propres de $\hat{\Sigma}^*$ sont assez mal estimées, car toutes les croix rouges (sauf la plus à gauche) sont censées estimer 1, alors qu'elles sont réparties presque uniformément entre 0.4 et 1.8. Cela signifie que les termes de variance dans les directions correspondantes sont mal estimés, et explique pourquoi utiliser $\hat{\Sigma}^*$ donne une estimation imprécise. Mais la fonction ℓ étant assez plate autour de 1, la grande variabilité des valeurs propres est atténuée par l'action de ℓ (les images de ces valeurs propres par ℓ étant comprises entre -0.4 et 0). De plus, comme ℓ croît fortement au voisinage de 0, elle permet clairement de distinguer la plus petite valeur propre estimée des suivantes. Cela justifie les bonnes performances des matrices $\hat{\Sigma}_{\mathbf{d}}^*$, $\hat{\Sigma}_{\hat{\mathbf{d}}}^*$, ainsi que $\hat{\Sigma}_{\mathbf{m}}^*$, et $\hat{\Sigma}_{\hat{\mathbf{m}}}^*$, visibles dans le tableau 4.2.

		Σ^*	$\hat{\Sigma}^*$	$\hat{\Sigma}^*_{\mathbf{m}} = \hat{\Sigma}^*_{\mathbf{d}}$	$\hat{\Sigma}^*_{\hat{\mathbf{d}}}$	$\hat{\Sigma}^*_{\hat{\mathbf{m}}}$
$n = 40$	$D'(\Sigma)$	37.4	39.3	37.4	37.5	37.4
	Erreur relative (%)	0	5.0	0.02	0.3	0.2
$E = 1.35 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.35	1.34	1.34	1.34	1.35
	Coefficient de variation (%)	2.7	9.4	2.7	2.5	2.0
$n = 70$	$D'(\Sigma)$	67.4	73.8	67.4	67.6	67.5
	Erreur relative (%)	0	9.5	0.01	0.3	0.2
$E = 1.35 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.35	1.33	1.34	1.35	1.35
	Coefficient de variation (%)	2.0	35	2.4	3.4	2.4
$n = 100$	$D'(\Sigma)$	97.4	111.9	97.4	97.7	97.6
	Erreur relative (%)	0	15.0	0.01	0.4	0.2
$E = 1.35 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.34	1.00	1.35	1.35	1.35
	Coefficient de variation (%)	2.5	90	2.3	5.1	3.7

Tableau 4.2 – Comparaison numérique de la divergence D' et de l'estimation de E pour les différentes matrices considérées section 4.2 et tableau 4.1, lorsque $\phi = \mathbb{I}_{\{\varphi \geq 0\}}$ avec $\varphi = \varphi_1$ la somme des variables indépendantes (2.1).

En projetant dans la direction optimale théorique ($\mathbf{d}_1^* = \mathbf{1}_n = \mathbf{m}^*/\|\mathbf{m}^*\|$), on observe (colonne $\hat{\Sigma}^*_{\mathbf{m}} = \hat{\Sigma}^*_{\mathbf{d}}$) que la divergence et l'estimation sont presque égales aux valeurs optimales (colonne Σ^*). Cela vient du fait que les matrices Σ^* et $\hat{\Sigma}^*_{\mathbf{m}}$ sont de la même forme, et déterminer $\hat{\Sigma}^*_{\mathbf{m}}$ nécessite uniquement l'estimation d'un paramètre de variance (égal à $\hat{v} = \mathbf{1}_n^\top \hat{\Sigma}^* \mathbf{1}_n$). Concernant les deux dernières colonnes du tableau, l'estimation de la direction de projection $\hat{\mathbf{1}}_n$ s'ajoute à celle de la variance $\hat{v}$, pour le calcul des matrices $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ et $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$. Celles-ci donnent malgré tout des résultats proches de la matrice optimale, et améliore significativement les performances obtenues par $\hat{\Sigma}^*$. En dimension 100, le coefficient de variation est légèrement supérieur pour les matrices $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$ et $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ (3.7% et 5.1% respectivement) que pour la matrice optimale (2.5%), du fait de cette double estimation (direction + variance), mais reste très inférieur à celui de la matrice empirique (90%).

On peut également noter que $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$ est légèrement plus précise que $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ quand la dimension grandit (coefficient de variation de 2.4% et 3.4% pour $n = 70$ respectivement, et de 3.7% contre 5.1% pour $n = 100$). Nous supposons que cela provient de l'estimation plus précise de $\hat{\mathbf{m}}^*$ comparé au vecteur propre $\hat{\mathbf{d}}_1^*$ de $\hat{\Sigma}^*$. En effet, pour évaluer $\hat{\mathbf{m}}^*$ on a besoin d'estimer n paramètres, alors que $\hat{\Sigma}^*$ nécessite l'estimation de plus de $n^2/2$ paramètres, et le vecteur propre est alors plus bruité que le vecteur moyenne.

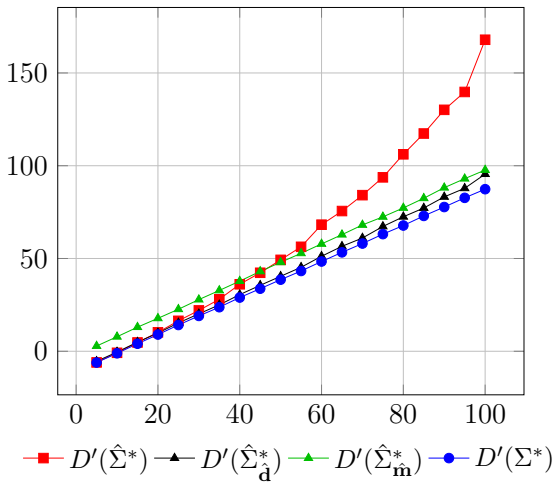
Finalement, les deux méthodes proposées permettent une nette amélioration de l'estimation en grande dimension. Dans cet exemple, où une projection en dimension 1 est suffisante, la projection sur le vecteur moyenne $\hat{\mathbf{m}}^*$ donne des résultats légèrement plus précis que pour le premier vecteur propre de $\hat{\Sigma}^*$. L'exemple suivant montre que projeter en plus d'une dimension peut être plus efficace que la projection sur le sous-espace de dimension 1 engendré par $\mathbf{m}^*$.

4.2.2 Exemple jouet dans le cas événement rare : un polynôme de degré 2

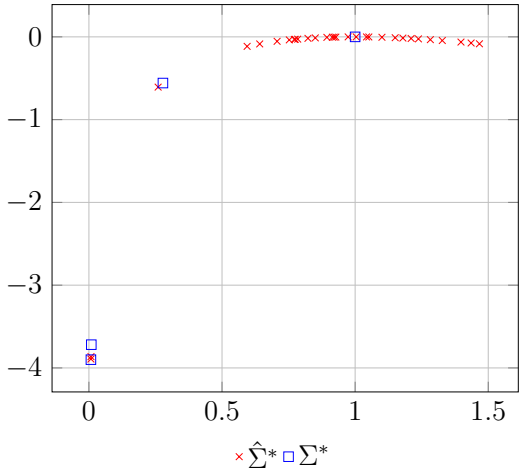
Le deuxième exemple jouet correspond à l'estimation de la probabilité d'événement rare avec la fonction d'état limite suivante :

$$\varphi_5 : \mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n \mapsto x_1 - 25x_2^2 - 30x_3^2 - 1. \tag{4.7}$$

Dans ce cas, la direction donnée par $\mathbf{m}^*$ est différente de $\mathbf{d}_1^*$ et l'algorithme 10 choisit deux directions différentes, ce qui n'était pas le cas dans l'exemple précédent. De ce fait, les matrices $\hat{\Sigma}^*_{\mathbf{d}}$ et $\hat{\Sigma}^*_{\mathbf{m}}$ sont différentes.



(a) Évolution de la divergence KL partielle D' en fonction de la dimension, pour la matrice de covariance optimale Σ^* (cercles bleus), la covariance empirique $\hat{\Sigma}^*$ (carrés rouges), et les matrices avec projection $\hat{\Sigma}^*_{\hat{\mathbf{d}}} (= \hat{\Sigma}^*_k)$ (triangles noirs), et $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$ (triangles verts).



(b) Images des valeurs propres $\ell(\lambda_i)$ des matrices Σ^* (carrés bleus) et $\hat{\Sigma}^*$ (croix rouges) en dimension $n = 30$.

Figure 4.5 – Évolution de la divergence KL partielle et images des valeurs propres (par ℓ) pour l'estimation de la probabilité d'événement rare avec la fonction φ_5 (4.7).

En effet, comme φ_5 dépend uniquement des trois premières variables et est paire en x_2 et x_3, on a $\mathbf{m}^* = m^*\mathbf{e}_1$ avec $m^* = \mathbb{E}(X_1 \mid X_1 \geq 25X_2^2 + 30X_3^2 + 1) \approx 1.9$ et

$$\Sigma^* = \begin{pmatrix} \lambda_1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \lambda_3 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 \end{pmatrix}.$$

Les coefficients de covariance de la sous-matrice $(\Sigma^*_{ij})_{1 \leq i,j \leq 3}$ sont nuls car ils sont égaux à l'intégrale d'une fonction impaire d'une variable de densité paire avec un conditionnement pair. Par exemple,

si $F(x) = \mathbb{P}(30\mathbf{X}_3^2 + 1 \leq x)$, alors en conditionnant en (X_1, X_2) on obtient :

$$\begin{aligned}\Sigma_{12}^* &= \mathbb{E}\left((X_1 - m^*)X_2 \mid X_1 - 25X_2^2 \geq 30X_3^2 + 1\right) \\ &= \frac{1}{E}\mathbb{E}\left[(X_1 - m^*)\mathbb{E}\left(X_2 F(X_1 - 25X_2^2) \mid X_1\right)\right]\end{aligned}$$

qui vaut bien 0 puisque $x_2 F(x_1 - x_2^2)$ est une fonction impaire de x_2, pour x_1 fixé, et X_2 a une densité paire. Les valeurs approchées des coefficients diagonaux sont $\lambda_1 \approx 0.278$, $\lambda_2 \approx 0.009$, $\lambda_3 \approx 0.0075$ et correspondent aux 3 plus petites valeurs propres indiquées par les carrés bleus sur la figure 4.5b.

		Σ^*	$\hat{\Sigma}^*$	$\hat{\Sigma}^*_{\mathbf{d}}$	$\hat{\Sigma}^*_{\mathbf{m}}$	$\hat{\Sigma}^*_{\hat{\mathbf{d}}}$	$\hat{\Sigma}^*_{\hat{\mathbf{m}}}$
$n = 30$	$D'(\Sigma)$	19.1	20.2	19.7	26.7	19.8	26.8
	Erreur relative (%)	0	5.5	2.9	39.9	3.7	40.1
$E = 1.51 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.52	1.54	1.49	1.43	1.51	1.45
	Coefficient de variation (%)	3.9	15.4	3.1	24.3	4.3	25.2
$n = 70$	$D'(\Sigma)$	59.1	65.6	59.7	66.7	60.1	66.8
	Erreur relative (%)	0	10.9	1.0	12.9	1.6	13.0
$E = 1.51 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.51	1.41	1.51	1.52	1.54	1.56
	Coefficient de variation (%)	3.1	32.6	3.2	25.5	5.0	32.5
$n = 100$	$D'(\Sigma)$	89.1	103.7	89.7	96.7	90.4	96.8
	Erreur relative (%)	0	16.4	0.6	8.6	1.4	8.7
$E = 1.51 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.50	1.09	1.50	1.51	1.49	1.48
	Coefficient de variation (%)	2.6	84	2.9	25.1	7.0	21.9

Tableau 4.3 – Comparaison numérique de la divergence D' et de l'estimation de E pour les différentes matrices considérées section 4.2 et tableau 4.1, lorsque $\phi = \mathbb{I}_{\{\varphi \geq 0\}}$ avec $\varphi = \varphi_5$ la fonction quadratique (4.7).

De plus, à l'aide de ce graphe on peut deviner que l'algorithme 9 sélectionne les 2 valeurs propres minimales (en dimension 40 sur la figure 4.5b, mais c'est aussi le cas en dimension 70 et 100). Celles-ci sont associées aux vecteurs propres $\mathbf{e}_2$ et $\mathbf{e}_3$ alors que $\mathbf{m}^*$ est proportionnelle à $\mathbf{e}_1$. Les directions suggérées par le théorème 4.1.1 sont donc bien différentes (et même orthogonales) de la moyenne optimale $\mathbf{m}^*$.

Cette différence se répercute sur la divergence de Kullback-Leibler comme on peut le voir sur la figure 4.5a. En effet, la matrice $\hat{\Sigma}^*_{\hat{\mathbf{d}}} (= \hat{\Sigma}^*_k)$ a une divergence très proche de la valeur optimale dans toutes les dimensions alors que $D'(\hat{\Sigma}^*_{\hat{\mathbf{m}}})$ est toujours légèrement supérieure. Néanmoins, les deux matrices projetées ont une divergence nettement inférieure à celle de $\hat{\Sigma}^*$ lorsque la dimension augmente.

Les résultats des simulations présentés dans le tableau 4.3 confirment les observations de la figure 4.5a. L'estimation se dégrade fortement avec la matrice $\hat{\Sigma}^*$ (84% de coefficient de variation en dimension 100), comme attendu. En revanche, la matrice $\hat{\Sigma}^*_{\mathbf{d}}$, avec les directions optimales exactes $\mathbf{e}_2$ et $\mathbf{e}_3$, donne des résultats très précis et presque identiques à la matrice optimale. Ce comportement est lié à l'évolution de la divergence KL, dont l'erreur relative passe de 5.5% en dimension $n = 30$ à 16.4% lorsque $n = 100$, pour la matrice $\hat{\Sigma}^*$, alors que l'erreur relative reste toujours en dessous de 3% pour $\hat{\Sigma}^*_{\mathbf{d}}$. De plus, cette dernière donne des résultats bien plus précis qu'en projetant sur $\mathbf{m}^*$ (coefficient de variation toujours autour de 3% contre environ 25% pour $\hat{\Sigma}^*_{\mathbf{m}}$), car

la direction donnée par la moyenne ($\mathbf{e}_1$) n'est pas optimale. Cependant, on peut noter que ne pas projeter est moins efficace en dimension 100 que projeter sur $\mathbf{m}^*$ (coefficient de variation de 84% pour $\hat{\Sigma}^*$ contre 25.1% pour $\hat{\Sigma}^*_{\mathbf{m}}$). Enfin, estimer les directions de projection dégrade légèrement la situation par rapport aux directions exactes. Le coefficient de variation passe en effet de 3.1% pour $\hat{\Sigma}^*_{\mathbf{d}}$ à 4.3% pour $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ en dimension 30, et de 2.9% à 7.0% en dimension 100. Concernant la matrice $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$, les résultats sont assez proches de ceux donnés par $\hat{\Sigma}^*_{\mathbf{m}}$, il semble donc que l'estimation de $\mathbf{m}^*$ soit assez précise jusqu'en dimension 100.

Encore une fois, on a pu constater l'efficacité des matrices du type Σ_k (3.4) pour l'estimation en grande dimension, comparé à la matrice empirique $\hat{\Sigma}^*$. Toutefois, la projection sur $\mathbf{m}^*$ n'étant pas optimale dans cet exemple, elle donne des résultats moins précis que lorsqu'on utilise les directions de projection définies dans le théorème 4.1.1, qu'elles soient exactes ou estimées.

4.2.3 Application en finance : probabilité de perte élevée d'un portefeuille

L'exemple qui suit est une application en finance de l'estimation d'événement rare, tirée de [Chan and Kroese, 2012] et [Bassamboo et al., 2008]. La probabilité recherchée est $E = \mathbb{P}(L(\mathbf{Z}) > 0)$, avec L la fonction de perte d'un portefeuille ("*portfolio loss function*") d'options financières, définie par :

$$L(\mathbf{z}) = \sum_{j=1}^{n} \mathbb{I}_{\{z_j \geq 0.5\sqrt{n}\}} - bn.$$

où b est choisi ici de sorte que la probabilité soit de l'ordre de 10^{-3} ($b = 0.45$ en dimension $n = 30$, $b = 0.3$ pour $n = 70$ et $b = 0.25$ pour $n = 100$). Les variables aléatoires Z_j sont dépendantes et données par :

$$Z_j = \left(qU + (1-q^2)^{1/2}\eta_j\right)\mu^{-1/2}$$

où $U \sim \mathcal{N}(0,1)$, $\eta_j \sim \mathcal{N}(0,9)$, $j = 1,\dots,n$, $\mu \sim \text{Gamma}(6,6)$ sont des variables indépendantes, et $q = 0.25$. Chaque Z_j est calculée avec la même réalisation de U et μ, c'est pourquoi elles sont dépendantes. Pour rester dans le cadre de variables gaussiennes standards, on pose $\eta_j = 3\widetilde{\eta}_j$, pour tout $j = 1,\dots n$ et $\mu = F_\Gamma^{-1}(F_{\mathcal{N}}(\widetilde{\mu}))$ avec $\widetilde{\eta}_j$, $\widetilde{\mu}$ des gaussiennes standards indépendantes et F_Γ, $F_{\mathcal{N}}$ les fonctions de répartition des lois Gamma$(6,6)$ et $\mathcal{N}(0,1)$ respectivement. De plus, en posant $X_1 = U, X_2 = \widetilde{\mu}$ et $(X_3,\dots,X_{n+2}) = \widetilde{\eta} \in \mathbb{R}^n$, on définit :

$$\varphi_6(\mathbf{X}) = \sum_{j=1}^{n} \mathbb{I}_{\{\Psi(U,\widetilde{\mu},\widetilde{\eta}_j) \geq 0.5\sqrt{n}\}} - bn \tag{4.8}$$

avec

$$\Psi(U,\widetilde{\mu},\widetilde{\eta}_j) = \left(qU + 3(1-q^2)^{1/2}\widetilde{\eta}_j\right)\left[F_\Gamma^{-1}\left(F_{\mathcal{N}}(\widetilde{\mu})\right)\right]^{-1/2}.$$

Ainsi la probabilité de pertes élevées $\mathbb{P}(L(\mathbf{Z}) > 0)$ peut être réécrite $\mathbb{P}(\varphi_6(\mathbf{X}) > 0)$, avec $\mathbf{X}$ un vecteur gaussien standard de dimension $n+2$. Les valeurs de référence de cette probabilité E sont rappelées dans le tableau 4.4 pour les dimensions $n = 30$, 70 et 100. Les paramètres optimaux $\mathbf{m}^*$ et Σ^* ne peuvent pas être calculées analytiquement, et sont estimées précisément par Monte-Carlo avec un budget très important. Numériquement, le premier vecteur propre $\mathbf{d}_1^*$ de Σ^* est indiscernable de la moyenne $\mathbf{m}^*$ (normalisée), et comme l'algorithme 9 sélectionne toujours une seule direction de projection ($k = 1$), on a $\hat{\Sigma}^*_{\mathbf{m}} = \hat{\Sigma}^*_{\mathbf{d}}$.

		Σ^*	$\hat{\Sigma}^*$	$\hat{\Sigma}^*_{\mathbf{m}} = \hat{\Sigma}^*_{\mathbf{d}}$	$\hat{\Sigma}^*_{\hat{\mathbf{d}}}$	$\hat{\Sigma}^*_{\hat{\mathbf{m}}}$
$n = 30$	$D'(\Sigma)$	33.4	34.6	33.6	33.7	33.7
	Erreur relative (%)	0	3.8	0.8	1.0	0.9
$E = 4.29 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	4.36	4.18	4.26	4.21	4.23
	Coefficient de variation (%)	11.3	19.9	9.7	11.1	10.3
$n = 70$	$D'(\Sigma)$	75.8	83.2	76.7	76.9	76.8
	Erreur relative (%)	0	9.7	1.1	1.4	1.2
$E = 2.36 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	2.35	3.21	2.37	2.31	2.36
	Coefficient de variation (%)	7.0	350	9.2	8.2	11.0
$n = 100$	$D'(\Sigma)$	106.0	122.3	107.3	107.7	107.4
	Erreur relative (%)	0	15.4	1.3	1.6	1.3
$E = 1.82 \cdot 10^{-3}$	Estimation moyenne ($\times 10^{-3}$)	1.79	1.15	1.87	1.84	1.81
	Coefficient de variation (%)	10.8	73	11.3	16.5	13.2

Tableau 4.4 – Comparaison numérique de la divergence D' et de l'estimation de E pour les différentes matrices considérées section 4.2 et tableau 4.1, lorsque $\phi = \mathbb{I}_{\{\varphi \geq 0\}}$ avec $\varphi = \varphi_6$ la fonction perte d'un portefeuille (4.8).

Les résultats du tableau 4.4 sont qualitativement similaires à ceux du premier exemple 4.2.1. La projection sur $\mathbf{d}_1^* = \mathbf{m}^*/\|\mathbf{m}^*\|$ améliore significativement l'estimation par rapport à la matrice $\hat{\Sigma}^*$ (sans projection), avec des coefficients de variation toujours proches de ceux obtenus avec la matrice optimale. Cette amélioration est toujours visible lorsqu'on estime les directions de projection $\hat{\mathbf{d}}_1^*$ et $\hat{\mathbf{m}}^*$, même si en dimension $n = 100$, $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ et $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$ ont un coefficient de variation légèrement supérieur à $\hat{\Sigma}^*_{\mathbf{d}}$ (respectivement 16.5%, 13.2% et 11.3%). De plus, $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$ est un peu plus efficace que $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ en dimension 100, au vu des coefficients de variation, car $\hat{\mathbf{m}}^*$ est un estimateur de $\mathbf{m}^* = \mathbf{d}_1^*$ plus précis que $\hat{\mathbf{d}}_1^*$, comme expliqué dans l'exemple 4.2.1.

4.2.4 Exemple jouet pour l'estimation de la constante de normalisation de la loi "*banana shape*"

On sort maintenant du cadre des événements rares et on revient sur l'exemple 3.1.3.3, où l'on veut échantillonner selon la loi en forme de banane, de densité π (2.3), et estimer sa constante de normalisation. La moyenne optimale étant nulle, on ne peut pas utiliser la méthode de projection sur $\mathbf{m}^*$, et on rappelle que la matrice de covariance optimale est $\Sigma^* = \mathrm{diag}(100, 19, 1, \ldots, 1)$. Avec cette matrice, l'algorithme 9, pour choisir le nombre de directions, ne permet pas de sélectionner le nombre optimal de dimensions. En effet, Σ^* a deux grandes valeurs propres (100 et 19) et toutes les autres sont égales à 1, donc l'idéal serait d'estimer les deux valeurs maximales. Cependant, la différence entre 100 et 19 étant nettement supérieure à celle entre 19 et 1 (et donc $\ell(19) - \ell(100)$ est largement plus grand que $\ell(1) - \ell(19)$), l'algorithme 9 sélectionne toujours une seule direction de projection ($\mathbf{d}_1^* = \mathbf{e}_1$, correspondant à la valeur propre 100). Les matrices $\hat{\Sigma}^*_{\mathbf{d}}$ et $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ ne sont donc pas optimales (elles s'approchent de la matrice $\mathrm{diag}(100, 1, \ldots, 1)$) mais apportent déjà une amélioration en grande dimension par rapport à $\hat{\Sigma}^*$, comme on peut l'observer dans le tableau 4.5. Le choix optimal du nombre de dimension étant $k = 2$, nous avons également effectué les simulations en imposant $k = 2$ dans l'algorithme 10 (une autre méthode de sélection est suggérée remarque 4.2.1 pour cet exemple). Les matrices ainsi construites sont notées $\hat{\Sigma}^*_{\mathbf{d}2}$ et $\hat{\Sigma}^*_{\hat{\mathbf{d}}2}$, et utilisent

les directions de projection $\mathbf{d}_1^*$ et $\mathbf{d}_2^*$ pour la première, et $\hat{\mathbf{d}}_1^*$ et $\hat{\mathbf{d}}_2^*$ pour la seconde. Les résultats numériques obtenus avec ces deux matrices ont aussi été intégrés dans le tableau 4.5.

		Σ^*	$\hat{\Sigma}^*$	$\hat{\Sigma}^*_{\mathbf{d}}$	$\hat{\Sigma}^*_{\hat{\mathbf{d}}}$	$\hat{\Sigma}^*_{\mathbf{d}2}$	$\hat{\Sigma}^*_{\hat{\mathbf{d}}2}$
$n = 40$	$D'(\Sigma)$	47.6	49.5	62.6	62.9	47.6	47.7
	Erreur relative (%)	0	4.1	31.7	32.4	0.1	0.4
$E = 1$	Estimation moyenne	0.999	1.011	0.841	0.847	0.990	0.996
	Coefficient de variation (%)	9.9	38	36	27	9.4	15
$n = 70$	$D'(\Sigma)$	77.6	84.0	92.6	93.0	77.6	77.9
	Erreur relative (%)	0	8.3	19.4	19.9	0.03	0.4
$E = 1$	Estimation moyenne	0.982	0.968	0.859	0.887	0.981	0.980
	Coefficient de variation (%)	6.8	76	32	57	5.7	8.0
$n = 100$	$D'(\Sigma)$	107.6	122.1	122.6	123.0	107.6	108.0
	Erreur relative (%)	0	13.6	14.0	14.4	0.02	0.4
$E = 1$	Estimation moyenne	0.989	1.564	1.170	0.895	0.982	0.973
	Coefficient de variation (%)	6.0	940	34	250	7.2	10.6

Tableau 4.5 – Comparaison numérique de la divergence D' et de l'estimation de $E = \int \pi(x)dx$ pour les matrices de covariance $\hat{\Sigma}^*_{\mathbf{d}}$, $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$, et $\hat{\Sigma}^*_{\mathbf{d}2}$, $\hat{\Sigma}^*_{\hat{\mathbf{d}}2}$ où g^* est égale à la densité de la loi "*banana shape*" π.

On peut voir que la projection sur les deux premiers vecteurs propres de Σ^* (colonne $\hat{\Sigma}^*_{\mathbf{d}2}$), donne des résultats aussi performants qu'avec la matrice optimale, et projeter sur les estimateurs, $\hat{\mathbf{d}}_1^*$ et $\hat{\mathbf{d}}_2^*$, permet de garder une bonne précision également. En effet, la divergence de Kullback-Leibler partielle de $\hat{\Sigma}^*_{\hat{\mathbf{d}}2}$ est toujours à moins de 0.4% de la valeur optimale et le coefficient de variation est légèrement plus élevé que celui de Σ^* (10.6% par exemple en dimension 100, contre 6% pour Σ^*) mais reste assez faible. En revanche, en ne projetant que sur le premier vecteur propre $\mathbf{d}_1^*$, la divergence s'accroit (entre 14 et 32% d'erreur) et l'estimation devient moins précise (coefficient de variation entre 32 et 36%, et la valeur moyenne de $\hat{E}_N$ est à environ 15% de la valeur théorique, quelle que soit la dimension). La précision diminue encore avec $\hat{\Sigma}^*_{\hat{\mathbf{d}}}$ (coefficient de variation de 57% en dimension $n = 70$, 250% pour $n = 100$) mais ces deux matrices sont tout de même plus efficaces que $\hat{\Sigma}^*$ en dimension 100 (qui a un coefficient de variation de plus de 900%). Ainsi, en grande dimension, il semble préférable d'utiliser une seule direction de projection que d'estimer la matrice entière. Malgré tout, on voit que le choix du nombre de directions est important et peut faire une grande différence dans la précision de l'estimation et la qualité de l'échantillonnage. Si l'algorithme 9 est efficace pour sélectionner le nombre de dimensions dans la majorité des cas que nous avons traités, il est ici mis en défaut. D'autres alternatives existent pour faire cette sélection (voir remarque 4.2.1), mais chacune peut devenir inefficace sur des cas particuliers. Nous préconisons donc malgré tout d'utiliser l'algorithme 9 de manière générale.

Remarque 4.2.1. Une alternative pour sélectionner le nombre de directions de projection est de choisir le plus petit entier k de sorte que le ratio $\dfrac{\sum_{i=1}^k \ell(\lambda_i)}{\sum_{i=1}^n \ell(\lambda_i)}$ soit plus grand qu'un réel $\rho \in]0, 1[$ (par exemple $\rho = 0.9$). Cette méthode permet, dans l'exemple 4.2.4 (avec $\rho = 0.9$), de choisir 2 directions jusqu'en dimension 100, mais dans des dimensions encore plus grandes, la matrice $\hat{\Sigma}^*$ étant de plus en plus bruitée, le nombre de directions choisi devient trop grand (par exemple

$k \approx 100$ pour $n = 300$). Pour rendre cette méthode plus performante, il faudrait adapter le réel ρ pour chaque exemple et chaque dimension.

4.2.5 Application à l'estimation d'une espérance : paiement d'une option asiatique

Le dernier exemple traité correspond à l'estimation de l'espérance E avec la fonction ϕ_3 définie en 3.13. Les paramètres optimaux $\mathbf{m}^*(\neq \mathbf{0})$ et Σ^* sont estimés par Monte-Carlo et le premier vecteur propre $\mathbf{d}_1^*$ de Σ^* est numériquement identique à $\mathbf{m}^*/\|\mathbf{m}^*\|$. L'algorithme 9 renvoyant toujours $k = 1$ (une seule direction de projection sélectionnée), on considère encore $\hat{\Sigma}^*_{\mathbf{d}} = \hat{\Sigma}^*_{\mathbf{m}}$.

		Σ^*	$\hat{\Sigma}^*$	$\hat{\Sigma}^*_{\mathbf{m}} = \hat{\Sigma}^*_{\mathbf{d}}$	$\hat{\Sigma}^*_{\hat{\mathbf{d}}}$	$\hat{\Sigma}^*_{\hat{\mathbf{m}}}$
$n = 16$	$D'(\Sigma)$	14.3	15.2	14.4	14.5	14.4
	Erreur relative (%)	0	5.7	0.3	1.2	0.5
$E = 2.45 \cdot 10^{-2}$	Estimation moyenne ($\times 10^{-2}$)	2.45	2.46	2.45	2.46	2.46
	Coefficient de variation (%)	1.7	6.7	1.6	2.2	1.7
$n = 40$	$D'(\Sigma)$	38.1	43.5	38.3	38.6	38.3
	Erreur relative (%)	0	14.1	0.5	1.5	0.6
$E = 2.04 \cdot 10^{-2}$	Estimation moyenne ($\times 10^{-2}$)	2.03	1.97	2.03	2.01	2.02
	Coefficient de variation (%)	1.7	35	3.2	3.9	2.1
$n = 100$	$D'(\Sigma)$	97.1	137.9	98.1	99.6	98.2
	Erreur relative (%)	0	42	0.9	2.5	1.1
$E = 1.87 \cdot 10^{-2}$	Estimation moyenne ($\times 10^{-2}$)	1.87	0.11	1.87	1.79	1.86
	Coefficient de variation (%)	3.1	94	2.6	9.1	2.5

Tableau 4.6 – Comparaison numérique de la divergence D' et de l'estimation de E pour les différentes matrices considérées section 4.2 et tableau 4.1, lorsque $\phi = \phi_3$ la fonction représentant le paiement d'une option asiatique (3.13).

Le tableau 4.6 montre l'amélioration apportée par les matrices avec projection par rapport à la matrice empirique, et ce dès la dimension 16. En effet, pour $\hat{\Sigma}^*$, la divergence D' a une erreur relative de 5.7% alors que toutes les autres matrices ont au maximum 1.2% d'erreur. Cela implique une meilleure précision de l'estimation, avec un coefficient de variation pour $\hat{\Sigma}^*$ de 6.7%, et de moins de 2.2% pour les autres matrices. Cette amélioration se confirme en dimension 40 et 100, où l'on remarque néanmoins que la projection sur $\hat{\mathbf{d}}_1^*$ donne des résultats un peu moins précis que la projection sur $\mathbf{d}_1^* = \mathbf{m}^*/\|\mathbf{m}^*\|$ ou sur $\hat{\mathbf{m}}^*$, comme précédemment (coefficient de variation de 9.1% contre 2.6 et 2.5% respectivement). On peut enfin noter que la matrice $\hat{\Sigma}^*_{\hat{\mathbf{m}}}$ est très performante, bien qu'on ne soit pas dans un cas d'estimation d'événement rare.

Remarque 4.2.2. Dans les exemples 4.2.1, 4.2.2 et 4.2.5, la matrice $\hat{\Sigma}^*_{\text{FIS}}$ définie dans la section 3.2.2 offre des performances équivalentes et même supérieures à celles des matrices considérées ici. En effet, dans ces cas le gradient est calculable facilement et permet d'obtenir les projections optimales très précisément, ce qui entraine des résultats d'estimation avec une faible erreur (proches de ceux donnés par la matrice optimale). En revanche, le sous-espace de projection FIS ne peut être déterminé dans les cas 4.2.3 (φ non différentiable) et 4.2.4 ($\phi \equiv 1$ et matrice H nulle), ce qui montre la limite de la méthode.

4.3 Conclusion

Les deux techniques de projection évoquées sections 4.1.2 et 4.1.3 ont montré leur efficacité pour construire une matrice de covariance proche de la matrice optimale et ainsi réaliser une estimation par échantillonnage préférentiel plus précise qu'avec la matrice empirique $\hat{\Sigma}^*$ en grande dimension. Ces projections permettent de réduire le nombre de paramètres de covariance à estimer et ainsi de diminuer les erreurs d'estimation tout en gardant un faible budget de simulation. La projection sur l'espace engendré par $\mathbf{m}^*$, ou sa valeur estimée $\hat{\mathbf{m}}^*$, est souvent performante dès que $\mathbf{m}^* \neq \mathbf{0}$. Cette projection est justifiée, dans le cadre des événements rares, par la propriété de queue légère de la loi normale, car les échantillons d'IS optimaux ont une faible variance dans la direction de $\mathbf{m}^*$. Elle n'est pas toujours optimale (voir exemple 4.2.2) et ne permet de projeter qu'en dimension 1, mais a l'avantage de rester efficace avec un faible budget de simulation. Les directions optimales de projection ont ensuite été déterminées en minimisant la divergence de Kullback-Leibler entre la densité optimale d'IS g^* et la famille de densités gaussiennes avec une matrice de covariance de la forme Σ_k (3.4), qui vit dans un sous-espace de $\mathcal{S}_n^+$ de dimension réduite. Ce résultat central est donné dans le théorème 4.1.1, qui affirme que les directions optimales sont des vecteurs propres de la matrice Σ^*. Dans tous les cas-tests considérés, projeter dans les directions optimales permet d'obtenir des estimations précises, en choisissant correctement le nombre de directions. Nous avons également noté que le premier vecteur propre $\mathbf{d}_1^*$ de Σ^* et la moyenne optimale $\mathbf{m}^*$ étaient souvent proches (comme dans les exemples 4.2.1, 4.2.3, et 4.2.5), mais dans ce cas, la projection sur $\hat{\mathbf{m}}^*$ donne des résultats légèrement plus précis qu'avec $\hat{\mathbf{d}}_1^*$, car l'estimation de ce dernier est plus bruité, comme expliqué dans la section 4.2.1.

Ainsi, les deux méthodes proposées pour réduire le nombre de paramètres estimés donnent des résultats prometteurs pour améliorer l'échantillonnage préférentiel en grande dimension. Cependant, dans ce chapitre, nous avons supposé que l'on savait générer des échantillons selon g^* pour estimer $\mathbf{m}^*$ et Σ^*. Or, en pratique cette étape n'est pas toujours aisée et peut être couteuse, et est généralement réalisée par des méthodes MCMC ou par des algorithmes adaptatifs d'échantillonnage préférentiel. L'objectif de la prochaine partie est donc de coupler les méthodes de projection proposées avec des algorithmes d'IS adaptatifs, visant à estimer une espérance.

Chapitre 5

Couplage d'un algorithme adaptatif d'IS avec une projection en petite dimension

Sommaire

L'objectif de ce chapitre est d'utiliser les méthodes de projection proposées dans le chapitre précédent pour améliorer un algorithme adaptatif d'échantillonnage préférentiel en grande dimension : l'algorithme d'entropie croisée, pour l'estimation de probabilités d'événements rares. Dans la littérature, [Uribe et al., 2021] ont déjà développé une méthode couplant la CE avec la projection dans le "*Failure-Informed Subspace*" (défini dans la section 3.2). L'algorithme iCEred qu'ils ont mis en place est présenté dans ce chapitre, mais comme il suppose la connaissance du gradient (qui est une limite importante) de la fonction d'intérêt, nous proposons deux nouvelles approches ne nécessitant pas le gradient. La première se base sur le calcul des vecteurs propres de la matrice de covariance, et couple la CE avec la méthode développée dans la section 4.1.3. La seconde intègre

la projection sur la moyenne optimale, expliquée dans la partie 4.1.2, à l'algorithme CE. Les algorithmes adaptatifs ainsi mis en place sont ensuite testés numériquement sur différents exemples d'estimation de probabilités d'événement rare.

5.1 L'algorithme iCEred

L'algorithme iCEred ("*improved Cross-Entropy method with failure-informed dimension reduction*") développé dans [Uribe et al., 2021] est une méthode récente d'estimation de probabilité d'événements rares en grande dimension, basée sur la méthode d'entropie croisée. L'idée principale est de projeter les échantillons dans le "*Failure-Informed Subspace*", présenté section 3.2, et de mettre à jour les paramètres de CE dans ce sous-espace de dimension réduite.

La détermination du FIS dans le cadre des probabilités d'événements rares a été détaillée dans la section 3.2.1 du chapitre 3. On rappelle simplement que la fonction indicatrice $\mathbb{I}_{\varphi(\cdot)\geq 0}$ est approchée par la fonction lisse $\psi(\cdot,\sigma) = F_{\mathcal{N}}(\varphi(\cdot)/\sigma)$, $\sigma \in \mathbb{R}_+^*$ (comme dans l'algorithme iCE (2)), afin d'évaluer le gradient de $\ln(\psi(\cdot,\sigma))$ (à σ fixé). On suppose ainsi, dans toute cette section, que la fonction φ est continument différentiable et que $\mathbb{E}_f(\|\nabla \ln \psi(\mathbf{X})\|^2) < +\infty$. Ce gradient permet d'estimer la matrice H (3.9), dont on détermine les valeurs propres et les vecteurs propres. La méthode générale pour trouver le FIS (et son complémentaire, le "*Complementary Subspace*", CS) est détaillée section 3.2.1. Ici, on présente l'algorithme iCEred, qui est une amélioration de l'algorithme iCE (2) en grande dimension à l'aide de la projection dans le FIS.

Étant donnés les vecteurs propres $\mathbf{d}_1, \ldots, \mathbf{d}_n$ de la matrice H, on définit la projection dans le FIS (de dimension k) $R^\top = (\mathbf{d}_1, \ldots, \mathbf{d}_k)^\top \in \mathbb{R}^{k\times n}$, et $R_\perp^\top = (\mathbf{d}_{k+1}, \ldots, \mathbf{d}_n)^\top \in \mathbb{R}^{(n-k)\times n}$, la projection dans le CS. Avec ces notations, le projecteur P_k^* (3.10) est égal à $RR^\top$. Ainsi, tout vecteur $\mathbf{x} \in \mathbb{R}^n$ se décompose de manière unique sur ces deux espaces : $\mathbf{x} = R\tilde{\mathbf{x}}_k + R_\perp \tilde{\mathbf{x}}_\perp$, où $\tilde{\mathbf{x}}_k = R^\top \mathbf{x} \in \mathbb{R}^k$ est la projection de $\mathbf{x}$ dans le sous-espace réduit, et $\tilde{\mathbf{x}}_\perp = R_\perp^\top \mathbf{x} \in \mathbb{R}^{n-k}$ la projection dans son orthogonal. De même, pour $\tilde{\mathbf{x}} = (\tilde{\mathbf{x}}_k, \tilde{\mathbf{x}}_\perp)$, la densité initiale se décompose en $f(\tilde{\mathbf{x}}) = f^k(\tilde{\mathbf{x}}_k) f^\perp(\tilde{\mathbf{x}}_\perp)$, où f^k est la densité $\mathcal{N}(\mathbf{0}, I_k)$, et $f^\perp$ est la densité $\mathcal{N}(\mathbf{0}, I_{n-k})$. La famille auxiliaire d'échantillonnage considérée est alors constituée des densités $h_{\mathbf{m}^k,\Sigma^k}(\tilde{\mathbf{x}}) = g^k_{\mathbf{m}^k,\Sigma^k}(\tilde{\mathbf{x}}_k) f^\perp(\tilde{\mathbf{x}}_\perp)$, avec $g^k_{\mathbf{m}^k,\Sigma^k}$ la densité gaussienne $\mathcal{N}(\mathbf{m}^k, \Sigma^k)$ dans $\mathbb{R}^k$.

Ainsi, la mise à jour des paramètres de CE se fait dans le sous-espace de dimension k et on obtient les expressions suivantes :

$$\mathbf{m}^{k*} = \frac{\mathbb{E}_f(\tilde{\mathbf{X}}_k \psi(\mathbf{X},\sigma))}{\mathbb{E}_f(\psi(\mathbf{X},\sigma))} \quad \text{et} \quad \Sigma^{k*} = \frac{\mathbb{E}_f((\tilde{\mathbf{X}}_k - \mathbf{m}^{k*})(\tilde{\mathbf{X}}_k - \mathbf{m}^{k*})^\top \psi(\mathbf{X},\sigma))}{\mathbb{E}_f(\psi(\mathbf{X},\sigma))} \tag{5.1}$$

avec $\tilde{\mathbf{X}}_k = R^\top \mathbf{X}$ et σ fixé.

En pratique, étant donné un échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_N \sim g_{\mathbf{m},\Sigma}$ à une itération t de l'algorithme, des nouvelles matrices de projection $(R^{(t+1)}, R_\perp^{(t+1)})$ et σ_{t+1} le paramètre de lissage mis à jour (voir l'algorithme iCE (2)), les paramètres à l'étape $t+1$ sont estimés par échantillonnage préférentiel par :

$$\hat{\mathbf{m}}^k_{t+1} = \frac{\sum_{i=1}^N \tilde{\mathbf{X}}_{k,i} \bar{L}_{t+1}(\mathbf{X}_i)}{\sum_{i=1}^N \bar{L}_{t+1}(\mathbf{X}_i)} \quad \text{et} \quad \hat{\Sigma}^k_{t+1} = \frac{\sum_{i=1}^N (\tilde{\mathbf{X}}_{k,i} - \hat{\mathbf{m}}^k_{t+1})(\tilde{\mathbf{X}}_{k,i} - \hat{\mathbf{m}}^k_{t+1})^\top \bar{L}_{t+1}(\mathbf{X}_i)}{\sum_{i=1}^N \bar{L}_{t+1}(\mathbf{X}_i)} \tag{5.2}$$

avec $\bar{L}_{t+1}(\mathbf{X}_i) = \psi(\mathbf{X}_i, \sigma_{t+1}) \dfrac{f(\tilde{\mathbf{X}}_i)}{g_{\mathbf{m},\Sigma}(\tilde{\mathbf{X}}_i)}$, et $\tilde{\mathbf{X}}_i = \left(R^{(t+1)}, R_\perp^{(t+1)}\right)^\top \mathbf{X}_i = (\tilde{\mathbf{X}}_{k,i}, \tilde{\mathbf{X}}_{\perp,i})$. D'autre part, pour trouver les sous-espaces FIS et CS, la matrice H est également estimée à chaque itération par :

$$\hat{H}_{t+1} = \frac{1}{\sum_{i=1}^{N} \tilde{L}_{t+1}(\mathbf{X}_i)} \sum_{i=1}^{N} \tilde{L}_{t+1}(\mathbf{X}_i) \left[\nabla \ln \psi(\mathbf{X}_i, \sigma_{t+1})\right] \left[\nabla \ln \psi(\mathbf{X}_i, \sigma_{t+1})\right]^\top \tag{5.3}$$

où $\tilde{L}_{t+1}(\mathbf{X}_i) = \psi(\mathbf{X}_i, \sigma_{t+1}) \dfrac{f^k(\tilde{\mathbf{X}}_{k,i})}{g^k_{\mathbf{m}_t^k, \Sigma_t^k}(\tilde{\mathbf{X}}_{k,i})}$. C'est à partir de cette matrice $\hat{H}_{t+1}$ que l'on calcule les vecteurs propres et valeurs propres permettant de construire le FIS.

Enfin, après avoir trouvé les vecteurs propres de $\hat{H}_{t+1}$, il faut exprimer les paramètres dans la nouvelle base (définie par $R^{(t+1)}$ et $R_\perp^{(t+1)}$). Ceux-ci sont donnés par :

$$\bar{\mathbf{m}} = \left(R^{(t+1)}, R_\perp^{(t+1)}\right)^\top R^{(t)} \hat{\mathbf{m}}_t^k \tag{5.4}$$

$$\bar{\Sigma} = \left(R^{(t+1)}, R_\perp^{(t+1)}\right)^\top \left(R^{(t)}, R_\perp^{(t)}\right) \begin{pmatrix} \hat{\Sigma}_t^k & 0 \\ 0 & I_{n-k} \end{pmatrix} \left(R^{(t)}, R_\perp^{(t)}\right)^\top \left(R^{(t+1)}, R_\perp^{(t+1)}\right) \tag{5.5}$$

En effet, $\bar{\mathbf{m}}$ et $\bar{\Sigma}$ sont les paramètres dans la base $\left(R^{(t+1)}, R_\perp^{(t+1)}\right)$, alors que dans l'ancienne base $\left(R^{(t)}, R_\perp^{(t)}\right)$, ces paramètres valaient par définition : $(\mathbf{m}_t^k, 0)$ et $\begin{pmatrix} \hat{\Sigma}_t^k & 0 \\ 0 & I_{n-k} \end{pmatrix}$. L'algorithme complet est basé sur le même schéma que iCE et est présenté dans l'algorithme 11.

En projetant les échantillons dans un sous-espace de petite dimension (le FIS), et en mettant à jour les paramètres de CE dans ce sous-espace, la méthode iCEred améliore sensiblement les algorithmes CE et iCE en grande dimension, en supposant que le gradient de la fonction d'état limite, φ, est disponible ou facile à obtenir. Selon les simulations numériques proposées dans [Uribe et al., 2021], iCEred permet d'estimer précisément des probabilités d'événement rare où la dimension atteint plusieurs centaines, alors que CE et iCE deviennent incapables de s'attaquer à des problèmes au-delà de la dimension 30, avec un budget de simulation identique.

Dans la suite de ce manuscrit, nous proposons de nouveaux algorithmes adaptatifs d'estimation améliorant la CE en grande dimension, basés sur les méthodes présentées dans le chapitre 4, et qui n'utilisent pas de gradient.

Algorithme 11 : iCEred : improved Cross-Entropy method with failure-informed dimension reduction

Données : Dimension de l'espace d'entrée n, coefficient de variation cible δ, seuil de tolérance de l'approximation ε, taille de l'échantillon par itération N, fonction d'état limite φ, nombre maximal d'itérations $t_{\max}$

Résultat : Estimation $\hat{E}_N$ de la probabilité $\mathbb{P}_f(\varphi(X) \geq 0)$

Initialisation : itération $t = 0$, paramètres gaussiens $\mathbf{m}_t = 0$, $\Sigma_t = I_n$, densité initiale $f = g_{0,I_n}$ et paramètre de lissage $\sigma_t = \infty$;

pour $t = 0 \dots t_{\max}$ **faire**

  si $t = 0$ **alors**

    Générer N échantillons indépendants $\mathbf{X}_1, \dots, \mathbf{X}_N \sim g_{\mathbf{m}_t,\Sigma_t}$ et poser $L_t(X_i) = 1$, pour tout $i = 1 \dots N$

  fin

  sinon

    Générer N échantillons indépendants selon la densité réduite $\tilde{\mathbf{X}}_{k,i} \sim g^k_{\hat{\mathbf{m}}^k_t,\hat{\Sigma}^k_t}$;

    Calculer les poids $L_t(\tilde{\mathbf{X}}_{k,i}) = f^k(\tilde{\mathbf{X}}_{k,i})/g^k_{\mathbf{m}^k_t,\Sigma^k_t}(\tilde{\mathbf{X}}_{k,i})$;

    Relever les échantillons en dimension n : $\mathbf{X}_i = R^{(t)}\tilde{\mathbf{X}}_{k,i} + R^{(t)}_{\perp}\tilde{\mathbf{X}}_{\perp,i}$;

  fin

  Calculer le coefficient de variation empirique du ratio entre la fonction indicatrice et son approximation, à partir des $\mathbf{X}_i$: $\widehat{\mathrm{cv}}_t = \dfrac{\sqrt{\widehat{\mathrm{Var}}(\mathbb{I}_{\{\varphi(\mathbf{X})\geq 0\}}/\psi(\mathbf{X},\sigma_t))}}{\hat{\mathbb{E}}(\mathbb{I}_{\{\varphi(\mathbf{X})\geq 0\}}/\psi(\mathbf{X},\sigma_t))}$;

  si $(\widehat{\mathrm{cv}}_t \leq \delta)$ ou $(t \geq t_{\max})$ **alors**

    Quitter

  fin

  Calculer $\sigma_{t+1} = \arg\min(\hat{\delta}_t(\sigma) - \delta)^2$ où le minimum est évalué sur $\sigma \in (0, \sigma_t)$ et $\hat{\delta}_t(\sigma)$ est le coefficient de variation des $\psi(\mathbf{X}_i, \sigma)L_t(\tilde{\mathbf{X}}_{k,i})$, $i = 1 \dots N$;

  Calculer les poids associés à la nouvelle indicatrice lissée pour tout i : $\tilde{L}_{t+1}(\mathbf{X}_i) = L_t(\tilde{\mathbf{X}}_{k,i})\psi(\mathbf{X}_i, \sigma_{t+1})$;

  Estimer la matrice $\hat{H}_{t+1}$ (5.3) à partir du gradient $\nabla \ln \psi(\mathbf{X}_i, \sigma_{t+1})$ et calculer ses valeurs propres, $\lambda_1 \geq \dots \geq \lambda_n$, et ses vecteurs propres, $\mathbf{d}_1, \dots, \mathbf{d}_n$;

  Trouver la dimension réduite k, tel que k soit le plus petit entier vérifiant : $\sum_{j=k+1}^{n} \lambda_j \leq \varepsilon$;

  Constuire la base du FIS : $R^{(t+1)} = (\mathbf{d}_1, \dots, \mathbf{d}_k)$ et du CS : $R^{(t+1)}_{\perp} = (\mathbf{d}_{k+1}, \dots, \mathbf{d}_n)$;

  Projeter les échantillons : $\tilde{\mathbf{X}}_{k,i} = \left(R^{(t+1)}\right)^{\top}\mathbf{X}_i$ et $\tilde{\mathbf{X}}_{\perp,i} = \left(R^{(t+1)}_{\perp}\right)^{\top}\mathbf{X}_i$;

  si $t = 0$ **alors**

    Prendre $\bar{L}_{t+1}(\mathbf{X}_i) = \psi(\mathbf{X}_i, \sigma_{t+1})$

  fin

  si $t > 0$ **alors**

    Calculer les paramètres $\bar{\mathbf{m}}$ (5.4) et $\bar{\Sigma}$ (5.5) dans la nouvelle base ;

    Calculer les poids associés $\bar{L}_{t+1}(\mathbf{X}_i) = \psi(\mathbf{X}_i, \sigma_{t+1})f(\tilde{\mathbf{X}}_i)/g_{\bar{\mathbf{m}},\bar{\Sigma}}(\tilde{\mathbf{X}}_i)$ où $\tilde{\mathbf{X}}_i = (\tilde{\mathbf{X}}_{k,i}, \tilde{\mathbf{X}}_{\perp,i})$;

  fin

  Estimer les nouveaux paramètres réduits $\hat{\mathbf{m}}^k_{t+1}$ et $\hat{\Sigma}^k_{t+1}$ avec (5.2).

fin

Estimer la probabilité : $\hat{E}_N = \dfrac{1}{N}\sum_{i=1}^{N} \mathbb{I}_{\{\varphi(\mathbf{X}_i)\geq 0\}} L_t(\tilde{\mathbf{X}}_{k,i})$.

5.2 Couplage de l'algorithme CE avec la projection sur les vecteurs propres de la matrice de covariance optimale

5.2.1 Mise en place des algorithmes

Dans cette section, nous proposons une amélioration de l'algorithme CE (1) à l'aide de la méthode de projection sur les vecteurs propres de Σ^*, définie dans la partie 4.1.3. Nous nous concentrons donc sur l'estimation de probabilités d'événement rare, où la fonction ϕ est une indicatrice du type $\mathbb{I}_{\{\varphi(\cdot)\geq 0\}}$, avec φ la fonction d'état limite. L'algorithme CE-P (12) (CE avec projections optimales) mis en place repose ainsi sur la méthode d'entropie croisée dans laquelle la matrice de covariance à estimer est Σ_k^* (4.4) (et non plus Σ^*). Nous présentons également la méthode iCE-P (13) (iCE avec projections optimales) analogue à CE-P mais basée sur iCE (2).

Algorithme 12 : CE-P : CE avec projections optimales

Données : dimension n, paramètre $\rho \in]0,1[$, taille de l'échantillon N, fonction d'état limite φ

Résultat : Estimation $\hat{E}_N$ de la probabilité d'événement rare $E = \mathbb{P}_f(\varphi(\mathbf{X}) \geq 0)$

1 Initialisation : Poser $t = 0$, $\mathbf{m}_t = \mathbf{0}$ et $\Sigma_t = I_n$;
2 Générer $\mathbf{X}_1, \ldots, \mathbf{X}_N$ indépendamment selon $g_{\mathbf{m}_t,\Sigma_t}$ et définir $L_t = f/g_{\mathbf{m}_t,\Sigma_t}$;
3 Évaluer $q_i = \varphi(\mathbf{X}_i)$ pour tout $i = 1, \ldots, N$;
4 Ranger les q_i dans l'ordre croissant : $q_{(1)} \leq \cdots \leq q_{(N)}$, et poser $\gamma_t = q_{(\lfloor(1-\rho)N\rfloor)}$;
5 Calculer les poids $w_i = w_i'/\sum_j w_j'$ avec $w_i' = \mathbb{I}_{\{q_i \geq \gamma_t\}} L_t(\mathbf{X}_i)$ pour tout i ;
6 **tant que** $\gamma_t < 0$ **faire**
7 Estimer $\mathbf{m}_{t+1} = \sum_{i=1}^N w_i \mathbf{X}_i$ et $\Sigma_{t+1}' = \sum_{i=1}^N w_i (\mathbf{X}_i - \mathbf{m}_{t+1})(\mathbf{X}_i - \mathbf{m}_{t+1})^\top$;
8 Calculer les valeurs propres $\lambda_1, \ldots, \lambda_n$ et les vecteurs propres associés $\mathbf{d}_1, \ldots, \mathbf{d}_n$ de la matrice Σ_{t+1}' où les λ_j sont rangées de sorte que $\ell(\lambda_1) \leq \cdots \leq \ell(\lambda_n)$;
9 Construire la matrice $\Sigma_{t+1} = \sum_{j=1}^k (\lambda_j - 1)\mathbf{d}_j \mathbf{d}_j^\top + I_n$, où k est obtenu par l'algorithme 9 avec en entrée $(\lambda_1, \ldots, \lambda_n)$;
10 Incrémenter t : $t \longleftarrow t + 1$;
11 Répéter les étapes 2, 3, 4 et 5 ;
12 **fin**
13 Estimer $\hat{E}_N = \dfrac{1}{N} \displaystyle\sum_{i=1}^N \mathbb{I}_{\{\varphi(\mathbf{X}_i)\geq 0\}} L_t(\mathbf{X}_i)$.

Les principales étapes de ces algorithmes sont les suivantes :

- Générer un échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_N$ indépendamment selon $g_{\mathbf{m}_t,\Sigma_t}$, la densité à l'étape t.
- Estimer les paramètres avec ces échantillons et les poids associés de CE (ou iCE) : $\mathbf{m}_{t+1}$ et Σ_{t+1}'.
- Calculer les valeurs propres $\lambda_1, \ldots, \lambda_n$ et les vecteurs propres associés $\mathbf{d}_1, \ldots, \mathbf{d}_n$ de la matrice Σ_{t+1}', où les λ_j sont rangées de sorte que $\ell(\lambda_1) \leq \cdots \leq \ell(\lambda_n)$.
- Construire la matrice $\Sigma_{t+1} = \sum_{j=1}^k (\lambda_j - 1)\mathbf{d}_j \mathbf{d}_j^\top + I_n$ (sur le modèle de Σ_k^*), où k est obtenu par l'algorithme 9 avec en entrée $(\lambda_1, \ldots, \lambda_n)$.

Ces étapes sont ensuite répétées jusqu'à ce que le critère d'arrêt soit vérifié. Les méthodes CE-P et iCE-P sont détaillées dans les algorithmes 12 et 13 respectivement.

Lorsque la dimension vaut n, estimer la matrice Σ_{t+1} à l'étape 9 de l'algorithme 12, entraine la réduction du nombre de paramètres pris en compte à chaque itération en passant de $n(n+3)/2$ pour CE et iCE, à $n(k+1)$ pour CE-P et iCE-P (n paramètres dans $\mathbf{m}_{t+1}$, k valeurs propres de Σ'_{t+1}, et $n-1$ coefficients pour chacun des k vecteurs propres orthonormés associés). Cela implique une diminution du nombre total d'erreurs d'estimation, et potentiellement, une plus grande précision dans l'estimation finale de la probabilité. Cependant, les vecteurs propres étant obtenus à partir de la matrice Σ'_{t+1}, qui est mal estimée en grande dimension, ils peuvent eux aussi être très imprécis et entrainer de mauvaises estimations. On évaluera numériquement l'efficacité des algorithmes 12 et 13 dans la section suivante.

Algorithme 13 : iCE-P : iCE avec projections optimales.

Données : dimension n, paramètre δ, taille de l'échantillon N, fonction d'état limite φ
Résultat : Estimation $\hat{E}_N$ de la probabilité d'événement rare $E = \mathbb{P}_f(\varphi(\mathbf{X}) \geq 0)$

1 Initialisation : poser $t = 0$, $\mathbf{m}_t = \mathbf{0}$, $\Sigma_t = I_n$ et $\sigma_t = \infty$;
2 Générer $\mathbf{X}_1, \ldots, \mathbf{X}_N$ indépendamment selon $g_{\mathbf{m}_t,\Sigma_t}$ et définir $L_t = f/g_{\mathbf{m}_t,\Sigma_t}$;
3 Calculer $q_i = \varphi(\mathbf{X}_i)$ pour tout $i = 1, \ldots, N$;
4 Évaluer $\widehat{\mathrm{cv}}$ le coefficient de variation empirique des $\mathbb{I}_{\{q_i \geq 0\}}/F_{\mathcal{N}}(q_i/\sigma_t)$;
5 **tant que** $\widehat{\mathrm{cv}} \geq \delta$ **faire**
6 Calculer $\sigma_{t+1} = \arg\min(\hat{\delta}_t(\sigma) - \delta)^2$ où le minimum est évalué sur $\sigma \in (0, \sigma_t)$ et $\hat{\delta}_t(\sigma)$ est le coefficient de variation des $F_{\mathcal{N}}(q_i/\sigma)L_t(\mathbf{X}_i)$;
7 Calculer les poids $w_i = w'_i / \sum_j w'_j$ où $w'_i = F_{\mathcal{N}}(q_i/\sigma_{t+1})L_t(\mathbf{X}_i)$;
8 Estimer $\mathbf{m}_{t+1} = \sum_{i=1}^N w_i \mathbf{X}_i$ et $\Sigma'_{t+1} = \sum_{i=1}^N w_i(\mathbf{X}_i - \mathbf{m}_{t+1})(\mathbf{X}_i - \mathbf{m}_{t+1})^\top$;
9 Calculer les valeurs propres $\lambda_1, \ldots, \lambda_n$ et les vecteurs propres associés $\mathbf{d}_1, \ldots, \mathbf{d}_n$ de la matrice Σ'_{t+1} où les λ_j sont rangées de sorte que $\ell(\lambda_1) \leq \cdots \leq \ell(\lambda_n)$;
10 Construire la matrice $\Sigma_{t+1} = \sum_{j=1}^k (\lambda_j - 1)\mathbf{d}_j\mathbf{d}_j^\top + I_n$, où k est obtenu par l'algorithme 9 avec en entrée $(\lambda_1, \ldots, \lambda_n)$;
11 Incrémenter t : $t \longleftarrow t+1$;
12 Répéter les étapes 2, 3 et 4 ;
13 **fin**
14 Estimer $\hat{E}_N = \dfrac{1}{N}\displaystyle\sum_{i=1}^N \mathbb{I}_{\{\varphi(\mathbf{X}_i) \geq 0\}} L_t(\mathbf{X}_i)$.

Remarque 5.2.1. Pour les simulations numériques, les matrices Σ'_{t+1} et Σ_{t+1} sont mises à jour en y ajoutant le terme εI_n, avec ε un réel strictement positif. Cet ajout permet d'éviter l'effondrement de la covariance vers une matrice singulière à cause des erreurs numériques. Cette modification est présente dans les algorithmes CE et iCE disponibles sur le site internet des auteurs [Papaioannou et al., 2019b], bien que non mentionnée dans leur article [Papaioannou et al., 2019a]. Dans toutes les simulations, ε est égal à $\varepsilon = 10^{-6}$.

5.2.2 Résultats numériques

Nous allons maintenant comparer les algorithmes CE-P et iCE-P à CE et iCE pour différentes dimensions. Dans toutes les simulations, les paramètres d'entrée sont fixés à $\rho = 0.1$ pour CE et

CE-P, $\delta = 1.5$ pour iCE, et $\delta = 3$ pour iCE-P. Ces choix ont été effectués de manière empirique pour CE-P et iCE-P, de sorte que les résultats soient les plus précis possible. Pour CE et iCE, nous avons gardé les paramètres conseillés par les auteurs dans [Rubinstein and Kroese, 2011] et [Papaioannou et al., 2019a]. Dans chacun des exemples, le budget de simulation est fixé et est égal à la taille de l'échantillon N multiplié par le nombre d'itérations. Pour obtenir une estimation moyenne de la probabilité E, 100 répétitions indépendantes de chaque algorithme sont effectuées, de façon à obtenir des estimations $\hat{E}_N^{(1)}, \ldots, \hat{E}_N^{(100)}$, d'en calculer la moyenne $\bar{E}_N$, ainsi que le biais relatif, $\dfrac{\bar{E}_N - E}{E}$, et le coefficient de variation, $\dfrac{\sqrt{\frac{1}{100}\sum_{i=1}^{100}(\hat{E}_N^{(i)} - E)^2}}{E}$. Ces valeurs sont regroupées dans les tableaux 5.1 et 5.2. Le nombre maximal d'itérations a été fixé à 10, et nous considérons donc que l'algorithme n'a pas convergé s'il ne s'est pas arrêté après 10 itérations. Nous avons noté "NC" (non convergent) dans les tableaux lorsque l'algorithme n'a pas convergé.

5.2.2.1 Exemple jouet : la somme des coordonnées

Le premier exemple considéré correspond à la fonction d'état limite φ_1 (2.1) déjà testée dans les chapitres précédents. Rappelons que la matrice optimale Σ^* est égale à Σ_k^* (4.4) avec $k = 1$ et $\mathbf{d}_1^* = \mathbf{1}_n$ (voir section 4.2.1). En pratique, dans les simulations de CE-P et iCE-P réalisées ici, le nombre de directions choisi vaut presque toujours $k = 1$ (il arrive parfois que $k = 2$), et la direction sélectionnée (ou une des deux lorsque $k = 2$) est bien une approximation de $\mathbf{d}_1^*$. Pour ce cas-test, nous fixons le budget de simulation à environ 30000 (suivant l'algorithme, le budget moyen sur les 100 simulations peut varier entre 29500 et 30700). Les résultats sont donnés dans le tableau 5.1.

		CE	CE-P	iCE	iCE-P
$n = 30$	Estimation moyenne ($\times 10^{-3}$)	0.78	1.21	1.35	1.35
	Biais relatif (%)	-42	-11	0.1	0.1
$E = 1.35 \cdot 10^{-3}$	Coefficient de variation (%)	54	19	2.0	1.3
	Taille d'échantillon N	8000	8000	10000	10000
$n = 70$	Estimation moyenne ($\times 10^{-3}$)	NC	1.05	NC	1.35
	Biais relatif (%)	NC	-22	NC	0.2
$E = 1.35 \cdot 10^{-3}$	Coefficient de variation (%)	NC	35	NC	1.7
	Taille d'échantillon N	NC	8000	NC	10000
$n = 100$	Estimation moyenne ($\times 10^{-3}$)	NC	1.18	NC	1.35
	Biais relatif (%)	NC	-12	NC	-0.3
$E = 1.35 \cdot 10^{-3}$	Coefficient de variation (%)	NC	34	NC	2.5
	Taille d'échantillon N	NC	8000	NC	10000

Tableau 5.1 – Comparaison des algorithmes CE, iCE, CE-P, et iCE-P pour la fonction d'état limite φ_1 (2.1). Le budget de simulation pour chaque algorithme est d'environ 30000.

Le premier point à souligner est la bonne performance des algorithmes avec projection comparés aux algorithmes de base. En effet, dès la dimension $n = 30$, CE-P est plus précis que CE (coefficient de variation de 54% pour CE contre 19% pour CE-P et biais relatifs respectifs de -42% et -11%), et dans les dimensions supérieures, CE ne converge plus alors que CE-P donne des résultats raisonnables même s'il n'est pas très précis (biais relatif de -22 à -12% et coefficient de variation inférieur à 35%). D'autre part, iCE et iCE-P sont tous deux très précis en dimension 30 avec un tel budget, vu que leur coefficient de variation ne dépasse pas 2%. Cependant, iCE ne

converge plus en dimension 70 et 100, alors que iCE-P garde un coefficient de variation inférieur à 2.5%. Ces résultats montrent que la méthode de projection suggérée par le théorème 4.1.1 permet d'augmenter la précision des algorithmes CE et iCE, en estimant la variance uniquement dans la direction du vecteur propre de Σ'_t ayant la plus petite valeur propre.

Par ailleurs, notons que iCE-P est nettement plus performant que CE-P dans ce cas (iCE est même meilleur que CE-P en dimension 30). Nous suggérons que cette supériorité est due au fait que iCE-P (et iCE) prend en compte plus d'échantillons que CE-P pour estimer les paramètres à chaque itération, et qu'il nécessite moins d'itérations pour converger (3 pour iCE-P, environ 3.8 en moyenne pour CE-P) donc la taille d'échantillon par itération est aussi plus grande pour iCE-P.

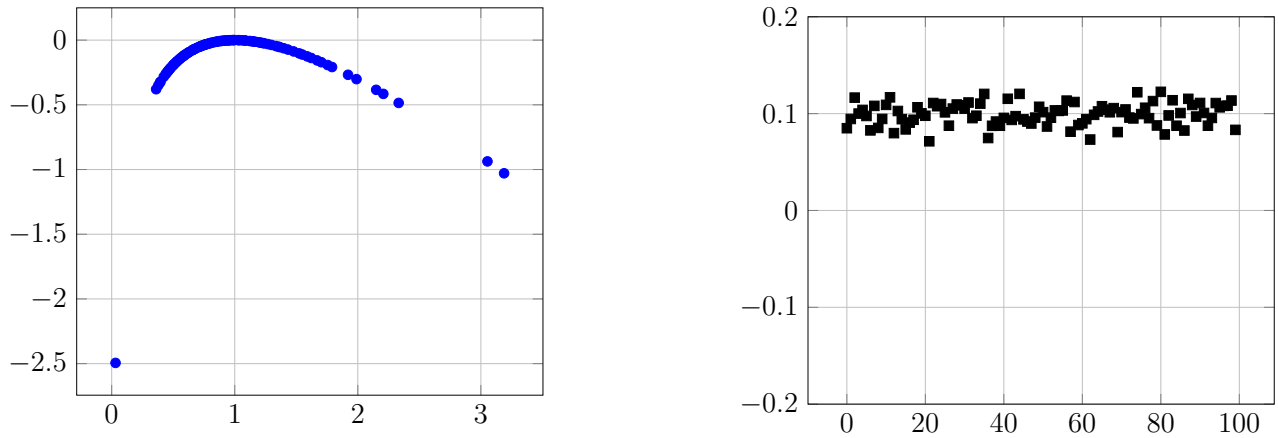


Figure 5.1 – Images par ℓ des valeurs propres de la matrice Σ'_t (à gauche) à la dernière itération de l'algorithme CE-P en dimension $n = 100$ et les coordonnées du vecteur propre associé (à droite) à la plus petite valeur propre, ce vecteur étant une estimation de $\mathbf{1}_n$.

Enfin, notons que le budget de simulation (≈ 30000) a été choisi de sorte que l'estimation de la matrice de covariance Σ'_t soit assez précise et que l'approximation de la direction optimale ($\mathbf{1}_n$) soit toujours sélectionnée. En effet, jusqu'en dimension 100, CE-P et iCE-P parviennent à bien identifier la plus petite valeur propre de Σ'_t et son vecteur propre associé. La figure 5.1 montre les images des valeurs propres de Σ'_t par la fonction ℓ (4.3) lors d'une réalisation de l'algorithme CE-P en dimension 100, ainsi que le vecteur propre sélectionné. La valeur minimisant la fonction ℓ est la plus petite valeur propre et le vecteur associé est bien une approximation de la direction optimale $\mathbf{1}_n$.

En revanche, dès que la dimension augmente, la matrice Σ'_t et donc ses valeurs propres sont de plus en plus mal estimées et le(s) vecteur(s) sélectionné(s) ne correspond(ent) plus à la direction optimale. C'est ce que l'on voit sur la figure 5.2 (tirée d'une réalisation de CE-P en dimension 200), où la valeur propre minimisant ℓ est la plus grande (la plus à droite sur le graphique de gauche) et le vecteur propre associé est très différent de la direction optimale. On a le même comportement lorsqu'on diminue le budget de simulation dans les dimensions inférieures ou égales à 100.

Malgré tout, dans ce cas particulier, on a observé dans la section 3.1.3.1 que choisir une direction de projection de manière aléatoire donnait des résultats très précis en grande dimension. En diminuant le budget ou en augmentant encore la dimension ($n > 100$), les algorithmes CE-P et iCE-P ne projettent pas sur la direction optimale théorique $\mathbf{1}_n$ (ou son approximation), mais ils restent performants (ce qui n'est pas le cas dans l'exemple suivant).

Ainsi, les algorithmes utilisant la méthode de projection définie en 4.1.3, CE-P et iCE-P, permettent de diminuer l'erreur d'estimation de $\mathbb{P}_f(\varphi_1(\mathbf{X}) \geq 0)$ en dimension 30, 70 et 100. Néanmoins,

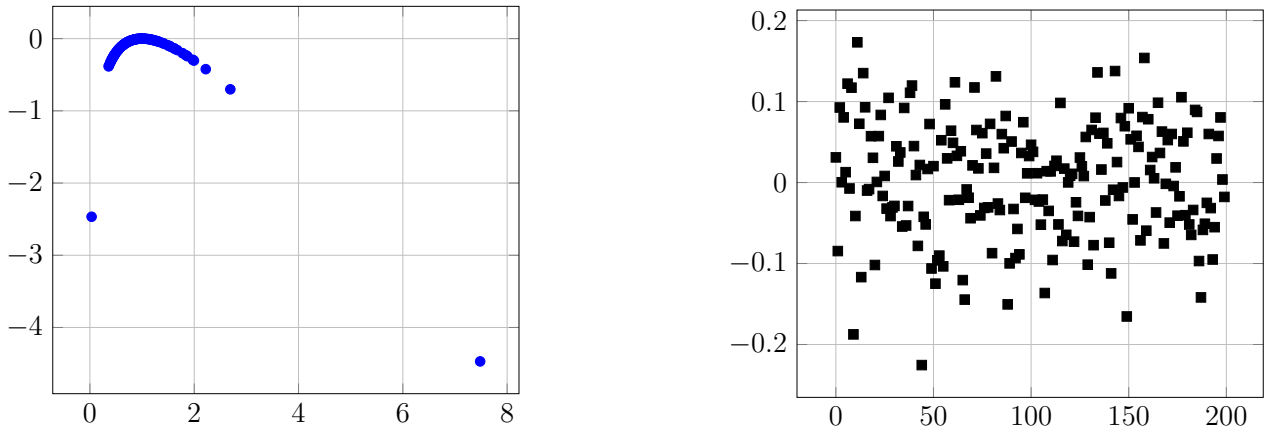


Figure 5.2 – Images par ℓ des valeurs propres de la matrice Σ'_t (à gauche) à la dernière itération de l'algorithme CE-P en dimension $n = 200$ et les coordonnées du vecteur propre associé (à droite) à la plus grande valeur propre qui minimise la fonction ℓ.

pour sélectionner la direction optimale de projection le budget doit être suffisamment élevé, et doit être augmenté si la dimension devient plus grande. Dans cet exemple, ne pas identifier la direction optimale n'a pas d'influence sur la convergence des algorithmes et dégrade assez peu l'estimation mais le cas-test suivant est un cas où le choix de la direction de projection a une grande influence.

5.2.2.2 Exemple jouet : un polynôme de degré 2

Dans ce second exemple, on considère la fonction φ_5 (4.7) pour l'estimation de la probabilité $E = \mathbb{P}_f(\varphi_5(\mathbf{X}) \geq 0)$. D'après la figure 4.5b, il y a en théorie deux directions de projection optimales à retenir ($k = 2$) pour estimer la matrice de covariance et ces deux vecteurs sont $\mathbf{e}_3$ (associé à la plus petite valeur propre de Σ^*) et $\mathbf{e}_2$ (associé à la deuxième plus petite valeur propre). Dans les simulations des algorithmes CE-P et iCE-P réalisées pour obtenir le tableau 5.2, on a toujours $k = 2$ également, et les deux vecteurs sélectionnés sont des approximations de $\mathbf{e}_2$ et $\mathbf{e}_3$. Le budget de simulation est fixé à environ 15000 (le budget moyen variant de 15000 à 15500 selon l'algorithme).

On remarque à nouveau que CE-P et iCE-P gardent un faible coefficient de variation (au maximum 12% en dimension 100 pour CE-P, et 3.8% pour iCE-P) et un biais relatif proche de zéro (entre -2.4 et -0.2% pour CE-P et entre -0.4 et 0.1% pour iCE-P), alors qu'avec le même budget, CE et iCE sont moins précis en dimension 30 et ne parviennent plus à estimer correctement la probabilité en dimension 70 et 100. Notons que l'algorithme iCE-P est légèrement plus performant que CE-P, pour les mêmes raisons que l'exemple précédent (une plus grande taille d'échantillon est utilisée pour l'estimation dans iCE).

L'efficacité de ces deux algorithmes est cependant vite dégradée lorsqu'on diminue le budget de simulation ou qu'on augmente la dimension. En effet, en dimension 300 et avec un budget de 15000, CE-P et iCE-P ne convergent plus car les directions optimales ne sont plus identifiées du fait de la mauvaise estimation de la matrice Σ'_t. La variance n'est alors plus estimée dans les directions influentes ($\mathbf{e}_2$ et $\mathbf{e}_3$), et la densité auxiliaire finale ($g_{\mathbf{m}_t,\Sigma_t}$) n'est pas assez proche de la densité optimale pour permettre la convergence des algorithmes. De même, si le budget est réduit à 10000 en dimension 100 par exemple, CE-P ne converge pas systématiquement et a un coefficient de variation proche de 100%.

Remarque 5.2.2. Dans les deux exemples traités, le gradient de φ est connu analytiquement et

		CE	CE-P	iCE	iCE-P
$n = 30$	Estimation moyenne ($\times 10^{-3}$)	1.46	1.51	1.53	1.51
	Biais relatif (%)	-3.2	-0.2	1.1	0.1
$E = 1.51 \cdot 10^{-3}$	Coefficient de variation (%)	18	2.8	7.2	2.4
	Taille d'échantillon N	5000	5000	5000	5000
$n = 70$	Estimation moyenne ($\times 10^{-3}$)	$3.2 \cdot 10^{-4}$	1.51	NC	1.50
	Biais relatif (%)	-100	-0.2	NC	-0.4
$E = 1.51 \cdot 10^{-3}$	Coefficient de variation (%)	100	4.6	NC	2.6
	Taille d'échantillon N	5000	5000	NC	5000
$n = 100$	Estimation moyenne ($\times 10^{-3}$)	0	1.47	NC	1.51
	Biais relatif (%)	-100	-2.4	NC	-0.3
$E = 1.51 \cdot 10^{-3}$	Coefficient de variation (%)	100	12	NC	3.8
	Taille d'échantillon N	5000	5000	NC	5000

Tableau 5.2 – Comparaison des algorithmes CE, iCE, CE-P, et iCE-P pour la fonction d'état limite φ_5 (4.7). Le budget de simulation pour chaque algorithme est d'environ 15000.

son évaluation n'est pas très couteuse. L'algorithme iCEred est alors très performant et surpasse CE-P et iCE-P, avec un budget similaire.

Par ailleurs, si on ne met à jour que la diagonale de la matrice de covariance (au lieu de la matrice pleine) dans les algorithmes CE et iCE (comme suggéré dans [Bourinet, 2018]), on aurait des résultats similaires voire parfois meilleurs qu'avec CE-P et iCE-P, avec un budget identique. En effet, comme la matrice de covariance optimale est proche de l'identité en grande dimension dans le premier exemple, et diagonale dans le second, il est pertinent d'estimer uniquement la diagonale pour réduire le nombre de paramètres. Cependant, on a vu que prendre en compte uniquement la diagonale pouvait être facilement mis en défaut (voir section 3.1.3.4), et on verra dans la section suivante que ce n'est pas toujours performant.

Finalement, les deux algorithmes définis dans cette section, CE-P (12) et iCE-P (13), donnent des résultats d'estimation prometteurs avec des coefficients de variation assez faibles en dimension 100 et moins. La projection sur les vecteurs propres de la matrice de covariance, associés aux valeurs propres minimisant ℓ, permet d'augmenter la précision de l'estimation en réduisant le nombre de paramètres estimés à chaque étape.

Néanmoins, le budget de simulation doit être suffisant pour que la matrice Σ'_t et ses éléments propres soient estimés précisément et les deux algorithmes ne convergent pas toujours dans les dimensions supérieures à 100, sans augmenter fortement le budget. Pour améliorer ces deux méthodes, une première piste serait d'estimer Σ'_t avec d'autres techniques plus efficaces en grande dimension pour estimer des matrices de covariance avec un petit budget (voir par exemple [Ledoit and Wolf, 2004] et [Ashurbekova et al., 2020]) plutôt que d'utiliser la matrice empirique. Une autre idée serait de déterminer les éléments propres à l'aide de méthodes plus robustes d'estimation des valeurs et vecteurs propres de matrices de covariance de grande dimension (comme dans [Mestre, 2008a], [Mestre, 2008b], [Nadakuditi and Edelman, 2008] et [Benaych-Georges and Nadakuditi, 2011]). Ces deux pistes sont des perspectives d'amélioration intéressantes des algorithmes proposés mais n'ont pas été développées au cours de cette thèse. En revanche, pour éviter que les directions de projection ne dépendent de l'estimation, souvent imprécise, d'une matrice de covariance de grande taille, nous avons proposé d'intégrer la méthode de projection sur la moyenne optimale $\mathbf{m}^*$ (présentée section 4.1.2) aux algorithmes CE et iCE. Le couplage de ces deux algorithmes avec la

projection sur $\mathbf{m}^*$ est présenté dans la section suivante.

5.3 Couplage de l'algorithme CE à la projection dans le sous-espace engendré par la moyenne

5.3.1 Mise en place des algorithmes CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$

Les résultats numériques (4.2) du chapitre 4 ont montré que projeter dans la direction de la moyenne optimale $\mathbf{m}^*$ (dès lors que $\mathbf{m}^* \neq 0$), ou de son approximation $\hat{\mathbf{m}}^*$, améliorait souvent l'estimation de E. C'est particulièrement le cas lorsqu'on estime une probabilité d'événement rare (voir les exemples 4.2.1, 4.2.2, 4.2.3). De plus, quand la direction optimale $\mathbf{d}_1^*$ coïncide avec $\mathbf{m}^*$, il vaut mieux utiliser l'estimation $\hat{\mathbf{m}}^*$ de $\mathbf{m}^*$ pour projeter que celle de $\mathbf{d}_1^*$, car cette dernière est moins précise puisqu'elle provient de la matrice de covariance empirique $\hat{\Sigma}^*$. Nous avons ainsi proposé, dans l'article [El Masri et al., 2021], deux algorithmes très simples à mettre en place : CE-$\mathbf{m}^*$ (14) (CE avec projection dans la direction de $\mathbf{m}^*$) et iCE-$\mathbf{m}^*$ (15) (iCE avec projection dans la direction de $\mathbf{m}^*$). Ils reposent sur l'itération des étapes suivantes :

- Générer un échantillon $\mathbf{X}_1, \ldots, \mathbf{X}_N$ indépendamment selon $g_{\mathbf{m}_t, \Sigma_t}$, la densité à l'étape t.
- Estimer la moyenne $\mathbf{m}_{t+1} = \sum_{i=1}^N w_i \mathbf{X}_i$ avec ces échantillons et les poids normalisés $\bar{w}_i$ de CE, ou iCE (définis dans les algorithmes 1 et 2).
- Définir la projection sur $\mathbf{m}_{t+1}$, $\mathbf{d} = \mathbf{m}_{t+1}/\|\mathbf{m}_{t+1}\|$, et projeter les échantillons $Y_i = \mathbf{d}^\top \mathbf{X}_i$.
- Estimer leur variance conditionnelle $\hat{v} = \sum_{i=1}^N w_i (Y_i - \|\mathbf{m}_{t+1}\|)^2$ et construire la matrice $\Sigma_{t+1} = (\hat{v} - 1)\mathbf{d}\mathbf{d}^\top + I_n$.

Contrairement aux méthodes CE-P et iCE-P, il n'est pas nécessaire d'estimer la matrice de covariance complète (Σ'_{t+1} dans les algorithmes 12 et 13), c'est pourquoi nous préférons d'abord projeter les échantillons puis estimer la variance $\hat{v}$ (étape 9 des algorithmes 14 et 15). Remarquons néanmoins que cette variance est bien égale à $\mathbf{d}^\top \Sigma'_{t+1} \mathbf{d}$ (car $\mathbf{d}^\top \mathbf{m}_{t+1} = \|\mathbf{m}_{t+1}\|$) et cela reviendrait au même d'estimer Σ'_{t+1} puis de calculer le produit matriciel $\mathbf{d}^\top \Sigma'_{t+1} \mathbf{d}$.

Le principal avantage des algorithmes CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ est qu'en dimension n, on estime seulement $n+1$ paramètres (n coefficients de $\mathbf{m}_{t+1}$ et 1 coefficient de variance $\hat{v}$) à chaque itération, contre $n(n+3)/2$ dans CE et iCE, et $n(k+1)$ dans CE-P et iCE-P. De plus, l'estimation de $\mathbf{m}_t$ à chaque itération reste assez précise pour des dimensions de quelques centaines, contrairement à la matrice de covariance empirique, ce qui permet de projeter efficacement. Ces deux algorithmes sont testés numériquement dans la partie suivante, où ils sont comparés à CE et iCE. Dans chaque exemple, nous appliquons également CE et iCE en ne mettant à jour que la diagonale de la covariance, que nous notons CEd et iCEd respectivement, comme suggéré dans [Bourinet, 2018]. En effet, le graphique 2.1 du chapitre 2 montre que CEd est très performant jusqu'en dimension 60, et c'est aussi le cas pour iCEd. À chaque itération, CEd et iCEd n'estiment que $2n$ paramètres (n pour la moyenne, et n pour la diagonale de la covariance), ce qui explique leur efficacité lorsque la dimension augmente comparés à CE et iCE, qui mettent à jour la matrice de covariance pleine. En revanche, une limite est l'annulation de tous les coefficients de covariance qui peuvent malgré tout être influents (voir le cas extrême 3.1.3.4). D'autre part, CEd et iCEd estiment à chaque étape n coefficients diagonaux qui ne sont pas tous influents, et leur mise à jour peut rajouter du bruit et dégrader le résultat final. La projection uniquement sur $\mathbf{m}^*$ permet d'éviter l'estimation

Algorithme 14 : CE-$\mathbf{m}^*$: CE avec projection dans la direction de $\mathbf{m}^*$

Données : dimension n, paramètre $\rho \in]0,1[$, taille de l'éachantillon N, fonction d'état limite φ

Résultat : Estimation $\hat{E}_N$ de la probabilité d'événement rare $E = \mathbb{P}_f(\varphi(\mathbf{X}) \geq 0)$

1 Initialisation : Poser $t = 0$, $\mathbf{m}_t = \mathbf{0}$ et $\Sigma_t = I_n$;
2 Générer $\mathbf{X}_1, \ldots, \mathbf{X}_N$ indépendamment selon $g_{\mathbf{m}_t,\Sigma_t}$ et définir $L_t = f/g_{\mathbf{m}_t,\Sigma_t}$;
3 Évaluer $q_i = \varphi(\mathbf{X}_i)$ pour tout $i = 1, \ldots, N$;
4 Ranger les q_i dans l'ordre croissant : $q_{(1)} \leq \cdots \leq q_{(N)}$, et poser $\gamma_t = q_{(\lfloor(1-\rho)N\rfloor)}$;
5 Calculer les poids $w_i = w'_i / \sum_j w'_j$ avec $w'_i = \mathbb{I}_{\{q_i \geq \gamma_t\}} L_t(\mathbf{X}_i)$ pour tout i;
6 **tant que** $\gamma_t < 0$ **faire**
7 Estimer $\mathbf{m}_{t+1} = \sum_{i=1}^N w_i \mathbf{X}_i$;
8 Définir $\mathbf{d}^\top = \mathbf{m}_{t+1}^\top / \|\mathbf{m}_{t+1}\| \in \mathbb{R}^{1\times n}$ la projection sur $\mathrm{Vect}(\mathbf{m}_{t+1})$;
9 Projeter les échantillons $Y_i = \mathbf{d}^\top \mathbf{X}_i$ et estimer leur variance conditionnelle $\hat{v} = \sum_{i=1}^N w_i (Y_i - \|\mathbf{m}_{t+1}\|)^2$ et poser $\Sigma_{t+1} = (\hat{v} - 1)\mathbf{d}\mathbf{d}^\top + I_n$;
10 Incrémenter t : $t \longleftarrow t + 1$;
11 Répéter les étapes 2, 3, 4 et 5 ;
12 **fin**
13 Estimer $\hat{E}_N = \dfrac{1}{N} \displaystyle\sum_{i=1}^N \mathbb{I}_{\{\varphi(\mathbf{X}_i) \geq 0\}} L_t(\mathbf{X}_i)$.

de coefficients "inutiles" (au sens où ils dégraderaient la performance sans apporter d'information supplémentaire) et présente l'avantage d'estimer la variance dans une direction influente. La comparaison de ces six algorithmes est réalisée dans la section 5.3.2.

Remarque 5.3.1. Comme dans la section précédente (voir remarque 5.2.1), les simulations numériques sont réalisées en ajoutant le terme εI_n dans la mise à jour de la matrice Σ_{t+1} afin d'éviter l'effondrement de la covariance. Autrement dit, on a $\Sigma_{t+1} = (\hat{v} - 1)\mathbf{d}\mathbf{d}^\top + (1+\varepsilon)I_n$ à l'étape 9 des algorithmes 14 et 15 avec $\varepsilon = 10^{-6}$.

5.3.2 Résultats numériques

Dans tous les exemples qui suivent nous comparons la précision des algorithmes CE, iCE, CEd, iCEd, CE-$\mathbf{m}^*$, et iCE-$\mathbf{m}^*$ pour l'estimation de la probabilité d'événement rare $E = \mathbb{P}_f(\varphi(\mathbf{X}) \geq 0)$ en faisant varier la dimension n. Les paramètres ρ et δ sont fixés à $\rho = 0.1$ pour les trois méthodes basées sur CE, $\delta = 1.5$ pour iCE, et $\delta = 3$ pour iCEd et iCE-$\mathbf{m}^*$. Les simulations sont effectuées de sorte à avoir un budget moyen autour de 8000. Comme dans la partie 5.2.2, nous calculons l'estimation moyenne de la probabilité, le coefficient de variation et le biais relatif sur 100 répétitions indépendantes de chaque algorithme, et ces valeurs sont reportées dans les tableaux avec la taille de l'échantillon par itération N. On rappelle que nous notons "NC" (non convergent) lorsqu'un algorithme n'a pas convergé en moins de 10 itérations.

5.3.2.1 Exemple jouet : la somme des coordonnées

Nous reprenons la fonction φ_1 (2.1) comme premier exemple. La direction donnée par la moyenne optimale théorique, $\mathbf{m}^*$, est ici égale à la direction optimale de projection $\mathbf{d}_1^* = \mathbf{1}_n$

Algorithme 15 : iCE-$\mathbf{m}^*$: iCE avec projection dans la direction de $\mathbf{m}^*$

Données : dimension n, paramètre δ, taille de l'éachantillon N, fonction d'état limite φ

Résultat : Estimation $\hat{E}_N$ de la probabilité d'événement rare $E = \mathbb{P}_f(\varphi(\mathbf{X}) \geq 0)$

1 Initialisation : poser $t = 0$, $\mathbf{m}_t = \mathbf{0}$, $\Sigma_t = I_n$ et $\sigma_t = \infty$;

2 Générer $\mathbf{X}_1, \ldots, \mathbf{X}_N$ indépendamment selon $g_{\mathbf{m}_t,\Sigma_t}$ et définir $L_t = f/g_{\mathbf{m}_t,\Sigma_t}$;

3 Calculer $q_i = \varphi(\mathbf{X}_i)$ pour tout $i = 1, \ldots, N$;

4 Évaluer $\widehat{\mathrm{cv}}$ le coefficient de variation empirique des $\mathbb{I}_{\{q_i \geq 0\}}/F_{\mathcal{N}}(q_i/\sigma_t)$;

5 **tant que** $\widehat{\mathrm{cv}} \geq \delta$ **faire**

6 Calculer $\sigma_{t+1} = \arg\min(\hat{\delta}_t(\sigma) - \delta)^2$ où le minimum est évaluer sur $\sigma \in (0, \sigma_t)$ et $\hat{\delta}_t(\sigma)$ est le coefficient de variation des $F_{\mathcal{N}}(q_i/\sigma)L_t(\mathbf{X}_i)$;

7 Calculer les poids $w_i = w'_i/\sum_j w'_j$ où $w'_i = F_{\mathcal{N}}(q_i/\sigma_{t+1})L_t(\mathbf{X}_i)$;

8 Estimer $\mathbf{m}_{t+1} = \sum_{i=1}^N w_i \mathbf{X}_i$ et poser $\mathbf{d} = \mathbf{m}_{t+1}/\|\mathbf{m}_{t+1}\|$;

9 Projeter les échantillons $Y_i = \mathbf{d}^\top \mathbf{X}_i$ et estimer leur variance conditionnelle $\hat{v} = \sum_{i=1}^N w_i (Y_i - \|\mathbf{m}_{t+1}\|)^2$ et poser $\Sigma_{t+1} = (\hat{v} - 1)\mathbf{d}\mathbf{d}^\top + I_n$;

10 Incrémenter t : $t \longleftarrow t + 1$;

11 Répéter les étapes 2, 3 et 4;

12 **fin**

13 Estimer $\hat{E}_N = \dfrac{1}{N} \displaystyle\sum_{i=1}^N \mathbb{I}_{\{\varphi(\mathbf{X}_i) \geq 0\}} L_t(\mathbf{X}_i)$.

(autrement dit le vecteur propre de Σ^* associé à la valeur propre minimisant ℓ). Nous sommes donc ici dans le cas idéal pour appliquer les algorithmes CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$.

Le tableau 5.3 regroupe les résultats des six algorithmes dans les dimensions $n = 30$, 100, et 300. Comme on a pu l'observer précédemment, avec un si faible budget, CE et iCE sont très imprécis en dimension 30 et ne convergent pas lorsque $n = 100$ et 300, alors que les quatre autres méthodes donnent des résultats très précis. En dimension 30 et 100, le coefficient de variation de ces quatre algorithmes reste inférieur à 7.4% et le biais relatif varie entre -0.5 et 1.1%. On remarque aussi que CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ sont légèrement plus précis que CEd et iCEd. Cette différence de précision est accrue en dimension 300 car les coefficients de variation de iCE-$\mathbf{m}^*$ et iCEd sont de 7.3 et 11% respectivement, et celui de CEd est plus de deux fois supérieur à celui de CE-$\mathbf{m}^*$ (28% contre 13%). Nous suggérons que CE-$\mathbf{m}^*$ est plus performant que CEd en grande dimension car un plus grand nombre de paramètres est mis à jour dans CEd et la plupart de ces paramètres n'a pas d'influence sur l'estimation. De ce fait, il y a plus d'erreurs d'estimation dans CEd que dans CE-$\mathbf{m}^*$ qui estime la variance uniquement dans la direction optimale. Par ailleurs, on peut noter que iCE-$\mathbf{m}^*$ et iCEd sont plus performants que CE-$\mathbf{m}^*$ et CEd respectivement, en ayant un coefficient de variation et un biais relatif souvent inférieur.

Enfin, il faut souligner que ces algorithmes sont très efficaces malgré le faible budget de simulation utilisé. En comparaison, CE-P et iCE-P ne convergent pas avec un tel budget. Les résultats donnés par CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ sont donc très performants, avec un faible budget et jusqu'en dimension 300, dans ce cas particulier où $\mathbf{m}^*$ est exactement la direction optimale pour estimer la variance.

		iCE	iCE-$\mathbf{m}^*$	iCEd	CE	CE-$\mathbf{m}^*$	CEd
$n = 30$ $P = 1.35 \cdot 10^{-3}$	Estim. moyenne ($\times 10^{-3}$)	0.011	1.35	1.36	$5.8 \cdot 10^{-5}$	1.34	1.35
	Coefficient de variation (%)	99	2.6	5.1	100	4.6	5.8
	Biais relatif (%)	−99	−0.1	0.9	−100	−0.5	0.4
	Taille d'échantillon N	1000	2700	3700	1000	2700	3400
$n = 100$ $P = 1.35 \cdot 10^{-3}$	Estim. moyenne ($\times 10^{-3}$)	NC	1.35	1.36	NC	1.35	1.37
	Coefficient de variation (%)	NC	4.6	6.8	NC	4.4	7.4
	Biais relatif (%)	NC	0.1	0.9	NC	−0.1	1.1
	Taille d'échantillon N	NC	2900	3700	NC	2700	3600
$n = 300$ $P = 1.35 \cdot 10^{-3}$	Estim. moyenne ($\times 10^{-3}$)	NC	1.35	1.36	NC	1.34	1.31
	Coefficient de variation (%)	NC	7.3	11	NC	13	28
	Biais relatif (%)	NC	−0.2	0.6	NC	−0.8	−3.2
	Taille d'échantillon N	NC	4000	3700	NC	2700	3700

Tableau 5.3 – Comparaison numérique de l'estimation de la probabilité E avec les six algorithmes (CE-$\mathbf{m}^*$, iCE-$\mathbf{m}^*$, CE, iCE, CEd et iCEd) lorsque la fonction d'état limite est la somme des coordonnées φ_1 (2.1).

5.3.2.2 Application : probabilité de perte élevée d'un portefeuille

Le deuxième exemple est l'application, déjà traitée dans la partie 4.2.3, qui consiste en l'estimation de la probabilité de perte élevée d'un portefeuille d'options financières. La fonction de perte est la fonction φ_6 définie en 4.8. Nous évaluons la performance des algorithmes sur cette fonction dans les dimensions $n = 30$, 100 et 250 (pour $n = 250$, la constante b dans l'expression de φ_6 est égale à 0.3). Ici encore la moyenne optimale a la même direction que le premier vecteur propre de Σ^* (voir section 4.2.3).

		iCE	iCE-m^*	iCEd	CE	CE-m^*	CEd
$n = 30$ $P = 4.29 \cdot 10^{-3}$	Estim. moyenne ($\times 10^{-3}$)	0.0028	4.29	4.38	2.70	4.24	4.32
	Coefficient de variation (%)	100	10.1	8.1	54	7.2	9.0
	Biais relatif (%)	−100	−0.1	2.1	−37	−1.3	0.8
	Taille d'échantillon N	1000	2700	3200	2600	2700	2700
$n = 100$ $P = 1.82 \cdot 10^{-3}$	Estim. moyenne ($\times 10^{-3}$)	NC	1.87	1.85	NC	1.84	1.79
	Coefficient de variation (%)	NC	8.5	7.1	NC	8.4	14
	Biais relatif (%)	NC	3.0	2.0	NC	1.3	−1.8
	Taille d'échantillon N	NC	3000	2700	NC	2700	2700
$n = 250$ $P = 1.0 \cdot 10^{-5}$	Estim. moyenne ($\times 10^{-5}$)	NC	1.01	0.049	NC	0.94	NC
	Coefficient de variation (%)	NC	110	98	NC	60	NC
	Biais relatif (%)	NC	1.0	−95	NC	−5.9	NC
	Taille d'échantillon N	NC	1600	1500	NC	2100	NC

Tableau 5.4 – Comparaison numérique de l'estimation de la probabilité E avec les six algorithmes (CE-$\mathbf{m}^*$, iCE-$\mathbf{m}^*$, CE, iCE, CEd et iCEd) lorsque la fonction d'état limite est la fonction de perte de portefeuille φ_6 (4.8).

En dimension 30, iCE et CE sont déjà très imprécis alors que les quatre autres algorithmes ont un coefficient de variation inférieur à 10% et un biais relatif inférieur à 2% en valeur absolue. Pour

$n = 100$, CE et iCE ne convergent plus, CE-$\mathbf{m}^*$, iCE-$\mathbf{m}^*$ et iCEd sont toujours très performants (coefficient de variation inférieur 8.5% et biais relatif entre 1 et 3%) et CEd est légèrement moins précis avec un coefficient de variation de 14.4%. Lorsqu'on atteint $n = 250$, CEd ne converge pas, à cause de la rapide dégénérescence des poids vers 0, et iCEd est très imprécis (-95% de biais relatif). CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ sont assez imprécis également (coefficient de variation de 60% et 110% respectivement) mais donnent malgré tout une estimation proche de la valeur de référence, avec un biais relatif de -5.9% et 1% respectivement. Si les résultats de CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ sont similaires en dimension 30 et 100, pour $n = 250$, CE-$\mathbf{m}^*$ est un peu plus précis car son coefficient de variation est presque deux fois plus petit que celui de iCE-$\mathbf{m}^*$.

La projection sur le vecteur de la moyenne permet, dans cet exemple aussi, d'améliorer l'efficacité des algorithmes CE et iCE en grande dimension. Il est même préférable de prendre l'estimation de $\mathbf{m}^*$ comme direction de projection pour estimer la covariance plutôt que d'estimer seulement sa diagonale. En effet, en grande dimension, CEd et iCEd estiment beaucoup de paramètres non influents ce qui induit de nombreuses erreurs d'estimation et peut entrainer une rapide dégénérescence des poids.

5.3.2.3 Un exemple utilisé en optimisation : la fonction de Ackley modifiée

La fonction de Ackley (voir [Surjanovic and Bingham, 2021]) est une fonction non convexe, souvent utilisée pour tester des algorithmes en optimisation, que nous utilisons ici pour évaluer la performance de CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ lorsque la dimension augmente. La fonction considérée ici est la suivante :

$$\varphi_7(\mathbf{x}) = 20 \exp\left(-0.2\sqrt{\frac{1}{n}\sum_{j=1}^{n}(a_j x_j - 3)^2}\right) + \exp\left(\frac{1}{n}\sum_{j=1}^{n}\cos\left(2\pi(a_j x_j - 3)\right)\right) - c_n \tag{5.6}$$

avec $\mathbf{a} = (a_1, \ldots, a_n) = \frac{2}{n}(0, 1, 2, \ldots, n-1) \in \mathbb{R}^n$ un vecteur fixé et c_n une constante réelle dépendant de la dimension et ajustée de sorte que la probabilité soit de l'ordre de 10^{-3}. La fonction de Ackley modifiée est tracée sur la figure 5.3, avec sa projection dans le plan $\{x_1 = 0\}$. Le domaine de défaillance correspond aux points (x_1, x_2) tels que $\varphi_7(x_1, x_2) \geq 0$, c'est-à-dire les valeurs de x_2 telles que la courbe bleue est au-dessus de la ligne rouge sur le graphique de droite de la figure 5.3, puisque la fonction ne dépend pas de x_1. La fonction de Ackley standard correspond à l'application φ_7 avec $a_i = 1$ pour tout i. Dans notre cas, on introduit le paramètre $\mathbf{a}$ pour casser la symétrie et éviter que $\mathbf{m}^*$ soit proportionnel à $(1, \ldots, 1)$ comme dans le premier exemple (5.3.2.1). La moyenne optimale $\mathbf{m}^*$, obtenue par Monte-Carlo avec un budget très important, est représentée sur la figure 5.4, avec le vecteur $\mathbf{a}$ (normalisé) en dimension 30. De plus, lorsqu'on calcule les valeurs propres de la matrice Σ^* (estimée avec un très grand budget), l'algorithme 9 donne une seule direction de projection optimale $\mathbf{d}_1^*$, également représentée sur la figure 5.4. La moyenne optimale ne semble donc pas être exactement égale au premier vecteur propre $\mathbf{d}_1^*$, qui lui se rapproche de $\mathbf{a}$, mais n'en est pas très éloignée. Cela peut expliquer l'efficacité des algorithmes CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$, dont les résultats de simulation sont donnés dans le tableau 5.5.

Ces résultats sont qualitativement similaires à ceux de l'exemple 5.3.2.1 avec la somme des coordonnées. En dimension 30, CE et iCE sont très imprécis (biais relatif d'environ -44 et -25% respectivement) alors que les quatre autres algorithmes donnent une bonne estimation avec un biais relatif de 1% ou moins et un coefficient de variation entre 7 et 13%. Lorsque la dimension augmente, CE et iCE ne convergent pas mais les résultats pour CE-$\mathbf{m}^*$, iCE-$\mathbf{m}^*$ et iCEd sont toujours performants, le biais relatif restant entre 1 et 3%, et le coefficient de variation entre 9

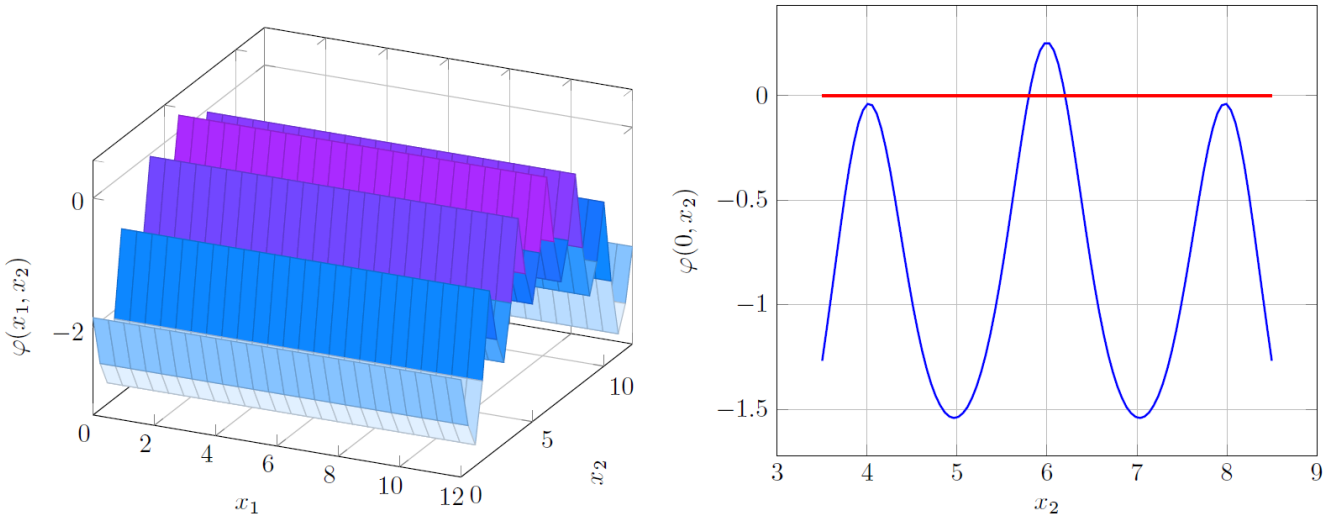


Figure 5.3 – Fonction de Ackley modifiée (5.6) en dimension 2 (à gauche) et sa projection dans le plan $\{x_1 = 0\}$ (à droite). La ligne rouge représente le seuil de défaillance.

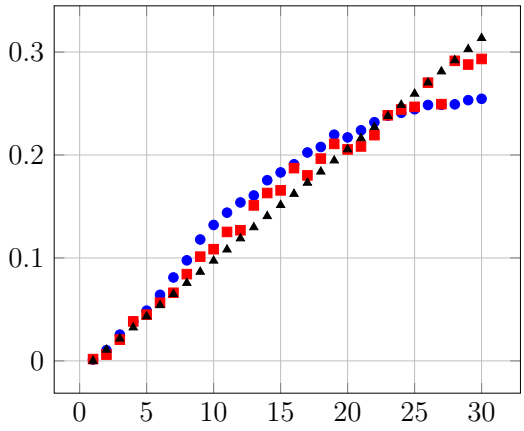


Figure 5.4 – Coordonnées des vecteurs $\mathbf{m}^*/\|\mathbf{m}^*\|$ (cercles bleus), $\mathbf{d}_1^*$ (carrés rouges), et $\mathbf{a}/\|\mathbf{a}\|$ (triangles noirs) en dimension $n = 30$.

et 30%. CEd est très proche de CE-$\mathbf{m}^*$ en dimension $n = 30$ et 100, et même légèrement plus précis, cependant en dimension 200, CEd a un grand coefficient de variation (120%) et un grand biais relatif (5.7%) alors que CE-$\mathbf{m}^*$ reste assez précis. Ainsi, CEd semble moins performant que CE-$\mathbf{m}^*$ en grande dimension car il estime trop de paramètres non influents. Enfin, on peut noter que les méthodes "iCE" sont un peu plus efficaces que les méthodes "CE".

5.3.2.4 Exemple où $\mathbf{m}^*$ n'est pas la direction optimale

Nous terminons cette section avec un exemple où $\mathbf{m}^*$ n'est pas la direction optimale mais où CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ restent suffisamment robustes pour estimer efficacement la probabilité E en grande dimension. On considère pour cela la fonction

$$\varphi_8(\mathbf{x}) = x_1 - 3x_2^2 - 3 \tag{5.7}$$

dont le domaine de défaillance est représenté sur la figure 5.5. On peut montrer facilement que la

		iCE	iCE-m^*	iCEd	CE	CE-m^*	CEd
$n = 30$ $P = 1.64 \cdot 10^{-3}$	Estim. moyenne ($\times 10^{-3}$)	1.23	1.62	1.65	0.92	1.62	1.66
	Coefficient de variation (%)	89	7.6	9.0	75	13	9.9
	Biais relatif (%)	-25	-1.1	0.6	-44	-1.2	0.9
	Taille d'échantillon N	1000	2700	2700	2700	2200	2700
$n = 100$ $P = 1.18 \cdot 10^{-3}$	Estim. moyenne ($\times 10^{-3}$)	NC	1.17	1.19	NC	1.22	1.20
	Coefficient de variation (%)	NC	13	10	NC	29	21
	Biais relatif (%)	NC	-1.2	0.9	NC	3.4	1.6
	Taille d'échantillon N	NC	2700	2700	NC	2200	2700
$n = 200$ $P = 1.72 \cdot 10^{-3}$	Estim. moyenne ($\times 10^{-3}$)	NC	1.71	1.69	NC	1.72	1.82
	Coefficient de variation (%)	NC	9.1	12	NC	27	120
	Biais relatif (%)	NC	-0.5	-1.5	NC	-0.1	5.7
	Taille d'échantillon N	NC	2700	2700	NC	2700	3700

Tableau 5.5 – Comparaison numérique de l'estimation de la probabilité E avec les six algorithmes (CE-$\mathbf{m}^*$, iCE-$\mathbf{m}^*$, CE, iCE, CEd et iCEd) lorsque la fonction d'état limite est la fonction de Ackley modifiée φ_7 (5.6).

moyenne optimale $\mathbf{m}^*$ est de la forme $\mathbf{m}^* = m^*\mathbf{e}_1$ (avec $m^* \approx 3.4$, évalué numériquement) et la variance Σ^* est une matrice diagonale dont les deux premiers termes diagonaux sont $\Sigma^*_{11} \approx 0.095$ et $\Sigma^*_{22} \approx 0.044$, et tous les suivants sont égaux à 1. Comme le domaine de défaillance est assez étroit dans la direction de $\mathbf{e}_2$, il faudrait diminuer la variance dans cette direction pour que suffisamment d'échantillons tombent dans la zone de défaillance. C'est pourquoi on trouve que $\mathbf{e}_2$ est le vecteur propre $\mathbf{d}^*_1$ de Σ^* associé à la plus petite valeur propre et la première des deux directions optimales données par la méthode 4.1.3. Mais $\mathbf{m}^*$ correspond à la seconde direction optimale $\mathbf{d}^*_2$ et est donc malgré tout une direction pertinente dans laquelle projeter la variance. Les algorithmes CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ ne tendent donc pas vers la densité optimale, comme on peut le voir sur le graphique de droite de la figure 5.5. Néanmoins, en grande dimension, CE-$\mathbf{m}^*$ converge et est capable d'estimer la probabilité alors que CE n'y parvient pas à cause de l'effondrement de la matrice de covariance (voir le graphique de gauche de 5.5).

C'est ce qu'on peut observer dans les résultats du tableau 5.6. Comme dans les exemples précédents, CE et iCE sont déjà très imprécis en dimension 30, alors que CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ sont très efficaces (coefficient de variation d'environ 11% et biais relatif entre 0.5 et 1%). L'algorithme iCEd est encore plus précis avec coefficient de variation de 5.2% mais CEd est moins performant (37% de coefficient de variation et -11% de biais relatif). En dimension $n = 100$, iCE-$\mathbf{m}^*$ et iCEd sont encore très précis car leur coefficient de variation reste entre 7 et 12% alors que celui de CE-$\mathbf{m}^*$ et CEd est d'environ 28 et 87% respectivement. CEd est particulièrement imprécis car son biais relatif est de -52% alors qu'il est inférieur à 1.3% pour les trois autres algorithmes convergents. Lorsque $n = 300$, iCEd et CEd ne convergent plus, à cause de la dégénérescence rapide des poids, alors que CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ sont toujours capables d'estimer la probabilité avec un faible biais relatif (-1.5% et 3.5% respectivement) malgré un important coefficient de variation (88% et 29% respectivement), surtout pour CE-$\mathbf{m}^*$.

Ainsi, même si $\mathbf{m}^*$ n'est pas la direction optimale, cet exemple montre que CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ sont capables d'estimer efficacement la probabilité en grande dimension alors que CE, iCE, CEd et iCEd n'y parviennent pas.

Remarque 5.3.2. Si on considère une parabole encore plus étroite, comme pour la fonction φ_4 (4.1),

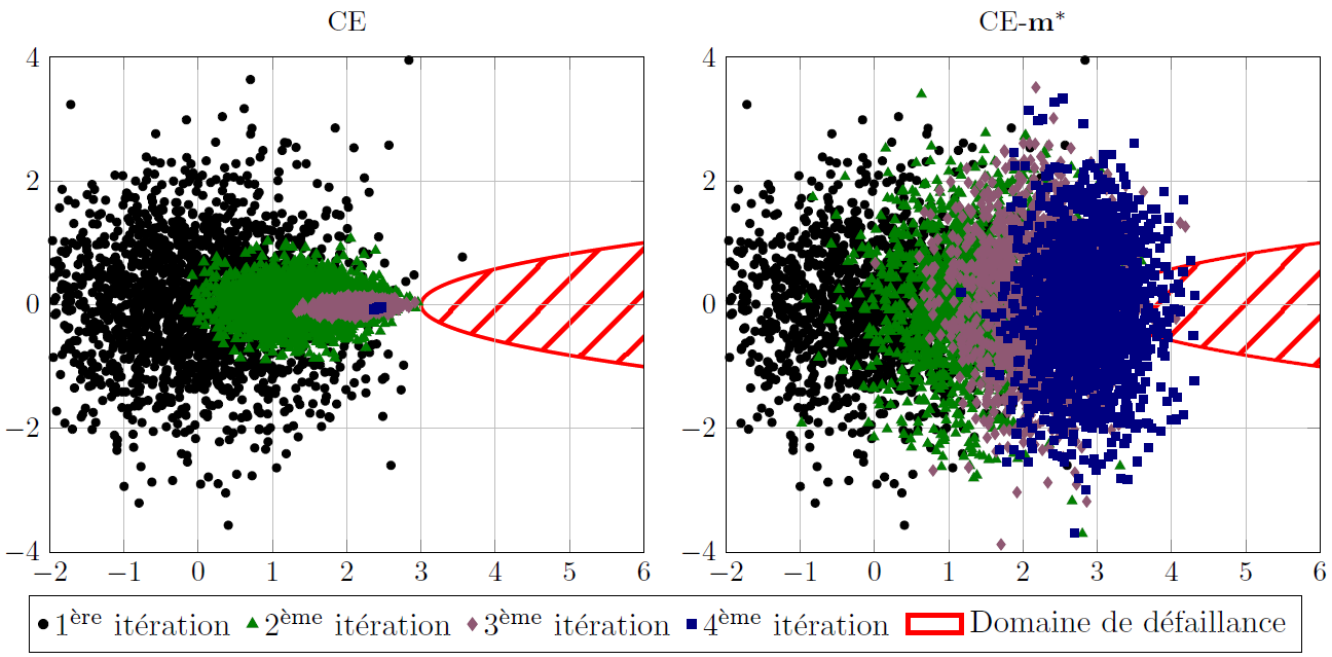


Figure 5.5 – Projection sur le plan (x_1, x_2) du domaine de défaillance de la fonction φ_8 (5.7) et des échantillons de chaque itération de CE (à gauche) et CE-$\mathbf{m}^*$ (à droite), lorsque la dimension vaut $n = 100$. Comme φ_8 ne dépend pas des autres coordonnées, la zone de défaillance est un cylindre autour de cette parabole.

alors CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ ne convergent pas non plus. En effet, dans ce cas trop peu d'échantillons tombent dans le domaine de défaillance car $\mathbf{m}^*$ n'est pas la direction optimale dans laquelle estimer la matrice de covariance. Cependant CE, CEd, iCE et iCEd ne convergent pas en grande dimension sur cet exemple. De plus, nous n'avons jamais trouvé de cas (vérifiant l'hypothèse d'unimodalité) où CE-$\mathbf{m}^*$ était moins performant que CE en grande dimension, et la plupart du temps il améliore nettement les résultats.

Remarque 5.3.3. Dans ce dernier exemple, les algorithmes CE-P et iCE-P donnent de meilleurs résultats que CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ en dimension 30 pour le même budget (coefficients de variation entre 5 et 10% environ). Cela s'explique par le meilleur choix de projection effectué dans CE-P et iCE-P, qui projettent sur la direction optimale $\mathbf{e}_2$. En dimension $n = 100$, CE-P et iCE-P ont une précision équivalente voire légèrement inférieure à CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$ respectivement avec un budget similaire (coefficients de variation autour de 15% pour iCE-P et 35% pour CE-P). La direction optimale est toujours détectée par CE-P et iCE-P, mais ils prennent en compte d'autres directions non influentes du fait de l'estimation imprécise de la matrice de covariance, ce qui contribue à l'augmentation du coefficient de variation. En revanche en dimension 300, avec un budget de simulation d'environ 8000, CE-P et iCE-P ne convergent pas toujours car ils projettent essentiellement dans des directions non influentes. Il faut augmenter suffisamment le budget pour obtenir des résultats précis. Une taille d'échantillon de $N = 6000$ et un budget total d'environ 20000 garantissent la convergence systématique de l'algorithme iCE-P et un coefficient de variation autour de 10%. Pour CE-P, il est nécessaire d'atteindre un budget d'environ 35000 (et $N = 12000$) pour obtenir de tels résultats.

Par ailleurs, dans les exemples 5.3.2.1, 5.3.2.3, et 5.3.2.4, l'algorithme iCEred est plus efficace

		iCE	iCE-m^*	iCEd	CE	CE-m^*	CEd
$n = 30$ $P = 2.9 \cdot 10^{-4}$	Estim. moyenne ($\times 10^{-4}$)	$2 \cdot 10^{-6}$	2.9	2.9	$9 \cdot 10^{-8}$	2.9	2.6
	Coefficient de variation (%)	100	11	5.2	100	11	37
	Biais relatif (%)	-100	1.0	0.2	-100	0.5	-11
	Taille d'échantillon N	1000	2700	2700	1000	1600	2000
$n = 100$ $P = 2.9 \cdot 10^{-4}$	Estim. moyenne ($\times 10^{-4}$)	NC	2.9	2.9	NC	3.0	1.4
	Coefficient de variation (%)	NC	11	7.2	NC	28	87
	Biais relatif (%)	NC	1.1	-0.3	NC	1.3	-52
	Taille d'échantillon N	NC	2700	2600	NC	1900	2000
$n = 300$ $P = 2.9 \cdot 10^{-4}$	Estim. moyenne ($\times 10^{-4}$)	NC	3.0	NC	NC	2.9	NC
	Coefficient de variation (%)	NC	29	NC	NC	88	NC
	Biais relatif (%)	NC	3.5	NC	NC	-1.5	NC
	Taille d'échantillon N	NC	2300	NC	NC	2400	NC

Tableau 5.6 – Comparaison numérique de l'estimation de la probabilité E avec les six algorithmes (CE-$\mathbf{m}^*$, iCE-$\mathbf{m}^*$, CE, iCE, CEd et iCEd) lorsque la fonction d'état limite est le polynôme de degré 2, φ_8 (5.7).

et plus précis que tous les algorithmes proposés dans cette section en grande dimension, le gradient des fonctions considérées étant facile à obtenir. En revanche, l'exemple 5.3.2.2 est un cas où iCEred n'est pas applicable, puisque la fonction n'est pas différentiable, ce qui montre les limites d'une méthode basée sur le gradient.

5.4 Conclusion

Dans ce chapitre, nous avons proposé d'intégrer les méthodes de projection définies dans le chapitre 4 à des algorithmes adaptatifs, en nous concentrant sur l'algorithme d'entropie croisée pour l'estimation de probabilités d'événement rare. Un tel couplage entre CE et projection dans un sous-espace a déjà été effectué dans l'article [Uribe et al., 2021], où ils reprennent la projection suggérée dans [Zahm et al., 2018]. L'algorithme iCEred (11) qu'ils ont développé donne des résultats d'estimation très précis en grande dimension, mais nécessite la connaissance du gradient de la fonction d'état limite. Nous avons donc tenté de coupler la méthode (4.1.3) de projection sur les vecteurs propres de la matrice de covariance avec la CE, afin d'éviter d'utiliser le gradient.

Les algorithmes CE-P (12) et iCE-P (13) ainsi construits améliorent les résultats de CE et iCE en grande dimension, mais ils nécessitent un budget de simulation assez grand pour converger, notamment lorsque la dimension dépasse la centaine. En effet, la matrice de covariance devient trop imprécise dès que la dimension augmente, ses vecteurs propres sont donc mal estimés et les directions de projection sont mal choisies. Les méthodes CE-P et iCE-P sont donc prometteuses mais sont efficaces pour des dimensions modérément grandes (entre 20 et 100) ou avec un budget assez grand (plusieurs dizaines de milliers) lorsque la dimension dépasse 100. En revanche, en utilisant la projection sur la moyenne optimale $\mathbf{m}^*$, les algorithmes CE-$\mathbf{m}^*$ (14) et iCE-$\mathbf{m}^*$ (15) développés dans ce chapitre, améliorent fortement les résultats de CE et iCE, pour de grandes dimensions (jusqu'à environ 300 dans nos exemples) et un petit budget (autour de 8000). Ils sont même souvent plus performants que les algorithmes CEd et iCEd, qui mettent à jour la diagonale de la matrice de covariance. En effet, les résultats suggèrent qu'il vaut parfois mieux projeter

dans une seule direction influente (par exemple $\mathbf{m}^*$), même si elle n'est pas optimale, plutôt que de projeter dans de trop nombreuses directions qui ne sont pas influentes et qui dégradent l'estimation. Cependant, projeter dans un sous-espace de dimension 1 est aussi une limite de cette approche, car certaines directions influentes peuvent aussi être omises.

Ainsi, si les quatre algorithmes proposés (CE-P, iCE-P, CE-$\mathbf{m}^*$ et iCE-$\mathbf{m}^*$) sont généralement moins performants que iCEred, lorsque le gradient de la fonction d'état limite est disponible, ils sont capables d'estimer précisément des probabilités d'événement rare en grande dimension, et sans calcul du gradient.

Conclusion et Perspectives

Résumé des principales contributions L'échantillonnage préférentiel et les algorithmes adaptatifs d'IS sont des outils efficaces pour l'estimation d'espérance et constituent un axe de recherche particulièrement actif ces dernières années, notamment pour les problèmes en grande dimension. En effet, les méthodes basées sur l'IS peuvent devenir inefficaces lorsque la dimension augmente. L'objectif principal de ces travaux de thèse est donc d'améliorer la précision de l'estimation par échantillonnage préférentiel paramétrique en grande dimension, tout en gardant un budget de simulation limité. Pour ce faire, nous proposons d'utiliser des projections dans des sous-espaces de petite dimension afin de réduire le nombre de paramètres (gaussiens) estimés dans l'IS. Comme la majorité des coefficients proviennent de la matrice de covariance, le but est de diminuer le nombre de paramètres estimés dans cette matrice en particulier.

La première contribution de ce manuscrit (chapitre 3) vise ainsi à justifier la pertinence d'une projection dans le cadre de l'échantillonnage préférentiel avec des lois gaussiennes. Nous avons montré dans un cadre simple que projeter les paramètres peut en effet entrainer une diminution de la divergence de Kullback-Leibler et donc potentiellement de l'erreur d'estimation. Ce résultat a été confirmé par des simulations numériques dans lesquelles des directions de projection naïves ont été utilisées pour approcher la matrice de covariance. Ces projections, choisies aléatoirement d'une part ou en prenant les directions canoniques d'autre part, permettent en effet de réduire le nombre de coefficients à estimer dans la matrice. Cela implique systématiquement une diminution de la divergence Kullback-Leibler et de l'erreur d'estimation par rapport à la matrice empirique (où tous les coefficients sont estimés) lorsque la dimension est élevée. Néanmoins, ces choix naïfs ne mènent pas toujours à des résultats très précis et la prise en compte des données du problèmes pour sélectionner des directions adaptées à chaque cas-test semble nécessaire pour améliorer encore les résultats.

La deuxième contribution (chapitre 4) consiste alors à déterminer des directions de projection influentes pour estimer la matrice de covariance. La première idée proposée est d'exploiter la direction donnée par la moyenne de la loi gaussienne pour projeter. Notons que cette suggestion est particulièrement pertinente dans l'estimation de probabilités d'événements rares. En effet dans ce cas, la variance estimée diminue dans la direction indiquée par la moyenne. Ce comportement est lié à la propriété de queue légère de la loi gaussienne. Cette première proposition est néanmoins limitée à une projection en dimension 1, ce qui peut être restrictif dans certains cas. La seconde proposition offre ainsi la possibilité de projeter sur plusieurs directions qui correspondent aux directions optimales déterminées en minimisant la divergence de Kullback-Leibler. Ces directions sont égales aux vecteurs propres de la matrice de covariance qui contribuent le plus à réduire la divergence Kullback-Leibler. Les deux idées de projection ont ensuite été testées numériquement sur des exemples d'estimation dans un cadre théorique. Les résultats de simulation ont montré dans les deux cas une amélioration de l'estimation par IS par rapport aux simulations sans projection ou aux projections naïves du chapitre 3. Cette contribution a fait l'objet d'un article actuellement

en cours de révision dans la revue *SIAM/ASA Journal on Uncertainty Quantification* :

> El Masri, M., Morio, J., and Simatos, F. (2021). Optimal projection to improve parametric importance sampling in high dimension. *arXiv preprint arXiv :2107.06091.*

La troisième et dernière contribution (chapitre 5) a consisté à intégrer les méthodes de projection développées à un algorithme adaptatif d'IS. En effet, en pratique il est souvent nécessaire d'estimer les paramètres progressivement, en passant par plusieurs étapes intermédiaires. Les deux méthodes de projection mises en place dans le chapitre 4 ont ainsi été couplées à l'algorithme d'entropie croisée (CE), et sa version améliorée (iCE), pour estimer des probabilités d'événements rares. Quatre algorithmes ont alors été testés numériquement sur des exemples analytiques en grande dimension et comparés aux algorithmes originaux de CE et d'iCE. Toutes les nouvelles méthodes ont donné des résultats d'estimation plus précis que CE et iCE. Cependant, la technique de projection sur les vecteurs propres de la matrice de covariance n'est efficace que pour des dimensions modérément grandes, de plusieurs dizaines environ. Cette limite est due au fait que les vecteurs propres dépendent de l'estimation de la covariance qui est elle-même imprécise lorsque la dimension est trop grande. Les algorithmes couplés à la projection dans le sous-espace engendré par la moyenne restent en revanche très performants dans des dimensions de quelques centaines. Ces résultats sont encourageants et ouvrent des perspectives intéressantes pour l'estimation en grande dimension de probabilités d'événements rares, ou plus généralement d'une espérance quelconque. Cette contribution est liée à l'article de journal suivant :

> El Masri, M., Morio, J., and Simatos, F. (2021). Improvement of the cross-entropy method in high dimension for failure probability estimation through a one-dimensional projection without gradient estimation. *Reliability Engineering & System Safety*, 216 :107991.

Perspectives La première piste d'amélioration concerne l'estimation des vecteurs propres de la matrice de covariance. En effet, ceux-ci sont imprécis lorsque la dimension est très grande puisque la covariance est elle-même mal estimée, et l'estimation finale de l'espérance est nettement dégradée. Deux idées principales peuvent être exploitées pour améliorer l'estimation des vecteurs propres. La première consiste à utiliser des estimateurs de la matrice de covariance plus robustes en grande dimension comme ceux évoqués dans la section 2.2, et suggérés dans [Ledoit and Wolf, 2004] et [Ashurbekova et al., 2020] par exemple. Ces estimateurs, basés sur des techniques de "*shrinkage*" ou contraction des paramètres, sont plus précis que la matrice empirique lorsqu'elle est de grande taille, ce qui pourrait également augmenter la précision des vecteurs propres. Néanmoins, le comportement de ce type d'estimateurs à l'intérieur d'un algorithme adaptatif d'IS reste à étudier, notamment lorsqu'ils sont calculés avec des poids d'importance et donc potentiellement confrontés au problème de dégénérescence des poids. La seconde idée pour améliorer l'estimation des vecteurs propres de la covariance est de directement faire appel à des méthodes d'estimation des valeurs et vecteurs propres en grande dimension proposées dans [Mestre, 2008a], [Nadakuditi and Edelman, 2008] ou [Benaych-Georges and Nadakuditi, 2011] par exemple. Ces techniques assurent une estimation efficace des éléments propres d'une matrice de covariance de grande taille et permettraient également d'améliorer la précision de la méthode de projection sur les vecteurs propres optimaux développée dans cette thèse. Comme précédemment, la question du comportement de ces approches à l'intérieur d'un algorithme adaptatif se pose et est à étudier.

Une deuxième piste pour améliorer les algorithmes mis en place dans ce manuscrit est d'étendre nos méthodes à des problèmes multimodaux. Pour ce faire, l'utilisation d'un mélange de plusieurs densités gaussiennes comme densité auxiliaire peut être envisagé. C'est un choix courant dans la

littérature d'AIS, notamment dans l'algorithme d'entropie croisée avec [Kurtz and Song, 2013] et [Geyer et al., 2019]. Une difficulté qui apparait dans les problèmes multimodaux est d'abord l'identification des modes et le choix du nombre de densités du mélange. Des méthodes de *clustering*, ou partitionnement, peuvent être envisagées pour détecter les modes, comme suggéré dans [Geyer et al., 2019]. Les techniques de projection proposées dans cette thèse pourraient alors être adaptées pour projeter les paramètres de chaque densité gaussienne du mélange. Avoir des algorithmes capables de résoudre des problèmes d'estimation multimodaux permettrait ainsi de traiter des cas-tests réels plus complexes en ingénierie ou dans d'autres domaines.

Ensuite, il serait intéressant de s'attaquer au problème de dégénérescence des poids en utilisant les techniques de projection. En effet, nous ne traitons pas directement ce problème dans cette thèse même si nos algorithmes semblent atténuer la dégénérescence en estimant une matrice de covariance dans un sous-espace de petite dimension. Une analyse du comportement des poids sous l'effet des projections pourrait permettre de comprendre comment éviter leur dégénérescence.

Enfin, le couplage de techniques de projection avec une méthode d'échantillonnage préférentiel non-paramétrique, inefficace en grande dimension, est une perspective intéressante. En effet, la mise en place d'algorithmes améliorant leur performance permettrait de traiter des problèmes particulièrement complexes, notamment les cas multimodaux et fortement non linéaire, en bénéficiant de la flexibilité des approches non-paramétriques.

Annexe A

Annexe

A.1 Calcul des paramètres gaussiens optimaux d'IS sur un exemple jouet

Dans le cas où la fonction d'intérêt est $\mathbb{I}_{\{\varphi_1 \geq 0\}}$ avec φ_1 la fonction somme des coordonnées (2.1), les paramètres optimaux, $\mathbf{m}^*$ et Σ^* (1.10), peuvent être calculés explicitement.

• *Calcul de* $\mathbf{m}^*$: Par symétrie de f (la densité $\mathcal{N}(0, I_n)$) et de φ_1 (toutes les variables peuvent être permutées sans modifier la valeur des fonctions), on a déjà $\mathbf{m}^* = m^* \mathbf{1}_n$, avec $\mathbf{1}_n = n^{-1/2}(1, \ldots, 1)^\top$ et $m^* = \sqrt{n}\mathbb{E}_f(X_1 |\ \varphi_1(\mathbf{X}) \geq 0)$. Cette dernière espérance conditionnelle vaut alors (pour $\beta = 3$) :

$$\begin{aligned}
\mathbb{E}_f(X_1 |\ \varphi_1(\mathbf{X}) \geq 0) &= \mathbb{E}_f\left(X_1 |\ X_1 \geq -\sum_{i=2}^{n} X_i + \beta\sqrt{n}\right) \\
&= \frac{1}{E}\mathbb{E}\left(X_1 \mathbb{I}_{X_1 \geq -Z\sqrt{n-1}+\beta\sqrt{n}}\right),
\end{aligned}$$

où Z est une variable aléatoire de loi $\mathcal{N}(0,1)$, indépendante de X_1 (de loi $\mathcal{N}(0,1)$ également), et où on rappelle que $E = \mathbb{P}_f(\varphi_1(\mathbf{X}) \geq 0)$. La constante β est fixée à 3 dans la fonction φ_1 pour les applications numériques, mais les calculs théoriques présentés ici sont effectués avec $\beta \in \mathbb{R}$ quelconque. En posant $s(z) = \beta\sqrt{n} - z\sqrt{n-1}$, on a ensuite :

$$\begin{aligned}
\mathbb{E}\left(X_1 \mathbb{I}_{X_1 \geq s(Z)}\right) &= \int_{\mathbb{R}} \int_{s(z)}^{+\infty} x_1 \exp(-x_1^2/2) \mathrm{d}x_1 \ \exp(-z^2/2) \frac{\mathrm{d}z}{2\pi} \\
&= \int_{\mathbb{R}} \exp\left(-s(z)^2/2\right) \exp(-z^2/2) \frac{\mathrm{d}z}{2\pi} \\
&= \int_{\mathbb{R}} \exp\left(-\frac{1}{2}\left(\beta^2 n - 2\beta z\sqrt{n}\sqrt{n-1} + z^2 n\right)\right) \frac{\mathrm{d}z}{2\pi} \\
&= \int_{\mathbb{R}} \exp\left(-\frac{1}{2}\left(z\sqrt{n} - \beta\sqrt{n-1}\right)^2\right) \exp\left(-\frac{\beta^2}{2}\right) \frac{\mathrm{d}z}{2\pi} \\
&= \int_{\mathbb{R}} \exp\left(-\frac{1}{2}\left(z' - \beta\sqrt{n-1}\right)^2\right) \exp\left(-\frac{\beta^2}{2}\right) \frac{\mathrm{d}z'}{2\pi\sqrt{n}} \qquad \text{(en posant } z' = z\sqrt{n}\text{)} \\
&= \frac{\exp\left(-\beta^2/2\right)}{\sqrt{2\pi n}},
\end{aligned}$$

la dernière égalité étant obtenue en utilisant le fait que la fonction : $z' \mapsto \exp\left(-\frac{1}{2}\left(z' - \beta\sqrt{n-1}\right)^2\right)(2\pi)^{-1/2}$ est une densité gaussienne, donc son intégrale vaut 1.

Finalement, on trouve $\boxed{m^* = \dfrac{\exp\left(-\beta^2/2\right)}{E\sqrt{2\pi}}}$.

• *Calcul de* Σ^* : Concernant la matrice Σ^*, on a $\Sigma^* = \mathbb{E}_f(\mathbf{X}\mathbf{X}^\top |\, \varphi_1(\mathbf{X}) \geq 0) - \mathbf{m}^*(\mathbf{m}^*)^\top$. Par des arguments de symétrie, on peut vérifier que l'espérance conditionnelle $\mathbb{E}_f(\mathbf{X}\mathbf{X}^\top |\, \varphi_1(\mathbf{X}) \geq 0)$ est une matrice dont tous les coefficients diagonaux sont égaux et les coefficients hors de la diagonale sont égaux entre eux. Il y a ainsi deux quantités à calculer, $\mathbb{E}_f(X_1^2 |\, \varphi_1(\mathbf{X}) \geq 0)$ et $\mathbb{E}_f(X_1X_2 |\, \varphi_1(\mathbf{X}) \geq 0)$. Pour la première, en reprenant la fonction s posée précédemment, on a :

$$\begin{aligned}\mathbb{E}_f(X_1^2 |\, \varphi_1(\mathbf{X}) \geq 0) &= \mathbb{E}(X_1^2 | X_1 \geq s(Z)) \\ &= \frac{1}{E}\int_{\mathbb{R}}\int_{s(z)}^{+\infty} x_1^2 \exp\left(-x_1^2/2\right)\exp\left(-z^2/2\right)\frac{\mathrm{d}x_1\mathrm{d}z}{2\pi}.\end{aligned}$$

L'intégrale par rapport à x_1 se calcule par une intégration par parties et donne :

$$\begin{aligned}\mathbb{E}_f(X_1^2 |\, \varphi_1(\mathbf{X}) \geq 0) &= \frac{1}{E}\int_{\mathbb{R}}\left[s(z)\exp\left(-s(z)^2/2\right) + E\right]\exp\left(-z^2/2\right)\frac{\mathrm{d}z}{2\pi} \\ &= \frac{\exp(-\beta^2/2)}{E\sqrt{2\pi}}\int_{\mathbb{R}}\left(\beta\sqrt{n} - z\sqrt{n-1}\right)\exp\left(-\frac{1}{2}\left(z\sqrt{n} - \beta\sqrt{n-1}\right)^2\right)\frac{\mathrm{d}z}{\sqrt{2\pi}} + 1 \\ &= m^*\int_{\mathbb{R}}\left(\beta\sqrt{n} - \frac{z'\sqrt{n-1}}{\sqrt{n}}\right)\exp\left(-\frac{1}{2}\left(z' - \beta\sqrt{n-1}\right)^2\right)\frac{\mathrm{d}z'}{\sqrt{2\pi n}} + 1 \\ &= m^*\left(\beta - \frac{\sqrt{n-1}}{n}\mathbb{E}\left(Z'\right)\right) + 1.\end{aligned}$$

L'espérance dans la dernière égalité est celle d'une gaussienne $\mathcal{N}(\beta\sqrt{n-1}, 1)$ et vaut donc $\beta\sqrt{n-1}$. On obtient finalement $\underline{\mathbb{E}_f(X_1^2 |\, \varphi_1(\mathbf{X}) \geq 0) = \dfrac{\beta m^*}{n} + 1}$.

La deuxième espérance à calculer est

$$\begin{aligned}\mathbb{E}_f(X_1X_2 |\, \varphi_1(\mathbf{X}) \geq 0) &= \mathbb{E}\left(X_1X_2 |\, X_1 + X_2 \geq \beta\sqrt{n} - \sum_{i=3}^{n} X_i\right) \\ &= \frac{1}{E}\mathbb{E}\left(X_1X_2\mathbb{I}_{X_1+X_2 \geq t(Z)}\right),\end{aligned}$$

où $t(z) = \beta\sqrt{n} - z\sqrt{n-2}$, et Z suit une loi normale $\mathcal{N}(0,1)$, indépendante de X_1 et X_2 toutes deux gaussiennes standards également. On a ensuite,

$$\begin{aligned}\mathbb{E}_f(X_1X_2 |\, \varphi_1(\mathbf{X}) \geq 0) &= \frac{1}{E}\int_{\mathbb{R}}\int_{\mathbb{R}}\int_{t(z)-x_2}^{+\infty} x_1\exp\left(-x_1^2/2\right)x_2\exp\left(-x_2^2/2\right)\exp\left(-z^2/2\right)\frac{\mathrm{d}x_1\mathrm{d}x_2\mathrm{d}z}{(2\pi)^{3/2}} \\ &= \frac{1}{E}\int_{\mathbb{R}}\int_{\mathbb{R}}\exp\left(-(t(z)-x_2)^2/2\right)x_2\exp\left(-x_2^2/2\right)\exp\left(-z^2/2\right)\frac{\mathrm{d}x_2\mathrm{d}z}{(2\pi)^{3/2}} \\ &= \frac{1}{E}\int_{\mathbb{R}}\int_{\mathbb{R}}x_2\exp\left(-\frac{1}{2}\left(\frac{t(z)}{\sqrt{2}} - \sqrt{2}x_2\right)^2\right)\exp\left(-t(z)^2/4\right)\exp\left(-z^2/2\right)\frac{\mathrm{d}x_2\mathrm{d}z}{(2\pi)^{3/2}} \\ &= \frac{1}{E}\int_{\mathbb{R}}\int_{\mathbb{R}}\frac{x_2'}{\sqrt{2}}\exp\left(-\frac{1}{2}\left(\frac{t(z)}{\sqrt{2}} - x_2'\right)^2\right)\exp\left(-t(z)^2/4\right)\exp\left(-z^2/2\right)\frac{\mathrm{d}x_2'\mathrm{d}z}{(2\pi)^{3/2}\sqrt{2}} \\ &= \frac{1}{E}\int_{\mathbb{R}}\mathbb{E}(X_2')\exp\left(-t(z)^2/4\right)\exp\left(-z^2/2\right)\frac{\mathrm{d}z}{4\pi}\end{aligned}$$

avec X_2' de loi $\mathcal{N}\left(\frac{t(z)}{\sqrt{2}}, 1\right)$, donc $\mathbb{E}(X_2') = t(z)/\sqrt{2}$ et alors :

$$\mathbb{E}_f(X_1X_2|\ \varphi_1(\mathbf{X}) \geq 0) = \frac{1}{4E\sqrt{\pi}}\mathbb{E}\left(t(Z)\exp\left(-t(Z)^2/4\right)\right).$$

Enfin, cette dernière espérance vaut :

$$\begin{aligned}
\mathbb{E}\left(t(Z)\exp\left(-t(Z)^2/4\right)\right) &= \int_{\mathbb{R}}(\beta\sqrt{n} - z\sqrt{n-2})\exp\left(-(\beta\sqrt{n} - z\sqrt{n-2})^2/4\right)\exp\left(-z^2/2\right)\frac{\mathrm{d}z}{\sqrt{2\pi}} \\
&= \int_{\mathbb{R}}(\beta\sqrt{n} - z\sqrt{n-2})\exp\left(-(\beta\sqrt{n-2} - z\sqrt{n})^2/4\right)\exp\left(-\beta^2/2\right)\frac{\mathrm{d}z}{\sqrt{2\pi}} \\
&= \int_{\mathbb{R}}\left(\beta\sqrt{n} - z'\frac{\sqrt{n-2}}{\sqrt{n}}\right)\exp\left(-(\beta\sqrt{n-2} - z')^2/4\right)\exp\left(-\beta^2/2\right)\frac{\mathrm{d}z'}{\sqrt{2n\pi}}.
\end{aligned}$$

La fonction $z' \mapsto \exp\left(-(\beta\sqrt{n-2} - z')^2/4\right)(2\sqrt{\pi})^{-1}$ est la densité de la loi $\mathcal{N}(\beta\sqrt{n-2}, 2)$ donc en mettant $\sqrt{2}$ en facteur on obtient

$$\mathbb{E}\left(t(Z)\exp\left(-t(Z)^2/4\right)\right) = \exp(-\beta^2/2)\sqrt{2}\left(\beta - \frac{\sqrt{n-2}}{n}\mathbb{E}(Z')\right)$$

avec Z' de loi $\mathcal{N}(\beta\sqrt{n-2}, 2)$, donc $\mathbb{E}(Z') = \beta\sqrt{n-2}$ et alors

$$\mathbb{E}\left(t(Z)\exp\left(-t(Z)^2/4\right)\right) = \frac{2\sqrt{2}}{n}\beta\exp(-\beta^2/2).$$

Finalement, on a $\mathbb{E}_f(X_1X_2|\ \varphi_1(\mathbf{X}) \geq 0) = \frac{\beta}{n}m^*$, et l'espérance $\mathbb{E}_f(\mathbf{X}\mathbf{X}^\top|\ \varphi_1(\mathbf{X}) \geq 0)$ est la matrice $\beta m^*\mathbf{1}_n\mathbf{1}_n^\top + I_n$, auquel il faut retrancher $\mathbf{m}^*(\mathbf{m}^*)^\top = (m^*)^2\mathbf{1}_n\mathbf{1}_n^\top$ pour avoir Σ^*. En notant $v^* = \beta m^* - (m^*)^2 + 1$, on obtient alors $\boxed{\Sigma^* = (v^* - 1)\mathbf{1}_n\mathbf{1}_n^\top + I_n}$ (la notation étant choisie de sorte que Σ^* soit de la forme Σ_k (3.4) du chapitre 3).

A.2 Échantillon généré selon la loi "*banana shape*"

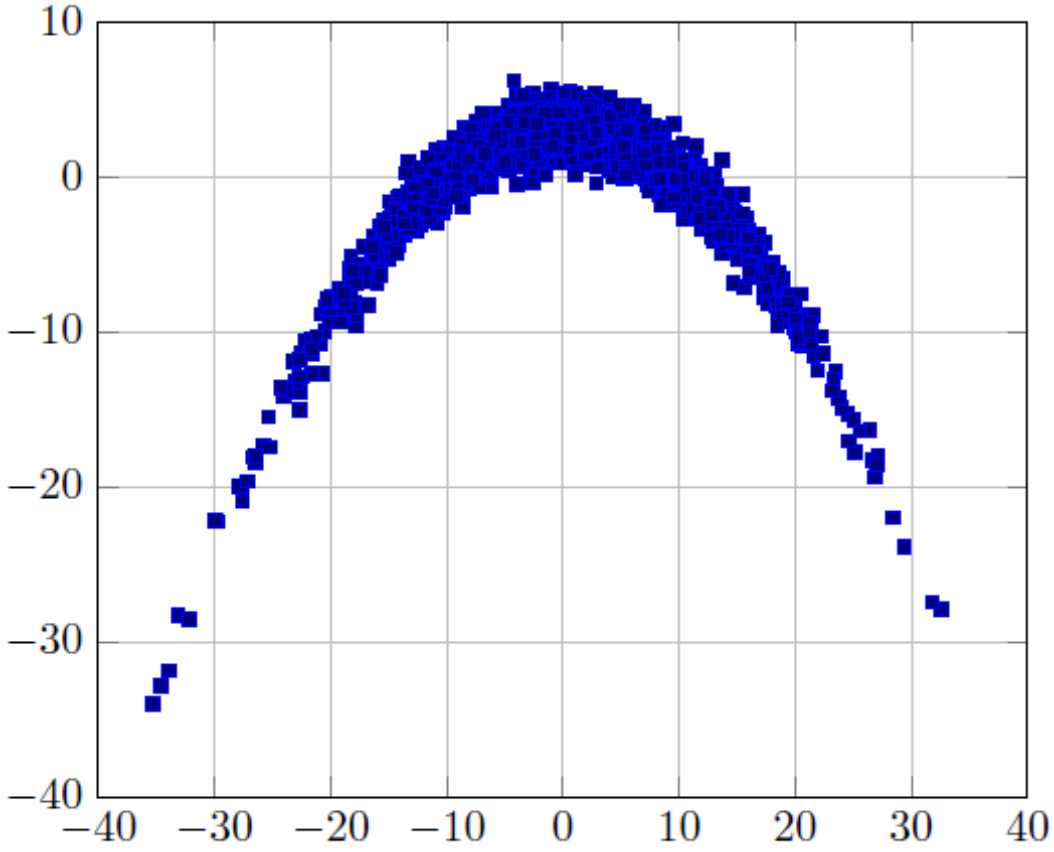


Figure A.1 – Échantillon de taille $N = 2000$ généré selon la loi "*banana shape*" en dimension 2.

Bibliographie

[Ashurbekova et al., 2020] Ashurbekova, K., Usseglio-Carleve, A., Forbes, F., and Achard, S. (2020). Optimal shrinkage for robust covariance matrix estimators in a small sample size setting. hal-02378034v3f.

[Au and Beck, 2003] Au, S. and Beck, J. (2003). Important sampling in high dimensions. *Structural Safety*, 25(2) :139–163.

[Au and Beck, 2001] Au, S.-K. and Beck, J. L. (2001). Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, 16(4) :263–277.

[Bassamboo et al., 2008] Bassamboo, A., Juneja, S., and Zeevi, A. (2008). Portfolio Credit Risk with Extremal Dependence : Asymptotic Analysis and Efficient Simulation. *Operations Research*, 56(3) :593–606.

[Benaych-Georges and Nadakuditi, 2011] Benaych-Georges, F. and Nadakuditi, R. R. (2011). The eigenvalues and eigenvectors of finite, low rank perturbations of large random matrices. *Advances in Mathematics*, 227(1) :494–521.

[Bengtsson et al., 2008] Bengtsson, T., Bickel, P., and Li, B. (2008). Curse-of-dimensionality revisited : Collapse of the particle filter in very large scale systems. In *Institute of Mathematical Statistics Collections*, pages 316–334. Institute of Mathematical Statistics, Beachwood, Ohio, USA.

[Botev et al., 2013] Botev, Z. I., L'Ecuyer, P., and Tuffin, B. (2013). Markov chain importance sampling with applications to rare event probability estimation. *Statistics and Computing*, 23(2) :271–285.

[Bourinet, 2018] Bourinet, J.-M. (2018). Reliability analysis and optimal design under uncertainty - Focus on adaptive surrogate-based approaches. Habilitation à diriger des recherches. Université Clermont Auvergne.

[Bugallo et al., 2017] Bugallo, M. F., Elvira, V., Martino, L., Luengo, D., Miguez, J., and Djuric, P. M. (2017). Adaptive Importance Sampling : The past, the present, and the future. *IEEE Signal Processing Magazine*, 34(4) :60–79.

[Cappé et al., 2008] Cappé, O., Douc, R., Guillin, A., Marin, J.-M., and Robert, C. P. (2008). Adaptive importance sampling in general mixture classes. *Statistics and Computing*, 18(4) :447–459.

[Cappé et al., 2004] Cappé, O., Guillin, A., Marin, J. M., and Robert, C. P. (2004). Population Monte Carlo. *Journal of Computational and Graphical Statistics*, 13(4) :907–929.

[Cérou et al., 2012] Cérou, F., Del Moral, P., Furon, T., and Guyader, A. (2012). Sequential Monte Carlo for rare event estimation. *Statistics and computing*, 22(3) :795–808.

[Cérou and Guyader, 2007] Cérou, F. and Guyader, A. (2007). Adaptive multilevel splitting for rare event analysis. *Stochastic Analysis and Applications*, 25(2) :417–443.

[Chan and Kroese, 2012] Chan, J. C. C. and Kroese, D. P. (2012). Improved cross-entropy method for estimation. *Statistics and Computing*, 22(5) :1031–1040.

[Chatterjee and Diaconis, 2018] Chatterjee, S. and Diaconis, P. (2018). The sample size required in importance sampling. *The Annals of Applied Probability*, 28(2) :1099–1135.

[Chen et al., 2011] Chen, Y., Wiesel, A., and Hero, A. O. (2011). Robust Shrinkage Estimation of High-Dimensional Covariance Matrices. *IEEE Transactions on Signal Processing*, 59(9) :4097–4107.

[Chib and Greenberg, 1995] Chib, S. and Greenberg, E. (1995). Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4) :327–335.

[Constantine, 2015] Constantine, P. G. (2015). *Active Subspaces : Emerging Ideas for Dimension Reduction in Parameter Studies*. Number 2 in SIAM Spotlights. Society for Industrial and Applied Mathematics, Philadelphia.

[Cornuet et al., 2012] Cornuet, J.-M., Marin, J.-M., Mira, A., and Robert, C. P. (2012). Adaptive Multiple Importance Sampling. *Scandinavian Journal of Statistics*, 39(4) :798–812.

[Der Kiureghian et al., 2005] Der Kiureghian, A. et al. (2005). First-and second-order reliability methods. *Engineering design reliability handbook*, 14.

[Dimov, 2008] Dimov, I. T. (2008). *Monte Carlo methods for applied scientists*. World Scientific.

[Ditlevsen and Madsen, 1996] Ditlevsen, O. and Madsen, H. O. (1996). *Structural Reliability Methods*. Wiley, Chichester ; New York.

[Doucet et al., 2009] Doucet, A., Johansen, A. M., et al. (2009). A tutorial on particle filtering and smoothing : Fifteen years later. *Handbook of nonlinear filtering*, 12(656-704) :3.

[El-Laham et al., 2019] El-Laham, Y., Elvira, V., and Bugallo, M. (2019). Recursive Shrinkage Covariance Learning in Adaptive Importance Sampling. In *2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 624–628. IEEE.

[El-Laham et al., 2018] El-Laham, Y., Elvira, V., and Bugallo, M. F. (2018). Robust Covariance Adaptation in Adaptive Importance Sampling. *IEEE Signal Processing Letters*, 25(7) :1049–1053.

[El Masri et al., 2021] El Masri, M., Morio, J., and Simatos, F. (2021). Improvement of the cross-entropy method in high dimension for failure probability estimation through a one-dimensional projection without gradient estimation. *Reliability Engineering & System Safety*, 216 :107991.

[Elvira et al., 2016] Elvira, V., Martino, L., Luengo, D., and Bugallo, M. F. (2016). Improving Population Monte Carlo : Alternative Weighting and Resampling Schemes. *arXiv :1607.02758 [stat]*.

[Elvira et al., 2019] Elvira, V., Martino, L., Luengo, D., and Bugallo, M. F. (2019). Generalized Multiple Importance Sampling. *Statistical Science*, 34(1) :129–155.

[Elvira et al., 2015] Elvira, V., Martino, L., Luengo, D., and Corander, J. (2015). A gradient adaptive population importance sampler. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4075–4079, South Brisbane, Queensland, Australia. IEEE.

[Engelund and Rackwitz, 1993] Engelund, S. and Rackwitz, R. (1993). A benchmark study on importance sampling techniques in structural reliability. *Structural safety*, 12(4) :255–276.

[Geman and Geman, 1984] Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6) :721–741.

[Geyer et al., 2019] Geyer, S., Papaioannou, I., and Straub, D. (2019). Cross entropy-based importance sampling using Gaussian densities revisited. *Structural Safety*, 76 :15–27.

[Glasserman, 2004] Glasserman, P. (2004). *Monte Carlo methods in financial engineering*, volume 53. Springer.

[Grace et al., 2014] Grace, A. W., Kroese, D. P., and Sandmann, W. (2014). Automated State-Dependent Importance Sampling for Markov Jump Processes via Sampling from the Zero-Variance Distribution. *Journal of Applied Probability*, 51(3) :741–755.

[Hastings, 1970] Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications.

[Hesterberg, 1988] Hesterberg, T. C. (1988). *Advances in importance sampling*. PhD thesis, Stanford University.

[Hinrichs et al., 2014] Hinrichs, A., Novak, E., Ullrich, M., and Woźniakowski, H. (2014). The curse of dimensionality for numerical integration of smooth functions ii. *Journal of Complexity*, 30(2) :117–143.

[Hohenbichler and Rackwitz, 1981] Hohenbichler, M. and Rackwitz, R. (1981). Non-Normal Dependent Vectors in Structural Safety. *Journal of the Engineering Mechanics Division*, 107(6) :1227–1238.

[Ihler et al., 2005] Ihler, A. T., Fisher, J. W., Moses, R. L., and Willsky, A. S. (2005). Nonparametric belief propagation for self-localization of sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4) :809–819.

[Ionides, 2008] Ionides, E. L. (2008). Truncated importance sampling. *Journal of Computational and Graphical Statistics*, 17(2) :295–311.

[Kahn, 1950] Kahn, H. (1950). Random sampling (Monte Carlo) techniques in neutron attenuation problems. I. *Nucleonics (US) Ceased publication*, 6(See also NSA 3-990).

[Katafygiotis and Zuev, 2008] Katafygiotis, L. and Zuev, K. (2008). Geometric insight into the challenges of solving high-dimensional reliability problems. *Probabilistic Engineering Mechanics*, 23(2-3) :208–218.

[Katafygiotis and Zuev, 2007] Katafygiotis, L. S. and Zuev, K. M. (2007). Estimation of small failure probabilities in high dimensions by adaptive linked importance sampling. *COMPDYN 2007*.

[Kawai, 2018] Kawai, R. (2018). Optimizing Adaptive Importance Sampling by Stochastic Approximation. *SIAM Journal on Scientific Computing*, 40(4) :A2774–A2800.

[Kim et al., 2000] Kim, Y. B., Roh, D. S., and Lee, M. Y. (2000). Nonparametric adaptive importance sampling for rare event simulation. In *2000 Winter Simulation Conference Proceedings (Cat. No. 00CH37165)*, volume 1, pages 767–772. IEEE.

[Koblents and Míguez, 2015] Koblents, E. and Míguez, J. (2015). A population Monte Carlo scheme with transformed weights and its application to stochastic kinetic models. *Statistics and Computing*, 25(2) :407–425.

[Kong et al., 1994] Kong, A., Liu, J. S., and Wong, W. H. (1994). Sequential imputations and Bayesian missing data problems. *Journal of the American statistical association*, 89(425) :278–288.

[Kroese et al., 2013] Kroese, D. P., Rubinstein, R. Y., and Glynn, P. W. (2013). The Cross-Entropy Method for Estimation. In *Handbook of Statistics*, volume 31, pages 19–34. Elsevier.

[Kullback and Leibler, 1951] Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The annals of mathematical statistics*, 22(1) :79–86.

[Kurtz and Song, 2013] Kurtz, N. and Song, J. (2013). Cross-entropy-based adaptive importance sampling using Gaussian mixture. *Structural Safety*, 42 :35–44.

[L'Ecuyer and Lemieux, 2002] L'Ecuyer, P. and Lemieux, C. (2002). Recent advances in randomized quasi-Monte Carlo methods. *Modeling uncertainty*, pages 419–474.

[Ledoit and Wolf, 2004] Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2) :365–411.

[Liu and Der Kiureghian, 1986] Liu, P.-L. and Der Kiureghian, A. (1986). Multivariate distribution models with prescribed marginals and covariances. *Probabilistic Engineering Mechanics*, 1(2) :105–112.

[Liu and Der Kiureghian, 1991] Liu, P.-L. and Der Kiureghian, A. (1991). Optimization algorithms for structural reliability. *Structural safety*, 9(3) :161–177.

[Madsen et al., 2006] Madsen, H. O., Krenk, S., and Lind, N. C. (2006). *Methods of Structural Safety*. Courier Corporation.

[Marin et al., 2012] Marin, J.-M., Pudlo, P., and Sedki, M. (2012). Consistency of the Adaptive Multiple Importance Sampling. *arXiv :1211.2548 [math, stat]*.

[Martino et al., 2017a] Martino, L., Elvira, V., and Louzada, F. (2017a). Effective sample size for importance sampling based on discrepancy measures. *Signal Processing*, 131 :386–401.

[Martino et al., 2015] Martino, L., Elvira, V., Luengo, D., and Corander, J. (2015). An adaptive population importance sampler : Learning from uncertainty. *IEEE Transactions on Signal Processing*, 63(16) :4422–4437.

[Martino et al., 2017b] Martino, L., Elvira, V., Luengo, D., and Corander, J. (2017b). Layered adaptive importance sampling. *Statistics and Computing*, 27(3) :599–623.

[Melchers, 1989] Melchers, R. (1989). Importance sampling in structural systems. *Structural safety*, 6(1) :3–10.

[Mestre, 2008a] Mestre, X. (2008a). Improved Estimation of Eigenvalues and Eigenvectors of Covariance Matrices Using Their Sample Estimates. *IEEE Transactions on Information Theory*, 54(11) :5113–5129.

[Mestre, 2008b] Mestre, X. (2008b). On the Asymptotic Behavior of the Sample Estimates of Eigenvalues and Eigenvectors of Covariance Matrices. *IEEE Transactions on Signal Processing*, 56(11) :5353–5368.

[Metropolis et al., 1953] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6) :1087–1092.

[Morio, 2012] Morio, J. (2012). Extreme quantile estimation with nonparametric adaptive importance sampling. *Simulation Modelling Practice and Theory*, 27 :76–89.

[Morio et al., 2014] Morio, J., Balesdent, M., Jacquemart, D., and Vergé, C. (2014). A survey of rare event simulation methods for static input–output models. *Simulation Modelling Practice and Theory*, 49 :287–304.

[Morio et al., 2021] Morio, J., Levasseur, B., and Bertrand, S. (2021). Drone ground impact footprints with importance sampling : Estimation and sensitivity analysis. *Applied Sciences*, 11(9) :3871.

[Nadakuditi and Edelman, 2008] Nadakuditi, R. and Edelman, A. (2008). Sample Eigenvalue Based Detection of High-Dimensional Signals in White Noise Using Relatively Few Samples. *IEEE Transactions on Signal Processing*, 56(7) :2625–2638.

[Nataf, 1962] Nataf, A. (1962). Determination des distribution dont les marges sont donnees. *Comptes Rendus de l Academie des Sciences*, 225 :42–43.

[Niederreiter, 1992] Niederreiter, H. (1992). *Random number generation and quasi-Monte Carlo methods*. SIAM.

[Papaioannou et al., 2019a] Papaioannou, I., Geyer, S., and Straub, D. (2019a). Improved cross entropy-based importance sampling with a flexible mixture model. *Reliability Engineering & System Safety*, 191 :106564.

[Papaioannou et al., 2019b] Papaioannou, I., Geyer, S., and Straub, D. (2019b). Software tools for reliability analysis : Cross entropy method and improved cross entropy method. Retrieved from `https://www.bgu.tum.de/era/software/software00/cross-entropy-method-and-improved-cross-entropy-method/`.

[Robert and Casella, 2004] Robert, C. P. and Casella, G. (2004). *Monte Carlo statistical methods*, volume 2. Springer.

[Roberts and Smith, 1994] Roberts, G. O. and Smith, A. F. (1994). Simple conditions for the convergence of the Gibbs sampler and Metropolis-Hastings algorithms. *Stochastic processes and their applications*, 49(2) :207–216.

[Rosenblatt, 1952] Rosenblatt, M. (1952). Remarks on a multivariate transformation. *The annals of mathematical statistics*, 23(3) :470–472.

[Rubinstein and Glynn, 2009] Rubinstein, R. Y. and Glynn, P. W. (2009). How to Deal with the Curse of Dimensionality of Likelihood Ratios in Monte Carlo Simulation. *Stochastic Models*, 25(4) :547–568.

[Rubinstein and Kroese, 2011] Rubinstein, R. Y. and Kroese, D. P. (2011). *The Cross-Entropy Method : A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning*. Springer, New York ; London.

[Rubinstein and Kroese, 2017] Rubinstein, R. Y. and Kroese, D. P. (2017). *Simulation and the Monte Carlo Method*. Wiley Series in Probability and Statistics. Wiley, Hoboken, New Jersey, third edition edition.

[Surjanovic and Bingham, 2021] Surjanovic, S. and Bingham, D. (2021). Virtual library of simulation experiments : Test functions and datasets. Retrieved October 8, 2021, from `http://www.sfu.ca/~ssurjano`.

[Uribe et al., 2021] Uribe, F., Papaioannou, I., Marzouk, Y. M., and Straub, D. (2021). Cross-entropy-based importance sampling with failure-informed dimension reduction for rare event simulation. *SIAM/ASA Journal on Uncertainty Quantification*, 9(2) :818–847.

[Wang and Song, 2016] Wang, Z. and Song, J. (2016). Cross-entropy-based adaptive importance sampling using von Mises-Fisher mixture for high dimensional reliability analysis. *Structural Safety*, 59 :42–52.

[Wang et al., 2013] Wang, Z., Zoghi, M., Hutter, F., and Matheson, D. (2013). Bayesian Optimization in High Dimensions via Random Embeddings. *Twenty-Third international joint conference on artificial intelligence*.

[Zahm et al., 2018] Zahm, O., Cui, T., Law, K., Spantini, A., and Marzouk, Y. (2018). Certified dimension reduction in nonlinear Bayesian inverse problems. *arXiv :1807.03712 [math, stat]*.

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

Résumé de la thèse

De nombreuses disciplines scientifiques s'intéressent à l'estimation d'espérances d'une fonction d'intérêt selon une certaine loi de probabilité. Cette fonction peut être considérée comme une boite noire, potentiellement couteuse à évaluer. Une méthode couramment utilisée pour estimer des espérances, tout en limitant le nombre d'appels à la boite noire, est la méthode stochastique d'échantillonnage préférentiel (Importance Sampling, IS) qui consiste à échantillonner selon une loi de probabilité auxiliaire au lieu de la loi initiale. L'estimateur d'IS est défini à partir de l'estimateur de Monte-Carlo, avec des poids d'importance, et converge presque sûrement vers l'espérance voulue, par la loi des grands nombres. Cependant, sa variance, et donc la précision de l'estimation, dépend fortement du choix de la densité auxiliaire. Une densité optimale d'échantillonnage préférentiel minimisant la variance peut être définie sur le plan théorique mais n'est pas connue en pratique. Une possibilité est alors de choisir la densité auxiliaire dans une famille paramétrique, avec laquelle il est facile de générer des échantillons, afin d'approcher la distribution optimale théorique. Des algorithmes adaptatifs (Adaptive Importance Sampling, AIS), qui estiment les paramètres de manière itérative, ont été développés pour trouver les paramètres optimaux permettant d'approcher la densité théorique visée. Mais lorsque la dimension de l'espace des paramètres augmente, l'estimation des paramètres se dégrade et les algorithmes d'AIS, et l'IS en général, deviennent inefficaces. L'estimation finale de l'espérance devient alors très imprécise, notamment du fait de l'accumulation des erreurs commises dans l'estimation de chaque paramètre.

L'objectif principal de cette thèse est ainsi d'améliorer la précision de l'IS en grande dimension, en réduisant le nombre de paramètres estimés à l'aide de projections dans un sous-espace de petite dimension. Nous nous concentrons particulièrement sur la recherche de directions de projection influentes pour l'estimation de la matrice de covariance dans un cadre gaussien unimodal (où l'on met à jour le vecteur moyenne et la covariance). La première piste explorée est la projection sur le sous-espace de dimension un engendré par la moyenne optimale. Cette direction est particulièrement pertinente dans le cas d'estimation d'une probabilité d'événement rare, car la variance semble diminuer selon cette direction. La seconde proposition correspond à la projection optimale obtenue en minimisant la divergence de Kullback-Leibler avec la densité visée. Cette seconde proposition permet de projeter dans un espace de plusieurs dimensions contrairement à la première, et permet d'identifier les directions les plus influentes. Dans un premier temps, l'efficacité de ces projections est testée sur différents exemples d'estimation d'espérances en grande dimension, dans un cadre théorique n'impliquant pas d'algorithmes adaptatifs. Les simulations numériques réalisées montrent une nette amélioration de la précision de l'estimation par IS avec les deux techniques de projection sur tous les exemples considérés. Ensuite, nous proposons un couplage de ces projections avec l'algorithme d'Entropie Croisée (Cross Entropy, CE), un algorithme d'AIS destiné à l'estimation de probabilités d'événements rares. L'efficacité de ces algorithmes est vérifiée sur plusieurs cas-tests avec un faible budget de simulation. La technique basée sur la projection dans les directions optimales permet d'obtenir des estimations très précises pour des dimensions modérément grandes (plusieurs dizaines). Le couplage avec la projection sur la moyenne reste en revanche performante dans des dimensions de quelques centaines dans la plupart des exemples. Dans tous les cas, les simulations montrent que les méthodes proposées sont plus précises que la CE classique en grande dimension avec un même budget.

Abstract

In many scientific fields, an important goal consists in estimating expectations of a function of interest according to a given probability distribution. The function can be considered as a computationally demanding black box function. Importance Sampling (IS) is a well-known method to estimate such integrals with a small simulation budget. It is a stochastic technique which consists in sampling from an auxiliary distribution instead of the initial one. The IS estimator is based on the Monte Carlo estimator with importance weights and converges almost surely to the unknown expectation, according to the law of large numbers. However, its variance, as well as the estimation accuracy, strongly depends on the choice of the auxiliary density. A theoretical optimal IS density, minimising the variance, can be defined but is unknown in practice. Hence, the auxiliary density can be chosen in a parametric family, which allows to easily generate samples, in order to approximate the optimal IS distribution. Adaptive Importance Sampling (AIS) algorithms have been developed to find optimal parameters allowing to approach the theoretical target density, by estimating parameters iteratively. However, when the dimension of the parameters space is growing, the parameters estimation is degrading and AIS algorithms, and IS more generally, become inefficient. Then the final expectation estimation becomes inaccurate, because of the accumulation of the parameters estimation errors.

Thus, the main goal of the thesis is the improvement of the accuracy of high dimensional IS, using projections in low dimensional subspaces to reduce the number of estimated parameters. We focus specifically on finding influential projection directions for the estimation of the covariance matrix in the Gaussian case (updating the mean vector and the covariance). The first suggested idea is the projection on the one-dimensional subspace spanned by the optimal mean vector. This direction is relevant in particular in the context of rare event probability estimation, because the variance decreases in this direction. The second projection is the optimal projection found by minimising the Kullback-Leibler divergence with the target density. This proposition allows to project in more than one direction contrary to the first technique, and identifies the most influential directions. We test the efficiency of both projections on various examples of expectation estimation, at first in a theoretical context and without adaptive algorithms. Numerical simulations show the significant improvement of the IS estimation accuracy with the two projection techniques and on all examples. We then implement an improvement of the Cross Entropy method (CE), an AIS algorithm for rare event probability estimation, using both projection methods. We check the efficiency of the proposed algorithms on some examples of rare event estimation with a small simulation budget. The projection on the optimal directions give accurate estimations in moderate dimensions (less than 100). The projection on the mean is still efficient in higher dimensions (a few hundreds) in most examples. In all cases, the numerical results show that our proposed algorithms outperform the classical CE by increasing the accuracy with the same budget.