Rapport de projet tutoré

UE Projet : MIS 2251
Année académique : 2020 - 2021

Etude par des modèles de mélange de données de durée de vie issues de spectroscopie par fluorescence

---

Réalisé par : Diamondra RAKOTONDRAZAKA, étudiant en 4e année du Cursus Master en Ingénierie (CMI) en sciences des données à l'Université Bretagne Sud (UBS) de Vannes, Charaf HOUDIF, étudiant en première année du master en sciences des données et modélisation statistique à l'UBS de Vannes

Tuteur : Pierre-Yves LOUIS

Tuteur et enseignant : Evans GOUNO

Résumé

Ce projet tutoré porte sur l'étude par des modèles de mélange de données de durée de vie issues de spectroscopie par fluorescence.

Nous expliquons d'abord ce qu'est le phénomène de fluorescence, qui est ici le contexte où se place notre travail. La fluorescence est la capacité de corps physiques (fluorophores) à émettre de la lumière suite à l'exposition à un rayonnement.

Ce qui nous intéresse ici est l'étude de la durée de ce phénomène. Les données utilisées dans ce projet portent sur ces "durées de vie" et sont hétérogènes, dans le sens où les valeurs peuvent venir de plusieurs fluorophores.

Il a donc fallu utiliser l'algorithme EM, afin de déterminer les paramètres de décroissance des durées de vie pour des nombres de fluorophores donnés. Ensuite sélectionner à l'aide du critère BIC, le meilleur modèle afin de savoir combien de fluorophores portent les données considérées et les paramètres de leurs distributions.

Nous exposons ensuite des résultats obtenus en s'appuyant sur les données d'une substance chimique nommée tryptophane, qui contient le comptage de photons (décroissant) d'un mélange de fluorophores. Cela caractérise le déclin de la durée de vie de fluorescence de ces fluorophores. Nous avons trouvé que ces données de mélanges portent sur 2 fluorophores et nous avons obtenu les estimations des paramètres de leurs distributions.

Remerciements

En préambule, nous souhaitons adresser nos remerciements à M. Pierre-Yves Louis pour nous avoir permis de réaliser ce projet pour l'établissement AgroSup Dijon, pour son accueil, pour ses conseils, pour sa disponibilité tout au long de ce projet, pour son implication dans nos recherches et sa participation au cheminement de ce rapport.

Nous tenons également à remercier M. Evans Gouno pour l'aide et les informations durant toute la période du stage.
Nous présentons nos sincères remerciements à tous nos enseignants pour leur formation.

Table des matières

1 Introduction **4**

2 Description du phénomène de fluorescence **5**
2.1 Définition de la fluorescence 5
2.2 Durée de vie de fluorescence 5
2.3 Absorption/Emission de photons 6

3 Description des données **7**

4 Modèles de mélange **8**
4.1 Loi de mélange 8

5 Algorithme EM **9**
5.1 Présentation de l'algorithme 9
5.2 Estimation par maximum de vraisemblance 9
5.3 Fonctionnement de l'algorithme EM 9
5.4 Formalisation de l'algorithme EM 10
5.5 Exemple pour un mélange de 2 densités exponentielles 10
5.6 Cas pour un nombre de densités exponentielles quelconque 12

6 Sélection du modèle **13**

7 Résultats **14**

8 Conclusion **15**

Annexes **17**

9 Création d'un modèle de mélange de 2 lois exponentielles sur des données simulées **18**

10 Autre méthode d'ajustement d'une loi exponentielle aux données portant sur un unique composé **21**

1 Introduction

Tout d'abord, de façon générale en analyse de données, l'un des principaux défis est d'adapter le choix de l'inférence statistique pour des échantillons lorsque des observations sont manquantes ou censurées.

Nous retrouvons le même cas de figure et la même contrainte dans notre étude sur la durée de vie de fluorescence. Notre projet consiste à réaliser une analyse des données sur le profil de désintégration d'une ou plusieurs molécules (composants de fluorescence).

L'objectif principal de notre projet est de répondre aux problématiques suivantes :

Comment déterminer le nombre optimal de composants pour obtenir un bon ajustement du modèle ?
Comment estimer de manière efficace les paramètres des distributions de chaque composant ?

Dans un premier temps nous allons effectuer une présentation compréhensible du phénomène de fluorescence, et une description détaillée des données utilisées pour notre étude. Ensuite dans un deuxième temps nous parlerons de quelques aspects généraux des modèles de mélange. De plus, nous allons effectuer une explication approfondie de l'algorithme EM qui nous permettra de résoudre les problématiques citées auparavant. Pour finir, nous présenterons les résultats de notre travail avant de conclure.

2 Description du phénomène de fluorescence

2.1 Définition de la fluorescence

La fluorescence est la propriété de certains corps physiques à émettre de la lumière par émission de photon, sous l'influence d'un rayonnement lumineux. On appelle fluorophore, un composant capable d'émettre cette lumière liée au phénomène de fluorescence.

Lors de l'influence d'un rayonnement, les électrons d'une molécule ou atome sont excités. Il survient alors l'absorption de l'énergie d'un photon par le corps, qui se retrouve dans un état électroniquement excité.

Le retour à l'état fondamental de ce dernier se fait par fluorescence ou phosphorescence. C'est à dire, l'émission d'un photon. La phosphorescence diffère à la fluorescence par une durée plus longue d'émission après excitation.

"Le terme fluorescence a été introduit par Stokes en 1852 dans une description du minéral fluorine qui émet de la lumière visible à la suite de l'éclairement avec des rayons UV invisibles." [1]

2.2 Durée de vie de fluorescence

Une caractéristique d'intérêt à étudier appartenant à la fluorescence est sa durée de vie. La durée de vie est le temps à partir de l'excitation de la molécule, jusqu'à l'émission du photon lié au phénomène de fluorescence ramenant la molécule à son état fondamental. Une molécule peut être identifiée grâce à cette durée de vie, qu'on appelle aussi profil de désintégration.

Le déclin de fluorescence d'une molécule $I(t)$ représente la concentration de molécules fluorescentes dans l'état excité depuis le moment d'excitation, suivant le temps t. Il est généralement exponentiel, de forme :

$$I(t) = a \exp^{-\frac{t}{\tau}}$$

Avec a une constante réelle et τ la durée de vie de fluorescence de la molécule (durée moyenne que la molécule soit en état excité avant son retour à l'état fondamental par émission d'un photon de fluorescence).

Cette durée de vie dépend de la molécule émettant le photon, du milieu physique où se trouve celui-ci et des interactions avec d'autres molécules. Cette étude donne l'accès à une grande variété d'informations concernant les processus moléculaires, et est très utile en biologie pour l'analyse de tissus et de cellules.

Dans un mélange de fluorophores, ce modèle est généralisé de la façon suivante :

$$I(t) = \sum_i \alpha_i \frac{1}{\tau_i} \exp^{-\frac{t}{\tau_i}}$$

Dans ce modèle multiexponentiel, α_i est la proportion des molécules des composantes dans leur milieu, on a $\sum_i \alpha_i = 1$. $\alpha_i = a_i \tau_i$ avec a_i une constante réelle et τ_i la durée de vie de fluorescence de la molécule du composé i. Le paramètre de décroissance τ_i suit une loi exponentielle.

2.3 Absorption/Emission de photons

Comme nous l'avons vu précédemment l'origine de l'excitation de la molécule est due à l'absorption d'un ou plusieurs photons (multiphotons), ce phénomène d'absorption fait passer la molécule d'un état basal à un état excité. Pour mieux comprendre ce phénomène, nous allons étudier le diagramme de Jablonski :

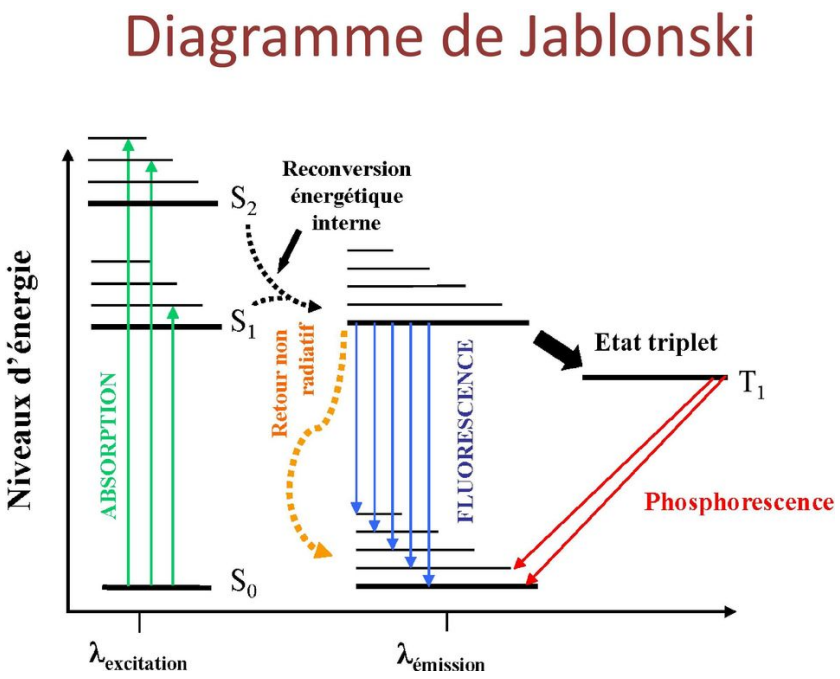

FIGURE 1 – Diagramme d'énergie dit de Jablonski

Dans le diagramme de Jablonski l'état basal de chaque molécule est représenté par l'état fondamental S_0, et les états excités par S_n avec $n \geq 1$. Chaque état excité possède plusieurs niveaux vibrationnels.

Suite à l'absorption de photons (suite à un rayonnement) la molécule passe de l'état fondamental S_0 à l'état excité S_n ,finalement le retour à l'état fondamental se fait en deux étapes :
- Première étape la transition $S_n \mapsto S_1$
- Deuxième étape la transition $S_1 \mapsto S_2$

La première transition se fait par une reconversion énergétique interne de la molécule, la deuxième transition c'est à dire le retour à l'état fondamentale s'effectue se fait par la libération de photons (la fluorescence).

3 Description des données

Afin de mettre en pratique les notions liées au traitement de données issues d'un mélange de distributions, le tuteur du projet tutoré nous a fourni des données répertoriées en deux parties.

La première partie contient des données de comptage, ces données représentent la variation du nombre de photons présents dans la molécule de la substance chimique étudiée (fluorescéine), suite à l'interaction de la fluorescéine et une énergie lumineuse pour créer un phénomène de fluorescence.

Ces données sont composées de 910 valeurs, chaque valeur définit le nombre de photons présents dans la molécule (fluorescéine) à l'instant t. Le comptage de photons s'effectue suivant un pas de temps constant et calibré afin d'obtenir des données cohérentes. Ce type de données sont généralement mesurées grâce à des compteurs de photons ou des spectromètres de fluorescence temporels.
Puis qu'il s'agit de données de comptage prétraitées, notre étude va plutôt se focaliser sur la décroissance du nombre de photons au cours du temps, puisque nos données de comptage démarrent à l'instant 0 lorsque le nombre de photons est à son maximum et diminue fortement par la suite, à cause de l'émission de photons.

En effet comme nous l'avons très bien vu à l'aide du diagramme de Jablonski (voir Figure 1), la fluorescence est l'émission de photons de la part de la molécule ,ce phénomène d'émission de photons est extrêmement rapide et s'effectue généralement en quelques nanosecondes , ce qui explique la décroissance rapide de nos données de comptage.

Le deuxième répertoire contient des mesures spectrophotométriques du tryptophane. Il s'agit ici aussi de données de comptage du nombre de photons (la molécule et la substance chimique changent) , cependant différemment des données du premier répertoire, ces données contiennent ,des mesures de comptage pour chaque valeur différentes de PH de la substance chimique (allant de 1 à 11), puisque le PH est un facteur potentiellement influent sur la durée vie de fluorescence. Pour chaque PH ,nous avons au plus 4 expérimentations différentes c'est à dire 4 séries de données de comptage différentes.

4 Modèles de mélange

4.1 Loi de mélange

Une loi de mélange est une fonction de densité f qui est une combinaison linéaire convexe de plusieurs fonctions de densité. Cela s'écrit de la manière suivante :

$$f(x) = \sum_{i=1}^{K} p_i f_i(x)$$

$f_1, ..., f_K$ sont les K densités constituant la loi de mélange, et $p_1, ..., p_K$ sont les poids donnés à chacune des densités. On a $\sum_{i=1}^{K} p_i = 1$, ils représentent la probabilité $\mathbb{P}(Z = i)$ de la variable aléatoire latente Z, sur la classe d'appartenance (cachée) de chaque observation issue de la loi de mélange.

Une réalisation d'une variable aléatoire suivant la loi f est générée par une réalisation de Z pour obtenir la classe d'appartenance $i \in \{1, ..., K\}$, puis la réalisation d'une variable aléatoire suivant la loi de la densité f_i.
Alors les densités f_i d'une loi de mélange, sont conditionnelles à $\{Z = i\}$.

L'objectif de l'étude de données suivant une loi de mélange est de faire de la classification ou du *clustering*. La classification étant ici l'affection d'une valeur de réponse (selon une variable à modalités), aux données en entrées. Et le *clustering* est la recherche de similitudes ou de schémas entre les observations d'un échantillon. On se trouve bien ici respectivement dans des études statistiques, paramétrique et non-paramétrique.

Concrètement, ce qu'on cherchera à tirer de données suivant une loi de mélange est l'estimation des paramètres des densités, et/ou trouver les classes d'appartenance (associées à la variable latente Z), et/ou trouver le nombre total de ces classes.

On a simulé sur R le mélange de 2 lois exponentielles, qu'on retrouve sur le Figure 6 en Annexes.
Pour un nombre n d'observations, on effectue un tirage z de loi de Bernoulli de paramètre $p \in [0, 1]$. Où p est la probabilité d'appartenance des données à la première densité, et 1-p à la deuxième. Selon la valeur de z, on tire une valeur de la loi exponentielle associée à la probabilité d'appartenance de la classe considérée. On itère ce processus n fois.

5 Algorithme EM

5.1 Présentation de l'algorithme

L'algorithme EM a été publié par les scientifiques Dempster, Laird et Rubin en 1977. Il s'agit d'un algorithme itératif particulièrement approprié pour trouver les paramètres du maximum de vraisemblance d'un modèle lorsque ce dernier dépend de variables latentes donc non-observables, ou lorsque l'expression de la vraisemblance du modèle ne permet pas le calcul de la maximisation.

5.2 Estimation par maximum de vraisemblance

Soit n observations étant les réalisations de n variables aléatoires indépendantes et identiquement distribuées $(X_1, ..., X_n)$. L'estimation par maximum de vraisemblance est une méthode statistique permettant d'estimer les paramètres inconnus d'une certaine loi de probabilité.
Tout d'abord, on établit la vraisemblance, cette dernière mesure la probabilité que des observations proviennent d'une loi donnée :

$$L(x_i \dots x_n\, \theta) = \prod_{i=1}^{n} f(x_i\, \theta)$$

Avec f la densité des X_i et $i = 1, ..., n$, s'ils suivent une loi discrète on a alors $f(x_i\, \theta) = \mathbb{P}(X_i = x_i)$ Plus généralement on utilise la log-vraisemblance, qui peut faciliter l'expression de densité jointes. Par exemple dans le cas de loi discrète on a :

$$\log(L(x_i, ...x_n, \theta)) = \ l((x_i \dots x_n\, \theta)) = \sum_{i=1}^{n} \log(P(X_i = x_i))$$

Ensuite on détermine l'estimateur du maximum de vraisemblance, c'est à dire la valeur de l'estimateur qui va nous permettre de maximiser cette fonction log-vraisemblance :

$$\widehat{\theta} = \text{argmax}_\theta\, l((x_i, ...x_n, \theta))$$

5.3 Fonctionnement de l'algorithme EM

Soit les données complètes C=(X,Z) avec des données incomplètes $X = (x_1, ...x_n)$ et des données cachées $Z = (z_1, ...z_n)$. On note la fonction log-vraisemblance des données complètes $l(\theta, X, Z)$ Comme nous l'avons évoqué précédemment l'algorithme EM est une procédure itérative qui s'effectue en deux étapes c'est à dire :

Etape E (*Expectation*) : comme son nom le laisse supposer cette étape procède à l'estimation des données inconnues, autrement dit la détermination des probabilités aux différentes classes d'appartenance cachée. Cela sachant les données observées et la valeur des paramètres déterminés à l'itération précédente (m-1).

$$Q(\theta|\theta^{(m-1)}) = \mathbb{E}_{Z|X,\theta^m}[l(\theta, X, Z|X, \theta)]$$

Etape M : couramment appelé phase *Maximisation*, comme son nom le laisse supposer cette étape procède donc à la maximisation de la vraisemblance. On effectue la mise à jour de l'estimation des paramètres.

$$\theta^{(m)} = \text{argmax}_\theta \, Q(\theta|\theta^{(m-1)})$$

5.4 Formalisation de l'algorithme EM

Entrée : Les données x,la valeur initiale $\theta^{(0)}$
Sortie : $\theta^{(m)}$
Data : x
Etape E θ **:**

$\mathrm{Q}(\theta|\theta^{(m-1)}) = \mathbb{E}_{Z|X,\theta^m}[l(X,Z|\theta)]$

return $\mathrm{Q}(\theta|\theta^{(m-1)})$

Etape M $Q(\theta|\theta^{(m-1)})$ **:**

$\theta^{(m)} = \text{argmax}_\theta \, Q(\theta|\theta^{(m-1)})$

return $\theta^{(m)}$

5.5 Exemple pour un mélange de 2 densités exponentielles

Pour illustrer l'illustration de cet algorithme, nous allons prendre le cas de données d'un mélange de 2 composés de fluorescence. On a donc les nombres de photons suivant de temps pour 2 composés désignant les classes d'appartenances cachées Z.

La formule du déclin de fluorescence pour un mélange de 2 fluorophores est donc :

$I(t) = \sum_{i=1}^{2} \alpha_i \frac{1}{\tau_i} \, \mathrm{e}^{-t/\tau_i}$

avec $\alpha_2 = 1 - \alpha_1$

Soit X, un n-échantillon iid de n observations issues de ce mélange et Z, les classes cachées désignant la distribution de chaque observation de X.
Par soucis de simplicité de calculs, on pose $\lambda = \frac{1}{\tau}$ Les paramètres que l'on va donc estimer sont $\lambda_1, \lambda_2, \alpha 1, \alpha_2$. On note l'ensemble de ces paramètres θ.

La vraisemblance des données complètes vaut :

$L(X|Z,\theta) = \prod_{i=1}^{n} \sum_{j=1}^{2} \mathbb{1}_{\{Z_i=j\}} \, \alpha_j \, \lambda_j \, \mathrm{e}^{-\lambda_j x_i}$

Et la log-vraisemblance vaut :

$l(X|Z,\theta) = \sum_{i=1}^{n} \sum_{j=1}^{2} \mathbb{1}_{\{Z_i=j\}}(-\lambda_j x_i + \log(\alpha_j) + log(\lambda_j))$

Comme les données sont incomplètes, la formule de la log-vraisemblance fait intervenir $\mathbb{P}(Z_i = j|X_i = x_i, \theta^{(m)})$ qu'on note $p_{i,j}$ avec $i = 1, ..., n$ et $j \in \{1, 2\}$. Cela définit la distribution a posteriori de Z_i connaissant X_i et $\theta^{(m-1)}$

Ce qui donne :

$Q(\theta|\theta^{(m-1)}) = \mathbb{E}_{Z|X,\theta^{(m-1)}}[l(X|Z,\theta)] = \sum_{i=1}^{n}\sum_{j=1}^{2} p_{i,j}(-\lambda_j x_i + log(\alpha_j) + log(\lambda_j))$

$= \sum_{i=1}^{n}[p_{i,1}(-\lambda_1 x_i + log(\alpha_1) + log(\lambda_1)) + p_{i,2}(-\lambda_2 x_i + log(1-\alpha_1) + log(\lambda_2))]$

Calcul des estimateurs du maximum de vraisemblance :

Ici on maximise $\mathbb{E}_{Z|X,\theta^{(m-1)}}[l(X|Z,\theta)]$ aux points λ_j et α_1 pour $j \in \{1,2\}$.

$\hat{\lambda_j}^{(m)}$:

$\frac{\partial Q(\theta|\theta^{(m-1)})}{\partial \lambda_j} = \sum_{i=1}^{n} p_{i,j}(-x_i + \frac{1}{\lambda_j})$

$\sum_{i=1}^{n} p_{i,j}(-x_i + \frac{1}{\lambda_j}) = 0$

$\Leftrightarrow \sum_{i=1}^{n} p_{i,j} x_i = \sum_{i=1}^{n} \frac{p_{i,j}}{\lambda_j}$

$\Leftrightarrow \hat{\lambda_j}^{(m)} = \frac{\sum_{i=1}^{n} p_{i,j}}{\sum_{i=1}^{n} p_{i,j} x_i}$ avec $j \in \{1,2\}$

$\hat{\alpha_1}^{(m)}$:

$\frac{\partial Q(\theta|\theta^{(m-1)})}{\partial \alpha_1} = \sum_{i=1}^{n} p_{i,1}\frac{1}{\alpha_1} - p_{i,2}\frac{1}{1-\alpha_1}$

$\sum_{i=1}^{n} p_{i,1}\frac{1}{\alpha_1} - p_{i,2}\frac{1}{1-\alpha_1} = 0$

$\Leftrightarrow \sum_{i=1}^{n} p_{i,1}\frac{1}{\alpha_1} = \sum_{i=1}^{n} p_{i,2}\frac{1}{1-\alpha_1}$

$\Leftrightarrow \sum_{i=1}^{n} p_{i,1}(\frac{1}{\alpha_1} - 1) = \sum_{i=1}^{n} p_{i,2}$

$\Leftrightarrow \sum_{i=1}^{n} \frac{p_{i,1}}{\alpha_1} - p_{i,1} = \sum_{i=1}^{n} p_{i,2}$

$\Leftrightarrow \sum_{i=1}^{n} \frac{p_{i,1}}{\alpha_1} = \sum_{i=1}^{n}\sum_{j=1}^{2} p_{i,j}$

$\Leftrightarrow \hat{\alpha_1}^{(m)} = \frac{\sum_{i=1}^{n} p_{i,1}}{\sum_{i=1}^{n}\sum_{j=1}^{2} p_{i,j}}$

Déroulement de l'algorithme :

On initialise $\alpha_1^{(0)}$, $\lambda_1^{(0)}$, $\lambda_2^{(0)}$

Etape E :

Calcul de $p_{i,j} = \mathbb{P}(Z_i = j|X_i = x_i, \theta^{(m-1)})$

Etape M :

Maximisation de $Q(\theta|\theta^{(m-1)})$ aux points de chaque paramètre à estimer.

Répétition des étapes E et M jusqu'à convergence.

5.6 Cas pour un nombre de densités exponentielles quelconque

Le calcul de dérivé partiel de $Q(\theta|\theta^{(m-1)})$ pour la maximisation de θ, se fait aisément pour un nombre de 2 densités de lois exponentielles.
Cependant, ce calcul se complique pour un mélange de plus de 2 densités. On a toujours la contrainte pour K densités, que $\sum_{j=1}^{K} \alpha_j = 1$.
En reprenant le raisonnement utilisé pour 2 densités, pour 3 densités on pose les α_1, α_2, $1 - \alpha_1 - \alpha_2$ pour respecter la contrainte.
Mais en dérivant $Q(\theta|\theta^{(m-1)})$ par rapport à α_1 (ou α_2), on ne peut pas isoler le paramètre lors de la maximisation car la dérivation laisse dans l'expression le terme du paramètre α_2.

Pour généraliser l'utilisation de l'algorithme EM pour un mélange de plus de 2 densités exponentielles, on va utiliser la méthode des multiplicateurs de Lagrange ce qui ajoute un paramètre en plus à estimer.
Dans ce cas de figure, pour un nombre de K densités exponentielles on a :

$Q(\theta|\theta^{(m-1)}) = \sum_{i=1}^{n}[\sum_{j=1}^{K} p_{i,j}(-\lambda_j x_i + \log(\alpha_j) + \log(\lambda_j)) + C(\sum_{j=1}^{K} \alpha_j - 1)]$

$\hat{\lambda_j}^{(m)} = \frac{\sum_{i=1}^{n} p_{i,j}}{\sum_{i=1}^{n} p_{i,j} x_i}$

$\hat{\alpha_j}^{(m)} = \frac{\sum_{i=1}^{n} p_{i,1}}{\sum_{i=1}^{n} \sum_{j=1}^{K} p_{i,j}}$

$\hat{C}^{(m)} = \sum_{i=1}^{n} \sum_{j=1}^{K} p_{i,j}$

Suite à la création de modèles de mélange pour différents nombres de classe, il va maintenant s'agir de sélectionner quel est le meilleur modèle parmi cela.

6 Sélection du modèle

Dans la démarche de base de *clustering* des données, on cherche bien à savoir quel est le meilleur regroupement possible de ces dernières. Afin d'avoir le nombre de densités du mélange (nombre de composants fluophores), et les paramètres des distributions (durée de vie de fluorescence de la molécule du composant).

Nous savons que nos données de fluorescence peuvent venir de 1 à 5 composants. Dans le cas mélanges d'exponentielles et estimation des paramètres avec l'algorithme EM, nous supposons que les données peuvent porter sur la durée de vie de fluorescence de 1 à 5 fluophores.
Nous allons donc estimer les paramètres de loi exponentielle pour ces nombres de densités et voir quel modèle s'ajuste le mieux aux données. Le critère de décision du meilleur modèle utilisé sera le critère d'information Bayésien (BIC).

Il se calcul de la façon suivante :

$BIC = -2\log(L) + K\log(n)$

Ici L est la vraisemblance estimée du modèle, K le nombre de paramètres libres et n le nombre d'observation.
La pénalité du modèle considéré dépend de la taille de l'échantillon de données et du nombre de paramètres.
Le modèle sélectionné est celui qui a la valeur minimum du critère BIC.

7 Résultats

Dans cette section présentant le résultat de nos recherches, nous prendons comme données de support le tryptophane. Ces données sont issues de mesures par spectroscopie, pour une valeur de pH à 5.

Commençons par afficher le nuages de points de ces données :

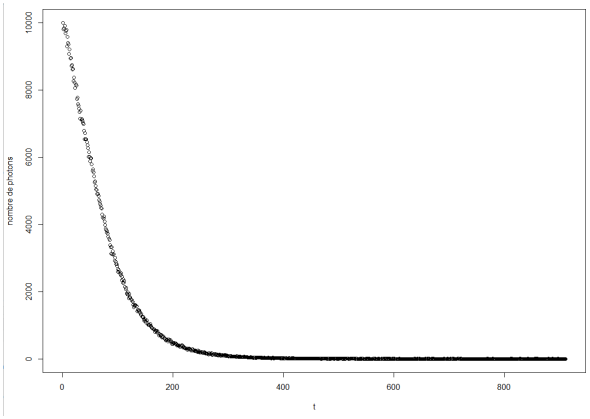

FIGURE 2 – Nuage de points des données du tryptophane

On a ici les comptages de photons pour une pas de temps t régulier, à l'allure décroissante. Ce qui montre le déclin de fluorescence de ou des molécules issues de la substance.
La valeur du nombre de photons suivant t n'est pas strictement décroissante, ce qui indique que les données mesurées peuvent être associées à plusieurs fluophores (donc plusieurs profils de désintégration).

Afin de déterminer le nombre de composants et leurs propres paramètres de décroissance, nous avons établi des modèles pour un nombre de 1 à 5 classes. Cela grâce à l'algorithme EM, et le choix s'est fait grâce aux critère BIC. Voici les résultats dans la Figure 3 :

K	λ_1	λ_2	λ_3	λ_4	λ_5	α_1	α_2	α_3	α_4	α_5	BIC
1	0.0012	-	-	-	-	1	-	-	-	-	14099.57
2	0.0005	0.1416	-	-	-	0.3820	0.6180	-	-	-	10612.74
3	0.0005	0.1416	0.1416	-	-	0.3820	0.3862	0.2318	-	-	11378.17
4	0.0005	0.1416	0.1416	0.1416	-	0.3820	0.2318	0.2318	0.1544	-	11865.47
5	0.0005	0.1416	0.1416	0.1416	0.1416	0.3820	0.2061	0.1373	0.1373	0.1373	12201.95

FIGURE 3 – Modèles obtenus suite à l'utilisation de l'algorithme EM

Le modèle choisi est le deuxième car c'est celui où le critère BIC est le plus petit. On conclut donc que la substance du tryptophane, où les mesures se sont faites pour un pH valant 5, contient l'information sur le profil de désintégration de 2 fluophores. Les paramètres estimés obtenus λ_1 et λ_2 d'un mélange de 2 exponentielles, correspondent aux paramètres de décroissance des 2 fluophores issus du tryptophane.

Il convient à présent d'afficher sur l'histogramme des données, le mélange des densités exponentielles pour les paramètres obtenus suite à l'estimation grâce à l'algorithme EM :

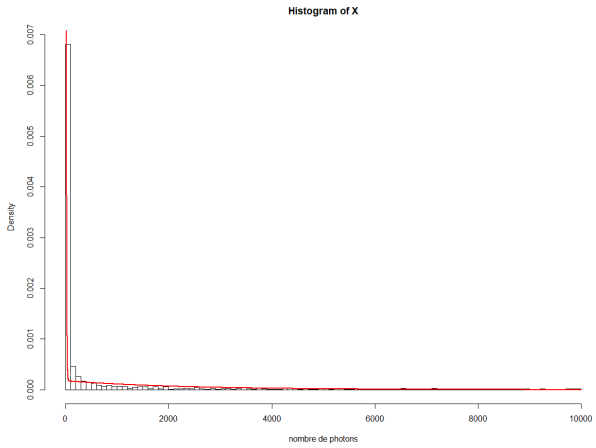

FIGURE 4 – Nuage de points des données du tryptophane

La densité donnée par le modèle de mélange biexponentielle considéré, est bien ajusté aux données.

8 Conclusion

Ce projet tuteuré nous a permis de mettre en pratique nos connaissances théoriques acquises durant notre formation, et utiliser l'un des logiciels les plus utiles en termes d'analyse statistique à savoir Rstudio.

L'algorithme EM très souvent utilisé dans le domaine de la modélisation, nous a permis d'estimer les paramètres de décroissance de mélange de fluophores. Ce qui constitue une étape intermédiaire à des études biologiques d'analyse de tissus et de cellules.

Nous nous sommes appuyé sur des données issues de la substance chimique, tryptophane où les mesures ont été prises dans un milieu où le pH était de 5. Nous avons vu que d'après la sélection de modèles multiexponentiels construits grâce à l'algorithme EM, que cette substance contient l'information sur le profil de désintégration de 2 fluophores.

Bien que le choix d'un modèle de modèle de mélanges exponentielles pour l'ajustement du modèle sur les données étudiées nous a semblé assez efficace, parallèlement on aurait pu aussi opter pour une approche des mélanges exponentielles par une loi gamma. Cette approche est souvent utilisée pour des échantillions à taille élevée, et est moins sensible à l'échec de convergence et *l'overfitting.* [6]

Références

[1] Tabea Rebafka.,Estimation dans le modèle d'empilement avec application aux mesures de la fluorescence, thèse de doctorat.Télécom ParisTech spécialité : signal et images. octobre 2009.

[2] Chit Yaw Fu Beng Koon Ng., Sirajudeen Gulam Razul., Fluorescence lifetime discrimination using expectation-maximization algorithm with joint deconvolution, rapport de recherche scientifique. novembre/décembre 2009.

[3] Frédéric Santos., L'algorithme EM : une courte présentation, rapport de recherche scientifique. août 2015.

[4] Nuno Vasconcelos., Expectation-Maximization, rapport de recherche scientifique. année inconnue.

[5] Fluorescence et phosphorescence, article, https://www.lachimie.fr/analytique/fluorimetrie/ (consulté le 20/04/2021)

[6] Daniel Sewell., Hajin Kim., Taekjip Ha., Ping Ma., A parameter estimation method for fluorescence lifetime data, rapport de recherche scientifique. juin 2015.

Annexes

```r
#initialisation des paramètres
lambda1 = 0.000000001
lambda2 = 0.000001
lambda3 = 0.000001
alpha1 = 0.2
alpha2 = 0.5
alpha3 = 0.3
C = 0.00001

#algorithme EM
EM_3 = function(X,lambda1,lambda2,lambda3,alpha1,alpha2,alpha3,eps,C){
  i=0
  lambda1_prec = lambda1
  lambda2_prec = lambda2
  lambda3_prec = lambda3
  alpha1_prec = alpha1
  alpha2_prec = alpha2
  alpha3_prec = alpha3
  C_prec = C
  convergence = F

  while(convergence == F) {
    #étape E
    vrais1 = alpha1*fexp(X,lambda1)
    vrais2 = alpha2*fexp(X,lambda2)
    vrais3 = alpha3*fexp(X,lambda3)
    vrais12 = vrais1 / (vrais1 + vrais2 + vrais3)
    vrais22 = vrais2 / (vrais1 + vrais2 + vrais3)
    vrais33 = vrais3 / (vrais1 + vrais2 + vrais3)
    #étape M
    alpha1 = sum(vrais12)/sum(vrais12 + vrais22 + vrais33)
    alpha2 = sum(vrais22)/sum(vrais12 + vrais22 + vrais33)
    alpha3 = sum(vrais33)/sum(vrais12 + vrais22 + vrais33)
    lambda1 = sum(vrais12)/sum(vrais12*X)
    lambda2 = sum(vrais22)/sum(vrais22*X)
    lambda3 = sum(vrais33)/sum(vrais33*X)
    C = sum(vrais12 + vrais22 + vrais33)

    #vérification du critère de convergence
    if (abs(lambda1 - lambda1_prec)<eps & abs(lambda2 - lambda2_prec)<eps & abs(lambda3 - lambda3_prec)<eps
        & abs(alpha1 - alpha1_prec)<eps & abs(alpha2 - alpha2_prec)<eps & abs(alpha3 - alpha3_prec)<eps
        & abs(C - C_prec)<eps)
    {convergence = T}

    lambda1_prec = lambda1
    lambda2_prec = lambda2
    lambda3_prec = lambda3
    alpha1_prec = alpha1
    alpha2_prec = alpha2
    alpha3_prec = alpha3
    C_prec = C
    #log-vraisemblance estimée par le modèle
    logL = sum(vrais12*(-lambda1*X + log(alpha1)+ log(lambda1)) + vrais22*(-lambda2*X + log(alpha2)+
    log(lambda2)) + vrais33*(-lambda3*X + log(alpha3)+ log(lambda3))) + C*(sum(alpha1 + alpha2 + alpha3 -1))
    i = i+1 #incrémentation du nombre d'itération
    }
  return(c(lambda1,lambda2,lambda3,alpha1,alpha2,alpha3,i, logL,C))}
```

FIGURE 5 – Exemple de code R pour l'algorithme EM faisant intervenir les multiplicateurs de Lagrange (K = 3 densités)

9 Création d'un modèle de mélange de 2 lois exponentielles sur des données simulées

La création de données simulées à pour but de vérifier le bon fonctionnement de l'algorithme EM, qui a été écrit à la main.
En effet, nous aurions pu utiliser la fonction EM.miexp de la librairie Renext qui calcul les paramètres de densités d'un mélange d'exponentielles suivant un echantillon de données et un nombre de densités. Mais pour appliquer le critère BIC, nous avons besoin de la log-vraisemblance et la fonction EM.miexp nous retournait des valeurs manquantes (NA) pour un nombre de densités supérieur à 2.

Voici la démarche qu'on a suivi avec le code R pour vérifier que le modèle créer grâce à l'algorithme EM est correct :

```r
#simulation des données

sim = NULL
sim = matrix(nrow=911,ncol=1)
class = NULL
class = matrix(nrow=911,ncol=1)

### Boucle permettant de tirer 1000 valeurs issues
### d'un mélange de 2 lois exponentielle :
for (i in 1:1000) {
  z = rbinom(1,1,0.4) # choix de la loi par tirage de Benoulli
  if (z == 1) {
    sim[i] = rexp(1,0.0005)
    class[i] = 1
  } else {
    sim[i] = rexp(1,0.0014)
    class[i] = 2
  }
}
```

FIGURE 6 – Simulation d'un mélange de 2 lois exponentielles

```r
#fonction de densité de la loi exponentielle
fexp = function(x,lambda){

  return(lambda * exp(-lambda*x))
}
```

FIGURE 7 – Fonction retournant la densité de la loi exponentielle

```r
#fonction retournant les paramètres estimés par l'algorithme EM
EM2_sim = function(X,lambda1,lambda2,alpha1,alpha2,eps){

  i=0 # initialisation du nombre d'itération

  # initialisation des paramètres à l'itération i-1 pour
  # la vérification de leur convergence
  lambda1_prec = lambda1
  lambda2_prec = lambda2
  alpha1_prec = alpha1
  alpha2_prec = alpha2

  convergence = F # booléen servant à indiquer si l'algorithme a convergé

  while(convergence == F) {
    # étape E
    vrais1 = alpha1*fexp(X,lambda1)
    vrais2 = alpha2*fexp(X,lambda2)
    vrais12 = vrais1 / (vrais1 + vrais2) # probas a posteriori p_{i,1}
    vrais22 = vrais2 / (vrais1 + vrais2) # probas a posteriori p_{i,2}

    # étape M
    alpha1 = sum(vrais12)/sum(vrais12 + vrais22)
    alpha2 = sum(vrais22)/sum(vrais12 + vrais22)

    lambda1 = sum(vrais12)/sum(vrais12*X)
    lambda2 = sum(vrais22)/sum(vrais22*X)

    #vérification du critère de convergence
    if (abs(lambda1 - lambda1_prec)<eps & abs(lambda2 - lambda2_prec)<eps
        & abs(alpha1 - alpha1_prec)<eps  & abs(alpha2 - alpha2_prec)<eps
    ){
      convergence = T}
    lambda1_prec = lambda1
    lambda2_prec = lambda2
    alpha1_prec = alpha1
    alpha2_prec = alpha2

    # calcul de la log-vraisemblance estimée par le modèle
    logL = sum(vrais12*(-lambda1*X + log(alpha1)+ log(lambda1)) +
                 vrais22*(-lambda2*X + log(alpha2)+ log(lambda2)))
    i = i+1 #incrémentation du nombre d'itération
  }

  # détermination des classes prédites par le modèle, de chaque observation
  # (au vu des probabilités d'appartenances Z)
  class_pred = list(rep(0,length(X)))
  for(l in 1:length(X)){
    if(vrais12[l]>0.5){
      class_pred[[1]][l]=1}
    else if (vrais12[l]<=0.5){
      class_pred[[1]][l]=2}
  }
  return(c(lambda1,lambda2,alpha1,alpha2,i,logL,class_pred))
}
```

FIGURE 8 – Code de l'algorithme EM pour un nombre de 2 classes

```
> #récupération des résultats
> res = EM2_sim(sim,lambda1,lambda2,alpha1,alpha2,eps)
>
> lambda1 = res[1]
> lambda2 = res[2]
> alpha1 = res[3]
> alpha2 = res[4]
> nb_it = res[5]
> logL = res[6]
> #class_pred = res[7]
>
> lambda1
[[1]]
[1] 0.000493027

> lambda2
[[1]]
[1] 0.001453291

> alpha1
[[1]]
[1] 0.4185454

> alpha2
[[1]]
[1] 0.5814546

> nb_it
[[1]]
[1] 432

> logL
[[1]]
[1] -7894.908

> #class_pred
>
> #calcul du taux d'erreur de classification par le modèle
> taux_erreurs = 0
> for(i in 1:length(class)){
+   if(class[i]!= class_pred[[1]][i]){
+     taux_erreurs = taux_erreurs + 1
+   }
+ }
> taux_erreurs = taux_erreurs/length(sim)
> taux_erreurs
[1] 0.2788145
```

FIGURE 9 – Résultats de l'algortihme EM et taux d'erreur du modèle obtenu

Nous obtenons un modèle ayant un taux d'erreur d'environ 27,9% de classification des observations aux 2 différentes distributions de lois exponentielles.

Après avoir effectuer plusieurs tests en relançant la simulation de données, on voit que le taux d'erreur tourne généralement autour de 30% .

Ce taux d'erreur peut être réduit ou stabilisé si on réduit la valeur d' ϵ, mais cela réduit la vitesse de convergence de l'algorithme.
On en conclut donc que la performance générale du modèle est plutôt satisfaisante.

10 Autre méthode d'ajustement d'une loi exponentielle aux données portant sur un unique composé

Dans le cas présent, on a des données de comptage de photons suivant le temps pour un seul fluophore. Nous ne sommes pas dans la situation de données de mélange.

Le modèle a été établi suite à la linéarisation grâce au logarithme népérien, de la fonction exponentielle :

$$\alpha \, \mathrm{e}^{-\beta x}$$

Voici le code décrivant cette démarche sur R :

```
# chargement des données
X = read.table("C:/Users/Ascensio/Downloads/ProjetTutoreVannes/fTrypt pH11 1.txt",header = T)

# variable à estimer y (nombre de photons)
y = X[,1]
y = y[y!=0] # on supprime les 0 pour ne pas avoir d'erreur dans le logarithme

ly=log(y)

#variable explicative (instants t successif)
t <- 1:length(ly)

# création du dataframe contenant les donnant
data1=data.frame(ly,t)

# création du modèle
model <- lm(ly ~ t, data=data1)

summary(model)

alpha <- exp(coef(model)[1])
beta <- - coef(model)[2]
alpha
beta

# affichage de la courbe des valeurs prédites par le modèle
#
plot(t,y, type = "l",ylab = "nombre de photons")
lines(t,alpha * exp(beta * t),type = "l",col="blue")
```

FIGURE 10 – Code R associé à la création d'un modèle exponentielle pour la description de la durée de vie d'un seul fluophore

Voici les résultats obtenu pour l'estimation des paramètres du modèle et l'affichage de la courbe des valeurs prédites sur les données :

```
> alpha <- exp(coef(model)[1])
> beta <- - coef(model)[2]
> alpha
(Intercept)
   10496.76
> beta
           t
0.005683497
```

FIGURE 11 – Paramètres estimés par le modèle exponentielle pour la description de la durée de vie d'un seul fluophore

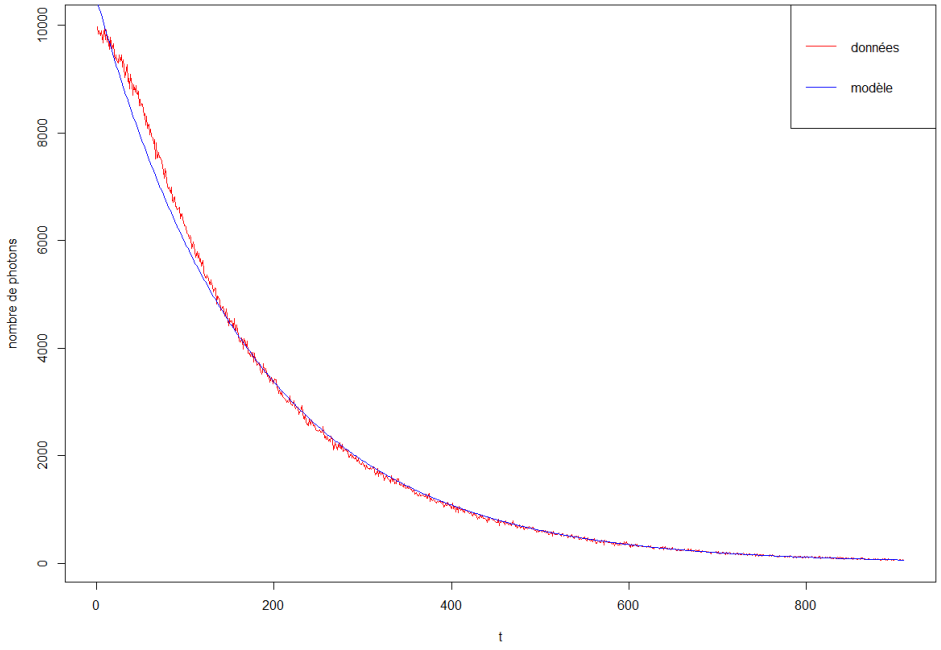

FIGURE 12 – Comparaison des courbes des valeurs prédites par le modèle et des données

Nous obtenons une courbe qui suit assez bien les données, ce qui pourrait indiquer que le modèle est de bonne qualité.

Voyons en détails les caractéristiques du modèle et notamment la valeur du coeffient de détermination R^2 :

```
> summary(model)

Call:
lm(formula = ly ~ t, data = data1)

Residuals:
     Min       1Q   Median       3Q      Max
-0.21914 -0.03689 -0.00724  0.03397  0.29611

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.259e+00  4.115e-03  2250.2   <2e-16 ***
t           -5.683e-03  7.825e-06  -726.3   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.06201 on 908 degrees of freedom
Multiple R-squared:  0.9983,    Adjusted R-squared:  0.9983
F-statistic: 5.275e+05 on 1 and 908 DF,  p-value: < 2.2e-16
```

FIGURE 13 – Caractéristiques du modèle exponentiel

Le coefficient de détermination est proche de 1, donc la prédiction fournie par le modèle est très bonne.