



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

**UNIVERSITE IBN KHALDOUN - TIARET**

# MEMOIRE

Présenté à :

FACULTÉ MATHÉMATIQUES ET INFORMATIQUE  
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

**MASTER**

Spécialité : Génie informatique (GI)

Par :

**Ait Hammou Hamza  
Korichi Badreddine**

Sur le thème

---

## **Développements des mécanismes basés sur l'apprentissage automatique pour la gestion du trafic routier dans les intersections.**

---

Soutenu publiquement le 03 /10 / 2021 à Tiaret devant le jury composé de :

Mr ABID KHALED

Grade Université MAA

Président

Mr HATTAB NOUREDDINE

Grade Université MAA

Encadreur

Mr MOKHTARI AHMED

Grade Université MAA

Examineur

2020-2021

# *Remerciements*

Nous tenons tout d'abord à remercier Allah le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce modeste travail.

En second lieu, nous tenons à remercier notre encadreur : pour ses précieux conseils et son aide durant toute la période du travail.

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail et de l'enrichir par leurs propositions.

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

# *Dédicace*

*Je dédie ce travail à :*

*Ma très cher Père*

*Ma très chère mère*

*Mes frères et Mes amis*

*Tous ceux j'aime et ceux qui m'aiment*

*Hamza Ait Hammou*

# Dédicace

Je dédie ce travail a :

Ma très chère mère رَحْمَهَا اللهُ

Mon très cher Père

Mes frères abdelmalek , abderahman et rayen

Mes amis spécialement : Benazddine aek, Guerroudj Med amine, Brahim

Laabaoui,Rabeh Fodhile Lmaati Nadhir , Benouna Amel ,Dussama bentamra,

Chachou Asma , Zahoiani Bahae, Zokel Ramy

Et spécialement monsieur hettab noureddine sans lui je ne pourrais jamais finir

ça

Tout la famille korichi..

Tous ceux j'aime et ceux qui m'aiment

KORICHI BADREDDINE

## Résumé

De nos jours, l'intelligence artificielle (IA) joue un rôle important dans la résolution de nombreux problèmes du monde réel.

Dans ce travail nous nous intéressons au domaine de circulation routière, spécialement la gestion des intersections. La motivation vient du fait que ce secteur nécessite de nouvelles perceptions et mécanismes basés sur l'IA pour freiner l'augmentation des problèmes issue de ce secteur, des problèmes qui engendre des coûts économiques et écologiques en croissance exponentielle.

L'objectif ultime de ce mémoire est de faire une recherche bibliographique "état de l'art" sur le sujet, ensuite intégrer les approches d'apprentissage automatique "ML" dans le contrôle et l'automatisation des feux de signalisation pour une circulation routière basique, en tenant compte des contraintes environnementales de la circulation routière étudiée.

Afin de mesurer l'impacte de l'ensemble des mécanismes développés, Les cadres informatique suivants Anaconda, spyder, Netlogo, et langage python sont visés dans ce Mémoire.

### المخلص

في هذا العمل نحن مهتمون بمجال المرور على الطرق. الدافع يأتي من حقيقة أن هذا القطاع يتطلب تصورًا جديدًا ، مما الهدف النهائي (يجعل من الممكن إبطاء الزيادة في المشكلات الناشئة عن هذا المجال مثل حظر التقاطعات) مفترق الطرق من هذا العمل هو دمج مناهج التعلم الآلي في أتمته إشارات المرور في حركة المرور الأساسية، مع مراعاة القيود البيئية لمهمة " حركة المرور المدروسة «. أخيرًا، قارنا بين أربع استراتيجيات تعلم خاضعة للإشراف للحركة المرورية المدروسة

من أجل قياس تأثير جميع الآليات التي تم تطويرها، تمت الإشارة إلى أطر عمل تكنولوجيا المعلومات التالية Anaconda و spyder و Netlogo و python language في هذه المذكرة.

### abstract

In this work we are interested in the field of road traffic. The motivation comes from the fact that this sector requires new perceptions, making it possible to slow down the increase in problems arising from this field such as blocking at intersections (crossroads). The ultimate objective of this work is to integrate machine learning approaches in the automation of traffic lights in basic road traffic, taking into account the environmental constraints of the "traffic studied" task. Finally, we compared between three supervised learning strategies for our studied circulation task

In order to measure the impact of all the mechanisms developed, the following IT frameworks Anaconda, spyder, Netlogo, and python language are referred to in this Memoir.

# Sommaire

Remerciements

Dédicaces

Résumé

Abstract

Sommaire

Introduction générale

## Chapitre I : Aperçu Général sur Le trafic routière

Introduction .....	15
1.1 Le trafic routier .....	15
1.1.1 Les autoroutes .....	16
1.1.2 Les intersections .....	16
1.2. Modélisation du trafic .....	17
1.2.1 Les différents types de modèles microscopiques .....	17
1.2.1.1 Les modèles de poursuite .....	18
1.2.1.2 Les modèles de particules .....	18
1.2.2 Les caractéristiques des modèles macroscopiques .....	18
1.2.2.1 Les variables du modèle macroscopique .....	18
1.3. La problématique urbaine .....	19
1.3.1 Position du problème.....	19
1.3.2 Le recueil des données .....	20
1.3.2 Les causes des dysfonctionnements au carrefour.....	20
1.3.2.1 Cas des dysfonctionnements externes .....	20
1.3.2.2 Cas des dysfonctionnements internes .....	21
1.4. Présentation du carrefour à feux étudié .....	21
1.4.1 Configuration du carrefour étudié .....	21
1.4.2 Caractéristiques du carrefour à feux étudié .....	22
1.4.3 Notion de conflit .....	23
1.4.4 Découpage en phases .....	24
1.4.5 Paramètres nécessaires au calcul de la durée des feux .....	25

1.5. Les différentes stratégies de régulation via les feux de signalisation .....	26
1.5.1 Stratégie prédéterminée ou cyclique .....	27
1.5.2 Stratégie semi-adaptative ou acyclique .....	27
1.5.3 Stratégie adaptative ou temps réel.....	27
1.6. Gestion de feux de carrefours – Position du problème .....	27
Conclusion .....	28

## **Chapitre II : Aperçu Général sur L'apprentissage automatique**

1-Introduction .....	30
2-Introduction sur l'apprentissage automatique .....	30
3-les solutions pour la gestion des intersections basées sur l'apprentissage automatique,.....	31
3.1-Définition de l'apprentissage automatique.....	31
3.2-Types d'apprentissage automatique.....	31
3.2.1-Apprentissage supervisé (Supervised Learning).....	32
3.2.2-Apprentissage non-supervisé (Unsupervised Learning).....	32
3.2.3-Apprentissage semi-supervisé .....	33
3.2.4-Apprentissage par renforcement (Reinforcement Learning) .....	34
3.3.1-classification .....	35
3.3.2-Les algorithmes de la classification.....	36
3.3.2.1-Arbres de décisions.....	36
3.3.2.2-Algorithmes k plus proches voisins (KNN) .....	38
3.3.2.3-Machine à Vecteur de Support (SVM) .....	41
3.3.2.4-Réseau Bayésien (Naïve bayes) .....	42
3.3.2.5Foret aléatoire (Random Forest) .....	43
3.4-métriques pour évaluer les algorithmes d'apprentissage automatique .....	46
3.4.1-Accuracy.....	46
3.4.3-Recall.....	46
3.4.4-F1Score.....	47
3.4.5-Matrice de confusion .....	47
4-Conclusion .....	48

## Chapitre III : Implémentation Et Evaluation

1. Introduction.....	50
2. Présentation de notre approche .....	50
3. Traitement de data set.....	51
3.1 Traitement les données avec Foda (Feature-Oriented Domain Analysis).....	51
3.2 Feature-Oriented Domain Analysis (Foda).....	52
3.2.1 Modélisation de la variabilité.....	52
4. Environnement de développement et Les outils techniques .....	54
4.1 Dentition du langage Python en informatique.....	54
4.2 Anaconda.....	54
4.3 Jupyter.....	55
4.4 Les bibliothèques utilisées.....	55
4.4.1 Numpy.....	55
4.4.2 Matplotlib.....	55
4.4.3 Pandas .....	56
4.4.4 Scikit-learn .....	56
5. 5. Apprentissage et Paramétrage des Modèles.....	56
5.1 Préparation du Dataset .....	56
5.2 Application de PCA.....	57
5.3 Algorithme LDA.....	58
6. Apprentissage du modèle.....	59
6.1 Algorithme de Arbre de décision.....	59
6.1.1 Explication du code d'algorithme Arbre de décision.....	59
6.2 Algorithme de Radom Forest .....	61
6.2.1 Explication du code d'algorithme Random forest .....	61
6.3 Algorithme de SVM .....	63
6.3.1 Explication du code d'algorithme SVM.....	63
6.4 Algorithme de KNN .....	65
7. Comparaison entre Les quatre algorithmes .....	67
8. Conclusion.....	70
Conclusion générale	
Bibliographie .	

## Liste des figures

Figure 1 : Intersection à quatre branches avec des voies différentes .....	16
Figure 2 : Configuration du carrefour étudié .....	21
Figure 3:Zone fonctionnelle d'un carrefour simple de deux routes à sens unique .....	22
Figure 4 : Exemples de conflits dans un carrefour.....	24
Figure 5: Structure du carrefour à 4 phases.....	25
Figure 6: Schéma général du Machine Learning .....	31
Figure 7 : Schéma général du Machine Learning.....	32
Figure 8 : Processus général de l'apprentissage non-supervisé.....	33
Figure 9 : Différentes étapes d'un processus semi-supervisé.....	34
Figure 10 : Processus d'apprentissage par renforcement.....	35
Figure 11 : Les types d'apprentissage automatique .....	35
Figure 12 : L'arbre de décision .....	37
Figure 13 : méthode de k plus proches voisins.....	39
Figure 14 : hyperplan optimal, vecteurs de support et marge maximale.....	41
Figure 15 :l'algorithme de foret aléatoire.....	44
Figure 16 : Présentation de notre approche .....	50
Figure 17 : Simulation Net Logo .....	51
Figure 18 : Les caractéristiques de la circulation routière .....	52
Figure 19 : Les caractéristiques de la route.....	53
Figure 20 : Les caractéristiques de la Véhicule.....	53
Figure 21 : Les caractéristiques des personne .....	54
Figure 22 : Les caractéristiques du feu de signalisation.....	54
Figure 23 : L'environnement Anaconda .....	55
Figure 24 : L'exécution de l'algorithme PCA.....	58
Figure 25 : L'exécution de l'algorithme LDA.....	58
Figure 26 : Arbre de décision organigramme .....	59
Figure 27 : Matrice de confusion de l'arbre de décision.....	60
Figure 28 : Organigramme d'algorithme Radom Forest .....	61
Figure 29 : Matrice de confusion de Foret Aléatoire .....	62
Figure 30 : classification des données avec SVM [33] .....	63
Figure 31 : Matrice de confusion de SVM .....	64
Figure 32 : classification des données avec KNN [34].....	65
Figure 33 : Matrice de confusion de KNN .....	66
Figure 34 : Comparaison entre les quatre algorithmes par apport Accuracy .....	67
Figure 35: Comparaison entre les quatre algorithmes par apport précision .....	69
Figure 36: Comparaison entre les quatre algorithmes par apport recall.....	69
Figure 37: Comparaison entre les quatre algorithmes par apport score.....	70

## Liste des tableaux

**Tableau 01** : les avantages et inconvénient de chaqu'un des algorithmes

**Tableau 02** : Comparaison entre les trois stratégies par apport Accuracy, Score

## Abréviations

**IA** : Intelligence Artificiel.

**AA** : Apprentissage Automatique.

**ML** : Machine Learning.

**LDA**: L'analyse discriminante linéaire

**FODA** : Feature-Oriented Domain Analysis.

**SVM** : Machine à Vecteurs de Support.

**NB** : Naïve Bayes.

**RF** : Random Forest.

**KNN** : k Nearest Neighbour.

**SPL** : Software using Product Lines

# Introduction générale

# Introduction générale

## 1. Contexte :

Le contexte du trafic routier suscite un vif intérêt scientifique depuis plus d'une cinquantaine d'années. Il est possible de décrire très communément le trafic routier comme étant le processus de transport depuis une origine et vers une destination par le moyen des infrastructures routières. Parmi ces infrastructures, nous citons les feux de signalisation comme le moyen de régularisation et de gestion le plus ancien. Le contexte général de ce mémoire est le trafic routier et le contexte spécifique c'est les feux de signalisation.

## 2. Problématique :

La spécificité du trafic de la circulation automobile est que l'infrastructure routière, en fait, conçue selon des normes évolutives, alors que chaque individu s'efforce d'améliorer son parcours tout en respectant les normes fixées à un niveau individuel.

Cette particularité fait du trafic automobile un phénomène difficile à analyser et à optimiser, ce phénomène possède un certain nombre de caractéristiques qui le rendent également difficile à modéliser et génère des problèmes au niveau de la fluidité du trafic pouvant être à l'origine de l'apparition de phénomènes de congestion et de formation des files d'attente avec les retards correspondants. Les retards causés par les congestions ont un impact important sur la qualité de vie des usagers.

Ce phénomène a aussi des conséquences directes sur l'augmentation de la consommation d'énergie. Le phénomène de congestion du trafic routier est ainsi un problème socio-économique crucial qui nécessite de rechercher des solutions adaptées pouvant être mises en œuvre en pratique.

Les programmes d'extension des infrastructures étant techniquement et juridiquement coûteux à mettre en œuvre, une utilisation optimale des infrastructures existantes, tant en termes de capacité qu'en termes de sécurité, est indispensable, ce qui ne peut être obtenu que par une meilleure compréhension des mécanismes qui régissent l'ensemble des phénomènes de trafic. Cependant, le contrôle du trafic, qui présente encore de nombreuses difficultés en raison de la complexité du problème à traiter, le contrôle intelligent du trafic peut être un moyen efficace d'éliminer ou au moins d'atténuer les effets de la congestion et de la formation de files d'attente.

Le moyen classique de prévenir ou de réduire ces impacts dans les villes modernes est de placer des panneaux de signalisation appropriés, grâce à l'utilisation de lumières colorées (rouge-orange/jaune-vert) qui peuvent aider à résoudre les conflits entre les différents flux de circulation aux carrefours. Le contrôle de la signalisation du trafic au niveau des carrefours se divise, généralement, en deux catégories : la stratégie de contrôle à plan de feux fixe, c'est-à-dire avec un cycle fixé, et la stratégie de contrôle adaptative, permettant des changements de durées de phases en fonction de la demande du trafic. Les deux stratégies sont généralement basées sur l'estimation du débit du trafic.

Schématiquement, un système de contrôle dynamique du trafic est un système qui contient des capteurs qui lui fournissent le nombre de véhicules sur certaines rues ou intersections du réseau routier. Cette information est ensuite envoyée à un ordinateur dont le rôle est de donner aux contrôles des valeurs de paramètres qui minimisent le trafic. Une fonction spécifique exprime la congestion du réseau routier, comme la longueur des files d'attente, le temps d'attente moyen et le temps moyen. Temps d'attente le plus long dans la file d'attente et temps d'évacuation total.

### **3. Objectif de travail :**

Dans ce mémoire, nous nous intéressons au contrôle des feux d'intersection afin de résoudre les problèmes liés à la congestion ; En ce sens, nous proposons une stratégie de contrôle des feux de circulation en utilisant l'apprentissage automatique ML. Le modèle de prédiction est basé sur quatre algorithmes de ML : Arbre de décision, système support vecteur machine (SVM), Forêt aléatoire RF et KNN. Le modèle obtenu permet de prédire la décision afin d'agir dans la circulation en fonction de plusieurs paramètres comme la rue, la voiture, les passages etc.

### **4. Organisation du travail :**

Ce mémoire comporte trois chapitres :

- Le premier chapitre : Aperçu Général sur Le trafic routier.
- Le second chapitre : Aperçu Général sur L'apprentissage automatique.
- Le troisième chapitre : Implémentation Et Evaluation.

# Chapitre I :

**Aperçu Général sur Le trafic routier**

# Chapitre I : Aperçu Général sur Le trafic routier

## Introduction

Les systèmes de transport ont toujours joué un rôle primordial dans le développement d'un pays; mais, depuis la seconde moitié du 20<sup>ème</sup> siècle, le phénomène de la congestion routière, en particulier, a induit, pendant les heures de pointe, un grand problème dû principalement, à l'accroissement rapide du nombre de véhicules et de la demande en transport.

Il constitue, en effet, un problème crucial pour la société, à cause des coûts qu'il engendre.

Généralement, la congestion du trafic apparaît lorsqu'un nombre assez important de véhicules utilise une infrastructure routière commune, caractérisée par une capacité limitée.

Pour réduire ou minimiser cette congestion, une commande adéquate du trafic devient une nécessité.

L'intérêt de contrôler les feux de circulation n'est plus à démontrer ; les feux tricolores permettent, en effet, de supprimer certains points conflictuels entre les différents flux de circulation aux carrefours tout en partageant, dans le temps, l'utilisation d'un même espace.

Ceci peut se réaliser par le choix des durées de chaque état des feux.

Cette approche, applicable à court et à moyen termes, consiste à mettre en œuvre des actions et des mesures afin d'assurer une circulation sûre et fluide du trafic et à augmenter, ainsi, la qualité de service aux usagers.

Dans ce chapitre, sont présentés, d'une manière détaillée, un carrefour à feux isolé, les différents éléments le caractérisant ainsi que des méthodes de commande classiquement utilisées pour sa gestion.

## 1. Le trafic routier

Le trafic routier se définit comme l'ensemble des phénomènes complexes qui résulte du déplacement d'usagers sur un réseau routier à capacité limitée [1].

Le trafic routier peut être défini comme étant le résultat des déplacements des véhicules sur le réseau routier. Ce phénomène résulte, en fait, de l'interaction de chaque usager de la route avec la réglementation ainsi qu'avec son environnement, constitué par l'infrastructure routière (dimensionnement géométrique de la route) et les autres usagers. Le trafic routier est, donc, un phénomène naturellement distribué au sein duquel les interactions entre les différents participants constituent le cœur du fonctionnement.

On a deux catégories de voies dont la différenciation est nécessaire dans le traitement des carrefours : les voies rapides (les autoroutes) et les voies de quartier (les intersections).

# Chapitre I : Aperçu Général sur Le trafic routier

Cette catégorisation est analysée sous des angles multiples : la longueur, la capacité (débit maximum admissible ou débit de saturation) et le mode d'exploitation.

## 1.1 Les autoroutes

Grâce à leurs topologies (multivoies, grandes lignes), les autoroutes offrent une bonne qualité de service. En effet, elles laissent circuler un grand nombre de véhicules, les vitesses des véhicules étant peu sensibles au changement de l'état du trafic [2].

Sur les routes urbaines, les vitesses de véhicules dépendent, par contre, de la densité du trafic.

## 1.2 Les intersections

Une intersection est située à la rencontre de plusieurs rues, déterminant des couloirs d'entrée et de sortie. Un couloir est caractérisé par sa largeur et le nombre de ses voies; certaines de ces voies (voies spéciales) peuvent être affectées par un flux particulier. Les courants des véhicules sont soit des courants directs, soit des courants de tourne à gauche, soit des courants de tourne à droite, figure 1. La mise en place d'un système de feux à un carrefour réalise une séparation dans le temps de l'admission de différents flux de véhicules



Figure 1 : Intersection à quatre branches avec des voies différentes

(Photo prise par José Navarro)

# Chapitre I : Aperçu Général sur Le trafic routier

## 2. Modélisation du trafic

Le développement rapide de la technologie automobile et l'augmentation énorme du trafic routier ont rendu nécessaire l'étude du trafic d'une manière mathématique précise, à travers des modèles capables de décrire sa dynamique dans l'espace et dans le temps.

Les chercheurs, dans les années 1950, ont réalisé un développement considérable de la théorie du trafic par l'entremise de plusieurs approches telles que :

- le véhicule suiveur [3].
- l'hydrodynamique [4].
- et les files d'attente [5].

Compte tenu de leur multiplicité, les modèles du trafic ont été classés en deux catégories :

- les modèles microscopiques
- les modèles macroscopiques.

La régulation du trafic, utilisant un certain nombre de variables de flux et de relations caractéristiques, a permis de mieux comprendre et prévoir les interactions inter-véhiculaires ainsi que le mouvement global du flux au niveau de ce système. Les variables de flux ont été classées en deux familles :

- les variables microscopiques qui sont inhérentes aux véhicules, considérés individuellement,
- les variables macroscopiques qui caractérisent le trafic en tant qu'entité globale et contextuelle [6].

### 2.1 Les différents types de modèles microscopiques

Les modèles microscopiques reposent sur une description du comportement individuel des véhicules, qui lorsqu'elle est fine, est réalisée en se basant sur les interactions entre les véhicules.

Ces modèles tentent de reproduire la façon avec laquelle réagit un véhicule en fonction de son environnement, c'est-à-dire l'infrastructure sur laquelle il circule et les véhicules qui l'entourent (distance avec les véhicules voisins, vitesse, accélération, ...).

En mentionne deux grandes classes à la modélisation microscopique :

- La première est celle caractérisant les modèles véhicule-suiveur [3].
- La seconde est basée sur les modèles de particules [7].

# Chapitre I : Aperçu Général sur Le trafic routier

## 2.1.1 Les modèles de poursuite

Les modèles de poursuite, appelés aussi modèles véhicule-suiveur, décrivent le comportement de tout véhicule par rapport au véhicule qui le précède.

Dans ces modèles, le conducteur joue un rôle primordial, en tant qu'élément de commande actif au sein de l'ensemble {véhicule, conducteur, route}, par des actions du conducteur sur le frein et sur l'accélérateur. Ces actions sont régies par une équation différentielle décrivant l'accélération du véhicule considéré, en respectant le comportement du véhicule qui le précède.

Les différents modèles véhicule-suiveur, existants, sont fondés sur le principe d'action réaction. [8]

Celui-ci se résume sur le fait qu'un conducteur va percevant un stimulus à l'instant  $t$  et réagit avec un certain décalage temporel, correspondant à son temps de réaction, noté  $\tau$ . La réponse est considérée proportionnelle à l'intensité du stimulus, mais dépend de la sensibilité du conducteur :

$$\text{Réaction}(t+\tau)=\text{sensibilité}\times\text{stimulus}(t)$$

Les modèles de poursuite considèrent que l'espace inter-véhiculaire est constant pour deux véhicules se déplaçant à la même vitesse.

## 2.1.2 Les modèles de particules

Son utilisation consiste à considérer les véhicules comme des particules interagissant entre elles. Le comportement des véhicules dans le réseau est décrit suivant leur déplacement dans la route découpée en un nombre fini de sites ou de cellules. La combinaison de tous les comportements des véhicules fournit une image globale des situations du trafic dans le réseau.

## 2.2 Les caractéristiques des modèles macroscopiques

Les variables macroscopiques du trafic permettent de décrire globalement le flot des véhicules sur une section de route et ignorent le comportement individuel des véhicules. Plus précisément, elles établissent des relations entre le débit du flux en nombre de véhicules par heure, sa densité en nombre de véhicules par kilomètre et sa vitesse moyenne en kilomètres par heure.

### 2.2.1 Les variables du modèle macroscopique

Le niveau macroscopique nécessite la définition d'un certain nombre de paramètres caractérisant le trafic dont la densité, le débit, la vitesse moyenne.

## Chapitre I : Aperçu Général sur Le trafic routier

### a. La densité du trafic

La densité caractérise la répartition des véhicules dans l'espace. La densité moyenne  $k(x_1, x_2, t)$  à un instant  $t$ , sur une section de route limitée par les points d'abscisses  $x_1$  et  $x_2$

### b. Le débit du trafic

Le débit  $q$ , correspondant à la répartition des véhicules dans le temps, entre les instants  $t_1$  et  $t_2$  et au point  $x$  de la route

### c. La vitesse moyenne

La mécanique des fluides nous enseigne qu'un débit  $q$  est égal au produit de la concentration  $k$  par la vitesse du flot  $v$  :  $q = k v$ . La vitesse moyenne pour un flot d'écoulement du trafic à l'instant  $t$  et au point  $x$

## 3. La problématique urbaine

### 3.1 Position du problème

La création d'un carrefour en milieu urbain ne doit pas se limiter à une réflexion ponctuelle à la fois dans l'espace et dans le temps. En effet, la problématique est plus complexe que celle du milieu interurbain où le bon fonctionnement d'un carrefour suppose :

- Qu'il soit adapté au type de route sur lequel il se situe
- Qu'il soit adapté au site et aux conditions d'utilisation
- Que sa configuration générale et sa conception de détails soient correctes.

Il est essentiel de prendre en compte le contexte environnant à l'échelle du quartier dans lequel se trouve le carrefour, voire au niveau de la ville ou de l'agglomération. Le fonctionnement d'un carrefour est aussi conditionné par le type :

- De la structure et du fonctionnement urbains
- Du maillage du réseau de voirie.

La différence avec le milieu interurbain est due à une répartition différente des types d'usagers (beaucoup plus de piétons, des deux-roues et des modes de transport en commun). De même, la typologie et la hiérarchie de la voie sur laquelle se trouve le carrefour lui confèrent une fonction particulière.

Ainsi, la prise en compte des politiques globales d'urbanisme, de déplacements et de projets d'infrastructures permet de mieux comprendre le fonctionnement d'un carrefour et de son évolution dans le temps.

# Chapitre I : Aperçu Général sur Le trafic routier

## 3.2 Le recueil des données

La conception d'un carrefour en milieu urbain exige de recueillir un nombre important de données du fait de la complexité du système. Les problématiques aux échelles du quartier, du réseau de voirie, du nœud de communication se superposent et conditionnent fortement les dysfonctionnements au niveau du carrefour.

Il existe différents domaines d'investigation pour la conception d'un carrefour.

Le premier est relatif à la vie locale ; ce domaine concerne le fonctionnement urbain (organisation du réseau de voirie et structure urbaine) et le fonctionnement des usagers (piétons, deux-roues, transports en commun, voitures et poids lourds).

Le second, relatif à l'approche fonctionnelle, s'intéresse au trafic et son déplacement, aux nuisances générées par la circulation automobile (bruit, pollution) et à la sécurité.

Le dernier est relatif à l'approche spatiale qui rassemble le site et son espace (morphologie et densité)

Le choix et le niveau d'investigation de ces données ne peuvent pas être prédéfinis.

L'analyse des approches et les enjeux du carrefour permettent de sélectionner et de doser les différentes recherches d'informations.

## 3.2 Les causes des dysfonctionnements au carrefour

Les dysfonctionnements sont de deux natures : externes et internes.

### 3.2.1 Cas des dysfonctionnements externes

L'étude d'un carrefour ne se limite pas à l'emprise même de l'intersection. Il est important d'élargir le champ d'investigations au fonctionnement des voies qui s'y croisent. Il faut, pour cela, prendre en compte les carrefours situés en amont et en aval ainsi que les conditions générales de la circulation dans le quartier, ou du moins dans la zone d'influence du carrefour, comme :

- Le mauvais fonctionnement des carrefours alentours, relatif aux problèmes de remontées de file d'attente d'un carrefour sur l'autre et à la mauvaise coordination entre les feux.
- Les mouvements perturbateurs systématiques, relatifs aux manœuvres fréquentes de poids lourds à proximité immédiate du carrefour (livraisons) et aux arrêts de courte durée de véhicules sur la chaussée (présence d'un commerce de proximité tel qu'un bureau de tabac, ou d'un distributeur de billets).
- Le stationnement gênant, relatif aux manœuvres fréquentes de stationnement sur une des voies (ou sur un arrêt de bus ou sur une voie de bus) générant des files d'attente pouvant remonter jusqu'au carrefour.

## Chapitre I : Aperçu Général sur Le trafic routier

- Les arrêts de bus mal positionnés ; en effet, un arrêt situé en aval d'un carrefour perturbe souvent moins la circulation générale qu'un arrêt placé en amont. Cette mesure permet également de positionner le passage piéton à l'arrière de l'arrêt de bus, au niveau du carrefour ;

### 3.2.2 Cas des dysfonctionnements internes

Ils sont relatifs aux paramètres dont l'influence peut être mise en évidence par un calcul de capacité théorique à partir de données de trafic comme :

- La longueur de stockage inadaptée au niveau de trafic des tourne-à-gauche,
- Le phasage et les plans de feux,
- La capacité de stockage à l'intérieur du carrefour.

## 4. Présentation du carrefour à feux étudié

### 4.1 Configuration du carrefour étudié

La configuration étudiée est un carrefour constitué :

- D'une intersection de quatre routes principales, avec deux choix de direction : le premier est formé avec des mouvements de tourner à droite et d'aller tout droit (voies 2, 5, 8, 11), le deuxième est celui des mouvements de tourner à gauche (voies 1, 4, 7, 10)
- Et quatre sorties de ce carrefour (voies 3, 6, 9, 12).

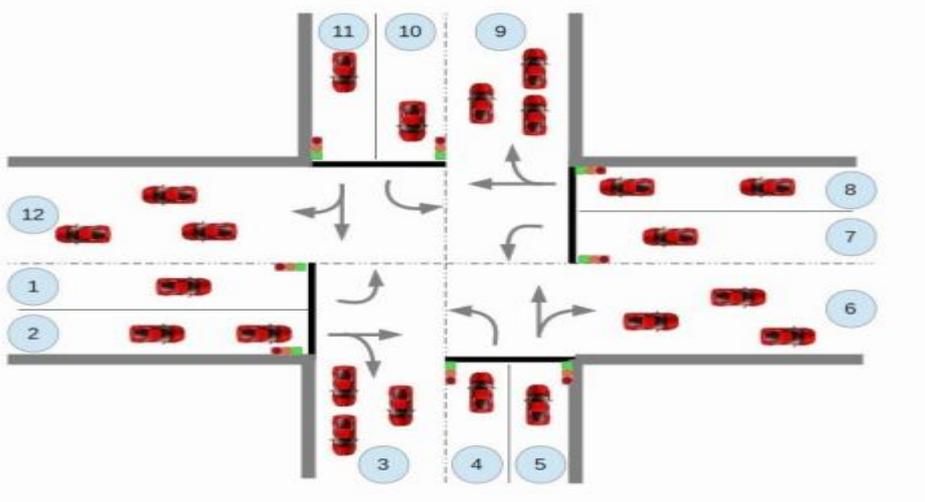


Figure 2 : Configuration du carrefour étudié

# Chapitre I : Aperçu Général sur Le trafic routier

## 4.2 Caractéristiques du carrefour à feux étudié

Les carrefours sont les lieux de croisement d'au moins deux routes où un flux de véhicules venant d'une même origine se divise en deux ou plusieurs flux vers plusieurs destinations suivant des coefficients appelés taux des mouvements tournants. Un tel croisement peut revêtir de nombreuses formes. Une route est caractérisée par sa longueur et son nombre de voies ainsi que le sens de la circulation et la vitesse maximale autorisée sur les voies.

La figure 3 en montre les zones fonctionnelles.

Dans chaque intersection, nous pouvons identifier trois zones fonctionnelles :

- Une zone de stockage, placée : en amont de la zone de conflit, constitue l'entrée empruntée par les véhicules ;
- Une zone de conflit, relative à l'espace de croisement des routes ; i.e. la ressource critique partagée par tous les véhicules qui traversent l'intersection ;
- Une zone de sortie, placée en aval de la zone de conflit, permet le soulagement de cette zone.

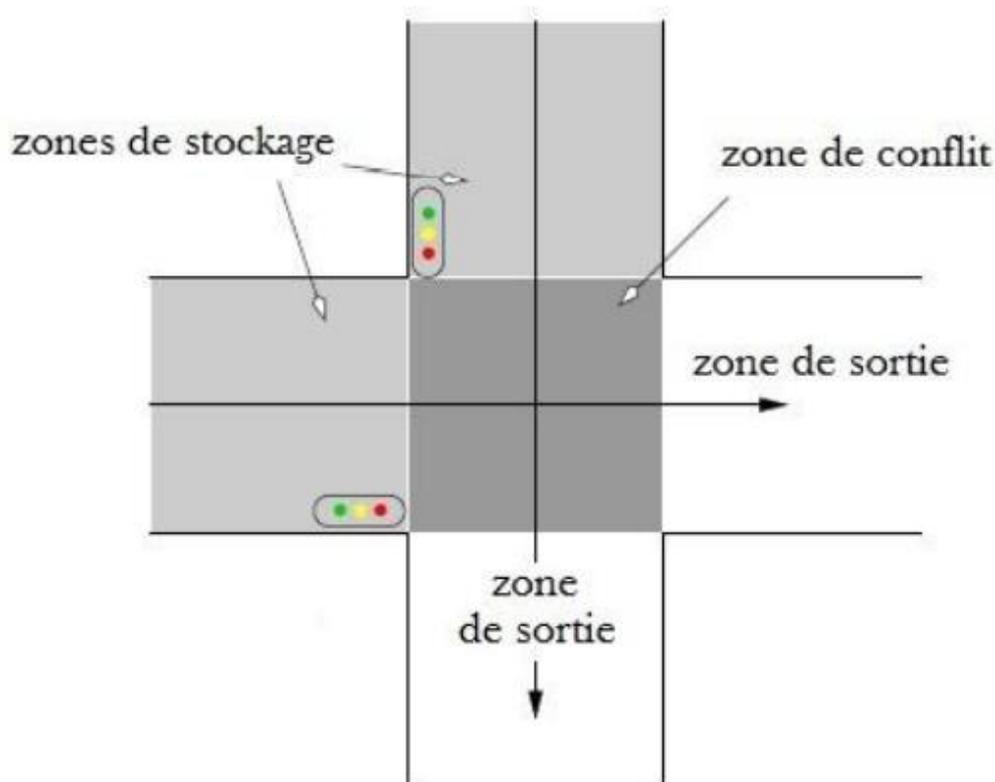


Figure 3: Zone fonctionnelle d'un carrefour simple de deux routes à sens unique

## Chapitre I : Aperçu Général sur Le trafic routier

Un vocabulaire technique commun, désignant le fonctionnement de la signalisation tricolore, est également utilisé par l'ensemble de la communauté des ingénieurs concernés par le domaine :

- L'état des feux : le signal lumineux commandant le passage libre (feu vert), toléré (feu orange) ou interdit (feu rouge) du trafic des véhicules ;
- Une phase : la durée pendant laquelle un feu passe par les trois états, c'est-à-dire effectue une séquence de signal lumineux ;
- Un cycle : la période pendant laquelle l'ensemble des feux effectuent leur phase permettant successivement l'admission de tous les courants de véhicules dans ce carrefour ;
- Un plan de feux : la durée et l'ordre de déroulement de l'allumage des trois états, pendant un cycle, de l'ensemble des groupes de feux ; un même carrefour peut suivre alternativement plusieurs plans de feux.

### 4.3 Notion de conflit

Il est nécessaire d'étudier le comportement de l'écoulement du trafic. En général, les flux de mouvements dans le carrefour sont séparés en groupes, recevant simultanément des signaux identiques. La zone de conflit d'un carrefour est constituée de l'ensemble de ses points conflictuels.

On peut distinguer trois types de conflits :

- Un conflit primaire : issu du croisement de deux mouvements sécants ; pour des raisons de sécurité, il est nécessaire de n'autoriser l'accès à un tel conflit qu'à un seul des mouvements concernés à la fois (points A et C).
- Un conflit secondaire : issu de la rencontre des mouvements conflictuels seulement gênants : tourne à gauche ou tourne à droite rencontrant des piétons. Ce type de conflit est résolu par les règles du code de la route et pour des raisons de sécurité, il arrive que ces mouvements ne soient pas autorisés à accéder la zone de conflit simultanément (points B et D).
- Un conflit tertiaire : dû à la géométrie d'un tronçon acceptant plusieurs modes, comme une voie étroite à double sens, ou, encore, une voie d'insertion trop courte (point E).

## Chapitre I : Aperçu Général sur Le trafic routier

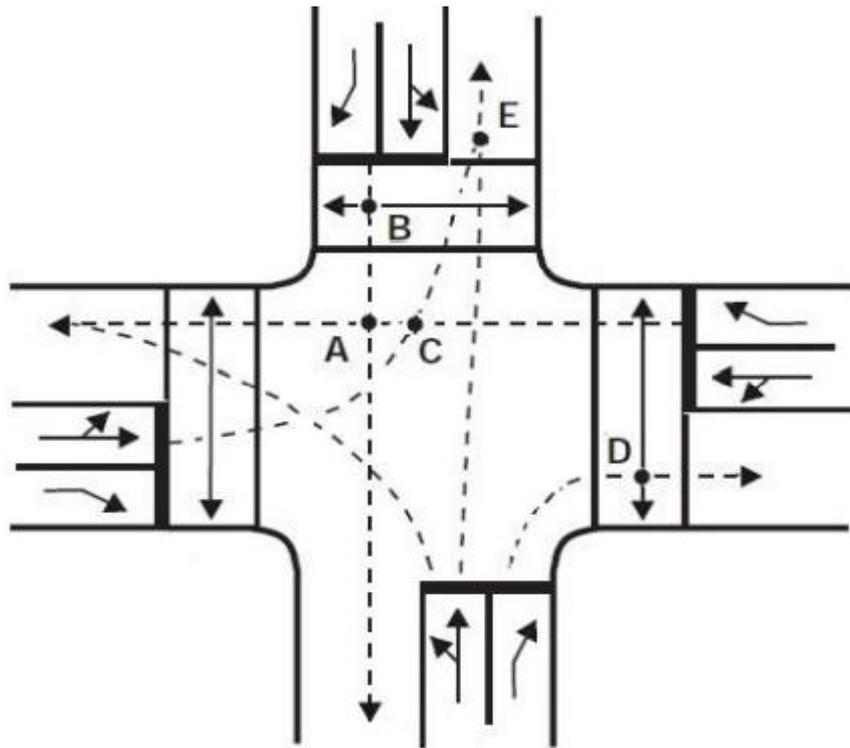


Figure 4 : Exemples de conflits dans un carrefour

### 4.4 Découpage en phases

Le découpage en phases, ou phasage, est un élément essentiel qui conditionne le bon fonctionnement d'un carrefour. Il doit :

- Être optimisé en fonction des emprises disponibles et de la géométrie du carrefour, après une analyse aussi exhaustive que possible des différents mouvements,
- Et faire apparaître toutes les phases du trafic, en particulier tourner à gauche.

Afin d'éviter les conflits entre les différents flux, sont utilisées, fréquemment, les notions de flux compatibles et de flux incompatibles. Évidemment, quand les trajectoires de deux flux de trafic ne se croisent pas, ces flux peuvent obtenir l'autorisation de traverser le carrefour simultanément ; nous qualifions ces deux flux de flux compatibles. Par exemple, dans la figure 5 , le flux 1 et le flux 7 constituent deux flux compatibles. A l'inverse, lorsque les trajectoires des deux flux se croisent, les flux sont en conflit et par conséquent incompatibles (par exemple, les flux 1 et 8), et leur mouvement simultané ne doit donc pas être autorisé.

## Chapitre I : Aperçu Général sur Le trafic routier

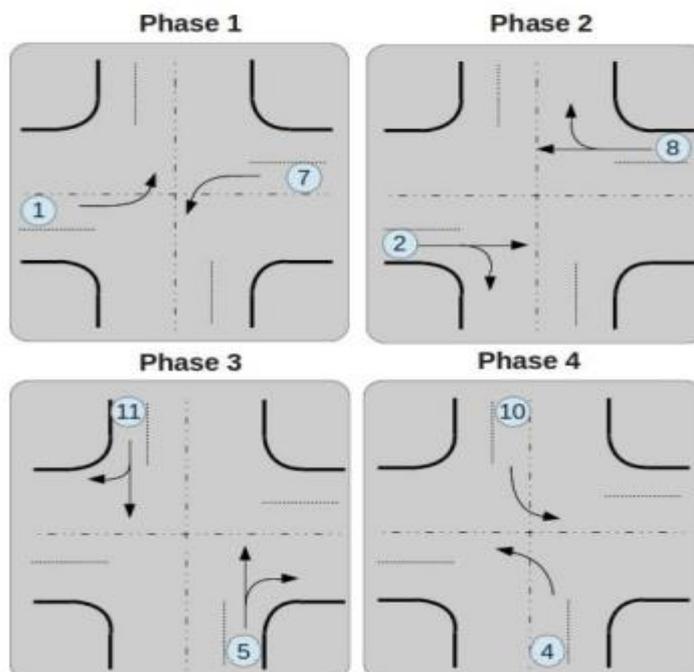


Figure 5: Structure du carrefour à 4 phases

### 4.5 Paramètres nécessaires au calcul de la durée des feux

Cependant, quelle que soit la méthode de commande appliquée à l'intersection, les indications des feux (vert, orange et rouge) doivent se succéder à l'intérieur d'un cycle défini comme étant la durée fixe ou variable, séparant deux passages successifs de l'ensemble des signaux par le même état [6].

Définitions :

- Vert effectif : le temps de vert effectif est la somme du temps de vert réel et du temps de l'orange qui est souvent de 3 ou 5 secondes selon la vitesse d'approche dans le cas d'un carrefour ordinaire ;
- Vert utile : le temps de vert utile est la différence entre le temps de vert effectif et celui du temps perdu ;
- Temps perdu : la somme du temps perdu au début du vert et celui en fin de phase ;
- Rouge utile : obtenu en retranchant, de la durée du cycle, le temps du vert utile ;
- Rouge intégral : la durée pendant laquelle aucun véhicule n'est admis dans le carrefour, C'est-à-dire lorsque l'état est rouge sur chaque feu.

Dans un carrefour à feux, la signalisation tricolore peut être régie de trois manières différentes [9]:

## Chapitre I : Aperçu Général sur Le trafic routier

- la gestion 1 : un feu est vert et les autres sont rouges. Le changement d'état des feux est régulé par un temporisateur qui applique une durée de vert identique pour chaque voie. Ainsi, pour chaque voie, le feu est vert pendant un quart de cycle ;
- la gestion 2 : deux feux opposés (Nord-Sud ou Est-Ouest) sont verts et les deux autres sont rouges. Le changement d'état des feux est régulé par un temporisateur qui applique une durée de vert identique pour chaque binôme de voies. Ainsi, pour chaque voie, le feu est vert pendant la moitié du cycle ;
- la gestion 3 : deux feux opposés (Nord-Sud ou Est-Ouest) sont verts et les deux autres sont rouges. Le changement d'état des feux est régulé selon la circulation.

### 5. Les différentes stratégies de régulation via les feux de signalisation

Un plan de feu d'un carrefour est déterminé par la spécification de quatre variables : la longueur du cycle, le plan de phase, la durée de chaque phase et enfin le décalage entre les phases.

Les stratégies de régulation en temps réel s'appuient généralement sur trois modules [10].

Le premier module est un module de collection et de traitement des mesures des variables du trafic. Plusieurs types de capteurs peuvent être utilisés pour mesurer les variables du trafic comme, par exemple, les boucles magnétiques qui permettent de mesurer la densité, le flux et le nombre de véhicules dans un tronçon de route ; les caméras vidéo peuvent, en plus, évaluer la longueur de la file d'attente devant un feu de signalisation ; etc.

Le deuxième est généralement un module de prédiction qui s'appuie sur un modèle de trafic plus ou moins sophistiqué. À partir des variables mesurées, ce module prédit les valeurs futures de ces mêmes ou / et d'autres variables.

Le troisième module d'optimisation détermine enfin, les paramètres optimaux des plans de feux qui permettent de minimiser le temps de parcours d'un flux donné, les temps d'attente aux carrefours ou le nombre de voitures dans le réseau.

Chaque stratégie a ses propres variables mesurées, son propre module de prédiction et son propre module d'optimisation. Les stratégies diffèrent aussi par les approches de résolution exploitées (programmation dynamique, heuristiques, réseaux de neurones, etc)

## **Chapitre I : Aperçu Général sur Le trafic routier**

### **5.1 Stratégie prédéterminée ou cyclique**

La première stratégie considère les cycles comme fixes pendant une période donnée. Ceci n'empêche pas que les durées des cycles peuvent changer d'une période à une autre.

Le rôle principal des plans de feux fixes est de mettre en marche cycliquement une série de phases pendant des durées données. Cette série de phases doit permettre de mieux répondre à la demande moyenne de la capacité estimée du carrefour. Dès le début des années 1980, deux systèmes de gestion du trafic ont émergé.

Il s'agit de SCOOT (Split Cycle Offset Optimisation Technique) et SCATS (Sydney Coordinated Adaptive Traffic System).

### **5.2 Stratégie semi-adaptative ou acyclique**

Cette famille de stratégies de régulation ne considère pas explicitement la notion de cycles, de décalages ou de durées des phases de vert. Les successions de phases de vert et de rouge sont perçues comme des instants de commutation. La durée du cycle ainsi que les durées du feu vert peuvent varier d'un cycle à un autre en fonction de la demande. L'algorithme, dit CRONOS (ContROl Of Networks by Optimization of Switchovers) [11], entrant dans le cadre de la micro-régulation se basant essentiellement sur les plans de feux fixes, ajuste les durées de commutation d'un cycle à un autre, en fonction de la demande enregistrée en amont.

### **5.3 Stratégie adaptative ou temps réel**

La recherche actuelle s'est orientée vers la réalisation de systèmes de régulation qui tiennent compte de l'évolution du trafic au cours du temps et qui réagissent en temps réel. C'est dans cet esprit que plusieurs types de méthode ont été élaborés pour ce mode de régulation, comme PRODYN [12].

## **6. Gestion de feux de carrefours**

**Position du problème** Les grandes villes possèdent un système de régulation du trafic permettant de gérer la circulation par une gestion des feux tricolores. La régulation du trafic se fait, en effet, par une adaptation permanente du fonctionnement des feux tricolores, aux conditions réelles de circulation.

Compte tenu de la complexité de l'implémentation des systèmes de régulation du trafic, il n'existe pas de méthode parfaite pour contrôler le flux dans une intersection routière. L'adéquation d'une méthode envisageable doit donc être très liée à la condition du trafic qu'il soit fluide, dense ou saturé.

# Chapitre I : Aperçu Général sur Le trafic routier

## 7. Conclusion

La régulation du trafic est un domaine vaste dans lequel plusieurs techniques et formes de signalisation sont utilisés pour faciliter la circulation des véhicules, augmenter la sécurité des usagers, réduire toute sorte de nuisances et rationaliser l'exploitation des infrastructures routières. Parmi les sujets les plus importants de la régulation du trafic, nous trouvons la gestion des carrefours à feux. Ces derniers constituent un terrain sensible pour les conflits de trafic et représentent un espace de convergence de différents acteurs du transport.

# Chapitre II :

## **APERCU GENERAL SUR L'APPRENTISSAGE AUTOMATIQUE**

## **Chapitre II : Aperçu Général sur L'apprentissage automatique**

### **1- Introduction**

Un réseau routier est conçu afin de permettre à ses usagers de se déplacer d'un point à un autre. Dans ce contexte, il est composé d'un ensemble de routes avec plus ou moins de voies, en fonction des zones d'activités et des lieux d'habitation. Les croisements entre les routes étant inévitables en milieu urbain, de par leur concentration, des intersections permettent de gérer les flux de véhicule s'entrant en conflit, pour leur sécurité et afin d'éviter les inter-blocages et de permettre aux automobilistes de changer de route. Ces intersections, ou dans certains cas des carrefours giratoires, régulent le trafic et appliquent généralement des règles de priorité à droite ou sont équipés de feux de circulation pour gérer les situations particulièrement dangereuses.

Dans ce chapitre, nous donnons en premiers les notions de base des feux tricolores, leurs significations, systèmes de transport intelligents. Ensuite nous allons présenter quelques solutions basées sur l'apprentissage automatique (ML) pour la gestion des intersection (feux de tricolores), On conclut ce chapitre avec quelques avantages et inconvénients de ces systèmes.

### **2 – Introduction sur Apprentissage automatique**

L'apprentissage automatique est une discipline qui consiste à appliquer des algorithmes à des jeux de données afin d'en extraire des modèles. Ceux-ci peuvent à leur tour être appliqués sur des données similaires à des fins de prédiction. Avec suffisamment de données, il est possible de formuler une approximation de la relation entre toutes les variables d'entrée et les valeurs particulières dites « cible ». On peut alors appliquer cette formule sur de nouvelles données d'entrée pour prévoir la valeur cible associée. Cette approche diffère de la programmation conventionnelle où une application est développée à partir de règles préalablement définies. Même si les concepts fondamentaux de l'apprentissage automatique existent depuis un certain temps, le domaine a récemment pris de l'ampleur en raison d'une part de l'amélioration de la performance des processeurs (en particulier graphiques), et d'autre part grâce à la disponibilité de grandes quantités d'information. Ces deux composantes qui sont essentielles à l'obtention de prévisions précises. Étant donné qu'il existe déjà suffisamment de littérature sur l'histoire de l'apprentissage automatique.[14]

## Chapitre II : Aperçu Général sur L'apprentissage automatique

### 3- Apprentissage automatique

#### 3.1- Définition de l'apprentissage automatique :

L'apprentissage automatique (" machine learning ") est une méthode utilisée en intelligence artificielle. Il s'agit des techniques qui analysent un ensemble de données afin de déduire des règles qui constituent de nouvelles connaissances permettant d'analyser de nouvelles situations.[15] C'est une technique qui sort un "modèle" à partir des "données" pouvant être selon plusieurs formats (des images, des sons, des vidéos, des valeurs numériques ou des signaux). Le concept de " l'apprentissage" apparait dans le fait que la technique analyse les données en entrée et trouve le modèle par soi-même au lieu d'avoir un humain pour le lui faire. On l'appelle apprentissage car la procédure ressemble à être entraîné avec les données (appelées données d'entraînement, jeu de données ou Training Data en anglais) pour résoudre le problème de trouver un modèle, qui `a son tour sera utilisé pour traiter d'autres données jamais vu avant. La figure suivante schématise le processus du machine learning,



Figure 6: Schéma général du Machine Learning

Le machine learning est créé pour résoudre les problèmes pour lesquels les modèles analytiques sont difficilement trouvés. Son idée principale est d'atteindre un modèle en utilisant les données d'entraînement quand les équations et les lois sont peu prometteuses.

#### 3.2 - Types d'apprentissage automatique :

Une première grande distinction à faire en "machine learning" est la différence entre les méthodes d'apprentissage :

## Chapitre II : Aperçu Général sur L'apprentissage automatique

### 3.2.1 - Apprentissage supervisé (Supervised Learning) :

C'est une méthode très similaire à l'apprentissage chez les humains. Le modèle prend en entrée les données et leurs sorties correctes (classes ou libellés cibles) et en s'entraînant pour les atteindre il " apprend " pour que l'algorithme devient capable, une fois entraîné, de prédire cette cible sur de nouvelles données non annotées.[16]

Ce type d'apprentissage automatique est basé sur l'utilisation de données pour l'ancien algorithme d'apprentissage. Les données sont identifiées comme entrées avec leurs sorties essentielles qui peuvent être décrit comme un signal de supervision. Le mécanisme d'apprentissage est décrit comme supervisé depuis la sortie correcte et l'algorithme d'apprentissage tente de prédire itérativement cette sortie et corrige ses paramètres pour réduire l'écart de variation entre sa production prévue et la production réelle. L'analyse du format des données permet de classer l'apprentissage supervisé en algorithme pour une fonction appelée classification si la sortie était discrète, et une fonction de régression si la sortie était continue (nombres réels).[17] La figure ci-dessous résume le processus de l'apprentissage supervisé.

**Remarques :** les quatre mécanisme développer sont de type d'apprentissage automatique supervisé



Figure 7 : Schéma général du Machine Learning

### 3.2.2 - Apprentissage non-supervisé (Unsupervised Learning) :

C'est dans le cas où l'entraînement du modèle repose sur l'utilisation des données non libellées en entrée. L'algorithme d'entraînement s'applique dans ce cas à trouver seul les similarités et distinctions au sein de ces données, et à regrouper ensemble celles qui partagent des caractéristiques communes. Donc, contrairement à l'apprentissage supervisé, cette méthode utilise un jeu de données en entrée sans sorties étiquetées pour entraîner l'algorithme d'apprentissage. Il n'y a pas de sortie correcte ou fautive sur chaque objet d'entrée et aucune

## Chapitre II : Aperçu Général sur L'apprentissage automatique

intervention humaine pour corriger ou ajuster comme dans l'apprentissage supervisé. L'objectif principal de l'apprentissage non supervisé est de savoir plus sur les données en identifiant les structures fondamentales ou les modèles de distribution qui se trouvent dans les données elles-mêmes. Apprend par lui-même, l'algorithme tente de représenter un motif d'entrée particulier tout en le reflétant sur la structure globale d'entrée. Ainsi, les différentes entrées sont regroupées en groupes en fonction des fonctionnalités et caractéristiques extraites de chaque objet d'entrée. Bien que l'algorithme n'attribue pas de noms aux résultats, il peut encore en produire, les différencier et utiliser certains d'entre eux pour assigner les nouveaux exemples.

Cette approche repose sur les données d'entrée et peut bien fonctionner quand il y a suffisamment de données disponibles pour l'utilisation. La figure suivante résume et montre le processus de ce type d'apprentissage.

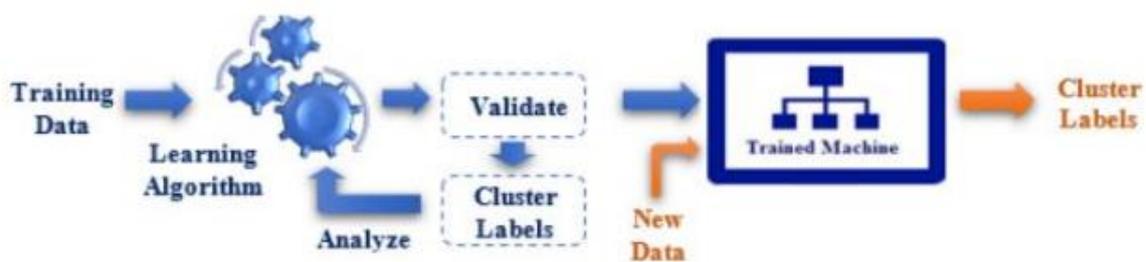


Figure 8 : Processus général de l'apprentissage non-supervisé

### 3.2.3- Apprentissage semi-supervisé :

Cette méthode se situe entre le supervisé et non supervisé : méthode d'apprentissage où nous avons une grande quantité de données d'entrée, dont certaines sont étiquetées et les autres ne le sont pas. Beaucoup de problèmes d'apprentissage de la vie réelle relèvent de ce domaine d'apprentissage automatique. La raison en est que semi-supervisé nécessite moins d'intervention humaine, car il utilise très peu de données étiquetées et une grande quantité de données non étiquetées car ces ensembles de données étiquetées sont très difficiles à collecter ainsi que coûteux et peuvent nécessiter accès aux experts du domaine. Les jeux de données non étiquetés de l'autre part sont moins chers et plus faciles à accéder. Les techniques d'apprentissage supervisé et non supervisé peuvent être utilisées pour former l'algorithme d'apprentissage en apprentissage semi-supervisé.

## Chapitre II : Aperçu Général sur L'apprentissage automatique

Des techniques d'apprentissage non supervisé peuvent être utilisées pour déplier les structures et les modèles cachés dans le jeu de données en entrée, tandis que les techniques d'apprentissage supervisé peuvent être utilisées pour faire des prédictions sur les données non étiquetées, nourrir les données de retour à l'algorithme d'apprentissage en tant que données d'apprentissage et utilisation pour l'acquisition de connaissances pour faire des prédictions sur de nouveaux ensembles de données. Ainsi, on peut dire que les données non étiquetées sont utilisées pour modifier ou redonner la priorité de prédiction ou l'hypothèse obtenue à partir de données étiquetées.[18]

La figure ci-dessous illustre les différentes étapes d'un processus semi-supervisé.

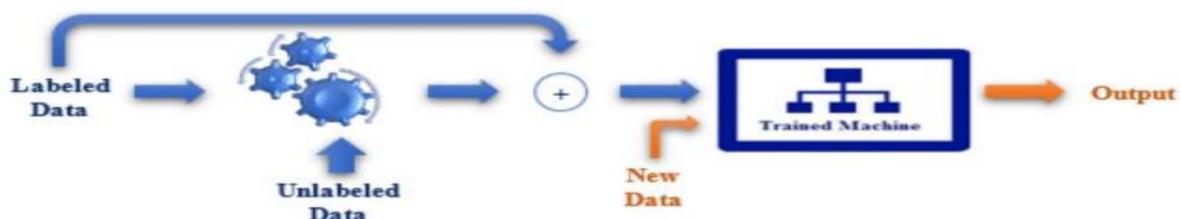


Figure 9 : Différentes étapes d'un processus semi-supervisé

### 3.2.4 - Apprentissage par renforcement (Reinforcement Learning) :

Dans ce cas, l'agent d'apprentissage prend en entrée les données, quelque sorties (état de l'environnement), et des observations de l'environnement pour ses sorties (actions). Il est utilisé généralement dans les cas où une interaction optimale est requise, comme dans le cas des jeux. L'apprentissage par renforcement c'est apprendre en interagissant avec l'environnement problématique. Un agent (algorithme) d'apprentissage par renforcement apprend de ses propres actions plutôt que d'être spécifiquement enseigné sur que faire. Il sélectionne les actions en cours en fonction des expériences passées (exploitation) et nouveaux choix (exploration). Ainsi, il peut être décrit comme un processus d'apprentissage par essais et erreurs.

Le succès d'une action est déterminé par un signal reçu par l'agent d'apprentissage de renforcement sous la forme d'une valeur numérique de récompense. L'agent cherche à apprendre à sélectionner des actions qui maximisent la valeur de la récompense numérique. Les agents d'apprentissage ont généralement des objectifs et ils peuvent détecter, dans une certaine mesure, l'état de l'environnement dans lequel ils se trouvent et donc prendre des mesures qui affectent l'état et le rapprocher de l'ensemble buts. L'apprentissage par

## Chapitre II : Aperçu Général sur L'apprentissage automatique

renforcement est différent du supervisé dans la manière dont chaque méthode acquiert des connaissances. La méthode d'apprentissage supervisé apprend à partir des exemples fournis par un superviseur externe, alors que l'apprentissage par renforcement utilise des interactions directes avec l'environnement du problème pour gagner connaissance. La figure suivante illustre le principe du processus de ce type d'apprentissage automatique.[19]

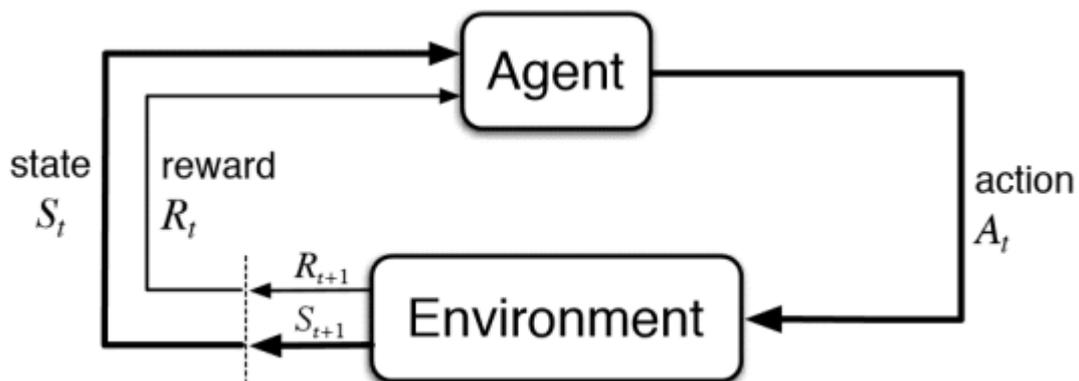


Figure 10 : Processus d'apprentissage par renforcement

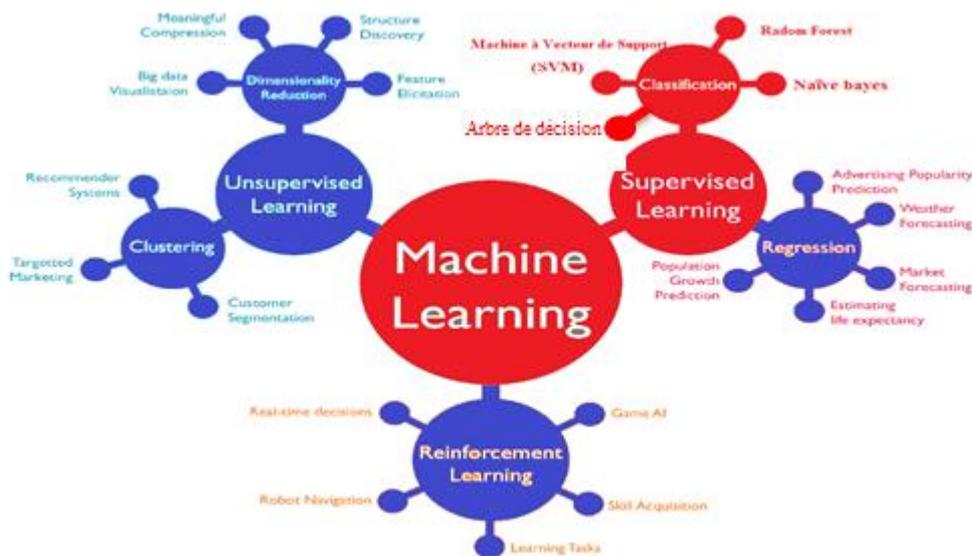


Figure 11 : Les types d'apprentissage automatique

**Remarque :** Nous basent sur (d'apprentissage automatique, Apprentissage supervisé, classification) pour résoudre notre approche.

## Chapitre II : Aperçu Général sur L'apprentissage automatique

### 3.3.1 – classification :

La classification est le processus de recherche ou de découverte d'un modèle (fonction) qui permet de séparer les données en plusieurs classes qualitatives. Dans la classification, l'appartenance au groupe du problème est identifiée, ce qui signifie que les données sont classées sous différentes étiquettes en fonction de certains paramètres, puis que les étiquettes sont prédites pour les données.

Le processus de la classification se déroule principalement en deux étapes : une phase d'apprentissage et une phase de test, Le prédicteur reçoit en entrée des données d'apprentissage qu'il utilise pour générer un modèle, il reçoit par la suite un fichier de test, dont il se servira pour prédire les nouvelles valeurs (0 : Rouge, 1 : Vert) en se basant sur le modèle préalablement généré. C'est le processus terminé, les données de test et les résultats des prédictions sont comparés pour traduire la précision.

### 3.3.2 - Les algorithmes de la classification :

Il existe plusieurs algorithmes pour la classification Dans notre solution, on a appliqué 4 algorithmes d'apprentissage automatique que nous allons comparer : Arbre de Décision, KNN, Machine à Vecteur de Support SVM, Forêt aléatoire (Random Forest),

#### 3.3.2.1 - Arbres de décision :

L'arbre de décision est une méthode de classification et prédiction, sa popularité repose en grande partie sur sa simplicité. Un arbre de décision est composé d'un nœud racine par lequel entrent les données, de nœuds feuilles qui correspondent à un classement de questions et de réponses qui conditionnent la question suivante.

Un arbre de décision est un outil d'aide à la décision représentant un ensemble de choix sous la forme graphique d'un arbre. Les différentes décisions possibles sont situées aux extrémités des branches (les « feuilles » de l'arbre), et sont atteintes en fonction de décisions prises à chaque étape. L'arbre de décision est un outil utilisé dans des domaines variés tels que la sécurité, la fouille de données, la médecine, etc. Il a l'avantage d'être lisible et rapide à exécuter. Il s'agit de plus d'une représentation calculable automatiquement par des algorithmes d'apprentissage supervisé.[20]

## Chapitre II : Aperçu Général sur L'apprentissage automatique

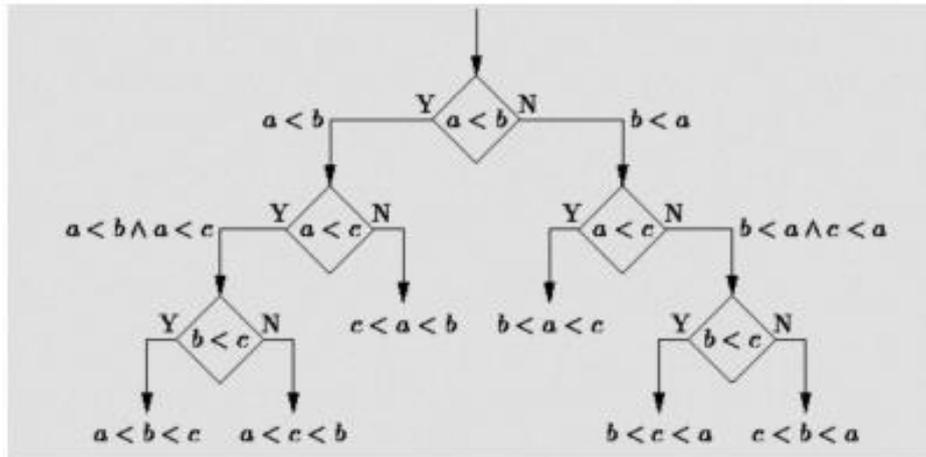


Figure 12 : L'arbre de décision

- **Algorithme d'arbre de décision générique**

**Entrée :** échantillon  $S$

**Début**

Initialiser l'arbre courant à l'arbre vide ; la racine est le nœud courant

**Répéter**

Décider si le nœud courant est terminal

**Si** le nœud est terminal **alors**

Lui affecter une classe

**Sinon**

Sélectionner un test et créer autant de nouveaux nœuds fils

Qu'il y a de réponses possibles au test

**FinSi**

Passer au nœud suivant non exploré s'il en existe

**Jusqu'à** obtenir un arbre de décision

**Fin**

\* **Les avantages :**

- La simplicité de compréhension et d'interprétation. C'est un modèle boîte blanche : si l'on observe une certaine situation sur un modèle, celle-ci peut être facilement expliquée à l'aide de la logique booléenne, au contraire de modèles boîte noire comme les réseaux neuronaux, dont l'explication des résultats est difficile à comprendre.

- Peu de préparation des données (pas de normalisation, de valeurs vides à supprimer, ou de variable muette)

## Chapitre II : Aperçu Général sur L'apprentissage automatique

- Le modèle peut gérer à la fois des valeurs numériques et des catégories. D'autres techniques sont souvent spécialisées sur un certain type de variables (les réseaux neuronaux ne sont utilisables que sur des variables numériques).
- Il est possible de valider un modèle à l'aide de tests statistiques, et ainsi de rendre compte de la fiabilité du modèle.
- Performant sur de grands jeux de données : la méthode est relativement économique en termes de ressources de calcul.

### \* Les inconvénients :

- L'apprentissage de l'arbre de décision optimal est NP-complet concernant plusieurs aspects de l'optimalité. En conséquence, les algorithmes d'apprentissage par arbre de décision sont basés sur des heuristiques telles que les algorithmes gloutons cherchant à optimiser le partage à chaque nœud, et de tels algorithmes ne garantissent pas d'obtenir l'optimum global.
- L'apprentissage par arbre de décision peut amener des arbres de décision très complexes, qui généralisent mal l'ensemble d'apprentissage. On utilise des procédures d'élagage pour contourner ce problème, certaines approches comme l'inférence conditionnelle permettent de s'en affranchir.
- Certains concepts sont difficiles à exprimer à l'aide d'arbres de décision. Dans ces cas, les arbres de décision deviennent extrêmement larges. Pour résoudre ce problème, plusieurs moyens existent, tels que la proportionnalisation, ou l'utilisation d'algorithmes d'apprentissage utilisant des représentations plus expressives (par exemple la programmation logique inductive).
- Lorsque les données incluent des attributs ayant plusieurs niveaux, le gain d'information dans l'arbre est biaisé en faveur de ces attributs. Cependant, le problème de la sélection de prédicteurs biaisés peut être contourné par des méthodes telles que l'inférence conditionnelle.

### 3.3.2.2 – Algorithme k plus proches voisins (KNN) :

L'algorithme de NearestNeighbour, connu sous le nom de KNN, est l'algorithme de base pour l'apprentissage automatique. Comprendre cet algorithme est un très bon endroit pour commencer à apprendre le machine learning, car la logique derrière cet algorithme est

## Chapitre II : Aperçu Général sur L'apprentissage automatique

incorporée dans de nombreux autres modèles de machine learning. L'algorithme de K NearestNeighbour fait partie de la partie classification de l'apprentissage supervisé.

K Nearest Neighbors est un algorithme de base qui stocke toutes les informations disponibles et prédit la classification des données non étiquetées sur la base d'une mesure de similarité. En géométrie linéaire lorsque deux paramètres sont tracés sur le système cartésien 2D, nous identifions la mesure de similitude en calculant la distance entre les points. La même chose s'applique ici, l'algorithme KNN fonctionne sur l'hypothèse que des choses similaires existent à proximité, simplement nous pouvons mettre dans les mêmes choses rester proches les uns des autres.[21]

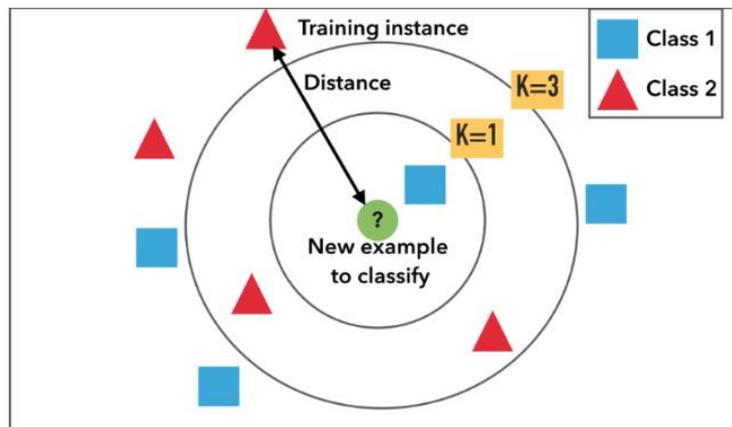


Figure 13 : méthode de k plus proches voisins

La fonction de distance euclidienne peut être calculée par l'équation suivante :

- Distance qui calcule la racine carrée de la somme des différences carrées entre les coordonnées de deux points :

$$D_{e(x,y)} = \sqrt{\sum_{i=1}^k (x_i - y_i)^2}$$

Si  $K = 1$ , alors le cas est simplement attribué à une classe de son plus proche voisin (nous utilisons «1» dans presque toutes les situations en mathématiques, nous pouvons modifier la valeur de  $K$  tout en entraînant les modèles dans l'apprentissage automatique et nous allons discuter plus en détail dans l'article)  $X$  et  $Y$  sont les valeurs sur les axes de coordonnées.

## Chapitre II : Aperçu Général sur L'apprentissage automatique

Si toutes les mesures de distance que nous obtenons seront des variables continues, mais nous avons besoin de valeurs discrètes lors de la classification, nous devons donc utiliser la distance de Manhattan pour y parvenir.

- la distance de Manhattan: calcule la somme des valeurs absolues des différences entre les coordonnées de deux points :

$$D_{m(x,y)} = \sum_{j=1}^k |x_j - y_j|$$

- **Algorithme KNN**

### Début Algorithme

Données en entrée :

- un ensemble de données **D**.
- une fonction de définition distance **d**.
- Un nombre entier **K**.

Pour une nouvelle observation **X** dont on veut prédire sa variable de sortie **y** Faire :

1. Calculer toutes les distances de cette observation **X** avec les autres observations du jeu de données **D**.
2. Retenir les **K** observations du jeu de données **D** les proches de **X** en utilisation la fonction de calcul de distance **d**.
3. Prendre les valeurs de **y** des **K** observations retenues :
  1. Si on effectue une régression, calculer la moyenne (ou la médiane) de **y** retenues
  2. Si on effectue une classification, calculer le mode de **y** retenues
4. Retourner la valeur calculée dans l'étape 3 comme étant la valeur qui a été prédite par K-NN pour l'observation **X**.

### Fin Algorithme

- **Les avantages :**
  - L'algorithme est simple et facile à mettre en œuvre.
  - Aucune hypothèse sur les données (linéaires, affines,).
  - L'algorithme est polyvalent. Il peut être utilisé pour la classification, la régression.
- **Les inconvénients :**
  - L'algorithme devient beaucoup plus lent à mesure que le nombre d'exemples d'apprentissage augmente.

## Chapitre II : Aperçu Général sur L'apprentissage automatique

- Le choix de la méthode de calcul de la distance ainsi que le nombre de voisins K peut ne pas être évident.
- L'étape de prédiction peut être lente.

### 3.3.2.3 -Machine à Vecteur de Support (SVM) :

Le SVM est un classificateur dit linéaire, ça veut dire que, dans le cas parfait, les données doivent être linéairement séparables. Ainsi notre corpus est représenté comme étant un espace vectoriel, où chaque donnée représentée par un point dans ce dernier. La problématique maintenant est de trouver le meilleur séparateur (ligne, plan ou hyperplan) qui partage notre corpus en deux catégories. L'espace entre ces deux catégories est appelé marge, qui est définie par les points (Vecteurs de support) les plus proches du séparateur, de part et d'autre. Le but étant essentiellement de maximiser cette marge, plus elle est grande meilleur est le résultat. Le classificateur se généralise bien avec les nouvelles données.

Toutefois, si les données ne sont pas linéairement séparables, la SVM peut être modifiée pour tolérer un minimum d'erreurs. Désormais, le but est de maximiser la marge et de minimiser l'erreur de classification. Une autre alternative pour parer à la non-séparabilité des données, est de passer à un espace de dimension supérieur.

La figure ci-dessous illustre la façon dont SVM départage les données dans le cas où elles sont linéairement séparables, de plus, elle choisit la séparation la plus optimale où la marge est maximale.[22]

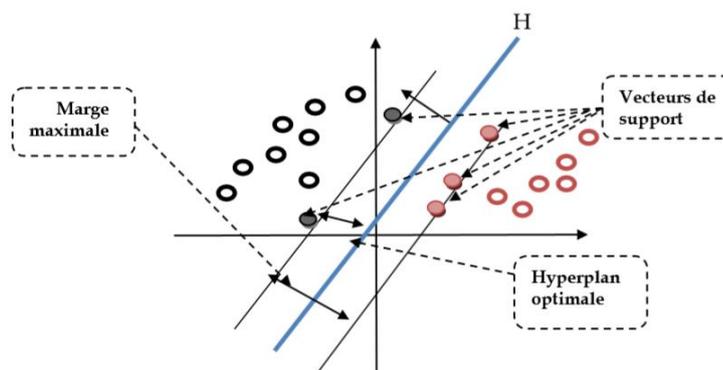


Figure 14 : hyperplan optimal, vecteurs de support et marge maximale.

## Chapitre II : Aperçu Général sur L'apprentissage automatique

- **L'algorithme de SVM**

**Méthode SVM**

- 1 : Optimiser le vecteur de poids  $w$
- 2 : Calculer le critère de rang  $R_c(X_j)$  pour tout  $j \in \{1, \dots, p\}$
- 3 : Éliminer la variable qui minimise  $R_c(X_j)$
- 4: Répéter les étapes 1 à 3 tant qu'il reste des variables

- **Les Avantages :**

Les SVM représentent plusieurs avantages parmi lesquels on peut citer par exemple :

- Fonctionne bien sur de plus petits data sets.
- Le faible nombre d'hyper paramètres utilisés par ces méthodes.
- Elles sont bien fondées théoriquement.
- Les résultats pertinents qu'on obtient avec les SVM en pratique.

- **Les Inconvénients :**

Malgré leurs performances, les SVM représentent aussi des faiblesses, parmi lesquelles on peut citer :

- Leur utilisation des fonctions mathématiques complexes pour la classification des corpus.
- Pour trouver les meilleurs paramètres, ce type d'algorithmes demande un temps énorme pendant les phases de test.

### 3.3.2.4 -Réseau Bayésien(Naïve bayes) :

Un réseau bayésien est un graphe orienté et acyclique qui contient des nœuds connectés entre eux par des arcs représentant des dépendances probabilistes entre les variables et des distributions de probabilités conditionnelles pour chaque variable. La classification naïve bayésienne est basée sur le théorème de Bayes avec une forte indépendance des hypothèses.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

$P(A)$  : la probabilité que l'hypothèse  $h$  soit vraie (indépendamment des données). Ceci est connu comme la probabilité a priori de  $h$ .

## Chapitre II : Aperçu Général sur L'apprentissage automatique

$P(B)$  : la probabilité des données (quelle que soit l'hypothèse). C'est ce qu'on appelle la probabilité a priori.

$P(A|B)$  : la probabilité de l'hypothèse  $h$  compte tenu des données  $D$ . C'est ce qu'on appelle la probabilité postérieure  
 $(B|A)$  : la probabilité des données  $d$  étant donné que l'hypothèse  $h$  était vraie. C'est ce qu'on appelle la probabilité postérieure[23].

- **Algorithme de Réseau Bayésien :**

D un ensemble de données avec leurs attribut prévues  $V$ ,  $p(w_k(v))$  la probabilité d'une valeur  $w_k$  tiré au hasard à partir d'une donnée quelconque de la classe  $v_j$  et  $p(v_j)$  la «Prior probabilités» de la classe  $v_j$ .

1. Collecter de tous les attributs apparaissant dans l'ensemble des données.

2. Calculer les termes  $p(v_j)$  et  $p(w_k | v_j)$

**Pour** chaque valeur  $v_j$  de  $V$  faire

*données  $j$* : les sous-ensembles de données de  $D$  pour lesquels la valeur prévue est  $v_j$ .

$$p(v_j) = \frac{|doonnes_j|}{|D|}$$

**Pour** chaque valeur  $w_k$  apparaît dans *données*

$$p(w_k | v_j) = \frac{|n_k + 1|}{n + |D|}$$

- **Les Avantages :**

- C'est relativement simple à comprendre et à construire.
- Ils faciles à former, même avec un petit jeu de données.
- Il n'est pas sensible aux caractéristiques non pertinentes.

- **Inconvénients :**

- Il implique que chaque fonctionnalité soit indépendante, ce qui n'est pas toujours le cas.

### 3.3.2.5 -Forêt aléatoire (Radom Forest)

L'algorithme des « forêts aléatoires » (ou Radom Forest parfois aussi traduit par forêt d'arbres Décisionnels) est un algorithme de classification qui réduit la variance des prévisions d'un arbre de décision seul, améliorant ainsi leurs performances. Pour cela, il combine de nombreux arbres de décisions dans une approche de type bagging. Chaque arbre de la forêt aléatoire est entraîné sur un sous ensemble aléatoire de données selon le principe du bagging, avec un sous ensemble aléatoire de Features (caractéristiques variables des données)

## Chapitre II : Aperçu Général sur L'apprentissage automatique

selon le principe des « projections aléatoires ». Les prédictions sont ensuite moyennées lorsque les données sont quantitatives ou utilisés pour un vote pour des données qualitatives, dans le cas des arbres de classification. L'algorithme des forêts aléatoires est connu pour être un des classeurs les plus efficaces « out-of-the-box » (c'est-à-dire nécessitant peu de prétraitement des données)[24].

### • Définition du Bagging

Le bagging a pour but de réduire la variance de l'estimateur, en d'autres termes de corriger l'instabilité des arbres de décision (le fait que de petites modifications dans l'ensemble d'apprentissage entraînent des arbres très différents).

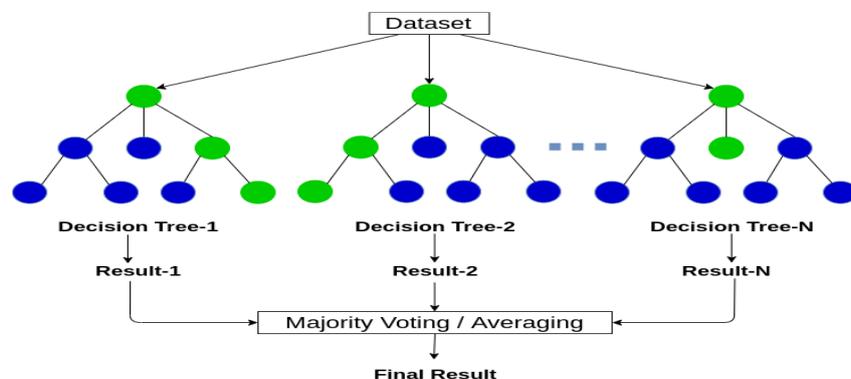


Figure 15 : l'algorithme de forêt aléatoire.

### • Algorithme de Forêt aléatoire

**Méthode Construire-Forêt - aléatoire**(D : données, N : nœud, M : méthode)

1. On applique M sur D afin de trouver le critère de division
2. En utilisant le critère de division, on divise le nœud N en sous nœuds
3. On met le nombre des sous nœuds de N dans la variable k

**Si**  $k > 0$  alors

On fabrique k sous nœuds  $N_1, \dots, N_k$  de N

On divise D en  $D_1, \dots, D_k$

**Pour**  $i=1$  jusqu'à k faire

Construire- Forêt - Aléatoire( $D_i, N_i, M$ )

**Fin pour**

**Fin Si**

**Fin de la procédure**

### • Avantages :

- Reconnaissance très rapide.
- Multi-classes par nature.

## Chapitre II : Aperçu Général sur L'apprentissage automatique

- Efficace sur inputs de grande dimension.
- Robustesse aux outliers.
- **Inconvénients :**
  - Apprentissage souvent long.
  - Les forêts aléatoires sont très bonnes pour les problèmes de classification mais légèrement moins bonnes pour les problèmes de régression. Contrairement à la régression linéaire, la régression forestière aléatoire est incapable de faire des prédictions en dehors de ses données d'apprentissage.
  - Les forêts aléatoires sont également des boîtes noires, contrairement à certains algorithmes d'apprentissage automatique traditionnels, il est difficile de regarder à l'intérieur d'un classificateur de forêt aléatoire et de comprendre les raisons de ses décisions. En outre, ils peuvent être lents à s'entraîner et à s'exécuter et produire des fichiers de grande taille.

	Avantages	Inconvénient
Arbre de décision	<ul style="list-style-type: none"> <li>-La simplicité de compréhension et d'interprétation</li> <li>-Le modèle peut gérer à la fois des valeurs numériques et des catégories.</li> <li>- Performant sur de grands jeux de données</li> </ul>	<ul style="list-style-type: none"> <li>-Certains concepts sont difficiles à exprimer à l'aide d'arbres de décision.</li> <li>- peut amener des arbres de décision très complexes</li> </ul>
Forêt aléatoire	<ul style="list-style-type: none"> <li>- Reconnaissance très rapide.</li> <li>- Multi-classes par nature.</li> <li>-Efficace sur inputs de grande dimension.</li> </ul>	<ul style="list-style-type: none"> <li>- Apprentissage souvent long.</li> <li>- sont très bonnes seulement pour les problèmes de classification</li> <li>-il est difficile de regarder à l'intérieur d'un classificateur</li> </ul>
KNN	<ul style="list-style-type: none"> <li>- L'algorithme est simple et facile à mettre en œuvre.</li> <li>- Aucune hypothèse sur les données (linéaires, affines,...).</li> <li>- L'algorithme est polyvalent. Il peut être utilisé pour la classification, la régression.</li> </ul>	<ul style="list-style-type: none"> <li>- Le choix de la méthode de calcul de la distance ainsi que le nombre de voisins K peut ne pas être évident.</li> <li>- L'étape de prédiction peut être lente.</li> </ul>

## Chapitre II : Aperçu Général sur L'apprentissage automatique

SVM	<ul style="list-style-type: none"><li>- Fonctionne bien sur de plus petits data sets</li><li>- Le faible nombre d'hyper paramètres utilisés par ces méthodes.</li><li>-Elles sont bien fondée théoriquement.</li></ul>	<ul style="list-style-type: none"><li>-utilisation des fonctions mathématiques complexes</li><li>- Moins efficace sur les jeux de données contenant du bruits</li></ul>
-----	--	---

Tableau 01 : les avantages et inconvénient de chaqu'un des algorithmes

### 3.4– métriques pour évaluer les algorithmes d'apprentissage automatique

La performance d'un algorithme de Machine Learning est directement liée à sa capacité à prédire un résultat. Lorsque l'on cherche à comparer les résultats d'un algorithme à la réalité, et pour cela nous utilisons les métriques suivantes :

#### 3.4.1- Accuracy :

Il s'agit du rapport entre le nombre de prédictions correctes et le nombre total d'échantillons d'entrée.

$$Accuracy = \frac{\text{Nombre de prédiction correcte}}{\text{Nombre total de prédiction}}$$

Il ne fonctionne bien que s'il y a un nombre égal d'échantillons appartenant à chaque classe.

#### 3.4.2- Précision :

Nombre de résultats positifs corrects divisé par le nombre de résultats positifs prédits par le classificateur.

$$précision = \frac{\text{positifs corrects}}{\text{positifs corrects} + \text{positifs incorrect}}$$

#### 3.4.3- Recall:

Il s'agit du nombre de résultats positifs exacts divisé par le nombre de tous les échantillons pertinents.

$$Recall = \frac{\text{positifs corrects}}{\text{positifs corrects} + \text{negative incorrect}}$$

## Chapitre II : Aperçu Général sur L'apprentissage automatique

### 3.4.4- F1Score :

Le score F1 est la moyenne harmonique entre la précision et le rappel. La plage du score F1 est [0, 1]. Il vous indique à quel point votre classificateur est précis (combien d'instances il classe correctement), ainsi que sa robustesse (il ne manque pas un nombre significatif d'instances).

Une précision élevée, mais un rappel inférieur, vous donne une précision extrême, mais il manque ensuite un grand nombre d'instances qui sont difficiles à classer. Plus le score F1 est élevé, meilleure est la performance de notre modèle. Mathématiquement, il peut être exprimé comme :

$$F1 = 2 * \frac{1}{\frac{1}{\text{précision}} + \frac{1}{\text{reccal}}}$$

### 3.4.5- Matrice de confusion :

La matrice de confusion comme son nom l'indique nous donne une matrice comme résultat et décrit la performance complète du modèle.

Supposons que nous ayons un problème de classification binaire. Nous avons quelques échantillons appartenant à deux classes : OUI ou NON. En outre, nous avons notre propre classificateur qui prédit une classe pour un échantillon d'entrée donné. En testant notre modèle sur 165 échantillons, nous obtenons le résultat suivant.

N =165	<b>Prédiction : non</b>	<b>Prédiction : oui</b>
<b>Actuel : non</b>	50	10
<b>Actuel : oui</b>	05	100

Il y a 4 termes importants :

Vrais positifs : Les cas dans lesquels nous avons prédit OUI et la sortie réelle était OUI.

Vrais négatifs : Les cas dans lesquels nous avons prédit NO et la sortie réelle était NO.

Faux positifs : Les cas dans lesquels nous avons prédit OUI et la sortie réelle était NON.

Faux négatifs : Les cas dans lesquels nous avons prédit NON et la sortie réelle était OUI.

## Chapitre II : Aperçu Général sur L'apprentissage automatique

### 4 -Conclusion :

Dans ce deuxième chapitre on a parlé de l'apprentissage automatique ses classes, ses solutions ses avantage et ses inconvénients, Notre vision est d'utiliser le volet superviser d'apprentissage automatique (Arbre de Décision, k plus proches voisins KNN, Machine à Vecteur de Support (SVM), Foret aléatoire (Radom Forest)) comme des stratégies pour administrer intelligemment une circulation typique dans le chapitre prochain.

# **Chapitre III :**

**Implémentation et évaluation**

# Chapitre III : implémentation et évaluation

## 1. Introduction

L'objectif de ce dernier chapitre est de présenter les étapes de l'implémentation et de l'évaluation de l'approche proposée. Une discussion qui commence par l'introduction des outils technologiques utilisés, l'environnement de développement utilisé et on termine par les étapes de la réalisation et l'évaluation des mécanismes implémentés.

## 2. Présentation de l'approche proposé :

Nous avons proposé un modèle de prédiction (système de feux de signalisation intelligence) basé sur L'apprentissage automatique superviser.

Ce système prend entrées les données réelles des différentes situations a la circulation avec contexte détaillé, ces données sont exploitées comme une base d'apprentissage par les algorithmes ML afin de générer une solution de coordination qui gères les interactions dans l'intersection.

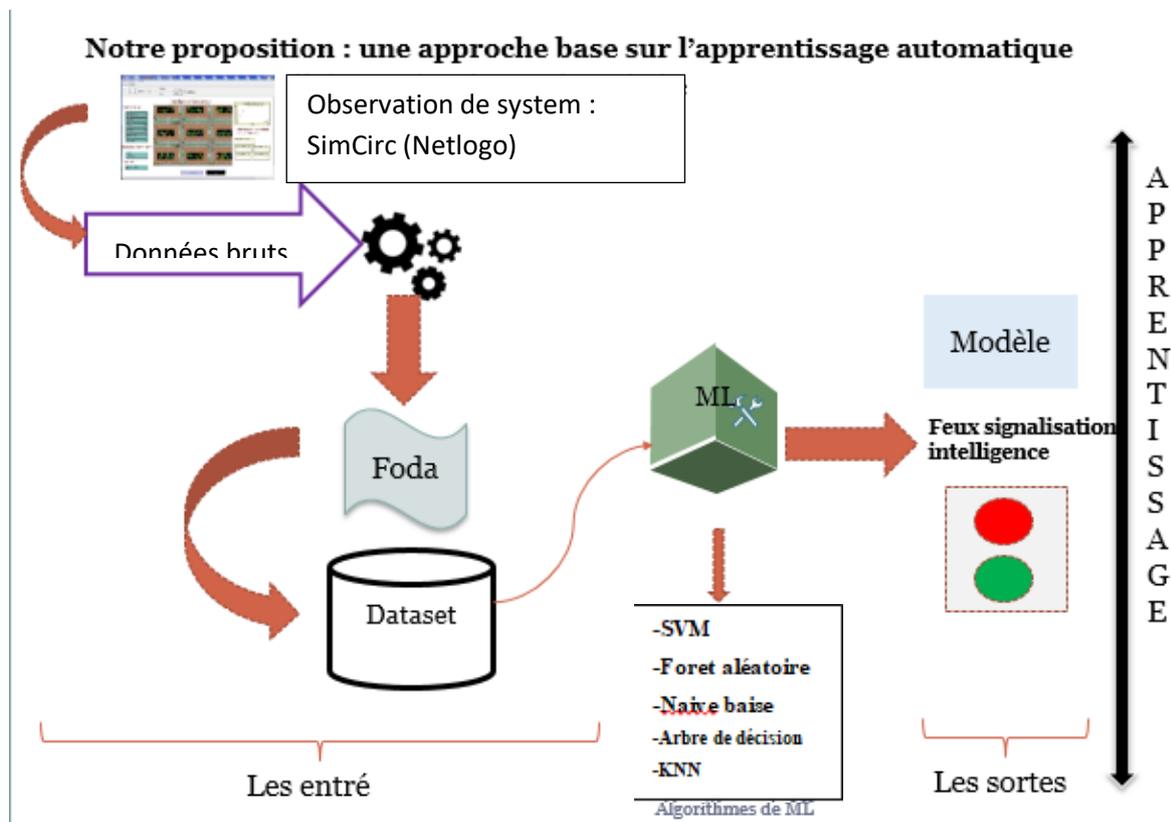


Figure 16 : Présentation de notre approche

## Chapitre III : Implémentation et évaluation

### 3. Traitement de data set :

La création de notre dataset passe par les étapes suivantes :

- créer la structure de dataset,
- extraction de les données
- alimente notre data set par des données conformes à Foda.

Dans ce mémoire on utilise le simulateur SimCirc [35] comme un outils fictive pour observer/déduire les paramètres/résultats du DataSet discerné. comme il représenté par la figure 17 .

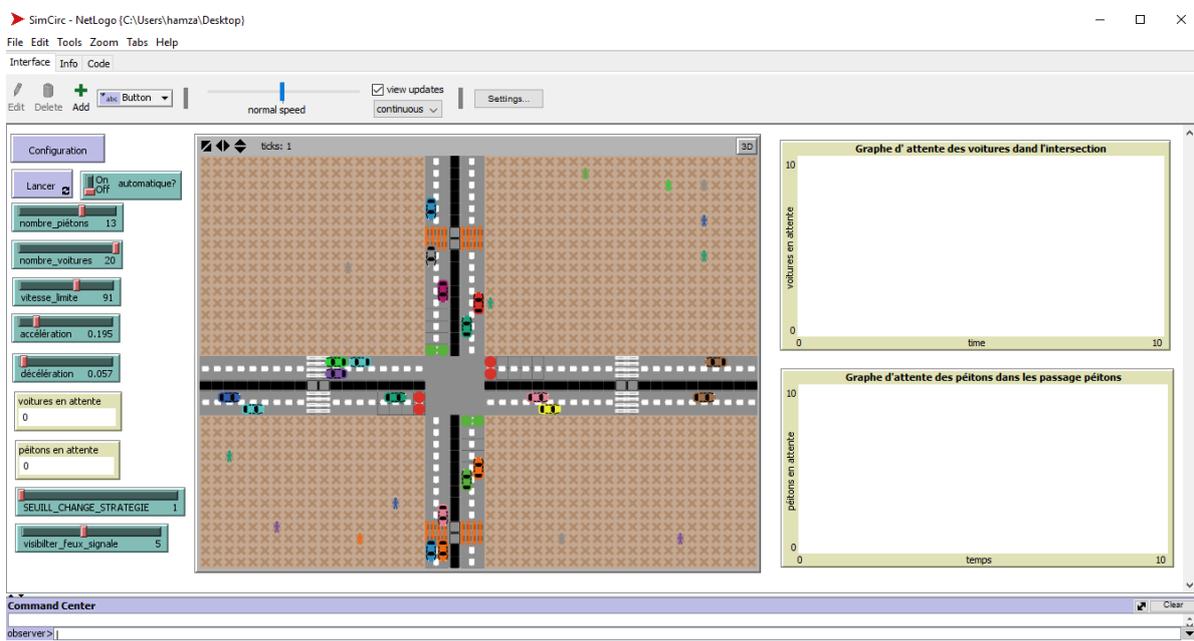


Figure 17 : Simulation Net Logo [35]

### Traitement les données avec Foda (Feature-Oriented Domain Analysis)

Pour analyser les paramètres de dataset à introduire dans la base, nous avons utilisé le diagramme de caractéristique SPL. Dans notre cas les attributs de notre data set représente les caractéristiques notre SPL par exemple : les feux de circulation, route, Véhicule, ...etc. Pour concevoir notre dataset il faut analyser le domaine nous avons utilisé le diagramme de caractéristique du SPL afin de d'écrire la variabilité.

## Chapitre III : Implémentation et évaluation

### 3.1 Feature-Oriented Domain Analysis (Foda):

La méthode FODA (Feature-Oriented Domain Analysis) a introduit au début des années 1990 le concept des fonctionnalités et l'utilisation de diagrammes de fonctionnalités (feature diagramme, FD) pour exécuter l'analyse du domaine ciblé par un logiciel. Bien que conceptuellement simples, ces diagrammes permettent de modéliser

La variabilité d'une ligne de produits au niveau de ses fonctionnalités. Afin de faciliter l'adoption du diagramme de fonctionnalités [25].

#### 3.1.1 Modélisation de la variabilité :

L'avantage d'utiliser ce modèle réside dans la possibilité modéliser la variabilité durant la gestion de circulation de véhicule dans un carrefour

##### A. Présentation Les caractéristiques de la circulation :

Nous représentons un espace de variante globale à l'aide d'un diagramme de fonctionnalités. Pour une première classification des facteurs d'impact, nous définissons trois groupes :

1- Route, 2- Véhicule, 3- Personne.

Les trois facteurs d'impact sont toujours présents lors de l'exécution et l'évaluation :

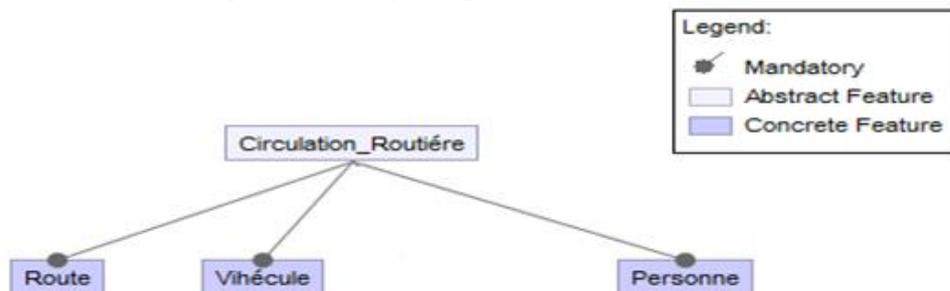


Figure 18 : Les caractéristiques de la circulation routière

## Chapitre III : Implémentation et évaluation

1. **Route** : c'est au sens littéral une voie terrestre aménagée pour permettre la circulation de véhicule, on a basé dans la route sur la voie et l'intersection.

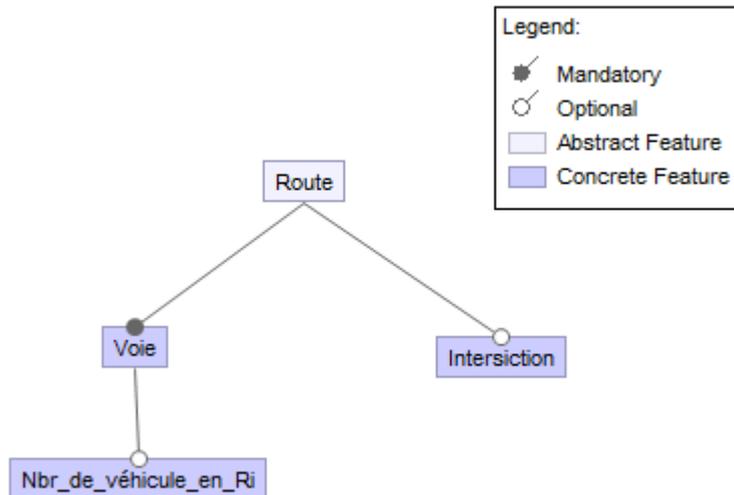


Figure 19 : Les caractéristiques de la route

**NB** :  $R(i)$  : représente le nombre de route.  $i = 4$

2. **Voiture** : la voiture est considérée comme l'un des moyens de transport les plus répandus à notre époque, ce que nous intéressent de la part de véhicule dans notre approche.

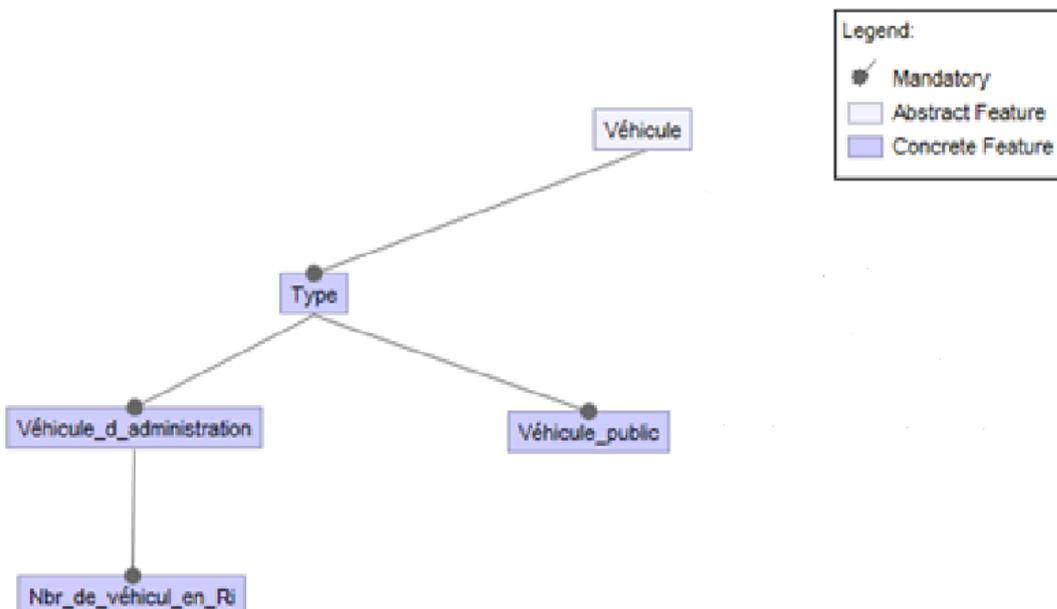


Figure 20 : Les caractéristiques de la Véhicule.

## Chapitre III : Implémentation et évaluation

### 3. Personnes

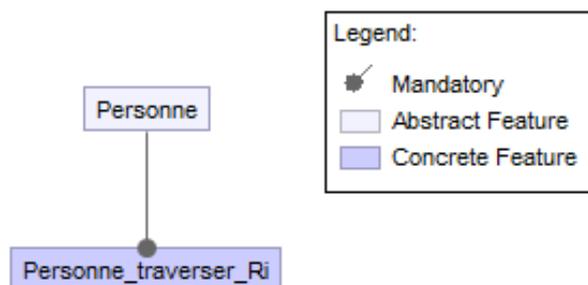


Figure 21 : Les caractéristiques des personne

### B. Présentation Les caractéristiques du feu de circulation :

Les paramètres spécifiques aux feux de circulation sont : Le Feu vert et Le Feu rouge

#### B.1.Couleur Rouge / Vert

#### B.2.Positionnement

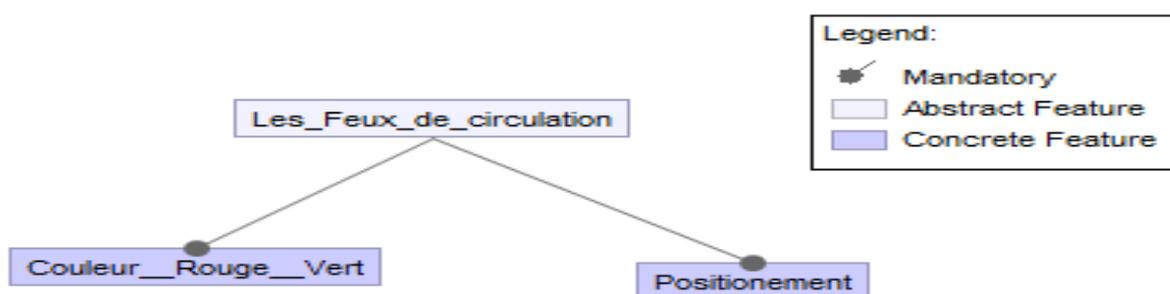


Figure 22 : Les caractéristiques du feu de signalisation.

## 4. Environnement de développement et Les outils techniques

### 4.1 Dentition du langage Python en informatique :

Python est le langage de programmation open source, est orienté objet et se veut relativement facile d'accès. Il est très utilisé au sein de la communauté scientifique et particulièrement dans le domaine de l'intelligence artificielle. Les principaux frameworks de machine learning et deep learning sont effectivement disponibles en Python. On a utilisé la version Python3.9.0 [26].

### 4.2 Anaconda :

Anaconda est une distribution open source pour python. Il est utilisé pour la science de données, L'apprentissage automatique, l'apprentissage en profondeur, etc. Avec la disponibilité de plus de trois cent Bibliothèques pour la science des données, il devient assez optimal pour tout programmeur de travailler Sur anaconda pour la science des données [27].

## Chapitre III : Implémentation et évaluation

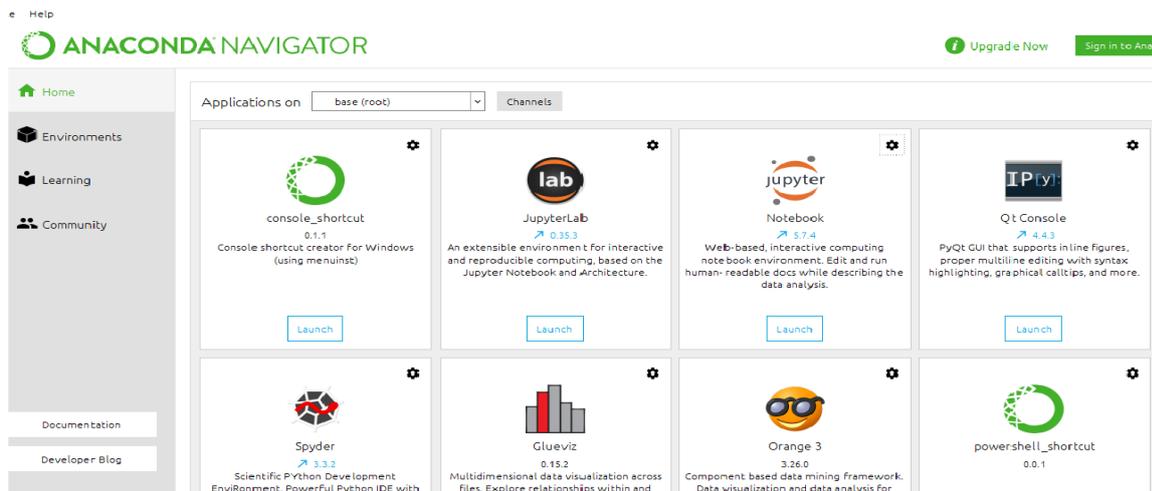


Figure 23 : L'environnement Anaconda

### 4.3 Jupyter :

Se présente comme un outil extrêmement simple à mettre en œuvre qui vous permettra de transformer vos Jupyter Notebooks en applications web ou en Dashboard quasiment automatiquement.



### 4.4. Les bibliothèques utilisées :

Dans le développement de notre solution, nous avons utilisé plusieurs bibliothèques : Numpy, Matplotlib, Pandas et Scikit-learn.

#### 4.4.1. Numpy :

Est une bibliothèque permettant d'exécuter des calculs numériques avec Python. Elle introduit une gestion facilitée des tableaux de nombres, des fonctions sophistiquées (division), on peut aussi l'intégrer le code C / C ++ et Fortran [28].

#### 4.4.2. Matplotlib :

Est une bibliothèque de traçage pour le langage de programmation Python et son extension mathématique numérique NumPy. Il fournit une API orientée objet permettant d'incorporer des graphiques dans des applications à l'aide de kits d'outils d'interface graphique à usage général tels que Tkinter, python, Qt ou GTK + [29].

## Chapitre III : Implémentation et évaluation

### 4.4.3. Pandas :

Pandas est une bibliothèque open source sous licence BSD fournissant des structures de données hautes performances et faciles à utiliser, ainsi que des outils d'analyse de données pour le langage de programmation Python. Pandas est un projet sponsorisé par NumFOCUS cela contribue.[30]

### 4.4.4. Scikit-learn :

Scikit-learn est une bibliothèque libre Python destinée à l'apprentissage automatique. Elle est développée par de nombreux contributeurs notamment dans le monde académique par des instituts français d'enseignement supérieur et de recherche comme Inria3 et Télécom Paris. Elle comprend notamment des fonctions pour estimer des forêts aléatoires, des régressions logistiques, des algorithmes de classification, et les machines à vecteurs de support. Elle est conçue pour s'harmoniser avec d'autres bibliothèques libres Python, notamment NumPy et SciPy [31].

## 1. Apprentissage et Paramétrage des Modèles :

Dans cette section, nous allons présenter l'apprentissage et paramétrage des modèles.

### 5.1 Préparation du Dataset :

Pour les algorithmes le jeu de données doit être en forme de xlsx ou CSV. et la dernière colonne doit contenir la décision et doit également être appelée (label).

```
18 data = pd.read_excel('C:/Users/Badro/Desktop/code finale/data finale/datafinale.xlsx')
19
```

On élimine toutes les valeurs manquantes dans la dataset.

Nous allons découper notre dataset en deux, données de test (20% des données), et données d'entraînement (80% des données).

```
81 # splitting the data test train split
82 x_train , x_test , y_train , y_test = train_test_split(X,y , test_size =0.2 , random_state = 20)
83 print(x_test, y_test)
84 print(x_train, y_train)
85
```

## Chapitre III : Implémentation et évaluation

### Avantages de la séparation train / test :

- Le modèle peut être formé et testé sur des données différentes de celles utilisées pour la formation.
- Les valeurs de réponse sont connues pour l'ensemble de données de test, donc les prédictions peuvent être évaluées.
- La précision des tests est une meilleure estimation que la précision de la formation des performances hors échantillon.

### 5.2. Application de PCA :

Pour exécuter PCA on a utilisé la bibliothèque Scikit-Learn de Python.

On utilise PCA pour :

- Améliore la visualisation les variables.
- Améliore les performances de l'algorithme.
- Les variables indépendantes deviennent moins interprétables.

La réalisation de PCA à l'aide de Scikit-Learn est un processus en deux étapes :

- Initialisez la classe PCA en passant le nombre de composants au constructeur.
- Appelez l'ajustement, puis transformez les méthodes en transmettant l'ensemble de fonctionnalités à ces méthodes.

La méthode de transformation renvoie le nombre spécifié de composants principaux, voici le code suivant :

```
50 from sklearn.decomposition import PCA
51 pca = PCA(2)
52 new_X = pca.fit_transform(X)
53 ratio = pca.explained_variance_ratio_
54
55 # Plotting The data
56 plt.scatter(new_X[:,0] , new_X[:,1] , c = y )
57 plt.legend()
58 plt.show()
59
```

La réalisation de PCA à l'aide de Scikit-Learn est un processus en deux étapes :

Initialisez la classe PCA en passant le nombre de composants au constructeur.

Appelez l'ajustement, puis transformez les méthodes en transmettant l'ensemble de fonctionnalités à ces méthodes.

La méthode de transformation renvoie le nombre spécifié de composants principaux,

## Chapitre III : Implémentation et évaluation

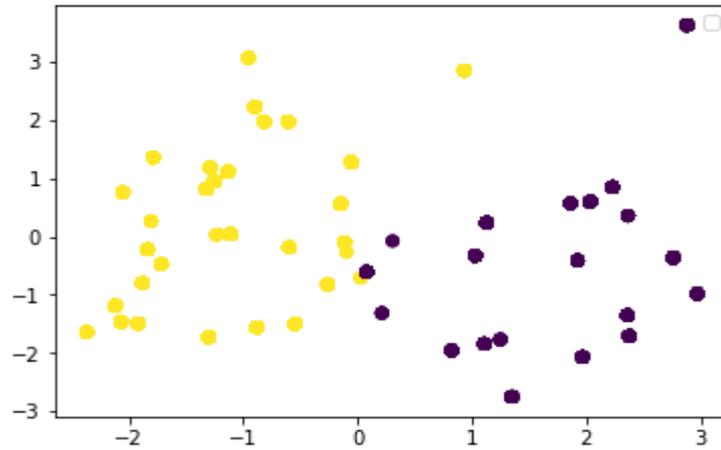


Figure 24 : L'exécution de l'algorithme PCA.

### 5.3. Algorithme LDA

L'analyse discriminante linéaire ou LDA est une technique de réduction de dimensionnalité. Il est utilisé comme étape de pré-traitement dans l'apprentissage automatique et les applications de classification de modèles. L'objectif de LDA est de projeter les caractéristiques d'un espace de dimension supérieure sur un espace de dimension inférieure afin d'éviter la malédiction de la dimensionnalité et également de réduire les ressources et les coûts dimensionnels.[32]

Le code suivant l'algorithme L'analyse discriminante linéaire ou quadratique

```
74 #trying LDA
75 lda = LinearDiscriminantAnalysis(n_components=1)
76 X_iris_LDA = lda.fit(X, y).transform(X)
77 print(X_iris_LDA)
78 print("Rappel des composantes identifiées par Le PCA :")
79 graph_acp2(new_X, y)
80
81 # spiting the data test train split
82 x_train , x_test , y_train , y_test = train_test_split(X,y , test_size =0.2 , random_state = 20)
83 print(x_test, y_test)
84 print(x_train, y_train)
85
```

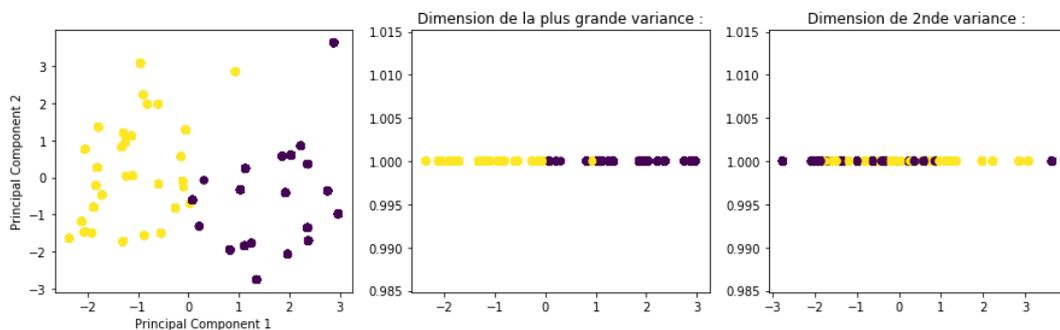


Figure 25 : L'exécution de l'algorithme LDA

## Chapitre III : Implémentation et évaluation

### 6. Apprentissage du modèle

Dans la recherche de modèles, on utilise des algorithmes d'apprentissage supervisé.

#### 6.1 Algorithme Arbre de décision :

Nous avons utilisé l'algorithme de l'arbre de décision pour obtenir un modèle de classification qui nous aide à décider la meilleure décision.

##### 6.1.1 Explication du code d'algorithme Arbre de décision :

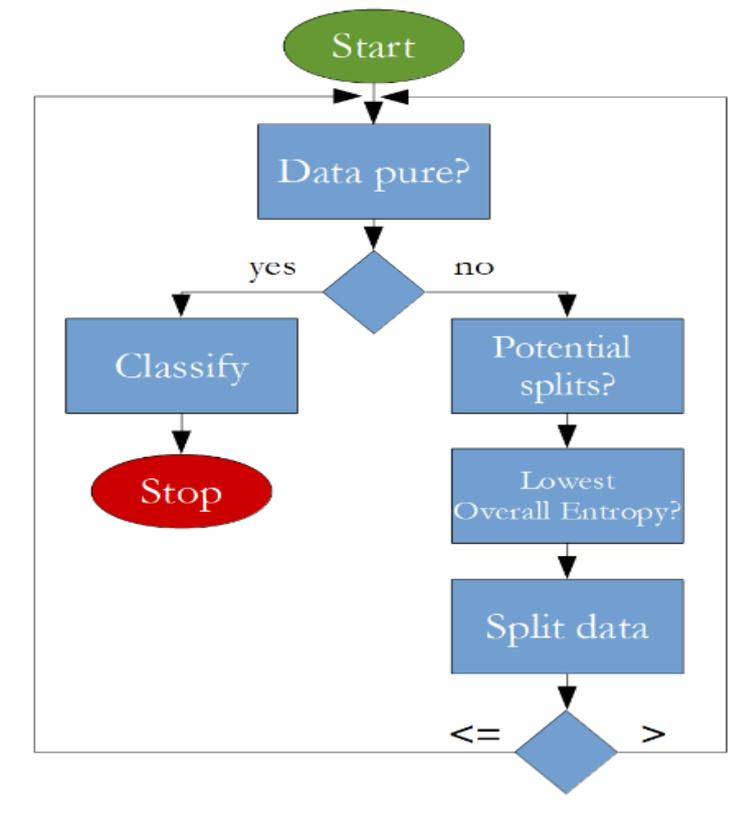


Figure 26 : Arbre de décision organigramme

L'idée principale est de diviser de manière récursive les échantillons d'apprentissage aussi efficacement que possible en utilisant des tests définis par les attributs, jusqu'à ce qu'un sous-échantillon contenant (presque) uniquement des exemples appartenant à la même classe soit obtenu.

## Chapitre III : Implémentation et évaluation

### Le code de l'algorithme Arbre de décision :

```
98 from sklearn.tree import DecisionTreeClassifier
99
100 X_train2, X_test2, y_train2, y_test2 = train_test_split(X, y, random_state=0)
101
102 clf = DecisionTreeClassifier(max_leaf_nodes=3, random_state=0)
103 clf.fit(X_train2, y_train2)
104 y_pred4 = clf.predict(X_test2)
105 cm = confusion_matrix(y_test2, y_pred4)
106 print("Accuracy Decision Tree :", accuracy_score(y_test2, y_pred4))
107 print("scort DecisionTree:", f1_score(y_test2, y_pred4))
108 print('Confusion Matrix is :', cm)
109 disp = plot_confusion_matrix(clf, X_test2, y_test2)
110 disp.ax_.set_title("plot")
111
```

### Résultat d'exécution :

- Precision et recall :

```
the precision is : 0.4942528735632184
the RECALL is : 0.7818181818181819
```

- Score et Accuracy :

```
Accuracy Decision Tree : 0.84
scort DecisionTree: 0.8360655737704917
```

- Matrice de confusion :

```
Confusion Matrix is : [[44  1]
 [12 43]]
```

La matrice ci-dessous explique la performance de l'algorithme par rapport aux échantillons d'essai.

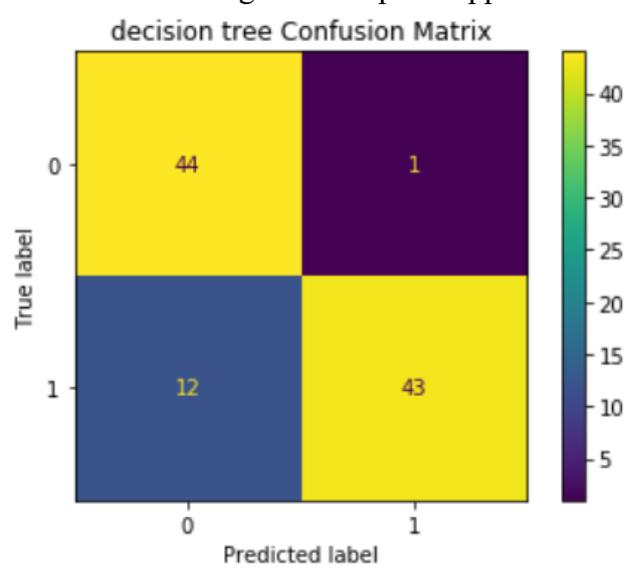


Figure 27 : Matrice de confusion de l'arbre de décision

## Chapitre III : Implémentation et évaluation

Dans la matrice l'axe des x représente les prédictions de l'algorithme (1 : prédiction vraie, 0 prédiction fausse), et l'axe des y représente les prédictions en réalité (1 : prédiction vraie, 0 prédiction fausse).

### 6.2 Algorithme de Radom Forest :

Nous avons utilisé l'algorithme de forêt aléatoire pour obtenir un modèle de classification qui nous aide à décider la meilleure décision.

#### 6.2.1 Explication du code d'algorithme Radom Forest :

L'idée est simplement d'utiliser de nombreux arbres de décision différents pour classer un nouvel exemple inconnu (au lieu d'avoir un seul arbre).

Et cela se passe en quatre étapes :

- Sélectionnez des échantillons aléatoires à partir d'une partie au hasard de dataset.
- Construisez un arbre de décision pour chaque échantillon et obtenez un résultat de prédiction à partir de chaque arbre de décision.
- Effectuez un vote pour chaque résultat prévu.
- Sélectionnez le résultat de la prédiction avec le plus de votes comme prédiction finale.

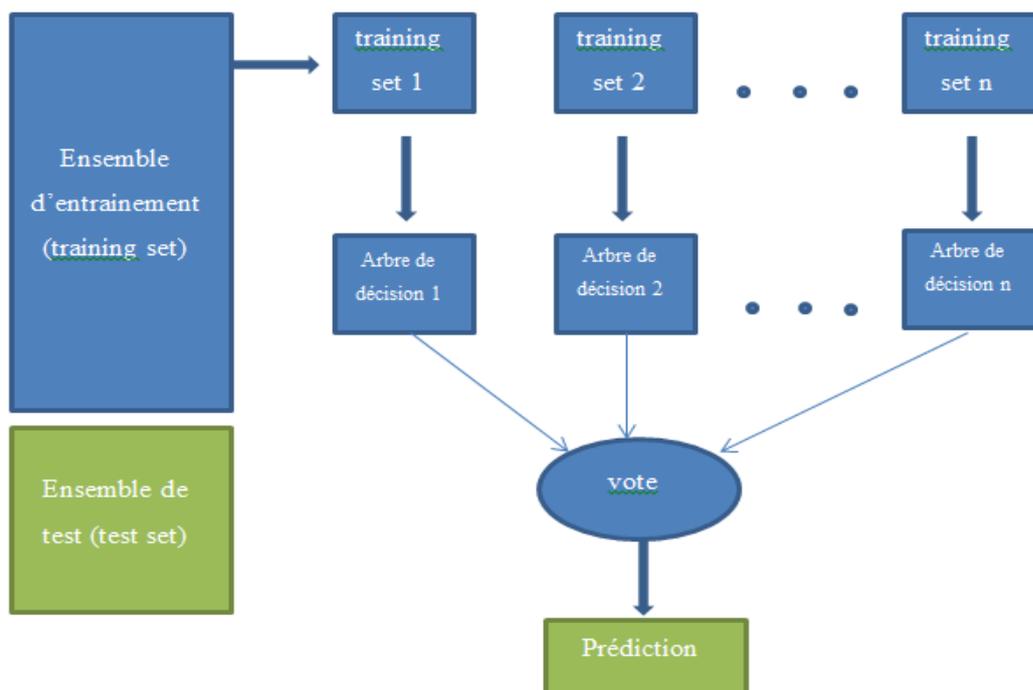


Figure 28 : Organigramme d'algorithme Radom Forest

## Chapitre III : Implémentation et évaluation

### Le code de l'algorithme de Foret Aléatoire :

```
from sklearn.ensemble import RandomForestClassifier
##### Random Forest
print('#####')
print('random forest algorithme')
h.DT = RandomForestClassifier(min_weight_fraction_leaf= 0.3 , criterion = 'entropy')
h.DT.fit(h.x_train,h.y_train)
h.y_pred1 = h.DT.predict(h.x_test)

print('The accuracy For Random Forest is ', (h.y_pred1 == h.y_test).mean())
cm = h.confusion_matrix(h.y_test, h.y_pred1)
```

### Résultat d'exécution :

- Précision et recall ;

```
LA precision est : 0.5925925925925926
the RECALL is : 0.8727272727272727
```

- Score et Accuracy :

```
random forest algorithme
The score For Random Forest: 0.8959999999999999
The accuracy For Random Forest is 0.87
```

- Matrice de confusion :

```
Confusion Matrix is : [[32 13]
 [ 3 52]]
```

La matrice ci-dessous explique la performance de l'algorithme par rapport aux échantillons d'essai.

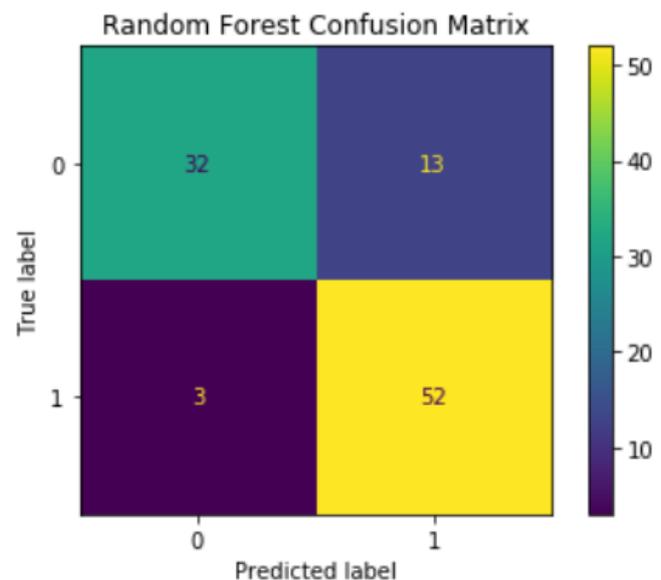


Figure 29 : Matrice de confusion de Foret Aléatoire

Dans la matrice l'axe des x représente les prédictions de l'algorithme (1 : prédiction vraie,

## Chapitre III : Implémentation et évaluation

0 prédiction fausse), et l'axe des y représente les prédictions en réalité (1 : prédiction vraie, 0 prédiction fausse).

### 6.3 Algorithme de SVM :

Nous avons utilisé l'algorithme de SVM pour obtenir un modèle de classification qui nous aide à décider la meilleure décision.

#### 6.3.1 Explication du code d'algorithme SVM :

L'idée générale est de séparer les données (ensemble des vecteurs) en entrée avec une frontière de séparation.

La frontière de séparation est choisie comme celle qui maximise la marge entre les points qui séparent une classe d'autre comme le montre la figure ci-dessous :

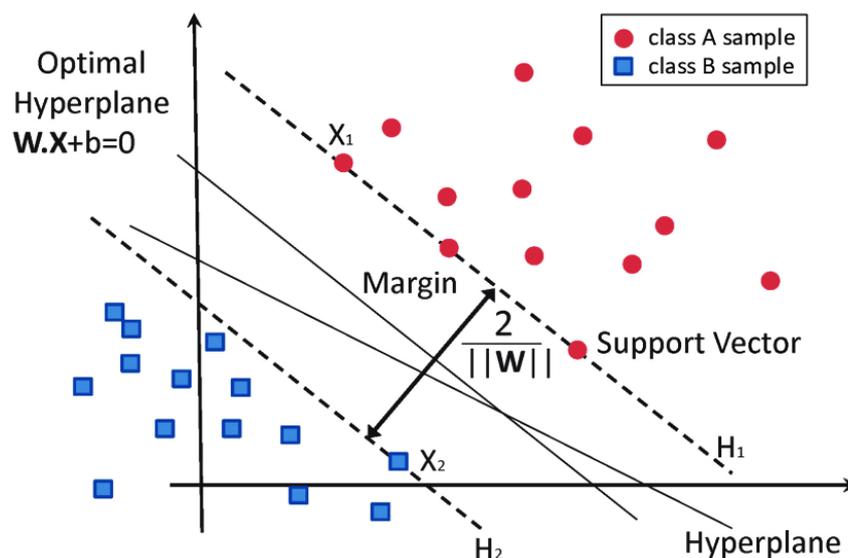


Figure 30 : classification des données avec SVM [33]

#### Le code de l'algorithme SVM :

```
75 print('SVM algorithme')
76 from sklearn.svm import SVC
77 ##### SVM
78 ▼ svc = SVC(kernel = 'sigmoid' , random_state = 12 , gamma =0.9 , C =12)
79 svc.fit(h.x_train, h.y_train)
80 h.y_pred2 = svc.predict(h.x_test)
81
82 #print("SVM",svc.predict(test))
83 print("Accuracy SVM:",h.accuracy_score(h.y_test, h.y_pred2))
84 print(" scort SVM:", h.f1_score(h.y_test, h.y_pred2))
85 cm = h.confusion_matrix(h.y_test,h.y_pred2)
86 print('Confusion Matrix is :', cm)
87 disp = plot_confusion_matrix(svc, h.x_test,h.y_test)
```

## Chapitre III : Implémentation et évaluation

### Résultat d'exécution :

- précision et recall ;

```
LA precision est : 0.5512820512820513
the RECALL is : 0.7818181818181819
```

- Score et Accuracy :

```
SVM algorithme
Accuracy SVM: 0.76
scort SVM: 0.8208955223880597
```

- Matrice de confusion :

```
Confusion Matrix is : [[35 10]
 [12 43]]
```

La matrice ci-dessous explique la performance de l'algorithme par rapport aux échantillons d'essai.

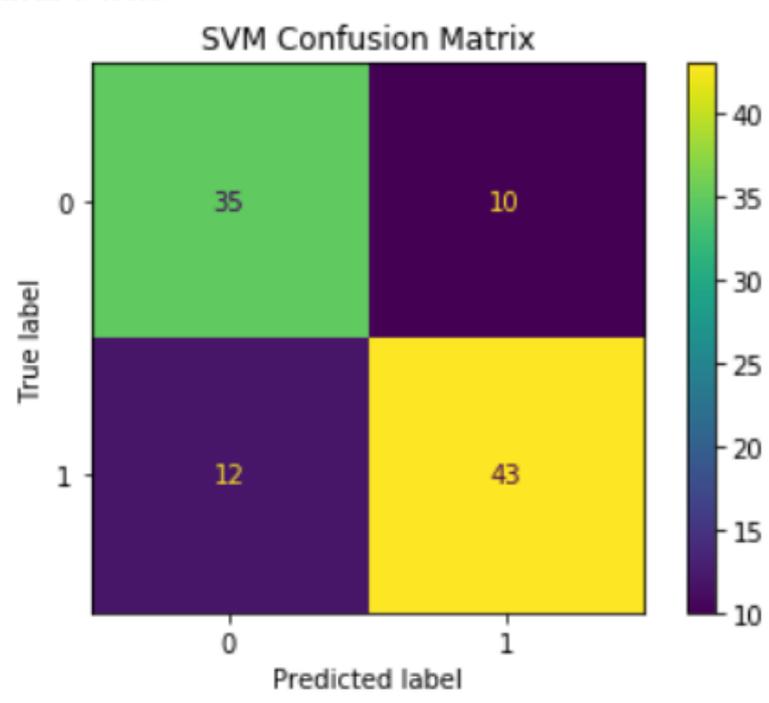


Figure 31 : Matrice de confusion de SVM

Dans la matrice l'axe des x représente les prédictions de l'algorithme (1 : prédiction vraie, 0 prédiction fausse), et l'axe des y représente les prédictions en réalité (1 : prédiction vraie, 0 prédiction fausse).

## Chapitre III : Implémentation et évaluation

### 6.4 Algorithme de KNN :

Nous avons utilisé l'algorithme de KNN pour obtenir un modèle de classification qui nous aide à décider la meilleure décision.

#### 6.4.1 Explication du code d'algorithme KNN:

Nous donnons à l'algorithme des données d'entrée étiquetées et il classera ces données en classes à l'aide du nombre k, et quand nous voulons classer un nouvel échantillon l'algorithme calculera la distance entre l'échantillon et les k voisins et il l'append à la classe qui a le grand nombre de voisins proches de l'échantillon comme le montre la figure ci-dessous :

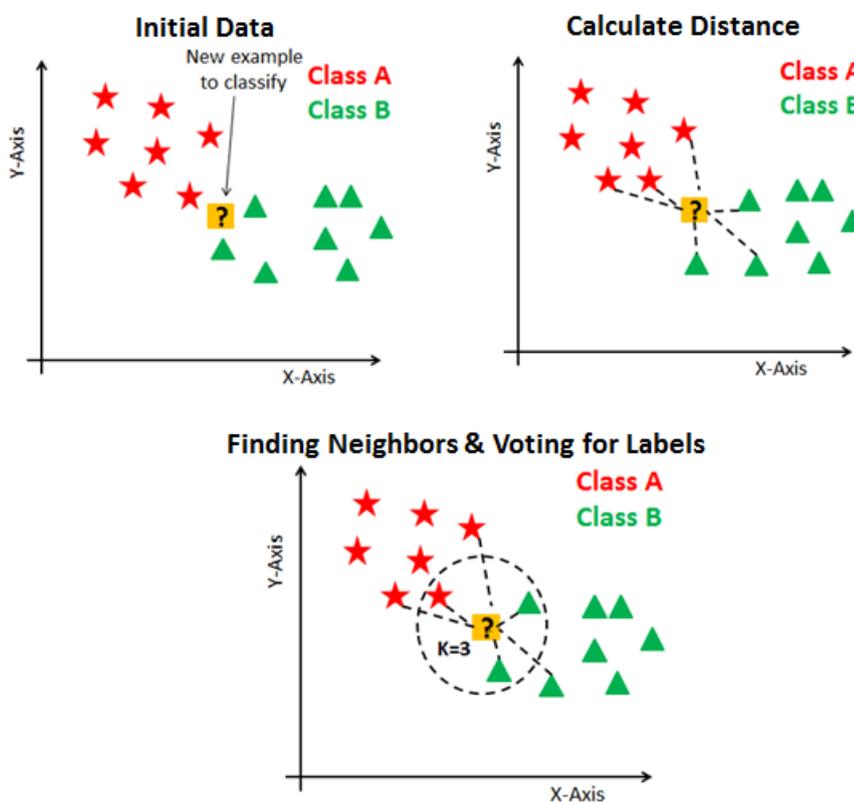


Figure 32 : classification des données avec KNN [34]

#### Le code de l'algorithme KNN :

```
57 from sklearn.neighbors import KNeighborsClassifier
58 dataset = pd.read_csv("C:/Users/Badro/Desktop/code finale/data finale/Classeur2.csv")
59 X = dataset.drop("Decision", axis=1)
60 y = dataset.Decision
61 X_train5, X_test5, y_train5, y_test5 = train_test_split(X, y, test_size=0.2)
62 knn = KNeighborsClassifier(n_neighbors=8)
63 knn.fit(X_train5, y_train5)
64 y_pred3 = knn.predict(X_test5)
65 cmK = confusion_matrix(y_test5, y_pred3)
66 print('Confusion Matrix is :', cmK)
67 print("Accuracy KNN :", accuracy_score(y_test5, y_pred3))
68 print("scort KNN:", f1_score(y_test5, y_pred3))
69 disp = plot_confusion_matrix(knn, X_test5, y_test5)
70 disp.ax_.set_title("plot")
```

## Chapitre III : Implémentation et évaluation

### Résultat d'exécution :

- Précision et recall ;

```
LA precision est : 0.5909090909090909  
the RECALL is : 0.9454545454545454
```

- Score et Accuracy :

```
Accuracy KNN : 0.88  
scort KNN: 0.8461538461538461
```

- Matrice de confusion :

```
Confusion Matrix is : [[36  9]  
 [ 3 52]]
```

La matrice ci-dessous explique la performance de l'algorithme par rapport aux échantillons d'essai

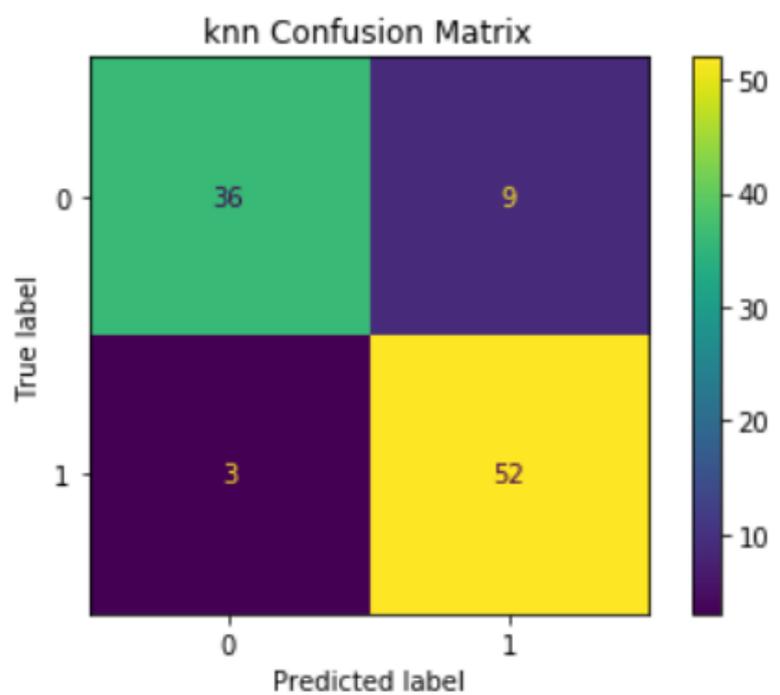


Figure 33 : Matrice de confusion de KNN

Dans la matrice l'axe des x représente les prédictions de l'algorithme (1 : prédiction vraie, 0 prédiction fausse), et l'axe des y représente les prédictions en réalité (1 : prédiction vraie,

0 Prédiction fausse).

## Chapitre III : Implémentation et évaluation

### 7. Comparaison entre Les quatre algorithmes :

Dans cette partie on utilise deux scénarios distinct détaillé comme suit :

#### 7.1. Scenario 01 :

##### A- Description :

Dans ce premier scenario on a utilisé les quatre mécanismes basés sur apprentissage automatique et on s'appuie sur la métrique d'évaluation accuracy.

$$Accuracy = \frac{\text{Nombre de prédiction correcte}}{\text{Nombre total de prédiction}}$$

##### B- Analyse

Nous avons comparé les quatre algorithmes de classification Arbre de décision, SVM, KNN et Foret aléatoire. Cette comparaison est basée sur les métriques de performances accuracy. Nous avons comparé la qualité du modèle en termes de la précision. Les résultats sont affichés dans la figure suivante.

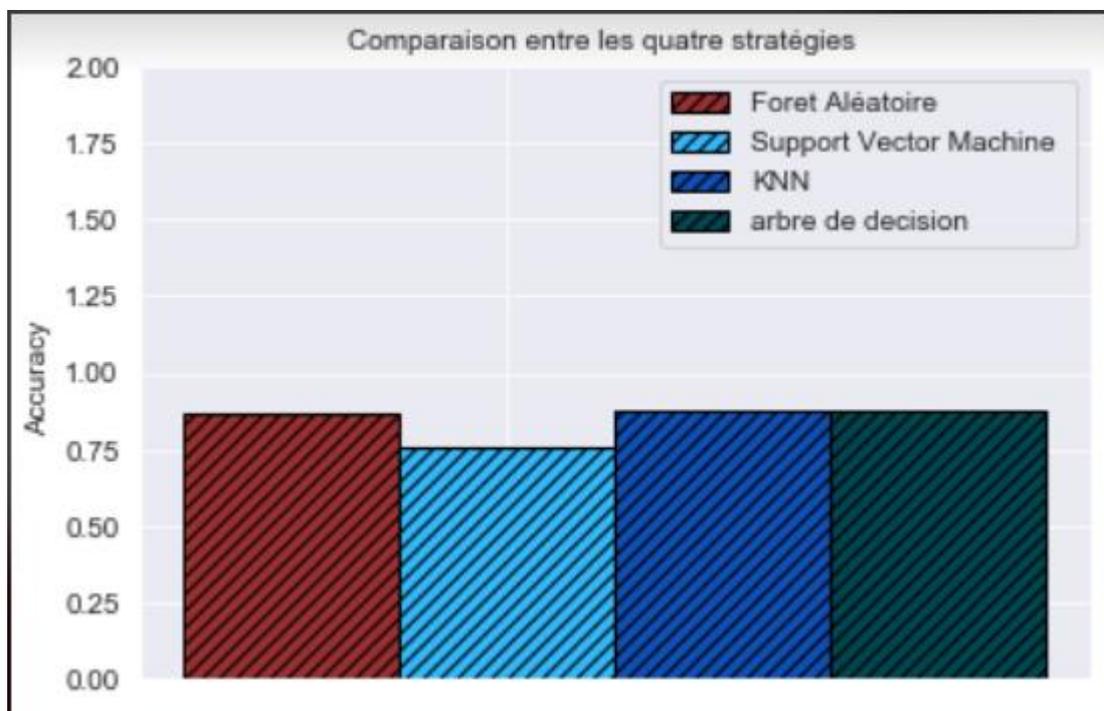


Figure 34 : Comparaison entre les quatre algorithmes par apport Accuracy

## Chapitre III : Implémentation et évaluation

### C-Comparaison entre les quatre algorithmes : par apport Accuracy:

Métrique d'évaluation	Accuracy
Foret Aléatoire	89 %
KNN	88 %
Arbre de décision	84 %
SVM	76 %

Tableau 02 : Comparaison entre les trois stratégies par apport Accuracy, Score

### D-Récapitulation :

Dans ce scenario, Foret Aléatoire, KNN et arbre de décision montrons les plus grande accarucy (89 %, 88%, 84%) dans les valeurs moyennes de précision de classification que Support Vector Machine (76 \%), la différence est statistiquement significative. Le modèle SVM est difficile de détecter le contexte correct à moins qu'un autre algorithme d'apprentissage Foret Aléatoire, KNN et arbre de décision. Nous concluons que le résultat obtenu avec les modèles : Foret Aléatoire et KNN sont les meilleurs.

### 7.2.Scenario 02 :

#### A-Description

Dans ce deuxième scenario on a utilisé les quatre mécanismes basé sur apprentissage automatique et on s'appuie sur les métriques d'évaluation précision, recall et score.

$$\text{précision} = \frac{\text{positifs corrects}}{\text{positifs corrects} + \text{positifs incorect}}$$

$$\text{Recall} = \frac{\text{positifs corrects}}{\text{positifs corrects} + \text{negative incorect}}$$

$$\text{score} = 2 * \frac{1}{\frac{1}{\text{précision}} + \frac{1}{\text{reccal}}}$$

## Chapitre III : Implémentation et évaluation

### B-Analyse:

Nous avons comparé les quatre algorithmes de classification Arbre de décision, SVM, KNN et Forêt aléatoire. Cette comparaison est basée sur les métriques de performances précision, recall et score. Nous avons comparé la qualité du modèle en termes de la précision. Les résultats sont affichés dans les figure suivante :

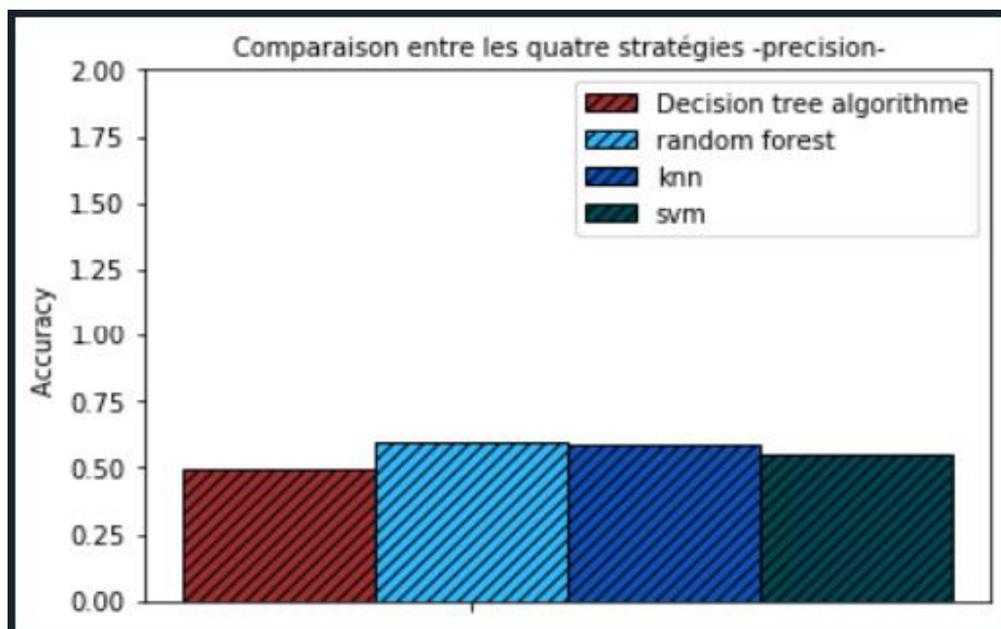


Figure 35: Comparaison entre les quatre algorithmes par apport précision

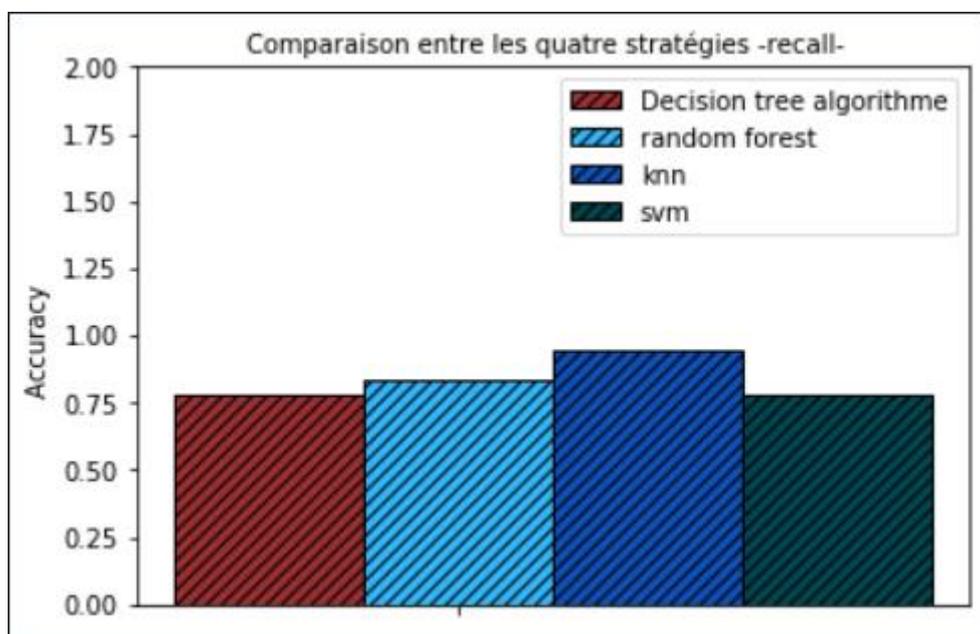


Figure 36: Comparaison entre les quatre algorithmes par apport recall

## Chapitre III : Implémentation et évaluation

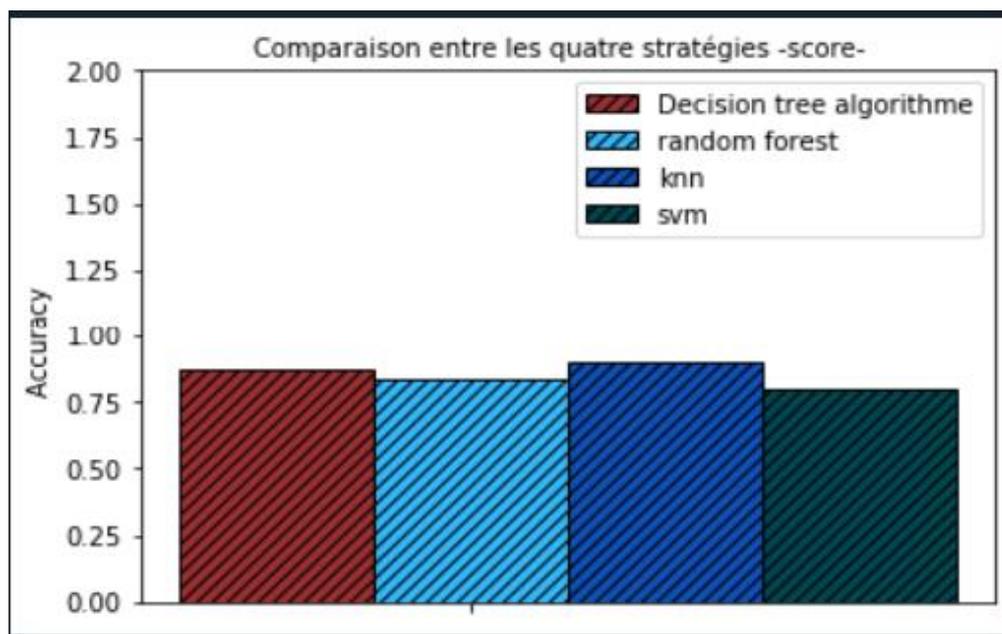


Figure 37: Comparaison entre les quatre algorithmes par apport score

### C-Récapitulation :

Dans ce scenario, Foret Aléatoire, KNN et arbre de décision montrons les plus grande score (88 %, 89%, 87%) dans les valeurs moyennes de précision de classification que Support Vector Machine (75 \%), la différence est statistiquement significative. Le modèle SVM est difficile de détecter le contexte correct à moins qu'un autre algorithme d'apprentissage Foret Aléatoire, KNN et arbre de décision. Nous concluons que le résultat obtenu avec les modèles : Foret Aléatoire et KNN sont les meilleurs.

### Synthèse :

à travers les résultats obtenu nous synthétisons qu'avec les mécanisme basés sur Foret Aléatoire et KNN sont les meilleurs mécanismes et leur taux de fausse prédiction est très faible par rapport au mécanismes basé sur SVM.

### 8. conclusion :

Dans ce chapitre nous avons, en premier lieu, présenté notre approche, les différents outils et langages que nous avons utilisés pour implémenter notre model. Après cela, nous avons expliqué comment nous avons extrait notre ensemble de données du simulateur netlogo, puis nous avons parlé des algorithmes de machine learning supervisé que nous avons utilisé dans notre approche et les résultats de chaque algorithme. Finalement nous avons fait une étude comparative sur le résultat des algorithmes avec les métriques de performance.

# **Conclusion générale**

## Conclusion générale

### Conclusion générale :

Dans ce mémoire, nous nous sommes intéressés à la commande des feux de signalisation en vue de résoudre les problèmes liés à la congestion ; dans ce sens, nous avons proposé cinq mécanismes pour rendre les feux de signalisation intelligente et non pas automatique en s'appuyant sur l'apprentissage automatique. Les mécanismes implémentés sont basés sur quatre algorithmes de l'apprentissage automatique : arbre de décision, support vecteur machine (SVM), k nearest neighbor (KNN) et forêt aléatoire (random forest). Les solutions obtenues permettent de prédire la décision afin d'agir dans la circulation en fonction de plusieurs paramètres comme la route, voitures, les passagers etc.

Dans le deuxième chapitre, nous avons présenté le domaine de la circulation routière, les composants et les variables élémentaires de ce domaine, la congestion, les méthodes pour gérer quelques problèmes dans ce domaine « la circulation routière » tel que les feux signalisation dans les intersections.

Dans le troisième chapitre, nous avons proposé notre contribution qui s'articule particulièrement sur les approches supervisées, notre intuition est d'utiliser le volet superviser des approches de l'apprentissage automatique (Forêt aléatoire, Machine à Vecteur de Support, KNN, arbre de décision), comme des mécanismes pour organiser intelligemment une circulation typique.

Dans l'étape de réalisation, nous avons eu l'opportunité de nous familiariser avec une panoplie d'outils récents telles que, Python, Anaconda, spyder, netlogo et des bibliothèque utilisée Numpy, Matplotlib, Pandas et Scikit-learn.

Notre effort sert à ouvrir une fenêtre sur les problèmes de gestion de circulation routière existante et de contribuer avec des solutions basées sur l'apprentissage automatique, une contribution qui nous semblent prometteurs et encourageants pour une application réelle. Comme perspective de notre travail, il serait intéressant d'approfondir cette recherche à court terme avec d'autre volet d'apprentissage automatique tels que les l'approches non supervisés. et à long terme avec l'apprentissage en profondeur (Deep-Learning).

## Bibliographie

- [1] : Doniec A., Espié S., Mandiau R. et Piechowiak S., Simulation de trafic routier en carrefour : gestion des comportements et des priorités. In International Workshop Logistique &Transport, Hammamet 2006.
- [2] : Yagar S., Corridor control system. Concise Encyclopaedia of Traffic and Transportation System, Editor Papageorgiou M., 1991, 79-83.
- [3] : Pipes L,A, An Operational Analysis of Traffic Dynamics. J. Appl. Phys., Vol. 3, n°24,1953, 274-281.
- [4] : Newell G.F. Mathematical models for freely flowing highway traffic. Operation Research, n°3, 1955, 176-186.
- [5] : Webster F.V., Traffic signal setting. Road Research Technical Paper, n° 39, Road Research Laboratory, London, 1958.
- [6] : Cohen S., Ingénierie du trafic routier: éléments de théorie du trafic et applications. Presses de l'Ecole Nationale des Ponts et Chaussées, 1990.
- [7] : Webster F.V., Traffic signal setting. Road Research Technical Paper, n° 39, Road Research Laboratory, London, 1958.
- [8] : Chandler F.E., Herman R., Montroll E.W., Traffic dynamics: studies in car following. Operation Research, n°6, 1958. , 165-184.
- [9] : Pierrich A. and Kevin V., Amélioration de la circulation routière à un carrefour. TER 2010, Promotion 50, Paris 2010.
- [10] : Papageorgiou M., Méthodes de contrôle automatique dans les transports et la gestion du trafic. Génie logiciel n° 75, 2006. , 13-25.
- [11] : Boillot F., Blosserville J., Lesort J., Motyka V., Papageorgiou M. and Sellam S.,Optimal signal control of urban traffic networks. 6th International Conference on Road Traffic Monitoring and Control, London 1992, 75-79.
- [12] : Henry J.J., The PRODYN Real time Traffic Algorithm. 4th IFAC/IFIP/IFORS International Conference on Control in Transportation Systems, 1983, 307-311.
- [13] : <https://www.djazairess.com/fr/letemps/47827> visite le 12/07/2021
- [14] : Mohamed Tlig, Coordination locale et optimisation distribuée du trafic de véhicules autonomes dans un réseau routier, Ecole doctorale IAEM Lorraine, 26 mars 2015.
- [15] : <http://www.psychomedia.qc.ca/lexique/definition/apprentissage-automatique> visite le 20/07/2021
- [16] : A. Cornuéjols, L. Miclet, Y.Kodratoff, " Apprentissage Artificiel, Concepts et algorithmes " ISBN 2-212-11020-0 , 20

- [17] : <https://www.lemagit.fr/conseil/Machine-Learning-comment-choisir-le-bon-framework> visite le 28/07/2021
- [18] : [www.intelligenttransport.com](http://www.intelligenttransport.com) visite le 02/08/2021
- [19] : <https://www.researchgate.net/publication/330815113> Speech Recognition Using Deep Neural visite le 15/08/2021
- [20] : <https://fr.education-wiki.com/> visite le 15/08/2021
- [21] : <https://moncoachdata.com/> visite le 15/08/2021
- [22] : Mohamadally Hasan et Fomani Boris," SVM : Machines à Vecteurs de Support ou Separateurs a Vastes Marges".Thèse. Versailles St Quentin, Francejanvier 2006.
- [23] : hassane hilali, Application de la classification textuelle pour l'extraction des regles d'association maximales, avril 2009.
- [24] : Kontonatsios, G., Korkontzelos, Y., Tsujii, J. I., & Ananiadou, S. Using a random forest classifier to compile bilingual dictionaries of technical terms from comparable corpora. In 14th Conference of the European Chapter of the Association for Computational Linguistics, (2014, April).
- [25] : David Broneske. On the Impact of Hardware on Relational Join Processing. Master thesis University of Magdeburg 19 aout 2013.
- [26] : Zakariya , ismail . . « pour quoi python il est meilleur. » analytics & insights,<https://analyticsinsights.io/pourquoi-python-est-populaire-en-data-science/>
- [27] : Anaconda Individual Edition <https://www.anaconda.com/products/individual>.
- [28] : Voilà, des dashboards à partir de vos Jupyter Notebooks.
- [29] : D. Harel et A. Pnueli. « On the development of reactive systems ». In: Logics and models of concurrent systems. 1985, p. 477–498.
- [30] : B. Cayla. Python Pandas – Tuto <https://pandas.pydata.org/> (Partie N°1).
- [31] : E. F.M. D. John Hunter Darren Dale. Lancement de l'initiative scikit-learn, bibliothèque logicielle de référence en machine learning (cf. p. 57).
- [32] : Priyankur ,sarkar .« pour quoi python il est meilleur. », What is LDA: Linear Discriminant Analysis for Machine Learning ,knowledge hut,30/09/2019 , <https://www.knowledgehut.com/blog/data-science/linear-discriminant-analysis-for-machine-learning>
- [33] : Paulino jose garcia nieto.« Classification of data by support vector machine (SVM) »,hard-Rock Stability Analysis for Span Design in Entry-Type Excavations with Learning Classifiers , June 2016,<https://www.researchgate.net/publication/>

[34] :[https://ichi.pro/fr/13-algorithmes-de-classification-d-apprentissage-automatique-pour la-science-des-donnees-et-leur-code-](https://ichi.pro/fr/13-algorithmes-de-classification-d-apprentissage-automatique-pour-la-science-des-donnees-et-leur-code-) visite le 20/08/2021

[35] : Gouchan Nour El Houda, Mémoire de master, Université Ibn Khaldoun ,Département Informatique, Spécialité Génie Logiciel 2016.