République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Kasdi Merbah Ouargla

Faculté des Nouvelles Technologies de l'Information et de la Communication

Département d'Informatique

Mémoire de Master

en Informatique

Spécialité : *informatique industriel*

Thème

Une application web pour la prédiction précoce du diabète basant sur les algorithmes d'apprentissage automatique

Encadré par

Dr : Abdelmadjid Youcefa

Réalisé par

Ing : Mayou NasserEddine

Ing :Belhachani Mohammed

2020/2021

Remerciements

Nous remercions le DIEU de nous avoir donné la patience, la santé et le courage pour réaliser ce travail. A travers ce modeste travail, nous tenons à remercier vivement notre encadreur **Dr : Abdelmadjid Youcefa** *pour ses conseils et ses encouragements qui nous ont permis de réaliser ce modeste travail Nous remercions sincèrement les membres du jury d'avoir accepté d'examiner et d'évaluer notre travail . Nous exprimons également notre gratitude à tous les professeurs et les enseignants qui ont collaboré à notre formation depuis notre premier cycle d''étude jusqu"à la fin de notre cycle universitaire. MERCI A TOUS*

Dédicaces

Nous dédions ce travail à nos parents à toute la famille. À tous les amis et bien sûr à tout le collègue durant la formation.

Nous dédions ce travail à tous les jours difficiles que nous avons vécu durant notre formation les cinq années de l'ingéniorat plus l'année de master

Résumé

Au cœur de ce mémoire, nous avons conçu et développé une application web pour la prédiction précoce du diabète de type 2, Afin d'éviter les risques de complications de cette maladie chronique sur la santé du patient. Pour atteindre cet objectif, nous avons utilisé des algorithmes d'apprentissage automatique supervisé (K neares neighbors, Decision Trees, Random Forest, Support Vector Machine, Naives Bayes) et le data set extrait de l'adresse web www.kaggle.com le data est de l'origine de l'hopital de Pima_Indian (USA)Les performances des classifieurs ont été comparées en fonction du taux de précision. Les plus hauts taux de classification obtenus par l'application de **Support_Vector_ Machine** et Random_**Forest** sont respectivement **81.16**% et **79.22**% en appliquant la méthode d'´évaluation train/test

Mots clés: IA , ML , Prédiction du diabète ,K nearest neighbors, Decision Trees,Random Forest, Support Vector Machine, Naives Bayes

Abstract

In this modest work, we designed and developed a web application for the early prediction of type 2 diabetes, in order to avoid the risk of complications of this disease on the patient's health .To achieve this goal, we used algorithms supervised machine learning (K nearest neighbors, Decision Trees, Random Forest, Support Vector Machine, Naives Bayes) and the data set extracted from the web address www.kaggle.com the data is from the Pima_Indian hospital (USA). The performance of classifiers was compared based on accuracy rate and model sensitivity. The highest classification rates obtained by the application of **Support_Vector_ Machine** and **Random Forest** are respectively **81.16**% and **79.22**%, by applying the method of evaluation train /test .

Key words : IA , ML , Diabetes prediction ,K nearest neighbors, Decision Trees, Random Forest, Support Vector Machine, Naives Bayes.

Table des matières

Table des figures

Liste des tableaux

Liste des abréviations

IA	Intelligence Artificielle
ML	Machine Learning
OMS	Organisation mondiale de la santé.
Ceed	Centre européen d´étude du diabète.
ASG	Auto surveillance glycémique
HAS	Haute autorité de santé
DT1	Diabète de type 1
DT2	Diabète de type 2

Introduction générale

L’apprentissage automatique est une discipline de l’intelligence artificielle qui cherche a trouver un moyen de créer des programmes informatiques qui s’améliorent automatiquement avec l’expérience. à travers ce mémoire de Master, nous intéresserons à l’utilisation des algorithmes d’apprentissage automatique pour la prédiction précoce du diabète de type 2 qui est un dysfonctionnement du système de régulation de la glycémie, Afin d'éviter les risques de complications de cette maladie chronique sur la santé du patient

Notre problématique nous permet de définir le diagnostic médical comme un processus de classification et l’utilisation de l’informatique devient de plus en plus fréquente pour mettre en œuvre cette classification bien que la décision de médecin soit le facteur le plus important dans le diagnostic. Les systèmes de classification sont d’une grande aide car ils réduisent les erreurs dues à la fatigue et au temps nécessaire au diagnostic.

La méthode utilisée dans ce travail est l’application des différents algorithmes de classification d’apprentissage supervisé

- K nearest neighbors
- Decision Trees
- Random Forest
- Support Vector Machine
- naïveBayes

aux données extrait de l'hôpital de Pima_Indian (USA) téléchargés à partir de l'adresse www.kaggle.com et déduire le meilleur algorithme qui donnera comme résultat une classification des patients en termes de taux du précision et de la sensibilité du modèle . Ce travail est organisé en trois principaux chapitres comme suit :

1. Le 1er chapitre présente un aperçu général sur la maladie du diabète, leurs différents types, les symptômes ainsi que le diagnostic et le traitement de la maladie et à la fin quelques préventions pour éviter le diabète.
2. Le 2éme chapitre donne un aperçu sur l'apprentissage automatique les algorithmes d'apprentissage supervisé qui peuvent nous aider à détecter l'apparition précoce du diabète et la méthode suivi pour manipuler les données afin de prédite du diabète type 2.
3. Le dernier chapitre présente d'abord une étude technique dans laquelle nous définissons l'environnement logiciel utilisé pour construire notre application, puis une définition et une analyse détaillé de la base des données utilisées. Ensuite, les résultats sont présentés, comparés et interprétés.

   Finalement nous finissons par une représentation des interfaces d'application d'apprentissage dans la prédiction du diabète type 2.

A la fin, ce travail est clôturé par une conclusion générale

Chapitre 1 : Généralités sur le diabète

1.1 Introduction

Le diabète, souvent appelé par les médecins diabète sucré, décrit un groupe de maladies métaboliques dans lesquelles la personne a une glycémie élevée (sucre dans le sang), soit parce que l'insuline la production est inadéquate, ou parce que les cellules du corps ne le font pas répondre correctement à l'insuline, ou aux deux. Les patients avec la glycémie présentera généralement une polyurie (fréquente miction), ils auront de plus en plus soif (polydipsie) et affamé (polyphagie).

1.2 La signification et l'origine du diabète sucré:

Le diabète vient du grec, et cela signifie un «siphon». Aretusle appadoce,

médecin grec au IIe siècle A.D., a nommé la condition diabainein. Il a décrit les patients qui passaient trop d'eau (polyurie) - comme un siphon. Le mot est devenu «diabète» à partir de l'adoption anglaise du diabète latin médiéval. En 1675, Thomas Willis ajouta mellitus au terme, bien qu'il soit communément appelé simplement comme le diabète. Mel en latin signifie «miel»; l'urine et le sang des personnes atteintes de diabète a un excès de glucose, et le glucose est doux comme le miel. Le diabète sucré pourrait littéralement signifie «siphonner de l'eau douce». Dans la Chine ancienne, les gens ont observé que les fourmis seraient attirées par certaines personnes l'urine, parce qu'elle était douce. Le terme «maladie des urines sucrées» a été inventé. [1]

1.3 Quelques aspects clés du diabète

• Le diabète est une maladie à long terme qui entraîne des taux de sucre dans le sang.

• En 2013, on estimait que plus de 382 millions de personnes partout dans le monde souffraient de diabète.

• Diabète de type 1 - le corps ne produit pas insuline. Environ 10% de tous les cas de diabète sont type 1.

• Diabète de type 2 - le corps ne produit pas suffisamment d'insuline pour un bon fonctionnement. Environ 90% de tous les cas de diabète dans le monde en sont taper.

• Diabète gestationnel - ce type affecte les femmes pendant la grossesse.

• Les symptômes du diabète les plus courants comprennent mictions fréquentes, soif et faim intenses, gain de poids, perte de poids inhabituelle, fatigue, coupures et ecchymoses qui ne guérissent pas, dysfonctionnement sexuel masculin, engourdissement et picotements dans les mains et les pieds.

• Si vous avez le type 1 et suivez un régime alimentaire sain, faites de l'exercice adéquat et prenez de l'insuline, vous pouvez conduire une vie normale.

• Les patients de type 2 doivent manger sainement, être physiquement active et testez leur glycémie. Ils peuvent aussi besoin de prendre des médicaments oraux et / ou de l'insuline pour contrôler la glycémie. [2]

1. 4 Types de diabète

1.4.1 Diabète de type 1:

Le corps ne produit pas d'insuline. Certaines personnes peuvent qualifier ce type d'insulino-dépendant diabète, diabète juvénile ou diabète précoce. Les gens développent généralement un diabète de type 1 avant leur 40^{e} année, souvent au début de l'âge adulte ou à l'adolescence. Type 1 le diabète est loin d'être aussi répandu que le diabète de type 2. Environ 10% tous les cas de diabète sont de type 1. Les patients atteints de diabète de type 1 devront prendre de l'insuline injections pour le reste de leur vie. Ils doivent également assurer glycémie adéquate en effectuant des tests sanguins et suivre un régime spécial.

1.4.2 Diabète de type 2:

Le corps ne produit pas assez l'insuline pour un bon fonctionnement, ou les cellules du corps ne réagit pas à l'insuline (résistance à l'insuline). Environ 90% de tous les cas de diabète dans le monde sont de type 2. Certains les personnes peuvent être en mesure de contrôler leur diabète de type 2 symptômes en perdant du poids, en suivant une alimentation saine, faire beaucoup d'exercice et surveiller leur sang les niveaux de glucose. Cependant, le diabète de type 2 est généralement un maladie évolutive – elle s'aggrave progressivement - et le patient devra probablement prendre de l'insuline, généralement sous forme de comprimés. Les personnes en surpoids et obèses ont un risque beaucoup plus élevé de développer un diabète de type 2 par rapport à ceux qui ont un poids corporel sain. Gens avec beaucoup de graisse

viscérale, également appelée obésité centrale, la graisse du ventre ou l'obésité abdominale sont

Particulièrement à risque. Le surpoids / l'obésité provoque la libération du corps produits chimiques qui peuvent déstabiliser le système cardiovasculaire du corps et les systèmes métaboliques. Être en surpoids,

Physiquement inactif et manger le les mauvais aliments contribuent tous à notre risque de développer un type 2 diabète. Les scientifiques estiment que l'impact des boissons gazeuses sucrées à risque de diabète peut être direct, plutôt que simplement une influence sur le poids corporel. le risque de développer un diabète de type 2 est également plus vieillir. Les experts ne savent pas vraiment pourquoi, mais disent qu'en vieillissant, nous avons tendance à prendre du poids et à devenir moins actif physiquement. Ceux qui ont un parent proche qui avaient / avaient un diabète de type 2, des personnes du Moyen-Orient, La descendance africaine ou sud-asiatique présente également un risque plus élevé de développer la maladie. Hommes dont les niveaux de testostérone sont faibles et présentent un risque plus élevé de développer un diabète de type 2.

1.4.3 **Diabète gestationnel:**

Ce type affecte les femmes pendant grossesse. Certaines femmes ont des niveaux très élevés de glucose dans leur sang et leur corps est incapable de

Produire suffisamment d'insuline pour transporter tout le glucose dans leurs cellules, ce qui entraîne une augmentation progressive des niveaux de glucose. Le diagnostic du diabète gestationnel est posé pendant la grossesse. La majorité des diabètes gestationnels les patients peuvent contrôler leur diabète grâce à l'exercice et à un régime. Entre 10 et 20% d'entre eux devront en prendre sorte de médicaments contrôlant la glycémie. Un diabète gestationnel non diagnostiqué ou non contrôlé peut augmenter le risque de complications lors de l'accouchement. [3]

1.5 Qu'est-ce que le prédiabète:

La grande majorité des patients le diabète de type 2 avait initialement un prédiabète. Leur sang les niveaux de glucose sont supérieurs à la normale, mais pas élevés suffisamment pour mériter un diagnostic de diabète. Les cellules du corps deviennent résistantes à l'insuline.

1.6 Le diabète est un trouble du métabolisme

Le diabète (diabète sucré) est classé comme un métabolisme désordre. Le métabolisme fait référence à la façon dont notre corps utilise nourriture digérée pour l'énergie et la croissance. La plupart de ce que nous mangeons se décompose en glucose. Le glucose est une forme de sucre dans le sang - c'est la principale source de carburant pour notre corps. Lorsque notre nourriture est digérée, le glucose fait son chemin dans notre circulation sanguine. Nos cellules utilisent le glucose pour produire de l'énergie et la crosans insuline - l'insuline permet nos cellules pour absorber le glucose. L'insuline est une hormone produit par le pancréas. Après avoir mangé, le pancréas libère automatiquement une quantité suffisante d'insuline pour déplacer le glucose présent dans notre sang dans les cellules, dès au fur et à mesure que le glucose pénètre dans les cellules, la glycémie diminue. Une personne diabétique a une condition dans laquelle le la quantité de glucose dans le sang est trop élevée (hyperglycémie). C'est parce que le corps ne produit pas suffisamment d'insuline, ne produit pas d'insuline ou a des cellules qui ne répondre correctement à l'insuline produite par le pancréas. Cette entraîne une accumulation excessive de glucose dans le sang. Cet excès de glucose sanguin finit par sortir du corps en urine. Ainsi, même si le sang contient beaucoup de glucose, le les cellules ne l'obtiennent pas pour leur énergie et leur croissance essentielles conditions. Comment déterminer si vous souffrez de diabète, de prédiabète ou ni l'un ni l'autre Les médecins peuvent déterminer si un patient a un métabolisme, prédiabète ou diabète dans l'un des trois manières (tests):

❖ Le test A1C

- au moins 6,5% signifie diabète

- entre 5,7% et 5,99% signifie prédiabète

- moins de 5,7% signifie normal

❖Le test FPG (glucose plasmatique à jeun)

- au moins 126 mg / dl signifie diabète

- entre 100 mg / dl et 125,99 mg / dl signifie prédiabète

- moins de 100 mg / dl signifie normal Une lecture anormale après le FPG signifie que le patient a une glycémie à jeun altérée (IFG) issance. Cependant, le glucose ne peut pas entrer dans nos cellules

❖L'OGTT (test oral de tolérance au glucose)

- au moins 200 mg / dl signifie diabète

- entre 140 et 199,9 mg / dl signifie prédiabète

- moins de 140 mg / dl signifie normal Une lecture anormale suite à l'OGTT signifie que le patient a une tolérance au glucose altérée (IGT)

[4]

1.7 La liste des complications possibles qui peuvent être causée par un diabète mal contrôlé :

❖**Complications oculaires** - glaucome, cataracte, diabétique rétinopathie et quelques autres.

❖**Complications du pied** - neuropathie, ulcères et parfois la gangrène qui peut exiger que le pied être amputé

Complications cutanées - les personnes atteintes de diabète sensible aux infections cutanées et aux troubles cutanés

Problèmes cardiaques - tels que les cardiopathies ischémiques, lorsque l'apport sanguin au muscle cardiaque est diminué

Hypertension - fréquente chez les personnes atteintes de diabète, ce qui peut augmenter le risque de maladie rénale, oculaire problèmes, crise cardiaque et accident vasculaire cérébral

Santé mentale - le diabète incontrôlé augmente le risque de souffrir de dépression, d'anxiété et autres troubles mentaux

Perte auditive - les patients diabétiques ont un risque plus élevé de développer des problèmes d'audition

Maladie des gencives - la prévalence est beaucoup plus élevée des maladies des gencives chez les patients diabétiques

Gastroparésie - les muscles de l'estomac s'arrêtent fonctionner correctement

Acidocétose - une combinaison de cétose et acidose; accumulation de corps cétoniques et acidité Dans le sang.

Neuropathie - la neuropathie diabétique est un type de nerf des dommages pouvant entraîner plusieurs problèmes.

HHNS (Hyperosmolar Hyperglycemic Nonketotic Syndrome) - la glycémie augmente trop, et il n'y a pas de cétones présentes dans le sang ou urine. C'est une situation d'urgence. Néphropathie - une tension artérielle incontrôlée peut conduire à une maladie rénale

MAP (maladie artérielle périphérique) - les symptômes peuvent comprennent des douleurs dans la jambe, des picotements et parfois des problèmes pour marcher correctement

Accident vasculaire cérébral - si tension artérielle, taux de cholestérol et la glycémie n'est pas contrôlée, le risque de l'AVC augmente considérablement

Dysfonction érectile - impuissance masculine.

Infections - personnes atteintes de diabète mal contrôlé sont beaucoup plus sensibles aux infections

Cicatrisation des plaies - les coupures et les lésions prennent beaucoup plus longtemps pour guérir.

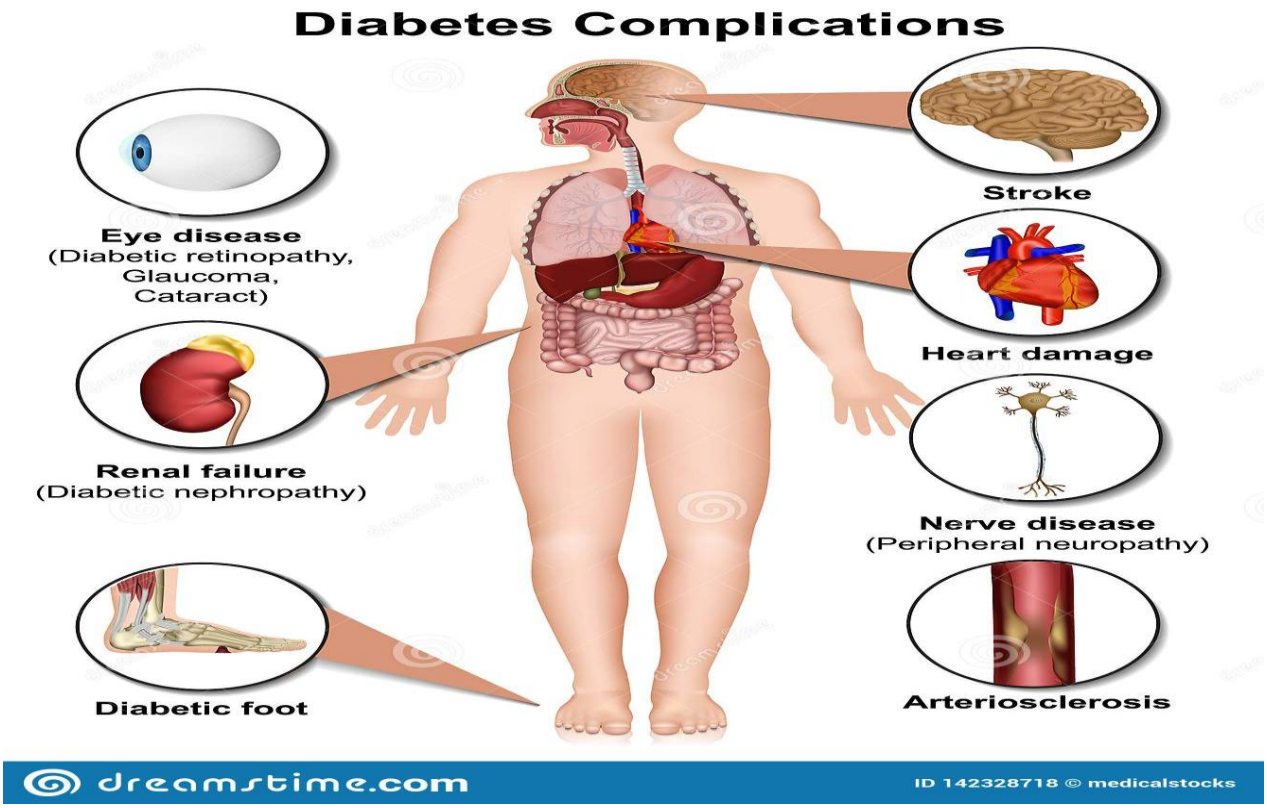


Figure 1.1 -Les Complications du diabète [5]

1.8 Quelques faits et mythes sur le diabète:

De nombreux Des «faits» sont évoqués dans la presse papier, les magazines et sur Internet concernant le diabète ; certains d'entre eux sont, en fait, mythes. Il est important que les personnes atteintes de diabète, de pré diabète, leurs proches, les employeurs et les écoles image précise de la maladie.

Quelques diabètes mythes:

Les personnes atteintes de diabète ne doivent pas faire d'exercice -

Faux ! L'exercice est important pour les personnes diabète, comme pour tout le monde. L'exercice aide gérer le poids corporel, améliore le système cardiovasculaire santé, améliore l'humeur, aide à contrôler la glycémie,

et soulage le stress. Les patients doivent discuter de l'exercice avec leur médecin en premier.

Les personnes grasses développent toujours un diabète de type 2 finalement - ce n'est pas vrai. Être en surpoids ou obèses augmente le risque de devenir diabétique, ils sont facteurs de risque, mais ne signifie pas qu'une personne obèse deviendra certainement diabétique. Beaucoup de gens avec le diabète de type 2 n'a jamais été en surpoids. Le la majorité des personnes en surpoids ne développent pas de type 2 diabète.

Le diabète est une nuisance, mais pas grave - les deux tiers des patients diabétiques meurent prématurément d'un accident vasculaire cérébral ou cardiopathie. L'espérance de vie d'une personne le diabète est de cinq à dix ans plus court que les autres les gens. Le diabète est une maladie grave.

Les enfants peuvent surmonter le diabète - ce n'est pas vrai.

Presque tous les enfants atteints de diabète ont le type 1 , les cellules bêta productrices d'insuline dans le pancréas ont été détruit. Ceux-ci ne reviennent jamais. Enfants diabétiques de type 1 devront prendre de l'insuline pendant le reste de leur vie, à moins qu'un remède ne soit trouvé journée

Ne mangez pas trop de sucre, vous deviendrez diabétique - ce n'est pas vrai. Une personne diabétique le type 1 a développé la maladie parce que leur système immunitaire système détruit les cellules bêta productrices d'insuline. Une alimentation riche en calories, qui peut rendre les gens surpoids / obèse, augmente le risque de développement de diabète de type 2, surtout s'il y a des antécédents de cette maladie dans la famille.

Je sais quand ma glycémie est élevée ou faible – des taux de sucre dans le sang très élevés ou faibles peuvent causer certains symptômes, tels que faiblesse, fatigue et soif extrême. Cependant, les niveaux doivent être fluctuant beaucoup pour que les symptômes se fassent sentir. Le seul Pour être sûr de votre taux de sucre dans le sang, testez-les régulièrement. Des chercheurs du L'Université de Copenhague, au Danemark, a montré que même de très légères augmentations de la glycémie augmentent considérablement le risque de cardiopathie ischémique.

Les régimes diabétiques sont différents de ceux des autres personnes -les diététiciens et nutritionnistes spécialisés recommander aux patients diabétiques sont en bonne santé ; sain pour tout le monde, y compris les personnes sans la maladie. Les repas doivent contenir beaucoup de légumes, fruits, grains entiers, et ils devraient être faible en sel et en sucre, et gras saturés ou trans. Les experts disent qu'il n'est pas nécessaire d'acheter d'aliments diabétiques car ils n'offrent aucun avantage particulier, par rapport aux produits sains dans lesquels nous pouvons acheter la plupart magasins.

Une glycémie élevée convient à certains, alors que pour d'autres, ils sont un signe de diabète - des taux élevés de sucre dans le sang ne sont jamais normaux pour personne. Quelque les maladies, le stress mental et les stéroïdes peuvent causer augmentations temporaires de la glycémie chez les personnes sans diabète. Toute personne dont le taux de sucre dans le sang ou le taux de sucre dans l'urine est supérieur à la normale doit être vérifiée pour le diabète par un centre de santé professionnel. Les diabétiques ne peuvent pas manger de pain, de pommes de terre ou de pâtes - les personnes atteintes de diabète peuvent manger des féculents.

Cependant, ils doivent garder un œil sur la quantité. Les féculents à grains entiers sont meilleurs, car c'est le cas des personnes sans diabète.

Une personne peut transmettre le diabète à une autre personne - PAS VRAI. Tout comme une jambe cassée n'est pas infectieux ou contagieux. Un parent peut transmettre, à travers leurs gènes à leur progéniture, une susceptibilité de développer la maladie.

Seules les personnes âgées développent un diabète de type 2 – choses changent. Un nombre croissant d'enfants et les adolescents développent un diabète de type 2. Experts dire que cela est lié à l'explosion de l'enfance taux d'obésité, mauvaise alimentation et sédentarité.

Je dois prendre de l'insuline, cela doit signifier mon diabète est grave - les gens prennent de l'insuline lorsqu'ils suivent un régime seul ou avec un régime avec injection orale ou non d'insuline les médicaments contre le diabète ne fournissent pas assez le contrôle du diabète, c'est tout. L'insuline aide le diabète contrôlé. Il n'a généralement rien à faire avec la gravité de la maladie.

Si vous êtes diabétique, vous ne pouvez pas manger de chocolats ou bonbons - les personnes atteintes de diabète peuvent manger des chocolats

et des bonbons s'ils les combinent avec de l'exercice ou mangez-les dans le cadre d'un repas sain.

Les patients diabétiques sont plus sensibles au rhumeet les maladies en général - une personne diabétique avec un bon contrôle du diabète n'est pas plus susceptible de tomber malade d'un rhume ou autre chose gens. Cependant, lorsqu'un diabétique attrape un rhume, leur diabète devient plus difficile à contrôler, alors ils ont un risque plus élevé de complications. Symptômes du diabète : les gens peuvent souvent souffrir de diabète et être complètement inconscient. La raison principale en est que les symptômes, vus seuls, semblent inoffensifs. Cependant, plus le diabète est diagnostiqué tôt, plus le il y a des chances que des complications graves, qui peuvent résulter d'avoir le diabète, peut être évité.

1.9 Liste des symptômes du diabète les plus courants:

Mictions fréquentes: êtes-vous allé au salle de bain pour uriner plus souvent récemment? Est-ce que tu remarques que vous passez la majeure partie de la journée à toilette ? Lorsqu'il y a trop de glucose (sucre) dans votre sang, vous urinerez plus souvent. Si ton l'insuline est inefficace, ou pas du tout, votre les reins ne peuvent pas filtrer le glucose dans le du sang. Les reins prélèveront de l'eau de votre sang afin de diluer le glucose - qui à son tour remplit votre vessie.

Soif disproportionnée : si vous urinez plus que d'habitude, vous devrez remplacer ce liquide perdu. Vous boirez plus que d'habitude. Avez-vous bu plus que d'habitude ces derniers temps?

Faim intense: comme l'insuline dans votre sang n'est pas fonctionné correctement, ou n'existe pas du tout, et votre les cellules ne reçoivent pas leur énergie, votre corps peut réagir en essayant de trouver plus d'énergie - de la nourriture. Vous serez avoir faim.

Gain de poids: cela peut être le résultat de ce qui précède symptôme (faim intense).

Perte de poids inhabituelle: elle est plus courante chez les personnes atteintes de diabète de type 1. Comme votre corps n'est pas en fabriquant de l'insuline, il cherchera une autre énergie source (les cellules ne reçoivent pas de glucose). Muscle les tissus et la graisse seront décomposés pour produire de l'énergie. Comme Le type 1 est d'apparition plus soudaine et le type 2 est beaucoup plus graduelle, la perte de poids est plus perceptible avec Type 1.

Augmentation de la fatigue : si votre insuline ne fonctionne pas correctement, ou n'y est pas du tout, le glucose ne sera pas entrer dans vos cellules et leur fournir énergie. Cela vous fera vous sentir fatigué et apathique.

Irritabilité: l'irritabilité peut être due à votre manque de énergie.

Vision floue : cela peut être causé par des tissus tiré de vos lentilles oculaires. Cela affecte vos yeux » capacité à se concentrer. Avec un traitement approprié, cela peut être traité. Il existe des cas graves de cécité ou de des problèmes de vision prolongés peuvent survenir. Les coupures et les ecchymoses ne guérissent pas correctement ou rapidement : Trouvez-vous que les coupures et les ecchymoses prennent beaucoup plus de temps que d'habitude pour guérir ? Quand il y a plus de sucre (glucose) dans votre corps, sa capacité à guérir peut être miné.

Plus d'infections cutanées et / ou à levures : plus de sucre dans votre corps, sa capacité à récupérer les infections sont affectées. Les femmes atteintes de diabète le trouvent particulièrement difficile à récupérer de la vessie et infections vaginales.

Démangeaisons cutanées: une sensation de démangeaisons sur votre peau parfois un symptôme du diabète.

Les gencives sont rouges et / ou en fées - Les gencives se détachent

des dents: si vos gencives sont sensibles, rouges et / ou enflé cela pourrait être un signe de diabète. Vos dents pourraient se détacher lorsque les gencives s'écartent de eux.

1.10 les traitement du diabète

1.10.1 Le traitement du diabète de type 1

Pour compenser, celle-ci doit être administrée artificiellement au quotidien par une injection sous cutanée d'insuline via une seringue, un stylo ou une pompe. Il s'agit d'un traitement d'insulinothérapie. Le diabète de type 1 touche plus souvent l'enfant, l'adolescent voire le jeune adulte.

Figure 1.2 - Seringue d'insuline [6]

1.10.2 Le traitement du diabète de type 2

Le traitement repose prioritairement sur des aliments équilibrée et pratique d'une activité physique régulière .si ces deux éléments sont insuffisants, il faudra ajouter un traitement par anti-diabétique oral. Le traitement à l'insuline peut s'avérer nécessaire, si les glycémies restent néanmoins élevées.[4]

Figure 1.3 - Le traitement hygiéno-diététiques [7]

1.10.3 Le traitement du diabète gestationnel

Selon le Centre européen d'étude du Diabète (Ceed) le traitement par le recours à l'insuline est nécessaire dans 50% des cas et dans quelques cas plus rares un traitement par anti-diabétique oral peut être mis en place. Dans tous les cas, des mesures hygiénodiététiques doivent rapidement être mises en place, avec la particularité qu'elles doivent prendre en compte `a la fois le diabète de la mère et les besoins nutritionnels du fœtus.[4]

1.11 **Conclusion**

Dans ce chapitre nous avons présenté la maladie du diabète, leur différent types, les symptômes ainsi que le diagnostic et le traitement de la maladie et a la fin nous avons cité quelques préventions pour éviter le diabète. Dans le prochain chapitre, nous présenterons des approches différentes d'aide au diagnostic préventif concernant les algorithmes de machine learning dans la prédiction du diabète de type 2

chapitre 2 : l'apprentissage automatique

2.1 Introduction

L'apprentissage automatique (ou artificiel)(*machine-learning*en anglais) est un des champs d'étude de l'intelligence artificielle. Commençons par la définition de l'AI et celle fournie dans l"avant-propos de (Cornuéjols*et al.*,2002)

Dans ce chapitre on va voir l'apprentissage automatique les algorithmes d'apprentissage supervisé qui peuvent nous aider a détecter l'apparition précoce du diabète et la méthode suivi pour manipuler les données afin de prédite du diabète type 2

C'est pour ça on va voir comment faire pour appliquer les différents algorithmes de classification d'apprentissage supervisé

- K nearest neighbors
- Decision Trees
- Random Forest
- Support Vector Machine
- naïveBayes

2.3 **L'apprentissage Automatique**

L'apprentissage automatique comporte généralement deux phases. La première consiste à estimer un modèle à partir de données, appelées observations, qui sont disponibles et en nombre fini, lors de la phase de conception du système. L'estimation du modèle consiste à résoudre une tâche pratique, telle que traduire un discours, estimer une densité de probabilité, reconnaître la présence d'un chat dans une photographie ou participer à la conduite d'un véhicule autonome. Cette phase dite « d'apprentissage » ou « d'entraînement » est généralement réalisée préalablement à l'utilisation pratique du modèle. La seconde phase correspond à la mise en production : le modèle étant déterminé, de nouvelles données peuvent alors être soumises afin d'obtenir le résultat correspondant à la tâche souhaitée. En pratique, certains systèmes peuvent poursuivre leur apprentissage une fois en production, pour peu qu'ils aient un moyen d'obtenir un retour sur la qualité des résultats produits.

Selon les informations disponibles durant la phase d'apprentissage, l'apprentissage est qualifié de différentes manières. Si les données sont étiquetées (c'est-à-dire que la réponse à la tâche est connue pour ces données), il s'agit d'un apprentissage supervisé. On parle de classification ou de classement[3] si les étiquettes sont discrètes, ou de régression si elles sont continués. Si le modèle est appris de manière incrémentale en fonction d'une récompense reçue par le programme pour chacune des actions entreprises, on parle d'apprentissage par renforcement. Dans le cas le plus général, sans étiquette, on cherche à déterminer la structure sous-jacente des données (qui peuvent être une densité de probabilité) et il s'agit alors d'apprentissage non supervisé. L'apprentissage automatique peut être appliqué à différents types de données, tels des graphes, des arbres, des courbes, ou plus simplement des vecteurs caractéristiques, qui peuvent être des variables qualitatives ou quantitatives continues ou discrètes.

L'intelligence artificielle qui se fonde sur des approches mathématiques et statistiques pour donner aux ordinateurs la capacité d'« apprendre » à partir de données, c'est-à-

dire d'améliorer leurs performances à résoudre des tâches sans être explicitement programmés pour chacune. Plus largement, il concerne la conception, l'analyse, l'optimisation, le développement et l'implémentation de telles méthodes. [9]

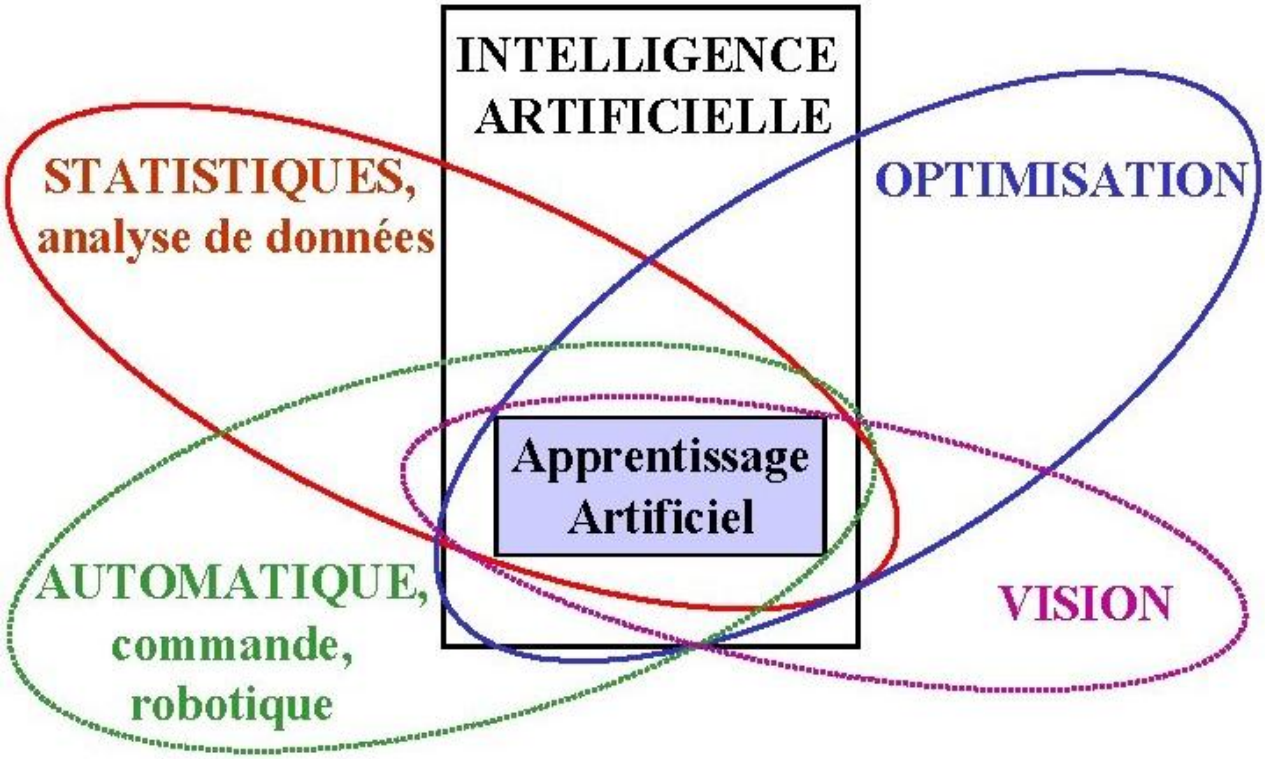


Figure 2.1 –L'apprentissage automatique avec les disciplines connexe [10]

2.3.1 Motivation

- Certaines tâches sont difficiles à programmer manuellement. Reconnaissance de formes, Traduction par machine, Reconnaissance de la parole, Aide à la décision, etc.
- Les données sont disponibles, qui peuvent être utilisé pour estimer la fonction de notre tâche.

2.3.2Domaines d'applications de l'apprentissage automatique :

L'apprentissage automatique s'applique à un grand nombre d'activités humaines et convient en particulier au problème de la prise de décision automatisée. Il s'agira, par exemple :

- D'établir un diagnostic médical à partir de la description clinique d'un patient ;
- De donner une réponse à la demande de prêt bancaire de la part d'un client sur la base de sa situation personnelle ;
- De déclencher un processus d'alerte en fonction de signaux reçus par des capteurs ;
- De la reconnaissance des formes [11]

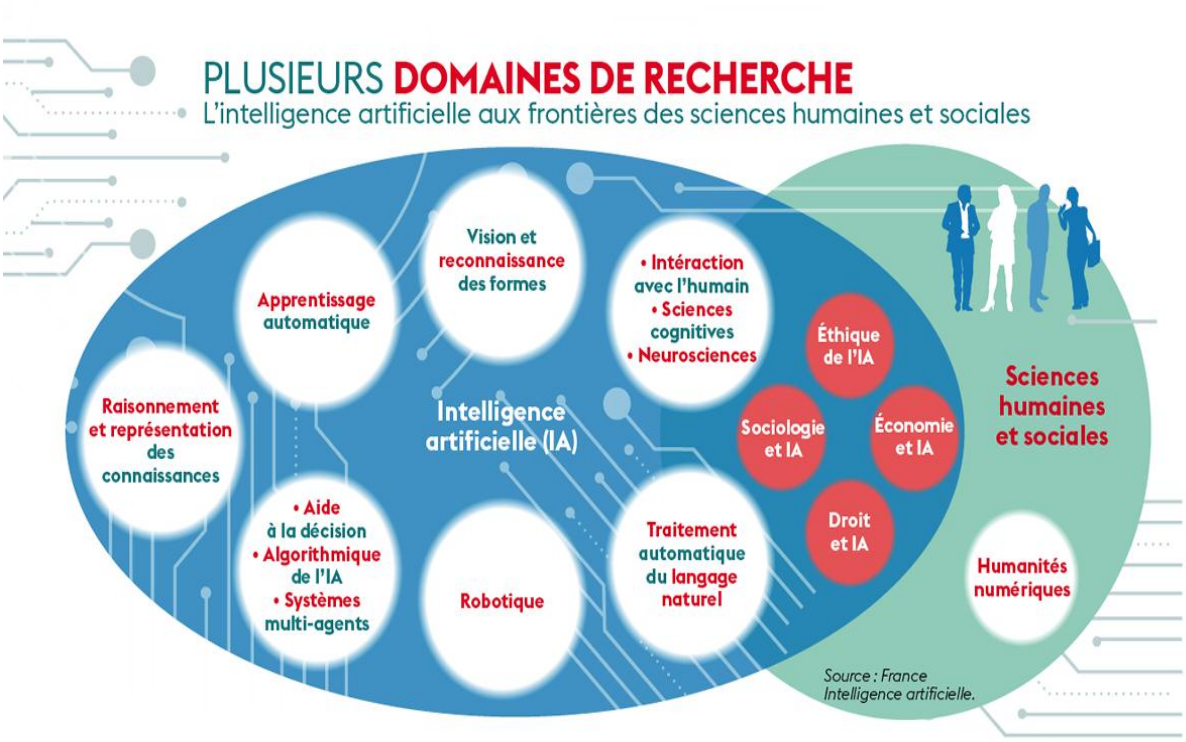


Figure 2.2-Domaines d'applications de l'apprentissage automatique [12]

2.3.4 Les différents types d'apprentissage automatique

Les algorithmes d'apprentissage peuvent se catégoriser selon le mode d'apprentissage qu'ils emploient :

- **L'apprentissage supervisé**

Si les classes sont prédéterminées et les *exemples* connus, le système apprend à classer selon un modèle de classement ; on parle alors d'apprentissage supervisé (ou d'analyse discriminante).

Un expert (ou oracle) doit préalablement correctement étiqueter des exemples. L'apprenant peut alors trouver ou approximer la fonction qui permet d'affecter la bonne «étiquette » à ces exemples. Parfois il est préférable d'associer une donnée non pas à une classe unique, mais une probabilité d'appartenance à chacune des classes prédéterminées (on parle alors d'apprentissage supervisé probabiliste). L'analyse discriminante linéaire ou les SVM sont des exemples typiques. Autre exemple : en fonction de *points communs* détectés avec les symptômes d'autres patients connus (les « exemples »), le système peut catégoriser de nouveaux patients au vu de leurs analyses médicales en risque estimé (probabilité) de développer telle ou telle maladie.

- **L'apprentissage non-supervisé**

Quand le système ou l'opérateur ne dispose que d'exemples, mais non d'étiquettes, et que le nombre de classes et leur nature n'ont pas été prédéterminés, on parle d'apprentissage non supervisé (ou clustering). Aucun expert n'est disponible ni requis. L'algorithme doit découvrir par lui-même la structure plus ou moins cachée des données

Le système doit ici dans l'espace de description (la somme des données) cibler les données selon leurs attributs disponibles, pour les classer en groupe homogènes

d'exemples. La similarité est généralement calculée selon la fonction de distance entre paires d'exemples. C'est ensuite à l'opérateur d'associer ou déduire du sens pour chaque groupe. Divers outils mathématiques et logiciels peuvent l'aider. On parle aussi d'analyse des données en régression. Si l'approche est probabiliste (c'est à dire que chaque exemple au lieu d'être classé dans une seule classe est associé aux probabilités d'appartenir à chacune des classes), on parle alors de *« soft clustering »* (par opposition au *« hard clustering »*)

Exemple : Un épidémiologiste pourrait par exemple dans un ensemble assez large de victimes de cancers du foie tenter de faire émerger des hypothèses explicatives, l'ordinateur pourrait différencier différents groupes, qu'on pourrait ensuite associer par exemple à leur provenance géographique, génétique, à l'alcoolisme ou à l'exposition à un métal lourd ou à une toxine telle que l'aflatoxine.

- **L'apprentissage semi-supervisé**

Effectué de manière probabiliste ou non, il vise à faire apparaitre la distribution sous-jacente des *« exemples »* dans leur espace de description. Il est mis en œuvre quand des données (ou *« étiquettes »*) manquent... Le modèle doit utiliser des exemples *non-étiquetés* pouvant néanmoins renseigner .Exemple : En médecine, il peut constituer une aide au diagnostic ou au choix des moyens les moins onéreux de tests de diagnostics.

- **L'apprentissage par renforcement**

L'algorithme apprend un comportement étant donné une observation. L'action de l'algorithme sur l'environnement produit une valeur de retour qui guide l'algorithme.

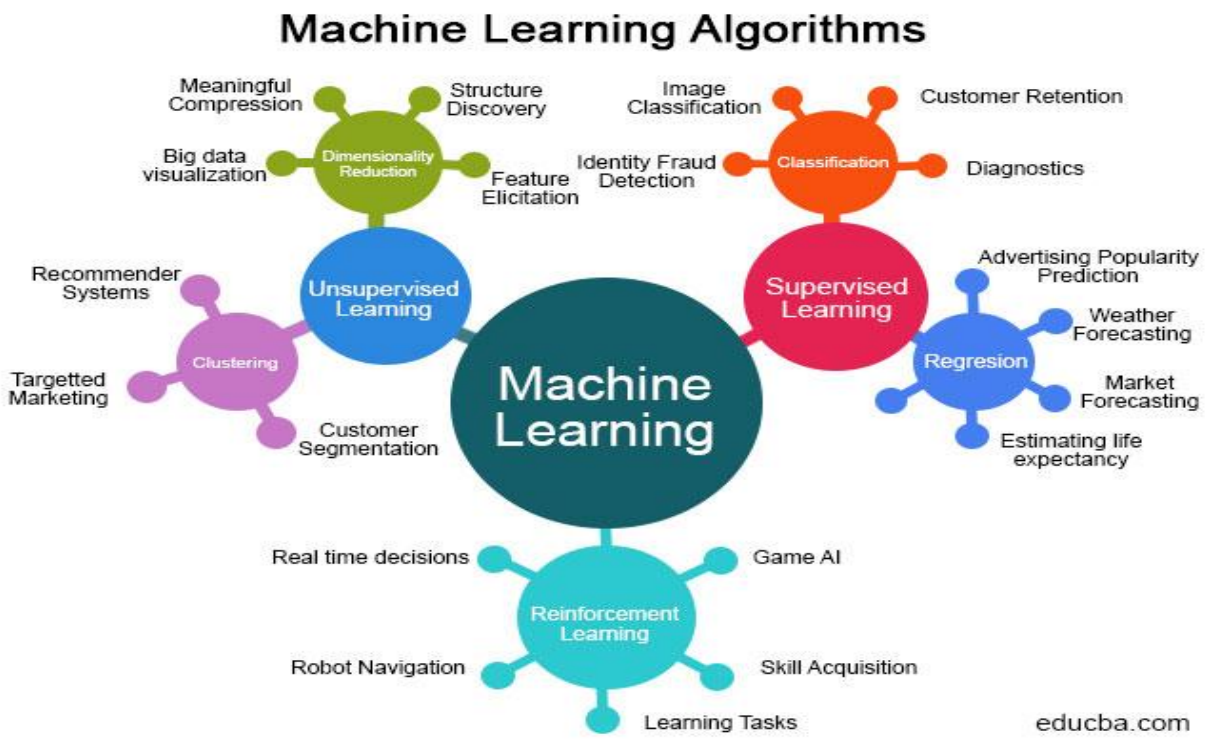


Figure2.3-les algorithmes d'apprentissage automatique [13]

2.4 L'apprentissage supervisé

2.4.1 Comment fonctionne l'apprentissage supervisé

Avec l'apprentissage supervisé, la machine peut apprendre à faire une certaine tâche en étudiant des **exemples** de cette tâche. Par exemple, elle peut apprendre à reconnaître une photo de chien après qu'on lui ait montré des millions de photos de chiens. Ou bien, elle peut apprendre à traduire le français en chinois après avoir vu des millions d'exemples de traduction français-chinois .D'une manière générale, la machine peut apprendre une **relation** qui en ayant **analysé** des millions d'exemples d'associations .La machine apprend à partir de milliers d'exemples x, y

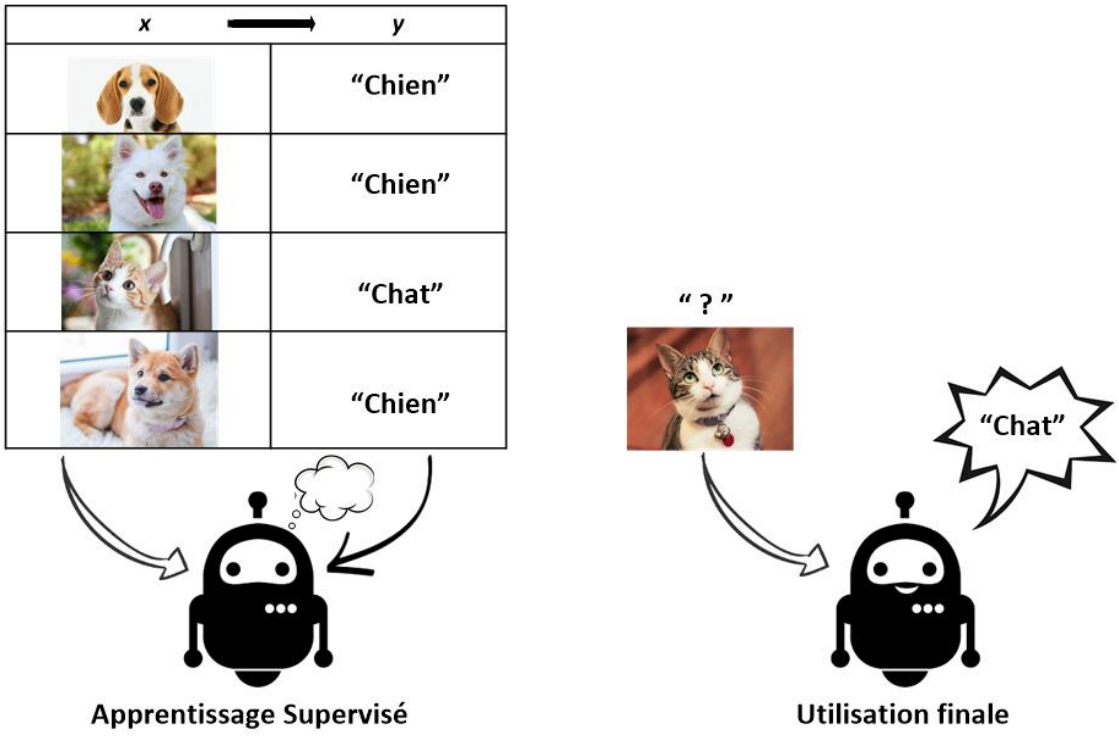


Figure 2.4-la fonctionnement de l'apprentissage supervisé [14]

2.4.2 Les étapes de l'apprentissage supervisé :

L'apprentissage supervisé fonctionne en 6 étapes

1. **Récolte des données** qui contient nos exemples
2. **préparation des données**
3. **Choisissez le ou les Modèles pertinents.**
4. **trainer le Modèle**
5. **tester le Modèle**
6. **Améliorer le Modèle**

Pas de panique ! Je vais vous expliquer tout ça dans les détails, étape par étape,

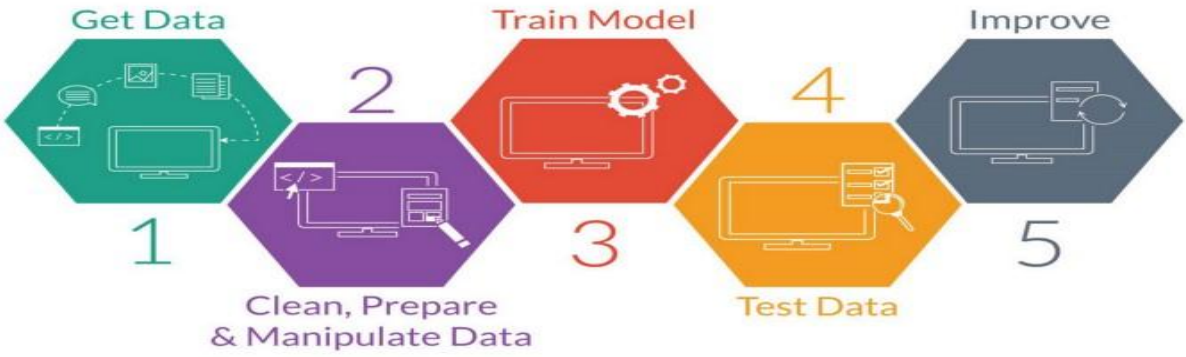


Figure 2.5-Les étapes de l'apprentissage supervisé [15]

- **Récolte des données** :

tout d'abord, rassemblez les données dont vous aurez besoin pour l'apprentissage automatique. Veillez à les rassembler sous une forme consolidée, afin qu'elles soient toutes contenues dans un seul tableau (*Flat Table*).

- **Préparation des données (*Data Wrangling*)** :

- il s'agit de préparer les données afin de les rendre exploitables par les algorithmes d'apprentissage automatique.
  - Nettoyage des données : trouvez les « Null », les valeurs manquantes et les données dupliquées. Il faut remplacer les « Null » et les valeurs manquantes par d'autres valeurs (ou les supprimer) et s'assurer de ne pas avoir de doublons.
  - Décomposition des données : les colonnes de texte contiennent parfois plus d'une information ; nous devons donc les diviser en autant de colonnes dédiées que

nécessaire. Si certaines colonnes représentent des catégories, convertissez-les en colonnes de type catégorie.

- Agrégation de données : regroupez certaines informations ensemble quand c'est pertinent
- Mise à l'échelle (*Data Scaling*) : cela permettra d'obtenir des données à une échelle commune, si ce n'est déjà le cas. La mise à l'échelle des données ne s'applique pas au label ou aux colonnes de catégories. Elle est nécessaire lorsqu'il y a une grande variation dans les plages de *features*.
- Mise en forme et transformation (*Data Shaping& Transformation*) : de catégoriel à numérique.

- Enrichissement des données : parfois, vous devrez enrichir les données existantes par des données externes afin de donner à l'algorithme plus d'informations avec lesquelles travailler, ce qui améliore le modèle (par exemple, des données économiques ou météorologiques).

- Visualisez vos données dans leur ensemble pour voir s'il existe des liens entre les colonnes. En utilisant des graphiques (*Charts*), vous pouvez voir les caractéristiques/*features* côte à côte et détecter tout lien entre les *features*, et entre les *features* et les *labels*.
  - Les liens entre les *features* nous permettent de voir si une *feature* donnée est directement dépendante d'une autre. Si c'est le cas, il se peut que vous n'ayez pas besoin des deux *features*.
  - Les liens entre une *feature* et le *label* nous permettent de voir si une *feature* aura un fort effet sur le résultat.
- Parfois, vous devrez générer des *features* supplémentaires à partir de celles qui existent déjà dans une classification (par exemple, lorsque l'algorithme choisi est incapable de différencier correctement les classes).

- Vous pouvez vous retrouver avec un nombre énorme de colonnes. Dans ce cas, vous devez choisir les colonnes que vous utiliserez comme *features*, mais si vous avez des milliers de colonnes (c'est-à-dire des *features* potentielles), vous devrez appliquer une réduction dimensionnelle. Il existe plusieurs techniques pour ce faire, notamment l'analyse en composantes principales ou ACP (*Principal Component Analysis*, PCA en anglais). L'ACP est un algorithme d'apprentissage non-supervisé qui utilise les colonnes existantes pour générer de nouvelles colonnes, appelées composantes principales, qui peuvent être utilisées ultérieurement par l'algorithme de classification.

- Divisez votre ensemble de données en trois parties : entraînement, test et validation.
  - Les **données d'entraînement** serviront à entraîner le ou les algorithmes choisis ;
  - Les **données de test** seront utilisées pour vérifier la performance du résultat ;
  - Les **données de validation** ne seront utilisées qu'à la toute fin du processus et ne seront, sauf nécessité, que très rarement examinées et utilisées avant afin d'éviter d'introduire un quelconque biais dans le résultat.

- **Choisissez le ou les Modèles pertinents.**

On peut lister un grand nombre d'algorithmes de l'apprentissage automatique et parmi eux on a

- Les machines à vecteurs support
- Le boosting
- Les réseaux de neurones pour un apprentissage supervisé ou non- supervisé
- La méthode des k plus proches voisins pour un apprentissage supervisé
- Les arbres de décision
- Les méthodes statistiques comme par exemple le modèle de mixture gaussienne
- La régression logistique

- L'analyse discriminante linéaire
- La logique floue
- Les algorithmes génétiques et la programmation génétique

Avant de commencer tout travail utilisant le Machine Learning, il est nécessaire de trouver l'algorithme permettant d'accomplir cette tâche.

Plusieurs outils sont donc à notre disposition notamment cet organigramme créé par Andreas Mueller qui permet assez facilement et rapidement de trouver l'algorithme associé à notre problème.

Ceci permet donc de créer le programme le plus optimisé possible par rapport aux tâches à accomplir. C'est un point non négligeable et surtout incontournable dans la création d'un programme utilisant le Machine Learning.

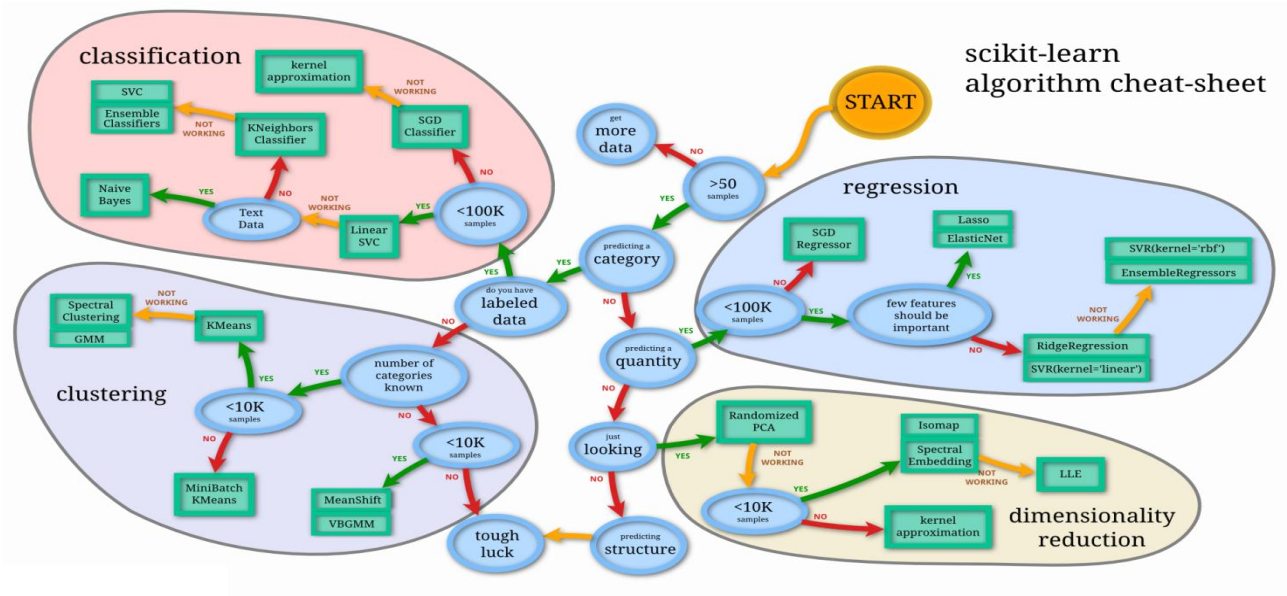


Figure 2.6- le plan de choix des algorithme des maching learning[16]

Selon les données utilisées qui on va les détaillés plus tard on a choisis les algorithmes suivants

- K nearest neighbors

- Support Vector Machine
- naïveBayes plus +

- (Decision Trees, Random Forest) qui sont les algorithmes à celui qui n'avait pas un bon algorithme

- **Trainer le Modèle**

C'est à cette étape que la magie opère ! L'ensemble de données se connecte à un algorithme et l'algorithme exploite une modélisation mathématique sophistiquée pour apprendre et développer des prédictions.

Ces algorithmes appartiennent généralement à l'une des trois catégories suivantes :

- Binaire - Classer en deux catégories
- Classification - Classer en plusieurs catégories
- Régression - Prédire un numérique

A la différence du modèle illustré plus haut, un modèle de Machine Learning ne repose pas sur une démonstration mathématique ou une équation physique. A la place, il est construit à partir de données, comme un modèle statistique.

Si par exemple votre Data set vous donne le nuage de point suivant, alors la machine devra trouver le modèle qui rentre le mieux dans ce nuage de point.

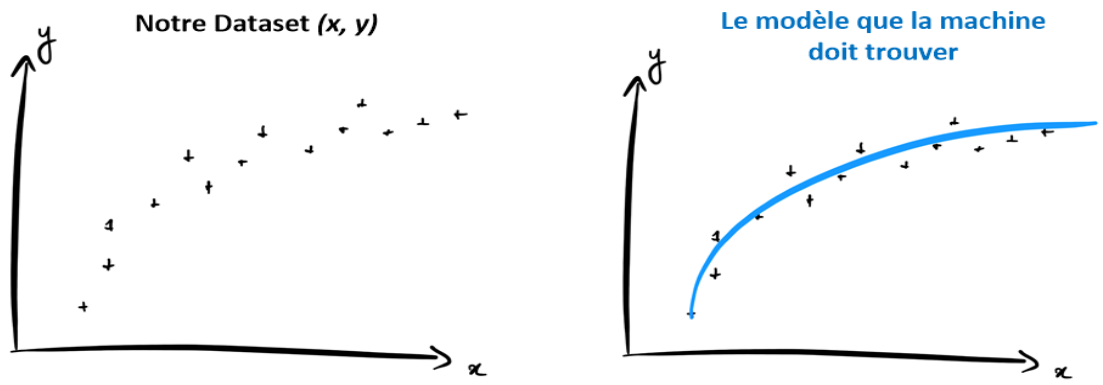


Figure 2.7- Aperçu de l'entrainement de modèle [17]

Cependant, ce n'est pas à la machine de faire tout le travail !

Dans les faits, c'est à nous de choisir le type de modèle (c'est-à-dire la fonction mathématique) et c'est à la machine de trouver les **coefficients** de cette fonction qui donnent les meilleurs résultats. Par convention on appelle ces coefficients les **paramètres** du modèle.

Par exemple, on peut choisir de développer un modèle **linéaire** et on laisse la machine trouver la valeur de et qui donne les meilleurs résultats. Ou bien on peut choisir un modèle **non-linéaire**, par exemple, ou sont les paramètres. Les possibilités sont infinies, mais nous verrons plus tard dans cette formation comment choisir un modèle plutôt qu'un autre.

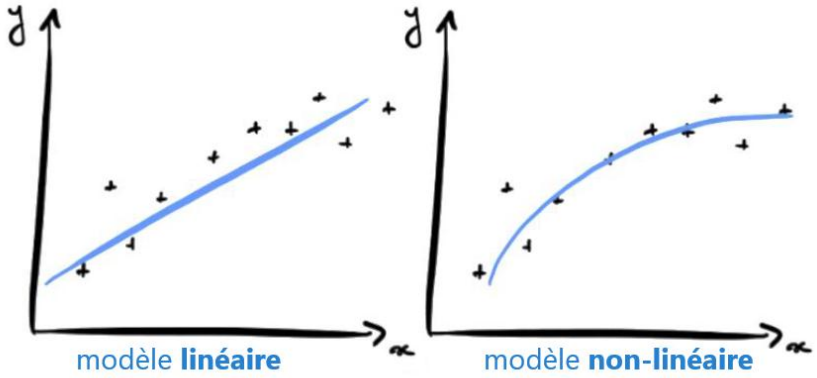


Figure 2.8-Apercu de deux types de models

Maintenant, la question est : comment faire pour que la machine trouve le meilleur modèle ? Autrement dit, comment faire pour que la machine apprenne

- **Tester le Modèle**

Il est maintenant temps de valider votre modèle entraîné. À l'aide des données de test de l'étape 3, nous vérifions la précision du modèle.

Pour que la machine trouve le meilleur modèle, il faut déjà qu'elle puisse **mesurer la performance** d'un modèle donné. Vous ne me croyez pas ? Imaginez que vous participiez à un concours de tir à l'arc. Comment savoir si vous êtes meilleur que votre voisin sans mesurer la **distance** entre vos flèches et le centre de la cible ? C'est impossible. Vous devez mesurer vos performances pour juger lequel de vous deux est le meilleur.

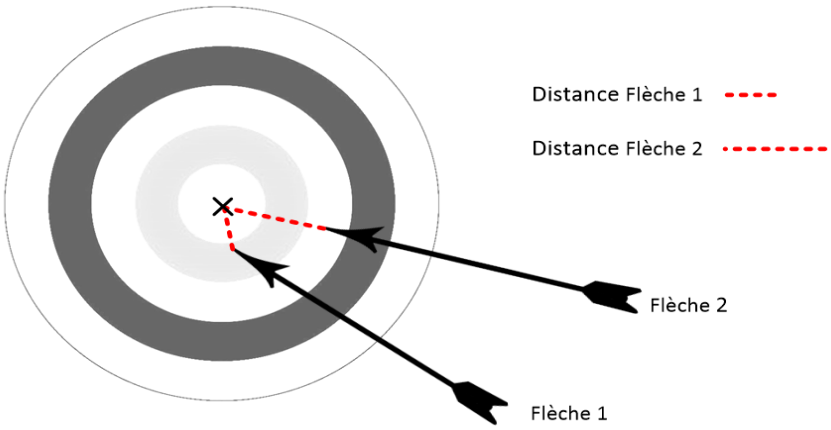


Figure 2.9- Aperçu la mesure de distance entre le centre etles flèches

C'est la même chose en Machine Learning. Pour savoir quel modèle est le meilleur parmi 2 candidats, il faut les **évaluer**. Pour cela, on mesure l'erreur entre un modèle et le Data, et on appelle ça a **Fonction Coût**.

Dans le cas d'une **régression**, on peut par exemple mesurer l'erreur entre la prédiction du modèle et la valeur qui est associée à ce dans notre Data. C'est similaire à l'idée de mesurer la distance entre votre flèche et le centre de la cible, qui n'est autre que le point qu'elle est sensée atteindre.

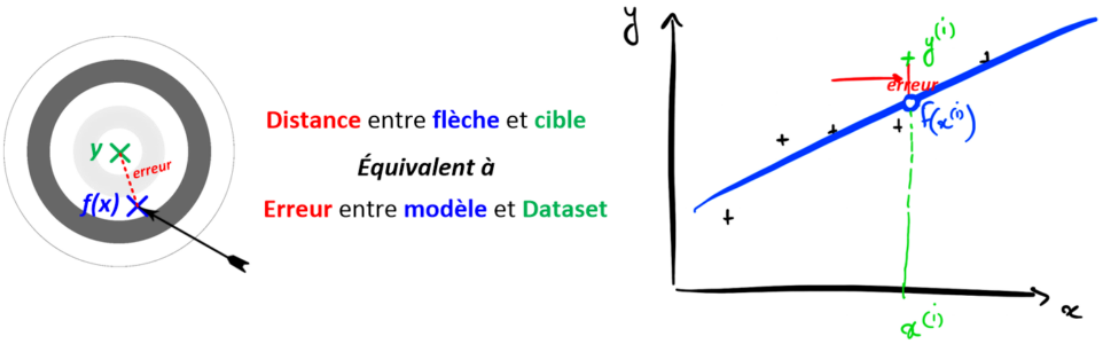
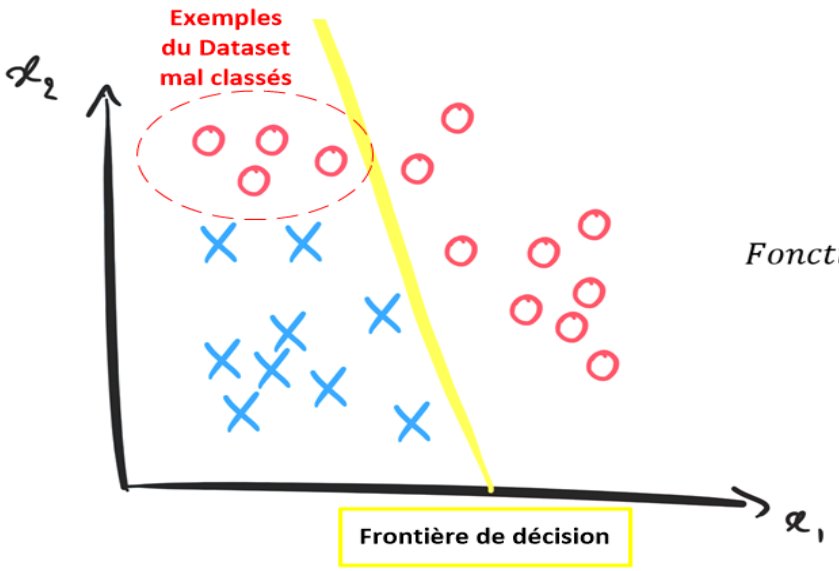


Figure 2.10- Aperçu de l'équivalence entre les deux mesures

Maintenant, dans le cas d'une **classification**, on peut construire notre Fonction Coût en mesurant le nombre d'exemples du Data set que notre modèle aura mal classé avec sa **frontière de décision**



$$Fonction\ Coût = \frac{\#\ Erreurs}{\#\ Exemples} = \frac{4}{22}$$

Figure 2.11- Aperçu de modèle de classification

Si les résultats ne sont pas satisfaisants, vous devez améliorer et recycler votre modèle ML (étape 4).

- **Améliorer le Modèle**

C'est en forgeant qu'on devient forgeron! Voici quelques mesures à prendre pour affiner votre modèle et améliorer la précision :

Passez en revue les résultats de votre modèle avec les parties prenantes de votre entreprise. Y a-t-il d'autres éléments de données qui méritent d'être ajoutés à votre modèle pour le rendre plus précise

Reconsidérez votre choix d'algorithme. Dans chaque classe d'algorithme, il existe des dizaines de choix d'algorithmes. Un algorithme différent peut mieux fonctionner pour vous

Ajustez les paramètres de votre algorithme choisi pour améliorer les performances. Parfois, de petits ajustements ont un impact significatif.

Parlons peu, parlons bien. Avoir un bon modèle, c'est avoir un modèle qui de petites erreurs. Logique ?

Ainsi, en Supervised Learning, la machine cherche les **paramètres** de modèle qui **minimisent** la **Fonction Coût**. C'est ça qu'on appelle l'apprentissage. Cette phrase est **très importante**. C'est l'essentiel de ce qu'il faut comprendre en Machine Learning. Pour trouver les paramètres qui minimisent la fonction Coût, il existe un paquet de stratégies.
On pourrait par exemple développer un algorithme qui tente au hasard plusieurs combinaisons de paramètres, et qui retient la combinaison avec la Fonction Cout la plus faible. C'est un peu comme organiser un concours d'archers pour ne garder que le meilleur. Cette stratégie est cependant assez inefficace la plupart du temps.

Une autre stratégie, **très populaire** en Machine Learning, est de considérer la Fonction Coût comme une fonction **convexe**, c'est-à-dire une fonction qui n'a qu'un seul minimum, et de chercher ce minimum avec un algorithme de minimisation appelé **Gradient Descent**. Nous le verrons en détails dans les prochains articles. Cette stratégie apprend de façon graduelle, et assure de converger vers le minimum de la fonction Coût (si convexe).

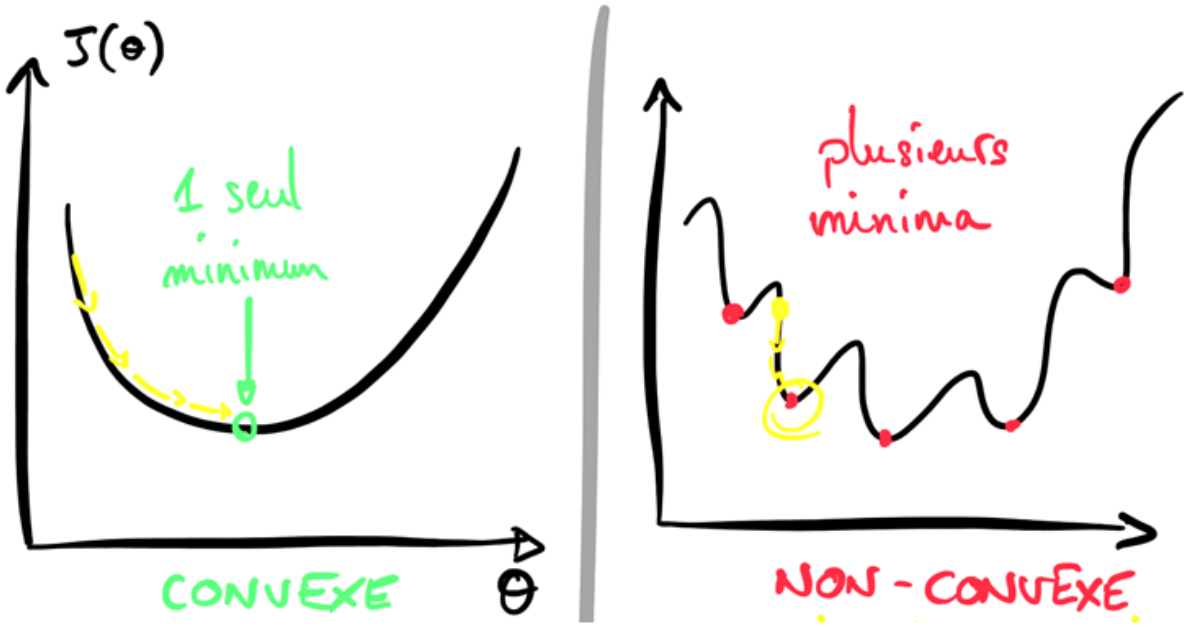


Figure 2.12-Aperçu de la minimisation de Coût

Dans le jargon, on appelle cette étape la **phase d'entrainement** du modèle. La machine choisit des paramètres pour le modèle, puis évalue sa performance (Fonction Coût) puis cherche des paramètres qui peuvent améliorer sa performance actuelle, etc.

Une fois la phase d'entrainement terminée… Vous avez un modèle de **MACHINE LEARNING ! BRAVO !** [18]

2.5 les algorithmes de l'apprentissage automatique utilisés

2.5.1Knearestneighbors (KNN)

K nearest neighbors (KNN) ou K plus proche voisins en français est l'un des méthodes d'apprentissage supervisé le plus simple, utilisé pour résoudre des problèmes de classification et de la régression. son fonctionnement est de classer les nouveaux points de données en fonction de la similarité aux points de données voisins.

➔ KNN est un algorithme qui ne fait aucunes hypothèses sur la structure des données et de la distribution, ce qui signifie qu'il s'agit d'un algorithme non paramétrique.

➔ Il est également appelé algorithme de l'apprenant paresseux, car il n'apprend pas immédiatement de l'ensemble d'apprentissage, mais stocke l'ensemble de données et, au moment de la classification, il exécute une action sur l'ensemble de données.

➔ KNN fonctionne par classification ou prédiction sur la base d'un nombre fixe (K) de points de données les plus proches de point d'entrée. Cela signifie que pour une valeur choisie de K, un point d'entrée serait classée ou devrait appartenir à la même classe que la classe la plus proche des nombre des points K voisins .[20] .

Exemple

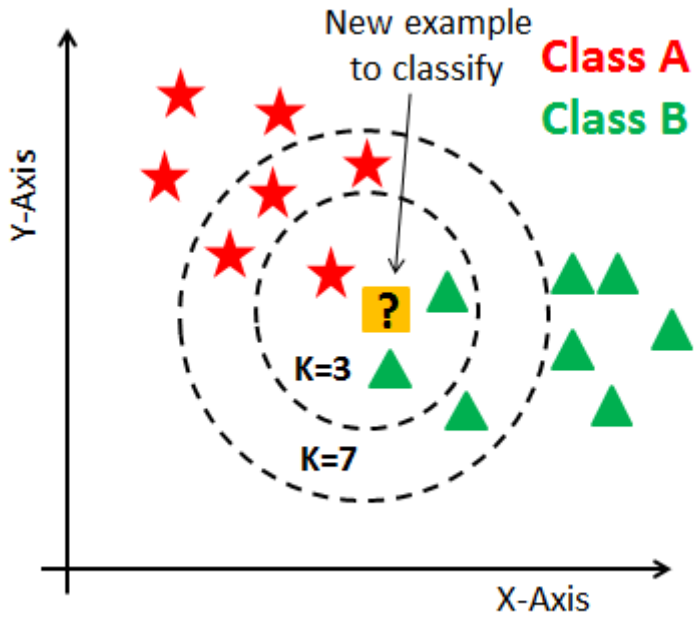


Figure 2.13- Exemple simple sur KNN

L'interprétation de l'exemple

Dans cet exemple nous avons une donnée non classée et tous les autres données sont classée (étoile et triangle) chacun avec leur classe (classe A et B).

- ➔ Si k=3 les données les plus proche du nouvelle donnée sont qui ont à l'intérieure de premier cercle, et la classe la plus prédominante c'est triangle (Classe B) car 2 triangles et seulement 1 étoile donc la donnée non classée sera classer un triangle (Classe B).
- ➔ Si k=7 les données les plus proches du nouvelle donnée sont qui ont à l'intérieure de deuxième cercle, et la classe la plus prédominante c'est l'étoile (Classe A) car on a 4 étoiles et 3 triangles donc le donnée non classée sera classer un étoile (Classe A).

La distance entre le point non classée et les plus proches voisins

La distance entre le point non classée et les plus proches voisins est mesuré en utilisant différents méthode comme : la distance uclidienne, la distance de Manhattan, la distance de Minkowski, celle de Jaccard, la distance de Hamming. . . etc, le fonction de distance est choisi en fonction de type de données qu'il manipule. Pour les données de même type la distance euclidienne est le bon candidat, et pour les données qui ne sont pas de même type la distance de Manhattan est le bonne mesure pour l'utiliser

Les représentations mathématiques de quelques distances

. Distance euclidienne

$$D_e(x, y) = \sqrt{\Sigma_{j=1}^{n}(x_j - y_j)^2}$$

. Distance Manhattan

$$D_m(x, y) = \Sigma_{i=1}^{k}|x_i - y_i|$$

Remarque

Le choix de la bonne valeur de K est un processus appelé réglage des paramètres est important pour une meilleure précision, la sélection des valeurs plus petites pour k aura une plus grand influence sur le résultat. Et pour la sélection des valeurs plus élevé de k auront des limites de décision plus lisses, ce qui signifie une variance plus faible mais un biais.

- **Algorithme de construction de KNN**

1. Sélectionnez le nombre K des voisins
2. Pour chaque exemple de l'ensemble de données :

2.1. Calculez la distance entre l'exemple de requête et l'exemple actuel à partir des données.

2.2. Ajouter la distance et l'index de l'exemple à une collection ordonnée

3. Trier cette collection de distances et d'indices du plus petit au plus grand (par ordre croissant) ordonnée par les distances.

4. Choisi les k premiers entrée de collections

5. Attribuer l'exemple de requête à la classe ou laquelle le nombre de k voisins est maximal (classe le plus fréquent).

- **Avantage de KNN**

1. Simple à implémenter
2. Gérer naturellement les cas multi classes
3. Peut être utilisé pour la classification et la régression

- **Inconvénients de KNN**
  1. le choix de la valeur de k (le nombre de voisins le plus proche)

     2. Le cout de calcul est élevé (pour chaque instance de l'ensemble de données on a besoin de calculer la distance)
  2. Stockage de données
  3. Sensible aux fonctionnalités non pertinentes

2.5.2 Decision Trees (Arbre de décision)

Decision Trees ou L'arbre de décision c'est un algorithme parmi les algorithmes d'apprentissage supervisé le plus utilisé et le plus pratique, qui est adapté pour résoudre tout type de problèmes (classifications ou régressions) telle-que :

➔ Un arbre de décision est une structure arborescente semblable à un organigramme ou un nœud interne représente une caractéristique (ou un attribut), la branche représente une règle de décision et chaque nœud feuille représente le résultat, cette structure aide pour prendre la décision.

➔ C'est un algorithme non-paramétrique signifie qu'il n'y a pas d'hypothèse sous-jacente sur la distribution des données.

Mesure de sélection d'attribut

Le principal problème qui se pose lorsque la construction d'un arbre de décision si comment choisi ou sélectionné le meilleur attribut pour le nœud racine et qui sépare mieux l'ensemble de données ? Pour résoudre ce problème il existe un technique qui appelé Mesure de sélection d'attribut ou ASM qui contient deux mesures principales et populaires sont :

1. Indice de Gini
2. Gain d'information

- **Exemple**

  C'est un petit exemple pour prédire si une personne est diabétique ou non, cette modèle contient trois attributs qui sont *minimun systolic blood pressure* , *age*et *glucose* avec deux classes diabétique et non-diabétique Dans cette exemple les attributs représente les nœuds interne, lequel basé pour l'arbre divise en branche, la fin de branche qui ne sépare plus est le feuille (la décision) où on peut prédire si un personne est diabétique ou non.

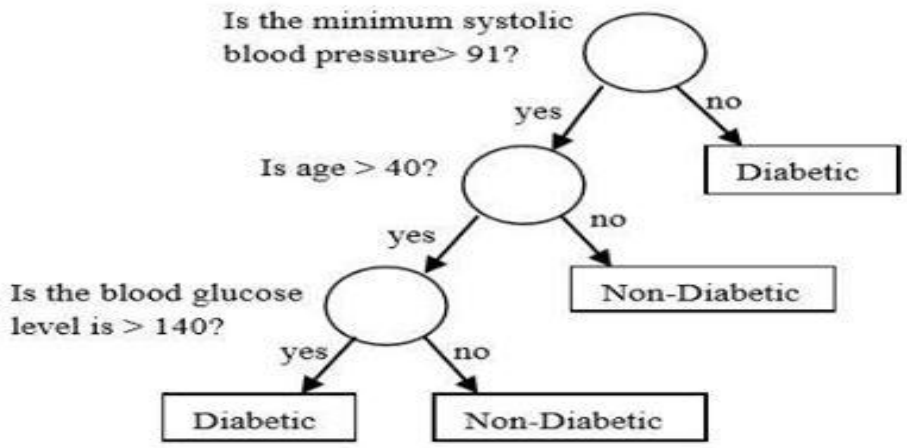


Figure 2.14- Arbre de décision

Cet arbre répondre à la question si un personne diabétique ou non ?

L'interprétation de l'exemple

Les arbres de décision sont bien adaptés aux problèmes de catégorisation où les attributs sont vérifiés pour déterminer une catégorie finale a cause de sa construction naturelle

*(*si. . . alors. . . sinon. . .) . Par exemple la Lecteur de l'exemple :

Si minimum, systolic blood pressure >91 = no , **alors** : personne=diabetic ;

Si minimum systolic blood pressure >91 = yes **AND** age >40 = no ,**alors**: personne = non-diabetic

Si minimum systolic blood pressure >91 = yes **AND** age >40 = yes **AND** glucose =no ,**alors**: personne = non-diabetic ;

Si minimum systolic blood pressure >91 = yes **AND** age >40 = yes **AND** glucose =yes ,**alors**: personne = diabetic ;

- **Algorithme de construction d'un arbre de décision**

1. **Sélectionne le meilleur attribut (nœud racine) :**

- pour chaque attribut le gain d'information est calculé, et celui qui il est maximal est sélectionner et des branches sont créer pour chaque valeurs de cette attribut .

2. **Continuez la division :** pour chaque branche s'étendant à partir de nœud, en répétant récursivement le processus

- 3. **Arrête la division si .**

3.1 nous obtenons un nœud pur, c'est-`a-dire un nœud qui ne contient que des points de données positifs ou négatifs .

3.2 nous obtenons très peu de points dans un nœud .

3.3 on atteint une certaine profondeur de l'arbre .

Avantage des arbres de décision .

1. faciles à expliquer et comprendre .

2. Fonctionne avec des données catégorielles et numériques .

.

3. peu couteux en termes de calcul

- **Inconvénient des arbres de décision .**

1. Il faut souvent plus de temps pour former le modèle.
2. L'arbre devient plus complexe à mesure qu'il s'approfondit.
3. Un petit changement dans les données peut entraîner un changement global de la structure de l'arbre de décision .

2.5.3 Random Forest (forêts aléatoires) .

Random Forest ou forêts aléatoires est un algorithme d'apprentissage supervisé très populaire Il est également utilisé pour les problèmes de régression ou de classification. Basé sur un ensemble des algorithmes d'apprentissage, qui est

un processus de combinaison de plusieurs algorithmes pour résoudre un problème complexe et améliorer les performances du modèle. C'est un algorithme qui créer de nombreux arbres de décision (c'est la raison pour laquelle il est appelé une forêt) sur divers sous-ensembles de l'ensemble de données. Elle prend la prédiction de chaque arbre et sur la base des votes majoritaires des prédictions, et elle prédit le résultat final et La figure suivant explique le fonctionnement et la structure d'algorithme

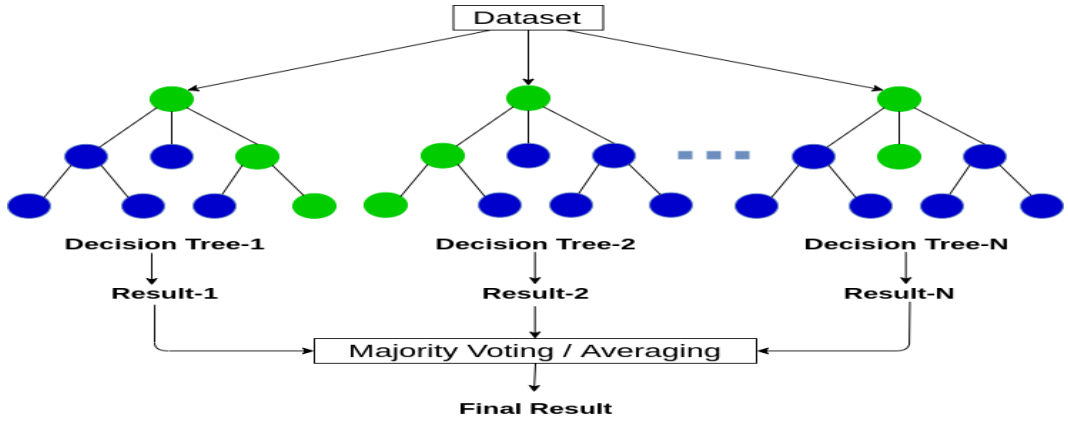


Figure 2.15- Structure de l'algorithme random forest

Exemple

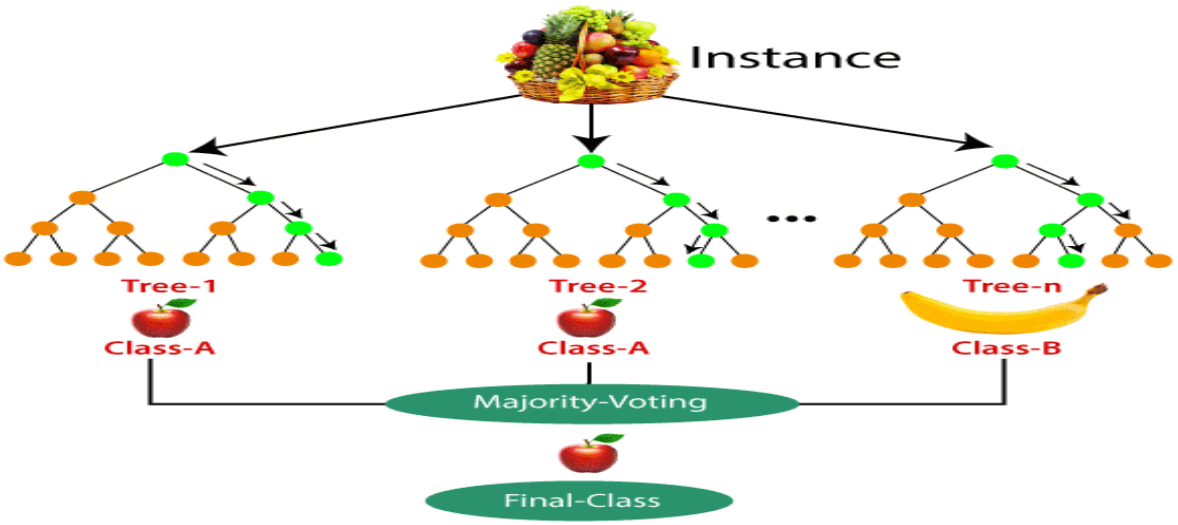


Figure 2.16- Un simple exemple sur l'algorithme random forest [19]

L'interprétation d'exemple

Dans cet exemple l'ensemble de données contenant un ensemble d'images de fruits classifié par l'algorithme random forest, cette ensemble est divisé en sous-ensembles et donné à chaque arbre de décision et dans la phase d'apprentissage chaque arbre produit un résultat de prédiction, et lorsqu'un nouveau point de données se produit, puis sur la base de la majorité des résultats, Random Forest prédit la décision finale (comme l'exemple dans l'image).

L'Algorithme de construction de Random forest

1. Sélectionnez des échantillons aléatoires à partir d'un ensemble de données d'entrainement.

2. Créer des arbres de décision pour chaque échantillon (sous-ensembles). Ensuite on obtient le résultat de prédiction de chaque arbre de décision
3. Pour les nouveaux points le vote sera effectué pour chaque résultat prédit.

4. sélectionnez le résultat de prédiction le plus voté comme résultat de prédiction final. [30]

Avantage de Random forest

1. Il s'agit de l'un des algorithmes d'apprentissage les plus précis disponibles. Pour de nombreux ensembles de données, il produit un classificateur très précis.
2. Il fonctionne efficacement sur de grandes bases de données. 3. Il dispose d'une méthode efficace pour estimer les données manquantes et maintient la précision lorsqu'une grande partie des données sont manquantes.

Inconvénient de Random forest

Le principal inconvénient de l'algorithme random forest est qu'un grand nombre d'arbres peut rendre l'algorithme trop lent et inefficace pour les prédictions en temps réel. En général, ces algorithmes sont rapides à entraîner, mais assez lents à créer des prédictions une fois qu'ils sont formés. Une prévision plus précise nécessite plusd'arbres, ce qui entraîne un modèle plus lent

2.5.4 Support Vector Machine (SVM)

Support Vector Machine ou SVM est l'un des algorithmes d'apprentissage supervisé les plus populaires, utilisé pour les problèmes de classification et de régression. Cependant, il est principalement utilisé pour les problèmes de classification dans l'apprentissage automatique. Le but de l'algorithme SVM est de créer la meilleure ligne ou limite de décision qui peut séparer l'espace à n dimensions en classes afin que nous puissions facilement mettre le nouveau point de données dans la bonne classe à l'avenir. Cette meilleure frontière de décision est appelée un hyperplan.

SVM choisit les points / vecteurs extrêmes qui aident à créer l'hyperplan. Ces cas extrêmes sont appelés vecteurs de support, et donc l'algorithme est appelé machine de vecteur de support.

Les diagrammes suivant illustre deux classes (classe des points bleus et classe des points roses) différant qui sont classés avec un hyperplan.

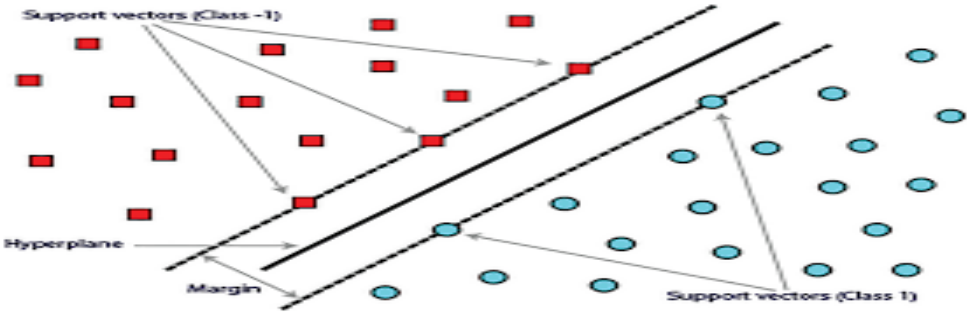


Figure 2.17-Séparation parfait de deux classes avec un hyperplan

Exemple

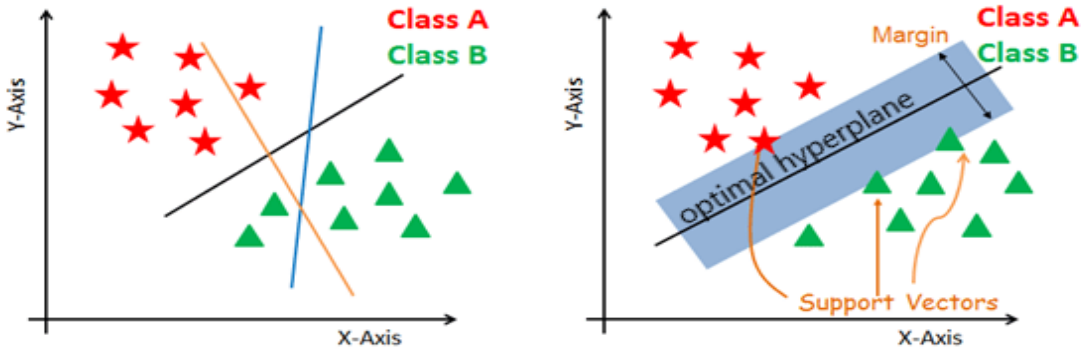


Figure 2.18- Un simple exemple sur le fonctionnement de l'algorithme SVM [20]

L'interprétation d'exemple

Dans cette exemple le jeu de données contient des étoiles et des triangles qui sont respectivement classé dans les classe A et B, dans la phase d'apprentissage le classificateur SVM consiste à trouve le meilleur hyperplan qui sépare parfaitement les deux classe, et classe correctement les nouveaux données ainsi comme les vecteurs de support crée une frontière de décision entre les deux

classes les nouveaux données sera classé à la base de ces vecteurs

Hyperplan et vecteur de support et marge dans l'algorithme SVM

Hyperplan :

Les hyperplans sont des limites de décision qui aident à classer les points de données dans un espace à n dimensions, ces points de données tombant de chaque côté de l'hyperplan peuvent être attribués à différentes classes. La dimension de l'hyperplan dépendent au nombre des entités dans le jeu de données, si le nombre d'entité égale à 2 l'hyperplan sera une ligne. Et si le nombre d'entité égal à 3 l'hyperplan devient un plan bidimensionnel .

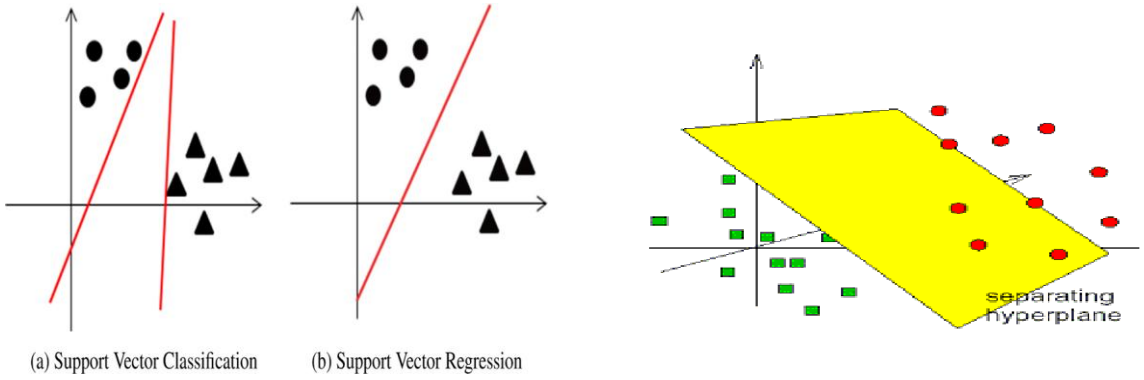


Figure 2.19- Hyperplan dans les entités 2D et 3D

➔ **Vecteur de support :**

Les vecteurs de support sont des points de données plus proches de l'hyperplan, et influencent la position et l'orientation d'hyperplan, la suppression de ces vecteur modifier la position de l'hyperplan.

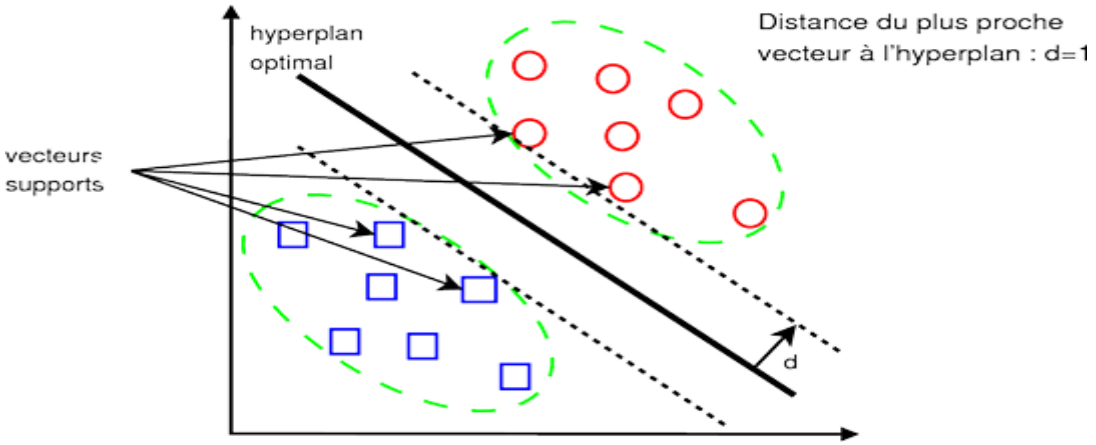


Figure 2.20- Les vecteurs de support

➔ **Marge :** c'est la distance entre les vecteurs de support et l'hyperplan. l'hyperplan optimal c'est qui a le plus grand marge, car une plus grande marge garantit que de légères déviations dans les points de données ne doivent pas affecter le résultat du modèle.

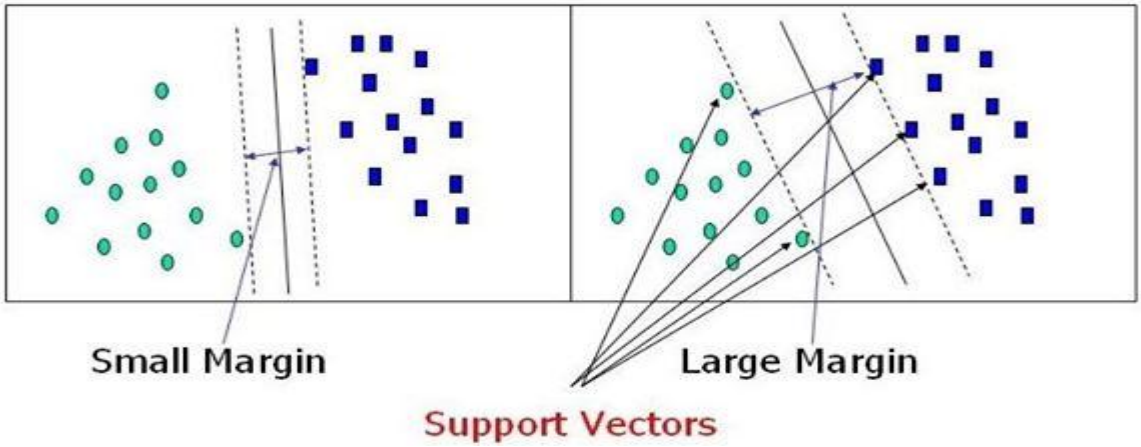


Figure 2.21- Marge dans l'algorithme SVM

Avantage de SVM

1. Il a la capacité de gérer de grands espaces fonctionnels.
2. Fonctionne bien avec même des données non structurées et semi-structurées comme du texte, des images et des arbres.
3. Il s'adapte relativement bien aux données de grande dimension.

Inconvénient de SVM

1. Il est sensible au bruit

2. Difficile de comprendre et d'interpréter le modèle final, les poids variables et l'impact individuel.
3. L'extension de la classification à plus de deux classes est problématique.

2.5.5Naïve_Bayes

naïve bayésienne fait partie des algorithmes d'apprentissage automatique supervisé qui sont principalement utilisés pour la classification. C'est un classificateur probabiliste simple basé sur l'application de *théorème de bayes* et qui aide à construire des modèles d'apprentissage automatique rapides qui peuvent faire des prédictions rapides. Naïve dans l'algorithme se réfère à l'hypothèse naïve que l'algorithme fait, qui est que chaque fonctionnalité est indépendante des autres fonctionnalités .

Théorème de bayes

Le théorème de Bayes (alternativement la loi de Bayes ou la règle de Bayes) décrit la probabilité d'un événement , basée sur la connaissance préalable des conditions qui pourraient être liées à l'événement. La formule est comme suit $P(A/B)=P(B/A)P(A)/P(B)$

Où : $P(A/B)$: la probabilité conditionnelle que l'événement A se produise, étant donné que B s'est produit. Ceci est également connu comme la probabilité postérieure .$P(A)$ et $P(B)$: probabilité de A et B sans égard l'un à l'autre

- **Avantage de naïve Bayes**
  1. fonctionne également bien dans la prédiction multi-classes.
  2. Lorsque l'hypothèse d'indépendance est vérifiée, un classificateur naïve Bayésienne fonctionne mieux que d'autres modèles.

3. Fonctionne mieux que les modèles plus compliqués lorsque l'ensemble de données est petit.

- **Inconvénient de naïve Bayes**

limitation de naïve Bayésienne est l'hypothèse de fonctionnalités indépendantes .Dans la vraie vie, il est presque impossible d'obtenir un ensemble de fonctionnalités complètement indépendants .

Remarque

Un bon modèle d'apprentissage choisit la fonction de prédiction qui réalise la plus faible erreur de prédiction

2.6 Conclusion

Dans ce chapitre, nous avons présenté les algorithmes d'apprentissage automatique qui peuvent nous aider a détecter l'apparition précoce du diabète, ce qui peut aider à réduire les risques des complications de cette maladie sur la santé du patient.

Dans l'étude qui suit, l'objectif principal est d'appliquer ces différents algorithmes (K nearest neighbors, Decision Trees, Random Forest, Support Vector Machine, Naives Bayes) de classification sur notre données

Chapitre 3 : La pratique de la prédiction par l'apprentissage automatique

3.1 Introduction

Dans ce dernier chapitre, nous présentons d'abord une étude technique dans laquelle nous définissons l'environnement logiciel utilisée pour construire notre application, puis nous définirons notre data set avec une description de ses caractéristiques et les étapes de pré-traitement des données (explorer, nettoyer, sélection de modèle ...) pour corriger les valeurs aberrantes et choisir le meilleur modèle a suivre.

A la fin, c'est la partie application ou nous fournissons des interfaces graphiques importantes développées pour clarifier les performances des activités du système et nous terminerons par une conclusion.

3.2Python

Python C'est un langage de programmation multi-paradigme et le langage de programmation dominant dans la data science avec de nombreuses implémentations ce qui le rend encore plus intéressant. Concernant le domaine de l'apprentissage automatique Python se distingue tout particulièrement en orant une pléthore de librairies de très grande qualité, couvrant tous les types d'apprentissages disponibles qui combine la facilite d'utilisation et d'apprentissage avec la puissance des librairies qu'elles possèdent. Parmi ces bibliothèques, nous avons utilisé [21]

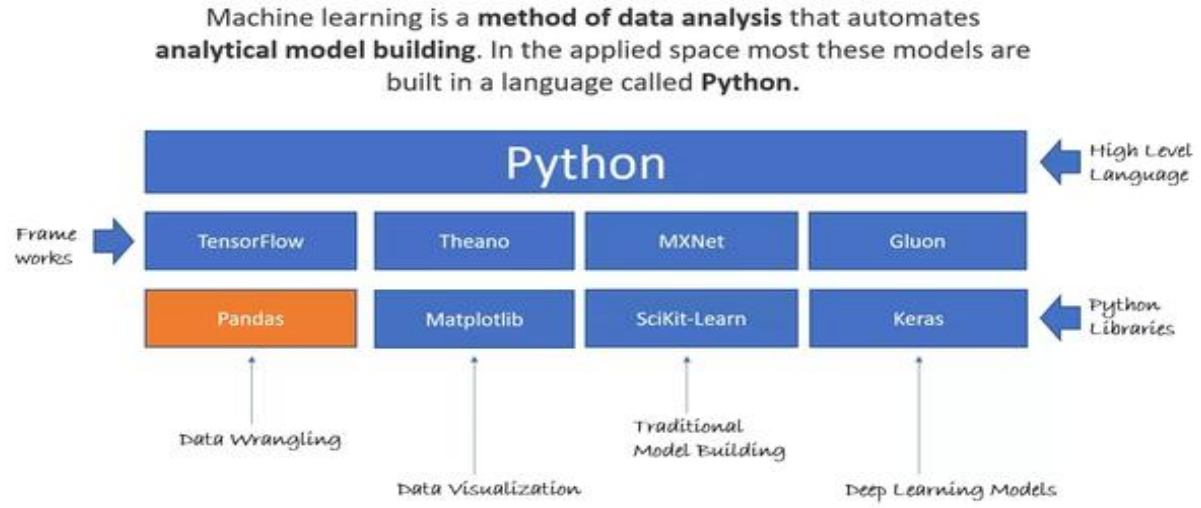


Figure 3.1 –Aperçu des Framework et libraires de python[22]

3.3Outils et Librairies utilisés

❖ Anaconda

Anaconda est une distribution python pour les applications de data science et d'apprentissage automatique. C'est un logiciel gratuit et open source qui contient plusieurs packages. Le principal avantage de l'utilisation d'anaconda est que, anaconda est comme un point central pour les bibliothèques qui auraient besoin pour le traitement de données,

L'analyse prédictive et les calculs scientifiques.

❖ jupyter notebook

Jupyter Notebook est un environnement de programmation qui prend en charge plusieurs langages de programmation, dont Python. Jupyter Notebook nous permet de créer des documents contenant du code, des équations, des visualisations et du texte. Ses utilisations comprennent le nettoyage et la transformation des données, la simulation numérique, la modélisation statistique, la visualisation des données, l'apprentissage automatique et bien plus encore.

➢ Matplotlib

Matplotlib est une bibliothèque complète pour créer des visualisations statiques, animées et interactives en Python.

➢ Seaborn

Seaborn est une bibliothèque de visualisation de données Python basée sur matplotlib . Il fournit une interface de haut niveau pour dessiner des graphiques statistiques attrayants et informatifs.

➢ Pandas

Pandas est une autre bibliothèque Python utilisée pour la manipulation et l'analyse des données, le point fort de cette bibliothèque est qu'elle possède une fonctionnalité importante appelée nettoyage des données qui résout le problème du temps passée à nettoyer les données dans un projet d'apprentissage automatique car de nombreux ensembles de données disponibles contiennent des champs vides ou nuls, ce qui peut avoir un impact négatif énorme sur notre modèle.

➢ NumPy

NumPy est une extension du langage de programmation Python, destinée à manipuler des tableaux multidimensionnels.

- **Scikit-learn**

elle est la bibliothèque Python la plus importante pour ce qui concerne l'apprentissage automatique telle que il contient de nombreux algorithmes (forêts aléatoires, des régressions logistiques, des algorithmes de classification, et les machines à vecteurs de support) .

- **Flask**

Flask est un petit Framework web Python léger, qui fournit des outils et des fonctionnalités utiles qui facilitent la création d'applications web en Python. Il offre aux développeurs une certaine flexibilité et constitue un cadre plus accessible pour les nouveaux développeurs puisque vous pouvez construire rapidement une application web en utilisant un seul fichier Python. Flask est également extensible et ne force pas une structure de répertoire particulière ou ne nécessite pas de code standard compliqué [23]

3.4 Définir l'ensemble des données et des variables utilisés

Cet ensemble de données provient à l'origine de l'Institut national du diabète et des maladies digestives et rénales. L'objectif de l'ensemble de données est de prédire de manière diagnostique si un patient est diabétique ou non, sur la base de certaines mesures diagnostiques incluses dans l'ensemble de données. Plusieurs contraintes ont été imposées à la sélection de ces instances à partir d'une base de données plus large. En particulier, tous les patients ici (768) sont des femmes d'au moins 21 ans d'origine indienne Pima. Contenu Les ensembles de données se composent de plusieurs variables prédictives médicales et d'une variable cible, Outcome. Les variables prédictives comprennent le nombre de grossesses que la patiente a eues, son IMC, son taux d'insuline, son âge, etc.

Les variables sont les suivants :

1. **Glucose** : Concentration plasmatique de glucose à 2 heures dans un test oral de tolérance au glucose.
2. **Prégnances** : Nombre de fois enceinte.
3. **BloodPressure** : Pression artérielle diastolique (mm Hg).
4. **SkinThickness** : Epaisseur de pli cutané du triceps (mm).
5. **Insulin** : Insuline sérique 2 heures (mu / ml).
6. **BMI** : (ou IMC) Indice de masse corporelle (poids en kg /taille en m)
7. **DiabetesPedigreeFunction** : Fonction généalogique du diabète.
8. **Age** : l'âge en années.
9. **Outcome** : variable de classe (0 ou 1) ou 0 indique que le patient ne soufre pas de diabète et 1 indique que le patient est diabétique. [24]

Variable	Description	La plage
Pregnancies	Nombre de fois enceinte	Minimum : 0 Maximum : 17
BloodPressure	Si un TA diastolique > 90 signifie une pression artérielle élevé (probabilité élevée de diabète) Un TA diastolique < 60 signifie une pression artérielle base (mois probabilité de diabète)	Minimum : 0 Maximum : 122
SkinThikness	Valeur estimé pour la graisse corporelle. épissure normal du pli cutané chez les femmes est de 23 mm. Une épissure plus élevée conduit à l'obésité et les chances de diabète augmente.	Minimum : 0 Maximum : 110
Insulin	Insuline sérique 2 heures (mu U/ ml) et niveau d'insuline normal 16-166 mUI/L, les valeurs au-dessus de cette plage peuvent être alarmante	Minimum : 0 Maximum : 799
BMI	(poids en kg / taille en m2) IMC de 18.5 à 20	Minimum : 0

	c'est normal IMC entre 25 et 30 situer dans une plage surpoids Et de 30 ou plus situer dans la fourchette d'obésité	Maximum : 80.6
DiabetePredigmeFunction	Fournit des informations sur les antécédentes chez les parents et la relation génétique avec les patients. Une fonction de pedigree plus élevée signifie que le patient plus susceptible de souffrir un diabète	Minimum : 0.078 Maximum : 2.42
Age	Age d'une personne en années	Minimum : 21 Maximum : 81
Outcome	Indique si une personne est diabétique ou non	0 (non diabétique) 1 (diabétique)

Table 3.1- Définition des variables

3.5 L'étude technique de la prédiction des diabètes type 2

Voici le schéma qui été suivi à cet implémentation

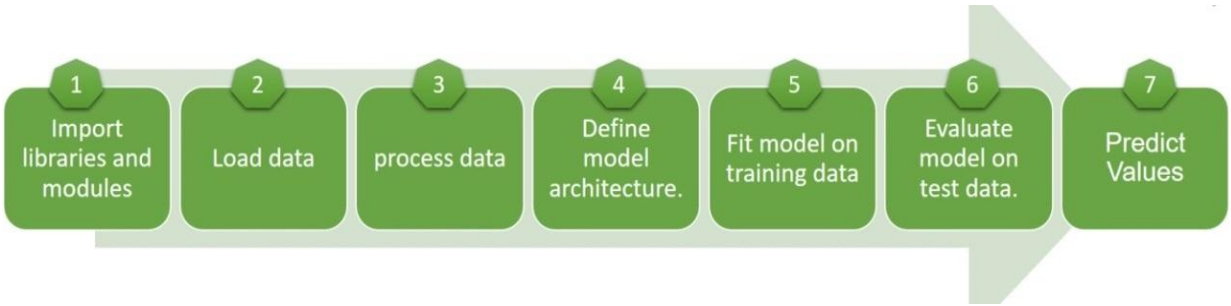


Figure 3.2-schéma d'implémentation de prédiction sur python [25]

3.5.1 Importation des Librairies

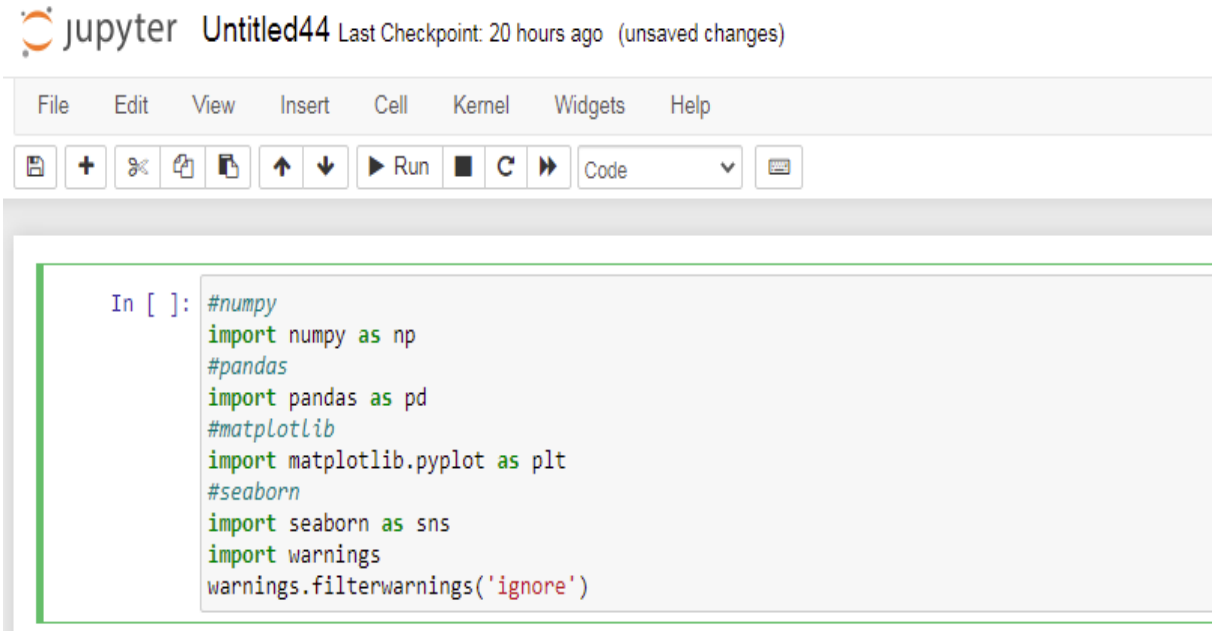


Figure 3.3- Importer les Librairies

3.5.2 Téléchargement des données

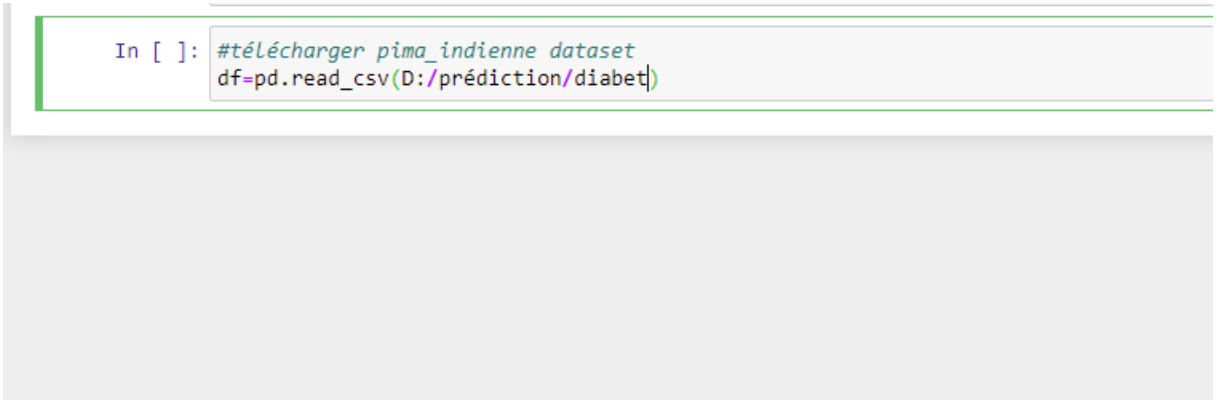


Figure 3.4 - Télécharger les données

3.5.3 Manipulation des données

```
In [6]: df.head()
```

Out[6]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Figure 3.5-Explorer les premiers (05) records de la data set

```
In [5]: # Display last five records of the data
        df.tail()
```

Out[5]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

Figure3.6-Explorer les derniers (05) records du data

```
In [4]: # Display first five records of data
        df.head(10)
```

Out[4]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
5	5	116	74	0	0	25.6	0.201	30	0
6	3	78	50	32	88	31.0	0.248	26	1
7	10	115	0	0	0	35.3	0.134	29	0
8	2	197	70	45	543	30.5	0.158	53	1
9	8	125	96	0	0	0.0	0.232	54	1

Figure 3.7-Explorer les premiers 10 records du data

```
In [9]: df.sample(10)
```

Out[9]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
754	8	154	78	32	0	32.4	0.443	45	1
381	0	105	68	22	0	20.0	0.236	22	0
506	0	180	90	26	90	36.5	0.314	35	1
282	7	133	88	15	155	32.4	0.262	37	0
302	5	77	82	41	42	35.8	0.156	35	0
107	4	144	58	28	140	29.5	0.287	37	0
755	1	128	88	39	110	36.5	1.057	37	1
745	12	100	84	33	105	30.0	0.488	46	0
231	6	134	80	37	370	46.2	0.238	46	1
170	6	102	82	0	0	30.8	0.180	36	1

figure 3.8-Explorer (10) records Aléatoire du data

```
In [ ]: #Number of rows and columns
        df.shape
```

No.of Rows=768

No.of Columns=9

Figure 3.9-Explorer le nombre des colonnes et des ligne dans le data

```
In [11]: df.dtypes
```

```
Out[11]: Pregnancies                 int64
         Glucose                     int64
         BloodPressure               int64
         SkinThickness               int64
         Insulin                     int64
         BMI                       float64
         DiabetesPedigreeFunction  float64
         Age                         int64
         Outcome                     int64
         dtype: object
```

Figure 3.10-Explorer les type des tous les colonnes de data

```
In [12]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Pregnancies               768 non-null    int64
 1   Glucose                   768 non-null    int64
 2   BloodPressure             768 non-null    int64
 3   SkinThickness             768 non-null    int64
 4   Insulin                   768 non-null    int64
 5   BMI                       768 non-null    float64
 6   DiabetesPedigreeFunction  768 non-null    float64
 7   Age                       768 non-null    int64
 8   Outcome                   768 non-null    int64
dtypes: float64(2), int64(7)
memory usage: 54.1 KB
```

Figure 3.11-Explorer des informations sur le data

Des informations de chaque colonnes de data, le nombre des records et le nombre des colonnes

```
In [12]: # Statistical summary
         df.describe()
```

Out[12]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876	33.240885	0.348958
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329	11.760232	0.476951
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	21.000000	0.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243750	24.000000	0.000000
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.372500	29.000000	0.000000
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	0.626250	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000	1.000000

Figure 3.12-Aperçudes valeur numériques statistiques

comme (min ,max , mean…) à chaque colonne notez que le minimum de tous les colonnes est zéro donc il faut les nettoyer

➢ Nettoyage de data

```
In [14]: df=df.drop_duplicates()

In [15]: # check the shape after drop the duplicates
         df.shape

Out[15]: (768, 9)
```

Figure 3.13-Supprimer les redoublant de la data

notez qu'ils n'existe Pas p-c-q il reste les mêmes nombre des lignes et colonnes

```
In [16]: df.isnull().sum()

Out[16]: Pregnancies                 0
         Glucose                     0
         BloodPressure               0
         SkinThickness               0
         Insulin                     0
         BMI                         0
         DiabetesPedigreeFunction    0
         Age                         0
         Outcome                     0
         dtype: int64
```

Figure 3.14-Compter les valeurs manquantes dans chaque colonne,

Notez qu'il n'existe aucune valeur manquante dans cet data

```
In [18]: print('No. of zero values in Glucose ',df[df['Glucose']==0].shape[0])
         No. of zero values in Glucose  5

In [19]: print('No. of zero values in BloodPressure ',df[df['BloodPressure']==0].shape[0])
         No. of zero values in BloodPressure  35

In [20]: print('No. of zero values in SkinThickness ',df[df['SkinThickness']==0].shape[0])
         No. of zero values in SkinThickness  227

In [21]: print('No. of zero values in Insulin ',df[df['Insulin']==0].shape[0])
         No. of zero values in Insulin  374
```

Figure 3.15-Vérifier l'emplacement des valeurs (nul) à chaque colonne

```
In [23]: df['Glucose']=df['Glucose'].replace(0,df['Glucose'].mean())
         print('No. of zero values in Glucose ',df[df['Glucose']==0].shape[0])
         No. of zero values in Glucose  0
```

Figure 3.16-Remplacer tous les zéro par la moyenne (mean) de colonne

En suite vérifier l'existence des zéro sur la colonne

```
In [24]: df.describe()
```

Out[24]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	4.400782	121.681605	72.254807	26.606479	118.660163	32.450805	0.471876	33.240885	0.348958
std	2.984162	30.436016	12.115932	9.631241	93.080358	6.875374	0.331329	11.760232	0.476951
min	1.000000	44.000000	24.000000	7.000000	14.000000	18.200000	0.078000	21.000000	0.000000
25%	2.000000	99.750000	64.000000	20.536458	79.799479	27.500000	0.243750	24.000000	0.000000
50%	3.845052	117.000000	72.000000	23.000000	79.799479	32.000000	0.372500	29.000000	0.000000
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	0.626250	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000	1.000000

Figure 3.17-Vérifier le minimum de toute la colonne notez qu'ils sont déférents de zéro

➢ Visualisation des donnés

```
In [20]: f,ax=plt.subplots(1,2,figsize=(10,5))
         df['Outcome'].value_counts().plot.pie(explode=[0,0.1],autopct='%1.1f%%',ax=ax[0],shadow=True)
         ax[0].set_title('Outcome')
         ax[0].set_ylabel('')
         sns.countplot('Outcome',data=df,ax=ax[1])
         ax[1].set_title('Outcome')
         N,P= df['Outcome'].value_counts()
         print('Negative (0): ',N)
         print('Positive (1): ',P)
         plt.grid()
         plt.show

         Negative (0):  500
         Positive (1):  268

Out[20]: <function matplotlib.pyplot.show(close=None, block=None)>
```

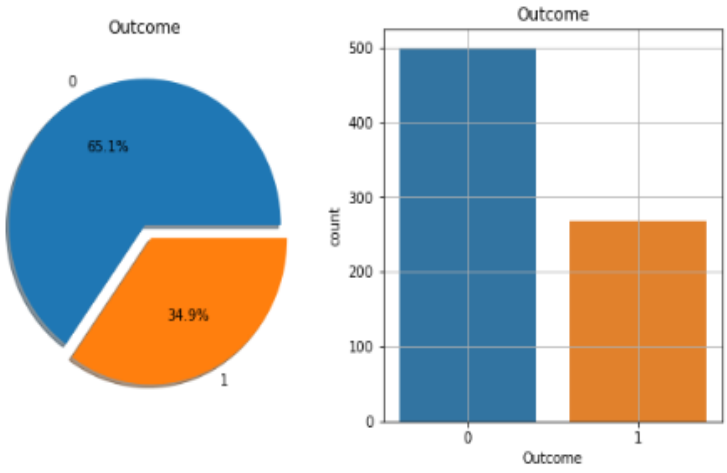


Figure 3.17- le graphe de comptage des colonnes

```
In [21]: df.hist(bins=10,figsize=(10,10))
         plt.show()
```

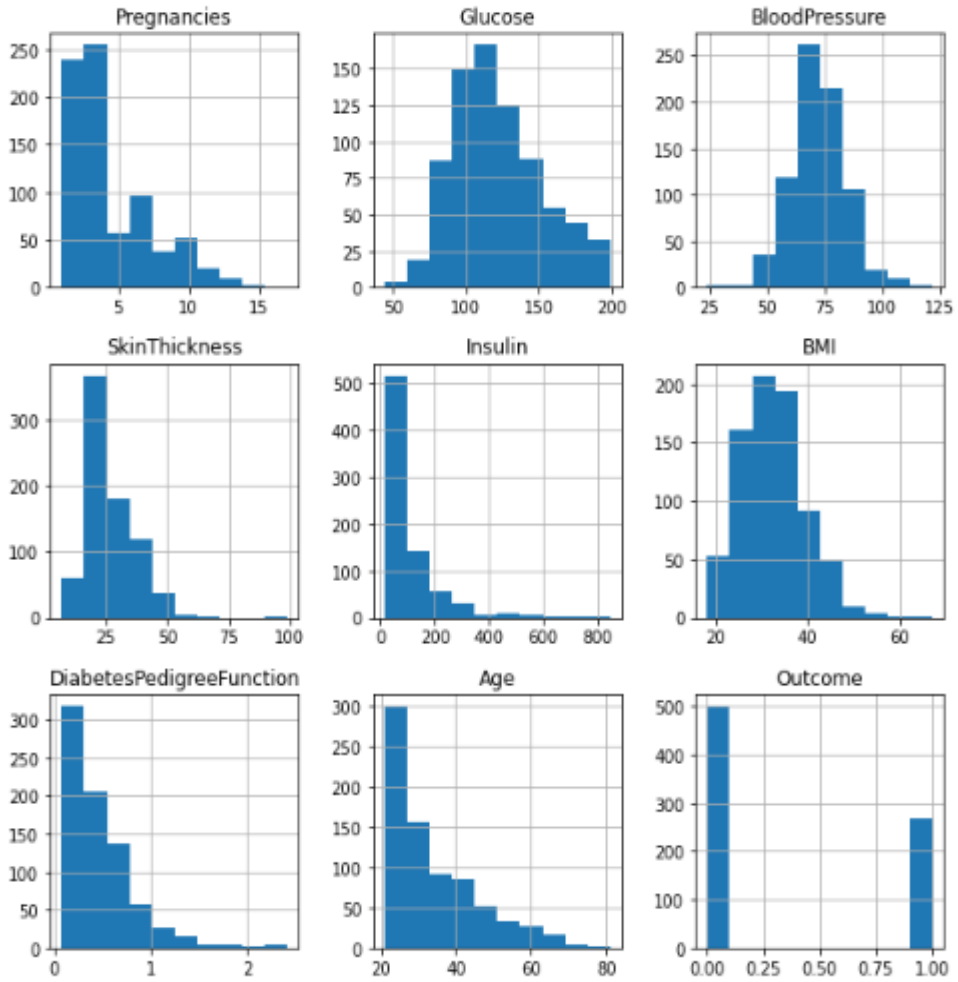


Figure 3.18 le graphe histogramme des colonnes

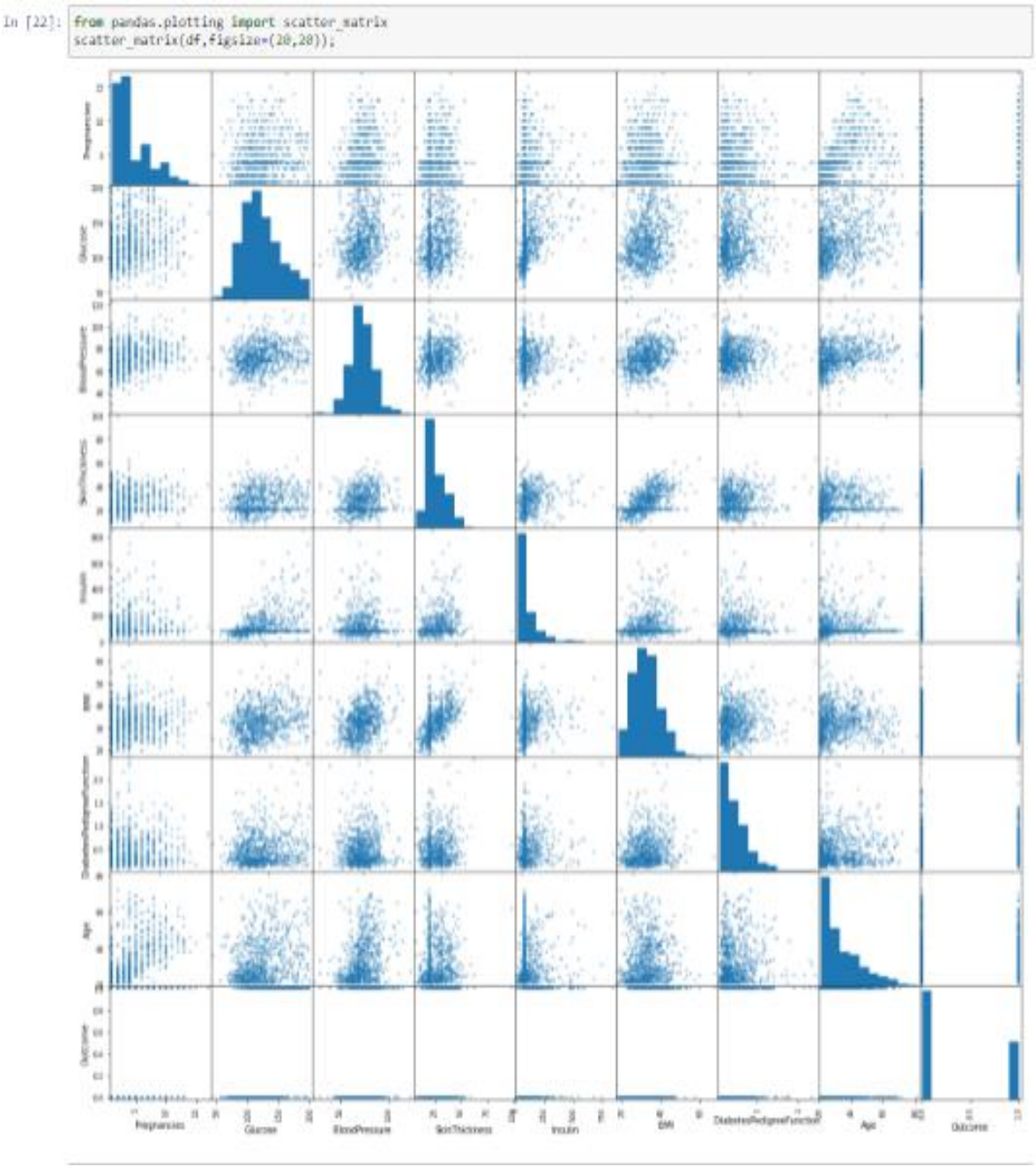


Figure 3.19- la matrice des graphes (scatterplot)

Ce graphe représente la relations entre chaque deux colonnes celui aidez à comprendre la matrice de corrélation

- **La matrice de corrélation :**

Est une donnée tabulaire chaque ligne et colonne représente une variable et chaque valeur de cette matrice est le coefficient de corrélation entre les variables représentées par la ligne et la colonne correspondantes. La matrice de corrélation est une métrique d'analyse de données importante qui est calculée pour résumer les données afin de comprendre la relation entre diverses variables et prendre des décisions en conséquence.

Les valeurs proches de +1 indiquent la présence d'une forte relation positive entre X et Y, tandis que celles proches de -1 indiquent une forte relation négative entre X et Y.

Des valeurs proches de zéro signifient qu'il n'y a aucune relation entre X et Y

```
In [2]: import seaborn as sns
corrmat=df.corr()
top_corr_features=corrmat.index
plt.figure(figsize=(10,10))
g=sns.heatmap(df[top_corr_features].corr(),annot=True,cmap="RdYlGn")
```

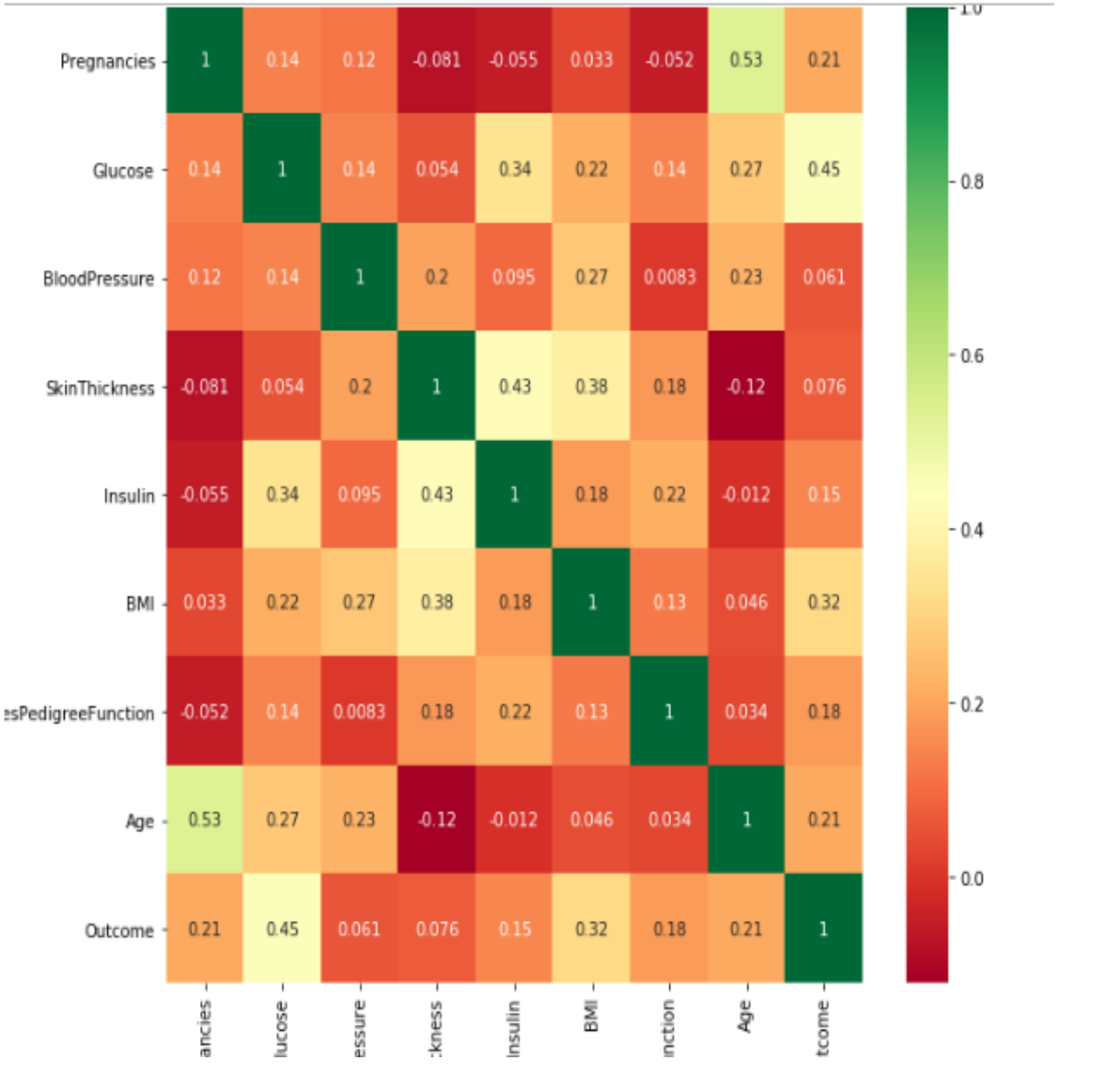


Figure 3.20 - La matrice de corrélation

➢ Diviser le data entre X et Y :

X : pour les colonnes d'entrer

Y : pour la colonne de outcome

```
In [27]: target_name ='Outcome'
         y=df[target_name]
         X=df.drop(target_name,axis=1)
```

```
In [28]: X.head()
```

Out[28]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age
0	6.000000	148.0	72.0	35.000000	79.799479	33.6	0.627	50
1	1.000000	85.0	66.0	29.000000	79.799479	26.6	0.351	31
2	8.000000	183.0	64.0	20.536458	79.799479	23.3	0.672	32
3	1.000000	89.0	66.0	23.000000	94.000000	28.1	0.167	21
4	3.845052	137.0	40.0	35.000000	168.000000	43.1	2.288	33

Figure 3.21-Apercu la division des données

➢ Scaling les caractéristiques du data

Est une méthode utilisée pour normaliser la plage des variables indépendantes elle est également connue sous le nom de normalisation des données Il existe déférents technique de normalisation mais on a choisi La standard technique

```
In [ ]: # Apply Standard Scaler
        from sklearn.preprocessing import StandardScaler
        scaler = StandardScaler()
        scaler.fit(X)
        SSX = scaler.transform(X)
```

Figure 3.22 Aperçu la normalisation du data

- **Diviser le data entre deux partie (train ,test)**

```
In [35]: from sklearn.model_selection import train_test_split
         X_train, X_test, y_train, y_test = train_test_split(SSX, y, test_size=0.2, random_state=7)

In [36]: X_train.shape,y_train.shape
Out[36]: ((614, 8), (614,))

In [37]: X_test.shape,y_test.shape
Out[37]: ((154, 8), (154,))
```

Figure 3.23 Aperçu la division des données

Donc on a les tableaux suivant

	X	Y
train	**(614 , 8)**	**614**
test	**(154 , 8)**	**154**

Table 3.2 – Aperçu la division du data

3.5.4 Définir les modèles et passer les données pour trainer

S

```
In [39]: from sklearn.neighbors import KNeighborsClassifier
         knn=KNeighborsClassifier()
         knn.fit(X_train, y_train)
Out[39]: KNeighborsClassifier()

In [41]: from sklearn.svm import SVC
         sv=SVC()
         sv.fit(X_train,y_train)
Out[41]: SVC()

In [40]: from sklearn.naive_bayes import GaussianNB
         nb=GaussianNB()
         nb.fit(X_train, y_train)
Out[40]: GaussianNB()
```

R

```
In [42]: from sklearn.tree import DecisionTreeClassifier
         dt=DecisionTreeClassifier()
         dt.fit(X_train,y_train)

Out[42]: DecisionTreeClassifier()

In [43]: from sklearn.ensemble import RandomForestClassifier
         rf=RandomForestClassifier(criterion='entropy')
         rf.fit(X_train,y_train)

Out[43]: RandomForestClassifier(criterion='entropy')
```

Figure 3.24- Définir les (05) modèles

3.5.5 Faire la prédiction à chaque modèle par la partie de test

```
In [47]: ## Making predictions on test dataset
         knn_pred=knn.predict(X_test)
```

3

```
In [49]: knn_pred.shape

Out[49]: (154,)
```

Figure 3.25-Faire la prédiction par les modèles

3.5.6 Evaluer les modèles

```
In [54]: # Train score & Test score of KNN
         print("Train Accuracy of KNN",knn.score(X_train,y_train)*100)
         print("Accuracy (Test) score of KNN",knn.score(X_test, y_test)*100)
         print("Accuracy score of KNN",accuracy_score(y_test,knn_pred)*100)

         Train Accuracy of KNN 81.10749185667753
         Accuracy (Test) score of KNN 74.67532467532467
         Accuracy score of KNN 74.67532467532467
```

KNN // La précision = 74.67

```
In [56]: # Train score & Test score of SVM
         print("Train Accuracy of SVM",sv.score(X_train,y_train)*100)
         print("Accuracy (Test) score of SVM",sv.score(X_test, y_test)*100)
         print("Accuracy score of SVM",accuracy_score(y_test,sv_pred)*100)

         Train Accuracy of SVM 81.92182410423453
         Accuracy (Test) score of SVM 83.11688311688312
         Accuracy score of SVM 83.11688311688312
```

SVM // la précision = 83.11

```
In [55]: # Train score & Test score of Naive-Bayes
         print("Train Accuracy of Naive Bayes",nb.score(X_train,y_train)*100)
         print("Accuracy (Test) score of Naive Bayes",nb.score(X_test, y_test)*100)
         print("Accuracy score of Naive Bayes",accuracy_score(y_test,nb_pred)*100)

         Train Accuracy of Naive Bayes 74.2671009771987
         Accuracy (Test) score of Naive Bayes 74.02597402597402
         Accuracy score of Naive Bayes 74.02597402597402
```

Naive_Bayes // la précision = 74.02

```
In [57]: # Train score & Test score of Decesion Tree
         print("Train Accuracy of Decesion Tree",dt.score(X_train,y_train)*100)
         print("Accuracy (Test) score of Decesion Tree",dt.score(X_test, y_test)*100)
         print("Accuracy score of Decesion Tree",accuracy_score(y_test,dt_pred)*100)

         Train Accuracy of Decesion Tree 100.0
         Accuracy (Test) score of Decesion Tree 79.22077922077922
         Accuracy score of Decesion Tree 79.22077922077922

In [58]: # Train score & Test score of Random Forest
         print("Train Accuracy of Random Forest",rf.score(X_train,y_train)*100)
         print("Accuracy (Test) score of Random Forest",rf.score(X_test, y_test)*100)
         print("Accuracy score of Random Forest",accuracy_score(y_test,rf_pred)*100)

         Train Accuracy of Random Forest 100.0
         Accuracy (Test) score of Random Forest 81.16883116883116
         Accuracy score of Random Forest 81.16883116883116
```

DecesionTree // la précision = 79.22

Random_Forest // la précision = 81.16

Figure 3.26 – évaluation des (05) modèle

Donc la Meilleur modèle c'est SVM avec précision de 83.11

3.6 L'application

Ci-dessous, nous fournissons nos interfaces d'application **"diabet_ prediction"** dans le but de permettre aux personnes des savoir s'ils ont le risque de développer un diabète avec un taux de prédiction bien défini .L'application avoir un adresse local après l'installation

http://127.0.0.1:5000/

Variable	Description	La plage
Pregnancies	Nombre de fois enceinte	Minimum : 0 Maximum : 17
BloodPressure	Si un TA diastolique > 90 signifie une pression artérielle élevé (probabilité élevée de diabète) Un TA diastolique < 60 signifie une pression artérielle base (mois probabilité de diabète)	Minimum : 0 Maximum : 122
SkinThikness	Valeur estimé pour la graisse corporelle. épisure normal du pli cutané chez les femmes est de 23 mm. Une épissure plus élevée conduit à l'obésité et les chances de diabète augmente.	Minimum : 0 Maximum : 110
Insulin	Insuline sérique 2 heures (mu U/ ml) et niveau d'insuline normal 16-166 mUI/L, les valeurs au-dessus de cette plage peuvent être alarmante	Minimum : 0 Maximum : 799
BMI	(poids en kg / taille en m2) IMC de 18.5 à 20 c'est normal IMC entre 25 et 30 situer dans une plage surpoids Et de 30 ou plus situer dans la fourchette d'obésité	Minimum : 0 Maximum : 80.6
DiabetePre digmeFunction	Fournit des informations sur les antécédentes chez les parents et la relation génétique avec les patients. Une fonction de pedigree plus élevée signifie que le patient plus susceptible de soufffrir un diabète	Minimum : 0.078 Maximum : 2.42
Age	Age d'une personne en années	Minimum : 21 Maximum : 81
Outcome	Indique si une personne est diabétique ou non	0 non diabétique 1 (diabétique)

Figure 3.27 – l'interface de l'application

La page d'accueil

La page d'accueil fournit des informations explicatives générales pour les diabétiques, ou nous expliquons leur hyper sensibilité à l'épidémie que nous connaissons actuellement. Cette page fournit des hyperliens vers les autres interfaces que constitue notre application Web. Ainsi que la forme de la prédiction

La forme de la prédiction

Figure 3.28 –la forme de prédiction

Le but de cette interface est de prédire si une personne est diabétique ou non avec un taux de prédiction, pour cela il doit remplir le formulaire ci-dessous qui contient les informations suivantes : Pregnancies, Glucose, BloodPressure, SkinThikness, Insulin, BMI, DiabetePedigreeFunction et Age.

La forme de resultat

Figure 3.29 – la forme de résultat

3.7 Conclusion

Dans ce chapitre, nous avons présenté les différents étapes de prétraitement des données tels que l'exploration et la visualisation des données ainsi le nettoyage des valeurs aberrantes. L'application des méthodes d'évaluation nous permet de sélectionner le modèle **SVM** comme le meilleur modèle qui a un taux de précision élevé. A la fin, on a développé une application web qui nous permet de prédire si une personne

Donnée est diabétique ou pas à partir de ces informations médicales

3.8 Conclusion générale

Dans cette Mémoire nous avons mené à faire une comparaison entre cinq algorithmes d'apprentissage automatique à savoir : l'arbre de décision, Randomforest, naive bayes, K nearest neighbors et support vector machine, les résultats expérimentaux obtenu pour l'ensemble de montre que support vector machine est meilleure que les autres algorithmes en terme de sa grande précision. Sur la base d'un algorithme support vector machine et que nous avons besoin d'un moyen pour rendre le modèle applicatif pour tout le monde nous avons développé une solution basé sur une application web dans le but d'aide les personnes de prédire s'il souffre de diabète En termes de perspective :

1. la construction d'une application Android parallèle avec notre application web permet d'aide les personnes qui sont diabétiques de suivre son situation médical, les médicaments, les rendez-vous médical ainsi leur état physique comme le sport et le régime alimentaire équilibré pour leur cas.

2. La prédiction de diabète avec l'approche de deep learning peut avoir
   Un bon résultat de précision

Bibliographie

[1] Organisation mondiale de la santé.(2016).Rapport mondial sur le diabète.88p

[2] Medtronic .Le Diabète En Quelque Mots.[en ligne].Disponible sur :https ://www.parlonsdiabete.com/parlons-diabete/le-diabete-en-quelques-mots

[3] fédération français de Cardiologie .Réduire le risque cardio-vasculaire LE

[4] CEED :Centre européen d´´etude du Diabète .Diabètes et complications.[en

[5] La macro angiopathie diabétique. Complications.[en ligne].(Mis `a jour en juin2015).Disponible sur :http://www.hegp.fr/diabeto/complicationmacro.html

[6] Doctissimo. Micro-angiopathie.[en ligne].Disponible sur : https ://www.doctissimo.fr/sante/dictionnaire-medical/micro-angiopathie

[7] CEED : Centre européen d´´etude du Diabète.le d´dépistage au cœur des actions de prévention.[en ligne].Disponible sur : http ://ceed-diabete.org/blog/diabete-ledepistage-au-coeur-des-actions-de-prevention/

[8] https://www.inprincipio.xyz/machine-learning/ (consulter 03/06/2020)

[9] Lev Kiwi.(2018).Apprentissage et Machine Learning.[en ligne].Disponible sur :
https ://levkiwi.ch/apprentissage-et-machine-learning/ (consulter 04/06/2020)
[10] Pensée Artificielle.Machine Learning pour d'ébutant : Introduction au Machine Learning.[en ligne].Disponible sur :http ://penseeartificielle.fr/introduction-au-machinelearning/ (consulter 04/06/2020)
[11] Ga¨el, P.Makina Corpus.(2017).Initiation au Machine Learning avec Python.[enligne].Disponible sur : https ://makina-corpus.com/blog/metier/2017/initiation-aumachine-learning-avec-python-pratique
[12] IlemonaS.Atawodi.(2019).A Machine Learning Approach to Network Intrusion Detection System Using K NearestNeighbor and Random Forest. de master : universit´e de Southern nearest-neighbor-classification-scikit-learn
[13] Benzaki, Y.Mr Mint .(2018).Introduction `a l'algorithme K Nearest Neighbors (KNN).[en ligne].Disponiblesur : https ://mrmint.fr/introduction-k-nearest-neighbors
[14] Gupta, P.to wards datascience.(2017).Decision Trees in Machine Learning.[enligne].Disponiblesur :https ://towardsdatascience.com/decision-trees-in-machinelearning-641b9c4e8052
[15] Choudhury, A.Analytics India Magazine.(2019).Beginner's Guide To Decisionsur : https //analyticsindiamag.com/beginners-

guide-to-decision-trees-why-arethey-crucial-for-data-science-applications/decision-trees-with-code-dc026172a284
[16] Ismaili, Z.(2019).Le Data Scientist . Apprentissage Supervisé Vs. Non Supervisé.[enligne].Disponible sur : https ://le-datascientist.fr/apprentissage-supervise-vs-nonsupervise neighbors-algorithm-6a6e71d01761
[17] java T point.Random Forest Algorithm.[en ligne].Disponible sur :https ://www.javatpoint.com/machine-learning-random-forest-algorithm
[18] tutorials point.Classification Algorithms - Random forest.[en ligne].Disponiblesur : https ://www.tutorialspoint.com/machine learning with python /machine learning with python classification algorithms random forest.htm
[19] java T point.Support Vector Machine Algorithm.[en ligne].Disponiblesur :https ://www.javatpoint.com/machine-learning-support-vector-machine-algorithm
[20] Tandel, A.to wards data science.(2017).Support Vector Machines | A Brief Overview.[en ligne].Disponiblesur :https ://towardsdatascience.com/support-vectormachines-a-brief-overview-37e018ae310f
[21] Sharma, N.heartbeat.Understanding the Mathematics behind Support Vector Machines.[en ligne].Disponiblesur : https ://heartbeat.fritz.ai/understanding-themathematics-behind-support-vector-machines-5e20243d64d5
[22] Gandhi, R.to wards data science.(2018).Support Vector

Machine | Introduction to Machine Learning Algorithms.[en ligne].Disponiblesur :https ://towardsdatascience.com/support-vector-machine-introduction-to-machinelearning-algorithms-934a444fca 47

[23] laptrinhx.(2019).Naive Bayes Unlocked.[en ligne].Disponible sur : https ://laptrinhx.com/naive-bayes-unlocked-1301819179/

[24] MEHIDI, D.MEDJOUDJ, S. (2018).Application des Méthodes d'Apprentissagedans la Prédiction du Diabète de Type 2. mémoire master :Intelligence Artificielle.Département Informatique. Faculté des Sciences Exactes.Université A.MIRA de Bejaia.79p

[25] Ebrahim, M.like geeks.(2020). Python Correlation Matrix Tutoriel.(mise a jour29-07-2020).[en ligne].Disponible sur : https ://likegeeks.com/python-correlationmatrix/

[26] Jupyter.(2017).Project Jupyter .[en ligne].Disponible sur :https ://jupyter.org/

[27] ResearchGate.Train-Test Data Split .[en ligne].Disponible sur :https ://www.researchgate.net/figure/Train-Test-Data-Split

Annexe

hémoglobineglyquée (HbA1c)

Ce test représente une "mémoire" globale de la glycémie sur les trois derniers mois, ilprend donc en compte tous les états, y compris la glycémie postprandiale.

autosurveillance glycémique (ASG)

autosurveillance glycémique est prescrite par le médecin en fonction du votre type dediabète et de votre type de traitement.elle est indispensable dans le diabète de type 1,nécessaire dans le diabète de type 2 insulinotraitéet variable pour les diabétiques de type2 non insulinotraités.cette autosurveillance sert principalement `a contrôler et prévenir lesdéséquilibres (hypo/hyperglycémies) et à adapter votre traitement. Elle permet aussi demesurer l'effet d'un aliment, d'une pratique sportive ou d'une activité physique sur saglycémie.