Outils statistiques pour la construction et le choix de modèles en fiabilité des logiciels

Mhamed-Ali El-Aroui

THÈSE

présentée par

Mhamed-Ali EL AROUI

Pour obtenir le titre de Docteur de

L'UNIVERSITÉ JOSEPH FOURIER – GRENOBLE 1

Spécialité

Mathématiques Appliquées

OUTILS STATISTIQUES POUR LA CONSTRUCTION ET LE CHOIX DE MODÈLES EN FIABILITÉ DES LOGICIELS

Thèse soutenue le 20 septembre 1996 devant la Commission d'Examen :

Bernard	YCART	Président du jury
Karama	KANOUN	Rapporteur
Jean-Pierre	RAOULT	Rapporteur
Jean	DIEBOLT	Examinateur
Olivier	GAUDOIN	Examinateur
Christian	LAVERGNE	Examinateur
Jean-Louis	SOLER	Directeur de thèse

Thèse préparée au sein du Laboratoire LMC de l'Institut IMAG de Grenoble

Mayara : Nous créerons les vents en ouragan. Nous créerons les tonnerres fracassants. Nous construirons le barrage. Nous construirons l'amour.

Ghaylane : OuiΓnous construironsΓnous créeronsΓnous enseignerons à cette terre le courage et la raisonΓl'énergie et la fermeté. Si fort nous secouerons ses habitants qu'ils abjureront leur prostrationΓleur lâchetéΓleur haine de l'eau et leur amour de l'aridité. Nous injecterons en eux notre parole et notre âme. Nous insufflerons la vie en toutes choses. Nous soufflerons un ouragan d'effroi. Nous construironsΓnous céerons d'authentique création. Car la force et l'élanΓc'est en nous qu'ils sont.

Ghaylane et Mayara s'en vont...

Maymouna : *les suivant du regard, ironise amèrement.* Ils vont créer les vents et les ouragans. Ils vont créer les tonnerres fracassants... Ils vont construire le barrage... Ils vont construire l'amour... Ah ! que soit maudite la gent des songe-creux !

Mahmoud Messaâdi

"Le barrage"

Ce travail a été réalisé au sein du *Laboratoire de Modélisation et Calcul* de l'institut *IMAG* de Grenoble. Je remercie les membres de l'équipe *Statistique et Modélisation Stochastique* pour leur accueil et leur soutien tout au long de ces années de thèse.

Je tiens à exprimer ma sincère reconnaissance à Monsieur Jean-Louis Soler, professeur à l'*Institut National Polytechnique de Grenoble*, qui a dirigé mes travaux et qui m'a sans cesse conseillé, aidé et encouragé.

Mes remerciements vont à Monsieur Bernard Ycart, professeur à l'*Université Joseph Fourier de Grenoble*, qui me fait l'honneur de présider le jury de soutenance.

Madame Karama Kanoun, chargée de recherche au *CNRS*, et Monsieur Jean-Pierre Raoult, professeur à l'*Université Paris V* ont bien voulu être rapporteurs de ce travail. Leurs remarques, leurs suggestions et leurs critiques ont amélioré la qualité de ce mémoire. Je les remercie très vivement pour le temps qu'ils y ont consacré.

Je remercie Monsieur Jean Diebolt, directeur de recherche au *CNRS*, pour avoir accepté de participer au jury. Ses conseils et son soutien m'ont permis d'achever ce travail.

Mes vifs remerciements s'adressent enfin à Monsieur Olivier Gaudoin, maître de conférences à l'*IUFM de Grenoble*, et à Monsieur Christian Lavergne, professeur à l'*Université Montpellier III*. Leur disponibilité, leurs encouragements et les nombreuses discussions que j'ai eu avec eux ont largement contribué à l'élaboration de ce travail.

Introduction

Cette thèse est consacrée à l'étude de méthodes statistiques pour l'évaluation de la fiabilité des logiciels.
L'utilisation croissante des systèmes informatisés dans tous les domaines donne une importance cruciale au problème de la fiabilité des logiciels.
En effetΓdes défaillances de logicielsΓpar exempleΓdans des systèmes de manœuvres aérospatialesΓde contrôle de réacteurs nucléairesΓd'assistance chirurgicale ou de transactions financières peuvent avoir des conséquences catastrophiques dans certaines utilisations critiquesΓet au moins des conséquences économiques dans la plupart des cas.
La mise en service de tout logiciel doit donc être précédée d'une période de tests et de validation permettant de garantir un niveau acceptable de fiabilité.

Plusieurs phénomènes agissent sur le logiciel au cours de son cycle de vie : fautes de conception d'origine humaineΓeffets variables des correctionsΓdifférents environnements d'utilisationΓinteraction avec d'autres logicielsΓetc. La complexité de ces phénomènes rend inévitable la présence de fautes dans tout logiciel de taille importanteΓet ceΓmême après la période de tests. Cette complexité des facteurs mis en jeu rend par ailleurs impossible toute tentative d'évaluation exacte du degré de fiabilité d'un logiciel.
Cela implique le recours à une modélisation stochastique de l'interaction du logiciel avec le monde externe. Cette modélisation permet d'utiliser des méthodes statistiques efficaces pour analyser les données de défaillance du logiciel et prédire sa fiabilité future.

Un logiciel constitue un exemple typique de "Système améliorable" (cf. [95]) dont les défaillances sont imputables à des fautes de conception. Sa fiabilité est donc susceptible d'évoluer au cours du tempsΓpar suite de leurs corrections. C'est ce qui le distingue d'un système réparable au sens traditionnelΓdont les performances sont toujours au plus égales à celles d'un système neuf. D'où la notion de "Croissance de fiabilité".

Par ailleurs une des caractéristiques essentielles du logiciel est l'absence de phénomène

d'usure ou de vieillissementΓce qui justifie l'utilisation généralisée de la loi de probabilité Exponentielle pour les durées de bon fonctionnementΓdans un grand nombre de modèles de fiabilité des logiciels.

Les premiers modèles mathématiques entièrement consacrés à l'étude de la fiabilité des logiciels ont été présentés au début des années soixante-dix notamment par Jelinski et Moranda en 1972 [47] et Littlewood et Verrall en 1973 [67].

La prise de conscience de l'importance du problème de la sûreté de fonctionnement des systèmes informatisés a ensuite suscité un très grand nombre d'études concernant la modélisation et l'évaluation statistique de la fiabilité des logiciels.

Des revues de synthèse de cette littérature ont été présentées par Xie [102Γ103]ΓSingpurwalla et Wilson [90] et Lyu et al [69].

Il est devenu clair aujourd'hui que la complexité et la diversité des phénomènes définissant les comportements des logiciels rendent impossible l'obtention d'un modèle universel utilisable dans toutes les études de fiabilité des logiciels.

La diversité des outils de conceptionΓdes procédés de tests et de corrections ainsi que la diversité des domaines et des profils d'utilisation font que chaque logiciel a ses propres particularités dont il faut tenir compte lors de l'évaluation de sa fiabilité. On pourra se référerΓà ce sujetΓpar exemple aux travaux de Laprie [57] et Kanoun [50].

Loin de vouloir ajouter de nouveaux modèles à la multitude de modèles déjà existantsΓ nous nous sommes proposés dans ce travail d'élaborer des méthodes statistiques permettant aux praticiens de construire et ensuite de valider leurs propres modèles en tenant compte des spécificités de leurs logiciels.

Ces outils de construction de modèlesΓtiennent compte des hypothèses générales en Fiabilité des LogicielsΓtout en permettant à chaque utilisateur d'intégrer d'une manière simple les particularités de son étude.

Pour démontrer la supériorité éventuelle des modèles ainsi construitsΓnous avons ensuite été amenés à étudier les outils statistiques pour le choix et la comparaison de modèles de fiabilité des logiciels.

Les différentes parties de ce travail ont nécessité l'utilisation d'un certain nombre d'outils mathématiques généraux Γnotamment les Modèles Linéaires Généralisés paramétriques et non paramétriquesΓl'Analyse statistique bayésienne et les Tests d'adéquation statistiques. Nous avons jugé utile de présenter brièvement chacun de ces outils avant son utilisation pour apporter une plus grande clarté à l'exposé.

Nous présentons dans le premier chapitre le cadre général et les concepts de base en Fiabilité des Logiciels : profil opérationnelΓensemble de fautesΓsollicitationsΓdéfaillancesΓ correctionsΓ versionsΓetc. Ils s'inscrivent dans le cadre d'une modélisation probabiliste générale de la vie d'un logicielΓbasée sur l'utilisation des processus aléatoires.
Cette modélisation permet par ailleurs de définir rigoureusement les principaux attributs de la fiabilité d'un logicielΓd'intégrer le caractère évolutif de celle ciΓainsi que le *MTTF*Γ le taux de défaillanceΓetc.

Nous terminons le premier chapitre par une revue des principaux modèles d'évaluation statistique de la fiabilité des logiciels.

Dans le deuxième chapitre nous utilisons la théorie des Modèles Linéaires Généralisés pour présenter des outils de construction et de choix de modèles en Fiabilité des Logiciels.
Ces outils ont l'avantage de pouvoir tenir compte des spécificités de chaque logiciel pour construire des modèlesΓaussi bien paramétriques que non paramétriquesΓayant de meilleures performances que les modèles usuels.

Le troisième chapitre est consacré aux méthodes statistiques bayésiennes en Fiabilité des Logiciels.
Après une revue critique des principales approches bayésiennes usuellement proposées dans ce domaineΓnous présentons un outil bayésien général pour la modélisation et l'évaluation de la fiabilité des logiciels.

Contrairement à la plupart des approches traditionnellesΠl'outil bayésien que nous présentons a l'avantage de pouvoir s'adapter aux différentes connaissances a priori des praticiens quant au comportement et aux spécificités de leurs logiciels. En effetΓcet outil est basé sur l'utilisation des algorithmes récents de simulation stochastiqueΓce qui permet d'éviter le choix de lois de probabilité a priori dont la seule justification est le plus souvent la facilité des calculs analytiques.

Dans le dernier chapitre nous étudions le problème de la validation et du choix de modèles en Fiabilité des Logiciels.
On ne trouve dans la littérature qu'un faible nombre d'outils empiriques permettant de faire le choix du modèle le plus adéquat parmi d'autres.

Nous discutons dans ce chapitre de l'utilisation des tests d'adéquation statistiques pour la validation de modèles de fiabilité des logiciels. Nous donnons ensuite un cadre théorique permettant de définir rigoureusement le critère du *u-plot* et d'en étudier les propriétés.

Ce critèreΓl'un des plus utilisés pour la validation des modèles de fiabilité des logicielsΓ n'a jusqu'ici pas été étudié de façon rigoureuse.
Nous montrons que ce critèreΓprésenté initialement comme un indicateur empiriqueΓpeut dans certains cas être considéré comme un test d'adéquation au sens statistique.

Cette démarche nous a par ailleurs amené à présenter un nouveau test séquentiel d'adéquation à une loi exponentielle de paramètre inconnu Γ et qui peut être utilisé dans un autre contexte.

Ce travail s'inscrit donc dans le cadre de la statistique appliquée Γ il souhaite contribuer à enrichir ou améliorer les méthodes généralement utilisées par les praticiens. Il porte à la fois sur les aspects théoriques Γ méthodologiques et pratiques en Fiabilité des Logiciels.

Table des matières

Chapitre 1

Méthodologie de l'évaluation de la fiabilité des logiciels

On commence ce chapitre par une présentation de la problématique et du cadre général de notre étude. Après avoir introduit la terminologie et les différents concepts utilisésΓon décrit les principaux facteurs agissant sur le comportement du logiciel au cours de son cycle de vie.

On présente ensuite une modélisation mathématique de la vie d'un logiciel. Cette modélisationΓbasée sur les processus aléatoiresΓpermet de définir rigoureusement les différentes mesures et attributs de la fiabilité.

A la fin de ce chapitre on présente une brève revue des principales classes de modèles statistiques de fiabilité des logiciels.

1.1 Problématique

1.1.1 Cadre général

Comme pour tout systèmeΓla fiabilité d'un logiciel mesure son aptitude à délivrer un service correct pendant une durée déterminée. Plus précisément on appelle **fiabilité** d'un logiciel la fonction du temps exprimée par la probabilité que le logiciel fonctionne sans défaillances pendant une période fixée et dans un environnement donné.

La fiabilité fait partie d'un concept plus global : la sûreté de fonctionnement qui regroupeΓ outre la fiabilitéΓles concepts de disponibilitéΓde maintenabilité et de sécurité (cf. Laprie [57]).

Le comportement d'un logicielΓet en particulier sa fiabilitéΓévolue au cours du temps en fonction de trois facteurs principaux :

- L'ensemble de ses fautes.
- Son profil d'utilisation : il décrit le comportement des utilisateurs du logiciel : choix des entréesΓfréquence des sollicitationsΓetc.

- Les modifications et les corrections que subit le logiciel au cours de son cycle de vie.

La complexité des phénomènes mis en jeu ainsi que l'incertitude concernant leurs effets et interactions font que toute évaluation de la fiabilité nécessite une modélisation aléatoire de la vie du logiciel considéré.

Le fonctionnement du logiciel étant observé pendant une période de temps donnéeΓl'objectif de cette étude est d'utiliser les méthodes statistiques pour l'estimation de la fiabilité et la prédiction du comportement futur du logiciel.

1.1.2 Concepts de base en Fiabilité des Logiciels

Dans tout ce travail on considère qu'un logiciel est un systèmeΓqui par l'intermédiaire d'un programme transforme des données d'entrée (instructionsΓchiffresΓimagesΓfichiersΓetc.) en données de sortie ou résultats.
Les **spécifications** du logiciel définissent quels doivent être les résultats fournis pour les différentes données d'entrée.

L'étude de la fiabilité des logiciels peut être faite dans le cadre général de l'étude de la fiabilité des systèmes améliorables.

Un **système améliorable** (cf. Soler [95]) est un système qui a des défaillances parce qu'il présente des fautes de conception. En supposant que l'on puisse corriger ces fautesΓ les performances du système se trouvent donc améliorées au cours du tempsΓcontrairement aux systèmes réparables dont les performances sont toujours au plus égales à celles d'un système neuf.

On considère les logiciels comme des cas particuliers de systèmes améliorables et on ne s'intéressera qu'aux problèmes de fonctionnement dus aux fautes de conception. Ceci implique qu'un logiciel ne vieillit pas et que sa fiabilité évolue au cours du temps au fur et à mesure des tentatives de suppression de ses fautes de conception. Dans le cas où les fautes de conception sont effectivement supprimées on observe alors une **croissance de fiabilité**.

Une **faute** du logiciel désigneraΓdans ce travailΓune de ses fautes de conception dues généralement à des imperfections de programmation perpétrées soit au cours du développement soit au cours des modifications ultérieures (cf. [57]).

Une **défaillance** survient quand pour une donnée d'entrée particulière on observe une différence entre le résultat fourni par le logiciel et le résultat prévu par les spécifications. Une défaillance peut être la manifestation d'une ou plusieurs fautes.

Une faute est un phénomène intrinsèque au programmeΓalors qu'une défaillance est un phénomène dynamique dépendant de la façon dont le logiciel est utilisé.

1.1.3 Evolution de la fiabilité d'un logiciel

L'évolution de la fiabilité d'un logiciel résulte de l'effet au cours du temps des différents facteurs agissant sur son comportement.
Ces facteurs évoluent au cours des différentes étapes du cycle de vie du logiciel. Rappelons que ces principales étapes sont : l'expression des besoinsΓla conceptionΓle développementΓ la validation pré-opérationnelle (phase de tests) et la vie opérationnelle (comprenant l'exploitation et la maintenance).

Ensemble des fautes d'un logiciel

En théorie on peut concevoir des logiciels parfaitsΓmais ceux-ci sont inaccessibles en pratique. A la sortie des étapes de conception et de développementΓtout nouveau produit logiciel de taille importante contiendra forcément des fautes de conception.

En phase de validation pré-opérationnelle on essaiera de supprimer le plus grand nombre de fautes. Mais malgré les tests rigoureux et systématiques et malgré le respect des normes et standards du génie logicielΓla plupart des logiciels de taille importante contiennent encore des fautes quand ils sont livrés.

L'ensemble des fautes continuera à évoluer au cours de la vie opérationnelle ; cette évolution est due aux activités de maintenance et à la livraison de nouvelles versions du logiciel.

Corrections et modifications du logiciel

Le logiciel subitΓtout le long de son cycle de vieΓun certain nombre de modifications qui font évoluer son ensemble de fautes et par conséquent sa fiabilité.

Ces modifications peuvent être dues à des changements de spécifications : changements ou ajouts de fonctionnalitésΓetc. Les modifications les plus fréquentes sont cependant celles dues aux corrections c'est-à-dire aux tentatives de suppression des fautes.

Plusieurs politiques de correction peuvent être envisageables. En phase de développement et de testsΓles corrections sont généralement introduites au fur et à mesure de l'observation des défaillancesΓceci évitera d'observer plusieurs défaillances associées à la même faute.
Une deuxième façon de faireΓfréquente en phase opérationnelleΓconsiste à n'effectuer les corrections qu'après l'activation et l'identification d'un certain nombre de fautes. Kanoun [50] parle alors de **correction par lots**.

Au cours de la vie opérationnelleΓlorsque l'utilisateur observe une défaillanceΓil reprend généralement le traitement en évitant l'utilisation de l'entrée défaillante. Il signale ensuite cette défaillance au constructeur. Quand ce dernier aura un nombre suffisant de réclamationsΓil lancera une nouvelle version où il effectuera toutes les corrections nécessaires.

L'activité de **maintenance** regroupe toutes les modifications qui ont lieu au cours de la vie opérationnelle.

Profil d'utilisation

La notion de fiabilité d'un logiciel est étroitement liée à la notion de profil d'utilisationΓ c'est-à-dire la manière dont il sera utilisé.
En vie opérationnelleΓle profil d'utilisation est appelé profil opérationnelΓil diffère d'un utilisateur à un autre.

Le profil opérationnel d'un utilisateur est spécifié par la fréquence de ses utilisations et les probabilités des sollicitations des différentes données d'entrée.

Deux utilisateurs ayant des profils opérationnels différents peuvent avoir deux perceptions différentes de la fiabilité du même logiciel.

La combinaison des profils opérationnels des différents groupes d'utilisateurs permet de définir le profil opérationnel moyen du logiciel (cf. Musa [76]).

Le profil d'utilisation évolue au cours du cycle de vie du logiciel. Au début de la phase de testsΓ le logiciel est généralement sollicité beaucoup plus qu'il ne le sera en phase opérationnelle. En fin de période de tests on essaiera au contraire de se rapprocher des conditions d'utilisation opérationnelle.

1.2 Modélisation aléatoire de la vie d'un logiciel

Nous présentons dans cette section l'approche adoptée pour la modélisation de la vie d'un logiciel. nous donnons ensuite les définitions et les outils mathématiques permettant de modéliser les différents facteurs décrits ci-dessus.

1.2.1 Approches "boite noire" et "boite blanche"

L'approche qu'on adoptera dans ce travail est l'approche appelée "boite noire" où on considère le logiciel comme une seule entité ou "boite noire".

L'effort de modélisation se concentre alors sur les interactions entre cette "boite noire" et le monde extérieur : sollicitationsΓdéfaillancesΓcorrections...

On tiendra compte cependant des différentes informations disponibles : profils d'utilisation, effets des corrections, environnements de tests... pour choisir les lois de probabilité adéquates des différents processus aléatoires mis en jeu.

Un deuxième type de modélisation, appelé approche "boite blanche", consiste à utiliser l'information disponible concernant la structure du logiciel étudié.
On trouve ainsi un certain nombre de modèles, appelés modèles structurels (cf. par exemple [17], [61] et [65]) où l'on tient compte de la structure du logiciel à travers sa décomposition en un certain nombre de composants principaux ou modules. L'interaction entre ces composants correspond à des transferts de contrôle de l'exécution.
L'exécution des programmes du logiciel est alors modélisée par un processus stochastique markovien désignant à chaque instant l'unique module actif.

Comme le souligne Ledoux [61], pour pouvoir appliquer les modèles structurels il faut adopter des hypothèses trop réductrices. Ces modèles nécessitent en outre un effort de collecte de données assez important puisqu'il faut recueillir des données de défaillance spécifiques aux différents modules ainsi qu'aux transferts de contrôle entre ces modules.

La difficulté de leur traitement numérique et l'absence de données expérimentales adéquates font que les modèles structurels sont pour l'instant très peu utilisés par les praticiens.

1.2.2 Définitions et outils mathématiques

Dans la modélisation utilisée dans ce travail, on considère que les instants de sollicitation du logiciel, les données d'entrée sollicitées ainsi que les instants de défaillance, sont des variables aléatoires. Les principaux outils de modélisation seront donc les processus aléatoires.

Les définitions et la terminologie employées ci-dessous sont inspirées des travaux de Gaudoin [36] et Soler [95].

Espace des données d'entrée

On suppose que l'on peut définir l'ensemble E de toutes les données d'entrée admissibles par le logiciel. On supposera que cet ensemble est invariant pendant la vie du logiciel.
Comme on considère que les données d'entrée sont choisies d'une façon aléatoire, il est nécessaire de munir E d'une tribu $\mathcal{A}$ des parties de cet ensemble représentant les événements d'entrée du logiciel.

L'espace mesurable $(E, \mathcal{A})$ sera l'**espace des entrées** du logiciel.

Processus de sollicitation et Profil opérationnel

Le logiciel est sollicité de façon aléatoire à la fois dans le temps et dans l'espace des entrées. Les instants successifs de sollicitation temporelle $0 < S_1 < S_2 \ldots$ forment un processus ponctuel sur $\mathbb{R}_+$.
A chacun des instants S_iΓle logiciel est sollicité avec une entrée aléatoire Z_i choisie selon une loi de probabilité Q_i dans E.
La suite des sollicitations est donc une suite de couples (S_i, Z_i) formant un processus ponctuel $\mathcal{S}$ sur l'espace produit $\mathbb{R}_+ \times E$.

Définition – 1.1 *on appelle* ***processus de sollicitation*** *du logiciel, le processus ponctuel sur* $\mathbb{R}_+$*, marqué dans* E *:*

$$\{\mathcal{S}_{[0,t]\times A}\}_{t\geq 0, A\in\mathcal{A}}$$

où pour tout $t \geq 0$ *et pour tout* $A \in \mathcal{A}$*,* $\mathcal{S}_{[0,t]\times A}$ *est le nombre de sollicitations* S_i *qui ont lieu sur l'intervalle* $[0, t]$ *dont les entrées associées appartiennent à* A*.*

Le ***profil opérationnel*** *est la loi de probabilité du processus de sollicitation.*

Un exemple particulier de profil opérationnel représentant assez bien les conditions d'utilisation générales des logiciels est le profil opérationnel Poissonnien homogène qui sera présenté dans la sous-section 1.4.3.

Processus des fautes

La notion de faute est liée à celle de donnée d'entrée. On confondra dans cette modélisation la faute avec l'ensemble des données d'entrée qui conduiront à sa manifestation. Une **faute de conception** est une partie $\mathcal{A}$-mesurable non vide de l'espace des entrées E.

Définition – 1.2 *A l'instant* t*, on appelle* ***faute totale*** $F_t \subset E$ *l'ensemble de toutes les données d'entrée pouvant provoquer des défaillances à l'instant* t*.*
On appelle ***processus des fautes****, la famille* $\{F_t\}_{t\geq 0}$*.*

Le processus des fautes est un processus aléatoire ensembliste à sauts. Ceux-ci interviennent aux instants des corrections.

Processus de défaillance

Une défaillance se produit à la $i^{ème}$ sollicitation si l'entrée correspondante Z_i active une fauteΓc'est-à-dire si Z_i appartient à la faute totale à l'instant S_i.

On appelle **instant d'une défaillance** l'instant de la sollicitation qui a entraîné cette défaillance.

Définition – 1.3 *Le* ***processus de défaillance*** *du logiciel est un processus ponctuel de* $\mathbb{R}_+$ *défini indifféremment par :*

- $T = \{T_i\}_{i\geq 1}$ *où* T_i *est l'instant de la* $i^{ème}$ *défaillance.*
- $X = \{X_i\}_{i\geq 1}$ *où* $X_i = T_i - T_{i-1}$ *(avec* $T_0 = 0$*) est la durée séparant la* $(i-1)^{ème}$ *de la* $i^{ème}$ *défaillance.*
- $N = \{N_t\}_{t\geq 0}$ *où* N_t *est le nombre cumulé de défaillances entre l'instant initial et l'instant* t*.*

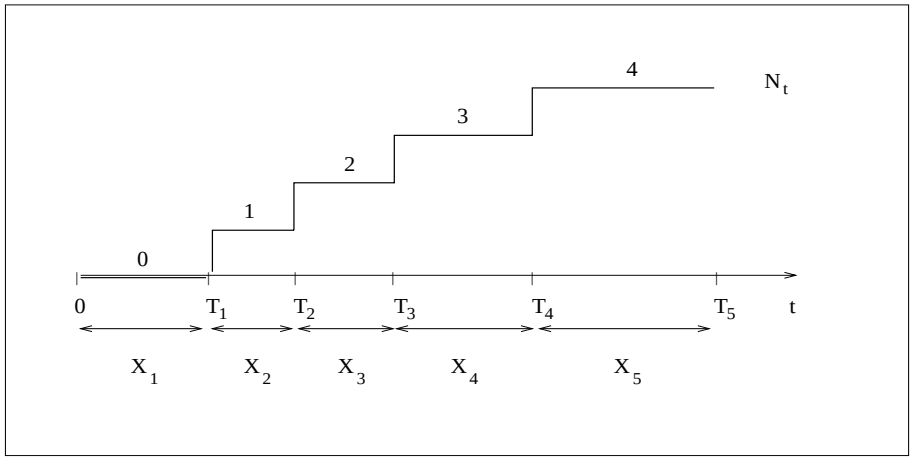

FIG. 1.1: Exemple d'une trajectoire du processus de défaillance.

Processus de correction

Une correction est une tentative de suppression de fautes. Elle a donc pour effet de modifier l'ensemble des fautes totales du logiciel. Plus précisément on a :

Définition – 1.4 *Une* ***correction*** *est une application de* $\mathcal{A}$ *dans* $\mathcal{A}$ *qui, à une faute totale avant correction* F*, associe une faute totale après correction* F'*.*

Une bonne correction aura pour effet de diminuer la taille de l'ensemble des fautes.

Définition – 1.5 *Le* ***processus de correction*** *du logiciel est un processus ponctuel sur* $\mathbb{R}_+$*, marqué dans* $\mathcal{A}$ *défini indifféremment par :*

- $C = \{C_i\}_{i\geq 1}$ *où* C_i *est l'instant de la* $i^{ème}$ *correction.*
- $Y = \{Y_i\}_{i\geq 1}$ *où* $Y_i = C_i - C_{i-1}$ *est la durée séparant la* $(i-1)^{ème}$ *de la* $i^{ème}$ *correction.*
- $K = \{K_t\}_{t\geq 0}$ *où* K_t *est le nombre cumulé de corrections effectuées entre l'instant initial et l'instant* t*.*

A chaque instant de correction C_i, on associe sa marque $F_{C_i} \subset E$ représentant la faute totale après la $i^{ème}$ correction.

Remarque – Le processus des fautes $\{F_t\}_{t\geq 0}$ est un processus de Markov. Ceci découle du fait que F_{C_i} est une transformationΓviala $i^{ème}$ correctionΓde $F_{C_{i-1}}$. La faute totale F_{C_i} ne dépend donc du passé qu'à travers $F_{C_{i-1}}$.

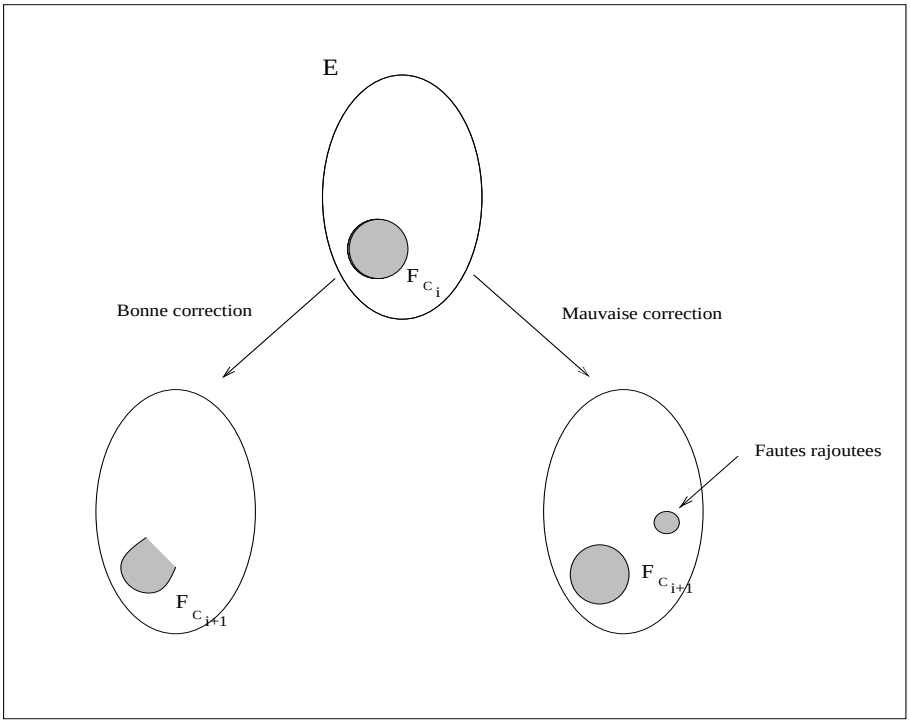

FIG. 1.2: Exemples de bonne et de mauvaise corrections.

Processus du comportement et histoire du logiciel

Définition – 1.6 *On appelle **processus du comportement** du logiciel, le processus qui décrit le comportement du logiciel au cours du temps. Ce processus résulte de l'interaction des processus de sollicitation, de défaillance et de correction. Il est noté :*

$$\Big\{\mathcal{S}_{[0,t]\times A}, N_t, K_t, F_t)\Big\}_{t\geq 0, A\in\mathcal{A}}.$$

Définition – 1.7 *On appellera **histoire** du logiciel à l'instant t, la filtration $\{\mathcal{H}_t\}_{t\geq 0}$ où $\mathcal{H}_t$ est la tribu engendrée par le passé du logiciel à l'instant t :*

$$\mathcal{H}_t = \sigma\left\{(\mathcal{S}_{[0,s]\times A}, N_s, K_s, F_s), 0 \leq s \leq t, A \in \mathcal{A}\right\}.$$

1.2.3 Interaction des différents processus

Pour modéliser rigoureusement l'évolution du logiciel au cours du tempsΓil faut tenir compte de l'interaction entre les différents processus mis en jeuΓen effet :

- Une sollicitation (S_i, Z_i) provoque une défaillance à l'instant S_i si l'entrée Z_i appartient à la faute totale F_{S_i}. Le processus de défaillance est donc influencé par le processus de sollicitation et le processus de faute.
- On procède à des corrections quand on observe un certain nombre de défaillancesΓ le processus de défaillance excite ainsi le processus de correction.
- Une correction a pour effet de modifier la faute totale du logiciel. Le processus de faute se trouve ainsi influencé par le processus de correction.

1.3 Approche statistique de l'évaluation de la fiabilité des logiciels

L'évaluation de la fiabilité d'un logiciel se fait par l'analyse du comportement passé et la prédiction du comportement futur de son processus de défaillance. Ceci nécessite l'utilisation d'un modèle probabiliste où l'on utilisera les diverses informations disponibles pour choisir les lois de probabilité des différents processus aléatoires mis en jeu.

Les données issues de l'observation du logiciel permettent ensuite d'estimer les paramètres du modèle pour analyser le comportement passé et prédire le comportement futur du logiciel.

La nature des données recueillies conditionne la façon de modéliser le processus de défaillance. On présente dans cette section deux types de modélisation selon que les observations sont des instants de défaillance ou des instants de correction.
On présente ensuite les différents attributs servant à mesurer la fiabilité d'un logiciel.
A la fin de la sectionΓon décrit la méthodologie générale de l'évaluation statistique de la fiabilité des logiciels.

1.3.1 Modélisation du processus de défaillance

Dans le cadre de l'étude de la fiabilitéΓon s'intéresse au comportement du processus de défaillance. Ce processusΓcomme tout processus aléatoire ponctuel (cf. [18] page 11)Γest complètement caractérisé par son intensité de défaillance conditionnelle :

Définition – 1.8 *On appelle* ***intensité de défaillance conditionnelle*** *la fonction aléatoire, définie à tout instant t par :*

$$\lambda_t = \lim_{dt \to 0} \frac{1}{dt} P(N_{t+dt} - N_t > 0 \mid \mathcal{H}_t^d)$$

où $\mathcal{H}_t^d$ représente l'histoire du processus sur l'intervalle de temps $[0,t]$Γc'est-à-dire la tribu engendrée par tous les événements pouvant influencer le processus de défaillance.

Dans le cas de la modélisation la plus générale l'histoire du processus de défaillance se confond avec l'histoire du logiciel :

$$\mathcal{H}_t^d = \mathcal{H}_t = \sigma\left\{(\mathcal{S}_{[0,s]\times A}, N_s, K_s, F_s), 0 \leq s \leq t, A \in \mathcal{A}\right\}.$$

Modéliser le comportement du processus de défaillance revient donc à modéliser la fonction intensité de défaillance conditionnelle et plus précisément l'influence des différents processus sur cette fonction.

Le modèle ainsi obtenu permettra ensuite de tenir compte des données issues de l'observation du passé du logiciel pour prédire le comportement futur de son processus de défaillance.

Hypothèse – 1 *On supposera dans la suite que la probabilité d'occurrence simultanée de deux défaillances est négligeable, c'est-à-dire :*

$$\forall t > 0 \ , \ P(N_{t+dt} - N_t \geq 2) = o(dt),$$

le processus de défaillance est alors dit ***ordonné****.*

Modèles à données de défaillance

Les données recueillies se résument en général aux instants d'occurrence des défaillances. Un modèle mathématiquement exploitable ne peut donc tenir compte explicitement de l'influence des processus de sollicitation et de faute.

La plupart des approches classiques en Fiabilité des Logiciels ne modélisent pas l'influence des processus de sollicitation et de faute. Le processus de défaillance est alors modélisé par un processus **auto-excité**Γson histoire $\mathcal{H}_t^d$ se réduit à la tribu engendrée par ses propres événements :

$$\mathcal{H}_t^d = \sigma\left(N_t, T_1, \ldots, T_{N_t}\right).$$

Proposition – 1.9 (Snyder [93] page 240) *La loi de probabilité d'un processus auto-excité ordonné $\{N_t\}_{t\geq 0}$ est complètement spécifiée par son intensité conditionnelle, donnée à chaque instant t par :*

$$\lambda_t = \lim_{dt\to 0} \frac{1}{dt} P(N_{t+dt} - N_t = 1 \mid N_t, T_1, \ldots, T_{N_t}).$$

Modéliser le processus de défaillance revient alors à proposer une expression analytique de la fonction λ_t.

L'hypothèse que le processus de défaillance est un processus auto-excité peut se justifier dans le cas où les corrections suivent immédiatement les défaillances. Les instants d'occurrence des corrections sont alors les mêmes que ceux des défaillances.

Modèles à données de correction

Pour certains logicielsΓles observations retenues sont les dates de correction et le nombre de défaillances entre deux corrections successives.
Dans ce casΓmodéliser le processus de défaillance par un processus auto-excité ne permet pas d'utiliser toutes les informations disponibles.

Soler[96] propose alors de considérer que les processus de défaillance et de correction sont deux processus aléatoires ponctuels mutuellement excitésΓspécifiés respectivement par leurs intensités relatives :

Définition – 1.10 *On appelle* ***intensité de défaillance relative*** *la fonction aléatoire définie en tout instant t par :*

$$\lambda_t^r = \lim_{dt \to 0} \frac{1}{dt} P(N_{t+dt} - N_t = 1 \mid N_t, T_1, \ldots, T_{N_t} ; K_t, C_1, \ldots, C_{K_t}).$$

Définition – 1.11 *On appelle* ***intensité de correction relative*** *la fonction aléatoire définie en tout instant t par :*

$$\mu_t^r = \lim_{dt \to 0} \frac{1}{dt} P(K_{t+dt} - K_t = 1 \mid N_t, T_1, \ldots, T_{N_t} ; K_t, C_1, \ldots, C_{K_t}).$$

On appelle **révision** d'un logiciel sa configuration entre deux corrections successives.

Pour différents protocoles de correction et différentes façons de modéliser l'interaction entre le processus de défaillance et le processus de correctionΓon peut obtenir différents estimateurs des attributs de la fiabilité (cf. Soler [95Γ96]).

Hypothèse – 2 Dans toute la suite de ce chapitre, ainsi que dans le chapitre suivant, on supposera que le processus de défaillance est un processus aléatoire auto-excité et que les corrections suivent instantanément les défaillances.

1.3.2 Attributs de la fiabilité des logiciels

Pour les systèmes améliorablesΓet en particulier les logicielsΓla fiabilité évolue au cours de la vie du système ; elle est évaluée à partir de l'étude du processus de défaillance.
Les attributs de la fiabilité sont donc les différentes fonctions décrivant le comportement du processus de défaillance au cours du temps. Ces fonctions sont décrites ci-dessous.

La fonction de fiabilité

A l'instant tΓla fonction de fiabilité représente la probabilité de ne pas avoir de défaillance au cours d'un intervalle de temps de durée déterminée débutant à l'instant t.
Cette fonction doit être redéfinie à chaque instant puisqu'elle est susceptible d'évoluer au cours de la vie du logiciel.

Définition – 1.12 *On appelle* ***fonction de fiabilité*** *à l'instant t, la fonction définie par :*

$$\forall \tau > 0,\ R_t(\tau) = P(N_{t+\tau} - N_t = 0).$$

A l'instant $t = T_n$Γinstant d'occurrence de la $n^{ème}$ défaillanceΓcette fonction de fiabilité vaut :

$$\forall \tau > 0,\ R_{T_n}(\tau) = P(X_{n+1} > \tau).$$

Le MTTF

A l'instant t le MTTF (Mean Time To Failure) représente l'espérance du temps d'attente de la prochaine défaillance. On a plus précisément :

Définition – 1.13 *A l'instant t, on appelle* ***MTTF*** *la quantité :*

$$MTTF_t = E(T_{N_t+1} - t).$$

En particulierΓsi on se place à l'instant T_n de la $n^{ème}$ défaillanceΓon a alors :

$$MTTF_{T_n} = E(X_{n+1}).$$

Remarque – Le $MTTF$Γquand il est finiΓpeut se calculer par la formule suivante :

$$MTTF_t = \int_0^\infty R_t(\tau) d\tau.$$

L'intensité de défaillance

A l'instant tΓl'intensité de défaillance représente la probabilité instantanée d'occurrence d'une défaillance sur l'intervalle $[t, t + dt]$. Plus précisément on a :

Définition – 1.14 *On appelle* ***intensité de défaillance*** *la fonction du temps :*

$$h(t) = \lim_{dt \to 0} \frac{1}{dt} P(N_{t+dt} - N_t = 1).$$

Il est important de noter que contrairement à l'intensité de défaillance conditionnelle notée λ_t dans la proposition 1.9 la fonction $h(t)$ ne suffit pas à caractériser un processus ponctuel auto-excité (cf. [18] page 9). Ces deux fonctions sont liées dans le cas d'un processus ponctuel ordonné par la relation suivante :

$$h(t) = E(\lambda_t).$$

On peut remarquer cependant que dans le cadre particulier d'un processus de Poisson non homogène (*NHPP*) les deux fonctions λ_t et $h(t)$ sont identiques puisque par définition un *NHPP* est un processus à accroissements indépendants sa fonction intensité conditionnelle ne dépend donc pas de son histoire.

Le nombre moyen de défaillances

Définition – 1.15 *On appelle* ***nombre moyen de défaillances****, ou aussi* ***fonction d'accumulation des défaillances*** *la fonction du temps m définie par :*

$$\forall t \geq 0 \ , \ m(t) = E(N_t).$$

Le ROCOF

Le ROCOF (Rate of OCcurrence Of Failures) ou taux instantané d'occurrence des défaillances correspond à la dérivée de la fonction d'accumulation des défaillances :

Définition – 1.16 *A l'instant t on appelle $ROCOF$ la quantité :*

$$ROCOF_t = m'(t) = \lim_{dt \to 0} \frac{E(N_{t+dt}) - E(N_t)}{dt}.$$

Le $ROCOF$ représente donc l'accroissement moyen du nombre de défaillances par unité de temps.

Remarque – Il est facile de montrer que le $ROCOF$ est égal à l'espérance mathématique de l'intensité de défaillance conditionnelle :

$$ROCOF_t = E(\lambda_t).$$

Fonction de hasard et taux de défaillance

Définition – 1.17 *On appelle* ***fonction de hasard*** *d'une v.a.r. T de densité f_T et de fonction de répartition F_T, la fonction définie par :*

$$\forall t \geq 0,\ z(t) = \frac{f_T(t)}{1 - F_T(t)}.$$

Quand la v.a.r. T représente une durée de vieΓon parle alors de taux de défaillance.

En Fiabilité des LogicielsΓon a souvent utilisé le terme "taux de défaillance" pour désigner le *ROCOF*.

On utilisera dans ce travail le terme **taux de défaillance** du logiciel pour désigner les fonctions de hasard des variables temps inter-défaillances X_i :

$$\forall i \geq 1,\ \forall x \geq 0,\ \lambda_i(x) = \frac{f_{X_i}(x)}{1 - F_{X_i}(x)}.$$

Remarques –

1. Ces taux de défaillance sont constants dans le cas où les v.a.r. X_i sont de lois exponentielles.

2. On a $\forall i \geq 1$ et $\forall x \geq 0$:

$$R_{t_i}(x) = exp\left[-\int_{t_i}^{t_i+x} \lambda_i(s)\, ds\right].$$

1.3.3 Approche globale de l'évaluation de la fiabilité des logiciels

L'approche globale pour l'évaluation de la fiabilité des logicielsΓtelle que décrite par Gaudoin et al [38] et Kanoun [50]Γpeut être décomposée en quatre étapes :

1. Une étape d'observation du logiciel étudié. A l'issue de cette étapeΓon dispose d'un certain nombre d'informations : environnement et profil d'utilisationΓprotocoles et effets des corrections...
   On dispose aussi d'un ensemble de données $x_1, \ldots, x_n$ décrivant en général le passé du processus de défaillance.
   Après s'être assuré de la qualité de ces donnéesΓon utilise les différents tests statistiques de tendance pour détecter le sens de l'évolution de la fiabilité au cours du temps.

2. Une étape de modélisation probabiliste où on tiendra compte des informations issues de la première étape pour proposer un modèleΓsouvent paramétrique : $\{P_\theta, \theta \in \Theta\}$ pour la loi de probabilité du processus de défaillance.

3. Une étape d'inférence et de prédictionΓoù on utilise les données collectées au cours de la première étape et le modèle de la deuxième étape pour estimer le paramètre θ à l'aide d'une procédure statistique appropriée :

$$\theta \simeq \hat{\theta}(x_1, \ldots, x_n).$$

On estime ensuite les différents attributs de la fiabilitéΓon a par exemple :

$$R_t(\tau) \simeq P_{\hat{\theta}(x_1,\ldots,x_n)}(N_{t+\tau} - N_t = 0).$$

4. Une étape de validation et de choix de modèles : cette étape permet de tester l'adéquation du modèle aux données observées. On comparera ensuite les performances du modèle considéré aux performances d'autres modèles.

1.4 Quelques modèles classiques de fiabilité des logiciels

Dans la littératureΓon trouve plusieurs classifications des modèles de fiabilité des logiciels.
Xie [102] par exempleΓpropose une classification se basant sur le type d'hypothèses probabilistes et de méthodes inférentielles utilisées dans les différents modèles. Il distingue ainsi plusieurs classes de modèlesΓparmi lesquelles :

- Les modèles markoviens : ce sont les modèles où le processus de défaillance $\{N_t\}_{t\geq 0}$ est supposé être un processus markovien. Cette hypothèse signifie queΓconditionnellement à l'état actuel du processus de défaillancesΓson état futur ne dépend pas de son état passé.
  L'intensité de défaillance dans ce cas sera une fonction discontinue constante entre deux défaillances successives.

- Les *NHPP* : Dans ces modèlesΓle processus de défaillance est modélisé par un processus de Poisson non homogène. Le nombre de défaillances observées jusqu'à l'instant t est alors une variable aléatoire de loi de Poisson de taux $m(t)$Γm étant une fonction paramétrique spécifiant le modèle *NHPP* utilisé.

- Les modèles bayésiens : ces modèles sont utilisés lorsqu'on dispose d'information a priori concernant le logiciel étudié. On utilise alors les méthodes d'inférence bayésiennes pour combiner l'information a priori et les observations issues des tests. On étudiera cette classe de modèles dans le chapitre 3.

- Les modèles métriques : dans ces modèles on donne une importance particulière aux mesures de complexité du logiciel. On cherche ensuite à établir une relation entre ces mesures de complexité et le nombre de défaillances du logiciel (cf. [53] et [69]).

Gaudoin [36] propose une autre classification se basant sur la forme de la fonction intensité de défaillance conditionnelle. Il dénombre ainsi quatre classes principales :

- Les modèles *ND* : où l'intensité conditionnelle de défaillance ne dépend (à travers une fonction ψ) que du Nombre de Défaillances survenues à chaque instant :

$$\forall t \geq 0 \ \Gamma \lambda_t = \psi(N_t).$$

- Les modèles *NDT* : où l'intensité conditionnelle ne dépend que du Nombre de Défaillances survenues à chaque instant et du Temps :

$$\forall t \geq 0 \ \Gamma \lambda_t = \psi(N_t, t).$$

- Les modèles *NDTE* : où l'intensité conditionnelle ne dépend que du Nombre de Défaillances survenues à chaque instantΓet du Temps Ecoulé depuis la dernière défaillance :

$$\forall t \geq 0 \ \Gamma \lambda_t = \psi(N_t, t - T_{N_t}).$$

- Les modèles *T* ou *NHPP* : où l'intensité n'est fonction que du Temps. Le processus de défaillance est alors un processus de Poisson non homogène :

$$\forall t \geq 0 \ \Gamma \lambda_t = \psi(t).$$

D'autres classifications ont été proposées par Bastani et RamamoorthyΓ[7]ΓMillerΓ[73]Γ TrachenbergΓ[98] et SingpurwallaΓet WilsonΓ[90].

On présente ci-dessous quelques unes des classes de modèles les plus utilisées.

1.4.1 Le modèle de Jelinski-Moranda et ses extensions

Le modèle de *Jelinski-Moranda* [47]Γprésenté en 1972Γest le premier modèle défini spécifiquement pour l'étude de la fiabilité des logiciels. Ce modèle qu'on notera dans la suite *JM*Γa donné suite à plusieurs généralisations et extensions.

Présentation du modèle

Jelinski et Moranda font les hypothèses suivantes :

- Avant le début des testsΓle logiciel contient un nombre fini mais inconnu N de fautes.
- Chaque faute détectée est supprimée en un temps négligeableΓaucune faute n'est introduite au cours des différentes corrections.
- A chaque instantΓl'intensité de défaillance conditionnelle est supposée être proportionnelle au nombre de fautes résiduelles :

$$\lambda_t = \Phi\,(N - N_t).$$

  La constante de proportionnalité $\Phi \in \mathbb{R}_+$ représente la qualité des différentes corrections. Cette qualité est supposée constante au cours du temps et indépendante des fautes supprimées.

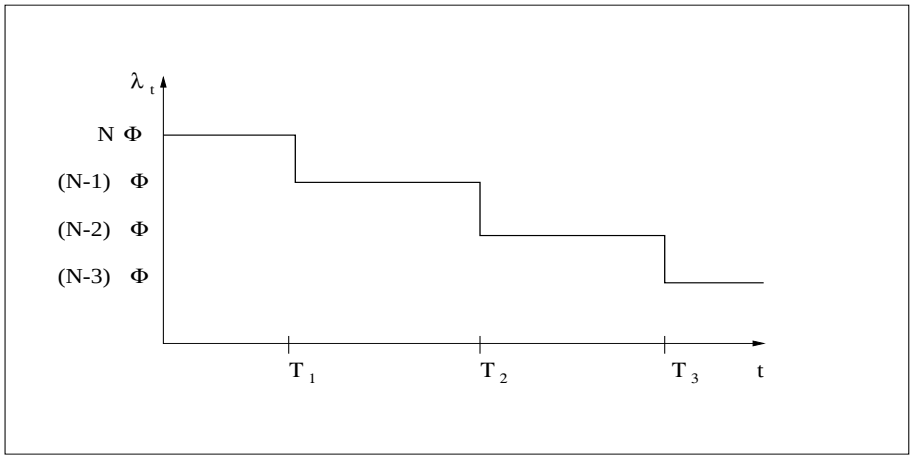

FIG. 1.3: Intensité de défaillance dans le modèle de Jelinski-Moranda.

Sous ces hypothèsesΓles temps inter-défaillances qui sont en nombre fini inconnu $X_1, X_2, \ldots, X_N$ sont des variables aléatoires réelles (v.a.r.) indépendantes de lois exponentielles :

$$X_i \sim Exp\,[\,\Phi\,(N - i + 1)\,].$$

A partir des observations $x_1, \ldots, x_n$ des n premiers temps inter-défaillancesΓon estime les paramètres N et Φ du modèle par la méthode du maximum de vraisemblance.

La simplicité des hypothèses de ce modèle fait qu'en pratiqueΓelles ne sont jamais vérifiées. Parmi ces hypothèsesΓles plus criticables sont :

- Le fait que les N fautes aient la même sévérité.
- L'hypothèse selon laquelle toutes les corrections sont parfaites.

A ces hypothèses trop simplistes s'ajoutent certains problèmes d'estimation des paramètres. En effetΓsous certaines conditions techniquesΓl'estimateur de maximum de vraisemblance de N est infini ou complètement aberrant (cf. [35] et [66]).
Littlewood et Verrall [68] donnent par ailleurs certaines conditions sur les observations des temps inter-défaillances x_i permettant d'avoir des estimations finies de N.

Extensions du modèle *JM*

C'est en essayant de résoudre les problèmes d'estimation du modèle *JM* que certains chercheurs ont utilisé les méthodes d'inférence bayésienne.

Meinhold et Sigpurwalla [72] et Langberg et Sigpurwalla [56] ont été les premiers à utiliser l'approche bayésienne pour estimer les paramètres N et Φ. Ils prennent différentes combinaisons de lois a priori pour N et Φ et donnent les différentes lois a posteriori associées.

Littlewood et Sofer [66] essayent de résoudre ces problèmes inférentiels en modifiant légèrement le modèle. Ils introduisent un nouveau paramètre $\lambda = \Phi\, N\Gamma$la fonction intensité de défaillance a alors la forme suivante :

$$\lambda_t = \lambda - \Phi\, N_t.$$

Ils choisissent des lois a priori *Gamma* pour les paramètres positifs λ et $\Phi\Gamma$et donnent sous ces hypothèses les expressions de la fiabilité a posterioriΓla loi a posteriori du taux de défaillance courantΓainsi que la loi a posteriori du nombre résiduel d'erreurs.

D'autres extensions bayésiennes du modèle *JM* peuvent être trouvées dans Jewell [48]Γ Raftery [81]ΓWright et Hazelhurst [100]Γetc.
Ces approches bayésiennes seront étudiées dans le chapitre 3. Au chapitre 2 on utilisera la théorie des modèles linéaires généralisés pour donner d'autres généralisations du modèle *JM*.

1.4.2 Les modèles *NHPP*

Dans cette classe de modèlesΓle processus de défaillance est modélisé par un processus de Poisson non homogène. A chaque instant $t\Gamma$la variable N_t est alors de loi de Poisson de paramètre $m(t)$:

$$P(N_t = n) = \frac{[m(t)]^n}{n!} e^{-m(t)}.$$

$m(t)$ représente le nombre moyen de défaillances ayant lieu sur $[0, t]$.

Rappelons (cf. par exemple [93] page 38) que le processus $\{N_t\}_{t\geq 0}$ est un processus de Poisson non homogène (*NHPP*)Γsi et seulement si il vérifie les propriétés suivantes :

1. $N_0 = 0\Gamma$
2. $\{N_t\}_{t\geq 0}$ est à accroissements indépendants :
   $\forall\ (t_0 < t_1 \ldots < t_n)\Gamma$les v.a.r. $(N_{t_1} - N_{t_0}), \ldots, (N_{t_n} - N_{t_{n-1}})$ sont indépendantesΓ
3. $P(N_{t+dt} - N_t = 1) = h(t)dt + o(dt)\Gamma\forall t \geq 0\Gamma$
4. $P(N_{t+dt} - N_t \geq 2) = o(dt)\Gamma\forall t \geq 0\Gamma$

où h est la fonction intensité de défaillance du processus.

Pour les processus *NHPP*Γ le nombre d'événements sur l'intervalle de temps $]t, t + s]$ est une v.a.r. de loi de Poisson de paramètre $m(t + s) - m(t)$.
Le nombre moyen de défaillances $m(t)$ est relié à la fonction intensité de défaillance par la relation suivante :

$$m(t) = \int_0^t h(s)ds.$$

La plupart des modèles *NHPP* supposent que la fonction intensité de défaillance est une fonction continue du temps. Cette hypothèse contredit le fait que toute correction effectuée

introduit forcément des modifications au logiciel qui engendrent des discontinuités dans les différents attributs de la fiabilité.

L'utilisation des modèles *NHPP* avec des fonctions intensité de défaillance continues peut cependant se justifier par le principe de *réparation minimale* (cf. Ascher [6]) qui énonce qu'un logiciel contenant beaucoup de fautes ne peut connaître que de très faibles variations de fiabilité.

On présente ci-dessous quelques uns des modèles *NHPP* les plus utilisés :

Le modèle de Crow

Le modèle de Crow [20] présenté en 1974Γappelé aussi modèle de Puissance [6] ou modèle de Duane [26]Γest l'un des plus anciens modèles *NHPP*.
Dans ce modèleΓl'intensité de défaillance a la forme suivante :

$$h(t) = \alpha\beta t^{\beta-1} \; \Gamma(\alpha,\beta) \in \mathbb{R}_+^{*2}.$$

La fonction de fiabilité est :

$$R_t(\tau) = exp\left[-\alpha\left((t+\tau)^\beta - t^\beta\right)\right].$$

Le nombre moyen de défaillances à l'instant t est donné par :

$$m(t) = \alpha\, t^\beta.$$

Le paramètre β représente le sens d'évolution de la fiabilité au cours du temps. Un β supérieur à un correspond à une décroissance de fiabilité alors qu'un paramètre β inférieur à un modélise une croissance de fiabilité.
Le paramètre α est un paramètre d'échelle.
Contrairement à la majorité des modèles de fiabilité des logicielsΓles estimateurs de maximum de vraisemblance des paramètres du modèle de Crow ont des expressions analytiques simples.
A l'instant t et après observation de n instants de défaillance $t_1, \ldots, t_n$Γles estimations de maximum de vraisemblance des paramètres α et β sont :

$$\hat{\alpha} = \frac{n}{t^{\hat{\beta}}} \text{ et } \hat{\beta} = \frac{n}{\sum_{i=1}^{n} ln(t/t_i)}.$$

Le modèle de Goel-Okumoto

Goel et Okumoto [42] présentent à leur tour en 1979 un modèle *NHPP* de fonction intensité de défaillance :

$$h(t) = \lambda e^{-\phi t} \text{ où } \lambda \in \mathbb{R}_+ \text{ et } \phi \in \mathbb{R}.$$

Le paramètre ϕ représente en quelque sorte la qualité de l'amélioration apportée par les corrections successives.
Le paramètre λ est un paramètre d'échelle qui représente le taux de défaillance initial.

La fonction de fiabilité dans ce modèle est :

$$R_t(\tau) = \begin{cases} exp\left[-\frac{\lambda}{\phi}e^{-\phi t}(1-e^{-\phi\tau})\right] & \text{si } \phi \neq 0 \\ exp(-\lambda\tau) & \text{si non.} \end{cases}$$

Lorsque $\phi > 0$Γla probabilité de ne plus observer de défaillances à partir de l'instant t vaut :

$$\lim_{\tau\to+\infty} R_t(\tau) = exp\left[-\frac{\lambda}{\phi}e^{-\phi t}\right],$$

cette probabilité non nulle permet de modéliser des logiciels qui peuventΓà partir d'un instant donnéΓne plus manifester de défaillances.

Le modèle hyperexponentiel de Kanoun-Laprie

Kanoun et Laprie [59] proposent un modèle *NHPP*Γappelé le modèle hyperexponentiel. Ils supposent dans ce modèle que l'intensité de défaillance tend vers une limite finie non nulle λ_r (appelée intensité de défaillance résiduelle). Ceci correspond au fait qu'en pratiqueΓun logiciel d'une complexité moyenne contiendra toujours quelques fautes et ne sera jamais parfait.

L'intensité de défaillance du modèle hyperexponentiel a la forme suivante :

$$\lambda(t) = \frac{\omega Z_1 e^{-Z_1 t} + (1-\omega)Z_2 e^{-Z_2 t}}{\omega e^{-Z_1 t} + (1-\omega)e^{-Z_2 t}}$$

où $\omega \in [0,1]$ΓZ_1 et $Z_2 \in \mathbb{R}_+^*$ sont les trois paramètres du modèle.

L'expression précédente a la même forme que celle du taux de défaillance d'une loi de Cox hyperexponentielle (cf. [19])Γd'où le nom du modèle.

L'intensité de défaillance résiduelle vaut :

$$\lambda_r = inf(Z_1, Z_2).$$

La fonction de fiabilité est :

$$R_t(\tau) = \frac{\omega e^{-Z_1(t+\tau)} + (1-\omega)e^{-Z_2(t+\tau)}}{\omega e^{-Z_1 t} + (1-\omega)e^{-Z_2 t}}$$

et le $MTTF$ est donné par :

$$MTTF_t = \frac{\omega(1/Z_1)e^{-Z_1 t} + (1-\omega)(1/Z_2)e^{-Z_2 t}}{\omega e^{-Z_1 t} + (1-\omega)e^{-Z_2 t}}.$$

Des extensions de ce modèleΓnotamment au cas du temps discretΓont été présentées par Kaâniche [49].

1.4.3 Les modèles à Profil Opérationnel Poissonnien Homogène

Introduction et propriétés

Le profil opérationnel Poissonnien homogène (*POPH*) est une modélisation simple mais assez générale du profil d'utilisation d'un logiciel.
Dans cette modélisation les sollicitations du logiciel sont supposées arriver d'une façon homogène dans le temps. Les entrées sollicitées sont supposées indépendantes entre elles indépendantes des instants de sollicitation et de même loi de probabilité Q sur E.

Remarque – On reprend ci-dessous la terminologie et les définitions de la section 1.2.

Définition – 1.18 *Le profil opérationnel est dit* ***Poissonnien homogène*** *quand :*

- *Le processus de sollicitation temporel est un processus de Poisson homogène d'intensité α.*
- *Les variables aléatoires entrées sollicitées Z_i sont indépendantes entre elles, indépendantes des instants de sollicitation et de même loi de probabilité Q sur E.*

Soler [94] donne un théorème permettant de spécifier les propriétés mathématiques du processus de défaillance sous les hypothèses d'un *POPH* :

Théorème – 1.19 *Pour un* POPH *avec corrections instantanées, il existe un processus de Markov $\Lambda = \{\Lambda_i\}_{i\geq 1}$ constitué de v.a.r. positives telles que, conditionnellement à $\{\Lambda_i = \lambda_i\}_{i\geq 1}$, les temps inter-défaillances X_i sont des v.a.r. indépendantes de lois exponentielles de paramètres respectifs λ_i. On a donc :*

$$\forall i \geq 1 \ , \ \textit{sachant } \Lambda_i = \lambda_i \ , \ X_i \sim Exp(\lambda_i).$$

On a par ailleurs :

$$\forall i \geq 1 \ , \ \Lambda_i = \alpha \ Q(F_{C_{i-1}}).$$

Remarque – Les v.a.r. Λ_i définies au théorème précédent seront dans la suite appelées **variables taux de défaillance**.

Gaudoin [36] montre que les variables taux de défaillance Λ_i vérifient les équations suivantes :

Théorème – 1.20 *Dans un* POPH *avec corrections instantanées, il existe deux suites de v.a.r. $(a_i)_{i\geq 1}$ (taux de bonne correction) et $(b_i)_{i\geq 1}$ (taux de mauvaise correction) à valeurs dans $[0,1]$, telles que les taux de défaillance $\{\Lambda_i\}_{i\geq 1}$ vérifient les équations :*

$$\forall i > 1, \ \Lambda_i = (1 - a_i - b_i)\, \Lambda_{i-1} + \alpha\, b_i.$$

Preuve – En général, une correction peut être en partie de bonne qualité et en partie de mauvaise qualité.
Ceci revient à dire que la $i^{ème}$ correction enlève une partie A_i de la faute totale $F_{C_{i-1}}$, mais en rajoute de nouvelles fautes représentées par une partie B_i de $\overline{F}_{C_{i-1}}$: complémentaire de $F_{C_{i-1}}$ dans l'ensemble des données d'entrée E.
La faute totale après la $i^{ème}$ correction est donc :

$$F_{C_i} = (F_{C_{i-1}} - A_i) \cup B_i \text{ où } A_i \in F_{C_{i-1}} \text{ et } B_i \in \overline{F}_{C_{i-1}}.$$

On conclut alors que :

$$Q(F_{C_i}) = Q(F_{C_{i-1}}) - Q(A_i) + Q(B_i).$$

Le taux de bonne correction a_i est donné par l'équation :

$$Q(A_i) = a_i Q(F_{C_{i-1}})$$

Comme $A_i \subset F_{C_{i-1}}$, on a alors $a_i \in [0, 1]$.

Le taux de mauvaise correction b_i est donné par l'équation :

$$Q(B_i) = b_i Q(\overline{F}_{C_{i-1}})$$

La v.a.r. b_i est à valeurs dans $[0, 1]$ car $B_i \subset \overline{F}_{C_{i-1}}$.

On obtient finalement l'équation :

$$Q(F_{C_i}) = (1-a_i-b_i)\, Q(F_{C_{i-1}}) + b_i.$$

En multipliant les deux membres de la formule précédente par α et en utilisant le théorème 1.19 on obtient le résultat énoncé. □

On décrit ci-dessous quelques modèles se basant sur les hypothèses du *POPH*.

Le modèle à double taux de correction déterministe

Dans ce modèle noté *MDTCD*, Gaudoin [36] suppose, pour simplifier, que les taux de correction $(a_i)_{i\geq 1}$ et $(b_i)_{i\geq 1}$ sont constants et déterministes :

$$\forall i \geq 1, a_i = a \text{ et } b_i = b.$$

Le premier taux de défaillance Λ_1 est supposé aussi déterministe. Les taux de défaillance successifs $\{\Lambda_i\}_{i\geq 1}$ sont alors des quantités déterministes notées $\{\lambda_i\}_{i\geq 1}$ et définies par :

$$\begin{cases} \lambda_1 &= \lambda \\ \lambda_{i+1} &= (1-a-b)\,\lambda_i + \alpha\, b \ , \forall i > 1. \end{cases}$$

Les temps inter-défaillances $\{X_i\}_{i\geq 1}$ sont alors des v.a.r. indépendantes de lois exponentielles de paramètres respectifs $\{\lambda_i\}_{i\geq 1}$.

Les différents paramètres de ce modèle peuvent être estimés par la méthode du maximum de vraisemblance.

Remarque – On peut remarquer que l'hypothèse selon laquelle les taux de mauvaise correction b_i sont constants n'est pas réaliste. En effet elle signifie que la taille des fautes introduites par les corrections augmente au fur et à mesure qu'on améliore le logiciel. Il est beaucoup plus réaliste de considérer un modèle où les taux de mauvaise correction sont décroissants.

Le Modèle Proportionnel Déterministe (*MPD*)

Dans la modélisation précédente on a supposé que l'effet des corrections est double. On peut simplifier ces hypothèses en supposant que l'effet d'une correction est soit bon soit mauvais. On n'aura ainsi qu'un seul taux de correction.
Sous cette hypothèse il est facile de prouver (cf. [36]) que les taux de défaillance $\{\Lambda_i\}_{i\geq 1}$ vérifient l'équation suivante :

$$\forall i \geq 1 \quad \Lambda_{i+1} = \gamma_i \cdot \Lambda_i \tag{1.1}$$

où Λ_1 et $(\gamma_i)_{i\geq 1}$ sont des v.a.r. positives indépendantes.

Parmi les modèles décrits par l'équation (1.1) et appelés **modèles proportionnels** on peut citer le modèle géométrique de Moranda [74] appelé aussi Modèle Proportionnel Déterministe (*MPD*) par Gaudoin et Soler [40].

Dans ce modèle les variables $(\gamma_i)_{i\geq 1}$ sont supposées déterministes et constantes plus précisément on a :

$$\forall i \geq 1 \quad \gamma_i = e^{-\theta},$$

où θ est un paramètre déterministe.

La variable taux de défaillance Λ_1 est aussi supposée déterministe et notée λ de telle sorte que la suite des taux de défaillance $\{\Lambda_i\}_{i\geq 1}$ est une suite de quantités déterministes notées $\{\lambda_i\}_{i\geq 1}$. Cette suite est définie par les équations :

$$\forall i \geq 1 \quad \lambda_i = e^{-\theta}\lambda_{i-1} = \lambda e^{-\theta(i-1)}.$$

Définition – 1.21 *On appelle* ***Modèle Proportionnel Déterministe*** *de paramètres* $\lambda \in \mathbb{R}_+$ *et* $\theta \in \mathbb{R}$*, le modèle de fiabilité des logiciels défini par l'hypothèse selon laquelle les v.a.r. temps inter-défaillances* X_i *sont indépendantes et de lois exponentielles :*

$$\forall i \geq 1 \ , \ X_i \sim Exp(\lambda\, e^{-\theta\,(i-1)}).$$

Le paramètre θ représente alors la qualité supposée constante des différentes corrections effectuées. Si les corrections sont de bonne qualité on aura $\theta > 0$; la suite des taux de défaillance a alors une décroissance géométrique.
Le deuxième paramètre de ce modèle λ est un paramètre d'échelle représentant le taux de défaillance initial.

Dans ce modèle on suppose que la proportion de fautes supprimées à chaque correction est proportionnelle à la taille de l'ensemble des fautes. S'il y a croissance de fiabilité la taille de l'ensemble des fautes va décroître ainsi que l'effet des corrections.

Après observation de n temps inter-défaillances $x_1, \ldots, x_n$ les estimations de maximum de vraisemblance de θ et λ sont données par les équations suivantes :

$$\begin{cases} \hat{\lambda} = \dfrac{n}{\sum_{i=1}^{n} e^{-\hat{\theta}(i-1)} x_i} \\ \\ \sum_{i=1}^{n} (n-2i+1)\, e^{-\hat{\theta}(i-1)} x_i = 0 \end{cases}$$

Gaudoin et Soler [40] donnent un certain nombre de propriétés statistiques de ces estimateurs.

Le Modèle Proportionnel Lognormal (*MPL*)

Gaudoin Lavergne et Soler [38] proposent un modèle généralisant le *MPD*. Ils supposent que les qualités des corrections successives sont des variables aléatoires $\{\Theta_i\}_{i\geq 1}$ indépendantes. En supposant que l'équipe de correcteurs a une manière de corriger assez régulière ils proposent un modèle où les v.a.r. $\{\Theta_i\}_{i\geq 1}$ sont de lois normales.

Définition – 1.22 *Le* ***Modèle Proportionnel Lognormal*** *de paramètres $\lambda \in \mathbb{R}_+$, $\theta \in \mathbb{R}$ et $\sigma^2 \in \mathbb{R}_+$ est le modèle où les v.a.r. temps inter-défaillances X_i sont de lois :*

$$X_i \sim Exp(\Lambda_i)$$

où :

- *Le premier taux de défaillance est déterministe :*

$$\Lambda_1 = \lambda$$

- *pour $i \geq 2$, les taux de défaillance Λ_i sont des v.a.r. données par les équations :*

$$\Lambda_{i+1} = e^{-\Theta_i}\ \Lambda_i$$

les Θ_i sont des v.a.r. indépendantes de même loi $\mathcal{N}(\theta, \sigma^2)$.

Remarques –

1. Les variables Λ_i sont alors de loi log-normale $[\, ln(\lambda) - (i-1)\,\theta, (i-1)\,\sigma^2\,]$ puisque les v.a.r. $ln(\Lambda_i)$ sont respectivement de lois $\mathcal{N}(ln(\lambda) - (i-1)\,\theta, (i-1)\,\sigma^2)$.
2. Dans le modèle *MPL*Γles v.a.r. X_i ne sont pas indépendantesΓleurs espérances et leurs variances sont :

$$E(X_i) = \frac{1}{\lambda} e^{-(i-1)(\theta+\sigma^2/2)} \text{ et } Var(X_i) = \frac{1}{\lambda^2} e^{-2(i-1)(\theta+\sigma^2/2)} \,[\, 2e^{(i-1)\sigma^2} - 1\,].$$

Les paramètres λΓθ et σ^2 de ce modèle peuvent être estimés en réécrivant le modèle sous la forme d'un modèle linéaire à deux composantes de la variance.

Un traitement bayésien de ce modèle sera présenté au chapitre 3.

1.5 Application des modèles

1.5.1 Traitement des données

Comme le souligne Kanoun [50]Γl'application des différents modèles de fiabilité des logiciels nécessite un traitement préliminaire des données de défaillance collectées.

La première étape de ce traitement consiste à s'assurer des bonnes conditions de la collecte des données : stabilité du profil d'utilisationΓbon enregistrement des défaillancesΓ absence de changements des spécifications... Cette partie du traitement des données doit se faire bien sûr en collaboration avec les équipes de développement du logiciel.

Il arrive cependant que certaines données collectées soient erronéesΓon peut dans ce cas utiliser des critères statistiques permettant de détecter les données aberrantes (cf. [2]). MaisΓcomme le souligne Kanoun [50]Γil est généralement préférable d'appliquer les modèles avec toutes les données disponibles et de n'enlever les valeurs aberrantes que si les résultats obtenus s'écartent significativement du comportement observé du logiciel.

Chaque modèle de fiabilité des logiciels a ses hypothèses propres concernant l'évolution de la fiabilité au cours du cycle de vie. Appliquer systématiquement un certain nombre de modèles aux données collectées sans tenir compte des hypothèses sous-jacentes aboutit généralement à des résultats non exploitables.

Il faut donc choisir les modèles en tenant compte de l'évolution réelle de la fiabilité du logiciel étudié. Cette évolution de la fiabilité est détectée par l'utilisation des tests de tendance.

1.5.2 Les tests de tendance

Ces tests permettent de savoir si la fiabilité "s'améliore" ou se "détériore" au cours du temps.

Généralement on peut répondre à cette question empiriquement en remarquant par exemple que les temps inter-défaillances sont de plus en plus grands. On dispose cependant de méthodes graphiques et de tests statistiques (cf. [6]) permettant de décider au vu des données si la fiabilité du logiciel s'améliore ou pas.
On pourra se référer à Gaudoin [36] pour une description détaillée de ces méthodes d'analyse de tendance.
On présente ci-dessous brièvement le plus utilisé de ces tests :

Le test de Laplace

Supposons qu'on observe un logiciel durant une période de temps $[0, T]$. Sous l'hypothèse d'une croissance de fiabilité les défaillances seront de plus en plus espacées et les observations des temps inter-défaillances $x_1, \ldots, x_n$ seront de plus en plus grands. Les instants de défaillance $t_1, \ldots, t_n$ seront plutôt proches de 0 que de T. La moyenne des t_i sera alors inférieure à $T/2$.

On conclura ainsi à une croissance de fiabilité si $\frac{1}{n}\sum_{i=1}^{n} t_i$ est significativement inférieure à $T/2$.

L'hypothèse nulle considérée ici est :

$$H_0 : \text{ “ il n'y a pas de tendance de fiabilité” .}$$

Pour des raisons de simplification on suppose que l'absence de tendance est équivalente à l'hypothèse que le processus des temps inter-défaillances est un processus de Poisson homogène.

Sous cette hypothèse nulle on utilise le théorème central limite pour obtenir le résultat de convergence suivant :

$$U_n = \frac{\sum_{i=1}^{n} T_i - n\,T/2}{\sqrt{nT^2/12}} \xrightarrow{Loi} \mathcal{N}(0,1).$$

Le test de Laplace consiste à calculer la valeur u_n observée de U_n et à conclure à une croissance de fiabilité si $u_n < \delta_\alpha$ et à une décroissance si $u_n > \mu_\alpha$ α étant le niveau de signification du test.
Si on note $F_{\mathcal{N}(0,1)}$ la fonction de répartition de la loi normale centrée réduite alors on a :

$$\delta_\alpha = F^{-1}_{\mathcal{N}(0,1)}(\alpha) \text{ et } \mu_\alpha = F^{-1}_{\mathcal{N}(0,1)}(1-\alpha).$$

Le test de Laplace permet ainsi de conclure avec une probabilité d'erreur α à la croissance ou non de la fiabilité sur la période observée.

1.5.3 Validation et comparaison de modèles

La présence de dizaines de modèles de fiabilité des logiciels nécessitent des techniques performantes de choix de modèles.

Iannino et al [45] proposent un certain nombre de critères à vérifier avant l'application des modèles :

- Validité des hypothèsesΓ
- applicabilitéΓ
- simplicitéΓ
- capacité à donner des estimations précises.

Ces critères permettent de sélectionner a priori un certain nombre de modèlesΓ cette sélection ne dépend que des propriétés intrinsèques des modèles.

L'application des modèles sur les jeux de données de défaillance permet de procéder à une deuxième sélection où l'on mesurera pour chaque modèle :

- sa capacité réplicative : c'est-à-dire son aptitude à reproduire le comportement passé du logiciel.
- sa capacité prévisionnelle : sa capacité à prévoir le comportement futur du logiciel.

Parmi les outils utilisés pour mesurer ces capacités on peut citer (cf. [1] et [60]) :

- le *u-plot* et le *y-plot*Γ
- la vraisemblance préquentielle (PLR)Γ
- le critère des résidusΓ
- le critère d'Akaike (cf. [3] et [54]).

Une étude détaillée du critère du *u-plot* est proposée au chapitre 4.

Chapitre 2

Modèles Linéaires Généralisés en Fiabilité des Logiciels

On s'intéresse dans ce chapitre à la classe des modèles de fiabilité des logiciels appelés modèles *ND* où la fonction intensité de défaillance conditionnelle λ_t ne dépend que du nombre de défaillances N_t.

On montre d'abord que certains modèles *ND* appartiennent à la famille des modèles linéaires généralisés.
En utilisant les propriétés générales de cette famille on présente des résultats nouveaux concernant ces modèles *ND*.
On présente ensuite différentes méthodes paramétriques et non paramétriques pour l'estimation de la fonction ψ reliant λ_t à N_t. Ces méthodes permettent de construire des modèles pouvant s'adapter aux spécificités de chaque jeu de donnéesΓils ont ainsi de meilleures qualités prévisionnelles que les modèles paramétriques usuels.
L'approche non paramétrique permet en outre de généraliser tous les modèles *ND* paramétriques.

2.1 Introduction

L'étude de la fiabilité d'un logiciel se fait à partir de l'étude de son processus de défaillance. Le comportement de ce processus est entièrement déterminé par sa fonction intensité de défaillance conditionnelle :

$$\lambda_t = \lim_{dt \to 0} \frac{1}{dt} P(N_{t+dt} - N_t = 1 | \mathcal{H}_t)$$

On s'intéresse ici aux modèlesΓnotés *ND* (à Nombre de Défaillances)Γoù λ_t ne dépend que de N_t :

$$\lambda_t = \psi(N_t),$$

la fonction ψ est une fonction à valeurs dans $\mathbb{R}_+$ spécifiant le modèle considéré.
On rappelle que la quantité $\lambda_t dt + o(dt)$ représente la probabilité d'occurrence d'une défaillance entre les instants t et $t + dt$. Cette probabilité instantanée est modifiée après

chaque correction. Il est donc naturel de faire dépendre λ_t de $N_t\Gamma N_t$ représente aussi bien le nombre de défaillances que le nombre de corrections effectuées jusqu'à l'instant t (cf. Hypothèse 2).

Sous l'hypothèse générale :

$$\lambda_t = \psi(N_t),$$

il est facile de montrer (cf. Snyder [93] page 265) que les variables aléatoires $X_1, X_2, \ldots$ sont indépendantes et de lois exponentielles de paramètres respectifs $\psi(i-1)\Gamma$ce que l'on note :

$$\forall i > 0 \ \Gamma X_i \sim Exp\,(\psi(i-1)).$$

A un changement de notation près on considèrera dans la suite que l'on aΓdans les modèles *ND* l'hypothèse suivante :

"$X_1, X_2, \ldots$ sont des v.a.r. indépendantes de lois :

$$\forall i > 0 \ \Gamma X_i \sim Exp\,(\psi(i))."$$

$\boldsymbol{H_{ND}}$

On s'intéressera dans ce chapitre aux différentes méthodes d'estimation de la **fonction taux de défaillance** ψ.

Parmi les modèles *ND* classiques on peut citer le modèle de Jelinski-Moranda [47] noté ici *JM*Γle modèle de Jelinski-Moranda Bayésien [66] ainsi que le modèle géométrique de Moranda [74] (ou modèle proportionnel déterministe [40]) noté ici *MPD*.

Les notations de ce chapitre étant fixéesΓnous consacrons la section 2 à rappeler la théorie des modèles linéaires généralisés.

Dans la section 3Γon montre que les modèles classiques de fiabilité des logiciels cités ci-dessus font partie de la famille des modèles linéaires généralisés (cf. aussi [34]). Ceci permet d'obtenir de nouvelles propriétés pour les modèles *JM* et *MPD*.

Dans la section 4 on présente une généralisation paramétrique de ces modèles *ND* classiques permettant d'avoir une meilleure adéquation aux jeux de données étudiés.

Dans la section 5Γon présente une approche non paramétrique permettant d'unifier tous les modèles *ND*. On présente à la fin de ce chapitre les résultats de l'application des différentes approches étudiées sur des jeux de données réels et simulés.

2.2 Modèles linéaires généralisés (*GLM*)

Les modèles linéaires généralisésΓintroduits par Nelder et Wedderburn en 1972 [78]Γ sont une extension des modèles linéaires classiques. Ils permettent de considérer une loi de probabilité autre que la loi gaussienne et un lien autre que l'identité.

Dans toute cette section on suppose qu'on est en présence de :

- Une variable à expliquer représentée par un vecteur aléatoire $X^{(n)} = (X_i)_{i\leq n}$ de $\mathbb{R}^n$. Dans le cadre de la Fiabilité des LogicielsΓ$X^{(n)}$ sera le vecteur des n premières variables temps inter-défaillances.

- Une observation $x^{(n)} = (x_i)_{i\leq n}$ du vecteur aléatoire $X^{(n)}$.

- Variables explicatives (ou régresseurs) $r_1, \ldots, r_p$Γvecteurs connus de $\mathbb{R}^n$. Ces vecteurs sont les colonnes d'une matrice R appelée **matrice plan d'expériences**.

On pourra trouver des études détaillées des modèles linéaires généralisés en particulier dans McCullagh et Nelder [71]ΓAntoniadis et al [4] et Fahrmeir et Tutz [33].

2.2.1 Définition d'un modèle linéaire généralisé

Définition – 2.1 *un* ***modèle linéaire généralisé*** *est un modèle paramétrique défini par les trois propriétés suivantes :*

1. *Les composantes du vecteur $X^{(n)}$ sont indépendantes.*

2. *Les lois de ces composantes appartiennent à une famille de lois P_α membre de la structure exponentielle naturelle au sens de Nelder (des exemples de telles familles sont les familles gaussienne, binomiale, Poisson, Gamma ...). Les v.a.r. X_i possèdent alors des densités non nulles s'écrivant sous la forme :*

$$f_{X_i}(x_i) = exp\left[\frac{\alpha_i x_i - b(\alpha_i)}{a(\phi)} + c(x_i, \phi)\right]. \tag{2.1}$$

   a, b et c sont des fonctions réelles connues caractérisant la famille de lois considérée. Les paramètres inconnus α_i sont appelés ***paramètres naturels****. ϕ est un paramètre réel appelé* ***paramètre de dispersion****.*
   On supposera dans la suite que le paramètre ϕ est un paramètre connu.

3. *Le vecteur des espérances $(E(X_i))_{i\leq n}$ est lié aux régresseurs linéaires $r_1, \ldots, r_p$ par la relation suivante :*

$$\forall i \leq n \ , \ g\left[E(X_i)\right] = \beta_1 r_{1,i} + \beta_2 r_{2,i} + \ldots + \beta_p r_{p,i},$$

   où :
   - g est une fonction réelle, connue, monotone différentiable appelée ***fonction de lien****.*
   - le paramètre $\beta = {}^t(\beta_1, \ldots, \beta_p)$ est un paramètre inconnu de $\mathbb{R}^p$.

Propriétés – De la définition précédente découlent les propriétés suivantes :

$$\forall i \leq n; \quad E(X_i) = b'(\alpha_i) \text{ et } Var(X_i) = b''(\alpha_i)a(\phi). \tag{2.2}$$

On peut en conclure que :

$$\forall i \leq n; \quad (g\,o\,b')(\alpha_i) = \sum_{j=1}^{p} \beta_j r_{j,i}. \tag{2.3}$$

La proposition suivante permettra d'utiliser les *GLM* pour traiter les données inter-défaillances en Fiabilité des Logiciels :

Proposition – 2.2 *Si $X_1, X_2, \ldots, X_n$ sont n v.a.r. de lois exponentielles :*

$$X_i \sim Exp(\lambda_i)$$

alors leurs densités respectives : $f_{X_i}(x) = \lambda_i\, exp(-\lambda_i x_i)$ peuvent s'écrire sous la forme (2.1) avec :

$$\alpha_i = -\lambda_i \ , \ a(\phi) = 1 \ , \ b(\alpha_i) = -ln(-\alpha_i) \ \textit{et} \ c(x_i, \phi) = 0.$$

Remarques et notations –

1. Les paramètres inconnus du *GLM* sont les vecteurs $\alpha = (\alpha_i)_{i\leq n}$ et βΓils sont liés par la relation (2.3). On ne s'intéressera dans la suite qu'à l'estimation du paramètre β.

2. Dans le cas particulier où le paramètre naturel α est une combinaison linéaire des régresseurs :

   $$\forall i \leq n\ \Gamma\alpha_i = \sum_{j=1}^{p} \beta_j r_{j,i},$$

   la fonction de lien g vérifie la relation : $g\,o\,b' = Id$Γet on parle alors de **fonction de lien canonique**.

3. Le vecteur $\sum_{j=1}^{p} \beta_j r_j$ qu'on notera η est appelé **prédicteur linéaire**. Sa $i^{ème}$ composante est notée η_i.

4. On notera dans la suite μ le vecteur $(E(X_i))_{i\leq n}$ et μ_i sa $i^{ème}$ composante.

2.2.2 Estimation de maximum de vraisemblance

Le problème d'inférence statistique qui se pose dans les modèles linéaires généralisés est l'estimation du paramètre inconnu β. Ce paramètre intervient dans la relation existant entre les espérances des v.a.r. X_i et les vecteurs de régression $r_1, .., r_p$ définissant le modèle :

$$\forall i \leq n\ \Gamma g[E(X_i)] = \sum_{j=1}^{p} \beta_j r_{j,i}.$$

Ce paramètre est généralement estimé par la méthode du maximum de vraisemblance. Dans un modèle linéaire généraliséΓla fonction de vraisemblance est :

$$L_{x_1,..,x_n}(\beta) \equiv \prod_{i=1}^{n} exp\left[\frac{\alpha_i x_i - b(\alpha_i)}{a(\phi)} + c(x_i,\phi)\right].$$

Notation – On note $\hat{\beta}_n$ l'estimateur de maximum de vraisemblance de β obtenu par maximisation de $L_{X_1,..,X_n}(\beta)$.

L'estimateur de maximum de vraisemblance $\hat{\beta}_n$ est obtenu par maximisation de la log-vraisemblance :

$$\mathcal{L}(\alpha) \equiv \sum_{i=1}^{n} \frac{\alpha_i X_i - b(\alpha_i)}{a(\phi)} + c(X_i,\phi)$$

Les paramètres α et β étant reliés par la relation (2.3) :

$$\forall i \leq n \ \Gamma(g\,o\,b')(\alpha_i) = \sum_{j=1}^{p} \beta_j r_{j,i}.$$

Les problèmes d'existence et d'unicité de $\hat{\beta}_n$ sont traités par exemple par Wedderburn [99].

Nelder et Wedderburn [78] suggèrent l'utilisation de la méthode des scores de Fisher pour l'évaluation numérique de $\hat{\beta}_n$.
Le schéma itératif découle alors de l'algorithme de Newton-Raphson où la matrice Hessienne :

$$Hess\mathcal{L}(\beta) = \frac{\partial^2 \mathcal{L}(\beta)}{\partial\beta\,\partial^t\beta}$$

est remplacée par son espérance.

Définition – 2.3 *Partant d'une estimation initiale $\hat{\beta}^{(0)}$, **la méthode des scores de Fisher** est décrite par le schéma itératif suivant :*

$$\hat{\beta}^{(k+1)} = \hat{\beta}^{(k)} + \left\{ E(-\frac{\partial^2 \mathcal{L}(\beta)}{\partial\beta\,\partial^t\beta}) \right\}^{-1} \frac{\partial \mathcal{L}(\beta)}{\partial\beta}.$$

Les dérivées sont évaluées au point $\hat{\beta}^{(k)}$.

Théorème – 2.4 *Dans le cas des* GLM, *la méthode des scores de Fisher se ramène au schéma suivant :*

$$\hat{\beta}^{(k+1)} = \hat{\beta}^{(k)} + ({}^t R\hat{W}^{(k)} R)^{-1} \ {}^t R\hat{W}^{(k)}\hat{Z}^{(k)}$$

où à chaque itération k on a :

- *$\hat{W}^{(k)}$ est la matrice diagonale définie par :*

$$\hat{W}_{ii}^{(k)} = \frac{1}{(g'(\hat{\mu}_i^{(k)}))^2\, b''(\hat{\alpha}_i^{(k)})}$$

- *Pour tout* $i \leq n$, $\hat{\mu}_i^{(k)}$ *et* $\hat{\alpha}_i^{(k)}$ *sont donnés par les relations suivantes :*

$$g(\hat{\mu}_i^{(k)}) = \sum_{j=1}^{p} \hat{\beta}_j^{(k)} r_{j,i} \text{ et } b'(\hat{\alpha}_i^{(k)}) = \hat{\mu}_i^{(k)}.$$

- $\hat{Z}^{(k)}$ *est le vecteur de* $\mathbb{R}^n$ *défini par ses composantes :*

$$\hat{Z}_i^{(k)} = (x_i - \hat{\mu}_i^{(k)})\, g'(\hat{\mu}_i^{(k)})$$

On pourra se référer à [71] page 40 pour une démonstration de ce théorème.

D'autres méthodes numériques pour le calcul de l'estimation $\hat{\beta}_n$ sont décrites par Antoniadis et al (cf. [4] page 147).

2.2.3 Propriétés asymptotiques

Fahrmeir et Kaufmann [32] montrent que sous certaines hypothèsesΓconcernant la matrice d'information de Fisher sur le paramètre $\beta\Gamma$l'estimateur $\hat{\beta}_n$ est consistant.

On précise ci-dessous quelques notations avant de donner le théorème de Fahrmeir et Kaufmann.

Notations et remarques –

1. $\mathcal{I}_n(\beta)$ est la matrice d'information de Fisher sur $\beta\Gamma$définie par :

$$\mathcal{I}_n(\beta) = E\left[-\frac{\partial^2}{\partial\beta\, \partial^t\beta} ln(L_{X_1,\ldots,X_n}(\beta)) \right].$$

   Pour les *GLM*Γon a le résultat suivant :

$$\mathcal{I}_n(\beta) = \frac{1}{a(\phi)}\ {}^tRWR$$

   où W est la matrice diagonale définie par ses éléments diagonaux :

$$W_{ii} = \frac{1}{(g'(\mu_i))^2\, b''(\alpha_i)}.$$

2. $\mathcal{I}_n^{1/2}(\beta)$ désigne la matrice vérifiant la propriété :

$$\mathcal{I}_n^{1/2}(\beta)\,.\,\mathcal{I}_n^{1/2}(\beta) = \mathcal{I}_n(\beta)$$

   On supposera dans la suite que cette matrice est inversibleΓsa matrice inverse est notée $\mathcal{I}_n^{-1/2}(\beta)$.

Théorème – 2.5 (Fahrmeir et Kaufmann [32]) *Si les deux hypothèses* FK_1 *et* FK_2 *décrites ci-dessous sont vérifiées, la suite des estimateurs de maximum de vraisemblance* $\hat{\beta}_n$ *est asymptotiquement gaussienne :*

$$\mathcal{I}_n^{1/2}(\beta)(\hat{\beta}_n - \beta) \overset{n\to+\infty}{\longrightarrow} \mathcal{N}(0, Id_p), \text{ en loi.}$$

***FK*$_1$** :

$$\lambda_{min}(\mathcal{I}_n(\beta)) \overset{n\to+\infty}{\longrightarrow} +\infty,$$

où $\lambda_{min}(\mathcal{I}_n(\beta))$ *est la plus petite valeur propre de la matrice d'information* $\mathcal{I}_n(\beta)$.

***FK*$_2$** :

$\forall \delta > 0, \forall \lambda \in \mathbb{R}^p \,/\, \| \lambda \| = 1$
$\max_{\gamma \in V_n(\delta)} \| \mathcal{I}_n^{-1/2}(\beta)\mathcal{I}_n(\gamma)\,{}^t\mathcal{I}_n^{-1/2}(\beta) - Id_p \|^{n\to+\infty} 0$, *en probabilité sous la vraie loi de probabilité* P *et sous* P_{β_n},

où $\beta_n = \beta + \delta\mathcal{I}_n^{-1/2}(\beta)\lambda$, β *étant le vrai paramètre à estimer.*

$$V_n(\delta) = \{\gamma \in \mathbb{R}^p / \| {}^t\mathcal{I}_n^{1/2}(\beta)(\beta - \gamma) \| \leq \delta\}.$$

Pour le cas particulier où la fonction de lien g *est la fonction de lien canonique, l'hypothèse* FK_2 *est remplacée par une hypothèse plus faible :*

***FK*$_2^*$** :

$$\forall \delta > 0, \max_{\gamma \in V_n(\delta)} \| \mathcal{I}_n^{-1/2}(\beta)\mathcal{I}_n(\gamma)\mathcal{I}_n^{-1/2}(\beta) - Id_p \|^{n\to+\infty} 0.$$

L'estimateur $\hat{\beta}_n$ est donc asymptotiquement de loi gaussienne : $\mathcal{N}(\beta, a(\phi)\,({}^tRWR)^{-1})\Gamma$ ceci permet de construire des intervalles de confiance et des tests d'hypothèses sur le paramètre β.

2.2.4 Qualité d'ajustement et déviance

Après avoir estimé les paramètres du modèle linéaire généralisé de famille de lois $P_\alpha\Gamma$il est intéressant de chercher un critère évaluant la qualité de l'ajustement effectué.

Un tel critère peut être obtenu à partir de la fonction log-vraisemblance :

Remarque – La fonction log-vraisemblance est considérée ici comme une fonction du vecteur des espérances $\mu\Gamma$elle est notée :

$$\mathcal{L}(x^{(n)}, \mu) \equiv lnL_{x_1,\ldots,x_n}(\beta).$$

La log-vraisemblance estimée est :

$$\mathcal{L}(x^{(n)}, \hat{\mu}) = lnL_{x_1,\ldots,x_n}(\hat{\beta}_n).$$

Un modèle linéaire généralisé ajuste bien les données si sa log-vraisemblance estimée est élevée.
Or parmi les modèles linéaires généralisés de famille de lois $P_\alpha\Gamma$le modèle ayant la plus grande log-vraisemblance estimée est le modèle plein :

Définition – 2.6 *Ayant n observations $x_1, \ldots, x_n$ et une famille de lois P_α membre de la structure exponentielle naturelle, on appelle* ***modèle plein*** *le modèle linéaire généralisé de famille de lois P_α ayant autant de paramètres que d'observations.*
Pour ce modèle on a :

$$\forall i \leq n \ , \ \hat{\mu}_i = g^{-1}(\sum_{j=1}^{p} \hat{\beta}_j r_{j,i}) \ = x_i.$$

La log-vraisemblance du modèle plein est alors notée $\mathcal{L}(x^{(n)}, x^{(n)})$.

Le modèle pleinΓn'a aucun pouvoir prédictif puisqu'il ne fait que coller aux données. Il permet cependant de majorer les fonctions log-vraisemblance sous tous les autres modèles linéaires généralisés de famille de lois P_α. Il permet ainsi de définir le critère de déviance permettant de comparer les qualités d'ajustement de plusieurs modèles :

Définition – 2.7 *Soit M un modèle linéaire généralisé de famille de lois P_α, dans ce modèle le vecteur des espérances μ est approché par $\hat{\mu}_M$.*
On appelle ***déviance*** *du modèle M la quantité :*

$$Dev(x^{(n)}, M) = 2[\, \mathcal{L}(x^{(n)}, x^{(n)}) - \mathcal{L}(x^{(n)}, \hat{\mu}_M)\,].$$

Un modèle M ajuste bien les observations si sa fonction de vraisemblance se rapproche de celle du *modèle plein*. Le modèle résume bien les observations s'il implique un faible nombre de paramètres et si sa déviance est assez faible.

2.2.5 Tests d'hypothèses

Soit M_1 le modèle linéaire généralisé défini par une famille de lois P_αΓune fonction de lien g et p régresseurs $r_1, .., r_p$. Dans ce modèle on a la relation :

$$E(X_i) = \mu_i = g^{-1}(\eta_i) = g^{-1}(\sum_{j=1}^{p} \beta_j r_j^i)$$

Notations –

1. On note E_p l'espace vectoriel de dimension p engendré par les régresseurs $r_1, \ldots, r_p$.
2. On note E_q un sous espace vectoriel de E_pΓde dimension $q < p$.

On souhaite tester l'hypothèse nulle H_0 : "$\eta \in E_q$" contre l'hypothèse H_1 : "$\eta \in E_p$".
Ce test nous permettraΓentre autresΓde choisir entre le modèle linéaire généralisé M_1 et un deuxième modèle M_0 défini par la même famille de lois P_αΓla même fonction de lien g mais seulement q régresseurs $r_{(1)}, .., r_{(q)}$.
$r_{(1)}, .., r_{(q)}$ sont q régresseurs choisis parmi les p régresseurs initiaux.

Il existe un certain nombre de tests permettant de tester l'hypothèse H_0. On peut considérer par exemple le **test de rapport de vraisemblances maximales**.

Pour ce testΓon rejette l'hypothèse H_0 contre l'hypothèse H_1Γsi la différence entre les vraisemblances maximales dans les deux modèlesΓnotées respectivement : $L(x^{(n)}, M_0)$ et $L(x^{(n)}, M_1)$ est jugée trop importante.

La zone de rejet de l'hypothèse H_0 au seuil α est donc :

$$D_\alpha = \left\{ x \in \mathbb{R}^n / \frac{L(x, M_0)}{L(x, M_1)} < C_\alpha \right\}.$$

Le modèle M_0 étant inclus dans le modèle M_1Γon a : $L(x, M_0) \leq L(x, M_1)$Γ$\forall x \in \mathbb{R}^n$.

La détermination de la zone D_α nécessite la connaissance de la loi du rapport $\frac{L(X^{(n)}, M_0)}{L(X^{(n)}, M_1)}$ sous l'hypothèse H_0. Ceci est donné par le résultat suivant :

Proposition – 2.8 (cf. [4] page 236) *Sous l'hypothèse H_0 on a :*

$$Dev(X^{(n)}, M_0) - Dev(X^{(n)}, M_1) \text{ converge en loi vers la loi du } \mathcal{X}^2_{p-q}.$$

L'hypothèse H_0 sera rejetée au seuil α si le vecteur des observations $x^{(n)}$ est dans la zone :

$$D_\alpha = \left\{ x \in \mathbb{R}^n / Dev(x, M_0) - Dev(x, M_1) \geq F^{-1}_{\mathcal{X}^2_{p-q}}(1-\alpha) \right\}.$$

2.3 Les modèles linéaires généralisés en Fiabilité des Logiciels

Certains modèles classiques en Fiabilité des Logiciels sont des modèles linéaires généralisés. C'est le cas par exemple pour le modèle proportionnel déterministe et le modèle de Jelinski-Moranda.
On utilise dans cette section les propriétés générales des modèles linéaires généralisés pour obtenir de nouveaux résultats concernant les deux modèles *MPD* et *JM*.

2.3.1 Le Modèle Proportionnel Déterministe (*MPD*)

Dans le modèle proportionnel déterministe (cf. [74] et [40]) de paramètres $\lambda \in \mathbb{R}_+$ et $\theta \in \mathbb{R}$Γ les variables temps inter-défaillances sont des variables indépendantes de lois exponentielles :

$$X_i \sim Exp(\lambda e^{-(i-1)\theta}).$$

Notations –

1. On posera dans la suite : $\boldsymbol{\nu} \equiv -ln\lambda$.

2. Si on a n temps inter-défaillances observés $x_1, x_2, \ldots, x_n\Gamma$les estimateurs de maximum de vraisemblance des paramètres (ν, θ) sont notés $(\hat{\boldsymbol{\nu}}_{\boldsymbol{n}}, \hat{\boldsymbol{\theta}}_{\boldsymbol{n}})$.

On montre ci-dessous que le fait d'écrire le *MPD* comme un modèle linéaire généralisé permet de préciser le comportement asymptotique des estimateurs de maximum de vraisemblance $\hat{\nu}_n$ et $\hat{\theta}_n$.

Le modèle proportionnel déterministe vu comme un *GLM*

Proposition – 2.9 *Le* MPD *est un modèle linéaire généralisé de famille de lois la famille des lois exponentielles, de fonction de lien la fonction* $g(x) = ln(x)$*, et de vecteurs de régression :* $r_1 = \begin{pmatrix} 1 \\ 1 \\ . \\ . \\ 1 \end{pmatrix}$ *et* $r_2 = \begin{pmatrix} 0 \\ 1 \\ . \\ . \\ n-1 \end{pmatrix}$.

Preuve – Il s'agit de démontrer que le *MPD* vérifie les trois propriétés de la définition – 2.1.
Les deux premières propriétés sont évidentes (cf. proposition – 2.2). Pour la troisièmeΓ exprimée par la relation :

$$\forall i \leq n \ \Gamma g\left[E(X_i)\right] = \beta_1 r_{1,i} + \beta_2 r_{2,i} + \ldots + \beta_p r_{p,i},$$

il suffit de remarquer que dans le *MPD* on a les relations :

$$\forall i \leq n \ \Gamma ln\left[E(X_i)\right] = \nu + \theta(i-1).$$

La troisième propriété est ainsi vérifiée avec :
- $g(x) = ln(x)\Gamma$
- $\beta = {}^t(\nu, \theta)\Gamma$
- $r_1 = \begin{pmatrix} 1 \\ 1 \\ . \\ . \\ 1 \end{pmatrix}$ et $r_2 = \begin{pmatrix} 0 \\ 1 \\ . \\ . \\ n-1 \end{pmatrix}$.

□

Le *MPD* fait ainsi partie de la famille des modèles linéaires généralisés. On montre dans ce qui suit que les hypothèses permettant d'établir la normalité asymptotique des estimateurs de maximum de vraisemblance sont vérifiées dans le cas du *MPD*.

Propriétés asymptotiques des estimateurs

Proposition – 2.10 *Dans le cas du* MPD*, les deux hypothèses FK_1 et FK_2 du théorème 2.5 sont vérifiées et permettent donc d'avoir la normalité asymptotique des estimateurs de maximum de vraisemblance $(\hat{\nu}_n, \hat{\theta}_n)$:*

$$({}^tRR)^{1/2}\left[\begin{pmatrix}\hat{\nu}_n\\ \hat{\theta}_n\end{pmatrix}-\begin{pmatrix}\nu\\ \theta\end{pmatrix}\right] \xrightarrow{Loi} \mathcal{N}(0, Id_2).$$

Preuve – Commençons par montrer que l'hypothèse FK_1 :

$$\lambda_{min}(\mathcal{I}_n(\beta)) \overset{n\to+\infty}{\longrightarrow} +\infty$$

est vraie. On a :

$$\mathcal{I}_n(\beta) = \frac{1}{a(\phi)}\ {}^tRWR$$

avec $a(\phi) = 1$ pour les lois exponentielles.

Or dans le cas du *MPD* la matrice R vaut $(r_0, r_1) = \begin{pmatrix}1 & 0\\ 1 & 1\\ . & .\\ . & .\\ 1 & n-1\end{pmatrix}$

et :

$$W = Id_n$$

en effet W est la matrice diagonale définie par :

$$W_{ii} = \frac{1}{[g'(\mu_i)]^2\ b''(\alpha_i)}$$

Pour le *MPD* on a $g(x) = ln(x)$ donc $g'(\mu_i) = \frac{1}{\mu_i}$.

Pour la loi exponentielle on a : $b(\alpha_i) = -ln(-\alpha_i)\Gamma$donc $b''(\alpha_i) = \frac{1}{\alpha_i^2}$.

Comme $\mu_i = b'(\alpha_i) = -1/\alpha_i\ \Gamma$on a ainsi :

$$\forall i \leq n\ \Gamma W_{ii} = 1.$$

Finalement on a

$$\mathcal{I}_n(\beta) = {}^tRR = \begin{pmatrix} n & \frac{n(n-1)}{2}\\ \frac{n(n-1)}{2} & \frac{n(n-1)(2n-1)}{6}\end{pmatrix}.$$

Cette matrice d'information ne dépend pas du paramètre β. Et on a :

$$\lambda_{min}(\mathcal{I}_n(\beta)) = \frac{1}{2}[n + \frac{n(n-1)(2n-1)}{6} - \sqrt{\Delta_n}].$$

$$\text{où } \Delta_n = [n + \frac{n(n-1)(2n-1)}{6}]^2 - 4[\frac{n^2(n-1)(2n-1)}{6} - \frac{n^2(n-1)^2}{4}].$$

On montre facilement qu'au voisinage de $+\infty$ on a l'équivalence suivante : $\lambda_{min}(\mathcal{I}_n(\beta)) \sim \frac{n}{4}$.
L'hypothèse FK_1 est ainsi bien vérifiée.

La deuxième hypothèse FK_2 est :

$$\max_{\gamma \in V_n(\delta)} \| \mathcal{I}_n^{-1/2}(\beta)\mathcal{I}_n(\gamma)\, {}^t\mathcal{I}_n^{-1/2}(\beta) - Id_p \| \overset{n\to+\infty}{\longrightarrow} 0 \ .$$

Cette hypothèse est vérifiée puisque la matrice $\mathcal{I}_n(\beta)$Γdans le cas du *MPD*Γne dépend pas de βΓon a alors :

$$\mathcal{I}_n^{-1/2}(\beta)\mathcal{I}_n(\gamma)\, {}^t\mathcal{I}_n^{-1/2}(\beta) = Id_p, \forall \gamma \in \mathbb{R}^p.$$

□

On peut ainsi utiliser les résultats du théorème 2.5 et obtenir les lois asymptotiques des estimateurs de maximum de vraisemblance des paramètres du *MPD*. On obtient alors les quatre corollaires présentés ci-dessous.

Corollaire – 1

$$\left\{\frac{n(n+1)}{2(2n-1)}\right\}^{1/2} (\hat{\nu}_n - \nu) \quad \overset{Loi}{\longrightarrow} \quad \mathcal{N}(0,1).$$

$$\left\{\frac{n(n+1)(n-1)}{12}\right\}^{1/2} (\hat{\theta}_n - \theta) \quad \overset{Loi}{\longrightarrow} \quad \mathcal{N}(0,1).$$

Quand le nombre d'observations n est assez grandΓon peut considérer que le vecteur $\begin{pmatrix} \hat{\nu}_n \\ \hat{\theta}_n \end{pmatrix}$ est de loi $\mathcal{N}(\begin{pmatrix} \nu \\ \theta \end{pmatrix}, ({}^tRR)^{-1})$.
La matrice $({}^tRR)^{-1}$ valant $\begin{pmatrix} \frac{2(2n-1)}{n(n+1)} & \frac{-6}{n(n+1)} \\ \frac{-6}{n(n+1)} & \frac{12}{n(n+1)(n-1)} \end{pmatrix}$

Le comportement asymptotique des variances des estimateurs de maximum de vraisemblance est décrit par les équivalences suivantes :

$$Var(\hat{\nu}_n) \sim \frac{2(2n-1)}{n(n+1)} \text{ et } Var(\hat{\theta}_n) \sim \frac{12}{n(n+1)(n-1)}.$$

Remarque – Gaudoin et Soler [40] montrent qu'une transformation adéquate des données permet d'écrire le *MPD* sous forme d'un modèle linéaire classique. Ils introduisent alors des estimateurs des moindres carrés notés : $(\tilde{\boldsymbol{\nu}}_{\boldsymbol{n}}, \tilde{\boldsymbol{\theta}}_{\boldsymbol{n}})$ dont les variances sont :

$$Var(\tilde{\nu}_n) = \frac{(2n-1)\pi^2}{3n(n+1)} \text{ et } Var(\tilde{\theta}_n) = \frac{2\pi^2}{n(n+1)(n-1)}.$$

Il est intéressant de comparer les variances des estimateurs $(\tilde{\nu}_n, \tilde{\theta}_n)$ et $(\hat{\nu}_n, \hat{\theta}_n)$:

$$Var(\hat{\nu}_n) \simeq \frac{6}{\pi^2} Var(\tilde{\nu}_n) \text{ et } Var(\hat{\theta}_n) \simeq \frac{6}{\pi^2} Var(\tilde{\theta}_n).$$

On montre donc ici que les estimateurs de maximum de vraisemblance ontΓ pour un nombre d'observations élevéΓdes variances inférieures à celles des estimateurs des moindres carrés.

Corollaire – 2 L'estimateur du paramètre d'échelle λ : $\hat{\lambda}_n = e^{-\hat{\nu}_n}$ est asymptotiquement de loi log-normale$(-\nu, Var(\hat{\nu}_n))$.

On a donc pour n assez grand :

$$E(\hat{\lambda}_n) \simeq \lambda(1 + \frac{2}{n}) \text{ et } Var(\hat{\lambda}_n) \simeq \frac{4\lambda^2}{n}$$

puisque si $X \sim$ log-normale(m, σ^2)Γon a $E(X) = e^{m+\frac{\sigma^2}{2}}$ et $Var(X) = e^{2m+\sigma^2}(e^{\sigma^2} - 1)$.
L'estimateur $\hat{\lambda}_n$ est donc asymptotiquement sans biais et consistant.

Comportement asymptotique de l'estimateur du *MTTF*

Après observation de n défaillancesΓl'expression du $MTTF$ dans le *MPD* est :

$$MTTF_n = E(X_{n+1}) = exp(\nu + n\theta).$$

Les paramètres (ν, θ) étant estimés par la méthode du maximum de vraisemblanceΓon en déduit un estimateur de $MTTF_n$:

$$\widehat{MTTF}_n = exp(\hat{\nu}_n + n\hat{\theta}_n).$$

En utilisant les résultats asymptotiques concernant les estimateurs $\hat{\nu}_n$ et $\hat{\theta}_n$ (Corollaire – 1)Γon montre que :

Corollaire – 3

$$\frac{\sqrt{n}}{2}\ \left\{(\hat{\nu}_n + n\hat{\theta}_n) - (\nu + n\theta)\right\} \xrightarrow{Loi} \mathcal{N}(0,1).$$

Par conséquent pour n suffisamment grandΓon peut considérer que :

$$\widehat{MTTF}_n \sim \text{log-normale}(\nu + n\theta, \frac{4}{n}).$$

On a pour n assez grand :

$$E(\widehat{MTTF}_n) \simeq exp(\nu + n\theta) \text{ et } Var(\widehat{MTTF}_n) \simeq \frac{4e^{2n\theta}}{n\lambda^2}.$$

Cet estimateur est donc asymptotiquement sans biaisΓsa variance est par contre assez importante.

On peut en outreΓpour n assez grandΓdonner des intervalles de confiance pour le $MTTF$. En effetΓen utilisant le corollaire – 3Γon montre facilement le résultat suivant :

Corollaire – 4 Pour n suffisamment grandΓon a un intervalle de confiance contenant le $MTTF$ avec une probabilité approximative de $1-\alpha$Γcet intervalle est :

$$I_{n,\alpha} = [\widehat{MTTF}_n\, exp(-\frac{2u_\alpha}{\sqrt{n}}), \widehat{MTTF}_n\, exp(\frac{2u_\alpha}{\sqrt{n}})],$$

où $u_\alpha = F^{-1}_{\mathcal{N}(0,1)}(1-\alpha/2)$ et $F_{\mathcal{N}(0,1)}$ est la fonction de répartition de la loi normale centrée réduite.

En choisissant $\alpha = 5\%$Γla largeur de cet intervalle de confiance est :

$$\begin{aligned} \mid I_{n,\alpha} \mid &= \widehat{MTTF}_n\, [\frac{4u_\alpha}{\sqrt{n}}] \\ &\simeq \widehat{MTTF}_n\, \frac{8}{\sqrt{n}}. \end{aligned}$$

On montre donc que pour avoir un intervalle de confiance de largeur : $\widehat{MTTF}_n$. 20%Γil faudrait avoir au moins 1600 observations.
L'intervalle de confiance proposé n'est donc utilisable que pour des jeux de données de taille importrante. Ceci n'est généralement pas le cas en Fiabilité des Logiciels.

2.3.2 Les modèles de Jelinski-Moranda

Dans le modèle JM initialΓles temps inter-défaillances $X_1, X_2, \ldots, X_N$ sont indépendants de lois exponentielles :

$$X_i \sim Exp((N-i+1)\,\Phi),$$

où $N \in \mathbb{N}$ et $\Phi \in \mathbb{R}_+$ sont les deux paramètres inconnus du modèle.
Dans une deuxième paramétrisation de ce modèleΓLittlewood et Sofer [66] font l'hypothèse suivante :

$$X_i \sim Exp(\lambda - \Phi(i-1)).$$

Dans les deux versions précédentes du modèle *JM*Γles variables temps inter-défaillances X_i sont indépendantes et de lois respectives $Exp(a+bi)$ Γ$\forall i \leq n$.

Notations –

1. On appellera dans la suite *Modèles de Jelinski-Moranda* les modèles de Fiabilité des Logiciels définis par l'hypothèse selon laquelle les temps inter-défaillances $X_1, X_2, \ldots$ sont indépendantes de lois exponentielles :

$$\forall i \geq 1 \ \Gamma X_i \sim Exp(a+bi), \ a \text{ et } b \in \mathbb{R} \text{ tels que } a+bi \geq 0.$$

2. L'estimateur de maximum de vraisemblance du paramètre $\beta = \begin{pmatrix} a \\ b \end{pmatrix}$ est noté : $\hat{\beta}_n \equiv \begin{pmatrix} \hat{a}_n \\ \hat{b}_n \end{pmatrix}$.

Les modèles de Jelinski-Moranda vus comme des *GLM*

Proposition – 2.11 *Tout comme le* MPD, *les modèles* JM *sont des modèles linéaires généralisés de famille de lois, la famille de lois exponentielles, de fonction de lien la fonction : $g(x) = 1/x$. Les vecteurs de régression considérés sont les mêmes que ceux du* MPD.

Remarques –

1. Les modèles *JM* et le modèle *MPD* sont issus du même type de modèles linéaires généralisésΓils ne diffèrent que par leurs fonctions de lien.
2. Dans les modèles *JM*Γla fonction de lien utilisée $g(x) = 1/x$ est la fonction de lien canonique pour la famille de lois exponentielles.

2.4 Généralisation polynômiale de quelques modèles *ND*

L'hypothèse commune aux modèles *JM* et *MPD* est :

$$\text{“}X_1, X_2, \ldots \text{ sont des v.a.r. indépendantes de lois :}$$

$$X_i \sim Exp\,[\,h(a+bi)\,]\text{”}$$

$\boldsymbol{H_{lin}}$

où :

- h est une fonction connue à valeurs dans $\mathbb{R}_+\Gamma$
- les paramètres a et b sont deux paramètres réels inconnus.

L'hypothèse H_{lin} définitΓpour différentes fonctions $h\Gamma$une famille de modèles qu'on peut appeler **modèles *ND* linéaires**.

Dans cette section on présente d'abord une méthode graphique permettant de mesurer l'adéquation du *MPD* aux jeux de données étudiés.
Cette méthode montre que pour certains jeux de donnéesΓle modèle *MPD* ne suffit pas à bien décrire les données observées. Dans ce casΓil est alors intéressant de généraliser l'hypothèse H_{lin} en remplaçant le prédicteur linéaire : $\eta_i = a + bi$ par un prédicteur polynômial : $\eta_i = \sum_{j=0}^{p} \beta_j i^j$.

On présente alors différents outils statistiques permettantΓaussi bien pour le *MPD* que pour le *JM*Γde choisir le polynôme approprié pour chaque jeu de données.

On obtient ainsi des modèles ***ND* polynômiaux** ayant une meilleure adéquation aux différents jeux de données étudiés.

2.4.1 Validation du *MPD*

Dans le modèle *MPD* de paramètres ν et $\theta\Gamma$on fait l'hypothèse suivante :

$$\text{“}X_1, X_2, \ldots \text{ sont des v.a.r. indépendantes de lois :}$$

$$X_i \sim Exp\,[\,exp(-\nu - \theta(i-1))\,]\text{.”}$$

$\boldsymbol{H_{MPD}}$

La procédure de validation du *MPD* est basée sur la proposition suivante :

Proposition – 2.12 *Sous l'hypothèse H_{MPD} on a :*

$$\forall i \geq 0\ ,\quad ln(X_i) = \theta(i-1) + (\nu - \gamma_E) + \epsilon_i$$

où :

- *la constante* γ_E *est la constante d'Euler :* $\gamma_E = 0.577..$
- $(\epsilon_i)_{i\geq 1}$ *sont des v.a.r. de loi Gumbel centrée de variance* $\frac{\pi^2}{6}$.

Preuve –

Sous l'hypothèse H_{MPD} on a pour tout $i \geq 1$:

$$X_i \sim Exp\,[\,exp(-\nu - \theta(i-1))\,]$$

donc :

$$\forall i \geq 1\ \Gamma\ \left[exp(-\nu - \theta(i-1))\right].X_i \sim Exp(1)$$

les v.a.r.

$$ln\,\{\,[exp(-\nu - \theta(i-1))]\,.\,X_i\,\}$$

sont alors i.i.d. de fonction de répartition F_G :

$$\forall x \in \mathbb{R}\ \Gamma\ F_G(x) = 1 - exp(-e^x),$$

F_G est la fonction de répartition d'une loi de *Gumbel* (cf. par exemple [86] page 48) de moyenne $-\gamma_E$ et de variance $\frac{\pi^2}{6}$.

On a finalement pour tout $i \geq 1$:

$$ln(X_i) = \theta(i-1) + (\nu - \gamma_E) + \epsilon_i$$

où les ϵ_i sont des v.a.r. de loi *Gumbel* de moyenne 0 et de variance $\frac{\pi^2}{6}$.

□

Sous l'hypothèse $H_{MPD}\Gamma$le graphe $(i, lnx_i)_{i\leq n}$ est approximativement rectiligne (cf. figure 2.1 où les données ont été simulées à partir de l'hypothèse H_{MPD}).

AinsiΓsi on dispose de n observations $x_1, \ldots, x_n\Gamma$le graphe $(i, lnx_i)_{i\leq n}$ permet d'avoir une idée sur la validité de l'hypothèse H_{MPD} sur le jeux de données étudié.

Les figures 2.2Γ2.3 et 2.4 représentent ces graphiquesΓainsi que leurs lissages splinesΓpour certains jeux de données de défaillances logicielles (jeux de données extraits de [75] et [36]).

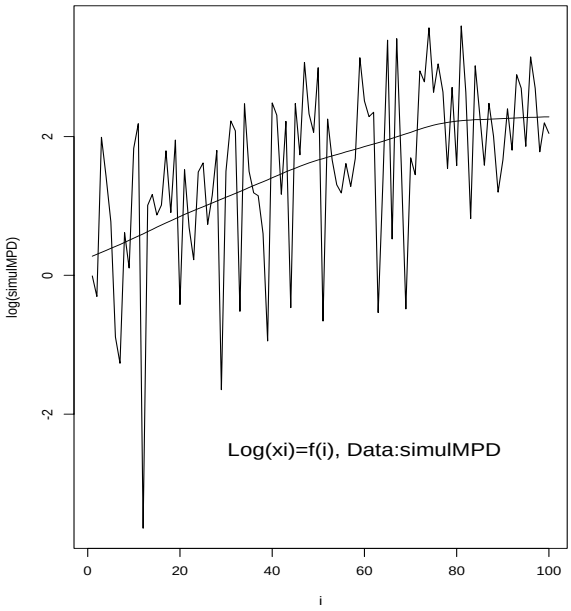

FIG. 2.1: Un jeu de données simulé à partir de H_{MPD}

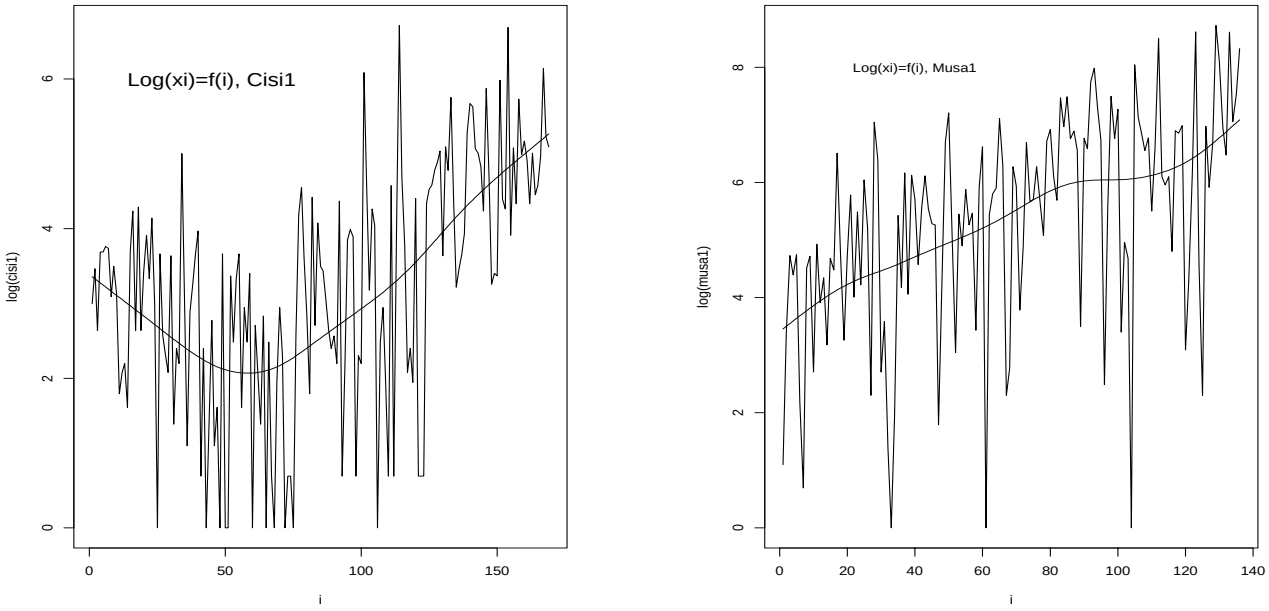

FIG. 2.2: Jeux de données : Cisi1 et Musa1

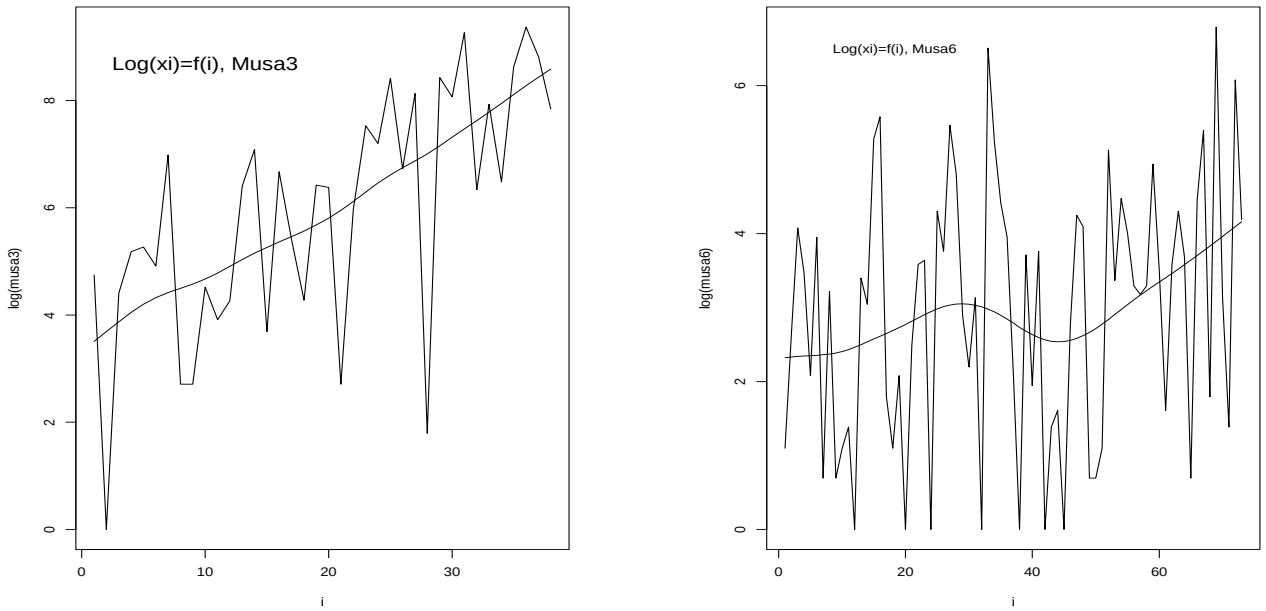

FIG. 2.3: Jeux de données : Musa3 et Musa6

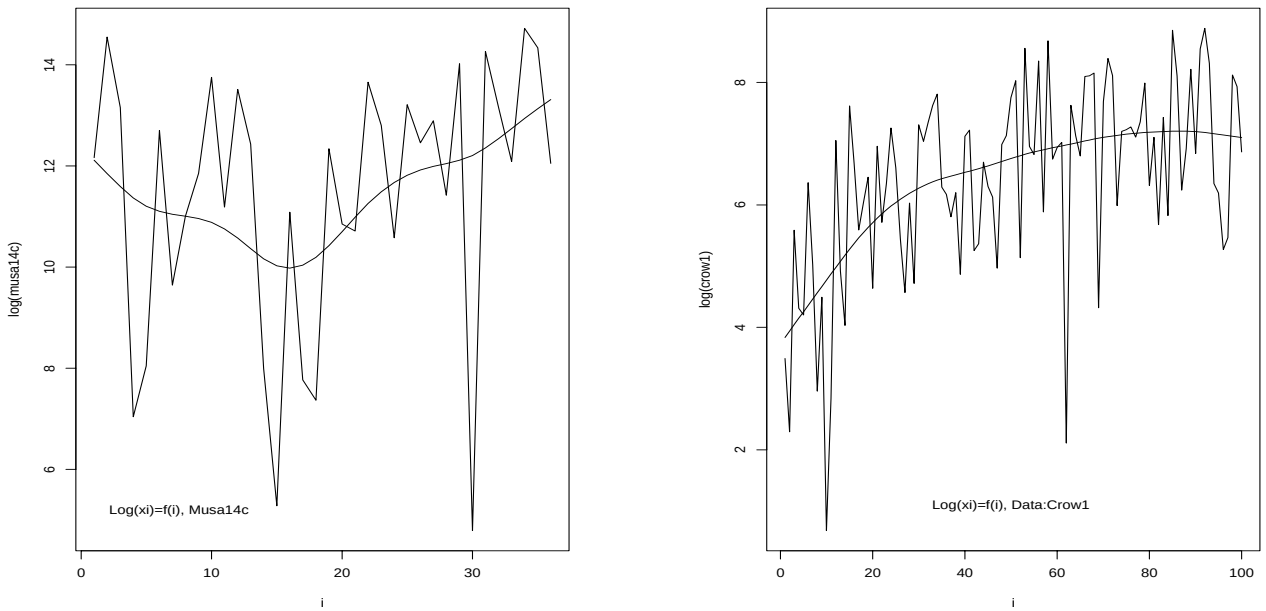

FIG. 2.4: Jeux de données : Musa14c et Crow1

Si dans certains cas (Musa1 et Musa3)Γla relation entre $(i)_{i\leq n}$ et $(lnx_i)_{i\leq n}$ est approximativement linéaireΓdans plusieurs autres cas (Cisi1Γ Musa6Γ Musa14cΓ Crow1)Γcette relation est plutôt polynômiale (voir le lissage spline de ces graphiques figures 2.2Γ2.3 et 2.4).
Dans ces derniers casΓune modélisation *ND* polynômiale :

$$\forall i \geq 1 \ \Gamma \ \ X_i \sim Exp\,[\,exp(P(i))\,]$$

où P est un polynôme de degré p (à déterminer) semble plus judicieuse.
En effet dans ce casΓon aurait : $ln(X_i) = P(i) - \gamma_E + \epsilon_i$Γce qui tient bien compte de la relation polynômiale trouvée graphiquement.

2.4.2 Les modèles *ND* polynômiaux (ND_{pol})

En tenant compte des remarques précédentesΓon se propose de généraliser les modèles *MPD* et *JM*Γou tout autre modèle *ND* linéaireΓen remplaçant l'hypothèse H_{lin} par l'hypothèse H_{pol} décrite ci-dessous.

Définition – 2.13 *On appelle* ***modèles* ND *polynômiaux*** *les modèles de fiabilité des logiciels décrits par l'hypothèse suivante :*

$$\text{“}X_1, X_2, \ldots \textit{ sont des v.a.r. indépendantes de lois :}$$

$$X_i \sim Exp\,[\,h(P(i))\,]\ \text{”}$$

$\boldsymbol{H_{pol}}$

où :

- *P est un polynôme de degré p inconnu,*
- *la fonction h est une fonction connue.*

On obtient ainsi une nouvelle famille de modèles faisant encore partie de la famille des modèles linéaires généralisés.
Pour un degré p fixéΓles vecteurs de régression sont choisis parmi les vecteurs :

$$r_0 \begin{pmatrix} 1 \\ . \\ . \\ 1 \end{pmatrix}, r_1 \begin{pmatrix} 1 \\ 2 \\ . \\ n \end{pmatrix}, .., r_p \begin{pmatrix} 1 \\ 2^p \\ . \\ n^p \end{pmatrix}.$$

Remarques –

1. On parlera de modèles ***MPD* polynômiaux** ou de modèles ***JM* polynômiaux** selon que l'on choisit $h(x) = exp(x)$ ou $h(x) = x$. On les notera respectivement MPD_{pol} et JM_{pol}.

2. La fonction h qu'on peut aussiΓpar abus de langageΓappeler fonction de lienΓest liée à la fonction de lien g des *GLM* telle que définie dans la définition – 2.1 par la relation :
$$g = [\frac{1}{h}]^{-1}$$
où $[\frac{1}{h}]^{-1}$ est la fonction réciproque de la fonction $\frac{1}{h}$.

La modélisation *ND* polynômiale peut être décomposée en trois principales étapes :

1. Choix de la fonction de lien hΓ
2. choix du degré et de la forme du polynôme utiliséΓ
3. estimation des paramètres du modèleΓi.e. des coefficients du polynôme P.

Les deux premières étapes permettent de choisir le modèle approprié.
Dans la troisième étapeΓles coefficients du polynôme considéré sont estimés par la méthode du maximum de vraisemblance. Les propriétés asymptotiques des estimateurs (cf. théorème 2.5) citées auparavant restent encore valables puisque les modèles *ND* polynômiaux sont des modèles linéaires généralisés.

On décrit ci-dessous les deux premières étapes qui permettent de choisir pour chaque jeu de données le modèle le plus approprié dans la famille des modèles *ND* polynômiaux.

2.4.3 Choix des polynômes appropriés

On suppose dans cette sous-section que la fonction de lien h a déjà été choisieΓle problème du choix de cette fonction sera discuté dans la sous-section suivante.

Il s'agit alors de choisir le degré et la forme du polynôme P de l'hypothèse H_{pol}. Ceci revient à estimer les entiers :
$$0 \leq i_1 \leq i_2 \ldots \leq i_q = p$$
tels que le polynôme P s'écrive sous la forme :
$$P(x) = \beta_{i_1} x^{i_1} + \beta_{i_2} x^{i_2} + \ldots + \beta_p x^p.$$

Pour déterminer le degré du polynôme P on peut s'aiderΓdans le cas du *MPD*Γde la procédure graphique décrite dans la sous-section 2.4.1 (cf. l'exemple d'application présenté ci-dessous).

MaisΓplus généralementΓon utilise dans le cas où les conditions du théorème 2.5 sont vérifiéesΓle test du rapport de vraisemblances maximalesΓdécrit dans la sous-section 2.2.5.

La procédure du choix du polynôme P peut alors se décomposer en trois étapes :

1. Choisir un degré p_0 assez élevé ($p_0 = 5$ par exemple).
   Ajuster les observations par un modèle *ND* polynômial de degré p_0.

   Ce premier modèle a une déviance :

   $$Dev(x^{(n)}, M_{p_0}).$$

2. Trouver l'entier $k \leq p_0$ tel que le test du rapport de vraisemblances maximales :
   - rejette l'hypothèse "$p = k-1$" contre l'hypothèse "$p = p_0$"
   - mais ne rejette pas l'hypothèse "$p = k$" contre l'hypothèse "$p = p_0$".
   On décide alors qu'un polynôme de degré k suffit à décrire convenablement les données.

3. Les estimateurs de maximum de vraisemblance des coefficients du polynôme P $\hat{\beta}_0, \hat{\beta}_1, .., \hat{\beta}_k \Gamma$sont asymptotiquement gaussiensΓon peut alors utiliser le test du rapport de vraisemblances maximales ou le test de *Student* pour tester les hypothèses de nullité "$\hat{\beta}_j = 0$"Γpour $j = 1, .., k-1$.

On présente ci-dessous un exemple de l'application de cette procédure de choix de modèles sur un jeu de données simulé.

Illustration

On simule des données $(x_i)_{i \leq 100}$ provenant de variables :

$$X_i \sim Exp\,[\,exp(P_{sim}(i))\,],$$

où

$$P_{sim}(i) = \beta_0 + \beta_1 i + \beta_2 i^2 \text{ avec } \beta_0 = -7\ \Gamma\beta_1 = 0.3 \text{ et } c = -0.004.$$

Le jeu de données obtenu est désigné par *simpoly1*.

Supposons maintenant qu'on ne connaisse pas le polynôme associé à ce jeu de données. On va appliquer la procédure décrite ci-dessus pour essayer de retrouver le polynôme optimal associé.

On choisit d'abord une fonction de lien $h(x) = exp(x)$ fonction de lien du *MPD*.

Etape 1 :
On commence par ajuster les données par un modèle *MPD* polynômial de degré $p_0 = 5\Gamma$ on obtient :

$$Dev(x^{(n)}, M_5) = 106.28$$

Etape 2 :

a) Le graphe $(i, lnx_i)_{i\leq n}$ a la forme d'un polynôme de degré 3.

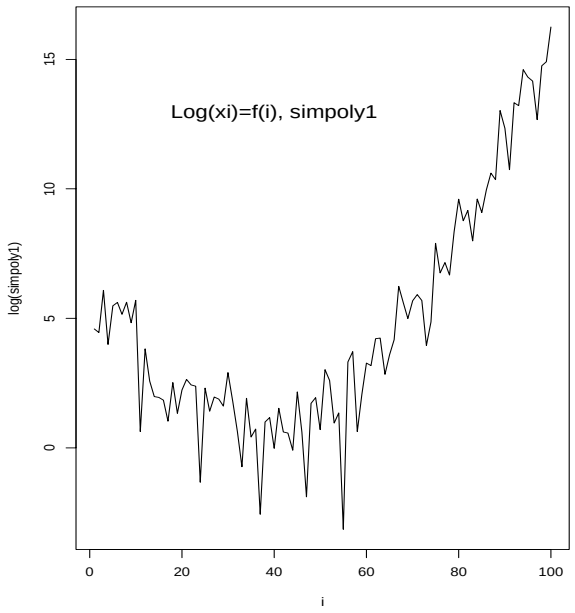

FIG. 2.5: Le graphe $(i, lnx_i)_{i\leq n}$ pour *simpoly1*

On teste alors l'hypothèse :

$$\text{“}p = 3\text{” contre l'hypothèse “}p = 5\text{”}$$

$$\text{On a } Dev(x^{(n)}, M_3) = 108.79$$

Comme :

$$Dev(x^{(n)}, M_3) - Dev(x^{(n)}, M_5) = 2.51 < F^{-1}_{\chi^2_2}(0.95) = 5.99$$

on ne rejette pas l'hypothèse “$p = 3$”.

b) On teste ensuite l'hypothèse :

$$\text{“}p = 2\text{” contre l'hypothèse “}p = 5\text{”}$$

$$\text{On a } Dev(x^{(n)}, M_2) = 108.96$$

Comme :

$$Dev(x^{(n)}, M_2) - Dev(x^{(n)}, M_5) = 2.68 < F^{-1}_{\chi^2_3}(0.95) = 7.81$$

on ne rejette pas l'hypothèse “$p = 2$”.

j	0	1	2
β_j	-7	0.3	-0.004
$\hat{\beta}_j$	-6.66	0.287	-0.0038

TAB. 2.1: Estimations des paramètres du Modèle MPD_{pol} (*simpoly1*).

c) On teste enfin l'hypothèse :

"$p = 1$ (MPD)" contre l'hypothèse "$p = 5$"

$$\text{On a } Dev(x^{(n)}, M_1) = 153.41$$

$$Dev(x^{(n)}, M_1) - Dev(x^{(n)}, M_5) = 47.13 > F^{-1}_{\chi^2_4}(0.95) = 9.48$$

L'hypothèse "$p = 1$" est ainsi rejetée contre l'hypothèse "$p = 5$".

Finalement l'entier recherché est $\boldsymbol{k = 2}$ et la forme retenue du polynôme P est bien celle du polynôme à partir duquel on a simulé les données :

$$P(i) = \beta_0 + \beta_1 i + \beta_2 i^2.$$

Etape 3 :
Le même type de tests que ceux utilisés dans l'étape 2 permettent ensuite de rejeter les hypothèses "$\beta_j = 0$" contre "$\beta_j \neq 0$"Γpour $j = 0, 1$.

Les algorithmes classiques de calcul des estimations de maximum de vraisemblance dans les modèles linéaires généralisés permettent alors d'estimer les paramètres β_0Γβ_1Γet β_2. Les estimations obtenues $\hat{\beta}_j$ sont comparéesΓau tableau 2.1Γaux vraies valeurs β_j utilisées pour simuler les données.

Remarque – Les résultats expérimentaux qui seront présentés dans la section 2.4.5 montrent que pour la plupart des jeux de données pour lesquels le degré p choisi est différent de 1Γles qualités prévisionnelles (mesurées par le critère du *u-plot*) du modèle *MPD* polynômial choisi sont nettement supérieures à celles du *MPD* classique.

2.4.4 Choix de la fonction de lien

Les modèles linéaires généralisés étudiés jusque là avaient des fonctions de lien connues :

$$g(x) = ln(x) \text{ pour le } MPD \text{ et } g(x) = \frac{1}{x} \text{ pour le } JM.$$

Ces fonctions de lien découlent généralement de la modélisation du phénomène physique étudié.

En Fiabilité des Logiciels par exemple, les fonctions de lien traduisent certaines hypothèses décrivant le logiciel étudié ou les procédés de correction de ses fautes.

On peut cependant se placer dans un cadre purement analytique et estimer la fonction de lien en utilisant les observations $(x_i)_{i\leq n}$.

La fonction de lien peut ainsi être estimée par des méthodes paramétriques (cf. Scallan et al [87]), par des méthodes semi-paramétriques (cf. Bonneu et al [10]) ou encore par des méthodes non paramétriques (cf. Hastie et Tibshirani [44]).

Pour les modèles *ND* où on n'a qu'un seul régresseur, l'approche *GLM* non paramétrique permet d'estimer l'effet composé de la fonction de lien et du régresseur.
Cette approche *GLM* non paramétrique sera utilisée dans la section suivante pour présenter un modèle *ND* non paramétrique généralisant tous les modèles *ND* paramétriques.

2.4.5 Résultats expérimentaux

On utilise ici la procédure décrite dans la section 2.4.3 pour choisir les modèles *MPD* et *JM* polynômiaux appropriés pour un certain nombre de jeux de données réels (cf. par exemple [75]). Les jeux de données considérés sont *Cisi1* (169 observations), *Musa1* (136), *Musa3* (38), *Musa6* (73), *Musa14c* (36) et *Crow1* (100) (cf. annexe B). Le dernier jeu de données a été simulé à partir du modèle de Crow ($\alpha=0.15$ et $\beta=0.5$).

Dans le tableau 2.2, on donne pour chaque jeu de données, les estimations des paramètres p et $(\beta_j)_{j\leq p}$ du modèle *MPD* polynômial choisi.

	$\hat{p}$	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$
Cisi1	3	-4.03	0.059	-0.00084	2.610^{-6}
Musa1	1	-4.53	-0.023		
Musa3	1	-4.44	-0.11		
Musa6	3	-2.16	-0.22	0.0069	-6.310^{-5}
Musa14c	2	-13.67	0.19	-0.059	
Crow1	2	-5.04	-0.066	0.00041	

TAB. 2.2: Estimations des paramètres des modèles MPD_{pol}

Dans le cas des jeux de données *Musa1* et *Musa3*, le polynôme choisi est de degré 1, le modèle polynômial coïncide donc avec le *MPD*, ce qui confirme que dans le cas de ces deux jeux de données, le modèle *MPD* est particulièrement adéquat.

Amélioration de la qualité d'ajustement

Dans le tableau 2.3 on présente les déviances des modèles MPD_{pol} et JM_{pol} qu'on compare à celles des modèles MPD et JM.

	MPD	$MPD_{pol}(p)$	JM	$JM_{pol}(p)$
Cisi1	253.74	227.43(3)	247.72	232.15(3)
Musa1	213.47	213.47(1)	228.01	212.02(3)
Musa3	61.80	61.80(1)	76.14	61.26(2)
Musa6	180.16	165.83(3)	179.50	160.32(5)
Musa14c	111.89	99.72(2)	110.42	99.57(2)
Crow1	123.28	115.89(2)	133.12	120.24(2)

TAB. 2.3: Déviances des différents modèles et valeurs choisies du paramètre p

Il ressort des résultats précédents que la généralisation polynômiale des modèles JM et MPD améliore leur adéquation aux jeux de données étudiés.
Ceci découle de la construction même des modèles polynômiaux puisqu'on les choisit par des tests sur la déviance. Ces tests permettent de faire un compromis entre le critère de déviance (qualité d'ajustement) et le critère de degré de liberté ou robustesse (nombre de paramètres).

On remarque par ailleurs que le modèle MPD_{pol} fournit des déviances inférieures ou comparables à celles du modèle JM_{pol} tout en ayant moins de paramètres.
Cette robustesse du modèle MPD_{pol} se traduit comme on le verra ci-dessous par un meilleur pouvoir prédictif.

Amélioration de la qualité prévisionnelle

En Fiabilité des Logiciels les modèles sont choisis en fonction de leur qualité prévisionnelle. Ce critère étudié au chapitre 4 mesure la capacité du modèle à bien prédire les observations futures.

Remarques –

1. La qualité prévisionnelle d'un modèle est mesurée ici par le **critère du *u-plot*** qui sera étudié au chapitre 4.
   La qualité prévisionnelle d'un modèle est d'autant plus grande que son critère *u-plot* est faible.

2. La procédure $u-plot$ a été implémentée sur les $n-20$ dernières données c'est-à-dire qu'à la première étape on utilise les observations $x_1, \ldots, x_{20}$ pour prédire l'observation x_{21}.

Les modèles MPD_{pol} et JM_{pol} ont généralement un nombre de paramètres supérieur à celui des modèles originaux MPD et JM. Ceci peut suggérer que les modèles MPD_{pol} et JM_{pol} sont moins robustes et ont donc de moins bonnes qualités prévisionnelles que les modèles MPD et JM.

Ceci est démenti par les résultats du tableau 2.4 donnant les critères *u-plot* de l'utilisation des différents modèles sur les six jeux de données étudiés.

	MPD	MPD_{pol}	JM_{pol}	JM
Cisi1	0.110	**0.104**	0.181	0.736
Musa1	0.111	**0.111**	0.164	0.539
Musa3	0.218	**0.218**	0.229	
Musa6	**0.222**	0.223	0.385	0.417
Musa14c	0.396	**0.158**	0.297	0.333
Crow1	**0.161**	0.174	0.203	0.840

TAB. 2.4: Critère du *u-plot*

Il ressort donc que le modèle MPD_{pol} aΓpour pratiquement tous les jeux de donnéesΓdes performances prédictives meilleures que celles du MPD.
D'autre partΓle modèle JM_{pol} améliore nettement les performances du modèle JM.
Le modèle MPD_{pol} aΓpour tous les jeux de données étudiésΓde meilleures performances prédictives que le modèle JM_{pol}.

On conclut ainsi qu'aussi bien au niveau de la qualité de l'ajustement qu'au niveau de la qualité prévisionnelleΓles modèles polynômiaux MPD_{pol} et JM_{pol} apportent une nette amélioration par rapport aux modèles originaux MPD et JM.

Le modèle MPD_{pol} semble fournir le meilleur compromis entre le critère de bonne adéquation aux données et le critère de robustesse. Ce bon compromis se traduit par de très bonnes qualités prévisionnelles.

2.5 Généralisation non paramétrique des modèles *ND*

Rappelons d'abord la propriété définissant les modèles *ND* :

$$\text{“}X_1, X_2, \ldots \text{ sont des v.a.r. indépendantes de lois :}$$

$$\forall i > 0 \ \Gamma X_i \sim Exp\,(\psi(i))\text{.”}$$

$$\boldsymbol{H_{ND}}$$

Les modèles *ND* rencontrés jusque là proposent tous une modélisation paramétrique de la fonction taux de défaillance ψ :

$$\psi(i) = \psi_0(i, \theta)$$

où :

- ψ_0 est une fonction connue à valeur dans $\mathbb{R}_+$
- θ un paramètre vectoriel à estimer.

Une généralisation naturelle de ces modèles *ND* paramétriques est obtenue en estimant ψ par des techniques non paramétriques.

Le modèle non paramétrique ainsi obtenu n'aura de bonnes qualités prévisionnelles que si l'on ajoute certaines conditions de régularité sur la fonction ψ.
On supposera dans la suite que ψ est une fonction réelle deux fois continûment différentiable.

Notation – $\boldsymbol{S_2}$ désignera dans la suite l'espace des fonctions réelles deux fois continûment différentiables.

Le modèle *ND* non paramétriqueΓqui sera présenté dans la sous-section 2.5.3Γfait partie de la famille des modèles linéaires généralisés non paramétriques.

On présente succinctement l'approche *GLM* non paramétriques dans la sous-section 2.5.2. Des présentations plus détaillées peuvent être trouvées dans Fahrmeir et Tutz [33] ou Green et Silverman [43].

On utilisera ensuite les méthodes et les algorithmes développés dans le cadre général des *GLM* non paramétriques pour obtenir des estimations non paramétriques de la fonction ψ. Ces estimations seront obtenues sous forme de splines cubiques.

Avant de décrire les *GLM* non paramétriques et d'étudier le modèle *ND* non paramétriqueΓ on rappelle ci-dessous brièvement quelques résultats concernant les splines cubiques.

2.5.1 Quelques rappels sur les splines cubiques

On suppose dans cette sous-section qu'on a n réels ordonnés $t_1, t_2, \ldots, t_n$ sur un intervalle $[a, b]$:

$$a < t_1 < t_2 < \ldots < t_n < b$$

On suppose en plus qu'on a des observations y_i bruitées d'une fonction inconnue f :

$$\forall i \leq n \ \Gamma y_i = f(t_i) + \epsilon_i.$$

Les splines cubiques jouent un rôle important dans l'obtention d'estimations lisses de la fonction f.

Définitions et propriétés

Définition – 2.14 *Une fonction s définie sur $[a,b]$ est une* ***spline cubique*** *de nœuds $(t_i)_{i\leq n}$ si les deux propriétés suivantes sont vérifiées :*

1. *sur chacun des intervalles $[a,t_1]$, $[t_1,t_2]$, ... , $[t_n,b]$ s est un polynôme cubique.*
2. *s est deux fois continûment différentiable sur $[a,b]$, c'est à dire que les fonctions s, s' et s'' sont continues aux nœuds t_i.*

Une spline cubique est dite ***naturelle*** *si ses dérivées secondes et troisièmes sont nulles aux points a et b.*

Remarque – Même si une spline cubique s est complètement spécifiée par le vecteur :

$$\underline{s} = (s(t_i))_{i\leq n}$$

il est plus pratique d'avoir aussi le vecteur $\gamma \in \mathbb{R}^{n-2}$ des dérivées secondes de s aux nœuds t_i :

$$\forall i \in [2,\ldots,n-1] \ \Gamma \ \gamma_i = s''(t_i).$$

Théorème – 2.15 *Deux vecteurs quelconques $\underline{s} \in \mathbb{R}^n$ et $\gamma \in \mathbb{R}^{n-2}$ définissent une spline cubique si et seulement si ils vérifient la relation suivante :*

$$^tQ\,\underline{s} \ = \ M\gamma$$

où :

- *$Q = (q_{ij})_{i\leq n, 2\leq j\leq n-1}$ est la matrice tri-diagonale $n\times(n-2)$ définie par ses composantes :*

$$q_{j-1,j} = h_{j-1}^{-1} \ , \ q_{jj} = -h_{j-1}^{-1} - h_j^{-1} \ \text{et} \ q_{j+1,j} = h_j^{-1}$$
$$q_{ij} = 0 \ \text{pour} \ \mid i-j \mid \geq 2.$$

- *$h_i = t_{i+1} - t_i$ pour $i = 1,\ldots,n-1$.*

- $M = (m_{ij})_{2 \leq i,j \leq n-1}$ *est la matrice tri-diagonale symétrique* $(n-2) \times (n-2)$ *définie par ses composantes :*

$$m_{ii} = \frac{1}{3}(h_{i-1} + h_i) \ , \ i = 2, \ldots, n-1$$

$$m_{i,i+1} = m_{i+1,i} = \frac{1}{6} h_i \ , \ i = 2, \ldots, n-2$$

$$m_{ij} = 0 \ pour \ \mid i - j \mid \geq 2.$$

Théorème – 2.16 *Si s est une spline cubique de nœuds $(t_i)_{i \leq n}$ spécifiée par le vecteur de ses valeurs $\underline{s}$, et le vecteur de ses dérivées secondes γ, on a alors avec les notations du théorème précédent :*

$$\int_a^b [\, s''(t) \,]^2 \, dt \ = \ {}^t\underline{s} \, K \, \underline{s}$$

où :

$$K = Q \, M^{-1} \, {}^t Q.$$

On pourra se référer à Green et Silverman [43] (page 13) pour les preuves des deux théorèmes précédents.

Splines cubiques et lissage

Soient $y_1, \ldots, y_n$ des observations bruitées d'une fonction f inconnue :

$$\forall i \leq n \ \Gamma y_i = f(t_i) + \epsilon_i.$$

On suppose que f est une fonction de l'espace S_2 des fonctions deux fois continûment différentiables.

L'estimation de f par une fonction $\hat{f}$ suffisamment lisse se fait généralement par la minimisationΓsur S_2Γde la somme pénalisée des carrés des résidus :

$$S(f) = \sum_{i=1}^{n} [\, y_i - f(t_i) \,]^2 + \delta \int_a^b [\, f''(t) \,]^2 \, dt.$$

Le terme de pénalité $\delta \int_a^b [\, f''(t) \,]^2 \, dt$ assure un certain degré de lissitude à l'estimation $\hat{f}$. Le paramètre de lissage δ représente l'importance relative de la contrainte de lissitude par rapport au critère d'adéquation aux observations.

le rôle important des splines cubiques dans les problèmes de lissage résulte de la proposition suivante :

Proposition – 2.17 *Si s est une spline cubique de nœuds $(t_i)_{i\leq n}$ et si $\tilde{f}$ est une fonction de S_2 telle que :*

$$\forall i \leq n \ , \ \tilde{f}(t_i) = s(t_i),$$

Alors on a :

$$\int_a^b [\ \tilde{f}''(t)\ ]^2\, dt \ \geq \int_a^b [\ s''(t)\ ]^2\, dt$$

Corollaire – La minimisation de $S(f)$ sur S_2 se réduit ainsi à sa minimisation sur l'espace des fonctions splines cubiques.

L'estimation de la fonction inconnue f se ramène alorsΓgrâce au théorème 2.16 à la recherche du vecteur $\underline{\hat{f}} = (\hat{f}(t_i))_{i\leq n}$ de $\mathbb{R}^n$ minimisant la quantité :

$$S(v) = \sum_{i=1}^{n} [\ y_i - v_i\ ]^2 + \delta \ {}^t v\, K v \ \text{ pour } v \in \mathbb{R}^n.$$

2.5.2 Les *GLM* non paramétriques

On présente ci-dessous l'approche modèles linéaires généralisés non paramétriques dans le cas où il n'y a qu'une seule variable explicative $r \in \mathbb{R}^n$.

Les modèles additifs généralisés [44] permettent par ailleurs de donner une généralisation non paramétrique des modèles linéaires généralisés dans le cas de plusieurs régresseurs.

On reprend dans cette sous-section les notations de la section 2.2.

Définition – 2.18 ***Les modèles linéaires généralisés non paramétriques*** *sont obtenus à partir des modèles linéaires généralisés tels que définis dans la section 2.2, en remplaçant la relation paramétrique :*

$$\forall i \leq n \ , \ (g\, o\, b')(\alpha_i) = \sum_{j=1}^{p} \beta_j r_{j,i}$$

par la relation non paramétrique :

$$\forall i \leq n \ , \ (g\, o\, b')(\alpha_i) = f(r_i)$$

où f est une fonction inconnue suffisamment lisse.

Remarque – Dans le cas où le choix de la fonction de lien g ne réduit pas l'ensemble des valeurs prises par la fonction $(g\, o\, b')^{-1}$Γil est possible de se ramener au modèle à lien canonique :

$$\forall i \leq n\ \Gamma \alpha_i = f_c(r_i)$$

on estimera alors directement la fonction f_c représentant l'effet composé de la fonction de lien et de la fonction de régression :

$$f_c = (g\,o\,b')^{-1}\,o\,f.$$

L'estimation non paramétrique de la fonction de régression f se fait par la méthode de la vraisemblance pénalisée brièvement décrite ci-dessous.

On se placera dans la suite de cette sous-section dans le cas du lien canonique. Le modèle considéré est alors décrit par les relations :

$$\forall i \leq n\ \Gamma\alpha_i = f(r_i).$$

Les algorithmes présentés sont facilement généralisables au cas d'un lien non canonique.

Méthode de la vraisemblance pénalisée

La fonction de vraisemblance associée aux *GLM* non paramétriques définis ci-dessus et aux observations $(x_i)_{i\leq n}$ est :

$$L_{x_1,..,x_n}(f) = \prod_{i=1}^{n} exp\,[\,\frac{f(r_i)x_i - b(f(r_i))}{a(\phi)} + c(x_i,\phi)\,].$$

La log-vraisemblance associée est :

$$\mathcal{L}(f) = \sum_{i=1}^{n} \frac{f(r_i)x_i - b(f(r_i))}{a(\phi)} + c(x_i,\phi)$$

Maximiser la quantité $\mathcal{L}(f)$ sur l'ensemble des fonctions $f \in S_2$ revient donc à maximiser la quantité :

$$\sum_{i=1}^{n}[\,f(r_i)x_i - b(f(r_i))\,]$$

Pour introduire la contrainte de lissitude de $f\Gamma$on introduit dans la quantité à maximiser le terme de pénalité :

$$\int [\,f''(t)\,]^2\,dt$$

pénalisant les estimations à courbures élevées.

La fonction f est alors estimée par la fonction $\hat{f}$ de S_2 maximisant la log-vraisemblance pénalisée :

$$S(f) = \sum_{i=1}^{n}[\,f(r_i)x_i - b(f(r_i))\,] - \frac{1}{2}\,\delta\,\int [\,f''(t)\,]^2\,dt.$$

Le paramètre de lissage δ représente l'importance donnée à la contrainte de lissitude.

Comme dans le cas du problème d'interpolationΓpour maximiser la quantité $S(f)$ sur l'ensemble S_2 des fonctions deux fois continûment différentiablesΓil suffit de la maximiser sur l'ensemble des fonctions splines cubiques. On a plus précisément :

Proposition – 2.19 *Pour toute fonction $\tilde{f} \in S_2$, il existe une spline cubique s ayant les mêmes valeurs que $\tilde{f}$ aux nœuds $(t_i)_{i\leq n}$ telle que :*

$$S(s) \geq S(\tilde{f}).$$

On peut se référer à [43] page 99 pour une preuve de cette proposition.

La recherche numérique de la spline cubique $\hat{f}$ maximisant $S(f)$ se faitΓpar exemple par une généralisation de la méthodes des scores de Fisher.
Avant de décrire cette méthodeΓon présente brièvement les différentes méthodes du choix du paramètre de lissage.

Paramètre de lissage et degré de liberté

Le paramètre de lissage δ représente le degré de lissitude imposée à l'estimation spline cubique $\hat{f}$ de f.
La valeur de δ est choisie en tenant compte des connaissances a priori concernant la régularité de la fonction f.
En l'absence de telles connaissancesΓle paramètre de lissage peut être estimé à partir des données observées en utilisant les méthodes de validation croisée (cf. [80]).

Le paramètre de lissage est lié à un deuxième paramètre : **le degré de liberté** obtenu comme généralisation de la notion de nombre de paramètres dans le cas paramétrique.
On pourra se référer à [43] page 110 pour une définition rigoureuse du degré de liberté dans le cadre des *GLM* non paramétriques.

En pratiqueΓla contrainte de lissitude de l'estimation $\hat{f}$ est spécifiée indifféremment par le paramètre de lissage ou par le degré de liberté.

Notation – Dans la suiteΓle paramètre degré de liberté sera noté ***dl***.

Méthodes des scores de Fisher pour les *GLM* non paramétriques

On cherche la spline cubique $\hat{f}$ de nœuds $(r_i)_{i\leq n}$ qui maximise la log-vraisemblance pénalisée :

$$S(f) = \sum_{i=1}^{n} [\, f(r_i)x_i - b(f(r_i))\,] - \frac{1}{2}\,\delta \int [\, f''(t)\,]^2\, dt.$$

En utilisant les notations de la sous-section 2.5.1 et le théorème 2.16 on a :

$$\int [\, f''(t)\,]^2\, dt \;=\; {}^t\underline{f}\, K\, \underline{f}$$

où :

- la matrice K est définie comme dans le théorème 2.16Γ

- le vecteur $\underline{f} \in \mathbb{R}^n$ est donné par ses composantes :

$$\underline{f}_i = f(r_i).$$

L'estimation non paramétrique de la fonction de régression f revient donc à la recherche du vecteur $\underline{\hat{f}} \in \mathbb{R}^n$ maximisant la quantité :

$$\sum_{i=1}^{n} [\, v_i x_i - b(v_i) \,] - \frac{1}{2} \delta \ {}^t v K v \quad \text{pour } v \in \mathbb{R}^n.$$

Proposition – 2.20 *L'algorithme des scores de Fisher permettant de trouver la spline cubique $\hat{f}$ maximisant $S(f)$ est décrit par le schéma itératif présenté ci-dessous.*
Ce schéma permet, à la $j^{ème}$ itération, de passer de la spline cubique $s^{(j)}$ représentée par $\underline{s}^{(j)} \in \mathbb{R}^n$ à la spline cubique $s^{(j+1)}$ représentée par $\underline{s}^{(j+1)} \in \mathbb{R}^n$:

$$\underline{s}^{(j+1)} = (V^{(j)} + \delta K)^{-1} V^{(j)} z^{(j)}$$

où à chaque itération j on a :

- *$z^{(j)}$ est le vecteur de $\mathbb{R}^n$ donné par ses composantes :*

$$z_i^{(j)} = s_i^{(j)} + \frac{x_i - b'(s_i^{(j)})}{b''(s_i^{(j)})}$$

- *$V^{(j)}$ est la matrice diagonale définie par :*

$$V_{ii}^{(j)} = b''(s_i^{(j)})$$

.

Une preuve détaillée de ce résultat peut être trouvée dans [43] page 100.

Remarque – Les *GLM* non paramétriques peuvent être considérés comme des cas particuliers des modèles additifs généralisés (*GAM*) (cf. Hastie et Tibshirani [44]). Dans les exemples d'application traités dans la suiteΓl'estimation non paramétrique de la fonction de régression f est faite par la procédure *gam* du logiciel ***S*** [15].

2.5.3 Les modèles *ND* non paramétriques (ND_{np})

Définition – 2.21 *On appelle modèles* **ND *non paramétriques*** *les modèles de fiabilité des logiciels décrits par l'hypothèse suivante :*

$$\text{“}X_1, X_2, \ldots \text{ sont des v.a.r. indépendantes de lois :}$$

$$\forall i > 0 \ , \ X_i \sim Exp\,(\psi(i))\text{.”}$$

$\boldsymbol{H_{NDnp}}$

Où ψ est une fonction inconnue deux fois continûment différentiable à valeurs dans $\mathbb{R}_+$.

Les modèles *ND* non paramétriques représentent une généralisation de tous les modèles *ND* paramétriques. En effet la seule hypothèse ajoutée par rapport à H_{ND} est une régularité minimale de la fonction taux de défaillance ψ.

Les modèles ND_{np} vus comme des *GLM* non paramétriques

L'estimation non paramétrique de ψ peut se faire dans le cadre des *GLM* non paramétriques.
En effet Les modèles *ND* non paramétriques peuvent être considérés comme des *GLM* puisque d'après l'hypothèse H_{NDnp} les paramètres naturels des lois des X_i s'écrivent :

$$\alpha_i = -\psi(i)$$

où f_λ est supposée être une fonction assez lisse.

Ceci définit un *GLM* non paramétrique de régresseur $(i)_{i\leq n}$ et de fonction de lien canonique. Les algorithmes présentés dans le cadre des *GLM* non paramétriques permettent ensuite d'estimer directement la fonction taux de défaillance ψ sous forme d'une spline cubique.

Choix du lien logarithmique

En choisissant le lien canonique on ne tient pas compte de la contrainte de positivité de la fonction ψ.
Pour ce faire on peut écrire ψ sous la forme :

$$\psi = exp(\xi)$$

et estimer la fonction ξ au lieu de ψ.

Le modèle non paramétrique associé s'écrit alors :

$$\alpha_i = -exp(\xi(i))$$

ou encore :

$$(gob')(\alpha_i) = \xi(i)$$

où $g(x) = -ln(x)$Γest la fonction de lien logarithmique (celle du *MPD*).

Si $\hat{\xi}$ est la spline cubique estimation de ξΓla fonction taux de défaillance ψ est alors estimée par :

$$\psi \simeq exp(\hat{\xi}).$$

En utilisant les modèles ND_{np} sur différents jeux de donnéesΓon remarque que les estimations de ψ obtenues par le lien canonique et le lien logarithmique sont toujours très proches. Quelques exemples sont présentés sur les figures 2.6 et 2.7.

Ceci est confirmé par le tableau 2.5 donnant les déviances des modèles ND_{np} sur les différents jeux de données.

	Cisi1	*Musa6*	*simpoly2.d*	*simsin1.d*
Lien canonique	227.72	160.62	116.24	137.44
Lien logarithmique	221.30	160.18	116.89	137.81

TAB. 2.5: Effet de la fonction de lien sur les déviances des modèles ND_{np}.

Remarques –

1. Dans tous les modèles ND_{np} présentés iciΓet sauf mention du contraireΓle paramètre de lissage est choisi de telle sorte que le nombre de degrés de liberté (dl) vaille 5.

2. Les jeux de données simpoly2.d et simsin1.d ont été simulés à partir de fonctions ψ^{poly2} et ψ^{sin1} connues :

$$\psi^{poly2}(i) = 10.5 - 0.6i + 0.016i^2 - 0.0001i^3 \text{ et } \psi^{sin1}(i) = \mid sin(\frac{i\pi}{50}) \mid +0.5.$$

3. On n'utilisera dans la suite que le modèle ND_{np} avec le lien logarithmique.

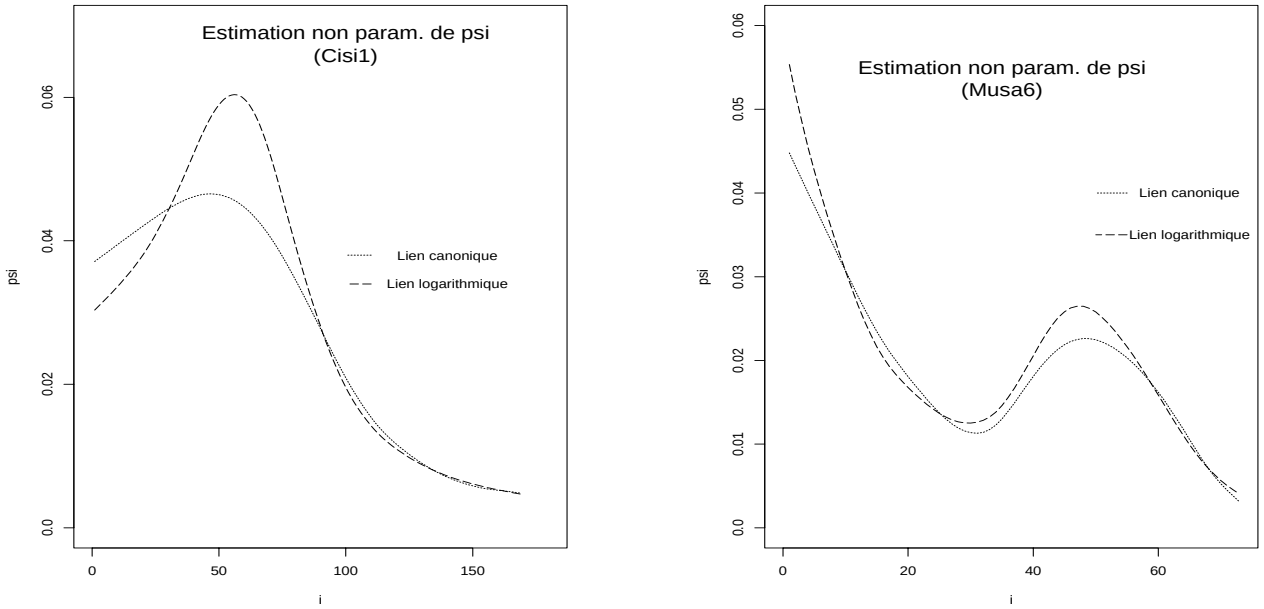

FIG. 2.6: Etude de l'effet de la fonction de lien sur l'estimation non paramétrique de ψ. Jeux de données réels : Cisi1 et Musa6.

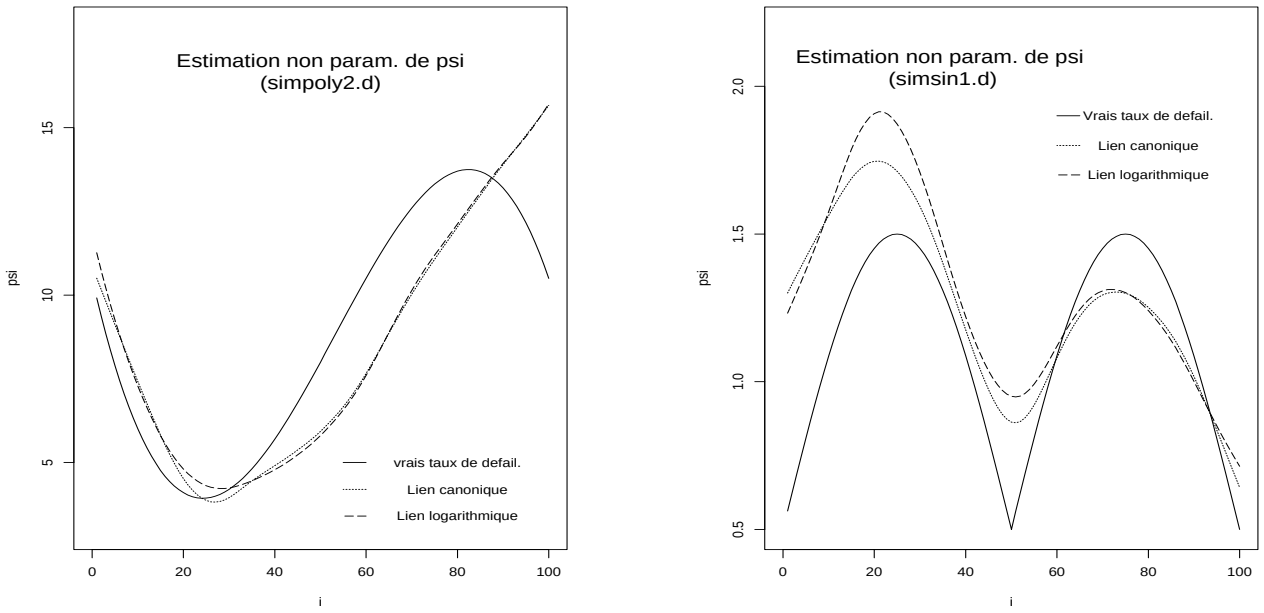

FIG. 2.7: Etude de l'effet de la fonction de lien sur l'estimation non paramétrique de ψ. Jeux de données simulés : simpoly2.d et simsin1.d.

2.5.4 Résultats expérimentaux

Qualités empiriques des estimateurs non paramétriques

Pour étudier empiriquement les qualités des estimateurs non paramétriques de $\psi\Gamma$ on simule un certain nombre de jeux de données à partir de l'hypothèse H_{NDnp} où la fonction taux de défaillance est une fonction connue (on prendra ψ^{poly2} et ψ^{sin1}) . On utilise ensuite l'approche ND_{np} pour obtenir pour chaque jeu de données une estimation particulière de ψ.
On étudie alors la variance de ces différentes estimations et on compare leur moyenne empirique à la vraie fonction ψ à partir de la quelle on a simulé les jeux de données.

On a donc simulé 50 jeux de données à partir de la fonction ψ^{poly2} et 50 autres jeux de données à partir de ψ^{sin1}.

Pour avoir une idée du biais de l'estimateur non paramétriqueΓon trace sur la figure 2.8 la moyenne empirique des 50 estimations non paramétriques de ψ^{poly2} et ψ^{sin1}.
On remarque alors que ces biais empiriques sont assez faibles.

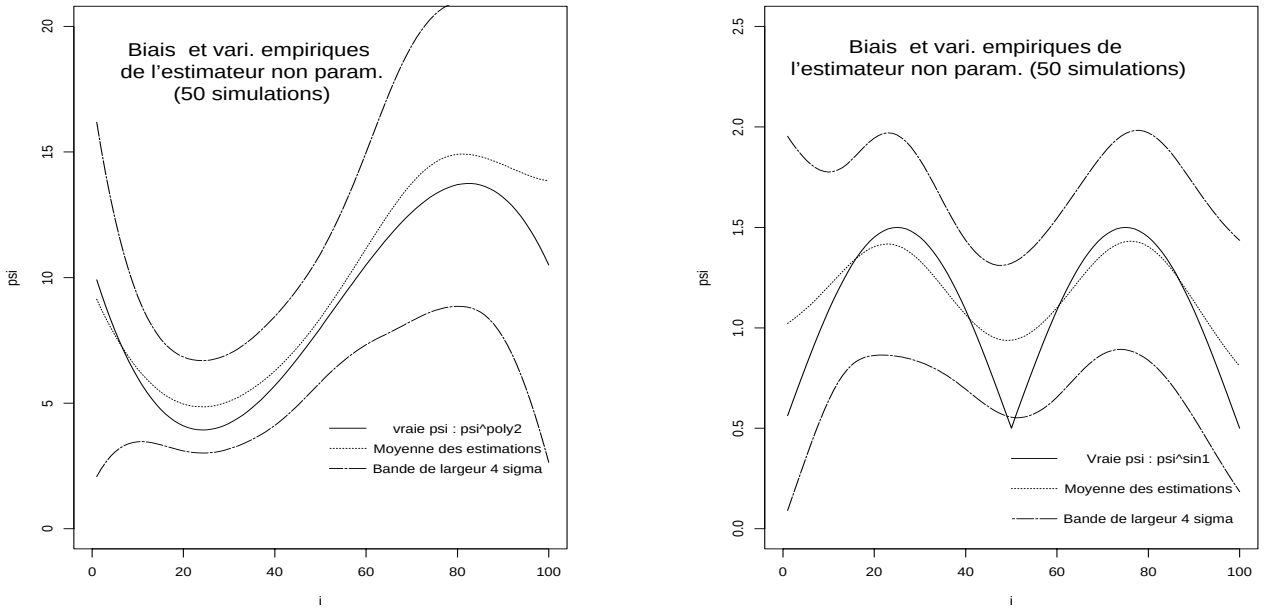

FIG. 2.8: Moyenne et bande de confiance empiriques des estimations non paramétriques de ψ^{poly2} et ψ^{sin1}.

Pour représenter la variance empirique des estimationsΓon trace autour de la moyenne des estimations de $\psi\Gamma$une bande de largeur $(4\sigma_i)_{i\leq n}$. Pour tout $i \leq n\Gamma\sigma_i$ désigne l'écart type empirique des 50 estimations de la quantité $\psi(i)$.

Cette bande représente une région de confiance empirique ayant approximativement une probabilité 95% de contenir l'estimation non paramétrique de ψ.

Les exemples traités ci-dessus montrent que l'estimateur non paramétrique a un comportement satisfaisant aussi bien au niveau du biais qu'au niveau de la variance. Une évaluation du *MISE* (Mean Integrated Square Errors) pourrait contribuer à l'évaluation des qualités de l'estimateur non paramétrique. Cette voie ne sera pas poursuivie dans ce travail.

Choix du paramètre degré de liberté

On étudie ici l'effet du paramètre *dl* sur les performances prédicitives du modèle ND_{np}.

On considère d'abord le jeu de données *Musa6*Γ les estimations non paramétriques de la fonction ψ correspondant à différentes valeurs du paramètre *dl* sont représentées sur la figure 2.9.

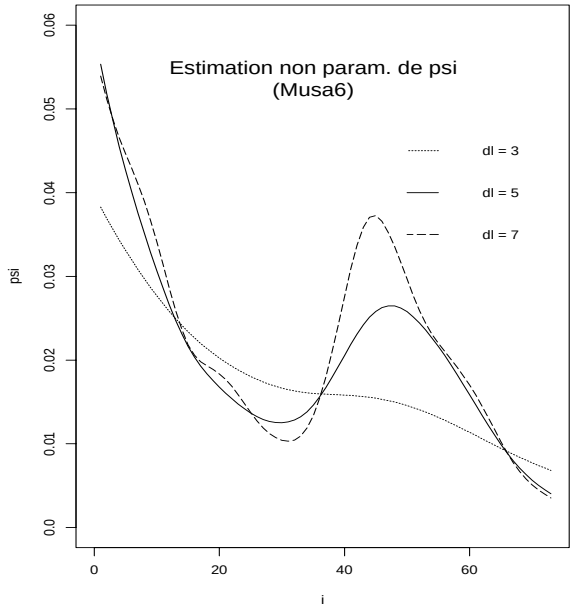

FIG. 2.9: Effet du paramètre de lissage sur l'estimation non paramétrique de ψ.

Dans le tableau 2.6Γon étudieΓpour le jeu de données *Musa6*Γl'effet du paramètre *dl* sur la qualité prévisionnelle et la déviance du modèle non paramétrique.

Remarque – La procédure *u-plot* a été mise en œuvre sur les $n-20$ dernières données.

	$dl=3$	$dl=5$	$dl=7$	$dl=9$
kolm $(u-plot)$	0.200	0.183	0.207	0.235
Déviance	173.88	160.16	152.47	143.31

TAB. 2.6: Effet du paramètre de lissage sur les qualités du modèle ND_{np}. (Musa6)

On remarque que pour le jeu de données *Musa6*, le modèle le plus robuste ($dl=3$) n'a pas le meilleur pouvoir prédictif. C'est le modèle ND_{np} à 5 degrés de liberté qui semble avoir les meilleures qualités prévisionnelles.

La qualité prévisionnelle d'un modèle semble résulter de sa capacité à assurer un bon compromis entre la robustesse et la qualité d'ajustement.

On étudie enfin, au tableau 2.7 l'influence du paramètre dl sur la qualité prévisionnelle du modèle ND_{np} pour différents jeux de données.

	$dl=3$	$dl=5$	$dl=7$
Cisi1	0.072	**0.068**	0.086
Musa1	0.099	**0.096**	0.101
Musa3	**0.165**	0.238	0.272
Musa6	0.200	**0.183**	0.207
Musa14c	0.214	0.202	**0.183**
Crow1	**0.070**	0.130	0.133

TAB. 2.7: Critère du *u-plot*

Le tableau précédent confirme le fait que le modèle le plus robuste ($dl=3$) n'a pas forcément la meilleure qualité prévisionnelle. Il est par ailleurs clair que la valeur optimale (d'un point de vue prédictif) du paramètre dl dépend du jeu de donné étudié.

Remarque – Pour simplifier l'utilisation de l'approche ND_{np}, on choisit de prendre dans la suite $\boldsymbol{dl=5}$.

Performances prédictives des modèles ND_{np}

On compare, pour différents jeux de données, les performances prédictives du modèle ND_{np} aux performances du modèle MPD_{pol} ainsi qu'aux performances d'autres modèles paramétriques utilisés en Fiabilité des Logiciels : le modèle de *Crow*, le modèle de *Goel-Okumoto* et le modèle de *Yamada-Ohba-Osaki* [104] noté YOO.

	ND_{np}	MPD_{pol}	MPD	$Crow$	$Goel-Okum.$	YOO
Cisi1	**0.068**	0.104	0.110	0.143	0.088	0.235
Musa1	**0.096**	0.111	0.111	0.133	0.142	0.405
Musa3	0.238	**0.218**	0.218	0.397	0.247	0.573
Musa6	**0.183**	0.223	0.222	0.214	0.223	0.341
Musa14c	0.202	**0.158**	0.396	0.405	0.468	0.302
Crow1	0.130	0.174	0.161	**0.083**	0.167	0.428

TAB. 2.8: Critère du *u-plot*.

Le modèle ND_{np} semble avoir dans la majorité des cas de meilleures qualités prévisionnelles que les autres modèles paramétriques classiques.

Il ressort des résultats précédents que les approches ND_{np} et MPD_{pol} fournissent toujoursΓ exception faite du jeu de données *Crow1*Γle modèle ayant le meilleur pouvoir prédictif.

Pour le jeu de données *Crow1*Γseul le modèle à partir duquel ont été simulées les données (le modèle de *Crow*) a de meilleures performances prédictives que le modèle ND_{np}.

2.6 Conclusion

L'utilisation des modèles linéaires généralisés nous a permis de généraliser et d'améliorer les performances des modèles *ND* classiques.
Les résultats et les algorithmes développés dans le cadre *GLM* nous ont permis de développer deux approches de construction et de choix de modèles : une approche paramétrique (ND_{pol}) et une approche non paramétrique (ND_{np}).

Ces approches ont l'avantage de pouvoir tenir compte des spécificités de chaque jeu de données pour donner des modèles ayant d'excellentes qualités d'ajustement.
A cette amélioration de la qualité d'ajustement s'ajoute une nette amélioration de la qualité prévisionnelle.
En effet les résultats expérimentaux ont montré que les modèles issus des approches ND_{pol} et ND_{np} ontΓdans la majorité des casΓdes qualités prédicitives meilleures que tous les autres modèles classiques.

Reste cependant le problème du choix entre l'approche paramétrique ND_{pol} et l'approche non paramétrique ND_{np}.

Les principaux avantages de l'approche ND_{pol} sont la facilité d'utilisation et la possibilité de donner des interprétations physiques aux différents paramètres.
Dans cette approche l'utilisateur choisit la fonction de lienΓle test de rapport de vraisemblances maximales permet ensuite de déterminer le nombre de paramètres et la méthode du maximum de vraisemblance permet de les estimer.

De son cotéΓl'approche non paramétrique a l'avantage de fournir une unification de tous les modèles *ND* puisqu'elle permet d'estimer l'effet combiné de la fonction de lien et de

la fonction de régression.
L'utilisateur de l'approche ND_{np} spécifie son modèle en choisissant le degré de lissitude de la fonction taux de défaillance, c'est-à-dire en choisissant le paramètre dl.

Une extension possible de l'approche ND_{np} pourrait consister à choisir le paramètre degré de liberté dl par des méthodes automatiques telles que la méthode de validation croisée adaptée au cadre *GLM* non paramétrique (cf. [44]).

Xiang et Wahba [101] ont proposé récemment une approche de test basée sur la distance de *Kullback-Leibler* permettant de tester, dans le cas d'un seul régresseur, l'hypothèse *GLM* paramétrique contre l'hypothèse *GLM* non paramétrique.
Ce test pourra être utilisé pour comparer les modèles issus des approches ND_{pol} et ND_{np}.

Dans le contexte de la Fiabilité des Logiciels, il est cependant préférable de faire le choix entre les approches ND_{pol} et ND_{np} en se basant sur les connaissances a priori disponibles, les objectifs de l'étude de fiabilité ainsi que sur les préférences de l'utilisateur.

Chapitre 3

L'analyse statistique bayésienne en Fiabilité des Logiciels

L'objectif de ce chapitre est la mise en œuvre d'un outil statistique bayésien général pour l'évaluation de la fiabilité des logiciels.

En partant des hypothèses usuelles de la Fiabilité des Logiciels on obtient un modèle général où les v.a.r. temps inter-défaillances X_i sont de lois exponentielles de paramètres λ_i. On présente ensuiteΓune analyse bayésienne générale du modèle précédent. On donne ainsi les expressions des estimateurs bayésiens des différents attributs de la fiabilitéΓles méthodes numériques permettant le calcul des estimations correspondantes ainsi que différents exemples d'hypothèses a priori envisageables.

L'approche bayésienne présentée dans ce chapitre a l'avantage de pouvoir s'adapter aux différents a priori que peuvent avoir les praticiens. Elle leur donne ainsi le moyen de construire leurs propres modèles. Ils n'auront pour ce faire qu'à préciser la forme de leurs connaissances a prioriΓils utiliseront ensuite les résultats et les algorithmes présentés dans ce chapitre.

3.1 Introduction

Le recours aux méthodes statistiques bayésiennes en Fiabilité des Logiciels a été envisagé pour répondre à certains défauts des méthodes inférentielles classiques (maximum de vraisemblance et moindres carrés) :

- ces méthodes classiques ne peuvent tenir compte d'une manière claire et précise des informationsΓsouvent subjectivesΓque peuvent avoir les ingénieurs ou les experts concernant le système étudié.
- Pour les systèmes à haute fiabilitéΓon peut ne pas observer de défaillancesΓles méthodes classiques sont donc difficilement applicables.
- Pour certains jeux de donnéesΓla fonction de vraisemblance peut ne pas avoir de maximum.

Le principal avantage qu'offre l'approche bayésienne en Fiabilité est la possibilité de traiter et d'utiliser séparément les deux sources d'information disponibles :

- Information issue de ce que "pensent" les ingénieurs et les experts.
- Information issue de l'observation du système étudié.

Dans le cas où les a priori des experts ne sont pas complètement erronésΓ l'approche inférentielle bayésienne donne de très bons estimateurs même si le nombre d'observations est assez faible.

Comme on va le voir dans la section 3.3Γla plupart des approches bayésiennes adoptées en Fiabilité des LogicielsΓintroduisent un certain nombre d'hypothèses mathématiques souvent artificielles. Si ces hypothèses simplifient énormément le calcul des estimateurs bayésiensΓelles éloignent souvent les modèles obtenus des connaissances physiques qu'ont réellement les experts.
Il faut en effet noter que les estimateurs issus de l'analyse bayésienne sont souvent donnés sous forme de rapports d'intégrales multiples non simplifiables.

L'élaboration de nouvelles techniques numériques performantes et faciles à mettre en œuvre devraitΓcomme on va l'illustrer dans ce chapitreΓpermettre aux concepteurs de modèles de se libérer de leurs contraintes techniques pour aller plus vers la modélisation des connaissances a priori des ingénieurs et autres experts.

On commenceΓdans la section 3.2Γpar rappeler les outils de base de l'analyse statistique bayésienne.

Une revue des principales études bayésiennes en Fiabilité des Logiciels est proposée dans la section 3.3.

On présente ensuiteΓdans la section 3.4Γune analyse bayésienne générale des modèles à lois exponentielles.

Cette analyse est ensuite affinée dans la section 3.5. On donne ainsi différents exemples d'a priori possibles et on présente des algorithmes numériques permettant le calcul des différentes estimations bayésiennes.
On donne enfin des exemples d'application de l'approche bayésienne développée dans ce chapitre.

3.2 L'approche statistique bayésienne

On suppose dans toute cette section qu'on a :

- un vecteur d'observations x appartenant à un sous ensemble $\mathcal{X}$ de $\mathbb{R}^n$Γ

- un modèle statistique paramétrique où x est une réalisation d'une loi de probabilité paramétrique P_θ spécifiée par sa densité par rapport à la mesure de Lebesgue sur $\mathbb{R}^n$:

$$f(x\,|\,\theta) \quad x \in \mathcal{X}.$$

  où θ est un paramètre vectoriel inconnu appartenant à un sous-ensemble Θ de $\mathbb{R}^p$.

L'approche inférentielle bayésienne dont les concepts de base sont brièvement rappelés ci-dessous permet d'estimer θ en tenant compte du vecteur des observations x du modèle paramétrique $f(x\,|\,\theta)$ et des éventuelles informations a priori résumées sous forme d'une loi de probabilité sur θ.

Une présentation détaillée de l'analyse statistique bayésienne peut être trouvée par exemple dans Robert [83] les notations et résultats de cette section en sont inspirés.

3.2.1 Concepts de base

Probabilité subjective

Une des bases de l'analyse statistique bayésienne est la notion de **probabilité subjective**. Cette notion est utilisée pour modéliser l'avis d'un individu concernant une proposition ou une hypothèse ("La probabilité d'une vie après la mort" "La probabilité qu'il fasse beau demain" etc.).
La notion de probabilité subjective différente de la notion de probabilité classique ou "fréquentiste" modélise un état d'incertitude ou un degré de croyance différent d'un individu à un autre.

Dans la majorité des analyses statistiques bayésiennes on utilise cette notion de probabilité subjective pour modéliser à travers des lois de probabilité les opinions des experts et le degré de confiance en ces opinions. Ces opinions sont ensuite confirmées ou infirmées par l'observation du fonctionnement du système physique étudié. La contribution de l'expert dans l'analyse statistique devient ainsi plus explicite.

Le modèle statistique bayésien

Définition – 3.1 *Dans le cadre d'une analyse bayésienne, on dispose généralement des opinions des experts. Ces opinions sont décrites par une loi de probabilité sur l'espace des paramètres Θ, appelée* ***loi a priori****.*

Notation – On notera dans la suite $\pi(\theta)$ la densité de la loi a priori de θ par rapport à une mesure de référence positive ν sur Θ.

Définition – 3.2 *Un* ***modèle statistique bayésien*** *est un modèle statistique paramétrique $f(x\,|\,\theta)$ où le paramètre inconnu θ est considéré comme une variable aléatoire de loi a priori : $\pi(\theta)$.*

Comme le souligne Robert [83]Γ"*le passage de la notion de paramètre inconnu à celle d'un paramètre aléatoire représente un saut délicat qui divise toujours les statisticiens*".
L'avantage de ce passage est d'introduire l'information a priori subjective ainsi que le degré de croyance en cette information.

Cette connaissance a priori est mise à jour au fur et à mesure de l'arrivée des observations. La mise à jour est faite par l'utilisation de la formule de Bayes donnant la loi a posteriori de θ :

Définition – 3.3 *La **loi a posteriori** de θ est sa loi conditionnellement à l'observation x. Cette loi est donnée par sa densité par rapport à ν :*

$$\pi(\theta \mid x) \equiv \frac{f(x \mid \theta)\pi(\theta)}{\int_\Theta f(x \mid \theta)\pi(\theta) d\nu(\theta)}.$$

La loi a posteriori combine ainsi l'information subjective de l'expert et l'information issue des observations x pour décrire l'information disponible concernant le paramètre θ.

Pour extraire une estimation ponctuelle de θΓon peut utiliser deux approches concurrentes :

- Estimateur de **maximum de vraisemblance bayésien**Γqui correspond au mode de la loi a posteriori $\pi(\theta \mid x)$.

- **Estimateur de Bayes** : cet estimateur est obtenu en minimisant l'espérance a posteriori d'une fonction de coût associée au problème considéré. Cette approche est détaillée dans la sous-section suivante.

Remarques –

1. Dans le **cas non-informatif**Γc'est-à-dire quand on ne dispose pas d'informations a prioriΓon peut encore utiliser l'approche bayésienne en prenant une densité a priori $\pi(\theta)$ constante.
   Si Θ est un ensemble borné Γla loi a priori ainsi définie est la loi uniforme sur Θ. Dans le cas contraireΓla fonction constante $\pi(\theta)$ n'est plus une densité de probabilitéΓon parle alors de **loi a priori impropre** ou généralisée.

2. L'estimateur de maximum de vraisemblance bayésien se confondΓdans le cas non-informatifΓavec l'estimateur de maximum de vraisemblance classique.

Le calcul bayésien peut aussi être utilisé dans un but prédictif. En effetΓsoient X et Y deux v.a.r. de densités respectives $f(x \mid \theta)$ et $g(y \mid \theta)$ et supposons que l'on dispose d'une observation x de X. La loi prédictive de YΓdéfinie ci-dessousΓpermet alors d'utiliser l'observation x pour améliorer les prédictions de Y :

Définition – 3.4 *On appelle **loi prédictive** de Y sa loi de probabilité conditionnellement à $X = x$. Sa densité est donnée par :*

$$g(y \mid x) = \int_\Theta g(y \mid \theta)\, \pi(\theta \mid x)\, d\nu(\theta).$$

3.2.2 Fonction de coût, risques et estimateurs de Bayes

Le but d'une analyse statistique bayésienne est souvent l'estimation de quantités dépendant du paramètre inconnu θ.
Plus généralementΓil s'agira de prendre une décision $\delta(x)$ (typiquement un estimateur de θ) dans un espace de décisions D en tenant compte de l'observation x.

L'utilisation de la décision $\delta(x)$ engendre un coût (ou une perte) fonction de la valeur du paramètre θ. Ce coût est évalué par une **fonction de coût** :

$$L : \Theta \times D \longrightarrow [0, \infty[.$$

L'analyse statistique doit fournir une règle de décisionΓδ fonction de $\mathcal{X}$ dans DΓpermettant de minimiser en un certain sens le coût $L(\theta, \delta(x))$.

Dans les problèmes réels il n'existe pas de fonctions de décision δ minimisant le coût $L(\theta, \delta(x))$Γpour tout θ et tout x. Il faut alors proposer des critèresΓpermettant de comparer différentes règles de décision.

Comparaison de règles de décision

L'approche fréquentiste considère un coût moyen sur toutes les valeurs possibles du vecteur d'observations xΓce coût moyenΓappelé **risque fréquentiste** est donné par :

$$R(\theta, \delta) = \int_{\mathcal{X}} L(\theta, \delta(x))\, f(x \mid \theta)\, dx.$$

L'utilité pratique de $R(\theta, \delta)$ est limitée de par sa dépendance du paramètre inconnu θ.

L'approche bayésienneΓau lieu d'intégrer la fonction coût $L(\theta, \delta(x))$ par rapport au vecteur d'observations xΓl'intègre par rapport à θ. On obtient alors le risque a posteriori :

Définition – 3.5 *Quand on a un vecteur d'observations x, on peut comparer deux fonctions de décision en utilisant* ***le risque a posteriori*** *donné par :*

$$\begin{aligned} \rho(\pi, \delta \mid x) &= E^{\pi}(L(\theta, \delta) \mid x) \\ &= \int_{\Theta} L(\theta, \delta(x))\, \pi(\theta \mid x) d\nu(\theta) \end{aligned}$$

Le risque a posteriori est donc l'espérance du coût par rapport à la loi a posteriori de θ.

On peut aussi comparer les règles de décision selon leur risque de Bayes :

Définition – 3.6 *Le* ***risque de Bayes*** *est l'espérance du risque fréquentiste par rapport à la loi a priori de θ :*

$$r(\pi, \delta) = E^{\pi}\left[R(\theta, \delta)\right] = \int_{\Theta} \int_{\mathcal{X}} L(\theta, \delta(x))\, f(x \mid \theta)\, dx\, \pi(\theta)\, d\nu(\theta).$$

Le risque de Bayes donne donc une valeur réelle et non pas une fonction de θ. Il permet ainsi de comparer deux fonctions de décision (ou deux estimateurs).
Le risque de Bayes permet d'introduire la notion d'estimateur de Bayes décrite ci-dessous.

Estimateurs de Bayes

Définition – 3.7 *On appelle* ***estimateur de Bayes*** *associé à une fonction de coût L et à une loi a priori π toute fonction de décision δ^π minimisant le risque de Bayes $r(\pi, \delta)$ sur l'espace D.*

La propriété suivante permet de donner un exemple d'estimateurs de Bayes :

Propriété – On peut construire un estimateur de Bayes en prenant pour chaque observation $x \in \mathcal{X}\Gamma$la décision $\delta^\pi(x)$ minimisant le risque a posteriori $\rho(\pi, \delta \mid x)$:

$$\forall x \in \mathcal{X} \ \Gamma \quad \delta^\pi(x) = Arg \min_{d \in D} \rho(\pi, d \mid x). \tag{3.1}$$

Preuve – On utilise le théorème de Fubini pour montrer que :

$$r(\pi, \delta) = \int_{\mathcal{X}} \rho(\pi, \delta \mid x) f(x)\, dx. \tag{3.2}$$

La fonction de décision δ^π définie par (3.1)Γest un estimateur de Bayes puisque pour toute règle de décision d de D on a :

$$\forall x \in \mathcal{X} \ \Gamma \quad \delta^\pi(x) \leq d(x)$$

par conséquentΓen utilisant l'équation (3.2)Γon a finalement :

$$\forall d \in D \ \Gamma \quad r(\pi, \delta^\pi) \leq r(\pi, d).$$

□

Choix des fonctions de coût

La possibilité d'introduireΓà travers la fonction de coûtΓles pertes et les conséquences des mauvaises estimations des paramètres inconnus est un avantage supplémentaire de l'analyse inférentielle bayésienne.
Il est par exemple logique de penser qu'un expert en logiciels saura évaluer les conséquences que provoquerait une sous-estimation ou une surestimation de la fiabilité de son logiciel.

Lorsque le manque d'informationΓou le manque de temps ne permet pas de spécifier la fonction coûtΓon peut utiliser des coûts classiquesΓsimples et bien étudiés. On en présente ci-dessous deux exemples.

Définition – 3.8 *La fonction de* ***coût quadratique*** *est donnée par :*

- *Si θ est un paramètre réel :*

$$L(\theta, \delta(x)) = c(\theta - \delta(x))^2$$

où c est une constante positive.

- *Si θ est un paramètre vectoriel :*

$$L(\theta, \delta(x)) = {}^t(\theta - \delta(x))\, Q\, (\theta - \delta(x))$$

où Q est une matrice symétrique définie positive.

Remarques –

1. Pour le coût quadratiqueΓl'estiamateur de Bayes de θ est donné par l'espérance a posteriori de θ conditionnellement à l'observation x :

$$\delta^\pi(x) = E^\pi(\theta \mid x).$$

2. Le coût quadratique donne l'estimateur de θ de variance a posteriori minimale.

Définition – 3.9 *Pour certains problèmes il est préférable d'utiliser une fonction* ***coût absolu*** *donné, dans le cas d'un paramètre réel, par :*

$$L(\theta, \delta(x)) = c \mid \theta - \delta(x) \mid .$$

L'estimateur de Bayes est alors donné par la médiane de la loi a posteriori $\pi(\theta \mid x)$.

3.3 Revue des approches bayésiennes en Fiabilité des Logiciels

On présente dans cette section les principales approches bayésiennes en Fiabilité des Logiciels.

Dans toutes ces approchesΓles auteurs présentent différentes analyses inférentielles bayésiennes de modèles paramétriques appartenant à l'une des classes suivantes :

- le modèle de *Jelinski-Moranda*
- les modèles *NHPP*
- les modèles à lois exponentielles où les v.a.r. temps inter-défaillances sont de lois exponentielles.

Remarque – Dans les différentes approches bayésiennes présentées ci-dessous la mesure de référence ν sera déduite du contexte.

3.3.1 Traitements bayésiens du modèle de *Jelinski-Moranda*

Le modèle de *Jelinski-Moranda* présenté dans la section 1.4 est l'un des modèles les plus utilisés par les praticiens. Il est aussi le modèle où la méthode de maximum de vraisemblance pose le plus de problèmes (cf. [66]) : estimateurs peu robustesΓestimations aberrantes notamment en cas de décroissance de fiabilitéΓsous-estimation systématique de la fiabilitéΓetc.

Ces problèmes ont été à l'origine des premières études bayésiennes en Fiabilité des Logiciels.
On a ainsi commencé par utiliser l'approche bayésienne pour estimer les deux paramètres du modèle *JM*

- N : nombre initial de fautes
- Φ : contribution de chaque faute à l'intensité de défaillance

en les considérant comme des variables aléatoires munies de lois a priori. On présente ci-dessous les principales extensions bayésiennes du modèle *JM*.

Langberg et Singpurwalla (1985)

Langberg et Singpurwalla [56] ont présenté une étude bayésienne assez générale où le paramètre N a une loi a priori discrète généraleΓspécifiée par la suite

$$[\,\pi_k = P(N=k)\,]_{k\geq 0}\,.$$

La loi a priori du paramètre Φ est une loi $Gamma(a,b)$ donnée par sa densité :

$$f_\Phi(x) = \frac{b^a}{\Gamma(a)}\, e^{-bx}\, x^{a-1}\ \Gamma \quad \forall x \geq 0.$$

Ils supposent par ailleurs que N et Φ sont a priori indépendantes.

La loi a posteriori du paramètre Φ conditionnellement à N est une loi $Gamma(a',b')$ où $a' = a+n$ et $b' = b + \sum_{i=1}^{n}(k-i+1)x_i$.

La loi a posteriori marginale de N est donnéeΓpour tout entier $k \geq n$Γpar :

$$P(N=k \mid x_1,\ldots,x_n) = \frac{\frac{k!}{(k-n)!}\,[\,a+\sum_{i=1}^{n}(k-i+1)x_i\,]^{-(b+n)}\,\pi_k}{\sum_{j=n}^{\infty}\frac{j!}{(j-n)!}\,[\,a+\sum_{i=1}^{n}(j-i+1)x_i\,]^{-(b+n)}\,\pi_j}$$

Langberg et Singpurwalla donnent ensuite deux cas particuliers de l'approche générale citée ci-dessus :

1. Ils supposent dans le premier cas que le paramètre N est connu. Ils montrent alors que la loi conjointe des variables temps inter-défaillances X_i est un mélange de lois de *Pareto* multivariées.

2. Dans le deuxième casΓils choisissent une loi a priori $Poisson(\theta)$ pour le paramètre N et ils supposent que le paramètre Φ est connu.
   La loi a posteriori de N est alors donnéeΓpour tout entier $k \geq n$Γpar :

   $$P(N=k \mid x_1,\ldots,x_n) = \frac{\theta^{k-n}}{(k-n)!}\,[\,exp(-\Phi t_n)\,]^{k-n}\,exp\,[\,-\theta exp(-\Phi t_n)\,]$$

   où $t_n = \sum_{i=1}^{n} x_i$.

   La loi a posteriori du nombre résiduel d'erreurs $N-n$ est dans ce cas une loi de *Poisson* de paramètre : $\theta\, e^{-\Phi t_n}$.

Jewell [48] reprend les a priori proposés par Langberg et Singpurwalla (une loi *Gamma* pour Φ et une loi de *Poisson* pour N) mais il suppose en plus que le paramètre de la loi de *Poisson* est une variable aléatoire ayant pour loi a priori une loi $Gamma(c,d)$ où c et d sont deux constantes de $\mathbb{R}_+^*$.
Ces hypothèses font que la loi a priori de N est la loi décrite par :

$$\forall k \in \mathbb{N}\ \Gamma\ P(N=k) = \frac{\Gamma(c+k)}{\Gamma(c)\,k!}\left(\frac{d}{1+d}\right)^c\left(\frac{1}{1+d}\right)^k,$$

dans le cas où le paramètre $c \in \mathbb{N}$Γla loi précédente est la loi de *Pascal* de paramètres $(c, \frac{1}{1+d})$.

Jewell donne dans son étude les lois a posteriori des paramètres N et Φ et s'intéresse à l'estimation du nombre d'erreurs résiduelles.

Csenki [21] part aussi des hypothèses de Langberg et Singpurwalla pour présenter une intéressante approche bayésienne prédictive.
Il montre que sous des lois a priori *Poisson* pour N et *Gamma* pour $\Phi\Gamma$la loi prédictive du prochain temps inter-défaillances X_{n+1} est une loi *Beta* inverse tronquée.
Il donne alors les expressions explicites de la densité prédictive de $X_{n+1}\Gamma$de la fonction de fiabilité et du taux de défaillance prédictifs.

Une étude semblable à celle de Csenki a été proposée par Wright et Hazelhurst [100].

Littlewood et Sofer (1987)

Littlewood et Sofer [66] ont essayé de résoudre les problèmes inférentiels du modèle *JM* en proposant une reparamétrisation du modèle. Les v.a.r. X_i sont toujours supposées indépendantes mais de lois :

$$X_i \sim Exp(\lambda - \Phi(i-1)).$$

Le paramètre entier N est ainsi remplacé par un paramètre réel λ représentant l'intensité de défaillance initiale. Ceci permet d'éviter les problèmes que pose l'estimation du paramètre entier N.

Les auteurs suggèrent ensuite d'utiliser une approche inférentielle bayésienne où les paramètres λ et Φ sont des v.a.r. indépendantes de lois a priori *Gamma*. Les valeurs des paramètres de ces lois a priori devant être choisies par l'utilisateur du modèle.

Dans l'implémentation de leur approcheΓLittlewood et Sofer supposent cependant qu'ils ne disposent d'aucune information a priori. Ils utilisent alors des lois a priori non-informatives impropres : le couple (λ, Φ) est muni d'une loi a priori uniforme donnée par la densité : $\pi(\lambda, \phi) = 1\Gamma\forall\, \lambda \geq 0$ et $\Phi \in \mathbb{R}$.

Les auteurs donnent sous ces hypothèses la loi a posteriori du couple (λ, Φ) ainsi que l'expression explicite de la fonction de fiabilité prédictiveΓla loi prédictive du prochain temps inter-défaillances et la loi a posteriori du nombre résiduel de fautes.

3.3.2 Traitements bayésiens des modèles *NHPP*

On suppose dans cette sous-section que le processus de défaillance est modélisé par un processus de Poisson non homogène (*NHPP*) de fonction intensité de défaillance $\lambda(t \mid \theta)$ où θ est un paramètre inconnu de $\Theta \subset \mathbb{R}^p$.

Notation et Rappels –

1. La fonction nombre moyen de défaillances sur l'intervalle $[0, t]$ est notée :

$$m(t \mid \theta) \equiv E(N_t) = \int_0^t \lambda(s \mid \theta) ds. \tag{3.3}$$

2. On rappelle que pour tout $t \geq 0$, la v.a.r. N_t est de loi de $Poisson\,[\, m(t \mid \theta)\,]$.

3. Après observation des n premiers instants de défaillance $t_1, \ldots, t_n$, la vraisemblance du paramètre θ est donnée par :

$$L(\theta; t_1, \ldots, t_n) = \prod_{i=1}^{n} [\, \lambda(t_i \mid \theta)\,] \; exp\,[\, - \int_0^{t_n} \lambda(s \mid \theta) ds\,].$$

Résultats communs à tous les modèles *NHPP*

Il est plus facile d'avoir les avis des experts sur des quantités ayant des significations physiques que d'avoir leurs avis sur le paramètre θ. C'est au modélisateur de traduire ensuite ces avis en lois a priori sur θ.

Campodónico et Singpurwalla [13] présentent une méthodologie, commune à tous les modèles *NHPP*, permettant d'exprimer les opinions des experts concernant le nombre de défaillances futures sous forme de loi a priori sur θ. Cette méthodologie sera brièvement décrite plus tard.

Le choix de la loi a priori $\pi(\theta)$ étant fait, il reste à calculer les estimateurs bayésiens des différentes variables d'intérêt. Les expressions de ces estimateurs, communes à tous les modèles *NHPP*, sont données ci-dessous.

La loi a posteriori de θ est donnée par :

$$f(\theta \mid t_1, \ldots, t_n) \propto L(\theta; t_1, \ldots, t_n)\, \pi(\theta).$$

La loi prédictive de $N_{t_n,t}$ (nombre de défaillances qui seront observées sur l'intervalle de temps $[t_n, t]$) est donnée, pour tout $k \geq 0$, par :

$$P(N_{t_n,t} = k \mid t_1, \ldots, t_n) = \int_{\theta \in \Theta} P(N_{t_n,t} = k \mid \theta; t_1, \ldots, t_n)\, f(\theta \mid t_1, \ldots, t_n)\, d\theta$$

rappelons que dans un modèle *NHPP* on a :

$$P(N_{t_n,t} = k \mid \theta; t_1, \ldots, t_n) = \frac{[\, m(t \mid \theta) - m(t_n \mid \theta)\,]^k}{k!} \; exp\,[\, -(m(t \mid \theta) - m(t_n \mid \theta))\,]\,.$$

La loi prédictive du prochain temps inter-défaillances est donnée par :

$$P(X_{n+1} \leq x \mid t_1, \ldots, t_n) = \int_{\theta \in \Theta} P(X_{n+1} \leq x \mid \theta; t_1, \ldots, t_n)\, f(\theta \mid t_1, \ldots, t_n)\, d\theta$$

où :

$$P(X_{n+1} \leq x \mid \theta; t_1, \ldots, t_n) = 1 - exp\,[\, -m(t \mid \theta) + m(t_n \mid \theta)\,].$$

On présente ci-dessous deux exemples d'approches bayésiennes pour les modèles *NHPP*.

Kyparisis et Singpurwalla (1985)

Kyparisis et Singpurwalla [55] ont été les premiers à proposer une approche bayésienne pour l'estimation et la prédiction dans un modèle *NHPP*. Ils ont considéré le modèle *NHPP* où le processus de défaillance est modélisé par un processus de *Weibull* donné par sa fonction intensité de défaillance :

$$\lambda(t \mid \alpha, \beta) = (\frac{\beta}{\alpha})(\frac{t}{\alpha})^{\beta-1}.$$

La valeur du paramètre β est liée à la tendance de la fiabilité du logiciel étudié. L'avis de l'expert concernant cette tendance est alors utilisé pour déterminer la loi a priori du paramètre β.

En cas d'absence de connaissances a prioriΓSingpurwalla et Kyparisis suggèrent l'utilisation d'une loi a priori uniforme $Unif[0, \alpha_0]$ pour α. Ils choisissentΓpour $\beta\Gamma$une loi a priori *Beta* à support dans $[\beta_1, \beta_2]$ de densité :

$$f(\beta) = \frac{\Gamma(k_1 + k_2)}{\Gamma(k_1)\Gamma(k_2)} \frac{(\beta - \beta_1)^{k_1-1}(\beta_2 - \beta)^{k_2-1}}{(\beta_2 - \beta_1)^{k_1+k_2-1}}$$

où k_1 et k_2 sont deux constantes à fixer.

Sous ces hypothèsesΓla loi a posteriori conjointe du couple (α, β) est donnée par sa densité :

$$f(\alpha, \beta \mid t_1, \ldots, t_n) \propto (\beta - \beta_1)^{k_1-1}(\beta_2 - \beta)^{k_2-1}(\frac{\beta}{\alpha})^n \prod_{i=1}^{n}(\frac{t_i}{\alpha})^{\beta-1} exp\,[\, -(\frac{t_n}{\alpha})^{\beta}\,].$$

La loi prédictive de la variable X_{n+k} : temps d'attente de la $k^{ème}$ prochaine défaillance est donnée par :

$$Pr(X_{n+k} \leq x \mid t_1, \ldots, t_n) = \int_0^{\alpha_0} \int_{\beta_1}^{\beta_2} [\, \int_0^{v(t_n,x)} \frac{v^{k-1} exp(-v)}{(k-1)!}\, dv\,]\, f(\alpha, \beta \mid t_1, \ldots, t_n)\, d\alpha\, d\beta$$

où

$$v(t_n, x) = (\frac{t_n + x}{\alpha})^{\beta} - (\frac{t_n}{\alpha})^{\beta}.$$

En particulierΓla loi a posteriori de X_{n+1} est une loi de *Weibull* tronquée à gauche au point t_n.

Campodónico et Singpurwalla (1994)

Campodónico et Singpurwalla [12] proposent d'utiliser l'approche bayésienne pour étudier le modèle de *Musa-Okumoto* [77] où l'intensité de défaillance est donnée par :

$$\lambda_t = \frac{\lambda}{1 + \lambda\theta t}.$$

Ils présentent dans leur étude une méthodologie permettant de passer des avis subjectifs des experts aux lois a priori des paramètres du modèle.

Dans cette méthodologie, on demande aux experts de donner leurs opinions concernant les nombres moyens de défaillances :

$$m_1 = E(N_{t_1}) \text{ et } m_2 = E(N_{t_2})$$

qui seront observées au bout de deux instants particuliers t_1 et t_2.
Ces opinions sont transformées en lois a priori sur les v.a.r. m_1 et m_2. Les auteurs proposent alors d'exprimer les deux paramètres λ et θ en fonction de m_1 et m_2 en résolvant numériquement le système suivant :

$$\begin{cases} m_1 = \frac{1}{\theta} ln(1 + \lambda\theta t_1) \\ m_2 = \frac{1}{\theta} ln(1 + \lambda\theta t_2) \end{cases}$$

Ces expressions analytiques de λ et θ permettent alors de déduire les lois a priori de λ et θ à partir des lois a priori de m_1 et m_2.

Campodónico et Singpurwalla donnent, sous forme d'intégrales implicites, les estimations a posteriori des différents paramètres d'intérêt.
Ils étudient ensuite la robustesse de leurs estimateurs en considérant différentes valeurs pour les constantes de leurs lois a priori.

Dans une autre étude (cf. [13]) les auteurs proposent une procédure générale permettant de traduire les opinions des experts dans le cadre général des processus aléatoires ponctuels.

Bunday et Al Ayoubi [11] présentent une approche bayésienne similaire. Ils considèrent trois classes de modèles *NHPP* où le processus des défaillances est modélisé successivement par un processus de *Pareto*, un processus de $Weibull$ et un processus de *Gumbel*.
Ils optent pour des lois a priori non informatives et utilisent l'approximation de *Lindley* (cf. [64]) pour le calcul numérique de leurs estimations.

3.3.3 Traitements bayésiens des modèles à lois exponentielles

Dans cette section, on considère la classe de modèles de Fiabilité des Logiciels où les v.a.r. temps inter-défaillances sont des v.a.r. indépendantes de lois exponentielles :

$$\forall i \geq 1 \quad X_i \sim Exp(\lambda_i).$$

H_{exp}

L'évolution des taux de défaillance λ_i résulte de l'effetΓgénéralement inconnuΓdes corrections effectuées.

En adoptant l'approche bayésienneΓles paramètres λ_i seront considérés comme des variables aléatoiresΓleurs lois a priori seront extraites de l'idée a priori qu'ont les experts à propos des effets réels des différentes corrections.

Notations – Les paramètres taux de défaillances λ_i considérés comme des v.a.r. seront notés Λ_i.

Modèle de Littlewood et Verrall (1973)

Littlewood et Verrall [67] supposent a priori que les v.a.r. Λ_i sont indépendandes de lois *Gamma* :

$$\forall i \geq 1\Gamma\ \Lambda_i \sim Gamma(\alpha, \psi(i)).$$

Comme $E(\Lambda_i) = \alpha/\psi(i)\Gamma$la fonction ψ traduit l'opinion a priori de l'expert concernant la tendance de la fiabilité du logiciel étudié. Une fonction ψ croissante impliquerait une croissance de fiabilité.

Dans le cas où la fonction ψ n'est pas connueΓon peut l'estimer en la supposant membre d'une famille de fonctions paramétriques $\{\psi(.,\beta);\ \beta \subset \mathbb{R}^k\}$. Le paramètre β peut être estimé par la méthode du maximum de vraisemblanceΓon parle dans ce cas d'approche bayésienne empirique.

Littlewood et Verrall proposent d'estimer β par une approche bayésienneΓils suggèrent une loi a priori conjointe uniforme pour le couple (α, β).

Mazzuchi et Soyer (1988)

Mazzuchi et Soyer [70] partent aussi de l'hypothèse $H_{exp}\Gamma$et supposent a prioriΓLittlewood et VerrallΓque les variables Λ_i sont indépendantes de lois a priori :

$$\Lambda_i \sim Gamma(\alpha, \psi(i))\ \Gamma\text{où}\ \psi(i) = \beta + \gamma i\Gamma\alpha > 0\Gamma\beta + \gamma > 0\ \text{et}\ \gamma > 0.$$

Les paramètres $\alpha\Gamma\beta$ et γ sontΓeux aussiΓconsidérés comme des variables aléatoires ayantΓ pour des raisons techniquesΓles lois a priori suivantes :

- Une loi uniforme pour α :

$$\forall \alpha \in [0, \alpha_0]\ \Gamma\ \pi(\alpha) = \frac{1}{\alpha_0}.$$

- Conditionnellement à γ on a :

$$\gamma + \beta \sim Gamma(a, b).$$

- Une loi a priori $Gamma(c, d)$ pour γ.

Dans leur approcheΓMazzuchi et Soyer demandent à l'utilisateur de fixer les valeurs des paramètres $\alpha_0 \Gamma a \Gamma b \Gamma c$ et d.

Mazzuchi et Soyer supposent par ailleurs que :

- La variable α est indépendante des variables β et γ.
- Pour $i \leq n \Gamma$conditionnellement à $\Lambda_i \Gamma$la v.a.r. X_i est indépendante des variables $\alpha \Gamma \beta \Gamma \gamma$ et $(\Lambda_j)_{j \neq i}$.

Ayant fait toutes ces hypothèsesΓMazzuchi et Soyer donnent alors les lois a posteriori des paramètres $\alpha \Gamma \beta \Gamma \gamma$ et $(\lambda_i)_{i \leq n}$.
Ils donnent aussi la loi prédictive du temps d'attente de la prochaine défaillance X_{n+1}.

Les auteurs utilisent ensuite l'approximation de *Lindley* (cf. [64]) pour le calcul numérique de leurs estimations.

Becker et Camaranipoulos (1990)

Becker et Camaranipoulos [8] présentent une approche itérative pour le choix des lois a priori des taux de défaillance λ_i. Ils considèrent en particulier que le logiciel peutΓau bout d'un certain nombre de correctionsΓdevenir parfaitΓc'est à dire ne contenant plus de fautes.

Dans leur approche ils supposentΓparadoxalementΓque les variations successives des taux de défaillances $(\Delta_i = \lambda_i - \lambda_{i+1})_{i \geq 1}$ sont des constantes connues par ailleurs.

Ils partent alors d'une loi a priori non-informative pour le premier taux de défaillance (i.e. $\pi(\lambda_1) = constante \Gamma \forall \lambda_1 \geq 0$) et mettent à jour$\Gamma$les lois a priori des v.a.r. Λ_i au fur et à mesure de l'arrivée des défaillances.

Après observation du premier temps inter-défaillances $x_1 \Gamma$la loi a posteriori de Λ_1 est donnée par sa densité :

$$\forall \lambda_1 \geq 0 \ \Gamma \ f_{\Lambda_1 | x_1}(\lambda_1) = \lambda_1 x_1^2 \, exp(-\lambda_1 x_1).$$

Après la correction qui suit cette première défaillanceΓle nouveau taux de défaillance est donné par : $\Lambda_2 = max(0, \Lambda_1 - \Delta_1)$.
La loi de Λ_2 conditionnellement à l'observation $x_1 \Gamma$est donc obtenue par un décalage à

gauche de la loi de Λ_1Γsa densité par rapport à la mesure somme de la mesure de Lebesgue sur $\mathbb{R}_+$ et de la mesure de Dirac en zéro est donnée pour tout $\lambda_2 \geq 0$ par :

$$f_{\Lambda_2|x_1}(\lambda_2) = 1_{\{\lambda_2=0\}} \int_0^{\Delta_1} \lambda_2 x_1^2 exp(-\lambda x_1)\, d\lambda + 1_{\{\lambda_2>0\}}\, (\lambda_2 + \Delta_1) x_1^2 exp\,[\,-(\lambda_2+\Delta_1)x_1\,].$$

Cette loi a une masse de probabilité non nulle au point 0Γce qui représente la probabilité que le logiciel ne contienne plus de fautes après la première correction.

Les auteurs proposent d'utiliser cette loi comme loi a priori pour la v.a.r. Λ_2.

Après observation du deuxième temps inter-défaillances x_2Γla loi a posteriori de Λ_2 est alors donnée par sa densité :

$$\begin{aligned} f_{\Lambda_2|x_1,x_2}(\lambda_2) &\propto f_{\Lambda_2|x_1}(\lambda_2)\,.\, f_{X_2|\lambda_2}(x_2) \\ &\propto \lambda_2(\lambda_2+\Delta_1)\, exp\,[\,-\lambda_2(x_1+x_2)\,]. \end{aligned}$$

En réitérant cette procédureΓils obtiennent les lois a priori adéquates pour les différents taux de défaillance. A chaque étapeΓla loi a priori du prochain taux de défaillance est obtenueΓpar décalageΓà partir de la loi a posteriori du taux de défaillance actuel.

Becker et Camaranipoulos montrent que toutes les lois a priori et a posteriori ainsi obtenues font partie d'une famille de lois fermée par décalage à gauche et par multiplication. Comme les lois des variables temps inter-défaillances X_i font aussi partie de cette famille de loisΓils obtiennent ainsi une famille de lois conjuguées donnée par l'expression générale de ses densités :

$$\forall \lambda \geq 0\ \Gamma \quad f(\lambda) = e^{-b\lambda} \sum_{j=0}^{n} a_j \lambda^j.$$

L'utilisation des propriétés de cette famille de lois permet d'avoir des expressions simples pour les estimateurs bayésiens des différentes variables d'intérêt.

Intéressante d'un point de vue théoriqueΓl'approche de Becker et Camaranipoulos est assez critiquable du point de vue pratiqueΓelle doit en effet être précédée de l'utilisation d'autres modèles permettant d'estimer les constantes $(\Delta_i)_{i\geq 1}$Γconstantes que Becker et Camaranipoulos supposent connues.

3.3.4 Conclusion

Les hypothèses et les lois a priori utilisées dans la majorité des approches bayésiennes présentées ci-dessus ne se justifient que par les simplifications qu'elles apportent aux expressions des différents estimateurs.

Ces hypothèsesΓassez techniques et souvent très éloignées des connaissances a priori des praticiensΓprésentent un handicap important pour l'utilisation pratique de ces études bayésiennes.

Pour résoudre ce problèmeΓon propose dans la section suivante une approche bayésienne générale où on se limitera à des hypothèses minimales assez consensuelles dans le contexte de la Fiabilité des Logiciels.

Jelinski-Moranda	Meinhold et Singpurwalla (1983) Langberg et Sinpurwalla (1985) Jewell (1985) Littlewood et Sofer (1987) Wright et Hazelhurst (1987) Csenki (1990)
Modèles *NHPP*	Kyparisis et Singpurwalla (1985) Bunday et Al Ayoubi (1990) Campodónico et Singpurwalla (1995)
Modèles à lois exponentielles	Littlewood et Verrall (1973) Mazzuchi et Soyer (1988) Becker et Camaranipoulos (1990)

TAB. 3.1: Principales approches bayésiennes en Fiabilité des Logiciels

3.4 Analyse bayésienne générale des modèles à lois exponentielles

On présente dans cette section une analyse bayésienne générale du problème de l'évaluation de la fiabilité des logiciels.

On part d'hypothèses assez généralesΓpour aboutir à une modélisation où les v.a.r. X_i sont de lois exponentielles.
On présente alors une analyse bayésienne de ce modèle et on donne les expressions des estimateurs des différents attributs de la fiabilité.
Les résultats obtenus sont assez généraux et permettent à l'utilisateur d'intégrer ses propres connaissances a priori du phénomène étudié.

3.4.1 Les modèles à lois exponentielles

Définition – 3.10 *On appellera dans la suite* ***modèles à lois exponentielles*** *les modèles de fiabilité des logiciels où les temps inter-défaillances X_i sont des v.a.r. indépendantes de lois exponentielles :*

$$\forall i \geq 1,\ X_i \sim Exp(\lambda_i)$$

les λ_i sont des constantes positives.

Justification des hypothèses

L'hypothèse des lois exponentielles est une hypothèse naturelle commune à un grand nombre de modèles de fiabilité des logiciels. Elle peut se justifier par les deux hypothèses suivantes :

1. absence de phénomène d'usure pour un logiciel
2. chaque défaillance du logiciel est immédiatement suivie par une correction.

Remarque – La deuxième hypothèse est souvent vérifiée en période de tests. Si ce n'est pas le casΓon peut tout de même s'y ramener en remplaçant les temps inter-défaillances par les temps séparant chaque correction de la défaillance qui la suit.

D'après la première hypothèseΓentre deux corrections successives le logiciel se comporte comme un système sans viellisementΓd'où le choix de la loi exponentielle pour les v.a.r. X_i.

Selon la deuxième hypothèseΓchaque défaillance est suivie d'une correctionΓet toute correction change les caractéristiques du logiciel dans le but d'améliorer sa fiabilité.

Ces modifications des caractéristiques du logiciel se traduisent par des paramètres différents pour les lois des v.a.r. X_i. Ces paramètres caractérisent ainsi les états de fiabilité du logiciel entre les corrections successives.

En adoptant des hypothèses minimales on aboutit ainsi à une modélisation où les v.a.r. X_i sont de lois exponentielles :

$$\forall i \geq 1\Gamma\ X_i \sim Exp(\lambda_i).$$

On peut par ailleurs supposer que les v.a.r. X_i sont indépendantes. Ceci s'explique par le fait qu'après chaque correction on a une nouvelle version du logiciel. La dépendance entre les versions successives étant entièrement modélisée par le lien entre les paramètres $\lambda_i\Gamma$il est naturel de supposer une indépendance stochastique des v.a.r. X_i.

Les modèles à lois exponentielles diffèrent entre eux par la façon dont est modélisée la relation entre les paramètres λ_i.
Dans le chapitre précédent par exempleΓon a étudié les modèles *ND* qui sont des modèles à lois exponentielles où les paramètres λ_i ne sont fonction que du nombre de défaillances observées.

On présente ci-dessous une approche générale permettant de modéliser le lien entre les paramètres λ_i.
Dans cette approcheΓbasée sur l'analyse statistique bayésienneΓtoutes les informations concernant l'état initial du logiciel et les effets des différentes corrections sont résumées par des lois de probabilité sur les paramètres λ_i.

3.4.2 Modélisation bayésienne exponentielle

Introduction

Pour affiner la modélisation décrite ci-dessusΓil faut modéliser le lien entre les paramètres λ_i. Le comportement de la suite λ_i reflète l'évolution de la fiabilité du logicielΓévolution due aux corrections effectuées au fur et à mesure de l'observation des défaillances.
Proposer un modèle d'évolution des λ_i revient donc à modéliser les effets des corrections successives du logiciel.

Dans un grand nombre de modèlesΓle lien entre les λ_i est modélisé par une approche paramétrique. Ces modèlesΓbien que simples à utiliserΓsont assez restrictifs puisqu'ils supposent une certaine forme pour les effets des correctionsΓforme qui est loin d'être vérifiée par tous les logiciels.

On se propose ici de modéliser l'évolution des **taux de défaillance** λ_i par une approche

bayésienne générale où chaque utilisateur à la possibilité d'introduire les spécificités de son problème.

L'approche statistique bayésienne permet d'utiliser les deux sources d'information suivantes :

- les observations $x_1, \ldots, x_n$ des n premiers temps inter-défaillancesΓ
- les informations a priori sur l'état initial du logiciel et les effets des différentes correctionsΓ

pour estimer et prédire la fiabilité du logiciel à travers :

- l'estimation des paramètres $\lambda_1, \ldots, \lambda_n$ et la prédiction des paramètres $\lambda_{n+1}, \lambda_{n+2}$, etc.
- la prédiction des prochains temps inter-défaillances X_{n+1}, X_{n+2}, etc.
- l'estimation de la fonction de fiabilitéΓdu $MTTF$Γetc.

Définitions et notations

La modélisation bayésienne de l'évolution des paramètres λ_i commence par les considérer comme des v.a.r.
Les informations a priori sur les effets des corrections et l'état initial du logiciel sont alors résumées par des lois de probabilité et des propriétés stochastiques pour les "v.a.r." λ_i.

Notations – On prendra dans la suite de ce chapitre les notations suivantes :

1. Pour éviter toute ambiguïtéΓon notera Λ_i les v.a.r. associées (par l'approche bayésienne) aux paramètres taux de défaillance λ_i que l'on souhaite estimer.
2. Pour alléger les notations on utilisera λ_i aussi bien pour désigner les taux de défaillance inconnus à estimer que les variables muettes intervenant dans les expressions des différentes intégrales utilisées ci-dessous.
3. On note Π la loi a priori du processus aléatoire $\{\Lambda_i\}_{i\geq 1}$.
4. Pour $i \geq 1$ on note $\Pi_i(\lambda_i)$ la densité de la loi marginale de la v.a.r. Λ_i par rapport à la mesure de Lebesgue sur $\mathbb{R}_+$.

   $\Pi_i(\lambda_i \,|\, \lambda_{i-1})$ désigne la densité de la loi a priori de la v.a.r. Λ_i conditionnellement à $\Lambda_{i-1} = \lambda_{i-1}$.
5. Pour $i \geq 1$Γ $\Pi_i(\lambda_1, \ldots, \lambda_i)$ est la densité de la loi a priori conjointe du vecteur $(\Lambda_1, \ldots, \Lambda_i)$ par rapport à la mesure de Lebesgue sur $\mathbb{R}_+^i$.

6. Si X et Y sont deux v.a.r. on note $f_{X|y}$ la densité de la loi de probabilité de X conditionnellement à $Y=y$.

7. On prend enfin les notation vectorielles suivantes :

$$\lambda^{(n)} \equiv (\lambda_1, \ldots, \lambda_n) \; \Gamma \; X^{(n)} \equiv (X_1, \ldots, X_n) \; \Gamma \; \Lambda^{(n)} \equiv (\Lambda_1, \ldots, \Lambda_n), \text{ etc.}$$

Définition – 3.11 *On appelle* ***modélisation bayésienne exponentielle*** *la modélisation générale où :*

- *les v.a.r. X_i sont de lois exponentielles de paramètres Λ_i aléatoires :*

$$\forall i \geq 1, \;\; X_i \sim Exp(\Lambda_i).$$

- *Conditionnellement à $\{\Lambda_i\}_{i\geq 1}$ les v.a.r. X_i sont indépendantes entre elles.*

- *Le processus aléatoire $\{\Lambda_i\}_{i\geq 1}$ est de loi a priori Π.*

Dans la modélisation précédenteΓla loi du vecteur $X^{(n)}$ conditionnellement à $\Lambda^{(n)}=\lambda^{(n)}$ est donnée par sa densité :

$$f_{X^{(n)}|\lambda^{(n)}}(x_1, \ldots, x_n) = \prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i}] \tag{3.4}$$

La loi de probabilité du vecteur $X^{(n)}$ est alors donnée par sa densité :

$$f_{X^{(n)}}(x_1, \ldots, x_n) = \int_{\mathbb{R}_+^n} \prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i}] \; \Pi_n(\lambda_1, \ldots, \lambda_n) \, d\lambda_1 \ldots d\lambda_n. \tag{3.5}$$

Dans l'approche statistique bayésienneΓl'estimation des taux de défaillance λ_i se fait à partir de la loi a posteriori du vecteur $\Lambda^{(n)}$.

3.4.3 Evaluation bayésienne de la fiabilité

Estimation bayésienne des taux de défaillance λ_i

Avant toute observationΓles connaissances a priori sont résumées par la loi a priori Π.

La mise à jour des a priori initiaux est effectuée en remplaçant la loi a priori Π par la loi de $\Lambda^{(n)}$ conditionnellement à $X^{(n)}=x^{(n)}$.
La densité de cette loi a posteriori est obtenue par la formule de Bayes :

$$f_{\Lambda^{(n)}|x^{(n)}}(\lambda_1,\ldots,\lambda_n) = \frac{\prod_{i=1}^{n}[\lambda_i e^{-\lambda_i x_i}]\ \Pi_n(\lambda_1,\ldots,\lambda_n)}{\int_{\mathbb{R}_+^n}\prod_{i=1}^{n}[\lambda_i e^{-\lambda_i x_i}]\ \Pi_n(\lambda_1,\ldots,\lambda_n)d\lambda_1\ldots d\lambda_n}. \tag{3.6}$$

Plusieurs estimateurs $\hat{\lambda}^{(n)}(X_1,\ldots,X_n)$ du vecteur $\lambda^{(n)}$ peuvent être extraits de la loi a posteriori précédente.

L'estimateur le plus utilisé est sans doute l'estimateur de Bayes relatif à la fonction de coût quadratique :

$$\hat{\lambda}^{(n)}(X_1,\ldots,X_n) = E(\Lambda^{(n)} \mid X^{(n)}) \tag{3.7}$$

Les estimateurs associés des taux de défaillance $\lambda_j\Gamma$pour $j = 1,\ldots,n$ sont alors :

$$\hat{\lambda}_j(X_1,\ldots,X_n) = \frac{\int_{\mathbb{R}_+^n}\ \lambda_j\ \prod_{i=1}^{n}[\lambda_i e^{-\lambda_i x_i}]\ \Pi_n(\lambda_1,\ldots,\lambda_n)d\lambda_1\ldots d\lambda_n}{\int_{\mathbb{R}_+^n}\prod_{i=1}^{n}[\lambda_i e^{-\lambda_i x_i}]\ \Pi_n(\lambda_1,\ldots,\lambda_n)d\lambda_1\ldots d\lambda_n} \tag{3.8}$$

Ces estimateurs s'expriment sous la forme d'intégrales multiples dont on présentera plus tard différentes méthodes de calcul.

Certaines spécificités du phénomène étudié peuvent inciterΓlors de l'estimation de $\lambda^{(n)}\Gamma$à choisir une fonction de coût :

$$L : \mathbb{R}_+^n \times \mathbb{R}_+^n \longrightarrow \mathbb{R}_+$$

différente de la fonction de coût quadratique.
Le calcul des estimations de Bayes $\hat{\lambda}_i(x_1,\ldots,x_n)$ devient alors plus délicat.

Il s'agit en effet de trouver les réels positifs $\hat{\lambda}^{(n)} \equiv (\hat{\lambda}_1,\ldots,\hat{\lambda}_n)$ minimisant la quantité :

$$E\,[\,L(\Lambda^{(n)},\hat{\lambda}^{(n)}) \mid x^{(n)}\,] = \int_{\mathbb{R}_+^n} L(\lambda^{(n)},\hat{\lambda}^{(n)})\, f_{\Lambda^{(n)}|x^{(n)}}(\lambda_1,\ldots,\lambda_n)\, d\lambda_1\ldots d\lambda_n. \tag{3.9}$$

le problème du choix de la fonction de coût sera discuté plus tard.

Un autre estimateur du vecteur $\lambda^{(n)}\Gamma$ l'estimateur de **maximum de vraisemblance bayésien**Γest donné par le mode de la loi a posteriori de $\Lambda^{(n)}$:

$$\hat{\lambda}^{(n)}(X_1,\ldots,X_n) = Argmax_{(\lambda_1,\ldots,\lambda_n)\in\mathbb{R}_+^n}\,[\,f_{\Lambda^{(n)}|X^{(n)}}(\lambda_1,\ldots,\lambda_n)\,]. \tag{3.10}$$

Lois prédictives et estimation de la fiabilité

L'approche bayésienne permet de prédire le prochain temps inter-défaillances X_{n+1} à partir de sa loi prédictive :

Proposition – 3.12 *Dans la modélisation bayésienne exponentielle, la loi prédictive de la v.a.r. X_{n+1} est donnée par sa densité :*

$$f_{X_{n+1}|x^{(n)}}(x_{n+1}) = \int_{\mathbb{R}_+^{n+1}} \lambda_{n+1} e^{-\lambda_{n+1} x_{n+1}} \, f_{\Lambda^{(n+1)}|x^{(n)}}(\lambda_1, \ldots, \lambda_{n+1}) \, d\lambda_1 \ldots \lambda_{n+1} \qquad (3.11)$$

où la densité de la loi a posteriori du vecteur $\Lambda^{(n+1)}$ est donnée par la formule de Bayes :

$$f_{\Lambda^{(n+1)}|x^{(n)}}(\lambda_1, \ldots, \lambda_{n+1}) = \frac{\prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i}] \; \Pi_{n+1}(\lambda_1, \ldots, \lambda_{n+1})}{\int_{\mathbb{R}_+^{n+1}} \prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i}] \; \Pi_{n+1}(\lambda_1, \ldots, \lambda_{n+1}) d\lambda_1 \ldots d\lambda_{n+1}}. \qquad (3.12)$$

Preuve – La densité de la loi prédictive s'écrit :

$$\begin{aligned} f_{X_{n+1}|x^{(n)}}(x_{n+1}) &= \int_{\mathbb{R}_+} f_{X_{n+1}|x^{(n)};\lambda_{n+1}}(x_{n+1}) \, f_{\Lambda_{n+1}|x^{(n)}}(\lambda_{n+1}) \, d\lambda_{n+1} \\ &= \int_{\mathbb{R}_+} \lambda_{n+1} e^{-\lambda_{n+1} x_{n+1}} \, f_{\Lambda_{n+1}|x^{(n)}}(\lambda_{n+1}) \, d\lambda_{n+1} \end{aligned}$$

La loi a posteriori de la v.a.r. Λ_{n+1} est obtenue à partir de la loi a posteriori du vecteur $\Lambda^{(n+1)}$:

$$f_{\Lambda_{n+1}|x^{(n)}}(\lambda_{n+1}) = \int_{\mathbb{R}_+^{n}} f_{\Lambda^{(n+1)}|x^{(n)}}(\lambda_1, \ldots, \lambda_{n+1}) \, d\lambda_1 \ldots d\lambda_n$$

On a ainsi le résultat énoncé. □

La loi prédictive permet de prédire la fiabilité du logiciel étudié. On peut par exemple prédire le temps d'attente de la prochaine défaillance en utilisant l'espérance, le mode ou la médiane de la loi prédictive.
On peut en outre parler de *MTTF* a posteriori donné par :

$$E(X_{n+1} \mid x^{(n)}) = \int_{\mathbb{R}_+} x_{n+1} \, f_{X_{n+1}|x^{(n)}}(x_{n+1}) \, dx_{n+1}. \qquad (3.13)$$

On peut aussi utiliser la fonction de fiabilité prédictive qui s'exprime en fonction de la densité de la loi prédictive :

$$\begin{aligned} \forall \tau \geq 0 \; R_{n+1}(\tau) &= P(X_{n+1} \geq \tau \mid x^{(n)}) \\ &= \int_{\tau}^{+\infty} f_{X_{n+1}|x^{(n)}}(x) \, dx. \end{aligned} \qquad (3.14)$$

La prédiction bayésienne de la fiabilité peut aussi se faire en utilisant la loi a posteriori du prochain taux de défaillance Λ_{n+1}.
En effet si $\hat{\lambda}_{n+1}$ est un estimateur de λ_{n+1} on peut estimer la fonction fiabilité par :

$$\forall \tau \geq 0 \ R_{n+1}(\tau) \simeq exp(-\hat{\lambda}_{n+1}\,\tau) \tag{3.15}$$

On termine cette sous-section par quelques remarques sur le choix de la fonction de coût.

Remarques sur le choix de la fonction de coût

Lorsqu'on utilise les estimateurs de Bayes pour estimer les effets des corrections du logiciel c'est-à-dire les taux de défaillance λ_i il n'y a pas de raisons particulières incitant à prendre une fonction de coût spécifique.
On peut alors choisir la fonction de coût quadratique qui donne des estimateurs de Bayes dont les propriétés sont bien connues.

Par contre quand on souhaite estimer la fiabilité du logiciel à l'issue de la période de tests il peut être intéressant d'utiliser une fonction de coût dissymétrique.
Ceci permet d'introduire dans l'approche statistique des informations a priori sur les différentes conséquences des mauvaises estimations de la fiabilité.

Il est par exemple logique de supposer que le coût d'une surestimation de la fiabilité soit plus élevé que celui d'une sous-estimation.
En effet une sous-estimation engendre généralement des tests redondants et inutiles alors qu'une surestimation peut induire des conséquences graves pour l'utilisateur et peut ainsi nuire gravement à l'image de marque du concepteur.

On peut trouver dans un travail de Canfield [14] un exemple d'une telle fonction de coût dissymétrique adaptée aux problèmes d'analyse bayésienne de la fiabilité.

On se contentera dans la suite de ce travail d'utiliser pour les estimateurs de Bayes la fonction de coût quadratique.
Notons cependant qu'il pourrait être intéressant de généraliser les résultats et les méthodes numériques décrites dans la suite du chapitre au cas de fonctions coût dissymétriques.

3.4.4 Propriétés a priori des taux de défaillance

Pour affiner la modélisation générale présentée ci-dessus il faut préciser davantage la forme des connaissances a priori disponibles. Ceci permettra de mieux spécifier les propriétés a priori du processus aléatoire $\{\Lambda_i\}_{i\geq 1}$.

Un premier type de connaissances a priori est lié à l'idée que se font les experts de l'évolution de la fiabilité au cours du temps.

Il est par exemple naturel de supposer que les corrections sont globalement bénéfiques d'où une tendance globale à la croissance de fiabilité.

Cette connaissance est modélisée par une hypothèse a priori de décroissance stochastique des v.a.r. Λ_i. Ce qui peut se traduire par la propriété suivante :

$$\forall i \geq 0\ \Gamma \forall l \in \mathbb{R}_+\ \Gamma\ P(\Lambda_i \geq l) \geq P(\Lambda_{i+1} \geq l). \tag{3.16}$$

Des connaissances plus précises sur l'environnement de correction permettent ensuite de mieux spécifier le modèle de décroissance stochastique des v.a.r. Λ_i.
Certain modèles (cf. [67]Γ[66] et [21]) supposent par exemple que les Λ_i sont des v.a.r. indépendantes.

Il est cependant plus naturelΓcomme on va l'expliquer plus tardΓde supposer a priori que les v.a.r. Λ_i sont **markoviennes**.

Taux de défaillance indépendants

L'hypothèse a priori la plus simple consiste à supposer les v.a.r. Λ_i **indépendantes**.

Le lien entre les taux de défaillance λ_i est alors modélisé par une même forme paramétrique pour les densités $\Pi_i(\lambda_i)$ des v.a.r. Λ_i :

$$\forall i \geq 1\ \Gamma\ \ \Pi_i(\lambda_i) = \psi(\lambda_i, i, \theta).$$

La fonction ψ modélise dans ce cas les effets des corrections du logiciel.

Le paramètre vectoriel θ peut être connu ou non. Dans le dernier cas il peut être estimé :

- soit par des méthodes "fréquentistes" telles que la méthode du maximum de vraisemblanceΓon parle alors d'approche **bayésienne empiriqueΓ**
- soit par une nouvelle approche bayésienneΓon aura alors besoin d'une loi a priori sur θ et on parle dans ce cas d'approche **bayésienne hiérarchique**.

Des exemples de telles approches ont été présentés par Littlewood et Verrall [67] et Mazzuchi et Soyer [70].

Remarquons par ailleurs que si les v.a.r. Λ_i sont supposées indépendantesΓla densité de la loi a posteriori du vecteur $\Lambda^{(n)}$ s'écrit :

$$f_{\Lambda^{(n)}|x^{(n)}}(\lambda_1, \ldots, \lambda_n) = \frac{\prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i)\,]}{\prod_{i=1}^{n} [\int_{\mathbb{R}_+}\ \lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i)\, d\lambda_i\,]}. \tag{3.17}$$

La densité de la loi a posteriori de la v.a.r. Λ_i est donc :

$$f_{\Lambda_i|x^{(n)}}(\lambda_i) = \frac{\lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i)}{\int_{\mathbb{R}_+}\ \lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i)\, d\lambda_i} \tag{3.18}$$

ce qui prouve que :

- conditionnellement à $X^{(n)}\Gamma$les v.a.r. $\Lambda_1, \ldots, \Lambda_n$ sont indépendantes entre ellesΓ
- conditionnellememt à $X_i\Gamma$la v.a.r. Λ_i est indépendante des v.a.r. X_j pour $j \neq i$.

L'hypothèse d'indépendance des Λ_i est cependant peu justifiée dans le contexte de la Fiabilité des Logiciels.
On lui préfère l'hypothèse selon laquelle les v.a.r. Λ_i sont markoviennes.

Taux de défaillance markoviens

Les paramètres λ_i décrivent l'évolution de la fiabilité du logiciel étudié. Il est donc assez naturel de supposer a priori que les v.a.r. Λ_i associées sont markoviennesΓ c'est-à-dire que pour tout $i\Gamma$la v.a.r. Λ_i estΓconditionnellement à $\Lambda_{i-1}\Gamma$indépendante des v.a.r. $\Lambda_{i-2}, \ldots, \Lambda_1$.

Cette hypothèse résulte du fait que l'état d'un logiciel après sa $i^{ème}$ correction est une transformationΓvia cette correctionΓde l'état du logiciel après la $(i-1)^{ème}$ correction.

L'état présent du logiciel ne dépend ainsi de son passé qu'à travers son état précédant la toute dernière correction. Ceci se traduit naturellement par une hypothèse **markovienne** sur les v.a.r. Λ_i.

La densité de la loi a priori du vecteur $\Lambda^{(n)}$ s'écrit alors :

$$\Pi_n(\lambda_1, \ldots, \lambda_n) = \Pi_n(\lambda_n \mid \lambda_{n-1}) \ldots \Pi_2(\lambda_2 \mid \lambda_1)\, \Pi_1(\lambda_1). \tag{3.19}$$

Le modèle d'évolution des v.a.r. Λ_i est alors entièrement déterminé par la donnée des densités : $\Pi_i(\lambda_i \mid \lambda_{i-1})$.

Les différents estimateurs bayésiens présentés dans la sous-section 3.4.3 s'écrivent alors plus simplement en fonction des densités $\Pi_i(\lambda_i \mid \lambda_{i-1})$. Ceci sera précisé dans la section suivante.

3.5 Modélisation exponentielle à taux de défaillance markoviens

On donne dans cette section les expressions des estimateurs bayésiens des différents attributs de la fiabilité dans le cas où les v.a.r. taux de défaillance sont markoviennes.

On présente ensuite différents exemples d'hypothèses a priori markoviennes. Pour chacun de ces exemples on implémente des méthodes numériques permettant de calculer les estimations bayésiennes des différents attributs de la fiabilité.

On présente à la fin de la section des exemples d'utilisation de cette approche bayésienne sur quelques jeux de données simulés.

3.5.1 Introduction et hypothèses du modèle

On a vu dans la section précédente que le contexte général de la Fiabilité des Logiciels justifie le choix d'une hypothèse a priori markovienne pour les v.a.r. Λ_i.

En restant dans un cadre très général on aboutit ainsi à la modélisation suivante :

Définition – 3.13 *On appelle* ***modélisation bayésienne exponentielle à taux de défaillance markoviens*** *(*BEM*) la modélisation générale où :*

- *les v.a.r. X_i sont de lois exponentielles de paramètres Λ_i* ***aléatoires*** *:*

$$\forall i \geq 1 \ , \ X_i \sim Exp(\Lambda_i)$$

$\boldsymbol{H_{BEM1}}$

- *conditionnellement à $\{\Lambda_i\}_{i\geq 1}$, les v.a.r. X_i sont indépendantes entre elles.*

$\boldsymbol{H_{BEM2}}$

- *Le processus $\{\Lambda_i\}_{i\geq 1}$ est un processus de* ***Markov****, sa loi a priori est donnée par la suite des densités :* $\Pi_i(\lambda_i \mid \lambda_{i-1})$.

$\boldsymbol{H_{BEM3}}$

Remarque – Un certain nombre de modèles bayésiens peuvent être considérés comme cas particuliers de la modélisation *BEM* définie ci-dessus.
On peut par exemple citer le modèle de Littlewood et Verrall [67]Γle modèle de Becker et Camaranipoulos [8] ainsi que la version bayésienne du modèle *Jelinski-Moranda* [66].

On a montré dans la section précédente que les hypothèses $H_{BEM1}\Gamma H_{BEM2}$ et H_{BEM3} de la définition ci-dessus sont des hypothèses naturelles dans le contexte de la Fiabilité des Logiciels.

En se plaçant dans le cadre de la modélisation *Profil Opérationnel Poissonnien Homogène* de la sous-section 1.4.3Γon peut donner une autre justification aux trois hypothèses précédentes. Ceci est expliqué ci-dessous.

L'approche Filtrage Optimal

Les hypothèses H_{BEM1}ΠH_{BEM2} et H_{BEM3} peuvent être obtenues en considérant une approche tout à fait différente de l'approche adoptée ici.

On peut en effet se placer dans le cadre de la modélisation proposée par Gaudoin et Soler [39] (cf. sous-section 1.4.3) où le profil opérationnel est modélisé par un *Profil Opérationnel Poissonnien Homogène* (*POPH*).

Soler [94] montre alors (cf. théorème 1.19) que si l'on suppose que les corrections sont de durées négligeables et qu'elles suivent immédiatement les défaillancesΓon aboutit exactement aux trois hypothèses H_{BEM1}ΠH_{BEM2} et H_{BEM3}.

L'approche *POPH* conduit alorsΓnon pas à une analyse bayésienneΓmais à un **modèle de Filtrage Optimal** discret non linéaire où le vecteur des observations est constitué des v.a.r. X_i. Les variables d'état sont les v.a.r. taux de défaillance Λ_i.

Dans ce modèle de FiltrageΓles équations des observations sont :

$$\forall i \leq n \ \Gamma \ \text{conditionnellement à } \Lambda_i = \lambda_i \text{ on a } \ X_i \sim Exp(\lambda_i). \tag{3.20}$$

Les équations décrivant l'évolution du système sont :

$$\forall i \leq n \ \Gamma \ \text{conditionnellement à } \Lambda_{i-1} = \lambda_{i-1} \ \Gamma \ \Lambda_i \sim \Pi_i(\lambda_i \,|\, \lambda_{i-1}). \tag{3.21}$$

Comme dans l'approche bayésienneΓ$\Pi_i(\lambda_i \,|\, \lambda_{i-1})$ désigne ici aussi la densité de la loi de probabilité de la v.a.r. Λ_i conditionnellement à $\Lambda_{i-1} = \lambda_{i-1}$.

L'estimation ou la prédiction des variables d'état au vu des observations se fait ensuite par l'utilisation de la formule de Bayes (cf. par exemple Jazwinski [46]).

L'estimation des paramètres λ_i dans la modélisation bayésienne exponentielleΓet le filtrage et la prédiction des variables Λ_i au vu des observations des v.a.r. X_i dans l'approche Filtrage s'effectuent en utilisant les mêmes outils.
On a en effet deux justifications et deux terminologies différentes pour un même modèle.

On se placera dans la suite de ce chapitre dans le cadre de la modélisation bayésienne exponentielle.

Remarque – Notons que la théorie du Filtrage Optimal a déjà été utilisée pour l'évaluation de la fiabilité des logiciels.

Singpurwalla et Soyer [89] supposent par exemple que les temps inter-défaillances sont des v.a.r. de lois log-normalesΓils obtiennent ainsi un modèle de filtrage gaussien.
Chen et Singpurwalla [16] choisissent des lois *Gamma* pour les v.a.r. X_i et des lois *Beta* pour les variables d'état Λ_i. Ils obtiennent ainsi un modèle de filtrage non gaussien pour lequel les estimateurs a posteriori ont des expressions explicites.

3.5.2 Evaluation bayésienne de la fiabilité

Les expressions des estimateurs bayésiens des différents attributs de la fiabilité données dans la sous-section 3.4.3 se simplifient lorsqu'on ajoute l'hypothèse markovienne H_{BEM3}. Ces simplifications sont décrites ci-dessous.

Estimation des taux de défaillance

Sous l'hypothèse H_{BEM3} la densité de la loi a priori du vecteur $\Lambda^{(n)}$ s'écrit

$$\Pi_n(\lambda_1, \ldots, \lambda_n) = \Pi_n(\lambda_n \mid \lambda_{n-1}) \ldots \Pi_2(\lambda_2 \mid \lambda_1)\, \Pi_1(\lambda_1). \tag{3.22}$$

Notation – Par abus de notationΓla densité de la loi a priori de la v.a.r. Λ_1 sera notée $\Pi_1(\lambda_1 \mid \lambda_0)$Γc'est-à-dire :

$$\Pi_1(\lambda_1 \mid \lambda_0) \equiv \Pi_1(\lambda_1).$$

La formule (3.6) de la densité de la loi a posteriori de $\Lambda^{(n)}$ se réécrit :

$$f_{\Lambda^{(n)}|x^{(n)}}(\lambda_1, \ldots, \lambda_n) = \frac{\prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i \mid \lambda_{i-1})\,]}{\int_{\mathbb{R}_+^n} \prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i \mid \lambda_{i-1})\,]\, d\lambda_1 \ldots d\lambda_n}. \tag{3.23}$$

Les densités des lois a posteriori marginales des v.a.r. Λ_j sont données pour tout $j \leq n$ par :

$$f_{\Lambda_j|x^{(n)}}(\lambda_j) = \int_{\mathbb{R}_+^{n-1}} f_{\Lambda^{(n)}|x^{(n)}}(\lambda_1, \ldots, \lambda_n)\, d\lambda_1 \ldots d\lambda_{j-1} d\lambda_{j+1} \ldots d\lambda_n. \tag{3.24}$$

On en déduit les expressions des estimateurs de Bayes des paramètres taux de défaillances $\lambda_1, \ldots, \lambda_n$:

$$\hat{\lambda}_j(X_1, \ldots, X_n) = \frac{\int_{\mathbb{R}_+^n} \lambda_j \prod_{i=1}^{n} [\lambda_i e^{-\lambda_i X_i}\, \Pi_i(\lambda_i \mid \lambda_{i-1})\,]\, d\lambda_1 \ldots d\lambda_n}{\int_{\mathbb{R}_+^n} \prod_{i=1}^{n} [\lambda_i e^{-\lambda_i X_i}\, \Pi_i(\lambda_i \mid \lambda_{i-1})\,]\, d\lambda_1 \ldots d\lambda_n}. \tag{3.25}$$

Prédiction du taux de défaillance λ_{n+1}

Pour prédire le taux de défaillance futur λ_{n+1} on doit réécrire la formule (3.12) de la densité de la loi a posteriori de Λ_{n+1}. Pour ce faire on utilise le résultat suivant :

Proposition – 3.14 *Dans une modélisation bayésienne exponentielle markovienne, la v.a.r. Λ_{n+1} est, conditionnellement à Λ_n, indépendante des v.a.r. $X_1, \ldots, X_n$. On a donc :*

$$f_{\Lambda_{n+1}|\lambda_n, x^{(n)}}(\lambda_{n+1}) = \Pi_{n+1}(\lambda_{n+1} \mid \lambda_n). \tag{3.26}$$

Preuve – Ce résultat résulte directement des hypothèses H_{BEM1} et H_{BEM3}.
En effet si on pose :

$$Y_1 \equiv (\Lambda_1, \ldots, \Lambda_{n-1}) \Gamma Y_2 \equiv \Lambda_n \Gamma Y_3 \equiv \Lambda_{n+1} \text{ et } Z \equiv X^{(n)},$$

les hypothèses H_{BEM1} et H_{BEM3} impliquent que :

- conditionnellement à $Y_2 \Gamma Y_3$ est indépendante de $Y_1 \Gamma$
- conditionnellement à $(Y_1, Y_2) \Gamma Z$ est indépendante de Y_3.

En considérant les densités des variables aléatoires $Y_1 \Gamma Y_2 \Gamma Y_3$ et Z par rapport aux mesures de Lebesgue associéesΓil résulte des deux hypothèses précédentes que :

$$f_{Y_1,Y_2,Y_3,Z}(y_1, y_2, y_3, z) = f_{Y_1,Y_2}(y_1, y_2)\, f_{Y_3|y_2}(y_3)\, f_{Z|y_1,y_2}(z)$$

en intégrant les deux termes de l'équation précédente par rapport à y_1 on obtient :

$$f_{Y_2,Y_3,Z}(y_2, y_3, z) = f_{Y_3|y_2}(y_3) \int_{\mathbb{R}_+^{n-1}} f_{Y_1,Y_2}(y_1, y_2)\, f_{Z|y_1,y_2}(z)\, dy_1$$

et en intégrant par rapport à y_3 :

$$f_{Y_2,Z}(y_2, z) = \int_{\mathbb{R}_+^{n-1}} f_{Y_1,Y_2}(y_1, y_2)\, f_{Z|y_1,y_2}(z)\, dy_1.$$

Finalement on a :

$$\begin{aligned} f_{Y_3|y_2,z}(y_3) &= \frac{f_{Y_2,Y_3,Z}(y_2, y_3, z)}{f_{Y_2,Z}(y_2, z)} \\ &= f_{Y_3|y_2}(y_3) \end{aligned}$$

d'où le résultat énoncé. □

La proposition précédente donne la loi a posteriori de la v.a.r. Λ_{n+1} :

Proposition – 3.15 *La loi a posteriori du taux de défaillance Λ_{n+1} est donnée par sa densité :*

$$f_{\Lambda_{n+1}|x^{(n)}}(\lambda_{n+1}) = \frac{\int_{\mathbb{R}_+^n} \Pi_{n+1}(\lambda_{n+1}\,|\,\lambda_n)\,\prod_{i=1}^n [\lambda_i e^{-\lambda_i x_i}\,\Pi_i(\lambda_i\,|\,\lambda_{i-1})]\,d\lambda_1\ldots d\lambda_n}{\int_{\mathbb{R}_+^n}\,\prod_{i=1}^n [\lambda_i e^{-\lambda_i x_i}\,\Pi_i(\lambda_i\,|\,\lambda_{i-1})]\,d\lambda_1\ldots d\lambda_n} \tag{3.27}$$

Preuve – Le résultat découle directement de l'écriture de la densité de la loi a posteriori de Λ_{n+1} sous la forme suivante :

$$f_{\Lambda_{n+1}|x^{(n)}}(\lambda_{n+1}) = \int_{\mathbb{R}_+} f_{\Lambda_{n+1}|x^{(n)},\lambda_n}(\lambda_{n+1})\, f_{\Lambda_n|x^{(n)}}(\lambda_n)\,d\lambda_n$$

□

On peut alors prédire le taux de défaillance λ_{n+1} après la $n^{ème}$ correction en prenant par exemple l'espérance a posteriori de Λ_{n+1} :

$$\hat{\lambda}_{n+1}(x_1,\ldots,x_n) = \int_{\mathbb{R}_+} \lambda_{n+1}\, f_{\Lambda_{n+1}|x^{(n)}}(\lambda_{n+1})\,d\lambda_{n+1}.$$

Loi prédictive du prochain temps inter-défaillances

Comme on l'a vu dans la sous-section 3.4.3Γl'approche bayésienne permet de prédire le prochain temps inter-défaillances en utilisant la loi prédictive de X_{n+1}Γc'est-à-dire sa loi de probabilité conditionnellement à $X^{(n)}=x^{(n)}$.

Proposition – 3.16 *Lorsqu'on suppose que les taux de défaillance sont markoviens, la densité de la loi prédicitive de X_{n+1} s'écrit :*

$$f_{X_{n+1}|x^{(n)}}(x_{n+1}) = \frac{\int_{\mathbb{R}_+^{n+1}}\,\prod_{i=1}^{n+1} [\lambda_i e^{-\lambda_i x_i}\,\Pi_i(\lambda_i\,|\,\lambda_{i-1})]\,d\lambda_1\ldots d\lambda_{n+1}}{\int_{\mathbb{R}_+^n}\,\prod_{i=1}^n [\lambda_i e^{-\lambda_i x_i}\,\Pi_i(\lambda_i\,|\,\lambda_{i-1})]\,d\lambda_1\ldots d\lambda_n}. \tag{3.28}$$

Preuve – Cette densité est donnée par :

$$f_{X_{n+1}|x^{(n)}}(x_{n+1}) = \int_{\mathbb{R}_+} f_{X_{n+1}|x^{(n)},\lambda_{n+1}}(x_{n+1})\, f_{\Lambda_{n+1}|x^{(n)}}(\lambda_{n+1})\,d\lambda_{n+1}.$$

Les hypothèses H_{BEM1} et H_{BEM2} permettent d'écrire :

$$f_{X_{n+1}|x^{(n)},\lambda_{n+1}}(x_{n+1}) = \lambda_{n+1} e^{-\lambda_{n+1}x_{n+1}}$$

on utilise enfin la formule (3.27) de la densité de la loi a posteriori de Λ_{n+1} pour obtenir le résultat énoncé.

□

A partir de la densité de la loi prédictive on obtient l'expression du *MTTF* a posteriori :

Proposition – 3.17 *Dans la modélisation bayésienne exponentielle à taux de défaillance markoviens, le* MTTF *a posteriori est donné par :*

$$E(X_{n+1} \mid x^{(n)}) = \frac{\int_{\mathbb{R}_+^{n+1}} \lambda_{n+1}^{-1} \Pi_{n+1}(\lambda_{n+1} \mid \lambda_n) \prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i} \Pi_i(\lambda_i \mid \lambda_{i-1})]\, d\lambda_1 \ldots d\lambda_{n+1}}{\int_{\mathbb{R}_+^n} \prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i} \Pi_i(\lambda_i \mid \lambda_{i-1})]\, d\lambda_1 \ldots d\lambda_n}. \tag{3.29}$$

Avant de présenter quelques méthodes numériques permettant de calculer les différentes estimations bayésiennes présentées ci-dessusΓon donne dans les deux sous-sections suivantes des exemples d'a priori markoviens particuliers permettant d'affiner l'analyse générale précédente.

3.5.3 Exemples d'a priori sur les effets des corrections

En utilisant des connaissances plus précises sur l'environnement de correctionΓon peut spécifier davantage les hypothèses a priori sur les v.a.r. Λ_i.

Dans les exemples donnés ci-dessousΓon utilise le principe de **maximum d'entropie** (cf. Kapur [51])Γqu'on présente brièvement ci-dessousΓpour traduire les différentes connaissances a priori des experts en lois de probabilité sur les v.a.r. Λ_i.

Le principe de maximum d'entropie : rappels

On considère une v.a.r. Y à valeurs dans $D_y \subset \mathbb{R}$. Soit f la densité de Y par rapport à la mesure de Lebesgue.

Définition – 3.18 *On appelle* ***fonction d'entropie*** *de Y la quantité :*

$$H(f) = -\int_{D_y} f(y)\, ln\, [f(y)\,]\, dy.$$

La fonction d'entropie est une mesure de la quantité d'incertitude sur Y.

Supposons que l'on dispose de certaines connaissances a priori sur la v.a.r. Y décrites par les égalités suivantes :

$$\forall r \leq m\Gamma \int_{D_y} f(y) g_r(y)\, dy = \bar{g}_r.$$

où :

- $(g_r)_{r\leq m}$ est une suite de fonctions réelles connuesΓ
- $(\bar{g}_r)_{r\leq m}$ est une suite de constantes connues.

Le principe de maximum d'entropie énonce alors que la loi de probabilité la plus vraisemblable pour Y est la loi de probabilité dont la densité maximise l'entropie $H(f)$ sous les contraintes suivantes :

$$\begin{cases} \displaystyle\int_{D_y} f(y)\,dy = 1. \\ \\ \forall r \leq m\Gamma \displaystyle\int_{D_y} f(y) g_r(y)\,dy = \bar{g}_r. \end{cases}$$

Exemples –

1. Si on connaît a priori la moyenne m et la variance σ^2 d'une v.a.r. $Y\Gamma$alors la loi de maximum d'entropie de Y est la loi normale : $\mathcal{N}(m, \sigma^2)$.
2. Si on connaît a priori que $Y \geq 0$ et que $E(Y) = m\Gamma$ alors la loi de maximum d'entropie de Y est la loi exponentielle : $Exp(m)$.

Une présentation détaillée du principe de maximum d'entropie et de ses applications peut être trouvée dans [51].

A priori exponentiels

Supposons que l'on sache a priori que les différentes corrections du logiciel vont avoir en moyenne un même effet sur l'évolution de la fiabilité.
Cette information peut être modélisée comme suit :

$$\forall i \geq 1\ \Gamma\ E(\Lambda_i \,|\, \lambda_{i-1}) = g(\lambda_{i-1}) \tag{3.30}$$

où g est une fonction réelle modélisant l'effet moyen d'une correction.

En ne tenant compte que de cette informationΓles lois a priori les plus "vraisemblables" au sens du principe de maximum d'entropie pour les v.a.r. Λ_i sont données pour tout entier i par :

$$\text{Conditionnellement à } \Lambda_{i-1} = \lambda_{i-1}\ \Gamma\ \Lambda_i \sim Exp(1/g(\lambda_{i-1})).$$

Dans cette modélisationΓune croissance de fiabilité s'exprime par la propriété "$g(x) \leq x$".

Un cas particulier d'a priori exponentiels correspond au cas où :

$$g(x) = e^{-\theta} \cdot x$$

le paramètre θ représente alorsΓcomme dans le modèle $MPD\Gamma$l'effet moyen d'une correction.

A priori uniformes

L'effet d'une correction peut être mesuré par la proportion de fautes éliminées (bonne correction) ou de fautes ajoutées (mauvaise correction).

On peut par exemple savoir a priori que :

- une bonne correction enlève dans le meilleur des cas $100.\alpha\%$ des fautes initialesΓ
- une mauvaise correction ajoute dans le pire des cas $100.\beta\%$ des fautes.

Cette information peut être modélisée par le fait queΓconditionnellement à $\Lambda_{i-1} = \lambda_{i-1}$ on a :

$$\Lambda_i \in [(1-\alpha)\,\lambda_{i-1}, (1+\beta)\,\lambda_{i-1}]. \tag{3.31}$$

La propriété précédente se traduitΓen utilisant le principe de maximum d'entropie par le choix de lois a priori uniformes pour les v.a.r. Λ_i.
Les lois a priori des Λ_i sont donc donnéesΓpour tout entier positif iΓpar :

$$\text{Conditionnellement à } \Lambda_{i-1}=\lambda_{i-1}\ \Gamma\ \Lambda_i \sim Unif[(1-\alpha)\lambda_{i-1}, (1+\beta)\lambda_{i-1}].$$

A priori log-normaux

L'effet d'une correction peut être modéliséΓcomme le suggèrent Gaudoin et al [38] (cf. définition 1.22)Γpar une décroissance géométrique des taux de défaillance :

$$\forall i > 1\Gamma\ \Lambda_i = e^{-\Theta_i}\,\Lambda_{i-1}$$

où les v.a.r. Θ_i représentent les effets des corrections successives.

Si on connaît a priori que les effets des corrections fluctuent autour d'un effet moyen θ avec une variance σ^2 (variance due à la fatigue des correcteursΓaux périodes de vacancesΓ aux primesΓetc.)Γc'est-à-dire si on sait a priori que :

$$\forall i > 1\ \Gamma\ E(\Theta_i) = \theta \ \text{ et } \ Var(\Theta_i) = \sigma^2 \tag{3.32}$$

les lois de maximum d'entropie des v.a.r. Θ_i sont alors des lois normales :

$$\forall i > 1\Gamma\ \Theta_i \sim \mathcal{N}(\theta, \sigma^2).$$

Par conséquentΓles lois a priori des v.a.r. Λ_i sont donnéesΓpout tout entier i par :

$$\text{Conditionnellement à } \Lambda_{i-1}=\lambda_{i-1} \Gamma \Lambda_i \sim \text{log-normale}(ln(\lambda_{i-1}) - \theta, \sigma^2).$$

Gaudoin et al [38] proposent un traitement non bayésien du modèle log-normal décrit ci-dessus.

Remarque – Dans tous les exemples cités ci-dessusΓles connaissances a priori doivent pouvoir permettre de préciser les valeurs des constantes intervenant dans les différentes lois a priori : $\theta\Gamma\lambda_0\Gamma\alpha\Gamma\beta\Gamma\sigma^2\Gamma$etc.

Avant de préciser les outils numériques permettant d'implémenter l'approche bayésienne markovienne et d'utiliser les a priori décrits ci-dessusΓon présente dans la sous-section suivante un quatrième exemple d'a priori où les v.a.r. Λ_i sont supposées à accroissements indépendants.
Ce dernier exemple conduit à des estimateurs bayésiens ayant des formules analytiques explicites.

3.5.4 Cas particulier : taux de défaillance à accroissements indépendants

Hypothèse – Si on suppose que les corrections ont des effets additifs sur les taux de défaillance et que ces effets sont indépendants entre euxΓon obtient un modèle où le processus aléatoire $\{\Lambda_i\}_{i\geq 1}$ est un processus à **accroissements indépendants**Γc'est-à-dire que les v.a.r. $\Lambda_i - \Lambda_{i-1}$ sont indépendantes entre elles. –

On montre ci-dessous que sous l'hypothèse précédenteΓles estimateurs bayésiens des taux de défaillance λ_i sont obtenus sous une forme explicite.

Notations – Après observation des n premières défaillancesΓon note :

$$\forall i \leq n-1 \Gamma U_i \equiv \Lambda_i - \Lambda_{i+1}$$

On prend par convention $U_n \equiv \Lambda_n$ de façon à avoir :

$$\forall i \leq n \Gamma \Lambda_i = \sum_{j=i}^{n} U_j. \tag{3.33}$$

Les v.a.r. U_i représentant les effets des corrections sont donc supposées indépendantes.

Les connaissances a priori sur les effets des corrections sont traduitesΓdans cette sous-sectionΓpar des lois a priori sur les v.a.r. U_i.

Notations – Dans cette sous-section Π_i désignera la densité de la loi a priori de la v.a.r. U_i.

Remarque – Le vecteur $\Lambda^{(n)}$ est par exemple à accroissements indépendants s'il est composé de n v.a.r. indépendantes rangées dans l'ordre décroissant.

Lois du vecteur $X^{(n)}$

En utilisant la formule (3.4) on écrit la densité de la loi de probabilité de $X^{(n)}$ conditionnellement à $\{U_i = u_i\}_{i \leq n}$ sous la forme :

$$\begin{aligned} f_{X^{(n)}|u^{(n)}}(x_1, \ldots, x_n) &= \prod_{i=1}^{n} \left[(u_i + \ldots + u_n)\, e^{-(u_i + \ldots + u_n)\, x_i} \right] \\ &= \prod_{i=1}^{n} \left[(u_i + \ldots + u_n)\, e^{-u_i\, (x_1 + \ldots + x_i)} \right]. \end{aligned} \tag{3.34}$$

OrΓon a pour tout n-uplet $(u_1, \ldots, u_n)$ de $\mathbb{R}^n$:

$$\prod_{i=1}^{n} (u_i + \ldots + u_n) = \sum_{(\alpha_1, \ldots, \alpha_n) \in A} u_1^{\alpha_1} \ldots u_n^{\alpha_n}$$

où l'ensemble A est décrit ci-dessous :

Notation –

1. A est l'ensemble des n-uplets d'entiers positifs $\alpha \equiv (\alpha_1, \ldots, \alpha_n)$ vérifiant : $\sum_{i=1}^{n} \alpha_i = n$ et identifiable à l'ensemble des $\boldsymbol{n!}$ chemins de l'arbre marqué représenté sur la figure 3.1.
2. Pout tout α dans $A\Gamma\alpha_i(\alpha)$ désigne la $i^{ème}$ composante du n-uplet α.
3. Rappelons que pour tout $i \geq 1$ on note :

$$T_i \equiv \sum_{j=1}^{i} X_j \quad \text{et} \quad t_i \equiv \sum_{j=1}^{i} x_j.$$

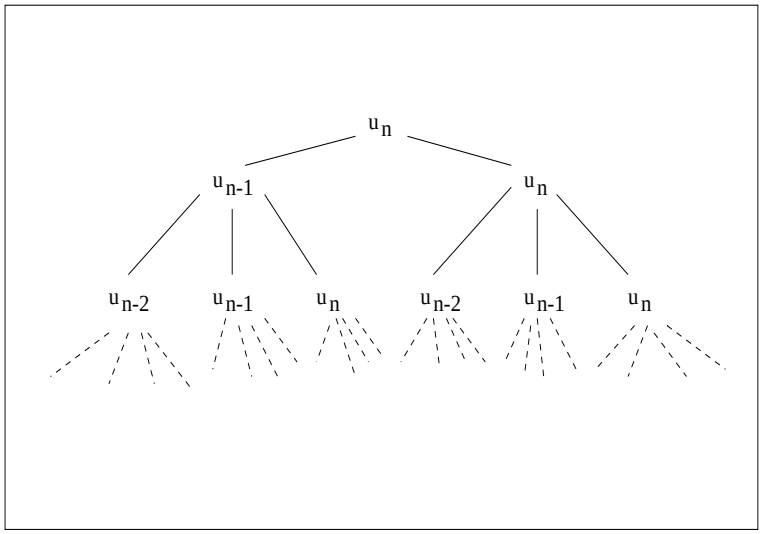

FIG. 3.1: Arbre dont les chemins (du sommet à une feuille) sont identifiables à tous les éléments α de A.

En utilisant les notations et résultats précédentsΓ la densité de la loi de probabilité de $X^{(n)}$ conditionnellement à $\{U_i = u_i\}_{i \leq n}$ s'écrit :

$$f_{X^{(n)}|u^{(n)}}(x_1, \ldots, x_n) = \sum_{\alpha \in A} \prod_{i=1}^{n} u_i^{\alpha_i(\alpha)} \, e^{-u_i t_i}. \tag{3.35}$$

En utilisant la formule précédente on montre le résultat suivant :

Proposition – 3.19 *La densité de la loi de probabilité du vecteur $X^{(n)}$ s'écrit :*

$$f_{X^{(n)}}(x_1, \ldots, x_n) = \sum_{\alpha \in A} \prod_{i=1}^{n} [\, a_i(\alpha_i(\alpha), t_i) \,]. \tag{3.36}$$

où pour tout α dans A et pout $i \leq n$ on note :

$$a_i(\alpha_i(\alpha), t_i) \equiv \int_{\mathbb{R}_+} u_i^{\alpha_i(\alpha)} \, e^{-u_i t_i} \, \Pi_i(u_i) \, du_i.$$

Remarque – Le résultat précédent permet aussi d'avoir la densité de la loi de probabilité du vecteur $(T_1, \ldots, T_n)$ sur le cône croissant de $\mathbb{R}_+^n$.

Estimateurs bayésiens des paramètres λ_i

On déduit ici les estimateurs de Bayes des paramètres λ_i à partir des estimateurs des paramètres $u_i = \lambda_i - \lambda_{i+1}$.

La loi a posteriori du vecteur $U^{(n)}$ est donnée par sa densité :

$$\begin{aligned} f_{U^{(n)}|t^{(n)}}(u_1,\ldots,u_n) &= \frac{f_{T^{(n)}|u^{(n)}}(t_1,\ldots,t_n)\ \prod_{i=1}^{n}\Pi_i(u_i)}{f_{T^{(n)}}(t_1,\ldots,t_n)} \\ &= \sum_{\alpha\in A}\frac{\prod_{i=1}^{n}[\,u_i^{\alpha_i(\alpha)}\,e^{-u_i t_i}\,\Pi_i(u_i)\,]}{\sum_{\alpha\in A}\prod_{i=1}^{n}[\,a_i(\alpha_i(\alpha),t_i)\,]}. \end{aligned} \tag{3.37}$$

Notations –

1. Pour tout α dans A et pour tout entier positif iΓon note G_i^α la loi de probabilité donnée par sa densité :

$$\forall u_i \in \mathbb{R}\ \Gamma\ g_i^\alpha(u_i;t_i) = \frac{u_i^{\alpha_i(\alpha)}\,e^{-u_i t_i}\,\Pi_i(u_i)}{a_i(\alpha_i(\alpha),t_i)}.$$

2. Pour tout α dans A on note :

$$p_\alpha(t_1,\ldots,t_n) = \frac{\prod_{i=1}^{n}a_i(\alpha_i(\alpha),t_i)}{\sum_{\alpha\in A}\prod_{i=1}^{n}[\,a_i(\alpha_i(\alpha),t_i)\,]}$$

En utilisant les notations précédentesΓla densité de la loi a posteriori de $U^{(n)}$ s'écrit :

$$f_{U^{(n)}|t^{(n)}}(u_1,\ldots,u_n) = \sum_{\alpha\in A}[\,p_\alpha(t_1,\ldots,t_n)\ \prod_{i=1}^{n}g_i^\alpha(u_i;t_i)\,] \tag{3.38}$$

la loi précédente est donc un mélange des lois produits données par l'ensemble :

$$\{\otimes_{i=1}^{n}G_i^\alpha\ \Gamma\ \alpha\in A\}$$

On peut ainsi en déduireΓpour $i \le n$Γla loi a posteriori de la v.a.r. U_i :

$$\sum_{\alpha\in A}p_\alpha(t_1,\ldots,t_n)G_i^\alpha$$

cette loi a pour densité :

$$\forall u_i \in \mathbb{R}\ \Gamma\ f_{U_i|t^{(n)}}(u_i) = \sum_{\alpha\in A}p_\alpha(t_1,\ldots,t_n)g_i^\alpha(u_i;t_i). \tag{3.39}$$

Proposition – 3.20 *Les estimateurs de Bayes (coût quadratique) des paramètres* $u_i = \lambda_i - \lambda_{i+1}$ *sont donnés par :*

$$\hat{u}_i(T_1, \ldots, T_n) = \sum_{\alpha \in A} p_\alpha(T_1, \ldots, T_n) \frac{a_i(\alpha_i(\alpha)+1, T_i)}{a_i(\alpha_i(\alpha), T_i)}. \tag{3.40}$$

Preuve – En utilisant la formule (3.39) on obtient pour $i \leq n$:

$$\begin{aligned} \hat{u}_i(T_1, \ldots, T_n) &= E(U_i \mid T^{(n)}) \\ &= \sum_{\alpha \in A} p_\alpha(T_1, \ldots, T_n) \int_{\mathbb{R}} u_i g_i^\alpha(u_i; t_i)\, du_i \\ &= \sum_{\alpha \in A} p_\alpha(T_1, \ldots, T_n) \frac{\int_{\mathbb{R}} u_i^{\alpha_i(\alpha)+1} e^{-u_i T_i} \Pi_i(u_i)\, du_i}{a_i(\alpha_i(\alpha), T_i)} \end{aligned} \tag{3.41}$$

d'où le résultat énoncé.

□

Les estimateurs des paramètres λ_i se déduisent des estimateurs $\hat{u}_i(T_1, \ldots, T_n)$ par les relations suivantes :

$$\forall i \leq n \ \Gamma \ \hat{\lambda}_i(X_1, \ldots, X_n) = \sum_{j=i}^{n} \hat{u}_j(T_1, \ldots, T_n). \tag{3.42}$$

Exemple

Supposons que la connaissance a priori disponible peut se traduire par le fait que le vecteur $\Lambda^{(n)}$ a la même loi qu'un vecteur constitué de n v.a.r. indépendantes de loi $Exp(1)$ (ou plus généralement $Exp(\delta)$) rangés dans l'ordre décroissant.

Sous cette hypothèseΓles v.a.r. $iU_i = i(\Lambda_i - \Lambda_{i+1})$ sont indépendantes de loi $Exp(1)$. On a par conséquent :

$$\forall i \leq n \ \Gamma \ U_i \sim Exp(i).$$

On a donc pour cet exemple : $\Pi_i(u_i) = ie^{-iu_i}$. Les fonctions a_i (cf. proposition 3.19) s'écriventΓpour $i \leq n$ et α dans A :

$$a(\alpha_i(\alpha), t_i) = \frac{i\Gamma(\alpha_i(\alpha)+1)}{(i+t_i)^{\alpha_i(\alpha)+1}}.$$

Les lois G_i^α sont donc ici des lois *Gamma* :

$$G_i^{\alpha} = Gamma(\alpha_i(\alpha)+1, i+t_i).$$

Les lois a posteriori des v.a.r. U_i sont ainsi des mélanges de lois *Gamma*.

Les estimateurs de Bayes des paramètres $u_i = \lambda_i - \lambda_{i+1}$ (cf. formule (3.41)) s'écrivent alors :

$$\hat{u}_j(T_1, \ldots, T_n) = \sum_{\alpha \in A} \frac{\prod_{i=1}^{n} \frac{\alpha_i(\alpha)!}{(i+T_i)^{\alpha_i(\alpha)}}}{\sum_{\alpha \in A} \prod_{i=1}^{n} [\, \frac{\alpha_i(\alpha)!}{(i+T_i)^{\alpha_i(\alpha)}} \,]} \tag{3.43}$$

Remarque – Le calcul des estimations données par la formule (3.43) nécessite le parcours de tous les $n!$ éléments de l'ensemble A.
Ceci représente un grand handicap pour l'exploitation des résultats précédents.
En pratique il devient impossible de calculer les estimations $\hat{u}_j(t_1, \ldots, t_n)$ dès que le nombre d'observations n dépasse la dizaine.

Conclusions

Le cas particulier où les Λ_i sont à accroissements indépendants a servi à montrer les difficultés rencontrées dès que l'on essaye de trouver des formules explicites des différents estimateurs bayésiens.

Au lieu d'adapter les connaissances a priori pour obtenir des estimateurs explicitesΓil est plus intéressantΓcomme on le verra dans la suiteΓde présenter des méthodes numériques permettant de calculer les estimations bayésiennes indépendamment de la forme précise des densités des lois a priori $\Pi_i(\lambda_i \mid \lambda_{i-1})$.

En procédant ainsi on obtient un outil bayésien général permettant d'adapter la modélisation exponentielle markovienne aux différents types de connaissances a priori que peuvent avoir les praticiens.

3.5.5 Méthodes simulatives pour le calcul des estimations bayésiennes

Dans la modélisation bayésienne exponentielle le calcul des estimations des différents attributs de la fiabilité se ramène généralement au calcul d'intégrales multiples non simplifiables.

Toute tentative pour avoir des estimateurs analytiques simples à calculer nécessiteΓcomme on l'a vu dans le cas des accroissements indépendantsΓdes hypothèses a priori très particulières.

Si l'on souhaite rester dans un cadre généralΓil faut proposer des méthodes numériques qui ne tiennent compte que des hypothèses générales de la modélisation bayésienne exponentielle.
Ces méthodes ne doivent pas dépendre de la forme précise des lois a priori des v.a.r. Λ_iΓ lois a priori qui varient d'une étude à une autre.

On peut trouver dans la littérature (cf. [91] et [92]) un certain nombre de méthodes numériques pour le calcul d'estimations bayésiennes s'exprimant en fonction d'intégrales multiples.
On en présente ci-dessous deux : la méthode "classique" de Monte-Carlo et l'algorithme d'échantillonnage de Gibbs.

On décrira ensuite l'utilisation de ces méthodes dans le cadre de la modélisation bayésienne exponentielle à taux de défaillance markoviens.

Méthode de Monte-Carlo : rappels

On décrit ici brièvement la méthode de Monte-Carlo. Des études détaillées de cette méthode peuvent être trouvées dans Rubinstein [85].

Supposons que l'on s'intéresse à l'estimation d'une quantité réelle :

$$y_n = g(\lambda_1, \ldots, \lambda_n)$$

fonction des taux de défaillance inconnus λ_i. La fonction g est une fonction intégrable connue à valeurs dans $\mathbb{R}$.

Dans une approche inférentielle bayésienneΓestimer y_n revient à calculer l'espérance a posteriori de la v.a.r. :

$$Y_n = g(\Lambda_1, \ldots, \Lambda_n)$$

Cette espérance s'écrit :

$$E\left[\, g(\Lambda^{(n)}) \mid x^{(n)} \,\right] = \int_{\mathbb{R}_+^n} g(\lambda_1, \ldots, \lambda_n)\, f_{\Lambda^{(n)}|x^{(n)}}(\lambda_1, \ldots, \lambda_n)\, d\lambda_1 \ldots d\lambda_n$$
$$= \frac{\int_{\mathbb{R}_+^n} g(\lambda_1, .., \lambda_n)\, f_{X^{(n)}|\lambda^{(n)}}(x_1, .., x_n)\, \Pi_n(\lambda_1, .., \lambda_n)\, d\lambda_1 .. d\lambda_n}{\int_{\mathbb{R}_+^n} f_{X^{(n)}|\lambda^{(n)}}(x_1, .., x_n)\, \Pi_n(\lambda_1, .., \lambda_n)\, d\lambda_1 .. d\lambda_n}$$

Cette espéranceΓcomme la plupart des estimations bayésiennes (cf. formules (3.25)Γ(3.27) et (3.29))Γs'exprime en fonction d'intégrales multiples de la forme :

$$I_n(\Phi) = \int_{\mathbb{R}_+^n} \Phi(\lambda^{(n)}, x^{(n)})\, d\lambda_1 \ldots d\lambda_n$$

où Φ est une fonction connue à valeurs dans $\mathbb{R}$.

Définition – 3.21 *La **méthode de Monte-Carlo** permet, par des simulations, d'estimer les intégrales multiples du type de $I_n(\Phi)$. Cette méthode est basée sur le résultat suivant conséquence de la loi des grands nombres :*

> *Soit q la densité d'une loi de probabilité définie sur $\mathbb{R}_+^n$ dont on sait simuler des réalisations et vérifiant la propriété :*
>
> $$\forall \lambda^{(n)} \in \mathbb{R}_+^n,\ \ \Phi(\lambda^{(n)}, x^{(n)}) \neq 0 \Longrightarrow q(\lambda^{(n)}) \neq 0.$$
>
> *L'intégrale $I_n(\Phi)$ qui peut se réécrire sous la forme :*
>
> $$I_n(\Phi) = \int_{\mathbb{R}_+^n} \frac{\Phi(\lambda^{(n)}, x^{(n)})}{q(\lambda^{(n)})} q(\lambda^{(n)})\, d\lambda_1 \ldots d\lambda_n$$
>
> *est bien approchée par :*
>
> $$\widehat{I}_{n,d}(\Phi) = \frac{1}{d} \sum_{k=1}^{d} \frac{\Phi(\lambda^{(n),k}, x^{(n)})}{q(\lambda^{(n),k})}$$
>
> *où :*
>
> - *d est un entier positif assez grand*
> - *$\lambda^{(n),1}, \ldots, \lambda^{(n),d}$ sont des simulations de réalisations indépendantes, de la loi de probabilité q.*

Remarques – $\widehat{I}_{n,d}(\Phi)$ est un estimateur sans biais convergeant de $I_n(\Phi)$Γsa variance vaut :

$$Var(\widehat{I}_{n,d}(\Phi)) = \frac{1}{d} \left[\int_{\mathbb{R}_+^n} \Phi^2(\lambda^{(n)}, x^{(n)})\, d\lambda_1 \ldots d\lambda_n - I_n^2(\Phi) \right].$$

Utilisation de la méthode de Monte-Carlo

On décrit ici l'utilisation de la méthode de Monte-Carlo pour le calcul des estimations :

- des taux de défaillance λ_i (cf. formule (3.25))
- du $MTTF$ a posteriori $E(X_{n+1} \mid x^{(n)})$ donné par la formule (3.29).

En utilisant la méthode de Monte-CarloΓon aura à calculer séparément les numérateurs et le dénominateur des expressions (3.25) et (3.29).

Considérons par exemple l'estimation du dénominateur qu'on notera D_n :

Notation – On note dans la suite :

$$D_n \equiv \int_{\mathbb{R}_+^n} \prod_{i=1}^{n} [\, \lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i \mid \lambda_{i-1})\,]\, d\lambda^{(n)}.$$

L'estimation de D_n est faite en utilisant la méthode de la définition 3.21 où l'on prend :

$$\Phi(\lambda^{(n)} \mid x^{(n)}) = \prod_{i=1}^{n} \left[\, \lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i \mid \lambda_{i-1})\right] \quad \text{et} \quad q(\lambda^{(n)}) = \Pi_n(\lambda_1, \ldots, \lambda_n).$$

En simulant d réalisations indépendantes $\lambda^{(n),1}, \lambda^{(n),2}, \ldots, \lambda^{(n),d}$ à partir de la loi a priori ΠΓon estime D_n par :

$$\hat{D}_{n,d} = \frac{1}{d} \sum_{k=1}^{d} \left[\prod_{i=1}^{n} \lambda_i^{(n),k}\, exp(-\lambda_i^{(n),k} x_i) \right].$$

La qualité de l'estimateur précédent dépend de la taille d de l'échantillon simulé ainsi que de la précision des lois a priori Π_i.
L'inégalité de Bienayme-Tchebichev permet de donner une indication sur la qualité de cette estimation :

$$\forall\, \epsilon > 0 \Gamma P\,[\,|\, D_n - \hat{D}_{n,d}\,| > \epsilon] \leq \frac{Var\,[\,\prod_{i=1}^{n} \Lambda_i exp(-\Lambda_i x_i)\,]}{d.\epsilon^2}.$$

Si la précision requise est spécifiée par exemple par :

$$P\,[\,|\, D_n - \hat{D}_{n,d}\,| > 0.1 D_n] \leq 0.1$$

On aura alors à générer un échantillon de taille :

$$d \simeq 1000\,.\, \frac{Var\,[\,\prod_{i=1}^{n} \Lambda_i exp(-\Lambda_i x_i)\,]}{E^2\,[\,\prod_{i=1}^{n} \Lambda_i exp(-\Lambda_i x_i)\,]}$$

Le nombre de simulations nécessaires d augmente avec le nombre de temps inter-défaillances observés $n\Gamma$il augmente aussi avec les variances des lois a priori $\Pi_i(\lambda_i \mid \lambda_{i-1})$.

Remarque – Dans le cas de jeux de données de défaillance de tailles assez élevées $(n > 20)$Γle nombre de simulations d nécessaires pour avoir de bonnes estimations de D_n devient très élevé.
On peut résoudre ce genre de problèmes en divisant le jeu de données traité en petits paquets d'observationsΓchacun de ces paquets est traité séparément en utilisant des lois a priori issues du paquet de données précédent.
On peut aussi utiliserΓdans le cas de grands jeux de donnéesΓl'algorithme d'échantillonnage de Gibbs décrit ci-dessous.

L'algorithme d'échantillonnage de Gibbs

La méthode de Monte-Carlo est conceptuellement simple mais pose des problèmes pratiques dans le cas de grands jeux de données.

L'algorithme d'échantillonnage de Gibbs est une méthode alternative assez simple à implémenter. On l'utilise ici pour le calcul des estimations des taux de défaillance λ_i données par l'expression (3.25).

Hypothèses – Supposons que l'on souhaite simuler des réalisations d'une loi de probabilité sur $\mathbb{R}^n$ spécifiée par sa densité h qui a une forme assez complexe.

Notations – Soit $Y^{(n)}$ un vecteur aléatoire de loi de probabilité h et $y^{(n)}$ une réalisation associée.
Pour tout $i \leq n$ on note :

1. y_{-i} le vecteur à $n-1$ éléments donné par :

$$y_{-i} \equiv (y_j)_{j \leq n, j \neq i}$$

2. $h(y_i \mid y_{-i})$ la densité de la loi de probabilité de la v.a.r. Y_i conditionnellement à $\{Y_j = y_j\}_{j \leq n, j \neq i}$.

L'algorithme d'échantillonnage de Gibbs permetΓsi on sait simuler des réalisations des lois $h(y_i \mid y_{-i})$Γde simuler le comportement d'une chaîne de Markov ergodique dont la loi stationnaire est h.
On extrait alors à partir des trajectoires de cette chaîne des simulations de réalisations indépendantes de la loi h.

Définition – 3.22 ***L'échantillonneur de Gibbs*** *fournit des trajectoires* $\{y^{(n),k}\}_{k \geq 1}$ *d'une chaîne de Markov dont la loi stationnaire est* h.
L'algorithme associé est décrit par les trois étapes suivantes :

1. *Choix arbitraire d'un vecteur initial* $y^{(n),0} = (y_1^0, \ldots, y_n^0)$.

2. *Passage du vecteur* $y^{(n),0}$ *au prochain état* $y^{(n),1}$ *: ce passage se fait en procédant à des tirages aléatoires à partir des lois conditionnelles* $h(y_i \mid y_{-i})$ *suivant le schéma suivant :*

$$\left\{\begin{array}{lll} y_1^1 & \textit{est tiré selon la loi} & h(y_1 \mid y_{-1}^0) \\ y_2^1 & \textit{est tiré selon la loi} & h(y_2 \mid y_1^1, y_3^0, \ldots, y_n^0) \\ y_3^1 & \textit{est tiré selon la loi} & h(y_3 \mid y_1^1, y_2^1, y_4^0, \ldots, y_n^0) \\ & \ldots & \\ y_n^1 & \textit{est tiré selon la loi} & h(y_n \mid y_{-n}^1) \end{array}\right.$$

3. *Passage de la réalisation* $y^{(n),k}$ *à la réalisation* $y^{(n),k+1}$ *qui se fait suivant le même schéma que ci-dessus.*

On obtient ainsi des réalisations $y^{(n),1}$, $y^{(n),2}$, *etc. d'une chaîne de Markov dont la loi stationnaire est* h.

Pour obtenir une approximation d'un échantillon de loi hΓil suffit donc de simuler une suite de réalisations $\{y^{(n),k}\}_{k\geq 1}$ selon l'algorithme de Gibbs décrit ci-dessusΓet d'en extraire judicieusement (à partir d'un rang élevé r et à intervalles réguliers) une sous-suite : $y^{(n),r}, y^{(n),r+s}, y^{(n),r+2s}$Γetc.

Des justifications théoriques et des indications supplémentaires sur la mise en œuvre de cet algorithme sont données par Smith et Roberts [92].

On décrit dans la sous-section suivante l'utilisation de l'algorithme de Gibbs dans le cadre de la modélisation bayésienne exponentielle à taux de défaillance markoviens.

3.5.6 Mise en œuvre de l'algorithme de Gibbs

On utilise ici l'algorithme d'échantillonnage de Gibbs pour calculer les estimations bayésiennes (moyennes et modes a posteriori) des taux de défaillance λ_i.

Plus généralementΓsupposons que l'on s'intéresse à l'estimation d'une quantité réelle :

$$y_n = g(\lambda_1, \ldots, \lambda_n)$$

où g est une fonction intégrable connue à valeur dans $\mathbb{R}$.

On s'intéresse donc au calcul numérique de la quantité :

$$E\,[\, g({\Lambda_n}^{(n)}) \mid x^{(n)}\,].$$

A priori markoviens généraux

L'algorithme de Gibbs permet de calculer $E\,[\,g({\Lambda_n}^{(n)})\mid x^{(n)}\,]$ en simulant des réalisations indépendantes $\{\lambda^{(n),k}\}_{k\geq 0}$ à partir de la loi a posteriori $f_{\Lambda^{(n)}|x^{(n)}}$.

Notation – Pour tout $i\leq n$ on note $f_{\Lambda_i|\lambda_{-i},x^{(n)}}$ la densité de la loi de probabilité de Λ_i conditionnellement à $X^{(n)}=x^{(n)}$ et à $\{\Lambda_j=\lambda_j\}_{j\leq n,j\neq i}$.

Hypothèse – Supposons que l'on sache simuler les lois de densités $f_{\Lambda_i|\lambda_{-i},x^{(n)}}$. –

L'algorithme de Gibbs permettant de simuler des réalisations $\{\tilde{\lambda}^{(n),k}\}_{k\geq 0}$ de la loi $f_{\Lambda^{(n)}|x^{(n)}}$ est ici décrit par le schéma suivant :

1. Choix arbitraire des valeurs initiales $\lambda^{(n),0}=(\lambda_1^0,\ldots,\lambda_n^0)$.
2. Passage du vecteur $y^{(n),0}$ à la réalisation $y^{(n),1}$: ce passage se fait en procédant à des tirages aléatoires à partir des lois conditionnelles$f_{\Lambda_i|\lambda_{-i},x^{(n)}}$ suivant le schéma suivant :

$$\left\{\begin{array}{lll}
\lambda_1^1 & \text{est tiré selon la loi} & f_{\Lambda_1|\lambda^0_{-1};\,x^{(n)}}\\
\lambda_2^1 & \text{est tiré selon la loi} & f_{\Lambda_2|\lambda_1^1,\lambda_3^0,\ldots,\lambda_n^0;\,x^{(n)}}\\
\lambda_3^1 & \text{est tiré selon la loi} & f_{\Lambda_3|\lambda_1^1,\lambda_2^1,\lambda_4^0,\ldots,\lambda_n^0;\,x^{(n)}}\\
 & \ldots & \\
\lambda_n^1 & \text{est tiré selon la loi} & f_{\Lambda_n|\lambda^1_{-n};\,x^{(n)}}
\end{array}\right.$$

   ceci achève une transition de $\lambda^{(n),0}$ vers $\lambda^{(n),1}$.
3. Passage de la réalisation $\lambda^{(n),k}$ à la réalisation $\lambda^{(n),k+1}$: ceci se fait suivant le même schéma que ci-dessus.

A partir d'un certain nombre d'itérations de l'algorithme de Gibbs on s'approche de l'état stationnaire de la chaîne de Markov. On peut alorsΓenΓ extrayant une sous-suite $\{\tilde{\lambda}^{(n),k}\}_{k\geq 0}$ à partir de la suite $\{\lambda^{(n),k}\}_{k\geq 0}$Γsimuler des réalisations indépendantes de la loi $f_{\Lambda^{(n)}|x^{(n)}}$.

La quantité $y_n=g(\lambda_1,\ldots,\lambda_n)$ est alors estimée par :

$$E\,[\,g(\Lambda^{(n)})\mid x^{(n)}\,]\simeq\frac{1}{d}\sum_{k=1}^{d}g(\tilde{\lambda}^{(n),k}).$$

Remarque – En simulant des réalisations de la loi $f_{\Lambda^{(n)}|x^{(n)}}$Γl'algorithme de Gibbs permet de calculer aussi bien l'estimation de Bayes (espérance a posteriori) de y_n que l'estimation de maximum de vraisemblance bayésien (mode a posteriori de la loi $f_{Y_n|x^{(n)}}$).

Rappelons que l'implémentation de l'algorithme précédent nécessite de savoir simuler les lois conditionnelles $f_{\Lambda_i|\lambda_{-i},x^{(n)}}$.

Ces simulations peuvent être faites en utilisant le résultat suivant :

Proposition – 3.23 *Dans la modélisation bayésienne exponentielle à taux de défaillance markoviens, on a* $\forall i < n$ *:*

$$f_{\Lambda_i|\lambda_{-i},x^{(n)}}(\lambda_i) \propto \lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i \mid \lambda_{i-1})\, \Pi_{i+1}(\lambda_{i+1} \mid \lambda_i)$$

pour $i = n$ *on a :*

$$f_{\Lambda_n|\lambda_{-n},x^{(n)}}(\lambda_n) \propto \lambda_n e^{-\lambda_n x_n}\, \Pi_n(\lambda_n \mid \lambda_{n-1}).$$

Preuve – En utilisant la formule de Bayes on écrit pour tout $i \leq n$:

$$\begin{aligned} f_{\Lambda_i|\lambda_{-i},x^{(n)}}(\lambda_i) &= \frac{f_{\Lambda^{(n)}|x^{(n)}}(\lambda^{(n)})}{f_{\Lambda_{-i}|x^{(n)}}(\lambda_{-i})} \\ &= \frac{\prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i \mid \lambda_{i-1})\,]}{\int_{\mathbb{R}_+} \prod_{i=1}^{n} [\lambda_i e^{-\lambda_i x_i}\, \Pi_i(\lambda_i \mid \lambda_{i-1})\,]\, d\lambda_i} \end{aligned}$$

d'où le résultat énoncé. □

La simulation de réalisations à partir des densités précédentes peut alors se faire en utilisant des méthodes de simulation par rejet (cf. [5] et [41]).

La méthode de rejet est utilisée ci-dessous dans le cas des a priori exponentielsΓuniformes et log-normaux présentés dans la sous-section 3.5.3.

L'algorithme de Gibbs pour des a priori exponentiels

D'après les résultats précédentsΓle calcul des estimations des taux de défaillance λ_i revientΓ en utilisant l'algorithme de GibbsΓà simuler des réalisations de v.a.r. dont les densités sont données par la proposition 3.23.

On choisit ici des lois a priori exponentielles données pour tout $i \geq 1$ par :

$$\text{Conditionnellement à } \Lambda_{i-1} = \lambda_{i-1} \;\Gamma\; \Lambda_i \sim Exp(\frac{1}{\lambda_{i-1}} e^{\theta}).$$

on a par conséquentΓpour tout $i \geq 1$:

$$\Pi_i(\lambda_i \,|\, \lambda_{i-1}) = \frac{e^{\theta}}{\lambda_{i-1}}\, exp(-\frac{\lambda_i}{\lambda_{i-1}} e^{\theta}).$$

Le résultat suivant permet de simuler des réalisations des lois de densités $f_{\Lambda_i|\lambda_{-i},x^{(n)}}$.

Proposition – 3.24 *Pour le modèle a priori choisi ici, on a :*

Pour $i=n$, $f_{\Lambda_n|\lambda_{-n},x^{(n)}}$ *est la densité d'une loi* $Gamma(2, x_n + \frac{e^\theta}{\lambda_{n-1}})$ *:*

$$f_{\Lambda_n|\lambda_{-n},x^{(n)}}(\lambda_n) \propto \lambda_n exp\,[\,-\lambda_n(x_n + \frac{e^\theta}{\lambda_{n-1}})\,].$$

Pour $i < n$ *on a :*

$$f_{\Lambda_i|\lambda_{-i},x^{(n)}}(\lambda_i) \propto g_{\xi_i}(\lambda_i)\ .\ h_{\xi_i}(\lambda_i)$$

où :

- g_{ξ_i} *est la densité d'une loi :* $Gamma(\xi_i + 2, x_i + \frac{e^\theta}{\lambda_{i-1}})$
- h_{ξ_i} *est la densité d'une loi* Inverse-Gamma *:* $IG(\xi_i, \lambda_{i+1}e^\theta)$
- ξ_i *est un réel introduit pour optimiser l'algorithme de simulation.*

Preuve – Pour $i=n$ le résultat découle directement de la proposition 3.23.
En utilisant cette même propositionΓon écrit pour $i < n$:

$$\begin{aligned}
f_{\Lambda_i|\lambda_{-i},x^{(n)}}(\lambda_i) &\propto \lambda_i e^{-\lambda_i x_i}\ \frac{e^\theta}{\lambda_{i-1}}\ exp(-\frac{\lambda_i}{\lambda_{i-1}}e^\theta)\ \frac{e^\theta}{\lambda_i}\ exp(-\frac{\lambda_{i+1}}{\lambda_i}e^\theta) \\
&\propto exp[-(x_i + \frac{e^\theta}{\lambda_{i-1}})\,\lambda_i]\ exp(-\frac{\lambda_{i+1}}{\lambda_i}\,e^\theta) \\
&\propto \lambda_i^{\xi_i+1}\,exp[-(x_i + \frac{e^\theta}{\lambda_{i-1}})\,\lambda_i] \quad \lambda_i^{-\xi_i-1}\,exp(-\frac{\lambda_{i+1}}{\lambda_i}\,e^\theta) \\
&\propto g_{\xi_i}(\lambda_i) \qquad . \qquad h_{\xi_i}(\lambda_i).
\end{aligned}$$

d'où le résultat énoncé.

□

D'après la proposition précédenteΓon peut simuler des réalisations des lois de densités $f_{\Lambda_i|\lambda_{-i},x^{(n)}}$ en utilisant la méthode de rejet qui revient ici à l'utilisation de l'algorithme suivant :

Répéter la simulation de réalisations $\tilde{\lambda}_i$ et $\tilde{u}$ où :

- $\tilde{\lambda}_i$ est simulée à partir de la loi $Gamma(\xi_i + 2, x_i + \frac{e^\theta}{\lambda_{i-1}})$
- $\tilde{u}$ est simulée à partir de la loi $Unif[0,1]$

jusqu'à satisfaction de la condition :

$$M_{\xi_i}.\tilde{u} < h_{\xi_i}(\tilde{\lambda}_i) \quad \text{où} \quad M_{\xi_i} = \sup_{x \in \mathbb{R}_+} h_{\xi_i}(x).$$

En effetΓla méthode de simulation par rejet (cf. par exemple [86]) résulte de la proposition suivante :

Proposition – 3.25 *Soient f et g deux densités de probabilité sur $\mathbb{R}$ et c une constante vérifiant :*

$$\forall x \in \mathbb{R}, \ cg(x) \geq f(x).$$

Le résultat suivant permet de simuler des réalisations d'une v.a.r. de densité f :
Soient X une v.a.r. de densité g et U une v.a.r. de loi $Unif[0,1]$ indépendante de X. La loi conditionnelle de X sachant que " $cUg(X) < f(x)$ " a pour densité f.

L'algorithme décrit ci-dessus se déduit de la proposition précédente en prenant :

$$f \equiv g_{\xi_i} h_{\xi_i} \Gamma \ g \equiv g_{\xi_i} \ \text{ et } c \equiv \sup_{x \in \mathbb{R}_+} h_{\xi_i}(x).$$

Remarque – Comme le font remarquer Arjas et Gasbarra [5]Γon peut optimiser la méthode de simulation par rejet en choisissant le paramètre ξ_i tel que les densités g_{ξ_i} et h_{ξ_i} aient leurs modes au même point.
Dans le cas particulier présenté iciΓcette valeur optimale du paramètre ξ_i est donnéeΓpour $i < n$Γpar :

$$\xi_i^* = [(\lambda_{i+1} e^\theta)\,(x_i + \frac{e^\theta}{\lambda_{i-1}})]^{1/2} - 1.$$

L'algorithme de Gibbs pour des a priori uniformes

Les méthodes de simulation décrites ci-dessus s'appliquent aussi au cas où les v.a.r. Λ_i ont des a priori uniformes :

$$\text{Conditionnellement à } \Lambda_{i-1} = \lambda_{i-1} \ \Gamma \ \Lambda_i \sim Unif[(1-\alpha)\lambda_{i-1}, (1+\beta)\lambda_{i-1}].$$

Dans ce cas les densités requises pour la mise en œuvre de l'algorithme de Gibbs s'écrivent pour $i < n$:

$$f_{\Lambda_i|\lambda_{-i},x^{(n)}}(\lambda_i) \propto \lambda_i e^{-\lambda_i x_i}\ 1_{\{\lambda_i \in [\, max(\alpha\lambda_{i-1},\lambda_{i+1}/\beta),min(\beta\lambda_{i-1},\lambda_{i+1}/\alpha)\,]\}}$$

Pour simuler des réalisations à partir de telles densités on utiliseΓcommeprécédemmentΓ la méthode de simulation par rejet.
On reconnaît en effet dans l'expression précédente le produit de la densité de la loi $Gamma(2, x_i)$ par la densité de la loi $Unif\,[\, max(\alpha\lambda_{i-1}, \lambda_{i+1}/\beta), min(\beta\lambda_{i-1}, \lambda_{i+1}/\alpha)\,]$.

L'algorithme de Gibbs pour des a priori log-normaux

Sous des lois a priori log-normales on a :

$$\text{Conditionnellement à } \Lambda_{i-1}=\lambda_{i-1}\ \Gamma\ \Lambda_i \sim \text{log-normale}(ln(\lambda_{i-1}) - \theta, \sigma^2).$$

On a donc pour tout entier $i \geq 1$:

$$\Pi_i(\lambda_i \,|\, \lambda_{i-1}) = \frac{1}{\lambda_i\sqrt{2\pi\sigma^2}}\, exp\,[\, \frac{(ln(\lambda_i) - ln(\lambda_{i-1}) + \theta)^2}{2\sigma^2}\,].$$

Pour utiliser l'algorithme de GibbsΓon a donc à simuler des réalisations de lois dont les densités sont donnéesΓpour tout $i \leq n$Γpar :

$$f_{\Lambda_i|\lambda_{-i},x^{(n)}}(\lambda_i) \propto \lambda_i e^{-\lambda_i x_i}\ \ \frac{1}{\lambda_i} exp\,[\, -\frac{1}{\sigma^2}\,(ln(\lambda_i) - \nu_i)^2\,]$$

où $\nu_i = \dfrac{1}{2}\,[\, ln(\lambda_{i+1}) + ln(\lambda_{i-1})\,]$.

On utilise à nouveau la méthode de simulation par rejet puisqueΓd'après la formule précédenteΓla densité $f_{\Lambda_i|\lambda_{-i},x^{(n)}}$ s'écrit comme le produit de la densité de la loi $Gamma(2, x_i)$ avec celle la loi log-normale$(\nu_i, \sigma^2/2)$.

Dans la section suivanteΓon expérimente l'approche bayésienne exponentielle sur des jeux de données simulés. On utilise aussi bien la méthode de Monte-Carlo que l'algorithme de Gibbs.

3.5.7 Résultats expérimentaux

Le principal avantage de l'approche bayésienne exponentielle à taux de défaillance markoviens (*BEM*) décrite ci-dessus est la possibilité qu'elle offre aux experts en logiciels de construire leurs propres modèles.

Pour exhiber les apports de l'approche *BEM*Γil est nécessaire que les jeux de données étudiés soient accompagnés de "rapports de progression de tests" décrivant : les protocoles de testsΓles corrections et les modifications effectuéesΓles origines des différentes défaillancesΓ les avis des équipes de tests sur les performances des équipes de développementΓetc.

L'échange et l'exploitation de ces rapports nécessitent une forte collaboration entre le statisticien et l'expert logiciel.

Pour démontrer l'applicabilité de l'approche *BEM*Γon l'appliquera ici sur des jeux de données simulés à partir de taux de défaillance λ_i **connus**.

On comparera alorsΓpour des a priori de différentes qualitésΓles estimations bayésiennes $\hat{\lambda}_i$ aux vrais taux de défaillance λ_i.
On comparera ensuite ces estimations bayésiennes aux estimations fournies par les modèles paramétriques usuels : *MPD*Γmodèle de *Crow* et modèle de *Goel-Okumoto*.

Remarque – Les résultats présentés ci-dessous n'ont qu'une valeur illustrative. Les qualités des estimations bayésiennes dépendent des qualités des informations a priori utiliséesΓces qualités varient d'une étude à une autre.

Simulation des jeux de données

On considère ici deux suites réelles positives $\boldsymbol{lam1} \equiv (lam1_i)_{i \leq 20}$ et $\boldsymbol{lam2} \equiv (lam2_i)_{i \leq 30}$ représentant deux suites de taux de défaillance.

Ces deux suitesΓreprésentées sur la figure 3.2Γsont utilisées pour simuler deux jeux de données inter-défaillancesΓnotés respectivement ***simlam1.d*** et ***simlam2.d***. Ces simulations sont faites à partir du modèle exponentiel :

$$X_i \sim Exp(\lambda_i).$$

AinsiΓla $i^{ème}$ observation x_i du jeu de données *simlam1.d* (resp. *simlam2.d*) est une réalisation de la loi $Exp(lam1_i)$ (resp. $Exp(lam2_i)$).

les observations x_i des jeux de données *simlam1.d* et *simlam2.d* sont représentées sur la figure 3.3.

On utilisera ci-dessous l'approche *BEM* pour estimer les taux de défaillance des jeux de données *simlam1.d* et *simlam2.d*. Ces estimations seront ensuite comparées aux vrais taux de défaillance $lam1$ et $lam2$.

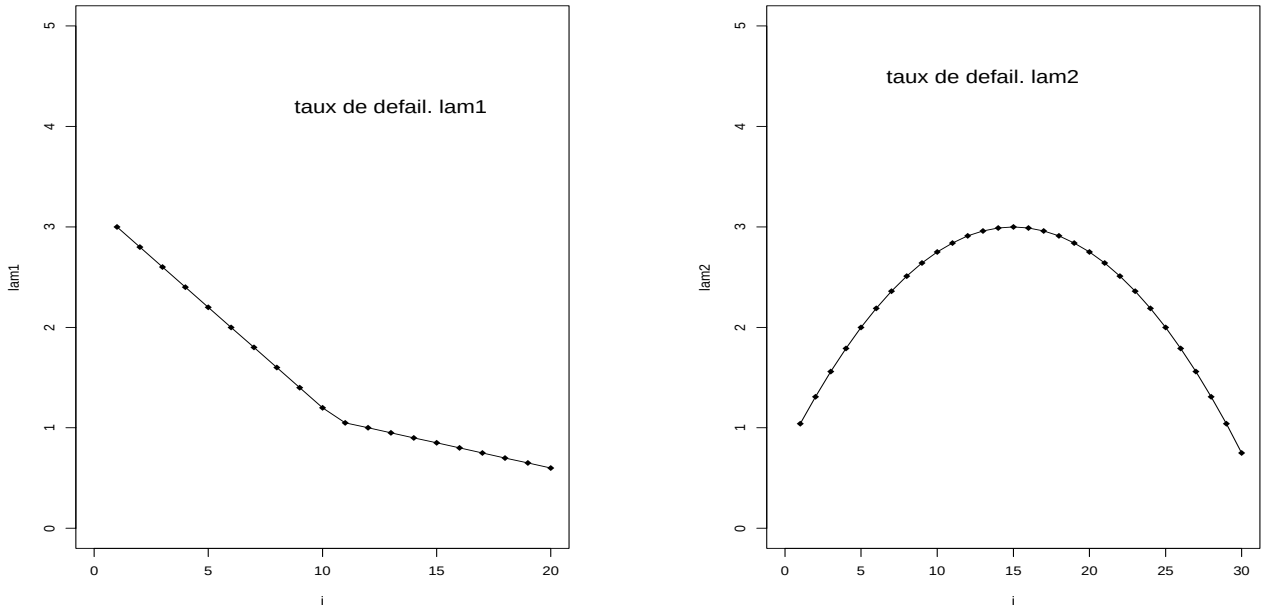

FIG. 3.2: Les suites de taux de défaillance $lam1$ et $lam2$

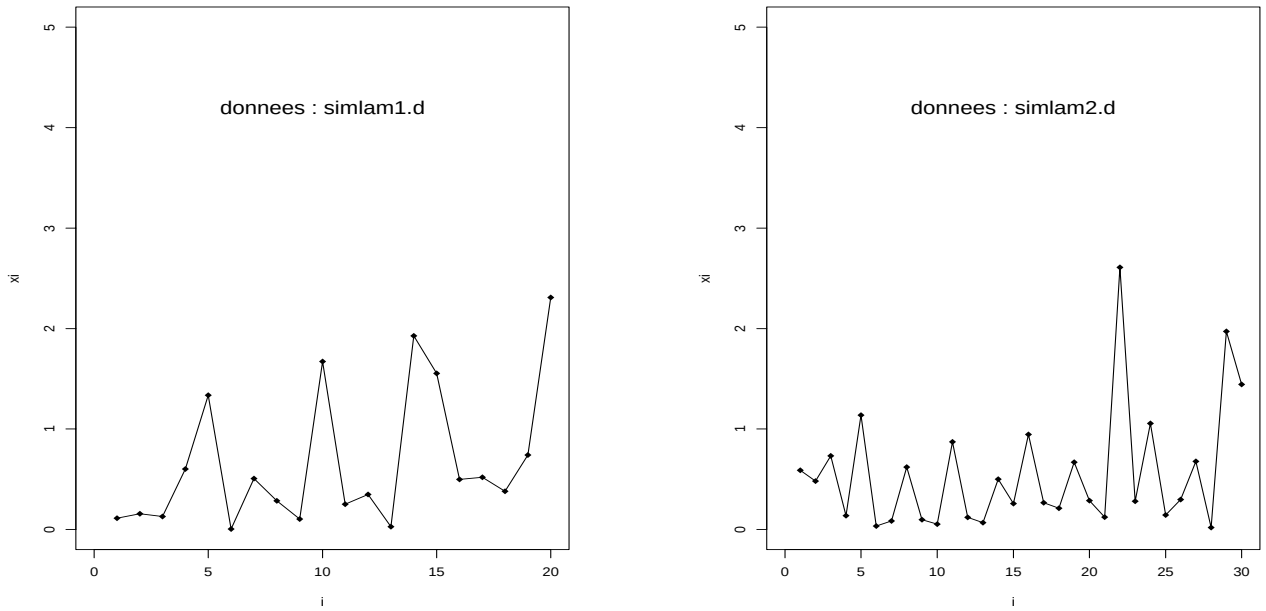

FIG. 3.3: Les jeux de données simulés *simlam1.d* et *simlam2.d*

Choix des hypothèses a priori (a priori log-normaux)

Pour appliquer l'approche *BEM* sur les jeux de données simulés *simlam1.d* et *simlam2.d* on suppose ici que les connaissances a priori disponibles impliquent des hypothèses a priori **log-normales** :

$$\text{Conditionnellement à } \Lambda_{i-1}=\lambda_{i-1} \ \Gamma \ \Lambda_i \sim \text{log-normale}(ln(\lambda_{i-1})-\theta, \sigma^2).$$

Les trois paramètres de ce modèle a prioriΓdont les valeurs sont choisies par l'utilisateurΓ sont :

- θ : représente l'idée des experts quant à l'évolution future de la fiabilité.

  En l'absence de telle connaissance a prioriΓon choisira $\theta=0$.

- σ^2 : représente l'idée que se font les experts au sujet des variations des effets des différentes corrections.

- λ_0 : représente le taux de défaillance initial. Les résultats expérimentaux montrent que les estimations $\hat{\lambda}_i$ sontΓpour σ^2 élevéΓpeu sensibles aux variations des valeurs de λ_0. On prendra dans la suite $\lambda_0=\lambda_1$.

Le "meilleur" choix a priori des valeurs des paramètres θ et σ^2 correspond au cas où on connaît a priori la moyenne empirique θ^* et la variance empirique σ^{*2} de la suite $[\, ln(\lambda_i/\lambda_{i-1}) \,]_{i\leq n}$.

Ces valeurs valent :

$$(\theta^*, \sigma^{*2}) = (0.085, 0.001) \ \text{ pour la suite } lam1 \ \text{ et } \ (\theta^*, \sigma^{*2}) = (0.011, 0.013) \ \text{ pour } lam2.$$

Avant de comparer les estimations fournies par l'approche bayésienne *BEM* aux estimations des modèles usuelsΓon étudie dans le paragraphe suivant la sensibilité des estimations $\hat{\lambda}_i$ aux variations des paramètres σ^2 et θ.

Sensibilité des estimations $\hat{\lambda}_i$ aux variations des paramètres a priori

On utilise ici aussi bien l'algorithme de Gibbs que la méthode de Monte-Carlo pour calculer les estimations bayésiennes (les espérances a posteriori) $\widehat{lam1}$ et $\widehat{lam2}$ des taux de défaillance des jeux de données *simlam1.d* et *simlam2.d*. Ceci est fait pour différentes valeurs des paramètres σ^2 et θ.

Le paramètre σ^2Γvariance du modèle a prioriΓreprésente aussi la qualité de l'information a priori.

Une faible valeur de σ^2 modélise une forte confiance dans l'information a prioriΓl'information apportée par les observations x_i est alors négligeable par rapport à l'information a priori.
Dans le cas opposéΓune forte valeur de σ^2 modélise un a priori "vague"Γl'information issue des observations devient alors prépondérante par rapport à l'information a priori.

Le paramètre $\sigma^2\Gamma$spécifie ainsi les contributions relatives des informations a priori et des observations à l'estimation des paramètres λ_i.
Ceci est illustré par la figure 3.4Γoù on étudie la sensibilité des estimations $\hat{\lambda}_i$ aux variations des valeurs a priori de σ^2.

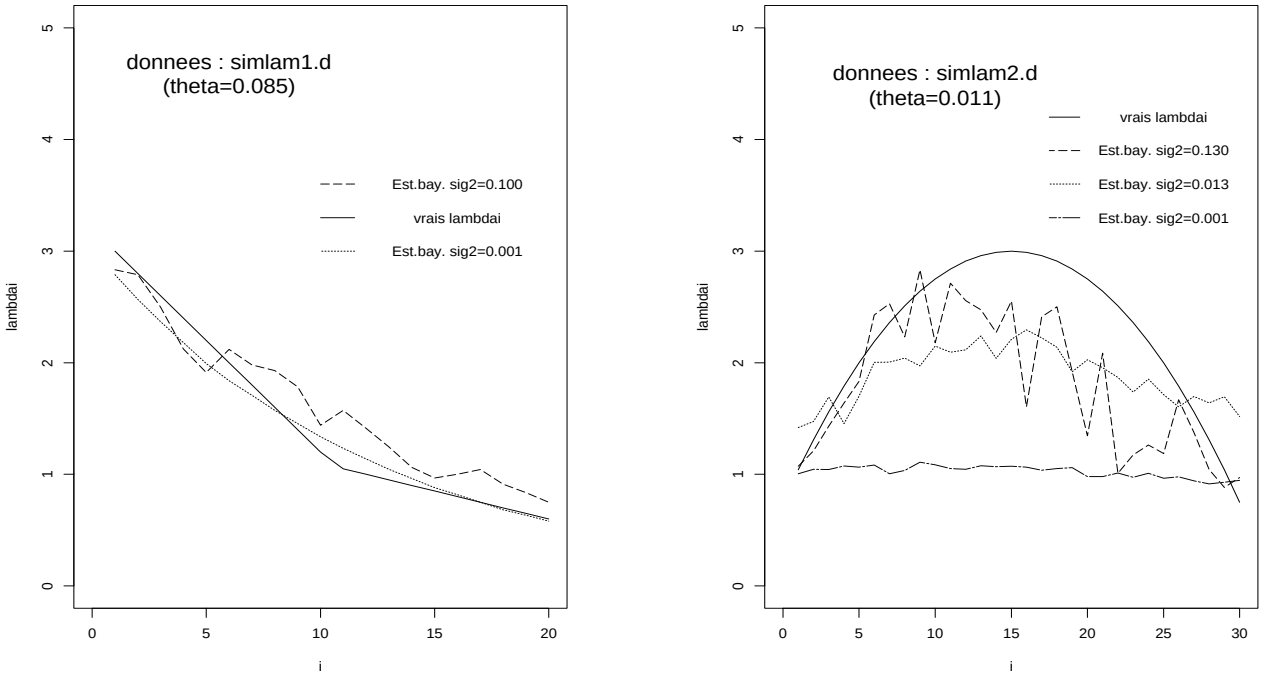

FIG. 3.4: Effet des variations de σ^2 sur les estimations $\hat{\lambda}_i$ $(\theta = \theta^*)$

La sensibilité des estimations $\hat{\lambda}_i$ aux variations des valeurs du paramètre θ dépend de la valeur de σ^2.

Pour une faible valeur de $\sigma^2\Gamma$l'information a priori est prépondérante par rapport aux observations. Les estimations $\hat{\lambda}_i$ sont dans ce cas très sensibles aux variations des valeurs a priori de θ.
Ceci est illustré sur la figure 3.5 pour le jeu de données *simlam1.d*.

Par contre si la valeur de σ^2 est suffisamment forteΓil y aura un certain équilibre entre l'information a priori et les observations. Le choix de la valeur a priori de θ aura dans ce cas une faible influence sur les estimations $\hat{\lambda}_i$.

Ceci est illustré sur la figure 3.5 pour le jeu de données *simlam2.d*.

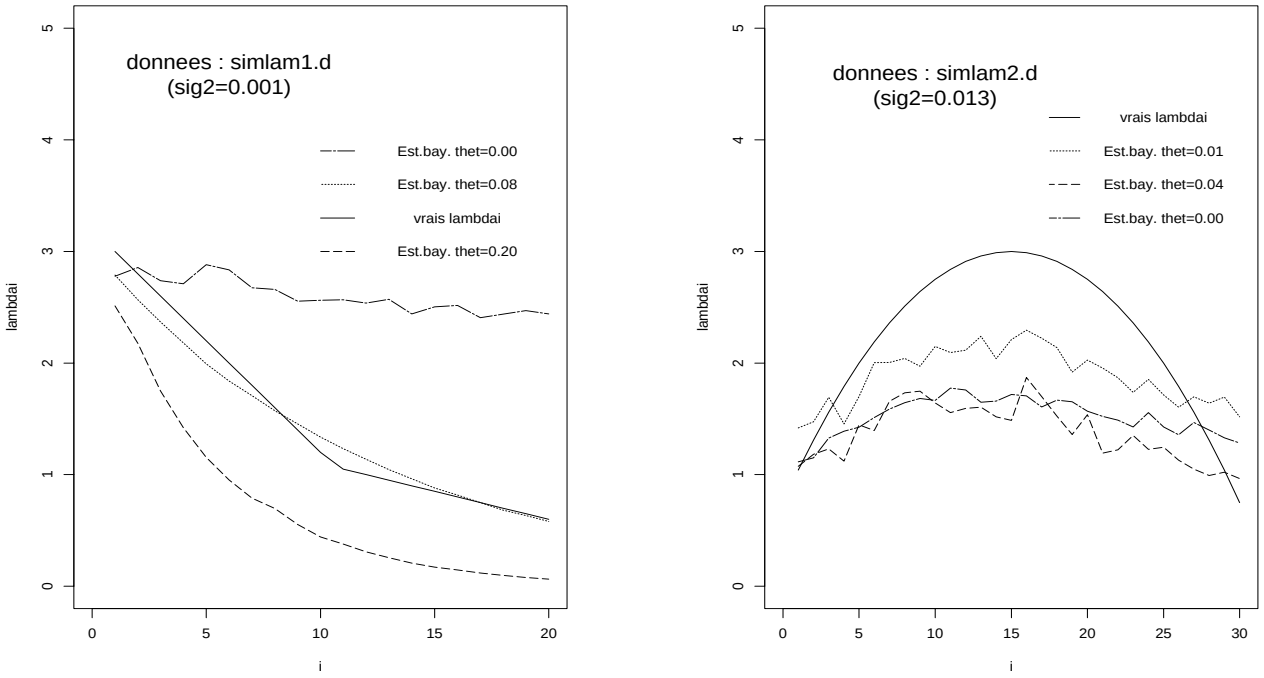

FIG. 3.5: Effet des variations de θ sur les estimations $\hat{\lambda}_i$ ($\sigma^2 = \sigma^{*2}$)

Comparaison avec les modèles usuels

On compare ici les estimations $\hat{\lambda}_i$ fournies par l'approche bayésienne *BEM* aux estimations fournies par le *MPD*Γle modèle de *Crow* et le modèle de *Goel-Okumoto*.

L'approche *BEM* a été utilisée avec des a priori log-normaux d'assez bonne qualité.

En effetΓmême si le choix du modèle a priori log-normal n'est pas un choix optimal pour les jeux de données *simlam1.d* et *simlam2.d*Γle choix des valeurs a priori des paramètres λ_0Γθ et σ^2 l'est. Ces valeurs ont en effet été choisies en tenant compte des vrais taux de défaillance $lam1$ et $lam2$ à partir desquelles ont été simulés *simlam1.d* et *simlam2.d*.
Ces valeurs sont :

- pour *simlam1.d* : $\lambda_0 = \lambda_1 = 3.00$Γ$\theta = \theta^* = 0.085$ et $\sigma^2 = \sigma^{*2} = 0.001$
- pour *simlam2.d* : $\lambda_0 = \lambda_1 = 1.04$Γ$\theta = \theta^* = 0.011$ et $\sigma^2 = \sigma^{*2} = 0.013$

La figure 3.6 représentant les estimations des taux de défaillance λ_i fournies par différents modèlesΓainsi que le tableau 3.2 des sommes des carrés des erreurs relatives $\sum_{i=1}^{n} \left[\frac{\hat{\lambda}_i - \lambda_i}{\lambda_i} \right]^2$Γ confirment que l'approche bayésienne *BEM* donne d'excellents résultats dans le cas de bons a priori.

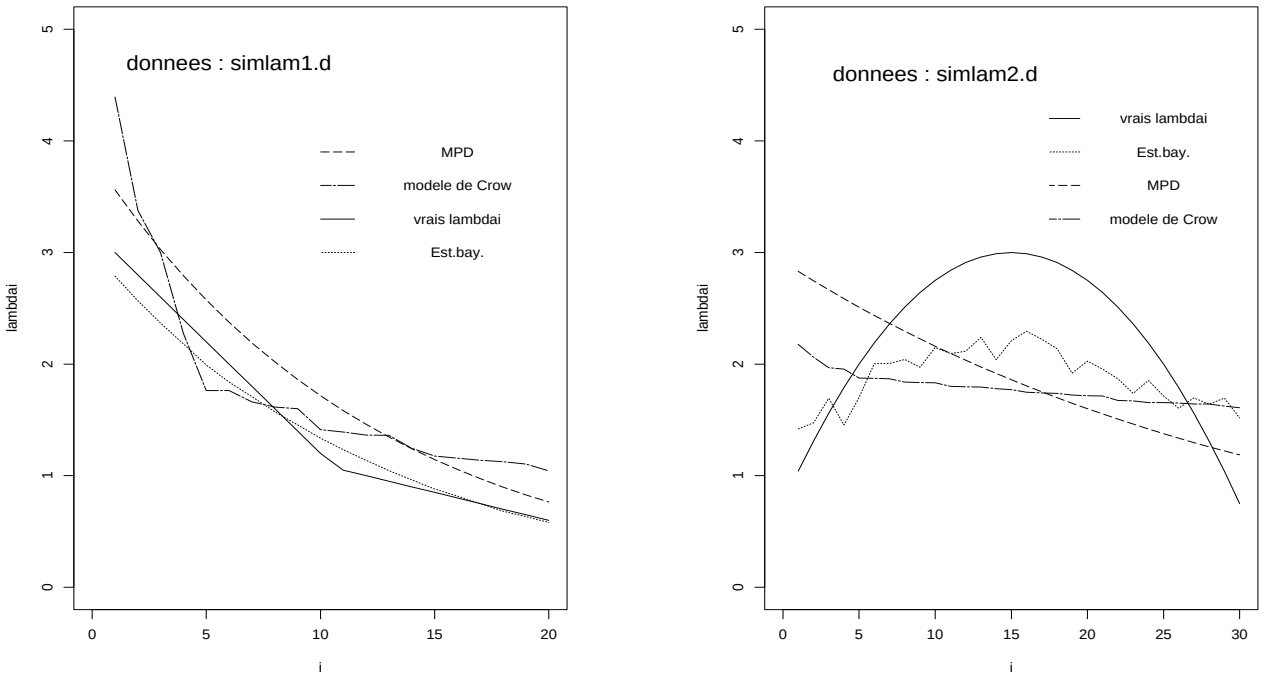

FIG. 3.6: Estimations $\hat{\lambda}_i$ fournies par différents modèles

	BEM	*MPD*	*Crow*	*Goel-Okumoto*
simlam1.d	**0.13**	1.92	2.98	2.48
simlam2.d	**2.90**	7.56	5.62	5.68

TAB. 3.2: Erreurs relatives $\sum_{i=1}^{n} [\ \frac{\hat{\lambda}_i - \lambda_i}{\lambda_i}\ ]^2$

3.6 Conclusion

Le principal objectif des études bayésiennes en Fiabilité des Logiciels présentées jusque là a été de trouver des estimateurs bayésiens s'exprimant sous des formes analytiques simples.

Cet objectif a souvent été atteint en utilisant des hypothèses a priori trop éloignées des réelles connaissances a priori des experts.

On s'est efforcé dans ce chapitre d'utiliser des hypothèses aussi générales que possibles. On a ainsi développé un outil bayésien basé sur des hypothèses minimales dans le contexte de la Fiabilité des Logiciels (absence d'usure) mais permettant d'intégrer différents types de connaissances a priori.

Cet outil général peut être considéré comme un outil d'aide à la modélisation permettant à chaque utilisateur d'introduire les spécificités de son problème à travers le choix des propriétés a priori des taux de défaillance λ_i.

Les résultats expérimentaux présentés à la fin de ce chapitre confirment que l'analyse bayésienne permet d'avoir d'excellentes estimations quand les hypothèses a priori sont de bonne qualité.

Ces a priori de bonne qualité nécessitent une forte collaboration entre le statisticien et les experts en génie logiciel.

Il reste donc à profiter de telles collaborations pour confronter l'approche bayésienne décrite dans ce chapitre aux réalités des problèmes industriels.

Chapitre 4

Validation et Choix de Modèles en Fiabilité des Logiciels

On présente dans ce chapitre quelques outils de validation de modèles de fiabilité des logiciels.

On commence par discuter l'utilisation des tests d'adéquation statistiques dans le cadre des hypothèses générales de la Fiabilité des Logiciels.

On introduit ensuite les outils mathématiques permettant de donner une définition formelle du critère du *u-plot*.
Ce critère a été présenté initialement par Littlewood et Verrall [67] comme un outil graphique de validation des modèles de fiabilité des logiciels.

On généralise ensuite les résultats expérimentaux de Downs et Scott [25] justifiant empiriquement l'utilisation du critère du *u-plot* comme un test d'adéquation statistique.

En essayant de donner une justification théorique à ces résultats expérimentaux on obtient un nouveau test "préquentiel" d'adéquation à une loi exponentielle de paramètre inconnu.

4.1 Introduction

L'abondance de modèles de fiabilité des logiciels et l'absence d'un modèle universel font que les praticiens se trouvent souvent confrontés à la difficulté du choix du modèle le plus adapté à leur problème.

Relativement peu de travaux ont été consacrés au problème de la comparaison et du choix de modèles de fiabilité des logiciels.
On peut par exemple citer : Keiller et al [52]ΓIannino et al [45]ΓAbdel-Ghaly et al [1]Γ Khoshgoftaar et Woodcock [54] et Downs et Scott [25].

De ces travaux ressortent quatre critères principaux [45] de validation a priori. Ils permettent d'évaluer les qualités intrinsèques des modèles indépendamment des données observées. Ces critères sont :

- La validité des hypothèses : le modèle considéré doit être basé sur des hypothèses plausibles et acceptables par les ingénieurs logiciels.

- L'applicabilité : un modèle doit pouvoir s'utiliser dans divers circonstances et cas de figures : différents environnements opérationnelsΓdifférentes étapes du cycle de vieΓetc.
Le modèle doit par ailleurs avoir une certaine robustesse vis à vis des écarts à ses hypothèses.

- La capabilité : un modèle doit être capable d'estimer avec une précision suffisante les attributs utilisés par les praticiens : $MTTF$Γ $ROCOF$Γ taux de défaillanceΓ fonction de fiabilitéΓetc.

- La simplicité : un modèle doit être conceptuellement simpleΓses fondements théoriques doivent être accessibles aux ingénieurs logiciels.
La collecte des données nécessaires à l'estimation de ses paramètres doit être facile et peu coûteuse. Les calculs sous-jacents doivent être facilement programmables et peu coûteux en temps de calcul.

Ces quatre critères permettent de faire une première sélection de modèles.

A ces critèresΓ s'ajoutent un certain nombre de critères de validation a posteriori qui permettentΓ au vu des données recueilliesΓ de choisir le modèle le mieux adapté. Ces critères de validation a posteriori permettent de mesurer (cf. Kanoun [50]) :

- la qualité réplicative : c'est la capacité du modèle à ajuster les données passées. Pour l'évaluer on peut par exemple utiliser les tests d'adéquation statistiques.

- la qualité prévisionnelle : c'est-à-dire la capacité du modèle à prédire les données de défaillance futures.
Un certain nombre d'outils empiriques ont été proposés pour l'évaluation de cette qualité prévisionnelle (cf. [1]).

Les qualités réplicative et prévisionnelle peuvent être évaluées par des outils statistiques assez semblables. Ces outils seront développés dans les sections suivantes.

4.2 Tests d'adéquation statistiques

La mesure de la qualité réplicative peut se faire par l'utilisation des tests d'adéquation statistiques. Ces tests permettent de juger la compatibilité du modèle considéré avec les données de défaillance observées.

Les principales familles de tests d'adéquation (cf. [22]) sont les tests du χ^2 et les tests basés sur la fonction de répartition empirique. On ne s'intéressera dans ce travail qu'aux derniers.

La plupart des tests d'adéquation proposés dans la littérature concernent le cas de modèles où les observations $x_1, \ldots, x_n$ sont issues de v.a.r. X_i i.i.d.
Or en Fiabilité des LogicielsΓla croissance de fiabilité implique des modèles où les v.a.r. temps inter-défaillances X_i ne sont pas i.i.d.

On précise ci-dessous le cadre général et les notations de cette section.
On rappelle ensuite quelques propriétés de la fonction de répartition empirique. Ces résultats permettent d'introduire le test de *Kolmogorov-Smirnov* : test d'adéquation à un modèle où les v.a.r. X_i sont i.i.d. de loi complètement spécifiée.
On parlera ensuite de l'adéquation à différents types de modèles selon que les v.a.r. X_i sont supposées i.i.d. ou nonΓet selon que les paramètres du modèle sont supposés connus ou inconnus.
On précisera pour chacun des cas les éventuelles applications pour les modèles de fiabilité des logiciels.

4.2.1 Cadre général et Notations

Hypothèses – On considère dans toute cette section un processus aléatoire réel $X = \{X_i\}_{i\geq 1}$. On note P^* sa loi de probabilité supposée inconnue.
On suppose qu'on dispose des observations $x_1, \ldots, x_n$ des n premieres v.a.r. $X_1, \ldots, X_n$.
–

Lorsque dans le modèle M les v.a.r. X_i sont supposées i.i.d.Γon note F^* leur fonction de répartition inconnue.

On s'intéresse ici au test de l'adéquation des données $x_1, \ldots, x_n$ à des modèles statistiques M appartenant à l'une des quatre familles suivantes :

- ***cas 0*** : les v.a.r. X_i sont supposées i.i.d. de fonction de répartition $F_M(., \theta_0)$ complètement spécifiée (paramètre θ_0 connu a priori)Γl'hypothèse nulle est alors :
$$H_0^{(0)} \equiv \text{ “ } F^* = F_M(., \theta_0) \text{ ”}.$$
- ***cas 1*** : les v.a.r. X_i ne sont pas i.i.d.Γleur loi de probabilité conjointe P_{θ_0} est complètement spécifiée (paramètre θ_0 connu a priori)Γl'hypothèse nulle est dans ce cas :
$$H_0^{(1)} \equiv \text{ “ } P^* = P_{\theta_0} \text{ ”}.$$
- ***cas2*** : les v.a.r. X_i sont i.i.d.Γleur loi de probabilité $F_M(., \theta)$ n'est pas complètement spécifiée (paramètre θ inconnu)Γl'hypothèse nulle est :
$$H_0^{(2)} \equiv \text{ “ } F^* \in \{F_M(., \theta), \theta \in \Theta\} \text{ ”}.$$
- ***cas3*** : c'est le cas le plus généralΓdans ces modèles les v.a.r. X_i ne sont pas i.i.d. et le paramètre θ n'est pas connu a priori. L'hypothèse nulle est alors :
$$H_0^{(3)} \equiv \text{ “ } P^* \in \{P_\theta, \theta \in \Theta \subset \mathbb{R}^k\} \text{ ”}.$$

Remarque – Nous nous n'intéressons pas ici à la puissance des tests d'adéquation dont l'étude supposerait de modéliser également une contre-hypothèse à M.

Des transformations adéquates des v.a.r. X_i peuvent ramenerΓcomme on le décrira plus tardΓle problème du test des modèles vérifiant les hypothèses des *cas* 1Γ2 ou 3 au problème du test d'adéquation dans le *cas 0*.

Pour tester l'adéquation à un modèle vérifiant les hypothèses du *cas 0* on peut utiliser le test de *Kolmogorov-Smirnov*.
Ce test est basé sur des propriétés asymptotiques de la fonction de répartition empiriqueΓ ces propriétés seront brièvement rappelées ci-dessous.

4.2.2 Propriétés de la fonction de répartition empirique

Hypothèse – On suppose dans cette sous-section que les v.a.r. X_i sont i.i.d. de fonction de répartition F^* continue et inconnue.

Notations – La fonction de répartition empirique associée aux v.a.r. $X_1, \ldots, X_n$ est notée $\mathbb{F}_n$:

$$\begin{aligned} \forall x \in \mathbb{R} \ \Gamma \ \mathbb{F}_n(x) &= \frac{1}{n} \sum_{i=1}^{n} 1_{\{X_i \leq x\}} \\ &= \frac{1}{n} [\text{ nombre des } X_i \leq x] \end{aligned}$$

Distance entre $\mathbb{F}_n$ et F^*

Plusieurs distances peuvent être utilisées pour mesurer l'écart entre les fonctions $\mathbb{F}_n$ et F^* (cf. [22] page 100).
On présente ci-dessous la distance de *Kolmogorov-Smirnov* :

Définition – 4.1 *La **distance de** **Kolmogorov-Smirnov** entre les fonctions $\mathbb{F}_n$ et F^* est donnée par :*

$$D_n = \sup_{x \in \mathbb{R}} | \mathbb{F}_n(x) - F^*(x) | .$$

Notations – Dans la suite de ce paragraphe on utilise les notations suivantes :

1. $D_n^+ \equiv \sup_{x \in \mathbb{R}} (\mathbb{F}_n(x) - F^*(x))$.
2. $D_n^- \equiv \sup_{x \in \mathbb{R}} (F^*(x) - \mathbb{F}_n(x))$.
3. $\forall i \leq n \ \Gamma \ U_i \equiv F^*(X_i)$.
4. $(U_i^*)_{i \leq n}$ est la suite ordonnée (croissante) obtenue à partir de l'échantillon $(U_i)_{i \leq n}$.

Propriétés –

1. Les v.a.r. U_i sont i.i.d. de loi $Unif[0,1]$
2. $D_n^+ = \max_{i\leq n}(\frac{i}{n} - U_i^*)$ et $D_n^- = \max_{i\leq n}(U_i^* - \frac{i-1}{n})$
3. $D_n = max(D_n^+, D_n^-) = max\,[\,\max_{i\leq n}(\frac{i}{n} - U_i^*)\ \Gamma\max_{i\leq n}(U_i^* - \frac{i-1}{n})\,]$.

Théorème – 4.2 (*Kolmogorov-Smirnov*)

$$\sqrt{n}D_n \overset{n\to+\infty,\,Loi}{\longrightarrow} \mathcal{L}_{KS}$$

où $\mathcal{L}_{KS}$ est une loi indépendante de F^*, à valeurs dans $\mathbb{R}_+$ appelée loi de Kolmogorov-Smirnov.
Sa fonction de répartition est donnée par :

$$\begin{aligned} \forall x \geq 0\ ,\ F_{KS}(x) &= \sum_{k=-\infty}^{+\infty} (-1)^k e^{-2k^2x^2} \\ &= 1 + 2\sum_{k=1}^{+\infty}(-1)^k e^{-2k^2x^2}. \end{aligned} \tag{4.1}$$

Remarques et Notations –

1. Les quantiles de la loi de *Kolmogorov-Smirnov* peuvent être trouvés dans des tables numériques (cf. par exemple [86] page 466).
2. Il est facile de démontrer l'égalité suivante :

$$D_n = \sup_{t\in[0,1]} |\ \mathbb{F}_n^u(t) - t\ |$$

où $\mathbb{F}_n^u$ désigne la fonction de répartition empirique associée aux v.a.r. $(U_i)_{i\leq n}$.

Le théorème de *Kolmogorov-Smirnov* peut être déduit (cf. Durbin [27]) à partir du résultat suivant :

Proposition – 4.3 ***Si*** *pour tout entier positif n on note $y_n \equiv \{y_n(t)\}_{t\in[0,1]}$ le processus défini par :*

$$\forall t \in [0,1]\ ,\ y_n(t) = \sqrt{n}\,(\mathbb{F}_n^u(t) - t)$$

alors *la suite de processus $(y_n)_{n\geq 1}$ converge en loi vers le* ***pont brownien*** *:*

$$\{y_n(t)\}_{t\in[0,1]} \overset{Loi}{\longrightarrow} \{\mathbb{B}(t)\}_{t\in[0,1]}$$

où $\{\mathbb{B}(t)\}_{t\in[0,1]}$ désigne le pont brownien sur $[0,1]$.

4.2.3 Adéquation à une loi complètement spécifiée

On utilise dans cette sous-section les résultats présentés ci-dessus pour tester l'adéquation des données $x_1, \ldots, x_n$ à un modèle paramétrique dont le paramètre est connu a priori. La procédure de test sera légèrement différente selon que les v.a.r. X_i sont supposées *i.i.d.* (*cas 0*) ou non (*cas 1*).

Cas 0 : les X_i sont i.i.d. (test de *Kolmogorov-Smirnov*)

Hypothèses – On suppose dans ce paragraphe que les v.a.r. X_i sont i.i.d. de fonction de répartition F^* inconnue.
Le modèle M dont on souhaite mesurer la qualité réplicative est complètement spécifié par la fonction $F_M(.,\theta_0)$ approchant F^*. Le paramètre θ_0 est ici supposé connu. –

Tester l'adéquation des données au modèle précédent revient à tester l'hypothèse :

$$H_0^{(0)} \equiv \text{ “ } F^* = F_M(.,\theta_0) \text{ ” } \quad \text{contre} \quad \text{ “ } F^* \neq F_M(.,\theta_0) \text{ ” } .$$

Pour ce faire on utilise le test 4.2 de *Kolmogorov-Smirnov* qui s'écrit sous l'hypothèse $H_0^{(0)}$ sous la forme suivante :

$$\sqrt{n} D_n = \sqrt{n} \sup_{x \in \mathbb{R}} \mid \mathbb{F}_n(x) - F_M(x,\theta_0) \mid \overset{n \to +\infty, \, Loi}{\longrightarrow} \mathcal{L}_{KS}.$$

Définition – 4.4 *Le **test de** **Kolmogorov-Smirnov** permet de tester l'hypothèse nulle $H_0^{(0)}$ en comparant $\sqrt{n} d_n$ (réalisation de la v.a.r. $\sqrt{n} D_n$) aux quantiles de la loi de* Kolmogorov-Smirnov.
La quantité d_n peut être calculée par la formule suivante :

$$d_n = max\,[\, \max_{i \leq n}(\frac{i}{n} - u_i^*) \;,\; \max_{i \leq n}(u_i^* - \frac{i-1}{n}) \,].$$

où :

- $\forall i \leq n, \quad u_i = F_M(x_i, \theta_0)$
- $(u_i^*)_{i \leq n}$ *est la suite ordonnée (croissante) obtenue à partir de l'échantillon* $(u_i)_{i \leq n}$.

Cas 1 : les X_i ne sont pas i.i.d.

Hypothèses – Dans les modèles de fiabilité des logiciels les v.a.r. X_i ne sontΓen généralΓ ni indépendantes ni équidistribuéesΓc'est l'hypothèse qu'on adopte dans ce paragraphe. On suppose donc que l'on souhaite tester l'adéquation des observations $x_1, \ldots, x_n$ à un modèle où le processus $\{X_i\}_{i \geq 1}$ est de loi P_{θ_0} de paramètre θ_0 connu. –

Ce problème se ramène au test de l'hypothèse :

$$H_0^{(1)} \equiv \text{ “ } P^* = P_{\theta_0} \text{ ” } \quad \text{contre} \quad \text{ “ } P^* \neq P_{\theta_0} \text{ ” } .$$

Le résultat suivantΓdû à Rosenblatt [84]Γpermet de transformer le problème précédent en un problème d'adéquation à un modèle où les variables sont i.i.d. (*cas 0*). On pourra alors utiliser le test de *Kolmogorov-Smirnov* décrit ci-dessus.

Théorème – 4.5 (Rosenblatt) *Soit $(X_1, \ldots, X_n)$ un vecteur aléatoire de fonction de répartition $F(x_1, \ldots, x_n)$ absolument continue.*

Soit T_X la transformation de $\mathbb{R}^n$ dans $\mathbb{R}^n$, qui au vecteur $(x_1, \ldots, x_n)$ fait associer le vecteur $(u_1, \ldots, u_n)$ défini par :

$$\begin{aligned} u_1 &= P(X_1 \leq x_1) \\ u_2 &= P(X_2 \leq x_2 \mid X_1 = x_1) \\ \ldots & \\ u_n &= P(X_n \leq x_n \mid X_1 = x_1, \ldots, X_{n-1} = x_{n-1}) \end{aligned}$$

Les v.a.r. $U_1, \ldots, U_n$ définies par :

$$(U_1, \ldots, U_n) = T_X(X_1, \ldots, X_n)$$

sont i.i.d. de loi $Unif[0,1]$.

Notations – La fonction qui à la $i^{ème}$ v.a.r. X_i associe la v.a.r. U_iΓselon le schéma décrit ci-dessusΓest notée :

$$U_i \equiv F_i(X_i \mid X_1, \ldots, X_{i-1}).$$

Le problème du test d'adéquation aux modèles du *cas 1* (hypothèse $H_0^{(1)}$) se ramèneΓgrâce au théorème précédentΓau problème du test de l'adéquation de la suite :

$$(u_1, \ldots, u_n) = T_X(x_1, \ldots, x_n)$$

à un échantillon de loi $Unif[0,1]$.

Si on connaît les expressions explicites des v.a.r. :

$$U_i = F_i(X_i \mid X_1, \ldots, X_{i-1})$$

en fonction des v.a.r. X_i on peut alors calculer les valeurs u_i et utiliser le test de *Kolmogorov-Smirnov* pour tester leur uniformité.
On en déduira ainsi la qualité de l'adéquation des observations $x_1, \ldots, x_n$ aux modèles vérifiant les hypothèses du *cas 1.*

Exemple : test d'adéquation au modèle de *Goel-Okumoto*

On utilise ici l'approche décrite ci-dessus pour tester l'adéquation d'un jeu de données $x_1, \ldots, x_n$ au modèle de *Goel-Okumoto* [42].

Le modèle de *Goel-Okumoto* présenté dans la section 1.4 est un modèle *NHPP* dont l'intensité de défaillance est donnée pour tout réel positif t par :

$$\lambda_{GO}(t) = \lambda e^{-\phi t} \text{ où } \lambda \in \mathbb{R}_+ \text{ et } \phi \in \mathbb{R}.$$

Notations –

1. On note $P_{GO(\lambda,\phi)}$ la loi de probabilité du processus aléatoire des temps inter-défaillances $\{X_i\}_{i\geq 1}$ dans le modèle de *Goel-Okumoto* de paramètres λ et ϕ.

2. Rappelons que pour tout entier positif i on note :

$$T_i \equiv \sum_{j=1}^{i} X_j \text{ et } t_i = \sum_{j=1}^{i} x_j.$$

Dans les modèles *NHPP* les v.a.r. X_i ne sont ni indépendantes ni équidistribuées.
On a cependant le résultat suivant (cf. Snyder [93] page 59) :

Proposition – 4.6 *Dans un modèle* NHPP *d'intensité de défaillance* $\lambda(t)$*, on a pour tout entier* $i \geq 1$ *:*

$$P(X_i \leq x_i \mid X_1 = x_1, \ldots, X_{i-1} = x_{i-1}) = 1 - exp[-\int_{t_{i-1}}^{t_i} \lambda(u)du\,].$$

En utilisant le théorème de Rosenblatt et la proposition précédente on obtient le résultat suivant :

Proposition – 4.7 *Sous l'hypothèse :*

$$H_0^{(GO)} \equiv \text{ “ } P^* \in \{P_{GO(\lambda,\phi)} \ , \quad \lambda \in \mathbb{R}_+ \ \textit{et} \ \phi \in \mathbb{R}\} \text{ ”}$$

les v.a.r. :

$$U_i = exp\,[\,\frac{\lambda}{\phi}(e^{-\phi T_i} - e^{-\phi T_{i-1}})\,]$$

sont i.i.d. de loi $Unif[0,1]$.

Le résultat précédent permet de tester l'adéquation des observations $x_1, \ldots, x_n$ à une loi de *Goel-Okumoto* dont les paramètres sont connus a priori.

Remarques –

1. L'approche précédente n'est pas spécifique au modèle de *Goel-Okumoto*. On peut la généraliser à tout modèle où on connaît l'expression des fonctions :

   $$F_i(x_i \mid x_1, \ldots, x_{i-1}) = P(X_i \leq x_i \mid X_1 = x_1, \ldots, X_{i-1} = x_{i-1})$$

   et où on connaît a priori les valeurs des paramètres.

2. En pratique on connaît rarement les valeurs des paramètres du modèle étudié. Dans ce cas il faut tester l'adéquation à une famille de lois et non plus à une loi connue.

4.2.4 Adéquation à une famille de lois

En pratique on est le plus souvent ramené à tester l'adéquation à un modèle dont les paramètres ne sont pas spécifiés.
Ceci revient à tester l'adéquation à une famille de lois de probabilité c'est-à-dire à tester l'hypothèse composée :

$$\text{ “ } P^* \in \{P_\theta \ \theta \in \Theta \subset \mathbb{R}^k\} \text{ ” } .$$

On présente dans le paragraphe suivant quelques approches relatives au cas où les v.a.r. X_i sont i.i.d. (*cas 2*).
On présentera ensuite une approche due à O'Reilly et Quesenberry [79] permettant de traiter aussi bien le cas i.i.d. (*cas 2*) que le cas général (*cas 3*).

Introduction : cas où les X_i sont i.i.d.

Hypothèses – Dans ce paragraphe on suppose à nouveau que les v.a.r. X_i sont i.i.d. de fonction de répartition F^* inconnue. –

On souhaite alors étudier la qualité réplicative d'un modèle M spécifié par la famille de fonctions paramétriques : $\{F_M(.,\theta) \ \theta \in \Theta\}$ approchant F^*.

Il faut donc tester l'hypothèse :

$$H_0^{(2)} \equiv \text{ “ } F^* \in \{F_M(.,\theta), \theta \in \Theta\} \text{ ” } \quad \text{contre} \quad \text{ “ } F^* \notin \{F_M(.,\theta), \theta \in \Theta\} \text{ ”.}$$

Le test de *Kolmogorov-Smirnov* tel que présenté dans la définition 4.4 ne peut être utilisé puisqu'on ne peut calculer les quantités :

$$u_i = F_M(x_i, \theta)$$

le paramètre θ étant inconnu.

Notation – Dans la suite de ce chapitre$\Gamma\hat{\theta}$ désignera un estimateur du paramètre θ.

Une première approche de test consiste à remplacer dans la distance de *Kolmogorov-Smirnov* :

$$D_n = \sup_{x \in \mathbb{R}} \mid \mathbb{F}_n(x) - F_M(x, \theta) \mid$$

le paramètre inconnu θ par son estimateur $\hat{\theta}(X_1, \ldots, X_n)$. On s'intéressera alors à la v.a.r. :

$$\hat{D}_n \equiv \sup_{x \in \mathbb{R}} \mid \mathbb{F}_n(x) - F_M\left[\, x, \hat{\theta}(X_1, \ldots, X_n)\,\right] \mid .$$

Mais sous $H_0^{(2)}\Gamma$ la suite des v.a.r. $\sqrt{n}\ \hat{D}_n$ ne converge pas forcément vers la loi de *Kolmogorov-Smirnov*.
Rien ne garantit par ailleurs que la loi asymptotique de la suite $\sqrt{n}\ \hat{D}_n$ ne dépend pas du paramètre inconnu θ.

Dans certains cas particuliersΓon peut cependant se ramener à des v.a.r. $\hat{D}_n$ dont la loi asymptotique ne dépend pas de θ.

David et Johnson [23] montrent par exemple que dans le cas où le paramètre θ est un paramètre réel de position (i.e. $F_M(x,\theta) = G(x-\theta)$)$\Gamma$ ou un paramètre d'échelle (i.e. $F_M(x,\theta) = G(x/\theta)$)$\Gamma$la suite $\sqrt{n}\ \hat{D}_n$ aΓsous certaines conditions sur l'estimateur $\hat{\theta}\Gamma$une loi asymptotique indépendante de θ.

Les conditions de David et Johnson sont vérifiées par exemple quand les lois considérées sont des lois normales ou exponentielles et quand les estimateurs utilisés sont ceux du maximum de vraisemblance.
Les tables des quantiles de la loi asymptotique de la suite de v.a.r. $\sqrt{n}\ \hat{D}_n$ (pour les lois normale et exponentielle) ont été fournies par Lilliefors [62] et [63].

Stephens [97] propose une variante de l'approche précédente où il estime le paramètre θ à partir d'une moitié (choix aléatoire) de l'échantillon $(X_i)_{i \leq n}$ notée : $(X_i^*)_{i \leq n/2}$.

En utilisant les résultats de Durbin [27] et Rao [82]ΓStephens montre que la loi asymptotique de la suite des v.a.r. :

$$\sqrt{n}\hat{D}_n^* \equiv \sqrt{n} \sup_{x \in \mathbb{R}} \mid \mathbb{F}_n(x) - F\left[\, x, \hat{\theta}(X_1^*, \ldots, X_{n/2}^*)\,\right] \mid$$

est la loi de *Kolmogorov-Smirnov*. On retrouve ainsi le *cas 0*. Ceci permet d'utiliser les tables standards des quantiles de la loi de *Kolmogorov-Smirnov*.

Une autre approche intéressante a été proposée par O'Reilly et Quesenberry [79].
Cette approche permetΓdans le cas où il existe une statistique exhaustive pour le paramètre θΓde se ramener au *cas 0* et doncΓà nouveauΓau test de *Kolmogorov-Smirnov*.

Cette approcheΓdécrite dans la suite de cette sous-sectionΓest basée sur les transformations intégrales de probabilité (*PIT*) (cf. [23] et [79]). Elle a l'avantage de traiter aussi bien le cas où les v.a.r. X_i sont i.i.d. (*cas 2*) que le cas général (*cas 3*).

Les transformations intégrales de probabilité (*PIT*)

Hypothèses – Les v.a.r. X_i ne sont plus supposées i.i.d. On se place dans la suite de ce chapitreΓsauf mention du contraireΓdans le cadre général du *cas 3*. –

On souhaite donc tester l'adéquation des observations $x_1, \ldots, x_n$ à un modèle paramétrique M spécifié par la structure statistique :

$$(\mathbb{R}_+{}^{\mathbb{N}}, \mathcal{B}(\mathbb{R}_+{}^{\mathbb{N}}), \mathcal{P}_M = \{P_\theta, \theta \in \Theta \subset \mathbb{R}^k\}).$$

Ce problème revient à tester l'hypothèse statistique :

$$H_0^{(3)} \equiv \text{ “ } P^* \in \mathcal{P}_M \text{ ” contre son alternative “ } P^* \notin \mathcal{P}_M \text{ ”.}$$

O'Reilly et Quesenberry [79] suggèrent de transformer la suite des v.a.r. $(X_i)_{i\leq n}$ en une suite de v.a.r. $(\bar{U}_i)_{i\leq n}$Γqui sontΓsous l'hypothèse nulle $H_0^{(3)}$Γi.i.d. de loi $Unif[0,1]$. Ce qui permet de se ramener au test de *Kolmogorov-Smirnov* standard.

Avant de donner le principal résultat de O'Reilly et QuesenberryΓ on donne d'abord quelques définitions et quelques notations.

Hypothèse – L'hypothèse principale dans l'approche de O'Reilly et Quesenberry est l'existence d'une statistique **exhaustive** HΓà valeurs dans $\mathbb{R}^l$Γpour le paramètre $\theta \in \Theta \subset \mathbb{R}^k$.
On notera dans la suite pour tout $i \leq n$:

$$H_i \equiv H(X_1, \ldots, X_i).$$

Notations – Sous l'hypothèse $H_0^{(3)}$Γla loi de probabilité P^* du processus $\{X_i\}_{i\geq 1}$ est égale à P_θ. Dans ce cas on note :

1. $F_n(x_1, \ldots, x_n; \theta)$ la fonction de répartition du vecteur $(X_1, \ldots, X_n)$.

2. Pour tout $i \leq n\Gamma$la fonction de répartition du vecteur $(X_1, \ldots, X_i)$ conditionnellement à H_n est notée :

$$F_i(x_1, \ldots, x_i \mid H_n) \equiv P(X_1 \leq x_1, \ldots, X_i \leq x_i \mid H_n).$$

3. Pour $i \leq n\Gamma$la fonction de répartition de X_i conditionnellement à $X_1, \ldots, X_{i-1}; H_n$ est notée :

$$\forall x_i \in \mathbb{R} \ \Gamma \ F_i(x_i \mid X_1, \ldots, X_{i-1}; H_n) \equiv P(X_i \leq x_i \mid X_1, \ldots, X_{i-1}; H_n).$$

Remarques –

1. Il est évidemment souhaitable que H ne soit exhaustive que relativement au modèle M dont on teste l'adéquation et non relativement à un sur-modèle qui inclurait une partie de la contre-hypothèse.

2. La fonction $F_n(x_1, \ldots, x_n \mid H_n)$ est l'estimateur de Rao-Blackwell de la fonction de répartition $F_n(x_1, \ldots, x_n; \theta)$.

3. L'exhaustivité de la statistique H fait que les fonctions :

$$F_i(x_1, \ldots, x_i \mid H_n) \text{ et } F_i(x_i \mid X_1, \ldots, X_{i-1}; H_n)$$

ne dépendent pas du paramètre θ.

On présente ci-dessus le résultat de O'Reilly et Quesenberry [79] permettant de se ramener au test de l'adéquation d'un échantillon à la loi $Unif[0,1]$ (*cas 0*) :

Théorème – 4.8 (O'Reilly et Quesenberry) *Soit α le plus grand entier positif inférieur ou égal à n tel que la fonction $F_\alpha(x_1, \ldots, x_\alpha \mid H_n)$ soit absolument continue.*
Sous l'hypothèse :

$$H_0^{(3)} \equiv \ \text{“ } P^* \in \{P_\theta, \theta \in \Theta \subset \mathbb{R}^k\} \text{ ”}$$

les v.a.r. :

$$\begin{aligned} \bar{U}_1 &= F_1(X_1 \mid H_n) \\ \bar{U}_2 &= F_2(X_2 \mid X_1; H_n) \\ &\ldots \\ \bar{U}_\alpha &= F_\alpha(X_\alpha \mid X_1, \ldots, X_{\alpha-1}; H_n) \end{aligned}$$

sont i.i.d. de loi $Unif[0,1]$.

Ces v.a.r. s'expriment en fonction des v.a.r. X_i indépendamment du paramètre θ.

Preuve – Une preuve détaillée de ce théorème est donnée dans O'Reilly et Quesenberry [79].
Notons que l'indépendance et la loi $Unif[0,1]$ des v.a.r. $\bar{U}_i$ sont une conséquence directe du théorème 4.5 de Rosenblatt.
L'utilisation de la statistique exhaustive H fait que les v.a.r. $\bar{U}_i$ sont indépendantes du paramètre inconnu θ. □

Le théorème précédent permet dans le cas où il existe une statistique exhaustive non triviale pour θ de ramener le test de l'hypothèse composée $H_0^{(3)}$ au test de l'hypothèse simple :

" les v.a.r. $\bar{U}_i$ sont i.i.d. de loi $Unif[0,1]$ ".

Pour pouvoir tester cette hypothèse simple il reste à obtenir les expressions explicites des v.a.r. $\bar{U}_i$ en fonction des X_i. Ces expressions vont dépendre du modèle considéré.

On présente ci-dessous deux exemples d'application du théorème de O'Reilly et Quesenberry. Le premier exemple dû à O'Reilly et Quesenberry (cf. [22] page 254) concerne l'adéquation d'un échantillon à une loi exponentielle.
Le deuxième exemple est donné par Gaudoin [37] il concerne le problème de l'adéquation au modèle *NHPP* de *Crow*.

Exemple 1 : test d'adéquation à une loi exponentielle

Hypothèse – Dans ce paragraphe les v.a.r. X_i sont supposées i.i.d. de fonction de répartition F^* inconnue. –

On souhaite tester l'adéquation des observations $x_1, \ldots, x_n$ à une loi exponentielle c'est-à-dire tester l'hypothèse nulle :

$$H_0^{(exp)} \equiv \text{ " } F^* = F_{exp}(.,\lambda) \text{ où } \lambda \in \mathbb{R}_+ \text{ et } F_{exp}(x,\lambda) = 1 - e^{-\lambda x} \text{ . "}$$

L'estimateur de maximum de vraisemblance de λ est donné par :

$$\hat{\lambda}_n = \frac{n}{\sum_{i=1}^n X_i},$$

et une statistique exhaustive est :

$$H_n = \sum_{i=1}^n X_i.$$

Le théorème de O'Reilly et Quesenberry ramène alors le problème du test de $H_0^{(exp)}$ au problème du test de l'adéquation des v.a.r. :

$$\bar{U}_i = F_i(X_i \mid X_1, \ldots, X_{i-1}; H_n)$$

à la loi $Unif[0,1]$.

Les expressions des v.a.r. $\bar{U}_i$ en fonction des X_i sont données par la proposition suivante :

Proposition – 4.9 (O'Reilly et Quesenberry) *Si les v.a.r. $X_1, \ldots, X_n$ sont i.i.d de loi $Exp(\lambda)$, $\lambda \in \mathbb{R}_+$, alors, pour $i = 1, \ldots, n-1$, les v.a.r. :*

$$\begin{aligned} \bar{U}_i &= F_i(X_i \mid X_1, \ldots, X_{i-1}; H_n) \\ &= 1 - \left(\frac{\sum_{j=i+1}^{n} X_j}{\sum_{j=i}^{n} X_j} \right)^{n-i} \end{aligned} \tag{4.2}$$

sont i.i.d. de loi $Unif[0,1]$.

Il en découle, par des changements d'indices, que les v.a.r. :

$$1 - \left(\frac{\sum_{j=1}^{i-1} X_j}{\sum_{j=1}^{i} X_j} \right)^{i-1} \quad , \quad i = 2, \ldots, n$$

sont aussi i.i.d. de loi $Unif[0,1]$.

La proposition précédente est une conséquence directe du théorème de O'Reilly et Quesenberry.
Le test de l'hypothèse composée $H_0^{(exp)}$ se ramène ainsi au test de *Kolmogorov-Smirnov*.

Exemple 2 : test d'adéquation au modèle de *Crow*

Le modèle de *Crow* présenté dans la section 1.4 est un modèle *NHPP* dont l'intensité de défaillance est donnée pour tout t réel positif par :

$$\lambda_{Cr}(t) = \alpha\beta\, t^{\beta-1} \text{ où } \alpha \text{ et } \beta \in \mathbb{R}_+^*.$$

Dans ce modèle les v.a.r. X_i ne sont pas i.i.d.

Notation – On note $P_{Cr(\alpha,\beta)}$ la loi de probabilité du processus aléatoire $\{X_i\}_{i\geq 1}$ dans le modèle de *Crow* de paramètres α et β.

Pour tester l'adéquation au modèle de *Crow* on s'intéresse au test de l'hypothèse :

$$H_0^{(Cr)} \equiv \text{ “ } P^* \in \{P_{Cr(\alpha,\beta)} \ \Gamma \ \alpha \text{ et } \beta \in \mathbb{R}_+^*\}. \text{ ”}$$

Remarques et Notations – Sous l'hypothèse $H_0^{(Cr)}$ on a (cf. proposition 4.6)

1. $P(X_i \leq x_i \mid X_1 = x_1, \ldots, X_{i-1} = x_{i-1}) = 1 - exp\,[\,-\alpha(t_i^\beta - t_{i-1}^\beta)\,]$.
2. Une statistique exhaustive bidimensionnelle est donnée par le couple :

$$[\,T_n, \sum_{i=1}^{n} ln(T_i)\,] \quad \text{où pour } i \leq n \;\Gamma\; T_i \equiv \sum_{j=1}^{i} X_j.$$

3. Pour $i \leq n\Gamma$la fonction de répartition de la loi de probabilité de $T_i\Gamma$conditionnellement à $T_1, \ldots, T_{i-1}$ et à $[\,T_n, \sum_{j=1}^{n} ln(T_j)\,]$ est notée :

$$\forall\, t_i \in \mathbb{R} \;\Gamma\; F_i\,[\,t_i \mid T_1, \ldots, T_{i-1}; T_n, \sum_{j=1}^{n} ln(T_j)\,].$$

Gaudoin [37] utilise le théorème de O'Reilly et Quesenberry pour montrer le résultat suivant :

Proposition – 4.10 (Gaudoin) *Sous l'hypothèse $H_0^{(Cr)}$, les v.a.r. V_i définies ci-dessous pour $i = 1, \ldots, n-2$ sont i.i.d. de lois $Unif[0,1]$. Leurs expressions ne dépendent pas des paramètres α et β :*

$$\begin{aligned} V_i &= F_i(T_i \mid T_1, \ldots, T_{i-1}; T_n, \sum_{j=1}^{n} ln(T_j)) \\ &= \frac{\phi(Y_i)}{\phi_i(Y_{i-1})} \end{aligned}$$

où :

- $\forall i \leq n\ , \quad Y_i \equiv ln(\dfrac{T_n}{T_i})$.
- *Les fonctions ϕ sont définies par :*

$$\phi_i(z) = \sum_{k=0}^{\lfloor \frac{1}{z} \sum_{j=i}^{n-1} Y_j \rfloor} C_{n-i}^{k} (-1)^k \left[\sum_{j=i}^{n-1} Y_j - kz \right]^{n-i-1}$$

  avec $\lfloor x \rfloor$ désignant la partie entière du réel x.

Le résultat précédent permet donc de se ramener au test d'adéquation à un échantillon de loi $Unif[0,1]$.

Rappelons que l'approche de O'Reilly et Quesenberry ne peut être utilisée que si l'on dispose de statistiques exhaustives non triviales. Ceci n'est pas le cas par exemple pour le modèle *MPD*.

Une deuxième difficulté pour l'application de cette méthode est le calcul des expressions des v.a.r. $\bar{U}_i$ du théorème de O'Reilly et Quesenberry. Ce calcul estΓcomme le montre l'exemple précédentΓsouvent très compliqué.

On présente dans la section suivante une approche alternative : la méthode du *u-plot*.

4.3 Un outil de mesure de la qualité prévisionnelle : le "*u-plot*"

En Fiabilité des LogicielsΓles modèles sont surtout utilisés pour prédire le comportement futur du processus de défaillance.
Ceci donne à la qualité prévisionnelle une grande importance dans le choix des modèles de fiabilité des logiciels.

Certains outils ont été proposés pour évaluer cette qualité prévisionnelle. Des listes assez exhaustives de ces outils sont données par Abdel-Ghali et al [1] et Ledoux [60].

L'un des outils les plus utilisés est le critère du *u-plot* initialement introduit par Littlewood et Verrall en 1973 [67]Γet étudié ensuite par Keiller et al [52]ΓAbdel-Ghali et al [1] et Downs et Scott [25].

On présente ci-dessous le critère du *u-plot* et on montre que cet outilΓprésenté initialement comme un indicateur graphique de qualité prévisionnelleΓpeut être utilisé dans certains cas comme un test d'adéquation statistique.

4.3.1 Cadre général et approche préquentielle

Hypothèses – On se place dans le même cadre que dans la section 4.2Γc'est-à-dire qu'on suppose que le processus $\{X_i\}_{i\geq 1}$ est de loi de probabilité P^* inconnue. –

On s'intéresse à la validation du modèle paramétrique M spécifié par la structure statistique paramétrique :

$$(\mathbb{R}_+{}^{\mathbb{N}}, \mathcal{B}(\mathbb{R}_+{}^{\mathbb{N}}), \mathcal{P}_M = \{P_\theta, \theta \in \Theta \subset \mathbb{R}^k\}).$$

On souhaite plus précisément évaluer la qualité prévisionnelle du modèle M au vu des observations $x_1, \ldots, x_n$ du vecteur $(X_1, \ldots, X_n)$.

L'approche prévisionnelle consiste à utiliser le modèle M et les observations $x_1, \ldots, x_n$ pour prédire le comportement futur du logiciel étudié en prédisant les lois de probabilité des v.a.r. X_{n+1}, X_{n+2}, etc.
Mesurer la qualité prédicitive revient à évaluer la qualité de ces prédictions.
Ceci peut être fait à l'aide de l'approche préquentielle (prédictive-séquentielle) présentée par Dawid [24] et décrite ci-dessous.

L'approche préquentielle

Définition – 4.11 ***L'approche préquentielle*** *est une approche itérative permettant d'évaluer la qualité prévisionnelle du modèle étudié.*
Chaque itération i de l'approche préquentielle se décompose en trois étapes :

1. *On divise les observations en deux groupes :*
$$(x_1, \ldots, x_i) \ \ et \ \ (x_{i+1}, \ldots, x_n)$$
2. *on prédit la loi de la v.a.r. X_{i+1} au vu uniquement des observations $x_1, \ldots, x_i$*
3. *cette prédiction est ensuite évaluée compte tenu de l'observation x_{i+1} de la v.a.r. X_{i+1}.*

En adoptant l'approche préquentielle on procéde donc comme si les observations arrivaient de manière séquentielle et qu'à la $i^{ème}$ itération on ne disposait que des observations $x_1, \ldots, x_i$ pour prédire la loi de probabilité de la v.a.r. X_{i+1}.

Notations – Dans la suite de cette section on note pour tout $i < n$:

1. $F^*_{i+1}(. \mid x_1, \ldots, x_i)$ la fonction de répartition de la "vraie" loi de probabilité de la v.a.r. X_{i+1} conditionnellement à $X_1 = x_1, \ldots, X_i = x_i$:
$$F^*_{i+1}(x \mid x_1, \ldots, x_i) = P^*(X_{i+1} \leq x \mid X_1 = x_1, \ldots, X_i = x_i).$$
2. $F_{i+1}(., \theta \mid x_1, \ldots, x_i)$ la fonction de répartition de la v.a.r. X_{i+1} conditionnellement à $X_1 = x_1, \ldots, X_i = x_i$ sous l'hypothèse "$P^* = P_\theta$" :
$$F_{i+1}(x, \theta \mid x_1, \ldots, x_i) = P_\theta(X_{i+1} \leq x \mid X_1 = x_1, \ldots, X_i = x_i).$$
3. Rappelons que $\hat{\theta}$ désigne un estimateur du paramètre θ.

A l'itération i de la procédure préquentielle la prédiction de la loi de X_{i+1} au vu des observations $x_1, \ldots, x_i$ peut se faire en estimant la fonction de répartion inconnue $F^*_{i+1}(. \mid x_1, \ldots, x_i)$ par :
$$F_{i+1}\left[\, ., \hat{\theta}(x_1, \ldots, x_i) \mid x_1, \ldots, x_i \,\right]. \tag{4.3}$$

Remarque – On peut aussi prédire la loi de X_{i+1} en utilisant une approche bayésienne où on considère une loi de probabilité a priori Π sur le paramètre θ.
Si on note $\Pi(\theta \mid x_1, \ldots, x_i)$ la densité a posteriori de θ la loi de probabilité de X_{i+1} peut alors être prédite en estimant $F^*_{i+1}(. \mid x_1, \ldots, x_i)$ par :
$$F^*_{i+1}(x \mid x_1, \ldots, x_i) \simeq \int_{\theta \in \Theta} F_{i+1}(x, \theta \mid x_1, \ldots, x_i)\ \Pi(\theta \mid x_1, \ldots, x_i)\, d\theta. \tag{4.4}$$

La difficulté de l'évaluation des prédictions précédentes vient du fait que pour tout $i \leq n$ on ne dispose que d'une seule observation x_{i+1} de la v.a.r. X_{i+1}.

Le critère du *u-plot* permet de contourner cette difficulté en donnant une évaluation globale de la qualité des prédictions (4.3).

4.3.2 Le critère du *u-plot*

Définitions et Notations

On présente ci-dessous quelques notations qu'on utilisera pour définir le critère du *u-plot*.

Notations – Soit p un entier fixé strictement inférieur à $n\Gamma$

1. on note $(\tilde{u}_i)_{p+1\leq i\leq n}$ les réalisations des v.a.r. $(\tilde{U}_i)_{p+1\leq i\leq n}$ définies par :

$$\begin{aligned}\tilde{U}_{p+1} &= F_{p+1}\,[\,X_{p+1},\hat{\theta}(X_1,\ldots,X_p)\mid X_1,\ldots,X_p\,]\\ \tilde{U}_{p+2} &= F_{p+2}\,[\,X_{p+2},\hat{\theta}(X_1,\ldots,X_{p+1})\mid X_1,\ldots,X_{p+1}]\\ &\ldots\\ \tilde{U}_n &= F_n\,[\,X_n,\hat{\theta}(X_1,\ldots,X_{n-1})\mid X_1,\ldots,X_{n-1}\,]\end{aligned}$$

2. on note $\tilde{\mathbb{F}}_{n,p}$ la fonction de répartition empirique associée aux v.a.r. $(\tilde{U}_i)_{p+1\leq i\leq n}$:

$$\forall x\in\mathbb{R}\ \Gamma\ \tilde{\mathbb{F}}_{n,p}(x)=\frac{1}{n-p}\sum_{i=p+1}^{n}1_{\{\tilde{U}_i\leq x\}}.$$

3. On note $(\tilde{U}_i^*)_{1\leq i\leq n-p}$ la suite ordonnée (croissante) obtenue à partir de la suite de v.a.r. $(\tilde{U}_i)_{p<i\leq n}$.

La qualité prévisionnelle d'un modèle M peut être mesurée par le critère du *u-plot* défini ci-dessous :

Définition – 4.12 *On appelle* ***critère du*** **u-plot** *la distance de* Kolmogorov-Smirnov *entre la fonction de répartition empirique de la suite* $(\tilde{u}_i)_{p+1\leq i\leq n}$ *et la fonction de répartition de la loi* $Unif[0,1]$.
Cette distance de Kolmogorov-Smirnov *est la réalisation de la v.a.r.* $\tilde{D}_{n,p}$ *donnée par :*

$$\begin{aligned}\tilde{D}_{n,p} &= \sup_{t\in[0,1]}\mid\tilde{\mathbb{F}}_{n,p}(t)-t\mid\\ &= max\,[\,\max_{1\leq i\leq n-p}(\frac{i}{n-p}-\tilde{U}_i^*)\ ,\ \max_{1\leq i\leq n-p}(\tilde{U}_i^*-\frac{i-1}{n-p})\,]\end{aligned}$$

Keiller et al [52] appellent **u-plot** *le graphe de la fonction de répartition empirique de la suite* $(\tilde{u}_i)_{p<i\leq n}$.

Remarques –

1. Il faut bien noter l'aspect séquentiel de l'approche *u-plot*Γpuisque les différents $\tilde{u}_i$ sont calculés à partir d'estimations différentes du paramètre θ.

2. Le choix de l'entier p est laissé à l'utilisateur. On peut cependant remarquer qu'un faible entier p engendre une mauvaise qualité des premières estimations de θ : $\hat{\theta}(x_1, \ldots, x_p)\Gamma\hat{\theta}(x_1, \ldots, x_{p+1})\Gamma$etc.
   Ce choix dépendra donc de la qualité de l'estimateur $\hat{\theta}$.

Définition – 4.13 *En pratique, on remplace souvent le critère* u-plot *associé à la v.a.r.* $\tilde{D}_{n,p}$ *par* ***l'indice*** **u-plot** *réalisation de la v.a.r.* $\tilde{K}_{n,p}$ *définie par :*

$$\tilde{K}_{n,p} = \sqrt{n-p}\,\tilde{D}_{n,p}.$$

L'indice *u-plot* permet d'éliminer l'effet taille des jeux de données.

Justification du critère du *u-plot*

On montre ci-dessous que des modèles qui ont de bonnes qualités prévisionnelles auront forcément des critères *u-plot* de faibles valeurs.

En effetΓsi le modèle étudié a un bon pouvoir prédictifΓles estimations (4.3) seront de bonne qualité.
Les fonctions prédictives $F_{i+1}\,[\,.,\hat{\theta}(x_1, \ldots, x_i) \mid x_1, \ldots, x_i\,]$ vérifieront alorsΓdans une certaine mesureΓles propriétés des fonctions $F^*_{i+1}(. \mid x_1, \ldots, x_i)$.

Le critère du *u-plot* est associé à la propriété de Rosenblatt (cf. théorème (4.5)) que vérifient les fonctions $F^*_{i+1}(. \mid x_1, \ldots, x_i)$:

Les v.a.r. U_i définies ci-dessous sont i.i.d. de loi $Unif[0,1]$:

$$\begin{aligned} U_1 &= F^*_1(X_1) \\ U_2 &= F^*_2(X_2 \mid X_1) \\ \ldots & \\ U_n &= F^*_n(X_n \mid X_1, \ldots, X_{n-1}) \end{aligned}$$

Si les estimations (4.3) sont de bonne qualitéΓles v.a.r. $\tilde{U}_i$ seront "proches" des v.a.r. U_i.
La suite $(\tilde{u}_i)_{1 \le i \le n-p}$ sera alors "proche" d'un échantillon de loi $Unif[0,1]$.

AinsiΓle modèle M sera validé si la fonction de répartition empirique de la suite $(\tilde{u}_i)_{1 \le i \le n-p}$ (i.e. le *u-plot*) est assez proche de la fonction de répartition de la loi $Unif[0,1]\Gamma$c'est-à-dire de la diagonale "$y = x$".
Cette proximité est mesurée par le critère du *u-plot* qui est la distance verticale maximale entre le *u-plot* et la diagonale.

Exemple – On considère le jeu de données *simulMPD.d* simulé à partir du modèle *MPD*.

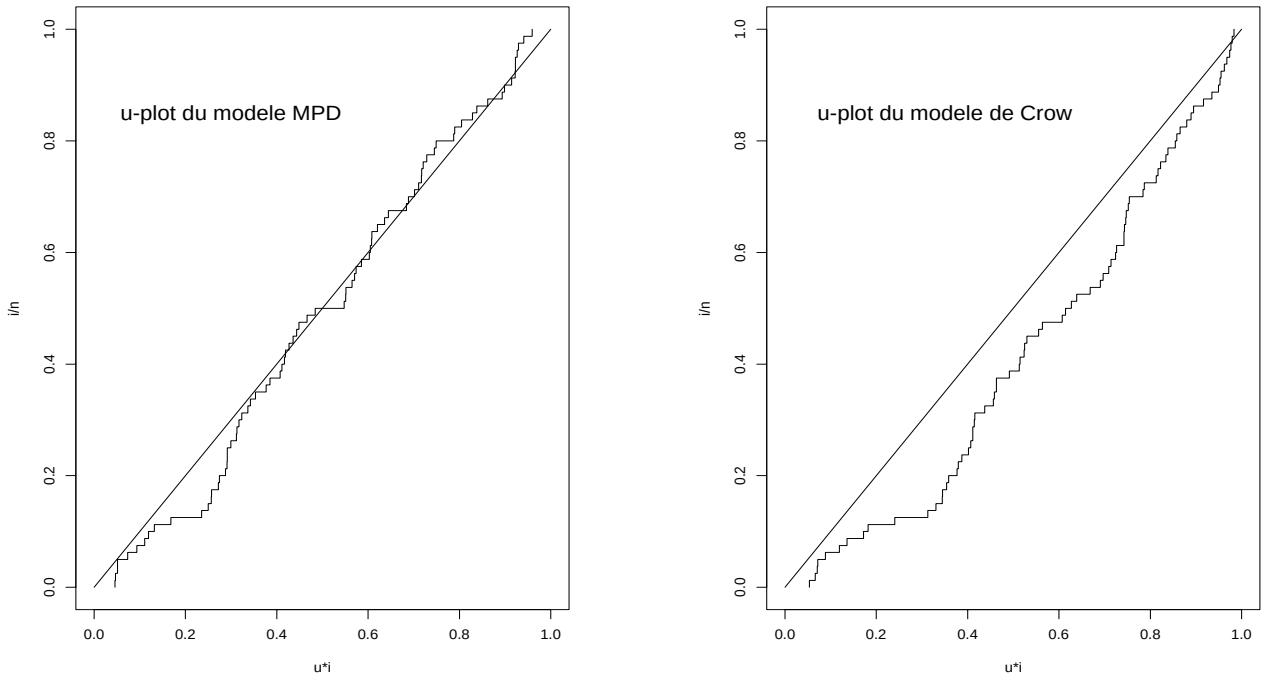

FIG. 4.1: Adéquation des données *simulMPD.d* au *MPD* et au modèle de *Crow*

La figure 4.1 représente les *u-plots* du *MPD* et du modèle de *Crow* relatifs au jeu de données *simulMPD.d*.
Il en ressort clairement que le *MPD* a une meilleure qualité prévisionnelle que le modèle de *Crow* puisque les valeurs de l'indice *u-plot* sont $\tilde{K}_{100,20} = 1.00$ pour le *MPD* et $\tilde{K}_{100,20} = 1.74$ pour le modèle de *Crow*.

Remarque – Keiller et al [52] précisent que le critère du *u-plot* est un simple indicateur graphique. Ils mettent en garde contre son utilisation comme un test statistique.

4.3.3 Le critère du *u-plot* vu comme un test statistique : justifications empiriques

On se pose ici la question de pouvoir utiliser le critère du *u-plot* comme un test d'adéquation statistique. C'est-à-dire peut-on comparer la réalisation $\tilde{k}_{n,p}$ de la v.a.r. $\tilde{K}_{n,p}$ à certains quantiles pour rejeter ou non le modèle M à un certain de seuil de signification ?

Cette question est liée à la déterminationΓsous l'hypothèse nulle :

$$\text{“ } P^* \in \mathcal{P}_M = \{P_\theta \; \Gamma\theta \in \Theta \subset \mathbb{R}^k\} \text{ ”}$$

de la loi de probabilité (asymptotique) de la v.a.r. indice *u-plot* $\tilde{K}_{n,p}$.

Aucune raison ne permet de dire a priori que les v.a.r. $\tilde{K}_{n,p}$ convergent en loi vers la loi de *Kolmogorov-Smirnov* ou vers une autre loi indépendante du paramètre θ.

Les résultats de Downs et Scott

Downs et Scott [25] montrent par des simulationsΓque dans les deux cas suivants :

- les v.a.r. X_i sont i.i.d. de loi exponentielle
- les v.a.r. X_i sont issues du modèle de *Jelinski-Moranda*

les lois empiriques des v.a.r. $\tilde{K}_{n,p}$ (pour n et p assez élevés) s'approchent d'une façon étonnante de la loi de *Kolmogorov-Smirnov*.

Ce résultat semble être dû à certaines propriétés de l'approche préquentielle.
En effet les lois des v.a.r. précédentes s'écartent clairement de la loi de *Kolmogorov-Smirnov* dès qu'on remplace l'approche préquentielle par une approche inférentielle classique où le paramètre θ est estimé une fois pour toutes en utilisant toutes les observations $x_1, \ldots, x_n$.

Downs et Scott [25] énoncent à partir de leurs résultats de simulation la conjecture suivante :

Conjecture – 4.14 (Downs et Scott) *Il est possible dans le cas des modèles de fiabilité des logiciels d'utiliser le critère du* u-plot *comme un test d'adéquation statistique.*
Les quantiles correspondants sont ceux de la loi de Kolmogorov-Smirnov.

Downs et Scott ne donnent cependant aucune raison théorique permettant d'expliquer leurs résultats empiriques.

On retrouve dans les paragraphes suivants les résultats empiriques de Downs et Scott (notamment le cas i.i.d. exponentiel).
On généralise ensuite ces résultats empiriques au *MPD* et au modèle de *Crow*.

On essaiera ensuite de donner une justification théorique à l'utilisation du critère du *u-plot* comme un test d'adéquation statistique.

Recherche des lois empiriques des v.a.r. $\tilde{K}_{n,p}$

On décrit dans ce paragraphe l'approche adoptée pour vérifier empiriquement la conjecture 4.14 qu'on réécrit sous la forme suivante :

Conjecture – 4.15 *Pour certains modèles, et pour des estimateurs particuliers de θ, la suite des v.a.r. indice* u-plot $\tilde{K}_{n,p}$ *converge en loi vers la loi de* Kolmogorov-Smirnov *:*

$$\tilde{K}_{n,p} \overset{n\to+\infty,\, Loi}{\longrightarrow} \mathcal{L}_{KS}.$$

Pour chaque modèle M on va vérifier la conjecture précédente en simulant plusieurs jeux de données à partir des hypothèses du modèle considéré. Ces simulations permettent ensuite de trouver la loi empirique de la v.a.r. $\tilde{K}_{n,p}$ sous l'hypothèse nulle " $P^* \in \mathcal{P}_M$ ". Les lois empiriques ainsi obtenues sont alors comparées à la loi de *Kolmogorov-Smirnov*.

Pour chacun des modèles considérésΓl'approche pratique qui sera adoptée peut se décomposer en cinq étapes :

1. on choisit une valeur particulière θ_0 du paramètre θ
2. on simule à partir de la loi P_{θ_0} m jeux de données différentsΓtous de taille n :

$$\begin{aligned} 1^{er} \text{ jeu :} &\quad x_1^{(1)}, \ldots, x_n^{(1)} \\ 2^{ème} \text{ jeu :} &\quad x_1^{(2)}, \ldots, x_n^{(2)} \\ &\ldots \\ m^{ème} \text{ jeu :} &\quad x_1^{(m)}, \ldots, x_n^{(m)} \end{aligned}$$

3. Pour chacun de ces jeux de données on calcule la réalisation $\tilde{k}_{n,p}^{(j)}$ de la v.a.r. $\tilde{K}_{n,p}$.
4. On trace la fonction de répartition empirique de la suite $(\tilde{k}_{n,p}^{(j)})_{1\le j\le m}$.
   Cette fonction est une estimation de la fonction de répartition de la loi de $\tilde{K}_{n,p}$ sous l'hypothèse nulle " $P^* \in \mathcal{P}_M$ ".
5. La fonction de répartition empirique tracée à l'étape 4 est comparée à la fonction de répartition de la loi de *Kolmogorov-Smirnov*.

Remarques –

1. Pour chacun des exemples qui seront traités ci-dessousΓon trace de la même façon que pour l'indice *u-plot* $\tilde{K}_{n,p}$Γla fonction de répartition empirique de la variable **indice complet** $\hat{K}_{n,p}$ où le paramètre θ est estimé une fois pour toutes en utilisant l'échantillon **complet** des observations :

   $$\hat{K}_{n,p} = \sqrt{n-p}\; max\,[\, \max_{1\le i\le n-p}(\frac{i}{n-p} - \hat{U}_i^*)\; \Gamma\; \max_{1\le i\le n-p}(\hat{U}_i^* - \frac{i-1}{n-p})\,]$$

   où toutes les v.a.r. $\hat{U}_i$ s'expriment en fonction du même estimateur de θ :

   $$\text{Pour } i=p+1,\ldots,n,\ \hat{U}_i = F_i\,[\, X_i, \hat{\theta}(X_1,\ldots,X_n) \mid X_1,\ldots,X_{i-1}\,].$$

2. Dans les exemples traités ci-dessous, et sauf mention du contraire, on prend :

$$n = 100, p = 20 \text{ et } m = 1000.$$

3. Les estimateurs $\hat{\theta}$ utilisés sont ceux du maximum de vraisemblance.

Cas où les X_i sont i.i.d. de loi exponentielle

Le modèle considéré dans ce paragraphe est spécifié par l'hypothèse selon laquelle les v.a.r. X_i sont i.i.d. de loi exponentielle :

$$\forall i \geq 1, \ X_i \sim Exp(\lambda), \ \lambda \in \mathbb{R}_+^*.$$

On souhaite vérifier empiriquement la conjecture 4.15 qui s'exprime ici sous la forme suivante :

Conjecture – 4.16 *Si les v.a.r. X_i sont i.i.d. de loi $Exp(\lambda)$, on a alors :*

$$\tilde{K}_{n,p} \overset{n\to+\infty, Loi}{\longrightarrow} \mathcal{L}_{KS}$$

où

- *la v.a.r. $\tilde{K}_{n,p}$ est donnée par la formule suivante :*

$$\tilde{K}_{n,p} = \sqrt{n-p}\,[\sup_{t\in[0,1]} \mid \frac{1}{n-p}\sum_{i=p+1}^{n} 1_{\{\tilde{U}_i \leq t\}} - t \mid]$$

- *Pour $i = p+1, \ldots, n$:*

$$\begin{aligned} \tilde{U}_i &= F_i(X_i, \hat{\lambda}(X_1, \ldots, X_{i-1}) \mid X_1, \ldots, X_{i-1}) \\ &= F_{exp}(X_i, \hat{\lambda}(X_1, \ldots, X_{i-1})) \\ &= 1 - exp\,[-\frac{(i-1)\,X_i}{\sum_{j=1}^{i-1} X_j}] \end{aligned}$$

En utilisant l'approche simulative décrite précédemment on obtient les fonctions de répartition empiriques représentées sur la figure 4.2.

Il est clair que la fonction de répartition empirique de l'indice *u-plot* $\tilde{K}_{n,p}$ est très proche de la fonction de répartition de la loi de *Kolmogorov-Smirnov*. Ceci n'est plus vrai pour l'indice complet $\hat{K}_{n,p}$.

Ce résultat semble par ailleurs indépendant de la valeur λ_0 à partir de laquelle on simule les jeux de données.

Ces résultats empiriques suggèrent ainsi l'utilisation du critère du *u-plot* comme un test d'adéquation statistique à un modèle où les v.a.r. X_i sont i.i.d. de loi exponentielle.

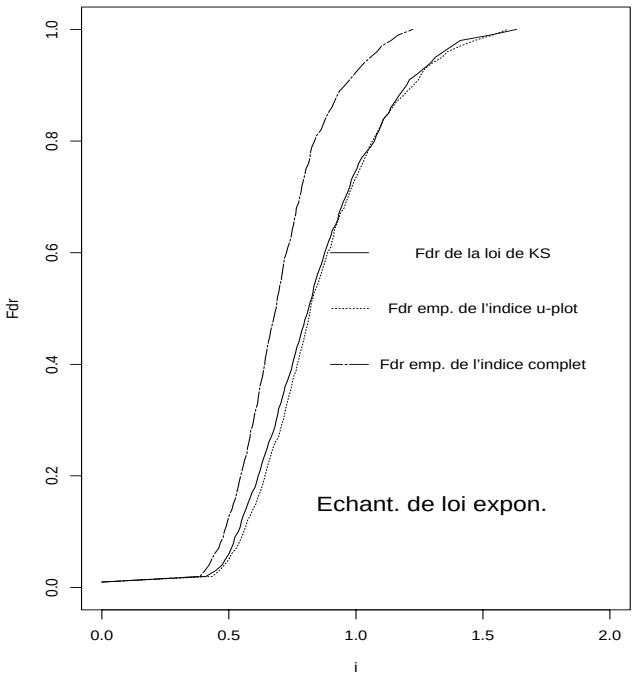

FIG. 4.2: Loi empirique de l'indice *u-plot* $\tilde{K}_{n,p}$ sous l'hypothèse $H_0^{(exp)}$.

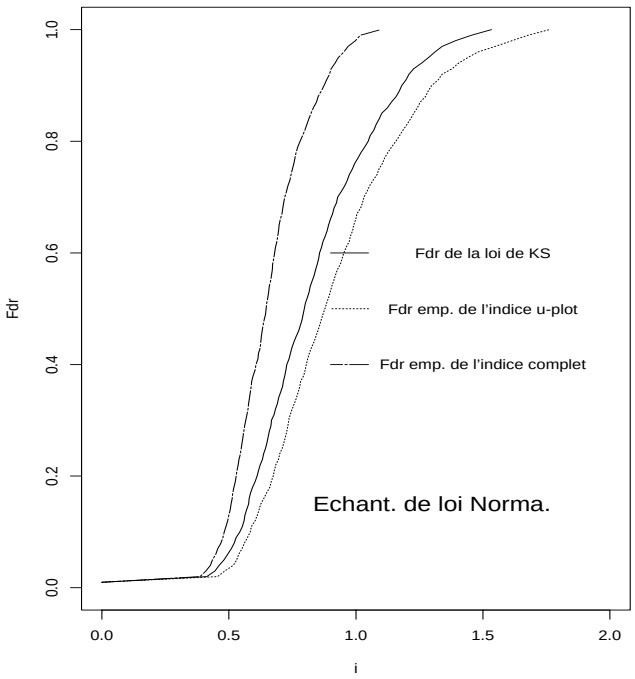

FIG. 4.3: Loi empirique de l'indice *u-plot* $\tilde{K}_{n,p}$ pour un échantillon gaussien

Remarque – Comme le montre la figure 4.3Γla conjecture 4.16 est fausse dans le cas de v.a.r. X_i i.i.d. de loi gaussienne $\mathcal{N}(m, \sigma^2)$.

Cas du modèle *MPD*

Rappelons que dans le modèle *MPD* de paramètres $\lambda \in \mathbb{R}_+$ et $\theta \in \mathbb{R}$ on a l'hypothèse suivante :

"$X_1, X_2, \ldots$ sont des v.a.r. indépendantes de lois :

$$X_i \sim Exp(\lambda e^{-\theta(i-1)}) \text{ "}$$

$\boldsymbol{H_{MPD}}$

Remarques – Rappelons aussi que :

1. sous les hypothèses du *MPD* on a pour tout entier i et pour tout $x \in \mathbb{R}_+$:

$$F_{i+1}(x; \lambda, \theta \mid x_1, \ldots, x_i) = 1 - exp\,[\,-\lambda e^{-\theta(i-1)}\ x\,]. \tag{4.5}$$

2. Les estimateurs de maximum de vraisemblance des paramètres λ et θΓnotés respectivement $\hat{\lambda}_n$ et $\hat{\theta}_n$ vérifient les équations :

$$\begin{cases} \hat{\lambda}_n = \dfrac{n}{\sum_{i=1}^{n} e^{-\hat{\theta}_n (i-1)} X_i} \\ \\ \sum_{i=1}^{n} (n-2i+1)\, e^{-\hat{\theta}_n (i-1)} X_i = 0 \end{cases}$$

Pour justifier empiriquement l'utilisation du critère *u-plot* comme un test d'adéquation statistique au modèle *MPD*Γil faut vérifier la conjecture suivante :

Conjecture – 4.17 *Sous les hypothèses du* MPD *on a :*

$$\tilde{K}_{n,p} \overset{n \to +\infty, Loi}{\longrightarrow} \mathcal{L}_{KS}$$

où

- *la v.a.r.* $\tilde{K}_{n,p}$ *est donnée par la formule suivante :*

$$\tilde{K}_{n,p} = \sqrt{n-p}\ \left[\sup_{t \in [0,1]} \mid \frac{1}{n-p} \sum_{i=p+1}^{n} 1_{\{\tilde{U}_i \leq t\}} - t \mid \right]$$

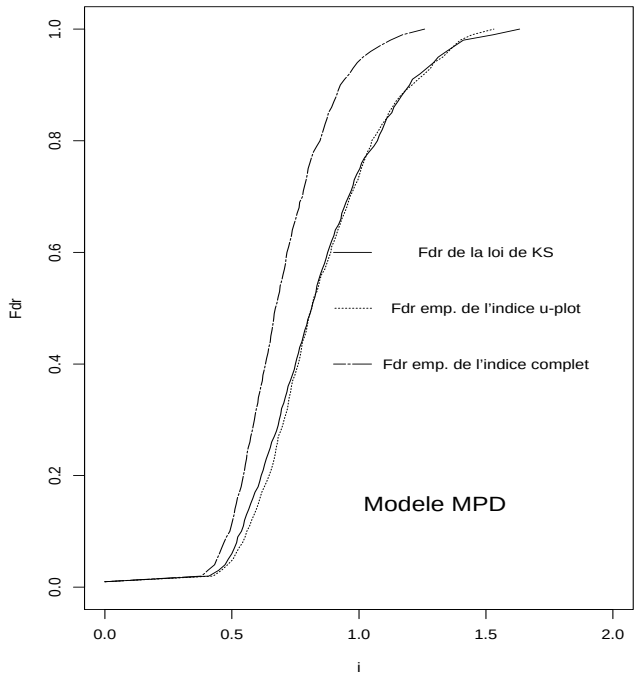

FIG. 4.4: Loi empirique de l'indice *u-plot* $\tilde{K}_{n,p}$ sous les hypothèses du *MPD* .

- *Pour* $i = p+1, \ldots, n$:

$$\begin{aligned} \tilde{U}_i &= F_i\left[\, X_i, \hat{\lambda}(X_1, \ldots, X_{i-1}), \hat{\theta}(X_1, \ldots, X_{i-1}) \mid X_1, \ldots, X_{i-1} \,\right] \\ &= 1 - exp\left[\hat{\lambda}(X_1, \ldots, X_{i-1})\, exp\left[\, -(i-1)\, \hat{\theta}(X_1, \ldots, X_{i-1}) \,\right] X_i\right]. \end{aligned}$$

On utilise à nouveau l'approche simulative décrite précédemment pour obtenir la fonction de répartition empirique de la v.a.r. indice *u-plot* $\tilde{K}_{n,p}$ sous les hypothèses du *MPD*. Cette fonction est représentée sur la figure 4.4.

Sous les hypothèses du *MPD*Γla loi empirique de la v.a.r. $\tilde{K}_{n,p}$ s'approche donc de la loi de *Kolmogorov-Smirnov*.

Cas du modèle *Crow*

Les résultats empiriques présentés dans les deux exemples précédents restent vrais dans le cas du modèle de *Crow*.

Remarques –

1. Rappelons que le modèle de *Crow* est un modèle *NHPP* d'intensité de défaillance donnée pour tout réel positif t par :

$$\lambda_{Cr}(t) = \alpha\beta\, t^{\beta-1} \quad \text{où } \alpha \text{ et } \beta \in \mathbb{R}_+^*.$$

2. Les estimateurs du maximum de vraisemblance des paramètres α et β notés respectivement $\hat{\alpha}_n$ et $\hat{\beta}_n$ sont :

$$\hat{\alpha}_n \equiv \hat{\alpha}(X_1, \ldots, X_n) = \frac{n}{T_n^{\hat{\beta}_n}} \quad \text{et} \quad \hat{\beta}_n \equiv \hat{\beta}(X_1, \ldots, X_n) = \frac{n}{\sum_i^n ln(T_n/T_i)}.$$

3. D'après la proposition 4.6 on aΓsous l'hypothèse $H_0^{(Cr)}$Γpour tout entier $i \leq n$:

$$\forall x \in \mathbb{R}_+ \Gamma\ F_i(x; \alpha, \beta \mid x_1, \ldots, x_{i-1}) = 1 - exp\,[\,-\alpha(t_i + x)^\beta + \alpha t_i^\beta\,].$$

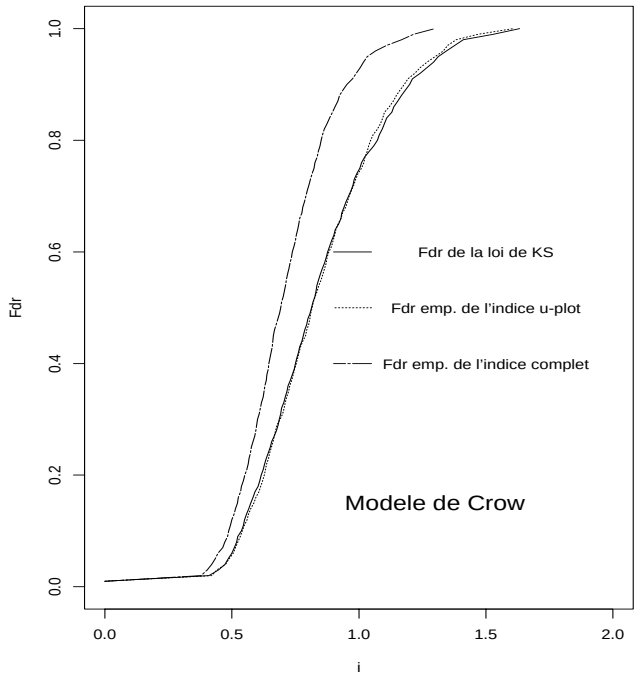

FIG. 4.5: Loi empirique de l'indice *u-plot* $\tilde{K}_{n,p}$ sous les hypothèses du modèle de *Crow*.

L'utilisation du critère du *u-plot* comme un test d'adéquation statistique au modèle de *Crow* est justifiée par la conjecture suivante :

Conjecture – 4.18 *Sous l'hypothèse* $H_0^{(Cr)}$ *du modèle de* Crow *on a :*

$$\tilde{K}_{n,p} \overset{n \to +\infty, Loi}{\longrightarrow} \mathcal{L}_{KS}$$

où

- *la v.a.r.* $\tilde{K}_{n,p}$ *est donnée par la formule suivante :*

$$\tilde{K}_{n,p} = \sqrt{n-p}\,[\,\sup_{t \in [0,1]}\,|\,\frac{1}{n-p}\sum_{i=p+1}^{n} 1_{\{\tilde{U}_i \leq t\}} - t\,|\,]$$

- *Pour* $i=p+1,\ldots,n$ *:*

$$\begin{aligned}\tilde{U}_i &= F_i\left[\, X_i, \hat{\alpha}(X_1,\ldots,X_{i-1}), \hat{\beta}(X_1,\ldots,X_{i-1}) \mid X_1,\ldots,X_{i-1}\,\right] \\ &= 1 - exp(-\hat{\alpha}_{i-1} T_i^{\hat{\beta}_{i-1}} + \hat{\alpha}_{i-1} T_{i-1}^{\hat{\beta}_{i-1}}).\end{aligned}$$

Une justification empirique de la conjecture précédente vient du fait que la fonction de répartition empirique de la v.a.r. $\tilde{K}_{n,p}$Γreprésentée sur la figure 4.5Γest très proche de la fonction de répartition de la loi de *Kolmogorov-Smirnov*.
Ceci n'est plus vrai pour l'indice complet $\hat{K}_{n,p}$ où l'approche préquentielle est remplacée par une approche inférentielle classique.

4.3.4 Un test préquentiel d'adéquation à la loi exponentielle

On donne dans cette sous-section une explication théorique aux résultats empiriques présentés précédemment dans le cas où les v.a.r. X_i sont i.i.d. de loi exponentielle.

Le premier intérêt des résultats théoriques présentés ci-dessous provient du fait qu'ils représentent une première étape de l'étude théorique des propriétés du critère du *u-plot*.

Ces résultats théoriques fournissent par ailleurs un nouveau test d'adéquation à une loi exponentielle de paramètre inconnu. Dans ce testΓqu'on peut appeler **test d'adéquation préquentiel**Γon utilise l'approche préquentielle pour se ramener à un test de *Kolmogorov-Smirnov* standard.

Hypothèse – Dans ce paragrapheΓles v.a.r. X_i sont supposées i.i.d. de fonction de répartition F^* inconnue. –

On souhaite tester l'adéquation des observations $x_1,\ldots,x_n$ à une loi exponentielleΓc'est-à-dire tester l'hypothèse nulle :

$$H_0^{(exp)} \equiv \text{ “ } F^* = F_{exp}(.,\lambda) \text{ où } \lambda \in \mathbb{R}_+ \text{ et } F_{exp}(x,\lambda) = 1 - e^{-\lambda x} \text{ . ”}$$

Le test d'adéquation préquentiel est basé sur le théorème suivant qui démontre la conjecture 4.16 :

Théorème – 4.19 *Sous l'hypothèse* $H_0^{(exp)}$*, la suite de v.a.r.* $\tilde{K}_{n,1}$ *définies ci-dessous converge en loi vers la loi de* Kolmogorov-Smirnov *:*

$$\tilde{K}_{n,1} \overset{n\to+\infty,\, Loi}{\longrightarrow} \mathcal{L}_{KS}$$

où :

- *les v.a.r.* $\tilde{K}_{n,1}$ *sont données par la formule suivante :*

$$\tilde{K}_{n,1} = \sqrt{n-1}\,\left[\, \sup_{t\in[0,1]} \mid \frac{1}{n-1}\sum_{i=2}^{n} 1_{\{\tilde{U}_i \le t\}} - t \mid \right]$$

- *Pour $i=2,\ldots,n$ les variables $\tilde{U}_i$ sont obtenues par une approche préquentielle :*

$$\begin{aligned}\tilde{U}_i &= F_{exp}\left[\, X_i, \hat{\lambda}(X_1,\ldots,X_{i-1})\,\right] \\ &= 1 - exp\left[-\frac{(i-1)\,X_i}{\sum_{j=1}^{i-1} X_j}\right]\end{aligned}$$

Le résultat précédent permet de définir un nouveau test d'adéquation à la loi exponentielle de paramètre inconnu :

Définition – 4.20 *Le **test d'adéquation préquentiel** consiste à tester l'hypothèse $H_0^{(exp)}$ en comparant la valeur $\tilde{k}_{n,1}$ aux quantiles de la loi de* Kolmogorov-Smirnov. *$\tilde{k}_{n,1}$ étant la réalisation de la v.a.r. indice* u-plot $\tilde{K}_{n,1}$ *associée aux observations $x_1,\ldots,x_n$ des v.a.r. $X_1,\ldots,X_n$.*

Remarque – Le paramètre p du critère *u-plot* (cf. sous-section 4.3.2) est choisi ici égal à un. Tous les résultats énoncés dans cette sous-section restent vrais (en remplaçant $\tilde{k}_{n,1}$ par $\tilde{k}_{n,p}$) pour un entier p fixé strictement inférieur à n.

La démonstration du théorème 4.19 sera faite en trois étapes :

- **Etape 1 :** On donne quelques propriétés des v.a.r. $\tilde{U}_i$ définies au théorème 4.19.

  On montre notamment que ces v.a.r. sont indépendantes mais non identiquement distribuéesΓet que la suite $(\tilde{U})_{i\geq 2}$ converge en loi vers la loi $Unif[0,1]$.

- **Etape 2 :** On présente un théorème dû à Shorack [88] donnant des conditions de convergence en loi de processus empiriques obtenus à partir d'une suite de v.a.r. indépendantes mais non identiquement distribuées.

- **Etape 3 :** On utilise enfin les résultats énoncés dans les deux étapes précédentes pour prouver le théorème 4.19.

Ces trois étapes sont détaillées ci-dessous.

Etape 1 : Propriétés des v.a.r. $\tilde{U}_i$

On s'intéresse ici aux propriétés des v.a.r. $\tilde{U}_i$ définies au théorème 4.19.

Proposition – 4.21 *Si les v.a.r. $X_1, \ldots, X_n$ sont i.i.d de loi $Exp(\lambda)$, $\lambda \in \mathbb{R}_+$, alors, pour $i = 2, \ldots, n$, les v.a.r. :*

$$\tilde{U}_i = 1 - exp\left[-\frac{(i-1)\, X_i}{\sum_{j=1}^{i-1} X_j} \right]$$

sont des v.a.r. indépendantes.

Preuve – La proposition 4.9 permet de montrer l'indépendance des v.a.r. :

$$\frac{\sum_{j=1}^{i-1} X_j}{\sum_{j=1}^{i} X_j} \; \Gamma \quad \text{pour } i = 2, \ldots, n$$

on en déduit l'indépendance des v.a.r. $\frac{\sum_{j=1}^{i} X_j}{\sum_{j=1}^{i-1} X_j}$Γet par conséquent celle des v.a.r. $\frac{X_i}{\sum_{j=1}^{i-1} X_j}$. D'où le résultat énoncé.

□

Proposition – 4.22 *Les lois de probabilité des v.a.r. $\tilde{U}_2, \ldots, \tilde{U}_n$ définies ci-dessus sont données par leurs fonctions de répartition notées $F_{\tilde{U}_i}$ qui s'écrivent pour $i = 2, \ldots, n$ $F_{\tilde{U}_i}(1) = 1$ et pour tout $u \in [0, 1[$:*

$$F_{\tilde{U}_i}(u) \equiv P(\tilde{U}_i \leq u) = 1 - \left[1 - \frac{ln(1-u)}{i-1} \right]^{-(i-1)}. \tag{4.6}$$

Preuve – Pour $i = 2, \ldots, n$ et pour $u \in [0, 1[$ on a :

$$\begin{aligned} P(\tilde{U}_i \leq u) &= P\left[1 - exp\,[\, -\frac{(i-1)\, X_i}{\sum_{j=1}^{i-1} X_j}\,] \leq u \right] \\ &= P\left[\frac{X_i}{\sum_{j=1}^{i-1} X_j} \leq -\frac{ln(1-u)}{i-1} \right] \end{aligned} \tag{4.7}$$

orΓpour $i = 2, \ldots, n$ on a :

$$X_i \sim Exp(\lambda) \equiv Gamma(1, \lambda) \;\text{ et }\; \sum_{j=1}^{i-1} X_j \sim Gamma(i-1, \lambda)$$

le rapport $\frac{X_i}{\sum_{j=1}^{i-1} X_j}$ est donc une v.a.r de loi $Beta(1, i-1)$ de fonction de répartition :

$$\forall x \in \mathbb{R}_+ \Gamma \; F_{Beta(1,i-1)}(x) = 1 - (1+x)^{-(i-1)}.$$

En utilisant le résultat précédent dans l'équation (4.7) on obtient le résultat énoncé. □

Corollaire – 1 La suite des v.a.r $\tilde{U}_i$ converge en loi vers la loi $Unif[0,1]$. On a en effet :

$$\forall u \in [0,1],\ F_{\tilde{U}_i}(u) \overset{i\to+\infty}{\longrightarrow} u. \tag{4.8}$$

Etape 2 : Théorème de Shorack

Le résultat énoncé ci-dessous a été présenté par Shorack [88] :

Théorème – 4.23 (Shorack) *Soient $U_1, \ldots, U_n$ des v.a.r. indépendantes de fonctions de répartition respectives $G_1, \ldots, G_n$ concentrées sur $[0,1]$.*
On note :

- *W_n la suite de processus aléatoires définie pour $t \in [0,1]$ par :*

$$W_n(t) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} [\, 1_{\{U_i \le t\}} - G_i(t) \,]$$

- *μ_n la fonction de covariance du processus W_n, définie pour s et t dans $[0,1]$ par :*

$$\begin{aligned} \mu_n(s,t) &= Cov\,[\, W_n(s), W_n(t) \,] \\ &= \frac{1}{n} \sum_{i=1}^{n} [\, G_i(min(s,t)) - G_i(s) G_i(t)] \,. \end{aligned}$$

- *$\bar{G}_n$ la suite de fonctions définies sur $[0,1]$ par :*

$$\bar{G}_n(t) = \frac{1}{n} \sum_{i=1}^{n} G_i(t)$$

Si *il existe deux fonctions réelles $\bar{G}$ (continue) et μ définies respectivement sur $[0,1]$ et $[0,1] \times [0,1]$ telles que :*

1. *Pour tout $t \in [0,1]$ on ait :*
$$\bar{G}_n(t) \overset{n\to+\infty}{\longrightarrow} \bar{G}(t)$$
2. *Pour tout s et $t \in [0,1]$ on ait :*
$$\mu_n(s,t) \overset{n\to+\infty}{\longrightarrow} \mu(s,t)$$

Alors *il existe un processus aléatoire gaussien W à trajectoire sur $[0,1]$, de moyenne nulle et de fonction de covariance μ tel que :*

$$W_n \overset{n\to+\infty, Loi}{\longrightarrow} W.$$

Etape 3 : Preuve du théorème 4.19

Pour démontrer le théorème 4.19 demontrons d'abord la proposition suivante :

Proposition – 4.24 *En reprenant les notations du théorème 4.19 on définit la suite des processus aléatoires* $\tilde{y}_n \equiv \{\tilde{y}_n(t)\}_{t\in[0,1]}$ *par :*

$$\forall n > 1,\ \forall t \in [0,1],\ \ \tilde{y}_n(t) = \sqrt{n-1}\,[\,\tilde{\mathbb{F}}_{n,1}(t) - t\,]$$

où $\forall n > 1$, *la fonction* $\tilde{\mathbb{F}}_{n,1}$ *est définie sur* $[0,1]$ *par :*

$$\tilde{\mathbb{F}}_{n,1}(t) = \frac{1}{n-1}\sum_{i=2}^{n} 1_{\{\tilde{U}_i \leq t\}}.$$

Sous l'hypothèse $H_0^{(exp)}$, *la suite des processus* $\tilde{y}_n$ *converge en loi vers le pont brownien :*

$$\{\tilde{y}_n(t)\}_{t\in[0,1]} \xrightarrow{Loi} \{\mathbb{B}(t)\}_{t\in[0,1]}.$$

Preuve – Pour démontrer la proposition 4.24 on écrit le processus $\tilde{y}_n$ sous la forme suivante :

pour $n > 1$ et pour $t \in [0,1]$:

$$\begin{aligned}\tilde{y}_n(t) &= \sqrt{n-1}\,[\,\frac{1}{n-1}(\sum_{i=2}^{n} 1_{\{\tilde{U}_i\leq t\}}) - t\,] \\ &= \frac{1}{\sqrt{n-1}}\sum_{i=2}^{n}[\,1_{\{\tilde{U}_i\leq t\}} - F_{\tilde{U}_i}(t)\,] + \sqrt{n-1}\,[\,\bar{F}_n(t) - t\,]\end{aligned}$$

on a ainsi :

$$\forall t \in [0,1]\ \Gamma\ \tilde{y}_n(t) = \tilde{W}_n(t) + d_n(t)$$

où :

- $\tilde{W}_n$ est une suite de processus aléatoires définis pour tout réel $t \in [0,1]$ par :

$$\tilde{W}_n(t) = \frac{1}{\sqrt{n-1}}\sum_{i=2}^{n}[\,1_{\{\tilde{U}_i\leq t\}} - F_{\tilde{U}_i}(t)\,]$$

- d_n est une suite de fonctions déterministes définies sur $[0,1]$ par :

$$d_n(t) = \sqrt{n-1}\,[\,\bar{F}_n(t) - t\,]$$

où :

$$\forall t \in [0,1]\ \Gamma\ \bar{F}_n(t) = \frac{1}{n-1}\sum_{i=2}^{n} F_{\tilde{U}_i}(t).$$

Par ailleurs :

- les processus $\tilde{W}_n$ ont la même forme que les processus W_n du théorème 4.23
- la condition 1 du théorème 4.23 est ici vérifiée puisqu'on a pour tout t dans $[0,1]$:

$$\bar{F}_n(t) \overset{n\to+\infty}{\longrightarrow} t,$$

ce résultat est obtenu à partir du corollaire 1 de la proposition 4.22 en utilisant la convergence au sens de Césaro.

- la condition 2 du théorème 4.23 est aussi vérifiée puisqu'on a pour tout s et t dans $[0,1]$:

$$Cov\,[\,\tilde{W}_n(s),\tilde{W}_n(t)\,] = \frac{1}{n-1}\sum_{i=2}^{n}[\,F_{\tilde{U}_i}(min(s,t)) - F_{\tilde{U}_i}(s)F_{\tilde{U}_i}(t)\,]$$

et par conséquent :

$$Cov\,[\,\tilde{W}_n(s),\tilde{W}_n(t)\,] \overset{n\to+\infty}{\longrightarrow} min(s,t) - st.$$

On peut donc utiliser le théorème 4.23 pour conclure la convergence en loi de la suite de processus $\tilde{W}_n$ vers le pont brownien :

$$\{\tilde{W}_n(t)\}_{t\in[0,1]} \overset{Loi}{\longrightarrow} \{\mathbb{B}(t)\}_{t\in[0,1]}.$$

Pour finir la preuve de la proposition 4.24Γon utilise le lemme suivantΓdémontré dans l'annexe A :

Lemme – 4.25 *La suite de fonctions $(d_n)_{n\geq 2}$ définies par :*

$$\forall t\in[0,1]\ ,\ \ d_n(t) = \sqrt{n-1}\ \left[\ \frac{1}{n-1}(\sum_{i=2}^{n} F_{\tilde{U}_i}(t)) - t\right]$$

où pour tout $i \geq 2$:

$$F_{\tilde{U}_i}(t) = 1 - \left[\ 1 - \frac{ln(1-t)}{i-1}\ \right]^{-(i-1)} \quad pour\ t\in[0,1[\ \ et\ \ F_{\tilde{U}_i}(1) = 1.$$

converge simplement vers la fonction nulle :

$$\forall t\in[0,1],\ \ d_n(t) \overset{n\to+\infty}{\longrightarrow} 0.$$

On en déduit la convergence en loi de la suite $\tilde{y}_n$ vers le pont brownien. □

On termine la preuve du théorème 4.19 en utilisant le résultat suivant (cf. par exemple Billingsley [9] page 105) :

$$\sup_{t\in[0,1]} |\mathbb{B}(t)| \sim \mathcal{L}_{KS}.$$

Comme la fonction "*sup*" est continue sur l'espace D des fonctions cad-lag définies sur $[0,1]$ muni de la métrique d de Skorokhod (cf. Durbin [27] page 18)Γ la propriété de Billingsley ([9] page 30) permet de passer des résultats :

$$\begin{cases} \sup_{t\in[0,1]} |\mathbb{B}(t)| \sim \mathcal{L}_{KS} \\ \\ \{\tilde{y}_n(t)\}_{t\in[0,1]} \xrightarrow{Loi} \{\mathbb{B}(t)\}_{t\in[0,1]} \end{cases}$$

au résultat suivant :

$$\tilde{K}_{n,1} = \sup_{t\in[0,1]} |\ \tilde{y}_n(t)\ | \xrightarrow{Loi} \mathcal{L}_{KS},$$

ceci termine la preuve du théorème 4.19.

□

4.4 Conclusions

On peut trouver dans la littérature d'autres outils généraux de validation et de choix de modèles de fiabilité des logiciels.

Citons par exemple :

- le critère de vraisemblance préquentielle proposé par Dawid [24] et utilisé en Fiabilité des Logiciels par Abdel-Ghaly et al [1]Γ
- le critère *AIC* d'Akaike [3] dont l'utilisation en Fiabilité des Logiciels a été proposée par Khoshgoftaar et Woodcock [54]Γ
- le critère du *y-plot* proposé par Keiller et al [52]
- ainsi que le *u-plot* généralisé de Downs et Scott [25].

L'avantage du critère du *u-plot* provient de la possibilité de son utilisation comme un test d'adéquation statistique.

Ce résultat a été justifié expérimentalement pour un certain nombre de modèles : cas i.i.d. de loi exponentielleΓle modèle de *Jelinski-Moranda*Γle *MPD* et le modèle de *Crow*.

L'étude des propriétés théoriques du critère *u-plot* dans le cas i.i.d. exponentiel nous a permis d'introduire un nouveau test d'adéquation à une loi exponentielle de paramètre inconnu.

Il reste donc à étudier les propriétés théoriques (notamment les lois asymptotiques) du critère du *u-plot* sous les hypothèses des modèles usuels en Fiabilité des Logiciels.

Ceci permettrait de donner des tests d'adéquation préquentiels permettant de valider les différents modèles usuels.

Conclusion

Nous avons étudié dans ce travail différents outils statistiques permettant de construire et de valider des modèles d'évaluation de la fiabilité des logiciels.

Les résultats présentés ont des objectifs et des intérêts divers. Certains sont d'ordre méthodologique : ils apportent une contribution à la pratique de la Fiabilité des Logiciels. D'autres ont plutôt un aspect théorique et constituent une contribution à la théorie statistique de l'analyse des durées de vie.

Ce travail apporte à l'ingénieur informaticien des outils lui permettant de mieux exploiter la grande quantité d'informations généralement recueillie en période de tests et de validation des logiciels. La pertinence de ces outils sera confirmée : nous l'espérons : à l'occasion de futures collaborations avec des constructeurs informatiques.

Ce travail comporte aussi quelques contributions à un domaine fécond de la statistique appliquée.

La théorie des modèles linéaires généralisés nous a permis d'abord de trouver de nouvelles propriétés de certains modèles usuels. Elle nous a permis ensuite de présenter deux approches : l'une paramétrique : l'autre non paramétrique : unifiant tous les éléments de la classe des modèles *ND*.

L'approche bayésienne : associée au principe du maximum d'entropie et aux algorithmes de simulation stochastiques : nous a permis de fournir une approche globale de modélisation et d'évaluation de la fiabilité des logiciels.

Le problème de la validation de modèles est sans doute le domaine de recherche le plus ouvert et le plus attrayant en Fiabilité des Logiciels.

Nous avons commencé par y apporter certaines réponses dans des cas particuliers. Plusieurs axes de recherche mériteraient d'être poursuivis.

Nous pensons notamment à la généralisation du test préquentiel à un plus grand nombre de modèles.

Une autre direction intéressante de rechercheΓ introduite par Laprie et Kanoun [58]Γ consiste à étudier les possibilités de combiner les estimations de la fiabilité des logiciels avec celles de la fiabilité des composants matériels pour l'évaluation de la fiabilité (voire la disponibilité) globale des systèmes informatisés.

Il est clair que l'importance de la Fiabilité des LogicielsΓen tant que discipline scientifiqueΓne cessera d'augmenter.

A cela plusieurs raisons :

- la concurrence croissante entre les constructeurs informatiquesΓ
- la prise de conscience de plus en plus grandeΓpar les utilisateursΓde l'aspect fiabilitéΓ
- l'émergence des "contrats fiabilité" obligeant le constructeur à dédommager l'utilisateur en cas de défaillances nombreusesΓ
- la prédominance de plus en plus affirmée des défaillances dues aux fautes de conception par rapport aux défaillances matérielles.

Ces facteurs donneront dans le futur une plus grande importance à l'évaluation de la fiabilité des systèmes améliorables.

Les outils statistiques présentés dans ce travail apportentΓnous l'espéronsΓune modeste contribution à cette discipline.

Annexe A

Lemme – 4.25 *La suite de fonctions $(d_n)_{n\geq 2}$ définies par :*

$$\forall t \in [0,1]\ ,\ \ d_n(t) = \sqrt{n-1}\,[\,\frac{1}{n-1}(\sum_{i=2}^{n} F_{\tilde{U}_i}(t)) - t\,]$$

où pour tout $i \geq 2$:

$$F_{\tilde{U}_i}(t) = 1 - \left[\,1 - \frac{ln(1-t)}{i-1}\,\right]^{-(i-1)} \quad \textit{pour } t \in [0,1[\ \ \textit{et} \ \ F_{\tilde{U}_i}(1) = 1.$$

converge simplement vers la fonction nulle :

$$\forall t \in [0,1],\ \ d_n(t) \overset{n\to+\infty}{\longrightarrow} 0.$$

Preuve – En faisant le changement de variable $u = ln(1-t)$Γet en définissant la suite de fonctions $(c_n)_{n\geq 1}$ définies sur $\mathbb{R}_-$ par :

$$\begin{array}{lll} \forall n \geq 1\ \Gamma\forall u \leq 0 \qquad c_n(u) & = & -d_{n+1}(1-e^u) \\ & = & \dfrac{1}{\sqrt{n}} \displaystyle\sum_{i=1}^{n} \left[(1-\frac{u}{i})^{-i} - e^u\right] \end{array}$$

on démontre le résultat du lemme précédent en démontrant que :

$$\forall u \leq 0\Gamma\ c_n(u) \overset{n\to+\infty}{\longrightarrow} 0.$$

Pour ce faireΓon écrit :

$$\left(1-\frac{u}{i}\right)^{-i} - e^u = \frac{u^2}{i}\,e^u + o(\frac{1}{i})$$

on en déduit que si a est un réel strictement positifΓil existe un entier n_0 tel que pour tout $i \geq n_0$ on a :

$$\left(1-\frac{u}{i}\right)^{-i} - e^u \leq \frac{1}{i}(u^2e^u + a).$$

On peut ainsi écrire :

$$c_n(u) = \left[\frac{1}{\sqrt{n}} \sum_{i=1}^{n_0-1} \left(1-\frac{u}{i}\right)^{-i} - e^u\right] + \left[\frac{1}{\sqrt{n}} \sum_{i=n_0}^{n} \left(1-\frac{u}{i}\right)^{-i} - e^u\right]$$

avec :

$$\frac{1}{\sqrt{n}} \sum_{i=1}^{n_0-1} \left(1 - \frac{u}{i}\right)^{-i} - e^u \overset{n\to+\infty}{\longrightarrow} 0$$

et :

$$\begin{aligned} \frac{1}{\sqrt{n}} \sum_{i=n_0}^{n} \left(1 - \frac{u}{i}\right)^{-i} - e^u &\leq \frac{1}{\sqrt{n}} \left(u^2 e^u + a\right) \sum_{i=1}^{n} \frac{1}{i} \\ &\leq \frac{1}{\sqrt{n}} (u^2 e^u + a) \left(1 + \int_1^n \frac{1}{x}\, dx\right) \\ &\leq \frac{1}{\sqrt{n}} (u^2 e^u + a) \left[1 + ln(n)\right]. \end{aligned}$$

Comme on a :

$$\forall u \leq 0 \Gamma\, \forall i \in \mathbb{N}^* \Gamma\ 0 \leq \left(1 - \frac{u}{i}\right)^{-i} - e^u$$

on conclut enfin que :

$$\forall u \leq 0 \Gamma\, c_n(u) \overset{n\to+\infty}{\longrightarrow} 0$$

d'où la preuve du lemme 4.25. □

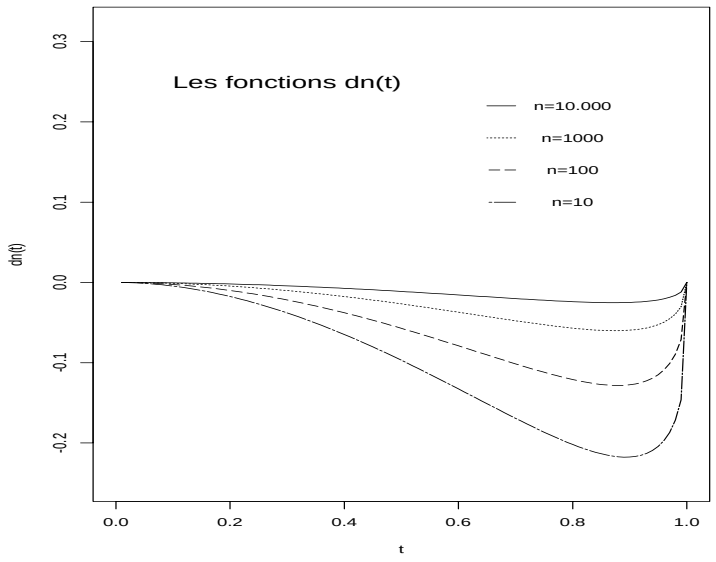

FIG. 4.6: Les fonctions d_n pour $n = 10, 100, 1000$ et $n = 10.000$.

Annexe B

On donne ci-dessous 6 jeux de données représentants des temps inter-défaillances. Les 5 premiers jeux représentent des données réelles collectées sur des logiciels en phase de test (cf. [75] et [36]). Le sixième jeu (*Crow1*) représente des données simulées à partir du modèle de *Crow*.

Les données sont à lire par colonnes.

Cisi1 - n = 169

20	63	16	2	11	97	316	50
32	20	3	1	13	2	76	161
14	1	5	7	9	55	25	76
40	39	1	19	79	825	32	309
40	13	39	10	2	108	39	147
43	10	1	1	9	44	51	176
42	8	1	2	47	8	198	135
22	38	29	2	54	11	291	76
33	4	12	1	49	7	279	150
23	11	28	13	2	82	159	86
6	9	39	66	10	2	150	97
8	149	5	95	9	2	125	144
9	19	19	37	440	2	69	465
5	3	12	17	101	77	356	187
40	18	30	6	24	92	109	163
69	26	1	83	71	98	26	
14	38	15	15	57	119	30	
73	53	8	59	1	133	29	
14	2	4	33	12	154	397	
31	11	17	31	19	38	81	
50	1	1	21	7	163	71	
28	4	12	14	2	119	805	

Musa1 - n = 136

3	120	227	21	529	860	108	22
30	26	65	233	379	983	1	75
113	114	476	134	44	707	3109	482
81	325	58	357	129	33	1247	5509
115	55	457	193	810	868	943	100
9	242	300	236	290	724	700	10
2	68	97	31	300	2323	875	1071
91	422	263	369	529	2930	245	371
112	180	452	748	281	1461	729	790
15	10	255	1	160	843	4897	6150
138	1146	197	231	828	12	447	3321
50	600	193	330	1011	261	386	1045
77	15	6	365	445	1800	446	648
24	36	79	1222	296	865	122	5485
108	4	816	543	1755	1435	990	1160
88	1	1351	10	1064	30	948	1864
670	7	148	16	1783	143	1082	4116

Musa 3 - n = 38

115	50	15	10571
1	71	390	563
82	606	1863	2770
178	1189	1337	652
194	40	4508	5593
136	788	834	11696
1077	222	3400	6724
15	72	6	2546
15	615	4561	
92	589	3186	

Musa 6 - n = 73

3	4	12	23	43	3	5	4
14	1	36	1	1	169	36	437
59	30	38	672	4	29	74	66
32	21	1	189	5	88	40	
8	196	74	83	1	55	2	
52	265	43	520	160	27	86	
2	6	236	8	70	24	221	
25	3	121	1	60	27	6	
2	8	18	41	2	140	891	
3	1	9	70	2	33	23	

Musa 14 c - n = 36

191520	15420	250680	228315	545280	1563300
2078820	60000	2965	51480	256980	513000
514560	140160	196	44820	396780	177660
1140	937620	65173	850080	91260	2469000
3120	72240	2370	361860	1225620	1678260
327480	737700	1581	39300	120	170760

Crow 1 - n = 100

33	17	1049	1137	1369	3061	1118	4406	1216	5153
10	1153	303	1544	191	170	8	3333	292	7198
267	137	571	2045	214	5193	2049	399	1682	4074
75	56	1416	2466	811	1047	1243	1339	339	572
67	2025	740	539	544	919	896	1378	6970	489
580	790	231	480	457	4220	3279	1441	3306	195
151	267	96	332	144	360	3318	1219	513	236
19	421	415	493	1079	5869	3470	1565	1011	3360
89	635	112	130	1251	851	75	2950	3693	2778
2	103	1491	1236	2324	1030	2182	554	933	957

Bibliographie

[1] Abdel-Ghali (A.A.)ΓChen (P.Y.) et Littlewood (B.). – Evaluation of Competing Software Reliability Predictions. *IEEE Transactions on Reliability*Γvol. SE-12Γn° 9Γ1986Γpp. 518–524.

[2] Aivazian (S.). – *Etude Statistique des Dépendances.* – Editions MIR MoscouΓ1970.

[3] Akaike (H.). – A new look at the statistical model identification. *IEEE Transactions on Automatic Control*Γvol. AC-10Γn° 19ΓDec. 1974Γpp. 716–723.

[4] Antoniadis (A.)ΓBerruyer (J.) et Carmona (R.). – *Regression non lineaire et applications.* – EconomicaΓ1992.

[5] Arjas (E.) et Gasbarra (D.). – Nonparametric Bayesian Inference from right censored survival DataΓusing the Gibbs sampler. *Statistica Sinica*Γno4Γ1994Γpp. 505–524.

[6] Ascher (H.). – *Repairable systems reliability.* – New York and Basel : Marcel DekkerΓ incΓ1984Γ*Lecture notes in statistics*Γvolume 7.

[7] Bastani (F.B.) et Ramamoorthy (C.V.). – *Handbook of Statistics,*Γchap. Software reliability. – ElsevierΓLondonΓ1989Γp.r. krishnaiah and c.r. rao édition.

[8] Becker (B.) et Camaranipoulos (L.). – A Bayesian Estimation Method for the Failure Rate of a Possibly Correct Program. *IEEE Trans. Software Engineering*Γ vol. 16Γn° 11Γ1990Γpp. 1307–1310.

[9] Billingsley (P.). – *Convergence of Probability Measures.* – Wiley and SonsΓ Inc.Γ 1968Γ*Wiley series in Probability and Mathematical Statistics.*

[10] Bonneu (M.)ΓDelecroix (M.) et Malin (E.). – *Semiparametric versus Nonparametric Estimation in Single Index Regression Model : a computational Aproach.* – Rapport techniqueΓGremaqΓToulouseΓ1992.

[11] Bunday (B.D.) et Al-Ayoubi (I.D.). – Likelihood and Bayesian Estimation Methods for Poisson Process Models in Software Reliability. *J. Quality and Reliability Management*Γvol. 7Γ1990Γpp. 9–18.

[12] Campodónico (S.) et Singpurwalla (N.D.). – A Bayesian Analysis of the Logarithmic-Poisson Execution Time Model Based on Expert Opinion and failure Data. *IEEE Transactions on Software Engineering*Γvol. 20Γn° 9Γ1994Γpp. 677–683.

[13] Campodónico (S.) et Singpurwalla (N.D.). – Inference and Predictions From Poisson Point Processes Incorporating Expert Knowledge. *JASA*Γvol. 90Γn° 429Γ1995Γpp. 220–226.

[14] Canfield (R.V.). – A Bayesian Approach to Reliability Estimation Using a Loss Function. *IEEE Transactions on Reliability*Γvol. R-19Γn° 2Γ1970Γpp. 13–16.

[15] Chambers (J.M.) et Hastie (T.J.). – *Statistical Models in S.* – Wadsworth & BrooksΓ 1992.

[16] Chen (Y.) et Singpurwalla (N.D.). – A Non-Gaussian Kalman filter model for tracking software reliability. *Statistica sinica*Γvol. 4Γ1994Γpp. 535–548.

[17] Cheung (R.C.). – A user oriented software reliability model. *IEEE Transactions on Software Engineering*Γvol. 2Γn° 6Γ1980Γpp. 118–125.

[18] Cox (D.R.) et Isham (V.). – *Point Processes.* – Chapman and HallΓ1980.

[19] Cox (D.R.) et Miller (H.D.). – *The Theory of Stochastic Processes.* – Chapman and HallΓLondonΓ1977.

[20] Crow (L.H.). – *Reliability and biometry- Statistical analysis of lifelength*Γchap. Reliability analysis for complex repairable systemsΓpp. 379–410. – SIAM PhiladelphiaΓ 1974.

[21] Csenki (A.). – Bayes predictive analysis of a fundamental software reliability model. *IEEE Trans. Reliability*Γvol. R-39 (2)Γ1990Γpp. 177–183.

[22] D'Agostino (R.B.) et Stephens (M.A.). – *Goodness-of-fit Techniques.* – New York and Basel : Marcel DekkerΓ incΓ 1986Γ *Statistics, textbooks and monographs*Γ volume 68.

[23] David (F.N.) et Johnson (N.L.). – The probability Integral Transformation when parameters are estimated from the sample. *Biometrika*Γvol. 35Γ1948Γpp. 182–190.

[24] Dawid (A.P.). – Statistical Theory : The Prequential Approach. *J. R. Statist. Soc. A*Γvol. 147Γ1984Γpp. 278–292.

[25] Downs (T.) et Scott (A.). – Evaluating the performance of software reliability models. *IEEE Transactions on Reliability*Γvol. 41Γn° 4ΓDec 1992Γpp. 518–524.

[26] Duane (J.T.). – Learning curve approach to reliability monitoiring. *IEEE Transactions on Aerospace*Γvol. AS 2Γn° 2Γ1964Γpp. 563–566.

[27] Durbin (J.). – *Distribution theory for tests based on the sample distribution function.* – PhiladelphiaΓSIAM PublicationsΓ1973Γ*Statistics textbooks and monographs.*

[28] El Aroui (M.A.). – Un test préquentiel d'adéquation à la loi exponentielle. *Une note à paraître aux Comptes Rendus de l'Académie des Sciences, Paris*ΓSeptembre 1996.

[29] El Aroui (M.A.) et Lavergne (C.). – Construction and Choice of Models in Software Reliability. *In : Proceedings of the Third ISI International Summer School on Model Choice and Design of Experiments.* – IzmirΓTurkeyΓSeptember 1995.

[30] El Aroui (M.A.) et Lavergne (C.). – Generalized Linear Models in Software Reliability : parametric and and semi-parametric approaches. *IEEE Transactions on Reliability*Γvol. 45Γn° 3ΓSept. 1996.

[31] El Aroui (M.A.) et Soler (J.L.). – A Bayes Nonparametric Framework for Software Reliability Analysis. *A paraître dans IEEE Transactions on Reliability*Γ1996.

[32] Fahrmeir (L.) et Kaufmann (H.). – Consistency and asymptotic normality of the maximum likelihood estimator in generalized linear models. *Annals of Statistics*Γ vol. 13Γn° 1Γ1985Γpp. 342–368.

[33] Fahrmeir (L.) et Tutz (G.). – *Multivariate Statistical Modelling Based on Generalized Linear Models.* – Springer-VerlagΓ1994.

[34] Font (V.). – *Une approche de la fiabilité des logiciels : modèles classiques et modèle linéaire généralisé.* – Thèse de doctoratΓUniversité Paul Sabatier de ToulouseΓ1985.

[35] Forman (E.H.) et Singpurwalla (N.D.). – Optimal time intervals for testing hypothesis on computer software errors. *IEEE Transactions on Reliability*Γvol. R-28Γ 1979Γpp. 250–253.

[36] Gaudoin (O.). – *Outils statistiques pour l'évaluation de la fiabilité des logiciels.* – Thèse de doctoratΓUniversité Joseph Fourier de GrenobleΓ1990.

[37] Gaudoin (O.). – *Tests d'adéquation aux modèles NHPP.* – Rapport technique n° à paraîtreΓGrenobleΓLMC-IMAGΓ1996.

[38] Gaudoin (O.)Γ Lavergne (C.) et Soler (J.L.). – A generalized geometric deeutrophication software reliability model. *IEEE Trans. Reliability*Γvol. 43(4)ΓDec. 1994Γpp. 536–541.

[39] Gaudoin (O.) et Soler (J.L.). – Modèles pour l'étude de la Fiabilité des systèmes présentant des fautes de conception. Application à l'évaluation de la Fiabilité des Logiciels. *Revue de Statistique Appliquée*Γvol. XXXXΓn° 2Γ1992Γpp. 91–98.

[40] Gaudoin (O.) et Soler (J.L.). – Statistical analysis of the geometric deeutrophication software reliability model. *IEEE Transactions on Reliability*Γvol. R-41Γn° 4ΓDec 1992Γpp. 518–524.

[41] Gilks (W.R.) et Wild (P.). – Adaptative rejection sampling for Gibbs sampling. *Applied Statistics*Γvol. 41Γ1992Γpp. 337–348.

[42] Goel (A.L.) et Okumoto (K.). – Time dependent error detection rate model for software reliability and other performance measures. *IEEE Transactions on Reliability*Γ vol. R-28Γn° 3Γ1979Γpp. 206–211.

[43] Green (P.J.) et Silverman (B.W.). – *Nonparametric Regression and Generalized Linear Models.* – Chapman and HallΓ1994Γ*Monographs on Statistics and Applied Probability.*

[44] Hastie (T.J.) et Tibshirani (R.J.). – *Generalized Additive Models.* – Chapman and HallΓ1990Γ*Monographs on Statistics and Applied Probability.*

[45] Iannino (A.)ΓMusa (J.D.)ΓOkumoto (K.) et Littlewood (B.). – Criteria for software model comparisons. *Transactions on Software Engeneering*Γvol. SE-10Γ1984Γpp. 687–691.

[46] Jazwinski (A.H.). – *Stochastic Processes and Filtering Theory.* – Academic PressΓ 1970.

[47] Jelinski (Z.) et Moranda (P.B.). – *Statistical computer performance evaluation*Γ chap. Software reliability researchΓpp. 465–497. – W. FreibergerΓAcademic PressΓ New-YorkΓ1972.

[48] Jewell (W.S.). – Bayesian extensions to a basic model of software reliability. *Software Engineering Journal*Γvol. SE-11Γn° 12ΓDec. 1985Γpp. 1465–1471.

[49] Kaâniche (M.). – *Modèle hyperexponentiel en temps continu et en temps discret pour l'évaluation de la croissance de la sûreté de fonctionnement.* – Thèse de doctoratΓ Institut National Polytechnique de ToulouseΓ1992.

[50] Kanoun (K.). – *Croissance de la Sûreté de Fonctionnement des Logiciels, Caractérisation-Modélisation-Evaluation.* – Thèse d'EtatΓInstitut National Polytechnique de ToulouseΓ1989.

[51] Kapur (J.N.). – *Maximum-Entropy Models in Science and Engineering.* – John Wiley & SonsΓ1989.

[52] Keiller (P.A.)ΓLittlewood (B.)ΓMiller (D.R) et Sofer (A.). – Comparison of software reliability predictions. *IEEE FCTS*Γvol. 13Γ1983Γpp. 128–134.

[53] Khoshgoftaar (T.M.) et Munson (J.C.). – Predicting software development errors using software complexity metrics. *IEEE J. Selected Areas in Commun*Γvol. SAC-8Γ 1990Γpp. 252–261.

[54] Khoshgoftaar (T.M.) et Woodcock (T.G.). – Software reliability model selection : a case study. *In : Proc. Int. Symp. on Software reliability Engeneering, ISSRE*Γpp. 183–191. – AustinΓTexasΓMay 1991.

[55] Kyparisis (J.) et Singpurwalla (N.D.). – Bayesian Inference for the Weibull Process with Applications to Assessing Software Reliability Growth and Predicting Software Failures. *In : Computer Science and Statistics 16 th Symp. Interface*Γpp. 57–64. – AtlantaΓGAΓ1985.

[56] Langberg (N.) et Singpurwalla (N.D.). – A unification of some software reliability models. *SIAM J. Scientific and Statistical Computation*Γvol. 6Γ1985Γpp. 781–790.

[57] Laprie (J.C.)ΓCourtois (B.)ΓGaudel (M.C.) et Powel (D.). – *Sûreté de fonctionnement des systèmes informatiques.* – Dunod informatiqueΓ1989.

[58] Laprie (J.C) et Kanoun (K.). – X-Ware Reliability and Availability Modeling. *IEEE Transactions on Software Engineering*Γvol. 18Γn° 2Γ1992Γpp. 130–147.

[59] Laprie (J.C)ΓKanoun (K.)ΓBeounes (C.) et Kaâniche (M.). – The KAT (knwoledge-action-transformation) approach to the modelling and evaluation of reliability and availability growth. *IEEE Transactions on Software Engineering*Γvol. 17Γ1991Γpp. 370–382.

[60] Ledoux (J.). – *Principaux modèles d'évaluation de la fiabilité du logiciel et techniques de validation de systèmes de prédiction : étude bibliographique.* – Rapport technique n° 667ΓRennesΓIRISAΓ1992.

[61] Ledoux (J.). – *Modèles markoviens : sur la caractérisation de l'agrégation faible et sur les modèles structurels pour l'évaluation de la sûreté de fonctionnement du logiciel.* – Thèse de doctoratΓUniversité de Rennes IΓ1993.

[62] Lilliefors (H.W.). – On the Kolmogorov-Smirnov test for normality with mean and variance unknown. *JASA*Γvol. 62Γ1967Γpp. 399–402.

[63] Lilliefors (H.W.). – On the Kolmogorov-Smirnov test for the exponential distribution with mean unknown. *JASA*Γvol. 64Γ1969Γpp. 387–389.

[64] Lindley (D.V.). – Approximate Bayesian Methods. *Trabajos Estadistica*Γvol. 31Γ 1980Γpp. 223–237.

[65] Littlewood (B.). – Software reliability model for modular program structure. *IEEE Transactions on Reliability*Γvol. R-28(3)Γ1979Γpp. 241–246.

[66] Littlewood (B.) et Sofer (A.). – A Bayesian modification to the Jelinski-Moranda software reliability growth model. *Software Engineering Journal*ΓMarch 1987Γpp. 30–41.

[67] Littlewood (B.) et Verrall (J.L.). – A bayesian reliability growth model for computer software. *Applied statistics*Γvol. 22Γ1973Γpp. 332–346.

[68] Littlewood (B.) et Verrall (J.L.). – On the likelihood function of a debugging model for computer software reliabiity. *IEEE Transactions on Reliability*Γvol. R-30ΓJune 1981Γpp. 145–148.

[69] Lyu (M.R.) et al. – *Handbook of Software Reliability Engineering.* – IEEE Computer Society Press and McGraw-Hill Book CompanyΓ1996.

[70] Mazzuchi (T.A.) et Soyer (R.). – A Bayes empirical bayes Model for software Reliability. *IEEE Trans. Reliability*Γvol. 37Γn° 2Γ1988Γpp. 248–254.

[71] McCullagh (P.) et Nelder (J.A.). – *Generalized Linear Models.* – Chapman and HallΓ1989Γ*Monographs on Statistics and Applied Probability.*

[72] Meinhold (R.J.) et Singpurwalla (N.D.). – Bayesian analysis of a commonly used model for describing software failures. *The Statistician*Γvol. 32Γ1983Γpp. 168–173.

[73] Miller (D.R.). – Exponential order statistic models of software reliability growth. *IEEE Transactions on Software Engineering*Γvol. SE 12(1)Γ1986Γpp. 12–24.

[74] Moranda (P.B.). – Event altered rate models for general reliability analysis. *IEEE Trans. Reliability*Γvol. R-28Γn° 5ΓDec 1979Γpp. 376–381.

[75] Musa (J.D.). – *Software reliability data.* – Rapport techniqueΓRome Air Development CenterΓRomeΓNew-YorkΓ1979.

[76] Musa (J.D.). – Operational Profiles in Software Reliability Engineering. *IEEE Software*ΓMarch 1993Γpp. 14–32.

[77] Musa (J.D.) et Okumoto (K.). – A Logarithmic Poisson Execution Time Model for Software Reliability Measurement. *In : Proceedings of the 7th International Conference on Software Engineering*Γpp. 230–237. – OrlandoΓ1984.

[78] Nelder (J.A.) et Wedderburn (R.W.M.). – Generalized Linear Models. *J. Roy. Statis. Soc. A*Γvol. 135Γ1972Γpp. 370–384.

[79] O'Reilly (F.) et Quesenberry (C.P.). – The conditional probability integral transformation and applications to obtain composite chi-square goodness of fit tests. *Annals of Statistics*Γvol. 1Γ1973Γpp. 74–83.

[80] O'Sullivan (F.)ΓYandell (B.S.) et Raynor (W.J.). – Automatic Smoothing of Regression Functions in Generalized Linear Models. *JASA*Γvol. 86Γn° 393Γ1986Γpp. 96–103.

[81] Raftery (A.E.). – Analysis of a simple debugging model. *Applied Statistics*Γvol. 37Γ 1988Γpp. 12–22.

[82] Rao (K.C.). – The KolmogorovΓ Cramer-von MisesΓ chi-square statistics for goodness-of-fit tests in the parametric case (abstract). *Bull. Inst. Math. Statist.*Γ vol. 1Γn° 87Γ1972Γpp. 133–136.

[83] Robert (C.). – *L'analyse statistique bayésienne.* – EconomicaΓ1992.

[84] Rosenblatt (M.). – Remarks on a multivariate transformation. *Annals of Mathematical Statistics*Γvol. 23Γ1952Γpp. 470–472.

[85] Rubinstein (R.Y.). – *Simulation and the Monte-Carlo method.* – John Wiley and sonsΓ1981.

[86] Saporta (G.). – *Probabilités, Analyse des Données et Statistique.* – Editions TechnipΓ 1990.

[87] Scallan (A.)ΓGilchrist (R.) et Green (M.). – Fitting parametric link functions in generalized linear models. *Comp. Statist. and Data Anal.*Γvol. 2Γ1984Γpp. 37–49.

[88] Shorack (G.R.). – The Weighted empirical process of row independent random variables with arbitrary distribution functions. *Statistica Neerlandica*Γvol. 33Γn° 4Γ1979Γpp. 169–189.

[89] Singpurwalla (N.D.) et Soyer (R.). – Non-homogeneous Autoregressive Processes for Tracking (Software) Reliability GrowthΓand their Bayesian Analysis. *J. R. Statist. Soc. B*Γvol. 54Γn° 1Γ1992Γpp. 145–156.

[90] Singpurwalla (N.D.) et Wilson (S.P.). – Software Reliability Modelings. *International Statistical Review*Γvol. 62Γ1994Γpp. 289–317.

[91] Smith (A.F.M). – Bayesian Computational methods. *Philos. Trans. Roy. Soc. Ser. A*Γvol. 337Γ1991Γpp. 369–386.

[92] Smith (A.F.M) et Roberts (G.O.). – Bayesian Computations via the Gibbs sampler and related Markov-Chain Monte-Carlo methods. *J. Roy. Statist. Soc. Ser. B*Γ vol. 55Γ1993Γpp. 3–23.

[93] Snyder (D.L.). – *Random point processes.* – WileyΓNew-YorkΓ1975.

[94] Soler (J.L.). – Modélisation des processus de risqueΓde défaillance et de correction. Application à la fiabilité des logiciels. *In : Proc. 6th Int'l. Conf. Reliability and Maintainability.* – StrasbourgΓFranceΓOct. 1988.

[95] Soler (J.L.). – *Fiabilité des systèmes : cours de DEA.* – Université Joseph Fourier de GrenobleΓ1995.

[96] Soler (J.L.). – Croissance de fiabilité des versions d'un logiciel. *Revue de Statistique Appliquée*Γvol. XLIVΓ1996Γpp. 5–20.

[97] Stephens (M.A.). – On the Half-sample method for Goodness-of-fit. *J. R. Statist. Soc. B*Γvol. 40Γn° 1Γ1978Γpp. 64–70.

[98] Trachtenberg (M.). – A general theory of software reliability modelling. *IEEE Transactions on Reliability*Γvol. R-39Γ1990Γpp. 536–541.

[99] Wedderburn (R.W.M.). – On the existence and uniqueness of maximum likelihood estimates fo certain Generalized Linear Models. *Biometrika*Γvol. 63Γ1976Γpp. 27–32.

[100] Wright (D.E.) et Hazelhurst (C.E.). – Estimation and Prediction for a simple software reliability model. *The Statistician*Γvol. 37Γ1988Γpp. 319–325.

[101] Xiang (D.) et Wahba (G.). – *Testing the generalized Linear Models Null Hypothesis versus Smooth Alternatives.* – Rapport technique n° 953ΓDepartment of StatisticsΓ University of Winsconsin MadisonΓOctober 1995.

[102] Xie (M.). – *Software Reliability Modelling.* – World ScientificΓ1991.

[103] Xie (M.). – Software Reliability Models : a selected annotated bibliography. *Software Testing, Verification and Reliability*Γvol. 3Γ1993Γpp. 3–28.

[104] Yamada (S.)Γ Ohba (M.) et Osaki (S.). – S-shaped reliability growth modeling for software error detection. *IEEE Transactions on Reliability*Γ vol. R 35Γ n° 5Γ 1983Γ pp. 475–478.