UNIVERSITE DE YAOUNDE I

****

ECOLE NATIONALE SUPERIEURE POLYTECHNIQUE

****

DEPARTEMENT DE GENIE INFORMATIQUE

UNIVERSITY OF YAOUNDE I

****

NATIONAL ADVANCED SCHOOL OF ENGINEERING

****

DEPARTMENT OF COMPUTER ENGINEERING

Classification non supervisée et suivi des processus de dynamique forestière

Mémoire de fin d'étude/Master of Engineering

Présenté et soutenu par

Bertrand TEGUIA TABUGUIA

En vue de l'obtention du

Diplôme d'Ingénieur de Conception de Génie Informatique

Encadreur académique :
Pr. Thomas Bouetou Bouetou
Professeur UYI, Chef de département du Génie Informatique

Encadreur professionnel :
Dr. Vivien ROSSI
Modélisateur statisticien HDR, cadre scientifique au Cirad.

Devant le jury constitué de :

Président : Crepin KOFANE, Pr
Rapporteur : Thomas Bouetou, Pr
Examinateur : Jules Tewa, MC
Invité : Vivien Rossi, HDR.

Année académique 2015-2016

Soutenu le 8 Juillet 2016

Classification Non Supervisée et suivi des processus de dynamique forestière : Cas du projet de recherche DynAfFor

« Les mathématiques ont des inventions très subtiles et qui peuvent beaucoup servir, tant à contenter les curieux qu'à faciliter tous les arts et à diminuer le travail des hommes »

RÉNÉ DESCARTES.

DÉDICACES

A mes frères et sœurs TABUGUIA

REMERCIEMENTS

Il y a un certain nombre de personnes que j'aimerais remercier : Je remercie très respectueusement le jury, je suis très honorer de voir mon travail évalué par de grands esprits scientifiques :

- Merci au **Professeur KOFANE TIMOLEON Crepin**, le président du jury,
- Merci au **Professeur BOUETOU BOUETOU Thomas**, le rapporteur, qui est aussi Enseignant d'Algèbre générale, également merci pour avoir accordé de son temps pour les précieux conseils dans mon travail et l'élaboration du mémoire, c'est un plaisir d'être encadrer par un amoureux des mathématiques quand on les aime.
- Merci au **Professeur TEWA Jean Jules**, l'examinateur, c'est un privilège de voir mon travail examiné par l'enseignant qui m'a donné mon plus beau cours de calcul des probabilités.
- Merci au **Docteur ROSSI Vivien**, l'invité, qui est aussi mon encadrant professionnel, également merci pour ce stage qu'il m'a accordé, pour ses bons conseils en modélisation statistique et pour ses nombreuses explications sur la dynamique des forêts qui furent très nécessaires pour l'aboutissement de mon travail.

Je remercie mon collègue de bureau et ancien camarade de classe **TONYE LISSOUK Éric**, qui travaillait aussi sur le projet DynaFfor, merci pour tous les moments de détentes et de réflexions passés ensemble.

Merci au corps enseignant et administratif de l'École Nationale Supérieure Polytechnique (ENSP) de Yaoundé, plus spécialement les enseignants chargés de cours et de travaux dirigés du Département de Mathématiques et Science Physique (MSP), en particulier **Dr. TAKOU Étienne** pour sa grande rigueur mathématique, **Dr. NDONG NGUEMA Eugène** pour son savoir faire mathématique, **Dr. ELOUNDOU Pascal** pour sa précision, **Dr. TETSAJIO** un génie du raisonnement en analyse, **Pr. TEWA Jules** pour sa logique du calcul des probabilités, **Pr. BOUETOU Thomas** pour le raisonnement mathématique et logique ; et du Département du Génie Informatique(GI), en particulier **Dr. KOUAMOU Georges** génie du logiciel, pour la capacité à formuler un problème dans le but de le modéliser, **Dr. BATCHAKUI Bernabé** pour son dévouement aux étudiants, se battant toujours pour qu'on soit les meilleurs, **Dr. MOUKOUOP Ibrahim** pour ses conseils en science, **Dr. NJAMPON** pour la théorie de l'estimation, **Dr. TOUSSILE Wilson** pour le cours d'analyse des données, dispensé dans une rare beauté, **Pr. NJIFENJOU Abdou** pour ses raisonnements pertinents vus en analyse numérique, **Dr. TCHANA Marie** pour sa douceur et sa rigueur dans l'enseignement, une maman scientifique, **M. NGANKAM Hubert** surdoué de l'informatique et dévoué à l'enseignement, **M. MBOUH Pride** pour ses encouragement à la connaissance, **Mme. TCHASSEUP**, merci à vous tous ! Et un grand merci au

Pr. NGABIRENG Marie pour la discipline de l'Ecole Nationale Supérieure Polytechnique de Yaoundé qui pousse les étudiants aux travail permanent.

Je dis également merci à toute la promotion 2016, en particulier à ceux du Génie Informatique, les années passées ensemble on fait de nous une famille, elles nous ont unis dans les partages de connaissances et d'idées, les moments de joie et de rigolade, les moments de tristesse, les débats, je pense particulièrement à mes camarades de pensées : **NGNAWE Jonas** camarade de ban, remplis de culture scientifique, avec qui j'ai passé beaucoup de temps et qui plusieurs fois m'a donné des coup de main sur tous les plans, merci mon frère, **DOKMEGANG Joël** riche en raisonnement mathématique et en culture informatique, très gentil avec moi, merci mon frère, je pense aussi à **WAPET Lavoisier** l'homme qui retient vite le nécessaire à chaque fois, à l'imminent délégué de classe **ATIBITA Jonas**, toujours à veiller au bien être de ses camarades, merci mon chef, merci à ma tendre amie **DJIEUFACK Larisse**, à **FODOUP Christian**, à **NDONNA Yacynth** mon compagnon très passionné de l'informatique, à tous merci. Merci aussi aux promotions adjacentes à la notre : la promotion de nos parrains, les ingénieurs de la promotion 2015, qui nous ont aidés avec beaucoup de conseil, et nos filleuls de la promotion 2017 qui nous ont souvent beaucoup encouragés dans nos travaux.

Merci mes frères des quartiers Manguier, de Tamtam, de Simbock et même de la ville de Douala qui m'ont toujours témoigné de l'amour par leurs présences et leurs encouragement. Merci à mes amis de l'amical du lycée de Mballa II, "AMISCI", avec qui tous les moments passés ensemble ont toujours été des encouragements pour la persévérance dans le travail. Merci mes chers amis.

Enfin j'exprime ma vive gratitude à tous, ma grande sœur **MALIEDJE Arielle** pour tous les efforts qu'elles s'est donnée pour que j'arrive jusque là, à mon grand frère **DJOKO Fabrice** qui a toujours su veillé à ce que je ne manque de rien, mes trois petits frères : **GUIATCHUENG Belissa**, **TABUGUIA Franck** et **YOUGO Reine** pour leurs présences fraternelles. Merci à la famille **FOUDJIN** qui s'est toujours préoccupé de moi, à la famille **FOGUAIN** pour leur amour à mon égard que je ne peux complètement exprimer, à la famille **GOMSI**, à la famille **TAMO**, à toute la grande famille du quartier de l'école de police pour leur grand soutien, à tous mes oncles, mes tantes, mes cousins, mes cousines, etc. Merci à tous. Merci à ma grand-mère **DJUMBOUN Bernadette** qui a toujours pensé à moi, me redonnant du courage par ses appels, et sans oublié mes mamans chérie : **maman MAMGUEM Suzanne** et **maman NOUBISSI Augustine**, des mères comme on en demande.

A tous, Merci !

AVANT PROPOS

Ce travail représente le résultat de cinq mois de stage académique, stage ingénieur effectué dans un bureau du département du Génie Informatique de l'École Nationale Supérieure Polytechnique pour le compte du CIRAD (Centre de coopération International en Recherche en Agronomie pour le Développement). Le CIRAD est un établissement public à caractère industriel et commercial (EPIC) français créé en 1984 et spécialisé dans la recherche agronomique appliquée aux régions chaudes. Dans le cadre du projet DynAfFor (Dynamique des Forêts d'Afrique central), le CIRAD est l'un des partenaires de recherche et d'enseignement. Ce projet vise à quantifier les effet sur la dynamique forestière et sur les processus qui la pilotent, C'est un projet de plus de six millions d'euros, financé par le Fonds Français pour l'Environnement Mondial (FFEM) et l'Agence Française de Développement (AFD) à hauteur d'environ 2.6 millions d'euros. Il se déroule sur une première phase de 5 ans depuis 2012, sous l'égide de la COMIFAC qui regroupe cinq Pays d'Afrique centrale : Cameroun, Congo, République Centrafricaine, République Démocratique du Congo [1]. C'est par la collaboration entre le CIRAD et UYI (l'Université de Yaoundé I) que nous sommes parvenu à travailler sur ce projet.

Notre travail s'est focaliser sur le moteur du logiciel DafSim construit dans le cadre du projet DynAfFor. Ce logiciel doit produire des simulations du comportement de la dynamique des forêts afin d'aider les exploitants forestiers sur les règles d'aménagement à prendre pour leur gestion durable en Afrique centrale.

RÉSUMÉ

Tout être vivant nait, grandi et meurt. Cela est le cœur même du comportement instable de l'évolution de son peuplement, on parle de dynamique. Nous nous intéressons à l'étude quantitative de la dynamique forestière en vue d'une exploitation parcimonieuse du bois d'œuvre de la forêt du bassin du Congo. Le but poursuivit est donc la modélisation statistique des trois processus de dynamique forestière : la croissance, la mortalité et la régénération. Ceci pour le développement d'un algorithme qui permet d'inférer sur l'évolution des arbres à partir de données collectés dans leurs environnements. Mais chaque arbre appartient à une espèce, et des espèces il y en a plusieurs, certaines ayant des comportements similaires. D'où le second objectif de clustering ou classification non supervisée consistant à regrouper les espèces selon les trois processus. En conséquence, ce mémoire étudie la classification non supervisée et sélectionne une méthode de classification qu'il couple à des modèles de régression issues d'une analyse sur les modèles des différents processus en forêt. C'est alors qu'il aboutit à une modélisation et un développement vérifiable par le biais d'une simulation cohérente de données.

Mots clés : dynamique forestière ; processus de dynamique forestière ; classification non supervisée ; régression ; simulation.

ABSTRACT

Every living being is born, grows up, and dies. This is the heart of the unstable behavior of the evolution of its population, it is called dynamic. We are interested in the quantitative study of the forest dynamic for parsimonious exploitation of timber from the Congo Basin forest. The aim pursued is thus the statistical modeling of the three forest dynamic processes : growth, mortality, and regeneration. This for the development of an algorithm that infers the evolution of trees from data collected in their environments. But each tree belongs to a species, and there are several species, some with similar behavior. Hence the second objective of clustering or unsupervised classification is the grouping of species according to the three processes. Accordingly, this essay studies clustering and selects a classification method coupled with regression models derived from an analysis of models of various forest processes. Then it leads to modeling and development verifiable through a coherent simulation of data.

Keywords : forest dynamics ; forest dynamics processes ; clustering ; regression ; simulation.

Principales Notations

- X : l'ensemble des variables explicatives (partie déterministe).
- Y : la variable aléatoire à expliquer.
- Z : variable désignant les données non observées.
- L : comme indice, notation du nombre de variables explicatives.
- s : indice d'espèce d'arbre.
- S : nombre d'espèce.
- X_s : tous les données de variables explicatives correspondant à l'espèce s
- k : indice de groupe.
- π_k : probabilité d'appartenance d'une espèce au groupe k.
- K : nombre de groupe.
- $x_{s,i} = (1, x_{s,i,1}, \cdots, x_{s,i,L})$: données de variables explicatives de l'espèce s à la ligne i du tableau de données (selon son ordre).
- $x_{s,j} = (1, x_{s,j,1}, \cdots, x_{s,j,L})$: j^{e} données de variables explicatives de l'espèce s.
- a : un arbre.
- $\mathcal{P}(\lambda)$: loi de Poisson de paramètre λ.
- $\mathcal{B}(n, p)$: loi Binomiale de paramètre n (nombre d'épreuve) et p (probabilité du succès).
- $\mathcal{M}(1, \pi_1, \cdots, \pi_K)$: loi multinomiale à une réalisation de proportions $\pi_1, \cdots, \pi_K$, $\sum_{k=1}^{K} \pi_k = 1$.
- $\mathbb{E}$: symbole de calcul de l'espérance mathématique.
- $\mathbb{P}$: symbole de calcul de la probabilité d'un évènement.
- ln : fonction logarithme népérienne.
- exp : fonction exponentielle.
- P : notation de la vraisemblance.

- L : Comme fonction, est la notation de la log-vraisemblance : logarithme de la vraisemblance.
- EM : Expectation-Maximization : algorithme de classification non supervisée.
- $\theta_k = (\theta_{k,0}, \cdots, \theta_{k,L})$: paramètre du groupe k.
- θ paramètre ou ensemble de paramètres à estimer.
- $\theta_{(m)}$: estimation de θ à l'itération m de l'algorithme EM.
- $\binom{n}{p}$: combinaison de p dans n : nombre de sous ensembles d'un ensemble à n éléments qui contiennent p éléments.
- $\dfrac{\partial f}{\partial x}$: dérivée partielle de la fonction f par rapport à la variable x.
- $\dfrac{\partial^2 f}{\partial x^2} = \dfrac{\partial f}{\partial x}\left(\dfrac{\partial f}{\partial x}\right)$: dérivée partielle seconde de la fonction f par rapport à la variable x (dérivée partielle de la dérivée partielle de la fonction f par rapport à la variable x, par rapport à la variable x).

Table des matières

Table des figures

Liste des tableaux

INTRODUCTION GÉNÉRALE

Contexte

La surface forestière mondiale est constituée à 50 % de forêt tropicale [11], cette dernière regroupe en son sein de grande richesse végétale et animale, et joue un grand rôle sur l'environnement. En effet, les forêts tropicales possèdent un stock de carbone d'une importance telle que la déforestation et l'utilisation irrationnelle de ses terres produirait des émissions de gaz à effet de serre non négligeable sur le plan mondiale. De plus, la régulation du climat, l'apport des matières premières, la préservation des fonctions du sol, sont d'autres secours qu'offrent ces forêts.
L'homme a donc besoin de comprendre, d'expliquer et de prédire le comportement des forêts tropicales afin de préserver la survie de cette abondance naturelle indispensable. Or la population forestière en milieu tropical est hétérogène, c'est-à-dire que le système d'évolution de la forêt est dynamique, il faut donc se servir de modèles de dynamique pour l'appréhension d'un tel peuplement. C'est ainsi qu'interviennent les modèles statistiques de dynamique des populations qui proposent des formalismes consistants et adaptables aux suivis des forêts. Nous nous focalisons alors sur l'individu « arbre » par l'étude des trois processus qui décrive sa dynamique, savoir : la régénération, la croissance et la mortalité.

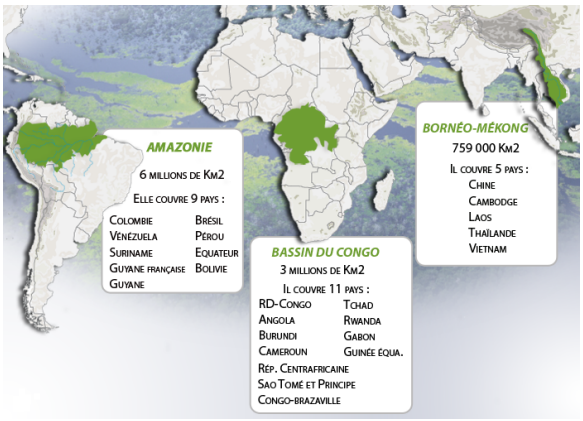

Figure 1 – Forêts tropicales. [2]

Problématique

Après la forêt d'Amérique du Sud dans le bassin de l'Amazon, la forêt tropicale humide (caractérisée par la plus grande richesse spécifique) la plus importante est celle du bassin du Congo en Afrique Centrale. L'inquiétude face aux conséquences de l'exploitation du bois d'œuvre de cette forêt a conduit des pays d'Afrique Centrale à une redéfinition des conventions forestières en vue de trouver des moyens de produire durablement du bois d'œuvre. Cela demande la quantification des processus qui pilotent la dynamique forestière : la croissance, la mortalité et le recrutement. C'est d'où naîtra le projet DynAfFor : Dynamique des forêts d'Afrique Centrale, où se développe le logiciel DafSim dont le moteur développé à l'usage de flexmix [1] a la fonction de construire les groupes de régression des paramètres des lois de probabilités suivis par chaque processus. Mais ce dernier présente des problèmes de performances, et aucune considération particulière lorsque les données sont collectées à des intervalles de temps irréguliers.

Motivations et Objectifs

La grande richesse spécifique des forêts tropicales disqualifie l'approche de modélisation par espèce qui demanderait un nombre d'observations irréalisables, et donc un modèle illusoire sur les données collectées. L'approche consiste donc à regrouper les espèces au comportement similaire et à calibrer la modélisation vers ces groupes (inconnus). Ce procédé de répartition d'individus dans les groupes que l'on ignore à base d'observations sur des variables pertinentes les décrivant constitue ce qu'on appelle en statistique : la classification non supervisée. Il s'agit ainsi pour nous d'en construire un qui nous permet de gagner en efficacité par rapport à flexmix qui est une bibliothèque de R dans un cadre plus générale.
Nous nous proposons donc de modéliser à base de l'algorithme EM (Expectation- Maximization), un algorithme de classification non supervisée approprié à la dynamique des forêts tropicales, et de le développer en langage R. Nous donnons aussi une version générale du modèle qui prend en compte l'irrégularité des intervalles de temps entre les inventaires de données forestiers avec l'utilisation d'une méthode d'interpolation pour la complétion des données manquantes.

Plan du mémoire

Notre travail comporte trois grandes parties organisées comme suit :

1. Une revue de littérature pour la définition des concepts clés tels que les processus de dynamique forestière, la classification et certains modèles de régression tout en présentant l'existant en ce qui concerne la modélisation de la solution.

2. La mise en oeuvre, où nous présentons notre modélisation de la solution avec une claire présentation théorique du principe de classification (non supervisée) utilisé. Cette partie

1. Bibliothèque du langage R permettant de faire des mélanges de modèles GLM de manière générale par l'algorithme EM

s'achève avec la production d'équations qui guideront le développement des algorithmes qui correspondent à notre contexte.

3. Nous terminons avec la partie Application consacrée à la présentation des résultats obtenus par l'implémentation de notre modélisation. On y présente ceux ci sur les données issues de simulation et on les compare avec ceux obtenus par l'utilisation du package [2] flexmix.

2. bibliothèque

Première partie

Revue de littérature

Introduction

En mathématiques appliquées et en informatique, la modélisation, c'est le cœur, et la finalité, c'est l'explication du réel dans la supposition de l'existence d'un modèle idéale auquel on veut accéder. Pour suivre l'évolution des forêts tropicales par le biais de modélisation des processus qui dirigent sa dynamique, une bonne compréhension de ceux-ci est de grande importance. De même, la connaissance des outils mathématiques et statistiques en particulier intégrant les modèles, requière une certaine assimilation permettant de mieux cerner le sens donné par ces théories à la réalité des forêts.
Cette partie a donc pour but d'apporter toute clairvoyance nécessaire à notre travail.

Chapitre 1

Processus de dynamique forestière

L'hétérogénéité forestière est à son niveau maximal lorsqu'on considère les arbres individuellement. Comme tout être vivant, un arbre nait, grandi et meurt. La prise en compte de l'apparition stochastique de ces phénomènes dans le temps fait qu'elles sont considérées comme des processus. Cependant, le suivi individuel des arbres est différent en ce sens que leurs positions déterminent leurs voisinages et leur statut compétitif dans la résistance au vent et l'absorption de la lumière, de l'eau et des nutriments du sol[3] [12]. Eu égard à cette considération des arbres, dans ce chapitre nous définissons les processus de dynamique forestière en présentant des approches de leurs modélisations qui permettent d'inférer sur l'évolution d'une forêt face à une perturbation naturelle ou humaine (exploitation) sur des observations temporelles et spatiales biens définies.

Ce chapitre est essentiellement inspiré du document [13].

FIGURE 1.1 – Compétition des arbres. [3]

1.1 Processus de croissance

1.1.1 Définition :

D'une manière générale, la croissance d'un individu peut se définir comme la variation dans le temps des variables qui permettent sa description. Pour un arbre, ces variables doivent retenir :

- l'absorption de l'eau
- l'absorption d'élément minéraux
- la nutrition azotée
- la photosynthèse

1.1.2 Modélisations :

La modélisation des processus de croissance consiste formellement à exprimer la variation temporelle des grandeurs descriptives de l'arbre en fonction de variables que l'on peut classer en trois catégories :

- Les grandeurs descriptives de l'arbre elles-mêmes(X), typiquement son diamètre à $1,30m$ (D), sa hauteur, etc.
- Les variables de compétition(C), appelées aussi indice de compétition
- Les variables de sites décrivant l'effet du milieu(S)

Cependant, la plupart des auteurs s'accorde à reconnaitre les variables de dimension comme les meilleurs prédicteurs de la croissance individuelle, la dimension traduisant à la fois les performances passées de l'individu ainsi que son statut social dans le peuplement. Les autres variables étant quasiment insignifiantes.

Le plus souvent, l'accroissement de X à l'instant t est prédit à partir de la valeur de X à cet instant :

$$\frac{dX}{dt} = f(X, C, S) \tag{1.1}$$

où f est appelé fonction de croissance. Et pour la cohérence du système, on ajoute des équations dites d'état qui relient les variables descriptives entre elles :

$$X_i = g(X_j)\ ,\ i \neq j \tag{1.2}$$

C'est donc la connaissance des formes des fonctions f de 1.1 et g de 1.2 qui fait le cœur de la modélisation de la croissance.
Mais, la modélisation pouvant se faire séparément sur les variables descriptives, nous présentons par la suite la forme des fonctions de croissance.

Forme de la fonction de croissance :
On distingue trois catégories de modèles pour la fonction de croissance :

1. Modèles linéaires polynomiaux :

$$f(X, C, S) = P(X, C, S, \theta)$$

où P désigne un polynôme de coefficients θ estimés par régression linéaire.

2. Modèles de type sigmoïde :
la forme générale de la croissance s'inspire d'un bilan anabolisme/catabolisme, la variable C n'est donc pas considérée :

$$f(X, S) = (\text{terme d'anabolisme}) - (\text{terme de catabolisme})$$

3. Modèles potentiel × réducteur :

$$f(X, C, S) = f_{pot}(X, S) \times r(C, S)$$

où f_{pot} est la fonction "potentiel" et r est la fonction réducteur. Ces modèles sont basés sur l'hypothèse fonctionnelle sous jacente la plus complète.
En pratique, la fonction potentiel est exprimée sous forme polynomiale ou sous forme sigmoïde. Elle dépend de la dimension de chaque individu.

Les modèles polynomiaux et les modèles de type potentiel × réducteur sont les plus fréquemment utilisés en modélisation individuelle, lorsque l'on travaille à partir de données transversales. C'est le cas dans la majeure partie des études de modélisation de la dynamique forestière en raison essentiellement du type de données disponibles.

1.2 Processus de mortalité

1.2.1 Définition :

La mortalité pour un arbre est l'état de dégradation continue et d'indifférence à l'environnement dans le temps qu'atteignent les variables de sa description. Du point de vue du modélisateur, on distingue deux types de mortalité :

- La mortalité régulière influencée par l'âge et la compétition
- La mortalité irrégulière dû aux catastrophes naturelles ou à l'exploitation.

1.2.2 Modélisations :

La mortalité accidentelle n'est pas en générale considérée dans la modélisation. La mortalité régulière se modélise quant à elle soit de façon déterministe soit de façon stochastique, ce dernier cas est le plus souvent observé. On considère alors que la probabilité de croissance de l'arbre est constante, variable ou dépend des caractéristiques de l'arbre et/ou du peuplement, la plus rependue étant cette dernière.
Les variables les plus recommandées servant à l'estimation de la probabilité de mourir d'un arbre sont :

- Le diamètre et/ou la surface terrière (surface à 1,30 m du sol) su sujet ;
- Un indice de compétition ;
- Une variable décrivant l'accroissement durant la période précédente, en diamètre ou en surface terrière.

Le modèle préconiser pour relier ces variables aux probabilités (ou taux) de mortalité des arbres est le modèle logistique.

1.3 Processus de Recrutement

1.3.1 Définition :

On entend par régénération l'ensemble des processus allant de la floraison d'un arbre adulte à l'apparition d'un nouvel individu dans le peuplement en passant par la production et la dissémination des graines, la germination et l'établissement d'une plantule autotrophe. La taille du nouvel individu peut- être quelconque : une fois fixée, elle définit le seuil de recrutement et l'on appelle recruté tout individu ayant franchi ce seuil durant la période étudiée [14].

1.3.2 Modélisations :

Étant données les difficultés rencontrées dans la description précise et la quantification des processus de régénération, il est compréhensible qu'une solution couramment retenue dans les modèles de dynamique forestière consiste à ignorer les étapes difficiles et à s'attacher à la modélisation du recrutement plutôt qu'à celle de la régénération. Le nombre et l'identité des individus franchissant le seuil de recrutement dépend alors directement de variables caractérisant les potentialités du site ou de la place disponible dans le peuplement sans lien explicite avec les pieds-mères potentiels. La fiabilité d'une telle approche pour l'appréhension de la dynamique forestière demande un contrôle expérimental d'hypothèses sur la régénération, et donc une communication avec le travail de terrain.
D'une manière générale, l'approche de recrutement consiste à considérer un effectif recruté au-dessus de 10 cm de diamètre constant, fonction de la surface terrière locale. On installe donc des dispositifs qui vont permettre de faire le décompte des arbres qui traverse le seuil fixé. Par considération des périodes d'inventaires et des effectifs observés (valeurs entières), la modélisation du recrutement se base immédiatement sur les modèles de régression de Poisson.

1.4 Conclusion

C'est par l'intégration des dynamiques individuelles des arbres dans un modèle de population que l'on parvient à un modèle de dynamique forestière, nous avons défini des processus de dynamique forestières et présenter les modélisations inhérentes à leurs suivis. Nous faisons ainsi un pas vers l'objectif de modélisation. Cependant, pour comprendre la liaison entre le formalisme du modèle et les outils mathématiques utilisés, il est important de pouvoir comprendre ces outils. La suite de cette première partie, est donc basée essentiellement sur l'existant mathématique utile à notre travail.

Chapitre 2

Classification non supervisée

Classer signifie "ranger par classe" et superviser, c'est "contrôler", donc littéralement la classification non supervisée est le fait de ranger sans contrôle. Sans clarification du but poursuivit, cette définition est assez abusive. Ce chapitre permet de savoir ce qu'est une classification et dans quel cas dit-on qu'elle est supervisée, il présente aussi les algorithmes traditionnels de classification non supervisée et se termine sur le choix de la méthode non supervisée la plus adaptée à notre contexte.

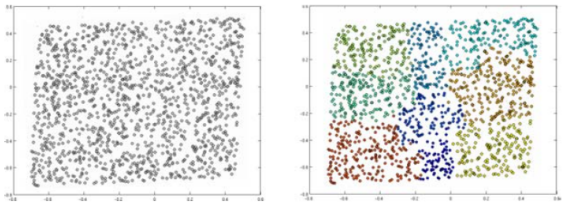

FIGURE 2.1 – Classification non supervisée : retrouver des groupes sur des données sans structure visible. [4]

2.1 Généralités et définitions

En statistique, on parle de classification lorsque l'objectif de modélisation est d'opérer l'identification de classes homogènes d'un ensemble d'individus à partir de leurs traits descriptifs (attributs, caractéristique, etc.). On distingue essentiellement deux types de classification : supervisée et non-supervisée [15].

2.1.1 Classification non-supervisée

Encore appelée "classification automatique", "clustering" ou "regroupement", la classification non supervisée est le fait d'identifier des groupes ou classes d'individus sans informations sur elles.
Supposons que l'on dispose d'un ensemble d'objets que l'on note par $X = \{x_1, x_2, \cdots, x_N\}$

caractérisé par un ensemble de descripteurs, l'objectif du clustering est de trouver l'ensemble des ensembles $G = \{C_1, C_2, \cdots, C_n\}$, en générale une partition de l'espace des individus de X, tel que tout x de X appartienne à un C de G. Ce qui revient à déterminer une fonction Y_s qui associe à chaque élément de X un ou plusieurs éléments de G. Il faut pouvoir affecter une nouvelle observation à une classe C de G. Les observations disponibles ne sont pas initialement identifiées comme appartenant à telle ou telle population [15]. Les méthodes prônées en classification non supervisée sont : la classification ascendante hiérarchique, les algorithmes de réallocation dynamique(k means) et les cartes auto-organisatrices(Kohonen)[16].

2.1.2 Classification supervisée

Dans le contexte supervisé on dispose déjà d'exemples dont la classe est connue et étiquetée. Les données sont donc associées à des labels des classes notés $L = \{l_1, l_2, \cdots, l_n\}$. L'objectif est alors d'apprendre à l'aide d'un modèle d'apprentissage [1] des règles qui permettent de prédire la classe des nouvelles observations ce qui revient à déterminer une fonction Cl qui à partir des descripteurs de l'objet associe une classe l, et de pouvoir aussi affecter toute nouvelle observation à une classe parmi les classes disponibles. Parmi les méthodes supervisées on cite : les k-plus proche voisins, les arbres de décision, les réseaux de neurones, les machines à support de vecteurs(SVM) et les classificateurs de Bayes [15].

2.2 Algorithmes de classification non supervisée

Le but de la classification non supervisée est de trouver une partition des données et de l'évaluer de la manière suivante [17] :

- Si aucune partition de référence n'est connue, vérifier que :
  - Les éléments proches sont dans un même ensemble (ou "cluster") de la partition
  - Les éléments éloignés sont dans deux ensembles différents de la partition

  Les rapprochement et les éloignements étant définit par l'introduction une mesure de distance entre les objets.
- Si une partition de référence est connue

  $\Rightarrow$ Mesurer la distance avec la partition de référence.

Il existe plusieurs méthodes de partitionnement des données, nous présentons par la suites les méthodes les plus implémentées dans les logiciels sur la preuve de leurs efficacités.

2.2.1 Les méthodes hiérarchiques

C'est la forme de classification la plus populaire, elle a l'avantage d'être interprétable visuellement à l'aide des arbres ou dendrogramme. On distingue deux types de classifications hiérarchiques :

1. champ d'étude de l'intelligence artificielle qui concerne la mise en oeuvre de méthodes permettant à une machine d'imiter les systèmes complexes.

La classification ascendante hiérarchique notée (C.A.H) qui se déroule comme suit : à partir des éléments terminaux, on forme de petites classes ne comportant que les individus les plus semblables, et à partir de celles-ci, on construit des classes de moins en moins homogènes jusqu'à obtenir la classe tout entière qui réunit tous les éléments terminaux [18]. En pratique, on parle de bottom-up(l'arbre se construit des feuilles vers la racine)[17].

Exemple en dimension 2 [17] :

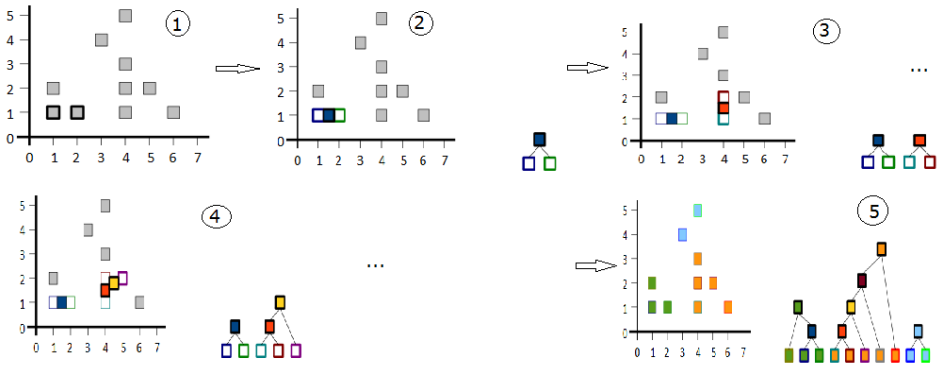

FIGURE 2.2 – Exemple en dimension 2 d'une classification hiérarchique ascendante

La classification descendante hiérarchique notée (C.D.H), il s'agit d'une dichotomie de la classe entière jusqu'à obtenir tous les éléments terminaux [18]. On parle de top-down car l'arbre se construit de la racine vers les feuilles [17].

2.2.1.1 Les étapes d'une classification hiérarchique [18]

Étape 1 : Sélectionner les individus à classer et les variables qui serviront pour critère de classification ;

Étape 2 : Définir une distance ou un indice d'écart entre paires d'individus. En générale on ramène le problème sur un espace vectoriel réel et on utilise la norme euclidienne ;

Étape 3 : Définir une règle de calcul des distances entre classe ;

Étape 4 : Déterminer un critère d'agrégation(minimiser la distance intraclasse ou maximiser la distance iterclasse) des individus dans les classes.

2.2.1.2 Algorithme générale d'une classification hiérarchique [18]

L'algorithme de classification hiérarchique se généralise de la manière suivante :

Algorithme

1. On calcule toutes les distances entre les individus constituant l'ensemble à classer. Supposant par exemple qu'on a n individus à classer, la matrice des distances (dite aussi de proximité) est symétrique. On lit donc $\frac{n(n+1)}{2}$ distances.
2. Le tri de ces distances permet de déterminer les deux éléments qui vont constituer une nouvelle classe. Puis on calcul les distances entre cette classe et les éléments restants que se soit des classes ou des individus.
3. On recommence l'étape 2 avec un élément de moins à chaque itération
4. Itérer jusqu'à ce qu'on ait agrégé tous les individus en une seule classe.

2.2.2 k-moyennes (k-means)

La méthode des $k-moyennes$ reste actuellement la méthode la plus utilisée surtout pour les grand fichier de données. Cette méthode à l'instar de la méthode hiérarchique, a l'avantage d'être efficace et très rapide. La classification hiérarchique a l'inconvénient d'user de toutes les ressources de l'ordinateur. Pour chaque point, elle calcul sa distance à tous les autres. La méthode hiérarchique est itérative et elle est inefficace pour les grands fichiers de données. Le principe de la méthode des $k-means$ c'est que la classification se fait sur la base du critère des plus proches voisins. Cela traduit le fait que chaque individu est affecté à une classe s'il est très proche de son centre de gravité.

Contrairement à la méthode hiérarchique, la méthode des $k-moyennes$ fixe le nombre de classes au départ. Ceci ne se fait pas par hasard, le plus souvent pour estimer ce nombre, on fait de la classification hiérarchique sur un échantillon représentatif des individus. Cette méthode apparait donc comme un complément de la méthode hiérarchique [18].

2.2.2.1 Principe [17] :

Plus généralement l'algorithme des $k-moyenne$ obéit aux étapes suivantes :

1. Choisir k points, par un tirage aléatoire sans remise et les considérer comme des centroïdes
2. Distribuer les points dans les k classes ainsi formées selon leur proximité aux centroïdes

3. Utiliser les barycentres des classes comme nouveaux centroïdes et répéter jusqu'à ce qu'il n'y ait plus de changement.

Exemple pour $k = 3$

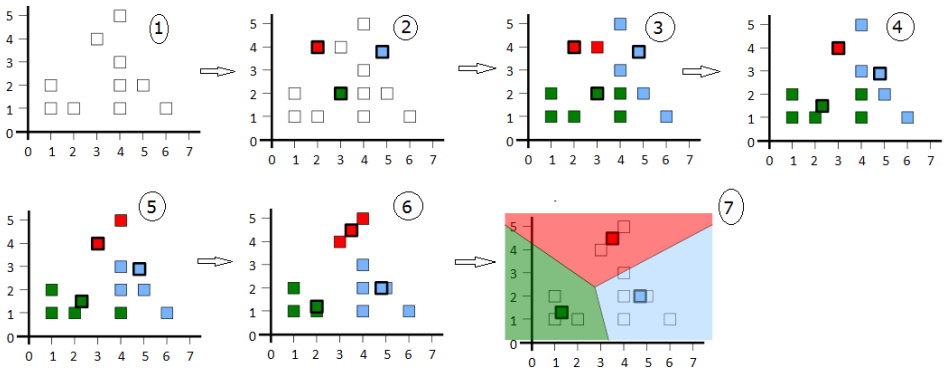

FIGURE 2.3 – Exemple en dimension 2 d'une classification par k=3-moyennes

Remarque :

- Dans cette méthode il y a déplacement des centres, on parle d'algorithme à centre mobile.
- Cette algorithme présente une forte dépendance au choix du nombre k de classe.

Une version stochastique de cette algorithme a été développé par Forgy.

2.2.3 Algorithme de Kohonen

Il s'agit d'un algorithme original de classification qui a été défini par Teuvo Kohonen, dans les années 80. L'algorithme regroupe les observations en classes, en respectant la topologie de l'espace des observations, on parle des cartes de Kohonen, cela veut dire qu'on définit a priori une notion de voisinage entre classe et que des observations voisines dans l'espace des variables appartiennent (après classement) à la même classe ou à des classes voisines [5].
Les voisinages entre classes peuvent être choisis de manière variée, mais en générale on suppose que les classes sont disposées sur une grille rectangulaire qui définit naturellement les voisins de chaque classe.

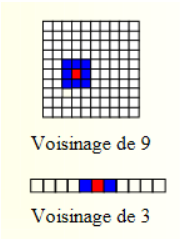

FIGURE 2.4 – Exemple de voisinage entre classes de l'algorithme de Kohonen à grilles carrées [5]

2.2.3.1 Principe de l'algorithme de Kohonen

La structure d'une carte de Kohonen peut être représentée comme un réseau de neurones avec une couche d'entrée, qui correspond à l'observation $z = (z_1, \cdots, z_n)$ de dimension n, et une couche de sortie, qui est composée d'un ensemble de neurones interconnectés et liés entre eux par une structure de graphe non orienté, noté C. Cette dernière définie une structure de voisinage entre les neurones.
Cette couche de traitement est appelée carte. Les neurones de la couche d'entrée sont entièrement connectés aux neurones de la carte, et les états de la couche d'entrée sont forcés aux valeurs des signaux d'entrée.
La topologie de la carte est apprise (ou calculée) par l'algorithme de gradient stochastique de Kohonen, auquel on fournit en entrée des données de dimension n (n pouvant être de dimension très grande) à analyser, ces données étant l'équivalent biologique des signaux du système nerveux. Chaque neurone de la carte correspond alors à un prototype du jeu de données, c'est à dire un individu fictif représentatif d'un ensemble d'individus réels proche de lui même(groupe d'individus). Un neurone de la carte est donc représenté par un vecteur de même dimension que les données. Les composantes de ce vecteur sont les "poids" (notés W) des connexions du neurone aux entrées du réseau, et sont également les coordonnées du prototype associé au neurone dans l'espace multidimensionnel de départ. La propriété d'auto-organisation de la carte lui permet de passer d'un état désorganisé à l'initialisation à un état organisé respectant la topologie des données[19].

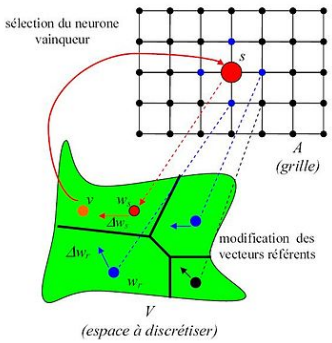

FIGURE 2.5 – Principe de l'algorithme de Kohonen [6]

Plus concrètement [19] :
Soit un réseau de M neurones, et notons k le nombre d'entrées, et $X = (x_1, \cdots, x_k)^t$ un vecteur d'entrée. Les vecteurs d'entrée sont extraits d'un ensemble d'apprentissage A. Cet ensemble contient $Card(A)$ vecteurs. Chaque neurone est caractérisé par un vecteur de poids $W_j = (\omega_{1j}, \cdots, \omega_{kj})^t$, j étant le numéro du neurone. En réponse à un vecteur d'entrée x, le neurone pour lequel la distance quadratique $\|W_j - x\|^2$ est minimale est appelé neurone vainqueur. On note :

$$O_j = \|W_j - x\|^2 = \sum_{i=1}^{K} (\omega_{ij} - x_i)^2 \tag{2.1}$$

On a l'algorithme suivant, nommé **algorithme d'apprentissage de la carte** :

Algorithme

1. Initialisation $t = 0$. Initialisation des vecteurs poids $\{W_1, \cdots, W_M\}$
2. $n = 1$, choix aléatoire d'une permutation ρ de $\{1, \cdots, Card(A)\}$
3. Présentation du vecteur $x_{\rho(n)}$ en entrée.
4. Calcul des sorties des neurones : O_j
5. Détermination du vainqueur (neurone ayant la plus faible sortie)
6. Modification des poids :

$$\triangle W_j = \alpha_{jk}(t)\,(x - W_j) \tag{2.2}$$

7. $n = n + 1$
8. Si $n \leqslant Card(A)$, aller à 3
9. $t = t + 1$
10. Si $t < T$, aller à 2

Les coefficients $\alpha_{jk}(t)$ sont de la forme $\alpha(t, d(j,k))$. d détermine la dimension du réseau. Pour les réseaux en dimension un, on a $d(j,k) = |j - k|$, et on prend un voisinage gaussien, qui donne de meilleurs résultats en pratique.

$$\alpha_{jk}(t) = \alpha_0 \exp\left(-\frac{(j-k)^2}{2\sigma_t}\right) \tag{2.3}$$

α_0 est nommé "vitesse d'apprentissage". Kohonen suggère des valeurs de l'ordre de 10^{-1}. L'écart type σ_t décroit avec t selon une loi exponentielle.

Remarque De tous ces algorithmes que nous venons de voir, le seul qui ne fixe pas le nombre de groupe ou de classe est la méthode hiérarchique, mais elle présente une faible efficacité sur un gros volume de données. Quant à l'algorithme de k-moyennes et l'algorithme de Kohonen, la fixation du nombre k de groupe utilise en général la méthode hiérarchique sur un volume réduit de données, on parle de classification à deux niveaux. D'autres parts l'ensemble de ces modèles se basent sur une approche géométrique pour la classification, ce qui peut alourdir l'objectif de la modélisation qui l'extraction de l'information utile à l'imitation de la génération des observations, nous introduisons par la suite, des modèles qui sont flexibles à la diversité et qui permettent par une approche probabiliste d'atteindre de grands résultats pour la classification.

2.3 Les modèles de mélange et la classification

2.3.1 Généralités

Définition 2.3.1 (Loi mélange)**.** *Une loi de mélange fini M sur un espace $\mathcal{X}$ est une loi de probabilité s'exprimant comme une combinaison convexe de plusieurs lois de probabilité $M_1, \cdots, M_K$ sur $\mathcal{X}$. Autrement dit, il existe K coefficients $\pi_1, \cdots, \pi_K$ ($\pi_k > 0$ et $\sum_{k=1}^{K} \pi_k = 1$) tels que, pour tout $x \in \mathcal{X}$,*

$$M(x) = \sum_{k=1}^{K} \pi_k M_k(x) \tag{2.4}$$

Les π_k et M_k sont respectivement appelées proportions et composantes du mélange .

Cette définition peut s'étendre au cas où on a une infinité de composante, si on considère un ensemble Ω de paramètres de lois f, si on a :

$$\int_{\Omega} \pi(\theta) d\theta = 1$$

alors,

$$f(x, \Omega) = \int_{\Omega} \pi(\theta) f(x, \theta) d\theta$$

est une densité mélange.

2.3.1.1 L'exemple gaussien [20]

Généralement, on suppose en outre que chaque composante P_k appartient à une famille paramétrique $P(., \alpha_k)$ et on note $P(.; \theta)$ la loi mélange associée à ce paramètre, $\theta = (\pi_1, \cdots, \pi_K, \alpha_1, \cdots, \alpha_K)$ désignant le paramètre de ce modèle. Le choix se restreint à des familles $P(.; \alpha_k)$ conduisant à des lois mélanges généralement identifiables [2] dans les situations d'intérêt.
Dans le cas continu où $\mathcal{X} = \mathbb{R}^d$, le modèle paramétrique le plus utilisé est la loi multinomiale :

$$P(., \alpha_k) = \mathcal{N}(\mu_k, \Sigma_k) \tag{2.5}$$

avec $\alpha_k = (\mu_k, \Sigma_k)$, $\mu_k \in \mathbb{R}^d$ désignant la moyenne de la composante k et $\Sigma_k \in \mathbb{R}^{d^2}$ la matrice de variance covariance correspondante.

2. loi dont les paramètres se détermine de façon unique

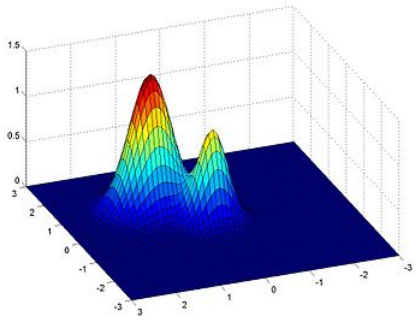

FIGURE 2.6 – Mélange de deux lois gaussiennes en dimension deux [7]

En statistique, on considère un ensemble de données comme la réalisation d'une loi mélange, et on cherche alors retrouver les paramètres de cette loi générative des données. Cette approche permet de partitionner les données en catégories suivant les proportions du mélange. Il s'agit bel et bien de la classification non supervisée. La recherche des paramètres de la loi mélange associée à ce genre de donnée par la méthode du maximum de vraisemblance(EMV) a conduit à la naissance d'un algorithme performant et très recommandé pour ce genre de situation : L'algorithme EM.

2.3.2 Algorithme EM (Expectation-Maximization)

L'algorithme EM [21] est une méthode d'estimation paramétrique s'inscrivant dans le cadre général du maximum de vraisemblance.
Lorsque les seules données dont on dispose ne permettent pas l'estimation des paramètres, et/ou que l'expression de la vraisemblance est analytiquement impossible à maximiser, l'algorithme EM peut être une solution. De manière grossière et vague, il vise à fournir un estimateur lorsque cette impossibilité provient de la présence de données cachées ou manquantes [22] : la connaissance de la répartition.

2.3.2.1 Principe générale de l'algorithme EM

L'algorithme EM tire son nom du fait qu'à chaque itération il opère deux étapes distinctes :

Etape Expectation souvent désignée comme « l'étape E », procède comme son nom le laisse supposer à l'estimation des données inconnues, sachant les données observées et la valeur des paramètres déterminée à l'itération précédente ;

Etape Maximization ou « étape M », procède donc à la maximisation de la vraisemblance, rendue désormais possible en utilisant l'estimation des données inconnues effectuée à l'étape précédente, et met à jour la valeur du ou des paramètre(s) pour la prochaine itération.

2.3.2.2 Notes sur EM

EM fait croître la vraisemblance, il converge très souvent vers un point stationnaire qui peut être typiquement le maximum global, un maximum local ou un point selle [20] si la vraisemblance en possède un bien-sûr. L'initialisation est très importante pour mener à bien la convergence de l'algorithme, il est recommandé de procéder à plusieurs maximisations en changeant les valeurs initiales.
L'objectif de détermination des composantes du mélange conduit aussi EM à un but de classement. Cette algorithme a la caractéristique de mener à un double objectif, celui de l'estimation des paramètres et celui du partitionnement. Le choix du nombre K de groupe(nombre de composante de la loi mélange) se fait en très souvent par l'un des trois critères de statistique mathématique de référence suivant :

- **AIC** (*An Information Criterion*) : Un "bon" modèle est celui qui utilise un compromis en terme de "biais-variance". Il y a indépendance avec le nombre d'observation.
- **BIC** (*Bayesian Information Criterion*) : un "bon" modèle est celui qui maximise la vraisemblance intégrée(sur les paramètres) lorsque chaque modèle en compétition est équiprobable *à priori*. Celle ci présente une dépendance avec le nombre d'observation.
- **ICL** (*Integrated Complete Likelihood*) qui est un critère BIC pénalisé par l'imbrication(pas d'adjacence entre classe), très adapté à EM pour assurer la partition.

Pour la classification, il est possible de définir une approche géométrique similaire au cas des centre mobile en utilisant les modèles parcimonieux gaussiens [20]. Mais très souvent la classification se fait par une variante de EM : **ECM** (C pour classification) qui poursuit se focalise beaucoup plus dans le sens du classement mais présente très souvent des estimateurs biaisés [3]. On utilise aussi le principe de **MAP**(*Maximum à Postériori*) qui consiste à considérer comme classe d'un individu ou d'un style d'individu, celle où il est le plus probable d'être par rapport à l'estimateur trouvé.

2.4 Conclusion

Nous avons présenté dans ce chapitre, les algorithmes permettant de faire le partitionnement d'un ensemble de données, nous avons parlé de la méthode hiérarchique qui est très performante pour un volume réduit de données, nous avons aussi présenter la méthode des k-moyennes et la méthode de Kohonen qui sont individuellement complémentaire à la méthode hiérarchique et permette d'attaquer de grand volume de donnée, celle de Kohonen étant la plus générale. Ensuite nous avons introduit le modèle le plus flexible et plus adaptatif aux données abondantes et diversifiés [20] : le modèle de mélange dont l'algorithme de classification utilisé est l'algorithme EM (Expectation-Maximization) qui a l'avantage de conduire autant à la classification qu'à un procédé de simulation pour la description comportementaliste et générative des individus de chaque classe.
Nous sommes ainsi parvenu à pourvoir faire le choix de notre procédure de classement, l'algorithme EM présentant un ensemble de caractéristiques adaptées à notre contexte, nous

3. l'espérance mathématique(moyenne) de l'écart à la vrai valeur n'est pas nulle(non négligeable)

l'utiliserons spécialement dans la deuxième partie de ce mémoire en donnant dans notre cas spécifique la preuve qu'elle conduit vers la solution désirée.
Cependant, étant donné que nous poursuivons également un but de d'inférence sur la dynamique individuelle des arbres, il serait judicieux pour nous de définir les méthodes de régression notamment citées sur les modèles de forêt, utiles à la bonne marche de notre travail.

Chapitre 3

Régression Logistique et régression de Poisson

En statistique, lorsque les variables aléatoires observées ne peuvent être considérées comme identiquement distribuées, même en présence d'indépendance, on recherche un modèle explicatif dit de régression.
De façon informelle, un modèle dit explicatif est un modèle exprimant une variable $\mathcal{Y}$, appelée *variable à expliquer*(ou réponse), comme fonction d'une ou plusieurs variables dites *variables explicatives* ou prédicteurs. Néanmoins, si l'entité $\mathcal{Y}$ est considérée comme une variable aléatoire Y, un terme aléatoire, caractérisant l'incertitude de la prédiction, doit être introduit d'une certaine manière dans l'équation du modèle.
Dans un *modèle de régression*, on cherche essentiellement à déterminer la variation de **l'espérance mathématique** de Y en fonction des variables explicatives [23].
Nous allons faire une brève présentation de deux modèles parmi les modèles GLM (Generalized Linear Models) : la régression logistique et de la régression de Poisson. Nous noterons $X_1, \cdots X_L$ les variables explicatives. Chaque section de ce chapitre présente la forme d'une fonction dite "fonction de lien" g qui permet de faire la liaison entre la partie déterministe dépendante du paramètre θ à estimer, et l'espérance μ de la partie aléatoire (la variable à expliquer). On a la relation :

$$g(\mu) = \theta_0 + \theta_1 X_1 + \cdots + \theta_L X_L = X\theta \tag{3.1}$$

$X = (1, X_1, \cdots, X_L)$, $\theta = (\theta_0, \cdots, \theta_L)^t$. On notera $x_i = (1, x_{i,1}, \cdots, x_{i,L})$ l'observation des variables variables explicatives pour un individu i, et dans le cas général on notera x pour parler d'un individu quelconque.

3.1 Régression logistique

Nous donnons la définition simplifiée de ce modèle, où on parle encore de classification binaire : la variable aléatoire observée ne peut prendre que deux valeurs que l'on identifie par les labels de $\{0, 1\}$.

Définition 3.1.1. *(Régression logistique) [24] Le modèle logistique propose une modélisation de la loi de $Y|X = x$ par la loi de Bernoulli de paramètre $p_\theta(x) = \mathbb{P}_\theta(Y = 1|X = x)$ telle*

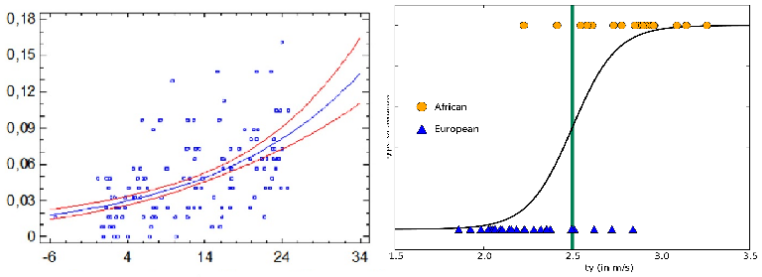

FIGURE 3.1 – Régression de Poisson et régression logistique [8]

que :

$$\ln\left(\frac{p_\theta(x)}{1-p_\theta(x)}\right) = \theta_0 + \theta_1 x_1 + \cdots + \theta_L x_L \tag{3.2}$$

ou encore

$$logit(p_\theta(x)) = x\theta$$

logit désignant la fonction bijective de $]0,1[$ *dans* $\mathbb{R}$ *:* $p \to \ln(p/(1-p))$.

En utilisant la réciproque de la fonction logit, L'égalité 3.2 s'écrit aussi :

$$p_\theta(x) = \mathbb{P}_\theta(Y=1|X=x) = \frac{\exp(x\theta)}{1+\exp(x\theta)} = [1+\exp(-x\theta)]^{-1} \tag{3.3}$$

Remarque :

- La fonction de lien est $g = \text{logit}$
- $\begin{cases} \mathbb{E}_\theta[Y|X=x] = \mathbb{P}_\theta(Y=1|X=x) \\ \mathbb{V}_\theta[Y|X=x] = \mathbb{P}_\theta(Y=1|X=x)\,(1-)\,\mathbb{P}_\theta(Y=1|X=x) \end{cases}$

Cette dernière remarque implique que la variance n'est pas constante en varie selon l'individu, on parle d'hétéroscédasticité.

3.2 Regression de Poisson

On parle aussi de modèle log-linéaire. Ce modèle est utilisé lorsque la variable à expliquer est une *variable de comptage*, par exemple :

- Le nombre de catastrophe aériennes sur une période donnée
- Le nombre de nouvel élève d'un lycée chaque rentrée scolaire
- Le nombre d'accident par jour sur une autoroute

Définition 3.2.1. *(Régression log-linéaire) Le modèle log-linéaire propose une modélisation de la loi de* $Y|X=x$ *par une loi de Poisson de paramètre* $\lambda = \lambda(x)$ *telle que :*

$$\ln\left(\mathbb{E}[Y|X=x]\right) = x\theta \tag{3.4}$$

Pour une nouvelle mesure x effectuée, le modèle log-linéaire va donc prédire la valeur $\exp(x\theta)$

Rappel : L'espérance et la variance d'une loi de Poisson c'est la valeur de son paramètre. Ceci permet de dire aussi que la régression de Poisson prédit la moyenne des valeurs observées.

Remarque : Pour la régression de Poisson la fonction $g = \ln$, qui a l'avantage d'être aussi bijective prenant les valeurs dans $[0, +\infty[$.

3.3 Conclusion

Nous venons de voir deux modèles GLM (Generalized Linear Models) qui permettent de faire de la régression sur des observations de variable aléatoire à valeur binaire ou de comptage, la première étant expliquée par un modèle logistique qui utilise la fonction logit (logarithme du rapport des chances) pour le lien avec les combinaisons linéaires des variables explicatives et la dernière étant expliquée par le modèle log-linéaire ou modèle de Poisson qui utilise le lien log (logarithme).
Nous terminons ainsi la présentation des outils théoriques de notre travail. Nonobstant, nous achevons cette partie par la présentation des outils d'implémentation utilisés.

Chapitre 4

Logiciels utilisés

Pour parvenir à l'application des sciences à la vie, l'ordinateur, reste le seul moyen d'automatisation et de minimisation de l'erreur que l'homme puisse utiliser. Son développement croissant apporte jour après jour un ensemble de programmes pouvant communiquer pour réaliser les tâches de l'humain. Pour faire de la statistique, elle offre des logiciels de programmation et de manipulation de données permettant de mettre en œuvre des aides à la décision. Nous utilisons pour notre cas, deux principaux logiciels : R et Notepad++.

FIGURE 4.1 – Logo du logiciel R et de l'éditeur Notepad++ [9] [10]

4.1 Le logiciel R

4.1.1 Présentation Générale

R est un logiciel Open Source (gratuit et libre en développement) développé en C pour les statistiques et les graphiques. Il donne une grande variété pour les calculs statistiques comme la modélisation linéaire ou non linéaire, les tests statistiques, la classification, et autres technique statistiques et graphiques.
Le grand atout de R est sa facilité à produire les graphiques avec des symboles ou des formules mathématiques intégrées. R est téléchargeable gratuitement sur `cran.r-project.org`.

4.1.2 Mode de fonctionnement

[25]

Deux moyens de fonctionnement :

(1) R est un interpréteur, il permet de programmer avec le langage S (on peut parler aussi de langage R). L'objet de base est un vecteur de données.
C'est un réel langage de programmation, c.-à-d. types de données, branchements conditionnels, boucles, organisation du code en procédures et fonctions, découpage en modules. Pour exécuter un programme R, on le transmet via un script ".r"

(2) R est un logiciel de statistique et de data mining, pilotée en ligne de commande. Il est extensible (quasiment) à l'infini via le système des packages.
Les instructions servent à manipuler les objets R c.-à-d. les ensembles de données, les vecteurs, les modèles, etc. Nombreuses fonctions graphiques disponibles via la fonction "plot()".
L'exécution se fait en introduisant la commande dans le terminal, manipulation interactive.

Les langages interprétés ont la caractéristique d'être lent en générale car l'exécution se fait de manière séquentiel, mais R présente une différence, car il possède des fonctions de bases (apply, eigen, svd, etc) qui sont compilées. Bref, les problèmes en R sont causés par les boucles. Cependant on conseil beaucoup la "vectorisation" en R, cela consiste à regrouper les données dans les vecteurs, matrices, liste ou data.frame et à faire les opérations en utilisant les fonctions de bases.

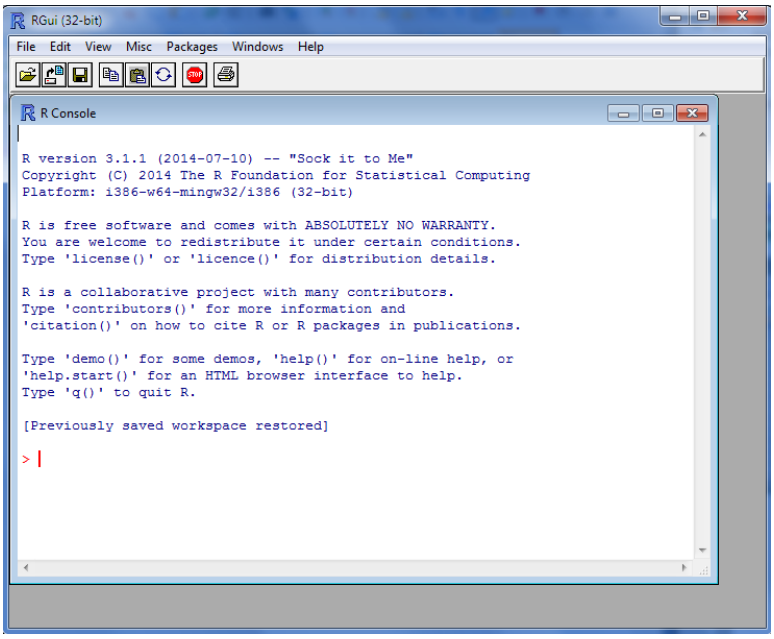

FIGURE 4.2 – Console de R

4.2 L'éditeur Notepad++

Notepad++ est un éditeur de texte générique qui est écrit en C++ (d'où son nom) par Don Ho, un informaticien basé à Paris diplômé de l'Université Paris VII - Diderot en 2000. Particulièrement performant, cet éditeur est distribué gratuitement sous licence GPL.
Il intègre plusieurs syntaxes de langage (parmi lesquelles le langage R) avec coloration syntaxique.

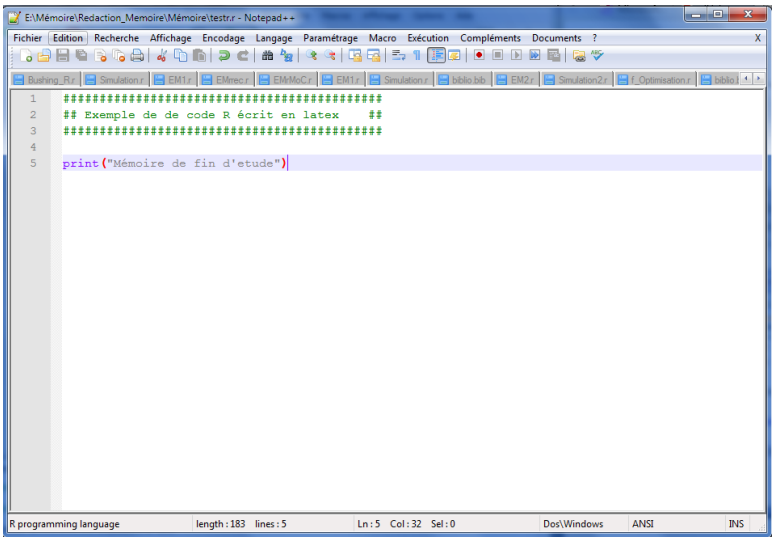

FIGURE 4.3 – Editeur Notepad++

4.3 Conclusion

Nous venons de voir deux logiciels dont un éditeur pour les scripts (programme sur fichier) et le langage R qui permet de faire l'application des statistiques. Le couplage de ces deux logiciels peut donc bien conduire à l'élaboration informatique d'un modèle statistique, particulièrement le notre.

Deuxième partie

Mise en œuvre

Introduction

Pour pouvoir suivre l'évolution d'une forêt, il est nécessaire de modéliser les trois composantes de sa dynamique : la croissance, la mortalité et la régénération. Vu globalement, une forêt tropical est faite d'un peuplement d'arbres hétérogènes rendant sa description complexe à cause de sa grande diversité spécifique. Pour résoudre ce problème, on se propose de regrouper les espèces d'arbres en groupes ayant un comportement similaire suivant les trois processus et les classes diamétriques. Cela va nous permettre de partitionner le peuplement forestier pour décrire l'évolution de chaque espèce dans chaque classe de diamètre et ainsi prédire la dynamique globale de la forêt.
DafSim est un logiciel de moteur DafMod, développé dans le cadre du projet DynAfFor pour la dynamique des forêts d'Afrique Centrale, notre travail est une contribution pour l'augmentation de la performance de DafMod pour le procédé de classification (non supervisée) par la modélisation d'un algorithme propre à notre problème. Nous fournissons aussi une approche d'extension de notre algorithme au cas générale des inventaires de données collectées à intervalle de temps quelconque, obstacle que nous surmontons en faisant usage d'interpolation pour la complétion des données.

Vue globale du problème

L'hétérogénéité d'un peuplement se décrit grâce aux modèles à base de fonctions de distribution. Ces modèles en écologie sont issus de la démographie humaine et ont été développés par Leslie en 1945 [13]. Ils résument la population par des fonctions de distribution sur une ou plusieurs variables. Il s'agit donc de suivre l'évolution de ces fonctions de distribution dans le temps. Nous nous intéressons à la distribution diamétrique à temps discret. En effet, la variation du diamètre des arbres est divisée en L catégories et on note $n_l(t)$ le nombre d'arbre qui sont dans la catégorie l pour le diamètre à la date t. L'état de la forêt est alors décrit par le vecteur :

$$N_t = (n_l(t))_{l \leqslant L}$$

Nous sommes ainsi rattachés par discrétisation du temps, aux chaînes de Markov qui sont au cœur des modèles linéaires dynamiques. L'évolution se définie alors de la manière suivante :

$$N_{t+1} = M(t, N_t)N_t$$

Où nous considérons un lien direct entre la matrice M de taille $(L+1) \times (L+1)$ et les effectifs $n_l(t)$ de chaque catégorie(modèle **densité-dépendant**). Pour rester dans le cadre

d'un modèle stochastique, on utilise un modèle de Usher qui permet d'obtenir une matrice stochastique(valeur positives et colonne de somme 1) :

$$M = C + R \quad \text{avec} \quad C = \begin{pmatrix} q_1 & 0 & \cdots & & 0 \\ p_1 & q_2 & \cdots & & 0 \\ & \ddots & \ddots & & \\ & \cdots & p_{L-1} & q_L & 0 \\ m_1 & \cdots & & m_L & 1 \end{pmatrix} \tag{4.1}$$

$$\text{et} \quad R = \begin{pmatrix} r_1 & \cdots & r_L & 0 \\ & & & \\ & 0 & & \\ & & & \end{pmatrix} \tag{4.2}$$

Où

- La matrice C est une matrice Markovienne, dont la chaîne de Markov associée a pour variable aléatoire la catégorie dans laquelle se situe un arbre arbitraire.
- q_l est la probabilité qu'un arbre reste dans la classe l en survivant.
- p_l est la probabilité qu'un arbre quitte la classe l pour la classe $l+1$ en survivant.
- $m_l = 1 - p_l - q_l$ $(l = 1, \cdots L-1)$ et $m_L = 1 - q_L$ sont les probabilité de mortalité dans la classe de l'indice.
- La matrice R représente le recrutement, r_l est le nombre moyen de recrut dans la classe l.

Cette matrice illustre trois transitions des classes résumées dans le schéma suivant :

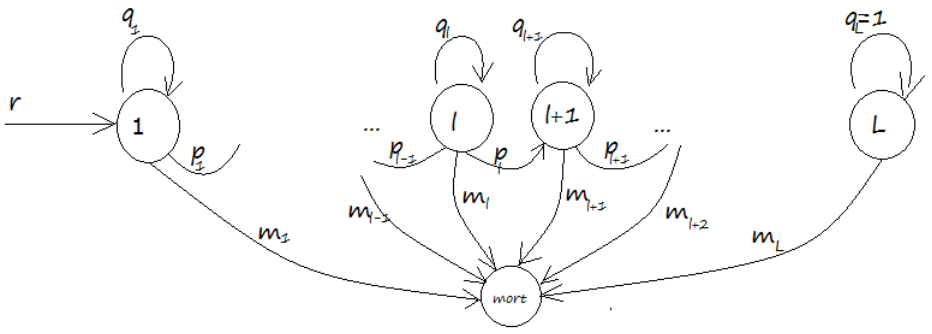

FIGURE 4.4 – Automate de transition des classes diamétriques

Notre but est d'inférer sur la dynamique interne d'une classe pour la détermination des p_l, $m_l\, r_l$. En se basant sur les études mener en biométrie nous savons que l'explication de la

dynamique d'une forêt se base sur trois principales variables : l'espèce, la surface terrière qui est le peuplement en terme de surface (la surface pour un arbre étant prise à la hauteur 1,30 m), et le diamètre. La forêt est donc partitionnée en L classes de diamètres, et les processus se définissent de la manière suivante :

- La croissance est le fait pour un arbre de passer de sa classe de diamètre à la classe de diamètre directement au dessus ;
- La mortalité, il s'agit ici de la mortalité naturelle, celle qui n'est pas liée à l'exploitation forestière.
- Le recrutement, qui est le fait qu'un arbre juvénile intègre la première classe de diamètre et devient pris en compte dans le peuplement.

Dans une classe, nous voulons pour chaque arbre d'une espèce, inférer sur les trois processus à l'aide de données de surfaces terrières de toutes les classes de diamètre. Ce qui nous fait L variables explicatives. Mais cela nous conduit à un modèle par espèce, ce qui n'est pas réaliste vu la richesse spécifique et la faible représentation de certaines espèces. D'où l'idée de regrouper les espèces pour chacun des trois processus séparément et de faire une régression en mélange.

La suite de notre travail se divise en deux grandes parties. Dans la première nous travaillons pour des inventaires à intervalles de temps réguliers, et la deuxième partie sera une généralisation de celle-ci aux inventaires à intervalles de temps quelconques.

Chapitre 5

Modélisation pour les intervalles de temps réguliers entre les inventaires

Lorsqu'on parle d'inventaires à intervalles de temps réguliers, on veut par là dire que le temps qui sépare deux inventaires consécutifs est constant, en générale il s'agit d'une année.

5.1 Modèle de mélange pour la mortalité

On représente les données de la manière suivante :

TABLE 5.1 – Tableau de données pour la mortalité, $i = 1, \cdots N$

Y		X			
m	n	Esp	X_1	$\cdots$	X_L
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$m_{s,i}$	$n_{s,i}$	s	$x_{s,i,1}$	$\cdots$	$x_{s,i,L}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Y désigne la variable aléatoire à expliquer à laide des données de la variable X qui est la partie déterministe.

- m désigne le nombre d'arbres morts
- n désigne le nombre d'arbres observés
- Esp désigne l'espèce. **NB :** les valeurs de cette colonne sont répétitives. On notera S le nombre d'espèce et n_s le nombre de fois que figure l'espèce s sur le tableau de données.
- $X_1, \cdots, X_L$ désignent les variables explicatives, chacune donnant la surface terrière de la classe diamétrique de son indice.
- Hypothèse : On admet l'indépendance entre les lignes du tableau de données.

Remarque : Dans la pratique, les lignes représentent les parcelles où plusieurs observations sur les arbres d'une espèce ont étés faites.

On pose $x_{s,i} = (1, x_{s,i,1}, \cdots, x_{s,i,L})$, le vecteur de données des variables explicatives de l'individu(les arbres de l'espèce s d'une parcelle) à la ligne i.

Étant donné que les données de chaque ligne représentent les observations sur plusieurs arbres, on se procède par la modélisation pour un seul arbre afin de parvenir à la modélisation pour une espèce.

Soit a un arbre de l'espèce de la ligne i et s l'espèce présente. Notons $y_{s,i,a}$ la variable aléatoire binaire à valeur dans {0, 1} donnant la mort(1) ou la vie(0) de l'arbre a. $y_{s,i,a}$ suit une loi de Bernoulli, on note $p_{s,i,a} = \mathbb{P}\{y_{s,i,a} = 1\}$ son paramètre.
Pour inférer sur la mortalité des arbres de la ligne i, on pourrait penser à un modèle logistique 3.2 à l'aide des variables explicatives. Cependant, les mêmes valeurs de variables explicatives servent à l'explication de la mortalité de tous les arbres à cette ligne, ainsi la probabilité de mourir $p_{s,i,a}$ est considéré indépendante directement à l'arbre a, et donc, il n'est pas question de régression logistique sur une ligne (homoscédasticité).

$$p_{s,i,a} = p_{s,i} \quad \forall a$$

En considérant la variable $y_{s,i}$ comme celle donnant la mortalité sur tous les $n_{s,i}$ arbres de la ligne i, $y_{s,i}$ est une répétition de loi de Bernoulli, donc une loi Binomiale $\mathcal{B}(n_{s,i}, p_{s,i})$. L'estimateur du maximum de vraisemblance de la probabilité de mourir sur les n_i arbres est $\dfrac{m_{s,i}}{n_{s,i}}$.

Or cette estimation est aussi inconnue puisqu'elle correspond à la partie aléatoire du tableau de données **??**. L'observation répétée sur les parcelles de l'espèce s donnant n_s données $x_{s,j}$, $j \in \{1, \cdots, n_s\}$, nous conduit alors à un problème de régression logistique car $\dfrac{m_{s,j}}{n_{s,j}} \in$]0, 1[est une probabilité. (on envisage pas la possibilité d'une totale mort de tous les arbres dans une parcelle, $m_{s,j} < n_{s,j}$).

Ainsi, le modèle de régression sur la mortalité des arbres d'une espèce, est le modèle logistique qui s'écrit :

$$\text{logit}\left(\frac{m_{s,j}}{n_{s,j}}\right) = \theta_0 + x_{s,j,1}\theta_1 + \cdots + x_{s,j,L}\theta_L = x_{s,j}\theta \quad j = 1, n_s \tag{5.1}$$

Remarque :

- Cette forme du modèle logistique est sa forme générale, on parle de modèle logistique à données groupés ou répétées [24]. L'observation de la variable réponse Y est vu comme le rapport $\dfrac{m}{n}$
- l'indice j est utilisé pour les lignes du tableau de données correspondant à l'espèce s, on utilise pas l'indice i parce que l'ordre n'est pas le même, ceci nous permet de nous affranchir de la définition d'une fonction injective entre les indices.

Nous sommes donc parvenu à trouver le modèle de régression pour une espèce. Or comme nous avons spécifier depuis l'introduction de cette partie que le nombre d'espèce est très divers et on retrouve la présence d'espèces rares. On fait donc l'hypothèse d'une loi mélange(mélange de loi logistique) sur l'ensemble des espèces. On note K le nombre de composantes du mélange. Il s'agira donc d'estimer le vecteur de paramètre $\theta = (\pi_1, \cdots, \pi_K, \theta_1, \cdots, \theta_K)$ où les π_k représentent les paramètres du mélange et $\theta_k = (\theta_{k,0}, \cdots, \theta_{k,L})$.

Ecriture de la loi mélange
Supposons que l'espèce s est du groupe k, on pose $\frac{m_{s,j}}{n_{s,j}} = p_{s,j,\theta_k}$. On a :

$$\mathbb{P}(y_{s,j} = m_{s,j}) = \binom{n_{s,j}}{m_{s,j}} p_{s,j,\theta_k}^{m_{s,j}} (1 - p_{s,j,\theta_k})^{n_{s,j} - m_{s,j}} \tag{5.2}$$

En considérant l'indépendance des observations sur l'espèce (hypothèse), on a

$$\mathbb{P}\{(y_{s,1}, \cdots, y_{s,j}, \cdots, y_{s,n_s}) = (m_{s,1}, \cdots, m_{s,j}, \cdots, m_{s,n_s})\} = \prod_{j=1}^{n_s} \binom{n_{s,j}}{m_{s,j}} p_{s,j,\theta_k}^{m_{s,j}} (1-p_{s,j,\theta_k})^{n_{s,j} - m_{s,j}} \tag{5.3}$$

En utilisant l'inverse de la fonction logit 3.2, si on pose

$$P(y_s | x_s, \theta_k) = \mathbb{P}\{(y_{s,1}, \cdots, y_{s,j}, \cdots, y_{s,n_s}) = (m_{s,1}, \cdots, m_{s,j}, \cdots, m_{s,n_s})\}$$

on a alors la k^{e} composante du mélange suivante :

$$P(y_s | x_s, \theta_k) = \prod_{j=1}^{n_s} \binom{n_{s,j}}{m_{s,j}} \Big(\frac{\exp(x_{s,j}\theta_k)}{1 + \exp(x_{s,j}\theta_k)} \Big)^{m_{s,j}} \Big(1 - \frac{\exp(x_{s,j}\theta_k)}{1 + \exp(x_{s,j}\theta_k)} \Big)^{n_{s,j} - n_{s,j}} \tag{5.4}$$

$$= \prod_{j=1}^{n_s} \binom{n_{s,j}}{m_{s,j}} \frac{\exp(x_{s,j}\theta_k)^{m_{s,j}}}{(1 + \exp(x_{s,j}\theta_k))^{n_{s,j}}} \tag{5.5}$$

D'où la loi mélange

$$\boxed{P(y_s | x_s, \theta) = \sum_{k=1}^{K} \pi_k P(y_s | x_s, \theta_k) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{n_s} \binom{n_{s,j}}{m_{s,j}} \frac{\exp(x_{s,j}\theta_k)^{m_{s,j}}}{(1 + \exp(x_{s,j}\theta_k))^{n_{s,j}}}} \tag{5.6}$$

Remarque : x_s et y_s font référence à tous les données qui correspondent à l'espèce s.

5.2 Modèle de mélange pour la croissance

On a le tableau 5.2

Dans notre cas de figure, la croissance se présente exactement comme la mortalité, en effet, la croissance est modélisé comme une variable binaire tout comme la mortalité, ainsi pour un arbre a quelconque, on a les deux éventualités suivante : {"*granir*", "*pas grandir*"}. Le

TABLE 5.2 – Tableau de données pour la croissance, $i = 1, \cdots N$

Y		X			
g	n	Esp	X_1	$\cdots$	X_L
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$g_{s,i}$	$n_{s,i}$	s	$x_{s,i,1}$	$\cdots$	$x_{s,i,L}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

procédé est donc exactement le même que celui de la mortalité(remplacement de m par g). On déduit alors la loi mélange suivante :

$$\boxed{P(y_s|x_s\theta) = \sum_{k=1}^{K} \pi_k P(y_s|x_s, \theta_k) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{n_s} \binom{n_{s,j}}{g_{s,j}} \frac{\exp(x_{s,j}\theta_k)^{g_{s,j}}}{(1+\exp(x_{s,j}\theta_k))^{n_{s,j}}}} \tag{5.7}$$

5.3 Modèle de mélange pour le recrutement

TABLE 5.3 – Tableau de données pour le recrutement, $i = 1, \cdots N$

Y	X			
r	Esp	X_1	$\cdots$	X_L
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$r_{s,i}$	s	$x_{s,i,1}$	$\cdots$	$x_{s,i,L}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Nous considérons directement dans cette partie l'idée de loi mélange, comme loi régissant les observations pour une espèce.
$r_{s,i}$ désigne pour l'espèce s, le nombre moyen de recrut à la ligne i. Si on note $y_{s,i}$ la variable aléatoire réponse de cette ligne, on sait d'après la définition du modèle de Poisson vu à la première partie 3.4 que $y_{s,i}$ suit une loi de poisson $\mathcal{P}(\lambda_{\theta_k,s,i})$. On a la relation :

$$\lambda_{\theta_k,s,i} = \exp(x_{s,i}\theta_k)$$

On a pour l'espèce s

$$\mathbb{P}\{(y_{s,1}, \cdots, y_{s,j}, \cdots, y_{s,n_s}) = (r_{s,1}, \cdots, r_{s,j}, \cdots, r_{s,n_s})\} = \prod_{j=1}^{n_s} \frac{\lambda_{\theta_k,s,j}^{r_{s,j}}}{r_{s,j}!} \exp(-\lambda_{\theta_k,s,j}) \tag{5.8}$$

On déduit alors la densité mélange de régression log-linéaire suivante :

$$\boxed{P(y_s|x_s, \theta) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{n_s} \frac{\exp\left(r_{s,j}x_{s,j}\theta_k - \exp(x_{s,j}\theta_k)\right)}{r_{s,j}!}} \tag{5.9}$$

$\theta = (\pi_1, \cdots, \pi_K, \theta_1, \cdots, \theta_K)$
Rappel de notation : l'indice j lié à l'espèce n'est pas dans pas dans le même ordre que l'indice i qui est plus générale et lié au tableau de données.

Maintenant il nous reste à formuler l'algorithme EM pour, la détermination des paramètres θ. Mais avant, présentons les preuves théoriques de la convergence de cet algorithme tout en ressortant son approche de maximisation de la vraisemblance. Et par la suite nous utiliserons ces résultats.

5.4 Appuis Théoriques de l'usage de EM (Expectation-Maximisation)

Notons par $P(y_s|x_s, \theta)$, la densité mélange associé aux observations, et par S le nombre d'observation. Note : ce que nous présentons ici est une généralisation pour le cas discret de la preuve de la robustesse de l'algorithme EM. Nous utilisons les notations de notre contexte afin de faciliter la compréhension de la suite du modèle.
Donc on a la loi mélange

$$P(y_s|x_s, \theta) = \sum_{k=1}^{K} \pi_k P(y_s|x_s\theta_k)$$

On a la vraisemblance suivante :

$$P(X;\theta) = \prod_{s=1}^{S} P(y_s|x_s, \theta) = \prod_{s=1}^{S} \sum_{k=1}^{K} \pi_k P(y_s|x_s\theta_k) \tag{5.10}$$

et la log-vraisemblance

$$L(X;\theta) = \sum_{s=1}^{S} \ln\Big(\sum_{k=1}^{K} \pi_k P(y_s|x_s\theta_k)\Big) \tag{5.11}$$

Remarque :

- Nous ne faisons pas fi de la variable Y, c'est juste que l'on se situe dans un contexte d'estimation, la régression étant implicite car comme on peut le remarquer, lors de la recherche du modèle mélange, c'est la statistique $\theta = (\pi_1, \cdots, \pi_K, \theta_1, \cdots, \theta_K)$ qui permettra de faire de l'inférence et du classement sur les données.
- Le passage à la log-vraisemblance transforme le P en L

Constatons que, la maximisation de la log-vraisemblance obtenu est impossible, cela à cause de l'ignorance des probabilités d'appartenance. C'est là qu'intervient l'algorithme EM, sa philosophie est de chercher un moyen de contourner l'obstacle pour atteindre le même objectif de maximisation de la vraisemblance. Et c'est ainsi qu'il résout deux problèmes : le problème du secret qui est celui de trouver ce qui est caché, et le problème du réel qui est celui de trouver ce qu'on cherche.
La démarche de EM c'est de rechercher une fonction Q que l'on sait maximiser et qui est telle que :

$$\begin{cases} L(X;\theta) \geqslant Q(\theta|\theta_{(m)}) \\ Q(\theta_{(m+1)}|\theta_{(m)}) \geqslant Q(\theta_{(m)}|\theta_{(m)}) \end{cases} \tag{5.12}$$

où m désigne une étape de l'algorithme et $\theta_{(m)}$ l'estimation de θ à cette étape. En effet l'algorithme EM est un algorithme itératif qui part d'un état initial quelconque et qui maximise Q à chaque itération en améliorant à chaque fois l'estimation du paramètre θ jusqu'à stabilité. A ce moment là, Q est soit sur un maximum global soit sur un maximum local [20].
Avant de déterminer Q, nous donnons un théorème clé pour la suite.

Théorème 5.4.1. ***Inégalité de Jensen.***
Soit f une fonction convexe définie sur un intervalle I. Si $x_1, \cdots, x_n \in I$ et $\lambda_1, \cdots, \lambda_n \geqslant 0$ tels que $\sum_{i=1}^{n} \lambda_i = 1$, alors :

$$f(\sum_{i=1}^{n} \lambda_i x_i) \leqslant \sum_{i=1}^{n} \lambda_i f(x_i)$$

La preuve de ce lemme, se fait par récurrence avec l'utilisation de la définition de la convexité. La fonction logarithme étant concave, l'inégalité du lemme se renverse pour elle.

Pour trouver Q, on s'appuie sur des données complétées par la variable Z inconnue. En notant $P(Z|X;\theta) = \mathbb{P}(Z|X;\theta)$, la probabilité de Z sachant X et le paramètre θ , on peut définir la log-vraisemblance complétée comme la quantité :

$$\begin{aligned} P(X,Z;\theta) = \mathbb{P}(X,Z;\theta) &= \mathbb{P}(Z|X;\theta)\mathbb{P}(X;\theta) \\ &\Rightarrow L(X,Z;\theta) = L(Z|X;\theta) + L(X;\theta) \\ &\Rightarrow L(X;\theta) = L(X,Z;\theta) - L(Z|X;\theta) \end{aligned}$$

En prenant membre à membre par l'espérance conditionnellement sur la variable Z et le paramètre courant $\theta_{(m)}$ on a :

$$L(X;\theta) = \underbrace{\mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X;\theta)\big]}_{\text{indépendant de Z}} = \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta)\big] - \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(Z|X;\theta)\big]$$

Posons $H(\theta|\theta_{(m)}) = \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(Z|X;\theta)\big]$, on a :

$$L(X;\theta) = \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta)\big] - H(\theta|\theta_{(m)})$$

Donc,

$$L(X;\theta) - L(X;\theta_{(m)}) = \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta)\big] - \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta_{(m)})\big] + H(\theta_{(m)}|\theta_{(m)}) - H(\theta|\theta_{(m)})$$

Or,

$$\begin{aligned}
H(\theta_{(m)}|\theta_{(m)}) - H(\theta|\theta_{(m)}) &= \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(Z|X;\theta)\big] - \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(Z|X;\theta_{(m)})\big] && (5.13)\\
&= \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(Z|X;\theta) - L(Z|X;\theta_{(m)})\big] && (5.14)\\
&= \mathbb{E}_{Z|X,\theta_{(m)}}\Big[\ln\left(\frac{\mathbb{P}(Z|X;\theta)}{\mathbb{P}(Z|X;\theta_{(m)})}\right)\Big] && (5.15)\\
\text{Par Jensen,}\quad &\leqslant \ln\left(\mathbb{E}_{Z|X,\theta_{(m)}}\Big[\frac{\mathbb{P}(Z|X;\theta)}{\mathbb{P}(Z|X;\theta_{(m)})}\Big]\right) && (5.16)\\
&= \ln\left(\int \frac{\mathbb{P}(Z|X;\theta)}{\mathbb{P}(Z|X;\theta_{(m)})}\mathbb{P}(Z|X;\theta_{(m)})dZ\right) && (5.17)\\
&= \ln\left(\int \mathbb{P}(Z|X;\theta)dZ\right) && (5.18)\\
&= \ln(1) = 0 && (5.19)
\end{aligned}$$

Donc,

$$L(X;\theta) - L(X;\theta_{(m)}) \geqslant \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta)\big] - \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta_{(m)})\big] \tag{5.20}$$

Ce qui signifie que maximiser $L(X;\theta)$ revient à maximiser $\mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta)\big]$

Considérons la suite $(\theta_{(m)})_{m\in\mathbb{N}}$ vérifiant :

$$\theta_{m+1} = \underset{\theta}{argmax}\Big\{|\theta - \theta_{(m)}| - |\theta^* - \theta|\Big\}$$

où

$$\theta^* = \underset{\theta}{argmax} L(X;\theta) \quad \text{(estimateur idéal)}$$

Or maximiser $L(X;\theta)$ revient aussi à maximiser $\mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta)\big]$, donc on a :

$$\mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta_{(m+1)})\big] \geqslant \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta_{(m)})\big] \tag{5.21}$$

5.20 et 5.21 vérifie 5.12, on déduit alors :

$$\boxed{Q(\theta|\theta_{(m)}) := \mathbb{E}_{Z|X,\theta_{(m)}}\big[L(X,Z;\theta)\big]} \tag{5.22}$$

Or pour nous, les données cachées sont la provenance des individus : l'ignorance du groupe de chaque observation. On discrétise donc la variable $Z = (Z_1, \cdots, Z_S)$, il s'agit de la connaissance à postériori sur les données et le paramètre estimé, du groupe de chaque individu(pour chaque espèce dans notre cas). On déduit alors :

$$Q(\theta|\theta_{(m)}) = \sum_{s=1}^{S}\sum_{k=1}^{K}\mathbb{P}(Z_s = k|X_s,\theta_{(m)})L(X_s, Z_s = k;\theta) \tag{5.23}$$

Algorithme EM

Etape E :

c'est la phase *Expectation*, pour l'estimation des données cachées qui empêchent de maximiser Q, dans notre cas il s'agit des $P(Z_s = k|X, \theta_{(m)})$.On se sert donc des données et des précédentes estimations pour les estimer.

Etape M :

c'est la phase *maximisation*, pour la maximisation de Q rendue désormais possible par l'estimation des données cachées, ce qui permettra de mettre à jour la valeur du paramètre à estimer.

$$\theta_{(m+1)} = \underset{\theta}{\operatorname{argmax}}\ Q(\theta|\theta_{(m)}) \tag{5.24}$$

On réitère E-M jusqu'à stabilité de Q.

Dans la suite on pose $\mathbb{P}(Z_s = k|X, \theta_{(m)}) = p_{s,k,m}$

5.5 Inférence par l'algorithme EM des modèles de croissance et mortalité

Nous écrirons les étapes de l'algorithme pour la croissance.
Pour la suite, le terme binomiale $\binom{n_{s,j}}{g_{s,j}}$ sera négliger car il n'intervient pas dans la maximisation de la vraisemblance.

On se situe à la m^{e} itération, avec $\theta_{(m)} = (\pi_{1,(m)}, \cdots, \pi_{K,(m)}, \theta_{1,(m)} \cdots, \theta_{K,(m)})$, comme valeur courante de θ.

On a la log-vraisemblance :

$$L(X;\theta) = \sum_{s=1}^{S} \ln\Big(\sum_{k=1}^{K} \pi_k \prod_{j=1}^{n_s} \frac{\exp(x_{s,j}\theta_k)^{g_{s,j}}}{(1+\exp(x_{s,j}\theta_k))^{n_{s,j}}}\Big) \tag{5.25}$$

Ainsi,

$$Q(\theta|\theta_{(m)}) = \sum_{s=1}^{S}\sum_{k=1}^{K} p_{s,k,m}\Big[\ln \pi_k \ + \sum_{j=1}^{n_s} g_{s,j}x_{s,j}\theta_k \ - \ n_{s,j}\ln\left(1+\exp(x_{s,j}\theta_k)\right)\Big] \tag{5.26}$$

5.5.1 Étape E

Pour $s = 1, S$, pour $k = 1, K$, faire :

$$\begin{aligned} p_{s,k,m} = \mathbb{P}\{Z_s = k|X_s, \theta_{(m)}\} &= \frac{\mathbb{P}\{X_s, Z_s = k|\theta_{(m)}\}}{\mathbb{P}\{X_s|\theta_{(m)}\}} \\ &= \frac{\mathbb{P}\{X_s|Z_s = k, \theta_{(m)}\}\mathbb{P}\{Z_s = k\}}{\sum_{v=1}^{K}\mathbb{P}\{X_s|Z_s = v, \theta_{(m)}\}\mathbb{P}\{Z_s = v\}} \\ &= \frac{\mathbb{P}\{X_s|Z_s = k, \theta_{(m)}\}\pi_{k,(m)}}{\sum_{v=1}^{K}\mathbb{P}\{X_s|Z_s = v, \theta_{(m)}\}\pi_{v,(m)}} \end{aligned} \tag{5.27}$$

$$= \frac{\pi_{k,(m)}\prod_{j=1}^{n_s}\frac{\exp(x_{s,j}\theta_{k,(m)})^{g_{s,j}}}{(1+\exp(x_{s,j}\theta_{k,(m)}))^{n_{s,j}}}}{\sum_{v=1}^{K}\pi_{v,(m)}\prod_{j=1}^{n_s}\frac{\exp(x_{s,j}\theta_{v,(m)})^{g_{s,j}}}{(1+\exp(x_{s,j}\theta_{v,(m)}))^{n_{s,j}}}} \tag{5.28}$$

Avec le logiciel R, tout ces calculs se font dans une matrice $S \times K$ à l'aide de fonction précompilée *apply*.

5.5.2 Étape M

5.5.2.1 Détermination de $\pi_{k,(m+1)}$

La détermination de $\pi_{k,(m+1)}$ est astucieuse et se généralise à tous les problèmes d'estimation par l'algorithme EM.

L'astuce est de savoir que quelque soit le nombre de groupe, on peut toujours le voir comme 2 groupes lorqu'on fixe un groupe en particulier. Pour calculer la dérivée par rapport à π_k, nous écrivons la fonction à maximiser correspondant au partitionnement : "le groupe k" et "l'ensemble des groupes distincts de lui". On peut donc écrire :

$$Q(\theta|\theta_{(m)}) = \sum_{s}^{S} \Big\{ \mathbb{P}(Z_s = k|X, \theta_{(m)}) \left(\ln \pi_k + \ln(\mathbb{P}\{X|Z_s = k\})\right) + \mathbb{P}(Z_s \neq k|X, \theta_{(m)}) \left(\ln(\mathbb{P}\{Z_s \neq k\}) + \ln(\mathbb{P}\{X|Z_s \neq k\})\right) \Big\} \quad (5.29)$$

$$\Rightarrow Q(\theta|\theta_{(m)}) = \sum_{s}^{S} \Big\{ p_{s,k,m} \left(\ln \pi_k + \ln(\mathbb{P}\{X|Z_s = k\})\right) + (1 - p_{s,k,m}) \left(\ln(1 - \pi_k) + \ln(\mathbb{P}\{X|Z_s \neq k\})\right) \Big\} \quad (5.30)$$

Ainsi, on a les dérivées :

$$\frac{\partial Q(\theta|\theta_{(m)})}{\partial \pi_k} = \sum_{s=1}^{S} \frac{p_{s,k,m}}{\pi_k} - \frac{1 - p_{s,k,m}}{1 - \pi_k} \quad (5.31)$$

$$\frac{\partial^2 Q(\theta|\theta_{(m)})}{\partial \pi_k^2} = \sum_{s=1}^{S} -\frac{p_{s,k,m}}{\pi_k^2} - \frac{1 - p_{s,k,m}}{(1 - \pi_k)^2} < 0\,, \forall\, \pi_k \quad (5.32)$$

$\pi_{k,(m+1)}$ est alors donnée par :

$$\boxed{\pi_{k,(m+1)} = \frac{1}{S} \sum_{s=1}^{S} p_{s,k,m}} \quad (5.33)$$

L'estimateur de π_k est intuitivement ce à quoi on s'imagine, en effet, π_k est la probabilité qu'une espèce appartienne au groupe k, et son estimateur à chaque itération de l'algorithme est la moyenne des probabilités à postériori qu'une espèce soit du groupe k.

5.5.2.2 Détermination de $\theta_{k,(m+1)}$

On a :

$$\frac{\partial Q(\theta|\theta_{(m)})}{\partial \theta_k} = \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \left(g_{s,j}\, x_{s,j} - n_{s,j} \frac{\exp(x_{s,j}\theta_k)}{1 + \exp(x_{s,j}\theta_k)} x_{s,j} \right) \quad (5.34)$$

$$\frac{\partial^2 Q(\theta|\theta_{(m)})}{\partial \theta_k^2} = -\sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} n_{s,j} \frac{\exp(x_{s,j}\theta_k)}{(1 + \exp(x_{s,j}\theta_k))^2}\, x_{s,j}^t x_{s,j} \quad (5.35)$$

$$\text{Note :} x_{s,j} = (1, x_{s,j,1}, x_{s,j,2}, \cdots, x_{s,i,L}) \text{ et } x_{s,j}^t = \begin{pmatrix} 1 \\ x_{s,j,1} \\ x_{s,j,2} \\ \vdots \\ x_{s,j,L} \end{pmatrix}$$

Cette "dérivée seconde" est la matrice Hessienne pour la variable multidimensionnel θ_k, $x_{s,j}^t x_{s,j}$ est une matrice carrée symétrique $(L+1)\times(L+1)$. Remarquons que cette Hessienne n'est pas défini positive quelque soit θ_k, car en effet $x_{s,j}^t x_{s,j}$ est définit positive(preuve triviale) $\forall j \in [\![1,\, n_s]\!]$, et comme chaque coefficient pondérant cette matrice est négatif, on déduit que cette Hessienne n'est pas défini positive en tout point θ_k. Donc Q admet un maximum global. Nonobstant, il n'est pas possible de résoudre analytiquement l'équation $\dfrac{\partial Q(\theta|\theta_{(m)})}{\partial \theta_k} = 0_{\mathbb{R}^{L+1}}$, on se sert donc de méthode d'optimisation. On a :

$$\sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \left(g_{s,j}\, x_{s,j} - n_{s,j} \frac{\exp(x_{s,j}\theta_k)}{1+\exp(x_{s,j}\theta_k)} x_{s,j} \right)$$
$$= \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \left(g_{s,j} \; - n_{s,j} \frac{1}{1+\exp(-x_{s,j}\theta_k)} \right) x_{s,j}$$

On cherche $\theta_{k,(m+1)}$ de la manière suivante :

$$\boxed{\theta_{k,(m+1)} = \underset{u\in\mathbb{R}^{L+1}}{\operatorname{argmin}} \left\| \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \left(g_{s,j} \; - n_{s,j} \frac{1}{1+\exp(-x_{s,j}u)} \right) x_{s,j} \right\|} \tag{5.36}$$

Les fonctions *optim*() et *nlm*() de R, permettent de minimiser aisément cette quantité. Et avec les fonctions *sapply lapply* et *apply* on parvient à effectuer toutes ces opérations sur des matrices et des listes.

Remarque : Nous prenons l'approche vectorielle parcequ'elle permet de regrouper un ensemble d'opération, et aussi parce qu'elle s'adapte le mieux à notre outil d'implémentation.

5.5.3 Conclusion

En résumé, l'inférence par EM sur la croissance(mortalité en changeant $g_{s,j}$ par $m_{s,j}$) des arbres de chaque espèce est donné par :

modèle d'inférence par EM pour la croissance et la mortalité

Etape E :

$$p_{s,k,m} = \frac{\pi_{k,(m)} \prod_{j=1}^{n_s} \frac{\exp(x_{s,j}\theta_{k,(m)})^{g_{s,j}}}{(1+\exp(x_{s,j}\theta_{k,(m)}))^{n_{s,j}}}}{\sum_{v=1}^{K} \pi_{v,(m)} \prod_{j=1}^{n_s} \frac{\exp(x_{s,j}\theta_{v,(m)})^{g_{s,j}}}{(1+\exp(x_{s,j}\theta_{v,(m)}))^{n_{s,j}}}}$$

Etape M :

$$\pi_{k,(m+1)} = \frac{1}{S}\sum_{s=1}^{S} p_{s,k,m}$$

$$\theta_{k,(m+1)} = \underset{u\in\mathbb{R}^{L+1}}{\operatorname{argmin}} \left\| \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \left(g_{s,j} - n_{s,j} \frac{1}{1+\exp(-x_{s,j}u)} \right) x_{s,j} \right\|$$

$$\theta_{(m+1)} = (\pi_{1,(m+1)}, \cdots, \pi_{K,(m+1)}, \theta_{1,(m+1)} \cdots, \theta_{K,(m+1)})$$

E-M jusqu'à $|Q(\theta_{(m+1)}|\theta_{(m)}) - Q(\theta_{(m)}|\theta_{(m)})| \approx 0 (< 10^{-6})$**.**

5.6 Inférence par l'algorithme EM du modèle de recrutement

On a la log-vraisemblance

$$L(X;\theta) = \sum_{s=1}^{S} \ln\left(\sum_{k=1}^{K} \pi_k \prod_{j=1}^{n_s} \frac{\exp\left(r_{s,j}x_{s,j}\theta_k - \exp(x_{s,j}\theta_k)\right)}{r_{s,j}!}\right) \tag{5.37}$$

D'où

$$Q(\theta|\theta_{(m)}) = \sum_{s=1}^{S}\sum_{k=1}^{K} p_{s,k,m}\Big[\ln \pi_k \ + \sum_{j=1}^{n_s} r_{s,j}x_{s,j}\theta_k \ - \ \exp(x_{s,j}\theta_k) \ - \ \ln(r_{s,j}!)\Big] \tag{5.38}$$

5.6.1 Étape E

$$p_{s,k,m} = \frac{\mathbb{P}\{X_s|Z_s = k, \theta_{(m)}\}\pi_{k,(m)}}{\sum_{v=1}^{K}\mathbb{P}\{X_s|Z_s = v, \theta_{(m)}\}\pi_{v,(m)}} \tag{5.39}$$

$$= \frac{\pi_{k,(m)}\prod_{j=1}^{n_s}\frac{\exp\left(r_{s,j}x_{s,j}\theta_{k,m} - \exp(x_{s,j}\theta_{k,m})\right)}{r_{s,j}!}}{\sum_{v=1}^{K}\pi_{v,(m)}\prod_{j=1}^{n_s}\frac{\exp\left(r_{s,j}x_{s,j}\theta_{v,m} - \exp(x_{s,j}\theta_{v,m})\right)}{r_{s,j}!}} \tag{5.40}$$

5.6.2 Étape M

5.6.2.1 Détermination de $\pi_{k,m+1}$

La détermination de $\pi_{k,m+1}$ est identique que celle faite pour la croissance et la mortalité. On obtient absolument les mêmes dérivée. Donc,

$$\pi_{k,(m+1)} = \frac{1}{S}\sum_{s=1}^{S} p_{s,k,m}$$

5.6.2.2 Détermination de $\theta_{k,m+1}$

On a :

$$\frac{\partial Q(\theta|\theta_{(m)})}{\partial \theta_k} = \sum_{s=1}^{S} p_{s,k,m}\sum_{j=1}^{n_s}\left(r_{s,j}\, x_{s,j} - \exp(x_{s,j}\theta_k)x_{s,j}\right) \tag{5.41}$$

$$\frac{\partial^2 Q(\theta|\theta_{(m)})}{\partial \theta_k^2} = -\sum_{s=1}^{S} p_{s,k,m}\sum_{j=1}^{n_s} x_{s,j}^t x_{s,j}\ \exp(x_{s,j}\theta_k) \tag{5.42}$$

On obtient également une dérivée seconde non définie positive.
l'équation du gradient de Q est aussi non linéaire et se résout par optimisation. Ici, il n'y a pas de transformation à faire, l'équation est assez simple, on déduit alors :

$$\boxed{\theta_{k,m+1} = \underset{u\in\mathbb{R}^{L+1}}{argmin} \left\| \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \left(r_{s,j} - \exp(x_{s,j}u)\right) x_{s,j} \right\|} \tag{5.43}$$

5.6.3 Conclusion

modèle d'inférence par EM pour le recrutement

Etape E :

$$p_{s,k,m} = \frac{\pi_{k,(m)} \prod_{j=1}^{n_s} \frac{\exp\left(r_{s,j}x_{s,j}\theta_{k,m} - \exp(x_{s,j}\theta_{k,m})\right)}{r_{s,j}!}}{\sum_{v=1}^{K} \pi_{v,(m)} \prod_{j=1}^{n_s} \frac{\exp\left(r_{s,j}x_{s,j}\theta_{v,m} - \exp(x_{s,j}\theta_{v,m})\right)}{r_{s,j}!}}$$

Etape M :

$$\pi_{k,(m+1)} = \frac{1}{S} \sum_{s=1}^{S} p_{s,k,m}$$

$$\theta_{k,m+1} = \underset{u\in\mathbb{R}^{L+1}}{argmin} \left\| \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \left(r_{s,j} - \exp(x_{s,j}u)\right) x_{s,j} \right\|$$

$$\theta_{(m+1)} = (\pi_{1,(m+1)}, \cdots, \pi_{K,(m+1)}, \theta_{1,(m+1)} \cdots, \theta_{K,(m+1)})$$

E-M jusqu'à $|Q(\theta_{(m+1)}|\theta_{(m)}) - Q(\theta_{(m)}|\theta_{(m)})| \approx 0 (< 10^{-6})$**.**

Chapitre 6

Modélisation pour les intervalles de temps irréguliers entre les inventaires

Le principe reste le même, mais l'utilisation des données de variables explicatives change. Nous allons montrer le pourquoi et présenter les aboutissants de la fonction Q à maximiser.

6.1 Cas de la mortalité et la croissance

On introduit le paramètre d, dont chaque valeur désigne pour une ligne la distance(en année) qui s'est écoulée entre les deux inventaires de données observées($init$ et end). Le tableau de données prend donc la forme suivante :

TABLE 6.1 – Tableau de données aux inventaires irréguliers : cas de la croissance

Y		X							
g	n	Esp	$X_1^{(init)}$	$\cdots$	$X_L^{(init)}$	$X_1^{(end)}$	$\cdots$	$X_L^{(init)}$	d
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$g_{s,i}$	$n_{s,i}$	s	$x_{s,i,1}^{(init)}$	$\cdots$	$x_{s,i,L}^{(init)}$	$x_{s,i,1}^{(end)}$	$\cdots$	$x_{s,i,L}^{(end)}$	$d_{s,i}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

NB : Si $d_{s,i} = 1$ alors les données de variables explicatives en ($init$) et (end) sont identiques.

La mortalité et la croissance dans notre contexte de forêts tropicales, ont un comportement tout à fait similaire, la croissance a un comportement d'état absorbant car dans une classe, on la considère similaire à la mort. les classes de diamètre étant de plage d'environ 10 cm, même en cinq ans ou plus il n'est pas réaliste d'envisager qu'un arbre ait changé deux fois de suite sa classe de diamètre. La croissance est un évènement rare.
Le modèle à intervalles de temps réguliers(d'un an) entre les inventaires représente pour nous le modèle idéale. C'est par le modèle régulier que nous définissons la croissance pour un arbre en fonction des surfaces terrières. L'ignorance des paramètres issus d'une régularité entre les inventaires doit donc s'exprimer par rapport aux paramètres d'inventaires réguliers. Ainsi, il

nous faut trouver le lien entre la probabilité de croissance d'inventaires distants de d années en fonction des probabilités d'inventaires des années non observées. On a le schéma suivant :

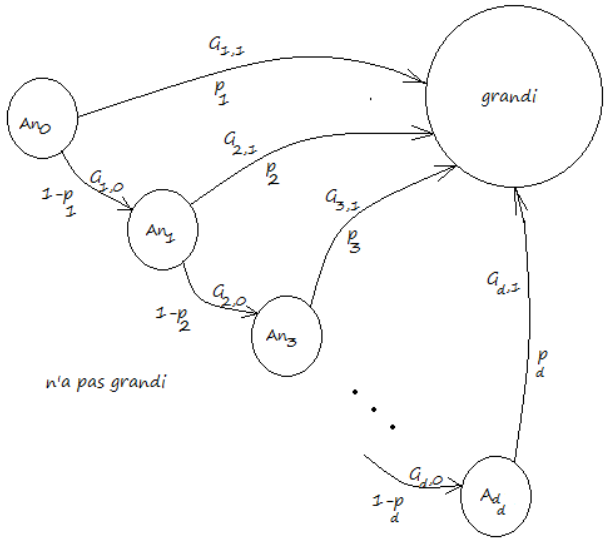

FIGURE 6.1 – Automate de croissance ou de mortalité pour d années

Dans cette figure, $G_{i,1}$: est l'évènement : " grandi l'année i" de probabilité p_i et $G_{i,0}$: "n'a pas grandir l'année i". Ainsi la formule du calcul de la probabilité de grandir sur n années est celle d'une réunion d'évènements, et on peut la calculée par la formule de Poincaré :

$$\begin{aligned}
\mathbb{P}\{G_{1,1} \cup G_{2,1} \cup \cdots \cup G_{d,1}\} &= \sum_{k=1}^{d} (-1)^{k-1} \sum_{1 \leqslant i_1 < \cdots < i_k \leqslant d} p_{i_1} p_{i_2} \cdots p_{i_k} && (6.1)\\
&= \sum_{i=1}^{d} p_i \prod_{j<i} (1-p_j) && (6.2)\\
&= 1 - \prod_{i=1}^{d} (1-p_i) && (6.3)
\end{aligned}$$

Par exemple pour $d = 2$, on a : $p = p_1 + p_2 - p_1 p_2 = p_1 + (1-p_1)p_2 = 1 - (1-p_1)(1-p_2)$

Cette relation va permettre de dupliquer la formule de la vraisemblance, on considèrera de nouvelles données issues de l'interpolation pour chaque ligne. Pour une ligne i d'espèce s, on notera $x_{s,i}^{(1)}, \cdots, x_{s,i}^{(d_{s,i}-1)}$ les $d_{s,i} - 1$ données complétées par interpolation, et on pose $x_{s,i}^{(d_{s,i})} = x_{s,i}^{end}$. Pour tenir compte de ces valeurs, on part de la formule 5.2.
Conditionnellement à l'appartenance au groupe k, à la ligne i la vraisemblance s'écrit alors :

$$P(x_{s,i}|\theta_k) = p_{s,i,\theta_k}^{g_{s,i}} (1 - p_{s,i,\theta_k})^{n_{s,i} - g_{s,i}}$$

Sachant que p_{s,i,θ_k} correspond à une probabilité de d années, en utilisant 6.3, on a :

$$P(x_{s,i}|\theta_k) = \left(1 - \prod_{t=1}^{d_{s,i}}(1 - p_{s,i,\theta_k}^{(t)})\right)^{g_{s,i}} \left(\prod_{t=1}^{d_{s,i}}(1 - p_{s,i,\theta_k}^{(t)})\right)^{n_{s,i}-g_{s,i}}$$

où $p_{s,i,\theta_k}^{(t)} = \dfrac{\exp(x_{s,i}^{(t)}\theta_k)}{1+\exp(x_{s,i}^{(t)}\theta_k)}$ Ainsi, la vraisemblance s'écrit :

$$P(X;\theta) = \prod_{s=1}^{S}\sum_{k=1}^{K}\pi_k \prod_{j=1}^{n_s}\left(1 - \prod_{t=1}^{d_{s,j}}\frac{1}{1+\exp(x_{s,j}^{(t)}\theta_k)}\right)^{g_{s,j}} \left(\prod_{t=1}^{d_{s,j}}\frac{1}{1+\exp(x_{s,j}^{(t)}\theta_k)}\right)^{n_{s,j}-g_{s,j}} \tag{6.4}$$

D'où la log-vraisemblance

$$L(X;\theta) = \sum_{s=1}^{S}\ln\left\{\sum_{k=1}^{K}\pi_k \prod_{j=1}^{n_s}\left(1 - \prod_{t=1}^{d_{s,j}}\frac{1}{1+\exp(x_{s,j}^{(t)}\theta_k)}\right)^{g_{s,j}} \left(\prod_{t=1}^{d_{s,j}}\frac{1}{1+\exp(x_{s,j}^{(t)}\theta_k)}\right)^{n_{s,j}-g_{s,j}}\right\} \tag{6.5}$$

Et la fonction à maximiser est

$$Q(\theta|\theta_{(m)}) = \sum_{s=1}^{S}\sum_{k=1}^{K} p_{s,k,m}\left[\ln\pi_k + \sum_{j=1}^{n_s} g_{s,j}\ln\left(1 - \prod_{t=1}^{d_{s,j}}\frac{1}{1+\exp(x_{s,j}^{(t)}\theta_k)}\right) - (n_{s,j}-g_{s,j})\sum_{t=1}^{d_{s,j}}\ln(1+\exp(x_{s,j}^{(t)}\theta_k))\right] \tag{6.6}$$

Remarque : Plus généralement utilisera la notation en exposant $(.)^{(t)}$ pour indicer les paramètres et les variables entre $(init)$ et (end)

6.1.1 Étape E

On a

$$\begin{aligned} p_{s,k,m} &= \frac{\mathbb{P}\{X_s|Z_s=k,\theta_{(m)}\}\pi_{k,(m)}}{\sum_{v=1}^{K}\mathbb{P}\{X_s|Z_s=v,\theta_{(m)}\}\pi_{v,(m)}} \\ &= \frac{\pi_{k,m}\prod_{j=1}^{n_s}\left(1-\prod_{t=1}^{d_{s,j}}\frac{1}{1+\exp(x_{s,j}^{(t)}\theta_{k,m})}\right)^{g_{s,j}}\left(\prod_{t=1}^{d_{s,j}}\frac{1}{1+\exp(x_{s,j}^{(t)}\theta_{k,m})}\right)^{n_{s,j}-g_{s,j}}}{\sum_{v=1}^{K}\pi_{v,m}\prod_{j=1}^{n_s}\left(1-\prod_{t=1}^{d_{s,j}}\frac{1}{1+\exp(x_{s,j}^{(t)}\theta_{v,m})}\right)^{g_{s,j}}\left(\prod_{t=1}^{d_{s,j}}\frac{1}{1+\exp(x_{s,j}^{(t)}\theta_{v,m})}\right)^{n_{s,j}-g_{s,j}}} \end{aligned} \tag{6.7}$$

6.1.2 Étape M

La détermination de $\pi_{k,m+1}$ est inchangée, nous nous intéressons à la recherche de $\theta_{k,m+1}$. On a :

$$\frac{\partial Q(\theta|\theta_{(m)})}{\partial \theta_k} = \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} g_{s,j} \frac{\partial}{\partial \theta_k} \ln\left(1 - \prod_{t=1}^{d_{s,j}} \frac{1}{1+\exp(x_{s,j}^{(t)}\theta_k)}\right) - (n_{s,j} - g_{s,j}) \sum_{t=1}^{d_{s,j}} \frac{x_{s,j}^{(t)} \exp(x_{s,j}^{(t)}\theta_k)}{1+\exp(x_{s,j}^{(t)}\theta_k)} \tag{6.8}$$

Or,

$$\frac{\partial}{\partial \theta_k} \ln\left(1 - \prod_{t=1}^{d_{s,j}} \frac{1}{1+\exp(x_{s,j}^{(t)}\theta_k)}\right) = \frac{\sum_{t=1}^{d_{s,j}} \frac{x_{s,j}^{(t)} \exp(x_{s,j}^{(t)}\theta_k)}{1+\exp(x_{s,j}^{(t)}\theta_k)}}{\prod_{t=1}^{d_{s,j}}(1+\exp(x_{s,j}^{(t)}\theta_k)) - 1} \tag{6.9}$$

On remarque un facteur commun, ce qui permet d'écrire après réduction

$$\frac{\partial Q(\theta|\theta_{(m)})}{\partial \theta_k} = \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \sum_{t=1}^{d_{s,j}} \Big(\frac{n_{s,j}}{\prod_{l=1}^{d_{s,j}}(1+\exp(x_{s,j}^{(l)}\theta_k)) - 1} - (n_{s,j} - g_{s,j})\Big)\Big(\frac{1}{1+\exp(-x_{s,j}^{(t)}\theta_k)}\Big) x_{s,j}^{(t)} \tag{6.10}$$

Contrairement au cas régulier, nous ne pouvons nous prononcer avec certitude quand à l'existence d'un maximum global. Toutefois, nous cherchons $\theta_{k,m+1}$ qui maximise la vraisemblance et qui vérifie :

$$\boxed{\theta_{k,m+1} = \underset{u\in\mathbb{R}^{L+1}}{argumin} \left\| \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \sum_{t=1}^{d_{s,j}} \Big(\frac{n_{s,j}}{\prod_{l=1}^{d_{s,j}}(1+\exp(x_{s,j}^{(l)}u)) - 1} - (n_{s,j} - g_{s,j})\Big)\Big(\frac{1}{1+\exp(-x_{s,j}^{(t)}u)}\Big) x_{s,j}^{(t)} \right\|} \tag{6.11}$$

Remarque : Nous écrivons toujours l'expression de la dérivée(gradient) de Q en scindant la partie réel de la partie vectorielle dans le but de pouvoir faire le calcul sans répétition et plus simplifier l'implémentation.

6.1.3 Conclusion

Ainsi, Dans le cas général on peut écrire pour la croissance(ou la mortalité) :

modèle d'inférence par EM pour croissance et la mortalité dans le cas générale

Etape E :

$$p_{s,k,m} = \frac{\pi_{k,m} \prod_{j=1}^{n_s} \left(1 - \prod_{t=1}^{d_{s,j}} \frac{1}{1+\exp(x_{s,j}^{(t)}\theta_{k,m})}\right)^{g_{s,j}} \left(\prod_{t=1}^{d_{s,j}} \frac{1}{1+\exp(x_{s,j}^{(t)}\theta_{k,m})}\right)^{n_{s,j}-g_{s,j}}}{\sum_{v=1}^{K} \pi_{v,m} \prod_{j=1}^{n_s} \left(1 - \prod_{t=1}^{d_{s,j}} \frac{1}{1+\exp(x_{s,j}^{(t)}\theta_{v,m})}\right)^{g_{s,j}} \left(\prod_{t=1}^{d_{s,j}} \frac{1}{1+\exp(x_{s,j}^{(t)}\theta_{v,m})}\right)^{n_{s,j}-g_{s,j}}}$$

Etape M :

$$\pi_{k,(m+1)} = \frac{1}{S} \sum_{s=1}^{S} p_{s,k,m}$$

$$\theta_{k,m+1} = \underset{u \in \mathbb{R}^{L+1}}{argumin} \Bigg\| \sum_{s=1}^{S} \sum_{k=1}^{K} p_{s,k,m} \sum_{j=1}^{n_s} \sum_{t=1}^{d_{s,j}} \Big(\frac{n_{s,j}}{\prod_{l=1}^{d_{s,j}} (1+\exp(x_{s,j}^{(l)} u)) - 1} - (n_{s,j} - g_{s,j}) \Big) \Big(\frac{1}{1+\exp(-x_{s,j}^{(t)} u)} \Big) x_{s,j}^{(t)} \Bigg\|$$

$$\theta_{(m+1)} = (\pi_{1,(m+1)}, \cdots, \pi_{K,(m+1)}, \theta_{1,(m+1)} \cdots, \theta_{K,(m+1)})$$

E-M jusqu'à $|Q(\theta_{(m+1)}|\theta_{(m)}) - Q(\theta_{(m)}|\theta_{(m)})| \approx 0 (< 10^{-6})$**.**

6.2 Cas du recrutement

TABLE 6.2 – Tableau de données aux inventaires irréguliers : cas du recrutement

Y	X							
r	Esp	$X_1^{(init)}$	$\cdots$	$X_L^{(init)}$	$X_1^{(end)}$	$\cdots$	$X_L^{(end)}$	d
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$r_{s,i}$	s	$x_{s,i,1}^{(init)}$	$\cdots$	$x_{s,i,L}^{(init)}$	$x_{s,i,1}^{(end)}$	$\cdots$	$x_{s,i,L}^{(end)}$	$d_{s,i}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Comme dans le cas précédant, on cherche à exprimer le paramètre de la loi de Poisson après d années, en fonction des paramètres des années intermédiaires. Nous partons du fait que le paramètre de la loi de Poisson est aussi son espérance et puisque nous sommes dans une situation de somme de telles lois(qui est encore une loi de Poisson), on déduit alors que le paramètre λ correspondant vérifie :

$$\begin{aligned}\lambda &= \mathbb{E}(\mathcal{P}(\lambda)) \\ &= \mathbb{E}(\mathcal{P}(\lambda_1) + \mathcal{P}(\lambda_2) + \cdots + \mathcal{P}(\lambda_d)) \\ &= \mathbb{E}(\mathcal{P}(\lambda_1)) + \mathbb{E}(\mathcal{P}(\lambda_2)) + \cdots + \mathbb{E}(\mathcal{P}(\lambda_d)) \\ &= \lambda_1 + \lambda_2 + \cdots + \lambda_d \end{aligned} \tag{6.12}$$

Dans le cas régulier, on a : $\lambda_{\theta_k,s,i} = \exp(x_{s,i}\theta_k)$, par 6.12, on déduit :

$$\lambda_{\theta_k,s,i} = \sum_{t=1}^{d_{s,i}} \lambda_{\theta_k,s,i}^{(t)} = \sum_{t=1}^{d_{s,i}} \exp(x_{s,i}^{(t)}\theta_k) \tag{6.13}$$

Ainsi, partant de 5.8, conditionnellement à l'appartenance au groupe k, on a la vraisemblance suivante pour l'espèce s :

$$P(x_s|\theta_k) = \prod_{j=1}^{n_s} \frac{\left(\sum_{t=1}^{d_{s,j}} \lambda_{\theta_k,s,j}^{(t)}\right)^{r_{s,j}}}{r_{s,j}!} \exp\left(-\sum_{t=1}^{d_{s,j}} \lambda_{\theta_k,s,j}^{(t)}\right) \tag{6.14}$$

$$= \prod_{j=1}^{n_s} \frac{\left(\sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_k)\right)^{r_{s,j}}}{r_{s,j}!} \exp\left(-\sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_k)\right) \tag{6.15}$$

On déduit la log-vraisemblance et la fonction Q à maximiser suivantes :

$$L(X;\theta) = \sum_{s=1}^{S} \ln\left(\sum_{k=1}^{K} \pi_k \prod_{j=1}^{n_s} \frac{\left(\sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_k)\right)^{r_{s,j}}}{r_{s,j}!} \exp\left(-\sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_k)\right)\right) \tag{6.16}$$

$$Q(\theta|\theta_{(m)}) = \sum_{s=1}^{S}\sum_{k=1}^{K} p_{s,k,m}\left[\ln \pi_k + \sum_{j=1}^{n_s} r_{s,j} \ln\left(\sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_k)\right) - \sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_k) - \ln(r_{s,j}!)\right] \tag{6.17}$$

6.2.1 Étape E

On a :

$$\begin{aligned} p_{s,k,m} &= \frac{\mathbb{P}\{X_s|Z_s=k,\theta_{(m)}\}\pi_{k,(m)}}{\sum_{v=1}^{K}\mathbb{P}\{X_s|Z_s=v,\theta_{(m)}\}\pi_{v,(m)}} \\ &= \frac{\pi_{k,(m)}\prod_{j=1}^{n_s}\frac{\left(\sum_{t=1}^{d_{s,j}}\exp(x_{s,j}^{(t)}\theta_k)\right)^{r_{s,j}}}{r_{s,j}!}\exp\left(-\sum_{t=1}^{d_{s,j}}\exp(x_{s,j}^{(t)}\theta_k)\right)}{\sum_{v=1}^{K}\pi_{v,(m)}\prod_{j=1}^{n_s}\frac{\left(\sum_{t=1}^{d_{s,j}}\exp(x_{s,j}^{(t)}\theta_v)\right)^{r_{s,j}}}{r_{s,j}!}\exp\left(-\sum_{t=1}^{d_{s,j}}\exp(x_{s,j}^{(t)}\theta_v)\right)} \end{aligned} \tag{6.18}$$

6.2.2 Étape M

$$\frac{\partial Q(\theta|\theta_{(m)})}{\partial\theta_k} = \sum_{s=1}^{S}p_{s,k,m}\sum_{j=1}^{n_s}\frac{r_{s,j}\sum_{t=1}^{d_{s,j}}x_{s,j}\exp(x_{s,j}^{(t)}\theta_k)}{\sum_{t=1}^{d_{s,j}}\exp(x_{s,j}^{(t)}\theta_k)} - \sum_{t=1}^{d_{s,j}}x_{s,j}\exp(x_{s,j}^{(t)}\theta_k) \tag{6.19}$$

$$= \sum_{s=1}^{S}p_{s,k,m}\sum_{j=1}^{n_s}\sum_{t=1}^{d_{s,j}}\left[\left(\frac{r_{s,j}}{\sum_{l=1}^{d_{s,j}}\exp(x_{s,j}^{(l)}\theta_k)}-1\right)\exp(x_{s,j}^{(t)}\theta_k)\right]x_{s,j}^{(t)} \tag{6.20}$$

D'où

$$\boxed{\theta_{k,m+1} = \underset{u\in\mathbb{R}^{L+1}}{argumin}\left\|\sum_{s=1}^{S}p_{s,k,m}\sum_{j=1}^{n_s}\sum_{t=1}^{d_{s,j}}\left[\left(\frac{r_{s,j}}{\sum_{l=1}^{d_{s,j}}\exp(x_{s,j}^{(l)}u)}-1\right)\exp(x_{s,j}^{(t)}u)\right]x_{s,j}^{(t)}\right\|} \tag{6.21}$$

6.2.3 Conclusion

modèle d'inférence par EM pour le recrutement dans le cas générale

Etape E :

$$p_{s,k,m} = \frac{\pi_{k,(m)} \prod_{j=1}^{n_s} \frac{\left(\sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_k)\right)^{r_{s,j}}}{r_{s,j}!} \exp\left(-\sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_k)\right)}{\sum_{v=1}^{K} \pi_{v,(m)} \prod_{j=1}^{n_s} \frac{\left(\sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_v)\right)^{r_{s,j}}}{r_{s,j}!} \exp\left(-\sum_{t=1}^{d_{s,j}} \exp(x_{s,j}^{(t)}\theta_v)\right)}$$

Etape M :

$$\pi_{k,(m+1)} = \frac{1}{S} \sum_{s=1}^{S} p_{s,k,m}$$

$$\theta_{k,m+1} = \underset{u \in \mathbb{R}^{L+1}}{argumin} \left\| \sum_{s=1}^{S} p_{s,k,m} \sum_{j=1}^{n_s} \sum_{t=1}^{d_{s,j}} \left[\left(\frac{r_{s,j}}{\sum_{l=1}^{d_{s,j}} \exp(x_{s,j}^{(l)} u)} - 1 \right) \exp(x_{s,j}^{(t)} u) \right] x_{s,j}^{(t)} \right\|$$

$$\theta_{(m+1)} = (\pi_{1,(m+1)}, \cdots, \pi_{K,(m+1)}, \theta_{1,(m+1)} \cdots, \theta_{K,(m+1)})$$

E-M jusqu'à $|Q(\theta_{(m+1)}|\theta_{(m)}) - Q(\theta_{(m)}|\theta_{(m)})| \approx 0 (< 10^{-6})$**.**

6.3 Conclusion

Ce modèle de recrutement dans le cas générale, achève ainsi notre modélisation. Mais avant, il faut noter que le principe de regroupement est le MAP (maximum à postériori) qui consiste à attribuer un groupe à une espèce si et seulement si après convergence de l'algorithme, sa probabilité à postériori (p_{s,k,m^*}) sur les données d'appartenir à ce groupe est maximale. En d'autres termes, étant donnée une espèce s, on a :

$$\text{groupe}(s) = \underset{k \in [\![1, K]\!]}{argmax}\, p_{s,k,m^*} \tag{6.22}$$

m^* étant l'itération de convergence.
Nous connaissons à présent quel algorithme appliquer sur les données, mais le cas où les inventaires sont distants de manière irrégulières fait apparaître de nouvelles données dont nous nous proposons de présenter la source dans le chapitre suivant.

Chapitre 7

Interpolation des données de variables explicatives

Dans la pratique, il n'est pas toujours possible de faire des collectes de données régulièrement. Car il suffit d'une intempérie ou de l'arrivé d'une situation indésirable qui endommage la procédure de collecte de données pour qu'aucun prélèvement ne se face ou dans le meilleur cas se face mal. Il semble donc intéressant de se donner une idée des valeurs de données manquantes afin d'appliquer le modèle sur des données qui respectent le format des entrées. Les valeurs de surface terrière étant quantitatives et monotone sur un intervalle de temps, après les avoir observé, on se rend compte qu'il n'est pas absurde de supposer une courbe linéaire donnant ces valeurs. C'est ce qui nous conduit au choix de l'interpolation linéaire pour combler le manque.

7.1 Présentation de la méthode d'interpolation (Inspiré de l'interpolation par noyau radial (RBF))

Il s'agit de trouver une relation entre les inventaires d'une même parcelle distant d'une distance d (en années). Pour une donnée absente, l'idée d'approximation de sa valeur est de pondérer les données présentes qui l'entourent en attribuant plus d'importance à la donnée qui lui est plus proche [26]. Nous entendons par données, le vecteur x de taille $L+1$ dont les composantes de 2 à $L+1$ représentent des valeurs de surface terrière.

On garde les notation du chapitre précédent : $x_{s,i}^{(t)}$ représente la t^{e} donnée manquante avec $t \in [\![1,\, d_{s,i}-1]\!]$ et $x_{s,i}^{(d_{s,i})} = x_{s,i}^{(end)}$ et $x_{s,i}^{(0)} = x_{s,i}^{(init)}$.
On veut se servir des données observées $x_{s,i}^{(init)}$ et $x_{s,i}^{(end)}$ pour approximer la valeur de $x_{s,i}^{(t)}$, $t \in [\![1,\, d_{s,i}-1]\!]$. Il s'agit de trouver une relation de la forme :

$$x_{s,i}^{(t)} = \omega_{(init)}(t) x_{s,i}^{(init)} + \omega_{(end)}(t) x_{s,i}^{(end)} \tag{7.1}$$

où les poids ω dépendent de l'inventaire manquante et la durée entre les inventaires observés. Il s'agit d'écrire la donnée $x_{s,i}^{(t)}$ comme barycentre des données $x_{s,i}^{(init)}$ et $x_{s,i}^{(end)}$ de telle enseigne

que les poids soient inversement proportionnelles à leurs distances à $x_{s,i}^{(t)}$. On choisit les poids de la manière suivante :

$$\omega_T(t) = \frac{\rho(|t-T|)}{\rho(t) + \rho(|T-t|)}$$

$T \in \{(init), (end)\}$, ρ est une fonction positive et décroissante des distances à t.
Nous choisissons $\rho(a) = d_{s,i} - a$. Ce choix nous permet de rejoindre la méthode d'interpolation linéaire. En fonction du comportement des données, on peut être amené à le définir autrement.

7.1.1 Exemple

Supposons que $d_{s,i} = 3$, et que $(init)$ corresponde aux données de 1988 et (end) à 1991. en appliquant 7.1 à 1989 $(t = 1)$ avec $\rho(a) = d_{s,i} - a$, on obtient :

$$\begin{cases} \rho(|(init) - t|) = \rho(|0 - 1|) = 3 - 1 = 2 \\ \rho(|(end) - t|) = \rho(|d_{s,i} - 1|) = 3 - (3 - 1) = 1 \\ \rho(t) = 3 - 1 = 2 \end{cases}$$

d'où

$$\begin{aligned} x_{s,i}^{(1)} &= \frac{3-1}{(3-1)+(3-2)} x_{s,i}^{(init)} + \frac{3-2}{(3-1)+(3-2)} x_{s,i}^{(end)} \\ &= \frac{2}{3} x_{s,i}^{(init)} + \frac{1}{3} x_{s,i}^{(end)} \end{aligned}$$

On peut remarquer que ce résultat est très intuitif. En effet, le poids pour un point non observé est définit comme étant le rapport entre l'écart complémentaire au point non observé et la valeur manquante, à la distance séparant les deux points observés ($(init)$ et end).

7.2 Formule de complétion des données

Plus généralement, comme on peut le remarquer dans l'exemple précédent, avec $\rho(a) = d_{s,i} - a$, on a la formule suivante :

$$\boxed{x_{s,i}^{(t)} = \frac{d_{s,i} - t}{d_{s,i}} x_{s,i}^{(init)} + \frac{t}{d_{s,i}} x_{s,i}^{(end)}} \tag{7.2}$$

7.3 Conclusion

Ce chapitre nous a présenté une méthode générale d'approximation des valeurs manquantes. Nous avons pu constater que celle ci dépend de la définition des poids de pondération des observations. Pour notre cas nous avons choisit comme poids le rapport entre l'écart complémentaire du point observé et la valeur manquante sur la distance séparant les deux points observés ($(init)$ et (end)), sur cette dernière. Par ailleurs, il existe d'autres moyens de définition des poids qui peuvent être liés directement à l'expérience, celles ci reposent entièrement sur la détermination de la fonction ρ qui permet d'avoir les poids de

pondération. On trouve par exemple dans la littérature [26], des méthodes permettant de la déterminer par une approche stochastique ou par considération de la fonction produisant les valeurs observées.

Chapitre 8

Simulation des données

Le modèle étant bien conçu, il faut à présent le tester. Cela demande à avoir des données dont on connais (ou on a une bonne estimation) les vrais paramètres (θ) afin de pouvoir mesurer l'incertitude ou la qualité du modèle. Or pour les problèmes complexes, faire les tests de cette façon n'est pas généralement possible. On fait donc une simulation des données. Celle ci se base sur le modèle conçu et permet, en utilisant l'aléa de produire des données dont on peut retrouver les paramètres du générateur avec l'utilisation du modèle. La simulation est donc importante car pour les systèmes complexe, c'est elle qui permet de valider les modèles. Ce chapitre présente la simulation des données pour le processus de croissance (identique à celui de la mortalité) tout en présentant l'approche pour le processus de recrutement dont le principe est le même.

8.1 Simulation des données pour le cas d'inventaires réguliers :

La génération du tableau de donnée se fait de la manière suivante :

On se donne K vecteurs θ_k de dimension $L+1$. Et on reparti de manière aléatoire un nombre S d'espèces dans les groupes : par exemple une partition de l'ensemble $\{1, \cdots, S\}$.

Principe [2] Il s'agit en fait, de faire des réalisations d'une loi mélange. On l'interprète comme la loi marginale de la variable aléatoire Y du couple de variables aléatoires (Y, Z) telle que :

$$Z \rightsquigarrow \mathcal{M}(1, \pi_1, \cdots, \pi_K) \quad \text{et} \quad Y|Z = z \rightsquigarrow \{\mathcal{B}(n, \frac{\exp((1,x)\theta_k)}{1+\exp((1,x)\theta_k)})\}_{z_k=1} \tag{8.1}$$

$z = (z_1, \cdots, z_K)$ étant un vecteur binaire de dimension K, n est choisit de manière aléatoire. $\mathcal{M}(1, \pi_1, \cdots, \pi_K)$ est la loi multinomiale de dimension K et d'ordre 1 et de paramètre $(\pi_1, \cdots, \pi_K)$. x est un vecteur de taille L que l'on choisit de manière aléatoire. L'expression : $\frac{\exp((1,x)\theta_k)}{1+\exp((1,x)\theta_k)}$ (obtenu par l'inverse de la fonction logit) fourni la probabilité permettant

de faire la réalisation binomiale pour une ligne ayant pour données de variables explicatives le vecteur x.

On déduit donc l'algorithme de simulation suivant :

A- Générer un échantillon $z = (z_1, \cdots, z_n)$ tel que les z_i soient une réalisation $i.i.d$ de Z.

B- Sélectionner les espèces correspondant à l'échantillon z, et prendre aléatoirement une espèce dans chaque groupe.

C- Générer un échantillon $y = (y_1, \cdots, y_n)$ tel que les y_i soient des réalisations indépendantes de $\{\mathcal{B}(n, \frac{\exp((1,x)\theta_k)}{1+\exp((1,x)\theta_k)})\}_{z_k=1}$. Ce qui permet d'avoir le x des variables explicatives, le n correspondant au nombre d'arbres observés et le nombre de succès correspond au nombre de grandi(croissance), ou de mort (mortalité).

D- Répéter les étapes A-B-C jusqu'au nombre de données désiré.

Il est important de noter que y_i est généré conditionnellement à z_i.

Le même principe s'applique pour le recrutement, juste en changeant la densité de la variable $Y|Z$ par une loi log-linéaire.

8.2 Cas des inventaires irréguliers

On refait identiquement la même chose que pour le cas régulier mais cette fois ci suivant un paramètre d tel que :

- Si pour une ligne on a $d > 1$ on simule deux valeurs du vecteur x des variables explicatives.
- Si $d = 1$, on simule une seule valeur de x.

Le d est considéré comme la durée entre les deux valeurs simulées. d peut être pris aléatoirement dans $[\![1, 3]\!]$.

Troisième partie

Application et résultats

Chapitre 9

Résultats

On simule avec l'algorithme de simulation en annexe, un tableau de 100 lignes de données de 30 espèces (représentés par les entiers de 1 à 30) issues de trois groupes distincts dont les paramètres theta (représente les θ_k dans le modèle) et Pi (représente les π_k du modèle) sont générés de manière aléatoire et sont fixés. Notre algorithme EM donne une estimation de ces paramètres en EM_Pi et EM_theta respectivement. On a pris le cas $L = 3$, donc les θ_k sont des éléments de $\mathbb{R}^4$. Selon la simulation, le regroupement idéale serait d'avoir les espèces de 1 à 10 dans le même groupe, de 11 à 20 dans le même groupe et de 21 à 30 dans le même groupe, les trois groupes étant distincts.

9.1 Résultats de notre algorithme

Après 3 minutes du lancé de l'algorithme, on a eu les résultats suivants :

TABLE 9.1 – Estimation des paramètres de régression pour un mélange de croissance (ou mortalité) de 30 espèces provenant de 3 groupes sur 100 lignes de données avec 3 variables explicatives

valeurs réelles			valeurs estimées		
θ_1	θ_2	θ_3	θ_1	θ_2	θ_3
0.2732180	0.6715684	-0.5759935	0.4011666	0.5980205	-0.6808234
-0.1537600	-0.5228565	-0.7054100	-0.1596489	-0.4714019	-0.7277253
-0.09930405	0.12545283	1.84788650	0.04924739	0.12193853	2.20622956
0.4325952	0.9186994	1.5026038	0.5024789	0.9135581	1.4450963

TABLE 9.2 – Estimation des proportions pour un mélange de croissance (ou mortalité) de 30 espèces provenant de 3 groupes sur 100 lignes de données avec 3 variables explicatives

valeurs réelles			valeurs estimées		
π_1	π_2	π_3	π_1	π_2	π_3
0.41	0.35	0.24	0.4683175	0.3020540	0.2296285

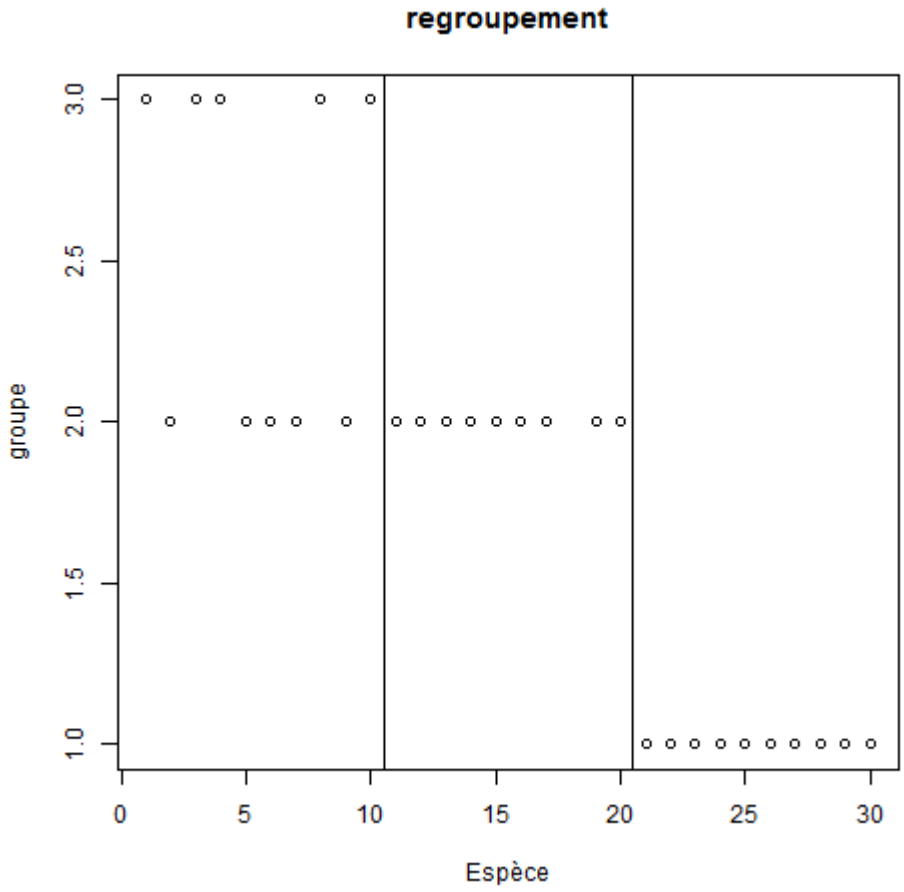

FIGURE 9.1 – Résultats du regroupement pour un mélange de croissance (ou mortalité) de 30 espèces provenant de 3 groupes sur 100 lignes de données avec 3 variables explicatives

Remarque : L'algorithme n'a pas numéroté les groupe de la même façon que la simulation. La figure 9.1 permet de savoir que son groupe 3 correspond au groupe 1 et vice versa. Et on déduit que seuls cinq espèces sont mal classées. Ce qui peut traduire la manque d'information pour ces espèces et une ressemblance avec ceux du groupe 2.

Ces résultats peuvent aussi s'illustrer avec la courbe logistique des probabilités de croissance (ou mortalité) estimées des différentes parcelles des espèces dans leurs groupes. On remarquera que pour cet exemple la courbe estimée est pratiquement identique à la courbe réelle dans les deux premiers groupes.
Les figures suivantes permettent de résumer la précision des estimations et la qualité du classement :

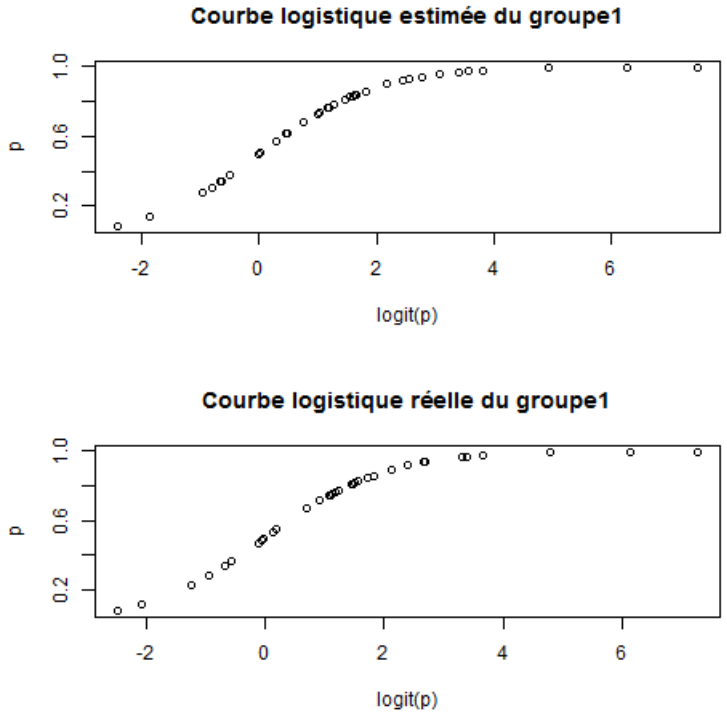

FIGURE 9.2 – Courbes logistiques du groupe 1

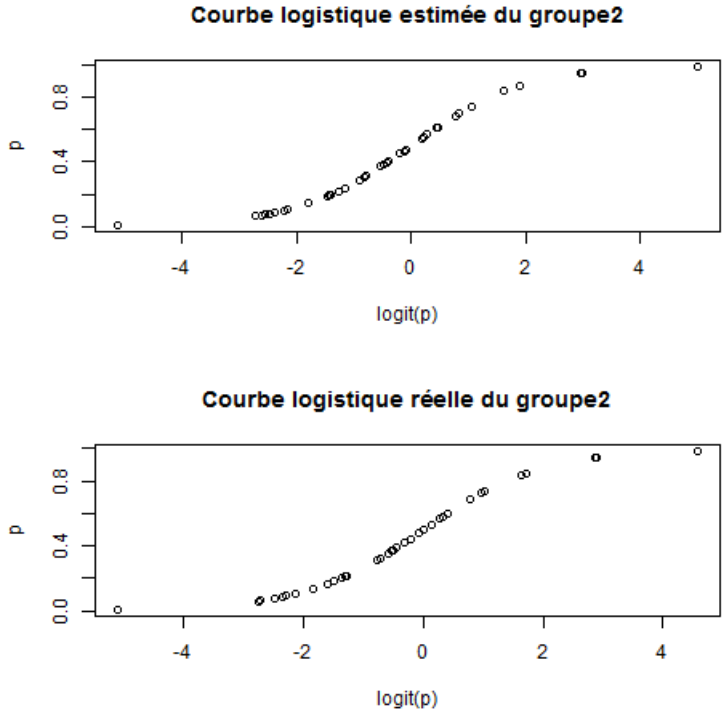

FIGURE 9.3 – Courbes logistiques du groupe 2

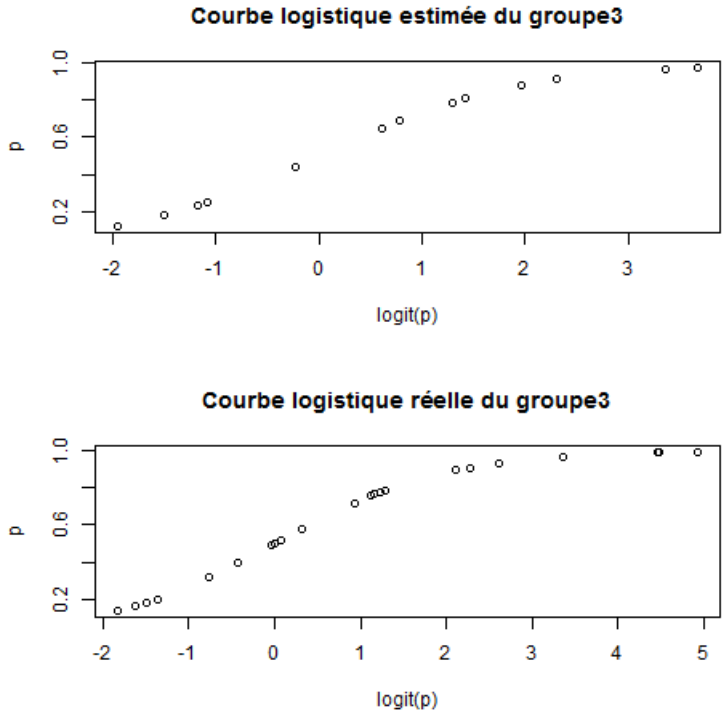

FIGURE 9.4 – Courbes logistiques du groupe 3

9.2 Résultats avec flexmix

En moins d'une minute, on obtient les résultats suivant :

TABLE 9.3 – Estimation des paramètres de régression avec flexmix, pour un mélange de croissance (ou mortalité) de 30 espèces provenant de 3 groupes sur 100 lignes de données avec 3 variables explicatives

valeurs réelles			valeurs estimées		
θ_1	θ_2	θ_3	θ_1	θ_2	θ_3
0.2732180	0.6715684	-0.5759935	0.39818511	0.6033668	-0.6808012
-0.1537600	-0.5228565	-0.7054100	-0.17120918	-0.4726924	-0.7276809
-0.09930405	0.12545283	1.84788650	0.04475755	0.1230896	2.2061643
0.4325952	0.9186994	1.5026038	0.50958455	0.9189840	1.4450973

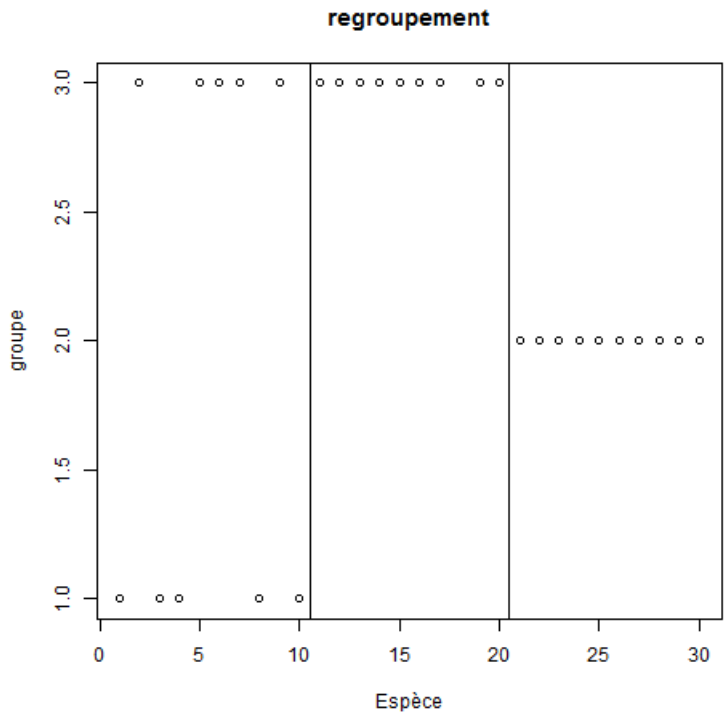

FIGURE 9.5 – Résultats du regroupement de flexmix pour un mélange de croissance (ou mortalité) de 30 espèces provenant de 3 groupes sur 100 lignes de données avec 3 variables explicatives

Remarque : flexmix ne fait pas le regroupement des espèces, mais des lignes du tableau de données, donc des parcelles, mais on retrouve le même classement que notre algorithme. Ceci permet de confirmer la qualité de notre algorithme et ouvre les porte pour le développement du modèle dans le cas des inventaires irrégulier.
A la différence de notre algorithme, l'interprétation des proportions du mélange ne correspond pas au modèle, ici, il s'agit des probabilités d'appartenance des parcelles aux différents groupes.
Notons cependant la vitesse de convergence de flexmix qui est bien meilleur.

Chapitre 10

Récapitulatif

Par rapport au test de la simulation précédente, on déduit le tableau suivant :

TABLE 10.1 – Tableau Comparatif

Tableau comparatif	Notre Algorithme	flexmix
Temps d'exécution	grand	petit
Précision	Très bonne	bonne
Classification	Excellente	Excellente

En somme, on peut dire que notre algorithme et flexmix convergent vers la même classification mais, font la différence au niveau de la précision des estimations, cela explique d'une certaine façon pourquoi en terme de temps, flexmix est meilleur que notre algorithme.
L'utilisation du package flexmix, nous rassure de notre développement et permet de valider notre démarche par la concordance que nous avons dans les résultats. Étant donné que flexmix limite notre modélisation de la dynamique forestière, ces résultats obtenus nous permettent d'étendre notre développement du modèle au cas des inventaires à intervalles de temps irréguliers.
Pour ce qui est de notre algorithme, on comprend que la recherche de la bonne solution augmente le temps d'exécution. En fonction des objectifs poursuivis, on peut donc penser à la définition des heuristiques [1] pour améliorer les résultats. Car en effet, si le but est d'avoir le bon résultat, le facteur temps peut être négligé, mais si l'on tien compte du facteur temps, on doit définir un plus grand niveau de tolérance s'adaptant au contexte. Ainsi, en couplant à tout cela une méthode d'optimisation pour améliorer le temps d'exécution, on est sûr d'avoir des résultats hautement meilleurs.

1. Critère permettant de joindre la qualité de la solution et le temps mis pour l'atteindre

CONCLUSION

Nous sommes parvenus au terme de notre étude. Il était question d'explorer la jointure entre la classification non supervisée et le suivi des processus de dynamique forestière. L'analyse des modèles des processus de dynamique forestière nous a fait ressortir deux modèles de régression : le modèle de régression logistique permettant d'inférer sur la probabilité de grandir ou de mourir des arbres dans une classe de diamètre, et la régression de Poisson qui permet d'inférer sur la régénération en ne considérant que le recrutement. Après avoir parcouru les traditionnels modèles de classification non supervisée : hiérarchique, k-moyennes, Kohonen et modèle de mélange, nous avons justifié le choix de l'algorithme EM (Expectation-Maximization) basé sur les modèle de mélange qui s'adapte le mieux à la dynamique forestière. Le couplage de régression et de classification nous a donc permis de faire un modèle de mélange de régression qui nous a fait aboutir à des modèles EM pour les trois processus dans le cas des inventaires à intervalle de temps régulier et quelconque. Ce dernier cas permettant d'arriver à notre objectif de généralisation à des inventaires espacés d'une manière quelconque. Dans le cas des processus de mortalité et de croissance, l'implémentation de notre modèle pour les inventaires en intervalle de temps réguliers nous a permis d'avoir la même classification que flexmix, mais des précisions différentes pour les paramètres estimés, ce qui traduit des résultats un problème de complexité dans l'optimisation faite dans notre algorithme.

En perspective, il serait intéressant d'aboutir à un programme formalisé pour permettre la réutilisation de l'implémentation du modèle conçu. Il s'agira du développement d'un package R ou dans un langage meilleur (langage C par lequel R a été développé), dédié à l'étude de la dynamique forestière. Il faudrait aussi, continuer à approfondir l'analyse des instructions utilisées dans le programme afin d'arriver à la solution package forestier la plus optimale.

Annexe : Algorithmes en langage R

Bibliographie

[1] Rossi Vivien. Projet de dynamique des forêts d'afrique centrale, 2014. www.dynaffor.org.

[2] Artus Isabelle. Forêts tropicales humides, avenir de la planète, sep 2011. www.rfi.fr.

[3] Pourquoi coupe-t-on les arbres de la forêt du pays-de-monts ?, nov 2015. www.lasemainevendeenne.fr.

[4] Cabanes Guénaël. Classification non supervisée à deux niveaux guidée par le voisinage et la densité, Avr 2011. Université Paris Nord.

[5] Cottrell Marie and Letremy Patrick. *Algorithme de Kohonen : classification et analyse exploratoire des données*. CNRS UMR 8595, 2003. Université Paris 1-Sorbonne.

[6] Schèma illustrant l'algorithme d'apprentissage de som, juil 2007. https ://fr.wikipedia.org.

[7] Densité mélange, nov 2005. https ://fr.wikipedia.org.

[8] titre1 :comment choisir les algorithme dans microsoft azure machine learning | titre2 :climate change : Implication for food-borne diseases (salmonella and food poisoning among humans in r. macedonia), 2016 | 2012. www.intechopen.com | https ://azure.microsoft.com.

[9] R : logiciel pour les statistiques et la gestion, mar 2014. http ://www.jawharafm.net.

[10] Dubois Daniel. Notepad++, l'éditeur de texte. http ://www.web-eau.net.

[11] Molto Quentin. *Estimation de la Biomasse en forêt tropicale*. PhD thesis, Université des Antilles et de la Guyane, 2012.

[12] Les stades de developpement de la forêt, fev 2009. www.scoutorama.org.

[13] Franc Alain, Gourley-Fleury Sylvain, and Picard Nicolas. *Une introduction a la modelisation des forets heterogenes(chapitre 2)*. Engref.

[14] Vanclay Jerome. Modelling forest growth and yield : applications to mixed tropical forests, 1994.

[15] Karem Fatma, Dhibi Mounir, and Martin Arnaud. Combinaison de classification supervisée et non supervisée par la théorie des fonctions de croyance. *International Conference on Belief Functions, Compiègne, France*, mai 2012.

[16] Besse Philippe and Laurent Beatrice. *Apprentissage Statistique : modélisation, prévision et datamining*. Université de Toulouse/Département de Génie Mathématique et Modélisation.

[17] Gambette Philippe. *Classification supervisée et non supervisée*. Université Paris-Est Marne-la-Vallée/Ingéniérie Linguistique, 2014.

[18] Hadd Mustapha. Classification de la population en catégories socio-économiques : méthodologie et application pratique, 1999. www.memoireonline.com.

[19] AMAJDA. Rapport de recherche sur le réseau de kohonen, fev 2015. Université Paris 1-Sorbonne.

[20] Biemacki Christophe. Pourquoi les modeles de melange pour la classification ? *CNRS et Universite de Lille 1, Villeneuve d'Ascq, France*, 2009.

[21] Dempster, Laird, and Rubin. Maximum likelihood from incomplete data via the em algorithm, 1977.

[22] Santos Frédéric. L'algorithme em : une courte présentation. *CNRS, UMR 5199 PACEA*, août 2009.

[23] Lejeune Michel. *Statistique La théorie et ses applications*. Springer, 2010. deuxieme edition.

[24] Rouvière Laurent. *Régression logistique avec R*. Université de Rennes 2, UFR Sciences Sociales.

[25] Rakotomalada Ricco. *Introduction à R Les bases du langage R*. Université Lyon 2.

[26] Avram Florin. Statistique spatiale, geostatistique, regression et interpolation. dec 2009.