

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Blida 1
Institut d'Aéronautique et des Études Spatiales
Département Études Spatiales



Mémoire de fin d'études présenté en vue de l'obtention du diplôme de
MASTER EN AÉRONAUTIQUE
Option : TÉLÉCOMMUNICATIONS SPATIALES

*Téledétection satellitaire de la surface terrestre
à l'aide des images SAR de Sentinel-1*

Proposé et encadré par :
Dr. AZMEDROUB Boussad

Réalisé par :
M. FERHAT Cherif Ismail
M. KHELFI Yahia Wail

Année 2024/2025

Remerciement

Nous tenons à exprimer notre profonde gratitude à Monsieur Boussad Azmedroub, notre encadrant, pour sa disponibilité, ses conseils avisés et son accompagnement tout au long de ce travail. Son encadrement rigoureux et bienveillant a été déterminant pour la réalisation de ce mémoire.

Nous souhaitons également rendre un hommage sincère à notre ami Zoair Mohammed Hayder, disparu trop tôt. Son esprit, sa gentillesse et son amour pour le savoir resteront à jamais gravés dans nos mémoires. Ce travail lui est dédié.

Nous remercions particulièrement Monsieur Zoair Tawhid , son père, pour son soutien moral et humain dans cette période difficile. Sa dignité et sa force nous ont profondément touchés.

Nos remerciements s'adressent aussi à l'ensemble des enseignants de l'Institut d'Aéronautique et des Études Spatiales pour la qualité de l'enseignement dispensé durant notre formation, ainsi qu'aux membres du jury pour avoir accepté d'évaluer notre mémoire avec attention et bienveillance.

Moi Khelfi Yahia Wail, je souhaite remercier mes chers parents, mes grands-parents, mon frère et ma sœur, mes amis ainsi que toute ma famille qui m'ont soutenu et encouragé durant tout mon parcours académique.

Moi FERHAT Cherif Ismail, je voudrais remercier mes chers parents, mes défunts grands-parents, mon frère bien-aimé ainsi que sa femme qui ont toujours été des piliers pendant tout mon parcours académique.

Enfin, nous remercions toutes les personnes qui, de près ou de loin, nous ont soutenues durant ce parcours.

Résumé

Ce mémoire porte sur l'utilisation des images radar Sentinel-1 pour la détection et l'analyse des changements de la couverture terrestre après des incendies de forêt. L'étude a été appliquée à la wilaya d'El Tarf, en Algérie, une région particulièrement exposée aux feux ces dernières années. Grâce à la technologie du radar à synthèse d'ouverture (SAR), nous avons pu suivre l'évolution du paysage indépendamment des conditions météorologiques. À l'aide du logiciel SNAP et d'outils de classification comme Random Forest, les images ont été prétraitées, classifiées et comparées entre deux dates pour détecter les zones touchées. Les résultats obtenus montrent l'efficacité des images SAR pour la surveillance environnementale post-incendie.

Abstract

This thesis focuses on the use of Sentinel-1 radar images to detect and analyze land cover changes following forest fires. The study was conducted in the El Tarf province of Algeria, a region heavily affected by wildfires in recent years. Using Synthetic Aperture Radar (SAR) technology, we were able to monitor landscape changes regardless of weather conditions. With SNAP software and classification methods like Random Forest, the radar images were preprocessed, classified, and compared between two dates to identify affected areas. The results demonstrate the effectiveness of SAR imagery in post-fire environmental monitoring.

Table des matières

Remerciement

Résumé i

Table des matières ii

Table des figures v

Liste des tableaux viii

Introduction Générale 1

1 Les Images SAR 2

1.1 Introduction 2

1.2 Radar à synthèse d'ouverture 2

1.3 Principe de formation des images SAR 3

1.4 Le satellite SENTINEL-1 5

1.4.1 Orbite 6

1.5 Résolution 7

1.5.1 Résolution radiale 8

1.5.2 Résolution azimutale 8

1.6 Bruit de l'image SAR 9

1.7 L'équation Radar 10

1.8 La polarimétrie 10

1.8.1 Les ondes électromagnétiques 11

1.8.2 Les types de polarisation 11

1.8.3 La matrice de Jones 14

1.8.4 La matrice de diffusion 15

1.8.5 Le vecteur cible 15

1.9	Les matrices de cohérence et de covariance	17
1.9.1	Matrice de covariance	17
1.9.2	Matrice de cohérence	17
1.9.3	La polarimétrie dans Sentinel-1	18
1.9.4	le rôle de la polarisation dans les forêts	18
1.10	La décomposition polarimétrique	20
1.10.1	Décomposition de pauli	20
1.10.2	Méthodes de Décomposition en Dual-Pol	21
1.11	Conclusion	22
2	Méthodes de classification	24
2.1	introduction	24
2.2	Les méthodes de classification supervisées	25
2.2.1	La méthode de classification SVM (Support Vector Machine)	25
2.2.2	Les réseaux de neurones convolutifs	31
2.2.3	La méthode de classification MLP (Multi Layer Perceptron)	37
2.2.4	La méthode de classification random forest	46
2.2.5	Comparaison entre les méthodes de classifications supervisées . . .	51
2.3	Les méthodes de classification non-supervisées	54
2.3.1	La méthode de Mean Shift	54
2.3.2	La méthode K-means	58
2.4	Conclusion	60
3	Résultats et analyses	62
3.1	Introduction	62
3.2	Site d'étude	62
3.3	Présentation des données	64
3.3.1	Présentation des images SLC	64
3.3.2	Présentation des images GRD	65
3.3.3	Le prétraitement des données	67
3.4	Traitement des images SLC	67
3.4.1	Ouverture de l'image SAR dans SNAP	68
3.4.2	Split	68
3.4.3	Split Orbit	69
3.4.4	Calibration	69

3.4.5	Deburst	70
3.4.6	Multi-looking	71
3.4.7	Terrain correction	71
3.5	Traitement des images GRD	73
3.5.1	Subset	74
3.5.2	Terrain correction	75
3.5.3	Calibrate	75
3.5.4	Speckle filtering	75
3.6	Classification avec Random Forest	77
3.6.1	Classifiacation avant le feu	77
3.6.2	Classification après le feu	79
3.7	Affichage des résultats sur Python	80
3.8	Conclusion	85
Conclusion Générale		86
Bibliographie		87

Table des figures

1.1	Première image SAR acquise en 1957 par l'université du Michigan	3
1.2	Antenne non statique	4
1.3	Principe d'acquisition d'image radar	5
1.4	Vue schématique de l'intensité spatiale Sentinel-1	6
1.5	Constellation de sentinel-1	7
1.6	Principe de la compression d'impulsion	8
1.7	Modes d'acquisitions d'images SAR	9
1.8	Polarisation elliptique	12
1.9	Polarisation circulaire	13
1.10	Polarisation linéaire	14
1.11	Image acquise par Copernicus Sentinel-1A le 2017-11-01 avec d'intenseur VV, image d'ambiance VH et composite de couleur RGB	18
1.12	schéma des effets de la polarisations dans les forets	18
1.13	Sentinel-1 C-band imagery (a) C_{VV} , (b) C_{VH} , (c) $ratio$, and (d) RGB composite $C_{VV}C_{VH}ratio$	19
2.1	deux classes linéairement séparables	27
2.2	Processus de maximisation de la marge	27
2.3	Exemple de deux classes non linéairement séparables.	28
2.4	Projection non linéaire des points dans un espace 3D. En bleu, on visualise une fonction de décision linéaire qui permet de séparer les classes une fois les données projetées dans un espace de plus grande dimension.	29
2.5	La fonction sigmoïde	29
2.6	La fonction de décision non linéaire	30
2.7	Méthodes OVA et OVO	30
2.8	CNN layers	31
2.9	Opération d'une convolution sur image de 5×5 pixel.	32
2.10	Max pooling	33

2.11	Le graphe de précision d'un réseau de neurone convolutifs	35
2.12	Le graphe d'erreur d'un réseau de neurone convolutifs	36
2.13	Relation entre le MLP et le IA	37
2.14	Diagramme du Perceptron Multi-Couches (MLP) avec quatre couches cachées.	38
2.15	Forward propagation	40
2.16	Fonction sigmoïde	41
2.17	La fonction tanh	41
2.18	La fonction RELU	42
2.19	Rétropropagation (backpropagation)	44
2.20	Descente de gradient	44
2.21	Root Mean Square Propagation	45
2.22	Arbre de décision	47
2.23	L'entropie	49
2.24	Le gain d'information	50
2.25	(a) une seule arbre de décision (b) 25 arbres de décision (c) 50 arbres de décision	51
2.26	Résultat du clustering avec Mean Shift. Trois clusters sont détectés auto- matiquement.	57
2.27	Exemple de classification K-means avec quatre points	59
3.1	Localisation et classes d'occupation du sol de la wilaya d'El Tarf	63
3.2	Images SLC Taref 03/09/2021	64
3.3	Images SLC Taref 05/09/2022	65
3.4	Image GRD du 29 aout 2021	66
3.5	Image GRD du 05 septembre 2022	66
3.6	Le traitement des données SAR des images SLC [1]	67
3.7	Ouverture des images SLC dur SNAP	68
3.8	split	68
3.9	Apply orbit file	69
3.10	Calibration	70
3.11	Deburst	70
3.12	Multi-looking	71
3.13	Terrain Correction	72
3.14	<i>Étapes de traitement SAR dans SNAP</i>	72

3.15	image GRD montrant notre site d'étude par le rectangle orange	73
3.16	Étapes de traitement des images GRD	73
3.17	accéder à subset sur SNAP 11	74
3.18	geo coordinates de notre subset	74
3.19	l'image GRD après subset	75
3.20	terrain correction 29/08/2021	75
3.21	accéder à calibrate sur SNAP 11	76
3.22	l'image du 29/08/2021 après calibration	76
3.23	l'image du 29/08/2021 après speckle filtering	76
3.24	Organigramme de détection des zones brûlées par Random Forest	77
3.25	les échantillons de classification sur l'image du 29/08/2021	78
3.26	classification random forest du 29/08/2021	78
3.27	classification random forest du 05/09/2022	79
3.28	l'image classifiée du 29/08/2021 sur PYTHON	82
3.29	l'image classifiée du 05/09/2022 sur PYTHON	82
3.30	les zones ayant subis des changements affichés sur Python	83
3.31	Carte de détection des changements post-incendie (forêt → végétation) . . .	83
3.32	Carte de détection des changements filtré post-incendie (forêt → végétation)	84
3.33	image optique prise par sentinel-2	84

Liste des tableaux

1.1	Table des bandes de fréquences radar	3
1.2	Paramètres orbitaux du satellite	7
1.3	Types de polarisation et leurs descriptions	14
1.4	Tableau des composantes de polarisation des ondes émises et reçues	15
2.1	Matrice de confusion pour un classifieur binaire	35
2.2	Fonctions d'erreur courantes en apprentissage automatique	43
2.3	Avantages et Inconvénients des SVM	52
2.4	Avantages et Inconvénients des réseaux de neurones convolutifs (CNN) . .	52
2.5	Avantages et Inconvénients des MLP (Multi-Layer Perceptron)	53
2.6	Avantages et Inconvénients de la méthode Random Forest	54
2.7	Avantages et inconvénients de l'algorithme Mean Shift	58

Introduction Générale

Dans un contexte mondial où la surveillance et la gestion durable des ressources naturelles sont devenues des préoccupations majeures, la télédétection satellitaire s'affirme comme un outil incontournable pour l'observation de la surface terrestre. Parmi les technologies les plus performantes, le radar à synthèse d'ouverture (SAR) se distingue par sa capacité à acquérir des images de haute résolution quelles que soient les conditions météorologiques ou d'éclairement. La mission Sentinel-1, développée par l'Agence Spatiale Européenne (ESA), met à disposition une grande quantité de données SAR qui ouvrent de nouvelles perspectives pour la cartographie et la détection des changements environnementaux.

Ce mémoire s'inscrit dans cette dynamique en exploitant les images SAR de Sentinel-1 pour la télédétection et la classification de la couverture terrestre, avec un focus particulier sur l'évaluation des changements post-incendie dans les zones forestières.

Le présent travail est structuré en trois chapitres :

Le premier chapitre introduit les notions fondamentales des images SAR. Il décrit le principe de fonctionnement du radar à synthèse d'ouverture, le satellite Sentinel-1 et ses caractéristiques techniques, la résolution des images, le traitement du bruit speckle et les concepts de polarimétrie qui permettent une meilleure interprétation des données radar.

Le deuxième chapitre est consacré aux différentes méthodes de classification utilisées pour l'analyse des images SAR. Il présente à la fois les approches supervisées (comme les SVM, les réseaux de neurones et les forêts aléatoires) et non supervisées (telles que le K-means et le Mean Shift), en détaillant leurs principes, leurs avantages et leurs limites.

Le troisième chapitre expose l'application pratique du traitement et de la classification des images SAR sur la zone d'étude. Il décrit le site d'étude, le prétraitement des données, le processus de classification réalisé sur SNAP et Python, et présente les résultats obtenus, accompagnés d'une analyse des changements détectés suite aux incendies.

À travers cette étude, nous mettons en évidence l'apport des images radar Sentinel-1 et des techniques de classification pour la surveillance et l'évaluation des impacts environnementaux, contribuant ainsi à une meilleure gestion des écosystèmes forestiers vulnérables.

Chapitre 1

Les Images SAR

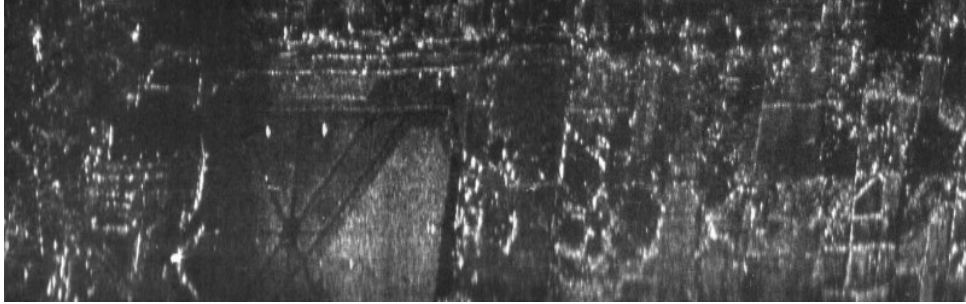
1.1 Introduction

Un radar est un dispositif électronique permettant de détecter, localiser et suivre des objets à distance grâce à l'utilisation d'ondes radio ou de micro-ondes. Le terme radar est l'abréviation de l'expression anglaise Radio Detection And Ranging, qui signifie en français Détection et Télémétrie par ondes radio. Son fonctionnement repose sur l'émission d'un signal électromagnétique, généralement constitué d'ondes radio ou de micro-ondes, qui se propage dans l'espace et se réfléchit lorsqu'il rencontre un obstacle. En analysant le temps écoulé entre l'émission et la réception du signal réfléchi, le radar est capable de mesurer la distance de l'objet détecté. Il peut également déterminer sa vitesse, sa direction de déplacement, et parfois même sa taille ou sa forme. Grâce à leur efficacité et leur précision, les radars sont utilisés dans de nombreux domaines, notamment la navigation aérienne, maritime et terrestre, la météorologie, les applications militaires, la surveillance du trafic, la détection d'obstacles ainsi que l'analyse des zones forestières... etc.

1.2 Radar à synthèse d'ouverture

Radar à synthèse d'ouverture SAR : Le Radar à synthèse d'ouverture appelé en anglais Synthetic Aperture Radar est une technologie radar capable de capturer et générer des images synthétiques de régions peu importe l'heure de la journée ou les conditions météorologiques de cette dernière. Le radar SAR a connu plusieurs systèmes d'acquisition à travers les années que ce soit des systèmes aériens (FSAR, AIRSAR) ou des systèmes spatiaux comme ceux des agences publiques telles que l'ESA avec Sentinel-1 et RADARSAT-2 et la NASA avec NISAR. Les premiers systèmes d'imagerie radar à

synthèse d'ouverture (SAR) ont vu le jour au début des années 1950, etc les premières captures ont été réalisées à la fin de cette période. La figure 1.1 présente la toute première image SAR, produite en 1957 par l'Université du Michigan à l'aide de lampes au mercure. Ce n'est qu'avec l'apparition du laser et les progrès en électronique et en informatique que la qualité des images s'est significativement améliorée. Depuis le lancement de SEASAR en 1978, premier radar SAR numérique embarqué sur un satellite, ces technologies ont apporté une quantité précieuse d'informations pour l'étude de notre planète.



[2]

FIGURE 1.1 – Première image SAR acquise en 1957 par l'université du Michigan

Ce système fonctionne généralement entre la bande P et la bande Ka et particulièrement dans la bande de fréquence L et la bande de fréquence C comme présenté dans le tableau ci-dessous.

P	0.025-0.390 GHz	133-76.9 cm
L	0.39-1.55 GHz	76.9-19.3 cm
S	1.55-4.20 GHz	19.3-7.1 cm
C	4.20-5.75 GHz	7.1-5.2 cm
X	5.75-10.90 GHz	5.2-2.7 cm
Ku	12.0-18.0 GHz	2.7-1.36 cm
Ka	22.0-36.0 GHz	1.36-0.83 cm

TABLE 1.1 – Table des bandes de fréquences radar

1.3 Principe de formation des images SAR

L'imagerie SAR (Synthetic Aperture Radar) repose sur l'utilisation d'un radar à antenne synthétique pour obtenir des données précises sur une zone donnée. Contrairement aux radars classiques dotés d'une antenne fixe, le SAR utilise une antenne en mouvement par rapport à la scène observée, ce qui permet d'améliorer la résolution des images.

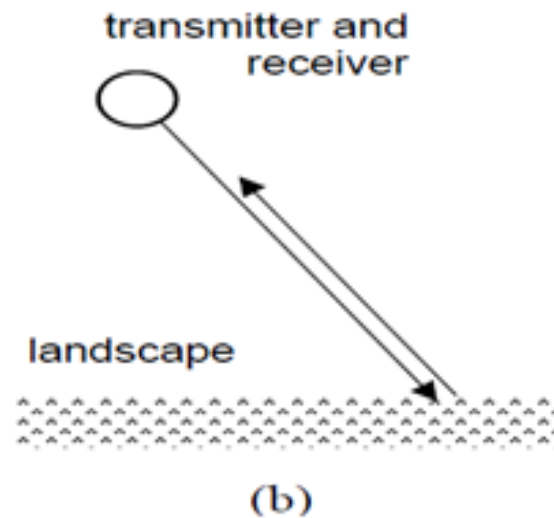


FIGURE 1.2 – Antenne non statique

Le processus de formation d'une image SAR se déroule en plusieurs étapes :

1. Émission du signal : Le radar SAR envoie des impulsions électromagnétiques (ondes radio ou micro-ondes) vers la scène cible. Ces impulsions sont émises de manière répétée selon une fréquence de répétition des impulsions (PRF), synchronisée avec la vitesse de déplacement de l'antenne.
2. Réception du signal réfléchi : L'antenne capte les signaux réfléchis par les objets présents dans la scène. Ces signaux contiennent des informations sur la distance des objets, l'intensité (amplitude) et la phase de l'onde réfléchie.
3. Échantillonnage et enregistrement des données : Les signaux reçus sont échantillonnés et enregistrés à différents instants, formant des échos radar. La fréquence de répétition des impulsions (PRF) influence la densité d'échantillonnage et la précision des données collectées.
4. Compression du signal : Pour optimiser le stockage et le traitement des données, les signaux bruts subissent une compression, généralement basée sur les caractéristiques du signal radar.
5. Correction des effets atmosphériques et topographiques : Les signaux enregistrés peuvent être altérés par des effets atmosphériques (comme l'humidité ou la pression) et des variations topographiques. Des corrections sont appliquées afin d'améliorer la précision de l'image finale.
6. Traitement de l'antenne synthétique : Le SAR simule une antenne virtuelle plus grande en combinant les échos captés sur une longue distance. Ce procédé améliore considérablement la résolution spatiale des images en traitant les données acquises à différents moments.

précises (mieux-contre 0,04) sur chaque axe, une grande précision d'attitude (environ 0,01 degré sur chaque axe) et une détermination de l'orbite en temps réel, ainsi qu'un système de propulsion dédié pour un contrôle précis de l'orbite. [4]

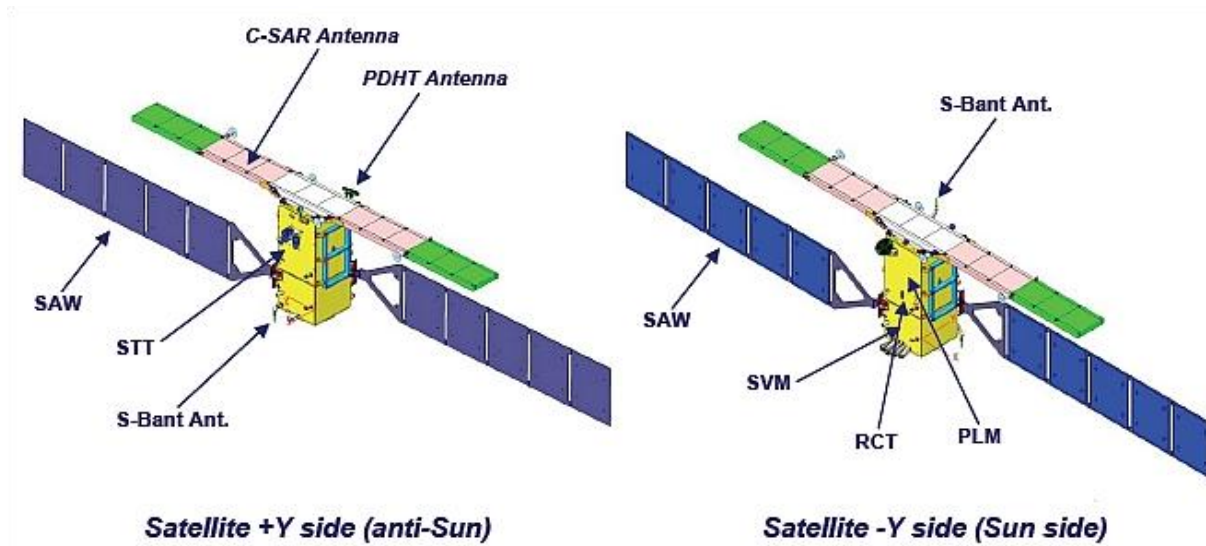


FIGURE 1.4 – Vue schématique de l'intensité spatiale Sentinel-1

L'orbite de référence sera maintenue à l'intérieur d'un tube orbital fixe sur la Terre d'un diamètre de 120 m (RMS) pendant l'exploitation normale.

La plate-forme satellite fournit des caractéristiques pour la gestion des systèmes de contrôle d'attitude et d'orbite, le traitement des données scientifiques avec une capacité de stockage de 1 410 Gb, la propulsion, la puissance, le contrôle thermique, l'autonomie des engins spatiaux et l'identification et la récupération de la détection des défaillances, et la communication avec le sol via la liaison descendante de données en X et la liaison optique par l'EDRS pour les données utiles à 520 Mbit/s. La masse totale de l'engin spatial au lancement est d'environ 2 300 kg (dont 130 kg de carburant).

1.4.1 Orbite

Sentinel-1 est sur une orbite proche de la polaire avec un cycle de répétition de 12 jours et 175 orbites par cycle pour un seul satellite. Sentinel-1A et Sentinel-1B partagent le même plan orbital avec une différence de phasage orbital de 180 degrés. Avec les deux satellites en fonctionnement, le cycle de répétition est de six jours.[5]

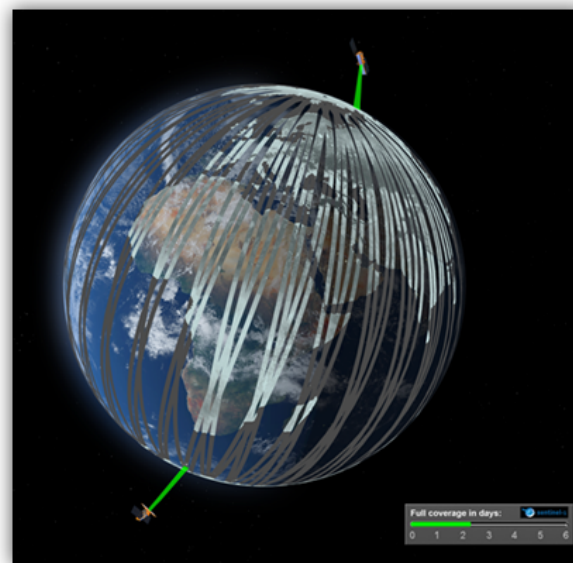


FIGURE 1.5 – Constellation de sentinel-1

Voici un tableau qui résume les paramètres orbitaux du satellite :

Altitude (km)	Inclinaison (°)	Période (min)	Cycle (jours)	Déviations (m)	Heure AN (h)
693	98.18	98,6	12	120	18

TABLE 1.2 – Paramètres orbitaux du satellite

1.5 Résolution

La sortie spatiale compte pour un grand pourcentage avec des systèmes d'imagerie où c'est la définition qui permet de distinguer les objets rapprochés. Une résolution élevée permet de distinguer et de commencer à séparer des objets jumeaux dans des cellules séparées et qu'une faible résolution conduit à la fusion des mêmes objets dans la même cellule. On refait principalement deux types de résolution : la résolution radiale et la résolution azimutale qui correspondent à la séparation des objets dans le plan orthogonal à l'axe de l'antenne.

1.5.1 Résolution radiale

La résolution radiale est valable pour la discrimination des objets sur la longueur de l'axe de l'antenne. La distance radiale au sol R_d fonction de l'angle d'incidence θ et de la durée des impulsions radar τ est donnée par : $R_d = \frac{c \tau}{2 \sin \theta}$. Pour obtenir une résolution spatiale optimale, on a besoin de systèmes radar à vision latérale (SLAR) car le pointeur nul n'apporte pas de résolution. La fantaisie d'un signal chirp (Compressed High Intensity Radar Pulse) a le résultat de l'amélioration de la résolution. Ce signal est exprimé par :

$$c(t) = p(t) \cos(w_0 + 0.5\alpha t)t = p(t) \cos(w_0 t + 0.5\alpha t^2) = p(t) \cos(2\pi f_0 t + 0.5\alpha t^2) \quad (1.1)$$

La montée de la résolution est due pour le reste à la compression impulsionnelle, faite par la corrélation entre le signal émis et le signal reçu.

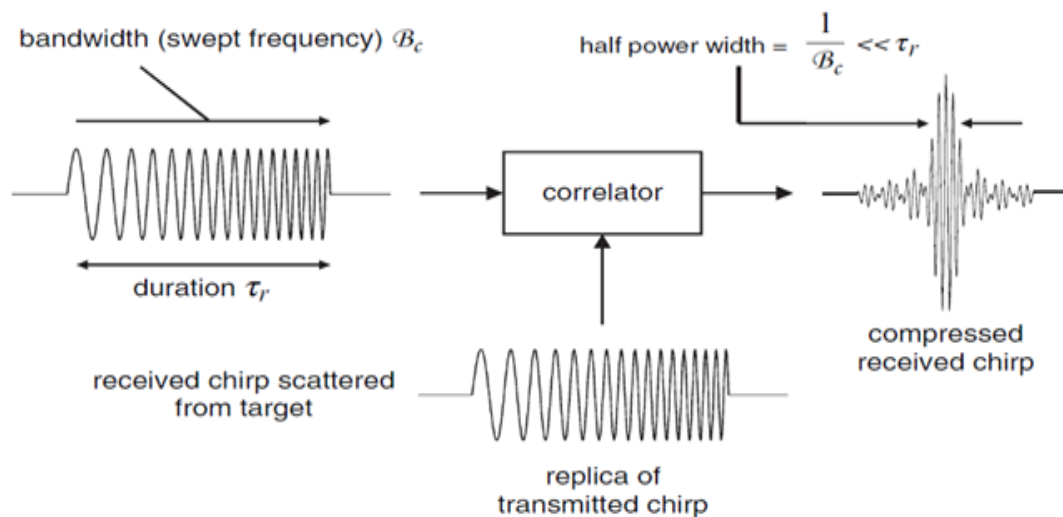


FIGURE 1.6 – Principe de la compression d'impulsion

1.5.2 Résolution azimutale

La résolution azimutale reflète la capacité à distinguer deux objets situés parallèlement à la trajectoire du radar. Elle correspond approximativement à la moitié de la longueur réelle de l'antenne, exprimée par la formule suivante : $R_a = \frac{l}{2}$. L'approche la plus simple pour améliorer cette résolution consiste à accroître la hauteur de l'antenne. Cependant, dans la pratique, augmenter la taille de celle-ci s'avère souvent complexe et peu efficace. C'est dans ce contexte que la technologie SAR (Synthetic Aperture Radar) entre en jeu. Elle permet de générer une ouverture synthétique en combinant les signaux rétrodiffusés

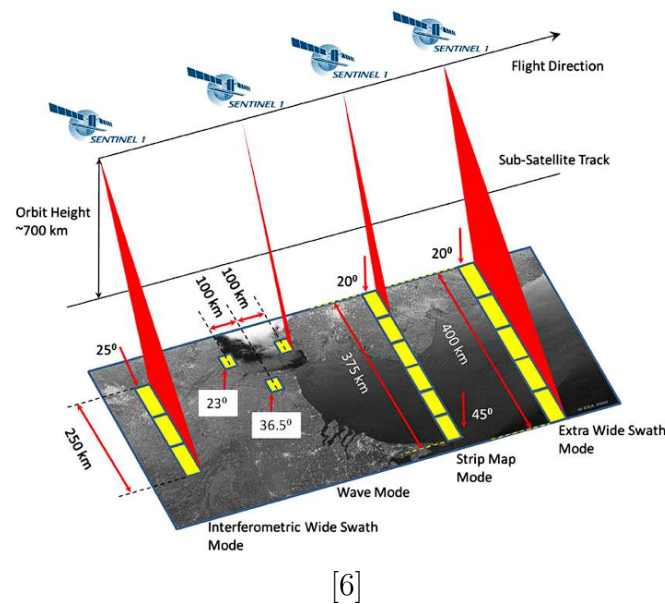


FIGURE 1.7 – Modes d'acquisitions d'images SAR

captés tout au long de la trajectoire du radar. Cette méthode simule ainsi une antenne bien plus grande que celle réellement utilisée. Les images produites par le SAR, tout comme les autres images numériques, sont constituées de pixels adjacents. La valeur attribuée à chaque pixel est déterminée par l'intensité du signal radar émis et réfléchi par la surface observée.

1.6 Bruit de l'image SAR

Lorsque l'onde radar est émise en direction de la cible, celle-ci renvoie non pas une seule onde mais plusieurs ondes qui interfèrent au niveau du radar de façon destructive, ces interférences affaiblissent les images produites par le SAR par un bruit appelé speckle. Le speckle a les caractéristiques d'un bruit multiplicatif. Pour bien extraire les informations thématiques et bien exploiter les images, il faut réduire ou éliminer ce bruit. D'où le développement de nombreuses méthodes de réduction du speckle, parmi ces méthodes le filtrage des images SAR. Un filtre idéal qui éliminerait tous les bruits n'existe pas en pratique, mais certains s'en rapprochent.

Les trois méthodes de filtrage du speckle sont :

- Méthode multi vues ou multi-looks : Il y a deux techniques pour cette méthode, la première est une technique spectrale qui permet de réduire la variance du speckle d'un facteur N au détriment de la résolution spatiale de l'image originale. La deuxième technique est une technique spatiale qui consiste à acquérir une image

pleine résolution SLC et à effectuer une moyenne de N pixels adjacents pour former une image basse résolution avec atténuation du speckle.

- Méthode Multifréquence, Multi-polarisation : utilise la matrice de covariance et les statistiques locales pour une zone homogène.
- Méthodes spatiales : Ces méthodes se subdivisent en deux familles, la première comporte les méthodes heuristiques qui permettent de réduire le speckle sans tenir compte de son caractère multiplicatif et sans connaissance à priori de ses lois de probabilité. La deuxième famille comporte les méthodes homomorphiques qui sont fondées sur le caractère multiplicatif du speckle.

1.7 L'équation Radar

L'équation radar est une formule fondamentale utilisée en télédétection et en radar pour déterminer la puissance du signal réfléchi par une cible et reçu par l'antenne du radar. Elle permet de comprendre comment la puissance du signal radar varie en fonction des paramètres du système et de l'environnement.

Voici la forme générale de l'équation radar :

$$P_r = \frac{P_t \cdot G_t \cdot G_r \cdot \lambda^2 \cdot \sigma}{(4\pi)^3 \cdot R^4 \cdot L} \quad (1.2)$$

Cette équation est essentielle pour concevoir des systèmes radar efficaces et comprendre leurs limites opérationnelles, évaluer la détectabilité des cibles, et optimiser les performances pour des applications spécifiques (militaires, météorologiques, civiles, etc...).

1.8 La polarimétrie

La polarimétrie est une technique qui exploite les propriétés de polarisation des ondes électromagnétiques pour améliorer la transmission et la réception des signaux. Elle est notamment utilisée dans les systèmes radar. La polarisation d'une onde décrit l'orientation du champ électrique dans l'espace (linéaire, circulaire ou elliptique).

En polarimétrie, on utilise différentes polarisations pour :

- Augmenter la capacité des canaux de communication (multiplexage polarimétrique).
- Réduire les interférences entre plusieurs signaux.
- Améliorer la détection dans les systèmes radar (discrimination des cibles).

1.8.1 Les ondes électromagnétiques

Les ondes électromagnétiques (OEM) sont des perturbations couplées des champs électrique (E) et magnétique (B) qui se propagent dans le vide ou dans un milieu matériel sans nécessiter de support matériel. Elles transportent de l'énergie et de l'information. Les lois fondamentales de l'électromagnétisme, telles que la propagation des ondes et leur interaction avec la matière, sont décrites par les équations de Maxwell.

Les équations de Maxwell expriment les relations fondamentales suivantes :[1]

$$\vec{\nabla} \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \quad (1.3)$$

$$\vec{\nabla} \times \vec{B} = \mu_0 \vec{J} + \mu_0 \varepsilon_0 \frac{\partial \vec{E}}{\partial t} \quad (1.4)$$

$$\vec{\nabla} \cdot \vec{E} = \frac{\rho}{\varepsilon_0} \quad (1.5)$$

$$\vec{\nabla} \cdot \vec{B} = 0 \quad (1.6)$$

ε_0 : permittivité de vide.

μ_0 : perméabilité magnétique dans le vide.

ρ : densité de charge.

J : densité de charge.

Dans le cas d'absence de charge $\rho = 0$, $J = 0$ et pour un milieu homogène, on peut obtenir une seule équation en combinant ces équations qui décrit la propagation du champ électrique, formulée de la manière suivante :

$$\nabla^2 \vec{E} + \frac{1}{c^2} \frac{\partial^2 \vec{E}}{\partial t^2} \quad (1.7)$$

1.8.2 Les types de polarisation

La polarisation est une propriété de la lumière. Cette dernière est une onde électromagnétique se déplaçant dans le vide à la vitesse $c=299\,792\,458$ m/s. Elle est composée d'un champ électrique (noté généralement E) et d'un champ magnétique (noté B), orthogonaux. À travers les équations de Maxwell, les champs E et B sont liés : la connaissance de l'un

suffit pour connaître l'autre. Aussi, pour simplifier — et c'est aussi la convention choisie en polarimétrie — on ne raisonne que sur le champ E .

Une onde électromagnétique plane se définit notamment par sa direction de propagation. Le plan perpendiculaire à la direction de propagation est appelé le plan d'onde. C'est dans le plan d'onde qu'évolue le champ E (et le champ B , mais nous n'en parlerons plus). À chaque instant, le champ E a une amplitude et une direction différente dans le plan d'onde. Dit autrement, si l'onde se propage en direction de l'observateur ce dernier verra le champ E former différents motifs dans le plan d'onde pendant son évolution temporelle. C'est cela qui va définir la polarisation de l'onde.

1. Polarisation elliptique : Si le champ E dessine une ellipse dans le plan d'onde, on parle de polarisation elliptique.

C'est le cas le plus général. On peut alors décomposer le champ électrique selon deux composantes perpendiculaires :

$$E_x = E_{x0} \cos(kz - \omega t) \quad (1.8)$$

$$E_y = E_{y0} \cos(kz - \omega t + \varphi), \text{ où } \omega = \frac{2\pi\lambda}{c} \quad (1.9)$$

est la pulsation de l'onde électromagnétique.

$$k = \frac{2\pi}{\lambda} \quad (1.10)$$

le nombre d'onde et φ le déphasage entre les deux composantes.

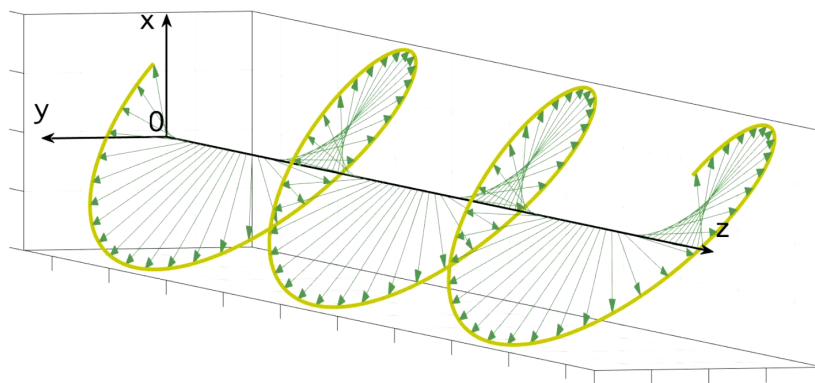


FIGURE 1.8 – Polarisation elliptique

2. Polarisation circulaire : Si le champ E dessine un cercle dans le plan d'onde, on parle de polarisation circulaire. Le sens de rotation de E définit une polarisation :

- Droite, si l'onde tourne dans le sens trigonométrique.

- Gauche, si l'onde tourne dans le sens horaire.

On peut se souvenir du sens de rotation en utilisant ses mains. En pointant son pouce vers soi, on regarde dans quel sens s'enroulent les autres doigts ; en choisissant la main qui permet de reproduire le sens de rotation de l'onde polarisée (main gauche ou main droite), on détermine le sens de la polarisation.

On peut alors écrire les composantes du champ électrique comme suit :

$$E_x = E_0 \cos(kz - \omega t) \quad (1.11)$$

$$E_y = E_0 \sin(kz - \omega t) \quad (1.12)$$

où E_0 est l'amplitude commune et $\varphi = \frac{\pi}{2}$.

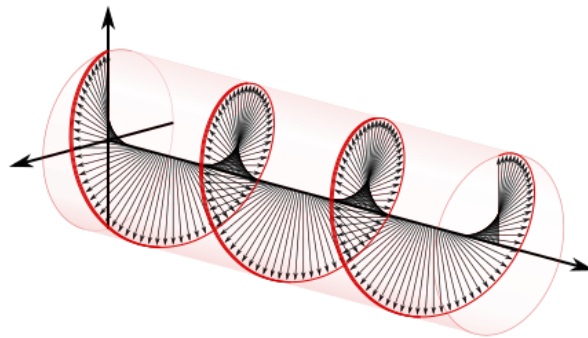


FIGURE 1.9 – Polarisation circulaire

3. Polarisation linéaire : Si le champ E décrit un segment dans le plan d'onde, on dit que la polarisation est linéaire. On peut la voir comme une polarisation elliptique pour laquelle l'un des deux axes de l'ellipse de polarisation serait réduit à un point, comme montré dans la figure 1.10 Le champ électrique s'écrit alors :

$$E_x = E_{x0} \cos(kz - \omega t) \quad E_y = E_{y0} \cos(kz - \omega t) \quad \text{avec} \quad \frac{E_{x0}}{E_{y0}} = \text{cste} \quad \text{et} \quad \varphi = 0. \quad (1.13)$$

4. Polarisation aléatoire : Le champ E varie de manière imprévisible sans direction privilégiée (lumière naturelle, sources thermiques).

On peut combiner les polarisations des ondes transmises et reçues en utilisant les paramètres H et V dont l'interrelation est bien définie. C'est pourquoi les systèmes qui

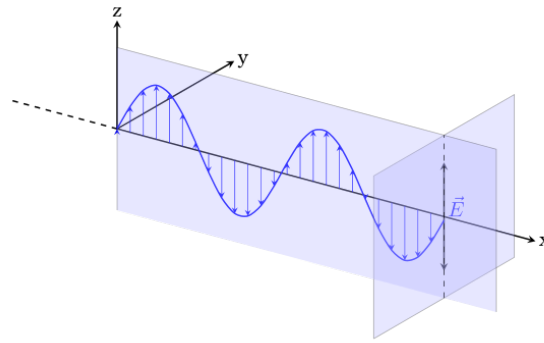


FIGURE 1.10 – Polarisation linéaire

transmettent et reçoivent ces deux polarisations linéaires sont couramment utilisés. Avec ce type de radar, il existe quatre combinaisons de polarisations de transmission-réception :

HH - transmission et réception horizontales

VV - transmission et réception verticales

HV - transmission horizontale et réception verticale, et

VH - transmission verticale et réception horizontale.

Polarisation	Description
Polarisation simple	- HH ou VV (éventuellement HV ou VH)
Bipolarisation	- HH et HV, VV et VH ou HH et VV
Polarisation alternée	- HH et HV alternant avec VV et VH
Polarimétrique	- HH, VV, HV et VH

TABLE 1.3 – Types de polarisation et leurs descriptions

1.8.3 La matrice de Jones

Le vecteur de Jones est un outil fondamental pour décrire mathématiquement l'état de polarisation d'une onde électromagnétique complètement polarisée. Les polarisations horizontale (H) et verticale (V) en sont les bases utilisées dans l'imagerie SAR.

Pour une onde plane se propageant selon l'axe \mathbf{Z} le champ électrique s'écrit :

$$\mathbf{E}(t) = \begin{pmatrix} E_x(t) \\ E_y(t) \end{pmatrix} = \begin{pmatrix} |E_x|e^{i(\phi_x - \omega t)} \\ |E_y|e^{i(\phi_y - \omega t)} \end{pmatrix} \quad (1.14)$$

Donc, sans la dépendance temporelle, le vecteur de **jones** s'écrit de la manière suivante :

$$\mathbf{J} = \begin{pmatrix} E_x \\ E_y \end{pmatrix} = \begin{pmatrix} |E_x|e^{i\phi_x} \\ |E_y|e^{i\phi_y} \end{pmatrix} \quad (1.15)$$

1.8.4 La matrice de diffusion

La matrice de diffusion ou matrice **S** appelé aussi matrice de SINCLAIR est un outil central pour décrire les interactions entre particules. Elle permet de calculer les amplitudes de transition entre des états asymptotiques (avant et après une interaction). Lorsque des ondes électromagnétiques se déplacent et entrent en contact avec un obstacle, elles interagissent avec l'obstacle en question. Au cours de cette rencontre, une partie de l'énergie de l'onde peut être absorbée par l'objet, tandis que le surplus est renvoyé sous forme de réflexion ou de diffusion. Ces phénomènes peuvent altérer la polarisation de l'onde réfléchie comparativement à celle de l'onde incidente. En analysant ces variations, on peut en déduire certaines caractéristiques de l'objet touché par le rayonnement. Dans l'imagerie SAR, on émet deux polarisations **H** ou **V** et la réception se fait sur deux canaux **H** et **V**.

Polarisation de l'onde émise	Polarisation de l'onde reçue	Composante mesurée
H	H	S_{HH}
	V	S_{HV}
V	H	S_{VH}
	V	S_{VV}

TABLE 1.4 – Tableau des composantes de polarisation des ondes émises et reçues

La matrice de diffusion est représentée par $\mathbf{S} = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix}$ en plus de ses complexes S_{ij} , les coefficients S_{11} et S_{22} représentent la même polarisation en émission et en réception, tandis que les coefficients S_{12} et S_{21} représentent une polarisation opposée entre émission et réception comme présentés dans la matrice ci-dessous :

$$\mathbf{S} = \begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix}$$

1.8.5 Le vecteur cible

Le vecteur cible constitue une représentation compacte de l'information polarimétrique contenue dans la matrice de diffusion **S**. Sa construction repose sur une projection mathématique

permettant d'extraire les caractéristiques essentielles de la cible observée.

Principe de construction

Le vecteur cible est obtenu par projection de la matrice \mathbf{S} sur une base orthogonale de matrices $\boldsymbol{\psi}$ appartenant à un groupe spécial unitaire [?, ?]. Cette opération s'exprime par :

$$\mathbf{k} = \frac{1}{2} \text{Trace}(\mathbf{S}\boldsymbol{\psi}) = [k_0, k_1, k_2, k_3]^T \quad (1.16)$$

Bases de projection usuelles

Deux bases principales sont couramment utilisées :

- a) **Base lexicographique** ($\boldsymbol{\psi}_L$) : Représentation directe des éléments de la matrice \mathbf{S} :

$$\boldsymbol{\psi}_L = \left\{ 2 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, 2 \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, 2 \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, 2 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \right\} \quad (1.17)$$

- b) **Base de Pauli modifiée** ($\boldsymbol{\psi}_P$) : Représentation physique des mécanismes de diffusion :

$$\boldsymbol{\psi}_P = \sqrt{2} \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & -j \\ j & 0 \end{bmatrix} \right\} \quad (1.18)$$

Expressions des vecteurs cibles

Forme générale (bistatique) :

$$\mathbf{k}_L = [S_{hh} \ S_{hv} \ S_{vh} \ S_{vv}]^T \quad (1.19)$$

$$\mathbf{k}_P = \frac{1}{\sqrt{2}} [S_{hh} + S_{vv} \ S_{hh} - S_{vv} \ S_{hv} + S_{vh} \ j(S_{hv} - S_{vh})]^T \quad (1.20)$$

Cas monostatique ($S_{hv} = S_{vh}$) :

$$\mathbf{k}_L = [S_{hh} \ \sqrt{2}S_{hv} \ S_{vv}]^T \quad (1.21)$$

$$\mathbf{k}_P = \frac{1}{\sqrt{2}} [S_{hh} + S_{vv} \ S_{hh} - S_{vv} \ 2S_{hv}]^T \quad (1.22)$$

Relation entre représentations

Les vecteurs \mathbf{k}_L et \mathbf{k}_P sont reliés par une transformation linéaire :

$$\mathbf{k}_L = \mathbf{A}\mathbf{k}_P = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & \sqrt{2} \\ 1 & -1 & 0 \end{bmatrix} \mathbf{k}_P \quad (1.23)$$

Cette formulation matricielle permet de convertir aisément entre les différentes représentations, offrant ainsi une grande flexibilité dans le traitement des données polarimétriques.

1.9 Les matrices de cohérence et de covariance

En polarimétrie radar, les matrices de covariance et de cohérence constituent deux représentations fondamentales d'une cible. Leur intérêt réside dans leurs propriétés mathématiques remarquables. Ces deux matrices sont complexes, de taille 3×3 [7].

1.9.1 Matrice de covariance

La matrice de covariance \mathbf{C} est définie à partir du vecteur cible \mathbf{k}_L :

$$\mathbf{C} = \langle \mathbf{k}_L \mathbf{k}_L^{*T} \rangle = \begin{bmatrix} \langle S_{hh} S_{hh}^* \rangle & \sqrt{2} \langle S_{hh} S_{hv}^* \rangle & \langle S_{hh} S_{vv}^* \rangle \\ \sqrt{2} \langle S_{hv} S_{hh}^* \rangle & 2 \langle S_{hv} S_{hv}^* \rangle & \sqrt{2} \langle S_{hv} S_{vv}^* \rangle \\ \langle S_{vv} S_{hh}^* \rangle & \sqrt{2} \langle S_{vv} S_{hv}^* \rangle & \langle S_{vv} S_{vv}^* \rangle \end{bmatrix} \quad (1.24)$$

1.9.2 Matrice de cohérence

De façon équivalente, la matrice de cohérence \mathbf{T} peut être exprimée à partir du vecteur \mathbf{k}_P :

$$\mathbf{T} = \langle \mathbf{k}_P \mathbf{k}_P^{*T} \rangle = \begin{bmatrix} \langle (S_{hh} + S_{vv})(S_{hh} + S_{vv})^* \rangle & \langle (S_{hh} + S_{vv})(S_{hh} - S_{vv})^* \rangle & 2 \langle S_{hv}^* (S_{hh} + S_{vv}) \rangle \\ \langle (S_{hh} - S_{vv})(S_{hh} + S_{vv})^* \rangle & \langle (S_{hh} - S_{vv})(S_{hh} - S_{vv})^* \rangle & 2 \langle S_{hv}^* (S_{hh} - S_{vv}) \rangle \\ 2 \langle S_{hv} (S_{hh} + S_{vv})^* \rangle & 2 \langle S_{hv} (S_{hh} - S_{vv})^* \rangle & 4 \langle S_{hv} S_{hv}^* \rangle \end{bmatrix} \quad (1.25)$$

1.9.3 La polarimétrie dans Sentinel-1

Sentinel-1 est un système de double polarisation SAR préservant en phase. Il peut émettre un signal en polarisation horizontale (H) ou verticale (V), puis recevoir à la fois dans les polarisations H et V. Les produits à double polarisation du complexe à simple apparence (SLC) de niveau 1 contiennent des valeurs complexes. En plus de l'intensité de rétrodiffusion qui peut être mesurée à partir de chaque polarisation unique, les informations de phase inter-canaux permettent d'effectuer une analyse améliorée des propriétés de rétrodiffusion.



FIGURE 1.11 – Image acquise par Copernicus Sentinel-1A le 2017-11-01 avec d'intenseur VV, image d'ambiance VH et composite de couleur RGB

1.9.4 le rôle de la polarisation dans les forêts

Il est essentiel de considérer la polarisation des ondes radar lorsqu'elles interagissent avec les forêts, car cela détermine la manière dont le signal interagit avec les troncs et les composantes de la canopée. La Figure 1.12 présente un schéma simplifié montrant comment les ondes longues et courtes en polarisations horizontale (H) et verticale (V) interagissent avec les forêts.

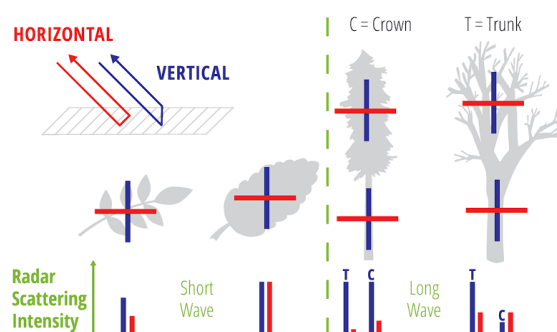


FIGURE 1.12 – schéma des effets de la polarisations dans les forêts

L'aspect le plus important est que la rétrodiffusion en co-polarisation (VV, HH) (c'est-à-dire avec les mêmes composantes d'émission et de réception) est généralement plus forte

pour les composantes de diffusion de surface, tandis que l'énergie mesurée en polarisation croisée (VH ou HV) (c'est-à-dire lorsque l'onde reçue est perpendiculaire à l'onde émise) est associée à la diffusion volumique. Ainsi, pour les applications liées à la biomasse, au suivi de la dégradation forestière et à l'identification des changements entre volumes et surfaces, les observations en polarisation croisée avec des images SAR sont indispensables. Les différences entre les images en co-polarisation et en polarisation croisée (bandes C et L) d'une plantation de palmiers à huile sont visibles dans la Figure 1.13.

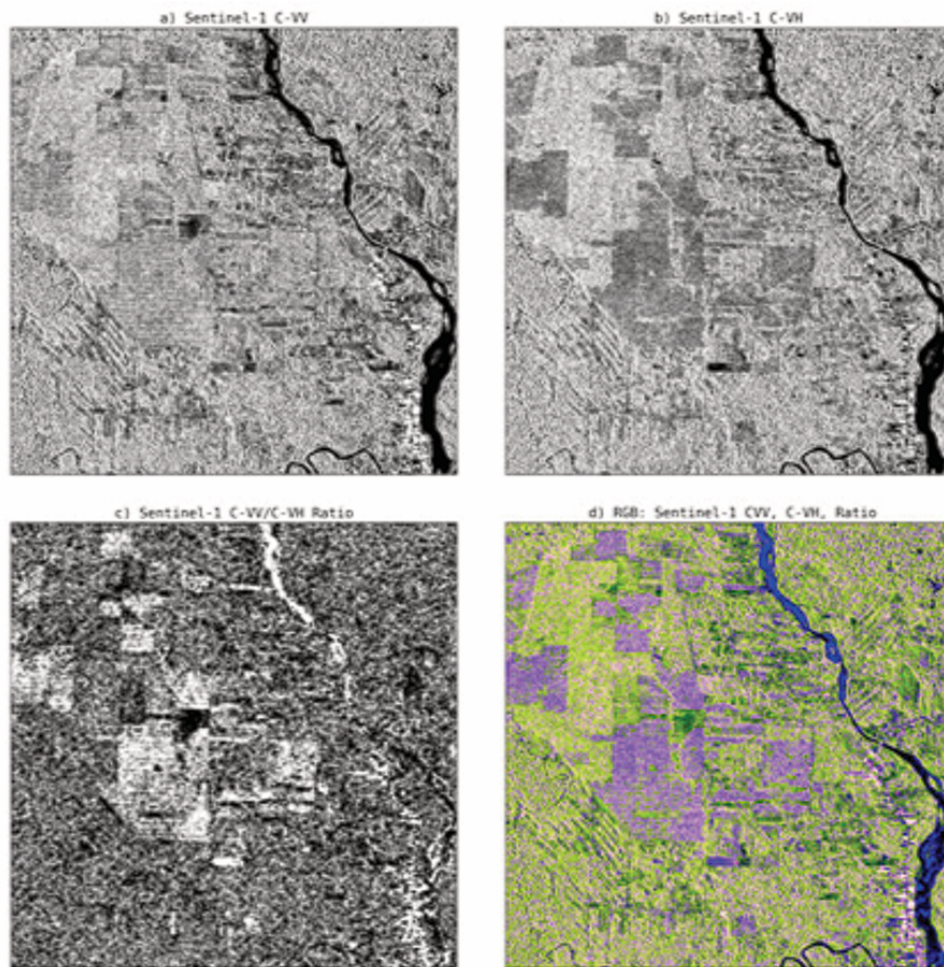


FIGURE 1.13 – Sentinel-1 C-band imagery (a) C_{VV} , (b) C_{VH} , (c) ratio, and (d) RGB composite C_{VV} C_{VH} ratio.

On observe clairement, aussi bien en L-HH qu'en C-VV, des ambiguïtés importantes dans les niveaux de gris entre les canopées forestières et les zones non forestières. En revanche, ces distinctions sont plus nettes et moins ambiguës dans les images en polarisation croisée.[8]

1.10 La décomposition polarimétrique

La décomposition polarimétrique est une technique essentielle en télédétection radar, permettant d'analyser la diffusion des ondes électromagnétiques par une cible. En exploitant les informations de polarisation, cette méthode décompose la matrice de diffusion en éléments cohérents ou incohérents, révélant ainsi les propriétés physiques et géométriques des surfaces observées. Les décompositions courantes, comme celles de Pauli, Krogager ou Freeman, facilitent l'interprétation des données polarimétriques pour des applications en agriculture, foresterie ou surveillance environnementale. Cette approche améliore significativement la caractérisation des cibles et leur classification.

1.10.1 Décomposition de pauli

Décomposition de Pauli La **décomposition de Pauli** est une méthode fondamentale en imagerie polarimétrique radar, utilisée pour décomposer la matrice de diffusion (ou matrice de Mueller) d'une cible en une somme de composantes élémentaires correspondant à des mécanismes de diffusion simples. Elle repose sur les **matrices de Pauli**, une base de matrices 2×2 utilisées en mécanique quantique et en traitement du signal radar.

Dans le cas d'une cible décrite par la **matrice de diffusion** $[S]$, la décomposition de Pauli l'exprime comme une combinaison linéaire de trois composantes principales :

- **Diffusion simple (ou surface)** : associée à une réflexion simple (typiquement une surface plane).
- **Diffusion double (ou dièdre)** : correspondant à une réflexion double (comme un coin formé par deux surfaces).
- **Diffusion volumique (ou helix)** : liée à des interactions multiples (végétation, forêt).

Cette décomposition permet une **visualisation en fausses couleurs** (RGB) où chaque mécanisme est représenté par une couleur distincte, facilitant l'interprétation des données polarimétriques. Elle est particulièrement utile pour la **classification des sols**, la **détection de structures artificielles** et l'**analyse de la végétation**. La base de Pauli en imagerie radar polarimétrique s'exprime sous la forme :

$$B_P = \{S_{P1}, S_{P2}, S_{P3}, S_{P4}\}$$

avec les matrices de base définies par :

$$B_P = \left\{ \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -j \\ j & 0 \end{pmatrix} \right\}$$

En configuration monostatique, le principe de réciprocité s'applique, conduisant à l'égalité $S_{HV} = S_{VH}$.

La décomposition de la matrice de diffusion S dans cette base s'écrit :

$$S = \alpha S_{P1} + \beta S_{P2} + \gamma S_{P3}$$

où les coefficients complexes sont déterminés par :

$$\begin{cases} \alpha = \frac{S_{HH} + S_{VV}}{2} \\ \beta = \frac{S_{HH} - S_{VV}}{2} \\ \gamma = S_{HV} \end{cases}$$

Cette décomposition admet une double interprétation physique :

- S_{P1} représente la diffusion par un trièdre (mécanisme de simple rebond)
- S_{P2} correspond à un dièdre à 0° (mécanisme de double rebond)
- S_{P3} modélise un dièdre à 45° (phénomène de dépolarisation)

Les éléments de la base de Pauli permettent ainsi une analyse physique des mécanismes de diffusion dominants dans une image radar polarimétrique.

1.10.2 Méthodes de Décomposition en Dual-Pol

Plusieurs méthodes existent pour décomposer le signal Dual-Pol et extraire des informations :

Décomposition de Pauli (adaptée à Dual-Pol)

Permet de séparer les mécanismes de diffusion dominants :

$$\begin{cases} k_1 = \frac{S_{HH} + S_{VV}}{\sqrt{2}} & \text{(Diffusion surfacique)} \\ k_2 = \frac{S_{HH} - S_{VV}}{\sqrt{2}} & \text{(Diffusion dièdre)} \\ k_3 = \frac{S_{HV} + S_{VH}}{\sqrt{2}} & \text{(Diffusion volumique)} \end{cases}$$

(En Dual-Pol, certaines composantes sont absentes, donc on adapte les équations.)

Décomposition en Cohérence (Dual-Pol Entropy/Alpha)

Utilisée pour estimer le degré d'aléatoire de la diffusion :

- **Entropie (H)** : Mesure le désordre de la diffusion
- **Angle Alpha (α)** : Indique le type de mécanisme dominant (surface, volume, dièdre)

Décomposition de Freeman-Durden (adaptée)

Sépare la diffusion en trois composantes :

- Diffusion de surface (sol lisse, eau)
- Diffusion de volume (forêt, végétation dense)
- Diffusion double-reflet (bâtiments, troncs d'arbres)

(En Dual-Pol, cette décomposition est simplifiée car certaines polarisations manquent.)

Indices Polarimétriques (Dual-Pol)

- **Ratio HH/HV** : Permet de distinguer les zones urbaines (HH dominant) des forêts (HV élevé)
- **Indice de Différence Normalisé (NDPI)** : Utile pour la détection des changements

1.11 Conclusion

Le Chapitre 1 a présenté une analyse approfondie des principes fondamentaux des images SAR (Synthetic Aperture Radar), en mettant en lumière leur fonctionnement, leurs caractéristiques techniques et leurs applications. Les points clés abordés incluent :

- **Fonctionnement des SAR** :
 - Utilisation d'antennes en mouvement pour générer des images haute résolution
 - Processus complet : émission d'impulsions, réception des signaux, et traitement des données
- **Satellite Sentinel-1** :

- Caractéristiques techniques et paramètres orbitaux
- Capacités polarimétriques pour l'observation terrestre
- **Résolution et bruit :**
 - Importance de la résolution radiale/azimutale
 - Techniques de réduction du bruit de speckle
- **Polarimétrie :**
 - Types de polarisation (HH, VV, HV, VH)
 - Matrices de Jones et de diffusion pour l'analyse des cibles
- **Applications forestières :**
 - Polarisation croisée (HV/VH) pour l'étude de la biomasse
 - Distinction entre diffusion de surface et volumique
- **Décomposition polarimétrique :**
 - Méthode de Pauli pour séparer les mécanismes de diffusion
 - Visualisation en fausses couleurs (RGB)

En résumé, ce chapitre a établi les bases théoriques et techniques de l'imagerie SAR, essentielles pour des applications en télédétection environnementale. Ces concepts fondamentaux préparent le terrain pour les méthodes de classification abordées dans les chapitres suivants.

Chapitre 2

Méthodes de classification

2.1 introduction

La classification, un des sujets majeurs de l'apprentissage automatique, concerne tout recoupage en groupes de choses ou de données conformément à ses attributs. Cette méthode permet d'expliquer et d'identifier la nature des phénomènes observés, comme ceux qu'il est possible d'observer dans un image ou une coupure de données radar. La classification est à distinguer en deux types de classification : supervisée et non supervisée qui permet de traiter des gros lots de données de quatre grosses masses de données et de mieux comprendre les objets terrestres.

- **La classification supervisée :**

La classification supervisée est l'une des approches les plus répandues en apprentissage automatique. Elle repose sur un processus dans lequel un algorithme est entraîné à partir d'un ensemble de données étiquetées, c'est-à-dire composées d'exemples pour lesquels on connaît déjà la classe à laquelle ils appartiennent.

L'objectif principal est de développer un modèle capable de prédire la classe correcte de nouveaux exemples, encore non étiquetés.

Dans ce cadre, chaque exemple d'apprentissage est défini par :

- Un ensemble de caractéristiques (ou attributs)
- Et une étiquette de classe associée.

L'algorithme d'apprentissage va utiliser ces données pour apprendre une fonction de classification, c'est-à-dire une fonction capable de faire le lien entre les caractéristiques et les classes. Une fois ce modèle entraîné, il peut ensuite être utilisé pour prédire la classe de nouvelles données.

- **Quelques méthodes de classification supervisée :**

- SVM (Support Vector Machine) : cette méthode cherche à trouver la frontière optimale entre les différentes classes. Elle est très efficace, notamment pour les problèmes de classification binaire.
- Réseaux de neurones (Neural Networks) : ils sont capables de modéliser des relations complexes entre les données. Lorsqu'ils sont constitués de plusieurs couches (réseaux profonds), on parle de Deep Learning, particulièrement efficace dans le traitement d'images et la reconnaissance de formes.
- Le MLP (Multi-Layer Perceptron) : est un réseau de neurones artificiels utilisé en apprentissage supervisé pour la classification et la régression, capable de modéliser des relations complexes entre les données grâce à ses couches cachées.
- Random Forest : c'est un ensemble d'arbres de décision utilisés en parallèle pour améliorer la précision du modèle. Elle est robuste face au sur-apprentissage et adaptée aux grandes bases de données.

2.2 Les méthodes de classification supervisées

2.2.1 La méthode de classification SVM (Support Vector Machine)

Le Support Vector Machine (SVM) est un algorithme d'apprentissage supervisé largement utilisé pour des tâches de classification et de régression. Il s'est imposé comme un outil performant en apprentissage automatique, notamment grâce à sa capacité à gérer des problèmes de classification complexes avec efficacité.

L'objectif du SVM est de maximiser la marge entre les données d'apprentissage les plus proches de l'hyperplan — appelées vecteurs de support — et l'hyperplan lui-même. Ces vecteurs de support jouent un rôle crucial, car ce sont eux qui définissent précisément la position de l'hyperplan optimal.

L'un des grands avantages du SVM réside dans sa capacité à généraliser efficacement, même à partir d'un nombre restreint d'exemples d'apprentissage. Cela signifie qu'il peut classer avec précision de nouvelles données jamais vues auparavant, ce qui en fait un outil particulièrement adapté aux applications à grande échelle.

Le SVM peut traiter aussi bien des données linéairement séparables que non linéairement séparables :

- Dans le premier cas, un hyperplan linéaire suffit.

- Dans le second cas, le SVM fait appel à des fonctions noyaux (ou kernels) pour projeter les données dans un espace de dimension supérieure, où une séparation linéaire devient possible.
- Dans le troisième cas, classification SVM multi classes.

- **Premier cas** : classes linéairement s'éparables

Imaginons un ensemble de données représentant différents types d'avions, caractérisés par deux attributs principaux : la vitesse maximale et la portée de vol. L'objectif est de classer ces avions en deux catégories distinctes : « **avion de chasse** » et « **avion de ligne** » .

Pour ce faire, on cherche à construire un hyperplan linéaire de la forme :

$$f(x) = w^T \phi(x) + b \quad (2.1)$$

Cet hyperplan permet de prédire à quelle classe appartient un avion donné, en appliquant une règle de classification basée sur la position de l'avion par rapport à la frontière définie par

$$g(x) = \begin{cases} +1 & \text{si } f(x) > 0 \\ -1 & \text{si } f(x) < 0 \end{cases} \quad (2.2)$$

Dans notre exemple :

Si : $f(x) > 0$ donc : cette classe correspond 'a des avions de chasse

Si : $f(x) < 0$ donc : cette classe correspond 'a des avions de ligne

Il est possible de tracer plusieurs hyperplans capables de séparer parfaitement les deux classes. Toutefois, l'objectif du SVM est de déterminer l'hyperplan optimal, c'est-à-dire celui qui maximise la marge entre les exemples des deux classes. Pour cela, il s'agit de calculer les poids optimaux de l'hyperplan à l'aide de techniques d'optimisation numérique, telles que la descente de gradient.

Les vecteurs de support jouent un rôle fondamental dans cette démarche. En effet, ce sont les exemples les plus proches de la frontière de décision, donc les plus difficiles à classer correctement. Ces points sont souvent situés dans des zones où les classes sont proches, voire se chevauchent légèrement.

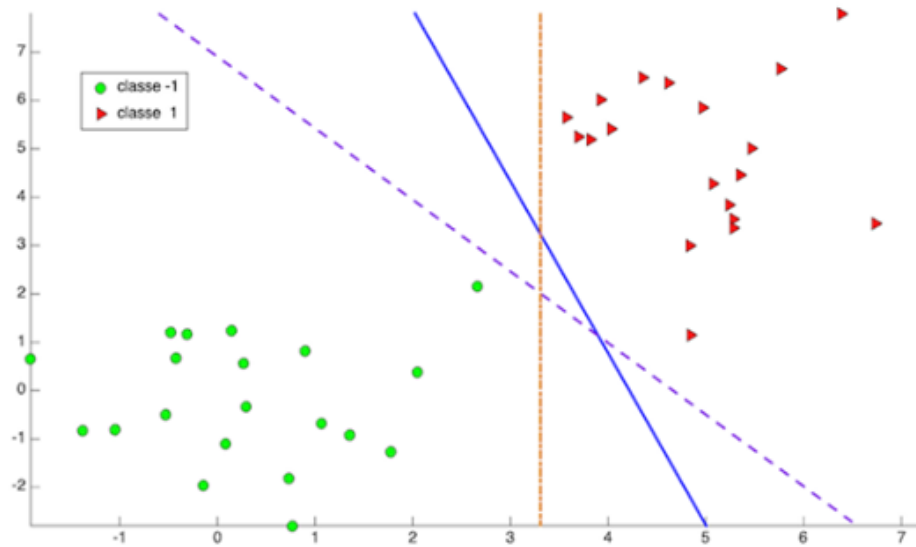


FIGURE 2.1 – deux classes linéairement séparables

En se focalisant sur ces vecteurs de support, le SVM cherche à maximiser la marge, c'est-à-dire à augmenter la distance entre ces points critiques et l'hyperplan. Ce mécanisme permet d'assurer une meilleure séparation des classes et améliore ainsi la capacité de généralisation du modèle sur de nouvelles données.

Par ailleurs, l'utilisation des vecteurs de support contribue à rendre l'algorithme plus efficace, en réduisant le nombre total de points nécessaires pour définir l'hyperplan. Cela se traduit par des économies de mémoire et un temps de calcul réduit lors de l'application du modèle à de nouveaux exemples.

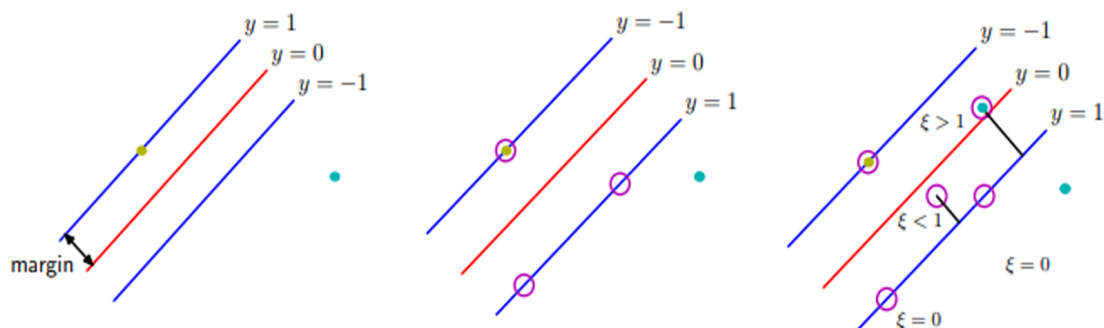


FIGURE 2.2 – Processus de maximisation de la marge

Une fois l'hyperplan de décision déterminé, il devient possible de classer de nouveaux exemples en les projetant dans l'espace des attributs. La classification se fait en identifiant de quel côté de l'hyperplan se trouve chaque exemple. Si l'exemple est positionné du côté correspondant à la classe « avion de chasse », il sera alors classé comme tel. À l'inverse, s'il se situe du côté associé aux « avions de ligne », il sera attribué à cette catégorie.

- **Deuxième cas :** Classes non linéairement séparables

Dans les situations où les données ne peuvent pas être séparées de manière linéaire dans leur espace d'origine, le SVM non linéaire intervient comme une solution efficace. Contrairement à la méthode linéaire qui cherche à tracer un hyperplan droit entre les classes, cette approche repose sur l'utilisation de transformations de données permettant de projeter les exemples dans un espace de dimension supérieure, où une séparation linéaire devient possible.

Comme illustré dans la figure 2.3, il est évident qu'aucun hyperplan linéaire ne permettrait de séparer correctement les deux classes dans l'espace d'origine.

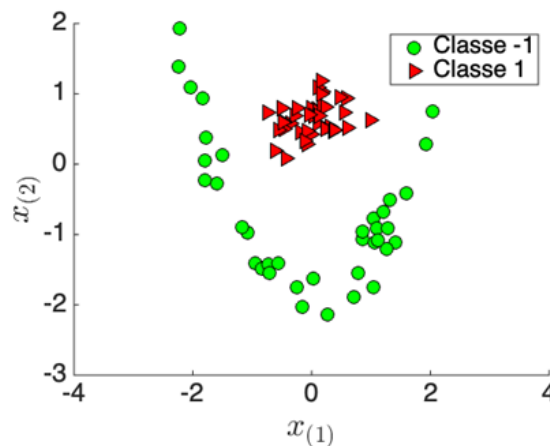


FIGURE 2.3 – Exemple de deux classes non linéairement séparables.

Le principe fondamental de cette méthode repose sur l'application d'une fonction de transformation, appelée fonction noyau (kernel). Cette fonction permet de calculer les produits scalaires dans l'espace transformé sans avoir à effectuer explicitement la transformation elle-même. Cette technique, connue sous le nom de « trick du noyau » (kernel trick), permet au SVM d'être entraîné dans un espace de dimension plus élevée tout en restant efficace en termes de calcul, sans qu'il soit nécessaire de connaître la transformation exacte appliquée aux données.

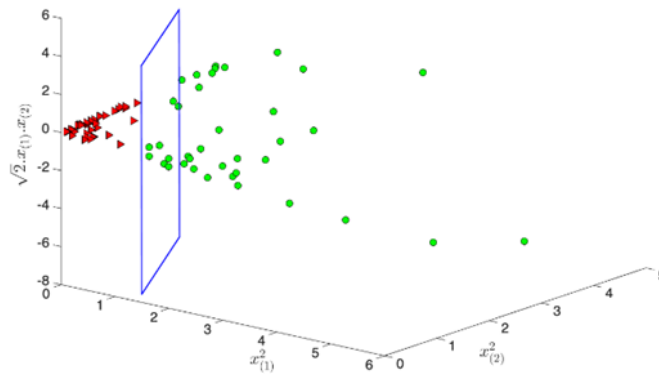


FIGURE 2.4 – Projection non linéaire des points dans un espace 3D. En bleu, on visualise une fonction de décision linéaire qui permet de séparer les classes une fois les données projetées dans un espace de plus grande dimension.

L'intégration de fonctions noyau dans les SVM permet de modéliser des frontières de décision complexes et non linéaires au sein des données. Parmi les noyaux les plus fréquemment utilisés, on retrouve le noyau gaussien (aussi appelé RBF – Radial Basis Function), le noyau polynomial et le noyau sigmoïde. Chaque noyau possède des caractéristiques propres, et le choix de l'un ou l'autre dépend principalement de la nature des données et du type de problème de classification à résoudre.

La classification non linéaire avec les SVM reprend les étapes fondamentales de la clas-

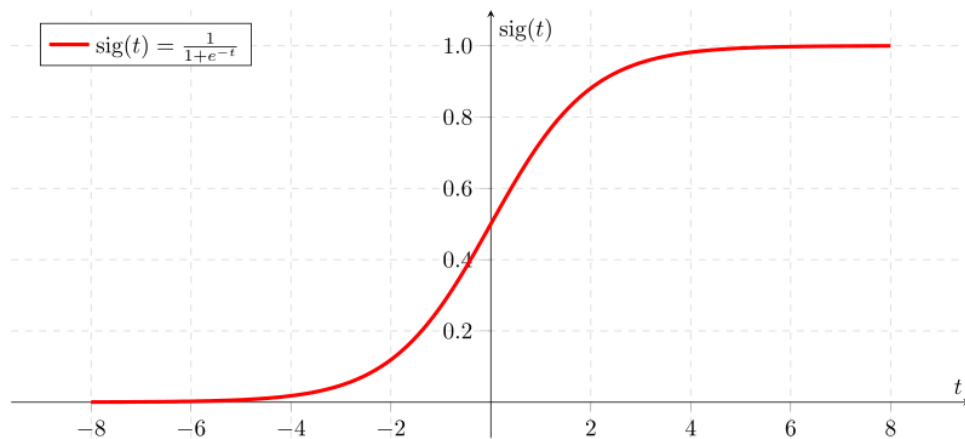


FIGURE 2.5 – La fonction sigmoïde

sification linéaire, mais se distingue par l'emploi d'une fonction noyau spécifique lors de la définition de l'hyperplan de décision. Le processus habituel inclut la récupération des données d'entraînement, la sélection d'un noyau pertinent, l'ajustement des paramètres du modèle via un algorithme d'optimisation, puis l'évaluation et la prédiction sur de nouvelles données.

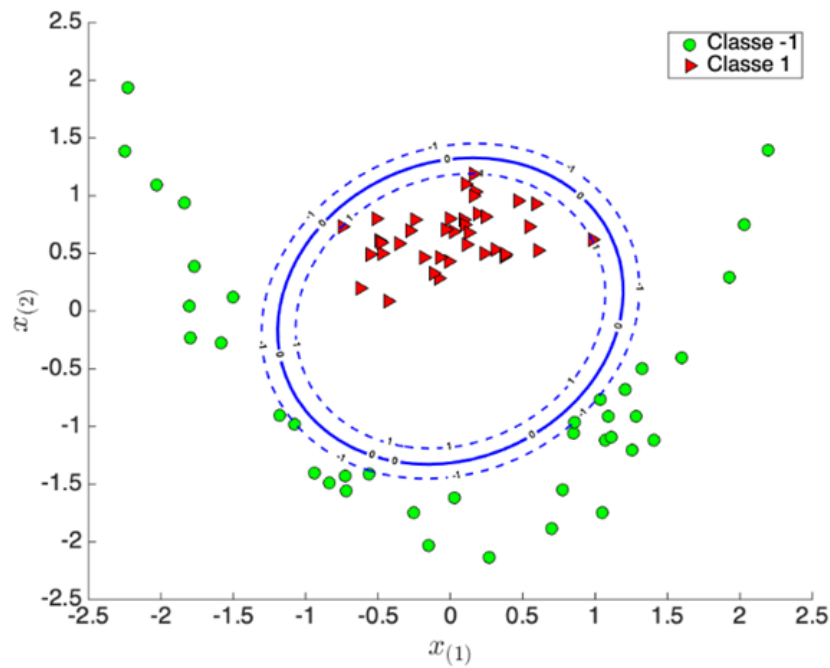
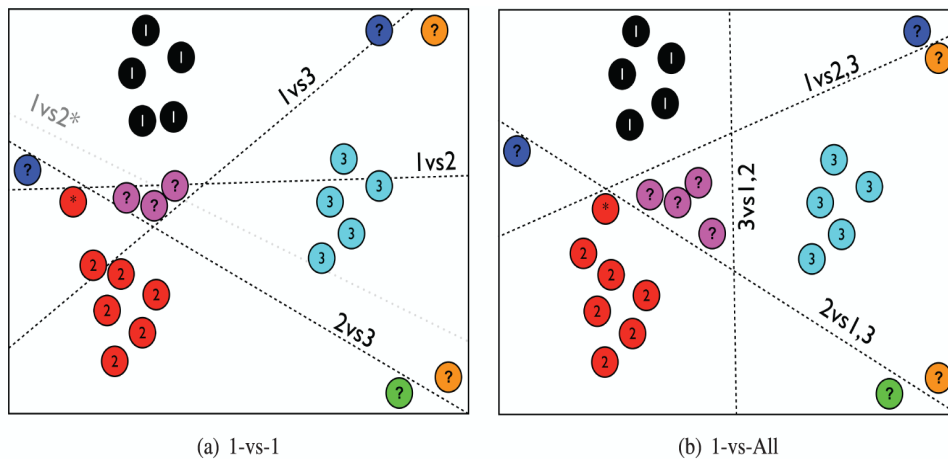


FIGURE 2.6 – La fonction de décision non linéaire

- **Troisième cas : classification SVM multi classes**

Le SVM peut également être appliqué à des problèmes de classification comportant plusieurs classes. Pour cela, plusieurs stratégies d'extension du SVM linéaire existent, notamment les approches One-vs-All (OvA) et One-vs-One (OvO).

Dans l'approche One-vs-All, un classifieur SVM est entraîné pour chaque classe, en considérant cette dernière comme la classe positive, tandis que toutes les autres sont regroupées comme classes négatives. Lors de la prédiction, l'échantillon est classé dans la catégorie correspondant au classifieur ayant fourni la plus haute valeur de décision.



(a) 1-vs-1

(b) 1-vs-All

FIGURE 2.7 – Méthodes OVA et OVO

Quant à la méthode One-vs-One, elle consiste à entraîner un classifieur SVM pour chaque paire possible de classes. Lors de la classification, chaque classifieur vote pour une classe, et l'exemple est attribué à la classe ayant obtenu le plus de votes parmi l'ensemble des comparaisons.

2.2.2 Les réseaux de neurones convolutifs

La classification d'images à l'aide des réseaux de neurones convolutifs (CNN – Convolutional Neural Networks) représente aujourd'hui l'une des approches les plus performantes en traitement d'images. Grâce à leur capacité à extraire automatiquement des caractéristiques visuelles hiérarchiques, les CNN sont particulièrement bien adaptés à l'analyse d'images complexes, comme celles issues de la télédétection ou des satellites d'observation de la Terre.

Les couches convolutionnelles permettent d'identifier des motifs tels que les textures, les contours, ou les formes, qui sont ensuite interprétés par des couches plus profondes pour effectuer la classification finale. Cette architecture hiérarchique rend les CNN extrêmement efficaces pour distinguer différentes classes d'objets ou de régions dans une image.

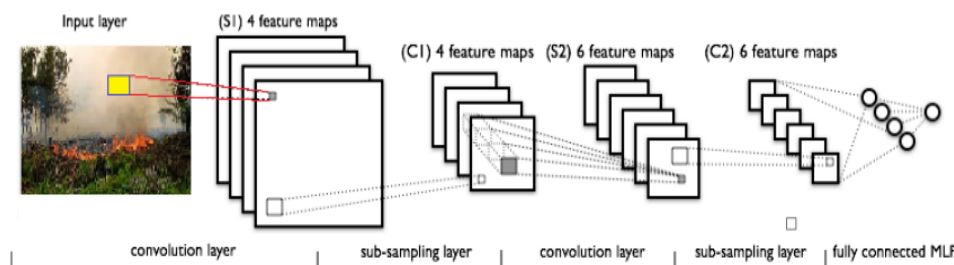


FIGURE 2.8 – CNN layers

Architecture de réseaux de neurones convolutionnels :

Couches convolutionnelles : La reconnaissance d'objets dans une image consiste à comparer l'image pixel par pixel afin d'en extraire des caractéristiques visuelles pertinentes. Si les réseaux de neurones multicouches (MLP) sont efficaces pour certaines tâches de traitement d'images, ils deviennent rapidement inadaptés face à des images de grande taille. Cela s'explique par la croissance exponentielle du nombre de connexions nécessaires avec l'augmentation de la dimension des images.

Par exemple, une image de taille

$200 \times 200 \times 3$ (200 pixels de largeur, 200 pixels de hauteur, et 3 canaux de couleur) impliquerait 120 000 entrées pour un seul neurone. Multiplier ce chiffre par le nombre total de neurones entraîne une charge computationnelle très élevée, difficile à gérer.

Pour pallier ce problème, les réseaux de neurones convolutionnels (CNN) introduisent une structure plus adaptée : la couche de convolution. Cette dernière permet de reconnaître des motifs ou formes locales sur une image en appliquant des filtres (ou noyaux) de petite taille, généralement 2×2 ou 3×3 pixels. Concrètement, pour chaque position du filtre sur la matrice d'entrée, la valeur correspondante dans la matrice de sortie est calculée comme la somme pondérée des pixels de l'image, chaque pixel étant multiplié par le poids correspondant du filtre. Ce mécanisme réduit fortement le nombre de connexions et permet d'extraire efficacement des caractéristiques locales, tout en préservant les structures spatiales. Notons que la taille de la matrice de sortie est généralement inférieure à celle de l'entrée (souvent réduite de deux lignes et deux colonnes lorsque aucun remplissage n'est appliqué) [9].

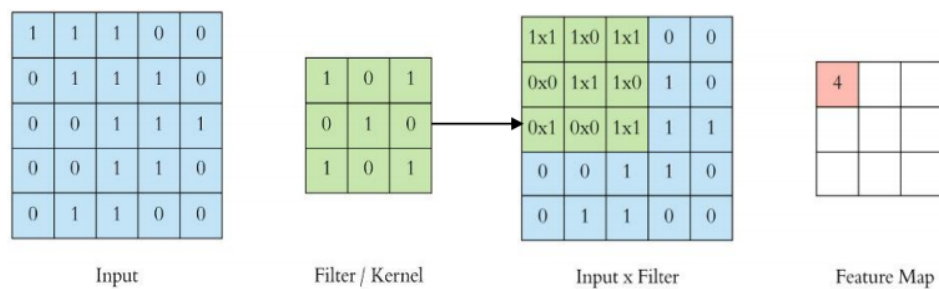


FIGURE 2.9 – Opération d'une convolution sur image de 5×5 pixel.

Couches de pooling : Après chaque couche convolutive, une couche de pooling (ou mise en commun) est souvent appliquée pour réduire la dimension spatiale des données. Cette opération de sous-échantillonnage permet de :

- Diminuer la complexité computationnelle.
- Contrôler le surapprentissage (overfitting).
- Introduire une invariance translationnelle.

Méthodes de pooling courantes :

Max pooling : Sélectionne la valeur maximale dans chaque fenêtre (ex : fenêtre 2×2).
Avantage : Préserve les features les plus saillantes.

Average pooling : Calcule la moyenne des valeurs. Avantage : Réduit le bruit.

Pooling apprenable : Utilise une combinaison linéaire pondérée des neurones (ex : learnable pooling).

Exemple illustratif : La Fig. 3 montre un max pooling avec une fenêtre de taille 2×2 et un stride de 2. Chaque bloc de 4 pixels est remplacé par sa valeur maximale, divisant ainsi la résolution spatiale par deux [10].

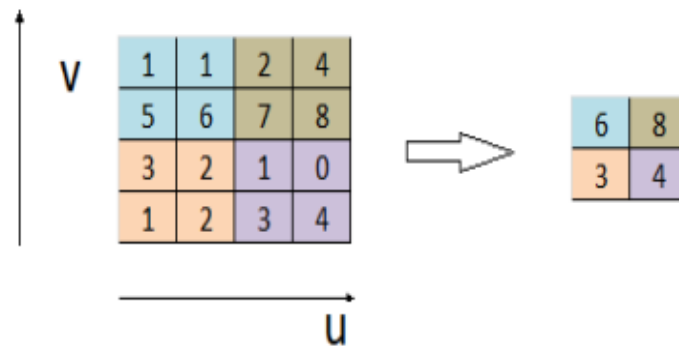


FIGURE 2.10 – Max pooling

L'entraînement d'un réseau de neurone convolutionnelle :

L'entraînement d'un réseau de neurones convolutif (CNN) consiste à déterminer et ajuster de manière empirique la valeur de ses poids. Le principe est le suivant : le CNN traite une image issue de la base de données d'entraînement et effectue une prédiction, c'est-à-dire qu'il indique à quelle classe il pense que cette image appartient.

Puisque la classe correcte de chaque image d'entraînement est connue à l'avance, il est possible de vérifier si la prédiction est correcte. Selon l'exactitude de la réponse, tous les poids du réseau sont mis à jour à l'aide d'un algorithme appelé rétropropagation du gradient de l'erreur.

Ce processus (prédiction, vérification et mise à jour des poids) est répété plusieurs fois sur l'ensemble des images de la base de données d'entraînement, afin que le modèle apprenne à mieux classer ces données.

Une fois l'entraînement terminé, on évalue les performances du modèle en lui présentant une base de données de validation. Cette base contient des images que le modèle n'a jamais vues auparavant. On mesure alors sa capacité à bien classer ces images en calculant son taux de précision, qui correspond à son pourcentage de bonnes classifications.

Analyse des performances du modèle après l'apprentissage :

À la fin du processus d'apprentissage, trois cas de figure peuvent se présenter :

- **Modèle performant sur les données d'entraînement et de validation :**
Le modèle a bien appris. Il parvient à reconnaître correctement aussi bien les images déjà vues que celles qu'il découvre pour la première fois.
- **Surapprentissage (Overfitting) :**
Le modèle est très performant sur les données d'entraînement mais obtient de moins bons résultats sur les données de validation. Cela indique une mauvaise capacité de généralisation. Pour y remédier, il est souvent nécessaire d'ajouter davantage d'images d'entraînement.
- **Sous-apprentissage (Underfitting) :**
Le modèle est peu performant aussi bien sur les données d'entraînement que sur celles de validation. Cela signifie qu'il n'a pas réussi à apprendre correctement. Dans ce cas, ajouter plus d'images ne suffit pas ; il faut envisager d'utiliser un modèle plus complexe ou mieux adapté au problème[11].

Indicateurs de performance d'un classifieur :

Matrice de confusion : Prenons l'exemple d'un classifieur binaire, c'est-à-dire un modèle qui prédit deux classes : **classe 0** et **classe 1**.

Pour évaluer les performances de ce classifieur, on distingue quatre types d'éléments classifiés :

- **Vrai positif (VP) :** élément de la classe 1 correctement prédit comme étant de la classe 1.
- **Vrai négatif (VN) :** élément de la classe 0 correctement prédit comme étant de la classe 0.
- **Faux positif (FP) :** élément de la classe 0 incorrectement prédit comme étant de la classe 1.
- **Faux négatif (FN) :** élément de la classe 1 incorrectement prédit comme étant de la classe 0.

Ces informations peuvent être rassemblées et représentées sous la forme d'une **matrice de confusion**.

Dans le cas d'un classifieur binaire, la matrice de confusion est :

En particulier, si la matrice de confusion est diagonale, le classifieur est parfait.

Classe réelle / Prédite	Classe 0	Classe 1
Classe 0	Vrai Négatif (VN)	Faux Positif (FP)
Classe 1	Faux Négatif (FN)	Vrai Positif (VP)

TABLE 2.1 – Matrice de confusion pour un classifieur binaire

Le graphe de précision : Un autre indicateur essentiel pour évaluer les performances d'un CNN est le **graphe de précision**.

Ce graphe représente l'évolution de la **précision du modèle** en fonction du **nombre d'itérations** réalisées lors de l'apprentissage et du test.

Il comporte généralement deux courbes :

- une courbe pour les données d'entraînement,
- une courbe pour les données de test.

Si les deux courbes atteignent une **précision de 100 %**, cela signifie que le classifieur est **parfait**, capable de reconnaître parfaitement aussi bien les données connues que les nouvelles données.

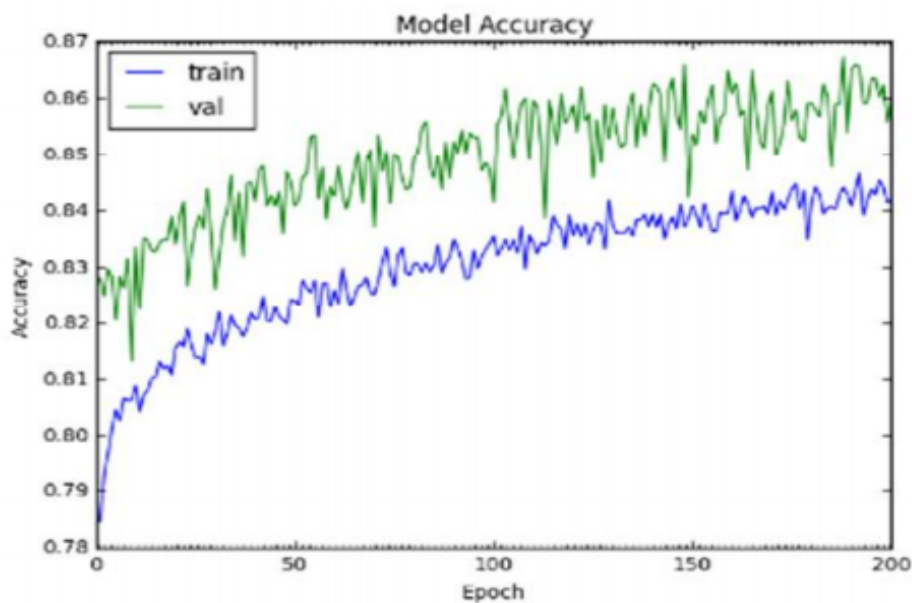


FIGURE 2.11 – Le graphe de précision d'un réseau de neurone convolutifs

Le graphe d'erreur : Le **graphe d'erreur** est une représentation de l'évolution du **taux d'erreur** du modèle en fonction du **nombre d'itérations** réalisées lors de l'apprentissage et du test.

Ce graphe comporte généralement deux courbes :

- une courbe pour les erreurs sur les données d'entraînement,

- une courbe pour les erreurs sur les données de test.

Lorsque l'erreur diminue sur les deux courbes, cela indique que le modèle apprend correctement. Si les erreurs tendent vers zéro pour les deux ensembles de données, on peut considérer que le classifieur est **parfait**.



FIGURE 2.12 – Le graphe d'erreur d'un réseau de neurone convolutifs

Indicateurs de bas : À partir de la matrice de confusion, plusieurs indicateurs permettent d'évaluer la performance d'un modèle de classification. Par exemple, pour évaluer la qualité de la prédiction sur la **classe 1**, on définit :

- **Précision (Precision)** : proportion d'éléments correctement prédits parmi tous ceux prédits comme appartenant à la classe 1.

La précision est donnée par la formule suivante :

$$\text{Précision} = \frac{VP}{VP + FP} \quad (2.3)$$

- **Rappel (Recall)** : proportion d'éléments de la classe 1 correctement prédits parmi tous les éléments réellement de la classe 1.

Le rappel est donné par la formule suivante :

$$\text{Rappel} = \frac{VP}{VP + FN} \quad (2.4)$$

- **F-mesure (F1-score)** : mesure de compromis entre la précision et le rappel.

La F-mesure est exprimée par la formule suivante :

$$\text{F-mesure} = \frac{2 \times \text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}} \quad (2.5)$$

Ces indicateurs peuvent être calculés pour chaque classe. La moyenne de ces indicateurs sur toutes les classes fournit une évaluation globale de la qualité du classifieur.

2.2.3 La méthode de classification MLP (Multi Layer Perceptron)

La classification par MLP (Multi-Layer Perceptron) est une méthode d'apprentissage automatique utilisée pour la classification de données. Le MLP s'inscrit dans le domaine de l'intelligence artificielle, et plus précisément dans le champ du machine learning, comme illustré dans la figure 2.8

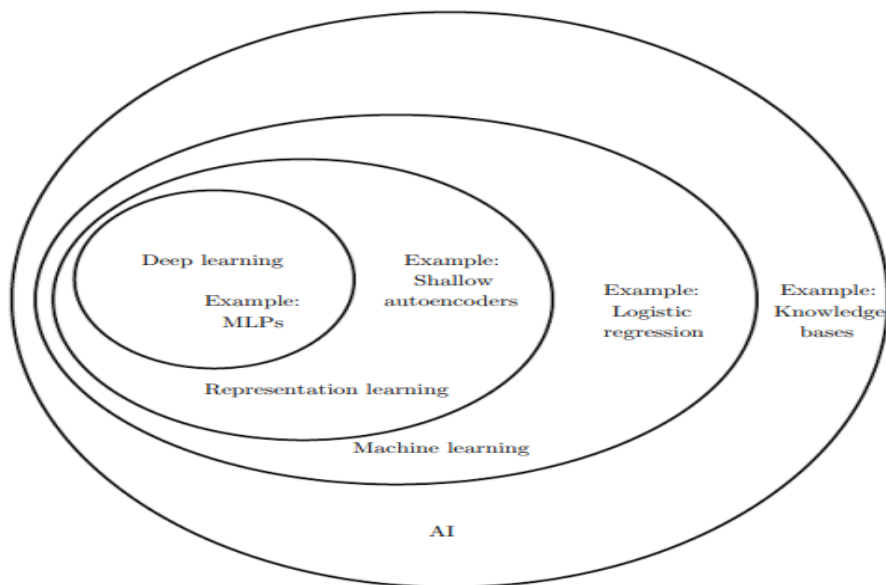


FIGURE 2.13 – Relation entre le MLP et le IA

Le MLP (Multi-Layer Perceptron) est un type de réseau de neurones artificiels, également connu sous le nom de réseau neuronal feedforward (Deep Feedforward Network ou Feedforward Neural Network). Il se compose de plusieurs couches de neurones interconnectées, organisées de manière séquentielle.

Le MLP fait partie du domaine du machine learning, qui vise à apprendre automatiquement à partir de données pour réaliser des prédictions ou des classifications. Il repose sur des algorithmes d'apprentissage supervisé permettant d'ajuster les poids des connexions

entre neurones, dans le but d'identifier les modèles et les relations sous-jacentes présentes dans les données d'entrée.

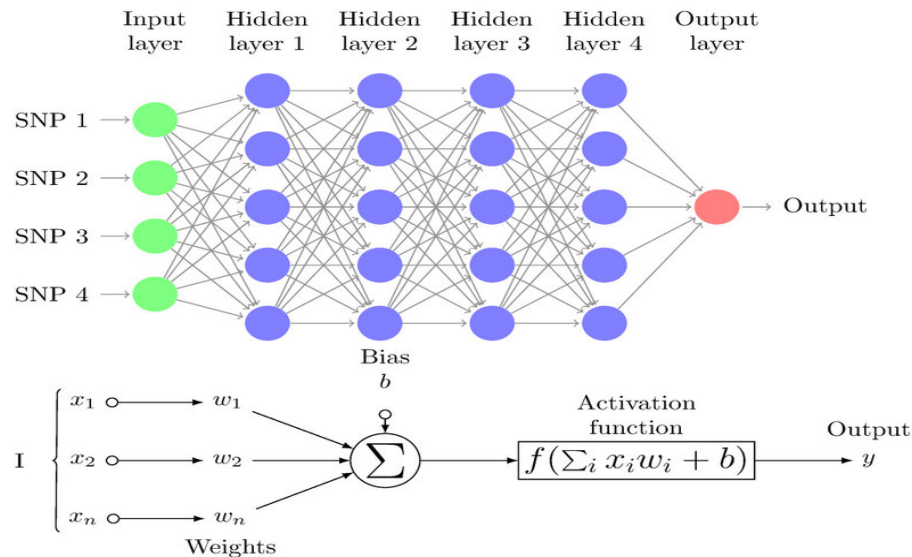


FIGURE 2.14 – Diagramme du Perceptron Multi-Couches (MLP) avec quatre couches cachées.

La mise en œuvre d'un classifieur basé sur un réseau de neurones MLP (Multi-Layer Perceptron) suit plusieurs étapes essentielles, permettant d'assurer un apprentissage efficace et une bonne capacité de généralisation.

Ces étapes principales sont :

1. Préparation des données :

Avant l'entraînement du modèle, les données doivent être prétraitées. Cela inclut généralement :

- La séparation en ensembles d'apprentissage et de test.
- La normalisation des caractéristiques.

Et d'autres techniques de traitement préalable adaptées au type de données concerné.

2. Définition de l'architecture du réseau :

L'architecture du réseau MLP est déterminée en spécifiant :

- Le nombre de couches cachées.
- Et le nombre de neurones par couche.

Chaque couche cachée permet d'extraire des représentations progressives et abstraites des données d'entrée.

Voici quelques architectures courantes de MLP :

- MLP à une seule couche cachée : Composé d'une couche d'entrée, d'une couche cachée et d'une couche de sortie. C'est l'architecture la plus simple, adaptée aux tâches de classification de base.
- MLP à plusieurs couches cachées : Intègre plusieurs couches intermédiaires entre l'entrée et la sortie. Cela permet d'apprendre des représentations hiérarchiques complexes, bien adaptées aux problèmes nécessitant une modélisation plus approfondie.
- MLP avec connexions résiduelles : Ce type d'architecture ajoute des connexions directes entre certaines couches, permettant à l'information de contourner une ou plusieurs couches. Cette stratégie facilite l'apprentissage profond et améliore la stabilité du modèle.
- MLP avec régularisation : Pour prévenir le surapprentissage (overfitting), diverses techniques de régularisation peuvent être employées, telles que : La régularisation L1 ou L2, le dropout, ou d'autres méthodes visant à limiter la complexité du modèle et améliorer sa capacité de généralisation.

Le choix de l'architecture dépend fortement de la complexité des données, des ressources de calcul disponibles, et des performances souhaitées.

3. Initialisation des poids :

Les poids synaptiques reliant les neurones sont initialisés de manière aléatoire ou suivant une distribution spécifique. Ces poids seront ajustés au cours de l'apprentissage pour minimiser l'erreur entre les sorties prédictives du réseau et les sorties attendues.

4. Propagation avant (Forward Propagation) :

Lors de cette étape, les données sont propagées à travers le réseau, couche par couche :

- À chaque neurone, une opération linéaire est effectuée (pondération de l'entrée).

- Puis, une fonction d'activation est appliquée pour introduire une non-linéarité.

La propagation commence à la couche d'entrée et se poursuit jusqu'à la couche de sortie, en passant par toutes les couches cachées.

Remarque sur la fonction d'activation :

Elle est essentielle pour permettre au réseau de modéliser des relations non linéaires entre les entrées et les sorties. Sans elle, le réseau se comporterait comme un simple modèle linéaire, limitant sa capacité à résoudre des problèmes complexes.

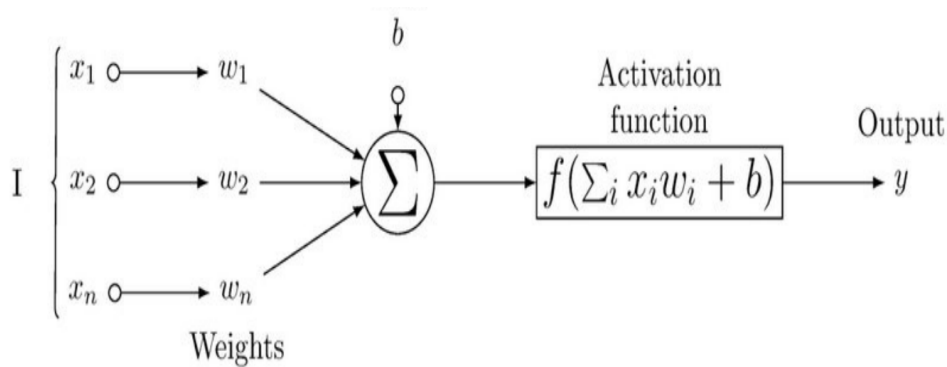


FIGURE 2.15 – Forward propagation

Parmi les différentes fonctions d'activation utilisées dans les réseaux de neurones, on peut citer :

- **la fonction sigmoïde** : Elle est définie par l'expression suivante :

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.6)$$

Cette fonction produit une sortie comprise entre 0 et 1, ce qui la rend particulièrement adaptée aux problèmes de classification binaire ou à la modélisation de probabilités. Grâce à sa forme en "S", elle permet de transformer des valeurs d'entrée continues en probabilités interprétables.

- **la fonction tangente hyperbolique (tanh)** : elle est définie par la fonction suivantes :

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.7)$$

Cette fonction produit des valeurs de sortie comprises dans l'intervalle (-1,1), ce qui la rend particulièrement utile pour gérer des données contenant à la fois des valeurs

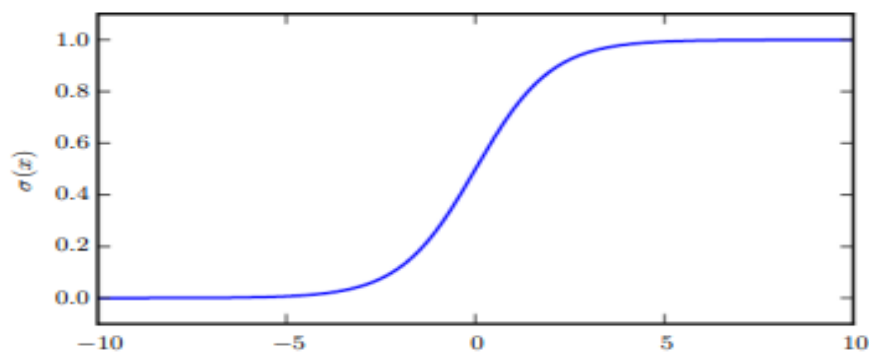


FIGURE 2.16 – Fonction sigmoïde

positives et négatives, tout en maintenant une sortie centrée autour de zéro.

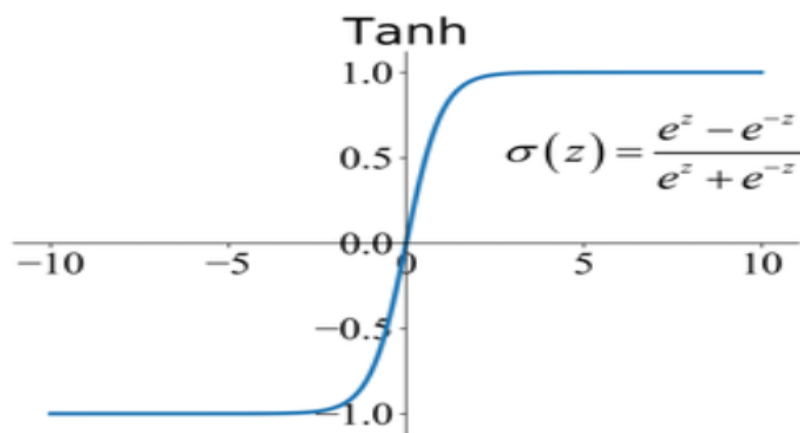


FIGURE 2.17 – La fonction tanh

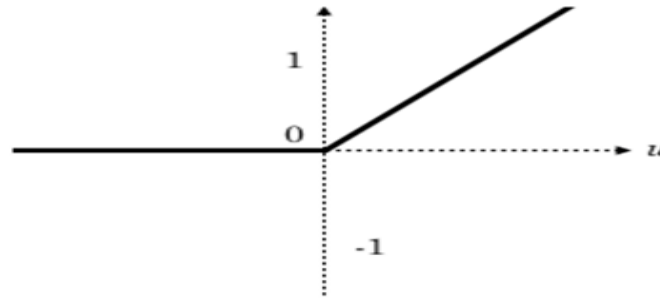
- **Fonction ReLU (Rectified Linear Unit)** : La fonction ReLU est l'une des fonctions d'activation les plus utilisées dans les réseaux de neurones, en particulier dans les réseaux profonds (deep learning), elle est définie comme

$$f(x) = \max(0, x) \quad (2.8)$$

Le choix de la fonction d'activation dépend fortement de la nature du problème ainsi que des caractéristiques des données. Chaque fonction d'activation peut influencer différemment le processus d'apprentissage du modèle, et certaines sont mieux adaptées que d'autres à des types spécifiques de tâches ou de données.

5. Calcul de l'erreur :

L'écart entre les prédictions du réseau et les valeurs réelles est évalué à l'aide d'une fonction de perte, telle que la moyenne des erreurs quadratiques (Mean Squared Error, MSE). Cette étape permet de mesurer la performance du modèle et de guider

FIGURE 2.18 – La fonction *RELU*

l'optimisation lors de l'apprentissage.

Note :

La fonction d'erreur (ou *loss function*) est une expression mathématique qui permet de mesurer l'écart entre les prédictions du modèle et les valeurs réelles issues des données d'apprentissage. Elle joue un rôle fondamental dans le processus d'apprentissage automatique, en particulier dans la classification à l'aide des réseaux MLP (Multi-Layer Perceptron), car elle guide l'optimisation des paramètres du modèle.

Il existe plusieurs fonctions d'erreur couramment utilisées, chacune étant choisie en fonction du type de problème (régression ou classification) et des caractéristiques des données.

Voici quelques exemples de fonctions d'erreur fréquemment utilisées :

La **fonction d'erreur** (ou *loss function*) est une fonction mathématique qui permet de mesurer l'écart entre les prédictions générées par un modèle et les valeurs réelles issues des données d'apprentissage. Elle joue un rôle central dans le processus d'apprentissage automatique, en particulier dans la classification à l'aide des réseaux de neurones MLP (*Multi-Layer Perceptron*), puisqu'elle guide l'ajustement des poids du réseau en minimisant l'erreur.

Le choix de la fonction d'erreur dépend du type de tâche (classification ou régression) et de la nature des données. Voici quelques fonctions d'erreur couramment utilisées :

Nom	Formule	Utilisation
Erreur quadratique moyenne (MSE)	$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$	Régression
Erreur absolue moyenne (MAE)	$MAE = \frac{1}{n} \sum_{i=1}^n y_i - \hat{y}_i $	Régression
Entropie croisée (Cross-Entropy)	$L = - \sum_{i=1}^n y_i \log(\hat{y}_i)$	Classification binaire ou multi-classes
Hinge Loss	$L = \sum_{i=1}^n \max(0, 1 - y_i \hat{y}_i)$	Support Vector Machines (SVM)
Divergence de Kullback-Leibler (KL)	$D_{KL}(P \parallel Q) = \sum P(x) \log \frac{P(x)}{Q(x)}$	Comparaison de distributions (classification)

TABLE 2.2 – Fonctions d’erreur courantes en apprentissage automatique

— 6. Rétropropagation (Backpropagation)

La **rétropropagation** est une étape essentielle dans l’apprentissage des réseaux de neurones. Elle consiste à *propager l’erreur en sens inverse*, depuis la couche de sortie jusqu’à la couche d’entrée, dans le but d’ajuster les poids du réseau et d’améliorer ses performances.

Ce processus repose sur le *calcul du gradient de la fonction de perte* par rapport aux poids, en appliquant la **règle de la chaîne** (*chain rule*). Chaque poids est mis à jour de manière à **minimiser l’erreur de prédiction**.

— Étapes de la rétropropagation :

- **Calcul de l’erreur** : On évalue l’écart entre la sortie prédite et la sortie réelle à l’aide d’une fonction de perte (par exemple MSE ou entropie croisée).
- **Calcul des gradients** : On détermine l’influence de chaque poids sur l’erreur, en calculant la dérivée partielle de la fonction de perte par rapport aux poids.
- **Propagation des gradients** : Les gradients sont transmis couche par couche vers l’arrière à l’aide de la règle de dérivation, afin de mettre à jour les couches précédentes.
- **Mise à jour des poids** : Chaque poids est ajusté selon l’algorithme de descente de gradient :

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} - \eta \cdot \frac{\partial \mathcal{L}}{\partial w_{ij}} \quad (2.9)$$

où :

- w_{ij} : poids entre le neurone j et i ,

- η : taux d'apprentissage (*learning rate*),
- \mathcal{L} : fonction de perte.

L'objectif de la rétropropagation est de **minimiser la fonction de perte** en ajustant les paramètres internes du réseau de manière optimale, améliorant ainsi la capacité de généralisation du modèle.

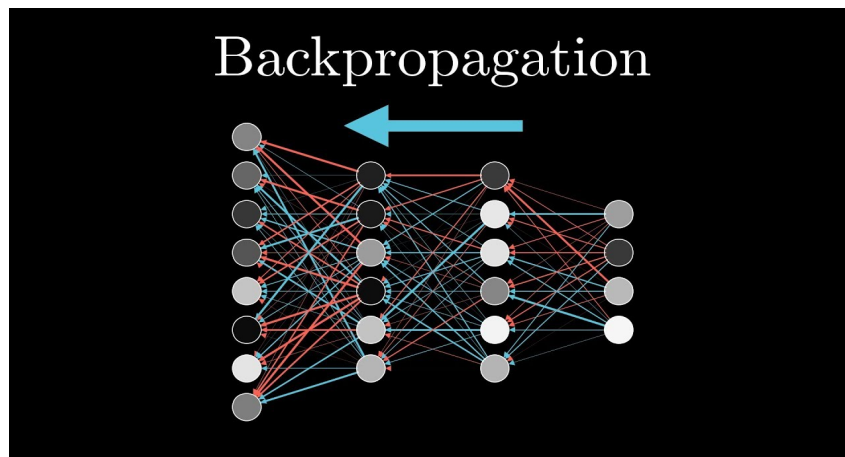


FIGURE 2.19 – Rétropropagation (*backpropagation*)

Note : Plusieurs algorithmes d'optimisation sont utilisés pour ajuster les poids dans les réseaux de neurones, notamment les MLP (Multi-Layer Perceptrons). Voici quelques-uns des plus courants :

- **Descente de gradient (Gradient Descent)** : C'est l'algorithme de base utilisé dans la rétropropagation. Il consiste à ajuster les poids dans la direction opposée au gradient de la fonction de perte. Plusieurs variantes en sont issues :

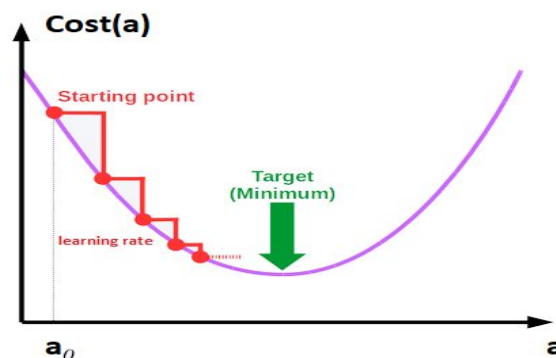


FIGURE 2.20 – Descente de gradient

- **SGD (Stochastic Gradient Descent)** : les poids sont mis à jour après chaque exemple.

- **Mini-batch Gradient Descent** : les poids sont mis à jour par petits lots d'exemples, ce qui permet un compromis entre rapidité et stabilité.
- **Gradient Descent with Momentum** : ajoute une composante d'inertie pour accélérer l'apprentissage et éviter les oscillations.
- **RMSprop (Root Mean Square Propagation)** : Cet algorithme ajuste le taux d'apprentissage de chaque poids en utilisant une moyenne mobile pondérée des carrés des gradients. Il est particulièrement adapté aux problèmes non stationnaires.

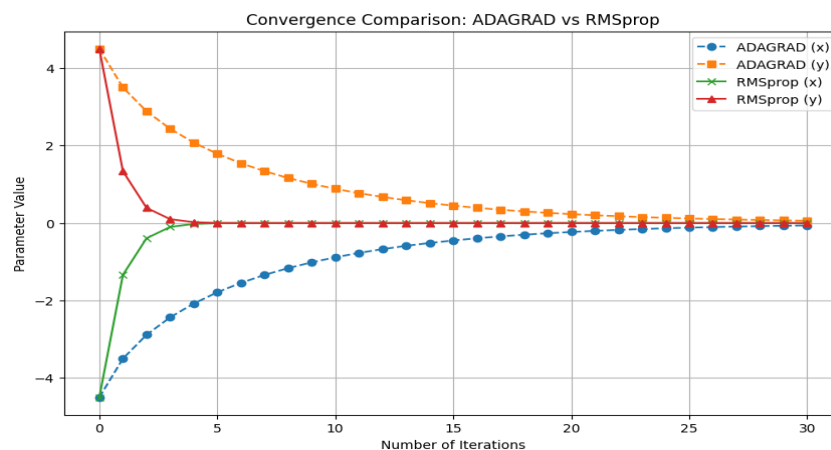


FIGURE 2.21 – Root Mean Square Propagation

Chaque méthode présente des avantages et des inconvénients, et le choix de l'algorithme dépend généralement du problème à traiter et des performances souhaitées.

— 7. Répétition des étapes 4 à 7 :

Les étapes de *propagation avant*, *calcul de l'erreur*, *rétropropagation* et *mise à jour des poids* sont répétées de manière itérative. Ce cycle d'apprentissage se poursuit soit pendant un **nombre fixe d'itérations** (appelées époques), soit jusqu'à l'atteinte d'un critère de **convergence**, lorsque la fonction de perte se stabilise ou que les performances de validation ne s'améliorent plus.

— 8. Évaluation et prédiction :

Une fois l'entraînement terminé, le réseau MLP peut être utilisé pour effectuer des **prédictions sur de nouvelles données**. Lors de cette étape, seule la propagation avant est utilisée, sans mise à jour des poids.

L'efficacité du modèle est ensuite évaluée à l'aide de plusieurs **indicateurs de performance**, parmi lesquels :

- la précision (*accuracy*),
- le rappel (*recall*),
- la matrice de confusion,

— la courbe ROC et l'AUC.

Ces mesures permettent d'apprécier la capacité du modèle à **généraliser** sur des données inconnues.

2.2.4 La méthode de classification random forest

La méthode de propagation Random Forest (ou forêt aléatoire) est un algorithme d'apprentissage automatique supervisé, largement utilisé pour des tâches de classification et de régression. Développé par Leo Breiman en 2001, ce modèle s'appuie sur le principe d'ensembling, combinant les prédictions de plusieurs arbres de décision pour améliorer la précision et réduire le surajustement (overfitting).

Le Random Forest se distingue par sa robustesse et sa capacité à gérer des jeux de données complexes, bruyants ou avec de nombreuses caractéristiques. Son fonctionnement repose sur deux mécanismes clés :

- **Le bagging** : Chaque arbre est entraîné sur un sous-ensemble aléatoire des données (avec rééchantillonnage, ou bootstrap).
- **La sélection aléatoire des caractéristiques** : À chaque nœud, seule une partie des variables est considérée pour la division, favorisant la diversité des arbres.

Grâce à cette approche, le Random Forest limite la variance du modèle tout en maintenant un biais faible, offrant des performances compétitives même sans réglage hyperparamétrique poussé. De plus, il fournit des indicateurs d'importance des variables, facilitant l'interprétation des résultats.

Dans des domaines comme la bio-informatique, la finance ou la télédétection, cette méthode est appréciée pour sa fiabilité et sa scalabilité. Son application à des problèmes de propagation (par exemple, la prédiction de la diffusion d'une maladie ou d'une information) en fait un outil polyvalent et puissant [12].

La télédétection consiste à acquérir des informations sur la surface terrestre à l'aide de capteurs embarqués sur des satellites, drones ou avions. Ces capteurs collectent des images multispectrales, hyperspectrales ou radar.

Pour interpréter ces images (ex. : cartographier l'occupation du sol, détecter des changements, identifier des cultures), on a besoin d'algorithmes de classification puissants. C'est là que Random Forest intervient. A cet effet, il est important de d'écrire le processus de formation d'un arbre de d'ecision (decision tree).

Arbre de décision (decision tree) :

Un arbre de décision est une méthode utilisée pour la classification ou la régression. Il se présente sous forme d'une structure arborescente dans laquelle chaque nœud représente une décision à prendre à partir d'un attribut.

L'arbre commence par un **nœud racine** qui contient l'ensemble des données d'entraînement. Ce nœud se divise en deux ou plusieurs **sous-nœuds** à l'aide d'un processus appelé *splitting*. Si un sous-nœud se divise à nouveau, il devient un **nœud de décision**. S'il ne se divise plus, on l'appelle **nœud terminal** ou **feuille**. L'ensemble formé par des nœuds de décision et des nœuds terminaux constitue une **branche** ou un **sous-arbre**.

Pour classer un exemple, on le fait passer de la racine jusqu'à une feuille. À chaque étape, une décision est prise selon la valeur d'un attribut. Chaque nœud représente un test, et chaque branche correspond à une réponse possible. Le chemin suivi permet d'aboutir à une décision finale.

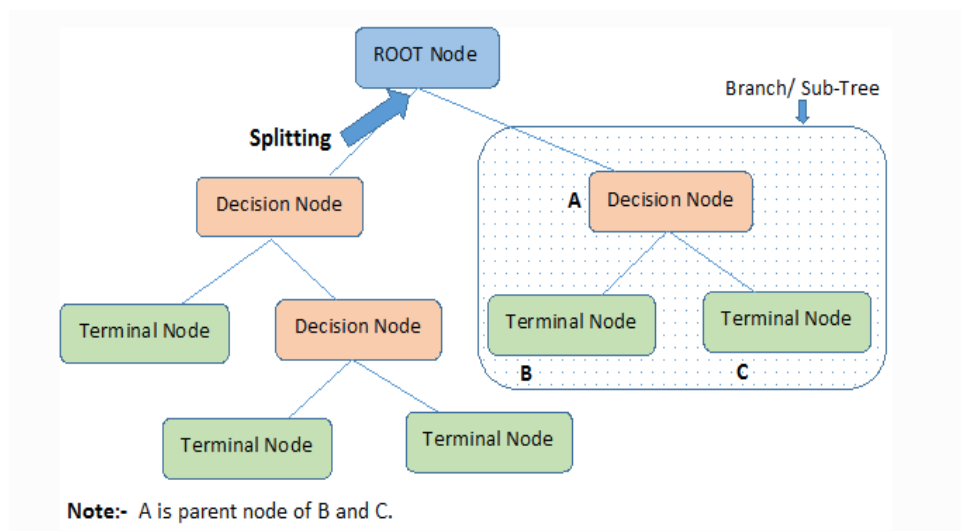


FIGURE 2.22 – Arbre de décision

L'objectif du *splitting* est de créer des sous-nœuds aussi homogènes que possible, c'est-à-dire contenant des exemples appartenant majoritairement à une seule classe. Cette homogénéité est mesurée par différents **critères de division**, parmi lesquels :

- **L'entropie** et le **gain d'information** (utilisés dans l'algorithme ID3),
- **Le gain de ratio** (utilisé dans l'algorithme C4.5),
- **L'indice de Gini** (utilisé dans l'algorithme CART),
- **L'erreur de classification**.

Voici quelques **algorithmes populaires** utilisés pour construire des arbres de décision :

- **ID3 (Iterative Dichotomiser 3)** : utilise le gain d'information pour choisir les attributs les plus pertinents à chaque division.
- **C4.5** : amélioration de ID3, capable de gérer les variables continues et les valeurs manquantes, utilisant le gain de ratio.
- **CART (Classification And Regression Tree)** : crée uniquement des arbres binaires et utilise l'indice de Gini pour mesurer la pureté.
- **MARS (Multivariate Adaptive Regression Splines)** : adapté aux problèmes de régression multivariée non linéaire[13].

Le **choix du nœud racine** est très important. Un mauvais choix peut réduire considérablement la performance du modèle. C'est pourquoi ces algorithmes utilisent des critères objectifs pour sélectionner l'attribut de départ :

• **L'entropie** : L'entropie est un concept fondamental en théorie de l'information, en statistique et en apprentissage automatique qui permet de quantifier le degré d'incertitude ou de désordre associé à une information. Initialement introduite par Claude Shannon en 1948 dans le cadre de la théorie de la communication, cette notion s'est révélée essentielle pour évaluer la quantité d'information contenue dans un message ou dans une distribution de probabilités. Plus l'entropie est élevée, plus l'incertitude est grande, et inversement[14].

Prenons l'exemple classique du lancer d'une pièce de monnaie. Dans le cas d'une pièce parfaitement équilibrée, où la probabilité d'obtenir "pile" est égale à celle d'obtenir "face" (soit 0,5 pour chaque), l'entropie est maximale. Cela signifie que le résultat est totalement imprévisible, et l'incertitude est à son comble. Mathématiquement, cela se traduit par une entropie de 1 bit, ce qui correspond à la quantité d'information nécessaire pour encoder le résultat d'un lancer.

En revanche, si la pièce est truquée et donne toujours "pile" (probabilité de 1 pour "pile" et 0 pour "face"), l'entropie est nulle. Il n'y a aucune incertitude, car le résultat est certain. Entre ces deux extrêmes, pour des probabilités intermédiaires, l'entropie varie continûment, atteignant son maximum à 0,5, comme le montre la courbe bien connue de l'entropie d'une variable binaire.

Pour une variable aléatoire discrète X pouvant prendre n valeurs distinctes avec des probabilités p_1, p_2, \dots, p_n , l'entropie $H(X)$ est définie par :

$$H(X) = - \sum_{i=1}^n p_i \log_2 p_i \quad (2.10)$$

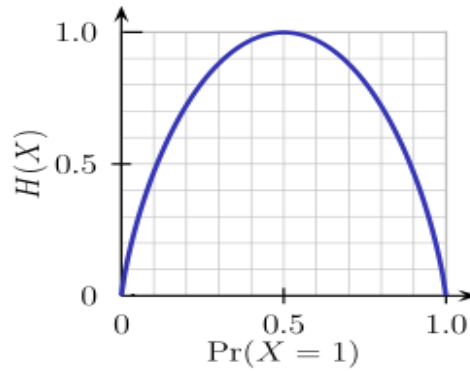


FIGURE 2.23 – L'entropie

Cette formule, bien que simple, a des implications profondes. Le logarithme en base 2 assure que l'entropie est mesurée en bits, ce qui correspond à l'interprétation en termes de quantité d'information. Notons que si l'une des probabilités p_i est nulle, le terme correspondant dans la somme est considéré comme nul, car $\lim_{p \rightarrow 0} p \log p = 0$.

L'entropie peut également être généralisée à des cas multivariés. Par exemple, pour un ensemble de données partitionné selon un attribut X , on peut définir l'entropie conditionnelle $H(T|X)$, qui mesure l'incertitude restante sur T après avoir observé X . Cette notion est cruciale dans des algorithmes comme les arbres de décision, où l'on cherche à réduire progressivement l'entropie des sous-ensembles créés.

$$H(T | X) = \sum_{i \in X} P(i) \cdot H(i) \quad (2.11)$$

• **Le gain d'information** Le Gain d'Information (IG) est une mesure clé en apprentissage automatique pour évaluer l'efficacité d'un attribut à segmenter un ensemble de données selon les classes cibles. Il quantifie la réduction d'incertitude (entropie) apportée par un attribut lors de la construction d'un arbre de décision.

$$\text{IG}(T, X) = H(T) - H(T | X) \quad (2.12)$$

Les arbres de décision sont des outils puissants en apprentissage supervisé. Toutefois, lorsqu'on les utilise sur un jeu de données comportant un grand nombre de colonnes (ou caractéristiques), ils présentent un risque élevé de **surapprentissage (overfitting)**.

En effet, un arbre de décision non contraint peut apprendre par cœur les données d'entraînement. Il peut aller jusqu'à créer une feuille distincte pour chaque observation, atteignant ainsi une précision de 100 % sur les données d'entraînement. Ce comportement nuit à sa capacité à généraliser sur de nouvelles données, ce qui entraîne une baisse significative

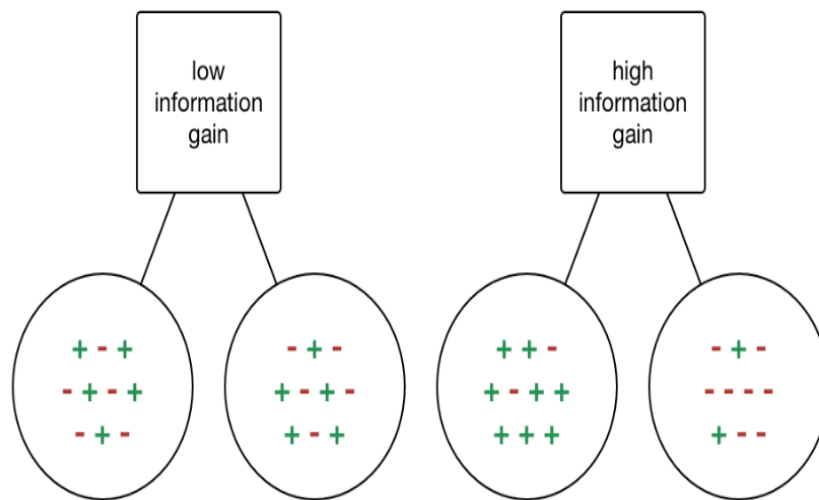


FIGURE 2.24 – Le gain d'information

de la performance sur l'ensemble de test ou en production.

Pour remédier à ce problème, on utilise l'algorithme **Random Forest**, qui repose sur une approche d'ensemble (*ensemble learning*) permettant de réduire la variance des modèles tout en conservant une bonne capacité de prédiction.

Le processus de construction d'une **Random Forest** se décompose en plusieurs étapes clés [15] :

1. Création d'un ensemble d'arbres de décision indépendants

Random Forest construit un grand nombre d'arbres de décision, chacun étant entraîné indépendamment des autres.

2. Échantillonnage aléatoire avec remplacement (méthode bootstrap)

Pour chaque arbre, un échantillon aléatoire des données d'entraînement est généré (avec remplacement). Cela signifie que chaque arbre est entraîné sur un sous-ensemble différent, ce qui introduit de la diversité entre les arbres.

3. Sélection aléatoire des caractéristiques à chaque division

Lors de la construction de chaque nœud de l'arbre, un sous-ensemble aléatoire de variables est sélectionné parmi toutes les caractéristiques. Cela réduit la corrélation entre les arbres et améliore la robustesse du modèle.

4. Construction des arbres de décision

Chaque arbre est ensuite construit en divisant récursivement les sous-ensembles de données en fonction des variables sélectionnées. Les critères de division peuvent être :

- l'indice de Gini (mesure d'impureté),

- ou l'entropie (basée sur la théorie de l'information).

L'objectif est de maximiser la pureté des nœuds à chaque étape.

5. Agrégation des prédictions finales

Une fois tous les arbres construits, les prédictions individuelles sont agrégées :

- Pour un problème de classification : on utilise un *vote majoritaire* entre les prédictions des arbres.
- Pour un problème de régression : on calcule la *moyenne* des prédictions numériques.

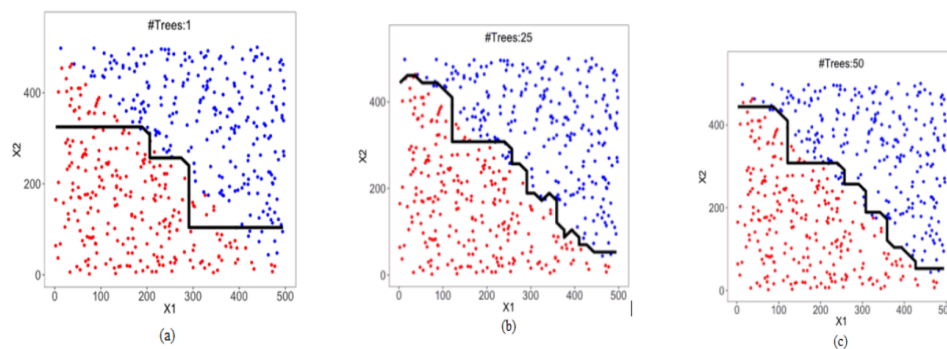


FIGURE 2.25 – (a) une seule arbre de décision (b) 25 arbres de décision (c) 50 arbres de décision

2.2.5 Comparaison entre les méthodes de classifications supervisées

La performance des méthodes de classification supervisée dépend de plusieurs facteurs cruciaux qui interagissent de manière complexe. Comprendre et maîtriser ces facteurs permet d'améliorer la précision des modèles de classification et de garantir leur efficacité dans des contextes réels. Les principaux éléments qui influencent cette performance sont : la qualité des données, la taille de l'échantillon d'apprentissage, le choix des caractéristiques pertinentes et les paramètres propres à chaque algorithme.

SVM (Support Vector Machine) :[16]

Bien que les SVM soient efficaces pour la classification, ils présentent plusieurs inconvénients. Leur complexité algorithmique élevée peut rendre leur entraînement coûteux, surtout sur de grands ensembles de données. Ils sont également sensibles au choix des hyperparamètres, et une mauvaise sélection peut nuire à la performance. De plus, les SVM ne gèrent pas bien les données déséquilibrées et sont souvent perçus comme des modèles

”boîte noire”, rendant leur interprétabilité limitée.

Avantages	Inconvénients
+ Excellente performance en haute dimension (p.ex. text mining)	– Complexité algorithmique élevée ($O(n^2)$ à $O(n^3)$)
+ Gère les frontières non-linéaires via le kernel trick	– Sensible au choix des hyperparamètres (C, γ)
+ Résistance au sur-apprentissage (bonne généralisation)	– Interprétabilité limitée (modèle ”boîte noire”)
+ Efficace sur petits/moyens jeux de données	– Performances médiocres sur données déséquilibrées
+ Théorie mathématique solide (maximisation de la marge)	– Nécessite un prétraitement des données (normalisation)
+ Peu sensible au ”fléau de la dimension”	– Difficile à adapter aux très grands datasets

TABLE 2.3 – Avantages et Inconvénients des SVM

Les réseaux de neurones convolutifs (CNN) :[17]

Bien que les réseaux de neurones soient très puissants, ils présentent certains inconvénients. Ils nécessitent des ressources computationnelles importantes, sont sensibles au surapprentissage et à la sélection des hyperparamètres, et manquent souvent d’interprétabilité. Le tableau suivant résume ces limitations.

Avantages	Inconvénients
Très performants pour l’analyse d’images, vidéos et données spatiales	Requiert une grande puissance de calcul (GPU recommandé)
Réduction du besoin de prétraitement grâce à l’extraction automatique de caractéristiques	Difficiles à interpréter (modèle ”boîte noire”)
Réduction du nombre de paramètres grâce aux filtres partagés	Nécessitent beaucoup de données pour un bon entraînement
Capables de détecter des motifs locaux (textures, bords, formes)	Sensibles aux changements de position, d’échelle ou de rotation sans ajustement
Efficaces dans les architectures modernes de deep learning	Temps d’entraînement long

TABLE 2.4 – Avantages et Inconvénients des réseaux de neurones convolutifs (CNN)

MLP (Multi Layer Perceptron)

Les MLP (Multi-Layer Perceptron) sont largement utilisés pour résoudre des problèmes de classification et de régression. Cependant, comme toute méthode, ils présentent certains avantages et inconvénients. Le tableau ci-dessous résume les principaux points à considérer lorsqu'on utilise cette méthode.

Avantages	Inconvénients
+ Capacité à modéliser des relations complexes entre les données	– Nécessite un grand nombre de données d'entraînement pour de bonnes performances
+ Utilisé pour des problèmes de classification et régression	– Entraînement long et coûteux en ressources computationnelles
+ Adapté à une grande variété de données (images, textes, etc.)	– Sensible au surapprentissage (overfitting) sans régularisation appropriée
+ Bonnes performances pour des problèmes non linéaires	– Manque d'interprétabilité, modèle "boîte noire"

TABLE 2.5 – *Avantages et Inconvénients des MLP (Multi-Layer Perceptron)*

Random forest :

La méthode Random Forest est une technique d'apprentissage supervisé fondée sur un ensemble d'arbres de décision. Elle est largement utilisée en raison de sa robustesse, de sa capacité à traiter des données complexes et de ses bonnes performances en classification comme en régression. Toutefois, comme tout algorithme, elle présente des points forts ainsi que certaines limitations. Le tableau suivant résume les principaux avantages et inconvénients associés à cette méthode.

- **La classification non-supervisée :** La classification non supervisée, aussi appelée clustering, est une technique d'apprentissage automatique où l'on cherche à regrouper des données similaires sans utiliser de labels prédéfinis. Contrairement à la classification supervisée, où chaque exemple d'entraînement est associé à une étiquette de classe connue, dans la classification non supervisée, l'algorithme doit découvrir lui-même la structure ou les motifs dans les données.

L'objectif principal de la classification non supervisée est d'identifier des groupes

Avantages	Inconvénients
Peut traiter efficacement des jeux de données contenant de nombreuses caractéristiques.	Moins facile à interpréter que des modèles simples comme les arbres de décision.
Réduit le risque de surapprentissage grâce à la combinaison des prédictions de plusieurs arbres.	Sensible aux données bruitées qui peuvent affecter la précision.
Convient aussi bien aux tâches de classification qu'à la régression.	Nécessite des ressources de calcul et mémoire importantes pour de grands ensembles d'arbres.
Permet d'estimer l'importance des caractéristiques dans le modèle.	—
Peut gérer des valeurs manquantes sans prétraitement complexe.	—

TABLE 2.6 – Avantages et Inconvénients de la méthode Random Forest

naturels ou des structures cachées dans un jeu de données. Cette approche peut être utilisée dans plusieurs domaines, notamment :

- **Segmentation de clients** en marketing pour cibler des groupes ayant des comportements similaires ;
 - **Regroupement d'images similaires** dans des bases de données d'images ou des systèmes de reconnaissance ;
 - **Détection d'anomalies**, comme la fraude ou les comportements inhabituels dans les données ;
 - **Réduction de la dimensionnalité** pour simplifier l'analyse ou la visualisation des données.
- **Quelques méthodes de classification non-supervisée :**
 - Mean Shift : Se base sur les pics de densité dans l'espace des caractéristiques.
 - K-Means : Regroupe les points en k clusters selon leur distance au centre de cluster.

2.3 Les méthodes de classification non-supervisées

2.3.1 La méthode de Mean Shift

L'algorithme Mean Shift est une méthode non paramétrique de regroupement de données (clustering) qui identifie les zones de forte densité dans un espace de caractéristiques

sans nécessiter de spécifier le nombre de clusters à l'avance. Il fonctionne en déplaçant itérativement une fenêtre vers la moyenne des points de données dans son voisinage, convergeant ainsi vers les modes (maximums locaux) de la distribution de densité. Cette approche est particulièrement utile pour détecter des structures complexes dans les données et est largement utilisée en traitement d'image et en vision par ordinateur[18].

Le principe de fonctionnement :

L'algorithme *Mean Shift* est une méthode de classification non supervisée basée sur l'analyse de la densité des données. Contrairement à des techniques comme K-Means, il ne nécessite pas de spécifier le nombre de clusters à l'avance. Il fonctionne en déplaçant progressivement une fenêtre de recherche vers les zones de plus forte densité de points dans l'espace des données, ce qui permet d'identifier automatiquement les centres de clusters. Cette approche est particulièrement utile pour détecter des structures complexes dans des jeux de données non étiquetés.

1. Principe de Base de Mean Shift

L'objectif de l'algorithme Mean Shift est de localiser les modes (maxima locaux) d'une fonction de densité de probabilité estimée à partir des données.

On utilise un noyau (par exemple, gaussien ou uniforme) pour estimer la densité autour de chaque point.

Chaque point est déplacé itérativement vers une région de plus haute densité selon la formule de Mean Shift.

2. Initialisation de Mean Shift

L'initialisation consiste à choisir les points de départ pour l'algorithme. Plusieurs approches existent :

(a) Initialisation par les Points de Données

Chaque point du dataset est utilisé comme centroïde initial. L'algorithme applique ensuite la procédure Mean Shift pour déplacer chaque point vers un mode.

Problème : Calculiquement coûteux si le dataset est grand.

(b) Initialisation par un Sous-Échantillonnage

Pour réduire la complexité, on peut initialiser Mean Shift sur un sous-ensemble des données (par exemple, via un échantillonnage aléatoire).

Moins précis, mais plus rapide.

(c) Initialisation sur une Grille (pour la Segmentation d’Image)

Dans le cas d’une image, on peut initialiser Mean Shift sur une grille régulière (pixels espacés) plutôt que sur tous les pixels.

Cela accélère le traitement tout en conservant une bonne approximation des régions homogènes.

(d) Initialisation via un Pré-Clustering (K-Means, etc.)

Une méthode hybride consiste à d’abord appliquer un algorithme rapide comme K-Means pour obtenir des centroïdes initiaux, puis à affiner avec Mean Shift.

3. Formule de Mean Shift

Pour chaque point initial x , le déplacement se calcule comme dans la formule 2.13 :

$$m(x) = \frac{\sum_{i=1}^n K(x - x_i)x_i}{\sum_{i=1}^n K(x - x_i)} - x \quad (2.13)$$

où K est le noyau (ex. : noyau gaussien), et x_i les points du dataset.

4. Critères d’Arrêt

L’algorithme s’arrête selon deux conditions :

- Le déplacement devient inférieur à un seuil de tolérance ε .
- Un nombre maximal d’itérations est atteint.

Exemple :

L’image ci-dessous illustre l’application de l’algorithme *Mean Shift* sur un ensemble de données synthétiques en deux dimensions. Cet algorithme de clustering non supervisé permet d’identifier automatiquement le nombre de groupes (ou clusters) présents dans les données, sans avoir à spécifier ce nombre à l’avance [19].

Nombre de clusters détectés : Le titre de la figure affiche *Estimated number of clusters* : 3, ce qui signifie que l’algorithme a trouvé trois regroupements distincts dans les données.

Couleurs et formes : Chaque point du nuage appartient à un cluster identifié par une couleur et un symbole particulier :

- Points **bleus** (ronds) pour le premier cluster.

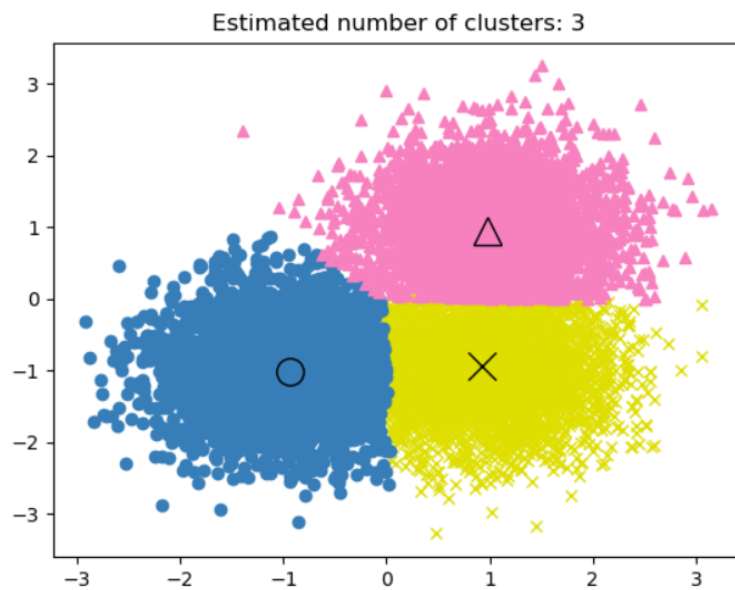


FIGURE 2.26 – Résultat du clustering avec Mean Shift. Trois clusters sont détectés automatiquement.

- Points **roses** (triangles) pour le second cluster.
- Points **jaunes** (croix) pour le troisième cluster.

Centroïdes détectés : Les grands symboles noirs au centre de chaque région représentent les *modes de densité* détectés par l'algorithme. Ils sont obtenus par déplacement itératif de chaque point vers la zone la plus dense selon la formule :

$$m(x) = \frac{\sum_{i=1}^n K(x - x_i)x_i}{\sum_{i=1}^n K(x - x_i)} - x \quad (2.14)$$

où K est un noyau (généralement gaussien) qui pondère la contribution des points voisins, et x_i sont les points du jeu de données.

Méthode : L'algorithme déplace progressivement chaque point vers un maximum local de densité. Une fois que les déplacements deviennent négligeables (selon un critère de convergence), le point est considéré comme fixé. Tous les points ayant convergé vers le même mode forment un même cluster.

Conclusion : Cet exemple montre la capacité de l'algorithme Mean Shift à détecter des regroupements naturels dans des données sans avoir à définir leur nombre au préalable. Contrairement à d'autres méthodes comme K-Means, il s'adapte à la forme et à la densité des clusters.

Les avantages et les inconvénients de Mean Shift :

Avantages de Mean Shift	Inconvénients de Mean Shift
Ne nécessite pas de spécifier le nombre de clusters à l'avance.	Coût computationnel élevé, surtout pour les grands ensembles de données.
Capable de détecter des clusters de formes arbitraires.	Sensible au choix de la largeur de bande (bandwidth).
Robuste aux valeurs aberrantes grâce au noyau.	Moins efficace pour les jeux de données très volumineux.
Approche non paramétrique basée sur la densité.	Performance réduite dans les espaces de grande dimension.
Peut localiser les modes de la densité (pas seulement du clustering).	Résultat parfois non déterministe selon l'implémentation.

TABLE 2.7 – *Avantages et inconvénients de l'algorithme Mean Shift***2.3.2 La méthode K-means**

Cet algorithme a été largement utilisé pour traiter de grands jeux de données en raison de sa rapidité. Nous étudions d'abord son fonctionnement, puis ses propriétés.

Principe

Supposons qu'il existe K classes distinctes. L'algorithme commence par sélectionner K centres de classes, notés μ_1, \dots, μ_K , parmi les individus. Ces centres peuvent être choisis soit par l'utilisateur pour leur représentativité, soit de manière aléatoire. Ensuite, les deux étapes suivantes sont répétées de manière itérative :

- Pour chaque individu qui n'est pas un centre de classe, on détermine le centre de classe le plus proche. Cela permet de définir K classes C_1, \dots, C_K , où :

$$C_i = \{\text{ensemble des points les plus proches du centre } \mu_i\}.$$

- Dans chaque nouvelle classe C_i , le centre de classe μ_i est recalculé comme le barycentre des points appartenant à C_i .

L'algorithme s'arrête selon un critère défini par l'utilisateur, qui peut être l'un des suivants :

- Le nombre maximal d'itérations est atteint.

- L'algorithme a convergé, c'est-à-dire que les classes ne changent plus entre deux itérations.
- L'algorithme a "presque" convergé, c'est-à-dire que l'inertie intra-classe ne s'améliore plus de manière significative.

Exemple

La figure 2.26 illustre le déroulement de l'algorithme sur un exemple avec quatre points : $a(-1, 1)$, $b(0, 1)$, $c(3, 0)$ et $d(3, -1)$, à classer en 2 groupes. On observe que, même si les centres de classes sont mal initialisés, l'algorithme converge en identifiant les "vraies" classes.

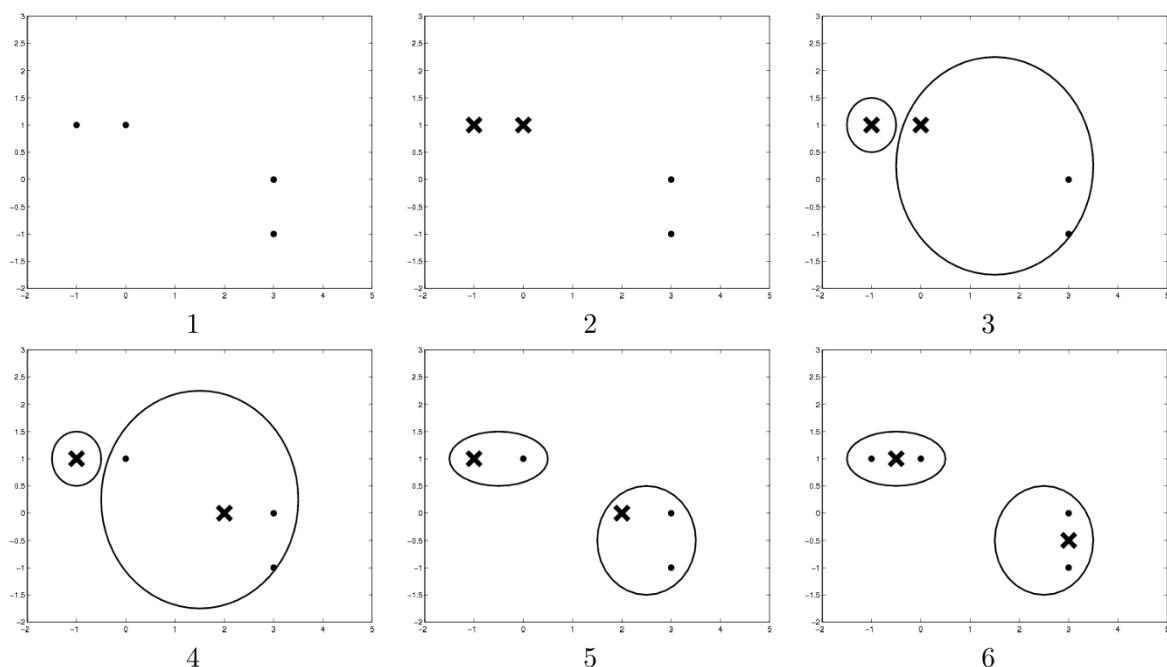


FIGURE 2.27 – Exemple de classification K-means avec quatre points

Propriétés de l'algorithme

Figure 2.27 Une illustration de l'algorithme K-means. (1) On dispose de 4 points à classer en deux classes. (2) À l'initialisation, deux de ces points sont choisis comme centres de classe. (3) Deux classes sont créées en regroupant les autres points en fonction du centre de classe le plus proche. (4) On définit les nouveaux centres de classe comme étant le barycentre des classes nouvellement créées. (5) On regroupe à nouveau les points. (6) On définit les nouveaux centres de classes. À l'étape suivante, rien ne change, l'algorithme a convergé.

Comment justifier l'algorithme des K -means dans le cadre de notre objectif de minimisation de l'inertie intra-classe I_W ? On peut démontrer (voir Annexes) que chacune des deux étapes de l'algorithme réduit I_W (ce qui est également observable visuellement dans l'exemple précédent). Ainsi, à chaque itération, la classification s'améliore au regard du critère I_W . Comme la marge d'amélioration est finie (l'inertie intra-classe ne pouvant pas être inférieure à celle de la partition optimale), l'algorithme est assuré de converger. En pratique, on remarque que très peu d'itérations suffisent dans la plupart des cas.

Cependant, il est essentiel de noter que le résultat final dépend fortement de l'initialisation de l'algorithme : selon les points choisis comme centres initiaux, les partitions obtenues peuvent varier significativement. Cette sensibilité à l'initialisation constitue le principal inconvénient de la méthode.

L'algorithme ne garantit pas la partition optimale globale, mais converge vers un minimum local de I_W . Pour atténuer ce problème, une solution consiste à exécuter l'algorithme plusieurs fois avec des initialisations différentes, puis à retenir la partition offrant la plus faible inertie intra-classe.[20]

2.4 Conclusion

Le Chapitre 2 a présenté une analyse approfondie des méthodes de classification supervisées et non supervisées, mettant en lumière leurs principes, leurs applications et leurs performances respectives. Les méthodes supervisées, telles que les SVM, les réseaux de neurones convolutifs (CNN), les MLP et les Random Forests, offrent des solutions robustes pour la classification des images SAR en s'appuyant sur des données étiquetées. Chacune de ces méthodes présente des avantages spécifiques, comme la capacité des SVM à gérer des frontières complexes grâce aux fonctions noyau, ou la puissance des CNN pour extraire automatiquement des caractéristiques hiérarchiques des images. Les Random Forests, quant à elles, se distinguent par leur robustesse et leur capacité à éviter le surapprentissage.

Les méthodes non supervisées, comme Mean Shift et K-means, permettent de regrouper des données sans étiquettes préalables, ce qui est particulièrement utile pour explorer des structures cachées dans les images SAR. Ces méthodes sont adaptées aux scénarios où les données étiquetées sont rares ou difficiles à obtenir.

En résumé, ce chapitre a souligné l'importance de choisir la méthode de classification adaptée en fonction des caractéristiques des données et des objectifs de l'étude. Les approches supervisées sont idéales pour des tâches précises nécessitant une haute performance, tandis que les méthodes non supervisées offrent une flexibilité précieuse pour l'exploration et l'analyse préliminaire des données. Ces concepts et outils ouvrent la voie à des applications avancées en télédétection, notamment pour la surveillance environnementale, la gestion des ressources naturelles et la cartographie des surfaces terrestres.

Chapitre 3

Résultats et analyses

3.1 Introduction

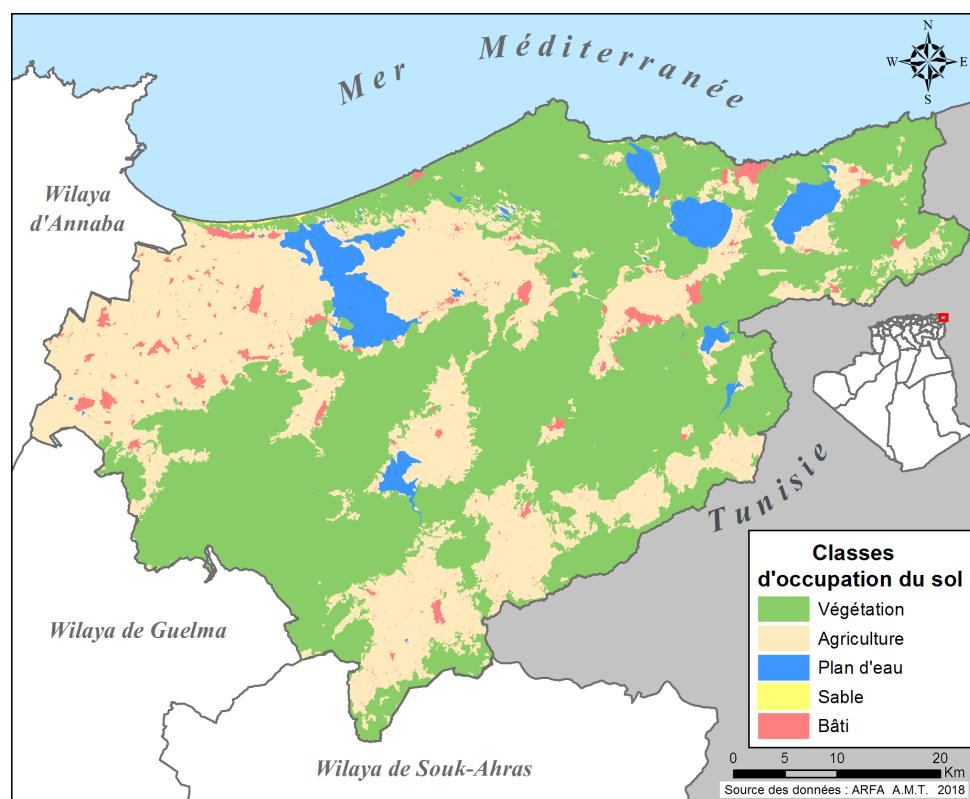
Dans notre projet, on cherche à détecter les zones forestières et à comparer ces zones après les feux de forêts en utilisant la méthode de classification supervisée RANDOM FOREST (RF) et aussi par les méthodes de classification non supervisée. On va utiliser des images de la wilaya de Tarf. Les images sont des images acquises à travers le site de l'agence spatiale européenne (Copernicus Browser) ; ce sont des images bi-polarimétriques (dual-pol) capturées par le satellite Sentinel-1 dont on a parlé dans le premier chapitre.

3.2 Site d'étude

Située à l'extrême nord-est de l'Algérie, la wilaya d'El Tarf est comprise entre les parallèles $36^{\circ}23'25''$ et $36^{\circ}57'7''$ de latitude Nord et $7^{\circ}39'49''$ et $8^{\circ}40'52''$ de longitude Est. Issue du découpage administratif de 1984, elle s'étend sur une superficie de 2 882 km² et comprend 24 communes. Elle est délimitée au nord par la mer Méditerranée, à l'est par la frontière algéro-tunisienne, à l'ouest par la wilaya d'Annaba, au sud-ouest par la wilaya de Guelma et au sud par la wilaya de Souk Ahras.

Le territoire de la wilaya d'El Tarf regroupe 5 classes d'occupation du sol figure 3.1 : végétation, agriculture, sable, plans d'eau et espace bâti. La végétation, composée essentiellement de forêts et de maquis, occupe une surface de 161 464 ha, soit 56,03 pour cent de la surface totale. L'agriculture avec 36,47 pour cent vient en seconde position comptabilisant une surface de 105 118 ha, elle comprend les terres de cultures, l'arboriculture et

les pâturages. Les zones de sable correspondent au cordon dunaire littoral localisé dans la partie nord de la wilaya avec une surface de 427 ha, soit 0,15 pour cent. La wilaya est aussi caractérisée par une importante zone humide, notamment au sein du parc national d'El Kala. Ces plans d'eau, qui englobent les lacs, les marais et les barrages, occupent une surface totale de 13 556 ha, soit 4,70 pour cent. Enfin, l'espace bâti correspond à 2,65 pour cent de la surface de la wilaya, soit 7 638 ha, dominé par l'habitat rural. En plus de 2 villes (El Kala et El Tarf) et 113 villages, on compte plus de 5 539 maisons rurales et 7 780 fermes éparses (figure3.1)[20].



[20]

FIGURE 3.1 – Localisation et classes d'occupation du sol de la wilaya d'El Tarf

3.3 Présentation des données

3.3.1 Présentation des images SLC

- La première image Sentinel-1 SLC utilisée dans cette étude a été acquise le 03 septembre 2021, soit avant l'événement incendiaire. Elle correspond à une scène capturée par le satellite Sentinel-1A, en mode Interferometric Wide Swath (IW), dans le cadre du programme Copernicus. Cette image est fournie au niveau 1 sous forme de produit SLC (Single Look Complex), dans un format SAFE contenant les bandes radar en coordonnées slant range, les métadonnées, et les fichiers d'annotation. Chaque pixel de l'image contient une valeur complexe (composantes I et Q), permettant d'analyser à la fois l'amplitude et la phase du signal radar. L'image présente une résolution spatiale d'environ 3,6 mètres en portée (slant range) et 22,5 mètres en azimuth, avec une largeur de fauchée d'environ 250 km. La polarisation utilisée est VV, ou VV/VH si le produit est en double polarisation. Cette image représente la forêt en état non brûlé et constitue la scène de référence pour l'analyse temporelle du changement (figure3.2).

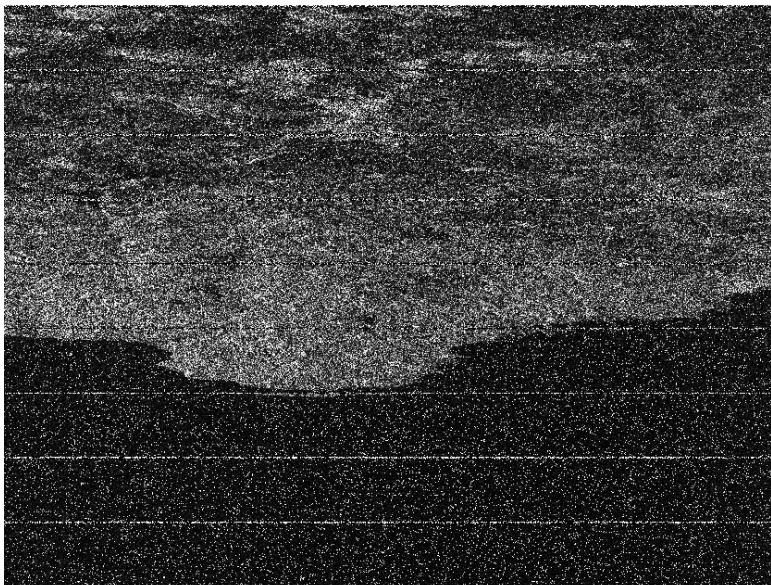


FIGURE 3.2 – Images SLC Taref 03/09/2021

- La seconde image Sentinel-1 SLC a été acquise le 5 septembre 2022, soit après la période des incendies ayant affecté la région d'étude située dans la wilaya d'El Tarf, à l'extrême nord-est de l'Algérie. Elle provient également du satellite Sentinel-1A, dans les mêmes conditions d'acquisition que la précédente (mode IW, polarisation

VV ou VV/VH). Il s'agit d'un produit SLC niveau 1, délivré au format SAFE, contenant des données radar complexes. Cette image conserve les mêmes caractéristiques techniques que la première, notamment en termes de résolution spatiale et de couverture. Elle représente une scène post-incendie et est exploitée afin de détecter les changements dans la diffusion du signal radar qui peuvent être liés à la perte de couverture végétale, à la modification de la rugosité de surface, ou à la présence de cendres et de sols nus (figure 3.3).

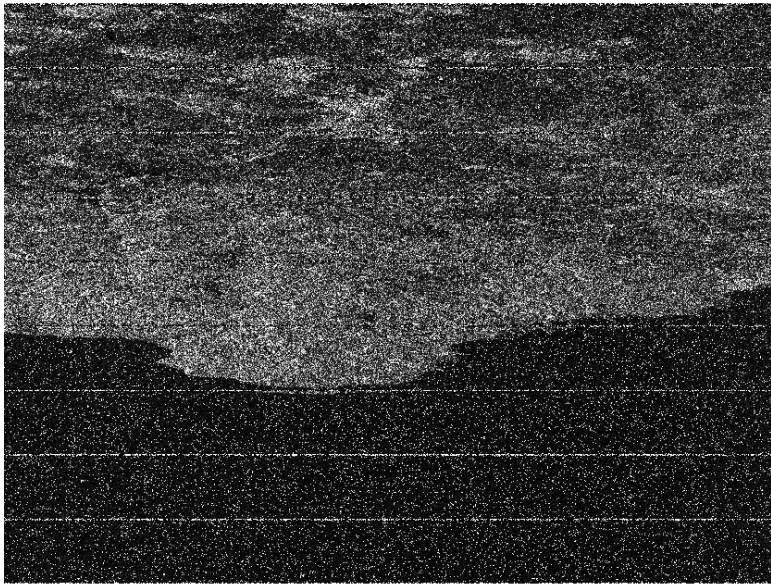


FIGURE 3.3 – Images SLC Taref 05/09/2022

3.3.2 Présentation des images GRD

- L'image présentée dans la figure 3.4 est une image SAR capturée le 29/08/2021 par le satellite Sentinel-1 en polarisation VH et VV couvrant intégralement la wilaya d'El Taref. Dans SNAP, la bande Amplitude VH affiche un raster de $26,35 \times 16,66$ pixels. Comme il s'agit d'un produit IW-GRD, le pas de pixel natif est d'environ $10 \text{ m} \times 10 \text{ m}$, ce qui se traduit par une emprise au sol voisine de 263 km (est-ouest) \times 167 km (nord-sud). Cette résolution permet de distinguer les principaux éléments hydrographiques, les zones urbaines et les formations forestières caractéristiques de la région d'El Taref, tout en conservant le niveau de détail requis pour l'analyse régionale.

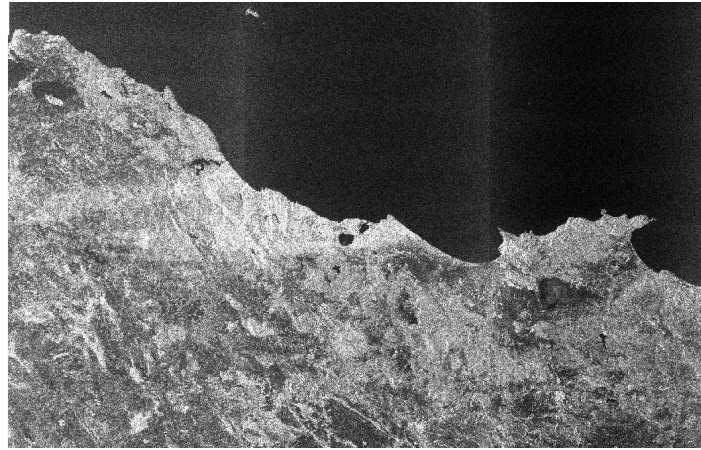


FIGURE 3.4 – Image GRD du 29 aout 2021

- L'image présentée dans la figure 3.5 est une image radar (SAR) acquise le 5 septembre 2022 par le satellite Sentinel-1, en double polarisation VH et VV. Elle couvre l'ensemble de la wilaya d'El Taref. Dans SNAP, la bande "Amplitude VH" est affichée sous forme d'un raster de $26,35 \times 16,66$ pixels. Étant un produit de type IW-GRD, cette image possède un pas de pixel natif d'environ 10 mètres par 10 mètres, ce qui correspond à une emprise au sol d'environ 263 km dans le sens est-ouest et 167 km dans le sens nord-sud. Cette résolution spatiale permet de distinguer clairement les principaux éléments du paysage régional, tels que le réseau hydrographique, les zones urbaines, ainsi que les formations forestières typiques de la région d'El Taref. Elle offre ainsi un niveau de détail adapté à une analyse à l'échelle régionale.



FIGURE 3.5 – Image GRD du 05 septembre 2022

3.3.3 Le prétraitement des données

1. accédez au site copernicus browser et ouvrir un compte sur le site pour pouvoir rechercher et télécharger les images concernés dans notre mémoire.
2. on zoom sur la région qu'on veut étudier et choisir la zone souhaité à l'aide su curseur .
3. on sélectionne sur le volet SEARCH le satellite sentinel-1 et on sélectionne le type d'image (SLC ou GRD).
4. on clique sur TIME RANGE et on choisit les dates concernés.
5. on clique sur le bouton search et on aperçoit plusieurs images selon la date et la zone.
6. dernièrement on télécharge l'image qui couvre notre site d'étude.

3.4 Traitement des images SLC

Les images sous forme SLC obtenues par Copernicus Browser de Sentinel 1 sont sous format brut. Afin de les exploiter, on suit les traitements présentés dans la figure 3.6 ci-dessous :

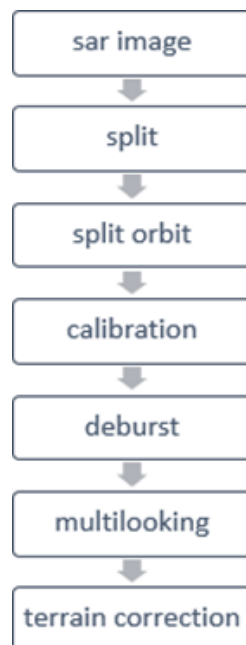


FIGURE 3.6 – Le traitement des données SAR des images SLC [1]

3.4.1 Ouverture de l'image SAR dans SNAP

L'ouverture de l'image SAR constitue la première étape du processus de traitement. Ces images généralement fournies par Sentinel-1 sont au format Zip .

On lance logiciel SNAP et puis en charge les deux images avec deux dates différents l'une a été acquise le 03 septembre 2021 et l'autre 05 septembre 2022.

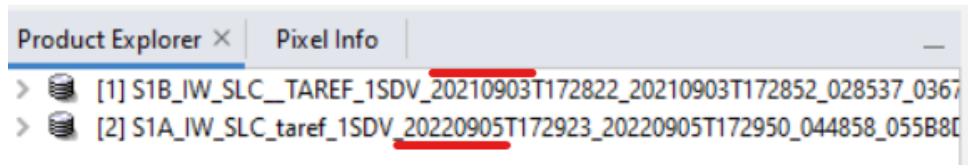


FIGURE 3.7 – Ouverture des images SLC dur SNAP

3.4.2 Split

Cette étape permet de sélectionner une sous-bande de l'image SAR. Le produit Sentinel-1 SLC contient généralement trois sous-bandules (IW1, IW2, IW3) correspondant aux différentes lignes de vol. Le Split permet d'extraire un seul IW (souvent IW2, qui couvre le centre de l'image) et la polarisation désirée (VV, VH, etc.).

Pour accéder à l'outil Split, on doit cliquer sur la barre de menu : Radar → Sentinel-1 TOPS → S-1 TOPS Split comme indiqué dans la figure 3.8.

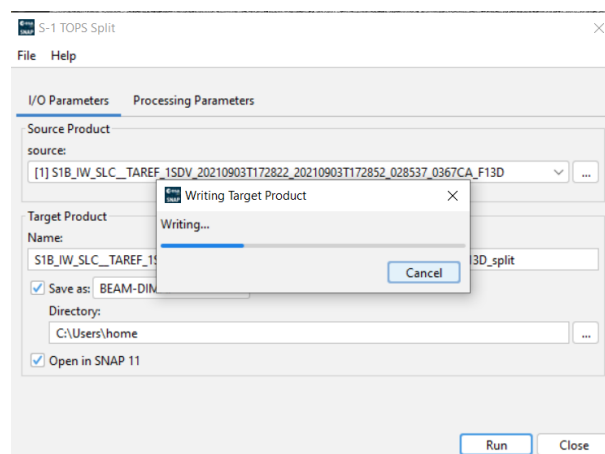


FIGURE 3.8 – split

3.4.3 Split Orbit

L'étape "Apply Orbit File" (souvent appelée Split Orbit de manière informelle) permet de corriger les erreurs de positionnement de l'orbite du satellite en utilisant des données d'orbite précises fournies après l'acquisition. Cela améliore significativement la géolocalisation des images SAR.

les procédures dans SNAP commencent par sélectionner le produit après Split et lancer l'outil "Apply Orbit File" par la barre de menu : Radar → Apply Orbit File comme indiqué dans la figure 3.9.

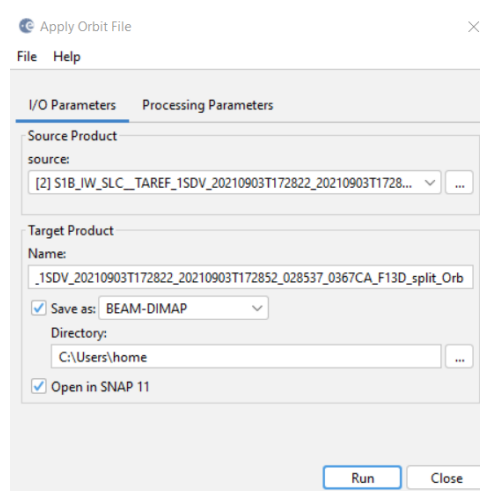


FIGURE 3.9 – Apply orbit file

3.4.4 Calibration

L'étape de calibration permet de convertir les données SAR brutes (valeurs numériques arbitraires) en mesures physiques significatives, telles que le σ^0 (sigma-nought), β^0 (beta-nought) ou γ^0 (gamma-nought). Cela rend les images comparables dans le temps et l'espace, en éliminant les effets instrumentaux et en assurant la cohérence radiométrique des produits. Ces coefficients représentent respectivement la rétrodiffusion normalisée selon différentes géométries d'acquisition, utiles pour les analyses quantitatives. Pour lancer l'outil de calibration en suit le menu suivant : Radar → Radiometric → Calibration comme indiqué dans la figure 3.10.

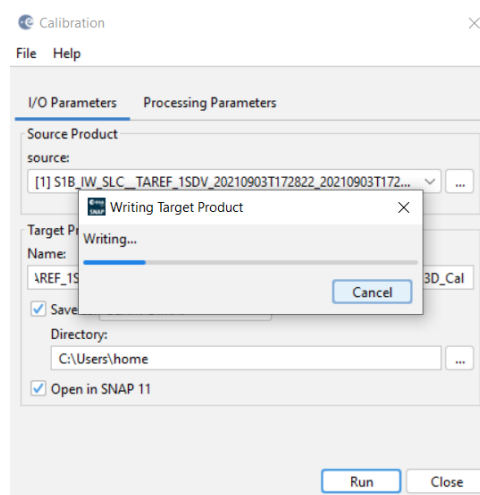


FIGURE 3.10 – Calibration

3.4.5 Deburst

L'imagerie Sentinel-1 SLC en mode TOPS est acquise par sous-blocs appelés "bursts" dans chaque sous-swath (IW1, IW2, IW3). Ces bursts sont discontinus et juxtaposés avec des recouvrements. L'étape de Deburst permet de fusionner ces bursts en une image continue, indispensable pour les traitements comme la mosaïque, la classification ou la géocodage. pour faire le Deburst en sélectionnant le produit calibré puis en ouvrant l'outil Deburst dans le menu : Radar → Sentinel-1 TOPS → S-1 TOPS Deburst comme indiqué dans la figure 3.11.

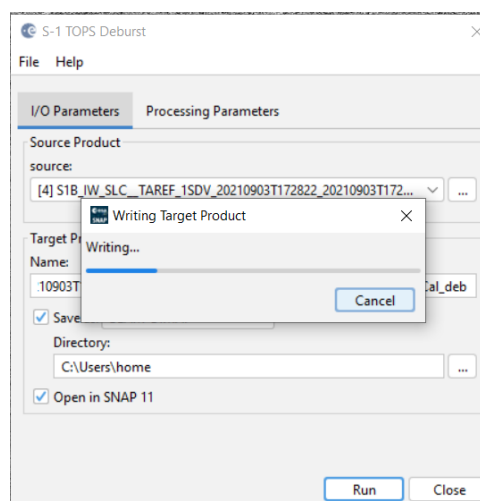


FIGURE 3.11 – Deburst

3.4.6 Multi-looking

L'étape de **Multilooking** permet de réduire le bruit inhérent aux images radar, connu sous le nom de *speckle*, en moyennant plusieurs pixels dans les directions azimutale et de portée. Cette opération améliore la qualité visuelle de l'image au détriment d'une légère perte de résolution spatiale, rendant l'image plus adaptée aux traitements visuels et cartographiques. Pour lancer l'outil Multilook en suivant les étapes suivantes comme indiqué dans la figure 3.12 : Radar → SAR Utilities → Multilook.

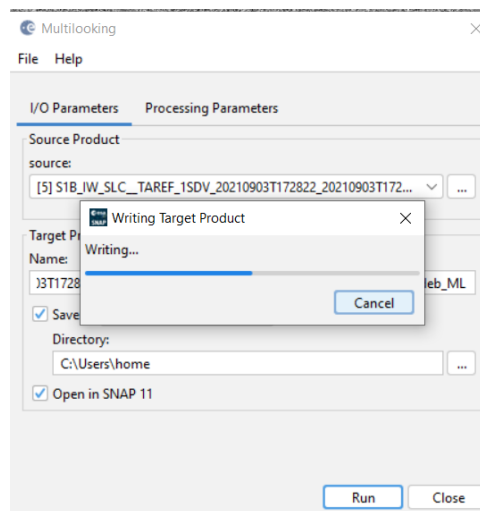


FIGURE 3.12 – Multi-looking

3.4.7 Terrain correction

La correction terrain permet de corriger les effets géométriques liés à la topographie et à la géométrie d'acquisition radar. Grâce à l'utilisation d'un MNT (Modèle Numérique de Terrain), cette étape projette les pixels de l'image SAR dans un référentiel géographique réel. Le résultat est une image géoréférencée et orthorectifiée, directement exploitable dans un SIG. Pour lancer l'outil de correction terrain on clique sur : Radar → Geometric → Terrain Correction → Range-Doppler Terrain Correction comme indiqué dans la figure 3.13.

Nous allons donc répéter toutes ces étapes de prétraitement pour la deuxième image. L'ensemble des étapes de prétraitement appliquées aux images Sentinel-1 SLC dans SNAP a permis de produire une série d'images intermédiaires reflétant l'évolution des données brutes vers une image radar géoréférencée et analysable. Le résultat obtenu à chaque étape est indiqué dans la figure 3.14.

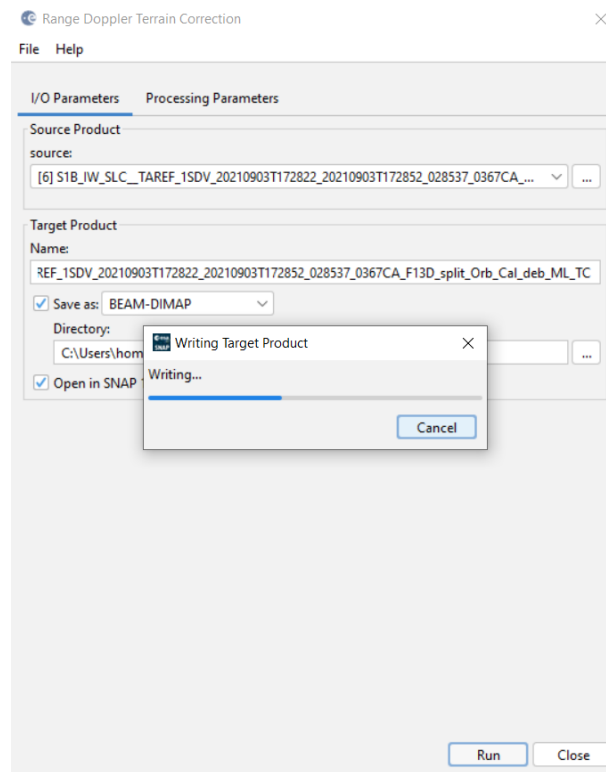
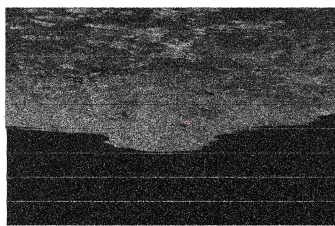
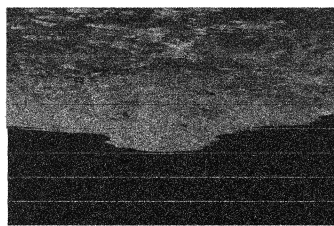


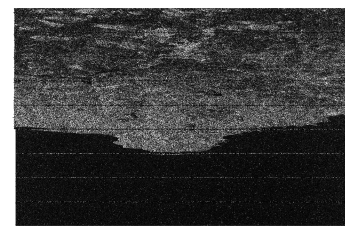
FIGURE 3.13 – Terrain Correction



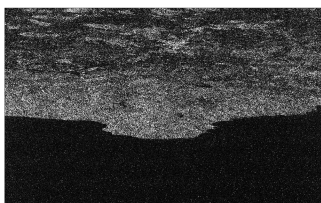
(a) Split



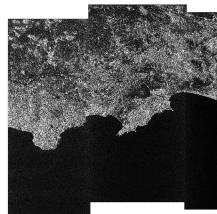
(b) Orbit



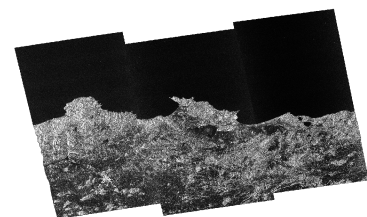
(c) Calibration



(d) Deburst



(e) Multi-looking



(f) Terrain correction

FIGURE 3.14 – Étapes de traitement SAR dans SNAP

3.5 Traitement des images GRD

On ouvre les deux images GRD de la même manière que les deux images SLC la figure ci-dessous montre la région sur laquelle on veut faire notre étude : Pour traiter les images

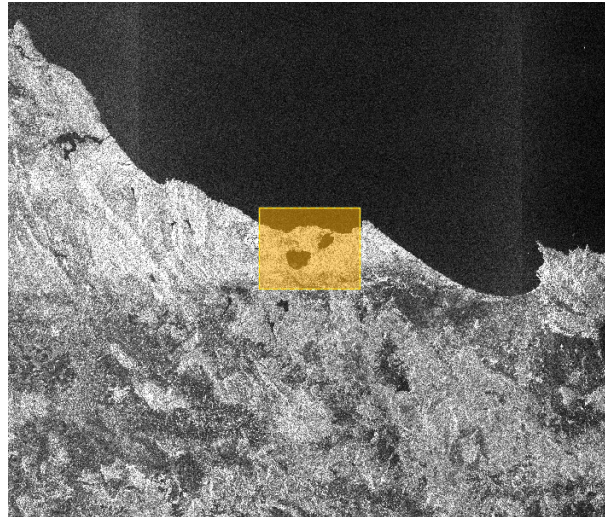


FIGURE 3.15 – image GRD montrant notre site d'étude par le rectangle orange

GRD on suit les étapes indiquées dans la figure 3.16 :

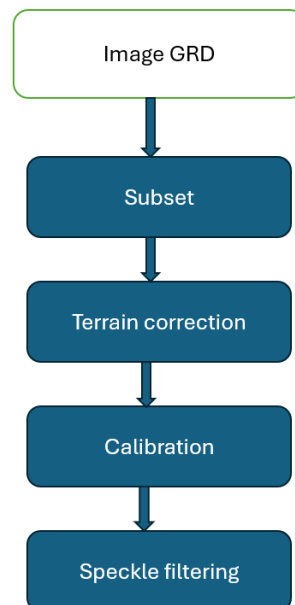


FIGURE 3.16 – Étapes de traitement des images GRD

3.5.1 Subset

la fonction subset est utilisée pour créer un sous-ensemble de l'image qu'on a téléchargée. on met les mêmes geo coordinates pour avoir le même subset :

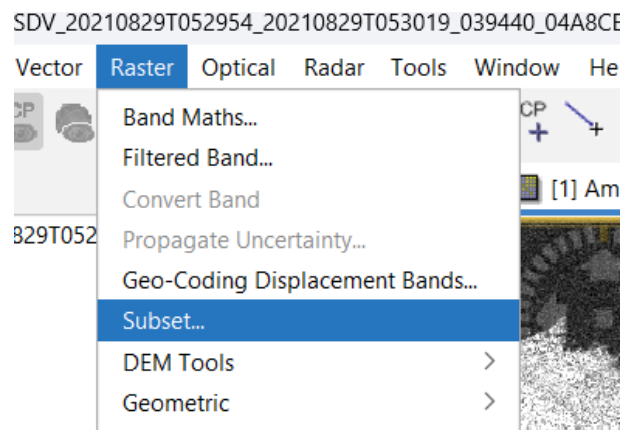


FIGURE 3.17 – accéder à subset sur SNAP 11

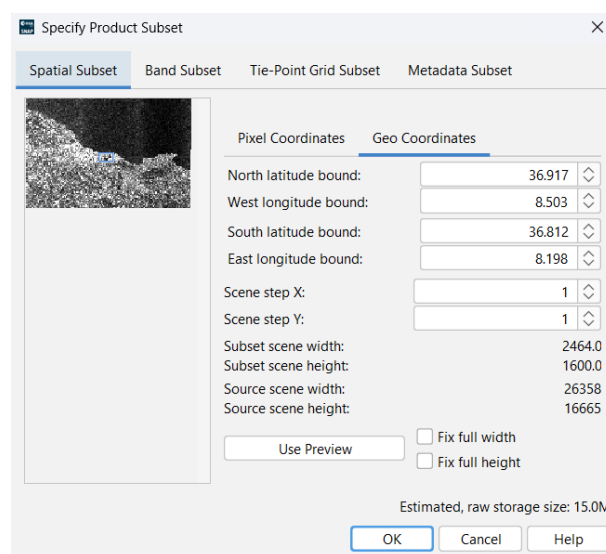
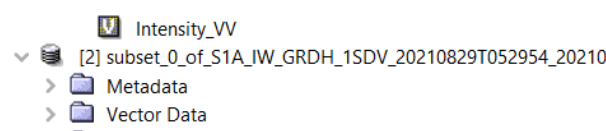


FIGURE 3.18 – geo coordinates de notre subset

dès qu'on appuie sur OK dans notre boîte de dialogue, notre subset sera créé



finallement notre subset a été créé comme présenté dans la figure 3.19 :

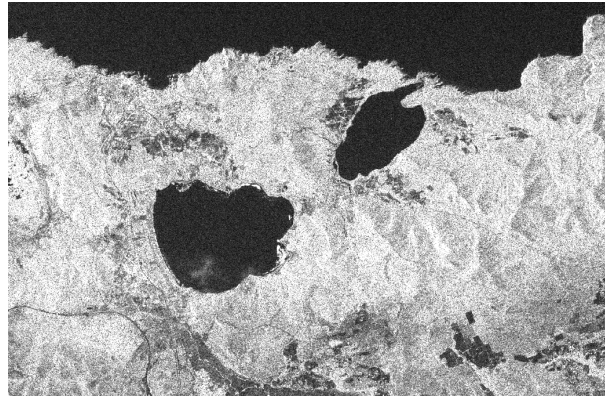


FIGURE 3.19 – l'image GRD après subset

3.5.2 Terrain correction

Le terrain correction est un processus qui corrige les distorsions géométriques des images satellitaires causées par le relief. Sur snap 11 on effectue la fonction terrain correction comme montré dans la figure 3.20 :



FIGURE 3.20 – terrain correction 29/08/2021

3.5.3 Calibrate

La fonction calibrate transforme les pixels d'une image Sentinel-1 en valeurs de sigma0. Sur SNAP 11, on accède à la fonction subset comme on a montré dans la figure 3.19 : notre image sar est donc montrée dans la figure 3.22

3.5.4 Speckle filtering

Le filtrage du speckle dans les images SAR vise à réduire le bruit granulaire tout en préservant les détails. On utilise souvent des filtres statistiques (comme Lee, Frost ou Kuan) notre image SAR après speckle filtering est montrée dans la figure 3.23

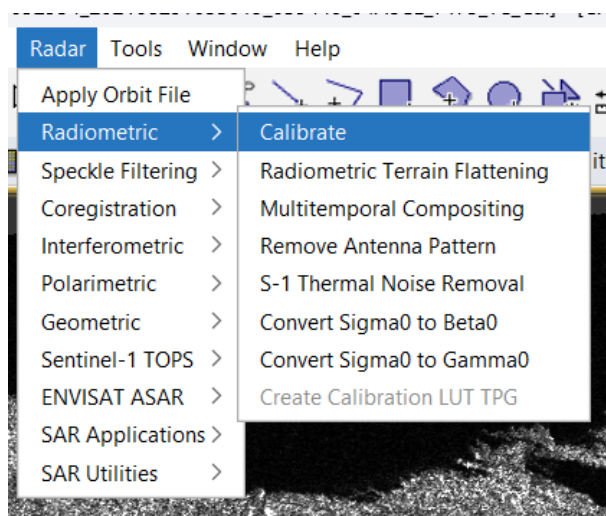


FIGURE 3.21 – accéder à calibrate sur SNAP 11

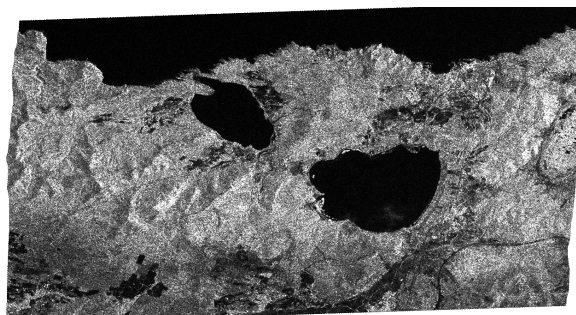


FIGURE 3.22 – l'image du 29/08/2021 après calibration

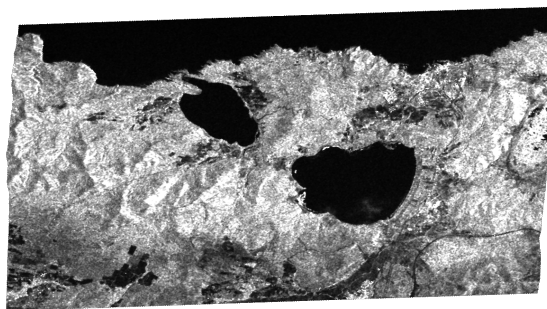


FIGURE 3.23 – l'image du 29/08/2021 après speckle filtering

3.6 Classification avec Random Forest

Après avoir traité l'image GRD sur SNAP 11 suivant les étapes précédentes, on commence notre classification en suivant le diagramme montré dans la figure 3.24

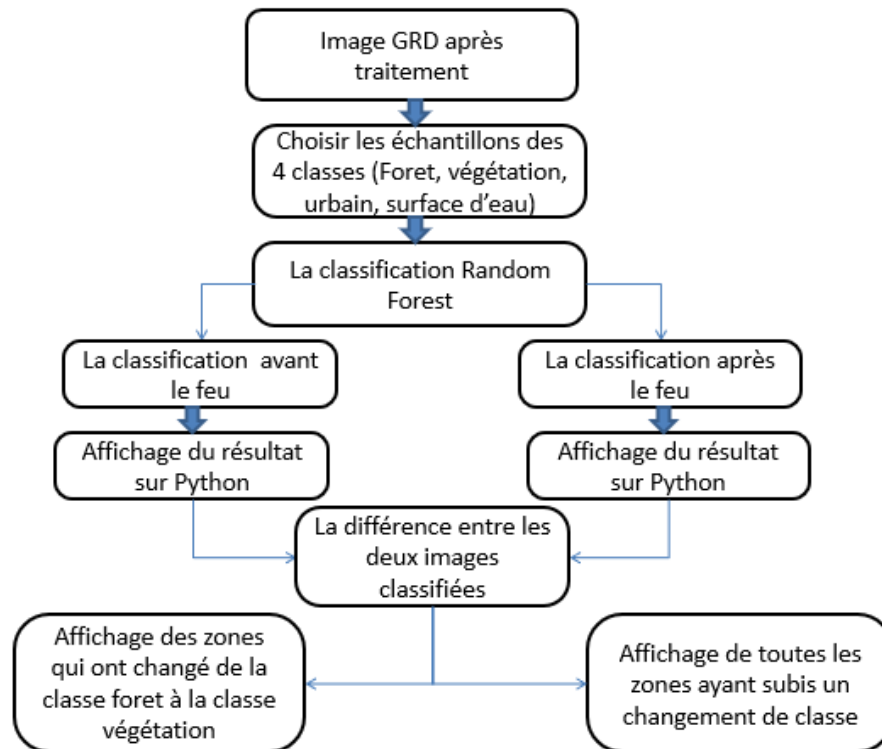


FIGURE 3.24 – Organigramme de détection des zones brûlées par Random Forest

3.6.1 Classifiacation avant le feu

Dans cette classification, on choisit 4 classes :1- forest (en vert), 2- urban(en rouge), 3- vegation(en jaune), 4- water ou surfacique (en bleu) comme représenté dans la figure 3.25 :

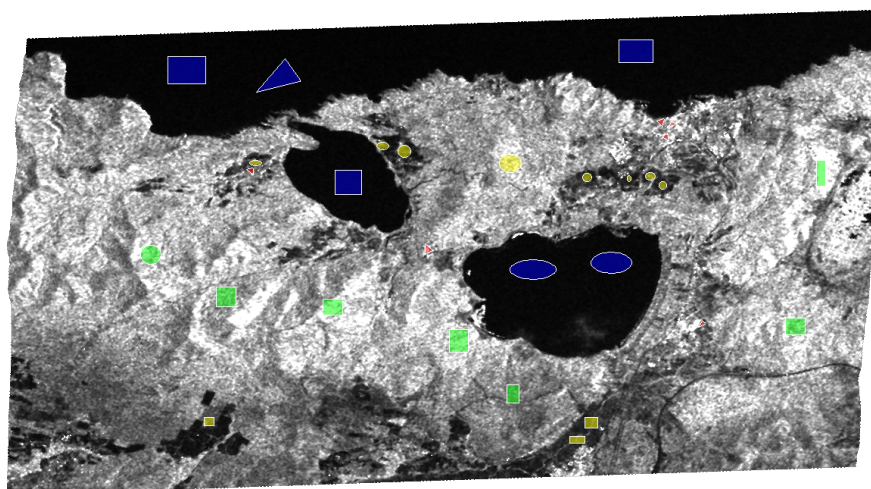
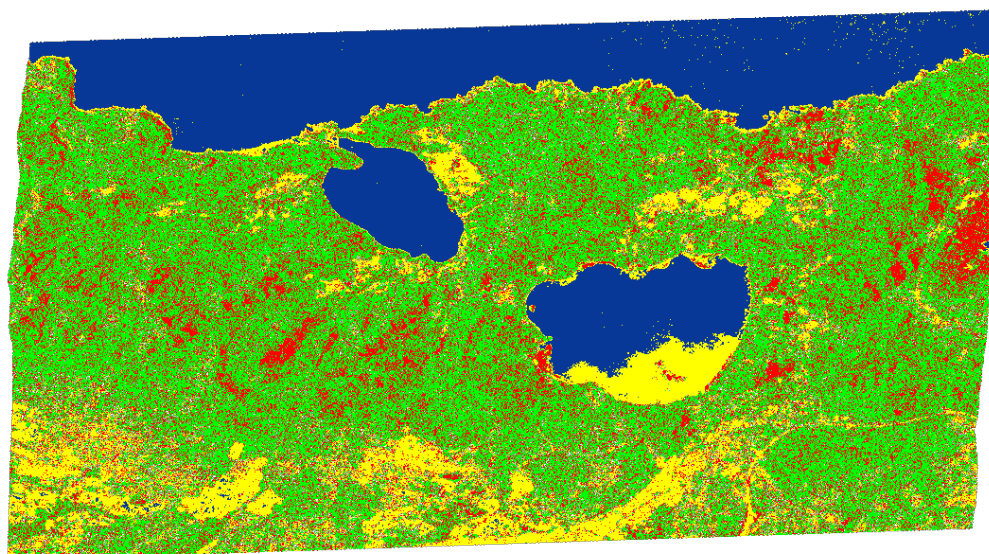


FIGURE 3.25 – les échantillons de classification sur l'image du 29/08/2021

après cette étape, on commence notre classification random forest sur snap. Une image de notre classification a été générée par SNAP comme montré dans la figure 3.26 :



Urbain	0
Végétation	1
Surface d'eau	2
Forêt	3

FIGURE 3.26 – classification random forest du 29/08/2021

Remarque :

On remarque que la classe forest est la classe la plus présente sur cette image. et on retrouve plusieurs zones de végétation et une minorité des zones urbaines comme dans la figure 3.1.

3.6.2 Classification après le feu

On refait la même classification mais cette fois sur l'image du 05/09/2022 (après le feu). et on remarque les zone ayant subis des différences le résultat est montré dans la figure 3.27.

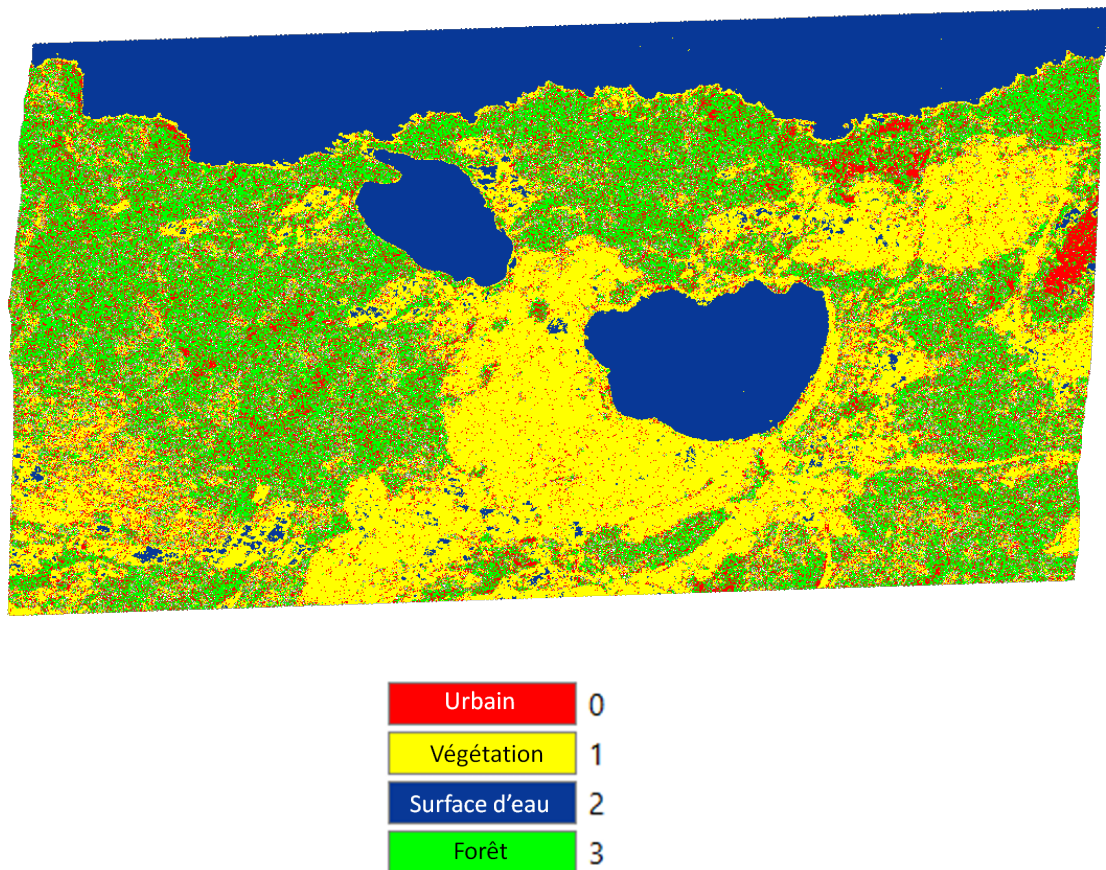


FIGURE 3.27 – classification random forest du 05/09/2022

Remarque :

On remarque dans cette image qu'une grande partie a changé de couleur du vert vers le jaune, donc de la classe forêt à la classe végétation, et ça, c'est dû aux feux de forêts.

3.7 Affichage des résultats sur Python

Après traitement et classification sur SNAP, on ouvre notre image sur Python.

- On commence par ouvrir l'image du 29/08/2021 (l'image classifiée avant le feu) 3.28.
- Ensuite on ouvre notre image du 05/09/2022 (l'image classifiée pendant le feu) figure 3.29.
- Pour comparer deux images GRD classifiées en Python, on commence par ouvrir les deux fichiers raster et lire leurs données sous forme de tableaux numériques grâce à la bibliothèque rasterio. On vérifie que les deux rasters ont exactement les mêmes dimensions pour garantir que chaque pixel correspond bien à la même position spatiale. Ensuite, on calcule la différence pixel par pixel en soustrayant la classe du second raster à celle du premier : cela génère une nouvelle matrice où chaque valeur indique s'il y a eu un changement de classe (valeur non nulle) ou non (zéro). On enregistre ensuite cette image de différence sous forme d'un nouveau fichier GeoTIFF en réutilisant les métadonnées du premier fichier pour conserver le même format et la même géoréférence. Enfin, on visualise le résultat sous forme d'image colorée pour localiser facilement les zones où des changements se sont produits entre les deux dates
- Après avoir calculé la différence entre deux images classifiées, il est souvent utile de repérer directement les pixels qui ont changé de classe. Pour cela, on crée un masque binaire de changement en comparant les deux images : l'instruction `labels1 != labels2` renvoie un tableau de valeurs `True` pour les pixels où les classes sont différentes, et `False` sinon. C'est un masque logique qui permet d'isoler uniquement les zones modifiées. Ensuite, pour savoir de quelle classe à quelle classe chaque pixel a changé, on peut calculer : $changemap = labels2 - labels1$
- On affiche visuellement les zones qui ont changé entre les deux images. Le masque binaire (change mask) est représenté en niveaux de gris, où les pixels en blanc correspondent aux endroits modifiés, et ceux en noir aux zones restées identiques. Cela permet de voir rapidement sur une image où les changements de classe ont eu lieu comme illustré la figure 3.30
- On récupère toutes les paires de classes correspondant aux pixels qui ont changé (classe initiale et classe finale), puis on extrait les combinaisons uniques de ces changements. Cela permet de connaître toutes les transitions possibles entre classes

qui ont eu lieu, sans doublons. Ensuite, on affiche ces changements uniques pour voir rapidement quelles transformations ont été observées.

- On compte le nombre de pixels pour chaque type de changement entre classes, en regroupant toutes les paires « classe d'origine → classe finale ». Cela permet de savoir précisément combien de pixels sont passés d'une classe à une autre, ce qui aide à mesurer l'importance de chaque transformation.
- On crée un masque binaire qui marque par True les pixels ayant changé spécifiquement de la classe 3 (forêt) à la classe 1 (agriculture). Cela permet d'isoler et d'étudier uniquement cette transformation précise parmi toutes les modifications détectées.
- On affiche l'image 3.31 du masque binaire correspondant à la transformation spécifique de forêt (3) vers agriculture (1). Les pixels en blanc indiquent où ce changement a eu lieu, ce qui permet de visualiser clairement la répartition spatiale de cette transition.
- On nettoie le masque des changements spécifiques en supprimant les petites régions isolées (moins de 50 pixels), qui peuvent être du bruit ou des erreurs. Cela permet de mieux se concentrer sur les zones significatives où la transformation a réellement eu lieu. Ensuite, on affiche le masque filtré pour visualiser les changements plus nets et mieux définis comme illustré dans la figure 3.32.

Remarque

On remarque que l'empreinte observée dans la figure 3.32 correspond fortement à celle de l'image optique de la figure 3.33, notamment en ce qui concerne les zones touchées par les feux de forêt qui sont affichées en blanc dans la figure 3.32, qui apparaissent similaires à celles de l'image optique (figure 3.33).

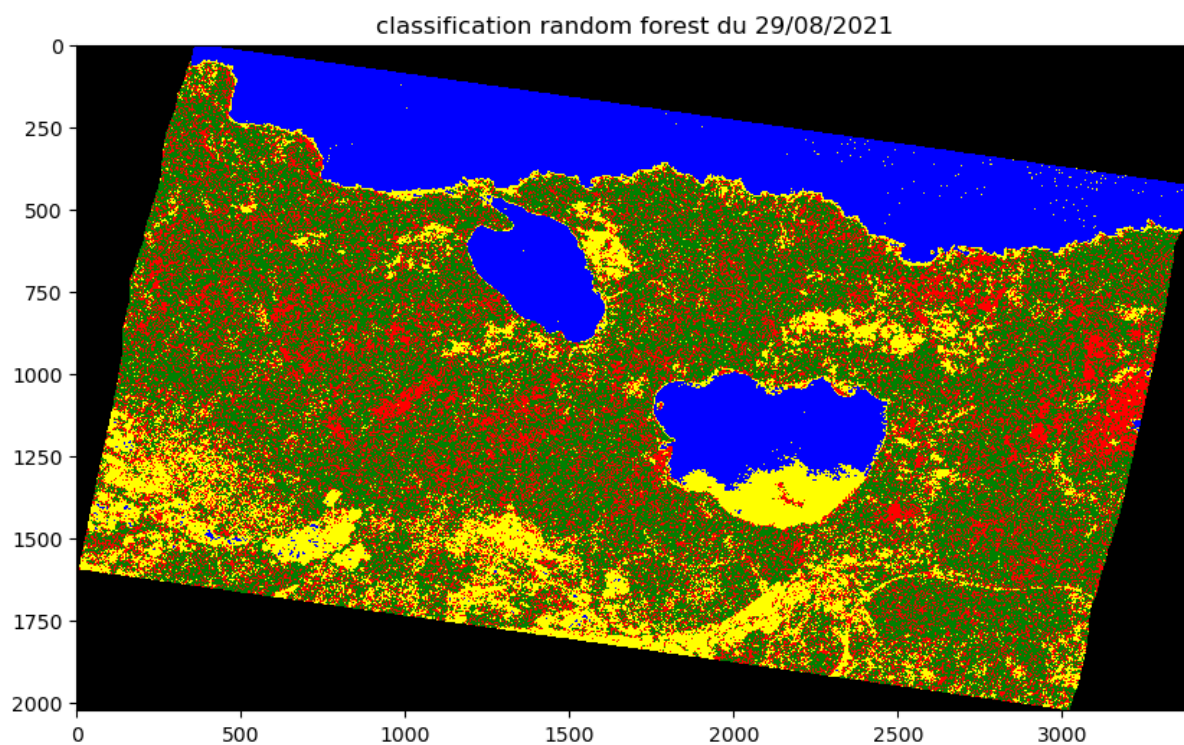


FIGURE 3.28 – l'image classifiée du 29/08/2021 sur PYTHON

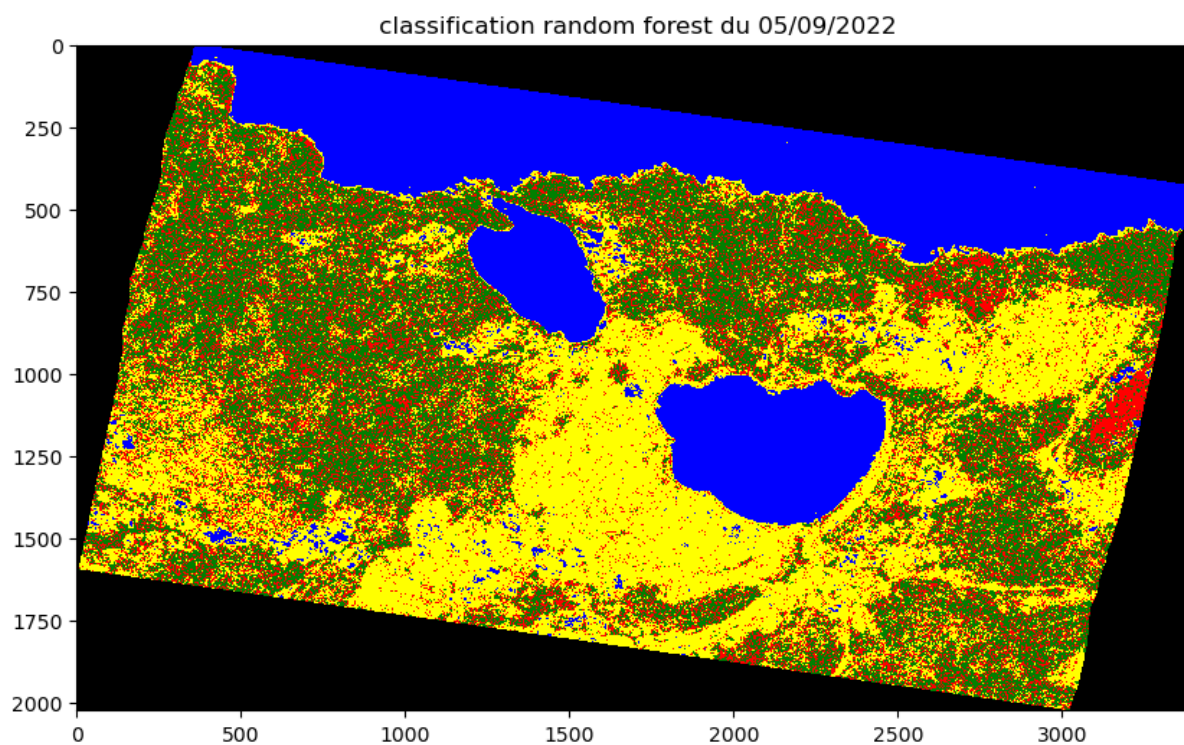


FIGURE 3.29 – l'image classifiée du 05/09/2022 sur PYTHON

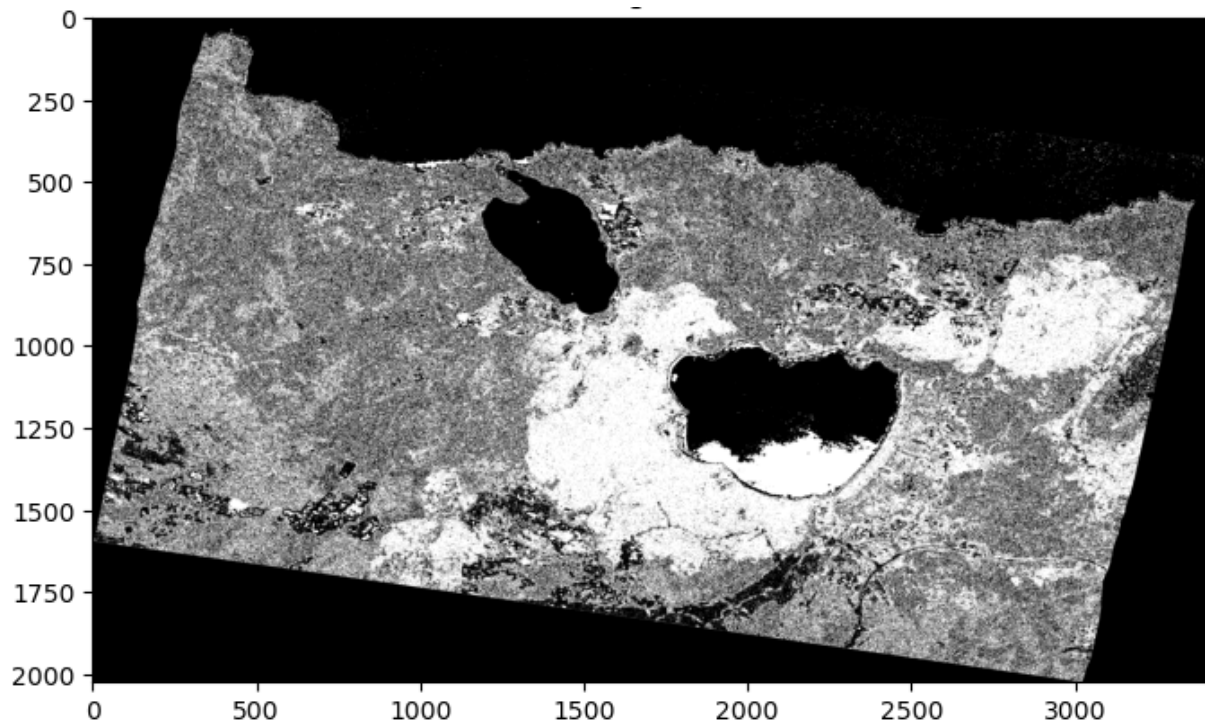


FIGURE 3.30 – les zones ayant subis des changements affichés sur Python

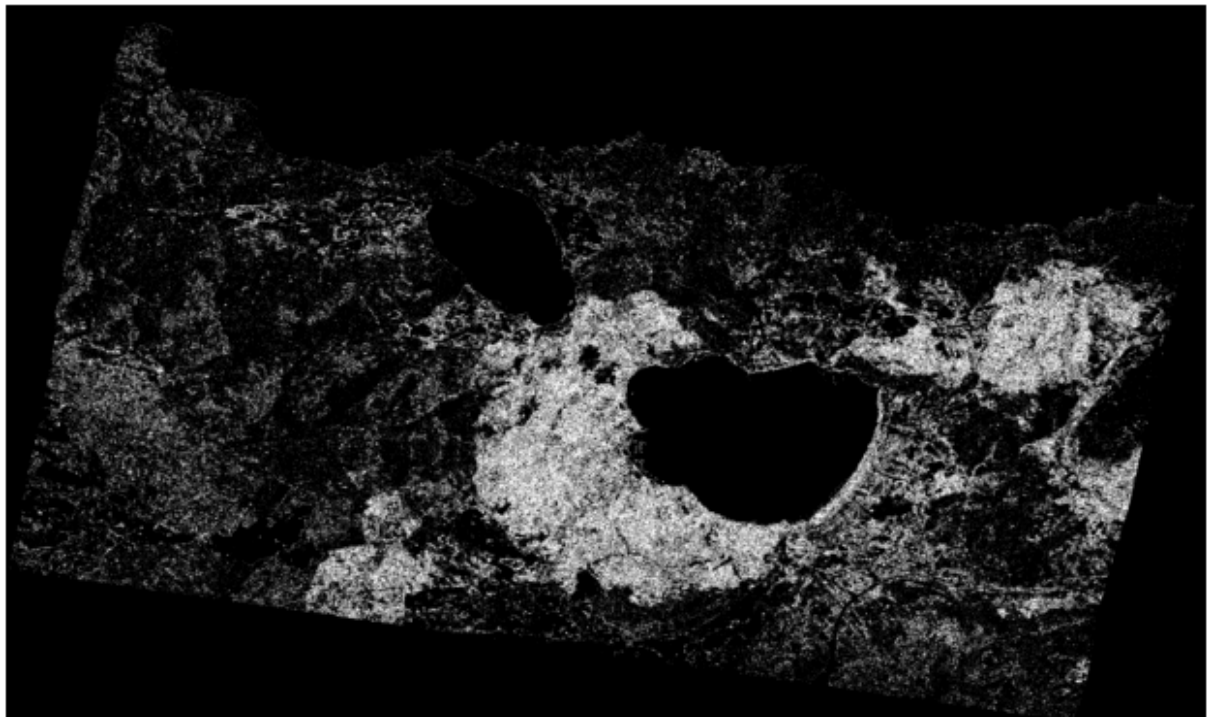


FIGURE 3.31 – Carte de détection des changements post-incendie (forêt → végétation)



FIGURE 3.32 – Carte de détection des changements filtré post-incendie (forêt → végétation)



FIGURE 3.33 – image optique prise par sentinel-2

3.8 Conclusion

Dans ce chapitre, nous avons appliqué l'ensemble des méthodes de traitement et de classification aux images SAR de Sentinel-1 afin d'analyser les changements survenus dans la wilaya d'El Tarf après un incendie de forêt.

Les prétraitements effectués sur les images SLC et GRD (calibration, correction topographique, filtrage du speckle) ont permis d'obtenir des données de qualité, exploitables pour la classification. Grâce à l'algorithme Random Forest, nous avons pu générer des cartes de classification pour deux dates : avant l'incendie (29 août 2021) et après l'incendie (5 septembre 2022).

Les résultats montrent clairement une réduction significative de la surface forestière, remplacée par des zones de végétation plus clairsemée ou du sol nu. L'affichage des changements sur Python a permis de visualiser précisément les zones touchées et de quantifier l'étendue des dégâts. La carte de détection des changements met en évidence la conversion de classes « forêt » vers « végétation » ou « sol découvert », ce qui valide l'efficacité de notre approche pour une surveillance rapide et précise des impacts environnementaux.

Ces résultats confirment que l'utilisation combinée des images Sentinel-1 et des méthodes de classification avancées constitue une solution fiable pour le suivi post-incendie, offrant ainsi une base précieuse pour la gestion durable des ressources forestières et la planification des actions de restauration.

Conclusion Générale

Ce mémoire a mis en évidence l'efficacité de l'imagerie radar Sentinel-1 pour la télédétection et l'analyse de l'évolution de la couverture terrestre, en particulier pour la surveillance des zones forestières après un incendie.

Dans un premier temps, nous avons rappelé les concepts théoriques relatifs aux images SAR et présenté les spécificités techniques du satellite Sentinel-1 qui permettent une acquisition continue des données, indépendamment des conditions météorologiques. Nous avons ensuite comparé plusieurs méthodes de classification adaptées aux images radar, en soulignant l'intérêt des approches supervisées comme Random Forest pour une classification précise et fiable.

L'application pratique réalisée sur la wilaya d'El Tarf a permis de démontrer concrètement l'apport de ces outils. Grâce à la méthode Random Forest, nous avons pu générer des cartes de classification détaillées pour deux dates clés, avant et après l'incendie. L'analyse des résultats a révélé une perte significative de la surface forestière, ainsi qu'un retrait de la végétation dense remplacée par des zones de végétation clairsemée et de sol nu. Ces changements ont été cartographiés et quantifiés avec une grande précision grâce au traitement et à l'affichage des données sous Python.

Les résultats obtenus illustrent parfaitement la capacité des images SAR à détecter et suivre les modifications de l'environnement dans des régions vulnérables aux incendies. Ils confirment aussi la robustesse de la chaîne de traitement mise en place, qui pourrait être utilisée pour un suivi régulier et automatisé des forêts, facilitant ainsi la planification des actions de restauration et la gestion durable des ressources naturelles.

En conclusion, les résultats atteints prouvent que la télédétection radar, associée à des méthodes de classification avancées, représente une solution efficace pour la surveillance environnementale post-incendie, apportant une contribution précieuse à la préservation et à la gestion raisonnée des écosystèmes forestiers.

Bibliographie

- [1] I. T. Trablesi and M. N. Berrabah, “Extraction des zones urbaines avec les images radar à synthèse d’ouverture (sar),” Master’s thesis, Université Saad Dahlab Blida 1, Institut d’Aéronautique et des Études Spatiales, 2023.
- [2] “History of synthetic-aperture radar,” Mar. 2025.
- [3] K. Benaouicha, “Géocodage et correction géométrique des images radar SAR SLC,” magister thesis, Faculté d’Électronique et d’Informatique, Université Houari Bou-médiène, Alger, Algérie, 2008. Soutenu publiquement le 18 mai 2008.
- [4] copernicus.eu, “S1 mission.”
- [5] ATG-Europe, “Solar orbiter - atg medialab,” 2020.
- [6] G. Veyssière, *Apport des mesures du radar à synthèse d’ouverture de Sentinel-1 pour l’étude des propriétés du manteau neigeux*. PhD thesis, Institut National Polytechnique de Toulouse (INPT), 2019.
- [7] A. Martini, *Téledétection d’un couvert neigeux en milieux alpins à partir de données SAR polarimétriques multi-fréquentielles et multi-temporelles*. PhD thesis, Université Rennes 1, 2005.
- [8] J. Kellndorfer, “Using SAR data for mapping deforestation and forest degradation,” in *The SAR Handbook : Comprehensive Methodologies for Forest Monitoring and Biomass Estimation* (A. I. Flores-Anderson, K. E. Herndon, R. B. Thapa, and E. Cherington, eds.), NASA and SERVIR Global, 2019.
- [9] H. BELHADJER and B. SAROUER, “Classification des images avec les réseaux de neurones convolutionnels,” mémoire de master académique, Université Mouloud MAMMERI de Tizi-Ouzou, Tizi-Ouzou, Algérie, 2018. Dirigé par Madame FELLAG.
- [10] S. FRIZZI, R. KAABI, M. BOUCHOUICHA, J.-M. GINOUX, F. FNAIECH, and E. MOREAU, “Détection de la fumée et du feu par réseau de neurones convolutifs,” mémoire de recherche, Aix Marseille Université, Marseille, France, 2017. Projet réalisé en collaboration avec l’Université de Toulon et l’ENSIT Tunis.
- [11] S. Gambs, *Survol de l’apprentissage machine*. 2023.
- [12] A. Liaw and M. Wiener, “Classification and regression by randomforest,” *R News*, vol. 2, pp. 18–22, Dec 2002.
- [13] J. R. Quinlan, “Induction of decision trees,” *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
- [14] C. E. Shannon, “A mathematical theory of communication,” *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.
- [15] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

-
- [16] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
 - [17] J. Dupont, “Avantages et inconvénients des réseaux de neurones,” 2023. Consulté en avril 2025.
 - [18] M. A. C.-P. nán, “A review of mean-shift algorithms for clustering,” *arXiv preprint arXiv :1503.00687*, 2015.
 - [19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Mean shift clustering example.” https://scikit-learn.org/stable/auto_examples/cluster/plot_mean_shift.html, 2025. Accessed : 2025-05-05.
 - [20] AgroParisTech, E. Lebarbier, and T. Mary-Huard, “Classification non supervisée,” 2008.