



Apprentissage d'arbre de décision pour le pilotage en ligne d'algorithmes de détection sur les électrocardiogrammes

François Portet, René Quiniou, Marie-Odile Cordier, Guy Carrault

► To cite this version:

François Portet, René Quiniou, Marie-Odile Cordier, Guy Carrault. Apprentissage d'arbre de décision pour le pilotage en ligne d'algorithmes de détection sur les électrocardiogrammes. 16e conférence Reconnaissance des Formes et Intelligence Artificielle (RFIA'08), Jan 2008, Amiens, France. inria-00266535

HAL Id: inria-00266535

<https://inria.hal.science/inria-00266535>

Submitted on 24 Mar 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Apprentissage d'arbre de décision pour le pilotage en ligne d'algorithmes de détection sur les électrocardiogrammes

Decision Tree Learning for On-Line Selection of Detection Algorithms in Electrocardiogram

François Portet¹ René Quiniou² Marie-Odile Cordier² Guy Carrault³

¹ Department of Computing Science, University of Aberdeen, Aberdeen AB24 3UE, UK

² IRISA, INRIA, Université de Rennes 1, Campus de Beaulieu, 35042 Rennes, France

³ LTSI, Université de Rennes 1, Campus de Beaulieu, 35042 Rennes, France

fportet@abdn.ac.uk, {quiniou, cordier}@irisa.fr, guy.carrault@univ-rennes1.fr

Résumé

Le nombre d'algorithmes de traitement du signal (compression, reconnaissance des formes, etc.) grandit progressivement ce qui rend de plus en plus difficile le choix de l'algorithme le plus adapté à une tâche particulière. Ceci est particulièrement vrai pour l'analyse automatique des électrocardiogrammes (ECG) notamment pour la détection des complexes QRS. Bien que chaque algorithme de la littérature se comporte de manière satisfaisante dans des situations normales, il existe des contextes où un algorithme est plus adapté que les autres, notamment en présence de bruit. Nous proposons une méthode de sélection qui choisit, en ligne, l'algorithme le plus adapté au contexte courant du signal à traiter. Les règles de sélection sont acquises par arbre de décision sur les résultats de performance de 7 algorithmes testés dans 130 contextes différents. Les résultats montrent la supériorité de l'approche proposée sur les algorithmes utilisés séparément. En outre, les performances des règles de sélection apprises sont très proches de celles des règles acquises par expertise, ce qui conforte notre approche.

Mots Clefs

Apprentissage automatique, arbre de décision, sélection d'algorithmes, monitoring cardiaque.

Abstract

As the number of signal processing algorithms (compression, pattern recognition, etc.) is continually increasing, it becomes more and more difficult to choose the most adapted to a particular task. This is particularly true for automatic QRS wave detection in electrocardiogram (ECG) analysis. Though each algorithm proposed in the literature behaves satisfactorily in normal situations, there are contexts where one algorithm performs better than the others, e.g. in the presence of noise. We propose a general

combination method that selects, on line, the algorithm that is the most adapted to the current context of the signal to process. The selection rules are acquired by a decision tree learning method from the performance measures of 7 algorithms in various instances of 130 contexts. The results show the superiority of the approach over the algorithms used separately. Furthermore, the performance of learned selection rules are quite close to those of expert rules supporting a machine learning approach.

Keywords

Machine Learning, decision tree, algorithm selection, cardiac monitoring.

1 Introduction

Les outils de traitement du signal sont intensément utilisés en médecine, dans l'industrie, la vidéosurveillance, etc. Le nombre de méthodes de traitement du signal pour exécuter certaines tâches (détection d'événements, filtrage, compression, identification de motifs,...) augmente constamment. Ceci est particulièrement vrai pour l'analyse automatique de l'électrocardiogramme (ECG), pour laquelle un très grand nombre d'algorithmes ont été proposés pour réaliser la détection d'une onde très importante : *le complexe QRS*. Choisir un algorithme pour une tâche spécifique est souvent un problème. En effet, même si certains algorithmes démontrent de bonnes performances globales, chaque algorithme peut échouer dans certains cas particuliers.

Afin de résoudre ce problème nous proposons de combiner plusieurs algorithmes pour réaliser la tâche de détection du complexe QRS dans les ECG. Nous avons constaté que, dans la plupart des domaines, les performances des algorithmes de traitement du signal sont influencées par le *contexte* du signal à traiter tel que le niveau de bruit ou le type d'information transporté par le signal. En effet, un signal est souvent bruité (surtout en

milieu clinique). En outre, le fait que le type, la quantité ou la qualité des informations contenues dans le signal varient peut perturber l'algorithme [1]. En se reposant sur une analyse du contexte du signal, l'algorithme qui est le plus approprié pour la situation peut être choisi plus convenablement. Dans notre approche, nous proposons d'extraire, par apprentissage d'arbre de décision, les règles de choix de 7 algorithmes de détection de QRS selon diverses situations, appelées *contextes*, qui représentent 130 combinaisons de signal d'ECG et de bruit clinique.

L'approche présente des similitudes avec le méta-apprentissage [2] ou les « ensembles de classificateurs » [3-5]. Cependant, elle en diffère sur certains points importants. Tout d'abord, les algorithmes ne sont pas des classificateurs. Bien que la détection puisse être vue comme un problème de classification, les méthodes de détection impliquées n'appartiennent pas toutes au domaine de l'apprentissage artificiel. De plus, l'approche ne consiste pas à combiner les sorties des algorithmes mais à choisir, en ligne, le meilleur algorithme parmi un ensemble d'algorithmes selon une évaluation du contexte courant. En effet, les sorties des algorithmes sont considérées comme qualitatives et ne peuvent pas être fusionnées facilement. De même, l'approche par vote n'est pas considérée car le vote implique l'exécution simultanée de la totalité des algorithmes pour produire toutes les sorties. Ceci n'est pas acceptable dans un système temps réel avec des ressources limitées tel qu'un système de monitoring cardiaque.

Le document est organisé comme suit. Le domaine du monitoring cardiaque et le problème du choix d'algorithme sont présentés dans la section 2. La méthode d'apprentissage est décrite dans la section 3. Puis, l'application à l'analyse d'ECG est présentée dans la section 4. La section 5 détaille une expérience dans laquelle un système de monitoring cardiaque utilise de telles règles de décision. En section 6, nous discutons l'approche et terminons par une brève conclusion.

2 Monitoring cardiaque

Les systèmes de monitoring cardiaque sont employés en routine dans les unités de soins intensifs (USI), pour prévenir et surveiller les pathologies cardiaques. Les arythmies, dont certaines sont les pathologies mortelles les plus fréquentes, sont particulièrement surveillées. Ces systèmes s'appuient sur l'analyse précise de la modalité non invasive la plus efficace pour surveiller l'activité cardiaque : l'électrocardiogramme (ECG).

2.1 Analyse de l'électrocardiogramme

Un électrocardiogramme est un enregistrement de l'activité électrique cardiaque. Cette activité électrique produit les contractions cardiaques à l'origine de la circulation du sang dans le corps. Le signal correspondant (appelé ECG) se compose d'ondes successives, à savoir l'onde P (liée à l'activité des oreillettes du cœur), le

complexe QRS et l'onde T (liées à l'activité ventriculaire). Comme indiqué sur la figure 1, dans un rythme sinusal (rythme normal), l'ordre normal des ondes est P - QRS - T répété régulièrement. Mais en présence d'une arythmie, telle qu'un rythme bigéminisé (deuxième partie du premier ECG de la figure 1), l'ordre et la forme des ondes diffèrent. De plus, comme dans tous problèmes de transmission de signal, de nombreux types de bruit viennent couramment corrompre le signal (mauvais contact d'une électrode sur la peau, bruit ambiant, etc.). La figure 1 montre un ECG corrompu par du bruit musculaire (activité électrique des muscles lorsque le patient bouge) à différents niveaux de rapport signal-à-bruit (RSB).

Le traitement automatique de l'ECG doit ainsi faire face à tous ces problèmes. La tâche principale en analyse de l'ECG est la détection du complexe QRS car il reflète l'activité ventriculaire du cœur (les chambres qui expulsent le sang dans le corps). Des informations majeures peuvent être dérivées de cette détection telles que la fréquence cardiaque et la forme d'onde du QRS.

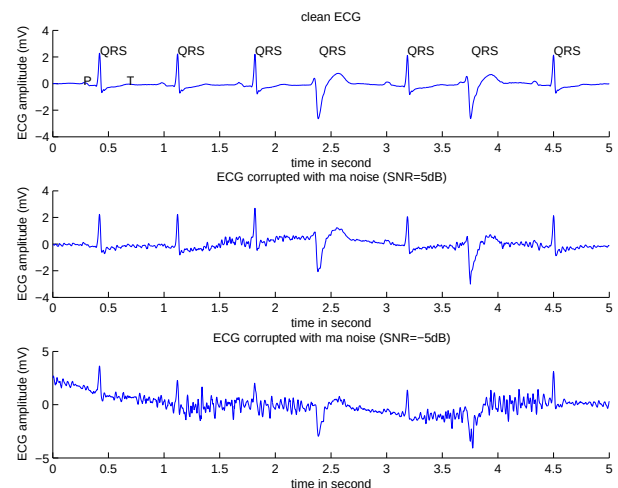


Fig. 1. Haut: ECG avec transition d'un rythme normal (0-2s) à un rythme bigéminisé (2-5s). Milieu: même ECG bruité avec bruit musculaire à RSB=5dB. bas: RSB=-5dB.

2.2 Traitement en ligne

En raison de son caractère d'urgence, le monitoring cardiaque exige le traitement des signaux en ligne. Pour certains programmes d'application temps réel, le signal d'entrée est traité buffer par buffer (c'est-à-dire segment par segment) afin de fournir des sorties aussitôt que possible. La taille du buffer peut être aussi grande que le signal, dans ce cas il s'agit d'une situation hors-ligne, ou aussi petite qu'un simple échantillon et dans ce cas le traitement correspond à des contraintes temps réel dures. Un exemple de système avec buffer est donné figure 2. Ce système avec buffer est celui adopté par le système de monitoring cardiaque IP-Calicot [6] pour l'identification en ligne des arythmies cardiaques. Le premier buffer recouvrant la toute première partie du signal d'entrée est

traité, puis le deuxième buffer, qui recouvre une partie du buffer précédent, est traité et ainsi de suite. Un algorithme doit être choisi pour traiter chaque buffer. En général, le même algorithme est employé pour traiter tous les buffers. Dans notre approche nous proposons de choisir le meilleur algorithme pour chaque buffer.

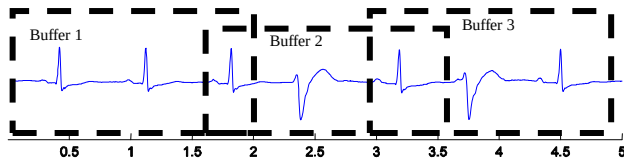


Fig. 2. Méthode de buffering avec recouvrement partiel dans le système de monitoring cardiaque IP-Calicot.

2.3 Détection du QRS

Beaucoup d'algorithmes de détection de QRS en temps réel ou en ligne ont été proposés ces 30 dernières années [7-9]. Ils utilisent diverses techniques : 1) filtres numériques ; 2) analyse temps fréquence et ondelettes [10; 11] ; 3) transformée linéaire et non linéaire [7; 12; 13] 4) analyse statistique et réseaux de neurones [14]; 5) grammaires de modèles [15] ; 6) programmation génétique [16] ; 7) fusion multi source [17]; etc. Depuis les années 90, les performances s'améliorent très lentement dans des situations non bruitées. Par exemple, le détecteur de Pan et Tompkins (1985) [12] avec un taux d'erreur (TE) de 0.675% est légèrement moins bon que celui de Christov (2004) [9] avec un TE de 0.435% dans des situations non bruitées. Cependant, si chaque algorithme donne satisfaction de manière générale, chacun présente des situations où il échoue. En effet, en raison de la grande différence entre les techniques employées, chaque détecteur de QRS réagit différemment aux différentes formes d'ondes et aux différents bruits.

La méthode proposée ici pour tirer parti en ligne des différents algorithmes se fonde sur une définition précise du contexte et sur une méthodologie d'acquisition des règles de sélection. L'approche est générale et pourrait convenir à d'autres systèmes de monitoring [18] (par exemple : les autres modules du monitoring cardiaque) ainsi qu'aux applications de traitement des signaux [19] (détection, transmission, compression, etc.), ou encore à d'autres domaines où plusieurs algorithmes peuvent être utilisés pour réaliser une tâche donnée [20].

3 Méthodologie pour l'acquisition de règles de sélection d'algorithmes

Dans une expérience précédente [1], nous avons décrit une méthode d'acquisition de règles expertes pour la sélection d'algorithme en fonction du contexte du signal. Cette expérience a souligné la complexité de la tâche (trouver un expert, interpréter la grande masse de données, etc.) et la nécessité d'une acquisition de connaissance automatisée. La section 3.1 discute les approches existantes pour combiner les détecteurs de

QRS. La section 3.2 détaille la définition de contexte pour chaque segment de signal à traiter. Enfin, la section 3.3 décrit la méthode d'apprentissage automatique utilisée pour acquérir les règles de sélection.

3.1 Combinaison de détecteurs de QRS

Quelques travaux utilisant la combinaison de détecteurs de QRS existent dans la littérature [7; 21; 22]. Dans la méthode de Moraes et collègues [21], lorsque le détecteur de QRS principal donne des résultats douteux, les sorties d'un deuxième algorithme sont utilisées. Dans l'approche de Meyer et collègues [7], lorsque les sorties du détecteur de QRS principal contredisent les sorties d'un deuxième algorithme, le détecteur principal est réexécuté après ajustement des paramètres. La méthode la plus proche de celle proposée dans cet article est celle de Soulas et collègues [22] qui choisissent un détecteur de QRS parmi une paire de détecteurs selon une analyse de l'énergie en sous-bandes du signal d'entrée (analyse de contexte). Comme remarqué dans l'introduction, aucune de ces approches ne fusionne les sorties des détecteurs en s'appuyant sur leur aspect qualitatif. L'algorithme à employer est choisi après une analyse, soit du signal d'entrée, soit de la qualité des résultats. On peut remarquer que ces études sont limitées à seulement deux détecteurs et n'ont pas été évaluées dans des situations difficiles.

Dans ces travaux, le choix est guidé par l'évaluation d'un « contexte », dont le contenu précis varie d'une approche à l'autre. Cependant, les règles de décision utilisées et les définitions des contextes ne sont guère généralisables et ne sont pas explicites. Ceci rend difficile leur réutilisation. De plus, dans certaines méthodes la constitution des contextes est basée sur l'expertise. Une telle méthode d'acquisition devient rapidement humainement ingérable lorsqu'un grand ensemble de contextes est considéré.

C'est pourquoi nous proposons une approche de sélection des meilleurs algorithmes de détection du QRS tenant compte du contexte, mais dont les règles de sélection sont explicites et acquises automatiquement. La méthode choisie pour l'apprentissage est l'induction d'arbre de décision pour des raisons d'interprétabilité humaine et de reproductibilité. De plus, l'arbre de décision semble bien adapté au traitement des données qualitatives et quantitatives. L'induction d'arbre de décision a été appliquée pour obtenir les règles de choix pour 7 détecteurs de QRS dans diverses situations, représentant 130 combinaisons d'arythmies et de bruits cliniques.

3.2 Définition du contexte

Un système de monitoring analyse le signal d'entrée (cf. 2.2) buffer par buffer afin de satisfaire les contraintes d'une exécution en temps réel. L'équation (1) fournit une définition simplifiée d'un signal bufferisé évoluant dans le temps.

$$Y(t) = S(t) + N(t) \quad (1)$$

où $Y(t)$ est un signal k -dimensionnel (par exemple une sortie de capteur) qui évolue avec le temps t . $Y(t)$ se compose de l'information utile $S(t)$ (par exemple l'information numérique) et du bruit ambiant $N(t)$ (par exemple interférence avec d'autres sources, déformation par la température ambiante, etc.). Le contexte est défini par une évaluation du type des informations transportées par S et une évaluation du bruit N . Nous insistons sur le terme « évaluation » car la séparation de S et de N est loin d'être évidente dans beaucoup d'applications de traitement du signal.

3.3 Acquisition de règles de sélection

Parmi les méthodes d'apprentissage utilisables pour apprendre des règles de sélection, l'apprentissage d'arbre de décision présente plusieurs avantages dans le cadre de notre application : (1) vis-à-vis de la masse des données présentes dans les problèmes de traitement du signal, l'apprentissage d'arbre de décision nécessite peu de préparation des données; (2) les contextes peuvent se composer de données numériques et catégoriques qui sont facilement manipulées par l'algorithme d'apprentissage d'arbre de décision ; (3) les règles dérivées des arbres de décision sont explicites et vérifiables par un expert ; (4) du fait leur intelligibilité, les règles peuvent servir à améliorer la connaissance des algorithmes et de leurs contextes d'utilisation ; enfin, (5) l'arbre de décision induit peut traiter une grande quantité de données en peu de temps, ce qui est requis dans des applications de traitement du signal en temps réel.

L'arbre de décision est induit à partir d'un ensemble d'apprentissage (TS – training set) composé d'exemples décrits par un ensemble d'attributs caractéristiques du contexte d'un segment de signal. Ce contexte est composé du type d'information r et du bruit n . La classe d'un tel exemple est la décision, c'est-à-dire l'algorithme le plus adapté devant être appliqué au contexte associé. L'algorithme 1 décrit la génération de l'ensemble d'apprentissage. Cet ensemble est réalisé 1) en produisant tous les contextes possibles ; 2) en exécutant chaque algorithme dans chaque contexte et 3) en décidant quel algorithme est le meilleur dans chaque contexte. Chacune de ces étapes est détaillée dans la section suivante.

4 Apprentissage de règles de sélection pour la détection de QRS

Cette section décrit l'application de la méthode au problème de détection de QRS. La section 4.1 détaille le choix retenu pour la définition de contexte et la section 4.2 décrit la méthode employée pour estimer le contexte en ligne. La section 4.3 présente l'ensemble d'algorithmes et la méthode d'évaluation. La section 4.4 décrit la composition de l'ensemble d'apprentissage. Enfin, les résultats d'apprentissage sont analysés en section 4.5.

Pour chaque $r \in R$ faire /* R est le domaine des différents types d'information */

Pour chaque $n \in N$ faire /* N est le domaine du bruit */

pour $t=1$ à T faire /* T est le nombre de réalisations (essais) */

1. échantillonnage : Tirer aléatoirement un segment ECG S de Q QRS contenant l'information r . /*prendre une période d'ECG de rythme r contenant Q QRS*/

2. addition de bruit : Modifier S en ajoutant le bruit n pour obtenir Y .

3. filtrage : Filtrer Y pour obtenir le signal Y^* .

4. évaluation de performance:

Pour chaque algorithme $d \in D$ **faire** /* D domaine des algorithmes */

exécuter d sur l'entrée Y^*

calculer le score de d

Fin pour

sélectionner l'algorithme d_{opt} ayant le score le plus élevé.

5. construction de l'ensemble d'apprentissage : Ajouter à l'ensemble d'apprentissage TS , l'exemple E défini par $r \times n \times d_{opt}$.

Fin pour

Fin pour

Fin pour

Algorithme 1. Construction de l'ensemble d'apprentissage

4.1 Définition du contexte de l'ECG

Suivant l'équation (1) nous définissons l'information utile S comme étant les différents rythmes cardiaques possibles et le bruit N comme les différents bruits qui peuvent être générés par l'environnement clinique. Les contextes de rythmes ont été définis d'après la base de données d'arythmies MIT-BIH [23]. Cette base de données se compose de 48 ECGs contenant 13 types d'arythmies et 4 types de rythmes sinusaux (normaux). La base de données entière dure 24 heures et contient 109.494 QRS annotés par des médecins. Le tableau 1 donne les détails des 10 rythmes que nous considérons dans la suite de l'étude pour composer le contexte de rythme.

Le bruit additif clinique a été défini selon la base de données *MIT-BIH Noise Stress Test Database* [24]. Cette base de données contient trois enregistrements de bruit, de 30 minutes chacun, enregistrés sur des volontaires actifs. Trois types d'enregistrements ont été extraits en choisissant les intervalles contenant principalement des ondulations de la ligne de base (bw), des artefacts dus à la commande musculaire du patient (ma), et des artefacts dus aux mouvements d'électrode (em). Nous avons utilisé ces bruits selon quatre Rapports Signal-à-Bruit (RSB) : 5, 0, -5 et -15dB. En conséquence $N = (\{ma, bw, em\} \times RSB) \cup \{no_noise\}$ représente le domaine du contexte de bruit et

contient 13 valeurs différentes dénotant 12 situations bruitées plus une situation sans bruit (*no_noise*). La figure 1 montre un ECG corrompu avec le bruit *ma* à RSB=5dB et à -5dB.

Symbole	Rythme	Type	Type de QRS
N	Normal rhythm	normal	normal
NL	Normal rhythm with lbbb	normal	lbbb
NR	Normal rhythm with rbbb	normal	rbbb
P	Paced rhythm	normal	paced
B	Bigeminy	arrhythmia	normal and ventricular
T	Trigeminy	arrhythmia	normal and ventricular
SBR	Sinus bradycardia	arrhythmia	normal
BII	Mobitz type II	arrhythmia	one QRS out of two missing
SVTA	Supraventricular tachycardia	arrhythmia	supraventricular
VT	Ventricular tachycardia	arrhythmia	ventricular

Table 1. Contextes de rythme considérés dans l'étude

4.2 Estimation du contexte en ligne

L'estimation du contexte en ligne est une tâche difficile et nous n'avons pas la place pour une étude approfondie dans cet article. Nous avons utilisé deux sources d'information pour l'estimation du contexte. Il s'agit, pour chaque buffer traité :

- 1- des annotations du rythme cardiaque *-r-* provenant des cardiologues et des annotations du type et niveau de bruit *-n-* provenant de la génération des signaux (et donc connu) ; et
- 2- d'une estimation en ligne les valeurs efficaces du signal (*RMS-Root Mean Square*) dans des sous-bandes de basse, moyenne et haute fréquences. On obtient ainsi, pour chaque buffer, les valeurs *sl*, *sm*, *sh* (*sl* = *signal low*, *sm* = *signal medium*, *sh* = *signal high*).

La première source d'information donne les valeurs des « vrais » contextes (vrais rythmes cardiaques et vraies valeurs du bruit ajouté). Cette information est obtenue hors-ligne (difficile à obtenir en ligne) et est de type qualitatif. Cette source va cependant nous permettre d'estimer la qualité du pilotage indépendamment du problème d'estimation du contexte.

La deuxième source d'information est, quant à elle, obtenue en ligne (au cours du traitement) et est de type quantitatif. Elle est utilisée comme heuristique pour évaluer le contexte de bruit et, dans une moindre mesure, le contexte de rythme (qui influence aussi le *RMS*). La valeur efficace est une heuristique courante utilisée pour l'évaluation du niveau du bruit [11]. Cette source va nous permettre d'estimer le contexte en complétant (ou remplaçant) la première source (qualitative) par une source plus précise (quantitative).

4.3 Evaluation des algorithmes

Sept détecteurs de QRS ont été choisis selon les critères suivants : capacité à travailler en temps réel, diversité des techniques, facilité d'exécution, et robustesse au bruit. Nous avons choisi ceux proposés par Benitez et collègues [11] (*benitez*), Gritzali [13] (*gritzali*), Suppappola et Sun [25] (*mobd*), Kadambe et collègues [10] (*kadambe*), Pan et Tompkins [12] (*pan*), plus deux détecteurs présentés comme *af2* et *df2* dans Freisen et collègues [26]. Ces deux derniers détecteurs sont sensibles à des types de bruits différents et sont, en ce sens, complémentaires.

Les sorties des détecteurs de QRS sont évaluées en les comparant aux annotations des médecins marquant les dates d'occurrence des QRS. Ainsi trois valeurs standards sont calculées: Vrais Positifs (VP - détections correctes), faux négatifs (FN - détections manquées) et faux positifs (FP - fausses alarmes). Ces valeurs sont ensuite utilisées pour calculer la sensibilité $Se=VP/(VP+FN)$, la Prédicativité Positive $PP=VP/(VP+FP)$ et la F-Mesure, $FM=2 \times PP \times Se / (PP+Se)$ (en supposant $\beta=1$).

4.4 Construction de l'ensemble d'apprentissage

Pour composer l'ensemble d'apprentissage nous avons utilisé l'algorithme 1 avec les paramètres suivants : $|N|=13$ types de bruit, $|R|=10$ types de rythme (donnant 130 types de contexte), $T=200$ réalisations de chaque type de contexte (donnant 26.000 réalisations de contextes), et $|D|=7$ détecteurs (donnant 182.000 calculs). La taille d'un fragment d'ECG a été fixée à $B=10$ battements (QRS). En tout, 9.500 QRS ont été utilisés représentant 8.70% de la base de données entière.

Selon le principe évoqué dans la section 3.2, chaque exemple de l'ensemble d'apprentissage est décrit par un 6-uplet $r \times n \times sl \times sm \times sh \times d$, où $r \in R$ (types de rythme), $n \in N$ (types de bruit), $sl, sm, sh \in \mathbf{R}$, $d \in D$ (ensemble d'algorithmes). La classe d de chaque exemple correspond au détecteur de QRS ayant obtenu la valeur de FM la plus élevée pour le contexte correspondant à l'exemple. Pour étudier différentes manières d'utiliser le contexte, quatre arbres de décision ont été appris :

- 1- DT1, qui emploie tous les attributs
- 2- DT2, qui emploie $r \times sl \times sm \times sh \times d$ (le bruit est décrit uniquement par le RMS des sous-bandes) ;
- 3- DT3, qui utilise $sl \times sm \times sh \times d$ (le contexte est décrit uniquement par le RMS des sous-bandes) ; et
- 4- DT4, qui utilise $r \times n \times d$.

4.5 Résultats d'apprentissage

L'algorithme J48 de la boîte à outils Weka [27] a été utilisé pour apprendre les arbres de décision. J48 est une mise en œuvre de l'algorithme C4.5 de Quinlan [28]. Les paramètres ont été fixés à 20 exemples au minimum par feuille, et à un facteur de confiance de 0,25 avec élagage. L'apprentissage des arbres a été effectué avec une validation croisée à 10 tours. Le tableau 2 donne les résultats de l'apprentissage.

Modèle	Attributs du contexte		Arbre appris		Validation croisée classifications correctes
	rythme	bruit	nb feuilles	taille	
DT1	r	n×sl×sm×sh	277	503	16601 (63,85%)
DT2	r	sl×sm×sh	235	461	16615 (63,90%)
DT3		sl×sm×sh	224	447	15702 (60,39%)
DT4	r	n	36	45	15850 (60,96%)

Table 2. Résultats de l'apprentissage

DT1 donne un résultat très moyen de 63.85% de classifications correctes mais avec l'arbre de plus grande taille tandis que DT2 réalise la même performance (63,90%) mais avec un arbre de taille nettement plus petite (meilleure généralisation). Ce résultat montre que l'attribut *n* contient des informations redondantes avec le triplet *sl×sm×sh* (DT1 contre DT2). C'est un résultat intéressant car il est beaucoup plus facile de calculer la valeur efficace dans trois sous-bandes que le type exact du bruit et du RSB réel. Pour DT3 les performances descendent à 60.39%. Ceci suggère que le contexte de rythme *r* est important. DT4, qui n'emploie pas l'information de sous-bande, mène à un arbre très petit (45) mais avec des performances plus faibles (61,0%). DT4 a également montré de faibles performances lors d'une expérience précédente [6].

5 Expérimentations et comparaison avec des règles expertes

Pour évaluer les performances du système de pilotage dans des situations réalistes (c'est-à-dire pour le traitement d'ECG en ligne), les arbres de décision ont été chargés dans le système de monitoring cardiaque IP-Calicot [6] qui permet de traiter des segments d'ECG selon une analyse du contexte. Les règles expertes acquises d'après la méthode décrite dans Portet et collègues [1] ont été testées sur l'ensemble des données de la présente étude, afin de les comparer à l'approche utilisant l'apprentissage automatique par arbre de décision. Par ailleurs, afin d'évaluer les performances maximales atteignables par le jeu de détecteurs de QRS, un test supplémentaire dénommé *idealSelection* a été effectué. Celui-ci consiste à exécuter tous les algorithmes sur chaque segment et à ne retenir que les sorties de l'algorithme le plus performant (selon FM). Ainsi *idealSelection* fournit les performances maximales atteignables et est utilisé comme étalon par la suite.

Un ensemble de test constitué de 11 enregistrements d'ECG totalisant 5 heures et 30 minutes et comprenant 10 contextes de rythmes différents a été construit à partir de la base de données d'arythmies MIT-BIH. Le bruit clinique (*bw*, *ma*, *em* à 4 RSB : 5, 0, -5 et -15dB) a été ajouté aléatoirement à l'ECG afin de contrôler le RSB (l'évaluation du bruit à partir d'ECG déjà bruité est toujours un problème non résolu dans l'analyse d'ECG). Le bruit clinique a été ajouté avec les probabilités suivantes : $P(\text{no_noise})=P(\text{bw})=P(\text{ma})=P(\text{em})=1/4$.

Excepté pour le contexte *no_noise*, le niveau de bruit respecte la distribution suivante : $P(5\text{dB})=1/2$, $P(-5\text{dB})=1/3$, et $P(-15\text{dB})=1/6$ pour reproduire des situations courantes dans les ECG cliniques.

Le tableau 3 synthétise les résultats pour tous les algorithmes, arbres de décision et règles expertes.

Détecteurs	Se ^{±σ} (%)	PP ^{±σ} (%)	FM ^{±σ} (%)	nombre de commutations
<i>af2</i>	91,16 ^{±3,94}	77,86 ^{±8,92}	83,99 ^{±6,54}	-
<i>Benitez</i>	94,77 ^{±2,39}	81,67 ^{±4,66}	87,74 ^{±3,45}	-
<i>df2</i>	73,26 ^{±18,58}	84,68 ^{±12,44}	78,56 ^{±15,80}	-
<i>Gritzali</i>	85,68 ^{±5,11}	71,36 ^{±5,37}	77,87 ^{±4,47}	-
<i>Kadambe</i>	92,79 ^{±3,08}	86,13 ^{±5,03}	89,34 ^{±3,61}	-
<i>Mobd</i>	95,29 ^{±3,11}	64,68 ^{±5,11}	77,05 ^{±4,10}	-
<i>Pan</i>	80,38 ^{±8,79}	85,97 ^{±5,75}	83,08 ^{±6,46}	-
<i>Expert rules</i>	93,72 ^{±2,62}	86,75 ^{±4,12}	90,10 ^{±3,05}	623
<i>DT1</i>	92,87 ^{±3,16}	86,36 ^{±4,63}	89,50 ^{±3,59}	459
<i>DT2</i>	93,35 ^{±2,80}	86,00 ^{±4,62}	89,52 ^{±3,36}	1276
<i>DT3</i>	93,05 ^{±3,11}	84,60 ^{±4,78}	88,62 ^{±3,53}	1407
<i>idealSelection</i>	93,24 ^{±2,92}	91,32 ^{±3,26}	92,27 ^{±2,96}	1484

Table 3. Résultats obtenus sur l'ensemble de test

Selon la F-Mesure (FM), les règles expertes surpassent toutes les autres méthodes (*idealSelection* est employé comme étalon). Le meilleur algorithme est *kadambe* avec FM=89,34%. Cependant, *benitez* montre une sensibilité supérieure Se=94,77% au *kadambe*. Ces deux algorithmes surpassent les autres avec une FM supérieure de 3.75%. Parmi les méthodes utilisant la sélection d'algorithmes, les règles expertes présentent les meilleurs résultats suivies de DT2 et de DT1 qui eux-mêmes présentent de meilleures performances que l'algorithme *kadambe*. Seul DT3 présente des résultats non satisfaisants.

Les règles expertes obtiennent l'écart type le plus faible pour FM (3,05%) suivi de DT2 ($\sigma=3,36\%$) tous deux inférieurs à l'écart type de l'algorithme *benitez* ($\sigma=3,45\%$). Ceci montre que les méthodes de sélection obtiennent des résultats plus stables que les algorithmes seuls.

idealSelection donne la limite supérieure pour FM=92,27%. Les règles expertes contribuent à remplir 25,93% de l'écart entre *kadambe*, le meilleur algorithme, et *idealSelection*. Ceci montre que la stratégie de sélection d'algorithme pourrait être améliorée avec des règles plus précises. Le nombre de commutations d'algorithmes est également un paramètre intéressant. *idealSelection* montre que les meilleures performances nécessitent 1484 commutations. DT2 est près de ce nombre (1276). Cependant, les règles expertes et DT1 atteignent un bon score avec bien moins de commutations (623 et 459). Ceci suggère que certains mauvais choix (par exemple la sélection de *benitez* au lieu de *kadambe*) effectués par les règles expertes et le DT1 ont des effets limités sur la détection.

6 Discussion

Les résultats révèlent une légère supériorité des règles expertes sur les arbres de décision. Les règles apprises par induction d'arbres de décision sont différentes des règles expertes. La supériorité des règles expertes peut être attribuée au fait que l'expert peut généraliser des résultats utiles et écarter les données douteuses. L'expert a également l'occasion de généraliser une règle en prenant en considération le coût du choix d'un algorithme plutôt qu'un autre. En effet, le coût du choix de *benitez* plutôt que *kadambe* (petite baisse de performance) n'est pas le même que le coût du choix de *mobd* plutôt que *kadambe* (importante baisse de performance). Ainsi, un arbre de décision tenant compte du coût de transition entre deux algorithmes devrait améliorer les performances. L'algorithme d'apprentissage d'arbre de décision avec « ranking » proposé par Todorovski et collègues [3] pourrait être utilisé. L'approche par vote pourrait également être considérée, non pas pour produire les sorties de tous les détecteurs, car la complexité du calcul reste critique dans un environnement aux ressources limitées, mais pour produire les sorties des k meilleurs algorithmes dans le contexte courant ($k < |D|$ dépendant des ressources disponibles) et appliquer un vote.

Une autre approche envisageable serait une approche Bayésienne. L'apprentissage des probabilités de succès de chaque algorithme sachant le contexte permettrait de représenter l'incertitude sur l'estimation du contexte et sur le choix d'un algorithme plutôt qu'un autre. On pourrait ainsi exécuter uniquement les algorithmes qui ont le plus de chance de réussir dans le contexte le plus probable. Ainsi, si $P(kadambe|contexte)=0,8$, $P(benitez|contexte)=0,3$ et $P(gritzali|contexte)=0,78$, alors, selon les ressources disponibles, *kadambe* et *gritzali* pourraient être exécutés et leurs résultats fusionnés.

L'approche par arbre de décision a aussi montré que l'information contextuelle appropriée se fonde non seulement sur le niveau de bruit, qui est le facteur le plus évident, mais également sur le rythme et le type de bruit qui sont souvent considérés comme moins perturbateurs. L'étude a également montré que l'analyse de trois sous-bandes de fréquences améliore davantage les résultats que l'utilisation simple du RSB et du type de bruit. Ce résultat est important pour concevoir l'analyseur de contexte pour le système de pilotage en ligne.

L'expérience entreprise avec des algorithmes de détection de QRS a montré qu'il reste des choses à améliorer. Selon *idealSelection*, la FM maximum atteignable est 92,27%. Les règles expertes atteignent 90,10%, améliorant le meilleur algorithme (*kadambe*) de 0,76% dans des contextes bruités. C'est une très bonne amélioration si l'on considère les algorithmes actuels décrits dans la littérature qui montrent une amélioration de sensibilité en général de moins de 1% [8; 9] même dans des situations non bruitées.

La méthode développée dans cet article peut s'appliquer à des systèmes auto-adaptatifs [20] pour lesquels on dispose de plusieurs algorithmes pour effectuer une même tâche

ou résoudre un problème donné. C'est notamment le cas dans certaines applications d'analyse d'images [19] ou de monitoring de systèmes industriels [18]. L'analyse des caractéristiques de tels algorithmes (complexité, précision...) apporte des informations importantes pour l'écriture des règles de sélection (pilotage) mais qu'il est souhaitable de compléter par des connaissances pragmatiques sur leur comportement à l'exécution. Notre approche consiste à caractériser les contextes d'exécution et d'évaluer les performances, en pratique, des algorithmes dans ces contextes. Des règles de sélection acquises de manière automatique permettent ensuite au système de s'adapter à la situation courante. L'inconvénient de la méthode provient essentiellement du fait qu'il faut envisager tous les contextes possibles. Cependant, l'utilisation de l'apprentissage automatique autorise une mise à jour aisée de l'ensemble des règles de décision lorsqu'un nouveau type de contexte est introduit ou lorsqu'il est nécessaire d'affiner les éléments définissant un contexte, par l'ajout de nouveaux attributs, par exemple.

7 Conclusion

Cet article a présenté une méthode d'acquisition automatique des règles de sélection pour l'adaptation en ligne d'un système de monitoring cardiaque. Sept algorithmes de détection de QRS de la littérature ont été mis en œuvre et testés dans 130 contextes cliniques. Un ensemble d'apprentissage, basé sur l'exécution des sept détecteurs de QRS sur 200 réalisations des différents contextes, a été utilisé pour apprendre des arbres de décision d'où des règles de sélection ont été extraites. Les résultats obtenus sur un ensemble de test ont montré que le choix des détecteurs de QRS selon le contexte augmente la F mesure de 0,76% en cas de règles expertes et de 0,18% en cas d'arbre de décision. C'est un bon résultat selon les améliorations typiquement indiquées dans la littérature. Les performances des règles de sélection apprises sont proches de celles des règles expertes et justifient l'utilisation d'une approche basée sur l'apprentissage artificiel.

Bien que la pratique courante en traitement du signal soit d'estimer le niveau du bruit pour adapter des seuils de décision dans les algorithmes, les résultats ont montré la pertinence d'employer une adaptation intelligente des détecteurs de QRS s'accordant non seulement avec le contexte de bruit mais également avec l'information transportée par le signal (contexte de rythme).

L'approche présentée est générale et peut s'appliquer à d'autres domaines du traitement du signal, tels que le traitement d'image ou le monitoring industriel. Nous projetons de l'appliquer aux autres étapes du monitoring cardiaque, comme la classification du QRS et la détection de l'onde P, qui sont des domaines de recherches moins développés et pour lesquels des résultats plus significatifs sont attendus.

Bibliographie

- [1] Portet F, Hernandez AI et Carrault G, Evaluation of real-time QRS detection algorithms in variable contexts, *Medical & Biological Engineering & Computing*, **43**(3), pp. 379-385, 2005.
- [2] Pfahringer B, Bensusan H, Giraud-Carrier C, Meta-learning by landmarking various learning algorithms, Dans *Seventeenth International Conference on Machine Learning*, 2000.
- [3] Todorovski L, Blockeel H et Zeroski S, Ranking with predictive clustering trees, Dans *13th European Conference on Machine Learning*, 2002.
- [4] Chawla NV, Moore TE, Hall LO, Bowyer KW, Kegelmeyer WP et Springer C, Distributed learning with bagging-like performance, *Pattern Recognition Letters*, **24**, pp. 455-471, 2003.
- [5] Chawla N, Elschrich S et Hall LO, Creating Ensembles of Classifiers, Dans *First IEEE International Conference on Data Mining*, 2001.
- [6] Portet F, Quiniou R, Cordier M et Carrault G, Learning Decision Tree for Selecting QRS Detectors for Cardiac Monitoring, Dans *11th Conference on Artificial Intelligence in Medicine (AIME 07)*, 2007.
- [7] Meyer C, Gavela J, Harris M, Combining algorithms in automatic detection of QRS complexes in ECG signals, *IEEE Transactions on Information Technology in Biomedicine*, **10**, pp. 468-475, 2006.
- [8] Darrington J, Towards real time QRS detection: A fast method using minimal pre-processing, *Biomedical Signal Processing and Control*, **1**, pp. 169-176, 2006.
- [9] Christov I, Real time electrocardiogram QRS detection using combined adaptive threshold, *BioMedical Engineering OnLine*, **3**, pp. 1-9, 2004.
- [10] Kadambe S, Murray R et Boudreaux-Bartels F, Wavelet transform-based QRS complex detector, *IEEE Transactions on Biomedical Engineering*, **47**, pp. 838-848, 1999.
- [11] Benitez D, Gaydecki P, Zaidi A et Fitzpatrick A, The use of the Hilbert transform in ECG signal analysis, *Computers in Biology & Medicine*, **31**(5), pp. 399-406, 2001.
- [12] Pan J et Tompkins WJ, A real-time QRS detection algorithm, *IEEE Transactions on Biomedical Engineering*, **32**, pp. 230-236, 1985.
- [13] Gritzali F, Towards a generalized scheme for QRS detection in ECG waveforms, *Signal Processing*, **15**, pp. 183-192, 1988.
- [14] Silipo R, Marchesi C, Artificial neural networks for automatic ECG analysis, *IEEE Transactions on Signal Processing*, **46**(85), pp. 1417-1425, 1998.
- [15] Kókai G, Alexin Z, Gyimóthy T, Classifying ECG Waveforms in Prolog, Dans *4th International Conference on the Practical Application of PROLOG (PAP96)*, 1996.
- [16] Poli R, Cagnoni S, Valli G, Genetic design of optimum linear and nonlinear QRS detectors, *IEEE Transactions on Biomedical Engineering*, **42** (11), pp. 1137-1141, 1995.
- [17] Hernández A, Carrault G, Mora F, Thoraval L, Passariello G, Schleich J-M, Multisensor fusion for atrial and ventricular activity detection in coronary care monitoring, *IEEE Transactions on Biomedical Engineering*, **46**(10), pp. 1186-1190, 1999.
- [18] Karsai G, Sztipanovits J, A Model-Based Approach to Self-Adaptive Software, *Journal of Intelligent Systems*, **14**(3), pp. 46-53, 1999.
- [19] Lejouad-Chaari W, Moisan S, Sevestre-Ghalila S, Rigault JP, Distributed Intelligent Medical Assistant for Osteoporosis Detection, Dans *29th International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC2007)*, 2007.
- [20] Robertson P, Laddaga R. Model Based Diagnosis and Contexts in Self Adaptive Software. In *Self-star Properties in Complex Information Systems, Conceptual and Practical Foundations*. Babaoglu Ö, Jelasity M, Montresor A, Fetzer C, Leonardi S, van Moorsel APA, van Steen M (Eds.). 2005. pp. 112-127.
- [21] Moraes J, Freitas M, Vilani F et Costa E, A QRS complex detection algorithm using electrocardiogram leads, Dans *Computers in Cardiology*, 2002.
- [22] Soulas T, Certen GL, Pichon JL et Carrault G, Algorithm switching in real time monitoring, Dans *Symposium on Electronics and Telecommunications (ETC)*, 1998.
- [23] Moody GB, Mark RG, The impact of the MIT-BIH Arrhythmia Database, *IEEE Transactions on Biomedical Engineering*, **20**(3), pp. 45-50, 2001.
- [24] Moody G, Muldrow W et Mark R, A noise stress test for arrhythmia detectors, Dans *Computers in Cardiology*, 1984.
- [25] Suppappola S et Sun Y, Nonlinear transforms of ECG signals for digital QRS detection: a quantitative analysis, *IEEE Transactions on Biomedical Engineering*, **41**, pp. 397-400, 1994.
- [26] Friesen GM, Jannett TC, Jadallah MA, Yates SL, Quint SR et Nagle HT, A comparison of the noise sensitivity of nine QRS detection algorithms, *IEEE Transactions on Biomedical Engineering*, **37**, pp. 85-98, 1990.
- [27] Witten IH et Frank E, *Data Mining: Practical machine learning tools and techniques*, Morgan Kaufmann (Ed.), 2005.
- [28] Quinlan JR, *C4.5: Programs for Machine Learning*, Morgan Kaufmann (Ed.), 1993.