

ÉCOLE DOCTORALE MIPTIS

LABORATOIRE : PRISME

THÈSE présentée par :

Arnaud Paris

soutenue le : **18 octobre 2016**

pour obtenir le grade de : **Docteur de l'université d'Orléans**

Discipline : **Informatique**

**Modélisation du comportement habituel de la
personne âgée dépendante en environnement
incertain pour la détection d'évolutions
et d'activités anormales**

RAPPORTEURS :

Alain DUTECH

CR INRIA-HDR, LORIA

Dan ISTRATE

ECC-HDR, Université Technologique de Compiègne

JURY :

Olivier SIMONIN

Professeur, INSA Lyon

Maria DI MASCOLO

Directeur de recherche, CNRS

Pascal DORÉ

Responsable recherche, Legrand

Nacim RAMDANI

Professeur, Université d'Orléans, Directeur de thèse

Remerciements

Je souhaite tout d'abord remercier mon directeur de thèse, Monsieur Nacim Ramdani, grâce à qui j'ai acquis une nouvelle vision sur le monde. Je souhaite également remercier mes encadrantes, Mesdames Nathalie Cislo et Selma Arbaoui, pour m'avoir donné la chance d'effectuer cette thèse et pour l'ensemble des corrections.

Un grand merci à Adnen, pour avoir été là, et pour le temps que tu m'as consacré, que ce soit pour les conseils ou les (très nombreuses) corrections.

Je souhaite exprimer toute ma gratitude à Messieurs Alain Dutech et Dan Istrate d'avoir accepté de rapporter ce travail. Un grand merci à Alain Dutech pour ses commentaires et remarques qui m'ont permis d'améliorer ce manuscrit et à Dan Istrate pour l'ensemble de ses conseils. Je tiens également à remercier Olivier Simonin, Maria Di Mascolo et Pascal Doré, pour avoir accepté de participer à mon jury de thèse. Un merci particulier à Pascal Doré, pour l'ensemble de ses conseils et questions tout au long de ce projet.

Je souhaiterais remercier les membres du pôle Capteur, Monsieur Yves Parman-tier et Madame Sophie Seuillet pour leur aide tout au long, ainsi qu'à Antoine qui m'a guidé à travers les méandres du projet, administratives comme techniques, durant la première année.

Je souhaiterais remercier le directeur de l'IUT, Gérard Poisson, pour m'avoir accueilli dans ses locaux, mais également pour l'ensemble de ses conseils, notamment sur l'intérêt de l'enseignement.

Je souhaiterais remercier un certain nombre de personnes du laboratoire qui m'ont aidé durant cette thèse. Un merci particulier à Laure, l'Élite du labo, pour sa bonne humeur, son efficacité et son soutien. Mathieu, pour l'ensemble des discussions que l'on a eu, mais surtout pour l'aide apportée. Sylvain, pour son soutien et pour sa conscience écologique. Cyril, pour son intégrité et sa gentillesse. Léo, pour la musique et les soi-

rées. Phillipe, pour son sens civique. Mame, pour son aide.

Je souhaiterais remercier l'ensemble du département GC pour l'accueil que j'ai reçu. Je souhaiterais notamment remercier Julien, Régis et Aurélien pour leur soutien et leur humour.

Un grand merci aux thésards du laboratoire. Notamment Mohamed-Hédi, partenaire sur le projet et ami, pour avoir supporté tout mes coups de gueule et m'avoir soutenu durant tout le projet. Merci à Sonia, de toujours alléger l'atmosphère. Merci à Christophe pour les conseil, les discussions et la preuve qu'on peut lier l'intelligence de l'esprit à celle du cœur. Merci à Juan pour sa présence studieuse, et pour continuer à faire des pieds et des mains. Merci à Lounis et Nabil pour leur humour. Julien pour son sérieux. Éric, pour ses passages furtifs.

Merci à toute ma famille de m'avoir soutenu durant ces longs moments et pour tout ce qu'ils m'ont apporté. Merci à mes amis pour leur soutien.

Les auteurs remercient la Communauté d'Agglomération Bourges Plus, les Conseils généraux du Cher et du Loir-et-Cher, les régions Centre et Limousin, le Ministère de l'Industrie, le Fonds Européen de Développement Économique et Régional (FEDER) et BPI France pour leurs financements, ainsi que les pôles de compétitivité S2E2, Elopsys, SCS et le pôle Risques pour leur implication.

Table des matières

1	Introduction	1
1.1	Introduction générale	2
1.2	Projet E-monitor'âge	3
1.3	Objectifs de la thèse	4
2	Contexte de la thèse et état de l'art	7
2.1	Smart homes et supervision	8
2.2	Apprentissage du comportement	17
2.3	Approche de la modélisation	26
3	Les modèles de Markov	33
3.1	Les méthodes d'apprentissage artificiel	34
3.2	L'hypothèse de Markov	36
3.3	Exemples de modèles de Markov	36
3.4	Algorithme des Méthodes de Markov caché	42
4	Apprentissage du comportement habituel de la personne	47
4.1	Caractéristiques du modèle	48
4.2	Description du modèle Gs-GsHSMM	50
4.3	Implémentation des algorithmes	53
4.4	Étude de la complexité	56
5	Application	57
5.1	Application à l'Apprentissage de modèle de déplacement habituel en living lab	58
5.2	Apprentissage sur données simulées	64
5.3	Application à l'Apprentissage en situation réelle	73
5.4	Conclusion	79
6	Étude des évolutions du modèle Gs-GsHSMM	81
6.1	Objectifs	82
6.2	Méthodes de calcul de distance	84

6.3	Analyse de l'évolution du comportement	87
6.4	Distance KL convergente	98
7	Conclusion et discussion	107
7.1	Résultats	108
7.2	Discussions	108
7.3	Perspectives	109
Annexes		I
A	Glossaire	III
B	Table des figures	V
C	Rappels des algorithmes	VII
D	Références	XV

Chapitre 1

Introduction

Sommaire

1.1	Introduction générale	2
1.2	Projet E-monitor'âge	3
1.3	Objectifs de la thèse	4

1.1 Introduction générale

Les projections de populations réalisées en 2010 par l'Insee ([Blanpain et Chardon \[2010\]](#)) prévoient une augmentation du nombre de personnes âgées de 75 ans ou plus de 5.2 millions en 2007 à 11.9 millions en 2060. Cette nette augmentation est accompagnée d'une multiplication par 2 du nombre de personnes âgées dépendantes entre 2010 et 2060 ([Charpin et Tili \[2011\]](#)). Si l'augmentation du nombre de personnes âgées est importante, le pourcentage de personnes âgées dans la population l'est tout autant. Si les plus de 75 ans représentaient 4.3% de la population en 1960, 8.5% en 2007, le scénario central de la projection prévoit qu'ils représentent 16.2% en 2060 et qu'une personne sur 3 aura plus de 60 ans. De ce fait, la prise en charge des personnes âgées dépendantes a été déclarée un enjeu majeur. Cette augmentation s'accompagne d'une hausse du ratio de personne âgée par personne active mais également d'une augmentation de l'âge moyen des aidants¹, qu'ils soient professionnels ou non.

L'augmentation du nombre de personnes âgées s'accompagne d'une augmentation du nombre de personnes en [Établissement d'Hébergement pour Personnes Âgées Dépendantes \(EHPAD\)](#), majoritairement dans le cadre d'accueil longue durée ([Volant \[2010\]](#)). En parallèle du nombre de résidents en [EHPAD](#), le profil du résident type évolue également. L'entrée se fait de plus en plus tardivement, avec des niveaux de dépendance de plus en plus élevée. Dans l'ensemble des établissements d'accueil, la proportion des personnes dépendantes ([GIR \(Groupe Iso-Ressources\) 1 à 4](#)) est de 89% en 2011 et augmente, la proportion des personnes très dépendantes ([GIR 1 et 2](#)) étant de 55%. Les [GIRs](#) sont utilisés en France par les professionnels afin de classer le niveau de dépendance de la personne et permet une évaluation de la quantité de soins ou services nécessaires à la personne. Il est calculé en fonction de la capacité de la personne à effectuer un ensemble de 10 activités. Chacune est notée de A (l'activité est effectuée correctement, spontanément et habituellement) à C (l'activité est effectuée ni seul, ni spontanément, ni habituellement). Le [GIR 1](#) correspond aux personnes confinées au lit et ayant perdu toute leur autonomie. À l'inverse, le [GIR 6](#) correspond à des personnes complètement indépendantes pour les activités de la vie courante. Ainsi, l'augmentation de la proportion des personnes ayant des [GIRs](#) faibles ou très faibles implique que les résidents en [EHPAD](#) ont globalement besoin de plus en plus de soins.

L'augmentation du nombre de résidents² et de la dépendance des résidents rend les conditions de travail des aidants professionnels très compliquées. [Dufour-Renouf](#)

1. Nous désignerons par "aidant" toutes personnes intervenant dans la prise en charge d'un résident (soignant, membre associatif, etc.).

2. Nous utiliserons "résident" pour désigner les occupants des chambres de l'[EHPAD](#).

[2006] décrit les conditions de travail des aidants, que ce soit les difficultés physiques ou psychologiques. Un indice de la pénibilité du travail en EHPAD est la fréquence des accidents du travail, près de deux fois supérieure à la moyenne nationale (KPMG [2014]). Plus spécifiquement, Evers *et al.* [2002] étudient le rapport entre les comportements agressifs des résidents et les burnout des aidants.

Les difficultés du personnel sont particulièrement marquées la nuit (Anesm [2012]). Les équipes sont alors très réduites, et composées majoritairement d'aides soignants ou d'agents des services hospitaliers. Une majorité d'EHPAD n'ont pas d'infirmière la nuit, ni sur place, ni en astreinte. De plus, les habitudes de vie ou les pathologies font que les résidents peuvent avoir une activité très importante la nuit. Si cela est particulièrement marqué dans les services pour les résidents atteints de pathologies comme la maladie d'Alzheimer, l'ensemble des EHPAD peuvent être concernés par des déambulations, des insomnies ou des troubles du sommeil divers. Cette problématique est gérée par certains EHPAD en mettant en place des activités sociales durant la nuit, mais cela implique une logistique et des coûts importants.

Compte tenu de l'ensemble de ces problématiques, les EHPAD sont appelés à améliorer la prise en charge des résidents et à améliorer les conditions de travail du personnel soignant en tirant profit des progrès technologiques réalisés.

1.2 Projet E-monitor'âge

Le projet FUI 14 E-monitor'âge a pour objectif de développer un système de supervision en EHPAD permettant d'améliorer la prise en charge des résidents et d'assister le travail des aidants via un système d'alarme "intelligente" et par la mise à disposition d'informations sur les activités de la personne dans sa chambre. Dans le but de répondre au mieux aux besoins des aidants, des utilisateurs finaux ont été intégrés dans le projet, afin de valider les besoins, d'exprimer les problématiques ou encore de tester les solutions proposées. Les travaux menés au sein du projet vont de la création de capteurs innovants à la mise en place d'interfaces dédiées pour le personnel soignant.

Dans un premier temps, l'objectif est d'acquérir dans les chambres les données permettant de caractériser au mieux le comportement de la personne au sein de la chambre via un ensemble de capteurs bien appropriés. Puis, dans un second temps, il s'agit d'analyser ces données afin de les convertir en informations intelligibles par les aidants, et d'alerter en cas d'activités anormales au sein de la chambre.

Cette solution est surtout utile la nuit, quand le nombre d'aidants est le plus réduit. Différentes situations nocturnes à détecter ont été mises en avant par les experts, comme par exemple : les risques de chutes, la déambulation, les difficultés respiratoires, les intrusions et l'altération de la qualité du sommeil. Seule la chute, sur laquelle nous ne travaillerons pas, renvoie automatiquement une alerte *a priori*, les autres éléments doivent être des alertes basées sur des comparaisons avec le comportement habituel de la personne. Par exemple, il peut être parfaitement normal pour un résident de passer plus de temps la nuit à déambuler qu'à dormir. Dans ce cas, l'absence de déambulation doit être détectée et signalée si elle devait se poursuivre. À l'inverse, la déambulation de la personne devra émettre une alerte si ce n'est pas le comportement habituel de la personne.

Dans le cadre du projet, différentes contraintes ont été mises en avant. Un exemple de contrainte est liée aux capteurs, ne devant pas être portés, afin de ne pas rajouter de gênes aux aidants et pouvant être considérés comme intrusifs pour les résidents. Une autre contrainte est le caractère individualisé de la solution, afin de prendre en compte les spécificités des résidents.

1.3 Objectifs de la thèse

Le but de ce travail de recherche, mené dans le cadre du projet E-monitor'âge, est de mettre en place un système de supervision visant à générer des alertes intelligentes dans le but de prévenir les aidants de l'état anormal d'un résident et assurer ainsi une meilleure prévention. Toutefois, les dérives lentes du comportement peuvent ne pas être détectées et représentent un problème à part entière.

Ces deux problèmes reposent sur une connaissance du comportement habituel du résident. Pour cela, le système de supervision doit être doté de capacités d'apprentissage capables de déterminer le meilleur modèle représentant le comportement habituel de la personne.

Notre objectif ne se limite pas à l'apprentissage du comportement habituel de la personne afin de détecter si le comportement actuel de la personne est anormal mais il s'étend à la détection des dérives lentes du comportement.

Pour cela, commençons par présenter une revue de la littérature sur la supervision en smarthomes et la modélisation du comportement humain. Ensuite, nous présentons les modèles de Markov, avant de proposer un modèle pour le comportement hu-

main habituel. Puis, et afin de prouver la fiabilité de l'approche proposée, nous réalisons une série de simulations et nous présentons une étude pour une éventuelle amélioration des modèles élaborés. Nous testerons par la suite sur différents cas, comme des données acquis en Living Lab ou des données simulées. Enfin, nous étudierons la détection d'évolution des modèles avant de conclure.

Chapitre 2

Contexte de la thèse et état de l’art

Sommaire

2.1 Smart homes et supervision	8
2.1.1 Définition et état de l’art sur les smart homes	8
2.1.2 Exemple de smart homes : GIS Madonah	13
2.1.3 Localisation de la personne en smarthomes	13
2.2 Apprentissage du comportement	17
2.2.1 Hors smarthomes	19
2.2.2 En smarthomes	19
2.3 Approche de la modélisation	26
2.3.1 Méthodes utilisées	27
2.3.2 Discussion sur les méthodes	29

2.1 Smart homes et supervision

2.1.1 Définition et état de l'art sur les smart homes

Une **maison intelligente** ou *smart home* (Chan *et al.* [2008], De Silva *et al.* [2012], Solaimani *et al.* [2013]) est un environnement équipé de capteurs permettant d'avoir des informations sur l'activité au sein de la zone équipée. Il est possible de coupler ces méthodes de détection à un système de supervision capable de prendre des décisions, mais aussi d'équiper l'environnement d'actionneurs, permettant au système d'agir sur l'environnement. L'utilisation la plus courante est d'allumer automatiquement les lumières quand une personne entre dans la pièce, ce qui peut se faire avec plus ou moins d'intelligence. On parle plus généralement de la domotique. La notion de **smart home** peut correspondre aussi bien à des maisons ou des appartements qu'à des bureaux. Dans le contexte de ce travail, on s'intéresse aux appartements d'un EHPAD, composés généralement d'une chambre et d'une salle d'eau.

Utilisation des smart homes

Si les **smart homes** couvrent de nombreux domaines techniques, allant de la conception de capteurs à la mise en place d'algorithmes traitant l'information et passant par la gestion des communications, les domaines d'applications sont également très vastes et variés.

Parmi les domaines d'applications se trouvent, par exemple, la sécurisation de locaux contre les intrusions (Xiang *et al.* [2005], Ayers et Shah [2001]), l'**Assistance à la vie autonome** ou "*Ambient Assisted Living*" (AAL) ou plus généralement le domaine de la santé, que ce soit dans l'aide à l'autonomie ou dans la détection de chute. Nous sommes dans une branche plus restreinte de la domotique qui est l'habitat intelligent pour la santé. De nombreux états de l'art traitent spécifiquement des **smart homes** pour la santé (Acampora *et al.* [2013], Hossain [2014], Amiribesheli *et al.* [2015], Demongeot *et al.* [2002]), que ce soit au niveau de capteurs spécifiques, liés à la santé de la personne, à la détection de comportement à risque par exemple la dénutrition, ou à la prévention comme prévenir le risque de chute (Mubashir *et al.* [2013]). Une grande partie des travaux sur les **smart homes** pour la santé concernent les personnes âgées.

Les capteurs

Ils représentent l'élément fondamental des **smart homes**. Ils sont utilisés pour la détection de différents types d'informations, avec des niveaux de précision et d'erreur différents. Ces capteurs dépendent étroitement de l'utilisation à laquelle ils sont dé-

diés. Ainsi, un thermomètre ne sera pas très utile pour détecter des intrusions, mais pourra permettre de gérer le confort des résidents et/ou optimiser le coût énergétique du [smart home](#). Parmi ces capteurs utilisés en [smart home](#), on peut citer :

Les capteurs intégrés au mobilier Ils nécessitent une installation préalable au mobilier et permettent d'identifier les activités au sein du [smart home](#) via les interactions avec le mobilier.

A titre d'exemple, on peut trouver les capteurs de pression qui sont installés au niveau des chaises ou des lits. Ils permettent de localiser précisément la personne. Par ailleurs, pour identifier la posture de la personne, il serait possible d'augmenter le nombre et la précision des capteurs.

On peut également citer dans cette classe de capteurs les wattmètres, installés sur un appareil électrique pour mesurer sa consommation. Ils sont très utilisés dans le cas de [smart homes](#) où l'objectif est de mesurer le coût énergétique. Ces capteurs sont également utilisés pour la caractérisation du comportement des personnes via leur utilisation des appareils électriques (voir [Shimada et al. \[2005\]](#)).

Les capteurs RFID se composent de deux éléments distincts. Les tags, installés sur les objets, répondent aux requêtes émises par le second élément, en renvoyant par exemple un identifiant. Cela permet d'identifier dans un [smart home](#) les objets utilisés, ou de les localiser par rapport à des lecteurs RFID. Afin de fournir des informations utiles, il faut que l'ensemble des objets que nous souhaitons suivre soient équipés de tags. De plus, une limite de ce système est la portée d'émission et de réception des lecteurs RFID. [Yamahara et al. \[2007\]](#) et [Costa et al. \[2012\]](#) présentent des exemples d'utilisation de RFID en [smart home](#).

Si les capteurs intégrés dans l'environnement peuvent permettre une bonne connaissance des activités de la personne, leur utilisation peut s'avérer compliquée, notamment en [EHPAD](#). En effet, bien qu'une grande partie du mobilier change peu, les personnes peuvent venir avec leur propre mobilier, ou celui-ci peut être déplacé. De même, installer des tags RFID sur les biens des résidents peut être compliqué et pourrait rajouter du travail aux aidants.

Les capteurs portés L'ensemble de ces capteurs permettent d'avoir un grand nombre de données sur les activités de la personne, mais également de ne pas confondre les personnes situées dans l'environnement. Pour assurer le fonctionnement de ces capteurs, il faut néanmoins s'assurer qu'ils soient portés par la personne, ce qui peut être

très contraignant si celle-ci ne s'en assure pas elle-même ou si elle le retire régulièrement.

Parmi les capteurs portés les plus utilisés, les accéléromètres permettent de mesurer les accélérations subies par l'objet. Les résultats seront donc différents que l'accéléromètre soit porté au niveau du poignet, comme une montre, ou au niveau du cou, comme un collier. Les accéléromètres ont de multiples utilisations. Dans le cas des gérontechnologies, leur principal objectif est de détecter les chutes brusques de la personne.

Pour avoir plus d'information concernant les activités de la personne, il est possible d'utiliser des gyroscopes. Ces capteurs permettent de mesurer les changements des positions angulaires de l'objet. Ils sont souvent combinés aux accéléromètres afin de caractériser les mouvements de la personne portant le capteur.

Si les capteurs portés permettent d'acquérir de nombreuses informations sur les activités de la personne, leurs utilisations peuvent être compliquées dans le cas de l'EHPAD, où les aidants doivent veiller à ce que les capteurs soient bien portés, et fonctionnent correctement. Par exemple, la plupart des capteurs portés fonctionnent grâce à des piles ou des batteries qui doivent être suffisamment chargées pour fonctionner. De ce fait, utiliser des capteurs intégrés à l'environnement, bien que pouvant être plus compliqué à mettre en place, peut être plus simple d'utilisation, pour les utilisateurs finaux.

Un exemple de méthode utilisant des capteurs portés est décrite par [Chamroukhi et al. \[2013\]](#). Certaines méthodes utilisent d'autres capteurs en complément, comme dans l'étude menée par [Costa et al. \[2012\]](#) où une caméra est intégrée, ou dans l'étude menée par [Hodges et al. \[2009\]](#) dans laquelle les auteurs utilisent un lecteur RFID portable comme capteur porté, et imposent de placer des tags RFID sur les objets de l'environnement, comme vu dans le paragraphe précédent.

Les capteurs dans le bâti Il existe un grand nombre de capteurs intégrés dans les bâtiments. Pour les [smart homes](#) situés en lieux publics ou assimilés, par exemple des bureaux, les caméras, couplées à des méthodes de traitement de l'image, permettent d'avoir un grand nombre d'informations ([Ayers et Shah \[2001\]](#)). L'étude de [Thi et al. \[2005\]](#) est un exemple d'application de caméras pour l'identification des activités spécifiques de la personne dans un [smart home](#) de type cuisine. Plus généralement, [Chaa-raoui et al. \[2012\]](#) ont réalisé un état de l'art de la vision par ordinateur pour la détection

des activités humaines. Les caméras sont néanmoins plus difficilement acceptables dans des environnements privés, compte tenu du caractère intrusif de celles-ci.

Il existe un grand nombre de capteurs moins intrusifs que les caméras et permettant d'avoir des informations sur l'activité au sein d'une [smart home](#). Parmi les capteurs les plus utilisés, on peut citer les capteurs de type [PIR](#).

Ceux-ci utilisent un capteur pyroélectrique, constitué d'un matériau qui émet une charge électrique lorsqu'il est exposé à un rayonnement infrarouge. Le corps humain, émettant naturellement un rayonnement infrarouge qui est focalisé par une lentille de Fresnel vers le capteur pyroélectrique. Un capteur de ce type serait néanmoins sensible à énormément de perturbations, par exemple le rayonnement solaire ou des vibrations. Afin de limiter cela, deux systèmes sont montés en parallèle, les signaux émis par les systèmes se soustrayant l'un l'autre. Les éléments sont donc montés de façon à ce qu'une personne passant devant le capteur active les capteurs pyroélectriques l'un après l'autre, comme représenté par la figure 2.1, alors qu'une perturbation activera les deux cellules pyroélectriques en même temps. Grâce à cela, quand la personne franchit le faisceau de détection du capteur, elle est détectée.

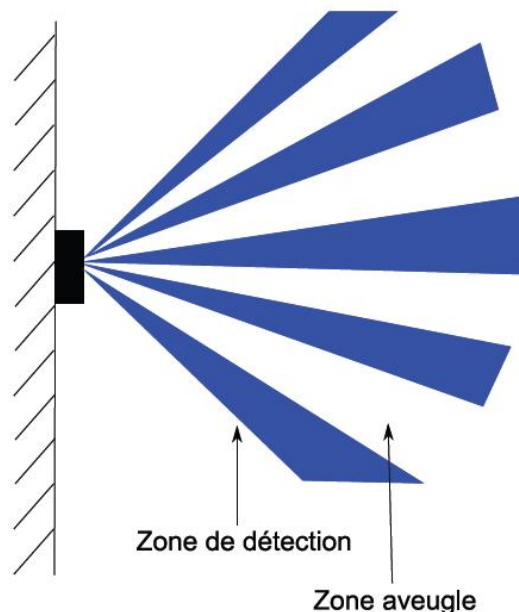


FIGURE 2.1 – Zone de détection d'un capteur PIR

Si ce type de capteur est très utilisé, notamment grâce à son faible coût et une technologie maintenant bien maîtrisée, ses capacités de détections restent limitées, détec-

tant uniquement des mouvements de chaleur dans un cône de détection précis. S'il est souvent considéré comme un capteur de présence, il s'agit davantage d'un capteur de mouvement, ce qui fait que si la personne reste immobile, lors d'une sieste, par exemple, au sein de la zone de détection, elle ne sera pas détectée, comme dans le cas d'une personne bougeant en activant les deux capteurs pyroélectriques en même temps. De plus, afin de réduire les fausses détections et augmenter la précision, le capteur est optimisé pour détecter les rayonnements infrarouges proches de ceux émis par le corps humain. Néanmoins, cela ne réduit pas le nombre de faux positifs à 0 et tout mouvement de chaleur proche de rayonnement du corps humain peut activer le capteur. Un animal ou un objet inorganique peut ainsi provoquer un mouvement de chaleur. Par exemple, une imprimante tirant un grand nombre de documents (Cook et Krishnan [2015]).

Des microphones peuvent également être installés dans un [smart home](#), et couplés à des méthodes de traitement du son. Cela permet d'identifier des sons spécifiques, notamment la voix humaine, afin de caractériser les activités au sein du [smart home](#), ou d'identifier des événements spécifiques. On trouve des exemples d'utilisation de microphones en [smart homes](#) dans les études de Brdiczka *et al.* [2009], Karaman *et al.* [2012] ou Medjahed *et al.* [2011].

Parmi les autres capteurs installés dans l'environnement et souvent utilisés dans les [smart homes](#), on peut noter les capteurs à contact magnétique, pour détecter si un ouvrant, c'est à dire une porte ou une fenêtre, est fermé ou non. Ces capteurs consistent en deux éléments, un aimant fixé sur la partie mobile et une partie électrotechnique sur la partie fixe. Ces deux éléments permettent d'agir sur un interrupteur, qui ferme un circuit quand l'aimant se rapproche de la partie électronique, ou au contraire de l'ouvrir en éloignant l'aimant. Cela permet, de façon sûre, de savoir à tout moment si la porte équipée est ouverte ou fermée.

Un autre ensemble de capteurs est composé des capteurs environnementaux. Ces capteurs ne fournissent pas directement d'informations sur l'activité de la personne, mais sur l'état du [smart home](#), via différents types de mesures, en fonction du capteur. Ainsi, un capteur, très utilisé, est le thermomètre, qui pourra permettre de réguler la température au sein de la [smart home](#). D'autres capteurs environnementaux très utilisés mesurent l'humidité ou la luminosité. D'autres capteurs environnementaux sont possibles, par exemple des capteurs permettant de mesurer la qualité de l'air via la détection de certaines particules spécifiques.

L'environnement de notre projet, c'est à dire l'[EHPAD](#), implique de fortes contraintes sur le choix des capteurs et sur la méthode de supervision. En effet, les capteurs utilisés dans le projet doivent être acceptables par des personnes âgées, pouvant avoir des faiblesses cognitives ou physiques. Il sera donc difficile d'envisager des capteurs portés, les aidants devant vérifier que les résidents ne les ont pas oubliés ou retirés.

2.1.2 Exemple de smart homes : GIS Madonah

Nous disposons dans le cadre du projet E-monitor'âge d'un [smart home](#), le GIS Madonah¹ que nous avons utilisé pour mener nos expérimentations et pour valider les approches que nous avons développées. C'est un living lab situé à l'[EHPAD](#) de Bellevue, à Bourges, dont le nom est l'acronyme de "Maintien et Assistance Diurne Ou Nocturne pour personnes Âgées et/ou Handicapées". Il s'agit d'un appartement de 100m², composé d'une pièce de vie, d'une chambre, d'une cuisine et d'une salle de bain, comme illustré par la figure 2.2. Ce living lab a été spécialement conçu pour des personnes âgées ou handicapées. Il est équipé de capteurs et de mobiliers spécifiques.

Le GIS Madonah est équipé notamment d'un ensemble de détecteurs [PIR](#), couvrant tout l'appartement et permettant de définir dans quelle pièce se trouve la personne. Au niveau de la cuisine et la salle d'eau, des capteurs de fuite d'eau permettent de couper automatiquement l'arrivée d'eau en cas de fuite. De plus, des thermomètres sont utilisés pour mesurer la température à l'intérieur de l'appartement, ainsi que dans le frigo.

Différentes solutions sont prévues pour assister le résident dans ses activités au sein du GIS Madonah. Ainsi, différents plans de travail ou étagères sont motorisés, permettant de régler leurs hauteurs. Les fenêtres et les portes sont également automatisées et peuvent être ouvertes ou fermées via des commandes, de même que le robinet de la baignoire est commandable à distance. De plus, un système de chemin lumineux permet d'allumer les lumières quand la personne se lève durant la nuit, réduisant ainsi les risques de chutes.

2.1.3 Localisation de la personne en smarthomes

Une problématique en [smart home](#) est de localiser la personne et suivre sa position. Dans le cas de [smart home](#) de type maison, avec plusieurs pièces clairement définies, la position de la personne peut être définie comme la pièce dans laquelle se trouve la personne. Si le [smart home](#) ne peut pas être défini comme plusieurs pièces, par exemple

1. GIS Madonah : <http://www.bourges.univ-orleans.fr/madonah/index.php>



FIGURE 2.2 – GIS Madonah

des bureaux avec de vastes pièces ou dans notre cas, une chambre d'EHPAD, il peut être intéressant de définir plus précisément la position de la personne. Pour cela, il est possible de définir la position de la personne via des capteurs spécifiques ou par des méthodes spécifiques, comme l'approche proposée par [Thi et al. \[2005\]](#), positionnant la personne dans une surface de $1m^2$ via un ensemble de 4 caméras. Une autre approche, proposée par [Andries et al. \[2015\]](#), permet de localiser et suivre les personnes, mais également de les reconnaître, ainsi que les objets, en utilisant des capteurs de pression au niveau du sol.

Dans le cadre du projet et de nos expérimentations, les données des capteurs [PIR](#) sont traitées grâce à la méthode décrite par [Amri et al. \[2015\]](#). Il s'agit d'une méthode de localisation de la personne dans un environnement intérieur grâce à des capteurs fournissant une zone où se situe la personne. Cette méthode prend en compte la dernière zone connue où se situait la personne et permet de filtrer les données aberrantes. Pour cela, elle se base sur la méthode de la q-relaxation ([Amri \[2015\]](#), [Jaulin \[2001\]](#)), visant à éliminer le minimum de données pour garantir une solution cohérente (voir figure [2.3](#)).

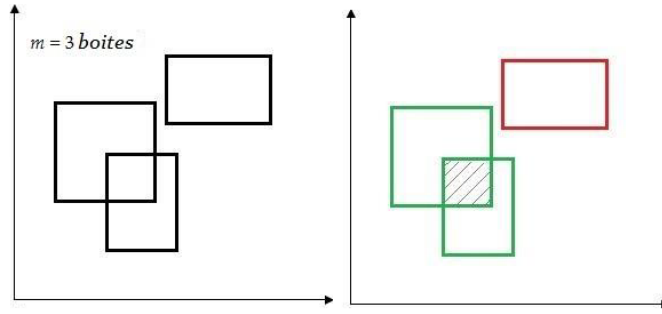


FIGURE 2.3 – Rejet des données aberrantes. Seuls les données en vert sont gardés, les données en rouge étant aberrantes.

Cette méthode fonctionne en deux étapes. Dans une première étape, la position de la personne est estimée grâce à sa dernière position connue et à un modèle de déplacement aléatoire. Ce dernier consiste à considérer que la personne peut se déplacer dans toutes les directions à la vitesse de 1 m/s à partir de sa dernière position connue. En effet, compte tenu de la typologie de l'environnement étudié, si on considère $(x(t), y(t))$ la position de la personne à un instant t , Δt le temps minimum entre deux détections capteurs, $v(t)$ la vitesse de la personne et θ la direction de la personne. La position de la personne à $t + 1$ peut alors être calculée avec la formule 2.1.

$$\begin{aligned} x(t+1) &= x(t) + \Delta t \cdot v(t) \cdot \cos(\theta) \\ y(t+1) &= y(t) + \Delta t \cdot v(t) \cdot \sin(\theta) \end{aligned} \quad (2.1)$$

Les données des capteurs PIR ne permettent de définir, ni la direction de la personne, ni sa position exacte (x, y) , et se limitent à l'information que la personne se situe dans sa zone de détection. On admet que cette zone peut être considérée comme un rectangle, défini par $(x_{\min}, x_{\max}, y_{\min}, y_{\max})$. Le modèle de marche aléatoire est donc défini dans ce cas par la formule 2.2.

$$\begin{aligned} x_{\min}(t+1) &= x_{\min}(t) - \Delta t \cdot v_{\max}(t) \\ y_{\min}(t+1) &= y_{\min}(t) - \Delta t \cdot v_{\max}(t) \\ x_{\max}(t+1) &= x_{\max}(t) + \Delta t \cdot v_{\max}(t) \\ y_{\max}(t+1) &= y_{\max}(t) + \Delta t \cdot v_{\max}(t) \end{aligned} \quad (2.2)$$

Dans une seconde étape, le nombre minimal q de données aberrantes est calculé grâce à la méthode de la q -relaxation. Pour cela, la valeur de la variable q est initialisée à 0 puis incrémentée jusqu'à ce que la solution soit atteinte. Si pour la solution trouvée il existe au moins une intersection non vide à $m - q$ données, où m l'ensemble des données, la solution est alors l'ensemble contenant toutes les intersections trouvées.

Afin d'optimiser ce calcul, une variante (Amri [2015], Jaulin [2001]), donnant le même résultat théorique est utilisée. Si l'intersection de l'ensemble des données est un ensemble non nul, alors le résultat est cette intersection. Sinon, nous considérons pour chaque zone, par dimension, le nombre de zones couvrantes. Pour cela, l'ensemble des données est parcouru. Les coordonnées minimums des données sont affectées d'une valeur +1 et les coordonnées maximums sont affectées d'une valeur -1, comme indiqué sur la figure 2.4. Le résultat est alors défini à partir des zones dont la valeur est la plus grande pour chaque dimension, c'est à dire que la borne inférieure est définie comme la borne inférieure minimum parmi les zones dont la valeur est la plus grande, et réciproquement, la borne supérieure est définie comme la borne supérieure la plus grande parmi les bornes supérieures des zones dont la valeur est la plus grande.

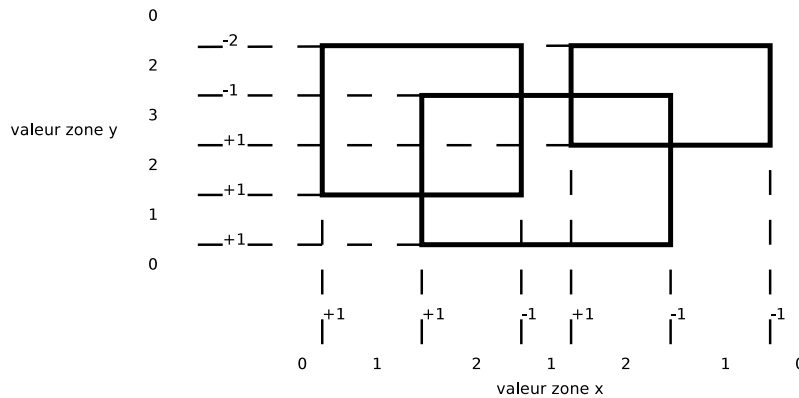


FIGURE 2.4 – Exemple de q-relaxation

Cette méthode ne nous permet de pas d'exploiter directement les données capteurs mais de connaître, à chaque pas de temps, une zone de l'environnement où la personne se situe, comme illustré à la figure 2.5. À partir de cette zone de localisation, il est possible de générer un vecteur d'observation à chaque pas de temps. Pour cela, on peut considérer différentes zones de la pièce, en fonction des résultats de la méthode de localisation et considérer, en fonction de ces zones, les vecteurs d'observations. Plusieurs solutions sont possibles pour cela. Une première solution est que pour chaque zone, l'élément du vecteur prend la valeur si la zone de localisation de la personne est couvrante, et 0 sinon. Une autre possibilité est de considérer le pourcentage de la zone recouverte par la zone de localisation de la personne.

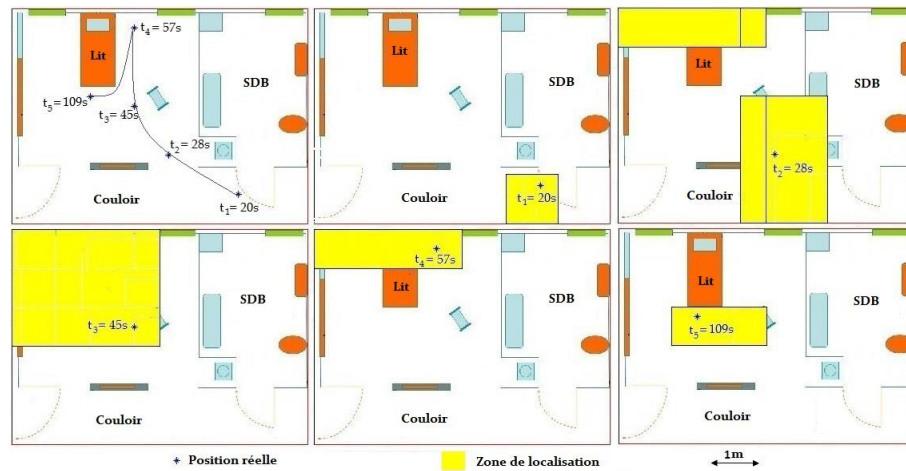


FIGURE 2.5 – Exemple de résultat de localisation de la personne. À chaque pas de temps, la personne est nécessairement dans la zone jaune.

2.2 Apprentissage du comportement

Plusieurs études portant sur la reconnaissance de l'activité ([Chaaroui et al. \[2012\]](#)) ou du comportement humain ([Vacher et al. \[2009\]](#)) ont été menées au cours de cette décennie. [Cook et Krishnan \[2015\]](#) présentent un état de l'art et une description des différentes méthodes et types de capteurs utilisés actuellement dans le cadre de la découverte et la reconnaissance du comportement humain.

Dans la littérature, de nombreuses études concernent la modélisation du comportement. Parmi celles-ci, certaines s'intéressent à modéliser le comportement d'une foule ([Wang et al. \[2011\]](#), [Mehran et al. \[2009\]](#)), ou aux comportements d'utilisation de services informatiques ([Tan et Xi \[2008\]](#)), ou encore à la navigation pour la surveillance maritime ([Brax et al. \[2012\]](#)). Nous considérons dans notre étude uniquement le modèle de comportement de la personne.

De plus, dans le cadre de ce travail, nous souhaitons modéliser spécifiquement le comportement de la personne en [EHPAD](#). La nature des [EHPAD](#) fait que certaines spécificités de comportement existent par rapport à la population globale. En effet, ces établissements sont voués, par définition, à accueillir des personnes âgées dépendantes. Le caractère dépendant des résidents en [EHPAD](#) peut être d'origine physique ou mentale mais le vieillissement des personnes implique en elle-même des spécificités. Ainsi, la gériatrie est une spécialité médicale qui permet de traiter les particularités des personnes de plus de 60 ans. Ainsi, la gérontopsychiatrie et la psychogériatrie ([Cohen et al. \[2014\]](#)) se développent, afin d'adapter les soins. Pour cela, des études sont menées, notamment sur les pathologies chez les résidents d'[EHPAD](#).

En plus de l'augmentation de la population présentant de multiples pathologies, des études ([Perrin-Haynes et al. \[2012\]](#)) ont montré que le pourcentage de personnes présentant des pathologies psychiatriques est important. Par exemple, en 2011, 42% des résidents d'EHPAD présentaient des syndromes démentiels, 36% des états dépressifs, et 16% des psychoses ou délires. De plus, une pathologie donnée peut provoquer des symptômes très différents. Les démences fronto-temporales, associées à des troubles du comportement et du langage, qui regroupent différentes pathologies, ne se manifestent pas de la même façon d'une personne à l'autre, avec des symptômes associés très hétérogènes, comme l'apathie, la désinhibition ou encore les stéréotypies comportementales, verbales ou motrices ([Ollat \[2002\]](#)). Si ces symptômes impliquent de grandes variations et des changements importants, d'autres symptômes influençant également le comportement sont associés aux démences fronto-temporales. De plus, on peut retrouver un certain nombre de ces symptômes comportementaux dans d'autres pathologies, par exemple la maladie d'Alzheimer ([Benoit et al. \[2003\]](#)). Les symptômes associés liés aux démences fronto-temporales sont ainsi très différents, contrairement à des pathologies définies à partir des symptômes, comme les dépressions mixtes ([Paris \[2012\]](#)), caractérisées par un épisode dépressif majeur et au moins trois symptômes hypomaniaques. D'autres symptômes, comme les comportements d'utilisation ([Lhermitte \[1983\]](#)) sont présents en EHPAD et affectent fortement le comportement des personnes.

Ainsi, afin de définir si un comportement est normal ou pas en EHPAD, il est nécessaire de prendre en compte les habitudes de la personne. Par exemple, si le système détecte que la personne n'est pas présente dans sa chambre au cours de la nuit, et que ce n'est pas habituel, alors une alerte doit être émise. Par contre, si la personne a l'habitude de déambuler durant la nuit, alors le système ne doit pas émettre d'alerte, la déambulation étant alors considérée comme normale. Par contre, des situations comme les chutes ne peuvent jamais être considérées comme habituelles. Ainsi, [Tran et al. \[2010\]](#) proposent une classification des comportements anormaux en fonction de la différence entre le comportement anormal et le comportement habituel de la personne.

La détection d'un comportement anormal, ou de l'évolution du comportement de la personne doit donc se faire via le prisme d'un comportement habituel. Le comportement habituel du résident est défini comme l'ensemble des activités que la personne peut effectuer de façon régulière au cours du temps. Cette définition se rapproche de la définition du comportement de [Charaoui et al. \[2012\]](#) où un comportement est défini

comme une compréhension à haut niveau sémantique de la situation.

La modélisation du comportement dépend fortement des données disponibles sur celui-ci. Les conditions d'acquisition d'informations étant très variables, tout dépend si l'on considère un environnement [smart home](#) ou non. Les différences entre ces deux aspects sont détaillées dans les sous-sections suivantes.

2.2.1 Hors smarthomes

La modélisation du comportement humain en dehors des [smart homes](#) implique des problèmes spécifiques, notamment au niveau de l'acquisition des données. En effet, il n'est pas possible d'avoir des données issues de capteurs installés au sein de l'environnement mais des données issues de capteurs portés par la personne peuvent être suffisantes. Toutefois, ces derniers doivent avoir une autonomie suffisante afin de pouvoir enregistrer et récupérer les données sans imposer de contraintes supplémentaires à la personne. Entre autres, le capteur est censé transmettre les données en permanence où que soit la personne, il doit être capable d'enregistrer l'ensemble des données jusqu'à ce que celles-ci puissent être récupérées.

Un exemple de l'apprentissage du comportement de la personne en dehors des [smart homes](#) est présenté par [Azam et al. \[2012\]](#). Dans cette étude, les auteurs font appel aux réseaux sans fil "Bluetooth". Le comportement de la personne est défini en fonction des réseaux Bluetooth qui sont détectés régulièrement lors des déplacements de la personne grâce à des méthodes de pattern matching. Si les habitudes d'une personne peuvent être considérées comme ses habitudes de mouvement, en fonction des endroits où la personne a l'habitude d'aller, cela implique que la personne se déplace suffisamment, mais également que les lieux visités sont localisés à proximité de réseaux sans fil détectables.

2.2.2 En smarthomes

Si le comportement de la personne peut être défini en dehors des [smart homes](#) comme ses habitudes de déplacement, il est possible d'acquérir davantage de données sur les activités de la personne dans un environnement équipé de capteurs. En considérant cet aspect, il est possible de distinguer, dans un premier temps, entre les études basées sur l'utilisation explicite des activités de la personne et celles cherchant à caractériser le comportement de la personne sans activité spécifique. Ensuite et dans un second temps, il est possible de s'intéresser aux méthodes caractérisant l'évolution du comportement.

ADL, iADL et leurs utilisations dans la détection des activités

Définition des ADL et iADL. [Katz et al. \[1963\]](#) décrivent l'échelle de Katz comme un outil qui sert à définir le niveau de dépendance d'une personne. Pour cela, les auteurs font appel aux [Activités de la vie quotidienne](#) ("*Activities of Daily Living*" (ADL)). Ils définissent ces dernières comme un ensemble d'activités qu'une personne peut faire chaque jour. Ces activités sont : se laver, s'habiller, aller aux toilettes, se mouvoir, manger. Un élément supplémentaire pris en compte est l'incontinence. Les [ADL](#) permettent d'avoir une échelle commune, fixe et facilement mesurable par un praticien afin d'évaluer le niveau de dépendance de la personne, en fonction des problèmes que la personne rencontre dans sa vie de tous les jours.

Dans le cadre de la domotique, il est très intéressant de pouvoir identifier les [ADL](#) des personnes présentes grâce aux capteurs. En effet, cela peut servir différents buts, que ce soit l'identification de situations anormales, comme dans notre cas, l'assistance à la personne, ou encore la mesure de son niveau de dépendance.

À partir des [ADL](#), [Lawton et Brody \[1969\]](#) ont développé ce qu'ils ont nommé les [activités de la vie quotidienne instrumentalisées](#) ("*Instrumental Activities of Daily Living*" (iADL)). Ces [iADL](#) permettent non seulement de mesurer le niveau de dépendance de la personne mais également de prendre en compte l'aspect cognitif de la dépendance par le biais de la prise en compte d'activités comme : savoir gérer son argent, téléphoner, faire le ménage, prendre ses médicaments, prendre des transports...

Méthodes détectant les ADL. Un grand nombre d'études se sont basées sur les [ADL](#) ([Cook et Krishnan \[2015\]](#), [Chaaroui et al. \[2012\]](#)). Certes, connaître l'activité effectuée par la personne, et à quel moment, est une information très importante ; et donc, définir les [ADL](#) effectuées par la personne, représente en lui-même un objectif qui a été considéré par un grand nombre de travaux de recherche.

Un premier exemple de ce type de méthodes est proposé par [Brdiczka et al. \[2009\]](#). Dans cette méthode, les auteurs proposent d'utiliser un modèle pour détecter chaque activité de la personne de manière spécifique, en utilisant des données issues de caméras et de microphones. Ils déterminent dans un premier temps la posture, la vitesse et la distance à une table de chaque personne grâce à des méthodes de vision par ordinateur. Ces informations sont combinées avec les données sonores afin de reconnaître un ensemble de six activités, deux individuelles et quatre collectives. Dans l'étude de [Chung et Liu \[2008\]](#), les auteurs définissent différentes activités de haut niveau grâce à la localisation spatiale de la personne et à cinq activités physiques définies grâce à des

caméras. Ces cinq activités de bas niveau permettent, en prenant également en compte la position et les données temporelles, notamment les durées, de définir des comportements comme : aller aux toilettes, s'asseoir sur une chaise pour regarder la télévision, faire une promenade et tomber du lit. Un grand nombre d'activités peuvent ainsi être définies, à partir du moment où les activités de bas niveau et les données temporelles permettent de les discriminer.

Krishnan et Cook [2014] proposent de classifier les activités de la personne en respectant la contrainte de résultats en temps réel. Cette méthode nécessite une étape de labellisation des données.

Dans leur méthode, **Medjahed et al. [2011]** proposent une approche hybride, basée sur différents types de capteurs, incluant des capteurs portés et des microphones. Chaque type de capteurs est traité séparément avant que les données ne soient fusionnées et synchronisées afin de détecter l'activité courante dans un ensemble de vingt activités différentes. Si l'ensemble des activités possibles est important, cette méthode ne permet néanmoins pas d'assurer l'exhaustivité de la liste, y compris dans le cas de résidents ayant des difficultés cognitives.

Kasteren et al. [2010] étudient différentes représentations et traitements des données capteurs afin d'apprendre, à partir de données labellisées, à reconnaître différentes activités. Il est à noter que les données ont été acquises en situation réelle, avec 10 à 14 activités différentes labellisées. Parmi ces labels se trouve l'activité "autres" qui représente entre 9.1% et 82.9% du temps de l'expérience.

Liu et al. [2007] utilisent des méthodes de vision par ordinateur pour détecter la posture et l'activité de la personne. Ces données permettent ensuite de détecter les **ADL**. Chaque nouvelle activité à détecter nécessite un système spécifique.

Le système proposé par **Noury et al. [2012]** permet de définir l'activité la plus probable parmi huit activités préalablement apprises. Pour cela, le système utilise des capteurs portés et des **PIR**. Quant à **Fang et al. [2014]**, ils utilisent les réseaux de neurones afin d'apprendre à identifier des **ADL** à partir de données labellisées.

Méthodes utilisant les ADL Les méthodes du paragraphe précédent ont pour objectif d'identifier les **ADL** de la personne. L'identification des activités est également utilisée par des auteurs comme moyen, afin d'atteindre d'autres objectifs liés à la personne.

Un premier exemple de ce type de méthodes est présenté par **Le et al. [2007]**. Ils utilisent une méthode statistique pour déterminer les activités de la personne grâce à la détection des états stationnaires ou en mouvement de la personne, en utilisant uniquement des capteurs **PIR**. Dans un premier temps, les données sont traitées afin de définir les états mobiles ou immobiles de la personne en fonction de la durée entre

deux activations, et du nombre d'activations. Ce traitement est effectué pour chaque pièce. En effet, les activités en fonction des différentes pièces étant fortement différentes, la discrimination entre les états mobile et immobile est différente en fonction des pièces. L'état de mobilité de la personne peut ensuite être traité afin de détecter différentes activités de la personne, et ainsi son niveau de dépendance. Cette méthode est intéressante du fait que l'auteur utilise peu de données *a priori*.

L'objectif de l'étude décrite par [Huang et al. \[2014\]](#) est de détecter des comportements anormaux. Ils utilisent pour cela un ensemble de caméras afin de reconnaître l'activité courante dans un ensemble d'activités grâce aux positions et aux déplacements de la personne. La durée de chaque activité est ensuite calculée et différentes distances sont par la suite utilisées afin de déterminer un comportement anormal en fonction du temps passé à chaque activité au cours de la journée.

Il est également possible de se concentrer sur des éléments spécifiques du comportement. Ainsi, [Chen et al. \[2013\]](#) cherchent à détecter des situations anormales, définies par les experts, à partir de routines de personnes âgées. Les éléments détectés sont spécifiques mais permettent de considérer des activités spécifiques de la personne, comme : aller aux toilettes plus de 4 fois dans la nuit, ne pas ouvrir le frigo pendant 24 heures, laisser le frigo ouvert pendant deux heures, ou ne pas ouvrir le frigo à des heures spécifiques. Toutefois, ces éléments sont d'une part fortement liés aux capteurs, mais également globalisés à l'ensemble de la population : aucune spécificité de la personne n'est prise en compte.

Méthode de modélisation sans activité

Plusieurs inconvénients des systèmes basés sur la détection des [ADLs](#) doivent être pris en compte. Un premier élément limitant l'utilisation des méthodes basées sur les [ADLs](#) est la nécessité d'utiliser des données labellisées. Néanmoins, il n'est pas toujours possible d'utiliser des données labellisées, compte tenu de la difficulté de labelliser de large quantité de données. Une solution est de labelliser des données spécifiques et de les considérer pour l'ensemble des personnes. Cela implique que le système ne prend pas en compte les spécificités de chaque résident, notamment dans le cas de personnes avec des difficultés cognitives. En effet, les [ADLs](#) sont un ensemble d'activités considérées comme normales pour un individu en pleine possession de ses capacités physiques et cognitives. Ceci n'est pas le cas de l'étude menée dans ce manuscrit, vu le niveau de dépendance des personnes résidant en [EHPAD](#). Par exemple, il peut être habituel pour un résident de déambuler, alors que la déambulation ne fait pas partie des [ADL](#).

De plus, il faut définir quelles **ADLs** labelliser afin d'être le plus représentatif. Si le nombre d'**ADLs** est très limité, il est possible de prendre en compte d'autres activités. Par exemple, la liste ATUS² qui considère plus de quatre cents activités différentes. Aucune base de données ne permet de prendre en compte autant d'activités et d'apprendre à les identifier efficacement à partir de données capteurs. À partir du moment où la liste ne contient pas l'ensemble des activités, il est nécessaire d'ajouter un label aux activités inconnues. Une solution fréquemment utilisée consiste à ajouter une activité "autres", qui peut correspondre à une grande partie du comportement de la personne.

Pour détecter des évolutions du comportement, comme par exemple une réduction du temps de déambulation pour quelqu'un ayant l'habitude de déambuler, le système de supervision doit détecter les **ADLs**, mais également l'ensemble des autres activités du résident. De ce fait, utiliser un système se basant uniquement sur les **ADLs**, dans un environnement où les résidents peuvent présenter des pathologies, notamment des démences séniles, n'est pas pertinent.

Parmi les méthodes qui ne sont pas basées sur la détection explicite des activités de la personne, un premier ensemble se compose de méthodes étudiant les zones au sein de la **smart home**.

Par exemple, l'approche développée par **Noury et Hadidi [2012]** consiste à étudier les transitions entre sept zones non recouvrantes et clairement identifiées dans la maison afin de modéliser le comportement. La présence de la personne dans une zone est connue grâce à des capteurs **PIR**. Les transitions entre les états peuvent être modélisées, grâce à sept modèles, chacun correspondant à une période de la journée. Ces modèles servent à déterminer un modèle d'activité de référence et chaque déviation de ce modèle peut indiquer une perte d'autonomie de la personne.

Les méthodes basées sur des zones spécifiques nécessitent que l'environnement soit suffisamment grand pour pouvoir définir des zones dont l'utilisation est clairement identifiée. Dans le cadre d'une maison, il est ainsi possible de différencier les différentes pièces. Dans notre cas d'étude, l'environnement se compose d'une chambre et d'une salle d'eau. En partant du postulat que les capteurs permettent de différencier avec certitude la pièce où se trouve la personne, ce qui n'est réellement pas toujours le cas, l'étude du comportement basée sur trois zones distinctes est extrêmement limitée.

Un autre type de méthodes étudie le niveau d'activité de la personne, sans s'intéresser à quelle activité spécifique la personne effectue.

2. American Time Use Survey, Bureau of Labor Statistics <http://www.bls.gov/tus/lexicons.htm>

L'objectif de l'étude menée par [Endelin et al. \[2013\]](#) est de définir des comportements anormaux de haut niveau, grâce à des capteurs de mouvements. Pour cela, les données sont traitées sur un pas de temps de 60 secondes pour définir dans chaque pièce l'état qui peut être mobile, immobile ou absent. Ces états correspondent à l'activité de la personne. Il est ensuite possible d'utiliser les données acquises durant deux semaines afin de définir statiquement si le comportement actuel de la personne correspond à son comportement habituel. Un comportement anormal de la personne correspondant alors à une variation de son niveau d'activité physique.

L'objectif de l'étude menée par [Botia et al. \[2012\]](#) est de détecter des situations d'alertes grâce aux périodes d'activités ou d'inactivités anormales, en se basant sur les règles d'un système expert. L'approche proposée est commune à l'ensemble des personnes, sans aucune forme de personnalisation.

[Poujaud et al. \[2008\]](#) étudient le temps entre deux détections de capteurs PIR afin d'identifier des périodes d'inactivités anormales. Les temps probables d'inactivités sont calculés pour chaque pièce et en divisant la journée en quatre périodes distinctes, le niveau d'activité de la personne dépendant de l'heure au cours de la journée. Si cette méthode ne permet pas de détecter spécifiquement des comportements anormaux, elle permet de détecter une réduction de l'activité de la personne à un moment donné sans aucun *a priori* sur les activités de la personne.

Un autre ensemble de méthodes caractérise le comportement de la personne à partir des données capteurs, sans chercher à donner un sens aux données.

Ainsi, la méthode [Chua et al. \[2011\]](#) considère les données capteurs comme les lettres d'un alphabet, et utilise une méthode d'apprentissage non-supervisé afin de trouver des mots qui apparaissent régulièrement. Le dictionnaire des mots ainsi définis va alors correspondre aux activités régulières de la personne. L'activité courante de la personne peut être ainsi déterminée en calculant la distance entre le mot actuel et ceux du dictionnaire.

Le système développé par [Cook et al. \[2013\]](#) combine la découverte puis la reconnaissance des activités, avec des contraintes de temps réel. La découverte d'activité se base sur la détection de motifs dans les données, tandis que la reconnaissance utilise les [Support Vector Machines \(SVM\)](#).

[Shimada et al. \[2005\]](#) cherchent à détecter des comportements anormaux à partir du comportement habituel de la personne, en se basant sur le fonctionnement des appareils électriques au sein de la maison. En effet, les activités de la personne sont associées au fonctionnement des appareils. Un réseau de neurones peut ainsi être entraîné sur ces données, permettant de détecter un comportement anormal. Néanmoins, cela implique que l'ensemble des activités qui ne sont pas liées aux appareils électriques ne

sont pas détectées, par exemple la chute d'une personne qui n'a pas allumé la lumière la nuit n'est pas détectée. De plus, dans le cas de personnes présentant des défaillances cognitives, l'utilisation d'appareils électriques peut ne pas suivre de modèle.

Un dernier type de méthodes ne considèrent pas directement la personne, mais des groupes de personnes. Ce type cherche à classer les groupes d'individus en fonction de différents paramètres pour ramener l'étude à l'évolution de ces groupes et à celle des personnes au sein de ces groupes (Rammal [2010]).

En conclusion, différents type de méthodes existent dans la littérature pour modéliser le comportement de la personne. Néanmoins, les comportements des personnes changent au cours du temps. Si le problème de modéliser le comportement de la personne a été fréquemment étudié dans la littérature, le problème de la détection des évolutions l'a été rarement.

Évolution du comportement

Une évolution du comportement correspond à un changement dans les habitudes de la personne au cours d'une période plus ou moins longue. Ces changements peuvent être de différentes natures. Il peut s'agir de l'apparition de nouveaux comportements (la personne commence à déambuler durant la nuit), ou au contraire la disparition d'anciens comportements (la personne ne sort plus à un moment donné).

Il peut également s'agir d'un changement dans un ancien comportement (heure de lever devenue plus tardive). Si un changement de comportement exceptionnel n'est pas forcément significatif, ce même changement répété au cours du temps peut indiquer une évolution de l'état global de la personne et de ses capacités. Les soignants doivent donc être informés de cette évolution, afin qu'ils puissent déterminer s'il s'agit d'une évolution normale pour la personne ou si cette évolution indique une pathologie comme par exemple dans le cas de changement du rythme du sommeil (Zhou *et al.* [2012]). Si le système doit détecter des dérives lentes du comportement, son objectif n'est pas de qualifier la normalité de cette évolution, mais d'en informer les aidants. Ces aidants sont en contact direct avec les résidents et peuvent juger de la pertinence de l'alerte en fonction des informations extérieures au système, comme dans le cas de la prise d'un nouveau médicament ou le changement de rythme extérieur ou encore à cause de certaines des conditions spécifiques...

Si de nombreuses études s'intéressent à la détection des ADL ou des comportements anormaux, peu d'entre elles se sont intéressées à la détection des changements

de comportement de la personne. Il s'agit dans certains cas d'un élément complémentaire à l'étude menée. Par exemple, étudier la perte d'autonomie peut se faire par une évolution du comportement de la personne, comme dans les travaux de [Noury et Haddidi \[2012\]](#).

Plusieurs de ces méthodes sont basées sur l'étude des mouvements de la personne, qui sont considérés comme un indicateur potentiel du changement de comportement ou d'état de la personne. Une étude très explicite a été présentée par [Hayes *et al.* \[2007\]](#). Cette étude utilise un capteur porté afin de définir la vitesse moyenne de déplacement de la personne, ainsi que de la différencier des autres personnes présentes dans l'environnement.

Un autre exemple est le système décrit par [Hayes *et al.* \[2004\]](#), où les auteurs déterminent la vitesse de déplacement d'une personne vivant seule chez elle. Pour cela, ils utilisent trois capteurs [PIR](#) avec un angle de détection réduit dans une zone en couloir, ce qui réduit les déplacements possibles de la personne. Ces capteurs permettent de déterminer la vitesse moyenne de déplacement de la personne.

Parmi les études sur la détection de l'évolution du comportement, on peut citer le travail de [Cook \[2006\]](#) qui propose un système de monitoring et d'assistance pour les personnes âgées. Les objectifs de ce système se résument à : l'identification des tendances de style de vie, la détection des anomalies dans les données actuelles et de la mise en place d'un système d'appel d'assistance. Le système est spécialement conçu pour l'assistance aux personnes âgées ou ayant des difficultés.

Si différentes méthodes permettent d'étudier le comportement de la personne en [smart homes](#), ces méthodes sont dépendantes des données utilisées. Si l'état de l'art met en avant l'importance de prendre en compte l'ensemble du comportement, l'étude doit également prendre en compte les méthodes utilisées.

2.3 Approche de la modélisation

Différentes méthodes peuvent être utilisées pour la modélisation du comportement de la personne en [smart home](#). Ces méthodes sont présentées dans la sous-section suivante.

2.3.1 Méthodes utilisées

Méthodes logiques

Un certain nombre de méthodes utilisent les méthodes logiques afin de détecter des situations anormales. Par exemple, la méthode décrite par [Chen et al. \[2013\]](#). Elle se base sur la détection de 7 événements, définis par les experts et détectés par 6 capteurs afin de caractériser la routine de la personne chez elle. Les situations détectées sont : se lever avant 9h30, se lever après 9h30, aller aux toilettes plus de 4 fois dans la nuit et les conditions d'ouverture du frigo. Plus précisément, que le frigo ne soit pas ouvert pendant 24 heures, ou spécifiquement à l'heure du petit-déjeuner, du déjeuner et du dîner, ou au contraire qu'il reste ouvert pendant deux heures. La dernière situation étant que sa température soit supérieure à 15°C.

Une autre approche utilisant la logique est décrite par [Merico et al. \[2010\]](#). Cette approche utilise les méthodes logiques afin de définir, grâce aux données de capteur, si différentes ADL ne sont pas effectuées, une alerte sera émise. De même, [Liu et al. \[2006\]](#) utilisent une méthode basée sur des règles logiques afin d'estimer l'état du smart home pour pouvoir lancer des alertes.

Logique floue

La logique floue combine les avantages des méthodes logiques aux méthodes probabilistes, afin de gérer les incertitudes, contrairement à un système complètement déterministe permettant ainsi d'avoir un système plus robuste.

La logique floue peut donc être utilisée afin de synchroniser un réseau de capteurs multi-modal et déduire quelle est l'activité de la personne au sein d'une liste. Par exemple, la méthode décrite par [Medjahed et al. \[2011\]](#) permet de reconnaître un ensemble de 20 activités au sein d'un smart home, notamment grâce à des capteurs portés et des microphones. La méthode proposée par [Noury et al. \[2012\]](#) permet de détecter l'activité la plus probable parmi une liste de huit activités. La logique floue permet pour cela de prendre en compte différentes connaissances *a priori* comme l'heure ou la localisation normale pour effectuer une activité.

Méthode du Web Sémantique

Basée sur un raisonneur sémantique, les méthodes du Web Sémantique ont été adaptées aux smart homes. Par exemple, [Botia et al. \[2012\]](#) présentent un système générant des alertes en fonction des périodes d'activité ou d'inactivité anormales. Le caractère anormal d'une période est défini en fonction de règles établies par les experts. D'autres systèmes utilisent les méthodes du Web Sémantique en smart home, comme

ceux proposés par [Tiberghien et al. \[2012\]](#), qui s'intéressent particulièrement au choix du raisonneur, ou le système proposé par [Chen et al. \[2012\]](#).

Méthodes statistiques

Les méthodes statistiques sont utilisées pour détecter des variations du comportement habituel par rapport à une moyenne habituelle. Ces méthodes peuvent se baser sur les détections, sans s'intéresser spécifiquement aux activités de la personne. Par exemple, [Endelin et al. \[2013\]](#) considèrent les situations anormales comme des variations du niveau d'activité de la personne. Ce niveau d'activité est défini pour chaque pièce en fonction des états mobile, immobile ou absent sur une période donnée. Les méthodes statistiques peuvent également être utilisées afin de détecter des activités spécifiques. Ainsi, la méthode décrite par [Le et al. \[2007\]](#) permet de détecter des activités spécifiques statistiquement grâce aux séquences de mobilité et d'immobilité dans chaque pièce.

Pattern matching

En considérant le comportement de la personne comme les activités que la personne fait régulièrement, il est possible de détecter ces activités régulières qui vont servir de modèle de comportement habituel de la personne. Détecter ces répétitions dans une séquence de données est connu sous le nom de problème de pattern matching.

L'approche proposée par [Chua et al. \[2011\]](#) consiste à considérer les détections des capteurs comme les lettres d'un alphabet. La détection des activités régulières de la personne sont assimilés à des mots qui apparaissent régulièrement. Cela permet de créer un dictionnaire des comportements habituels et ainsi chaque activité est identifiée à partir de la distance entre les mots actuels et ceux du dictionnaire.

Réseau de neurones

Les réseaux de neurones sont des modèles inspirés des neurones biologiques et très utilisés dans le domaine de l'Intelligence Artificielle, et plus particulièrement pour les problèmes de classification. Un des avantages des réseaux de neurones est qu'en dehors de la typologie (nombre de couches et de neurones par couches et choix des fonctions), aucune connaissance *a priori* du problème n'est nécessaire pour apprendre, permettant ainsi d'éviter des biais dû au choix du modèle.

Dans le cadre des [smart homes](#), les réseaux de neurones peuvent être utilisés pour modéliser les habitudes de la personne. Par exemple, la méthode présentée par [Shimada et al. \[2005\]](#) consiste à apprendre grâce à un réseau de neurones, le fonctionne-

ment habituel des appareils électriques dans un [smart home](#). Ce réseau de neurones peut ensuite être utilisé afin de détecter des utilisations inhabituelles des appareils électriques en indiquant un comportement inhabituel.

Il est également possible d'utiliser les réseaux de données pour classifier des [ADLs](#) spécifiques, et ainsi apprendre à les identifier ([Fang et al. \[2014\]](#)).

Modèles de Markov

Les méthodes basées sur différents modèles de Markov sont très utilisées dans le cadre des [smart homes](#). Différents modèles et leurs utilisations dans le cadre des [smart homes](#) sont décrits dans le chapitre 3.

Les modèles de Markov sont utilisés de différentes manières dans les [smart homes](#), que ce soit pour modéliser le comportement global de la personne, une activités spécifiques ou même une caractéristique précise du comportement de la personne. L'utilisation des méthodes de Markov implique peu de contraintes et présente de nombreux avantages, notamment la robustesse vis-à-vis du caractère incertain des données.

Méthodes hybrides

Différentes études combinent plusieurs méthodes afin de bénéficier de leurs différents avantages. Par exemple, l'approche décrite par [Cook \[2006\]](#) combine des méthodes de data minig, de compression, de Markov, de calcul de distance entre pattern et de logique, via un système multi-agents. L'ensemble de ces méthodes permet d'identifier les tendances de style de vie, de détecter des anomalies dans les données actuelles et de mettre en place un système d'alerte en réduisant les *a priori*.

Un autre exemple est présenté par [Cook et al. \[2013\]](#). Leur objectif est la détection des activités d'une personne avec la contrainte du temps réel. Pour cela, ils considèrent deux problèmes distincts : la reconnaissance et la découverte d'activité. Pour la découverte d'activité, l'auteur utilise des méthodes de détection de pattern dans les données, tandis que pour la reconnaissance, il utilise les [SVM](#).

La méthode décrite par [Yuan et Herbert \[2013\]](#) a pour objectif de détecter si la situation actuelle est une situation normale, anormale ou d'urgence, en utilisant la logique floue et les k-means.

2.3.2 Discussion sur les méthodes

L'environnement intelligent, dit [smart homes](#) étant très présent dans la littérature, un grand nombre de méthodes y ont été appliquées, et présentant à la fois de multiples avantages et inconvénients.

Les méthodes à base de règles ou de connaissances, c'est à dire les méthodes basées sur la logique, y compris la logique floue ou les méthodes du Web Sémantique ne sont pas utilisées pour modéliser le comportement humain mais pour émettre des alertes. Ils se basent sur des ensemble de règles, ce qui implique que le système est bien maîtrisé, mais qui reste très peu personnalisable.

Par exemple, déterminer une période de lever normale est peu pertinente dans le cas d'une personne déambulante ou présentant des troubles du sommeil. De plus, excepté la logique floue, les autres logiques sont des méthodes déterministes, peu robustes face à la variabilité du comportement humain.

Les méthodes statistiques permettent de détecter des variations d'une variable acquises par des capteurs ou par un traitement sur les données. Elles permettent donc de détecter des évolutions au cours du temps via un facteur précis, comme par exemple la vitesse de déplacement ou un niveau d'activité moyenne. Il faut généralement traiter les données afin qu'elles soient utilisables par la méthode statistique. Si ces méthodes permettent de détecter des évolutions du comportement, même partiellement, au cours du temps, elles ne permettent pas de caractériser le comportement actuel de la personne. De plus, les détections des évolutions sont restreintes par le traitement préalable des données.

Les méthodes de classification permettent de caractériser les activités de la personne afin de pouvoir ensuite les identifier. Une restriction de ce genre de méthodes est l'utilisation de données labellisées, ne permettant pas de détecter des activités rares. De plus, cela ne permet pas de prendre en compte l'ensemble du comportement de la personne, comme vu dans la sous-section 2.2.2, mais uniquement des activités spécifiques apprises. De plus, si ces méthodes permettent de détecter des activités habituelles de la personne, celles-ci restent complètement indépendantes ; les transitions entre les activités, ou leurs probabilités d'occurrence n'étant pas étudiées, bien que potentiellement très caractéristiques de l'évolution du comportement.

Les méthodes de pattern matching sont utilisées pour détecter la répétition de séquences dans les données afin de définir le comportement de la personne, permettant ainsi la modélisation du comportement en réduisant le nombre d'*a priori*. Des modèles peuvent ensuite être utilisés afin de détecter ses évolutions. Ces méthodes sont donc particulièrement adaptables, aussi bien en fonction de la personne que des capteurs utilisés, une des seules contraintes étant que les capteurs soient suffisamment explicites. De même que pour les méthodes de classification, les liens entre les activités spécifiques ne sont pas étudiés.

Les réseaux de neurones peuvent être utilisés de multiples manières, aussi bien pour apprendre des activités spécifiques, comme les méthodes de classification, que pour détecter un comportement inhabituel, sans prendre en compte des activités spécifiques. Ils permettent d'acquérir les résultats en réduisant énormément le nombre d'*a priori*.

De même que pour les réseaux de neurones, les méthodes basées sur les modèles de Markov ont plusieurs utilisations différentes. Elles permettent des niveaux d'expressivité très différents, avec des avantages ou des contraintes spécifiques. L'utilisation d'une approche markovienne nécessite néanmoins le respect de l'hypothèse de Markov, et des bases de données suffisantes afin de pouvoir apprendre les paramètres des modèles. En contrepartie, ces modèles peuvent nécessiter peu d'*a priori* en dehors du nombre d'états et sont aisément personnalisables. C'est pour toutes ces raisons que le chapitre suivant (Chapitre 3) sera dédié aux méthodes de Markov.

Chapitre 3

Les modèles de Markov

Sommaire

3.1 Les méthodes d'apprentissage artificiel	34
3.1.1 L'apprentissage supervisé	34
3.1.2 L'apprentissage non-supervisé	35
3.2 L'hypothèse de Markov	36
3.3 Exemples de modèles de Markov	36
3.3.1 Les chaînes de Markov	36
3.3.2 Modèle de Markov Caché	37
3.3.3 Modèle semi-markovien caché	39
3.3.4 Mixture of Gaussian-HMM	41
3.4 Algorithme des Méthodes de Markov caché	42
3.4.1 Problème 1 : calcul de la vraisemblance d'une séquence observée	42
3.4.2 Problème 2 : définir les états cachés les plus probables	43
3.4.3 Problème 3 : apprentissage du modèle à partir des observations	44

3.1 Les méthodes d'apprentissage artificiel

L'apprentissage est lié au domaine de l'Intelligence Artificielle (Russell et Norvig [2009]). Il est défini comme étant la capacité pour un système, à partir d'un ensemble suffisant d'exemples d'un problème donné, de trouver les paramètres permettant de résoudre ce problème, principalement sur des cas qui n'ont pas été observés auparavant, conférant ainsi au système étudié une forme de robustesse. Un élément très important des algorithmes d'apprentissage consiste en leur capacité de généralisation à partir d'un ensemble d'exemples spécifiques et représentatifs du problème.

Les modèles de comportements humains sont complexes, partiellement inconnus, propres à chaque personne et évoluent au cours du temps. Il n'est donc pas possible de les construire tous manuellement. C'est pour cela que nous utiliserons des méthodes d'apprentissage pour les obtenir. Il existe deux catégories d'algorithmes d'apprentissage, dépendant du type de données utilisées.

3.1.1 L'apprentissage supervisé

L'apprentissage supervisé désigne les méthodes d'apprentissage où les données utilisées pour apprendre sont labellisées préalablement. La labellisation consiste à ajouter à chaque élément du corpus d'apprentissage un label, c'est-à-dire le résultat que l'on souhaite obtenir pour cet élément. L'apprentissage consiste donc à déterminer les paramètres permettant, pour l'ensemble du corpus d'apprentissage, de retrouver le label à partir des données.

Un exemple de problème utilisant l'apprentissage supervisé est la classification. Ce type de problème consiste à déterminer en fonction des données d'entrées quelle est la classe correspondante, par exemple la reconnaissance de caractère manuscrit (Lecun *et al.* [1995]). Un autre type de problème est la régression, lorsque le label que l'on cherche à déterminer à partir des données d'entrée fait partie de l'ensemble $\mathbb{R}$.

Parmi les modèles permettant l'apprentissage supervisé, on peut citer les SVM, les réseaux de neurones, les méthodes des plus proches voisins ou les classifications bayésiennes. Il n'est cependant pas toujours possible d'utiliser des méthodes d'apprentissage supervisé, pour les simples raisons qu'elles ne correspondent pas au type du problème traité ou que les données ne sont pas labellisables.

3.1.2 L'apprentissage non-supervisé

Quand la labellisation n'est pas possible, dans le cas où on ne connaît pas les résultats que l'on souhaite obtenir, ou encore si le problème est de très grande dimension, les méthodes supervisées ne peuvent pas être utilisées. Par exemple, dans le cas de problèmes dits de clustering où il s'agit de déterminer l'existence de classes potentiellement utiles sans connaissance *a priori* et de les identifier, ou plus généralement trouver des connaissances utiles dans une grande base de données (Jain *et al.* [1999]). Dans ce cas, il est nécessaire d'utiliser des méthodes d'apprentissage non supervisées.

Différentes méthodes permettent de résoudre les problèmes dans le cas du non-supervisé, chacune avec ses avantages et ses inconvénients. On peut citer à titre d'exemple le modèle bio-inspiré des réseaux de neurones, les méthodes basées sur les distances comme les SVM ou les k plus proches voisins ou des méthodes de type arbres de décision.

Si on considère que les activités des personnes constituent des classes inconnues, nous sommes dans le cas d'un problème de clustering avec des liens entre les classes, et ainsi l'utilisation de méthodes non supervisées devient justifiée. Un autre facteur qui vient renforcer la raison de ce choix est le caractère personnel du comportement de chaque individu. En effet, chaque résident a un comportement différent ; il peut avoir des activités atypiques et peut effectuer des activités d'une façon qui ne corresponde pas à celles apprises, par exemple en cas de comportement d'utilisation tel que décrit par Lhermitte [1983]. Ces éléments sont accentués par le fait que les personnes peuvent présenter des troubles physiques ou cognitifs, comme présentés dans la partie 2.2.2. La nécessité que l'apprentissage soit personnalisé, étant donné les différences entre les personnes, rend impossible l'utilisation de labels et donc des méthodes supervisées. De plus, le cas des nouveaux arrivants dont le comportement n'a jamais été observé auparavant et qui peuvent présenter des particularités par rapport aux comportements de la base d'apprentissage est une raison supplémentaire qui justifie encore une fois le choix d'une telle méthode.

Dans le cadre de notre étude, nous utiliserons plus précisément des méthodes d'apprentissage basées sur des modèles probabilistes, dont le principal avantage est l'aspect robuste qu'il peut leur conférer, notamment aux situations incertaines ou inconnues, particulièrement utile dans notre cas, comme vu dans le chapitre 2.

3.2 L'hypothèse de Markov

Les modèles de Markov sont un ensemble de modèles et de méthodes qui leur sont appliquées, très utilisés dans le domaine de l'Intelligence Artificielle (Russell et Norvig [2009]). L'ensemble de ces modèles ont pour point commun l'hypothèse de Markov, spécifiant que l'état du système à un instant t donné dépend exclusivement de l'état du système à l'instant précédent $t-1$, et que la connaissance de l'ensemble de l'historique du processus n'est donc pas utile. Cette hypothèse est formalisée par l'équation 3.1. Dans cette hypothèse, l'état s du système appartient à l'ensemble dénombrable des états S du système.

$$P(s_{t+1}|s_t, s_{t-1}, s_{t-2}...) = P(s_{t+1}|s_t) \quad (3.1)$$

Les modèles de Markov sont donc des processus stochastiques. L'hypothèse de Markov permet de définir directement le modèle des chaînes de Markov qui seront présentées dans la sous-section 3.3.1. Néanmoins, de nombreux modèles se basant sur l'hypothèse de Markov ont été développés. Chacun de ces modèles permettant d'exprimer de nouvelles expressivités. Parmi les modèles les plus utilisés, le [Modèle de Markov caché ou Hidden Markov Model \(HMM\)](#) sera décrit dans la sous-section 3.3.2, ou les processus de décision de Markov ou *Markov Decision Process* (MDP) ajoutant un choix d'actions agissant sur le système (Puterman [1994]).

Les modèles de Markov ont de nombreux domaines d'application, notamment le traitement de la parole, la robotique, le traitement d'images, la maintenance industrielle... En fonction des problèmes spécifiques, plusieurs types de modèles peuvent être utilisés. Nous nous intéresserons ici à quelques modèles de Markov pouvant nous être utiles. Nous ne prendrons pas en compte l'ensemble des modèles avec prise de décision. En effet, nous souhaitons apprendre un modèle à partir d'observations, sans avoir d'actionneurs permettant d'agir sur le système. Dans notre étude de cas, cela reviendrait à vouloir agir sur le comportement de la personne.

3.3 Exemples de modèles de Markov

3.3.1 Les chaînes de Markov

Les Chaînes de Markov est un type de modèle basé sur l'hypothèse de Markov. Dans les modèles markoviens, un état s_i appartient à un ensemble S fini ou dénombrable qui est l'ensemble des états permettant de décrire le système. De plus, une seconde hypothèse est que les probabilités de transition entre les états, que l'on nommera A ,

soient stationnaires, c'est à dire qu'ils ne changent pas comme formalisé par l'équation $P(s_{t+1} = j | s_t = i) = P(s_{u+1} = j | s_u = i) \forall u, t$. Un état est donc composé d'un certain nombre de variables. Pour que la propriété d'indépendance à l'historique soit respectée, il est nécessaire que l'état S_n soit suffisamment complet par rapport au problème (par exemple heure et jour pour modéliser une file d'attente qui devient plus longue à midi...). Dans cette modélisation, le temps est discret.

La figure 3.1 représente l'exemple d'une chaîne de Markov à 4 états avec ses probabilités de transition, y compris les auto-transitions.

Un exemple d'application des Chaîne de Markov dans le cadre de la détection d'activité est décrit par [Noury et Hadidi \[2012\]](#) et présenté dans la sous-section 2.2.2. Dans cette étude, l'un des deux modèles testés pour représenter les transitions entre les zones du domicile est une chaîne de Markov.

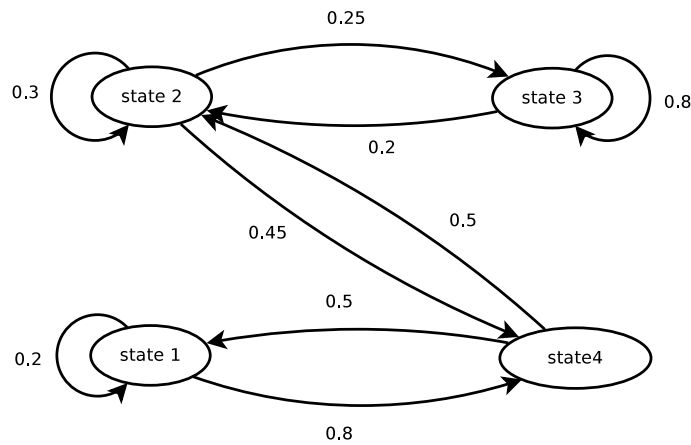


FIGURE 3.1 – Exemple de chaîne de Markov à 4 états

3.3.2 Modèle de Markov Caché

Les [HMM \(Rabiner \[1989\]\)](#) permettent de modéliser des processus markoviens où l'état du système n'est pas directement connu mais où, à chaque pas de temps, le système émet une observation o dont la probabilité dépend de l'état courant du système : $P(o_t | S_T)$. C'est un cas particulier des réseaux bayésiens dynamiques.

Un [HMM](#) est défini par un 3-tuple (π, A, B) avec π l'ensemble des fonctions de probabilité π_n tel que π_i est la probabilité que s_i de S soit l'état initial, A l'ensemble des fonctions de probabilité a_{nm} tel que a_{ij} est la probabilité de la transition de l'état s_i de S à l'état s_j de S et B l'ensemble de fonctions $b_n(k)$ tel que $b_i(k)$ est la probabilité que l'état s_i de S provoque l'observation k . On notera le modèle $\lambda = (\pi, A, B)$. Dans un

3.3. Exemples de modèles de Markov

modèle **HMM**, l'état s_{t+1} ne dépend que de l'état s_t et l'observation o_t ne dépend que de l'état s_t , comme décrit dans la figure 3.2. Dans cet exemple, le **HMM** se compose de 3 états (S1, S2 et S3) et de 4 observations (a, b, c et d), dont deux exclusives à un état. Ainsi, l'observation "a" ne peut être émise que par l'état 1, avec une probabilité de 2/3, alors que l'observation "d" ne peut être émise que par l'état 3 avec une probabilité de 1/6.

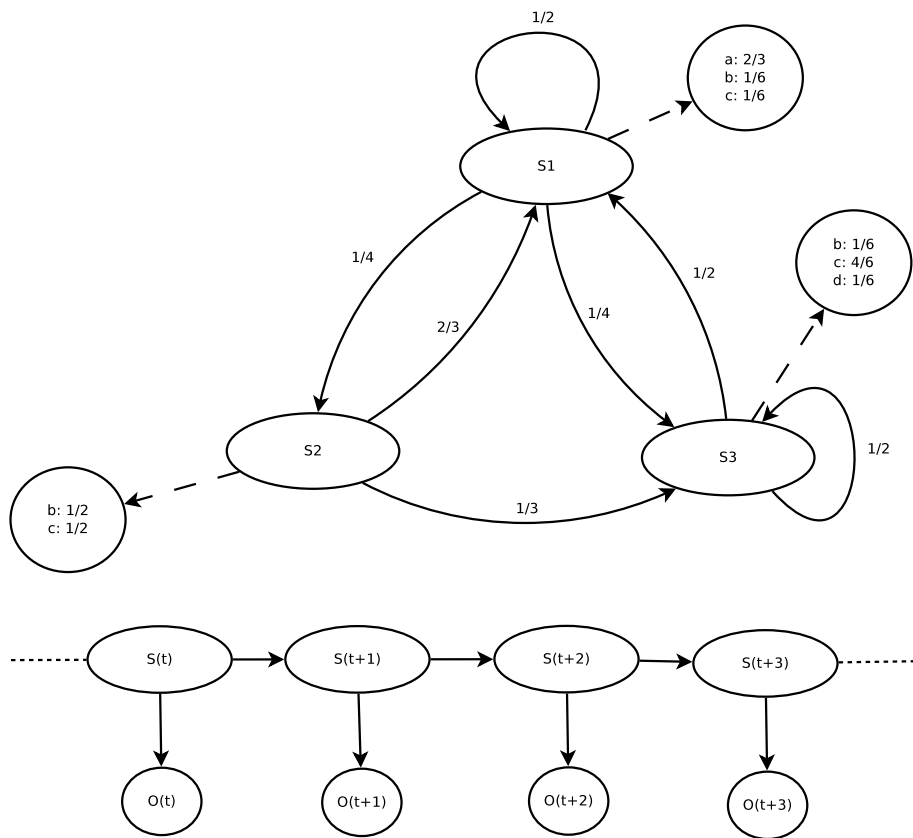


FIGURE 3.2 – Exemple de **HMM**. À chaque pas de temps, le modèle émet une observation (a, b, c ou d) en fonction de l'état courant.

Un **HMM** modélisant un processus markovien, l'état s_{t+1} ne dépend que de s_t . De plus, l'observation o_t ne dépend que de l'état s_t du système. Les **HMM** peuvent permettre de corriger des informations a posteriori, en effet, connaître l'état s_t permet de prédire l'état s_{t+1} mais également de corriger l'état s_{t-1} .

Parmi les méthodes utilisant les **HMM** dans le cadre de la supervision en **smart home** et du comportement humain, la méthode de [Brdiczka et al. \[2009\]](#), vue dans la partie 2.2.2, apprend pour chacune des 6 activités un **HMM**, et qui est ensuite utilisée pour reconnaître l'activité la plus probable. [Yamahara et al. \[2006\]](#) utilisent les

HMM afin d'apprendre les actions d'une personne en fonction des objets qu'elle utilise. **Chung et Liu [2008]** utilisent des modèles de Markov cachés hiérarchiques (HC-HMM), une extension des **HMM**, pour modéliser et discriminer les comportements à partir des postures physiques de la personne acquises préalablement. De même, **Karaman et al. [2012]** proposent une méthode basée sur une caméra et un micro porté par le résident afin de détecter différentes activités en utilisant des hierarchical-HMM (HHMM). **Kasteren et al. [2010]** testent plusieurs modèles pour reconnaître les activités, y compris un modèle de type **HMM**. L'objectif du système proposé par **Zhang et Zhang [2013]** est de détecter un ensemble d'activités dans un **smart home**. Pour cela, ils utilisent des **HMM** d'ordre 2, c'est à dire où s_t dépend de s_{t-1} et s_{t-2} . Dans ce cas, l'hypothèse de Markov est affaiblie selon la formule $P(s_t | s_{t-1}, s_{t-2} \dots) = P(s_t | s_{t-1}, s_{t-2})$.

Les **HMM** peuvent également être hybridés avec d'autres méthodes. Par exemple, pour remplir les objectifs vus en 2.2.2, **Cook [2006]** combine un premier traitement avec des méthodes de Data Mining ; un choix d'actions basé sur un Processus de décision de Markov hiérarchique partiellement observable (HPOMDP). Ensuite, une base de connaissances permet d'assurer que le choix effectué est sûr pour la personne.

3.3.3 Modèle semi-markovien caché

Les **Modèles Semi-Markoviens cachés** ou *Hidden Semi-Markov Model* (**HSMM**) (**Dewar et al. [2012]**, **Yu [2010]**) sont une variantes des **HMM** permettant de modéliser la durée de séjour des états de façon probabiliste et explicite.

Le modèle **HSMM**, également appelé **Modèle de Markov caché à Durée Explicite** ou *Explicit Duration-Hidden Markov Model* (**EDHMM**), peut être défini de deux façons différentes. Il peut être considéré comme un **HMM** où chaque état peut émettre plus d'une observation, en fonction de l'état, comme présenté par la figure 3.3. Il peut également être considéré comme un **HMM** où le système bloque les transitions entre les états pendant un certain nombre de pas de temps, en fonction de l'état, comme présenté par la figure 3.4. Dans cet exemple, les flèches représentent les liens de probabilités entre les éléments, tandis que les pointillés sont les liens automatiques (les durées qui diminuent, ou les états qui ne peuvent pas changer). Pour représenter cela, la transition d'un état à l'autre est assurée par la probabilité de transition $a_{(i,d')(j,d)}$ et les observations par la probabilité d'observation $b_{(j,d)}(O_{t+1:t+d})$ avec $i, j \in S$, $d \in D$, S l'ensemble des états et D l'ensemble des durées. On peut également les considérer comme des **HMM** où les transitions entre deux états ne sont possibles qu'au bout d'une durée d , définie de façon probabiliste à partir de l'état. Dans les autres cas, l'état reste constant.

3.3. Exemples de modèles de Markov

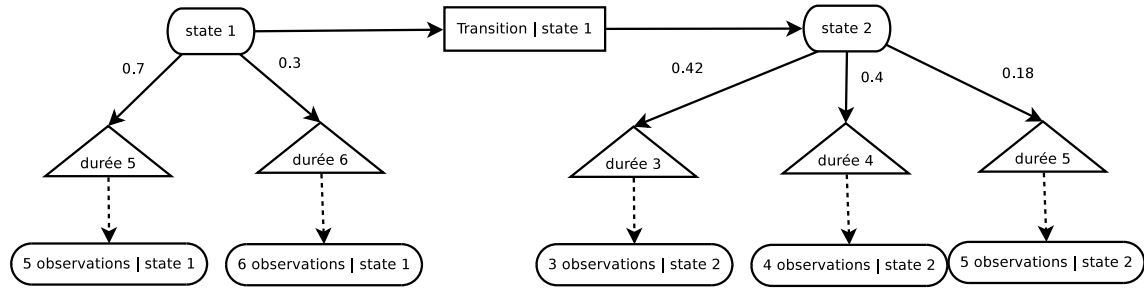


FIGURE 3.3 – Exemple de représentation d'un **HMM** avec plusieurs émissions par état

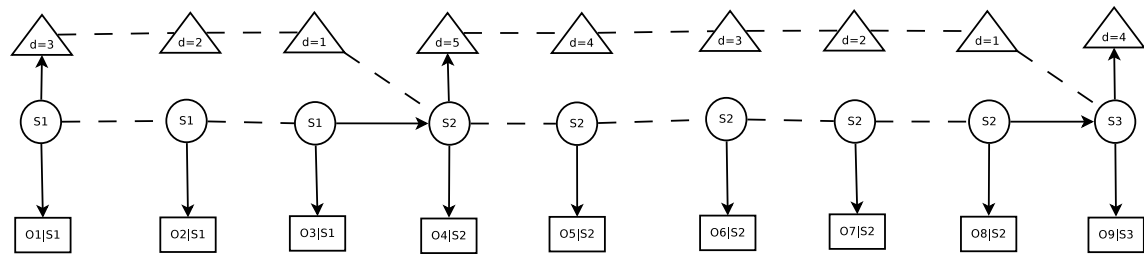


FIGURE 3.4 – Exemple de représentation d'un **HMM** avec transition interdite durant une période de stationnement

Les **HMM** sont modélisables par un tuple (A, B, D, π) , avec π , A et B équivalents à ceux des **HMM**, à part que A n'autorise pas les auto-transitions¹, c'est à dire $a_{ii} = 0 \forall i$ et D les probabilités de stationnement dans un état, permettant de modéliser explicitement le temps passé dans un état, selon la formule (3.2).

$$\begin{cases} P(s_t | s_{t-1}, d_{t-1}) = P(s_t | s_{t-1}, A) \text{ et tirer } P(d_t | s_t = i, D), \text{ si } d_{t-1} = 1 \\ P(s_t | s_{t-1}, d_{t-1}) = 1 \text{ et } d_t = d_{t-1} - 1, \text{ si } d_{t-1} > 1 \text{ et } s_t = s_{t-1} \\ P(s_t | s_{t-1}, d_{t-1}) = 0 \text{ et } d_t = d_{t-1} - 1, \text{ si } d_{t-1} > 1 \text{ et } s_t \neq s_{t-1} \end{cases} \quad (3.2)$$

Yu [2010] présente les différentes évolutions et méthodes liées au **HMM** et fait un état de l'art des méthodes les utilisant, notamment concernant la reconnaissance d'activité humaine. Un exemple d'utilisation des **HMM** dans la littérature est Tan et Xi [2008]. Leur étude consiste à détecter des intrusions dans un système informatique. Pour cela, ils apprennent, grâce à un **HMM** le comportement habituel des utilisateurs du système, un comportement malveillant étant alors défini comme un comportement

1. Les durées de stationnement dans un état permettent en effet de modéliser de façon plus explicite et précis les auto-transitions.

différent du comportement normal. Afin de calculer la différence par rapport au comportement habituel, ils utilisent l'entropie d'informations moyennes des symboles visibles comme métriques.

Les [HSMM](#), ainsi que les modèles développés en les utilisant, sont fréquemment utilisés pour la modélisation du comportement ou pour la supervision de [smart home](#). Ainsi, [Duong et al. \[2009\]](#) utilisent une variante des [HSMM](#), les Coxian Semi-Hidden Markov Model (CxHSMM), combinée à un modèle Switching Hidden Semi-Markov Model (SHSMM), où la durée suit une loi mathématique bien définie et où les variables de la loi sont apprises. Le but est d'apprendre des [ADL](#) dans une smart home composée d'une cuisine. Dans ce cadre, la matrice d'observations est préalablement définie. Différentes lois de durée sont comparées, permettant ainsi de réduire le nombre de variables à apprendre. [Thi et al. \[2005\]](#) présentent une variante des [HSMM](#), les switch-HSMM, appliquée dans la reconnaissance des activités des personnes. Ils présentent une approche pour la détection des durées d'activités anormales et l'expérimentent dans une cuisine équipée de 4 caméras permettant de positionner la personne dans une surface de 1 m^2 . Pour l'expérimentation, un scénario, contenant un nombre connu d'états, est défini.

3.3.4 Mixture of Gaussian-HMM

La représentation (π, A, B) des [HMM](#) présente un inconvénient où l'on suppose que les observations sont de dimension 1. Ce qui n'est pas le cas dans notre étude. Pour pouvoir gérer au mieux l'ensemble des informations disponibles, il est possible de combiner l'ensemble des variables pour créer une "mégavariable" dont les valeurs sont l'ensemble des combinaisons pouvant être pris par chaque variable individuelle. Cette méthode est néanmoins limitée au niveau de son expressivité et dans le cas où le nombre des observations peut exploser. Il est donc nécessaire de considérer des observations d'ordre supérieur.

[Rabiner \[1989\]](#) propose d'utiliser des densités d'observations continues, permettant, par rapport aux observations discrètes, de gérer des observations d'ordre supérieurs à 1, mais également des observations continues. Pour cela, les probabilités d'observations B sont redéfinies avec $b_i(O) = \sum_{m=1}^M c_{im} R[O, \mu_{im}, U_{im}]$ avec O un vecteur d'observation à un instant donné, c_{im} le coefficient de mélange de l'état i , R une densité, généralement une fonction gaussienne, avec un vecteur de moyenne μ et une matrice de covariance U . Ce type de modèle peut être appelé [Mélange de gaussienne-HMM](#) ou [Mixture of Gaussian-HMM \(MoG-HMM\)](#).

Cette méthode peut être généralisée, comme défini par Yu [2010], en proposant des distributions paramétriques des observations. Le principal avantage de cette méthode est de réduire le nombre de paramètres.

Si les différents types de modèles de Markov ont des expressivités intéressantes pour représenter un système donné, plusieurs algorithmes ont été créés afin de résoudre des problèmes spécifiques grâce aux méthodes de Markov.

3.4 Algorithme des Méthodes de Markov caché

Rabiner [1989] définit trois grands problèmes liés aux HMM, applicables également aux modèles dérivés de ceux-ci. Nous noterons λ un modèle, avec l'ensemble des paramètres le définissant. Les trois problèmes sont donc :

1. Pour une séquence d'observations O donnée et un modèle λ donné, calculer $P(O|\lambda)$, la probabilité que le modèle ait généré la séquence d'observation.
2. Pour une séquence d'observations O donnée et un modèle λ donné, trouver la séquence d'état caché S la plus probable, c'est à dire $P(S_{1..T}|\lambda, O)$.
3. Pour une séquence d'observations O donnée, calculer le modèle λ maximisant $P(O|\lambda)$.

Nous allons revenir sur ces 3 problèmes et présenter des moyens pour les résoudre. Dans le cadre de notre étude, nous nous intéresserons plus spécifiquement aux problèmes 1 et 3.

3.4.1 Problème 1 : calcul de la vraisemblance d'une séquence observée

Le premier problème proposé est de déterminer la probabilité qu'un modèle λ ait émis une séquence d'observations donnée O , c'est à dire $P(O|\lambda)$. Dans le cas de notre étude, ce problème apparaît quand on souhaite calculer la proximité entre le modèle de comportement habituel de la personne, et les données acquises au cours de la journée, afin de détecter si la personne a eu un comportement inhabituel, et donc, potentiellement anormal.

Afin de calculer $P(O|\lambda)$ de façon efficace, il est possible d'utiliser une procédure de type forward-backward (Rabiner [1989]). Cette dernière fait appel à deux variables : la variable forward α et sa variable réciproque backward β . Afin de définir la probabilité $P(O|\lambda)$, seule la partie forward est nécessaire. Où $\alpha_t(i) = P(O_1, O_2, \dots, O_t, S_t = s_i | \lambda)$, est la

probabilité des observations jusqu'à l'instant t et de l'état à l'instant t connaissant le modèle. Cette variable $\alpha_t(i)$ peut être calculée récursivement par l'équation 3.3.

$$\begin{aligned} \alpha_1(i) &= \pi_i b_i(O_1) \quad \forall i \\ \alpha_t(i) &= \left[\sum_{j=1}^N \alpha_{t-1}(j) a_{ji} \right] b_i(O_t) \quad 1 < t \leq T \quad \forall i \end{aligned} \quad (3.3)$$

La probabilité que la séquence d'observations O ait été générée par le modèle λ peut donc être définie par $P(O|\lambda) = \sum_N \alpha_T(i)$, avec T la longueur de la séquence d'observations O et N le nombre d'état.

La vraisemblance d'une séquence observée dépend principalement du modèle et des valeurs de la séquence d'observations, mais est également inversement proportionnelle à la longueur de la séquence². De ce fait, la vraisemblance tend vers 0 pour toute séquence de longueur importante, sauf dans le cas particulier où les probabilités sont de 1. Par ailleurs, et afin d'éviter les problèmes liés à la représentation de très petites variables par la machine, il sera plus intéressant de calculer la log-vraisemblance.

3.4.2 Problème 2 : définir les états cachés les plus probables

Le second problème consiste à trouver les états cachés du système les plus probables à chaque instant, permettant d'expliquer une séquence d'observation et connaissant le modèle. En d'autres termes, le problème consiste à trouver la séquence Q permettant de maximiser $P(Q|\lambda, O)$.

Dans le cadre de notre problème, cela consiste à trouver les activités qu'a eu la personne au cours de la journée à partir des données des capteurs et de son modèle de comportement habituel. Néanmoins, le comportement habituel de la personne étant construit sans *a priori*, il est peu intéressant pour les aidants de savoir si la personne est dans l'état 1 ou 2 de son modèle, à moins de savoir à quelle activité correspond l'état. Étant donné que nous ne cherchons pas, dans cette étude, à donner un sens sémantique aux états, définir les états du système est peu intéressant pour nous.

Une première façon de résoudre ce problème consiste à définir, pour chaque pas de temps, l'état du système qui a la probabilité la plus importante. Cela est traduit par la variable $\gamma_t(i)$, définie comme la probabilité d'être dans l'état s_i à l'instant t . γ peut être calculée par l'équation 3.4, où α et β sont les variables de la procédure forward-backward.

2. $P(O|\lambda)$ étant calculé comme une multiplication de probabilités et pouvant être considérée comme étant égale à $\prod_T P(O_t|s_t, \lambda)$.

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^N \alpha_t(j)\beta_t(j)} \quad (3.4)$$

α a été définie dans le cadre du problème 1. Sa variable complémentaire est la variable backward $\beta_t(i) = P(O_{t+1}, O_{t+2}, \dots, O_T | S_t = s_i, \lambda)$. Elle définit les probabilités des observations à partir de l'instant t , connaissant le modèle λ et l'état à l'instant t . De même que $\alpha_t(i)$, $\beta_t(i)$ peut être calculée efficacement de façon récursive par la formule 3.5.

$$\begin{aligned} \beta_T(i) &= 1 \quad \forall i \\ \beta_t(i) &= \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \quad 1 \leq t < T \quad \forall i \end{aligned} \quad (3.5)$$

Compte tenu de la définition de γ , la séquence d'états la plus probable, connaissant le modèle et la séquence d'observations, peut alors être définie par l'état maximisant la valeur de γ_t à chaque pas de temps, c'est à dire $s_t = \arg\max_i \gamma_t(i)$ avec $1 \leq t < T$.

Une limite de cette méthode est le critère d'optimisation, cherchant à chaque pas de temps l'état le plus probable sans prendre en compte le caractère séquentiel des états. Cela peut poser problème, s'il y a des transitions nulles par exemple. D'autres critères peuvent être pris en compte, comme pour l'algorithme de Viterbi (voir annexe 6). Ce dernier permet de définir la séquence la plus probable, connaissant le modèle, grâce à la variable δ , définit selon l'équation 3.6.

$$\delta_t(i) = \max_{s_1, s_2, \dots, s_{t-1}} P(s_1, s_2, \dots, s_{t-1}, s_t = i, O_1, O_2, \dots, O_t | \lambda) \quad (3.6)$$

3.4.3 Problème 3 : apprentissage du modèle à partir des observations

Le dernier problème lié aux HMM est de trouver le modèle λ , avec $\lambda = (\pi, A, B)$ permettant de représenter au mieux un système donné inconnu et dont les seules informations sont une séquence d'observations O générée par ce système, c'est à dire trouver λ tel que $P(O|\lambda)$ soit maximale. Une solution pour résoudre ce problème est l'algorithme de Baum-Welch (Rabiner [1989]). Il s'agit d'une méthode itérative permettant de résoudre localement $\arg\max_{\lambda} P(O|\lambda)$. Pour cela, elle se base sur la méthode forward-backward, définit précédemment, pour le calcul des variable α , β et γ .

Afin de résoudre le problème, une nouvelle probabilité, ξ , est définie. Cette variable correspond à la probabilité d'être dans l'état S_i à l'instant t et à l'état S_j à l'instant $t + 1$ pour un modèle λ donné et une séquence d'observations O donnée, c'est à dire la probabilité $\xi_t(i, j) = P(S_t = s_i, S_{t+1} = s_j | O, \lambda)$. ξ peut être calculée, grâce aux variables définies précédemment, par la formule 3.7.

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{k=1}^N \sum_{l=1}^N \alpha_t(k) a_{kl} b_l(O_{t+1}) \beta_{t+1}(l)} \quad (3.7)$$

L'ensemble de ces éléments permettent de recalculer les paramètres selon les équations (3.8).

$$\begin{cases} \overline{\pi_i} = \gamma_1(i) \\ \overline{a_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \\ \overline{b_j(k)} = \frac{\sum_{t=1 \text{ et } O_t=v_k}^T \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)} \end{cases} \quad (3.8)$$

Les trois étapes de l'algorithme Baum-Welch se résument à l'étape forward pour calculer α , l'étape backward pour calculer β et l'étape de mise à jour du modèle. Ces étapes sont répétées jusqu'à la convergence du modèle, c'est à dire jusqu'à ce que le nouveau modèle calculé ait une vraisemblance inférieure à celle du modèle précédent. L'algorithme Baum-Welch appliqué au modèle [HMM](#) est donné en annexe 1, et une version synthétisée pour le modèle [HSMM](#) élaboré est présentée en annexe 2.

L'algorithme Baum-Welch étant sensible aux maximums locaux, pour cela nous utilisons une variantes Naïve consistant à entraîner, grâce à l'algorithme Baum-Welch, n modèles à partir de paramètres d'origines aléatoires. Le modèle gardé comme résultat de l'apprentissage est celui dont la vraisemblance par rapport à la séquence d'observations apprise est la plus importante.

Ces formules sont la base théorique de l'algorithme Baum-Welch. Elles ne peuvent néanmoins pas être directement utilisées pour implémenter l'algorithme dans un certain nombre de conditions. Par exemple à cause du problème de précision des machines et α tendant vers 0, il n'est pas possible de calculer directement la valeur de α quand il y a un grand nombre d'observations. Des adaptations de l'algorithme, résolvant ces problèmes sont disponibles, par exemple grâce à une méthode de mise à l'échelle et de normalisation des variables présentée par [Rabiner \[1989\]](#), ou par le passage au logarithme.

Chapitre 4

Apprentissage du comportement habituel de la personne

Sommaire

4.1	Caractéristiques du modèle	48
4.1.1	L'incertain	48
4.1.2	Le temps	48
4.1.3	La durée	49
4.1.4	La personnalisation	49
4.2	Description du modèle Gs-GsHMM	50
4.2.1	Modèle MoG-HSMM	50
4.2.2	Gestion de la durée de stationnement	50
4.2.3	Gestion des observations	51
4.2.4	Gestion de l'état initial	52
4.2.5	Limites du modèle	52
4.3	Implémentation des algorithmes	53
4.3.1	Apprentissage des modèles	53
4.3.2	Comparaison des modèles	55
4.4	Étude de la complexité	56

4.1 Caractéristiques du modèle

L'étude du contexte (Chapitre 2) met en évidence que la modélisation du comportement humain peut prendre en compte différentes caractéristiques et que différents niveaux de granularité sont possibles. Ceci est également vrai pour la détection des situations critiques et imprévisibles. Néanmoins, compte tenu des objectifs de détection, il est nécessaire que le modèle de comportement habituel de la personne permette de discriminer correctement un certain nombre de situations. Il est alors nécessaire qu'il puisse avoir la capacité de représenter un certain nombre de caractéristiques, telles que : l'incertain, le temps et la durée.

4.1.1 L'incertain

Le comportement humain ne peut pas être considéré comme un système déterministe, particulièrement dans le cas de personnes présentant des difficultés cognitives, comme expliqué précédemment. Le modèle devant être représentatif du comportement humain, il en est de ce fait fortement impacté.

De plus, les informations sur l'environnement traitées les algorithmes sont obtenues par des capteurs. Or, les informations fournies par ces derniers sont limitées à leur mode de fonctionnement. Par exemple, un capteur de mouvement comme un PIR renverra les mêmes données s'il n'y a personne dans son champ de détection ou si la personne présente est immobile, comme vu dans la partie 2.1.1. Si une partie des données peut être filtrée afin de supprimer des données erronées, dû à un mauvais fonctionnement du capteur, certaines données restent peu fiables, fortement bruitées ou peu significatives de l'activité de la personne.

Au regard de ces éléments, le système doit être capable de gérer l'incertitude. Cette incertitude peut être gérée en utilisant des modèles stochastiques, comme il est proposé par Russell et Norvig [2009].

4.1.2 Le temps

Le modèle doit permettre de différencier les situations normales de celles qui ne le sont pas en fonction du moment de la réalisation de l'activité, dans le cas où un résident effectue une activité à un moment qui ne lui est pas habituel. Pour Lee *et al.* [2013], le temps est également un facteur de mesure du comportement. En effet, s'il est normal que le cycle circadien des personnes âgées évolue, certaines évolutions peuvent être des signes de problèmes, par exemple dans le cas d'agitation nocturne et

de somnolence diurne. De plus, si une personne a l'habitude d'effectuer une activité à un moment donné, la réalisation de cette même activité durant une autre période de la journée peut indiquer l'existence d'un problème, par exemple faire sa toilette durant la nuit alors que la personne la fait habituellement durant la matinée. Dans ce contexte, [Demongeot et al. \[2002\]](#) présentent une plate-forme expérimentale dans le cadre du maintien à domicile mettant en avant le rythme circadien de la personne. De plus, [Zhou et al. \[2012\]](#) étudient les conséquences de différents types de démences sur le comportement, mais aussi sur les rythmes circadiens et du sommeil. Le rythme circadien personnel est pris en compte ainsi que le jour de la semaine ou les saisons dans l'année.

4.1.3 La durée

La durée d'une activité peut indiquer un comportement anormal si la durée est particulièrement longue ou courte, comparée à la durée habituelle ([Lee et al. \[2013\]](#)). Afin de pouvoir détecter le caractère anormal, il est important de se référer à la durée moyenne de l'activité pour la personne, et non pas à une moyenne en général. En effet, de la même façon qu'une activité peut être effectuée différemment, la durée d'une activité peut être très variable d'une personne à l'autre. Par exemple, certaines pathologies font que les personnes peuvent avoir des difficultés à rester assises trop longtemps. De ce fait, il arrive que dans des [EHPAD](#), certaines personnes peuvent marcher et manger en même temps. Pour ces personnes, l'activité "manger" n'est pas liée au fait que la personne soit assise, et que l'activité dure généralement plus longtemps que la durée moyenne.

Une grande variation de la durée pour effectuer une activité par rapport à la durée habituelle peut indiquer que la personne est en difficulté. De plus, l'évolution de la durée moyenne d'une activité au cours du temps peut être révélatrice de changement de son état général comme dans le cas de variations des durées du cycle du sommeil tel que décrit par [Zhou et al. \[2012\]](#).

4.1.4 La personnalisation

En général, du fait d'avoir des activités spécifiques, chaque personne a un comportement propre, qui peut être très différent de celui des autres. Pour que le modèle soit le plus proche des habitudes de la personne et pour pouvoir détecter des comportements inhabituels, ou des changements d'habitude de vie, le modèle doit être personnel et par conséquent doit être appris pour chaque personne. Cette caractéristique n'est pas directement liée au modèle mais au fait qu'un modèle donné corresponde au

comportement habituel d'une seule personne, et qu'il n'est donc pas possible d'utiliser un même modèle pour plusieurs personnes.

L'ensemble de ces éléments font que les modèles de comportement sont complexes, partiellement inconnus, uniques pour chaque individu et peuvent évoluer au cours du temps. Dans la suite, nous proposons un modèle à partir de l'hybridation de deux modèles capables de modéliser le comportement de la personne à un instant donné. Par ailleurs, pour que le modèle corresponde au mieux au comportement de la personne, celui-ci devra être appris pour chaque personne et pour que les caractéristiques du domaine soient respectées, un modèle est proposé, en combinant les modèles [HSMM](#) et [MoG-HMM](#) pour toutes les raisons détaillées en section 3.3.

4.2 Description du modèle Gs-GsHSMM

4.2.1 Modèle MoG-HSMM

La modélisation du comportement habituel nécessitant de prendre en compte les durées de stationnement dans un état, mais également de gérer des vecteurs d'observations avec des valeurs continues, issues des données des capteurs filtrées, nous combinons les deux modèles [HSMM](#) et [MoG-HMM](#) afin d'obtenir un nouveau modèle ayant l'expressivité des deux modèles.

De plus, afin de réduire la complexité du modèle et les temps de calcul, nous simplifions le modèle afin de n'utiliser que les éléments qui nous intéressent pour résoudre le problème posé. Nous obtenons ainsi le modèle MoG-GsHSMM. Toutefois, ce modèle ne correspond pas exactement aux besoins identifiés. Ainsi, différentes modifications sont nécessaires afin que le modèle soit plus simple et plus expressif. Pour cette fin, on aboutit au modèle [Gaussienne-Modèle Semi-Markovien caché Gaussien ou *Gaussian Hidden Semi-Markov Model*](#) (Gs-GsHSMM).

4.2.2 Gestion de la durée de stationnement

Comme première modification apportée au modèle de base, nous considérons que les durées de stationnement dans un état suivent une loi gaussienne, propre à l'état. En effet, on suppose qu'une activité donnée a une durée moyenne autour de laquelle se répartissent les valeurs réelles des durées de l'activité. Ce qui justifie l'utilisation d'une distribution de type gaussienne. Cette hypothèse est relativement faible, les gaussiennes pouvant modéliser un grand nombre de phénomènes naturels. D'autres lois pourraient potentiellement mieux représenter la durée de certaines activités, néanmoins, les lois

gaussiennes permettent de simplifier le modèle et de modéliser, même globalement, les durées moyennes. Ainsi, l'utilisation d'une loi gaussienne permet, non seulement, de simplifier le modèle mais aussi de réduire le temps d'apprentissage.

Pour chaque état, la probabilité de stationnement D est définie par les paramètres de la gaussienne, μ la moyenne et σ l'écart-type. De plus, les gaussiennes sont tronquées. En effet, les algorithmes d'apprentissage impliquent une durée $D_{\max}$ de stationnement maximale dans un état, ce qui implique que nous considérons chaque gaussienne entre 1 et une durée de stationnement maximale. La gaussienne est ensuite normalisée afin que $\sum_d P(d|s_i) = 1$ soit vérifiés. Il est important de noter que les gaussiennes sont continues et pourraient donc permettre que les valeurs des durées soient continues. Toutefois, comme le modèle considéré est à temps discret, on se limitera à la sélection des valeurs entières.

4.2.3 Gestion des observations

La seconde modification apportée consiste, pour chaque dimension du vecteur d'observations, à ne considérer qu'une gaussienne par état, les valeurs des données issues des capteurs utilisés étant supposées modélisables par une fonction gaussienne, c'est à dire être centrées autour d'une valeur moyenne, en respectant un écart-type donné. Si la première hypothèse est relativement faible dans le cadre de notre application, la seconde est fortement liée à la nature des données d'entrée, et pourrait, avec l'ajout de nouveaux capteurs par exemple, devoir être supprimée. La nature des capteurs actuellement utilisés permet néanmoins de la maintenir dans le cadre de notre expérimentation. Les avantages de maintenir l'hypothèse sont très intéressants pour le temp de calcul et le nombre de paramètres.

Les probabilités d'observation τ de chaque état sont définies par un ensemble de gaussiennes, chacune correspondant à un capteur ou à une donnée spécifique. Cette modélisation implique que les probabilités d'émission de chacun des capteurs sont indépendantes par rapport aux autres capteurs. Cette hypothèse est fausse. Par exemple, si les capteurs de mouvement émettent, il y a de fortes probabilités qu'il y ait au moins une personne dans la pièce, et donc que le niveau de CO2 augmente. Les probabilités d'émission des capteurs [PIR](#) et de CO2 sont donc liées. Il serait possible, afin de corriger cette erreur, de considérer une gaussienne de dimension n au lieu de n gaussiennes de dimension 1. L'hypothèse d'indépendance est néanmoins maintenu dans le cadre de ces expérimentations. En effet, afin de respecter la contrainte d'amélioration du modèle, notamment avec l'ajout de nouveaux capteurs, il est plus simple d'ajouter des gaussiennes supplémentaires, plutôt que d'augmenter la dimension d'une gaus-

sienne. Cela implique néanmoins un biais qu'il sera nécessaire de corriger en faisant évoluer le modèle afin d'intégrer l'aspect de dépendance entre les capteurs.

Le modèle **Gs-GsHSMM** n'est pas lié à un type donné de capteurs, à partir du moment où les données émises par les capteurs peuvent être considérées dans $\mathbb{R}$. Compte tenu du vecteur d'observations, plusieurs types de capteurs peuvent être considérés en même temps.

Une contrainte forte sur les capteurs étant leur fréquence d'émission. En effet, afin d'apprendre correctement le comportement habituel au sein de la chambre, il est nécessaire que la fréquence des capteurs soit suffisamment importante pour que la valeur soit significative. Ainsi, un capteur ne communiquant qu'une fois par jour, par exemple, n'est pas significatif de l'activité de la personne au cours de la journée : la valeur pouvant être considérée constante quelle que soit l'activité. Au contraire, un capteur pouvant communiquer des valeurs différentes au moins toutes les minutes pourra être significatif de l'activité de la personne, pour des activités dont la durée est de l'ordre de la minute.

4.2.4 Gestion de l'état initial

Une dernière hypothèse que nous faisons est liée au caractère circadien du comportement humain. Le comportement humain dépendant du temps de façon extrêmement cyclique, définir un instant t_0 n'est pas significatif, et l'instant initial à partir duquel on lance l'apprentissage n'est pas du tout représentatif du comportement du résident.

De ce fait, nous considérons les probabilités d'état initial comme équiprobables, afin de simplifier le modèle et de réduire le nombre de paramètres à apprendre, permettant ainsi d'accélérer légèrement le temps de calcul.

4.2.5 Limites du modèle

Si le modèle doit permettre de modéliser le comportement habituel de la personne, celui-ci présente différentes limites.

Les **Gs-GsHSMM** sont discrets dans le temps, ce qui est en contradiction avec le comportement humain qui est continu. Cela n'est pas un problème si l'on considère un pas de temps suffisamment faible. En effet, si les résidents ont des activités continues dans le temps, les informations qui les concernent ne sont accessibles que via

des capteurs. Or, ces capteurs ne fournissent pas des informations de manière continue mais en fonction du type de capteurs et d'une façon généralement discrète (toutes les secondes par exemple), ou épisodique (lorsqu'il y a détection). De ce fait, prendre un pas de temps inférieur à celui des capteurs ayant la fréquence la plus élevée n'est pas utile. Mais en plus, il ne permet pas d'avoir des informations complémentaires et complexifie le système. Si le choix de discrétiser le temps posait problème, il serait possible d'utiliser une variante du modèle de Markov, considérant le temps continu.

Les **Gs-GsHSM** sont donc définis par un tuple (π, A, Δ, τ) où π est la distribution de probabilités de l'état initial, A est les probabilités de transitions entre les états, avec les auto-transitions interdites ($a_{ii} = 0 \forall i$), Δ est la distribution de probabilité de stationnement dans un état et τ la distribution de probabilité des observations. Δ et τ sont composés de distributions gaussiennes et se composent des paramètres gaussiens, c'est à dire μ les moyennes des gaussiennes et σ leurs écarts-types. De plus, pour les probabilités de stationnement Δ , il y également les variables de normalisation de chaque gaussienne, afin que le modèle vérifie la condition : $\sum_d P(d|s_i) = 1$.

En excluant π non utilisée dans le cadre de cette étude, le modèle est défini par $n_e(n_e + 2(n_o + 1) - 1)$ paramètres, avec n_e le nombre d'état caché et n_o le nombre de gaussiens pour le vecteur d'observation. L'ensemble de ces paramètres doivent être appris, à partir des données des capteurs, afin de caractériser le comportement de la personne. De plus, afin de valider les modèles appris, il est possible de calculer la probabilité que ceux-ci ait généré une séquence d'observations qui n'a pas été utilisée pour l'apprentissage. Nous sommes donc dans le cadre des problèmes 1 et 3 des **HMM**, définis dans la section 3.4.

4.3 Implémentation des algorithmes

4.3.1 Apprentissage des modèles

Afin de déterminer les modèles du comportement habituel de chaque résident, nous utilisons l'algorithme Baum-Welch, décrit dans le cas des **HMM** dans la partie 3.4.3. Pour cela, nous nous basons sur l'implémentation des **HSM** tel que proposée par **Yu et Kobayashi [2003]**, et présentée dans l'algorithme 2, auquel nous rajoutons des ensembles d'observations gaussiennes et nous apportons des modifications pour les durées de stationnement afin qu'elles soient modélisées par des gaussiennes.

Ainsi, pour permettre l'apprentissage des paramètres des gaussiennes modélisant

les observations, nous faisons appel à une implémentation existante¹. Calculer les paramètres nécessite de définir de nouvelles variables, devant être calculées durant la phase backward (Rabiner [1989]). Ces variables sont S_y , S_{yy} et w , calculées selon les formules 4.1.

Ces formules utilisent la variable γ tel que décrite dans la partie 3.4.2.

$$\begin{aligned} S_y(i, k) &= \sum_T \gamma(i) O_{t,k} \quad \forall i, k \\ S_{yy}(i, k) &= \sum_T \gamma(i) O_{t,k}^2 \quad \forall i, k \\ w(i) &= \sum_T \gamma(i) \quad \forall i \end{aligned} \tag{4.1}$$

Ces variables sont utilisées pour mettre à jour chacune des variables d'observations, c'est à dire les paramètres μ et σ des gaussiennes, selon la formule 4.2. La mise à jour des paramètres d'observations gaussiennes, en ne prenant pas en compte de variables de mixture, est proche des formules utilisées avec les modèles HMM, avec la probabilité γ définissant la probabilité de l'état à chaque instant t .

$$\begin{aligned} \mu_{i,k} &= \frac{S_y(i, k)}{w_i} \\ \sigma_{i,k} &= \sqrt{\frac{S_{yy}(i, k)}{w_i} - \left(\frac{S_y(i, k)}{w_i} \right)^2} \end{aligned} \tag{4.2}$$

Une autre modification par rapport à l'implémentation des HSMM est l'utilisation de fonctions gaussiennes tronquées afin de modéliser les durées de stationnement dans les états. Pour définir les paramètres μ et σ des gaussiennes, nous utilisons la méthode proposé dans Yu *et al.* [2006]. Celle-ci consiste, pour chaque état, à maximiser la fonction $\arg \max_{\Delta_m(d)} \sum_d \overline{\Delta_m(d)} \log \Delta_m(d)$ avec $\overline{\Delta_m(d)}$ calculée tel qu'indiqué dans l'algorithme 3. Pour cela, nous utilisons une descente de gradient avec un pas calculé avec la méthode du gradient combiné. Nous calculons ensuite les variables de normalisation.

L'algorithme 3 présente le pseudocode de l'algorithme de Baum-Welch dans le cas des Gs-GsHSMM. Cet algorithme permet d'apprendre, à partir des observations, les paramètres du modèle. De plus, l'algorithme Baum-Welch étant sensible aux maximums locaux, nous utilisons une variante Naïve consistant à entraîner n modèles à partir de paramètres d'origines aléatoires. Seul le modèle maximisant la vraisemblance est gardé comme résultat de l'apprentissage.

1. <http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html>

Afin de pouvoir apprendre des modèles corrects, notre implémentation de l'algorithme doit prendre en compte différentes contraintes, notamment dû aux caractéristiques de notre problématique. Ainsi, pour apprendre un modèle de comportement habituel significatif, il est nécessaire de prendre en compte suffisamment de données. En considérant un pas de temps d'une durée d'une minute, chaque journée complète apprise augmente la longueur de la séquence d'observations de 1440 pas de temps. Cela implique que différents problèmes peuvent apparaître, par exemple dû aux différences d'ordre des variables, fortement supérieur à la longueur de la mantisse. De plus, la longueur de la séquence d'observations augmente le temps de calcul, mais également la quantité de mémoire nécessaire, à cause des variables α et β .

Deux paramètres doivent être choisis afin d'initialiser l'algorithme : le nombre d'états cachés et la durée de stationnement maximale dans un état. Ces deux paramètres dépendent du problème que l'on souhaite modéliser. Pour la modélisation du comportement humain, la durée de stationnement maximale peut être déterminée en fonction de la période du sommeil ou de la période d'absence de la chambre au cours de la journée c'est-à-dire en fonction de ce qui est le plus long. Ces deux paramètres affectent la viabilité du modèle, et il est donc nécessaire de les choisir correctement.

4.3.2 Comparaison des modèles

La log-vraisemblance est calculée comme défini dans le problème numéro 1, de la sous-section 3.4.1. Seule la partie forward de l'algorithme de Baum-Welch est donc gardée. La log-vraisemblance d'un modèle par rapport à une séquence d'observations donnée peut ainsi être calculée. Si l'algorithme présenté dans la sous-section 4.3.1 peut ainsi être réutilisé, les contraintes sont beaucoup moins fortes, notamment par rapport à la durée de la séquence d'observations. En effet, si l'apprentissage nécessite une séquence d'observations suffisamment longue, impliquant différents problèmes, calculer la log-vraisemblance peut se faire sur une durée relativement réduite et les problèmes sont donc beaucoup moins importants. Calculer la log-vraisemblance d'un modèle est donc direct par rapport à l'apprentissage.

Le calcul de la log-vraisemblance se fait néanmoins par rapport à une séquence d'observations défini et le résultat en est complètement dépendant. Il faut donc choisir correctement la séquence. Dans le cas où l'on souhaite calculer la vraisemblance des données acquises durant la dernière période par rapport à un modèle de comportement habituel, la séquence d'observations est choisie directement. Dans le cas des expérimentations sur des données simulées, la séquence d'observations utilisée pour

calculer la vraisemblance doit être pertinente, mais également être différente de la séquence d'observations utilisée pour l'apprentissage, afin de ne pas fausser les résultats.

4.4 Étude de la complexité

Murphy [2002] propose un calcul pour la complexité de l'algorithme forward-backward appliqué aux HSMM. Il estime cette complexité $O(TQ^2D_{\max}^2)$, avec : T la longueur de la séquence observée, Q le nombre d'états et $D_{\max}$ la durée maximum de stationnement dans un état. L'algorithme proposé par Yu et Kobayashi [2003] a une complexité de $O((QD_{\max} + Q^2)T)$. Les modifications apportées à l'algorithme modifie peu la complexité du forward-backward. En effet, l'ensemble des variables α et β peut être calculé de façon équivalente, bien que le calcul des $b_i(O_t)$ avec $1 \leq i \leq Q$, $1 \leq t \leq T$ doit prendre en compte l'ensemble des éléments du vecteur d'observations. Deux variables supplémentaires sont à calculer, pour chaque gaussienne d'observations, ce qui porte la complexité à $O((QD_{\max} + Q^2 + Qk)T)$, où k est la dimension du vecteur des observations.

La mise à jour du modèle a une complexité plus élevée. En effet, A et τ peuvent être mises à jour avec une complexité de $O(Q(Q + Q + k))$. La mise à jour de Δ se fait par une descente de gradient. La complexité de cet algorithme étant relativement faible, cela ramène la complexité générale de la mise à jour du modèle à $O(Q(Q + Q + k + 1))$.

Une itération de l'algorithme d'apprentissage consiste à un forward-backward et à la mise à jour des variables. Chaque itération de l'algorithme de Baum-Welch a donc une complexité de $O(Q((D_{\max} + Q + k)T + 2Q + k))$. Le nombre d'itérations n'est pas connu à l'avance, la méthode itérant jusqu'à convergence où un nombre d'itérations maximums est atteint. De plus, avec la méthode Naïve, l'algorithme est exécuté plusieurs fois, augmentant le nombre d'itérations à effectuer à $O(IQ((D_{\max} + Q + k)T + 2Q + k))$ avec I le nombre d'itérations.

Chapitre 5

Application

Sommaire

5.1 Application à l'Apprentissage de modèle de déplacement habituel en living lab	58
5.1.1 Résultats	59
5.1.2 Discussions	63
5.2 Apprentissage sur données simulées	64
5.2.1 Données simulées	64
5.2.2 Étude des paramètres du modèles par plan d'expérience	66
5.2.3 Étude des modèles issues des données simulées	68
5.2.4 Discussions	72
5.3 Application à l'Apprentissage en situation réelle	73
5.3.1 EHPAD d'Ambazac	73
5.3.2 Discussion	78
5.4 Conclusion	79

Afin de tester la validité du modèle pour apprendre le comportement habituel de la personne dans un EHPAD grâce aux données acquises par les capteurs, nous testons celui-ci sur différentes situations.

5.1 Application à l'Apprentissage de modèle de déplacement habituel en living lab

Dans un premier temps, nous testons l'applicabilité de la méthode au sein du GIS Madonah, décrit dans la partie 2.1.2. Dans le cas de nos expérimentations, nous avons décidé d'utiliser uniquement la chambre et le couloir menant à la salle d'eau, et de ne pas prendre en compte la partie pièce de vie et la cuisine. Cela permet d'avoir une configuration plus proche de ce que peut être une chambre d'EHPAD, notamment au niveau de la surface de l'environnement et de faire en sorte que les zones de détection des capteurs se recouvrent.

Nous avons utilisé un réseau de capteurs PIR spécifique, indépendant des autres capteurs du GIS et plus proche, du point de vue fonctionnement, des technologies utilisées dans le cadre du projet E-monitor'âge, notamment via l'utilisation d'un réseau KNX, impliquant des contraintes spécifiques. Ce réseau est composé de huit capteurs PIR connectés à un room controller, lui-même connecté à un ordinateur permettant l'acquisition des données. Six des capteurs sont placés par groupe de deux, un à 30cm du sol, et un à 1m20, placé à l'envers afin de détecter exclusivement les éléments en hauteur, comme indiqué par la figure 5.1. Dans le cadre de l'apprentissage du comportement de déplacement, nous utilisons exclusivement les 5 capteurs situés en hauteur. L'ensemble de ces capteurs forment un quadrillage de la zone qui nous intéresse. Les capteurs qui émettent sont donc différents en fonction de l'endroit où se situe la personne, permettant ainsi d'utiliser la méthode de localisation décrite dans la sous-section 2.1.3.

Dans la suite, nous proposons un scénario, présenté par la figure 5.1. Ce scénario est décrit comme suit : la personne, dans le lit, se lève à gauche, rejoint le couloir, attend, puis retourne dans le lit par la droite et attend. Ce scénario est simple permettant d'avoir des données dont nous maîtrisons le contenu, et nous permet également d'identifier facilement les problèmes pouvant se présenter. De plus, ce scénario est cyclique, permettant, de façon très limitée, de représenter un cycle circadien, ou au moins, une répétition dans le comportement. Ce scénario a ainsi pu être enregistré huit fois pour former la base d'apprentissage. Compte tenu des capteurs utilisés, et du peu de données, cette base d'apprentissage permet d'apprendre uniquement le pat-

tern du déplacement de la personne, et la séquence d'observations fait que les états correspondent à la position de la personne, les observations étant les probabilités que la personne se situe dans une zone donnée.

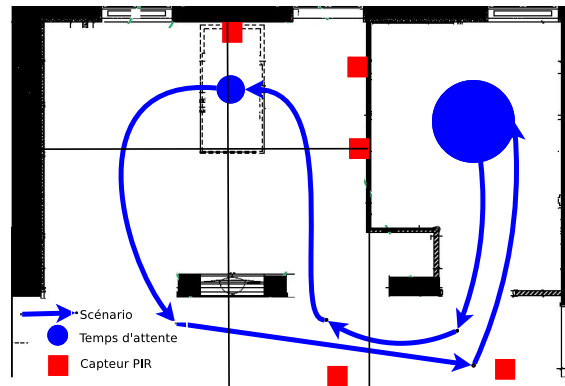


FIGURE 5.1 – Scénario effectué

Les données acquises par la méthode de localisation permettent de former un pavage de l'environnement où l'on sait, à chaque instant, si la personne est ou non. La méthode de localisation permettant d'avoir plus d'une centaine de pavés différents, en grande partie recouvrante, nous ne pouvons pas utiliser les résultats directement. À partir des données de localisation, il s'avère indispensable de proposer un pavage de l'environnement, qui soit le plus significatif pour les données. Par exemple, aucune zone ne peut avoir un côté inférieur à 1m. De plus, l'ensemble des zones fournies par la méthode de localisation doit être représenté, et les zones ne doivent pas être couvrantes. Cette méthode fournit un pavage de dix pavés. Une observation va consister en dix valeurs et chacune correspond à un pavé de la zone. Les valeurs prennent, à chaque instant, la valeur 1 si l'intersection du pavé de localisation issue de la méthode et du pavé du zonage est non-vide, et 0 sinon.

Les observations étant ainsi générées à partir des données de capteurs et de la méthode de localisation de la personne, nous testons l'apprentissage du modèle de déplacement.

5.1.1 Résultats

L'algorithme de Baum-Welch permet, à partir des données filtrées, d'apprendre l'ensemble des paramètres du modèle permettant de représenter le scénario. Étant donné que nous connaissons le scénario à partir duquel les données ont été acquises, les résultats peuvent être validés empiriquement. Différents apprentissages ont été effectués, en modifiant les entrées, c'est à dire le nombre d'états du modèle et la durée

de stationnement maximum dans un état.

Compte tenu des données utilisées, notamment la simplicité du scénario, le nombre d'états est réduit. Deux états sont significatifs pour représenter le scénario et sont identifiables dans l'ensemble des apprentissages, représentant la zone du lit et la zone de la salle d'eau. Au-delà de quatre les états se répètent et ne sont pas significatifs. L'influence de la durée de stationnement maximum dans un état a également été testée. Dans le cadre de l'expérience présentée, nous avons considéré quatre états et une durée maximale de deux minutes.

Deux types de modèles sont appris. Un exemple du résultat le plus courant est présenté ensuite, avec deux états définissant la zone de la salle d'eau. Dans l'autre cas, l'un des états modélise la zone entre la salle d'eau et la zone lit et sert de transitions entre les deux zones. Ce cas, bien que modélisant mieux le scénario, n'est appris que rarement, du fait du peu de données. Cette zone a néanmoins plus de probabilité d'apparaître dans le cas où on considère davantage d'états cachés.

L'analyse du résultat est effectuée en 3 étapes : les probabilités d'observations pour chaque état, les probabilités de transitions entre états et les probabilités de durées de stationnement de chaque état.

Probabilités d'observation

Les probabilités des observations émises par chaque état sont représentées par des gaussiennes. Pour chaque état, dix gaussiennes, définies par leurs moyennes et leurs variances, sont associées. Chaque gaussienne correspond à une zone du GIS, et définit les probabilités que le système émette une observation dans la zone correspondante quand le système est dans l'état correspondant.

Les résultats du modèle sont continus, étant calculé stochastiquement grâce aux gaussiennes. Or, les observations des données apprises ont des valeurs discrètes : chaque zone émet une observation ou non. Les valeurs continues ne sont pas significatives, il est donc nécessaire de discrétiser les valeurs générées par le modèle. Pour cela, les valeurs inférieures à 0.5 signifient que la personne n'est pas détectée dans la zone, et les valeurs supérieures à 0.5 indiquent le contraire, c'est à dire que la personne est détectée dans la zone. Nous considérons donc pour chaque état la probabilité de présence de la personne.

La figure 5.2 présente les probabilités de présence de la personne dans chaque zone. Étant donné que nous n'utilisons pas de capteurs dans la salle d'eau, quand la personne est dans celle-ci, elle est détectée dans le couloir y menant.

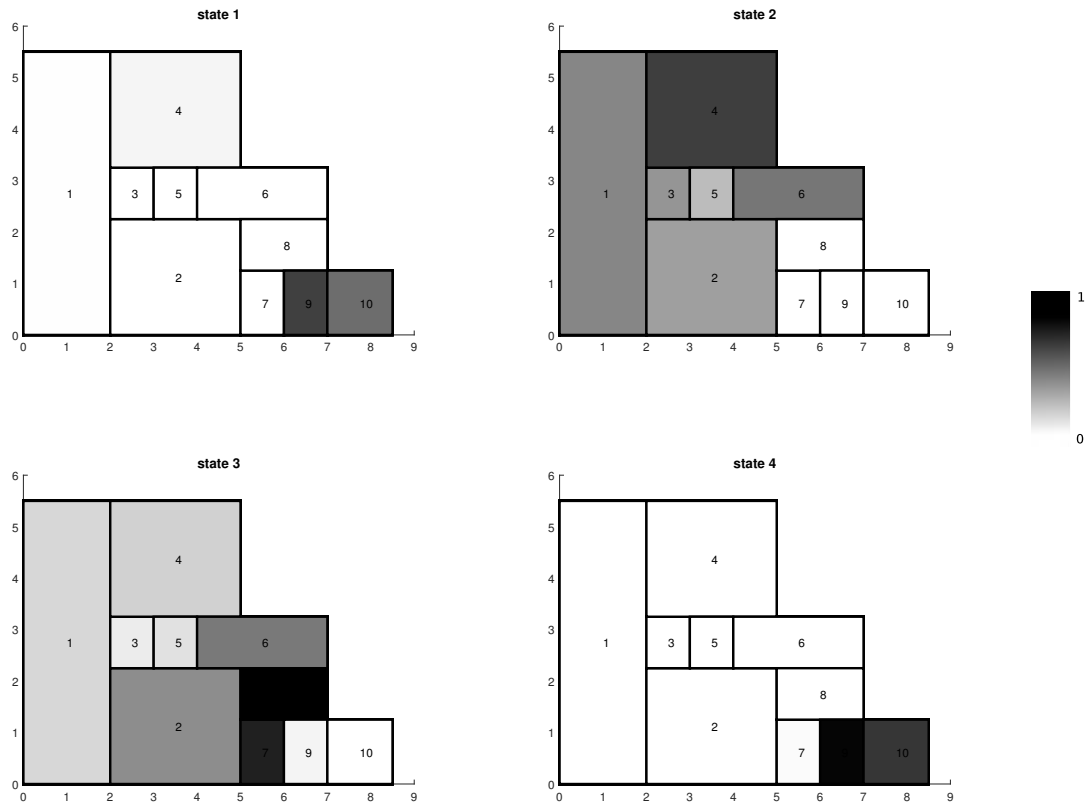


FIGURE 5.2 – Probabilité de présence dans chacun des états

L'analyse de la figure 5.2 nous permet de donner un sens aux états. Les états 1 et 4 sont très similaires. Il y a de fortes probabilités de présence de la personne dans les zones 9 et 10, qui correspondent au couloir menant à la salle d'eau. Il est important de noter que la probabilité de présence de la personne dans certaines zones est très faible, voire nulle. Dans l'état 2, les probabilités de présence de la personne dans les zones autour du lit sont très importantes, tandis qu'elles sont faibles dans les autres zones. Enfin, l'état 3 correspond à de fortes probabilités de présence de la personne dans la zone comprise entre le couloir et les zones autour du lit. Il s'agirait donc d'un état transitoire entre les différentes zones, et donc entre les états. Cela doit être confirmé par l'analyse de la matrice de transition. Les observations sont particulièrement intéressantes du fait de probabilités nulles dans certaines zones, impliquant que la personne ne peut pas être dans ces zones quand le système est dans ces états.

Probabilités de transition

Les transitions entre les états du modèle sont définies par la matrice A, présentée par la figure 5.3 et qui traduit les changements entre les états. Étant donné que les données utilisées indiquent la position de la personne, la matrice de transition va correspondre aux changements entre les états.

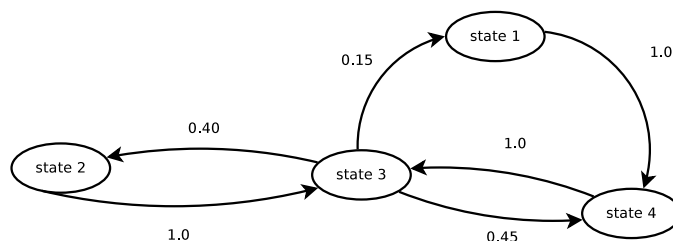


FIGURE 5.3 – Probabilité de transition entre les états

Trois des états présentent des transitions particulières, chacun ayant une probabilité de transition de 1 vers un autre état et les autres probabilités de transition étant nulles. De plus, le caractère transitoire de l'état 3 est confirmé par la matrice de transition. En effet, le passage des états correspond au couloir à celui correspondant aux zones autour du lit ne peut se faire qu'en passant par l'état 3. Cela permet de valider que des règles de transition physique peuvent être correctement apprises par le modèle.

Néanmoins, si on considère moins d'états, ou que la zone transitoire n'est pas correctement apprise, les propriétés de la matrice de transition ne pourraient pas être apprises et elles ne seraient pas très significatives. En effet, seules deux zones seraient alors correctement apprises, et les transitions entre ces zones ne seraient pas très significatives. Cela peut notamment s'expliquer par la simplicité de notre scénario et le nombre de fois qu'il a été enregistré.

Probabilités de stationnement

La durée de stationnement dans un état correspond au temps que la personne va rester dans une zone donnée avant de changer. La durée est définie par une moyenne et un écart-type, qui sont les paramètres de la gaussiennes. Cela va nous permettre, par exemple, de différencier le comportement d'une personne dormant six heures de celui d'une personne dormant huit heures, mais également si la personne dort exactement 7 heures ou entre 6 et 8 heures.

Les probabilités des durées de stationnement des états sont présentées dans la figure 5.4 et montrent différentes situations. Dans le scénario que nous avons utilisé, il y a un long temps d'attente dans la salle d'eau. On retrouve cela, du fait de la présence de deux états correspondant au couloir, dont l'état 1. Celui-ci a une moyenne de durée d'attente importante. Les probabilités de durées des états correspondent empiriquement au scénario appris.

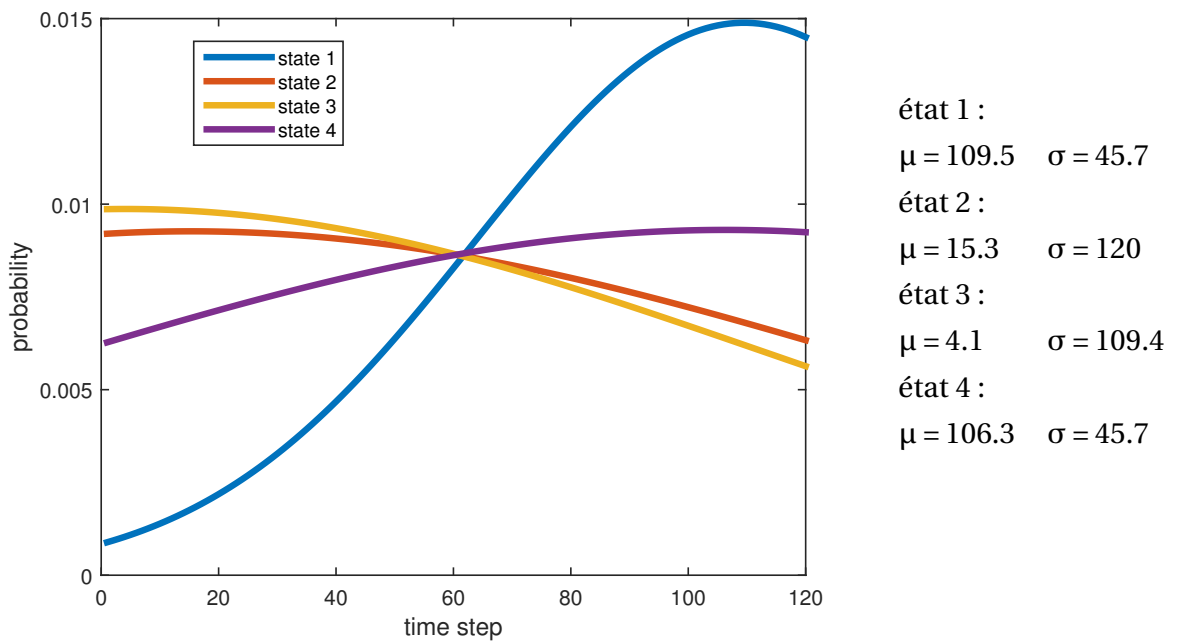


FIGURE 5.4 – Probabilité de durée des états

5.1.2 Discussions

Les différents paramètres du modèle appris correspondant au scénario effectué, le modèle de déplacement habituel de la personne peut être correctement appris dans le cas d'un scénario dans le living lab. Cette validation empirique doit néanmoins être discutée. En effet, il est à noter que le scénario est extrêmement simple, et un modèle avec une faible expressivité pourrait permettre de le définir.

De plus, seules les positions de la personne sont prises en compte, or le comportement de la personne ne se résume pas à sa position. Par exemple, connaître la position permet de définir que la personne est dans le lit, mais ne permet pas de différencier si la personne dort ou a une activité dans son lit comme lire. Pour cela, il faudrait prendre

en compte d'autres modalités de données, et donc, d'autres types de capteurs.

Une autre limite de notre expérimentation est le choix des paramètres initiaux du modèle, c'est à dire le nombre d'états cachés et la durée de stationnement maximale dans un état. Le scénario étant très simple et maîtrisé, nous pouvons définir facilement la valeur des paramètres. Néanmoins, la vraisemblance des modèles appris est fortement dépendante de ces paramètres. La vraisemblance pouvant être considérée comme la proximité du modèle à la réalité. Ainsi, maximiser la vraisemblance est un élément très important, notamment pour pouvoir ensuite utiliser le modèle pour détecter les évolutions. On observe par exemple, qu'à partir de 3 états, deux d'entre eux correspondent au couloir permettant d'accéder à la salle de bain. Utiliser un modèle contenant 4 états permet l'apparition d'un état transitoire entre les zones autour du lit et celles du couloir, qui est un état extrêmement significatif pour notre modèle.

S'il est possible de choisir ces paramètres à partir d'*a priori* sur le type d'activité de la personne, le temps qu'elle dort et qu'elle est à l'extérieur de sa chambre, ce choix n'est pas optimal. Une manière de définir ces paramètres de manière optimale serait d'augmenter la valeur des paramètres jusqu'à maximiser un critère donné. Cette méthode est utilisée pour définir le nombre d'états cachés d'un [HMM](#) en utilisant le critère d'Akaike ([Parzen et al. \[2012\]](#)). Une limite de cette méthode est que nous avons deux paramètres dépendants à optimiser : le nombre d'états et la durée de stationnement. Une amélioration de notre méthode serait donc de trouver un critère permettant d'optimiser les deux paramètres simultanément.

L'utilisation des modèles pour apprendre des modèles de déplacement habituel ont fait l'objet d'une publication dans [Paris et al. \[2015\]](#). Les tests effectués au sein du GIS Madonah permettent de valider la possibilité d'apprendre des modèles de déplacement habituel de la personne grâce à des capteurs [PIR](#) et à la méthode de localisation de la personne, présentée dans la sous-section [2.1.3](#). Dans la suite, nous testerons notre méthode sur des données simulées.

5.2 Apprentissage sur données simulées

5.2.1 Données simulées

Nous générons artificiellement des séquences d'observations, représentant les activités basiques (se lever, se laver, ouvrir la fenêtre, passage d'un aidant...) au sein d'une chambre d'[EHPAD](#). Pour cela, nous considérons un scénario simple permettant de simuler les données des capteurs sur une journée. Il est possible de répéter ce scénario

un grand nombre de fois afin de générer plus de données. Ce qui permettra d'avoir des données stochastiques plus représentatives et se rapprochant des données réelles.

Le scénario se compose d'un ensemble de 15 événements par jour, dont la durée depuis le dernier événement ou le moment de l'occurrence est défini aléatoirement. Ces événements permettent de définir un certain nombre de variables, comme le nombre de personnes présentes, l'état de la fenêtre ou le type de mouvements, qui sont utilisés pour générer aléatoirement un ensemble de 6 observations, à chaque pas de temps, et qui sont nécessaires pour l'algorithme d'apprentissage. Ces observations sont le moment du pas de temps dans la journée, le nombre de détections dans les zones du sas, de la chambre et de la salle d'eau, la présence dans le lit, ainsi que le niveau de CO₂.

Si nous pouvons ainsi simuler un grand nombre de jours différents, ce modèle reste une simulation, qui n'est pas représentative des problèmes pouvant apparaître en situation réelle. Par exemple, nous considérons que les capteurs fonctionnent correctement, et qu'ils ne présentent pas de défauts. De même, l'ensemble des données est généré à partir de probabilités identiques, or les données réelles varient en fonction du jour de la semaine, les activités étant notamment différentes entre les jours de la semaine et le samedi ou le dimanche. Ces variations peuvent dépendre des [EHPAD](#), ou aux changements des aidants, ou des activités proposées aux résidents ou encore, ils peuvent être extérieurs, dûs aux visites, par exemple.

Les données sont utilisées afin d'apprendre des modèles en faisant varier les paramètres du nombre d'états et de durée maximum, mais également le nombre de jours de données simulées utilisées dans l'apprentissage. Cela permet de définir le nombre de jours nécessaires afin que le modèle soit pertinent.

Afin de comparer les différents modèles, nous calculons leur vraisemblance par rapport à un ensemble d'observations. Il s'agit du premier problème des [HMM](#), décrit dans la section [3.4](#). L'algorithme permettant de calculer la vraisemblance est modifié pour le modèle [Gs-GsHMM](#) afin de prendre en compte les observations par gaussiennes et les durées de stationnement. Pour que les résultats soient les plus significatifs, nous les comparons à un même ensemble de données simulées qui n'est pas utilisé pour l'apprentissage. De plus, les jours simulés sont ordonnés. Ainsi, si nous considérons deux apprentissages avec 20 jours puis 30 jours de données simulées, les observations des 30 jours sont constituées des mêmes données pour les 20 premiers jours, auquel s'ajoutent les données simulées des jours 21 à 30.

Dans la sous-section suivante, nous envisageons d'utiliser les plans d'expériences afin d'évaluer les modèles appris grâce aux données simulées.

5.2.2 Étude des paramètres du modèles par plan d'expérience

L'utilisation de données simulées permet de mettre en place un plan d'expérience permettant d'évaluer l'influence des différents paramètres, notamment le nombre de jours de données utilisé pour la séquence d'observation.

Les plans d'expériences

On considère un processus dont le résultat ou les propriétés dépendent de multiples paramètres. Si ces différents paramètres ne s'influencent pas entre eux, il est possible de trouver la valeur optimale de chacun des paramètres indépendamment des autres paramètres. Au contraire, si les paramètres s'influencent entre eux, le choix de la valeur des paramètres doit se faire en prenant en compte la valeur de l'ensemble des paramètres. Pour cela, il serait nécessaire de tester l'ensemble des combinaisons possibles pour trouver la meilleure combinaison, ce qui peut être problématique si les tests nécessitent beaucoup de temps ou de moyens. Ainsi, l'objectif des plans d'expériences est de déterminer un résultat précis en un nombre minimum mais suffisant d'essai.

Les plans d'expériences permettant de déterminer la valeur d'un ensemble de paramètres afin d'optimiser le résultat en réduisant le nombre de tests nécessaires. Pour cela, l'influence entre les différents paramètres doit être déterminée. Un exemple de plan d'expériences est le plan en étoile, où à partir d'une première expérience, on fait varier à chaque fois l'un des paramètres.

Dans le cadre de notre expérimentation sur les données simulées, nous lançons un seul apprentissage avec la méthode Naïf pour chaque série de paramètres ; les résultats étant fortement stochastiques, mais au vu du caractère aléatoire de l'algorithme, une variabilité peut se présenter. Il serait possible de faire une série d'expériences pour chaque ensemble, afin de moyenner les résultats. Cela implique néanmoins d'augmenter le temps de calcul. La solution choisie a donc été d'augmenter le nombre de modèles d'origines de l'algorithme Naïf, et ainsi augmenter les chances de trouver un modèle correct.

Nous utilisons un plan d'expérience afin d'évaluer l'influence des différents para-

mètres en utilisant des données simulées. Nous évaluons ainsi trois paramètres différents. Le premier paramètre étudié est le nombre de jours utilisés pour apprendre le modèle, c'est à dire la quantité de données, sachant que celles-ci présentent un caractère cyclique. Le second paramètre est le nombre d'états du modèle et le dernier paramètre est la durée de stationnement maximum dans un état.

L'ensemble de ces paramètres peut prendre un grand nombre de valeurs. Par exemple, il serait ainsi possible d'apprendre un modèle avec, au minimum 2 états mais également avec un grand nombre d'états. Des tests ont ainsi été effectués sur les données simulées avec des modèles à 44 états. De même, la durée de stationnement maximal dans un état peut fortement varier. Un pas de temps ayant une durée de 60 secondes, la durée de stationnement peut s'exprimer en minutes. Le choix de la durée de stationnement maximum dépend des données que l'on souhaite apprendre et peut prendre de nombreuses valeurs.

Application

Dans le cadre de l'application aux données simulées, nous avons testé, pour chacun des paramètres, deux valeurs. Les résultats sont présentés dans le tableau 5.1. Le choix des valeurs permet d'avoir un plan orthogonal, afin de ne pas biaiser les résultats.

TABEAU 5.1 – Résultats des essais

Nombre de jours	Nombre d'états	Durées de stationnement maximum	log-vraisemblance
20	5	120	-252772
20	5	210	-248752
20	10	120	-225008
20	10	210	-231314
40	5	120	-248413
40	5	210	-254025
40	10	120	-240871
40	10	210	-216107

Dans un premier temps, nous calculons les effets des différents paramètres. Les expérimentations effectuées précédemment permettent de déterminer que l'effet du nombre de jours utilisé pour l'apprentissage est de -392.5 . L'effet du nombre d'états est de 22665.5 et l'effet de la durée de stationnement maximum est de 4216.5 .

La valeur des effets permet notamment de caractériser l'influence des différents paramètres entre les valeurs choisies. Le choix du nombre de valeurs est particulièrement significatif, mais les autres paramètres ont également une forte influence. De plus, les valeurs choisies pour les expérimentations ont été définies empiriquement, parmi un très grand choix de possibilités. Dans ce cas, il serait possible d'avoir d'autres résultats avec un autre choix de paramètres. Les tests actuellement effectués permettent de définir qu'il est globalement plus efficace de choisir un modèle avec 10 états cachés qu'avec 5, et une durée de stationnement maximal de 210 pas de temps plutôt que 120 pas de temps. Cela ne permet néanmoins pas, actuellement, de définir la valeur optimale pour les différents paramètres. Pour cela, il serait nécessaire d'étudier les variations de chacun des paramètres.

5.2.3 Étude des modèles issues des données simulées

Les résultats sont fortement bruités et seules les variations globales peuvent être interprétées. En effet, bien que la log-vraisemblance soit calculée en utilisant les mêmes observations de références, les modèles sont calculés grâce à l'algorithme stochastique de Baum-Welch. La version Naïve permet de calculer plusieurs modèles et seul celui maximisant la vraisemblance est gardé. Si cela permet que le résultat final soit correct, il n'est pas possible d'assurer que le modèle trouvé soit optimal.

Plus le nombre d'exécutions de l'algorithme de Baum-Welch est important, plus les chances de trouver un modèle ayant une vraisemblance importante augmente, au prix d'une augmentation du temps de calcul. De ce fait, de nouveaux modèles appris avec les mêmes paramètres et les mêmes observations seraient différents de ceux obtenus lors des tests effectués. Ainsi, certains modèles peuvent présenter des log-vraisemblances fortement différentes de celles de modèles appris avec des paramètres proches. De plus, ces résultats ne sont pas moyennés par plusieurs tests.

Afin de confirmer les variations, certains tests ont été effectués deux fois, permettant ainsi d'avoir deux ensembles de valeurs. On considère alors les valeurs maximum et minimum. Il faut noter qu'il ne s'agit pas des valeurs maximums ou minimums de la log-vraisemblance de modèles appris pour un ensemble de paramètres d'entrées définis, mais des valeurs trouvées pour des apprentissages Naïfs distincts. Ces valeurs distinctes permettent de vérifier si les variations apparaissent aléatoirement, dû au caractère stochastique de l'algorithme de Baum-Welch, ou si une variable a bien une influence précise sur la qualité du modèle. Dans ce cas, les valeurs minimums et maximums présentent les mêmes variations, permettant de définir l'influence des para-

mètres.

Nombre d'états cachés

La variation du nombre d'états cachés, sur la même base d'apprentissage et en gardant les autres paramètres constant permet d'étudier l'influence du paramètre sur la log-vraisemblance. Différents cas sont testés et présentés sur la figure 5.5. On observe globalement une augmentation de la log-vraisemblance quand le nombre d'états cachés croît. Celle-ci n'est néanmoins pas constante, et semble converger. Ainsi, l'augmentation de la log-vraisemblance est très importante pour des modèles avec peu d'états et devient de plus en plus faible, et n'est plus significative à cause de la variabilité des résultats dû au caractère stochastique de l'apprentissage.

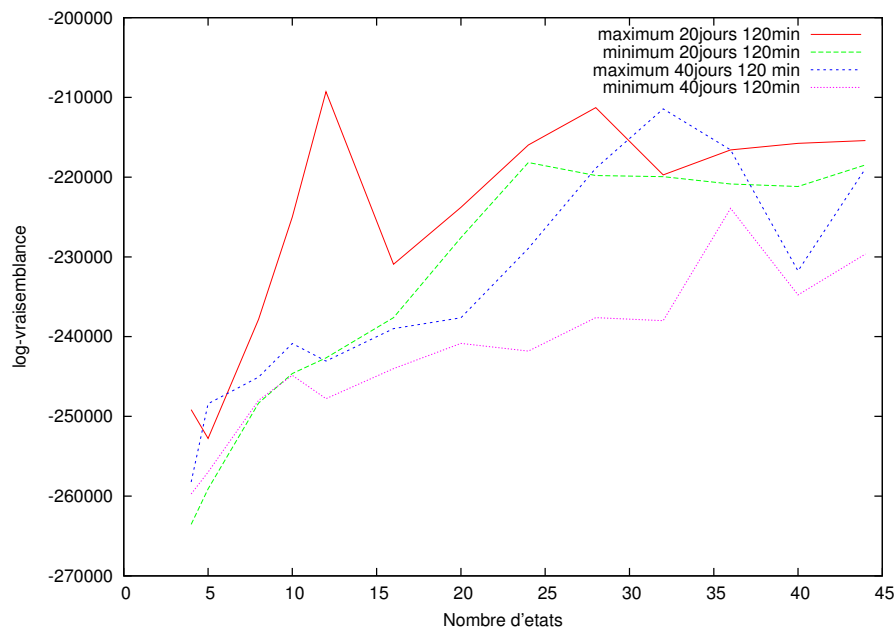


FIGURE 5.5 – Variation de la log-vraisemblance en fonction du nombre d'état cachés

Si on retrouve globalement les mêmes variations, la progression de la courbe est différente à cause de l'impact des autres paramètres. Cela confirme que le choix du nombre d'états cachés doit se faire en fonction des autres paramètres et que l'utilisation de la log-vraisemblance uniquement ne permet pas de trouver une valeur optimale.

On observe, pour 12 états cachés, un résultat particulièrement élevé de la log-vraisemblance. C'est un cas spécifique, caractérisant le caractère stochastique de l'algorithme, mais qui reste moins significatif par rapport aux autres résultats. En effet, les

résultats pour ces paramètres sont en général inférieurs à ce cas particulier.

Les différents tests semblent confirmer le caractère convergent, avec un maximum qui varie en fonction de la durée de stationnement maximal. Il y a néanmoins de fortes variations si les valeurs sont proches dans le cas des tests avec 20 jours et 120 minutes. Au contraire, ceux avec 40 jours et 120 minutes présentent de plus grandes variabilités et la différence entre les résultats maximums et minimums est plus importante.

Compte tenu que la log-vraisemblance est globalement croissante, il serait possible d'utiliser un critère comme [critère d'information d'Akaike ou Akaike Information Criterion \(AIC\)](#) ou [critère d'information bayésien ou Bayesian Information Criterion \(BIC\)](#) pour optimiser le nombre d'états, comme dans le cas des [HMM](#). Néanmoins, le choix du nombre d'états étant dépendant des autres paramètres, il est nécessaire d'étudier également leurs variations.

Durée de stationnement maximum

La variation de la log-vraisemblance en fonction de la durée de stationnement maximum dans un état est étudiée. Pour cela, la log-vraisemblance est calculée pour des modèles appris avec des paramètres constants et en ne faisant varier que la durée de stationnement maximum. Les résultats sont présentés par la figure [5.6](#).

Les tests menés en prenant 20 jours de données et 12 états sont doublés afin de vérifier la pertinence des résultats, tandis que ceux effectués pour 40 jours et 10 états confirment les résultats.

On observe de grandes variations de la log-vraisemblance des modèles appris, sans qu'une structure globale n'apparaisse. Celles-ci ne sont néanmoins pas exclusivement dûes au bruit. En effet, les résultats des tests avec 20 jours et 12 états présentent les mêmes changements et une grande symétrie, hormis dans le cas d'une durée de stationnement de 120 minutes, où la différence entre le minimum et le maximum est la plus importante. Les résultats présentent des fluctuations globalement moins importantes en augmentant la durée de stationnement maximale. Cette réduction n'implique pas que les modèles appris avec des durées de stationnement important soient plus intéressantes que d'autres appris précédemment, les valeurs pouvant être inférieures, y compris par rapport à des valeurs minimums précédemment appris.

Le second cas étudié confirme les résultats précédents. On observe que si la valeur de la convergence est proche, les durées pour lesquelles la log-vraisemblance est la plus intéressante, sont différentes du premier cas. De même, les variations sont diffé-

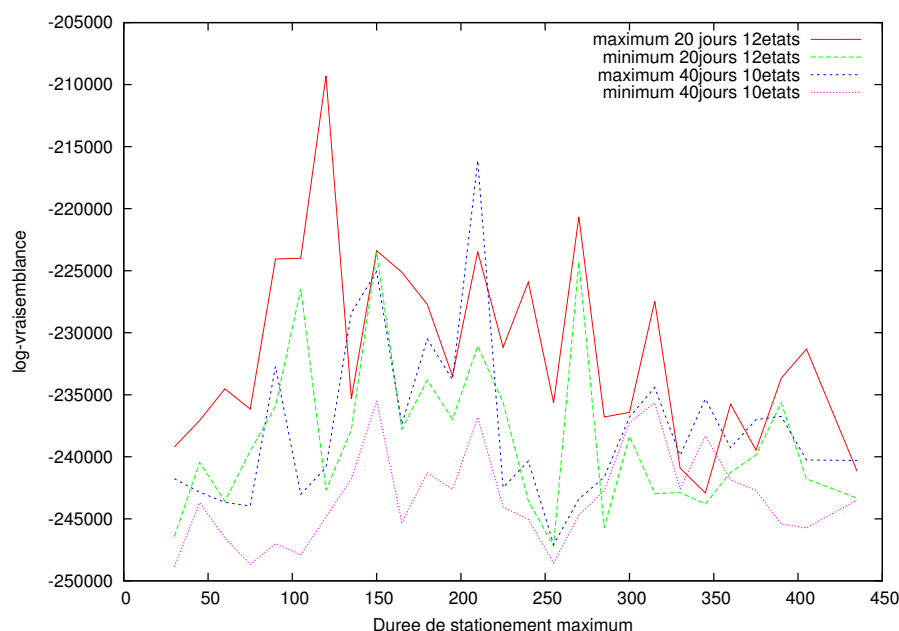


FIGURE 5.6 – Variation de la durée de stationnement maximum dans un état

rentes une fois de plus, bien que les valeurs des différents paramètres soient proches. Ce qui confirme l'influence des autres paramètres sur le choix de la durée de stationnement maximal.

Les tests menés confirment la nécessité de définir une durée de stationnement maximum optimale, variant en fonction du nombre d'états. Les variations de la log-vraisemblance en fonction de la durée de stationnement maximum ne permet pas de sélectionner la valeur optimale en l'augmentant jusqu'à la maximisation d'un critère comme [AIC](#) ou encore jusqu'à la convergence.

Nombre de jours de la séquence d'apprentissage

Les variations de la log-vraisemblance des modèles appris en fonction du nombre de jours utilisés dans la base d'apprentissage sont présentées dans la figure [5.7](#).

On observe globalement une diminution de la log-vraisemblance quand le nombre de jours utilisés augmente. Seuls les tests avec 24 états cachés et une durée de stationnement de 210 minutes présentent une légère augmentation dans le cas où peu de données sont utilisées.

Les données sont générées par simulation. Elles ne présentent pas d'imperfections comparées aux données acquises en situation réelle. On peut supposer que très peu de données suffisent à les modéliser correctement. Les jours supplémentaires provoquent

5.2. Apprentissage sur données simulées

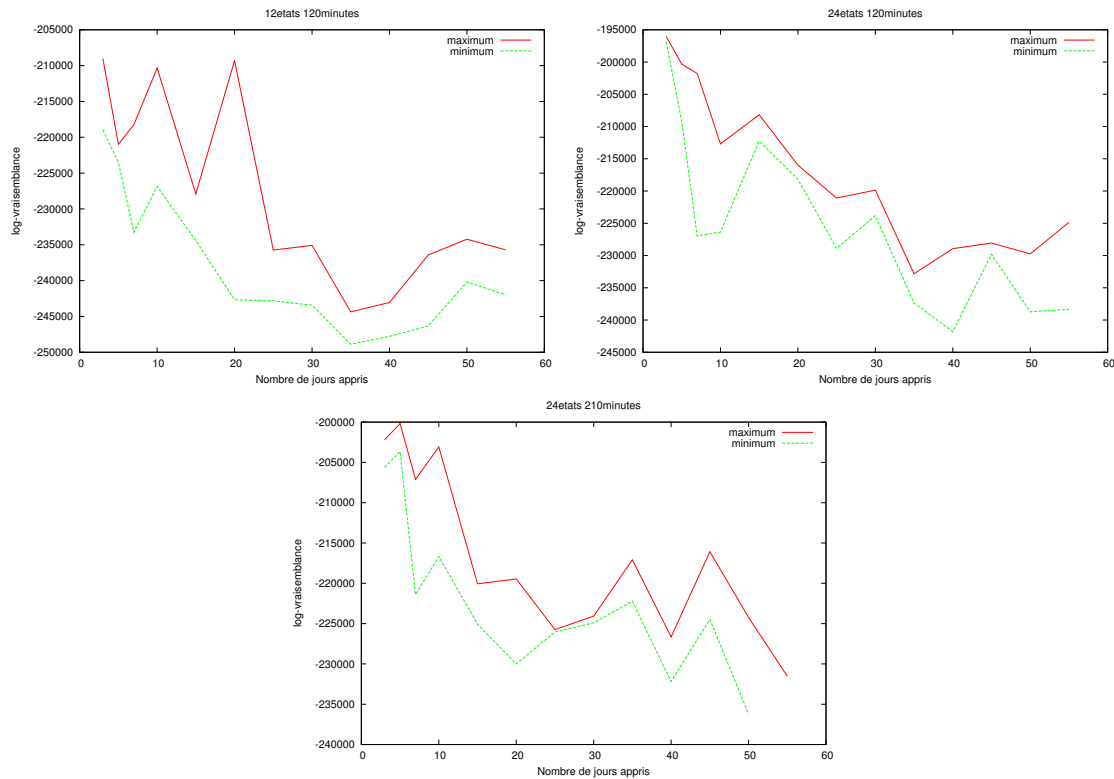


FIGURE 5.7 – Variations de la log-vraisemblance en fonction du nombre de jours appris

raient alors un sur-apprentissage, expliquant les diminutions de la log-vraisemblance. Dans ce cas, l'augmentation de la log-vraisemblance, pour un faible nombre de jours utilisés, indiquerait alors une phase de sous-apprentissage, quand il n'y a pas assez de données. Cela est particulièrement vrai pour les modèles avec un grand nombre d'états, ce qui implique un grand nombre de paramètres à apprendre, et dans les cas où les paramètres permettent d'apprendre des modèles de qualité.

Ces variations indiquent, dans le cadre des données artificielles, un nombre de jours utilisés pour l'apprentissage optimal, et dépendant des autres paramètres. Si cette valeur optimale est faible dans le cas de l'apprentissage des données simulées, on peut supposer que, les données réelles étant plus compliquées à apprendre, cette valeur est beaucoup plus importante dans le cas de données acquises en situation réelle.

5.2.4 Discussions

Bien que permettant de confirmer certaines hypothèses, l'étude des données reste limitée. En effet, les modèles appris restent basés sur des données simulées, qui ne sont pas représentatives de l'ensemble des variations en situations réelles. Cela peut par

exemple expliquer pourquoi peu de jours simulés permettent d'apprendre le modèle. Néanmoins, les données simulées valident la nécessité d'optimiser les paramètres. Cette optimisation doit se faire de manière efficace mais également en prenant en compte l'interaction entre paramètres.

De plus, les résultats sont fortement stochastiques. S'il est possible d'observer des variations globales, effectuer un seul test, ou même deux tests distincts, n'est pas très significatifs. Il serait plus intéressant d'évaluer la moyenne calculée sur un grand nombre d'expérience. Cela implique néanmoins un temps de calcul important, ce qui n'est pas possible dans le cas d'une application en situation réelle. Sachant que le choix des paramètres optimaux doit se faire en un temps de calcul acceptable.

Si le nombre d'états et de la durée de stationnement maximale du modèle ont de fortes influences sur la vraisemblance du modèle appris, le choix du nombre de jours de données utilisées pour l'apprentissage a également une influence sur l'utilisation de la méthode en situation réelle. En effet, déterminer le nombre de jours minimum afin de permettre un apprentissage efficace permettrait de réduire considérablement le temps de calcul, compte tenu de la complexité de l'algorithme utilisé, mais également de mettre plus souvent à jour le modèle afin que celui-ci reste le plus proche du véritable comportement de la personne.

Après avoir testé l'apprentissage des modèles sur des données simulées, nous envisageons, dans la section suivante, de le tester sur des données acquises en situation réelle.

5.3 Application à l'Apprentissage en situation réelle

5.3.1 EHPAD d'Ambazac

Dans le cadre du projet, plusieurs chambres de l'[EHPAD](#) d'Ambazac ont été équipées d'un ensemble de capteurs ce qui nous a permis d'avoir une base de données en situation réelle, à la fois au niveau de la typologie des chambres que du comportement des résidents. En effet, les données sont issues des chambres habitées par des résidents de l'[EHPAD](#). Cela implique également qu'il n'y a pas de scénario auquel se référer, afin de tester nos résultats. Mais ces derniers peuvent être partiellement validés par les aidants, compte tenu de leurs connaissances du comportement, des habitudes de vie et des pathologies des résidents.

5.3. Application à l'Apprentissage en situation réelle

Différents capteurs sont testés en situation réelle au sein de l'EHPAD d'Ambazac, notamment des capteurs d'alitement ou environnementaux. Nous utilisons l'ensemble des capteurs PIR, de façon équivalente à l'expérimentation précédente, avec la méthode développée par Amri *et al.* [2015]. Dans chacune des chambres équipées, un ensemble de 5 capteurs PIR. Ces capteurs sont placés de façon standardisée dans l'ensemble des chambres qui ont toutes le même plan et comme représentées sur la figure 5.8.

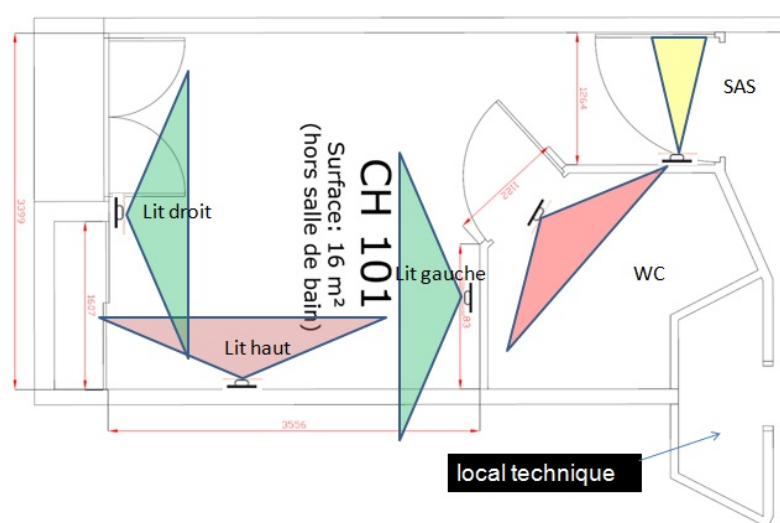


FIGURE 5.8 – Plan d'une chambre de la maison de retraite Ambazac avec position des capteurs.

Le capteur SAS est situé au niveau de la porte et orienté verticalement, afin de détecter les entrées et sorties de la chambre. Un second capteur, "lit haut", est placé au dessus du lit placé à l'envers, afin de détecter les personnes debout uniquement. Les capteurs "lit fenêtre" et "lit salle d'eau" sont placés de part et d'autre du lit à 30cm du sol afin de détecter les pieds des personnes. Le dernier capteur est placé dans la salle d'eau. Les données capteurs servent à nourrir une base de données que nous pouvons utiliser pour nos tests.

Un capteur supplémentaire est utilisé pour mesurer la teneur en CO₂ dans l'environnement, mesure réalisées en ppm. Celle-ci augmente naturellement quand une personne est présente dans une pièce non ventilée, du fait de la respiration. Cette mesure permet donc de détecter, au bout d'un certain temps, la présence d'une personne quand elle ne bouge pas et notamment la nuit.

Test 1

Données utilisées Dans le cadre de cette expérience, nous disposons d'une base de données de 26 jours consécutifs. Les 5 derniers jours sont exclus de la base d'apprentissage afin de servir de base de validation pour les modèles appris. Ainsi, notre base d'apprentissage est limitée à 21 jours consécutifs.

Compte tenu du nombre de jour et des activités que nous souhaitons détecter, utiliser comme pas de temps la seconde n'est pas forcément très pertinent. La complexité de l'apprentissage dépendant fortement du nombre de pas de temps observé, il peut être plus intéressant de considérer un pas de temps plus long. Cela implique un nouveau paramètre, qui est la période du pas de temps. La méthode de [Amri et al. \[2015\]](#) permettant de fournir des résultats toutes les secondes, nous convertissons les données en un vecteur d'observation, utilisable par l'algorithme d'apprentissage. Pour cela, nous considérons le pourcentage d'activation des zones durant la période du pas de temps choisi.

Les activités de la personne dépendant de l'heure, aux données issues des capteurs [PIR](#), nous ajoutons une variable temporelle, dont la valeur dépend de l'heure et des minutes. Étant donné que nous nous intéressons particulièrement au comportement la nuit, nous ne convertissons pas directement l'heure en un entier, notamment pour éviter une rupture à minuit. Nous convertissons donc en décalant de 12h, afin que la rupture se fasse à midi, quand les résidents ne sont généralement pas dans leurs chambres. Sans oublier que nous intégrons les données issues du capteur de CO2.

Chaque observation se compose donc de 7 éléments. La figure 5.9 présente un exemple, sur une durée de 24 heures des données des capteurs [PIR](#) utilisés dans le cadre de l'apprentissage.

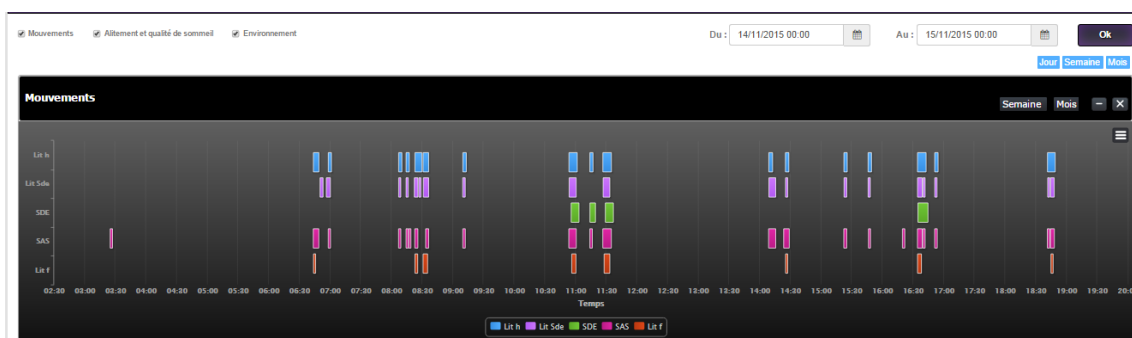


FIGURE 5.9 – Données des capteurs PIR

Résultats Différents ensembles de paramètres ont été utilisés, que ce soit le nombre d'états cachés, la durée de stationnement maximum ou la durée du pas de temps pour générer la séquence d'observations. En fonction de ces paramètres, différents modèles ont pu être appris.

Ces modèles ne semblent pas être très pertinents et la séquence observée n'est pas correctement apprise. En effets, on observe notamment que les écart-types appris, que ce soit pour les observations ou les durées de stationnement sont globalement très importants, par rapport à l'ensemble des valeurs. Cela implique que les valeurs correspondantes sont globalement équiprobable sur l'ensemble des valeurs possibles. Les modèles généralisent donc énormément les données, et ne sont pas représentatifs du comportement de la personne.

Test 2

Pour ce test, nous utilisons une base d'apprentissage de 23 jours, acquises dans la même chambre, et une base de validation de 3 jours, acquise 15 jours plus tard. Comme dans le cadre de l'expérience 1, nous testons l'apprentissage en considérant un pas de temps d'une durée de 60 secondes. Nous considérons pour cette expérimentation un vecteur d'observations de 6 éléments : l'heure et le niveau de CO2 restant identiques à l'expérimentation 1, le nombre de secondes durant lesquelles les capteurs **PIR** sont activés, deux éléments correspondant au nombre de seconde d'activation d'un capteur spécifique (les capteurs "SAS" et "salle d'eau") et la somme du nombre de seconde d'activation des trois capteurs situés dans la chambre.

Comme précédemment, nous faisons varier le nombre d'état et la durée de stationnement maximum afin de tester les effets sur la log-vraisemblance.

Variations des paramètres Les modèles sont appris en faisant varier le nombre d'états, ainsi que la durée de stationnement maximum dans un état. Comme dans le cas des données simulées, nous effectuons deux apprentissages pour chaque couple de paramètres, afin de réduire le caractère stochastique des évolutions possibles. On obtient donc deux représentations des évolutions, présentées par la figure 5.10.

Comme pour les données simulées, on observe une évolution cohérente des différentes valeurs. Si l'évolution en fonction du nombre d'états est particulièrement significative, la durée de stationnement maximum a également une influence sur la qualité de l'apprentissage. Dans la suite nous envisageons d'évaluer l'influence de cette durée.

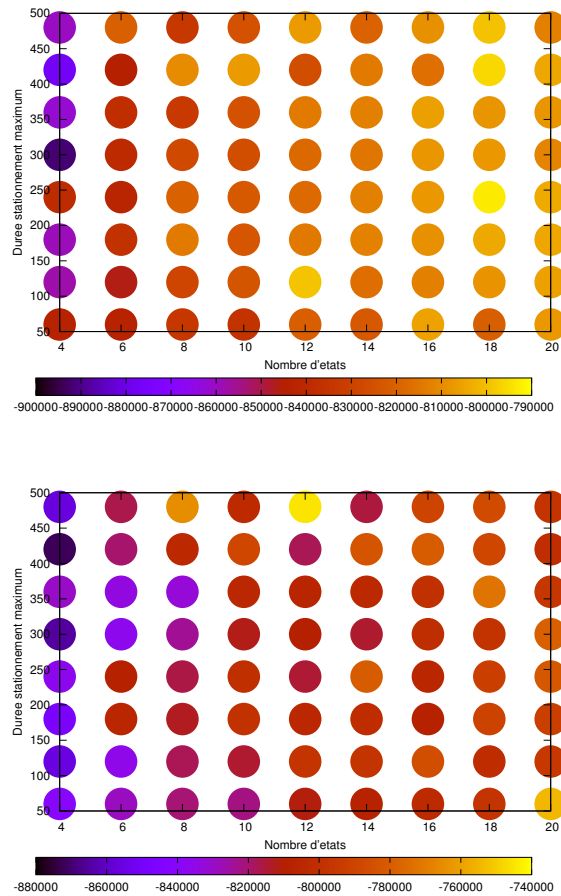


FIGURE 5.10 – Variations de la log-vraisemblance en fonction du nombre d'états et de la durée de stationnement maximum.

Pour cela, nous apprenons des modèles avec des séquences d'apprentissage de différentes longueurs.

Les résultats sont présentés à la figure 5.11. Les tests effectués avec 20 états et une durée de stationnement maximum de 420 minutes permettent de définir deux phases. La première phase correspond aux apprentissages avec moins de 21 jours de données utilisées, tandis que la seconde phase correspond de 21 à 25 jours. La première phase correspond à des log-vraisemblances inférieures à -107000, tandis que pour la seconde phase, elles sont comprises entre -87000 et -95000. Cette variation permet de définir, dans le cadre de cette expérimentation, le nombre de jours minimum de la séquence d'observations permettant l'apprentissage. Étant donné que cette valeur est très proche du nombre de jours que nous pouvons utiliser, si nous pouvons apprendre le modèle de comportement habituel de la personne, celui-ci n'est pas optimal. Il serait

5.3. Application à l'Apprentissage en situation réelle

nécessaire d'utiliser davantage de données afin d'augmenter la vraisemblance des modèles appris. Ce qui permettrait de rapprocher davantage les modèles appris au comportement habituel de la personne.

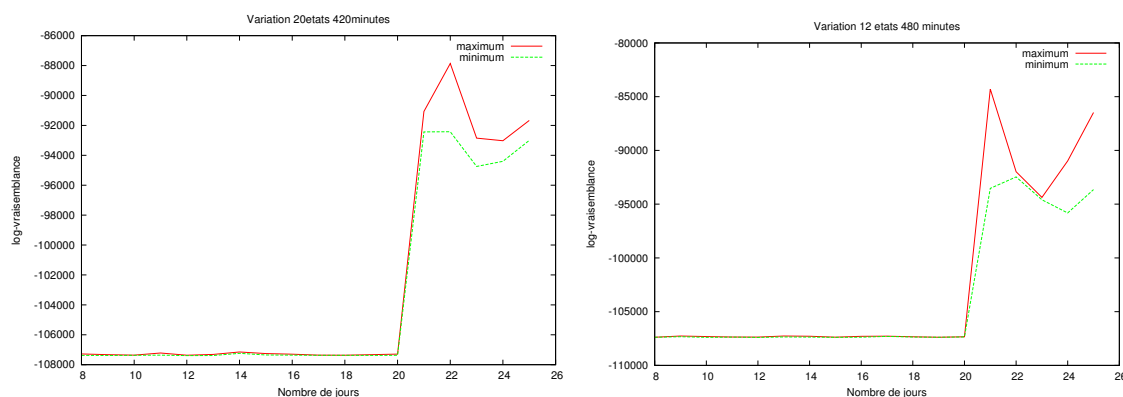


FIGURE 5.11 – Variations de la log-vraisemblance en fonction du nombre de jours utilisés pour l'apprentissage.

Ces résultats doivent être validés par les tests effectués avec 12 états et une durée de stationnement maximum dans les états de 480 minutes. On observe dans le cadre de ces tests les même phases que dans le cas des premières expérimentations, avec un passage de log-vraisemblances inférieures à -107000 à des log-vraisemblances supérieures à -93000. Le nombre de jours de la base d'apprentissage utilisés pour passer d'une phase à l'autre est le même, confirmant la quantité de données nécessaire pour apprendre le modèle. Si le nombre de jours est le même dans le cadre des deux tests présentés, celui-ci n'est pas fixe. Par exemple, dans le cadre des tests avec 20 états et une durée de stationnement maximum de 480 minutes, le changement de phase se fait pour 25 jours appris. Si la base d'apprentissage est la même, le nombre de paramètres à apprendre est différent, comme vu dans la partie 4.2.5. Afin de pouvoir déterminer le nombre de jours optimal à utiliser afin d'apprendre le modèle, il serait nécessaire de tester l'évolution de la log-vraisemblance, notamment en utilisant davantage de jours de données.

5.3.2 Discussion

L'échec de l'apprentissage peut être expliqué par le fait que la base d'apprentissage n'est pas suffisante pour apprendre l'ensemble des paramètres. Pour valider cette hypothèse, il serait nécessaire de tester l'apprentissage sur une base de données contenant plus de jours, étant donné l'évolution de la log-vraisemblance en fonction du

nombre de jours appris. Il est également possible que les données ne soient pas suffisantes pour caractériser le comportement de la personne. Il est possible que plus de données permettrait de caractériser les différentes activités effectuées dans la chambre, et donc de pouvoir caractériser des états. Pour cela, il serait nécessaire d'avoir plus de capteurs, avec des modalités différentes. De plus, il faudrait également tester la méthode dans le cas d'un résident ayant davantage d'activités. En effet, le résident de la chambre utilisé a un comportement apathique. De ce fait, il y a peu d'activités effectuées par le résident et un grand nombre des mouvements dans la chambre sont effectués par les aidants. À l'inverse, il y a de très nombreuses périodes d'inactivité dans la chambre, y compris quand le résident s'y trouve.

5.4 Conclusion

Les applications effectuées permettent de valider l'utilisation des [Gs-GsHSMM](#) afin d'apprendre des modèles de déplacement habituel dans un cas maîtrisé, ou un modèle de comportement habituel au sein d'une chambre simulée. L'application du modèle dans des conditions réelles reste à discuter. En effet, le modèle utilisé ne prend pas en compte l'ensemble des problématiques pouvant apparaître en condition réelle. Afin d'apprendre un modèle de comportement habituel efficace, il pourrait être nécessaire d'utiliser une base d'apprentissage contenant plus de données.

5.4. Conclusion

Chapitre 6

Étude des évolutions du modèle Gs-GsHSMM

Sommaire

6.1 Objectifs	82
6.2 Méthodes de calcul de distance	84
6.2.1 Distance hybride Kullback-Leibler et Bhattacharyya	84
6.2.2 Distance KL appliquée au Gs-GsHSMM	85
6.2.3 HSD	85
6.2.4 Distance statistique normalisée	86
6.3 Analyse de l'évolution du comportement	87
6.3.1 Modèle évolutif de test	87
6.3.2 Comparaisons des distances appliquées au modèle	89
6.3.3 Influence de l'apprentissage sur le calcul des distances	94
6.3.4 Calcul des distances sur modèle évolutif appris	95
6.3.5 Discussion sur les méthodes	97
6.4 Distance KL convergente	98
6.4.1 Contraintes	98
6.4.2 Présentation et implémentation	99
6.4.3 Résultats	101

6.1 Objectifs

La détection d'anomalie est une classe de problèmes qui consiste à trouver si un ensemble de données est conforme au comportement supposé d'un système et sinon quelle partie des données ne correspond pas. Ce type de problèmes et les méthodes utilisées pour les résoudre ainsi que différents domaines d'applications sont décrites par [Chandola *et al.* \[2012\]](#). [Gupta *et al.* \[2014\]](#) présentent les méthodes pour la détection d'anomalies dans le cadre d'évolution temporelle. Plusieurs méthodes de détection d'anomalies utilisent des modèles de Markov, par exemple la méthode décrite par [Goernitz *et al.* \[2015\]](#).

Nous considérons un cas spécifique du problème de la détection d'anomalie où le système étudié a un comportement inconnu et peut évoluer dans le temps. Le problème n'est plus alors de chercher une anomalie dans les données, mais de trouver une anomalie entre les données et une modélisation du système généré grâce à des données antérieures. L'anomalie va alors correspondre à une évolution du système. Un problème complémentaire est alors de caractériser l'évolution du système. Compte tenu du système, nous supposons qu'il est complexe et inconnu, modélisable par un [Gs-GsHMM](#) tel que décrit dans la section 4.2.

Étant donné que notre modèle représente le comportement habituel de la personne, détecter des évolutions dans le modèle doit nous permettre de détecter des dérives lentes du comportement. Pour cela, le modèle du comportement habituel de la personne est appris plusieurs fois, par exemple tous les mois. Il est donc possible d'avoir plusieurs modèles à différents moments. En calculant la distance entre ces derniers, il serait possible de détecter les évolutions des habitudes de la personne au cours du temps.

Compte tenu du type de modèle, nous pouvons considérer trois types d'évolutions de base, chacune est liée à un élément du modèle. Un premier ensemble de changements est lié à la matrice de transition A . Ce qui correspond dans notre application aux activités que la personne va passer plus, ou moins, de temps à effectuer au cours du temps. Un second type de changement concerne les probabilités de stationnement dans les états Δ . Ces modifications ont plus de sens, étant représentatives du temps que passe une personne à effectuer une activité. Le dernier ensemble est lié aux probabilités d'observation τ . En première approximation, nous pouvons considérer que ces évolutions vont être liées aux capteurs, pouvant par exemple correspondre à un capteur devenant moins précis, ou cessant d'émettre. Néanmoins, des changements dans la façon d'effectuer une activité peuvent être représentés par les probabilités d'obser-

vation, par exemple, si la personne ne se déplace plus dans l'ensemble de la chambre. Une méthode pour calculer la distance entre deux modèles doit permettre de détecter tous les types d'évolutions. En effet, les distributions de deux modèles peuvent être très différentes, alors que la différence n'affecte qu'un seul paramètre. Cette contrainte peut néanmoins être relâchée, compte tenu du cadre de notre étude. En effet, on peut supposer qu'une évolution de l'état général de la personne affectera au moins deux éléments du modèle.

D'autres contraintes doivent être prises en compte. En effet, afin de détecter une évolution du comportement, il faut calculer les distances entre les modèles au cours du temps. De ce fait, la méthode de calcul de distance doit pouvoir retourner un résultat en temps de calcul acceptable.

Différentes méthodes permettent de calculer la distance entre deux modèles de Markov, notamment entre des HMM. Par exemple, la méthode présentée par Silva et Narayanan [2004] se base sur la divergence de Kullback-Leibler (KL) dans le cas spécifique de modèle gauche-droite, très utilisé dans la reconnaissance de la parole. Falkhausen *et al.* [1995] se basent également sur la divergence de KL, ainsi que sur Viterbi pour calculer efficacement la distance. Pour cela, il est nécessaire que les deux modèles testés aient le même nombre d'états. La méthode présentée par Xie *et al.* [2005] utilise le taux d'entropie relatif. Pour cela, ils génèrent une séquence d'observations qui est utilisée pour calculer récursivement les états d'information pour chacun des modèles. La séquence d'observations étant générée de manière stochastique, la longueur T de cette dernière a une influence sur la qualité de la méthode. Pour avoir un résultat exact, il faut faire tendre T vers l'infini. Il existe d'autres méthodes pour calculer la distance entre deux HMM. Nous choisissons d'en tester plus spécifiquement plusieurs d'entre elles que nous décrirons dans la section suivante.

Les méthodes testées l'ont été afin que chacune ait une approche spécifique du problème. Ainsi, si la première est très géométrique, la seconde se base sur la génération d'observations des modèles, la troisième sur des propriétés spécifiques des modèles de Markov, et la quatrième est statistique et se base sur les événements propres aux modèles. Ces différentes approches, et leurs applications, ont donc été étudiées.

6.2 Méthodes de calcul de distance

6.2.1 Distance hybride Kullback-Leibler et Bhattacharyya

La première distance consiste à calculer séparément les distances entre les matrices de transitions A d'une part, et les distances entre les distributions gaussiennes des durées Δ et des observations τ .

Pour calculer la distance entre les matrices de transition, on utilise la divergence de **KL**, défini par $D_{KL}(P, Q) = \sum_i P_i \log \frac{P(i)}{Q(i)}$, avec P et Q deux distributions de probabilités discrètes. La divergence de **KL** n'est pas symétrique, donc on calcule la distance comme la moyenne des distances $D_{KL}(P, Q)$ et $D_{KL}(Q, P)$. Il serait possible d'utiliser d'autres méthodes de calcul de distance entre des distributions de probabilité que la divergence de **KL**. **Österreicher** [2002] présente ainsi différentes divergences permettant de calculer la distance entre deux distributions de probabilités discrètes et leurs propriétés. La distance entre les matrices de transition est la somme des distances, élément par élément, entre les matrices.

Pour calculer la transition entre les éléments Δ et τ des modèles, qui sont définis par des distributions gaussiennes, on utilise la distance de Bhattacharyya. Celle-ci est appliquée entre les gaussiennes équivalentes des modèles et leurs valeurs sont sommées afin d'avoir un résultat global.

Il est à noter que la méthode d'apprentissage ne respecte pas l'ordre des états. Calculer la distance entre deux modèles λ_1 et λ_2 consiste donc à calculer toutes les distances entre le modèle λ_1 et les modèles définis par toutes les inversions de l'ordre des états possibles de λ_2 , et seule la distance minimale est gardée.

La divergence de **KL** a l'inconvénient d'être sensible à la division par 0, impliquant que la distance entre deux modèles peut tendre vers ∞ , alors que la réalisation des deux distributions est extrêmement proche. Par exemple, soit trois distributions de probabilité P_1 , P_2 et P_3 , tels que $P_1(r = 0) = 0$, $P_2(r = 0) = 0.001$, $P_3(r = 0) = 0.002$ et $P_i(r = 1) = \overline{P_i}(r = 0)$. Ces trois distributions ont des réalisations très proches. La distance entre les distributions P_2 et P_3 sera faible alors que les distances avec la distribution P_1 sera ∞ . Afin d'éviter ce problème, nous ajoutons ϵ aux éléments nuls du dénominateur pour éviter les divisions par 0. Cette variable doit être suffisamment petite pour ne pas fausser les résultats.

6.2.2 Distance KL appliquée au Gs-GsHMM

Cette distance est une application de la distance décrite par [Rabiner \[1989\]](#) pour les [HMM](#). La distance entre deux modèles λ_1 et λ_2 est définie par la formule 6.1, avec O une séquence d'observation générée par le modèle λ_2 durant T pas de temps. De même que pour la distance précédente, cette méthode n'est pas symétrique. La séquence O étant générée de façon probabiliste, la distance calculée entre deux modèles n'est pas déterministe et les résultats sont différents à chaque calcul. Les différences avec la méthode appliquée aux [HMM](#) résident dans la génération de la séquence d'observations et dans le calcul de la log-vraisemblance, qui doivent être adaptés aux [Gs-GsHMM](#).

$$D_{\text{KL}_{\text{Rabiner}}}(\lambda_1, \lambda_2) = \frac{1}{T} (\log(P(O|\lambda_1)) - \log(P(O|\lambda_2))) \quad (6.1)$$

Le nombre de pas de temps de la séquence d'observation O est un facteur important pour le calcul de la distance. Le nombre de pas de temps doit être suffisamment important pour être représentatif du modèle générateur.

Afin de calculer le plus précisément la distance, il faudrait faire tendre T vers l'infini ($T \rightarrow \infty$). Néanmoins, un grand nombre de pas de temps implique également une augmentation importante du temps de calcul, à la fois pour la génération de la séquence O et pour calculer les log-vraisemblances.

La séquence d'observation générée doit donc être suffisamment longue afin que celle-ci soit représentative du modèle. Pour cela, la longueur doit être choisie en prenant en compte les modèles étudiés, notamment les durée de stationnement maximales dans un état, ou le nombre d'états, en particulier quand certaines probabilités de transition sont faibles. T doit donc être suffisamment important pour que tous les états significatifs du modèle aient de fortes probabilités d'être représentés.

6.2.3 HSD

La [distance stationnaire HMM](#) ou *HMM Stationary Distance (HSD)*, décrite par [Zeng et al. \[2010\]](#), est utilisée pour calculer la distance entre deux Gaussian-HMM, en utilisant l'équation 6.2 avec $F_i(x)$ la distribution cumulative stationnaire du modèle λ_i . $F_i(x)$ est définie par $F_i(x) = \int_x (\sum_{j=1}^N \Pi_j b_j(x))$, avec Π_j la probabilité stationnaire de l'état j . La probabilité stationnaire d'un modèle de Markov peut être considérée comme le temps passé dans un état donné au bout d'un certain nombre d'itérations. Afin de simplifier les calculs, les auteurs proposent de ne pas intégrer sur l'ensemble des valeurs mais de sommer sur un échantillon de valeurs représentatives. Nous gar-

dons cette méthode pour notre application.

$$D_{\text{HSD}}(\lambda_1, \lambda_2) = \int_x |F_1(x) - F_2(x)| dx \quad (6.2)$$

Pour adapter [HSD](#) aux [Gs-GsHSM](#), deux étapes sont nécessaires. La première est de prendre en compte les probabilités de stationnement, pour calculer les probabilités stationnaires. Pour cela, nous utilisons la méthode décrite par [Barbu et al. \[2012\]](#). Celle-ci consiste à calculer les probabilités stationnaires en utilisant, au lieu du nombre de visites dans chaque état, la durée de séjour à chaque visite dans un état. La probabilité stationnaire pour l'état i se calcule alors comme $\Pi_i = \frac{\sum_{k=1}^m \text{et } s_k=i \cdot X_k}{\sum_{k=1}^m X_k}$ avec m le nombre de changements d'état dans la séquence générée, X_k le temps de séjour entre les changements d'états $k-1$ et k et s_k l'état du système avant le changement d'état k . Afin de calculer la probabilité stationnaire, m augmente jusqu'à la convergence du résultat.

La seconde étape consiste à prendre en compte l'ensemble des observations possibles. Pour cela, la distribution cumulative stationnaire est calculée pour une liste d'observations représentatives des modèles. Cette liste est créée comme décrit dans l'algorithme 7. Elle est composée de tous les arrangements des valeurs binaires, et des N valeurs représentatives pour les valeurs continues. Pour un vecteur avec a valeurs binaires et b valeurs continues, la liste se compose donc de $2^a N^b$ éléments. Si N est trop petit, la distance calculée n'est pas significative des valeurs continues, mais plus N est grand, plus le temps de calcul est important.

Un avantage de la méthode est qu'elle prend en compte l'ensemble des paramètres du modèle. Il serait possible d'améliorer les résultats, en considérant les observations continues.

6.2.4 Distance statistique normalisée

Cette distance est une adaptation de la distance statistique normalisée pour [HMM](#) présentée par [Lu et al. \[2013\]](#). Elle définit la distance entre deux modèles comme la probabilité jointe qui permet aux deux modèles d'être statistiquement équivalents.

Cette distance consiste à calculer les probabilités jointes des événements des deux modèles, définit comme l'ensemble des éléments $\Pi_i a_{ij} \forall i, j$, calculés pour les deux modèles. L'équivalence statique des deux modèles est testé en utilisant le test du χ^2 sur une séquence d'états générée grâce à l'un des modèles. Tant que les deux modèles ne sont pas statistiquement équivalents, les événements dont les probabilités sont infé-

rieures à la probabilité jointe courante sont supprimés du modèle. La matrice de transition doit alors être normalisée afin de vérifier $\sum_j a_{ij} = 1 \forall i$ et les états absorbants sont supprimés. On passe alors à la probabilité jointe suivante. Le résultat est alors la probabilité courante quand les deux modèles sont équivalents.

Pour adapter le modèle au [Gs-GsHSMM](#), plusieurs modifications sont nécessaires. Comme pour [HSD](#), décrite dans la sous-section 6.2.3, il faut calculer les distributions stationnaires des modèles en prenant en compte les probabilités de stationnement. Les événements qui sont supprimés (les probabilités dans la matrice de transition) deviennent nuls. Dans le cas des [Gs-GsHSMM](#), les auto-transitions ne sont pas définies dans la matrice de transition A mais dans les durées de stationnement dans un état. Cela pose globalement peu de problèmes si les durées sont importantes. Néanmoins, il est nécessaire de prendre en compte l'ensemble des événements, y compris les probabilités de stationnement. Ainsi, si l'un de ces événements est supprimé, il faudrait alors supprimer le temps de stationnement pour l'état. Cela implique que l'on considère l'ensemble des durée de stationnement comme un seul événement. Une façon de faire consiste à réduire l'écart-type des probabilités de stationnement, afin d'annuler les probabilités de stationnement les plus éloignées de la moyenne.

Pour calculer le nombre minimum de pas nécessaire pour définir si les modèles sont équivalents, nous multiplions le nombre de pas définis grâce à la matrice de transition par la durée de stationnement maximum. La dernière modification est, pour définir si deux modèles sont équivalents, de générer une séquence d'états qui prenne en compte les probabilités de stationnement.

Le fait de prendre l'ensemble des durée de stationnement comme un seul événement réduit l'importance des probabilités de stationnement dans les états. Afin que celles-ci soient correctement prises en compte, il serait nécessaire de les considérer comme plusieurs événements significatifs, et ne pas procéder à la normalisation des probabilités de transition et de stationnement.

6.3 Analyse de l'évolution du comportement

6.3.1 Modèle évolutif de test

Afin de comparer les différentes méthodes utilisées pour le calcul de la distance entre deux modèles [Gs-GsHSMM](#), nous utilisons des modèles connus avec 4 états que nous pouvons faire évoluer de l'un vers l'autre. Cette évolution se fait de manière linéaire. Le modèle ainsi réalisé est évolutif et peut changer de manière indépendante

6.3. Analyse de l'évolution du comportement

selon les différents paramètres A , Δ et τ .

Les matrices de transition A initiale et finale sont représentées par la figure 6.1. Les modifications dans la matrice de transitions consistent à changer les probabilités de transitions, incluant des probabilités devenant nulles ou au contraire, des probabilités nulles devenant non-nulles. Seule la diagonale, représentant les auto-transitions du modèle, reste constante et égale à zéro, afin de respecter le modèle.

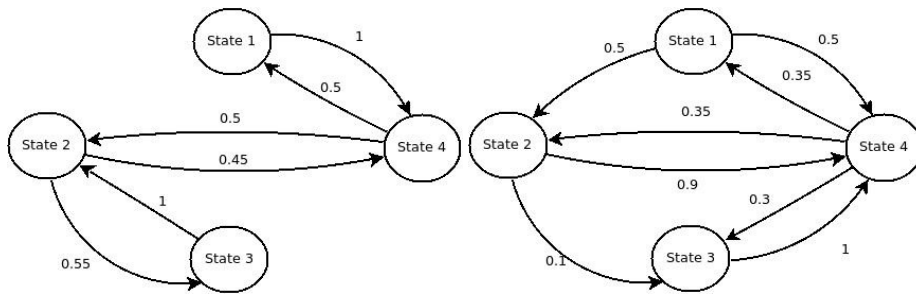


FIGURE 6.1 – Matrices de transition initiale et finale

La figure 6.2 présente l'évolution des probabilités de durées dans un état. Celle-ci concerne la moyenne et la variance des gaussiennes qui modélisent les probabilités de durées. Afin de respecter les contraintes du modèle, le facteur de normalisation doit également être recalculé.

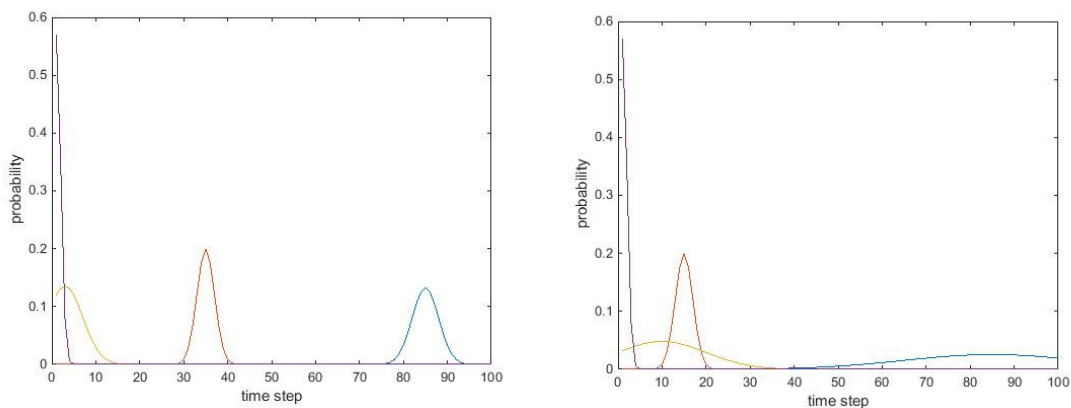


FIGURE 6.2 – Probabilités de durées initiales et finales

Les observations du modèle se composent de 4 valeurs binaires et d'une valeur continue. Il y a donc 5 gaussiennes par état, ce qui fait que τ se compose en tout de 20 gaussiennes. Comme pour les probabilités de durées dans les états, les probabilités d'observations peuvent évoluer en moyenne et en variance.

6.3.2 Comparaisons des distances appliquées au modèle

Nous comparons les méthodes en calculant les distances entre les différentes versions du modèle évolutif.

Distance hybride Kullback-Leibler et Bhattacharyya

Dans un premier temps, la distance hybride est utilisée pour calculer la distance entre les modèles successifs λ_t et λ_{t-1} , chacun des modèles étant proche de son successeur. La figure 6.3 montre les distances calculées pour des modèles où Δ et τ évoluent, la distance gaussienne Bhattacharyya permettant de détecter ces évolutions. Au contraire, l'évolution de la matrice de transition A n'est pas détectée, excepté pour la première et la dernière évolution. Dans ces deux cas, des probabilités nulles deviennent non-nulles, ou au contraire, des probabilités non-nulles deviennent nulles. Ces transitions sont des cas spécifiques et peu représentatives des évolutions habituelles d'un système.

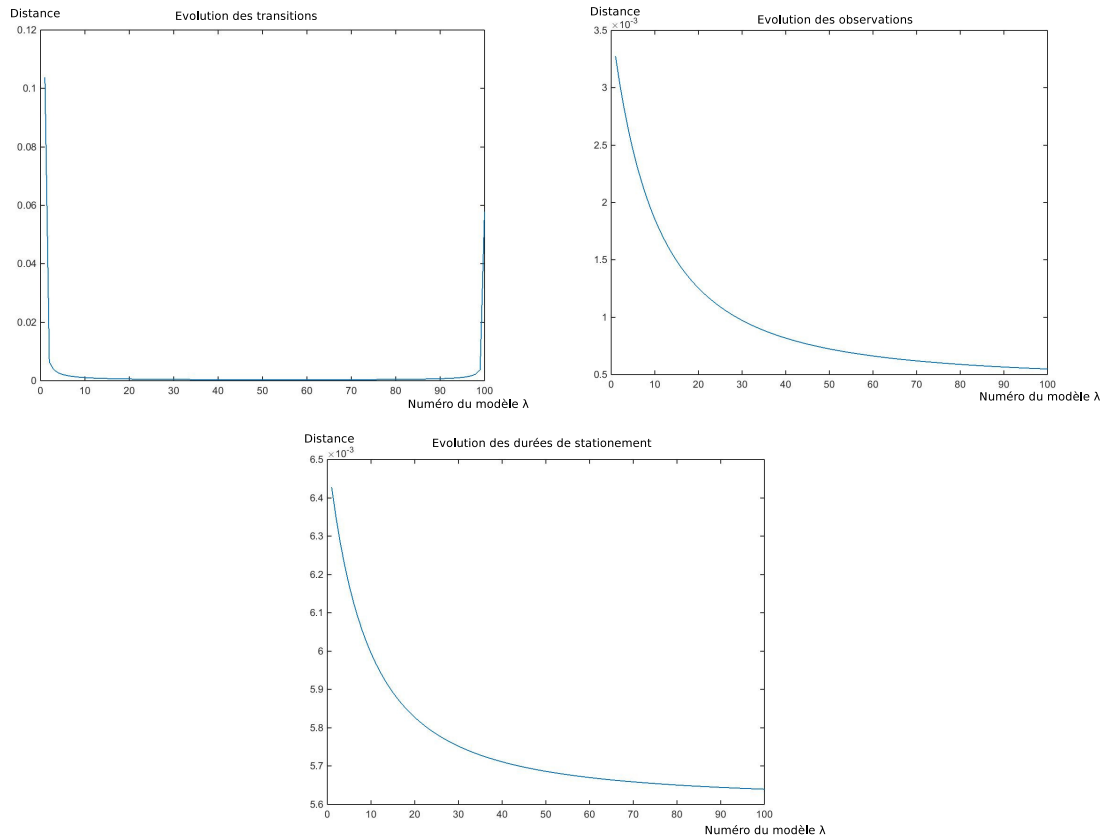


FIGURE 6.3 – Évolution des distances avec la méthode hybride pour les modèles λ_t et λ_{t-1}

Dans un second temps, nous calculons les distances entre le modèle λ_t et le modèle λ_0 . Les différences deviennent alors de plus en plus importantes, et toutes les évo-

lutions peuvent être détectées, comme montré par la figure 6.4.

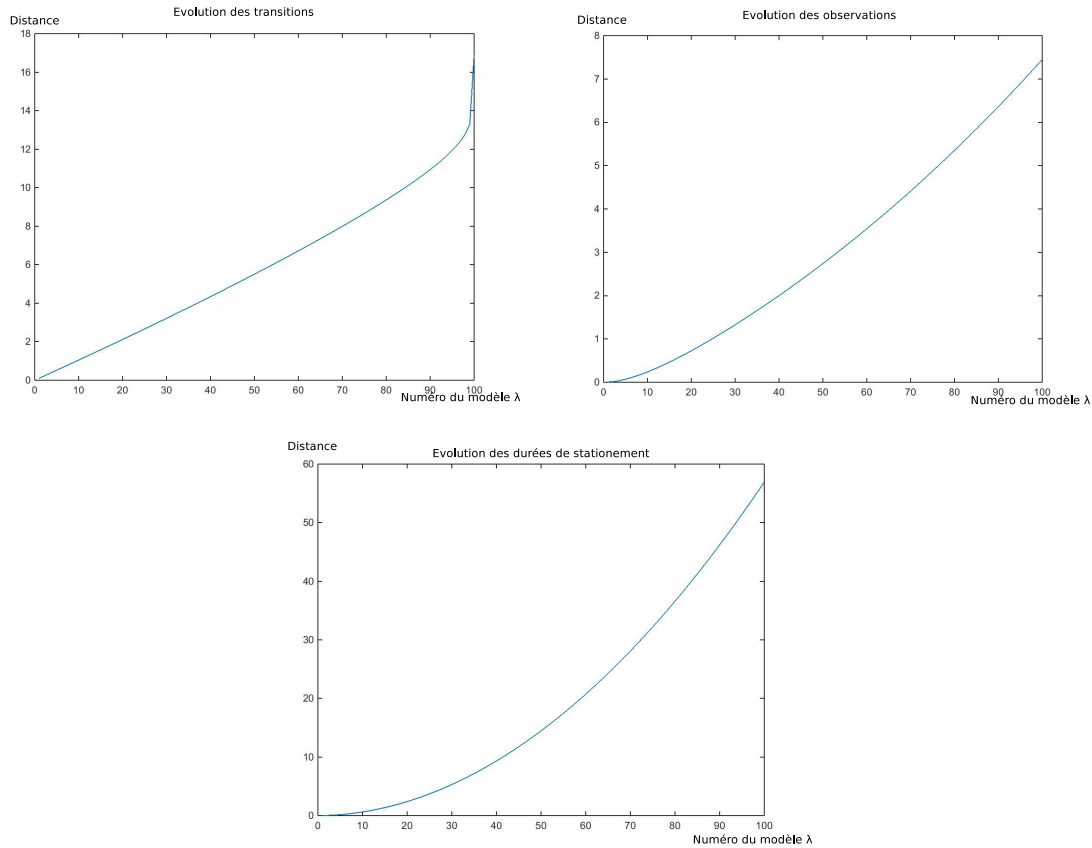


FIGURE 6.4 – Évolution des distances avec la méthode hybride pour les modèles λ_t et λ_0

Distance KL appliquée au Gs-GsHSM

Contrairement à la première méthode, la distance KL est définie de manière probabiliste et donc la distance entre deux modèles λ_t et λ_{t-1} n'est pas déterministe. Le résultat est donc très variable et non analysable. S'il y a aussi un fort bruit entre les distances λ_t et λ_0 , il est possible d'identifier d'importantes variations pour tous les types d'évolution, principalement pour l'évolution du paramètre Δ , qui définit les observations, comme illustré par la figure 6.5.

HSD

Comme pour la distance KL, la distance calculée avec la méthode HSD entre deux modèles λ_t et λ_{t-1} donne des résultats très différents mais seule l'évolution de Δ est détectable. Vu les variations importantes constatées au niveau des autres évolutions.

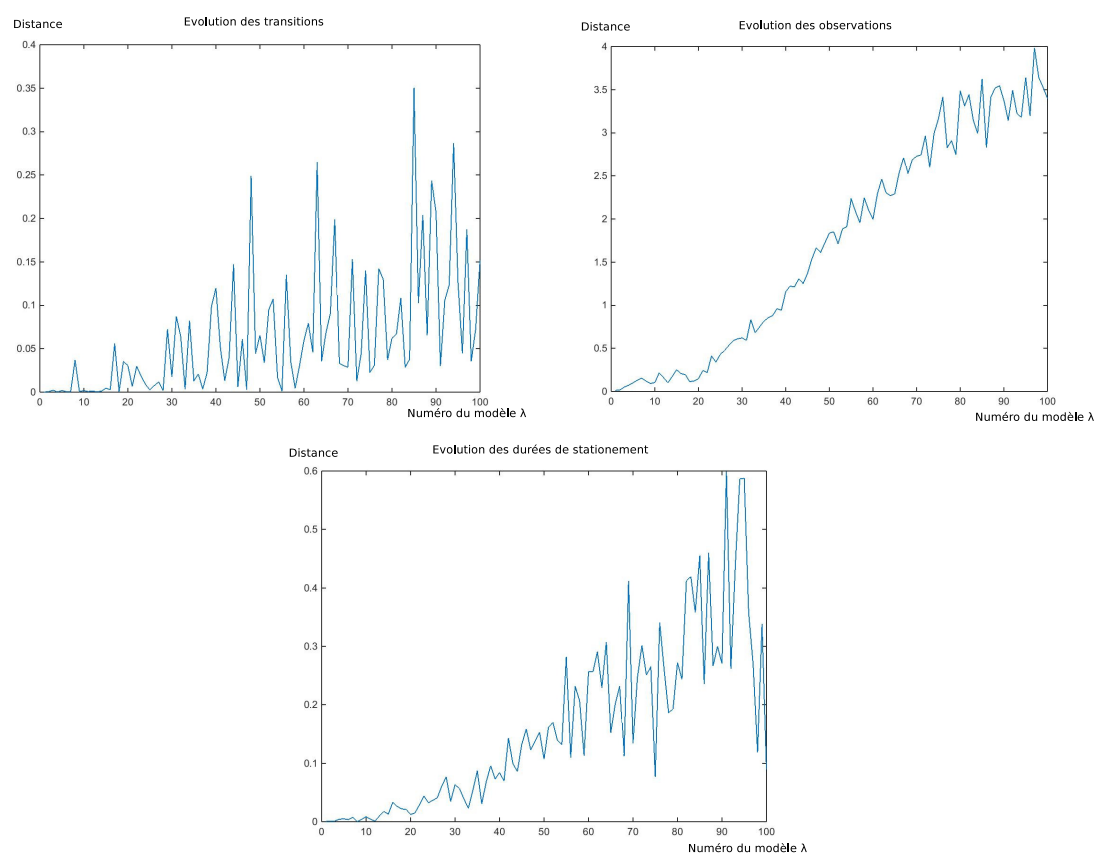


FIGURE 6.5 – Évolution des distances avec la méthode KL pour les modèles λ_t et λ_0

6.3. Analyse de l'évolution du comportement

L'étude de l'évolution des distances entre les modèles λ_t et λ_0 permet de définir plus précisément les évolutions du modèle. Si la distance dans le cas de l'évolution des paramètres de la durée de stationnement τ est à peu près constante, ce n'est pas le cas des autres types d'évolutions, comme montré par la figure 6.7. Dans le cas de l'évolution des paramètres d'observations Δ , la distance augmente globalement. La distance pour l'évolution de la matrice de transition A présente un maximum au milieu de l'évolution ensuite la distance est diminuée.

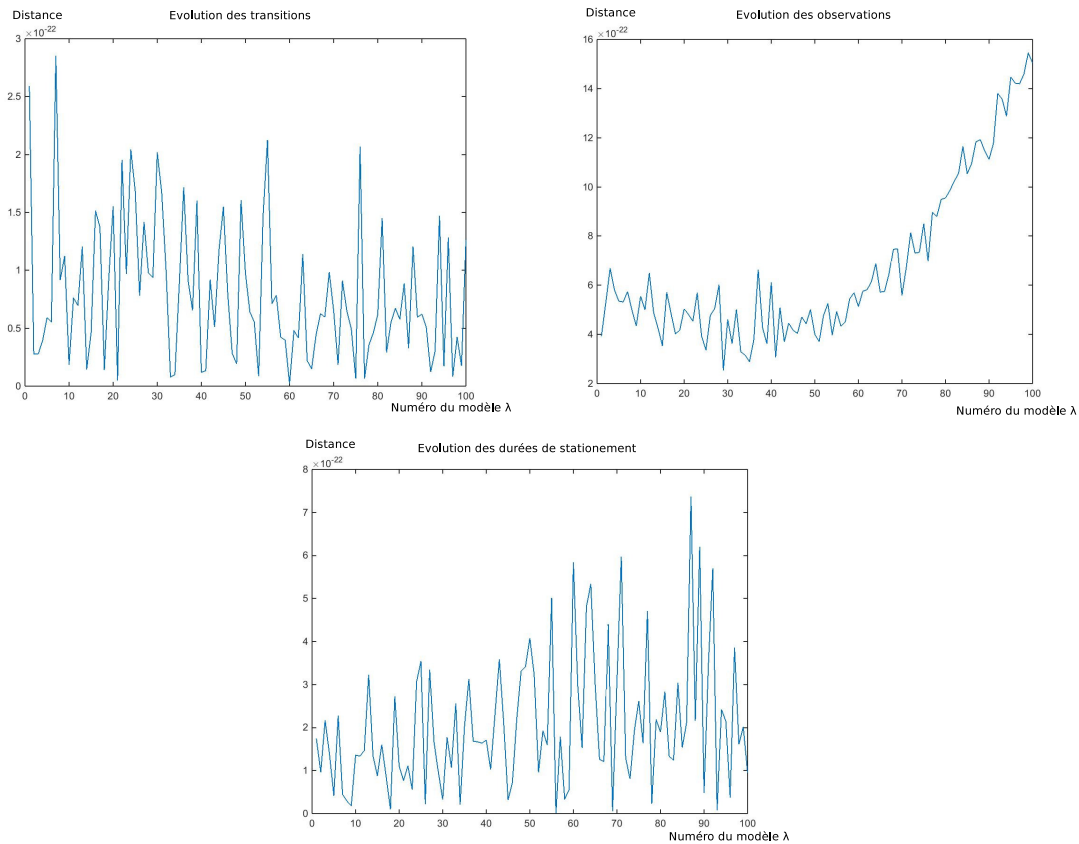


FIGURE 6.6 – Évolution des distances avec la méthode HSD pour les modèles λ_t et λ_{t-1}

Distance statistique

Bien que la méthode de calcul des distances statistiques soit la plus intéressante théoriquement, le temps de calcul reste importante. De ce fait, nous ne testons cette distance que pour 10 pas de temps, contrairement aux autres méthodes qui sont testées pour 100 pas de temps.

La comparaison entre les modèles λ_0 et λ_t fournit des résultats très bruités et ne permet pas de détecter les évolutions du modèle. En effet, aucune variation n'apparaît, quelles que soient les évolutions. S'il est possible que des variations très importantes

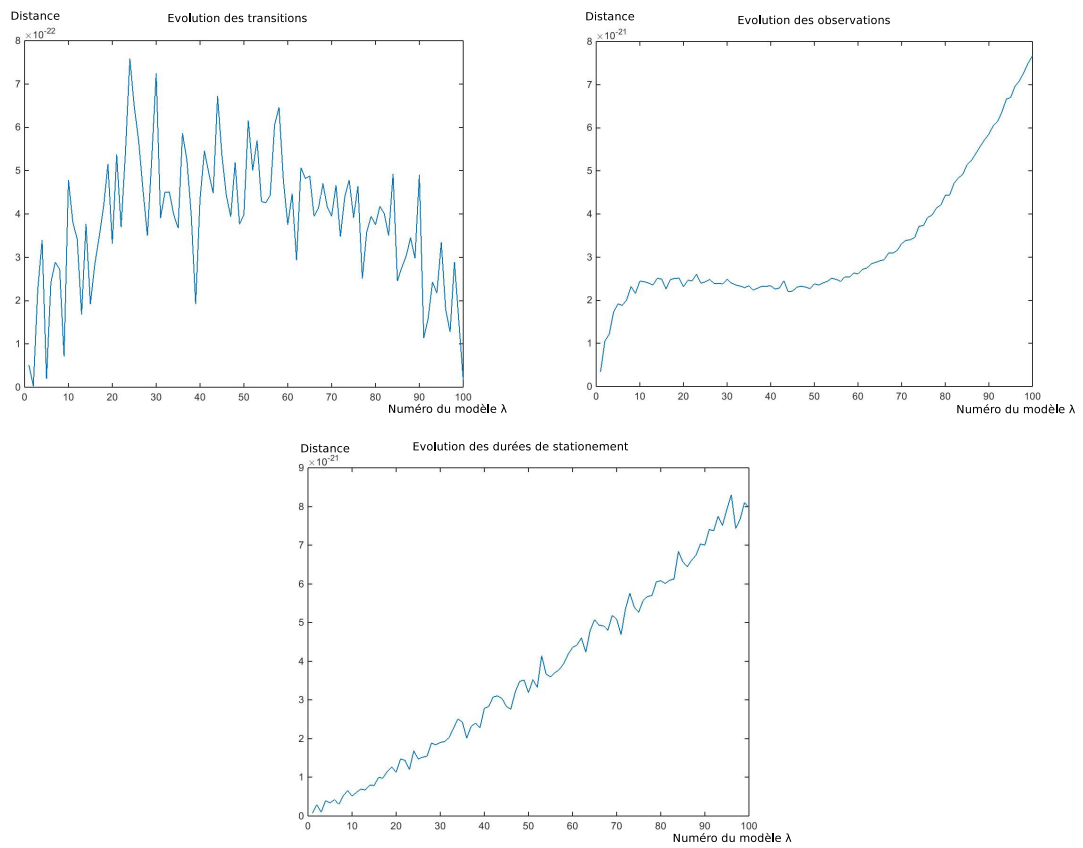


FIGURE 6.7 – Évolution des distances avec la méthode HSD pour les modèles λ_t et λ_0

puissent être détectables, ce n'est pas le cas de variations moins importantes.

Compte tenu des résultats de la méthode par rapport aux autres, mais également des contraintes liées au temps de calcul nécessaire pour la méthode, celle-ci ne paraît pas très intéressante pour résoudre notre problématique, et nous ne la testons pas plus.

6.3.3 Influence de l'apprentissage sur le calcul des distances

Nous souhaitons évaluer l'influence de l'algorithme d'apprentissage Baum-Welch sur les méthodes de calcul des distance, afin d'identifier le bruit dû à l'algorithme. En effet, si l'ensemble de ces méthodes permettent de calculer la distance entre deux modèles, nous souhaitons calculer la distance entre des modèles appris, ce qui peut rajouter des variabilités et des incertitudes par rapport à des modèles acquis directement et créés manuellement. Pour cela, générons 100 séquences d'observations ($O^1, O^2, \dots, O^{100}$) à partir d'un même modèle λ , que nous utilisons par la suite pour apprendre les modèles ($\lambda_1, \lambda_2, \dots, \lambda_{100}$), tel que représenté par la figure 6.8. Puis nous calculons la distance entre ces modèles appris et le modèle d'origine ($D(\lambda, \lambda_i)$) et entre le modèle et le modèle appris précédemment ($D(\lambda_i, \lambda_{i-1})$). Pour chacun des apprentissages, nous testons 50 modèles.

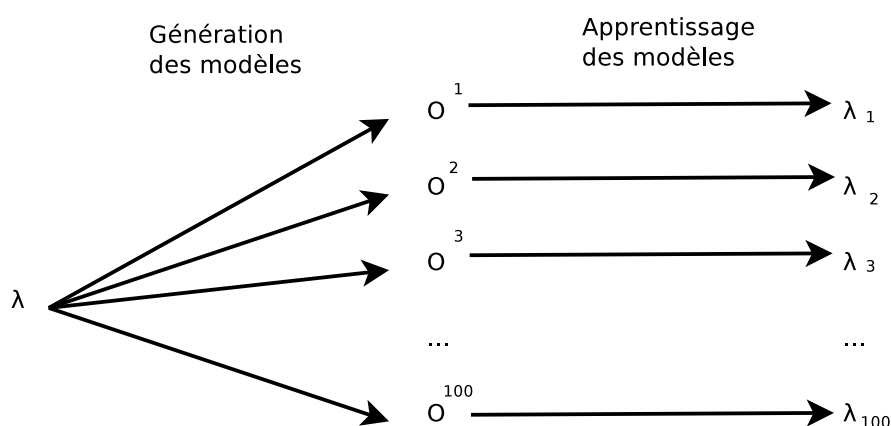


FIGURE 6.8 – Génération des modèles appris expérimentaux

Dans le cas de la distance hybride, la distance moyenne entre le modèle appris et le modèle d'origine est de 392. Cette valeur est très largement supérieure à la distance maximum calculée entre les modèles. Néanmoins, la moyenne des distances entre les modèles appris successivement est de 217. Cela semble indiquer que le modèle initial n'est pas le modèle le plus significatif pour les séquences générées. Par exemple, cela pourrait être dû à des états dont les observations seraient proches. Ce résultat reste à valider par les autres méthodes de distances. La moyenne des distances entre les mo-

dèles appris reste néanmoins très supérieure par rapport au modèle connu. Les variations dues à l'apprentissage seraient alors trop importantes pour permettre la détection de l'évolution du modèle.

L'étude des résultats avec la méthode [KL](#) appliquée au [Gs-GsHMM](#) confirme la différence entre le modèle d'origine et les modèles appris, avec une moyenne de 7.29 dans le cas de la distance avec le modèle d'origine, et de 0.34 dans le cas de la distance avec le modèle appris précédent. Cette distance utilise une séquence d'observations générée de manière probabiliste et semble moins influencée par l'apprentissage. En effet, la moyenne dans le cas de la distance avec le modèle précédent est inférieure à la distance maximum observée dans le cas de l'évolution des paramètres d'observations. Celle-ci devrait donc pouvoir être détectée.

Les résultats avec [HSD](#) confirment que les distances calculées avec le modèle initial sont supérieures à celles calculées avec le modèle précédent. En effet, la moyenne dans le premier est de l'ordre de 10^{-66} alors que dans le second cas, elle est de l'ordre de 10^{-76} . Il est à noter une itération spécifique, dont la distance est très supérieure aux autres. Les distances sont néanmoins très inférieures à celles calculées dans le cadre de l'évolution de modèles qui est de l'ordre de 10^{-22} . Il semblerait donc que le [HSD](#) soit peu sensible à l'apprentissage. Cela doit donc être validé dans le cadre de modèle appris avec une évolution du modèle.

6.3.4 Calcul des distances sur modèle évolutif appris

Nous souhaitons valider les résultats acquis précédemment en calculant les distances avec des modèles appris avec l'algorithme de Baum-Welch à partir de données issues du modèle évolutif, vu dans la partie [6.3.1](#). Compte tenu du temps de calcul nécessaire pour l'apprentissage, nous faisons évoluer le modèle 10 fois ($\lambda_{evo10}, \lambda_{evo20}, \dots, \lambda_{evo100}$) afin de calculer les distances. Les modèles évolutifs sont utilisés pour générer des séquences d'observations ($O^{10}, O^{20}, \dots, O^{100}$). Ces séquences sont ensuite apprises afin de générer les modèles appris ($\lambda_{app10}, \lambda_{app20}, \dots, \lambda_{app100}$), comme présenté dans la figure [6.9](#). Nous calculons ensuite les distances entre le modèle d'origine et les modèles λ_{app} .

Les résultats de la méthode avec la distance hybride ne permettent pas de détecter une évolution du modèle. En effet, aucune variation globale n'apparaît et les résultats semblent très fortement bruités. Cela confirme la forte sensibilité de la méthode à l'apprentissage, entraînant des difficultés d'utilisation de la méthode dans le cas d'utilisation de modèles acquis par apprentissage.

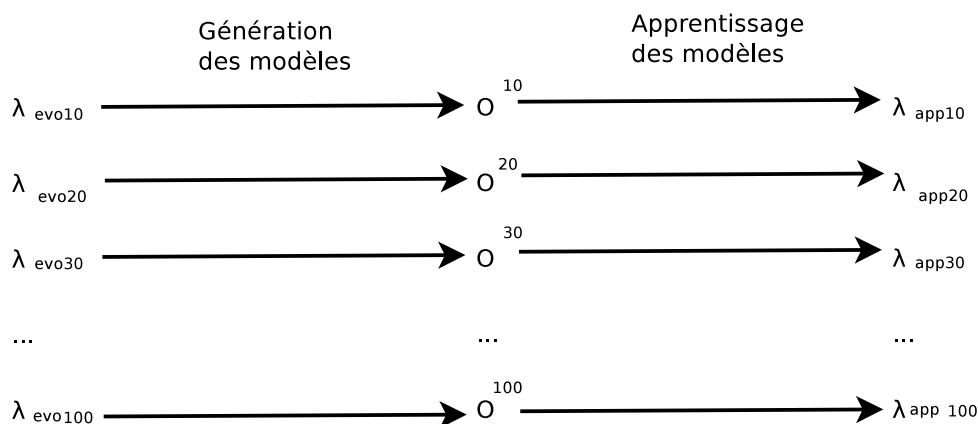


FIGURE 6.9 – Génération des modèles évolutifs appris

La méthode de calcul de distance [KL](#) permet globalement une meilleure détection de l'évolution du modèle, comme vu sur la figure [6.10](#). En effet, les 7 premières itérations permettent de détecter une augmentation globale de la distance avec le modèle initial. On observe, pour l'itération 8, une distance beaucoup plus faible, avant qu'elle n'augmente de nouveau. Il est possible que, du fait des évolutions du modèle évolutif, les modèles sont proches, à un moment donné du modèle d'origine. Cela doit néanmoins être validé. Cela semble confirmer que cette méthode est globalement moins sensible à l'apprentissage que la méthode hybride et permettrait de détecter des évolutions importantes du modèle.

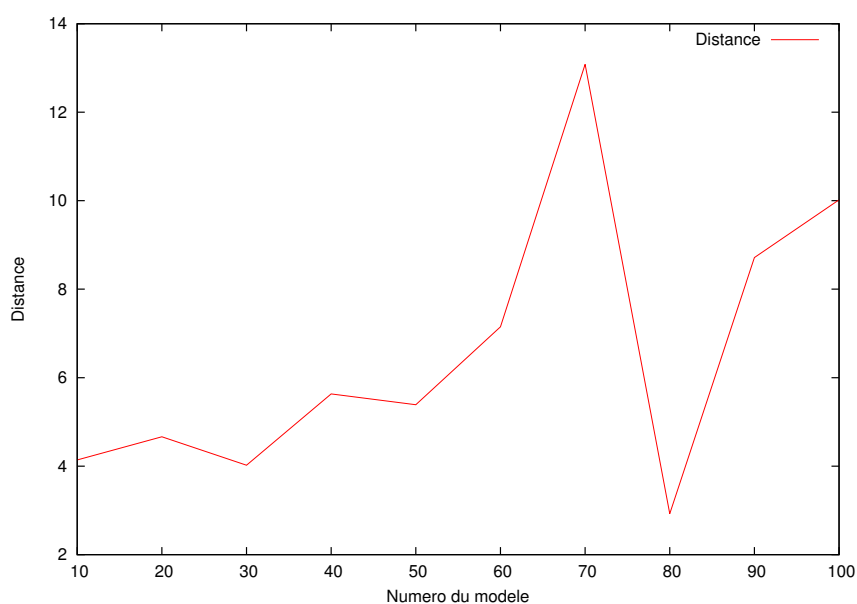


FIGURE 6.10 – Évolution des distances avec la méthode KL pour les modèles appris

Si les résultats précédents semblent indiquer que la méthode [HSD](#) est moins sensible à l'apprentissage, cela ne semble pas être validé par les derniers résultats. En effet, la distance calculée ne semble pas évoluer de façon cohérente avec l'évolution du modèle.

6.3.5 Discussion sur les méthodes

La comparaison des méthodes de calcul de distances avec le modèle évolutif permet de mettre en avant, sur un cas spécifique, les qualités et les défauts des méthodes, notamment en prenant en compte l'influence de l'algorithme d'apprentissage. Cela permet de mettre en avant la méthode [KL](#) appliquée au [Gs-GsHSM](#), qui semble être la plus intéressante. Bien que les évolutions peu importantes ne soient pas détectées, la méthode est peu sensible à l'apprentissage, et permet de détecter les évolutions selon l'ensemble des paramètres, en un temps de calcul acceptable.

Les tests ont été effectués sur un modèle que nous avons généré. Si cette première approche permet d'étudier différentes possibilités pour détecter les évolutions d'un système, la méthode doit être validée dans le cadre de notre application et notamment prendre en compte les caractéristiques du comportement humain. Les méthodes doivent donc être testées sur des données acquises en situations réelles, afin de définir ce que sont ces caractéristiques. Ainsi, il faut définir des évolutions significatives pour les aidants, détectables et éviter toute alerte inutile. Cela revient à définir, par exemple, la durée minimum nécessaire entre l'acquisition des données utilisées pour l'apprentissage des deux modèles pour que les différences soient significatives et utiles. Ces éléments peuvent être définis grâce à des exemples acquis en situation réelle et à l'expertise des aidants, et grâce à certaines variables spécifiques, telle que les pathologies du résident ou son niveau de dépendance.

Un autre élément à prendre en compte est la vraisemblance du modèle. En effet, une amélioration proposée pour augmenter celle-ci serait de définir de manière dynamique le nombre d'états et la durée de stationnement maximum. Autrement, la méthode utilisée pour déterminer la distance devrait pouvoir calculer la distance entre deux modèles dont les structures ne sont pas identiques. Cette contrainte supplémentaire exclurait des méthodes comme la méthode hybride [KL](#) et Bhattacharyya, qui calcule la distance élément par élément. Contrairement aux autres méthodes, dont l'unique contrainte est d'exiger le même nombre d'observations à chaque pas de temps.

6.4 Distance KL convergente

L'étude des distances présentes dans la littérature permet de mettre en avant les avantages et les contraintes des différentes méthodes. À partir des différents résultats, nous pouvons discuter les contraintes imposées pour le calcul des distance entre deux [HMM](#) et présenter une nouvelle méthode, basée sur la distance [KL](#), que nous appliquons au [Gs-GsHMM](#)

6.4.1 Contraintes

Les méthodes de calcul de distances doivent respecter quatre propriétés, qui permettent de définir un système métrique : définies positivement ($D(\lambda_1, \lambda_2) \geq 0$), commutative ($D(\lambda_1, \lambda_2) = D(\lambda_2, \lambda_1)$), réflexive ($D(\lambda_1, \lambda_1) = 0$) et respectant l'inégalité triangulaire ($D(\lambda_1, \lambda_2) + D(\lambda_2, \lambda_3) \geq D(\lambda_1, \lambda_3)$). Dans le cas des distances entre des modèles de Markov, on observe que ces propriétés ne sont pas toujours respectées, pour des raisons volontaires. [Falkhausen et al. \[1995\]](#) expliquent qu'une distance non-symétrique permet de caractériser le fait qu'un modèle puisse expliquer un autre modèle sans que la réciproque ne soit vraie.

Une autre propriété qui peut être considérée, compte tenu du caractère stochastique des modèles, est le déterminisme de la distance. En effet, il est possible d'avoir certaines distances, comme la distance [KL](#), dont le résultat est stochastique, et donc qui peut changer à chaque évaluation. Cela implique des variations importantes, rendant l'étude de l'évolution des distances impossibles, comme expliqué précédemment. Il faut donc que les résultats acquis, en calculant plusieurs fois la distance entre deux même modèles, soient le plus proche possible pour être significatifs et utilisables.

Pour que la méthode proposée soit utilisable en situation réelle, il faut que la complexité soit suffisamment faible pour fournir un résultat en un temps acceptable.

De plus, pour la distance [KL](#), il faut avoir le même nombre d'observations pour que les modèles puissent être comparables permettant ainsi de calculer des distances entre deux modèles dont la typologie, c'est à dire le nombre d'états cachés et la durée de stationnement maximum, est différente. Cela est par exemple le cas avec la méthode de distance statistique normalisé.

6.4.2 Présentation et implémentation

Nous proposons une nouvelle méthode, basée sur la distance KL, présentée dans la sous-section 6.3.2 et qui présente une limite au niveau du résultat qui est fortement stochastique. Afin d'assurer que le calcul converge, il faut augmenter le nombre de pas de temps utilisé pour calculer la vraisemblance, ce qui a pour effet d'augmenter le temps de calcul. En effet, la distance KL dépend de $P(O|\lambda_1)$ et de $P(O|\lambda_2)$, dont le résultat est variable en fonction de T , la longueur de la séquence d'observations O .

L'objectif est donc choisir la taille de la séquence d'observation la plus faible possible, permettant la convergence du résultat. Pour cela, une méthode consiste à augmenter la longueur de la séquence d'observations jusqu'à ce que le résultat converge, quand l'inéquation suivante est vérifiée $|D_{KL1}(\lambda_1, \lambda_2) - D_{KL2}(\lambda_1, \lambda_2)| < \epsilon$ avec ϵ un réel suffisamment petit et D_{KL1} et D_{KL2} sont calculées avec deux séquences d'observations différentes. Cette condition d'arrêt peut se simplifier en appliquant la formule 6.3, avec O_1 et O_2 deux séquences d'observation générées aléatoirement par le modèle λ_2 et t le nombre de pas de temps correspondent aux séquences O_i .

$$\frac{|\log(P(O_1|\lambda_1)) - \log(P(O_1|\lambda_2)) - \log(P(O_2|\lambda_1)) + \log(P(O_2|\lambda_2))|}{t} < \epsilon \quad (6.3)$$

Cette condition permet, à partir du moment que les log-vraisemblances tendent vers une même valeur, que la distance converge en un nombre de pas de temps minimum. Ce qui est traduit par la formule 6.4 et qui n'est pas garanti par tous les modèles de Markov. La convergence a néanmoins été prouvée sous différentes hypothèses, par exemple par Leroux [1992]. Une des hypothèses est que la propriété d'ergodicité soit respectée. Dans ce cas, il est possible d'atteindre, à partir de tous les états, en un ou plusieurs pas de temps, l'ensemble des états du système. Étant donné que la distance entre les modèles de Markov a pour objectif d'être utilisée suite à un apprentissage, nous ne pouvons pas assurer que ces propriétés soient automatiquement satisfaites.

$$\sum_{t \rightarrow \infty} \log(P(O_1|\lambda_i)) = \sum_{t \rightarrow \infty} \log(P(O_2|\lambda_i)) \quad \forall i \text{ et } O_1 \neq O_2 \quad (6.4)$$

La distance est alors définie selon la formule 6.5 et peut être considérée comme la moyenne des distances D_{KL1} et D_{KL2} . Si la condition 6.3 n'est pas validée au bout d'un certain nombre de pas de temps, par exemple si les log-vraisemblances ne convergent pas vers la même valeur, cela permet de moyenner la distance calculée.

$$D(\lambda_1, \lambda_2) = \frac{|\log(P(O_1|\lambda_1)) - \log(P(O_1|\lambda_2)) + \log(P(O_2|\lambda_1)) - \log(P(O_2|\lambda_2))|}{2t} \quad (6.5)$$

La méthode se base sur le calcul de la log-vraisemblance de deux modèles, ce qui revient au premier problème des [HMM](#), présenté dans la sous-section [3.4.1](#). Nous appliquons ici sur les [Gs-GsHSM](#). Il serait néanmoins possible de calculer pour des modèles [HMM](#) ou d'autres variantes, tant qu'il est possible de calculer la log-vraisemblance sur des séquences d'observations générées aléatoirement.

Nous implémentons la méthode en nous basant sur l'algorithme de calcul de vraisemblance présenté dans la sous-section [4.3.2](#). Il s'agit en effet de calculer les vraisemblance de deux modèles différents et sur deux séquences d'observations distinctes. La longueur de ces séquences d'observation est inconnue, étant donné que nous souhaitons la réduire. De plus, il n'est pas nécessaire de connaître l'ensemble des séquences, mais uniquement l'observation courante, et les éléments permettant de générer les observations futures, c'est à dire l'état courant et le nombre de pas de temps avant de changer d'état. Il est donc possible de calculer simultanément l'ensemble des log-vraisemblances, en respectant les contraintes typologique des modèles. On obtient alors le pseudo-code présenté dans l'algorithme [4](#). Sa complexité algorithmique est de $O(T(N_1(N_1 + D_1) + N_2(N_2 + D_2 + 2 \cdot D_2 \cdot k)))$, avec N_i le nombre d'état, D_i la durée de stationnement maximal, i le numéro du modèle, T le nombre de pas de temps avant la convergence et k le nombre de gaussiennes observées. Il est possible de réduire la complexité de l'algorithme si on considère que les deux modèles ont la même typologie. Ce qui ramène la complexité à $O(T(N(N + D)))$.

En considérant le cas où les modèles ont des typologies différentes, en considérant que les modèles sont composés de 20 états, ont une durée maximum de 480 pas de temps et que les observations se composent de 6 valeurs, il faut 60 secondes pour effectuer 10000 pas de temps. En comparaison, l'apprentissage d'un modèle avec la même typologie prend, en moyenne, 13.75 heures. Ces tests ont été effectués sur une machine avec un processeur dual core de 2.5GHz, 4Go de mémoire RAM, sur un environnement Windows 7. Le nombre de pas de temps nécessaire pour calculer les distances, dans le cas de différentes tests, sont présentés dans la partie [6.4.3](#).

L'algorithme doit permettre de calculer la distance en utilisant la plus petite séquence d'observation possible. Il faut néanmoins que cette distance soit suffisante pour que le résultat ne soit pas un cas particulier, où la condition [6.3](#) est vérifiée alors que le résultat n'est pas celui attendu et qui serait calculé avec une séquence d'observations plus longue. En effet, lors des premiers pas de temps du calcul de la vraisem-

blance, on observe que celle-ci peut évoluer très rapidement, donnant des résultats très différents avant de commencer à converger vers une valeur. Nous ne testons donc pas la condition d'arrêt lors des n premiers pas de temps, afin que les vraisemblances se stabilisent et que les résultats puissent être significatifs. n dépend des modèles, par exemple de leurs durées maximales de stationnement.

Parmi les propriétés des distances métriques, on remarque que la propriété de symétrie n'est pas respectée. En fonction des besoins, cela peut être un avantage, comme expliqué par Falkhausen *et al.* [1995], ou un inconvénient. Dans ce cas, une solution pour symétriser le résultat est de faire la moyenne des distances $D(\lambda_1, \lambda_2)$ et $D(\lambda_2, \lambda_1)$, comme pour la distance KL (Rabiner [1989]). Calculer séparément les deux distances a néanmoins pour effet de doubler le coût en temps de calcul. Une solution envisageable est de calculer l'ensemble des huit log-vraisemblances simultanément, en générant quatre séquences d'observations distinctes, tant que les deux distances n'ont pas convergées.

6.4.3 Résultats

Comme pour les autres distances, nous testons la méthode sur différents tests, en utilisant le modèle évolutif décrit dans la sous-section 6.3.1.

Comparaison des distances appliquées au modèle

Dans un premier temps, nous testons la méthode sur différentes évolutions du modèle, afin de tester que le résultat converge bien et de façon déterministe et aussi pour pouvoir évaluer le bruit sur un cas simple. Nous générons 1000 pas de temps avant de tester la condition d'arrêt, et le nombre de pas de temps maximum est de 1 000 000.

La figure 6.11 montre l'évolution des distances entre le modèle d'origine et les différentes variantes du modèle évolutif. En faisant varier uniquement la matrice de transition A , on observe que le bruit est non nul et qu'il augmente proportionnellement avec la distance entre les modèles. Les distances sont néanmoins croissantes et l'évolution du modèle est visible.

L'évolution des paramètres d'observations Δ est également visible. Comme dans le cas de l'évolution des paramètres de A , le bruit est fortement dépendant de la distance entre les modèles d'observations. De plus, comme pour HSD, on observe que l'évolution n'est pas linéaire, contrairement à A . Ces observations sont confirmées par l'évolution des paramètres de stationnement dans les états τ . Celle-ci est détectable,

6.4. Distance KL convergente

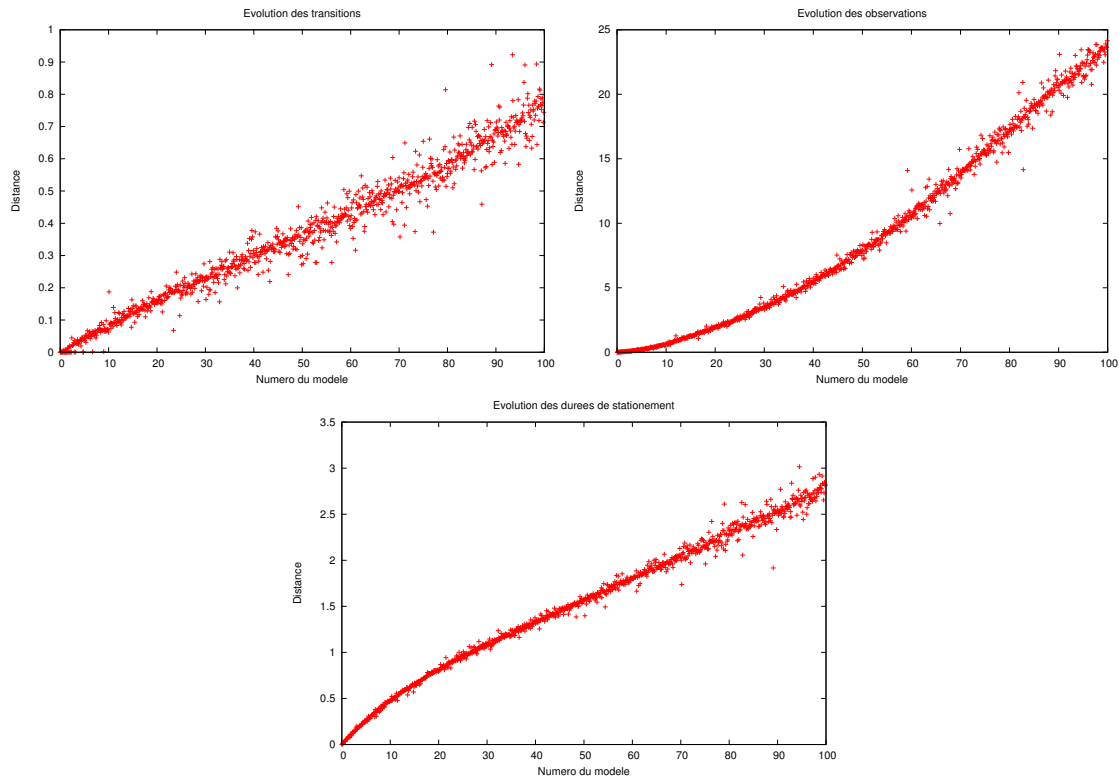


FIGURE 6.11 – Distance calculée entre le modèle d’origine et les variantes du modèle évolutif

bien que légèrement bruitée, surtout pour les distances importantes. De plus, l’évolution des paramètres d’observations Δ est non linéaire.

Si l’ensemble des évolutions est détectable, tous les changements ne sont pas détectables de la même façon. Dans le cas de l’expérience menée sur le modèle évolutif, on observe ainsi que dans le cas des différences les plus importantes, la distance dans le cas de la matrice de transition A est de 1, alors que celle dans le cas des paramètres d’observation est de 24.

Afin de vérifier la convergence, nous étudions, pour l’ensemble de ces tests, le nombre de pas de temps avant l’arrêt. La figure 6.12 présente les résultats. Les moyennes du nombre de pas avant la convergence sont de 190743.012 pour l’évolution des transitions, de 142466.57 pour l’évolution des observations, et de 177082.042 pour l’évolution des durées de stationnement. Pour l’ensemble de ces cas, la moyenne est donc très inférieure à la limite maximum. On observe néanmoins que cette limite est atteinte pour un grand nombre de cas. Néanmoins, compte tenu des distances calculées, les cas limites semblent liés à la précision des résultats et à la valeur choisie pour ϵ dans la condition d’arrêt, et non pas à une impossibilité de converger.

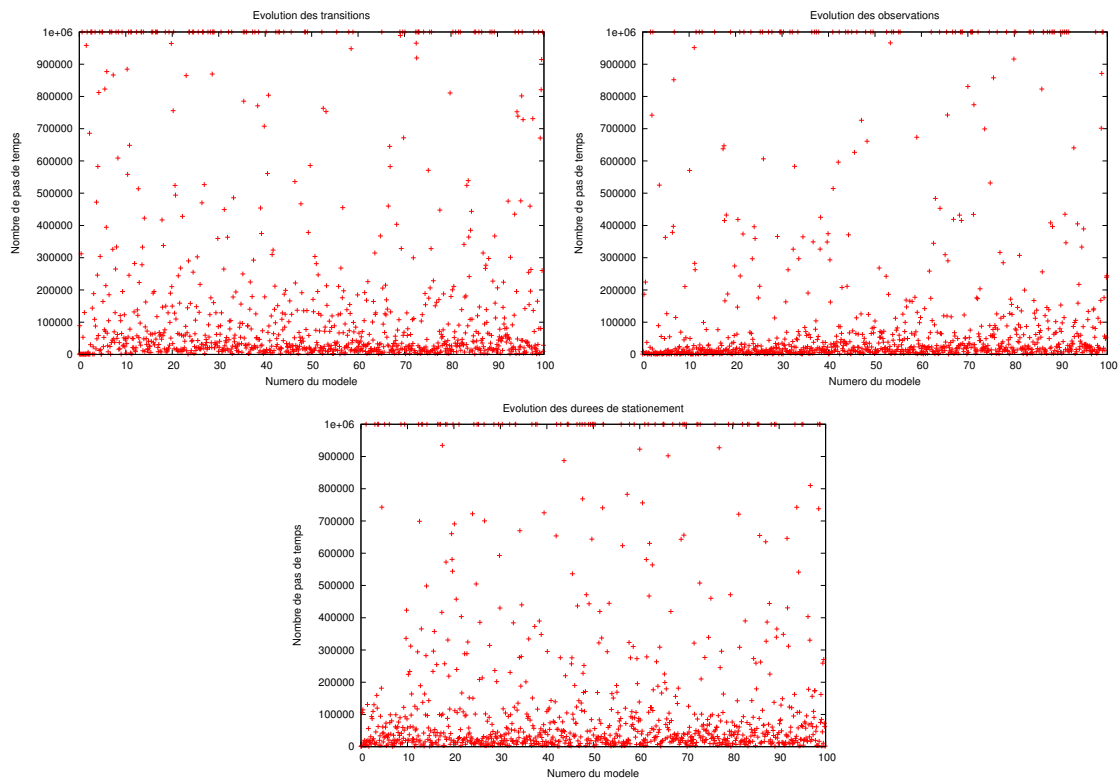


FIGURE 6.12 – Nombre de pas de temps avant convergence du calcul entre le modèle d’origine et les variantes du modèle évolutif

Si les évolutions entre les modèles directs semblent détectables, il est nécessaire de tester l’effet de l’algorithme d’apprentissage sur la méthode.

Influence de l’apprentissage sur le calcul des distances

Nous testons l’influence de l’apprentissage sur la méthode de calcul de distance en testant la distance entre le modèle d’origine et cent modèles acquis par apprentissage en utilisant des séquences d’observations issues du premier modèle. La moyenne des distances obtenues est de 9.38 et la variance est de 5.27.

Les variations issues de l’apprentissage sont supérieures aux différences calculées dans le cas des évolutions de la matrice de transition A . Il est néanmoins inférieur aux différences dans le cas de l’évolution des paramètres d’observations Δ . Les modifications de la matrice de transition ne serait donc pas détectable, alors que des modifications suffisantes des autres paramètres le serait. Des évolutions importantes du modèles seraient donc détectables. Il est néanmoins nécessaire de caractériser le bruit dû

à l'apprentissage afin de caractériser à partir de quelle valeur la distance est significative d'une évolution.

Ces résultats doivent être confirmés en testant la méthode sur des modèle appris grâce aux données du modèle évolutif.

Calcul des distances sur modèle évolutif appris

Comme dans la partie 6.3.4, nous apprenons des modèles ($\lambda_{app1}, \lambda_{app2}, \dots, \lambda_{app100}$) à partir de séquences d'observations générées grâce à différentes versions du modèle évolutif. Nous utilisons ces modèles afin de calculer les distances avec le modèle d'origine ($D(\lambda_0, \lambda_{appi})$). Nous considérons un nombre de pas de temps minimum de 1000, et maximum de 10 000 000. Les résultats sont présentés par la figure 6.13. Ces résultats confirment la possibilité de détecter des évolutions du modèle si elles sont suffisantes, malgré un bruit non nul. Dans le cas des modèles proches du modèle initial, on remarque que la propriété de réflexivité n'est pas respectée. Cela confirme qu'en dessous d'une certaine valeur, une distance n'est pas significative. De plus, les tests sur les modèles évolutifs appris confirme l'augmentation du bruit quand la distance augmente.

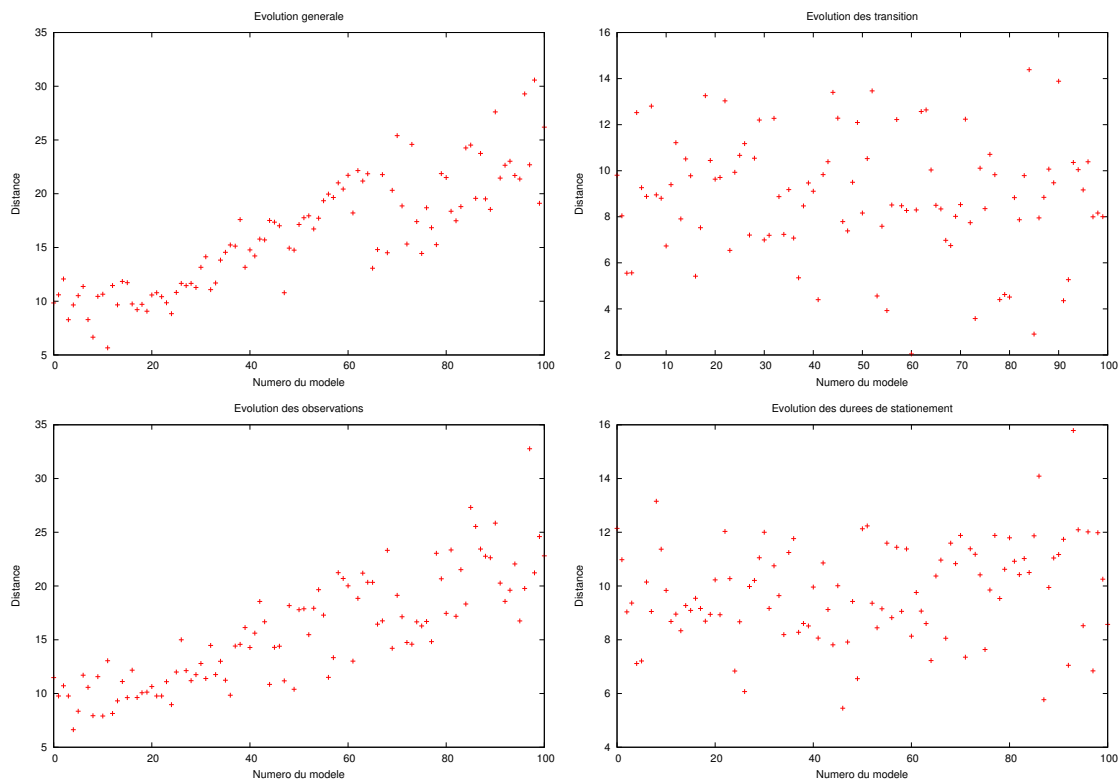


FIGURE 6.13 – Évolution des distances pour les modèles appris

La figure 6.14 présente le nombre de pas de temps pour calculer les distances des modèles appris à partir des séquences générées en faisant varier l'ensemble des paramètres. La moyenne du nombre de pas de temps est alors de 460 901. La valeur maximum a été atteinte pour l'un des calculs, sur les 100 effectués. À l'inverse, le nombre de pas de temps minimum pour trouver une distance est de 1181. Si la méthode ne permet donc pas de réduire de manière systématique le nombre de pas de temps nécessaire, elle permet, dans le cas de modèle appris, de réduire les variations des distances en un nombre de pas de temps moyen acceptable.

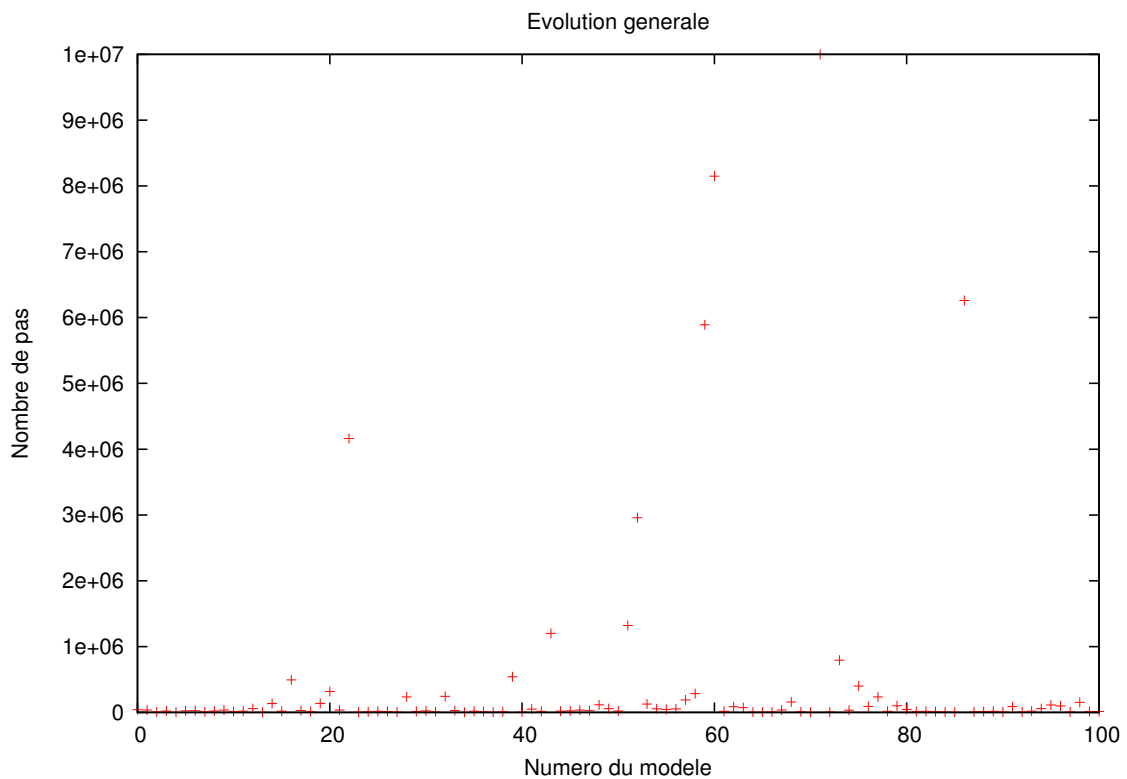


FIGURE 6.14 – Nombre de pas de temps pour calculer les distances des modèles appris

Pour confirmer les résultats précédents, nous testons la méthode en calculant la distance entre le modèle d'origine et des modèles appris en utilisant des données générées grâce au modèle évolutif, en faisant varier un seul paramètre à la fois. Si l'augmentation de la distance est bien détectable en faisant varier les paramètres d'observations Δ . À l'inverse, les variations de A et τ ne sont pas détectables. En effet, si on considère que l'apprentissage implique, dans le cas de notre modèle, un bruit moyen de 9.38 et une variance de 5.27, la distance directe entre les modèles est très inférieure pour les variations de A et de τ . L'évolution n'est donc pas détectable dans ce cas.

Discussions

La méthode proposée permet de calculer la distance entre deux modèles [Gs-GsHSM](#) en réduisant le temps de calcul. De plus, la méthode est globalement robuste à l'apprentissage par rapport aux distances testées précédemment. Différents points doivent néanmoins être discutés.

Comme pour l'ensemble des méthodes de calcul de distances, les résultats ont été acquis en utilisant un modèle évolutif spécifique. Certaines conclusions peuvent donc être liées à celui-ci. Par exemple, les méthodes ont des sensibilités différentes aux variations des paramètres A , τ et Δ , avec des valeurs maximum de l'ordre de 35 pour τ , et de l'ordre de 16 pour les autres durées. Or ces paramètres n'ont pas une évolution identique et d'autres modèles, avec des changements différents, pourraient fournir d'autres résultats.

Cela implique que si l'ensemble des méthodes fonctionnent théoriquement, le choix définitif devra se faire en prenant en compte les spécificités des situations réelles. Cela signifie qu'il faudra tester les méthodes sur des situations réelles, mais également caractériser, avec les aidants, quelles différences sont significatives et quelles différences le sont moins.

En plus des données utilisées pour tester la méthode, il faut prendre en compte le caractère stochastique de la méthode. Les expériences menées directement sur les modèles, sans apprentissage, indiquent que si le bruit n'est pas important, permettant de détecter les évolutions, celui-ci n'est pas nul. Ainsi, la distance n'est pas complètement déterministe, et une distance faible n'est pas détectable.

Une autre limite est due à l'apprentissage. Les tests effectués montrent qu'avec les méthodes les plus robustes à l'apprentissage, les distances importantes sont détectables. Néanmoins, l'apprentissage implique un bruit important et les résultats sont très variables. Cela implique que si la distance entre deux modèles est insuffisante par rapport à la variabilité, les résultats ne sont pas significatifs et la distance entre deux modèles ne peut pas être calculée. Les méthodes utilisées ne permettent pas de détecter des faibles variations pour les modèles appris, mais uniquement des variations importantes. La robustesse à l'apprentissage de la méthode, bien que non nulle, reste faible et pourrait faire le sujet d'une amélioration.

Chapitre 7

Conclusion et discussion

Sommaire

7.1 Résultats	108
7.2 Discussions	108
7.3 Perspectives	109
7.3.1 Évolutions du modèle Gs-GsHMM	109
7.3.2 Évolutions de l'apprentissage	110
7.3.3 Méthode de calcul des distances	111

7.1 Résultats

Afin d'assister les aidants en [EHPAD](#), nous souhaitons mettre en place une méthode permettant de détecter des comportements inhabituels ou des évolutions du comportement chez les résidents, en prenant en compte les différentes spécificités, que ce soit les personnes ou la typologie de l'environnement [smart home](#). Nous proposons pour cela d'utiliser des modèles de Markov, les [Gs-GsHMM](#), afin de modéliser le comportement propre de chaque personne, appris à partir de données capteurs. Ces modèles peuvent ensuite être utilisés afin d'évaluer la vraisemblance des données actuelles ou la distance entre différents modèles. Ces informations permettant d'informer les aidants des activités inhabituelles et potentiellement problématiques des résidents.

L'apprentissage des modèles a été testé sur différents types de données afin d'évaluer les limites de la méthode, notamment dans le cas de l'utilisation de données acquises en situation réelle. De même, différentes méthodes de calcul de distance entre des modèles de Markov ont été évaluées sur différentes situations et une nouvelle méthode permettant de calculer cette distance en réduisant le temps de calcul a été proposée.

7.2 Discussions

Une première limite de notre travail est liée aux données utilisées pour expérimenter les méthodes développées. En effet, nous ne disposons pas de suffisamment de données acquises en situations réelles pour apprendre et valider notre modèle. Les méthodes sont donc testées sur des données générées artificiellement. Ces simulations restent néanmoins limitées et ne sont pas complètement représentatives des problématiques liées aux conditions réelles.

Pour passer aux conditions réelles, aussi bien pour l'apprentissage que pour le calcul des distances appliquées à des données réelles, un certain nombre de paramètres reste à définir, afin que les résultats soient cohérents et correspondent aux besoins des aidants. Par exemple, il faudrait définir à partir de quelle valeur le changement est significatif d'une évolution afin de réduire le nombre de faux positifs. De même, la quantité de données nécessaires afin que le modèle soit significatif doit être définie. Cet élément est également lié au temps nécessaire entre deux apprentissages.

Le passage aux conditions réelles doit également prendre en compte l'ensemble des capteurs. Dans les expérimentations actuelles, l'acquisition des données est fortement liée aux [PIR](#). Le modèle [Gs-GsHMM](#) n'est pas lié à un type spécifique de capteur,

tant que leurs émissions puissent être exprimées comme un réel. L'expressivité des modèles dépend fortement des données utilisées, le modèle ne pouvant apprendre que les informations dont il dispose. Cela signifie qu'il faudrait tester l'apprentissage avec différents capteurs afin d'optimiser le système. Cela revient à définir les capteurs qui ne sont pas significatifs et ceux qui sont redondants pour pouvoir les retirer des séquences apprises. À l'inverse, il faut également définir les données supplémentaires permettant de mieux caractériser le comportement de la personne, par exemple le jour de la semaine.

De plus, il faudrait tester la pertinence des modèles [Gs-GsHSMM](#) dans le cadre de différents types de pathologies. Si le modèle doit prendre en compte un certain nombre de caractéristiques, comme les durées ou l'incertain. Certaines pathologies sont caractérisées par d'autres éléments, comme la répétition d'une activité spécifique au cours d'une période donnée. Le nombre de répétitions ne doit pas dépasser un seuil. En effet, si la répétition d'une même activité peut être normale, la répétition un trop grand nombre de fois peut indiquer que la personne a des difficultés cognitives ou physiques, par exemple un comportement d'utilisation, décrit par [Lhermitte \[1983\]](#). Le seuil où le nombre de répétitions devient anormal est défini en fonction des habitudes de la personne. Certains seuils pouvant être considérés comme critiques. De plus, l'évolution du nombre de fois que la personne effectue une activité au cours du temps peut également être significative de l'état de la personne, comme le nombre de lever au cours d'une nuit.

7.3 Perspectives

Si les données acquises en situation réelle ne permettent pas actuellement de valider la méthode dans le cas réel, l'apprentissage a pu être testé sur des données simulées, ainsi que les méthodes appliquées pour le calcul des distances entre les modèles. Différentes limites de la méthode ont pu être mises en avant. S'il est nécessaire de valider les méthodes sur davantage de données, notamment acquises en situation réelle, certaines modifications seraient possibles afin d'améliorer théoriquement les résultats.

7.3.1 Évolutions du modèle Gs-GsHSMM

Si le modèle [Gs-GsHSMM](#) doit permettre de modéliser le comportement humain, certains éléments doivent être modifiés. Ainsi, si le modèle prend en compte différentes caractéristiques permettant de différencier des comportements, plusieurs autres

caractéristiques ne le sont pas. Ainsi, le modèle actuel ne prend pas en compte de façon explicite les répétitions d'une activité, comme vu précédemment. Dans le modèle actuel, une activité qui se répète pourrait être détectée via une activité dont la probabilité de transition est élevée mais dont la durée est courte, ou par deux états dont les probabilités de transitions l'un vers l'autre sont élevées. La répétition d'une activité n'est néanmoins pas détectée directement, ni son nombre d'occurrence durant une période donnée. Afin que le modèle corresponde davantage au comportement de la personne, il serait nécessaire d'améliorer celui-ci afin d'augmenter son expressivité en prenant en compte l'ensemble des caractéristiques du comportement de la personne. Une autre solution serait de gérer ces caractéristiques via des variables extérieures et de les introduire dans les observations.

Dans la section 5.1, on observe les probabilités de transitions depuis l'état correspondant à la zone entre le couloir et les zones du lit ne permettent pas de différencier le sens de passage. En effet, l'état est équivalent que la personne va vers le couloir ou vers les zones du lit. Quand le modèle atteint l'état transitoire, après la période de stationnement, l'état change en fonction de la matrice de probabilité. Si cela n'est pas faux au niveau du modèle, ce n'est pas forcément très représentatif du comportement habituel d'une personne. Afin d'augmenter l'expressivité du modèle, il serait possible d'augmenter l'ordre du modèle de Markov, en passant par exemple à un modèle d'ordre 2, c'est à dire où l'état du modèle ne dépend pas seulement du dernier état, mais également de l'avant-dernier, c'est à dire $P(s_{t+1}|s_t, s_{t-1}, s_{t-2}...) = P(s_{t+1}|s_t, s_{t-1})$. Le fait d'étendre l'hypothèse de Markov à l'ordre 2 pourrait permettre de mieux modéliser le comportement, mais cela impliquerait l'augmentation du nombre de variables à apprendre.

Un autre élément à prendre en compte est le choix des observations. Les expérimentations en situation réelle ont été effectuées en utilisant un nombre très réduit de capteurs, notamment des PIR. Le modèle n'étant pas lié à un type spécifique de capteurs, d'autres types de capteurs pourront être utilisés, plus tard, afin d'améliorer le modèle de comportement habituel de la personne. Augmenter la taille du vecteur d'observations a néanmoins un coût en temps de calcul et certaines données ne sont pas forcément pertinentes. Il faudra donc définir quel ensemble de données est le plus significatif et utile pour apprendre le modèle de comportement habituel de la personne.

7.3.2 Évolutions de l'apprentissage

Comme vu dans la sous-section 3.4.3, l'algorithme de Baum-Welch permet de trouver un modèle optimisant la séquence d'observations, à partir d'un modèle original. Le résultat de l'algorithme dépend donc de ce dernier. Afin d'optimiser le résultat, l'algo-

l'algorithme Baum-Welch est lancé plusieurs fois avec des modèles définis aléatoirement. Afin de réduire le temps de calcul, il serait possible de réduire le nombre d'exécutions avec des modèles aléatoires, en utilisant des modèles initiaux issues d'une bibliothèque de modèles de comportement habituel type. Pour construire celle-ci, il serait nécessaire d'avoir une base de données contenant de nombreux exemples de comportement habituel acquis en situation réelle. Il serait possible d'utiliser une méthode comme les k-means en calculant les distances entre les modèles grâce à l'une des méthodes présentées dans le chapitre 6, afin de déterminer un ensemble significatif de modèles de comportements observables en EHPAD. Si cela permet de réduire le temps de calcul, cette méthode nécessite un nombre suffisant de modèles de comportements acquis en situation réelle comparable. En plus du problème d'acquisition des données, il est possible que les spécificités entre les EHPAD soient plus significatives que les différences entre les comportements. Par exemple, bien que les capteurs soient placés de manière standardisée, leur position peut varier en fonction de la typologie des chambres. De plus, tous les capteurs ne sont pas forcément actifs dans l'ensemble des chambres, créant un biais.

Une autre évolution possible, visant à améliorer la modélisation du comportement de la personne serait d'adapter les différents paramètres, c'est à dire le nombre d'états cachés et la durée de stationnement maximum. L'optimisation de ces derniers implique différentes contraintes, notamment dûes au fait qu'elles soient fortement liées (pour plus de détails, revoir le chapitre 5). Le choix des valeurs doit donc se faire en parallèle, et ne peut pas se faire en augmentant une valeur jusqu'à trouver le maximum d'un critère donné. Il faut donc trouver une méthode permettant d'optimiser les paramètres par rapport à un critère donné à définir.

En plus des paramètres du modèle, le nombre de jours appris doit également être déterminé afin que le modèle soit pertinent en un temps de calcul acceptable. Plus le nombre de jours est important, plus le temps de calcul est important. Néanmoins, il est nécessaire de prendre suffisamment de données pour que les modèles soient pertinents. Cela implique que le nombre de jours utilisés doit être optimisé. Cela pourra se faire en ayant suffisamment de données et il faudra identifier les paramètres à considérer afin de définir le nombre minimal de jours.

7.3.3 Méthode de calcul des distances

La méthode testée pour calculer la distance entre deux Gs-GsHSMM est utile pour des distances suffisantes. Bien que le bruit issu de la méthode soit faible pour des modèles proches, il est considérable dans le cas de modèles appris. En effet, les varia-

tions dûes à l'apprentissage sont très importantes et seules les distances suffisamment grandes sont significatives. Cela traduit une nouvelle contrainte, la robustesse à l'apprentissage. Les méthodes testées ont différents niveaux de robustesses, mais aucune ne présente des résultats complètement conformes aux résultats théoriques. Une évolution nécessaire du travail est donc de mettre en place une méthode de calcul de distance entre deux [Gs-GsHMM](#) assurant une forte robustesse à l'apprentissage.

Si l'on considère la méthode actuelle, le résultat n'est pas déterministe. On observe des variations, notamment dans le cas des distances importantes. La convergence des vraisemblances n'assure donc pas un résultat déterministe, et le bruit doit être étudié afin de déterminer son origine et trouver des critères permettant de réduire ces variations.

Annexes

A Glossaire

- AAL** Assistance à la vie autonome ou *Ambient Assisted Living*. 6
- ADL** Activités de la vie quotidienne ou *Activities of Daily Living*. Ensemble d'activités effectuées quotidiennement permettant d'évaluer le niveau de dépendance de la personne.. 18–21, 23, 25, 26, 38
- AIC** critère d'information d'Akaike ou *Akake Information Criterion*. 71, 72
- BIC** critère d'information bayésien ou *Bayesian Information Criterion*. 71
- EDHMM** Modèle de Markov caché à Durée Explicite ou *Explicit Duration-Hidden Markov Model*. Également appelé HSMM.. 37
- EHPAD** Établissement d'Hébergement pour Personnes Âgées Dépendantes. 2, 3, 6–8, 10–12, 15, 16, 20, 47, 56, 62, 66, 96, 99
- GIR** Groupe Iso-Ressources. Classification permettant de d'expliciter le niveau d'autonomie de la personne. L'échelle de classification va de 1 à 6, 6 représentant le groupe des personnes complètement autonome, et 1 le groupe des personnes ayant perdu toute leur autonomie. 2
- Gs-GsHSMM** Gaussienne-Modèle Semi-Markovien caché Gaussien ou *Gaussian-Gaussian Hidden Semi-Markov Model*. 48, 50–52, 66, 76, 79–81, 85, 87–89, 93, 96, 97, 99, 100
- HMM** Modèle de Markov caché ou *Hidden Markov Model*. Modèle de Markov où les états sont cachés, et des observations émises à chaque pas de temps.. 34–37, 39, 42, 43, 51, 52, 62, 66, 71, 77–80, 88, 89
- HSD** Distance stationnaire HMM ou *HMM Stationary Distance*. Méthode de calcul de distance appliqué pour les HMM. 79, 80, 83, 85, 86, 91
- HSMM** Modèle Semi-Markovien caché ou *Hidden Semi-Markov Model*. Variante des HMM modélisant explicitement la durée de stationnement dans chaque état de manière probabiliste.. 37–39, 43, 48, 51, 53
- iADL** Activités de la vie quotidienne instrumentalisées ou *Instrumental Activities of Daily Living*. Variante des ADL ajoutant un ensemble d'activité nécessitant l'utilisation d'objets.. 18
- KL** divergence de Kullback-Leibler. Méthode de calcul de distance entre deux distributions de probabilités. 77, 78, 82, 83, 85–88, 90
- MoG-HMM** Mélange de Gaussienne-HMM ou *Mixture of Gaussian-HMM*. Variante des HMM où les observations sont modélisé par une mixture de gaussiennes.. 39, 48

PIR Passive Infra Red. Capteurs de présence conçus pour détecter des sources de chaleur en mouvement dans leurs zones de détection, grâce à des cellules pyroélectriques.. [9](#), [11–13](#), [19](#), [21](#), [22](#), [24](#), [46](#), [49](#), [56](#), [62–64](#), [97](#), [98](#)

smart home Environnement équipé de capteurs fournissant des informations sur l'état de la zone. [6–12](#), [17](#), [21](#), [24–27](#), [36–38](#), [96](#)

SVM Support Vector Machines. Méthode d'apprentissage supervisé, construisant une frontière de décision maximisant la distance avec les points appris ([Russell et Norvig \[2009\]](#)). [22](#), [27](#), [32](#), [33](#)

B Table des figures

2.1	Zone de détection d'un capteur PIR	11
2.2	GIS Madonah	14
2.3	Rejet des données aberrantes. Seuls les données en vert sont gardés, les données en rouge étant aberrantes.	15
2.4	Exemple de q-relaxation	16
2.5	Exemple de résultat de localisation de la personne. À chaque pas de temps, la personne est nécessairement dans la zone jaune.	17
3.1	Exemple de chaîne de Markov à 4 états	37
3.2	Exemple de HMM. À chaque pas de temps, le modèle émet une observation (a, b, c ou d) en fonction de l'état courant.	38
3.3	Exemple de représentation d'un HSMM avec plusieurs émissions par état	40
3.4	Exemple de représentation d'un HSMM avec transition interdite durant une période de stationnement	40
5.1	Scénario effectué	59
5.2	Probabilité de présence dans chacun des états	61
5.3	Probabilité de transition entre les états	62
5.4	Probabilité de durée des états	63
5.5	Variation de la log-vraisemblance en fonction du nombre d'état cachés	69
5.6	Variation de la durée de stationnement maximum dans un état	71
5.7	Variations de la log-vraisemblance en fonction du nombre de jours appris	72
5.8	Plan d'une chambre de la maison de retraite Ambazac avec position des capteurs.	74
5.9	Données des capteurs PIR	75
5.10	Variations de la log-vraisemblance en fonction du nombre d'états et de la durée de stationnement maximum.	77
5.11	Variations de la log-vraisemblance en fonction du nombre de jours utilisés pour l'apprentissage.	78
6.1	Matrices de transition initiale et finale	88
6.2	Probabilités de durées initiales et finales	88
6.3	Évolution des distances avec la méthode hybride pour les modèles λ_t et λ_{t-1}	89
6.4	Évolution des distances avec la méthode hybride pour les modèles λ_t et λ_0	90
6.5	Évolution des distances avec la méthode KL pour les modèles λ_t et λ_0	91
6.6	Évolution des distances avec la méthode HSD pour les modèles λ_t et λ_{t-1}	92

6.7 Évolution des distances avec la méthode HSD pour les modèles λ_t et λ_0	93
6.8 Génération des modèles appris expérimentaux	94
6.9 Génération des modèles évolutifs appris	96
6.10 Évolution des distances avec la méthode KL pour les modèles appris	96
6.11 Distance calculée entre le modèle d'origine et les variantes du modèle évolutif	102
6.12 Nombre de pas de temps avant convergence du calcul entre le modèle d'origine et les variantes du modèle évolutif	103
6.13 Évolution des distances pour les modèles appris	104
6.14 Nombre de pas de temps pour calculer les distances des modèles appris	105

C Rappels des algorithmes

Algorithm 1 Algorithme Forward-Backward pour HMM (Rabiner [1989])

Require: modèle HMM racine (π, a, b) , séquence d'observation O

$$\alpha_1(i) = \pi_i b_i(O_1)$$

for $t = 1 \rightarrow T - 1$ **do**

$$\alpha_{t+1}(j) = [\sum_{i=1}^N \alpha_t(i) a_{ij}] b_j(O_{t+1})$$

end for

$$\beta_T(i) = 1$$

for $t = T - 1 \rightarrow 1$ **do**

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$$

end for

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}$$

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j)$$

$$\bar{\pi}_i = \gamma_1(i)$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$\bar{b}_j(k) = \frac{\sum_{t=1}^T \text{et } O_t = v_k \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)}$$

Return : Modèle appris $(\bar{\pi}, \bar{a}, \bar{b})$

Variables de l'algorithme 1 :

$\alpha_t(i)$ Probabilité des observations jusqu'au pas de temps t et que l'état soit i à ce pas de temps ($P(O_1, O_2, \dots, O_t, S_t = s_i | \lambda)$)

$\beta_t(i)$ Probabilité des observations à partir du pas de temps t et que l'état soit i à ce pas de temps ($P(O_{t+1}, O_{t+2} \dots O_T | S_t = s_i, \lambda)$)

$\xi_t(i, j)$ Probabilité d'être dans l'état i à l'instant t et à l'état j à l'instant $t + 1$ pour un modèle λ donné et une séquence d'observations O donnée ($P(S_t = s_i, S_{t+1} = s_j | O, \lambda)$)

$\gamma_t(i)$ Probabilité d'être dans l'état i à l'instant t pour un modèle λ donné et une séquence d'observations O donnée

Variables de l'algorithme 2 :

- $\alpha_{t|x}(m, d)$ Variable forward. Probabilité que, à l'instant t , l'état soit m et la durée soit d , connaissant la séquence d'observations $O_1, \dots, O_x$
- r_t^{-1} Prédiction à un pas de temps de l'observation à l'instant t , connaissant la séquence d'observations $O_1, \dots, O_{t-1}$ ($P(O_t|O_1, \dots, O_{t-1})$)
- $b_m^*(O_t)$ Rapport entre la probabilité filtrée $\alpha_{t|t}(m, d)$ et la probabilité prédite $\alpha_{t|t-1}(m, d)$
- $\mathcal{E}_t(m)$ Probabilité que, à l'instant t , l'état soit m et que la durée de stationnement restant de l'état soit 1, connaissant la séquence d'observations $O_1, \dots, O_t$
($P(s_t = m, d_t = 1|O_1, \dots, O_t)$)
- $\mathcal{S}_1(m)$ Probabilité, à l'instant t que la durée de stationnement de l'état courant soit 1 et que l'état à l'instant $t+1$ soit m , connaissant la séquence d'observations $O_1, \dots, O_t$
($P(s_{t+1} = m, d_t = 1|O_1, \dots, O_t)$)
- $\beta_t(m, d)$ Variable backward. Rapport entre les probabilités $\alpha_{t|T}(m, d)$ sur les probabilités prédites $\alpha_{t|t-1}(m, d)$
- $\mathcal{E}_t^*(m)$ Rapport des probabilités d'émission de la séquence d'observations $O_t, \dots, O_T$ sachant que, à l'instant t , l'état est m et qu'il vient d'y avoir une transition, sur la probabilité d'émission $O_t, \dots, O_T$, connaissant la séquence d'observation $O_1, \dots, O_{t-1}$
$$O_{t-1} \left(\frac{P(s_t = m, d_{t-1} = 1|O_1, \dots, O_{t-1})}{P(O_t, \dots, O_T|O_1, \dots, O_{t-1})} \right)$$
- $\mathcal{S}_t^*(m, d)$ Rapport des probabilités d'émission de la séquence d'observations $O_t, \dots, O_T$ sachant que, à l'instant $t-1$, l'état est m et la durée de stationnement restant est de 1, sur la probabilité d'émission $O_t, \dots, O_T$, connaissant la séquence d'observation $O_1, \dots, O_{t-1}$
$$O_{t-1} \left(\frac{P(s_{t-1} = m, d_{t-1} = 1|O_1, \dots, O_{t-1})}{P(O_t, \dots, O_T|O_1, \dots, O_{t-1})} \right)$$
- $\xi_t(m, n)$ Probabilité de la transition de l'état m à l'état n à l'instant t
- $\gamma_t(m)$ Probabilité que l'état à l'instant t soit m , connaissant la séquence d'observation.
- $\eta_t(m, d)$ Probabilité, à l'instant t , que l'état soit m , pour une durée de stationnement de d , et que l'état à $t-1$ ne soit pas m

Algorithm 2 Algorithme Forward-Backward pour HSMM (Yu *et al.* [2006])**Require:** modèle HSMM racine (π, a, b, p) , séquence d'observation O

$$\alpha_{1|0}(m, d) = \pi_m p_m(d)$$

$$r_1^{-1} = \sum_{m,d} \alpha_{1|0}(m, d) b_m(O_1)$$

$$b_m^*(O_1) = \frac{b_m(O_1)}{r_1^{-1}}$$

$$\mathcal{E}_1(m) = \alpha_{1|0}(m, 1) b_m^*(O_1)$$

$$\mathcal{S}_1(m) = \sum_n \mathcal{E}_1(n) a_{nm}$$

for $t = 2 \rightarrow T$ **do**

$$\alpha_{t|t-1}(m, d) = \mathcal{S}_{t-1}(m) p_m(d) + b_m^*(o_{t-1}) \alpha_{t-1|t-2}(m, d+1)$$

$$r_t^{-1} = \sum_{m,d} \alpha_{t|t-1}(m, d) b_m(o_t)$$

$$b_m^*(o_t) = \frac{b_m(o_t)}{r_t^{-1}}$$

$$\mathcal{E}_t(m) = \alpha_{t|t-1}(m, 1) b_m^*(o_t)$$

$$\mathcal{S}_t(m) = \sum_n a_{mn} \mathcal{E}_t^*(n)$$

end for

$$\beta_T(m, d) = b_m^*(o_T)$$

$$\mathcal{E}_T^*(m) = \sum_d p_m(d) \beta_T(m, d)$$

$$\mathcal{S}_T^*(m) = \sum_d p_m(d) \beta_T(m, d)$$

$$\gamma_T(m) = \sum_{d \geq 1} \alpha_T(m, d)$$

$$\xi_T(m, n) = \alpha_{T-1}(m, 1) a_{mn} b_n(O_T) (\sum_{d \geq 1} p_n(d) \beta_T(n, d))$$

for $t = T-1 \rightarrow 1$ **do**

$$\beta_t(m, d) = \begin{cases} \mathcal{S}_{t+1}^*(m) b_m^*(o_t) & \text{si } d = 1 \\ \beta_{t+1}(m, d-1) b_m^*(o_t) & \text{si } d > 1 \end{cases}$$

$$\mathcal{E}_t^*(m) = \sum_d p_m(d) \beta_t(m, d)$$

$$\mathcal{S}_t^*(m) = \sum_n a_{mn} \mathcal{E}_t^*(n)$$

$$\xi_t(m, n) = \alpha_{t-1}(m, 1) a_{mn} b_n(O_t) (\sum_{d \geq 1} p_n(d) \beta_t(n, d))$$

$$\gamma_t(m) = \gamma_{t+1}(m) + \sum_{n \neq m} (\xi_{t+1}(m, n) - \xi_{t+1}(n, m))$$

end for

$$\eta_t(m, d) = (\sum_{n \neq m} \alpha_{t-1}(n, 1) a_{nm}) b_m(O_t) p_m(d) \beta_t(m, d)$$

$$\overline{\pi_m} = \frac{\gamma_1(m)}{\sum_{n=1}^M \gamma_1(n)}$$

$$\overline{a_{mn}} = \frac{\sum_{t=1}^T \xi_t(m, n)}{G(m)}$$

$$\overline{p_m(d)} = \frac{\sum_{t=1}^T \eta_t(m, d)}{\sum_{d=1}^D \sum_{t=1}^T \eta_t(m, d)}$$

$$\overline{b_m(v_k)} = \frac{\sum_{t=1}^T \gamma_t(m) \delta(o_t - v_k)}{\sum_k \sum_{t=1}^T \gamma_t(m) \delta(o_t - v_k)}$$

Return : Modèle appris $(\overline{\pi}, \overline{a}, \overline{b}, \overline{p})$

Variables de l'algorithme 3 différentes de l'algorithme 2 :

$\gamma_t(m)$ Probabilité que l'état à l'instant t soit m , connaissant la séquence d'observation.
Sert de poids pour les moyennes pondérées déterminant les paramètres d'observations

w_i Somme sur des poids

$S_y(m, k)$ Somme, pour le k -ème élément du vecteur d'observation, de la valeur courante pondéré par la probabilité d'être dans l'état m

$S_{yy}(m, k)$ Somme, pour le k -ème élément du vecteur d'observation, du carré de la valeur courante pondéré par la probabilité d'être dans l'état m

$\Delta_m(d)$ Mise à jour de la probabilité que la durée de stationnement de l'état m soit d sans le passage aux fonctions gaussiennes tronquées. ($\overline{p_m(d)}$)

Algorithm 3 Algorithme Baum-Welch pour Gs-GsHSMM**Require:** modèle Gs-GsHSMM racine (a, B, p) , séquence d'observation O

$$\alpha_{1|0}(m, d) = p_m(d) / N$$

$$r_1^{-1} = \sum_{m,d} \alpha_{1|0}(m, d) b_m(o_1)$$

$$b_m^*(o_1) = \frac{b_m(o_1)}{r_1^{-1}}$$

$$\mathcal{E}_1(m) = \alpha_{1|0}(m, 1) b_m^*(o_1)$$

$$\mathcal{S}_1(m) = \sum_n \mathcal{E}_1(n) a_{nm}$$

for $t = 2 \rightarrow T$ **do**

$$\alpha_{t|t-1}(m, d) = \mathcal{S}_{t-1}(m) p_m(d) + b_m^*(o_{t-1}) \alpha_{t-1|t-2}(m, d+1)$$

$$r_t^{-1} = \sum_{m,d} \alpha_{t|t-1}(m, d) b_m(o_t)$$

$$b_m^*(o_t) = \frac{b_m(o_t)}{r_t^{-1}}$$

$$\mathcal{E}_t(m) = \alpha_{t|t-1}(m, 1) b_m^*(o_t)$$

$$\mathcal{S}_t(m) = \sum_n a_{mn} \mathcal{E}_t^*(n)$$

end for

$$\beta_T(m, d) = b_m^*(o_T)$$

$$\mathcal{E}_T^*(m) = \sum_d p_m(d) \beta_T(m, d)$$

$$\mathcal{S}_T^*(m) = \sum_d p_m(d) \beta_T(m, d)$$

$$\gamma_T(m) = \sum_{d \geq 1} \alpha_T(m, d)$$

$$\xi_T(m, n) = \alpha_{T-1}(m, 1) a_{mn} b_n(o_T) (\sum_{d \geq 1} p_n(d) \beta_T(n, d))$$

for $t = T-1 \rightarrow 1$ **do**

$$\beta_t(m, d) = \begin{cases} \mathcal{S}_{t+1}^*(m) b_m^*(o_t) & \text{si } d = 1 \\ \beta_{t+1}(m, d-1) b_m^*(o_t) & \text{si } d > 1 \end{cases}$$

$$\mathcal{E}_t^*(m) = \sum_d p_m(d) \beta_t(m, d)$$

$$\mathcal{S}_t^*(m) = \sum_n a_{mn} \mathcal{E}_t^*(n)$$

$$\xi_t(m, n) = \alpha_{t-1}(m, 1) a_{mn} b_n(o_t) (\sum_{d \geq 1} p_n(d) \beta_t(n, d))$$

$$\gamma_t(m) = \gamma_{t+1}(m) + \sum_{n \neq m} (\xi_{t+1}(m, n) - \xi_{t+1}(n, m))$$

$$w_m = w_m + \gamma_t(m)$$

$$S_y(m, k) = S_y(m, k) + \gamma_t(m) o_{t,k}$$

$$S_{yy}(m, k) = S_{yy}(m, k) + \gamma_t(m) o_{t,k}^2$$

end for

$$\Delta_m(d) = \mathcal{S}_1(m) p_m(d) \beta_1(m, d)$$

$$\overline{a_{mn}} = \frac{\sum_{t=1}^T \xi_t(m, n)}{G(m)}$$

$$p_m(d) = \text{descenteGradient}(\sum \Delta_m(d) \log p_m(d))$$

$$\overline{\sigma_{B_{m,k}}} = \sqrt{\frac{S_{yy}(m, k)}{w_m} - \left(\frac{S_y(m, k)}{w_i} \right)^2}$$

$$\overline{\mu_{B_{m,k}}} = \frac{S_{ym,k}}{w_m}$$

Return : Modèle appris $(\bar{a}, \bar{B}, \bar{p})$

Algorithm 4 Algorithme de calcul de distance entre deux Gs-GsHSMM - Initialisation

Require: modèle Gs-GsHSMM $\lambda_1 : (\pi, a, b, p)$, modèle Gs-GsHSMM $\lambda_2 : (\pi, a, b, p)$

$(o_1, o_2, s_1, s_2, d_1, d_2) = \text{initialisationObservation}(\lambda_2)$

for $n = 1 \rightarrow \lambda_1.\text{NbEtat}$ **do**

for $d = 1 \rightarrow \lambda_1.\text{DureeMax}$ **do**

$\alpha_{i1}(n, d) = \lambda_1.\pi_n \lambda_1.p_n(d)$

end for

$r_{i1} = r_{i1} + \alpha_{i1}(n, d) \lambda_1.b_n(o_i)$

end for

for $n = 1 \rightarrow \lambda_2.\text{NbEtat}$ **do**

for $d = 1 \rightarrow \lambda_2.\text{DureeMax}$ **do**

$\alpha_{i2}(n, d) = \lambda_2.\pi_n \lambda_2.p_n(d)$

end for

$r_{i2} = r_{i2} + \alpha_{i2}(n, d) \lambda_2.b_n(o_i)$

end for

$\text{lkh}_{ij} = \log(r_{ij})$

for $n = 1 \rightarrow \lambda_1.\text{NbEtat}$ **do**

$b_{mi1}^*(o_i) = \frac{\lambda_1.b_m(o_i)}{r_{i1}}$

$\mathcal{E}_{1i1}(m) = \alpha_{i1}(m, 1) b_{mi1}^*(o_i)$

for $m = 1 \rightarrow \lambda_1.\text{NbEtat}$ **do**

$\mathcal{S}_{1i1}(n) = \mathcal{S}_{1i1}(n) + \mathcal{E}_{1i1}(m) \lambda_1.a_{nm}$

end for

end for

for $n = 1 \rightarrow \lambda_2.\text{NbEtat}$ **do**

$b_{mi2}^*(o_i) = \frac{\lambda_2.b_m(o_i)}{r_{i2}}$

$\mathcal{E}_{1i2}(m) = \alpha_{i2}(m, 1) b_{mi2}^*(o_i)$

for $m = 1 \rightarrow \lambda_2.\text{NbEtat}$ **do**

$\mathcal{S}_{1i2}(n) = \mathcal{S}_{1i2}(n) + \mathcal{E}_{1i2}(m) \lambda_2.a_{nm}$

end for

end for

$t = 1$

Algorithm 5 Algorithme de calcul de distance entre deux Gs-GsHSM - Partie 2

```

while  $(\text{lkh}_{11} - \text{lkh}_{12} - \text{lkh}_{21} + \text{lkh}_{22}) / t > \epsilon$  do
   $t = t + 1$ 
   $(o_1, o_2, s_1, s_2, d_1, d_2) = \text{generationObservation}(\lambda_2, s_1, s_2, d_1, d_2)$ 
  for  $n = 1 \rightarrow \lambda_1.\text{NbEtat}$  do
    for  $d = 1 \rightarrow \lambda_1.\text{DureeMax}$  do
       $\alpha_{i1}(n, d) = \mathcal{S}_{(t-1)i1}(n) \lambda_1.p_n(d) + \lambda_1.b_n^*(o_i) \alpha_{i1}(n, d+1)$ 
       $r_{i1} = r_{i1} + \alpha_{i1}(n, d) \lambda_1.b_n(o_i)$ 
    end for
  end for
  for  $n = 1 \rightarrow \lambda_2.\text{NbEtat}$  do
    for  $d = 1 \rightarrow \lambda_2.\text{DureeMax}$  do
       $\alpha_{i2}(n, d) = \mathcal{S}_{(t-1)i2}(n) \lambda_2.p_n(d) + \lambda_2.b_n^*(o_i) \alpha_{i2}(n, d+1)$ 
       $r_{i2} = r_{i2} + \alpha_{i2}(n, d) \lambda_2.b_n(o_i)$ 
    end for
  end for
   $\text{lkh}_{ij} = \text{lkh}_{ij} + \log(r_{ij})$ 
  for  $n = 1 \rightarrow \lambda_1.\text{NbEtat}$  do
     $b_{ni1}^*(o_i) = \frac{\lambda_1.b_n(o_i)}{r_{i1}}$ 
     $\mathcal{E}_{ti1}(n) = \alpha_{i1}(n, 1) b_{ni1}^*(o_i)$ 
    for  $m = 1 \rightarrow \lambda_1.\text{NbEtat}$  do
       $\mathcal{S}_{ti1}(n) = \mathcal{S}_{ti1}(n) + \lambda_1.a_{nm} \mathcal{E}_{ti1}(m)$ 
    end for
  end for
  for  $n = 1 \rightarrow \lambda_2.\text{NbEtat}$  do
     $b_{ni2}^*(o_i) = \frac{\lambda_2.b_n(o_i)}{r_{i2}}$ 
     $\mathcal{E}_{ti2}(n) = \alpha_{i2}(n, 1) b_{ni2}^*(o_i)$ 
    for  $m = 1 \rightarrow \lambda_2.\text{NbEtat}$  do
       $\mathcal{S}_{ti2}(n) = \mathcal{S}_{ti2}(n) + \lambda_2.a_{nm} \mathcal{E}_{ti2}(m)$ 
    end for
  end for
end while
Return : Distance calculé  $(\text{lkh}_{11} - \text{lkh}_{12} + \text{lkh}_{21} - \text{lkh}_{22}) / 2t$ 

```

Algorithm 6 Algorithme de Viterbi (Rabiner [1989])

Require: modèle HMM (π, a, b) , séquence d'observation O

$$\delta_1(i) = \pi_i b_i(O_1)$$

$$\psi_1(i) = 0$$

for $t = 2 \rightarrow T$ **do**

$$\delta_t(i) = \max_{1 \leq j \leq N} [\delta_{t-1}(j) a_{ji}] b_i(O_t)$$

$$\psi_t(i) = \arg \max_{1 \leq j \leq N} [\delta_{t-1}(j) a_{ji}]$$

end for

$$s_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)]$$

for $t = T \rightarrow 2$ **do**

$$s_t^* = \psi_{t+1}(s_{t+1}^*)$$

end for

Return : Séquence d'état la plus probable s

Algorithm 7 Génération de la liste L_O des observations

for all dimension i du vecteur d'observation **do**

if l'observation i est binaire **then**

$$\text{choix}_i = [0, 1]$$

else

$$V_{\min} = \min(\mu_1(s, i) - 4 * \sigma_1(s, i), \mu_2(s, i) - 4 * \sigma_2(s, i)), \forall s$$

$$V_{\max} = \max(\mu_1(s, i) + 4 * \sigma_1(s, i), \mu_2(s, i) + 4 * \sigma_2(s, i)), \forall s$$

$$p = (V_{\max} - V_{\min}) / N$$

$$\text{choix}_i = [V_{\min} + j * p] \quad \forall j \in [1, 2, \dots, N]$$

end if

end for

L_O est la combinaison de tous les choix_i

D Références

- Acampora, G., D. Cook, P. Rashidi et A. Vasilakos. 2013, «A Survey on Ambient Intelligence in Healthcare», vol. 101, n° 12, doi :10.1109/JPROC.2013.2262913.A, p. 2470–2494. [8](#)
- Amiribesheli, M., A. Benmansour et A. Bouchachia. 2015, «A review of smart homes in healthcare», *Journal of Ambient Intelligence and Humanized Computing*, doi :10.1007/s12652-015-0270-2, ISSN 1868-5137. [8](#)
- Amri, M.-H. 2015, *Fusion ensembliste de données pour la surveillance des personnes dépendantes en habitat intelligent*, Thèse de doctorat, Université d'Orléans. [14](#), [16](#)
- Amri, M.-H., Y. Becis, A. Didier et N. Ramdani. 2015, «Indoor Human/Robot Localisation using Robust Multi-modal Data Fusion», in *IEEE International Conference on Robotics and Automation (ICRA)*. [14](#), [74](#), [75](#)
- Andries, M., O. Simonin et F. Charpillet. 2015, «Localisation of humans, objects and robots interacting on load-sensing floors», *IEEE Sensors Journal*, vol. 6, n° 4, p. 1026–1037. [14](#)
- Anesm. 2012, «Qualité de vie en Ehpad (volet 3) - La vie sociale des résidents en EH-PAD», . [3](#)
- Ayers, D. et M. Shah. 2001, «Monitoring human behavior from video taken in an office environment», *Image and Vision Computing*, vol. 19. [8](#), [10](#)
- Azam, M. A., J. Loo, S. K. A. Khan, U. Naeem, M. Adeel et W. Ejaz. 2012, «Behavioural patterns analysis of low entropy people using proximity data», *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 3, n° 3, p. 21–40. [19](#)
- Barbu, V., J. Bulla et A. Maruotti. 2012, «Estimation of the stationary distribution of a semi-Markov chain», *Journal of Reliability and Statistical Studies*, vol. 5, p. 15–26. [86](#)
- Benoit, M., P. Staccini, P. Brocker, T. Benhamidat, C. Bertogliati, L. Lechowski, D. Tortrat et P. H. Robert. 2003, «Symptômes comportementaux et psychologiques dans la maladie d'Alzheimer : Résultats de l'étude REAL.FR», *Revue de Medecine Interne*, vol. 24, n° SUPPL. 3, doi :10.1016/S0248-8663(03)80690-2, p. 319–324, ISSN 02488663. [18](#)
- Blanpain, N. et O. Chardon. 2010, «Projections de population à l'horizon 2060 : Un tiers de la population âgé de plus de 60 ans», *INSEE Première*, vol. 1320, p. 4. [2](#)

- Botia, J. a., A. Villa et J. Palma. 2012, «Ambient Assisted Living system for in-home monitoring of healthy independent elders», *Expert Systems with Applications*, vol. 39, n° 9, doi :10.1016/j.eswa.2012.01.153, p. 8136–8148, ISSN 09574174. [24](#), [27](#)
- Brax, N., E. Andonoff et M.-P. Gleizes. 2012, «A Self-adaptive Multi-Agent System for Abnormal Behavior Detection in Maritime Surveillance», *Agent and Multi-Agent Systems. Technologies and Applications*, vol. 7327, p. 174–185. [17](#)
- Brdiczka, O., M. Langet, J. Maisonnasse et J. Crowleu. 2009, «Detecting human behavior models from multimodal observation in a smart home», *IEEE Transactions on automation science and engineering*, p. 1–10. [12](#), [20](#), [38](#)
- Chaaaraoui, A. A., P. Climent-Pérez et F. Flórez-Revuelta. 2012, «A review on vision techniques applied to Human Behaviour Analysis for Ambient-Assisted Living», *Expert Systems with Applications*, vol. 39, n° 12, doi :10.1016/j.eswa.2012.03.005, p. 10 873–10 888, ISSN 09574174. [10](#), [17](#), [18](#), [20](#)
- Chamroukhi, F., S. Mohammed, D. Trabelsi, L. Oukhellou et Y. Amirat. 2013, «Joint segmentation of multivariate time series with hidden process regression for human activity recognition», *Neurocomputing*, vol. 120, doi :10.1016/j.neucom.2013.04.003, p. 633–644, ISSN 09252312. [10](#)
- Chan, M., D. Estève, C. Escriba et E. Campo. 2008, «A review of smart homes- present state and future challenges.», *Computer methods and programs in biomedicine*, vol. 91, n° 1, doi :10.1016/j.cmpb.2008.02.001, p. 55–81, ISSN 0169-2607. [8](#)
- Chandola, V., A. Banerjee et V. Kumar. 2012, «Anomaly detection for discrete sequences : A survey», *Knowledge and Data Engineering, IEEE Transactions on*, vol. 24, n° 5, p. 823–839. [82](#)
- Charpin, J.-M. et C. Tlili. 2011, «Perspectives démographiques et financières de la dépendance», *rapport du groupe*, , n° 2. [2](#)
- Chen, C., D. Zhang, L. Sun, M. Hariz et B. Jean-Bart. 2013, «AQUEDUC : Improving Quality and Efficiency of Care for Elders in Real Homes», *in International Conference on Smart Homes and Health Telematics - ICOST*. [22](#), [27](#)
- Chen, L., C. D. Nugent et H. Wang. 2012, «A Knowledge-Driven Approach to Activity Recognition in Smart Homes», *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, n° 6, doi :10.1109/TKDE.2011.51, p. 961–974, ISSN 1041-4347. [28](#)
- Chua, S.-L., S. Marsland et H. W. Guesgen. 2011, «Unsupervised Learning of Human Behaviours», *in AAAI*, p. 319–324. [24](#), [28](#)

- Chung, P.-C. et C.-D. Liu. 2008, «A daily behavior enabled hidden Markov model for human behavior understanding», *Pattern Recognition*, vol. 41, n° 5, doi :10.1016/j.patcog.2007.10.022, p. 1572–1580, ISSN 00313203. [20](#), [39](#)
- Cohen, L., T. Desmidt et F. Limosin. 2014, «La psychiatrie de la personne âgée : enjeux et perspectives», *Annales Médico-psychologiques, revue psychiatrique*, vol. 172, n° 9, doi :10.1016/j.amp.2014.09.001, p. 781–784, ISSN 00034487. [17](#)
- Cook, D. J. 2006, «Health Monitoring and Assistance to Support Aging in Place», *Universal Computer Science*, vol. 12, n° 1, doi :10.3217/jucs-012-01-0015, p. 15–29, ISSN 0958695X. [26](#), [29](#), [39](#)
- Cook, D. J. et N. C. Krishnan. 2015, *Activity Learning : Discovering, Recognizing, and Predicting Human Behavior from Sensor Data*, John Wiley & Sons. [12](#), [17](#), [20](#)
- Cook, D. J., N. C. Krishnan et P. Rashidi. 2013, «Activity discovery and activity recognition : a new partnership.», *IEEE transactions on cybernetics*, vol. 43, n° 3, doi : 10.1109/TSMCB.2012.2216873, p. 820–8, ISSN 2168-2275. [24](#), [29](#)
- Costa, A., J. C. Castillo, P. Novais, A. Fernández-Caballero et R. Simoes. 2012, «Sensor-driven agenda for intelligent home care of the elderly», *Expert Systems with Applications*, vol. 39, n° 15, doi :10.1016/j.eswa.2012.04.058, p. 12 192–12 204, ISSN 09574174. [9](#), [10](#)
- De Silva, L. C., C. Morikawa et I. M. Petra. 2012, «State of the art of smart homes», *Engineering Applications of Artificial Intelligence*, vol. 25, n° 7, doi :10.1016/j.engappai.2012.05.002, p. 1313–1321, ISSN 09521976. [8](#)
- Demongeot, J., G. Virone, F. Duchêne, G. Benchetrit, T. Hervé, N. Noury et V. Rialle. 2002, «Multi-sensors acquisition, data fusion, knowledge mining and alarm triggering in health smart homes for elderly people.», *Comptes rendus biologies*, vol. 325, n° 6, p. 673–82, ISSN 1631-0691. [8](#), [49](#)
- Dewar, M., C. Wiggins et F. Wood. 2012, «Inference in Hidden Markov Models with Explicit State Duration Distributions», *IEEE Signal Processing Letters*, vol. 19, n° 4, doi : 10.1109/LSP.2012.2184795, p. 235–238, ISSN 1070-9908. [39](#)
- Dufour-Renouf, F. 2006, «Développer un projet d'animation en EHPAD au service de la vie sociale des résidents, École Nationale de la Santé Publique», . [2](#)
- Duong, T., D. Phung, H. Bui et S. Venkatesh. 2009, «Efficient duration and hierarchical modeling for human activity recognition», *Artificial Intelligence*, vol. 173, n° 7-8, doi : 10.1016/j.artint.2008.12.005, p. 830–856, ISSN 00043702. [41](#)

- Endelin, R., S. Renouard, T. Tiberghien, H. Aloulou et M. Mokhtari. 2013, «Behavior Recognition for Elderly People in Large-Scale Deployment», in *11th International Conference on Smart Homes and Health Telematics, ICOST 2013*, p. 61–68. [24](#), [28](#)
- Evers, W., W. Tomic et A. Brouwers. 2002, «Aggressive behaviour and burnout among staff of homes for the elderly.», *International journal of mental health nursing*, vol. 11, n° 1, doi :10.1080/01612840151136975, p. 2–9, ISSN 0161-2840. [3](#)
- Falkhausen, M., H. Reininger et D. Wolf. 1995, «Calculation of distance measures between hidden markov models», in *EUROSPEECH*, p. 4–7. [83](#), [98](#), [101](#)
- Fang, H., L. He, H. Si, P. Liu et X. Xie. 2014, «Human activity recognition based on feature selection in smart home using back-propagation algorithm.», *ISA transactions*, doi : 10.1016/j.isatra.2014.06.008, p. 1–10, ISSN 1879-2022. [21](#), [29](#)
- Goernitz, N., M. Braun et M. Kloft. 2015, «Hidden Markov Anomaly Detection», in *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, p. 1833–1842. [82](#)
- Gupta, M., J. Gao, C. C. Aggarwal et J. Han. 2014, «Outlier Detection for Temporal Data : A Survey», *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, n° 1, doi : 10.1109/TKDE.2013.184, ISSN 1041-4347. [82](#)
- Hayes, T., M. Pavel, N. Larimer et I. Tsay. 2007, «Simultaneous assessment of multiple individuals», *IEEE Pervasive Computing/IEEE Computer Society [and] IEEE Communications Society*, vol. 6, p. 36—43. [26](#)
- Hayes, T. L., M. Pavel et J. a. Kaye. 2004, «An unobtrusive in-home monitoring system for detection of key motor changes preceding cognitive decline.», in *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 4, ISBN 0780384393, ISSN 1557-170X, p. 2480–3, doi:10.1109/IEMBS.2004.1403715. [26](#)
- Hodges, M., M. Newman et M. Pollack. 2009, «Object-Use Activity Monitoring : Feasibility for People with Cognitive Impairments.», in *AAAI Spring Symposium : Human Behavior Modeling*, p. 13–18. [10](#)
- Hossain, M. A. 2014, «Perspectives of human factors in designing elderly monitoring system», *Computers in Human Behavior*, vol. 33, doi :10.1016/j.chb.2013.12.010, p. 63–68, ISSN 07475632. [8](#)

- Huang, B., G. Tian, H. Wu et F. Zhou. 2014, «A method of abnormal habits recognition in intelligent space», *Engineering Applications of Artificial Intelligence*, vol. 29, doi : 10.1016/j.engappai.2013.12.010, p. 125–133, ISSN 09521976. [22](#)
- Jain, A., M. Murty et P. Flynn. 1999, «Data clustering : a review», *ACM computing surveys (CSUR)*. [35](#)
- Jaulin, L. 2001, «Applied interval analysis : with examples in parameter and state estimation, robust control and robotics», *Springer Science & Business Media*, vol. 1. [14](#), [16](#)
- Karaman, S., J. Benois-Pineau, V. Dovgalecs, R. Mégret, J. Piquier, R. André-Obrecht, Y. Gaëstel et J.-F. Dartigues. 2012, «Hierarchical Hidden Markov Model in detecting activities of daily living in wearable videos for studies of dementia», *Multimedia Tools and Applications*, , n° 1, doi :10.1007/s11042-012-1117-x, p. 1–30, ISSN 1380-7501. [12](#), [39](#)
- Kasteren, T. L. M., G. Englebienne et B. J. A. Kröse. 2010, «An activity monitoring system for elderly care using generative and discriminative models», *Personal and Ubiquitous Computing*, vol. 14, n° 6, doi :10.1007/s00779-009-0277-9, p. 489–498, ISSN 1617-4909. [21](#), [39](#)
- Katz, S., A. B. Ford, R. W. Moskowitz, B. A. Jackson et M. W. Jaffe. 1963, «Studies of Illness in the Aged. The Index of ADL : A Standardized Measure of Biological and Psychosocial Function», *JAMA : the journal of the American Medical Association*, vol. 185, p. 914—919. [20](#)
- KPMG. 2014, «Observatoire des EHPAD 2014», . [3](#)
- Krishnan, N. C. et D. J. Cook. 2014, «Activity recognition on streaming sensor data», *Pervasive and Mobile Computing*, vol. 10, doi :10.1016/j.pmcj.2012.07.003, p. 138–154, ISSN 15741192. [21](#)
- Lawton, M. P. et E. M. Brody. 1969, «Assessment of older people : self-maintaining and instrumental activities of daily living.», *The Gerontologist*, vol. 9, n° 3, p. 179–86, ISSN 0016-9013. [20](#)
- Le, X. H. B., M. Di Mascolo, A. Gouin et N. Noury. 2007, «Health smart home : towards an assistant tool for automatic assessment of the dependence of elders.», in *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2007, ISBN 1424407885, ISSN 1557-170X, p. 3806–3809, doi : 10.1109/IEMBS.2007.4353161. [21](#), [28](#)

- LeCun, Y., L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drucker, I. Guyon, U. Müller, E. Säckinger, P. Simard et V. Vapnik. 1995, «Comparison of learning algorithms for handwritten digit recognition», in *International Conference on artificial neural networks*, p. 53–60. [34](#)
- Lee, D., S. Helal et Y. Sung. 2013, «Assessing Behavioral Responses in Persuasive Ubiquitous Systems», in *International Conference on Smart Homes and Health Telematics - ICOST*. [48](#), [49](#)
- Leroux, B. G. 1992, «Maximum-likelihood estimation for hidden Markov models», *Stochastic processes and their applications*, vol. 40, n° 1, p. 127–143. [99](#)
- Lhermitte, F. 1983, «'Utilization behaviour' and its relation to lesions of the frontal lobes.», *Brain : a journal of neurology*, vol. 106 (Pt 2), p. 237–55, ISSN 0006-8950. [18](#), [35](#), [109](#)
- Liu, C.-D., P.-C. Chung, Y.-N. Chung et M. Thonnat. 2007, «Understanding of human behaviors from videos in nursing care monitoring systems», *Journal of High Speed Networks*, vol. 16, n° 1, p. 91–103. [21](#)
- Liu, J., J. C. Augusto et H. Wang. 2006, «Considerations on Uncertain Spatio-Temporal Reasoning in Smart Home Systems», in *International Conference on Applied Artificial Intelligence (FLINS 2006)*, World Scientific Publishing Co. Pte. Ltd., Singapore, ISBN 9789812566904, p. 817–824, doi :10.1142/9789812774118_0115. [27](#)
- Lu, C., J. M. Schwiier, R. M. Craven, L. Yu, R. R. Brooks et C. Griffin. 2013, «A normalized statistical metric space for hidden Markov models», *IEEE Transactions on Cybernetics*, vol. 43, n° 3, doi :10.1109/TSMCB.2012.2216872, p. 806–819, ISSN 21682267. [86](#)
- Medjahed, H., D. Istrate, J. Boudy, J.-L. Baldinger et B. Dorizzi. 2011, «A pervasive multi-sensor data fusion for smart home healthcare monitoring», *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, doi :10.1109/FUZZY.2011.6007636, p. 1466–1473. [12](#), [21](#), [27](#)
- Mehran, R., A. Oyama et M. Shah. 2009, «Abnormal Crowd Behavior Detection using Social Force Model», *IEEE conference on Computer Vision and Pattern Recognition*, , n° 1. [17](#)
- Merico, D., A. Mileo, S. Pinardi et R. Bisiani. 2010, «A Logical Approach to Home Healthcare with Intelligent Sensor-Network Support», *The Computer Journal*, vol. 53, n° 8, doi :10.1093/comjnl/bxn074, p. 1257–1276, ISSN 0010-4620. [27](#)

- Mubashir, M., L. Shao et L. Seed. 2013, «A survey on fall detection : Principles and approaches», *Neurocomputing*, vol. 100, doi :10.1016/j.neucom.2011.09.037, p. 144–152, ISSN 09252312. [8](#)
- Murphy, K. P. 2002, «Hidden semi-Markov models (HSMMs)», Rapport technique November. [56](#)
- Noury, N., P. Barralon, N. Vuillerme et A. Fleury. 2012, «Fusion of Multiple Sensors Sources in a Smart Home to Detect Scenarios of Activities in Ambient Assisted Living», *International Journal of E-Health and Medical Communications*, vol. 3, n° 3, doi:10.4018/jehmc.2012070103, p. 29–44, ISSN 1947-315X. [21](#), [27](#)
- Noury, N. et T. Hadidi. 2012, «Computer simulation of the activity of the elderly person living independently in a Health Smart Home.», *Computer methods and programs in biomedicine*, vol. 108, n° 3, doi :10.1016/j.cmpb.2012.07.004, p. 1216–28, ISSN 1872-7565. [23](#), [26](#), [37](#)
- Ollat, H. 2002, «Démence fronto-temporale – Un diagnostic à savoir évoquer», *Neuropsychiatrie : Tendances et Débats*, vol. 18, p. 21–25. [18](#)
- Österreicher, F. 2002, «Csiszár's f-divergences-Basic properties», *RGMIA Res. Rep. Coll*, p. 1–13. [84](#)
- Paris, A., S. Arbaoui, N. Cislo, A. El-Amraoui et N. Ramdani. 2015, «Using Hidden Semi-Markov Model for learning behavior in smarthomes», *IEEE International Conference on Automation Science and Engineering (CASE)*, p. 752–757. [64](#)
- Paris, V. 2012, «Regard sur le concept de dépression mixte», *Université de Franche-Comté*. [18](#)
- Parzen, E., K. Tanabe et G. Kitagawa. 2012, *Selected Papers of Hirotugu Akaike*, Springer Science & Business Media. [64](#)
- Perrin-Haynes, J., J. Chazal, C. Chantel et I. Falinower. 2012, «Les personnes âgées en institution», *Dossiers solidarité et santé, Direction de la Recherche, des Études, de l'Évaluation et des Statistiques*. [18](#)
- Poujaud, J., N. Noury et J.-E. Lundy. 2008, «Identification of inactivity behavior in smart home.», in *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2008, ISBN 9781424418152, ISSN 1557-170X, p. 2075–8, doi :10.1109/IEMBS.2008.4649601. [24](#)

- Puterman, M. L. 1994, *Markov Decision Processes : Discrete Stochastic Dynamic Programming*, *Wiley Series in Probability and Mathematical Statistics : Applied Probability and Statistics*, vol. 10, John Wiley & Sons, Inc., ISBN 0471619779, 672 p.. [36](#)
- Rabiner, L. 1989, «A tutorial on hidden Markov models and selected applications in speech recognition», *Proceedings of the IEEE*, vol. 77, p. 257–286. [37](#), [41](#), [42](#), [44](#), [45](#), [54](#), [85](#), [101](#), [VII](#), [XIV](#)
- Rammal, A. 2010, *Modélisation multi-agent dans un processus de gestion multi acteur, application au maintien à domicile*, Thèse de doctorat, Université Toulouse III. [25](#)
- Russell, S. et P. Norvig. 2009, *Artificial Intelligence : A Modern Approach*, 3rd edition, ISBN 0136042597, doi :10.1017/S0269888900007724. [34](#), [36](#), [48](#), [IV](#)
- Shimada, Y., T. Matsumoto et S. Kawaji. 2005, «Development of human daily behavior model and it's application», in *Proc. of the 16th IFAC World Congress*. [9](#), [24](#), [28](#)
- Silva, J. F. et S. Narayanan. 2004, «A statistical Discrimination Measure for Hidden Markov Models based on Divergence», in *INTERSPEECH*. [83](#)
- Solaimani, S., W. Keijzer-Broers et H. Bouwman. 2013, «What we do - and don't - know about the Smart Home : an analysis of the Smart Home literature», *Indoor and Built Environment*, doi :10.1177/1420326X13516350, ISSN 1420-326X. [8](#)
- Tan, X. et H. Xi. 2008, «Hidden semi-Markov model for anomaly detection», *Applied Mathematics and Computation*, vol. 205, n° 2, doi :10.1016/j.amc.2008.05.028, p. 562–567, ISSN 00963003. [17](#), [40](#)
- Thi, V., H. Hung et Q. Dinh. 2005, «Activity recognition and avnormality detection with the switching hidden semi-markov model», *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, p. 838—845. [10](#), [14](#), [41](#)
- Tiberghien, T., H. Aloulou, M. Mokhtari et J. Biswas. 2012, «Semantic Reasoning in Context-Aware Assistive Environments to Support Ageing with Dementia», in *11th International Semantic Web Conference*. [28](#)
- Tran, A., S. Marsland, J. Dietrich, H. Guesgen et P. Lyons. 2010, «Use cases for abnormal behaviour detection in smart homes», in *International Conference on Smart Homes and Health Telematics - ICOST*, p. 144–151. [18](#)
- Vacher, M., A. Fleury, N. Noury, F. Portet et J.-F. Serignat. 2009, «Complete Sound and Speech Recognition System for Health Smart Homes : Application to the Recognition of Activities of Daily Living», *Recent Advances In Biomedical Engineering*, p. 1–30. [17](#)

- Volant, S. 2010, «L'offre en établissements d'hébergement pour personnes âgées en 2011», *Études et Résultats, Direction de la recherche, des études, de l'évaluation et des statistiques (DREES)*, vol. 877. [2](#)
- Wang, B., M. Ye, X. Li, F. Zhao et J. Ding. 2011, «Abnormal crowd behavior detection using high-frequency and spatio-temporal features», *Machine Vision and Applications*, vol. 23, n° 3, doi :10.1007/s00138-011-0341-0, p. 501–511, ISSN 0932-8092. [17](#)
- Xiang, T., S. Gong et Q. Mary. 2005, «Video Behaviour Profiling and Abnormality Detection without Manual Labelling», . [8](#)
- Xie, L., V. A. Ugrinovskii et I. R. Petersen. 2005, «Probabilistic distances between finite-state finite-alphabet hidden Markov models», in *Automatic Control, IEEE Transactions on*, vol. 50, p. 505–511. [83](#)
- Yamahara, H., H. Takada, H. S et Hikawa. 2007, «An individual behavioral pattern to provide ubiquitous service in intelligent space», *WSEAS Transactions on* [9](#)
- Yamahara, H., H. Takada et H. Shimakawa. 2006, «A behavioral pattern adapted to individual for providing ubiquitous service in intelligent space», in *Proc. 5th WSEAS Intl. Conf. on*, p. 84–92. [38](#)
- Yu, S.-Z. 2010, «Hidden semi-Markov models», *Artificial Intelligence*, vol. 174, n° 2, doi : 10.1016/j.artint.2009.11.011, p. 215–243, ISSN 00043702. [39](#), [40](#), [42](#)
- Yu, S.-z. et H. Kobayashi. 2003, «An Efficient Forward–Backward Algorithm for an Explicit-Duration Hidden Markov Model», vol. 10, n° 1, p. 11–14. [53](#), [56](#)
- Yu, S.-Z., H. Kobayashi et L. Fellow. 2006, «Practical Implementation of an Efficient Forward – Backward Algorithm for an», *ieee transactions on signal processing*, vol. 54, n° 5, p. 1947–1951. [54](#), [IX](#)
- Yuan, B. et J. Herbert. 2013, «Hybrid Reasoning Framework for CARA Pervasive Healthcare», in *11th International Conference on Smart Homes and Health Telematics, ICOST 2013*, p. 126–133. [29](#)
- Zeng, J., J. Duan et C. Wu. 2010, «A new distance measure for hidden Markov models», *Expert Systems with Applications*, vol. 37, n° 2, doi :10.1016/j.eswa.2009.06.063, p. 1550–1555, ISSN 09574174. [85](#)
- Zhang, C. et L. Zhang. 2013, «Activity Recognition in Smart Homes Based on Second-Order Hidden Markov Model», vol. 7, n° 6, p. 237–244. [39](#)

Zhou, Q. P., L. Jung et K. C. Richards. 2012, «The management of sleep and circadian disturbance in patients with dementia.», *Current neurology and neuroscience reports*, vol. 12, n° 2, doi :10.1007/s11910-012-0249-8, p. 193–204, ISSN 1534-6293. [25](#), [49](#)

Modélisation du comportement habituel de la personne âgée dépendante en environnement incertain pour la détection d'évolutions et d'activités anormales

Résumé :

Des projections réalisées sur les perspectives démographiques et financières de la dépendance prévoient, en France, une nette augmentation de la population des plus de 80 ans, accompagnée d'une multiplication par 2 du nombre de personnes âgées dépendantes entre 2010 et 2060. Afin de gérer l'augmentation du nombre de personnes âgées dépendantes, les EHPAD (Etablissement d'Hébergement pour Personne Âgées Dépendantes) sont appelés à améliorer la prise en charge des résidents et à améliorer les conditions de travail du personnel soignant. C'est dans ce contexte, que nous avons développé un système de supervision permettant de détecter, via un ensemble de capteurs, des évolutions du comportement ou encore, le comportement anormal d'une personne âgée.

La détection des comportements anormaux dans le cadre de la supervision est un sujet de recherche qui a été largement étudié dans la littérature ; ce qui n'est tout de même pas le cas de l'analyse des variations des activités de la vie de tous les jours, prenant en compte les spécificités du comportement de la personne au cours du temps.

Ainsi, nous avons proposé un modèle de Markov, permettant d'apprendre, avec le moins d'*a priori* possible, le modèle de comportement habituel au sein de la chambre. Le modèle proposé a été testé sur des données acquises en Living Lab (GIS-Madonah^a). Nous avons également proposé une nouvelle approche pour calculer la distance entre deux modèles de Markov, afin d'évaluer l'évolution du comportement au cours du temps. Ces méthodes devront permettre, non seulement de déterminer la probabilité du comportement actuel de la personne par rapport à son comportement habituel ; mais également, de détecter des évolutions lentes du comportement de la personne.

Mots clés : Modèle de Markov, Détection d'évolution, Environnement incertain, Comportement humain, Comportement inhabituel, Smart homes

^a. GISMadonah : <http://www.bourges.univ-orleans.fr/madonah/index.php>

environment for the detection of evolutions and abnormal activities

Abstract :

Due to demographic changes, it is expected that the number of French having over 80 years will increase drastically and the number of dependent elderly people will grow twice between 2010 and 2060. To manage this increasing number of dependent elderly person, nursing homes are required to improve the care of residents and to improve the working conditions of health workers.

In this context, we plan to develop a monitoring system, based on a set of sensors, to detect modifications in the behavior of a person, and unusual behavior.

Detection of abnormal activities in smart homes is an important topic of research, unlike the detection of the evolutions of behavior, which take into account the specifics activities of the person in time.

Thus, we proposed a Markov model which allow to learn the usual behavior in the room, with a reduced number of *a priori*. The model is try on data acquired on a Living Lab (GIS Madonah). We proposed a new method to compute the distance between two markov models, to estimate the evolution of the behavior. These methods allow to compute the probability of the current activities with the usual behavior, and the slow evolutions of the behavior.

Keywords : Markov models, Detection of evolution, Uncertain, Human behavior, Unusual behavior, Smart homes