

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique



Université Abderrahmane Mira

Faculté de Technologie

Département d'Automatique,

Télécommunication et d'Electronique



Projet de Fin d'étude

Pour l'obtention du diplôme de Master

Filière : Télécommunication

Spécialité : Réseaux et Télécommunication.

Thème

*Reconnaissance Automatique de la Modulation
à base des Séparateurs à Vaste Marge (SVM)*

Préparé par :

- *ABDELGHEFFAR Lydia*
- *BOUAMARA Rosa*

Dirigé par :

Mr TOUNSLIM

Examiné par :

Mr AZNIM

Mr KHIREDINE.A

Année universitaire : 2020/2021

Remerciements

Avant tout, nous remercions le bon Dieu de nous avoir aidées à accomplir ce modeste travail et qui nous a donné la force de mener à bien cette tâche.

Au nom de la rigueur scientifique et de la vertu du travail bien accompli qu'il nous a inculquées durant ses enseignements, nous tenons à remercier notre promoteur, Dr TOUNSI Mohamed, enseignant-chercheur à l'université A. Mira de Bejaia pour son encadrement, sa confiance et ses encouragements constants durant ces longs mois de travail.

Nous tenons à remercier vivement les honorables membres de notre jury M. AZNI et M. KHIREDINE, pour l'intérêt qu'ils ont bien voulu porter à notre travail en acceptant son évaluation.

Nos remerciements vont également à Monsieur BEKNADJ Dalil pour l'aide qu'il nous a apportée ainsi qu'à toutes les personnes qui ont contribué de près ou de loin à l'aboutissement de ce travail, particulièrement nos parents qui nous ont toujours encouragées et soutenue ainsi qu'à tout(e)s nos ami(e)s de la promotion 2021.

« Rosa, Lydia »

Dédicaces

Je dédie ce modeste travail à :

Mon père :

J'espère qu'il trouvera dans ce travail les valeurs qu'il m'a transmises, notamment la rigueur, la méthode, la patience et la persévérance.

Ma mère :

J'espère qu'elle trouvera aussi dans les résultats de ce travail, tout ce qu'elle m'a transmis : la générosité, la créativité et le courage d'aller jusqu'au bout.

A mes et sœurs (Kenza, Tiziri, Lynda, Katia, Damia, Radia) et mon frère (Merbah) qui même de loin, m'ont toujours soutenue et accompagnée de prières pendant toutes ces années d'études.

A mes nièces (Massilia, Marissa, Qoqo, Massilva) et mes neveux (Massilas et Idris).

A mes cousines et cousins (sylvia)

A mes chères amies : Safia, Lydia, Damia, et Lynda.

A ma binôme ABDELGHEFFAR Lydia et toute sa famille.

A tous mes ami(e)s de la promotion Télécommunications, particulièrement : Lyes, Aissa, Linda, Yasmina, Rafik, Yasmine, Maria, et Leila.

Et à tous ceux qui ont cru en mes capacités pendant tout mon parcours.

« Rosa »

Dédicaces

Je dédie ce modeste travail à :

Mes très chers parents en témoignage de ma reconnaissance envers leur soutien constant durant toutes mes études, et tous les efforts, les sacrifices qu'ils ont consentis pour mon éducation et ma formation, je leur serai éternellement reconnaissante, que Dieu les garde.

A mes chers grands-parents: El hadi, Nadia.

A mes frères Belkacem, Hamza, Yahia.

A ma tante : Kouka et toute les autres.

A mes oncles.

A mes cousins et cousines.

A Ma chère amie : Lisa

A ma binôme BOUAMARA Rosa et toute sa famille.

A mes ami(e)s de promotion : Lyes, Rafik, Yasmina, Linda, Aissa, Yasmine

Et à tous ceux qui ont cru en mes Capacités durant toute ma formation.

« *Lydia* »

Table de matières

Liste des acronymes	viii
Introduction générale.....	1

CHAPITRE I: INTRODUCTION A LA RADIO COGNITIVE

I.1 Introduction.....	4
I.2 Radio logicielle (Software Radio)	4
I.2.1 Radio logicielle restreinte (SDR).....	4
I.3 La radio cognitive (RC)	5
I.3.1 Les principales fonctions de la RC	5
I.3.2 Les types de détection de spectre	6
I.3.2.1 Détection dans un environnement coopératif	7
I.3.2.2 Détection dans un environnement non-coopératif	7
I.4 La reconnaissance automatique de la modulation (AMR).....	8
I.4.1 AMR à base de la théorie ML.....	8
I.4.2 AMR à base d'extraction des caractéristiques (FB)	9
I.5 Processus AMR à base de l'extraction des caractéristiques	9
I.5.1 Les statistiques d'ordres supérieurs (SOS)	9
I.5.2 Classification.....	11
I.5.2.1 Les SVMs (Séparateur à Vaste Marge)	11
I.5.2.2 ANNs (Artificial Neural Network)	12
I.5.2.3 La méthode des k plus proches voisins (KNN)	13
I.6 Conclusion	14

CHAPITRE II: SYSTEMES MIMO ET MODULATION SPATIALE

II.1 Introduction	16
II.2 Les systèmes MIMO.....	16
II.2.1 Principe de la technologie MIMO	16
II.2.2 Techniques de diversité	17
II.2.4 Les différentes Configurations MIMO	19

II.2.4.1 Configuration MISO (multiple Input Single Output)	20
II.2.4.2 Configuration SIMO (Single Input multiple Output)	20
II.2.5 Algorithmes de réception associés aux systèmes MIMO	21
II.2.5.1 Critère de forçage à zéro (ZF)	21
II.2.5.2 Détecteur par la minimisation d'erreur quadratique moyenne (MMSE detector).....	22
II.2.5.3 Récepteur a Maximum vraisemblance (ML ou MV)	22
II.2.5.4 Annulations successives d'interférences ordonnées (V-BLAST)	22
II.3 Systèmes SM-MIMO.....	23
II.3.1 La modulation spatiale (SM)	23
II.3.2 Principe de fonctionnement de la modulation spatiale (SM)	23
II.3.3 SM en émission	24
II.3.4 SM en réception.....	26
II.3.5 Avantages et inconvénients de la modulation spatiale (SM).....	26
II.4 SM-OFDM (Spatial Modulation-Orthogonal Frequency Division Multiplexing)	27
II.5 Conclusion	28

Chapitre III: Les Séparateurs à Vaste Marge (SVM)

III.1 Introduction	30
III.2 Apprentissage statistique.....	30
III.2.1 Apprentissage supervisé (supervised Learning).....	31
III.2.2 Apprentissage non supervisé (Unsupervised Learning).....	32
III.2.3 Apprentissage par renforcement.....	32
III.2.4 Notions d'apprentissage automatique (ML : Maching Learning).....	33
III.3 Les séparateurs à vaste marge (SVM).....	34
III.3.1 Historique	35
III.3.2Types des SVM	36
III.3.2.1 SVM linéaire	36
III.3.2.2 SVM non linéaire	37
III.3.3 Les SVMs binaires	37

III.3.4 Les SVMs multi-classes	43
III.3.4.1 Un-contre-Reste (One-vs-Rest)	43
III.3.4.2 Un-contre-un (one-vs-one)	44
III.3.5 Architecture générale d'une machine à vecteurs supports	45
III.3.6 Avantages et inconvénients de la machine à vecteurs de support	46
III.4 Conclusion	47

Chapitre IV: Reconnaissance automatique des types de modulations

IV.1 Introduction	49
IV.2 Méthodologie des simulations	49
IV.2.1 Application « Classification Learner / MATLAB »	50
IV.2.2 Librairie LIBSVM	50
IV.3 Métriques d'évaluation des performances des classifieurs	51
IV.3.1 La matrice de confusion	51
IV.3.2 La Courbe ROC	52
IV.3.3 AUC (Aire sous la courbe ROC)	52
IV.4 Résultats de simulations pour différentes configurations antennaires	52
IV.4.1. Simulations sous LIBSVM	53
IV.4.1.1 Paramètres des classifieurs SVM optimaux	53
IV.4.2 Simulations sous « Classification Learner »	60
IV.4.2.1 Évaluation des taux de bonne classification (exactitude du classifieur VM)	61
IV.4.2.2 Évaluation des matrices de confusion	62
IV.4.2.3 Évaluation des AUC (aire sous la courbe ROC)	63
IV.4.2.4. Évaluation des matrices de confusion	65
IV.4.2.5 Évaluation du taux de bonne classification	67
IV.5. Conclusion	68
Bibliographie	70
Annexe A	75
Annexe B	81

Liste des acronymes

1vs1	Un -contre-Un
1vsR	Un -contre-Reste
3D	3 Dimensions
ADC	Analog to Digital Convert
AMR	Automatic Recognition of Modulation
ANN	Artificial Neural Network
AUC	Air Under the Courbe
BER	Bit Error Rate
CP	Cyclic Prefix
C-SVC	C-Support Vector Classification.
EE	Energy efficiency
Epsilon-SVR	Epsilon-Support Vector Regression.
FB	Features Based
HOC	Higher-Order Cumulants
HOM	Higher-Order Moments
HOS	Higher Order Statistic
ICI	Inter-Channel Interferences
ISI	Interferences Inter-Symbole
KKT	Karush, Kuhn et Tucker
KNN	K-Nearest Neighbours
LIBSVM	Library for Support Vector Machines
MIMO	Multiple Input Multiple Output
MISO	Multiple Input Single Output
MDQP	Modulation par Déplacement de Phase en Quadrature
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
MRE	Minimisation Risque Empirique
MV	Maximum Vraisemblance
Nu-SVC	Nu-Support Vector Classification.
Nu-SVR	Nu-Support Vector Regression.
OFDM	Orthogonal Frequency-Division Multiplexing
OSIC	Ordered Successive Interference Cancellation

PAM	Pulse-Amplitude Modulation
PSK	Phase-Shift Keying
PU	Primary User
QAM	Quadrature Amplitude Modulation
RBF	Radial Basis Function
RC	Radio Cognitive
RF	Radio Frequency
ROC	Receiver Operating Characteristic
SDR	Software Defined Radio
SE	Spectral efficiency
SIMO	Single Input Multiple Output
SISO	Single Input Single Output
SNR	Signal-to-Noise Ratio
SM	Modulation Spatial
SM-MIMO	Spatial Modulation- Multiple –Input Multiple-Output
SOS	Statistiques d’Ordres Supérieurs
SP	Sensory Perception
SU	Secondary User
SVM	Séparateur à Vaste Marge
SNR	Signal to Noise Ratio
TA	Transmission Antenna
TEB	Taux d’Erreur Binaire
V-BLAST	Vertical – Bell – Laboratories –Layered –Space – Time
ZF	Forcing to Zero

INTRODUCTION GENERALE

Introduction générale

Les réseaux sans fil ont connu au cours des deux dernières décennies des avancées technologiques considérables. Les équipements qui constituent les réseaux actuels implémentent diverses applications inconcevables dans un passé récent. Les nouvelles générations de réseaux reposent sur l'utilisation de plusieurs technologies émergentes, telles que la radio-cognitive, les stratégies antennaires MIMO...etc. L'utilisation de ces technologies permet une meilleure qualité des communications sans fil, mais au prix de grands efforts dans la recherche et la conception d'algorithmes innovants permettant par exemple, une gestion et une allocation rigoureuses et efficaces des ressources radio.

En effet, la technologie MIMO permet au réseau de disposer d'une nouvelle ressource, à savoir la ressource spatiale (plusieurs antennes) [1]. Cette technologie peut être utilisée soit pour maximiser les débits de transmission, soit pour minimiser le taux d'erreur. Tandis que la technologie radio-cognitive pousse les concepteurs des techniques d'allocation des ressources à repenser la gestion du spectre radio.

Auparavant, l'activation de toutes les antennes émettrices dans les systèmes MIMO, réduit l'optimisation de l'efficacité énergétique, c'est pour cela que le concept de la modulation spatiale est apparu, et qui a induit de nouvelles structures d'antennes plus compactes et reconfigurables tout en ne conservant qu'une seule chaîne de transmission radio (RF) pour assurer une faible complexité [3].

La reconnaissance automatique du type de modulation (AMR) est l'une des composantes cruciales du concept de la RC pour des applications civiles et militaires. L'AMR est un domaine non-coopératif de la théorie des communications traitant de la signalisation du type de modulation d'un signal de communication reçu [4], elle est considérée comme une caractéristique importante d'un système RC qui accentue la capacité de celui-ci d'enrichir sa connaissance de l'environnement de transmission. En effet, elle lui permet de sélectionner le processus de démodulation approprié, au niveau du récepteur, où plusieurs algorithmes sont utilisés pour récupérer le signal (ZF, MMSE, V-BLAST,...).

Dans le cadre de ce travail, nous visons la problématique de détection du type de modulation à base des classificateurs Séparateurs à Vaste Marge (SVM), qui sont des classes d'algorithmes d'apprentissage supervisés, connus pour leurs solides garanties théoriques, leur grande flexibilité ainsi que leur simplicité d'utilisation. Le modèle SVM définit un hyperplan qui permet de séparer au mieux les données de différentes classes à l'aide d'une frontière aussi « simple » que possible [5]. L'objectif de notre travail vise l'implémentation dans un récepteur cognitif, d'une détection

automatique du type de modulation des systèmes MIMO à modulation spatiale (SM-MIMO). En utilisant les SVM.

Notre travail est synthétisé dans ce présent mémoire et organisé en quatre chapitres :

- Le premier chapitre est une introduction à la radio cognitive (RC) où sont présentés les principes de cette radio avec ses fonctionnalités, et ses composants. La fonctionnalité qui retient notre intérêt est la reconnaissance automatique de modulation (AMR) dans un système de communication.
- Le second chapitre est réservé à la technologie MIMO et la modulation spatiale pour une association dans des systèmes SM-MIMO ;
- Le troisième chapitre traite de la méthode de classification choisie, à savoir les Séparateurs à Vaste Marge (SVM) ;
- Le quatrième chapitre constitue notre contribution, où nous donnons des résultats de simulation de la reconnaissance automatique du type de modulation de systèmes SM-MIMO. Les classifieurs SVM utilisés sont issus de l'application « Classification Learner » sous l'interface « Machine Learning and Deep Learning » de MATLAB 2020 et d'une librairie LIBSVM de l'université de Taïwan. Tous nos résultats, sous les deux classifieurs, y sont reportés.
- Nous terminons par une conclusion générale à nos investigations avec quelques perspectives possibles pour notre travail, suivie des références bibliographiques utilisées.

CHAPITRE I

INTRODUCTION A LA RADIO COGNITIVE

I.1 Introduction

Les ressources spectrales sont devenues de plus en plus précieuses avec la prolifération rapide de standards et de services de radiocommunication. Il est aujourd'hui largement reconnu que les systèmes sans fil de communications numériques n'exploitent pas l'intégralité de la bande de fréquence disponible, on appelle ces ressources non exploitées des 'espaces libres' ou opportunités, c'est de cela qu'est né le besoin d'introduire la radio cognitive (RC) qui est apparue comme une solution naturelle aux problèmes de pénurie spectrale. L'évolution des technologies radio a en effet créé un besoin de nouvelles fonctionnalités comme la détection, l'accès et le partage du spectre ainsi que la mobilité spectrale. Nous commencerons ce premier chapitre par la présentation du concept de la radio cognitive ainsi que ses différentes composantes et fonctionnalités. Puis, nous introduirons la reconnaissance automatique de modulation (AMR) dans un système de communication.

I.2 Radio logicielle (Software Radio)

C'est grâce aux travaux de Joseph Mitola que le terme radio logicielle est apparu en 1999 pour définir une classe de radio reprogrammable et reconfigurable [6]. Il y a qui définit de la radio logicielle idéale, qui consiste à une dématérialisation complète de l'interface radio, constituée uniquement d'une antenne et d'un ADC (convertisseur analogique-numérique), dans laquelle les fonctions typiques de l'interface radio généralement réalisées en matériel, telles que la fréquence porteuse, la largeur de bande du signal, la modulation qui sont maintenant définies sous forme logicielle. La radio logicielle moderne intègre également l'implantation logicielle des procédés de cryptographie, codage correcteur d'erreur, codage source de la voix, de la vidéo ou des données. Son concept est considéré comme une manière de rendre les usagers, les fournisseurs de services et les fabricants plus indépendants des normes. Ainsi, avec cette solution, les interfaces radio peuvent, en principe, être adaptées aux besoins d'un service particulier pour un usager particulier dans un environnement donné à un instant donné.

I.2.1 Radio logicielle restreinte (SDR)

La SDR (Software Defined Radio) est un système de communication radio qui peut s'adapter à n'importe quelle bande de fréquence et recevoir n'importe quelle modulation en utilisant le même matériel. Ce concept permet de résoudre des problèmes de gestion dynamique du spectre et d'appliquer n'importe quelle modulation tout en utilisant le même support matériel [7]. C'est-à-dire le SDR idéal peut s'adapter automatiquement aux nouvelles fréquences et aux nouvelles modulations.

I.3 La radio cognitive (RC)

La RC est une forme de communication sans fil dans laquelle un émetteur/récepteur peut détecter intelligemment les canaux de communication qui sont en cours d'utilisation et ceux qui ne le sont pas, et peut se déplacer vers les canaux inutilisés. Ceci permet d'optimiser l'utilisation des fréquences radio disponibles du spectre tout en minimisant les interférences avec d'autres utilisateurs.

C'est une radio intelligente programmée et configurée dynamiquement, elle permet d'adapter chaque appareil aux conditions spectrales du moment et offre donc aux utilisateurs un accès plus souple, efficace et complet à cette ressource. Sa capacité d'adaptation est mise en œuvre autour d'un système clé appelé "radio logicielle", qui lui offre les fonctionnalités de flexibilité et de reconfiguration nécessaires.

La RC nécessite une gestion alternative du spectre qui est la suivante : un utilisateur dit secondaire (SU pour Secondary User), pourra à tout moment accéder à des bandes de fréquence libres, c'est-à-dire, non occupées par l'utilisateur dit primaire (PU pour Primary User), qui possède une licence sur cette bande. L'utilisateur secondaire devra céder ces bandes de fréquences une fois le service terminé ou une fois qu'un utilisateur primaire aura montré son désir de connexion.

I.3.1 Les principales fonctions de la RC

- **Détection du spectre (Spectrum sensing):** Détecter le spectre non utilisé et le partager sans interférences avec d'autres utilisateurs. La détection des utilisateurs primaires est la façon la plus efficace pour détecter les espaces blancs du spectre. Son but est d'obtenir le statut du spectre (libre /occupé), de sorte que le spectre peut être consulté par un utilisateur secondaires en vertu de la contrainte d'interférence [8].
- **Gestion du spectre (Spectrum management):** Capturer les meilleures fréquences disponibles et décider de la meilleure bande de spectre pour répondre aux exigences des utilisateurs [8].
- **Mobilité du spectre (Spectrum mobility):** C'est le processus qui permet à l'utilisateur de la RC de changer sa fréquence de fonctionnement. Les réseaux radio cognitifs essaient d'utiliser le spectre de manière dynamique en permettant à des terminaux radio de fonctionner dans la meilleure bande de fréquence disponible, de maintenir les exigences de communication transparentes au cours de la transition à une meilleure fréquence.
- **Partage du spectre (Spectrum sharing) :** La radio cognitive doit avoir accès au spectre et le partager avec plusieurs autres utilisateurs (SU ou PU). Il consiste à distribuer le spectre entre les utilisateurs principaux et secondaires [8].

- **Décision sur le spectre (Spectrum decision) :** L'attribution d'un canal à un utilisateur RC est basée sur la disponibilité du spectre, et doit tenir compte de plusieurs critères comme la qualité de service exigée par l'application, le besoin en bande passante de l'utilisateur, etc....

I.3.2 Les types de détection de spectre

L'une des exigences les plus fondamentales des systèmes de radiocommunication cognitifs consiste à éviter les interférences pour les utilisateurs primaires potentiels se trouvant à proximité. Pour garantir que les utilisateurs secondaires n'interféreront pas avec les utilisateurs primaires, ils doivent détecter l'existence du signal de l'utilisateur primaire dans les environnements sans-fil. Ce processus de détection est réalisé en écoutant l'environnement sans-fil, et il est connu sous le nom de « détection du spectre ».

La détection de spectre est l'élément clé de la communication dans les RC, c'est la capacité de déterminer et de détecter si l'utilisateur PU est absent ou présent, cela va permettre à l'utilisateur SU d'utiliser son canal ou bien de passer à une autre bande pour éviter les interférences. Cette opération permet d'identifier l'état d'utilisation du spectre (occupé ou vacant) et de caractériser le signal occupant le spectre par sa fréquence porteuse, sa modulation et sa bande passante [9].

La détection du spectre est modélisée par un problème de test d'hypothèses binaire comme suit :

$$\begin{cases} H_0: r(t) = n(t) & (PU \text{ est absent}) \\ H_1: r(t) = h.s(t) + n(t) & (PU \text{ est présent}) \end{cases} \quad (\text{I.1})$$

Où :

H_0 désigne l'hypothèse d'absence du PU sur le canal de fréquence considéré ;

H_1 Dénote l'hypothèse de la présence du PU sur le canal considéré ;

$r(t), s(t), n(t), h$ sont respectivement le signal reçu (par l'utilisateur SU), le signal transmis (par l'utilisateur PU), le bruit et le gain du canal.

Il y'a trois catégories de techniques de détection de spectre: détection dans un environnement coopératif, détection dans un environnement non-coopératif, et détection basée sur les interférences.

On s'intéresse juste aux deux premières catégories.

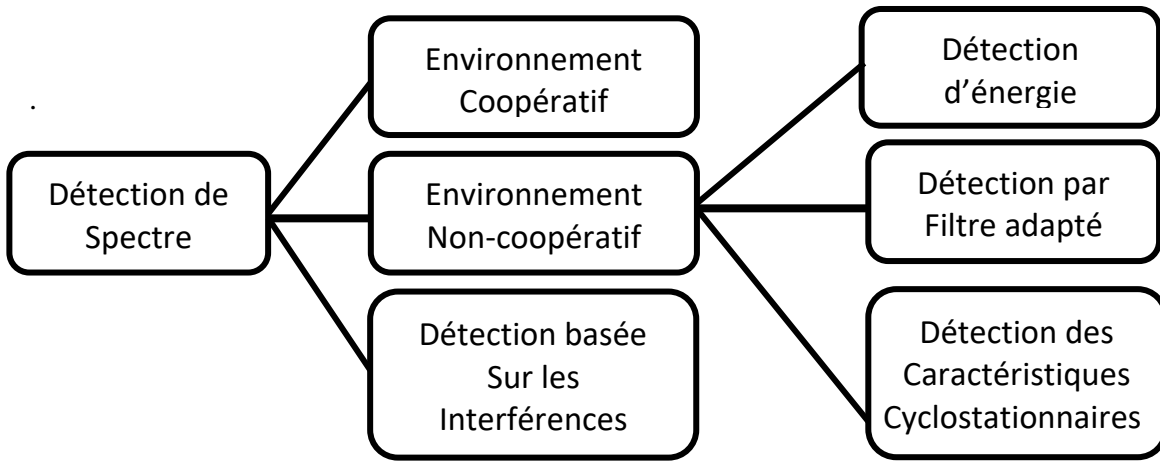


Figure I.2 : Types de détection de spectre

I.3.2.1 Détection dans un environnement coopératif

La détection coopérative a été proposée, pour remédier à des problèmes tels que le nœud caché. Elle augmente considérablement la probabilité de détection. Elle permet aux Sus de coopérer et consolider leurs efforts de détection afin de parvenir à une conclusion plus précise sur la disponibilité du spectre. Quel que soit la méthode utilisée, l'échange d'information se fait à travers d'un canal dédié. Différentes approches peuvent être établies [10]:

- **L'approche centralisée :** Chaque nœud effectue sa propre détection et l'envoie à une entité centrale qui englobe toutes les détections de tous les utilisateurs SU [11].
- **L'approche distribuée :** Ne nécessite pas de centre de fusion en revanche, chaque nœud reçoit les détections des nœuds voisins et les combine avec la sienne pour prendre la décision [11].

I.3.2.2 Détection dans un environnement non-coopératif

On l'appelle détection de l'émetteur, où l'utilisateur SU se base sur la détection du signal de l'utilisateur PU. Il existe plusieurs types, dont on citera [12]:

- **Détection d'énergie :** C'est une méthode non cohérente, appelé aussi « radiomètre ». Elle ne nécessite pas la connaissance du signal du PU. L'idée de base est de calculer l'énergie du signal reçu, pour ensuite la comparer à un seuil prédéfini. Cette méthode est utilisée lorsque le SU ne rassemble pas assez d'informations sur le PU. Elle est limitée en raison de ses mauvaises performances dans les environnements à faible SNR.
- **Détection par filtre adapté :** L'utilisation de ce filtre nous permettra d'avoir un fort contraste entre le signal et le bruit. Afin de détecter la présence du PU on passe le signal par un filtre, celui-ci fait en sorte d'augmenter l'amplitude du signal utile et de supprimer l'amplitude du bruit en même temps. Si à la sortie on remarque un pic dans l'amplitude du signal cela voudra dire que le PU est présent à cet instant, si non le PU est absent.

- **Détection des caractéristiques cyclostationnaires** : Dans ce type de détection, on utilise la fonctionnalité statique intégrée d'un signal ; le bruit de fond et les interférences ne sont pas liés au domaine temporel ou fréquentiel ; ainsi, si le SU possède les statistiques des informations de la fonction de corrélation du signal du PU, il peut augmenter la précision de détection.
- **Détection par covariance** : La covariance est la mesure par laquelle un lien entre deux signaux est calculé. Elle est généralement différente pour un signal et pour le bruit, c'est la raison pour laquelle elle est utilisée pour détecter s'il y a une présence d'un signal électrique ou pas.
- **Détection par ondelettes** : La large bande est représentée comme un train de sous-bandes consécutives dans laquelle la densité spectrale de puissance est constante. La détection par ondelettes est appliquée dans cette large bande, pour déterminer les singularités observées dans les bords des sous bandes ou les fréquences qui les délimitent. Elle permet une meilleure résolution en fréquence en un temps meilleur comparé à d'autres.

I.4 La reconnaissance automatique de la modulation (AMR)

La reconnaissance automatique de la modulation (AMR) est un domaine non-coopératif de la théorie des communications traitant de la signalisation du type de modulation d'un signal de communication reçu. Elle consiste à identifier, au niveau du récepteur d'une chaîne de transmission, la constellation à laquelle appartiennent les symboles du message transmis. Cette reconnaissance est nécessaire dans de nombreux scénarios de communication, qu'ils soient coopératifs ou non coopératifs. Cette technique est utilisée pour identifier le type de modulation utilisé par les PU, ce qui permet de décider si le spectre est libre ou non [4]. Afin de définir le type de signaux inconnus reçus, elle permet d'extraire des paramètres ou des caractéristiques. Grâce à l'AMR de la radio cognitive, le récepteur peut détecter le changement de modulation sans qu'il ait besoin de transmettre des informations supplémentaires.

Les méthodes de recherche sur l'AMR sont principalement divisées en deux catégories : l'une est basée sur la théorie du maximum de vraisemblance (ML) et l'autre est la méthode basée sur l'extraction des caractéristiques (FB), et dans notre cas nous nous intéressons à la seconde méthode que nous détaillons ci-après.

I.4.1 AMR à base de la théorie ML

L'idée principale des méthodes ML est la théorie Bayésienne, qui doit acquérir autant d'informations que possible sur le canal de transmission pour concevoir la fonction de vraisemblance. Il permet d'analyser et comparer les caractéristiques statistiques des signaux reçus avec les meilleurs seuils de décision obtenus par le calcul. Cette méthode permet d'avoir des résultats approximatifs.

I.4.2 AMR à base d'extraction des caractéristiques (FB)

La méthode FB permet d'extraire des attributs caractérisant les signaux reçus afin de concevoir un classifieur approprié pour la reconnaissance de la modulation. Il est nécessaire d'extraire les fonctionnalités clés pour que l'algorithme puisse identifier correctement les différents types de modulation.

I.5 Processus AMR à base de l'extraction des caractéristiques

Le processus AMR à base de méthodes de l'extraction des caractéristiques comprend trois étapes, prétraitement du signal, extraction de caractéristiques et reconnaissance de modulation en utilisant les caractéristiques extraites. Ce processus est illustré dans la figure (I.3). Le but principal du prétraitement du signal est de réduire le bruit du signal, ce qui est utile pour l'extraction des caractéristiques à l'étape suivante [9].

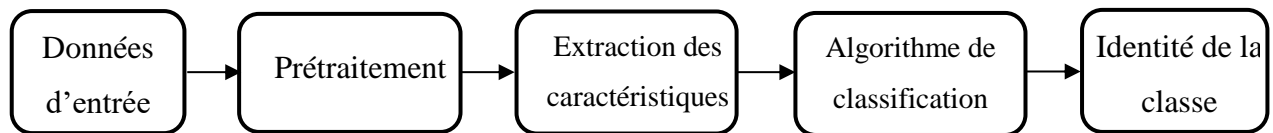


Figure I.3 : Processus AMR à base de l'extraction de caractéristiques.

I.5.1 Les statistiques d'ordres supérieurs (SOS)

Les Statistiques d'Ordre Supérieur à 2 (Higher Order Statistics), autrement dit les moments (HOM) et cumulants (HOC) d'ordre supérieur à 2, sont utilisées essentiellement en complément aux statistiques d'ordre 2, utilisées essentiellement dans les algorithmes d'identification du type de modulation. Elles donnent une description plus complète des données et de leurs propriétés, comme elles fournissent suffisamment d'informations pour classer efficacement chaque modulation. Ces statistiques assurent des performances élevées de classification des signaux de modulation numérique.

Le moment d'un signal est égal à l'espérance du signal élevée à une puissance déterminée par l'ordre du moment. La moyenne d'un signal est le moment du premier ordre, tandis que le moment du deuxième ordre indique généralement la puissance du signal.

Les moments d'ordre supérieur (HOM) sont définis par l'équation suivante [13] :

$$M_{pq}(\vec{v}) = E[\vec{v}^p \vec{v}^{*q}] \quad (\text{I.2})$$

Où : p et q sont des entiers non négatifs, E désigne l'espérance, et \vec{v}^* est le conjugué du vecteur \vec{v} .

Les moments de second ordre (HOM) peuvent être exprimés par :

$$M_{20}(\vec{v}) = E[(\vec{v})^2] \text{ et } M_{21}(\vec{v}) = E[|\vec{v}|^2] \quad (\text{I.3})$$

De manière similaire, les moments peuvent être estimés comme suit :

$$C_{pq} = [\bar{v}, \dots, \bar{v}, \bar{v}^*, \dots, \bar{v}^*] \quad (\text{I.4})$$

Où: $[\bar{v}, \dots, \bar{v}]$ désigne le terme $p-q$, et $[\bar{v}^*, \dots, \bar{v}^*]$ désigne le terme q .

En utilisant l'équation I.4 ci-dessus, $C_{20}(\bar{v})$ et $C_{21}(\bar{v})$ peuvent être écrits comme suit :

$$C_{20}=[\bar{v}, \bar{v}] \text{ Et } C_{21}=\text{Cum}[\bar{v}, \bar{v}^*] \quad (\text{I.5})$$

Les cumulants HOCs sont exprimés en fonction des moments HOMs d'un signal. Les relations entre certains des HOC les plus couramment utilisés et les HOM sont listées ci-après [14] :

$$\begin{aligned} C_{40} &= M_{40} - 3M_{20}^2 \\ C_{41} &= M_{41} - 3M_{21}M_{20} \\ C_{42} &= M_{42} - 2M_{21}^2 - |M_{20}|^2 \\ C_{60} &= M_{60} - 15M_{20}M_{40} - 30M_{20}^3 \\ C_{61} &= M_{61} - 5M_{40}M_{21} - 10M_{41}M_{20} + 3M_{21}M_{20}^2 \\ C_{62} &= M_{62} - 6M_{42}M_{20} - 8M_{41}M_{21} - M_{40}M_{22} + 6M_{22}M_{20}^2 + 24M_{20}M_{21}^2 \\ C_{63} &= M_{63} - 9M_{21}M_{42} + 12M_{21}^3 - 3M_{20}M_{43} - 3M_{22}M_{41} + 18M_{20}M_{21}M_{22} \end{aligned} \quad (\text{I.6})$$

Les valeurs des cumulants pour chaque type de modulation, sont uniques et peuvent donc être utilisés comme caractéristiques de la classification. Le tableau suivant représente les valeurs théoriques des cumulants pour certains types de modulations.

<i>Modulation</i>	C_{20}	C_{21}	C_{40}	C_{41}	C_{42}
2-PAM	1.0000	1.0000	-2.0000	-2.0000	-2.0000
4-PAM	1.0000	1.0000	-1.3600	-1.3600	-1.3600
8-PAM	1.0000	1.0000	-1.2381	-1.2381	-1.2381
B-PSK	1.0000	1.0000	-2.0000	-2.0000	-2.0000
Q-PSK	0.0000	1.0000	0.0000	0.0000	-1.0000
8-PSK	0.0000	1.0000	0.0000	0.0000	-1.0000
4-QAM	0.0000	1.0000	-0.6800	0.0000	-0.6800
16-QAM	0.0000	1.0000	-0.6800	0.0000	-0.6800
64-QAM	0.0000	1.0000	-0.6191	0.0000	-0.6191

Tableau I.1 : Cumulant théorique des modulations M-PAM, M-PSK et M-QAM

I.5.2 Classification

La classification est la dernière étape du processus AMR à base des méthodes d'extraction des caractéristiques. C'est la partie décisionnelle de l'ensemble du processus d'identification des types de modulation. Elle consiste à étiqueter des objets suivant les classes auxquelles ils appartiennent. Les systèmes de classification comportent trois étapes :

- L'apprentissage, qui va s'efforcer de trouver la description de l'espace des observations qui traduit le mieux l'association avec les classes correspondantes. C'est une étape clé dans les systèmes de reconnaissance. On distingue deux types d'apprentissage : apprentissage supervisé et apprentissage non-supervisé [15].
- Le test, qui permet d'évaluer les performances du classifieur pour un apprentissage donné.
- La validation, qui est un processus permettant d'estimer les performances du classifieur.

Les principaux classifieurs utilisés dans l'AMR sont : les séparateurs à vaste marge (SVM) (que nous utiliserons durant notre présent travail), les réseaux de neurones artificiels (ANN), et les K plus proches voisins (KNN).

I.5.2.1 Les SVMs (Séparateur à Vaste Marge)

Les SVMs (Support Vector Machine (en anglais) ou Séparateur à Vaste Marge (en français)) développés dans les années 1990 par Vapnik, sont un ensemble de techniques d'apprentissage destinées à résoudre des problèmes de discrimination, c'est-à-dire décider à quelle classe appartient un échantillon. Le succès de cette méthode est justifié par les solides bases théoriques qui la soutiennent, leur grande flexibilité, ainsi que leur simplicité d'utilisation. Les SVM reposent sur deux idées clés : la notion de marge maximale et la notion de fonction noyau.

Le principe des SVMs est simple : il ont pour but de séparer les données en classes à l'aide d'une frontière aussi « simple » que possible, de telle façon que la distance entre les différents groupes de données et la frontière qui les sépare soit maximale. Cette distance est aussi appelée « marge », Comme le montre la figure (I.4) [16].

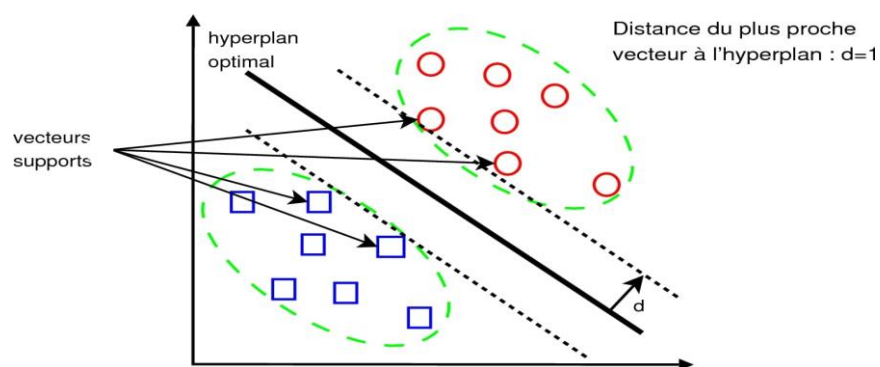


Figure I.4 : Classifieur SVM (dans cas linéaire)

Cette notion de frontière suppose que les données soient linéairement séparables, ce qui est rarement le cas, c'est pour ça qu'ils reposent souvent sur l'utilisation de fonctions « noyaux ». Ces fonctions mathématiques permettent de séparer les données en les projetant dans un feature space (un espace vectoriel de plus grande dimension, la figure (I.5)). La technique de maximisation de marge permet, quant à elle, de garantir une meilleure robustesse face au bruit et donc un modèle plus généralisable.

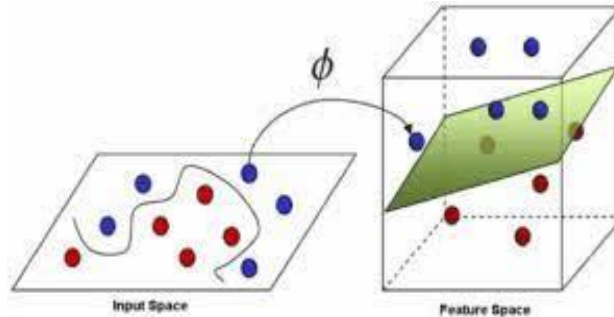


Figure I.5: Classifieur SVM (dans le cas non-linéaire)

I.5.2.2 ANNs (Artificial Neural Network)

Aujourd'hui le domaine des réseaux de neurones artificiels est en pleine expansion. Il a connu un fort développement depuis les années 1940. L'idée fondatrice des réseaux de neurones était d'imiter le fonctionnement du cerveau humain. On considère généralement qu'un réseau de neurones est constitué d'un grand ensemble d'unités (ou neurones), ayant chacune une petite mémoire locale. Ces unités sont reliées par des canaux de communication, qui transportent des données numériques. Ce type de classifieur est largement utilisé lors de la reconnaissance des formes pour ses bonnes performances, et de sa capacité de généralisation [17].

Un réseau de neurones multicouche se compose d'une série de couches d'unités ou neurones auxquels sont attachés des poids. Chaque neurone effectue une tâche relativement simple : recevoir les informations externes ou les sorties des neurones de la couche précédente et les utiliser pour calculer sa propre sortie (nommée activation) qui se propage aux neurones connectés de la couche suivante. Une deuxième tâche consiste à ajuster les poids. On distingue généralement trois types de couches dans un réseau de neurones :

- **La couche d'entrée :** Constituée des neurones qui reçoivent les informations de l'extérieur du réseau et renvoient les résultats intermédiaires aux neurones de la couche suivante.
- **La couche de sortie :** Qui calcule les résultats finaux et fournit la réponse du classifieur.
- **Les couches cachées :** Qui se situent entre la couche d'entrée et la couche de sortie.

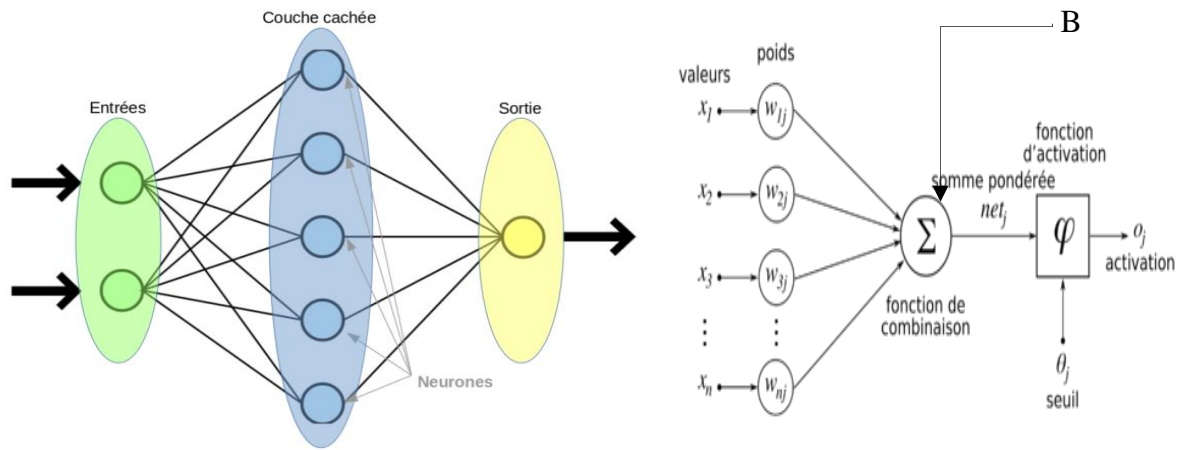


Figure I.6 : Structure d'un classifieur ANN multicouches.

Les valeurs w_i représentent les poids associés à chaque entrée x_j . φ est la fonction dite d'activation, b est le biais, et O_j définit la sortie de chaque neurone. Un système doté d'un réseau de neurones apprend à effectuer une tâche en analysant des exemples pour s'entraîner. Ces exemples ont préalablement été étiquetés afin que le réseau puisse savoir ce dont il s'agit.

I.5.2.3 La méthode des k plus proches voisins (KNN)

L'algorithme des k plus proches voisins ou k-Nearest Neighbors (kNN) est un algorithme d'apprentissage automatique supervisé simple et facile à mettre en œuvre, qui peut être utilisé pour résoudre les problèmes de classification et de régression, comme il est aussi utilisé dans des algorithmes d'identification de type de modulation. Il repose exclusivement sur le choix de la métrique de classification [18]. Il est "non paramétrique" (seul k doit être fixé) et se base uniquement sur les données d'entraînement. Le paramètre k doit être déterminé par l'utilisateur ($k \in \mathbb{N}$). Le principe de cet algorithme de classification est très simple. On lui fournit un ensemble de données d'apprentissage D, une fonction de distance d, et un entier k. Pour tout nouveau point de test x, pour lequel il doit prendre une décision, l'algorithme recherche dans D, les k points les plus proches de x au sens de la distance d, et attribue x à la classe qui est la plus fréquente parmi ces k voisins. Un exemple d'algorithme KNN est illustré dans la figure (I.7).

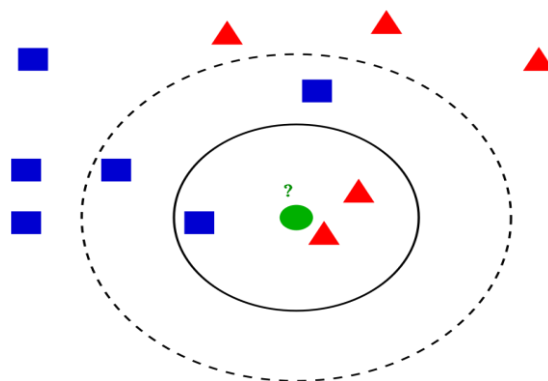


Figure I.7 : Classifieur KNN

Dans cette figure, nous disposons de deux classes de données (carré bleu et triangle rouge). L'échantillon de test (cercle vert) pourrait être classé soit dans la première classe de carré bleu ou la seconde classe de triangles rouges. Si $k = 3$ (cercle en ligne pleine) il est affecté à la seconde classe car il y a deux triangles et seulement un carré dans le cercle considéré. Si $k = 5$ (cercle en ligne discontinue) il est affecté à la première classe (3 carrés face à deux triangles dans le cercle externe).

I.6 Conclusion

Dans la première partie de ce chapitre, nous avons présenté les principes de la radio cognitive qui est un domaine alliant l'intelligence artificielle aux télécommunications, en passant par une petite description de la radio logicielle. Dans la deuxième partie, nous avons présenté les méthodes de reconnaissance automatique de modulation ainsi que les types des classifieurs qu'elles utilisent (ANN, SVM, et KNN) dans les systèmes de communication. Nous nous intéressons plus aux techniques de détection et de classification automatiques du type de modulations dans des systèmes MIMO utilisant des réseaux d'antennes à la fois à l'émission et/ou à la réception. Le prochain chapitre sera ainsi consacré à la présentation de cette technologie antennaire.

CHAPITRE II

SYSTEMES MIMO ET MODULATION

SPATIALE

II.1 Introduction

Les systèmes modernes de radiocommunication doivent fournir des débits de données de plus en plus élevés qui ne peuvent être assurés par les méthodes conventionnelles d'utilisation de bandes passantes plus larges ou de types de modulation d'ordres supérieurs. De nouveaux procédés d'utilisation des canaux de transmission ont été développés. Les systèmes à antennes multiples (Entrées multiples, Sorties Multiples - MIMO) sont l'une des technologies les plus puissantes. Ils consistent en une architecture de transmission à plusieurs antennes à l'émission et à la réception. Cette diversité spatiale permet d'améliorer considérablement le débit et la qualité de transmission. MIMO est utilisée dans les radars, les réseaux sans fil et les réseaux mobiles permettant des transferts de données à plus longue portée et avec un débit plus élevé.

II.2 Les systèmes MIMO

Les systèmes MIMO (Multiple Input Multiple Output) ont été introduits en 1997 par Jack Winters. Ils désignent une technologie d'antennes destinée aux communications sans fil, et consistant à utiliser des réseaux d'antennes, N_t à l'émission et N_r à la réception dans une bande de fréquence donnée. Ce procédé permet de résoudre les problèmes d'encombrement et de limitation de capacité des réseaux sans fil large bande [19]. Ils introduisent une nouvelle forme de diversité connue sous le nom de diversité spatiale, cette technique cherche à exploiter les phénomènes d'évanouissement et de multi-trajets. En effet, le récepteur reçoit et combine plusieurs copies d'un même signal envoyées à partir de plusieurs antennes et subissant des évanouissements indépendants. Par conséquent, le récepteur prend la bonne décision permettant de reconstituer le message d'origine sans erreur. L'utilisation de plusieurs antennes, sert à améliorer la fiabilité de transmission en réduisant la probabilité d'erreur (gain en diversité) et à augmenter le débit de transmission (gain de multiplexage) [20,21]. Des méthodes robustes de traitement du signal du côté du récepteur sont nécessaires pour reconstituer le message d'origine sans erreur [22] du fait que tous les signaux transmis se mélangent.

II.2.1 Principe de la technologie MIMO

Le principe de la technologie MIMO consiste à émettre simultanément dans un même canal, N flux d'informations de même fréquence sur N_t antennes d'émission $\{s_i\}$ ($i=1 \dots N_t$), où chaque flux est reçu par N_r antennes $\{y_j\}$ ($j=1 \dots N_r$), et des traitements adéquats sont appliqués à la réception. Le flux transmis par les antennes peut être le même afin d'améliorer la diversité en émission, ou bien différent pour augmenter la capacité de l'émetteur [22]. Un algorithme à la réception permet ensuite d'identifier les différents flux en vue de les restituer en un seul flux.

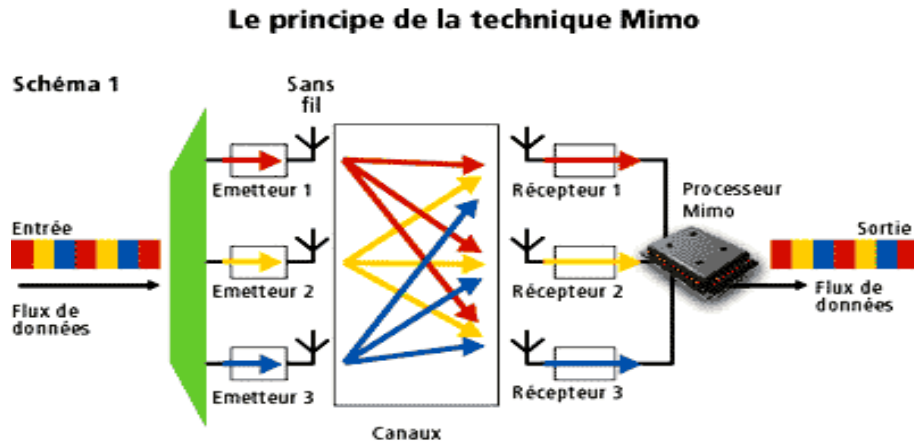


Figure II.1: Principe de la technologie MIMO.

Sur la figure ci-dessus, chaque trajet entre une antenne émettrice et une antenne réceptrice est caractérisé par un gain complexe h_{ij} représentant le coefficient d'évanouissement entre le $i^{ème}$ émetteur et le $j^{ème}$ récepteur. À la réception, chaque antenne reçoit la somme des symboles s_i transmis simultanément par chacune des N_t antennes émettrices. Le signal y_j reçu par la $j^{ème}$ antenne peut par conséquent s'écrire de manière discrète [23]:

$$y_j = \sum_{i=0}^{N_t} h_{ji} s_i + n_j \tag{II.1}$$

Où n_j est un bruit additif gaussien associé à l'antenne de réception.

L'équation (II-1) peut être reformulée sous forme matricielle comme suit :

$$Y = H.S + N \tag{II.2}$$

avec :

$$H = \begin{bmatrix} h_{1,1} & h_{1,2} & \dots & h_{1,N_r} \\ \vdots & \vdots & \ddots & \vdots \\ h_{N_r,1} & h_{N_r,2} & \dots & h_{N_r,N_t} \end{bmatrix} \tag{II.3}$$

$$S = [s_1, \dots, s_{N_t}]^T \tag{II.4}$$

$$Y = [y_1, \dots, y_{N_r}]^T \tag{II.5}$$

et :

$$N = [n_1, \dots, n_{N_r}]^T \tag{II.6}$$

II.2.2 Techniques de diversité

La diversité est une technique utilisée dans les systèmes de transmissions radio-mobiles afin de lutter contre l'évanouissement causé par les multi-trajets lors de la transmission. Le principe est de transmettre plusieurs copies de la même information sur plusieurs canaux avec une puissance comparable et un évanouissement indépendant. Les principales formes de diversité sont listées ci-après :

- **Diversité temporelle** : Elle consiste à émettre plusieurs répliques d'un même signal (ou des versions redondantes) dans des intervalles temporels (Cf. Figure II.2) séparés d'au moins le temps de cohérence du canal T_c (temps pendant lequel le canal est considéré stationnaire).

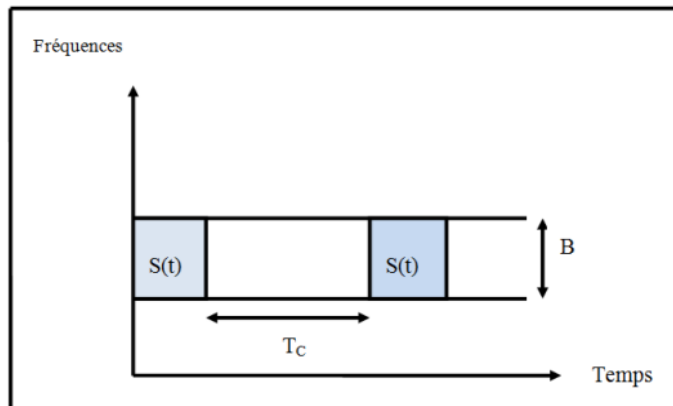


Figure II.2: Transmission avec une diversité temporelle.

- **Diversité fréquentielle** : Elle est efficace lorsque les évanouissements du canal sont sélectifs en fréquence. Ce type de diversité revient à émettre le même signal (ou des versions redondantes) sur plusieurs fréquences porteuses, dont l'écartement fréquentiel est d'au moins la bande de cohérence du canal B_c (Figure II.3).

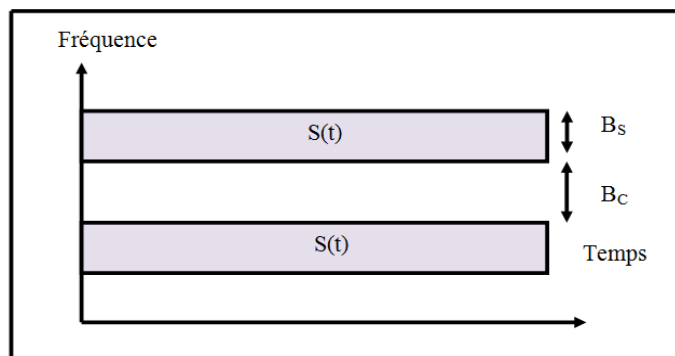


Figure II.3 : Transmission avec une diversité fréquentielle.

- **Diversité spatiale (ou diversité d'antennes)** : Elle consiste à émettre simultanément un même message sur différentes antennes à l'émission, séparées dans l'espace d'au moins la distance de cohérence (distance minimale pour obtenir des évanouissements indépendants) (Figure II.4). Les signaux reçus sur chacune des antennes de réception sont ensuite remis en phase et sommés de façon cohérente. Cela permet d'augmenter le rapport signal sur bruit (grâce au gain de diversité). L'inconvénient majeur de ce type de diversité est l'encombrement car en théorie les antennes doivent être espacées d'au moins $0,5\lambda$ pour que les sous-canaux MIMO soient décorrélés les uns des autres [24].

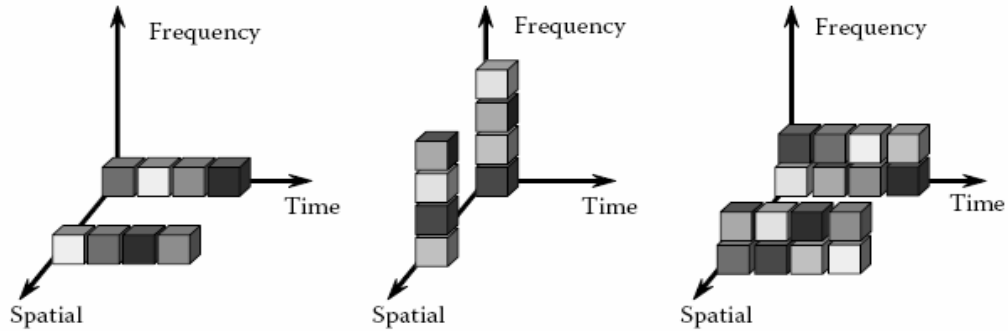


Figure II.4 : Différents types de diversité spatiale

- **Multiplexage spatial MIMO** : Chaque message est découpé en sous-messages. On transmet simultanément les sous-messages différents sur chacune des antennes d'émission. Les signaux reçus sur les antennes de réception sont réassemblés pour reconstituer le message d'origine. Le multiplexage MIMO permet d'augmenter les débits de transmission (grâce au gain de multiplexage) [24].
- **Le MIMO-Beamforming** : La technique de Beamforming correspond à un filtre spatial qui permet de constituer des faisceaux directs, pour favoriser le gain dans une direction souhaitée et atténuer la puissance de l'onde dans les autres directions. Lors de la transmission, la focalisation du faisceau offre la capacité de concentrer l'énergie émise par le canal radio vers un récepteur spécifique. En ajustant la phase et l'amplitude des signaux transmis, il est possible d'obtenir une addition constructive des signaux correspondants au niveau du récepteur. Cela augmente la puissance du signal reçu et donc le débit de l'utilisateur. De même, à la réception, le Beamforming assure la capacité de collecter l'énergie du signal d'un émetteur spécifique et de limiter les interférences entre utilisateurs [24].

II.2.4 Les différentes Configurations MIMO

Lors de la transmission, la quantité d'information transmise est limitée par la capacité du canal (limite de Shannon). Pour un lien simple, c'est à-dire une antenne à l'émission et une autre à la réception (système dit SISO - Single Input Single Output - (figure II.5)), la capacité maximale du canal est donnée par [25] :

$$C = W \cdot \log_2 \left(1 + \frac{P}{N_o \cdot W} \right) \quad (\text{II.7})$$

Où P est la puissance du signal utile [W], N_o la densité spectrale de puissance du bruit [W/Hz] et W la bande passante utilisée [Hz].

Pour rendre la capacité indépendante de la bande passante W du canal, la notion d'efficacité spectrale η a été définie comme le rapport entre la capacité du canal et la bande passante :

$$\eta = \frac{C}{W} = \log_2\left(1 + \frac{P}{N_0 \cdot W}\right) \quad (\text{II.8})$$

Physiquement cela se traduit par la quantité d'information que l'on peut envoyer par chaque Hertz de bande passante. Autrement dit, l'efficacité spectrale renseigne sur la limite à laquelle on peut exploiter la bande passante disponible [25]. Pour repousser cette limite, des améliorations de débit, tout en gardant une complexité matérielle raisonnable, ont été proposées en exploitant la technique de diversité spatiale en émission et/ou en réception. C'est ainsi que sont apparues différentes configurations (ou formats) MIMO avec des avantages et des inconvénients différents pouvant être équilibrés afin de fournir la solution optimale pour une application donnée.

II.2.4.1 Configuration MISO (multiple Input Single Output)

Ces systèmes MISO utilisent plusieurs antennes à l'émission et une seule antenne en réception (figure II.5). Chaque antenne émet le même symbole d'énergie $\frac{E_s}{N_t}$. Autrement dit, la puissance d'émission est divisée sur le nombre d'antennes, telle que $P = \sum_{i=1}^{N_t} P_i$, avec P_i est la puissance émise sur chaque antenne. Si P_i est la même pour toutes les antennes, alors $P = N_t \cdot P_i$ [26]. Dans ce cas, la capacité du canal MISO est donnée comme suit :

$$C = W \cdot \log_2\left(1 + N_t \cdot \frac{P_i}{N_0 W}\right) = W \cdot \log_2\left(1 + \frac{P}{N_0 W}\right) \quad (\text{II.9})$$

L'équation (II.7) peut s'écrire de la même façon que l'équation (II.9), mais l'intérêt de cette technique réside dans le fait que dans les multi trajets, la probabilité d'évanouissement dans N_t antennes est inférieure à la probabilité d'évanouissement sur une seule antenne.

II.2.4.2 Configuration SIMO (Single Input multiple Output)

C'est une technique qui utilise plusieurs antennes en réception pour lutter contre l'évanouissement dû au canal. Le signal reçu est additionné et le rapport signal sur bruit total est la somme des rapports RSB de chacune des antennes de la réception. La capacité du canal SIMO est donnée par la formule suivante [24] :

$$C = W \cdot \log_2\left(1 + N_r \cdot \frac{P}{N_0 W}\right) \quad (\text{II.10})$$

N_r étant le nombre d'antennes réceptrices.

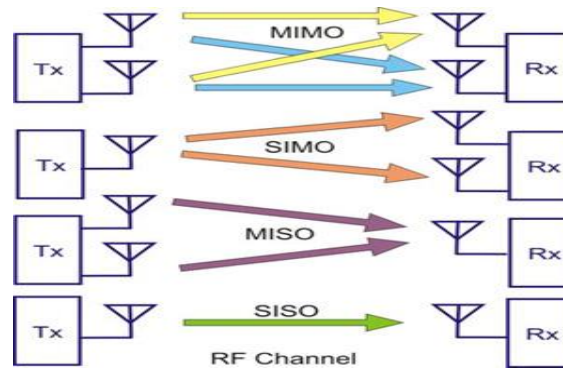


Figure II.5 : Différences entre formats SISO, SIMO, MISO et MIMO.

II.2.5 Algorithmes de réception associés aux systèmes MIMO

Dans un système MIMO, il existe de nombreux algorithmes de réception possibles qui peuvent être utilisés pour récupérer des symboles avec un accroissement minimal du bruit. Les moins compliqués sont les récepteurs linéaires basés sur le critère obligatoire zéro (ZF) ou la minimisation de l'erreur quadratique moyenne (MMSE). Afin d'obtenir un détecteur capable de réduire le temps de calcul avec une efficacité acceptable, beaucoup de travail a été fait dans [25] [26]. Ce travail se concentre sur les deux aspects suivants:

- *La méthode heuristique* : est une méthode approximative, mais elle peut fournir des solutions de faible complexité (ZF, MMSE, ML).
- *Méthode précise* : elle atteint de très bonnes performances plus rapidement qu'une recherche exhaustive (V-BLAST).

Le choix d'une technique repose sur deux critères: en termes de taux d'erreur, les performances doivent être aussi faibles que possible par rapport à la meilleure détection, et le mécanisme doit être le plus complexe possible.

II.2.5.1 Critère de forçage à zéro (ZF)

Le zéro forcing c'est le récepteur linéaire non adaptative le plus simple basé sur le critère de forçage à zéro c'est-à-dire il force la réponse impulsionnelle à être nulle à tout instant sauf à l'instant zéro. C'est un filtre qui consiste à inverser exactement la fonction de transfert du canal H, il a été proposé pour assurer la suppression des interférences inter-symbole (ISI) et séparer les symboles transmis dans un même canal d'une manière parfaite. Le forçage à zéro peut provoquer une amplification du bruit si la valeur singulière minimale de H est trop petite. Supposant que H est inversible, le vecteur de symboles estimés s'effectue en moyen de l'équation suivante :

$$\hat{s} = H^{-1}r \tag{II.11}$$

Où : r désigne le vecteur reçu.

Lorsque H est mal conditionné, le récepteur ZF donne des résultats corrects pour des SNR élevés, mais pour de faibles SNR le bruit perturbe fortement les performances. Le principal avantage de ZF est sa simplicité. Malheureusement, cette détection ne prend pas en considération la variance du bruit [27], ce qui limite les performances de ce critère. Pour remédier à ce problème, l'erreur quadratique moyenne est minimisée.

II.2.5.2 Détecteur par la minimisation d'erreur quadratique moyenne (MMSE detector)

La technique MMSE (Minimum Mean Square Error) est directement basée sur le ZF que nous venons de voir. Tout comme cette dernière, elle est linéaire et non adaptative. Le but de l'égaliseur linéaire à erreur quadratique minimale est de minimiser l'erreur globale due au bruit et à l'interférence mutuelle des signaux transitant par le même canal. D'autre part, quand le bruit est négligeable ou bien le rapport $\frac{N_t}{SNR}$ tend vers zéro à haut SNR, l'égalisation MMSE converge vers le ZF. Son expression est :

$$\hat{s} = (H^*H + \frac{N_t}{SNR})^{-1} H^*r \quad (\text{II.12})$$

Ce récepteur résiste mieux au bruit que le récepteur ZF, mais ne sépare pas parfaitement les sous canaux [27].

II.2.5.3 Récepteur à Maximum vraisemblance (ML ou MV)

Le récepteur à maximum de vraisemblance (ML) est un récepteur non-linéaire qui offre les meilleures performances en termes de taux d'erreurs binaire (BER). Il est basé sur le critère de minimisation de la puissance de bruit sur le vecteur reçu "r". Cependant, ce récepteur présente également la plus grande complexité et augmente de façon exponentielle avec le nombre d'antennes d'émission. Si les vecteurs émis "s" sont équiprobables, ce qui est le cas puisque les symboles s_i sont équiprobables, Donc le récepteur est optimal. En supposant un bruit gaussien et un état de canal parfaitement connu côté récepteur, le récepteur MV donne les symboles estimés \hat{s} grâce à la formule [28]:

$$\hat{s} = \arg \min \|y - Hs\|^2 \quad (\text{II.13})$$

Où la recherche de la distance minimale s'effectue sur tous les mots de code s possibles.

II.2.5.4 Annulations successives d'interférences ordonnées (V-BLAST)

Le V-BLAST est une technique non linéaire et adaptative qui a été développée en 1998, il s'agit d'un égaliseur à retour de décision, adapté à la structure des systèmes MIMO. L'égaliseur peut utiliser soit le critère du forçage à zéro, soit le critère du MMSE.

Son principe est le suivant : le symbole de l'émetteur le plus favorisé (possédant le meilleur TEB (Taux d'erreur binaire) suivant le critère considéré) est démodulé en premier. Sa contribution au

vecteur reçu r est ensuite annulée, ce qui augmente le SNR sur les autres émetteurs (à chaque bonne décision). Cette étape est répétée jusqu'au dernier émetteur, le moins favorisé. Ce récepteur est aussi noté dans la littérature OSIC (Ordered Successive Interference Cancellation) [28]. Comme pour tous les égaliseurs à retour de décision, Nous pouvons constater que cette technique comporte tout de même un certain risque, son principal inconvénient est la propagation des erreurs. Une fois qu'une mauvaise décision a été prise sur la valeur d'un symbole, une mauvaise contribution est retirée au vecteur r , ce qui entraîne que les symboles suivants seront presque certainement mal estimés. C'est la raison pour laquelle l'ordonnancement est utilisé pour minimiser la propagation des erreurs.

II.3 Systèmes SM-MIMO

Les systèmes à entrées multiples et sorties multiples (MIMO) offrent une augmentation significative de l'efficacité spectrale (SE) par rapport aux systèmes à antenne unique, Les avantages de l'utilisation d'antennes multiples à l'émission et à la réception sont exploités dans de nombreux systèmes de communication. Pour améliorer les performances des systèmes MIMO ainsi que le rendement de l'efficacité énergétique (EE) et l'efficacité spectrale des systèmes MIMO, un nouveau concept de transmission a ainsi été associé à ces systèmes, c'est la modulation spatiale (SM) qui a induit de nouvelles structures d'antennes plus compactes et reconfigurables tout en ne conservant qu'une seule chaîne de transmission radio (RF) pour assurer une faible complexité [29].

II.3.1 La modulation spatiale (SM)

La modulation spatiale (SM), une technique de transmission récemment développée qui utilise des antennes multiples, permet de surmonter les inconvénients des systèmes MIMO. Elle a apparue au début des années 2000, comme un concept de transmission prometteur pour les dispositifs à faible complexité et à faible consommation d'énergie, et consiste à exploiter l'indice des antennes d'émission ou des antennes de réception pour transmettre des bits d'informations.

II.3.2 Principe de fonctionnement de la modulation spatiale (SM)

Le concept principal de la modulation spatiale est de transmettre les données en utilisant l'indice des antennes. Elle exploite la position des antennes d'émission et les symboles de constellation pour transmettre des informations via le canal MIMO. Le principe de base est de faire correspondre un bloc de bits d'information à deux unités porteuses d'information, contenant chacune $\log_2(N_t) + \log_2(M)$ bits, $\log_2(N_t)$ et $\log_2(M)$ étant le nombre de bits nécessaires pour identifier une antenne d'émission dans le réseau d'antennes et un symbole dans le diagramme de constellation du signal, respectivement, où le premier, un numéro d'antenne d'émission unique choisi dans un ensemble d'antennes d'émission, et le second, un symbole choisi dans un diagramme

complexe de constellation de signaux (PSK/QAM) [30]. L'utilisation du numéro d'antenne d'émission comme unité porteuse d'information augmente l'efficacité spectrale globale (par $\log_2(N_t)$). À un moment donné, une seule antenne d'émission est active à partir de laquelle le symbole est transmis implicitement (ce qui signifie qu'une seule chaîne RF est nécessaire, réduisant ainsi considérablement la complexité matérielle du système), tandis que toutes les autres antennes d'émission envoient une puissance nulle. Ces deux parties dépendent des bits de données entrants. Au niveau du récepteur, un algorithme de combinaison de taux de réception est utilisé pour récupérer le bloc de bits d'information transmis [31]. Pour cela on utilise la modulation SM avec une transmission OFDM afin d'effectuer un codage de données et analyser le taux d'erreur des symboles (BER).

II.3.3 SM en émission

Un émetteur est constitué de deux blocs d'entrée, le premier bloc de $\log_2(N_t)$ bits détermine l'indice de l'antenne active j du domaine spatial et le deuxième bloc de $\log_2(M)$ bits est utilisé pour moduler le symbole de la constellation s (voir la figure II.6) [32].

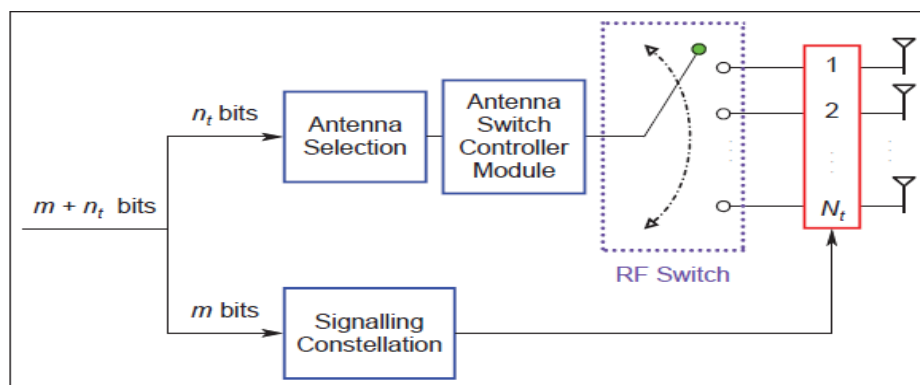


Figure II.6 : Schéma de principe de l'émetteur SM

Les bits d'information sont modulés sur un diagramme de constellation tridimensionnel (3D) fourni par le réseau d'antennes, (où certains bits sont mis en correspondance avec les indices TA (Transmission Antenna)), qui généralise le diagramme de constellation du signal bidimensionnel (complexe) connu des schémas de modulation PSK/QAM. La nouvelle constellation ainsi obtenue est souvent appelée « diagramme de constellation spatiale ». La figure II.8 présente un exemple d'un tel diagramme avec $N_t = 4$ antennes et une constellation $M = 4$ en modulation par déplacement de phase en quadrature (MDPQ) [32].

À l'émission, le codeur traite les bits d'information par blocs de quatre bits chacun. Dans la première utilisation du canal illustrée à la figure II.8, le bloc de bits à coder est "1110". Les deux premiers bits, "11," déterminent l'unique antenne d'émission (TA) active 'TX3' tandis que les deux seconds bits, "10," déterminent le symbole PSK/QAM émis. La TA activée peut changer à chaque

utilisation de canal en fonction des bits d'information d'entrée. Ainsi, la commutation TA est un moyen efficace de faire correspondre les bits d'information aux indices TA et d'augmenter le débit de transmission.

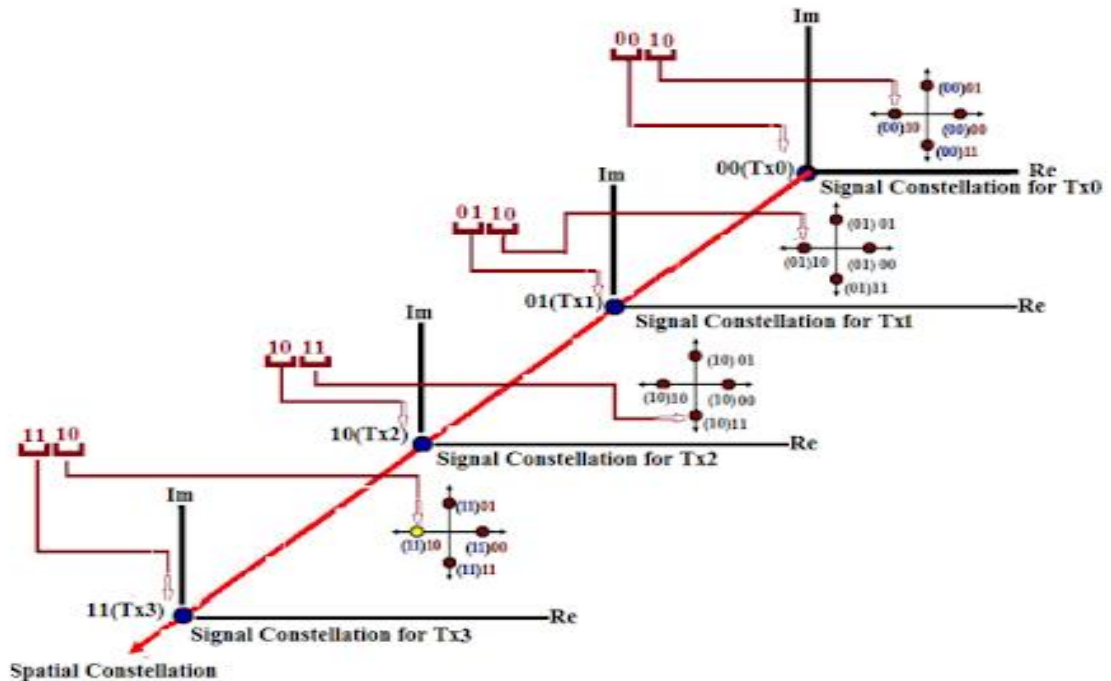


Figure II.8 : Schéma conceptuel du diagramme spatial 3-D.

Un tableau exemplaire illustrant la correspondance SM pour une modulation 4-QAM :

Bits	Bit de symbole	Symboles	Bits spatiaux	Index d'antenne
0000	00	+1+j	00	1
0001	01	-1+j	00	1
0010	10	-1-j	00	1
0011	11	+1-j	00	1
0100	00	+1+j	01	2
0101	01	-1+j	01	2
0110	10	-1-j	01	2
0111	11	+1-j	01	2
1000	00	+1+j	10	3
1001	01	-1+j	10	3
1010	10	-1-j	10	3
1011	11	+1-j	10	3
1100	00	+1+j	11	4
1101	01	-1+j	11	4

1110	10	-1-j	11	4
1111	11	+1-j	11	4

Tableau II.1 : Un tableau de correspondance SM avec une modulation 4-QAM.

II.3.4 SM en réception

Le récepteur du schéma SM doit gérer les deux tâches principales suivantes : détection de l'antenne d'émission active, et détection du symbole de données transmis par cette antenne pour la démodulation des bits mappés en constellation de symboles M-aires [29]. L'émetteur effectue un prétraitement afin de focaliser le signal vers l'antenne ciblée. Ensuite, le récepteur détecte l'antenne ciblée parmi ses N_r antennes, et l'indice d'antenne estimé est converti en valeurs binaires préétablie correspondant à une table de correspondance spatiale prédéfinie. Cette approche permet une démodulation de faible complexité, mais elle est sous-optimale et est affectée par une sensibilité élevée à la démodulation de l'indice TA.

Afin de surmonter ces limitations, les chercheurs ont proposé un démodulateur optimal ML à faible complexité de calcul, qui permet de décoder conjointement l'indice TA et les symboles de constellation transmis. Ce détecteur optimal doit rechercher conjointement toutes les antennes d'émission et tous les symboles de constellation pour décoder les bits. La différence de performance entre la démodulation en deux étapes et la démodulation ML optimale est que même si l'index TA et les symboles sont codés indépendamment dans SM-MIMO, ils seront effacés ensemble pendant la transmission du canal. Par conséquent, toute tentative d'effectuer une démodulation indépendante entraînera une perte de performances non négligeable.

II.3.5 Avantages et inconvénients de la modulation spatiale (SM)

Par rapport aux autres techniques MIMO, le SM présente de nombreux avantages, notamment qui sont :

- La SM permet d'obtenir un gain de multiplexage tout en maintenant l'absence d'ICI [33], puisqu'une seule antenne est active à la fois ;
- La synchronisation des antennes d'émission n'est pas nécessaire ;
- La complexité et le coût de l'émetteur sont donc considérablement réduits du fait qu'une seule chaîne RF est utilisée..
- SM offre une flexibilité intrinsèque pour échanger le nombre d'antennes d'émission avec l'ordre de modulation dans le domaine du signal pour atteindre le débit de données souhaité.
- SM peut fonctionner efficacement si $N_r < N_t$ puisque les antennes de réception sont utilisées pour obtenir seulement un gain de diversité [34].

Cependant, ces avantages apportent des compromis de base qui doivent être surmontés.

- La Modulation spatiale est très sensible aux canaux imparfaits et corrélés ;
- Le récepteur doit connaître parfaitement le canal pour détecter les données ;
- Le nombre de bits mappés dans la SM est limité par la taille physique de l'appareil connecté au réseau sans fil ;
- Au moins deux antennes d'émission sont nécessaires pour exploiter le concept SM.

II.4 SM-OFDM (Spatial Modulation-Orthogonal Frequency Division Multiplexing)

L'OFDM (orthogonal frequency division multiplexing) est né dans les années 50-60, c'est un procédé de codage de signaux numériques par répartition en fréquences orthogonales qui consiste à transmettre les données de manière simultanée sur N porteuses modulées, utilisée dans des applications sans fil pour sa résistance aux bruits impulsifs, aux canaux multi-trajets et les canaux sélectifs en fréquence plusieurs

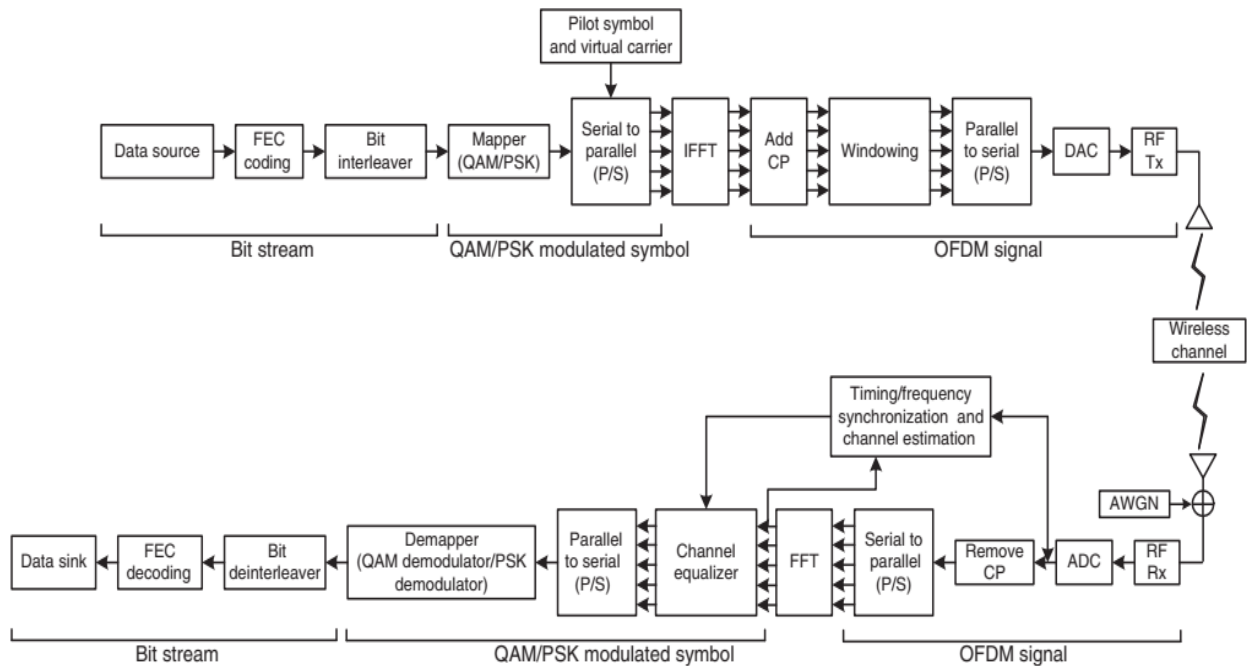


Figure II.9: Diagramme-bloc de la chaîne de transmission OFDM

La figure ci-dessus illustre le diagramme-bloc d'un système de communication utilisant l'OFDM, après un codage Bit/Symboles des bits à transmettre, Ces bits seront modulés où le modulateur génère des symboles complexes. Ces symboles sont successivement regroupés par blocs de N (conversion série/parallèle), puis ils sont utilisés comme entrées du bloc IFFT qui amène le signal dans le domaine temporel. Le bloc IFFT prend ces N symboles à la fois (N étant le nombre de sous-porteuses dans le système) où chaque symbole d'entrée a une durée appelée période symbole. Après le passage au domaine temporel, vient l'étape de l'insertion d'un intervalle de garde au début du bloc où la fin du bloc est y recopiée, qu'on nomme préfixe cyclique (CP) servant à combattre l'effet

de dispersion. Il faut vérifier que la longueur de l'intervalle de garde est plus grande que la dispersion maximale causée par le canal de telle sorte à éviter l'apparition de l'ISI.

À la réception, le signal est ramené en bande de base, l'intervalle de garde CP est supprimé. Le bloc FFT est utilisé pour traiter le signal reçu et le convertir en spectre fréquentiel pour récupérer les sous-porteuses. Après une reconversion parallèle/série, une dernière étape de démodulation des sous-porteuses est effectuée.

Le multiplexage par répartition en fréquence orthogonale à modulation spatiale (SM-OFDM) est une nouvelle technique OFDM qui transmet des informations supplémentaires par les indices d'antenne actifs des sous-porteuses, où chaque symbole est véhiculé par une sous-porteuse accompagnée par un symbole spatial, de sorte que pour chaque sous-porteuse, il n'y a qu'un seul émetteur actif. Cela procure une robustesse contre les interférences inter-antennes dans les sous-porteuses qui permet d'améliorer le débit de données.

II.5 Conclusion

Dans ce chapitre, nous avons décrit la technologie MIMO de déploiement de réseaux d'antennes à l'émission et à la réception afin d'améliorer la qualité du rapport, le débit de transmission et donc une meilleure qualité de service. Nous avons aussi explicité un schéma modulation spatiale qui améliore les performances des systèmes MIMO avec une faible complexité.

Chapitre III

Les Séparateurs à Vaste Marge (SVM)

III.1 Introduction

Au cours des années 1990, un nouveau type d'algorithme d'apprentissage a été mis au point, sur la base des résultats de la théorie de l'apprentissage statistique de Vladimir Vapnik [5] les séparateurs à vaste marge (SVM : Support Vector Machines) qui permettent une classification binaire optimale par apprentissage supervisé. Ces machines d'apprentissage théoriquement intelligentes utilisent un concept central des SVM, les fonctions noyaux, pour des tâches d'apprentissage pouvant être complexes. Ces machines fournissent un cadre modulaire qui peut être adapté à différentes tâches et domaines par le choix de la fonction noyau et de l'algorithme de base. Elles surclassent les réseaux de neurones dans divers domaines de recherche et d'ingénierie tels que le diagnostic médical, le marketing, la biologie, la reconnaissance de formes et de caractères manuscrits.

Le présent chapitre introduit les fondements mathématiques des SVMs, basés sur l'apprentissage statistique, leurs différentes formes et leur optimisation.

III.2 Apprentissage statistique

L'apprentissage statistique est un champ d'étude de l'intelligence artificielle fondé sur des approches mathématiques et statistiques pour donner aux ordinateurs la capacité d'apprendre à partir des données. Ce processus est réalisé par un apprenant dans l'objectif de généraliser la fonction apprise à de futurs exemples inconnus (exemples d'entraînement) [35].

En commençant à sélectionner et à préparer un ensemble de données d'entraînement. Ces données seront utilisées pour nourrir le modèle de Machine Learning pour apprendre à résoudre le problème pour lequel il est conçu. Ensuite, il faut sélectionner un algorithme à exécuter sur l'ensemble de données d'entraînement. Le type d'algorithme à utiliser dépend du type et du volume de données d'entraînement et du type de problème à résoudre.

On passe après à l'entraînement de l'algorithme, il s'agit d'un processus itératif. Des variables sont exécutées à travers l'algorithme, et les résultats sont comparés avec ceux qu'il aurait dû produire. « Les poids » et « le biais » peuvent ensuite être ajustés pour accroître la précision du résultat. On exécute ensuite de nouveau les variables jusqu'à ce que l'algorithme produise le résultat correct la plupart du temps. L'algorithme, ainsi entraîné, est le modèle de Machine Learning. On finit par l'utilisation et l'amélioration du modèle où on utilise le modèle sur de nouvelles données, dont la provenance dépend du problème à résoudre. Par exemple, un modèle de Machine Learning conçu pour détecter les spams sera utilisé sur des emails. On distingue trois types d'apprentissage :

III.2.1 Apprentissage supervisé (supervised Learning)

C'est un apprentissage en présence d'un superviseur (professeur, expert). Autrement dit, les données utilisées pour l'entraînement sont déjà « étiquetées » [36]. Par conséquent, le modèle d'apprentissage sait déjà ce qu'il doit chercher (motif, élément...) dans ces données d'appartenance fournies en couples (entrée x , sortie y). Parmi les algorithmes supervisés, on distingue **les algorithmes de classification** (prédictions d'une variable qualitative) et **les algorithmes de régression** (prédictions d'une variable quantitative). En fonction du problème à résoudre, on utilisera l'un de ces deux archétypes.

1) **La classification** : Est un type d'apprentissage supervisé qui consiste à attribuer automatiquement une catégorie (ou une classe) à des données dont on ne connaît pas la catégorie à partir d'exemples de données étiquetées. On se basant sur la minimisation du nombre d'erreurs de classification.

- **L'objectif de la classification supervisée** : Est principalement de définir des règles permettant de classer des objets (entrées) en classes (catégories) à partir de variables qualitatives ou quantitatives caractérisant ces objets. On dispose au départ d'un échantillon dit d'apprentissage dont le classement est connu, cet échantillon est utilisé pour l'apprentissage des règles de classement. Il est nécessaire d'étudier la fiabilité de ces règles pour les comparer et les appliquer, évaluer les cas de sous-apprentissage ou de sur-apprentissage (complexité du modèle). On utilise souvent un deuxième échantillon indépendant, dit de validation ou de test.
- **Fonctionnement**: On va commencer par soumettre au classifieur les informations des entrées et qui en sortie nous fournira les catégories des données. Pour atteindre notre objectif, le classifieur doit être une fonction ou un modèle mathématique auquel, quand on lui soumettra les entrées, il produira la sortie la plus proche que possible de celle désirée. Il est impossible de construire une fonction généraliste qui fonctionnerait pour toutes les entrées et pour toutes les sorties. C'est pourquoi il va falloir construire cette fonction automatiquement en utilisant les données d'entrée et de sortie d'exemples ou d'échantillons que l'on a du problème que l'on souhaite résoudre. En clair, nous allons procéder à ce qu'on appelle couramment en machine Learning à l'entraînement du classifieur. Cette procédure se déroule comme suit :

À partir des données que l'on a du problème, deux lots sont constitués aléatoirement:

- Un lot d'apprentissage que l'on nommera base d'apprentissage et un lot de test que l'on nommera base de test.

- Le premier lot servira à entraîner le classifieur tandis que le second servira à tester le classifieur.

Lors de l'entraînement, les exemples sont soumis un à un dans un ordre aléatoire au classifieur, pour chaque exemple soumis, deux cas sont possibles.

- Le premier cas, le classifieur trouve la réponse correcte, on passe alors au suivant.
- Le second cas où la réponse du classifieur est incorrecte. Dans ce cas, l'algorithme d'entraînement répare le classifieur afin qu'il sorte la bonne réponse et l'on peut alors passer à l'exemple suivant. Après cette phase dite d'entraînement du classifieur, vient alors une phase de test. Durant cette phase, des exemples qui n'ont jamais été vus par le classifieur durant son entraînement, lui sont soumis. Si les réponses du classifieur sont correctes, l'on considère qu'il a été bien entraîné, à l'inverse le classifieur est mal entraîné et il est alors important d'en trouver la cause.

2) La régression : Est l'une des techniques d'apprentissage supervisé les plus populaires dans la prédiction d'une valeur continue. On cherche à apprendre une fonction à partir d'un ensemble de points et des valeurs qu'elle prend sur ces points. La réponse est réelle et on cherche la fonction optimale au sens des moindres carrés, si le modèle est très fort pour capturer des relations linéaires entre variables explicatives et variable à expliquer, surtout grâce à ses variantes (version avec régularisation pour éviter le sur-apprentissage). Il peut être non performant quand il y'a des relations plus complexes qu'une simple linéarité entre les variables. Par exemple, on peut utiliser ce modèle pour prédire le prix d'une maison sachant sa taille, le nombre de chambres et le lieu où elle se trouve.

III.2.2 Apprentissage non supervisé (Unsupervised Learning)

C'est le cas inverse où l'entraînement se fait sur des données sans étiquettes. Dans ce cas, il parcourt les données sans aucun indice, et tente d'y découvrir des motifs ou des tendances récurrentes. Parmi les modèles non-supervisés, on distingue **les algorithmes de clustering** (pour trouver des groupes d'objets similaires), **d'association** (pour trouver des liens entre des objets) et **de réduction dimensionnelle** (pour choisir ou extraire des caractéristiques).

III.2.3 Apprentissage par renforcement

Dans ce cas d'apprentissage, l'algorithme apprend en essayant plusieurs fois d'atteindre un objectif précis. Il pourra essayer toutes sortes de techniques pour y parvenir. Le modèle est récompensé s'il s'approche du but, ou pénalisé s'il échoue.

Dans ce qui suit, nous nous plaçons dans un contexte d'apprentissage supervisé, l'objectif de notre travail étant d'utiliser les séparateurs à vaste marge pour la classification de type de modulation.

III.2.4 Notions d'apprentissage automatique (ML : Maching Learning)

- **Apprenant** : C'est la machine ou système d'apprentissage construisant le processus d'induction (plus précisément c'est l'algorithme d'apprentissage utilisé) [36].
- **Données d'apprentissage** : Ce sont les entrées X issus d'une population et formant un ensemble ou échantillon d'exemples, de taille finie. Ces données sont des vecteurs générés dans \mathbb{R}^d (ayant d attributs ou caractéristiques) indépendamment les uns des autres selon une loi de probabilité fixée mais inconnue $P(x)$. Un oracle attribue à chaque entrée x une valeur y , dite sortie désirée (une étiquette, un label, une catégorie, ou parfois un concept), selon une loi $P(y/x)$ également fixée mais inconnue. Une base d'apprentissage est donc l'ensemble $\chi_m \subset X$, de m observations ou réalisations :

$$\chi_m = \{(x_1, y_1), (x_2, y_2), \dots (x_n, y_n)\} \quad \text{(III.1)}$$

Indépendantes et identiquement distribuées (i.i.d.) selon la loi jointe :

$$p(x, y) = p(x) \cdot p(y|x) \quad \text{(III.2)}$$

- **Fonction cible** : C'est la fonction d'apprentissage. En pratique, elle est approchée par une fonction f_{a^*} choisie selon un critère d'induction dans un espace d'hypothèses (une famille de fonctions implémentables ou modèles d'apprentissage) $F_a = \{f(x, a) / a \in A\}$ où A est un ensemble de paramètres abstraits, pour fournir la meilleure approximation de la réponse y générée par l'oracle [36].
- **Critère d'induction**: c'est la règle d'inférence à partir de l'échantillon d'apprentissage. Ce critère est basé sur l'idée de minimisation d'une fonctionnelle de risque, $R(a)$, définie comme l'espérance d'une certaine fonction de décision (fonction de perte ou mesure de désaccord) $D(y, f(x, a))$:

$$R(\alpha) = \int D(y, f(x, \alpha)) \cdot dP(x, y) \quad \text{(III.3)}$$

Une autre formulation consiste à exprimer $R(a)$ par un ensemble $\{Q(f(x, y), a) / a \in A\}$ de fonctions de coût :

$$R(\alpha) = \int Q(y, f(x, \alpha)) \cdot dP(f(x, y)) \quad \text{(III.4)}$$

Où la mesure de probabilité $P(f(x, y))$ est inconnue. L'objectif de l'apprentissage est donc de trouver une fonction $Q(f(x, y), \alpha^*)$ qui minimise le risque réel $R(\alpha)$. Or seule une base finie d'exemples $\{(x_i, y_i)\}_{1 \leq i \leq m}$ est donnée selon un tirage i.i.d. respectant $P(f(x, y))$ (la taille de la population X qui est très grande, voire infinie). La seule solution consiste alors à trouver une fonction $Q(f(x, y), \alpha_m)$ qui minimise le risque empirique défini sur χ_m :

$$R_{\text{emp}}(a) = \frac{1}{m} \sum_{i=1}^m Q(f(x_i, y_i), a) \quad \text{(III.5)}$$

C'est seulement sur la validité de ce principe, dit de minimisation du risque empirique (MRE), que la règle d'inférence, déduite de l'échantillon χ_m , peut être généralisée à l'ensemble de la population.

- **Généralisation** : C'est la capacité d'un apprenant à appliquer, lors d'une phase de test (donc sur une base de test ou ensemble de nouveaux exemples qui lui sont inconnus), la règle d'inférence déduite du processus d'apprentissage. Le pouvoir de généralisation exprime la pertinence du principe inductif utilisé et de ce fait l'optimalité du modèle d'apprentissage sélectionné [36].

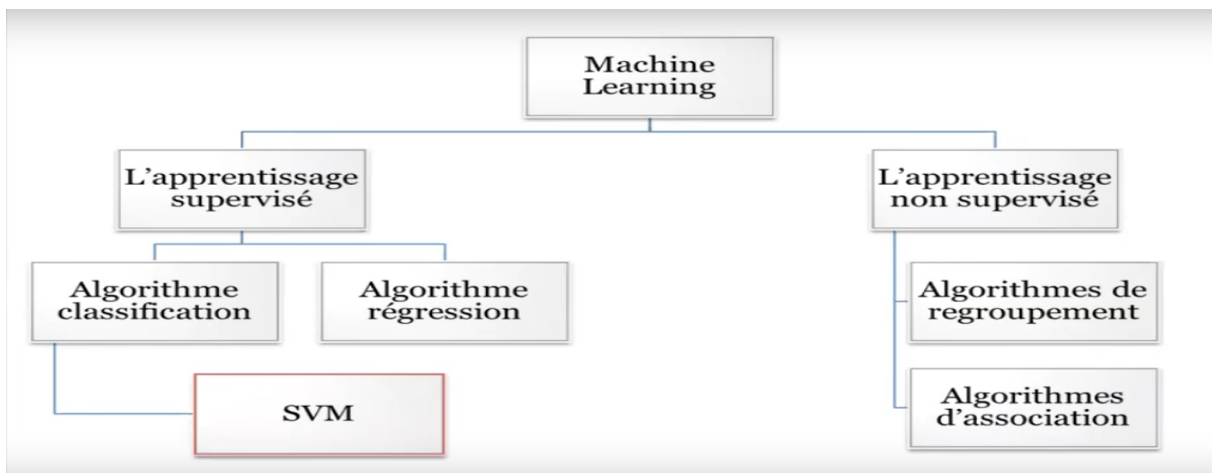


Figure III.1: Les types d'apprentissage statistique.

III.3 Les séparateurs à vaste marge (SVM)

Les SVMs sont un ensemble de techniques d'apprentissage supervisé, destinées à résoudre des problèmes de discrimination (c'est-à-dire décider à quelle classe appartient un échantillon), ou de régression (c'est-à-dire prédire la valeur numérique d'une variable), elles reposent sur une théorie mathématique solide à l'inverse des méthodes de réseaux de neurones. L'idée principale derrière ce classifieur revient à chercher une frontière de décision linéaire entre deux classes dans un espace approprié, l'hyperplan étant tracé de manière à maximiser la distance de chaque point à l'hyperplan, de sorte que cette frontière soit la plus discriminante possible, et que notre classification soit ainsi la plus pertinente possible. Pour des données non linéairement séparables, on utilise le « kernel trick » ou bien l'astuce du « noyau » qui consiste à projeter les données dans un autre espace de dimension supérieure permettant d'augmenter la séparabilité des données avec la marge la plus large [37].

Ci-dessous un exemple simple en dimension 2 (on considère que nos données sont décrites par seulement 2 features, nos vecteurs sont donc des points dans le plan). L'algorithme va chercher à séparer les points rouges et verts en traçant une droite optimale, qui offre une marge maximale (**noir**) entre les 2 classes.

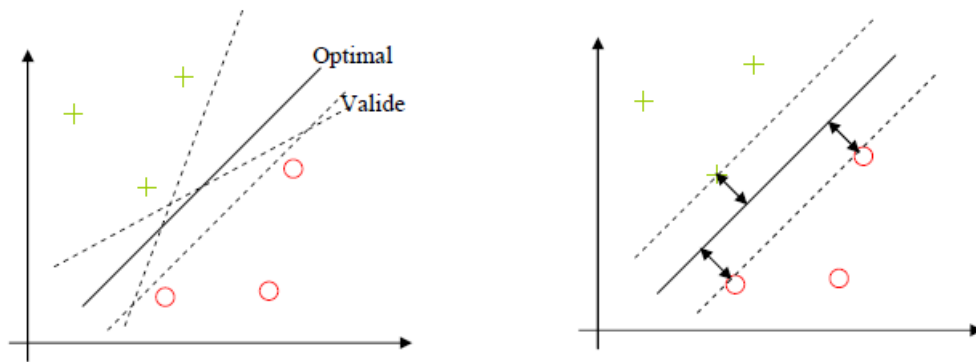


Figure III.2: exemple d'un séparateur à vaste marge.

Les points se trouvant sur les 2 droites formant la marge sont « les vecteurs supports », d'où le nom de l'algorithme.

III.3.1 Historique

Les séparateurs à vastes marges reposent sur deux idées clés : la notion de marge maximale et la notion de fonction noyau. Ces deux notions existaient depuis plusieurs années avant qu'elles ne soient mises en commun pour construire les SVM.

L'idée des hyperplans à marge maximale a été explorée dès 1963 par Vladimir Vapnik et A. Lerner, et en 1973 par Richard Duda et Peter Hart dans leur livre *Pattern Classification*. Les fondations théoriques des SVM ont été explorées par Vapnik et ses collègues dans les années 70 avec le développement de la Théorie de Vapnik-Chervonenkis, et par Valiant.

L'idée des fonctions noyaux n'est pas non plus nouvelle: le théorème de Mercer date de 1909, et l'utilité des fonctions noyaux dans le contexte de l'apprentissage artificiel, a été montrée dès 1964 par Aizermann, Bravermann et Rozoener.

Ce n'est toutefois qu'en 1992 que ces idées seront bien comprises et rassemblées par Boser, Guyon et Vapnik dans un article, qui est l'article fondateur des séparateurs à vaste marge. L'idée des variables ressorts, qui permet de résoudre certaines limitations pratiques importantes, ne sera introduite qu'en 1995. À partir de cette date, qui correspond à la publication du livre de Vapnik, les SVM gagnent en popularité et sont utilisés dans de nombreuses applications.

Un brevet américain sur les SVM est déposé en 1997 par les inventeurs originaux [38].

III.3.2 Types des SVM

Il existe deux types principaux des SVM :

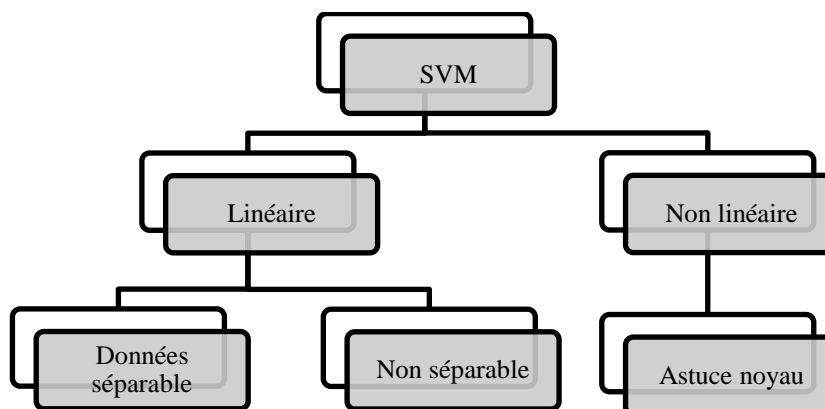


Figure III.3: types de SVM

III.3.2.1 SVM linéaire

Parmi les modèles des SVM linéaire, on constate les cas linéairement séparables et les cas non linéairement séparables. Les SVM linéaire sont utilisé pour les données linéairement séparables, ce qui signifie que si un ensemble de données peut être classé en deux classes à l'aide d'une seule ligne droite, ces données sont appelées données linéairement séparables [39].

- **Cas linéairement séparables :** C'est le cas le plus simple des SVM, car ils permettent de trouver facilement le classificateur linéaire. Dans ce cas on cherche l'hyperplan d'équation $\langle w, x \rangle + b = w^T x + b = 0$ tel que la **marge** (qui peut être vue comme la distance entre les observations les plus proches de l'hyperplan et l'hyperplan) soit maximale.
- **Cas linéairement non séparables :** Malheureusement, dans la plupart des cas, les données ne sont pas linéairement séparables. Il va donc falloir que nous acceptions de faire des erreurs, autrement dit que certains points de notre jeu d'entraînement se retrouvent du mauvais côté de la frontière de la zone d'indécision.

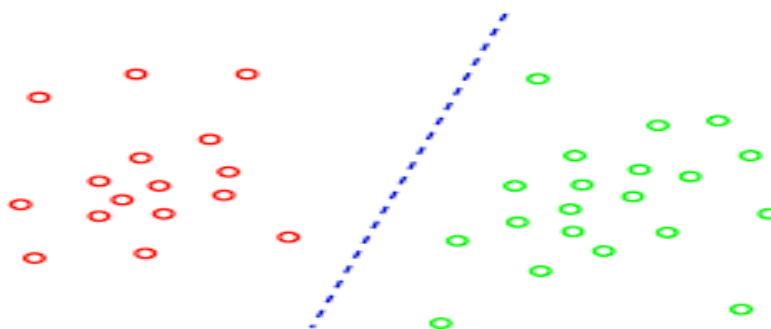


Figure III.4 : cas linéairement séparable.

III.3.2.2 SVM non linéaire

Lorsqu'un problème n'est pas linéairement séparable dans l'espace d'entrée, le SVM linéaire ne peut pas trouver un hyperplan de séparation robuste qui minimise le nombre de points de données mal classés et qui se généralise bien. Pour cela, on utilise **l'astuce du noyau (kernel trick)**, qui permet d'utiliser un **classifieur linéaire** pour résoudre un problème non linéaire. L'idée est de transformer l'espace de représentation des données d'entrée en un espace de plus grande dimension, dit « espace de redescription », à l'aide d'une fonction ϕ , où un classifieur linéaire peut être utilisé et obtenir de bonnes performances. La transformation d'espace inverse ϕ^{-1} , permet de retourner à l'espace d'origine après la résolution dans le nouvel espace. En effet, intuitivement, plus grande est la dimension de l'espace de redescription, plus élevée sera la probabilité de pouvoir trouver un hyperplan séparateur entre les exemples [35].

L'utilisation de l'astuce du noyau rend les SVM plus intéressants et surtout plus riches puisqu'on peut toujours chercher de nouveaux noyaux qui peuvent être mieux adaptés à la tâche qu'on veut accomplir. Les trois noyaux les plus utilisés sont : le noyau linéaire, le noyau polynomial et le noyau gaussien noté aussi RBF (Radial Basis Function).

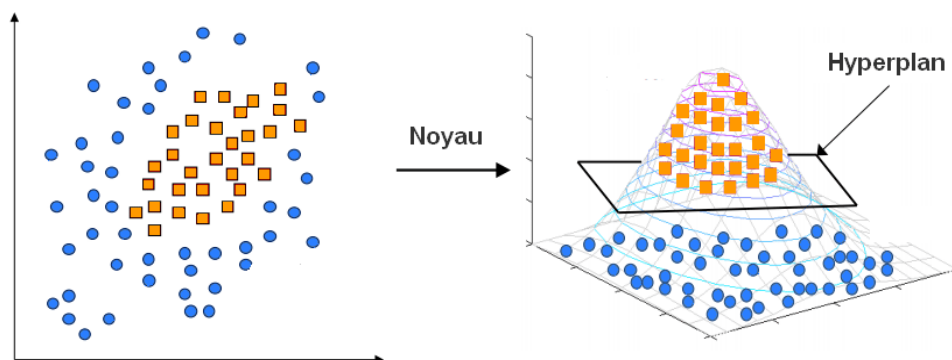


Figure III.5 : Illustration d'un SVM non-linéaire (kernel trick).

III.3.3 Les SVMs binaires

L'algorithme des SVM a pour objectif de trouver la séparation entre deux classes d'objets (+1 ou -1), avec l'idée que plus la séparation est large, plus la classification est robuste, on parle alors de classification binaire. Si les données sont linéairement séparables, il existe une infinité d'hyperplans séparateurs qui classifient correctement les données, l'algorithme sélectionne l'hyperplan qui sépare le jeu d'observations en deux classes distinctes de façon à maximiser la distance entre l'hyperplan et les observations les plus proches de l'échantillon d'apprentissage (voir figure III.6). Si un tel hyperplan existe, c'est-à-dire si les données sont linéairement séparables, on parle d'une machine à vecteurs supports à marge dure (Hard margin).

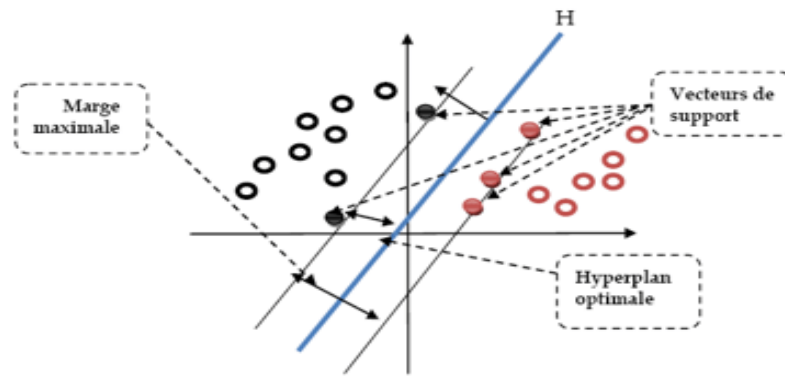


Figure III.6 : Hyperplan optimal, marge et vecteurs de support.

L'hyperplan $w \cdot x + b = 0$ représente un hyperplan séparateur H des deux classes, et la distance qui le sépare de l'exemple le plus proche s'appelle marge (la marge maximale), égale à $\gamma = \frac{2}{\|w\|}$. Cette marge est délimitée par 2 hyperplans H^+ et H^- parallèles à H, qui ont les équations $w \cdot x + b = 1$ et $w \cdot x + b = -1$ respectivement et qui se situe chacun à une distance $\frac{1}{\|w\|}$ de H. La marge entre ces deux derniers s'appelle la région de généralisation de la machine d'apprentissage, La maximisation de cette région est l'objectif de la phase d'entraînement qui consiste à rechercher l'hyperplan qui maximise la marge. Un tel hyperplan est appelé "hyperplan de séparation optimale ». Les points positifs les plus proches de H, situés sur H^+ , et les points négatifs les plus proches de H, situés sur H^- , s'appellent les **vecteurs de support** [40].

• SVM à marge dure

L'hyperplan séparateur est représenté par l'équation suivante :

$$H(x) = w^T x + b \tag{III.6}$$

Où w est un vecteur de m dimensions et b est un terme. La fonction de décision, pour un exemple x , peut être exprimée comme suit :

$$\begin{cases} \text{Classe} = 1 & \text{Si } H(x) > 0 \\ \text{Classe} = -1 & \text{Si } H(x) < 0 \end{cases}$$

Puisque les deux classes sont linéairement séparables, il n'existe aucun exemple qui se situe sur l'hyperplan, c.-à-d. qui satisfait $H(x)=0$. Il convient alors d'utiliser la fonction de décision suivante :

$$\begin{cases} \text{Classe} = 1 & \text{Si } H(x) > 1 \\ \text{Classe} = -1 & \text{Si } H(x) < -1 \end{cases}$$

Les valeurs +1 et -1 à droite des inégalités peuvent être des constantes quelconques +a et -a, mais en divisant les deux parties des inégalités par a, on trouve les inégalités précédentes

Qui sont équivalentes à l'équation suivante :

$$y_i(w^T x_i + b) \geq 1, i = 1 \dots n \tag{III.7}$$

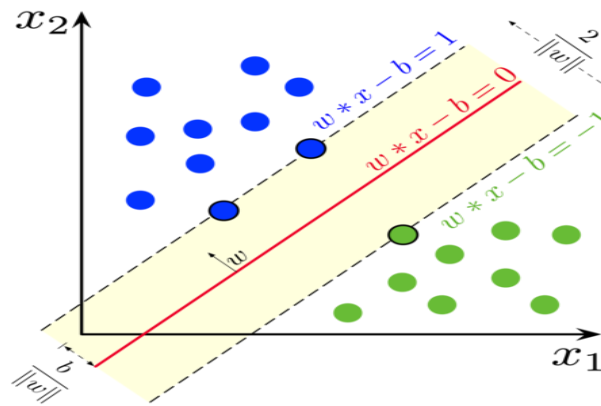


Figure III.7: SVM binaire.

La distance d'un point x_i à l'hyperplan (H) vaut : $d(x_i) = \frac{|\langle w, x_i \rangle + b|}{\|w\|}$.

Il ressort ainsi que la recherche de l'hyperplan optimal (H) revient à trouver ses paramètres (w, b) maximisant cette marge ($\frac{2}{\|w\|}$) ou bien minimisant le carré ($\frac{\|w\|^2}{2}$), sous les contraintes (III.7). Il s'agit là d'un problème d'optimisation d'une fonction objectif sous contraintes:

$$\begin{cases} \text{Min} \frac{1}{2} \|w\|^2 & \longleftarrow \text{Problème primal} \\ \text{Avec : } y_i(w, x_i + b) \geq 1 \forall i = 1 \dots m \end{cases} \quad \text{(III.8)}$$

Dans ce problème, les variables sont w et b , c.-à-d. que le nombre de variables est égal à $m+1$. Dans ce cas le problème est convertit en un problème dual.

Un problème dual est un problème fournissant la même solution que le primal mais dont la formulation est différente. On passe du problème primal au dual en introduisant les multiplicateurs de Lagrange pour chaque contrainte. Il s'agit de faire rentrer les contraintes dans la fonction objective et de pondérer chacune d'entre elles par une variable duale :

$$L(w, b, a) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m a_i (y_i (\langle w, x_i \rangle + b) - 1) \quad \text{(III.9)}$$

$$\text{Avec : } a_i \geq 0 \quad \forall i = 1 \dots m$$

Les variables duales a_i sont les multiplicateurs de Lagrange.

L'optimum de la fonction objective L peut être obtenu en la minimisant par rapport à w et b et en la maximisant par rapport aux a_i . À l'optimum de la fonction objective, ses dérivées par rapports aux variables w et b s'annulent ainsi que le produit des a_i aux contraintes, et des conditions dites de Karush, Kuhn et Tucker (KKT) :

$$\left\{ \begin{array}{l} \frac{\partial}{\partial w} L(w, b, a) = w - \sum_{i=0}^m a_i y_i x_i = 0 \quad \text{(III.10)} \\ \frac{\partial}{\partial b} L(w, b, a) \sum_{i=0}^m a_i y_i = 0 \quad \text{(III.11)} \\ \frac{\partial}{\partial a} (L, b, a) = -y_i (< w, x_i > + b) + 1 \leq 0 \quad \text{(III.13)} \\ a_i [y_i (< w, x_i > + b) - 1] = 0 \quad \forall i = 1 \dots m \quad \text{(III.14)} \end{array} \right.$$

Nous verrons plus loin que la condition (III.14), dite condition KKT complémentaire, est très intéressante. Ces équations permettent de réexprimer le vecteur.

$$w = \sum_{i=1}^m a_i y_i x_i \quad \text{(III.15)}$$

Et de reformuler le Lagrangien sous une forme duale :

$$L(w, b, a) = L(a) = \sum_{i=1}^m a_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m a_i y_i a_j y_j < x_i, x_j > \quad \text{(III.16)}$$

Le problème dual d'optimisation est alors :

$$\left\{ \begin{array}{l} \max \sum_{i=0}^m a_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m a_i y_i a_j y_j < x_i, x_j > \\ \text{Avec : } \left\{ \begin{array}{l} \sum_{i=1}^m a_i y_i = 0 \\ a_i \geq 0 \quad \forall i = 1..m \end{array} \right. \end{array} \right. \quad \text{(III.17)}$$

La résolution du dual permet donc de calculer le vecteur w à moindre coût, cependant cette formulation ne fait à aucun moment apparaître le terme b . Pour calculer ce dernier nous devons utiliser les variables primales :

$$b = - \frac{\max_{y_i = -1} < w, x_i > + \max_{y_i = +1} (< w, x_i >)}{2} \quad \text{(III.18)}$$

- **SVM à marge souple**

Le classificateur de marge maximale ne peut pas être utilisé dans la plupart des problèmes réels : si les données ont été affectées par le bruit, il n'y a pas de séparation linéaire entre elles. Dans ce cas, le problème d'optimisation ne peut pas être résolu. Pour surmonter cette nouvelle contrainte, nous allons introduire une notion de « tolérance » faisant appel à une technique dite des **variables ressort** (*slack variables*).

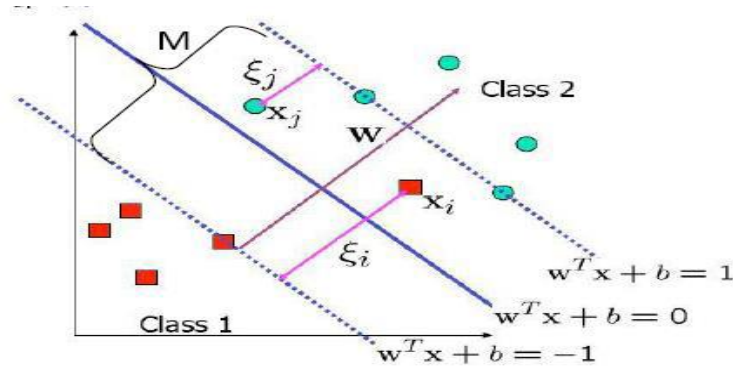


Figure III.8 : SVM à marge souple

Les points d'apprentissage mal classés (outliers) sont marqués par x_i et x_j

Le problème d'optimisation initial était :

$$\begin{cases} \text{Min} \frac{1}{2} \|w\|^2 \\ \text{Avec : } y_i(w, x_i + b) \geq 1 \forall i=1 \dots m \end{cases} \quad (\text{III.19})$$

La technique des variables ressort permet de construire un hyperplan en admettant des erreurs, mais en les minimisant, ce qui amène à assouplir les contraintes en introduisant les variables ressort $\xi_i \geq 0$ dans la définition des contraintes :

$$\begin{cases} y_i(w^T x_i + b) \geq +1 - \xi_i \text{ Si } y = +1 & (\text{III.20}) \\ y_i(w^T x_i + b) \geq -1 + \xi_i \text{ Si } y = -1 & (\text{III.21}) \\ y_i(w^T x_i + b) \geq 1 - \xi_i \text{ Si } \forall i=1..n & (\text{III.22}) \end{cases}$$

Quand une erreur de classification intervient, la variable ξ_i a une valeur plus grande que 1, donc $\sum_i \xi_i$ est une borne supérieure du nombre d'erreurs à l'apprentissage.

De là, un moyen naturel pour pénaliser les erreurs est de remplacer la fonction précédente à minimiser par :

$$\frac{1}{2} \|w\|^2 + \sum_{i=1}^n \xi_i \quad (\text{III.23})$$

Autrement dit, on cherche à maximiser la marge en s'autorisant pour chaque contrainte une erreur positive ξ_i la plus petite possible. Le coefficient C est défini comme un paramètre de régularisation, il donne un compromis entre la marge et le nombre d'erreurs admissibles. L'ajout du terme $C \sum_{i=1}^m \xi_i$ peut être considéré comme une mesure d'une certaine quantité de mauvaise classification. Ainsi, une faible valeur de C entraîne une faible tolérance [5].

En suivant la même démarche du Lagrangien que précédemment, nous aboutissons à la forme duale suivante :

$$L(w, b, \xi, a, \beta) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m \xi_i - \sum_{i=1}^m a_i (y_i \langle w, x_i \rangle + b - 1 + \xi_i) - \sum_{i=0}^m \beta_i \xi_i \quad (\text{III.23})$$

$$\left\{ \begin{array}{l} \frac{\partial}{\partial w} L(w, b, \xi, a, \beta) = w - \sum_{i=0}^m a_i y_i x_i = 0 \quad \text{(III.24)} \\ \frac{\partial}{\partial b} L(w, b, \xi, a, \beta) = \sum_{i=0}^m a_i y_i = 0 \quad \text{(III.25)} \\ \frac{\partial}{\partial \xi_i} L(w, b, \xi, a, \beta) = C - a_i - \beta_i = 0 \quad \text{(III.26)} \\ L_D = \sum_{i=0}^m a_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m a_i y_i a_j y_j \langle x_i, x_j \rangle \quad \text{(III.27)} \end{array} \right.$$

La seule différence avec le cas linéairement séparable est que :

$$C - \alpha_i - \beta_i = 0 \quad \text{(III.29)} \\ \beta_i \geq 0$$

$$\alpha_i \leq C \quad \forall i \in 1 \dots m$$

Le problème d'optimisation est équivalent au cas de la marge maximale, avec une contrainte additionnelle ($a_i \leq C$).

Cette formulation est connue comme « contrainte de boîte », car chaque valeur est limitée par 0 d'un côté et par C de l'autre. C s'est révélé être un compromis entre la précision et la régularisation (le contrôle de l'erreur).

Les conditions KKT sont désormais :

$$[y_i (\langle w, x_i \rangle + b) - 1 + \xi_i] = 0 \quad \text{(III.30)} \\ \beta_i \cdot \xi_i = (C - \alpha_i) = 0 \quad \forall i \in 1 \dots m$$

Ces conditions impliquent que les variables d'écart soient différentes de zéro quand $\alpha_i = C$ c'est à dire, quand leur marge est moins de $\frac{1}{\|w\|}$. Les vecteurs pour lesquels $0 \leq \alpha_i \leq C$: sont considérés comme vecteurs supports.

Il existe trois statuts différentiels :

- $\alpha_i = 0$: $\xi_i = 0$ et $y_i (\langle w, x_i \rangle + b) > 1$: Exemple bien classé, n'est pas un vecteur support ;
- $0 < \alpha_i < C$: $\xi_i = 0$ et $y_i (\langle w, x_i \rangle + b) = 1$: Exemple bien classé, sur hyperplan canonique.
 $\xi_i < 1$ et $y_i (\langle w, x_i \rangle + b) < 1$: Exemple bien classé mais dans la marge (VS)
- $\alpha_i = C$: $\xi_i > 1$ $y_i (\langle w, x_i \rangle + b) < 0$: Exemple mal classé (outlier)

III.3.4 Les SVMs multi-classes

Dans leur version la plus simple, les SVM sont appliqués à la classification binaire, cependant, les problèmes du monde réel sont dans la plupart des cas multi-classes. Diverses stratégies ont ainsi été mises au point pour résoudre le problème de multi-classification. Pour pouvoir traiter plus de deux classes il convient d'apporter les modifications nécessaires [41].

Les méthodes des machines à vecteur support multiclasse, simplifient le problème multiclasse à une composition de plusieurs hyperplans binaires permettant de tracer les frontières de décision entre les différentes classes [42]. Ces méthodes décomposent l'ensemble d'exemples en plusieurs sous-ensembles représentant chacun un problème de classification binaire. Pour chaque problème un hyperplan de séparation est déterminé par la méthode SVM binaire. On construit lors de la classification une hiérarchie des hyperplans binaires qui est parcourue de la racine jusqu'à une feuille pour décider de la classe d'un nouvel exemple. La généralisation dans le cas multi-classes peut se faire de deux façons différentes:

III.3.4.1 Un-contre-Reste (One-vs-Rest)

C'est la méthode la plus simple et la plus ancienne. Selon la formulation de Vapnik, elle consiste à déterminer pour chaque classe k un hyperplan $H_k(w_k, b_k)$ la séparant de toutes les autres classes. Cette classe k est considéré comme étant la classe positive (+1) et les autres classes comme étant la classe négative (-1), ce qui résulte, pour un problème à K classes, en K SVM binaires. Un hyperplan H_k est défini pour chaque classe k par la fonction de décision suivante :

$$H_k(x) = \text{sign}(\langle w_k, x \rangle + b_k) \quad \text{(III.31)}$$

$$= \begin{cases} +1 & \text{Si } f_k(x) > 0 \\ 0 & \text{sinon} \end{cases}$$

$f_k(x)$ indique la distance entre x et l'hyperplan qui sépare la classe k des autres. Plus cette valeur est élevée, plus nous sommes convaincus que x appartient à la classe k .

La valeur retournée de l'hyperplan permet de savoir si x appartient à la classe k ou non. Dans le cas où il n'appartient pas à k ($H_k(x) = 0$), nous n'avons aucune information sur l'appartenance de x aux autres classes. Pour le savoir, on présente x à tous les hyperplans, ce qui donne la fonction de décision de l'équation suivante :

$$k^* = \arg \max_{1 \leq k \leq K} (H_k(x)) \quad \text{(III.32)}$$

Si une seule valeur $H_k(x)$ est égale à 1 et toutes les autres sont égales à 0, on ne conclut que x appartient à la classe k . Le problème est que l'équation peut être vérifiée pour plus d'une classe, ce qui produit des régions d'ambiguïté, et l'exemple x est dit non classifiable. La figure III.9 représente un cas de séparation de 3 classes.

Pour surmonter cette situation, la méthode 1vsR utilise le principe de "le gagnant prend tout" ("winner-takes-all") : la classe k retenue est celle qui maximise $f_k(x) = \langle w_k, x \rangle + b_k$ de l'équation :

$$k^* = \operatorname{arg} \max_{1 \leq k \leq K} (w_k, x) + b_k \quad (\text{III.33})$$

La méthode 1vs R peut être utilisée pour découvrir même les cas de rejet où un exemple n'appartient à aucune des k classes (Figure III.9). Pour cela, on prend les deux fonctions de décision les plus élevées, puis on calcule leur différence, si elle est au-dessous d'un certain seuil, l'exemple est rejeté.

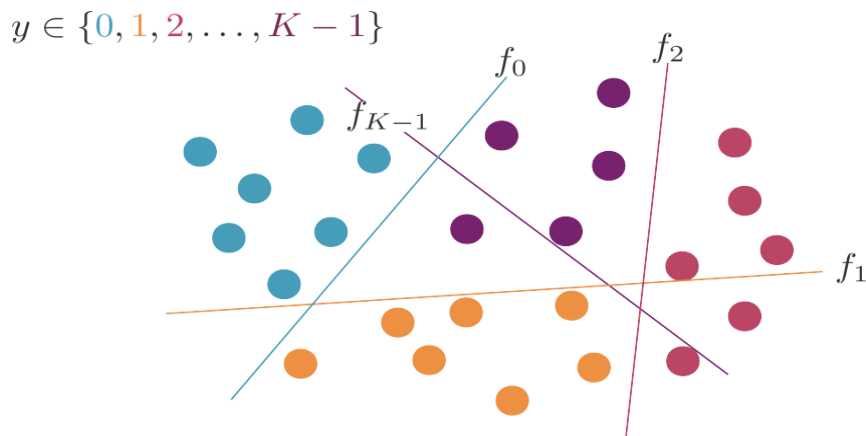


Figure III.9 : Résolution des cas d'indécision dans la méthode 1vs R (chaque droite sépare - parfois avec des erreurs - les points d'une classe de tous les autres).

III.3.4.2 Un-contre-un (one-vs-one)

L'approche Un-Contre-Un est un cas spécial des méthodes de décomposition proposées par Dietterich et al [43] pour résoudre des problèmes à plusieurs classes. Elle est appelée aussi "pairwise". Elle consiste à utiliser un classifieur pour chaque paire de classes. Pour k classe on aura $\frac{k(k-1)}{2}$ classifieur binaire (le nombre de fonctions de décision est égale aux nombre de classifieurs), le nombre des séparateurs est habituellement plus grand a celui des séparateurs sur la méthode un contre tous, par exemple pour un $k=10$ on a besoin de 45 classifieurs binaires, chaque classe effectue un vote pour affecter un point x , la classe majoritaire après vote sera celle à qui le point est affecté.

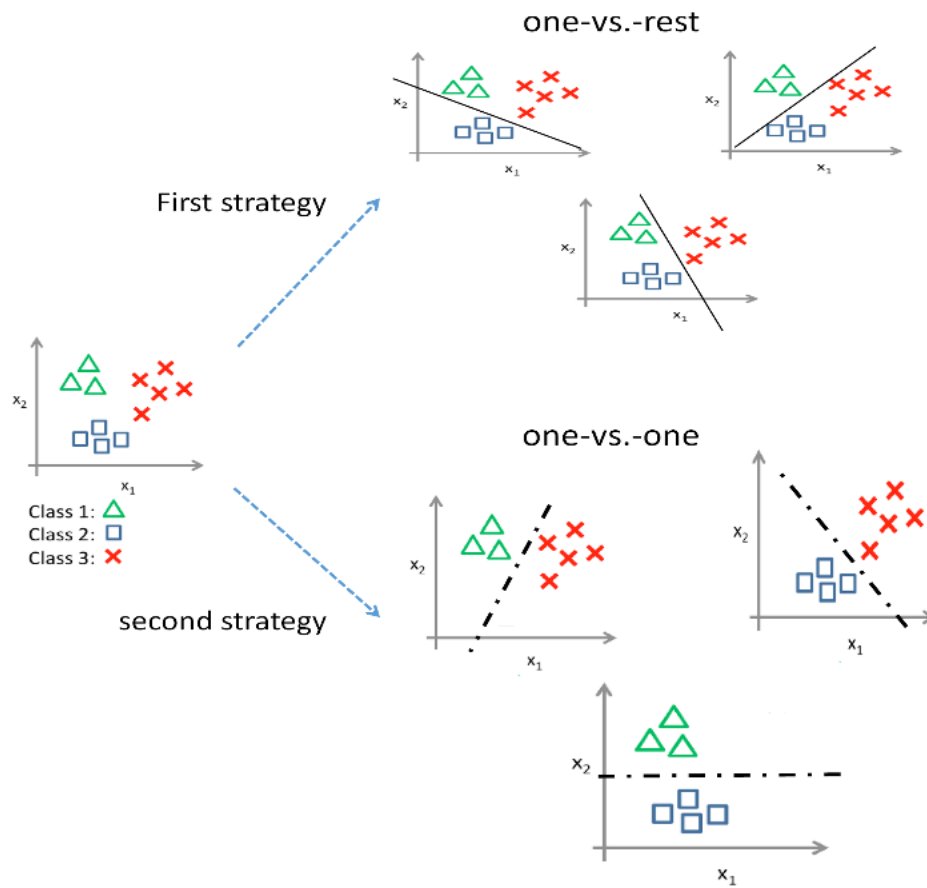


Figure III.10 : cas d'un SVM multiclasse.

III.3.5 Architecture générale d'une machine à vecteurs supports

Une machine à vecteur support, recherche à l'aide d'une méthode d'optimisation, dans un ensemble d'exemples d'entraînement, des exemples, appelés vecteurs support, qui caractérisent la fonction de séparation. La machine calcule également des multiplicateurs associés à ces vecteurs. Les vecteurs supports et leurs multiplicateurs sont utilisés pour calculer la fonction de décision pour un nouvel exemple. Le schéma de la figure 2.13 résume l'architecture générale d'une SVM dans le cas de la reconnaissance des chiffres manuscrits.

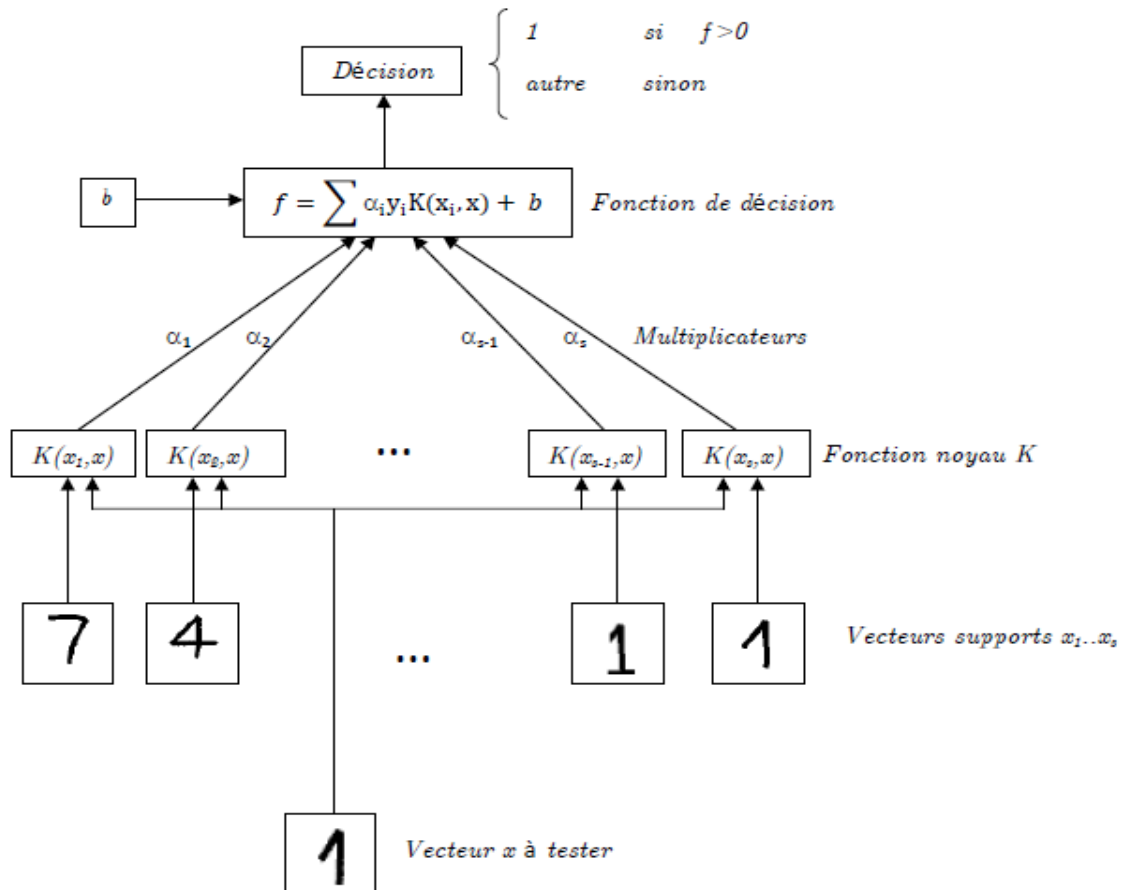


Figure III.10 : exemple d'une Architecture d'une machine à vecteur support

La fonction noyau K est utilisée pour calculer la distance entre le vecteur à tester x et chaque vecteur support dans l'espace de caractéristique. Les résultats sont ensuite linéairement combinés en utilisant les multiplicateurs de Lagrange α_i et ajoutés au biais b. Le résultat final f permet de décider à propos du nouveau vecteur : si $f(x)$ est positive, il s'agit du chiffre "1", sinon, il s'agit d'un autre chiffre.

III.3.6 Avantages et inconvénients de la machine à vecteurs de support

- L'apprentissage du modèle est relativement facile.
- Le modèle s'adapte relativement bien aux données de grande dimension.
- Le SVM est une alternative utile aux réseaux neuronaux.
- Il est utile pour les données linéairement séparables et non linéairement séparables.
- Les SVM sont beaucoup plus efficaces quand on ne dispose que de peu de données d'entraînement ou les autres algorithmes n'arriveraient pas à généraliser correctement.
- Elles ont une base théorique solide.
- Différentes fonctions noyau peuvent être spécifiées.

- Le compromis entre la complexité du classificateur et l'erreur peut être contrôlé explicitement.

Malgré leurs performances, les SVM représentent aussi des faiblesses, notamment celles-ci :

- Elles utilisent des fonctions mathématiques complexes pour la classification.
- Les machines à support de vecteurs demandent un temps énorme durant les phases de test.

III.4 Conclusion

Dans ce troisième chapitre, nous avons abordé des notions d'apprentissage statistique, qui sont à la base de l'avènement des SVM. Nous avons ensuite explicité les bases théoriques permettant l'optimisation de la marge de séparation utilisée par cette méthode pour la recherche d'un hyperplan optimal pour séparer au mieux des ensembles de données. Nous avons ainsi exposé les cas linéairement séparables et les cas non linéairement séparables qui nécessitent l'utilisation de fonction noyau (kernel) pour changer d'espace. Cette méthode est applicable pour des tâches de classification à deux ou plusieurs classes.

Chapitre IV

*Reconnaissance & classification
automatiques des types de modulation*

IV.1 Introduction

La reconnaissance automatique de modulations (AMR) consiste à identifier, au niveau du récepteur d'une chaîne de transmission, la constellation à laquelle appartiennent les symboles du message reçu. Cette reconnaissance est nécessaire dans de nombreux scénarios de communication, qu'ils soient coopératifs ou non coopératifs [44]. La classification de la modulation est une étape intermédiaire entre la détection et la démodulation du signal, et joue un rôle clé dans diverses applications civiles et militaires.

Ce chapitre dresse notre travail simulation fait sous environnement MATLAB. Nous utilisons deux classifieurs SVM pour l'AMR de modulations numériques de systèmes SM-MIMO sous diverses configurations antennaires pour différentes conditions de transmission (diverses valeurs de rapport signal sur bruit SNR).

IV.2 Méthodologie des simulations

Avant d'entamer la tâche de classification objet de notre intérêt dans ce mémoire, une étape d'extraction de caractéristiques pertinentes de chaque type modulation est nécessaire. Pour ce faire, nous commençons nos simulations par récolter de grandes bases de données (40000 échantillons) formées de caractéristiques pertinentes des différentes modulations utilisées par nos systèmes SM-MIMO. Ces données sont issues de codes MATLAB fournis pour la détection, avec un égalisateur linéaire à la réception, des symboles de constellations numériques BPSK, QPSK, 64-QAM, 128-QAM avant leur démodulation. Ces codes utilisent un détecteur de type zéro forcing (ZF).

Les données récoltées sont les statistiques d'ordre supérieur (HOS) des classes de modulations choisies. Ces données sont stockées et rappelées lors de nos simulations de classification SVM. Un prétraitement permet de normaliser les données pour uniformiser l'effet d'éventuels bruits. Avant d'entamer la tâche de classification, nos données sont réparties aléatoirement en deux ensembles :

- un ensemble d'apprentissage (comprenant 90% des données) pour l'entraînement ;
- et un ensemble de test (les données restantes : 10%) pour la validation du classifieur retenu lors de la phase d'apprentissage.

Nos simulations ont été conduites sous le logiciel MATLAB, pour la tâche d'extraction des statistiques d'ordre supérieur des modulations et pour la tâche de classification. Nous précisons que les classifieurs SVM utilisés sont issus de l'application « Classification Learner » sous l'interface « Machine Learning and Deep Learning » de MATLAB 2020 et d'une librairie LIBSVM de l'université nationale de Taiwan. Des descriptions fonctionnelles de ces deux classifieurs sont données dans l'annexe A en fin de ce mémoire.

IV.2.1 Application « Classification Learner / MATLAB»

Classification Learner est une application hébergée sous MATLAB qui permet d'explorer l'apprentissage automatique supervisé à l'aide de diverses méthodes de classification: arbres de décision, machines à vecteurs de support, régression logistique, les k-plus proches voisins, et les réseaux neuronaux.

Le « classification Learner » réalise la tâche de classification comme suit :

- L'entraînement de plusieurs modèles de classifieurs SVM multi-classes B-PSK, Q-PSK, 64QAM et 128-QAM avec les données de l'ensemble d'apprentissage (en variant les fonctions noyau et les valeurs de leurs hyperparamètres). Le modèle retenu est celui qui assure l'optimalité. Les valeurs de ses hyperparamètres sont données par l'application ;
- La validation du modèle retenu lors de la phase d'apprentissage est faite avec les données de l'ensemble de test.
- Les performances du classifieur sont évaluées à l'aide du taux de classification obtenu, de matrices de confusion et de courbes ROC.

IV.2.2 Librairie LIBSVM

LIBSVM est une bibliothèque dédiée aux Machines à Vecteurs de Support développée par deux chercheurs du département Informatique de l'Université nationale de Taïwan depuis les années 2000, l'objectif étant d'aider les utilisateurs à appliquer facilement les SVM à leurs applications. Elle est disponible en langage C++ et en JAVA, et compatible avec diverses plateformes logicielles (Python, R, MATLAB, Perl, ...). LIBSVM met en œuvre l'algorithme d'optimisation minimale séquentielle (SMO) pour la classification par vecteurs de support (C-SVC, nu-SVC), la régression (epsilon-SVR, nu-SVR) et l'estimation de distribution (SVM à une classe). LIBSVM supporte l'apprentissage multi-classes et l'estimation de probabilité pour des valeurs de confiance appropriées après l'application du modèle appris sur un ensemble de données de classification.

Le code d'apprentissage SVM de cette bibliothèque est souvent réutilisé dans d'autres boîtes à outils d'apprentissage automatique open source. Des liaisons et des ports existent pour des langages de programmation tels que Java, MATLAB, R et Python.

Nous allons intégrer ce logiciel pour la classification par vecteurs de support, sur notre MATLAB afin de pouvoir faire une classification multi-classes.

La dernière version (version 3.25, avril 2021) peut être obtenue en téléchargeant son fichier zip sur www.csie.ntu.edu.tw/~cjlin/libsvm/. Le paquet comprend le code source de la bibliothèque en C++ et Java, Un fichier README contenant des explications détaillées est fourni.

L'installation de cette dernière est expliquée dans l'Annexe A.

IV.3 Métriques d'évaluation des performances des classifieurs

IV.3.1 La matrice de confusion

La matrice de confusion, appelée également matrice d'erreurs, est un tableau croisé $N \times N$ (où N est le nombre de classes) qui enregistre le nombre d'occurrences entre deux évaluateurs, la classification réelle et la classification prévue, comme le montre la figure IV.1. Cette matrice est utilisée pour mesurer la qualité d'un système de classification, son utilité réside dans sa facilité d'interprétation. Ces matrices sont utilisées dans les statistiques, les modèles d'apprentissage automatique et d'autres applications de l'intelligence artificielle [14].

		Predicted class			
		A	B	C	D
True class	A	AA	AB	AC	AD
	B	BA	BB	BC	BD
	C	CA	CB	CC	CD
	D	DA	DB	DC	DD

Figure IV.1: matrice de confusion multiclasse.

Il est utile d'examiner la matrice de confusion pour les problèmes multi-classes où chaque ligne correspond à une classe réelle, et chaque colonne correspond à une classe estimée ou prédite. La matrice de confusion montre chaque classe dans les données d'évaluation et le nombre ou le pourcentage de prédictions correctes et de prédictions incorrectes en comptant toutes les combinaisons possibles. Les classes sont répertoriées dans le même ordre dans les lignes que dans les colonnes, donc les éléments correctement classés sont situés sur la diagonale principale de haut à gauche en bas à droite et ils correspondent au nombre de fois où les deux évaluateurs sont d'accord.

- **Vrais négatifs (VN)** : Le nombre d'exemple négatifs que le modèle a correctement classés comme négatifs.
- **Faux négatifs (FN)** : Le nombre d'exemple positifs que le modèle a incorrectement classés comme négatifs.
- **Faux positive (FP)** : Le nombre d'exemple négatifs que le modèle a incorrectement classés comme positifs.

- **Vrais positifs (VP)** : Le nombre d'exemple positifs que le modèle a correctement classés comme positifs (AA par ex).

IV.3.2 La Courbe ROC

Les courbes ROC (Receiver Operating Characteristics) furent inventées pendant la Seconde Guerre mondiale pour montrer la séparation entre les signaux radar et le bruit de fond. Cette courbe est un graphique représentant les performances d'un modèle de classification pour tous les seuils de classification [14]. Cette courbe trace le taux de vrais positifs en fonction du taux de faux positifs :

- Le **taux de vrais positifs (TVP)** est l'équivalent à la métrique d'évaluation Rappel ' Ψ '. Il est donc défini comme suit :

$$\frac{FN}{VP+FN} \quad (\text{IV.1})$$

- Le **taux de faux positifs (TFP)** est défini comme suit :

$$\frac{Fp}{VN+F} \quad (\text{IV.2})$$

Pour calculer les points d'une courbe ROC, nous pourrions calculer efficacement l'aire sous cette courbe, ou AUC, grâce à un algorithme de tri.

IV.3.3 AUC (Aire sous la courbe ROC)

AUC signifie "aire sous la courbe ROC", Cette valeur mesure l'intégralité de l'aire à deux dimensions située sous l'ensemble de la courbe ROC (par calculs d'intégrales) de (0,0) à (1,1). L'AUC fournit une mesure agrégée des performances pour tous les seuils de classification possibles. On peut interpréter l'AUC comme une mesure de la probabilité pour que le modèle classe un exemple positif aléatoire au-dessus d'un exemple négatif aléatoire. Les valeurs d'AUC sont comprises dans une plage de 0 à 1. Plus cette région se rapproche de 1, plus le classifieur est performant [14].

IV.4 Résultats de simulations pour différentes configurations antennaires

Dans cette section, nous présentons les résultats de nos simulations de classification sous la librairie LIBSVM et sous l'application « Classification Learner ». Les simulations de récolte des statistiques HOS (40 000 données) pour caractériser les types de modulations (10000 données pour chacun des 4 types) ne sont pas reportées dans notre mémoire. Notons que la détection est réalisée pour 4 valeurs de rapports RSB (SNR= [-15 0 15 30]) pour plusieurs configurations MIMO.

IV.4.1. Simulations sous LIBSVM

Nos simulations sous LIBSVM nous fournissent des figures de classification pour chaque configuration antennaire, les valeurs optimales des hyperparamètres des classifieurs SVM et de leurs matrices de confusion. La pertinence de chaque classifieur est basée sur le taux de bonne classification (Accuracy) pour chaque valeur SNR considérée.

La figure suivante montre un exemple de répartition des 40000 échantillons de statistiques HOS récoltées pour les quatre types de modulation numériques (classe 1 pour les 10000 HOS de la PSK, classe 2 pour la BPSK, classe 3 pour la 64-QAM, et classe 4 pour la 128-QAM).

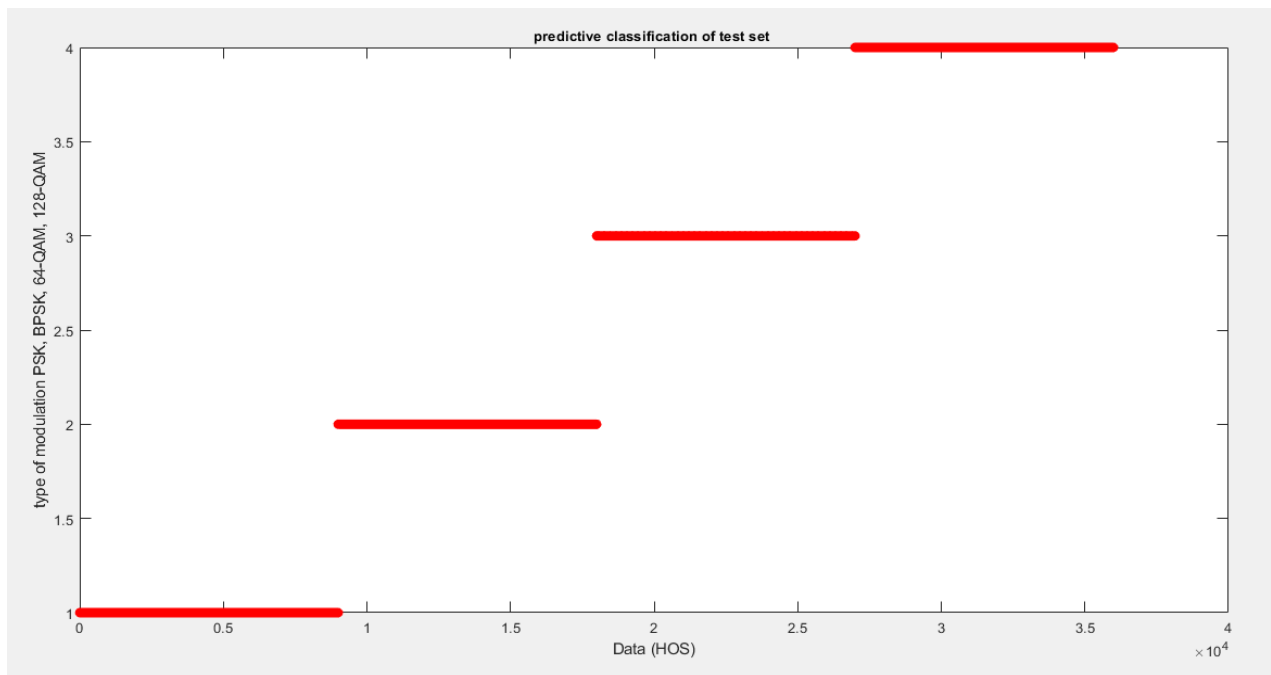


Figure IV.2: Répartition des échantillons (HOS) en 4 classes de types de modulation.

IV.4.1.1 Paramètres des classifieurs SVM optimaux

Le classifieur SVM optimal obtenu après chaque phase d'apprentissage est obtenu avec une optimisation croisée des valeurs des hyperparamètres du modèle. Nos simulations nous fournissent les valeurs optimales de ces hyperparamètres ainsi que divers paramètres de simulation.

Des exemples de résultats sont donnés dans le tableau IV.1 pour une même configuration MIMO ($N_t \times N_r = 2 \times 8$) pour 3 valeurs de SNR (-15 dB ; +15 dB; 30 dB). Les divers paramètres sont listés ci-après :

- **#iter** est le nombre d'itérations,
- **nu** utilisé pour choisir le type de fonction noyau,
- **obj** est la valeur objective optimale du problème SVM dual,

- **rho** est le terme de décalage pour la fonction de décision (-b),
- **nSV** est le nombre de vecteurs de support standard ($0 < a_i < c$),
- **nBSV** est le nombre de vecteurs de support sur la frontière (bornés) ($a_i = c$),
- **total nSV** est le nombre total de vecteurs de support (puisque'il n'y a qu'un seul modèle de classification total nSV = nSV, mais pour plusieurs classes, il s'agit de la somme des nSV pour chaque modèle de classification),
- **C** et **g** ce sont des paramètres des fonctions noyau,
- l'**Accuracy** c'est l'exactitude du classifieur ou taux de bonne classification.

SNR -15dB	<pre> optimization finished, #iter = 1012 nu = 0.999835 obj = -1000.812209, rho = 0.000075 nSV = 2000, nBSV = 1957 Total nSV = 4000 -c 1 -g 0.1 Accuracy = 25.0306% (9011/36000) (classification) </pre>
SNR 15dB	<pre> optimization finished, #iter = 4053 nu = 0.135042 obj = -3158.631203, rho = -0.092374 nSV = 549, nBSV = 193 Total nSV = 1162 -c 16 -g 0.0625 Accuracy = 86.4083% (31107/36000) (classification) </pre>
SNR 30dB	<pre> optimization finished, #iter = 993 nu = 0.037394 obj = -545.206851, rho = -0.028108 nSV = 146, nBSV = 55 Total nSV = 320 -c 11.3137 -g 0.0625 Accuracy = 96.9583% (34905/36000) (classification) </pre>

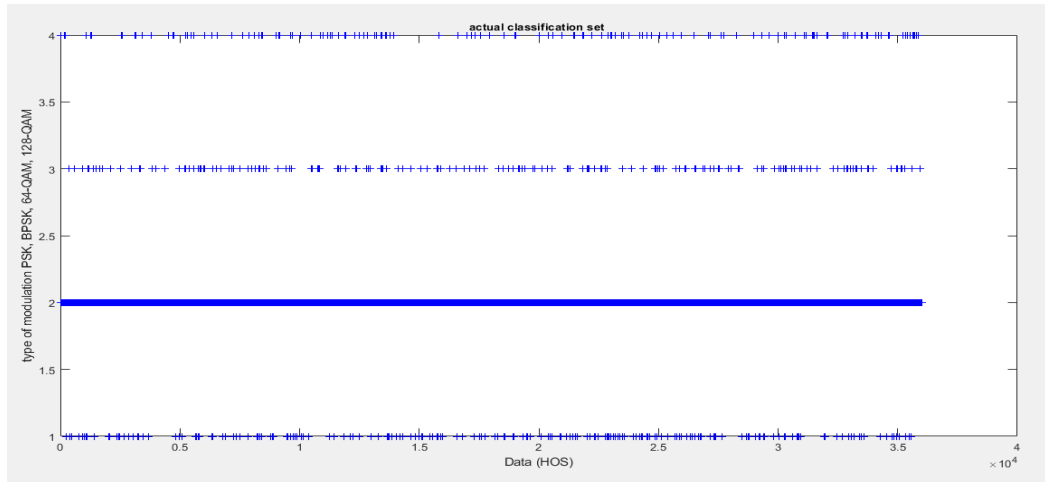
Tableau IV. 1: Résultats d'optimisation pour divers RSB [Nt*Nr=2*8]

Ce tableau révèle que le classifieur SVM retenu utilise une fonction noyau gaussien pour chaque cas. Les nombres de vecteurs supports standards et bornés sont réduits quand le rapport SNR s'améliore et la classification est beaucoup plus correcte vu que les symboles sont moins bruités.

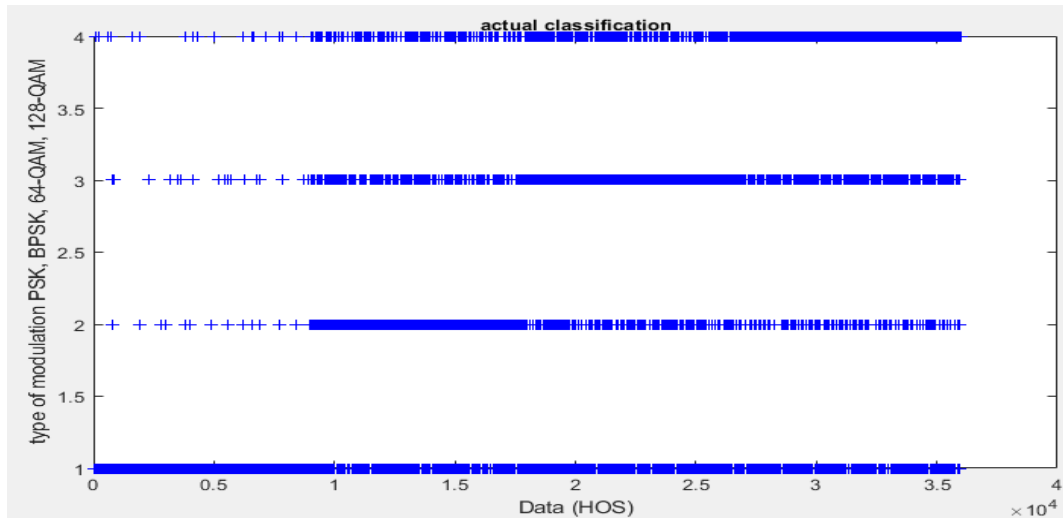
IV. 4.1.2 Classification des symboles de l'ensemble de validation

D'après la figure ci-dessous obtenue pour une configuration antennaire $[N_t * N_r = 2 * 8]$, la répartition des symboles de l'ensemble de validation sur les 4 types (classes) de modulation, n'est satisfaisante que pour des valeurs SNR élevées. Pour un SNR faible (-15 dB), presque tous les symboles sont considérés du même type de modulation (BPSK dans ce cas).

SNR -15dB



SNR 15dB



SNR 30dB

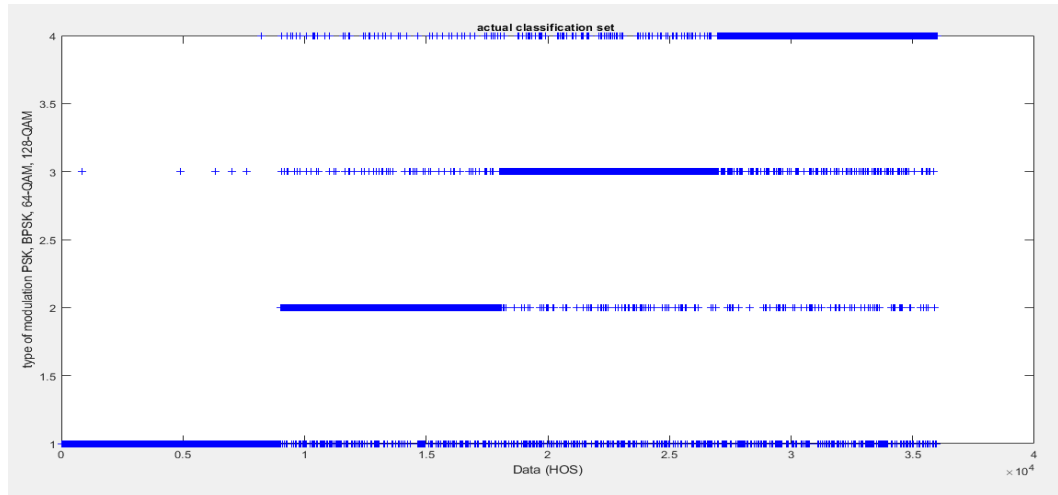


Figure IV. 2: Classification des symboles de l'ensemble de validation
[SNR :-15 ; 15, 30 dB ; Configuration MIMO $N_t*N_r= 2*8$].

IV. 4.1.3 Évaluation des matrices de confusion

- Résultats obtenus pour une configuration antennaire [$N_t*N_r= 2*8$]

		Classes réelles			
		BPSK	QPSK	64 QAM	128 QAM
SNR -15dB	Classes prédites				
	BPSK	0.52% (47)	96.61% (8875)	0.51% (46)	0.35% (32)
	QPSK	0.5% (45)	96.62% (8876)	0.46% (42)	0.41% (37)
	64 QAM	0.71% (64)	98.44% (8860)	0.34% (43)	0.17% (33)
	128QAM	0.53% (48)	98.42% (8858)	0.27% (49)	0.2% (45)
		Classes réelles			
		BPSK	QPSK	64 QAM	128 QAM
SNR 15dB	Classes prédites				
	BPSK	99.42% (8948)	0.14% (13)	0.21% (19)	0.22% (20)
	QPSK	9.52% (857)	83.4% (7506)	10.32% (929)	2.83% (255)
	64 QAM	9.05% (815)	3.7% (333)	80.93% (7284)	6.31% (568)
	128QAM	9.28% (836)	2.31% (208)	6.52% (587)	81.87% (7369)

SNR 30dB	Classes prédites	Classes réelles				
			BPSK	QPSK	64 QAM	128 QAM
		BPSK	99.93% (8994)	0% (0)	0.05% (5)	0.01% (1)
		QPSK	2.05% (185)	96.58% (8693)	0.74% (67)	0.61% (55)
		64 QAM	1.93% (174)	0.87% (79)	96% (8640)	1.18% (107)
	128QAM	2.04% (184)	0.8% (72)	1.84% (166)	95.31% (8578)	

Tableau IV. 2: Les Matrices de confusion pour divers RSB [$N_t \cdot N_r = 2 \cdot 8$]

Dans cette étape, on va comparer les matrices de confusion de la configuration antenneaire [$N_t \cdot N_r$] = [2*8] en fonction de leur classification correcte pour les différents SNR. Nous constatons que les meilleurs performances sont obtenues en augmentant le SNR, où pour le SNR= 30dB on remarque que plus de 8900 (par modulation) de nos données sont correctement classées avec une exactitude qui dépasse 98%.

Remarque : La matrice de confusion du SNR = 0 dB donne presque les mêmes résultats que le SNR=-15 dB (Les matrices du SNR=0dB se trouve dans l'Annexe B).

- **Résultats obtenus pour une configuration antenneaire [$N_t \cdot N_r = 4 \cdot 8$]:**

SNR	Classes prédites	Classes réelles			
		BPSK	QPSK	64 QAM	128 QAM
SNR -15dB	BPSK	0.27% (25)	0.32% (29)	0.29% (26)	99.11% (8920)
	QPSK	0.26% (24)	0.32% (29)	0.26% (24)	99.03% (8913)
	64 QAM	0.22% (20)	0.23% (21)	0.48% (44)	99.05% (8915)
	128QAM	0.21% (19)	0.22% (20)	0.34% (31)	99.22% (8930)
SNR 15dB	BPSK	97.40% (8766)	0.84% (76)	0.81% (73)	0.94% (85)
	QPSK	15.12% (1361)	73.45% (6611)	6.67% (601)	4.74% (427)
	64 QAM	14.53% (1308)	5.76% (519)	67.64% (6088)	12.05% (1085)
	128QAM	21% (1890)	3.94% (355)	9.55% (860)	71.05% (6395)
SNR 30dB	BPSK	99.99% (8999)	0% (0)	0.01% (1)	0% (0)
	QPSK	2.41% (217)	95.95% (8636)	0.85% (77)	0.77% (70)
	64 QAM	2.35% (212)	0.85% (77)	94.78% (8531)	2% (180)
	128QAM	2.45% (221)	0.48% (44)	1.6% (144)	95.78% (8621)

Tableau IV.3 : Les Matrices de confusion pour divers SNR [Nt*Nr=4*8]

• Résultats obtenus pour une configuration antennaire [Nt*Nr=8*8]

SNR	Classes prédites	Classes réelles			
		BPSK	QPSK	64 QAM	128 QAM
SNR -15dB	BPSK	0.17% (16)	99.33% (8940)	0.23% (21)	0.37% (23)
	QPSK	0.3% (18)	99.30% (8937)	0.33% (30)	0.16% (15)
	64 QAM	0.21% (19)	99.26% (8934)	0.34% (31)	0.19% (16)
	128QAM	0.25% (21)	99.28% (8936)	0.27% (25)	0.2% (18)
SNR 15dB	BPSK	99.42% (8948)	0.14% (13)	0.21% (19)	0.23% (20)
	QPSK	9.52% (857)	83.4% (7506)	4.25% (382)	2.83% (255)
	64 QAM	9.05% (815)	3.7% (333)	80.93% (7284)	6.31% (568)
	128QAM	9.28% (836)	2.31% (208)	6.52% (587)	81.87% (7369)
SNR 30dB	BPSK	99.91% (8992)	0.02% (2)	0.06% (6)	0% (0)
	QPSK	2.41% (259)	95.95% (8527)	0.85% (118)	0.77% (96)
	64 QAM	2.35% (238)	0.85% (76)	94.78% (8478)	2% (208)
	128QAM	2.45% (233)	0.48% (85)	1.6% (152)	95.78% (8530)

Tableau IV.4: Les Matrices de confusion pour divers RSB [Nt*Nr = 8*8]

Après avoir associées nos données aux algorithmes, nous avons constaté que plus de 95% des instances ont été classées correctement. Nos caractéristiques ont donnée de bons résultats (sauf pour des SNR faibles -15dB par ex) avec une précision entre 0.94 (94%) et 0.99 (99%), cependant les meilleures performances sont obtenues avec le SNR= 30dB.

IV.4.1.4 Taux de bonne classification (exactitude des classifieurs)

La figure suivante représente l'exactitude des classifieurs pour différentes configurations antennaires ($N_t \cdot N_r = 2 \cdot 8, 4 \cdot 8, 8 \cdot 8$) en fonction des différents SNR = [-15, 0, 15 et 30] dB.

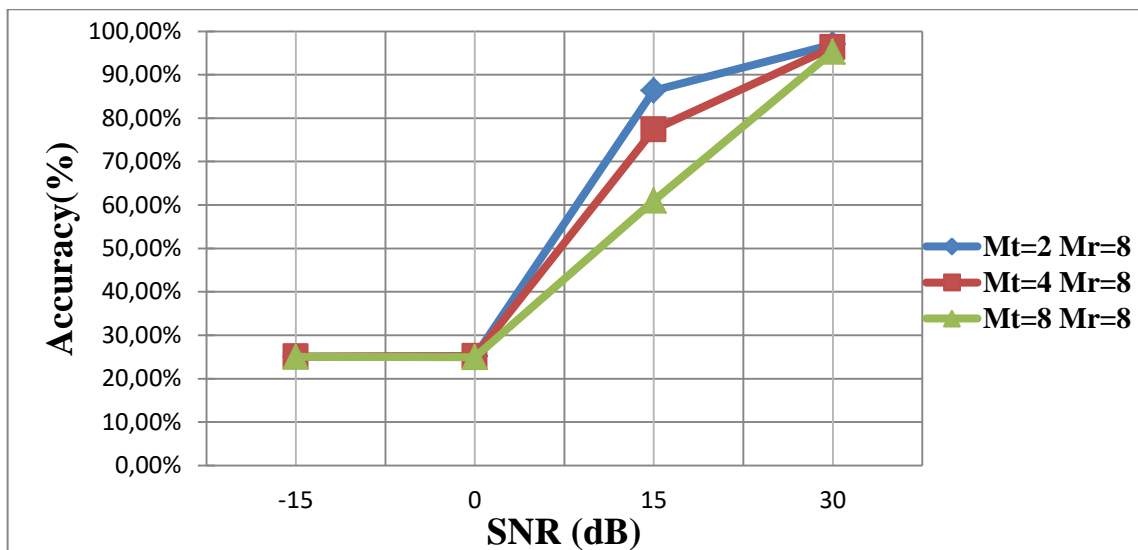


Figure IV. 4 : l'Accuracy des configurations antennaires pour les différents SNR.

La figure ci-dessus donne les taux de bonne classification obtenus par les classifieurs retenus par la LIBSVM pour diverses configurations antennaires. Les résultats montrent que le classifieur SVM a une meilleure exactitude :

- lorsque le rapport SNR est élevé quelle que soit la configuration antenne ;
- lorsque le nombre d'antennes réceptrices est élevé (dans le cas où $N_t=2$ et $N_r=8$).

Remarque : Les matrices de confusion pour le SNR= 0 dB sont données dans l'annexe B.

IV.4.2 Simulations sous « Classification Learner »

Dans cette deuxième méthode de classification utilisée nous avons choisi la configuration antenne $[N_t \cdot N_r] = [8 \cdot 8]$ pour laquelle on a eu des meilleurs résultats en terme d'exactitude et courbes ROC.

Pour avoir les résultats de classification, nous devons choisir les paramètres internes du modèle et au lieu de sélectionner manuellement ces options, nous avons utilisé l'optimisation des hyper paramètres dans l'application pour automatiser la sélection des valeurs d'hyper paramètres.

L'application affiche un diagramme **d'erreur de classification minimale** lors de l'exécution ce processus, À chaque itération, l'application essaie une combinaison différente de valeurs d'hyper paramètres et met à jour le tracé avec l'erreur de classification de validation minimale.

Ces hyper paramètres de classification sont : **multiclass method** qui permet de savoir la méthode de classification utilisée, **box constraint level** qui contrôle la pénalité maximale imposée aux

observations qui violent la marge, **karnel scale** paramètres d'échelles pour les données d'entrées et **karnel function** qui nous indique la fonction karnel utilisée, La figure IV. 5 illustre cette phase.

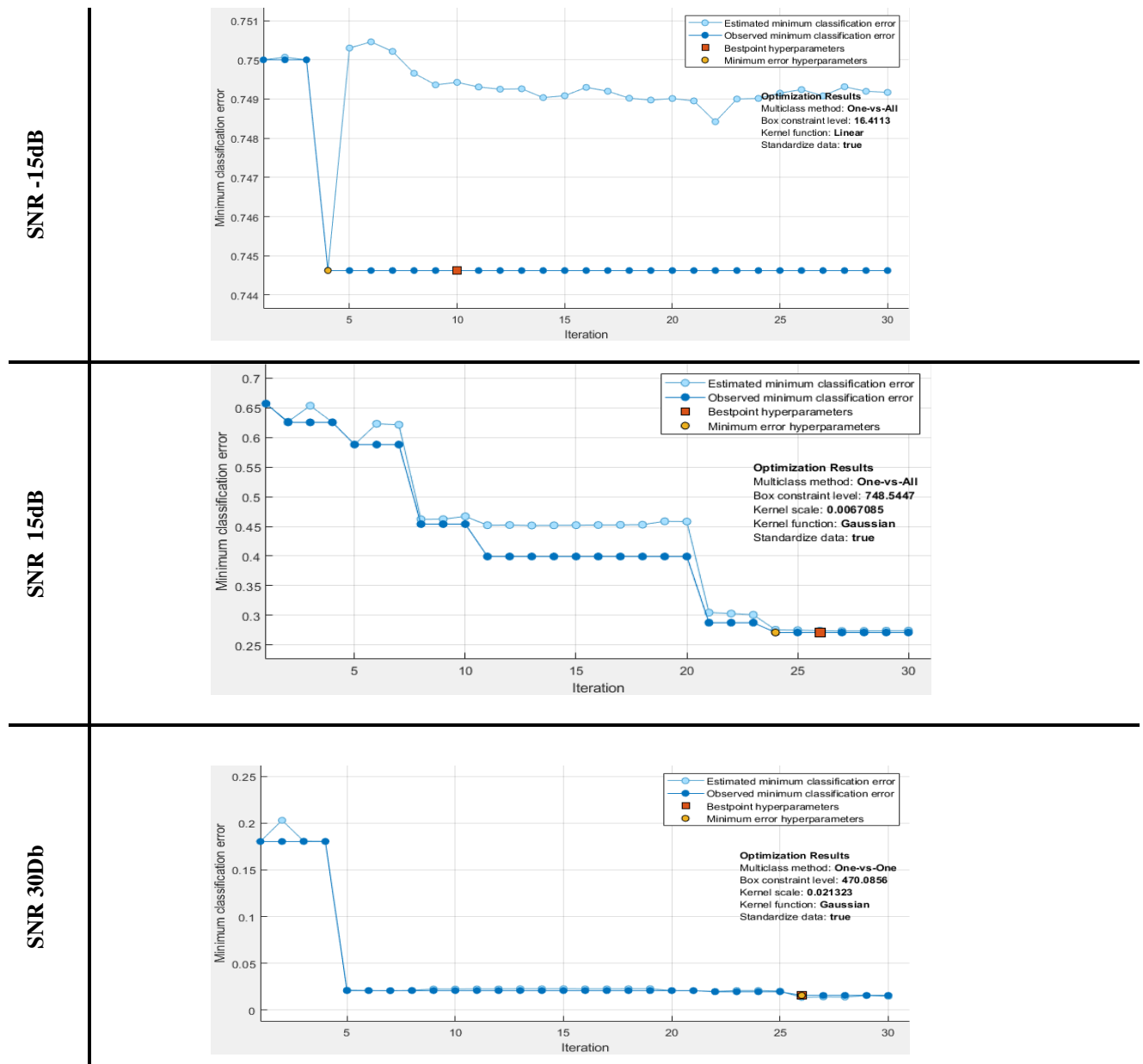


Figure IV. 5 : l'optimisation de la config (Nt*Nr=8*8) pour les divers SNR

IV.4.2.1 Évaluation des taux de bonne classification (exactitude du classifieur VM)

L'évaluation du classifieur est basée sur la probabilité d'identification correcte (Accuracy) en fonction de SNR (dB).

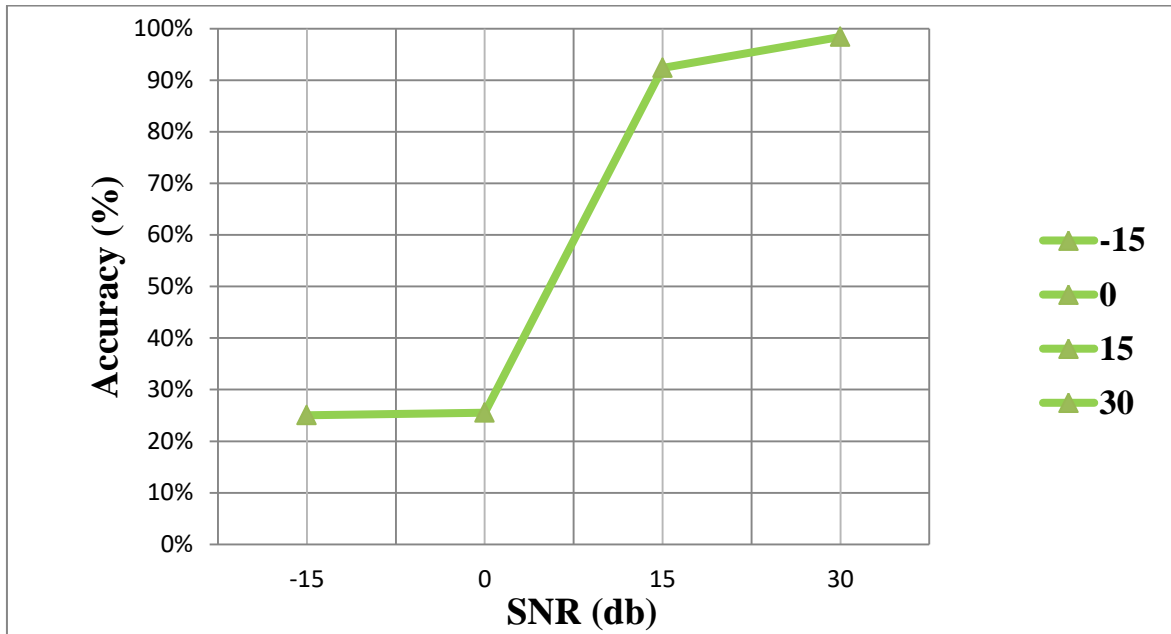


Figure IV.7 : l'Accuracy du ($N_t \cdot N_r = 8 \cdot 8$) pour les différents SNR

D'après la figure résultante, Nous remarquons que la performance de notre classifieur est limitée en termes de probabilité moyenne d'identification pour des niveaux faibles de SNR inférieur à 0 car elles sont évaluées qu'aux environs du 25 %, et 92,2% pour SNR=15.

Lorsque le SNR est élevé (dans le cas ou SNR =30db) l'exactitude est supérieur à 98%, cependant nous avons constaté que les performances atteintes par le classifieur en terme d'Accuracy s'augmente proportionnellement avec le SNR.

IV.4.2.2 Évaluation des matrices de confusion

Afin d'évaluer plus précisément les performances de classifieur, on utilise des métriques d'évaluation tel que la matrice de confusion.

La figure qui suit représente les matrices de confusions de la configuration antennaire $N_t \cdot N_r = 8 \cdot 8$ pour les trois SNR= [-15 15 30] dB.

Remarque :

- La matrice de confusion du SNR = 0 dB donne presque les mêmes résultats que le SNR=-15 dB (Les matrices du SNR=0dB se trouve dans **l'Annexe B**).

SNR	Classes prédites	Classes réelles			
		BPSK	QPSK	64QAM	128QAM
SNR -15dB	BPSK	(55,4%) 4995	(3,8%) 342	(3,8%) 342	(37,0%) 3330
	QPSK	(55,6%) 5004	(4,3%) 387	(4,3%) 387	(35,8%) 3222
	64 QAM	(56,3%) 5067	(5,2%) 477	(4,3%) 387	(34,2%) 3078
	128QAM	(54,5%) 4905	(4,5%) 405	(3,7%) 333	(37,3%) 3357
SNR 15dB	BPSK	89,8% (8082)	5,3% (477)	2,3% (207)	2,6% (226)
	QPSK	8,2% (738)	68,9% (6201)	12,3% (1107)	10,6% (945)
	64 QAM	7,2% (648)	10,0% (900)	56,5% (5085)	26,4% (2376)
	128QAM	6,5% (585)	6,0% (540)	14,1% (1269)	73,4% (6606)
SNR 30dB	BPSK	99,8% (8982)	0,1% (9)	0,1% (9)	0% (0)
	QPSK	0,8% (72)	98,5% (8865)	0,5% (45)	0,1% (9)
	64 QAM	0,7% (63)	0,3% (27)	97,2% (8748)	1,9% (171)
	128QAM	0,4% (36)	0,1% (9)	1,1% (99)	98,3% (8847)

Tableau IV.1: matrice de confusion pour [Nt*Nr = 8*8] avec divers SNR

Dans cette étude, il est question de comparer les matrices de confusion sous différents SNR en fonction des prédictions (correcte et incorrecte). Nous remarquons que les classifieurs les plus performants sont obtenus pour un SNR élevé (30 dB) pour les 4 types de modulations.

IV.4.2.3 Évaluation des AUC (aire sous la courbe ROC)

Dans cette partie, nous avons utilisé une autre métrique d'évaluation pour mesurer les performances des classifieurs SVM, appelée l'aire sous la courbe (AUC).

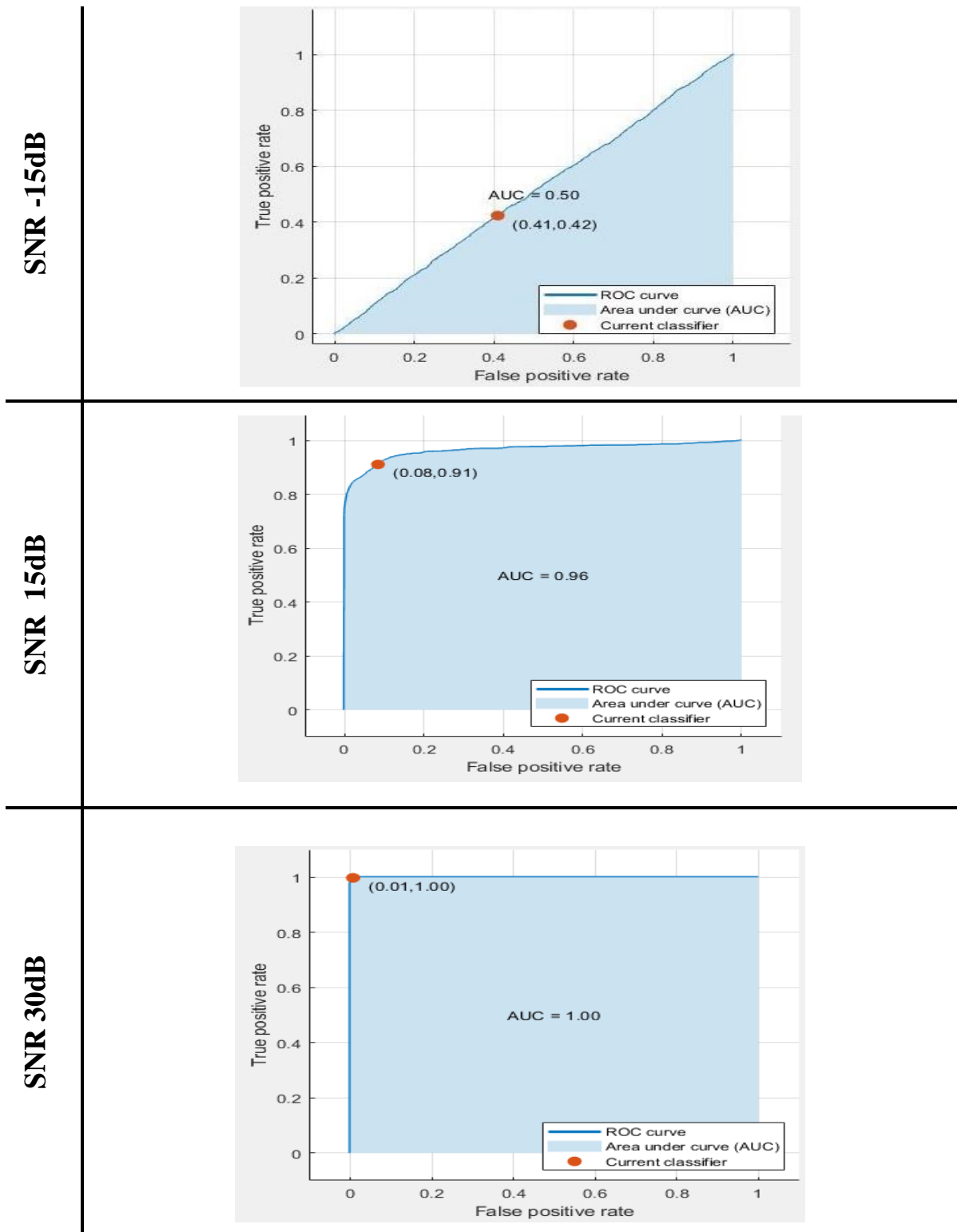


Figure IV.6: Courbe ROC pour $M_t * M_r : 8*8$ avec divers SNR

L'allure de la première courbe montre clairement que notre classifieur est aléatoire pour un SNR à faible valeur (-15dB). Les performances du classifieur s'améliorent nettement pour de meilleures valeurs du SNR (15 dB et 30 dB). La valeur AUC obtenue avec 30 dB indique un classifieur parfait (AUC =1).

IV.4.2.4. Évaluation des matrices de confusion

- Les matrices de confusions pour [Nt*Nr = 4*8]

SNR	Classes prédites	Classes réelles			
		BPSK	QPSK	64 QAM	128 QAM
SNR -15dB	BPSK	94,8% (8532)	0,9% (81)	0% (0)	4,3% (387)
	QPSK	93,5% (8415)	1,2% (99)	0% (0)	5,3% (477)
	64QAM	94,5% (8505)	1% 90	0% (0)	4,5% (405)
	128QAM	94,2% (8478)	0,4% (36)	0% (0)	5,4% 477
SNR 15dB	BPSK	91,1% 8199	5,0% (450)	2,1% (189)	1,8% (162)
	QPSK	8,9% (8011)	68,8% (6192)	11,8% (1062)	10,5% (945)
	64 QAM	8,5% (765)	8,9% (8011)	59,3% (5337)	23,4% (2106)
	128QAM	7,9% (711)	5,7% (513)	15,3% (1377)	71,1% (6408)
SNR 30dB	BPSK	99,7% (8981)	0,1% (9)	0,1% (9)	0,1% (9)
	QPSK	0,3% (72)	98,9% (8865)	0,4% (45)	0,4% (9)
	64 QAM	0,5% (63)	0,4% (27)	97,2% (8748)	1,5% (171)
	128QAM	0,4% (36)	0,1% (9)	1,1% (99)	98,26% (8844)

Tableau IV.2: matrice de confusion pour [Nt* Nr =4*8]

Les valeurs d'AUC de la configuration [Nt* Nr =4*8] dans le cas où SNR= -15, 15 et 30 dB sont répertoriées dans le tableau suivant :

SNR _{dB}	-15 dB	15dB	30dB
AUC	0.50	0.75	1.00

- Les matrices de confusions pour [Nt* Nr : 2*8]

SNR	Classes prédites	Classes réelles			
		BPSK	QPSK	64 QAM	128 QAM
SNR -15dB	BPSK	0% 0	0,7% 63	4,7% 423	94,6% 8514
	QPSK	0% 0	1,5% 135	3,8% 342	94,8% 8532
	64 QAM	0% 0	0,9% 81	4,3% 387	94,8% 8532
	128Q AM	0% 0	1,3% 117	4,3% 387	94,5% 8505
SNR 15dB	BPSK	97,6% 8784	1,1% 99	0,7% 63	0,6% 54
	QPSK	2,4% 216	92,6% 8334	3,2% 288	1,8% 162
	64 QAM	2,1% 189	1,8% 162	87,8% 7902	8,2% 738
	128Q AM	1,9% 171	1,0% 90	5,3% 477	91,8% 8262
SNR 30dB	BPSK	99,7% 8973	0,2% 18	0% 0	0,1% 9
	QPSK	0,2% 18	99% 8910	0,6% 54	0,3% 27
	64 QAM	0,4% 36	0,3% 27	97,5% 8775	1,8% 162
	128Q AM	0,7% 63	0,4% 36	0,7% 63	98,3% 8847

Tableau IV. 2.3: matrice de confusion pour [Mt* Mr : 2*8]

Les valeurs d'AUC de la configuration [Nt* Nr =2*8] dans le cas où SNR= -15, 15 et 30 dB sont répertoriées dans le tableau suivant :

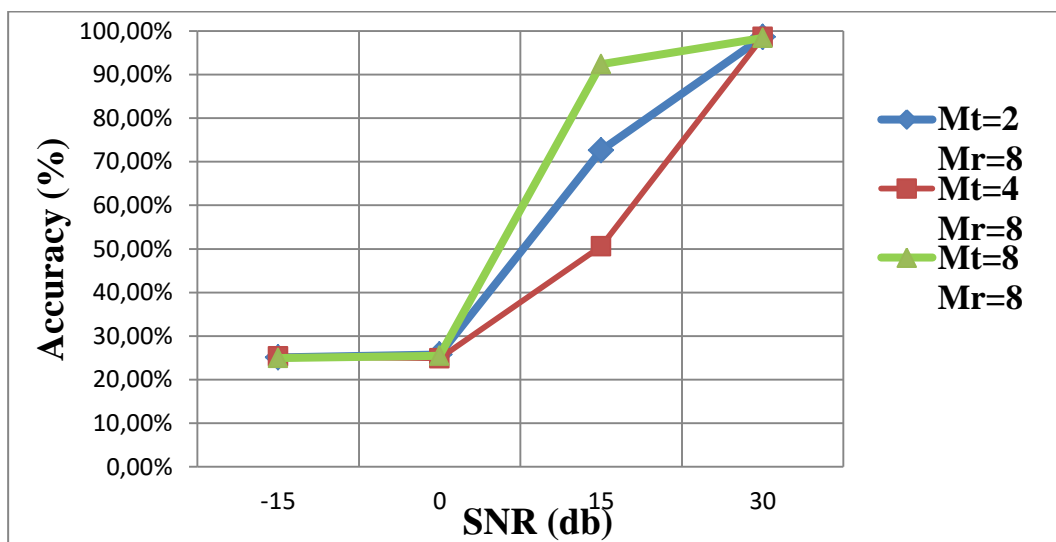
SNR _{dB}	-15 dB	15dB	30dB
AUC	0.50	0.96	1.00

Dans ces tableaux nous avons représenté les matrices de confusion des deux autres configurations antennaires en fonction de leur classification correcte.

D'après les résultats obtenus de ces matrices de confusion pour les 4 types de modulations, on trouve que les meilleures performances sont obtenues par la configuration ($M_t = M_r = 8$). Ce qui signifie que la diversité spatiale a un rôle crucial et primordial pour assurer de meilleures performances de classification.

IV.4.2.5 Évaluation du taux de bonne classification

La figure suivante représente le taux de bonne classification des classifieurs SVM sous différentes configurations antennaires ($2 \times 8, 4 \times 8, 8 \times 8$) :



Figures IV.7 : L'Accuracy des config.

Il ressort de ces courbes que les résultats obtenus en termes d'exactitude, le classifieur SVM est de plus en plus performant lorsqu'on augmente le SNR (98% pour un SNR=30 dB).

Nous remarquons que les meilleurs résultats sont obtenus pour la configuration $N_t = N_r = 8$ ce qui signifie que la diversité spatiale permet une meilleure détection des symboles et une classification correcte du type de modulation de ces symboles.

IV.5. Conclusion

Dans ce dernier chapitre, nous avons présenté nos simulations de classification automatique des types de modulations numériques de symboles de systèmes SM-MIMO. Après une étape d'extraction des caractéristiques pertinentes pour les 4 types de modulation utilisés, nous avons entraîné des classifieurs SVM issus de deux applications puissantes (LIBSVM, Classification Learner). L'évaluation des performances des classifieurs retenus, a été faite avec diverses métriques. Les résultats obtenus indiquent que les classifieurs automatiques retenus sont très performants pour des valeurs SNR élevées.

Conclusion générale

L'émergence de la radio-cognitive (RC) permet d'implémenter des capacités de cognition dans les émetteurs et les récepteurs pour s'adapter et tirer profit de l'évolution de leurs environnements radio de façon dynamique et autonome. L'une des fonctionnalités importantes des récepteurs RC est la reconnaissance automatique de modulation (AMR), très utilisée dans diverses applications où la surveillance des communications est stratégique, puisqu'elle permet d'extraire des caractéristiques représentatives du type de signaux inconnus reçus.

Le travail mené dans ce mémoire a pour objectif d'explorer le domaine de la classification automatique basée sur l'apprentissage statistique pour permettre une efficacité accrue dans la dernière phase de la reconnaissance AMR. Il s'agit explicitement de maîtriser l'utilisation de classifieurs SVM pour la reconnaissance automatique de types de modulations numériques utilisés dans des systèmes MIMO à modulation spatiale.

Nos investigations nous ont conduites à utiliser deux types de classifieurs SVM : la librairie LIBSVM et l'application « Classification Learner » de MATLAB. Les différentes simulations effectuées sur des bases de données de taille conséquente nous ont démontré la forte capacité de généralisation des classifieurs SVM. Nos résultats ont révélé de très bonnes performances notamment pour des conditions de transmission favorables (SNR élevés) et une bonne diversité spatiale des systèmes SM-MIMO.

Sur le plan pédagogique, nous avons saisi cette opportunité afin d'approfondir nos connaissances dans le domaine des télécommunications, plus précisément sur la radio cognitive, les systèmes MIMO et l'apprentissage statistique.

Nous espérons que notre travail puisse inspirer de nouvelles études sur les systèmes SM-MIMO pour exploiter d'autres algorithmes de détection des symboles pour améliorer les performances des classifieurs SVM même pour des SNR défavorables, ou explorer d'autres techniques pour la reconnaissance automatique des symboles et pas uniquement leur type de modulation.

Bibliographie

- [1] Driouch, El Mahdi (2014). « Allocation des ressources dans les réseaux sans fil émergents » Thèse. Montréal (Québec, Canada), Université du Québec à Montréal, Doctorat en informatique.
- [2] Glisic S., «ADVANCED WIRELESS NETWORKS Cognitive, Cooperative and Opportunistic 4G Technology», Second Edition, University of Oulu, Finland
- [3] S. U. Hwang, S. Jeon, S. Lee, and J. Seo, “Soft-output ML detector for spatial modulation OFDM systems,” *IEICE Electron. Exp.*, vol. 6, no. 19, pp. 1426–1431, Sep. 2009.
- [4] Abdessamad, El Rharras & Rachid, Saadane & Wahbi, Mohammed & Hamdoun, A. « Signal Detection and Automatic Modulation Classification Based Spectrum Sensing Using PCA-ANN with Real Word Signals », 2014. *Applied Mathematical Sciences*. 8. 7959-7977. 10.12988/ams.2014.49736.
- [5] V. Vapnik, “The Nature of Statistical Learning Theory”. Springer Verlag, New York, USA, 1995.
- [6] J. Mitola and G. Maguire “Cognitive radio: Making software radios more personal”, *IEEE Personal Communications* August 1999.
- [7] A. Metref “Contribution à l’étude du problème de synchronisation de porteuse dans le contexte de la Radio Intelligente ”, Novembre 2010.
- [8] E. Hossain, D. Niyam, Zhu Han, “Dynamic Spectrum Access and management in cognitive radio networks”, Cambridge University Press 2009.
- [9] R.Kamath, “Spectrum sensing in cognitive radio networks”, International Conference on Control, Communication and Computing, 2010.
- [10] W. Ejaz, N. ul Hasan, S. Lee, H-S. Kim, “IS: Intelligent spectrum sensing scheme for cognitive radio networks, *Eurasip Journal on Wireless Communications and networking*, 2013
- [11] Kandeepan Sithamparanathan, Gianmarco Baldini et Radoslaw Piesiewicz; « UWB Cognitive Radios »
- [12] Mahmood A.Abdulsattar and Zahir A.Hussain, « Energy Detection for spectrum sensing in cognitive radio: survey», September 2012, in *International Journal of Computer Networks & Communications (IJCNC)* vol 4.
- [13] C. Yang, Z. He, Y. Peng, Y. Wang and J. Yang, "Deep Learning Aided Method for Automatic Modulation Recognition," in *IEEE Access*, vol. 7, pp. 109063-109068, 2019.
- [14] Hakim TAYAKOUT, « Détection itérative par Turbo codage combinant différentes techniques à base de systèmes MIMO », Thèse de doctorat en Electronique, Ecole Nationale Polytechnique, 2019.

- [15] R. O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, Second edition. New York: Wiley-Interscience, 2001.
- [16] Mohamadally Hasan et Fomani Boris, "SVM : Machines à Vecteurs de Support Ou Separateurs a Vastes Marges" Versailles St Quentin, France janvier 2006.
- [17] Mobien, Mohammed. (2013). An Overview of Feature-Based Methods for Digital Modulation Classification. 2013 1st International Conference on Communications, Signal Processing and Their Applications, ICCSPA 2013. 10.1109/ICCSPA.2013.6487244.
- [18] HAROUS Walid, MRAIN Zakaria, « Reconnaissance automatique de type de modulation dans les réseaux MIMO coopératifs » mémoire pour l'obtention du diplôme d'ingénieur d'état en électronique.
- [19] M. BENYAROU, F.T.BENDIMERAD, F. DEBBAT, « Multi-User Detection for Multibeam-MIMO-Multi-Carrier-CDMA systems with MMSE adaptive algorithm » International Journal of Computer Science Issues IJCSI, Volume 10, 2013.
- [20] I.Ouachani, —Performance analysis of communication systems wireless operator Micro and Macro Diversityll, University of Paris XI Rosary, Discipline: Automatic and Signal Processing, June 28, 2005.
- [21] Olivier Berder. «Optimisation et stratégies d'allocation de puissance des systèmes De transmission multi antennes ». Université de Bretagne Occidentale, 20 décembre 2002.
- [22] Kaïs MABROUK. Conception et réalisation d'un système de Télécommunications MIMO avec formation numérique de Faisceaux en réception ; Calibrage aveugle du Démodulateur triphasé Zéro IF et comparaison au démodulateur classique à 2 voies I et Q. Thèse de doctorat, Ecole nationale supérieur des télécommunications, Décembre 2008.
- [23] A. BENAMAR, N. BELKHATIR, 'Performance Evaluation of a Generic Deployment Infrastructure for Component-based S/W Engineering', Proceeding of 6th International Conference on Software Engineering Advances (ICSEA 2011), October 23-29, Barcelona, Spain, pp. 380-385, ISBN : 978-1-61208-165-6.
- [24] L. Zhang, "A study of IEEE 802.16a OFDM-PHY Baseband" Master thesis performed in Electronics Systems, Department of Electrical Engineering, Linköping Institute of Technology, 16 February 2005.
- [25] J. Paulraj et al. "An Overview of MIMO Communications-A Key to Gigabit Wireless", Proceedings of the IEEE, Vol. 92, No. 2, February 2004.
- [26] A. Chaboub, "Etude des algorithmes de traitement de signal requis pour un système de Communication MIMO". Thèse de doctorat, Ecole de technologie supérieur, Université du Québec, Montréal, 21 Mai 2003.

- [27] B. Hassibi et H. Vikalo. On the Sphere-Decoding Algorithm I. Expected complexity. *IEEE Trans. Signal Process.*, vol. 53, pages 2806–2818, 2005.
- [28] K. Lee et J. Chun. ML Symbol Detection Based on the Shortest Path Algorithm for MIMO Systems. *IEEE Trans. Signal Process.*, vol. 55, pages 5477–5484, 2007.
- [29] David Gesbert, Mansoor Shafi, Da-shan Shiu, Peter J Smith, and Ayman Naguib. From the ory to practice: An overview of mimo space-time coded wireless systems. *IEEE Journal on selected areas in Communications*, 21(3) :281–302, 2003.
- [30] M. Di Renzo, H. Haas, and P. M. Grant, “Spatial modulation for multiple-antenna wireless system: A survey,” *IEEE Commun. Mag.*, vol. 49, no. 12, pp. 182–191, Dec. 2011.
- [31] Marco Di Renzo, Harald Haas, Ali Ghayeb, Shinya Sugiura and Lajos Hanzo, 2014: Spatial Modulation for Generalized MIMO: Challenges, Opportunities, and Implementation, *Proceedings of the IEEE*, vol. 102, no. 1, pp.56-103.
- [32] V. Tarokh, H. Jafarkhani, and R. Calderbank, “Space-time block codes from orthogonal designs,” *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1456–1467, Jul. 1999.
- [33] J. Mietzner, R. Schober, L. Lampe, W. H. Gerstacker, et P. A. Hoher, "Multiple-antenna techniques for wireless communications - A comprehensive literature survey", *IEEE Commun. Surveys Tuts.* Vol. 11, n° 2, pp. 87-105, 2e trimestre 2009.
- [34] R. Y. Mesleh, H. Haas, S. Sinanovic, C. W. Ahn et S. Yun, "Spatial modulation", *IEEE Trans. Veh. Technol.*, vol. 57, no. 4, pp. 2228-2241, juillet 2008.
- [35] V. Vapnik, *the nature of statistical learning theory* Wiley, New York, 1998
- [36] M. Tounsi “ cyclage actif en mode MLI des modules de puissance IGBT Application des SVM pour LE Diagnostic des Défauts de Vieillesse Thermique “. Thèse de doctorat, université Abderahmane Mira Bejaia, 23 octobre 2011.
- [37] P. Mahé : " Noyaux pour graphes et Support Vector Machines pour le criblage virtuel de molécules ". Rapport de stage, DEA MVA 2002/2003, Septembre 2003
- [38] Nicolas Turenne, “Apprentissage automatique (Machine Learning) "INRA ,2006.
- [39] H. Mohamadally et B. Fomani, " SVM : Machines à Vecteurs de Support ou Séparateurs à Vastes Marges" Versailles St Quentin, France. janvier 2006.
- [40] H. Mohamadally, B. Fomani SVM : Machines à Vecteurs de Support ou Séparateurs à Vastes Marges Versailles St Quentin, France 16 janvier 2006.
- [41] J. Milgram, Contribution à l’intégration des machines à vecteurs de support au sein des systèmes de reconnaissance des formes : Application à la lecture automatique de l’écriture manuscrite. MONTRÉAL, LE 29 JUIN 2007.
- [42] C. Chatelain, S. Adam, Y. Lecourtier. (2007). Optimisation multi-objectif pour la Sélection de modèles SVM. Rapport de recherche. Université de Rouen, France.

- [43] Thomas G.Dietterich and G.Bakiri. “Solving multiclass learning problems via error correcting output codes”. *Journal of Artificial Intelligence Research*, 2: pages 263–286, 1995.
- [44] A. Swami and B. Sadler, “Hierarchical digital modulation using cumulants,” *IEEE Trans. Commun.*, vol. 48, no. 3, pp. 416-429, Mar. 2000.

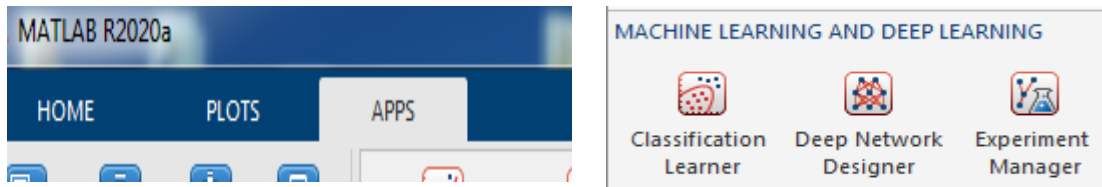
Annexes

Annexe A

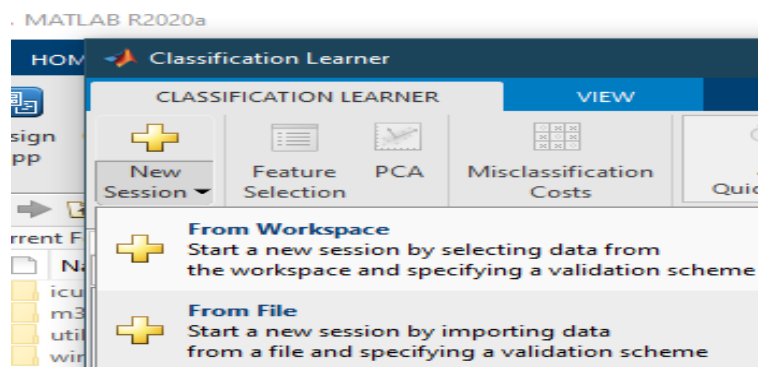
A.1 Classification Learner

Pour faire appel à cette application :

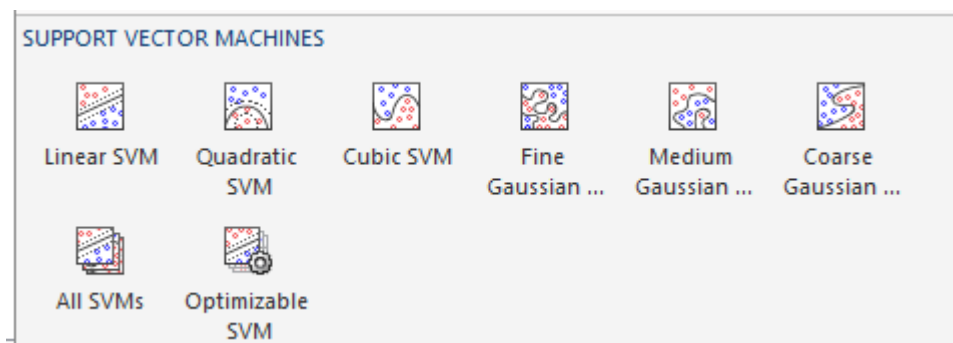
1. Dans l'onglet **Apps**, dans le groupe **machine Learning and deep Learning**, on clique sur **classification Learner (figure IV.1)**.



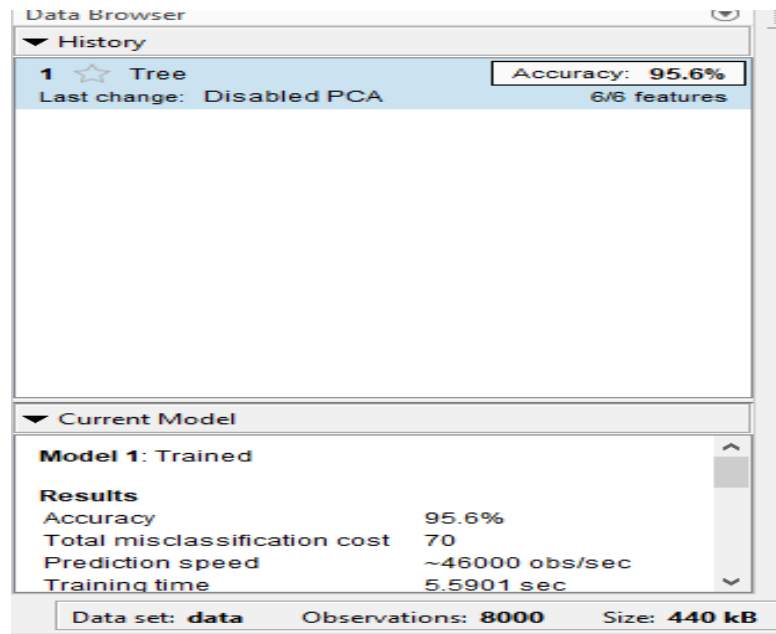
2. On clique sur **Nouvelle session** et sélectionner dans **From Workspace** les données à utiliser, comme des variables prédicteurs et des variables réponses.



3. Dans l'onglet **Classification Learner**, dans la section **Model Type**, on clique sur la flèche vers le bas pour développer la liste des classificateurs, on choisit **Support Vector Machines**, On clique sur **Train**.

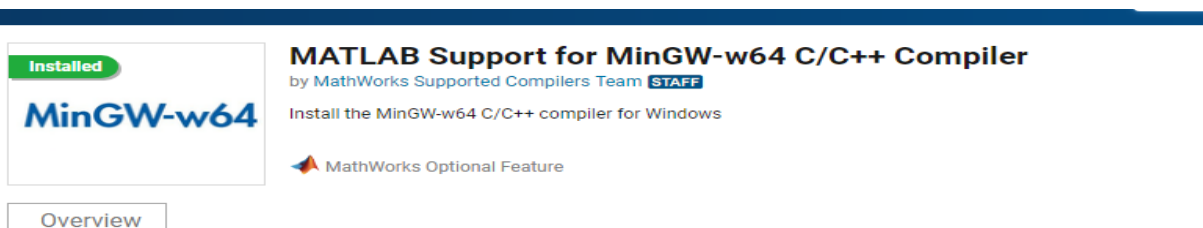


4. Une sélection de types de modèles apparaît dans la liste de l'historique. Lorsqu'ils terminent leur formation, le meilleur pourcentage de précision (Accuracy) est mis en évidence dans un encadré.



A.2 LIBSVM

1. Pour commencer on doit d'abord Télécharger la libsvmtoolbox, qui consiste à ajouter la boîte à outils au chemin de Matlab son site de téléchargement de la libsvmtoolbox, elle est téléchargé sous forme de **fichier zip**.
2. Puis, télécharger sur la machine le compilateur "**MinGW64 Compiler(C)**" selon votre machine est en 64 ou 32 bits, pour l'utiliser comme compilateur C.

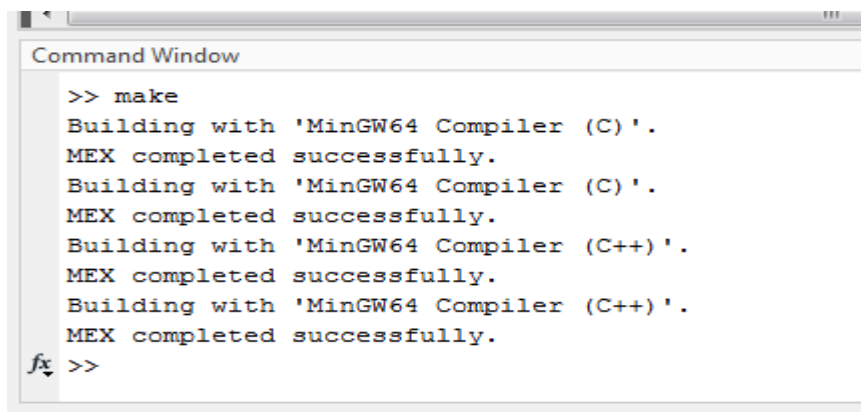


3. Dézipper le fichier libsvm-3.25.zip téléchargé, puis déplacer le fichier Windows à l'intérieur du dossier dézippé vers Matlab.



4. Ajouter le dossier LIBSVM dans Matlab.

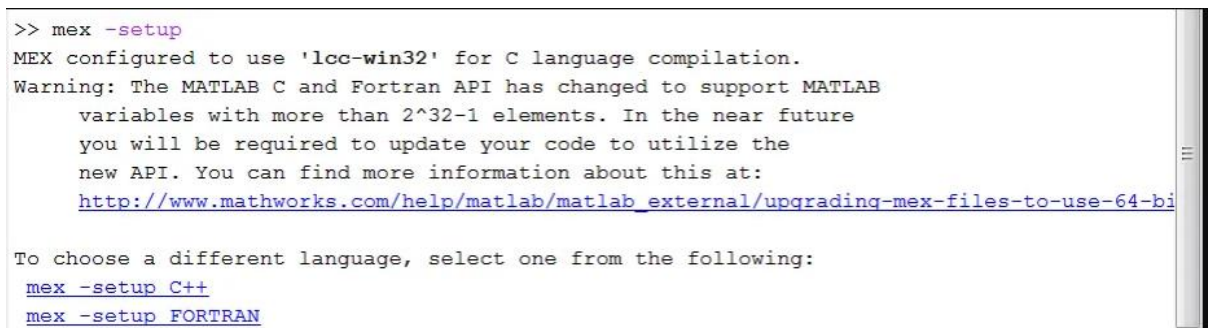
5. Dans la fenêtre de la ligne de commande de MATLAB, taper **make** pour compiler.



```
Command Window

>> make
Building with 'MinGW64 Compiler (C)'.
MEX completed successfully.
Building with 'MinGW64 Compiler (C)'.
MEX completed successfully.
Building with 'MinGW64 Compiler (C++)'.
MEX completed successfully.
Building with 'MinGW64 Compiler (C++)'.
MEX completed successfully.
fx >>
```

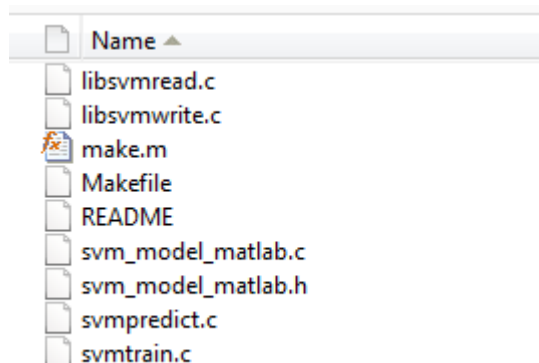
6. Ensuite entrer **mex -setup**, si ce qui suit apparaît, on clique sur la première ligne de **mex -setup C++**, ceci est le succès de l'installation. Et si y a un problème faut changer les CFLAGS de 13 lignes en COMPELAGS.



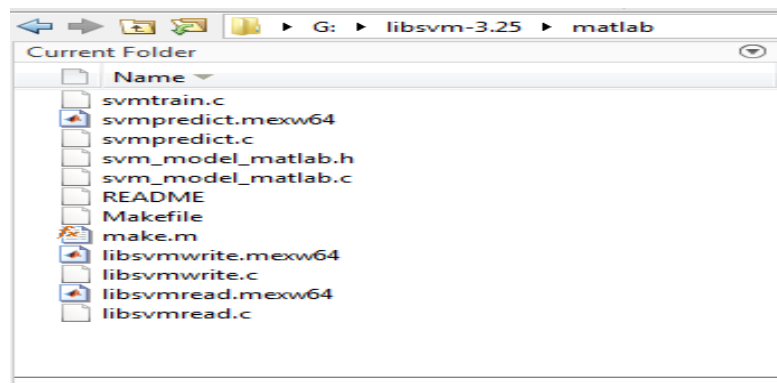
```
>> mex -setup
MEX configured to use 'gcc-win32' for C language compilation.
Warning: The MATLAB C and Fortran API has changed to support MATLAB
variables with more than 2^32-1 elements. In the near future
you will be required to update your code to utilize the
new API. You can find more information about this at:
http://www.mathworks.com/help/matlab/matlab\_external/upgrading-mex-files-to-use-64-bit-integers

To choose a different language, select one from the following:
mex -setup C++
mex -setup FORTRAN
```

Pour compiler, on va trouver le fichier README via Matlab dans le dossier décompressé. Vous pouvez voir les informations suivantes dans la fenêtre de MATLAB quatre fichiers **mex** sont obtenus en compilant.



7. Et quand l'installation sera fini on peut constater à ce moment, la fenêtre MATLAB comporte quatre fichiers mex supplémentaires sont obtenus, qui sont les fichiers qui viennent d'être compilés avec le suffixe mexw64.



8. On trouve un fichier heart_scale. Entre le code dans l'éditeur pour tester.

A.2.1 Fonctionnalités de la LIBSVM

Après la compilation, l'utilisateur peut faire appel aux fonctionnalités de la librairie en ligne de commande, ci-dessous un descriptif des commandes les plus basiques :

- Normalisation des données : svm-scale

La normalisation des données est préconisée dans la plupart des traitements d'analyse, la LIBSVM appuie cette pratique et propose une fonctionnalité pour la normalisation : svm-scale, elle recommande une normalisation linéaire utilisant les intervalles [-1; +1] ou [0; 1], en utilisant bien sur la même intervalle pour le jeu de données d'apprentissage et de test. Pour faire appel à cette commande il faut taper en ligne de commande :

svm-scale -y lower upper

Où **lower** et **upper** constituent les bornes de l'intervalle à utiliser pour la normalisation, généralement [0,1] ou [-1,1].

- Apprentissage : svmtrain

La commande svmtrain permet de lancer le processus d'apprentissage sur les données, elle prend en entrée un format de données spécifique, décrit plus loin, et génère un fichier de modèle. La commande pour utiliser cette fonctionnalité est :

svmtrain [options] training_set_file [model_file]

Où **training_set_file** constitue le fichier d'apprentissage et **model_file** le nom du modèle à générer.

L'usage de la svmtrain dans Matlab est

```
matlab>model = svmtrain(training_label_vector, training_instance_matrix [, 'libsvm_options'])
```

- Prédiction : svmpredict

Génère les classes prédites ou les valeurs prédites (en cas de régression), en se basant sur un Modèle entraîné, la commande à utiliser est la suivante :

svmpredict test_file model_file output_file

Où **test_file** est fichier de test, ou les données que l'on souhaite prédire, à noter qu'il doit également être sous le format LIBSVM et **model_file** le nom du modèle à générer.

Après prédiction, la fonction `svmpredict` compare les résultats de la prédiction avec les labels du fichier de test, pour en déduire la justesse de la prédiction et par la même occasion le taux d'erreur commis.

```
[predicted_label, accuracy, decision_values/prob_estimates] = svmpredict (testing_label_vector, testing_instance_matrix, model [, 'libsvm_options']);
```

```
[predicted_label] = svmpredict (testing_label_vector, testing_instance_matrix, model [, 'libsvm_options']);
```

A.2.2 Paramètres SVM pris en charge par la librairie :

Afin d'entamer le processus d'apprentissage avec la LIBSVM, certains paramètres sont à renseigner selon qu'on souhaite faire une classification, une régression ou autre, le choix des bons paramètres est déterminant pour obtenir des résultats satisfaisants. Ci-dessous la liste de tous les paramètres LIBSVM possibles. À noter que dans le cas où ces paramètres ne sont pas renseignés, la librairie utilise les valeurs par défaut, précisées ci-après également:

-s svm_type: C'est le type de l'algorithme SVM à utiliser, peut être l'une des fonctions: `C_SVC`, `NU_SVC`, `ONE_CLASS`, `EPSILON_SVR`, `NU_SVR`, et peut prendre respectivement les valeurs suivantes :

- 0: Pour une classification de type C-SVM
- 1: Pour une classification de type nu-SVM
- 2: Pour une classification de type one-class-SVM
- 3: Régression de type epsilon-SVM
- 4: Régression de type nu-SVM

Par défaut le type SVM utilisé est `C_SVC`.

-t kernel_type: C'est le type de la fonction noyau à utiliser, peut être défini à : `LINEAR`, `POLY`, `RBF` ou `SIGMOID`, et par conséquent prendre les valeurs :

- 0: Pour utiliser une fonction linéaire de formule $=u*v$
- 1: Pour une fonction polynomiale de formule $= (\text{gamma}*u*v + \text{coef0})^{\text{degré}}$
- 2: Pour une fonction Radiale de formule $= \exp (-\text{gamma}*|u-v|^2)$
- 3: Pour une fonction sigmoïde de formule $= \tanh (\text{gamma}*u*v + \text{coef0})$
- 4: Pour créer un kernel dans le fichier de test (cf. documentation LIBSVM pour le détail).

Tels que : **u'** représente la transposé du vecteur contenant les valeurs des attributs de l'ensemble d'apprentissage, et **v** le vecteur des labels (étiquettes). Le **gamma**, **degré** et **coef0** sont des paramètres (rentrés pas l'utilisateur). Par défaut le type de la fonction noyau utilisée est `RBF`, la documentation LIBSVM recommande l'utilisation de la fonction `RBF` pour plusieurs raisons, entre autre parce que `RBF` gère le cas où la relation entre les labels et les attributs est non linéaire.

Paramètre des fonctions noyau :

-d degree: Paramètre degré de la fonction noyau, par défaut 3

-g gamma : Paramètre gamma de la fonction noyau, par défaut 1

-r coef0 : Paramètre coef0 de la fonction noyau, par défaut 0.

Paramètres dépendants du type SVM choisi :

-c cost: C'est le paramètre C (coût), qui représente la pénalité de l'erreur, à renseigner lors de l'utilisation du type SVM C-SVC, epsilon-SVR et nu-SVR, par défaut le coût est égal à 1

-wi weight: pour changer le paramètre C à $\text{weight} * C$, s'il n'est pas renseigné weight est égale à 1 sa valeur par défaut, et par conséquent neutre.

-n nu : Paramètre nu du type nu-SVC, One-class-SVM et nu-SVR, par défaut 0.5

-p epsilon : Paramètre epsilon de la fonction de perte (Loss Function) pour le type epsilon-SVR, par défaut égal à 0.1

Paramètres SVM généraux :

-m cachesize: Pour paramétrer la taille mémoire (Mbits) allouée au noyau, par défaut 100

-e epsilon : C'est le critère d'arrêt de l'apprenant, par défaut 0.001. Il est recommandé d'utiliser une valeur de 0.00001 pour le nu-SVC, et 0.001 pour les autres types.

-h shrinking: Pour activer/désactiver l'heuristique de shrinking, par défaut mis à 1

-b probability-estimates: Pour activer/désactiver l'apprentissage avec probabilité, par défaut à 0

-v n : paramètre spécifique au mode validation-croisée (cross-validation), le n est le nombre de sous-divisiones de l'ensemble d'apprentissage (n-fold), par défaut désactivé.

-q: Mode silencieux, pour omettre l'affichage des messages, par défaut mis à 0 à noter que la librairie propose un utilitaire pour vérifier la validité des paramètres avant de faire appel à la fonction d'apprentissage, via la commande `svm_check_parameter()`.

Annexe B

B. Matrice de confusion du SNR=0dB

Nous présentons dans cette partie les matrices de confusion de SNR=0 pour les trois configurations antennaire utilisée dans les simulations.

B.1 pour la LIBSVM

$N_t * N_r = 2 * 8$	Classes réelles				
		BPSK	QPSK	64 QAM	128 QAM
	Classes prédites	BPSK	QPSK	64 QAM	128 QAM
		87,18% 7847	4,3% 387	4,73% 426	3,46% 312
		87,31% 7858	4,45% 401	4,7% 423	3,53% 318
	86,24% 7762	4,78% 431	4,98% 447	4,0% 360	
$N_t * N_r = 4 * 8$	Classes réelles				
		BPSK	QPSK	64 QAM	128 QAM
	Classes prédites	BPSK	QPSK	64 QAM	128 QAM
		95,56% 8601	1,46% 132	1,52% 137	1,44% 130
		94,94% 8545	1,56% 141	1,94% 175	1,54% 139
	95,2% 8568	1,47% 133	2,0% 180	1,32% 119	
	94,97% 8548	1,63% 147	1,9% 171	1,48% 134	
$N_t * N_r = 8 * 8$	Classes réelles				
		BPSK	QPSK	64 QAM	128 QAM
	Classes prédites	BPSK	QPSK	64 QAM	128 QAM
		0,7% 63	0,36% 33	0,45% 41	98,47% 8863
		0,51% 46	0,32% 29	0,47% 43	98,68% 8882
	0,73% 66	0,26% 24	0,53% 48	98,46% 8862	
	0,55% 50	0,38% 35	0,65% 59	98,4% 8856	

Tableau IV. 2.3: matrice de confusion pour SNR-0db pour divers config

B.2 Pour le Classifier Learner

		Classes réelles				
		BPSK	QPSK	64 QAM	128 QAM	
$N_t * N_r = 2 * 8$	Classes prédites	BPSK	12,4% 1116	24,1% 2169	53,3% 4797	10,2% 918
		QPSK	10,8% 972	24,9% 2241	54,0% 4860	10,2% 918
		64 QAM	9,9% 891	23,8% 2142	55,3% 4977	10,9% 981
		128QAM	10,9% 981	24,3% 2187	54,4% 4896	10,4% 936
$N_t * N_r = 4 * 8$	Classes prédites	BPSK	0,1% 9	3,8% 342	96,0% 8640	0,1% 9
		QPSK	0,1% 9	3,6% 324	96,0% 8640	0,2% 18
		64 QAM	0% 0	3,9% 351	96,0% 8640	0,1% 9
		128QAM	0% 0	3,8% 342	96,1% 8649	0,1% 9
$N_t * N_r = 8 * 8$	Classes prédites	BPSK	18,6% 1674	49,2% 4428	14,8% 1332	17,3% 1557
		QPSK	17,2% 1548	48,9% 4401	15,2% 1368	18,8% 1692
		64 QAM	17,6% 1584	49,5% 4455	16,6% 1494	16,3% 1467
		128QAM	16,9% 1521	50,2% 4518	15,1% 1359	17,8% 1602

Tableau IV. 2.3: matrice de confusion pour SNR=0 pour divers config

Résumé

Le travail présenté dans le cadre de ce mémoire concerne la diversité coopérative avec laquelle nous avons combiné trois techniques très utilisées dans le domaine des communications sans fil à savoir : le système MIMO, le système MIMO à modulation spatiale (SM-MIMO) et radio intelligente, et ce dans le but d'améliorer d'avantage les performances et la qualité des transmissions. La reconnaissance automatique de type de modulation (AMC) est un composant important des radios cognitives (RC), elle permet aux utilisateurs d'écouter le spectre et de y détecter les espaces blancs existants. Ce projet porte sur l'étude des performances d'un système de classification automatique de type de modulation, basée sur l'apprentissage automatique : les séparateurs à Vaste Marge (SVM) qui visent une identification optimale des types de modulation. Les résultats de simulations obtenus ont démontré de meilleures performances des solutions proposées et une grande robustesse face aux erreurs d'estimation des canaux.

Mot Clés :

Radio Cognitive, Diversité coopérative, Système MIMO, SM-MIMO, Radio intelligente, La reconnaissance automatique de type de modulation, Apprentissage automatique, SVM.

Abstract

The work presented in this thesis concerns the cooperative diversity with which we have combined three techniques widely used in the field of wireless communications, namely: MIMO, Space Modulated MIMO (SM-MIMO) and Smart Radio, with the aim of further improving the performance and quality of transmissions. Automatic modulation type recognition (AMC) is an important component of cognitive radios (CR), allowing users to listen to the spectrum and detect existing white spaces. This project investigates the performance of an automatic modulation type classification system, based on machine learning: Support Vector Machines (SVM), which aims at an optimal identification of modulation types. The simulation results obtained have shown better performances of the proposed solutions and a high robustness against channel estimation errors.

Keywords:

Cognitive Radio, Cooperative Diversity, MIMO System, SM-MIMO, Intelligent Radio, Automatic modulation type recognition, Machine learning, SVM.