Machine learning : Les algorithmes d'apprentissage

Lien vers le cours : https://www.linkedin.com/learning/machine-learning-les-algorithmes-d-apprentissage/bienvenue-dans-machine-learning-les-algorithmes-d-apprentissage

# Introduction

## Bienvenue dans « Machine learning : Les algorithmes d'apprentissage »

Le machine learning a montré ces dernières annéesqu'il était une technique extrêmement performante pour réaliserles tâches cognitives diverses.Il permet de reconnaître des objets dans des images,traduire automatiquement des textes,jouer aux échecs ou au go, et plus généralementclasser des observations à partir d'exemples.Mais la mise en œuvre de tout cela nécessite de maîtrisertout un ensemble de concepts.Et c'est justement ce que nous verronsensemble à travers la présentationde modèles de machine learning,et de techniques pour les apprendre à partir de données,ainsi que leur utilisation sur un exemple concretprogrammé en Python, et utilisant la librairiescikit-learn. Je m'appelle Céline Robardet, je suis professeur à l'Insa Lyon,et ensemble nous apprendrons comment utiliserles techniques de machine learning dans vos projets.Vous êtes prêt ? Alors allons-y !

# Découvrir l’intelligence artificielle et le machine learning

## Définir l'IA en quelques mots

Par le développement des connaissances scientifiques,l'homme cherche à comprendre le monde et son fonctionnement.Appartenant au monde, l'Homme fait naturellement partiedu champ d'étude de la science.La science s'est alors intéressée depuis des milliers d'annéesà comprendre comment une telle connaissance est élaborée.Quels sont les mécanismes qui sous-tendent le raisonnement ?Comment, à partir d'une perception de l'environnement,est-on capable de comprendre une situation ?Le domaine de l'intelligence artificielle va encore plus loin,en cherchant à reproduire ces comportements intelligentsde manière artificielle, c'est-à-direavec des ordinateurs.Il s'agit de créer des machines qui réalisentdes processus cognitifs, les tâches que notre cerveauréalise continuellement de manière autonome dans des environnementscomplexes et changeants.L'intelligence artificielle englobe actuellement une grandevariété de sous-domaines allant des plus générauxtels que l'apprentissage, la perception,à des tâches plus spécifiques, telles que jouer aux échecs,prouver des théorèmes mathématiques,écrire de la poésie, diagnostiquer des maladies.L'intelligence artificielle systématise et automatiseles tâches intellectuelles, et est donc potentiellementpertinente pour tout domaine d'activité intellectuelle.En ce sens, c'est vraiment un champ universel.L'intelligence artificielle s'est développée à partir des années50 selon deux axes principaux.Le premier axe oppose les approches selonqu'elles visent à imiter l'homme ou sa rationalité.Alors que l'autre axe oppose pensée et action.Une tension existe entre les approches centréessur l'Homme et les approches centrées sur sa rationalité.Une approche centrée sur l'humain doit être une approche empirique,impliquant hypothèses et confirmations expérimentales.Alors qu'une approche rationaliste implique une combinaison d'outilsmathématiques et d'ingénierie.Chaque groupe a à la fois décrié et aidé l'autre.Le test proposé par Alan Turing en 1950 a pour but de fournirune définition opérationnelle de l'intelligence artificielle.Face à la pluralité des définitions de l'IA,dans un lieu souvent à polémiques et controverses,il a suggéré un test basé sur l'indiscernabilité des entitésindéniablement intelligentes que sont les êtres humains.L'IA réussit le test si un interrogateur humain,après avoir posé quelques questions,ne peut pas dire si les réponsesproviennent d'une personne ou d'une machine.Notons que programmer un ordinateur capable de réussirce test nécessite d'être capable de traiter le langage naturelpour permettre à la machine de communiquer avec l'homme,de représenter des connaissances pour stocker le savoiret permettre de raisonner automatiquement,afin de répondre aux questions, et de tirer de nouvelles conclusions.Enfin la machine doit être capable d'apprendreet de s'adapter à de nouvelles circonstances.Dans les années 50/60, le développementde l'IA est fulgurant.Par exemple, en 1955, Newell et Simon conçoiventle programme Logic theorist qui permet de démontrerautomatiquement 38 desMathematica d'Alfred North Whitehead et Bertrand Russel.Résultats extrêmement impressionnants puisquepour la première fois une machine est capable de raisonnement.Puis l'IA rencontre des difficultés.Les premiers systèmes mis au point sur de petits problèmesjoués s'avèrent incapables de passer à l'échelle pour traiterdes problèmes plus difficiles.Ces premiers systèmes comportent peu ou pas de connaissances,effectuent de simples manipulations syntaxiques.Les succès initiaux ont été possibles parce queles problèmes étaient réduits, et trouver la solutionnécessitait de considérer une poignée de combinaisons.Plutôt que de produire un système intelligent unique,le développement de l'IA s'est focalisé sur l'élaborationde machines qui réalisent des tâches spécifiques,souvent mieux et plus vite que l'être humain.Des exemples de ces applications sont la conduitede voitures autonomes, la traduction automatique,les assistants personnels intelligents,comme par exemple Siri, la reconnaissance faciale,les jeux (échecs, go), la détection de fraudes,les systèmes de recommandation, les systèmes d'aideau diagnostic. L'apprentissage automatique permet de réaliser une grandepartie de ses tâches spécialisées.C'est pourquoi le machine learning est si populaire actuellement.Pour résumer, il y a eu de nombreuses tentativesde construction d'une machine intelligente qui remontetrès loin dans le passé.On s'est alors rendu compte que de nombreux stagesqui sont réalisés de manière assez simple par des humainssont très difficiles à faire réaliser par un ordinateur.Ainsi les ambitions extravagantes du début sont généralementrestées non atteintes.Aujourd'hui, les ordinateurs peuvent réussir et parfoismême mieux que les humains de nombreux problèmes spécifiques.Pour cela, les algorithmes de machine learning sontles plus performants.

## Apprendre pour une machine

Considérons certaines des choses que nous faisonsau quotidien sans effort : reconnaître visuellementdes personnes, des lieux, des objets,comprendre le langage parlé.Comment peut-on programmerune machine pour faireces choses ? Il est difficile d'établir un programme informatiqueeffectuant ces activités, car nous n'avons que trèspeu de conscience introspective du fonctionnement de notre cerveau.Comment reconnaissons-nous un ami ?Quels traits du visage sont caractéristiques d'une personne ?Les chercheurs en intelligence artificielle ont essayé pendantdes décennies de mettre au point des procédures de calcul pour cetype de tâches, et ceci s'est avéré extrêmement difficile.L'apprentissage automatique adopte une approche différente :collecter de nombreuses données et faire en sorte qu'un algorithmedétermine automatiquement un bon comportement à partir des données.Si vous essayez de décrire un programme pour distinguerdifférentes catégories d'objets, par exemple des arbreset des chiens, vous pouvez d'abord collecterun ensemble d'images de chaque type d'objet d'objets,puis utiliser un algorithme d'apprentissage automatiquepour former un modèle qui classera une imagedans une catégorie ou une autre.Peut-être que votre modèle apprendra à voir d'une manièreanalogue au système visuel humain, ou peut-être adoptera-t-ilune approche totalement différente.Quoi qu'il en soit, le processus entier peut êtrebeaucoup plus facile que de spécifier à la main les caractèresdécrivant chacune des classes.En plus d'être plus facile, il y a deux autres raisonspour lesquelles nous pourrions vouloir utiliser l'apprentissagemachine pour résoudre un problème donné.Un système peut devoir s'adapter à un environnement en mutation.Par exemple, les programmes générant des spamsessaient constamment de trouver des moyensde tromper nos classificateurs de courriers électroniquespour spams. Les algorithmes de classification doiventdonc s'adapter en permanence.Un algorithme d'apprentissage peut être plus performantque ses programmeurs humains.Les algorithmes d'apprentissage sont devenus des championsdu monde dans des jeux comme les dames,les échecs, récemment le jeu de go.Cela serait impossible si les programmes ne faisaientque ce qu'on leur avait explicitement dit de faire.De manière assez similaire à la façon dont les humains apprennent,les machines utilisent des exemples associés à unedécision pour apprendre un modèle que l'on peut ensuiteutiliser pour déterminer la meilleure décisionà associer à de nouveaux cas.C'est ce qu'on appelle la capacité de généralisation de la méthode.Bien entendu, l'apprentissage humain est bien plussophistiqué que même les algorithmes d'apprentissageautomatique les plus avancés.Mais les ordinateurs ont l'avantage de disposerd'une grande capacité de mémorisation et detraitement de données.Les expériences se présentent sous la forme de données historiques,et ils l'utilisent afin de créer et d'optimiser un modèle qui ala possibilité de généraliser.Selon la définition de Tom Mitchell,une machine apprend si ses performancesà réaliser une tâche donnée,telle que mesurée par un score calculable,s'améliore avec l'expérience, c'est-à-dire avecle nombre d'exemples considérés.Il existe trois types différents d'apprentissage automatique.En apprentissage supervisé, nous avons des exemplesdu comportement souhaité.Supposons que l'on essaye de reconnaître des imagesd'arbres et de chiens, nous allons Tout d'abordcollecter des images d'arbres et de chiensen utilisant le fait que nous savonsquelle image représente un arbre, et quelle autre représente un chien.Nous apprenons un modèle qui est ensuite utilisépour classer de nouvelles images.Pour l'apprentissage par renforcement,nous n'avons pas d'exemple de comportement,mais nous disposons d'une méthode permettant de déterminerla qualité d'un comportement.C'est ce que l'on appelle un signal de récompense.Par analogie, on peut penser au chien que l'on entraînepour exécuter des tours.Un exemple serait de former un agent à jouer à des jeux vidéo,où le signal de récompense est le score du joueur.Enfin, dans l'apprentissage non supervisé,nous n'avons ni étiquette ni signal de récompense.Nous ne disposons que de nombreuses données,et souhaitons rechercher des modèles dans ces données.Par exemple, nous avons peut-être beaucoup d'exemplesde patients atteints d'autisme, et souhaitons identifierles différents sous-types de la maladie.Ce cours se concentre sur l'apprentissage supervisé.Il s'agit du type d'apprentissage automatique le mieux compris,car par rapport à l'apprentissagenon supervisé, ou par renforcement,les problèmes d'apprentissagesupervisé sont beaucoup plus facilesà formuler de manière mathématique.Un algorithme de machine learning consiste à apprendreles paramètres d'un modèle à partir d'exemples.Un exemple est une description numérique d'un objet associéà une catégorie ou une étiquette.Et l'algorithme ajuste les paramètres du modèleen fonction d'un ensemble d'exemples appeléséchantillons d'apprentissage.Une fois le modèle appris, il peut être utilisé pour prédirel'étiquette de nouveaux objets.On peut mesurer les performances du modèle sur un deuxièmeensemble d'exemples appelés échantillons de test.Il existe de nombreux algorithmes d'apprentissagesupervisé très efficaces et largement applicables,dont beaucoup seront abordés dans ce cours.Pour conclure, on peut se demander pourquoi le machinelearning suscite actuellement un tel intérêt.Il y a bien entendu la réussite très médiatiséeautour du jeu de go, mais on peut se rendrecompte dans différents outils qu'on peut utiliserquotidiennement que des progrès importantsont été obtenus. Alors je pense au traducteur automatique,aux outils de recherche ou de recommandation.On a pris conscience de la grande quantité de donnéesnumériques qui sont aujourd'hui disponibles et des opportunitésque leur traitement offre dans énormément de domaines.

## Comprendre quelques exemples

Le champ d'application des techniquesde machine learning est extrêmement large et touche un grandnombre de secteurs.Considérons le secteur du commerce par exemple.De nombreuses problématiques liées à l'anticipation de comportementsclient peuvent être traitées à l'aide du machine learning.Par exemple, on peut chercher à prévoirla demande liée à un produit.Tout d'abord, on a besoin de données décrivant le produitet le contexte d'achat.On peut utiliser des variables telles que la date de livraison,le jour de la semaine, le créneau horaire,des informations sur le produit, la marque, le nom du produit,la catégorie, le prix, la réduction, des informationssur le magasin, son type, son emplacement,et d'autres données externes, telles que la météo,les vacances, le contexte économique,l'inflation, le chômage, ainsi qu'une informationparticulière qu'on nomme la vérité terrain,et qui correspond à la disponibilité ou au contrairela rupture de stock pour l'article dansun contexte d'achat particulier.On construit alors automatiquement un modèle comme une fonction quia un ensemble de valeurs sur chacune des variables associe la valeurdisponibilité ou rupture de stock.De telle sorte que les valeurs prédites soient à la foisle plus possible concordantes avec la vérité terrain,et aussi justes lorsqu'on lui présente des cas encorenon considérés lors de l'apprentissage du modèle.Une fois ce modèle construit, on peut l'utiliserpour mieux gérer les stocks et la chaîne logistique,ou encore orienter le client vers des produits similaires,mais encore en stock.On peut aussi chercher à prévoir le chiffre d'affairesgénéré par une promotion ou une campagne publicitaire.Les modèles utilisés traditionnellement sontconstruits par des experts qui choisissentun petit ensemble de variables explicativesdu chiffre d'affaires.Avec une approche machine learning,on laisse la machine déterminer les variables à prendre en comptepour chaque catégorie de produits.Pour cela, on considère un grand nombre de variables décrivantle contexte d'une promotion : le type de produit,l'importance de la remise client, la durée de la promotion,les ventes moyennes hors promotions,l'exposition sur le prospectus, la taille de la photo,l'emplacement dans le prospectus, l'expositionet l'emplacement au magasin,gondole, allée centrale, hauteur de rayonnage,la mécanique de promotion, 2+1 gratuit,réductions immédiates, points de fidélité,le type de produit, l'élasticité de la promotion,combien de consommateurs réagissent à la promotion,la pression concurrentielle, les autres promotionsmenées par les concurrents, la saisonnalité.La vérité terrain est ici le chiffre d'affaires généré.L'algorithme d'apprentissage génère des milliers de modèless'appuyant sur des sous-groupes de variables explicativesqui sont ensuite combinées pour obtenir un modèleprédictif unique intégrant potentiellement l'ensembledes variables.On améliore ainsi l'exactitude des prévisions et le modèlequi continue à s'affiner au fur et à mesure quede nouvelles promotions sont réalisées et intégrées dansla base d'apprentissage.Le machine learning est également d'un grand intérêtdans le secteur industriel.Il peut être utilisé pour prévoir les défaillancesd'un système de production, et ainsi permettre d'interveniravant qu'une machine ne tombe en panne.À partir des données de fonctionnement et del'historique des interventions et des pannes,le machine learning permet d'estimerla probabilité d'une défaillancefuture d'un équipement donné.La maintenance est réalisée sur la base d'une estimation de l'étatde santé des pièces d'équipement.On peut alors avancer la détection des défaillances et permettreune pré-intervention grâce à des outils de prédiction.Pour cela, on utilise un ensemble de capteurs qui sont installéssur les machines à surveiller.Selon l'équipement considéré, on peut avoir directement des logsd'événements ou d'incidents, par exemple pour les ordinateurs.À partir de ces logs, on peut effectuerla prédiction de pannes.Pour d'autres types de machines, par exempledes machines rotatives,différents types de mesures sont utilisés pour détecterles signes avant-coureurs de panne.Les mesures acoustiques utilisent des ultrasons pour évaluer l'étatdes équipements et permettent de détecter rapidement la présencede défauts mécaniques, de fuite, ou encorede problèmes électriques.La thermographie utilise des capteurs de température,typiquement des caméras infrarouges,pour obtenir le profil thermique des équipements.Enfin l'analyse vibratoire permet de préciser le diagnostic.Il s'agit d'analyser les vibrations émisespar la machine pour détecter et identifier des défautsde fonctionnement.Les techniques de maintenance prédictive sont conçuespour aider à anticiper les défaillances des équipements,afin de permettre une planification anticipéede la maintenance corrective, évitant ainsi les pannesimprévues des équipements, améliorant la qualitéde service pour les clients, et réduisant les coûtssupplémentaires engendrés par une maintenanceexcessive des stratégies de maintenance préventive.Les gains en termes de disponibilité,de sécurité et de coûts opérationnels sont à la foisimmédiats et significatifs.

# Recueillir et mettre en forme les données

## Définir les données pour le machine learning

La qualité d'un modèle de machine learning dépend des données.C'est l'élément indispensable, et c'est d'ailleurs grâceà la disponibilité croissante de grandes quantités de donnéesnumériques que le machine learning est devenu si populaireces dernières années.Toutefois, il ne suffit pas de mettre toutes les donnéesque l'on a sous la main en entrée d'un algorithme d'apprentissagepour obtenir un bon modèle.Il faut identifier les bonnes caractéristiques,celles qui influencent potentiellementla variable cible que l'on chercheà prédire. Mais au fait, qu'est-ce qu'une donnée? Une donnée, c'est le couple que forme une définition,c'est- à-dire un concept et une mesure sur un objet particulier.La définition est appelée attribut ou variable,et est caractérisée par un type, numérique, symbolique.Créer un jeu de données, c'est alors choisir les objetsque l'on va observer pour être représentés dans les donnéeset les attributs que l'on va recueillir sur ces objets,en faisant abstraction de tout le reste.Le mot "données" est donc trompeur.La définition, la mesure et les objets observés sontproduits de façon sélective par l'observateur humain,le data scientist, et non donnéesde manière objective.Quelles sont les étapes à suivre pour construireun jeu de données ? Il faut d'abord recueillir des données brutes,puis identifier les bons attributs,ainsi que la variable à prédire.Puis on doit s'intéresser aux objets observés.Il peut être nécessaire de choisir une stratégied'échantillonnage appropriée lorsque l'ona trop d'observations, par exemple pour les donnéesen ligne potentiellement infinies.Enfin, il faut fractionner les données enun échantillon d'apprentissage et un échantillon de tests.Les données sont stockées sous de nombreusesformes et formats : des signaux, des images,du texte, des valeurs.Il faut alors les transformer en un format compatibleavec l'algorithme de machine learning utilisé.Les algorithmes de machine learning acceptent généralementles données sous forme d'une liste de n vecteurs de dimension d.Chaque vecteur représente un objetet la IM entrée d'un vecteur correspondà la mesure du IM attribut sur cet objet.Si les données proviennent de différentes sources,il faut vérifier la cohérence de format des données,c'est-à-dire que toutes les valeurs d'un attributsoient écrites dans le même format.En particulier, il faut faire attention aux dates,aux valeurs monétaires, aux adresses.La forme d'entrée doit être identique pour l'ensembledu jeu de données.Alors il est tentant d'inclure autant de données que possibleà cause du Big Data.Mais ce n'est pas une chose à faire.Il faut utiliser le bon sens pour déterminerles attributs pertinents par rapport à la tâcheque l'on cherche à apprendre.Par exemple, si l'on souhaite prédire quels clients sontsusceptibles de faire des achats importants sur un site en ligne,l'âge des clients, leur adresse, leur sexe peuventêtre de bons prédicteurs.En revanche, leur numéro de portable ne va pas êtreutile dans cette tâche.C'est à ce moment-là qu'une expertise surle problème à traiter joue un grand rôle.Il est bien évidemment crucial de s'assurer de la fiabilitédes données et de la mesurer.La fiabilité fait référence au degré auquel vous pouvezfaire confiance à vos données.Est-ce que la variable de classe comporte des erreurs ?Les valeurs des attributs sont-elles bruitées ?Vous pouvez effectuer des vérifications aléatoirespour répondre à ces questions.La qualité dépend aussi des objets observés.Ont-ils été bien choisis ?Sont-ils représentatifs des données que l'on auraà traiter en production ?Ont-ils des valeurs manquantes ?Y a-t-il des exemples en double ?Des observations systématiques du jeu de données sont nécessairespour répondre à ces questions.Il peut être difficile de collecter suffisammentde données dans un projet d'apprentissage automatique,mais il peut aussi arriver que l'on ait trop de données.Dans ce cas, il est nécessaire de sélectionnerun sous-ensemble d'exemples, d'échantillonner les données.Comment sélectionner un tel sous-ensemble ?À quelle granularité faut-il échantillonner les données ?Devons-nous utiliser des requêtes aléatoires,suivre un utilisateur en particulier ?La réponse dépend du problème considéré et des donnéesnécessaires au calcul des valeurs d'attributs.Si l'on a besoin de connaître le comportement d'un utilisateur,il faudra échantillonner au niveau de l'utilisateur.Si l'on a besoin de connaître les événementssurvenus précédemment, il faudra considérerdes fenêtres temporelles.Un autre aspect à bien considérer est la répartitiondes modalités de la variable de classe dans l'échantillon.Un ensemble de données de classificationavec des proportions de classes asymétriquesest appelé déséquilibré.Certaines classes, les classes majoritaires,apparaissent dans une grande proportion des exemples.Les autres sont dites minoritaires.Dans un tel cas, il peut être nécessaire d'appliquer une techniqued'échantillonnage particulière.On peut Tout d'abord apprendre notre modèle sur un échantillonrespectant la vraie distribution.Si le modèle n'a pas de bonnes performances en généralisation,on peut chercher à l'améliorer à l'aide de techniquesde sous- échantillonnage et de surpondération.Le sous-échantillonnage consiste à apprendre le modèle sur unéchantillon dans lequel la classe majoritaire est sous représentéepar rapport à sa proportion dans l'ensemble des données.Et la technique de surpondération consiste à ajouter un poidsaux exemples de la classe majoritaire pour compenserle sous-échantillonnage dans l'évaluation des performancesde la méthode.Regardons sur un exemple comment cela fonctionne.Lors de l'étape 1, on sous-échantillonnela classe majoritaire.Si l'on cherche à apprendre un modèle sur des donnéespour lesquelles on a un exemple positif pourexemples négatifs, nous pouvonssous-échantillonner d'un facteur 20 par exemple pour prendre10 exemples négatifs.On aura ainsi un exemple positif pour 10 exemples négatifs.Cela va faciliter l'apprentissagede notre modèle. Lors de l'étape 2, nous donnerons un poidsplus important aux exemples de la classe majoritaire,c'est-à-dire aux exemples de l'ensemble qui aété sous-échantillonné.Puisque nous avons sous-échantillonnéla classe majoritaire par un facteur de 20,le poids d'un exemple de cette classe doit être de 20.Il peut sembler étrange d'ajouter des poids aux exemples dela classe que nous avons sous-échantillonnée,mais nous essayons de faire en sorte que notre modèleaméliore la classe minoritaire.Alors pourquoi augmenter le poids de la classe majoritaire ?Grâce au sous-échantillonnage, nous voyons plus souventla classe minoritaire, ce qui améliore la qualitéde l'apprentissage.Et grâce à la surpondération, nous pouvons calculer les mesuresde qualité du modèle de façon à toujours pouvoir les interprétercorrectement en termes de probabilités observablesdans les données réelles.Une fois la base d'exemples construite,il faut la partager en un échantillon d'apprentissageset un échantillon de tests.Le partitionnement aléatoire est approprié pour de nombreuxproblèmes de machine learning, mais ce n'est pastoujours le cas.Notamment lorsque l'on considère des données séquentielles,par exemple des données générées par une applicationdisponible sur Internet.Dans ce cas, on peut par exemple recueillirles données pendant 30 jours,utiliser les données des 29 premiers jourspour apprendre le modèle, et celles du dernier jour pour le tester.Pour les systèmes en ligne, les données d'apprentissagesont plus anciennes que les données utilisées en production.Cette technique conserve ce décalage entre l'apprentissageet le test. Toutefois, les partitions basées sur le tempsfonctionnent mieux avec des jeux de données très volumineux,tels que ceux contenant des dizaines de millionsd'exemples. Dans les projets avec moins de données,les distributions des deux échantillons finissentpar être très différentes.

## Prétraiter et transformer les données

La qualité d'un modèle d'apprentissage peut généralementêtre améliorée par prétraitement et transformation des données.Une étape essentielle est le nettoyage des données.Cela consiste à traquer les éventuelles erreurs quiont pu se glisser dans les données : des données qui sonten dehors de l'intervalle de valeurs possibleset les données manquantes.La gestion des données manquantes est délicate.Ignorer les observations comportant des valeursmanquantes peut d'une part réduire drastiquementla taille du jeu de données, et compromettre la constructiond'un bon modèle, et d'autre part introduireun biais de sélection, si les valeurs manquantes ne sontpas réparties de manière uniforme, mais sontcaractéristiques d'une sous-population des observations.Il faut donc adopter une autre approche.Dans une première étape, vous pouvez visualiserla répartition des valeurs manquantes par attributet par observation.Vous pouvez alors décider de supprimer les observationscomportant trop de valeurs manquantes ou les attributstrop peu renseignés.Dans une deuxième étape, vous pouvez remplacer les valeursmanquantes restantes par la moyenne ou la médiane des valeurspour un attribut numérique, ou par la valeur la plus fréquentepour un attribut catégoriel.Cette approche standard souvent très efficace permet de donnerune valeur à l'attribut et de pouvoir ainsi utiliserun algorithme d'apprentissage, le modèle, sur les données.En revanche, cela diminue artificiellement la dispersiondes valeurs et peut introduire des corrélations de manièreartificielle. Ensuite, on peut chercher à dériver de nouvellescaractéristiques, de nouveaux attributs à partirdes attributs de nos données.En effet, afin d'accroître les performances du modèle appris,il peut être nécessaire de décomposer certains attributsou d'en modifier d'autres.De nombreux algorithmes d'apprentissageautomatique ne peuvent pas être utilisés directementsur les attributs catégoriels.Ils exigent que toutes les variables soient numériques.Cela signifie que les attributs catégoriels doivent être convertisen variables numériques.Ceci peut être réalisé en associant à chaquecatégorie une valeur numérique.La couleur verte prend la valeur 1.La couleur bleue, la valeur 2, et la couleur rouge,la valeur 3.Cependant, certains algorithmes de machinelearning qui reposent sur des calculsde distance peuvent être biaisés parce codage. Leur couleur verte sera plus proche de la couleur bleue,que de la couleur rouge, sans que cela n'aitun sens dans les données.Ainsi, on préfère transformer les attributs catégorielsen autant d'attributs binaires qu'ils ont demodalités distinctes.On aura ainsi autant de variables factices qu'il y a de couleurs.La première variable factice prendra la valeur1 pour les observations dont la couleur est verte,et 0 sinon.La deuxième fera de même pour la couleur bleue.Et la troisième sera utilisée pour encoder la couleur rouge.Ainsi deux observations seront à une distance nullesi elles ont la même modalité, la même couleurdans notre exemple, ou auront une distancequi sera la même, quelles que soient les modalitésdistinctes qu'elles prennent.Certains attributs peuvent être complexes,et leur décomposition en plusieurs parties aidera à capturerdes relations plus spécifiques.Par exemple les attributs de type date comportentbeaucoup d'informations, et doivent être décomposés.Par exemple, si vous pensez qu'il existeune relation entre le jour de la semaineet d'autres attributs, alors il faudra dériver un nouvelattribut catégoriel indiquant le jour de la semaine.D'autres informations, telles que le momentde la journée, le mois, ou la saison,peuvent fournir à l'algorithme des informations plus pertinentes.Considérons maintenant les attributs numériques.Le fait qu'un attribut numérique puisse prendredes valeurs très différentes, et pour certainesobservations que cette valeur puisse êtretrès grande, peut engendrer des erreursde calcul numérique lors de l'apprentissage du modèle.De plus, on peut avoir plusieurs attributs avec des plagesde valeurs très différentes, par exemple l'âge et le revenu.Et lors du calcul du modèle, les valeurs élevées d'un attributpeuvent alors prendre le dessus et en quelquesorte masquer l'attribut de plus faible amplitude.Par exemple, si un attribut A prend ses valeursdans l'intervalle [0, 5], et qu'un attributB prend ses valeurs dans l'intervalle [100,200], le calcul de la distance euclidiennebasé sur ces deux attributs sera dominépar l'attribut B.Il est donc nécessaire d'anticiper ce problème.C'est ce que permet le redimensionnementdes données. La méthode la plus simple appelée miseà l'échelle min-max consiste à redimensionnerl'intervalle des valeurs de l'attribut pour qu'ilsoit égal à l'intervalle [0;1].Une mise à l'échelle min-max est généralement effectuéevia l'équation suivante, où x est la valeur d'origine et x'est la valeur mise à l'échelle.Par exemple, supposons que nous ayons les données de poidscomprises entre 50 et 120 kilos.Pour redimensionner ces données, nous soustrayons d'aborddu poids de chaque observation et divisons le résultat par 70,la différence entre les poids maximum et minimum.La normalisation d'une variable consiste quant à elleà déterminer la moyenne et l'écart type de la variable.Ensuite, nous soustrayons la moyenne de chaque valeur,et nous divisons le résultat par l'écart-type.Cela a pour effet que les valeurs de cette variable dansles données ont une moyenne nulle, lors de la soustractionde la moyenne au numérateur, et une variance unitaire.Alors ce procédé est largement utilisé pour la normalisationdans de nombreux algorithmes d'apprentissage automatique,par exemple, les machines à vecteurs de support,la régression logistique, les réseaux de neuronesartificiels. Les deux approches sont utilisées dans différentes applications.La mise à l'échelle min-max induit des écarts-types plus petits,ce qui peut supprimer l'effet des valeurs aberrantes.La mise à l'échelle min-max est un bon choix lorsque 1,on connaît les limites supérieures et inférieuresapproximatives de nos données, avec peu ou pasde valeurs aberrantes, et 2, lorsque les données sontapproximativement uniformément réparties sur cette plage.Un bon exemple est l'âge.La plupart des valeurs d'âges se situent entre 0 et 90,et chaque partie de la plage compte un nombreimportant de personnes.En revanche, on utilisera la normalisation surla variable revenu, car seules quelques personnesont des revenus très élevés.Ainsi la limite supérieure de l'intervalle est très élevée,et la plupart des valeurs se retrouveraient dans une petitepartie de l'intervalle [0,1].Il peut être pertinent d'essayer les deux approches et d'évaluerleur influence sur un modèle particulier.Enfin, parfois, il peut être avantageux de convertirdes valeurs numériques en valeurs catégorielles.Cela peut être réalisé par exemple en divisanttoute la plage de valeurs en un certain nombrede groupes. Ces groupes peuvent être de même amplitude numérique,une durée de 5 ans par exemple, ou bien contenir le mêmenombre d'observations, des quantiles.Il est nécessaire de convertir une variable numériqueen variable catégorielle lorsque celle-ci n'est pasen dépendance linéaire avec la variable à prédire,et que par exemple, plusieurs intervalles disjointssont liés à une modalité de la variable de classe.Pour conclure, les transformations à appliquer aux données sontà la fois dépendantes des données brutes etde l'algorithme d'apprentissage utilisé.Il peut être nécessaire de procéder de manièreitérative en essayant plusieurs transformationset en étudiant leur impact sur les performances du modèle.

## Appréhender les grandes dimensions

Une étape cruciale dans la préparation des donnéesconsiste à choisir le nombre de variables à utiliserpour décrire les observations.Cette décision dépend en réalité de nombreux facteurs :est-ce que la variable apporte de l'information,les variables sont-elles corrélées,de combien d'observations étiquetées dispose-t-on ?En machine learning on travaille souventavec des vecteurs de grande dimensionpour créer un modèle.Ce qu'on appelle dimension c'estle nombre de variables décrivant nos observations.Lorsque la dimension est supérieure à 3,il est impossible de visualiser cet espace et il estalors difficile de déceler les éventuels problèmes.Pour donner une intuition des problèmes qui se passenten grande dimension, on peut considérer un carré et uncercle inscrit dans ce carré.Le cercle occupe environ 78 % du carré.En dimension 3, si l'on considère une boule inscrite dans un cube,la boule occupe 52 % du cube.Lorsque l'on augmente encore les dimensions,l'hyper sphère occupe une partie de plus en pluspetite de l'hyper cube.En dimension 6 la boule occupe moins de 10 % du cube et moinsde 1% en dimension 9.En traitement de données cela implique que,si on augmente la dimensionnalité de nos données,nous aurons besoin d'augmenter de manière exponentiellele nombre d'observations pour compenser l'immensitéde l'espace c'est-à-dire la rareté des données sinonnous risquons d'avoir un modèle ayant peu de capacitéde généralisation.Les modèles simples appris sur de grandsensembles d'observations sont généralementplus performants que des modèles sophistiquésappris sur de petits ensembles de données.Dans de nombreux cas réels, le fait d'avoir un grandnombre de variables conduit à une diminutionde la précision et du rappel du modèle.Il peut être alors intéressant de diminuer la dimensiondes données pour obtenir une meilleure généralisationdu modèle. La réduction de la dimensionnalité consisteà résumer de grands vecteurs de donnéesdécrivant nos observations en une représentationde moindres dimensions, c'est-à-dire par un vecteurde plus petite dimension qui résume l'informationcontenue dans une donnée brute.L'analyse en composantes principales est une techniquelargement utilisée pour réduire la dimension des données.Cette technique consiste à rechercher un espacede plus faible dimension.Chaque nouvelle dimension appelée composante est définie parune combinaison linéaire des variables d'origine et est indépendantedes autres composantes.Enfin ces composantes sont triées par ordred'importance c'est-à-dire que la premièrecomposante définit la direction selonlaquelle les observations varient le plus.C'est la direction qui répartit le mieuxles données dans l'espace.La deuxième composante est à la fois orthogonaleà la première et celle selon laquelleles observations varient le plus et ainsi de suite.Schématiquement l'ACP consiste à calculer une matrice qui résumecomment toutes les variables sont liées les unes aux autrespuis elle la décompose en deux composantes distinctes,l'une indiquant la direction des données et l'autresa magnitude c'est-à-dire l'importancede cette nouvelle dimension pour décrire les données.Sur le schéma on voit les deux directions principales des donnéesdécrites par les axes X et Y, la direction rougeet la direction verte.Dans ce cas la direction rouge est la plus importante.On peut alors représenter ces mêmes données en prenantpour axe les deux directions principales qui sontdes combinaisons de nos variables d'origine X et Y.Le schéma contient les données transformées avec l'axedes X correspondant à la direction rouge et l'axedes Y à la direction verte.En identifiant les directions les plus importantes,nous pouvons réduire la dimensionnalité des données.De plus, comme les nouvelles variables sont des combinaisonslinéaires des variables d'origine, nous conservons les informationsimportantes qu'elles contiennent.Plus techniquement l'ACP consiste à rendre les variables d'originecomparables en les centrant, cela consiste à ce que chaquevariable ait une moyenne égale à 0 et souvent en les réduisant,cela consiste à ce que chaque variable aitun écart type égal à 1.On considère ensuite la matrice de covariance obtenue à partirde ces variables transformées qui permet de quantifierla variation de chaque variable par rapport à chacune des autres.Ensuite on utilise une technique d'algèbre linéaire pour décomposerla matrice de covariance en un produit de vecteurs orthogonaux.C'est la décomposition en vecteurs propres.On retient ensuite un petit nombre de composantesqui explique la majorité de la variance contenuedans les données d'origine.L'idée qu'il faut retenir c'est quel'on cherche un espace dans lequel les donnéessont le plus dispersé possible pour queles variables ou composantes soientle plus informatif possible.

# Découvrir quatre modèles incontournables du machine learning

## S'initier à la méthode Naïve Bayes

La méthode d'apprentissage automatique appelée NaïveBayes repose sur la notion de probabilités et plusprécisément sur la notion de probabilités conditionnelles.Quelle est la probabilité qu'une proposition soitvraie sachant les informations dont on dispose ?Plus le nombre de chances est faible,plus la probabilité tend vers zéroet plus on a de certitudesplus la probabilité tend vers la valeur1. Considérons un exemple.Nous avons quatre variables décrivantles conditions climatiques : le temps, la température,l'humidité, la force du vent et l'on cherche à prédiresi le court de tennis va être occupé ou non.Quelle est la probabilité que l'on joue au tennis ?Sans information additionnelle, la probabilité a priori que l'onjoue au tennis est égale à 9/sur quatorze correspondant au fait que l'on joue au tennis.Maintenant quelle est la probabilité que la propositionjouer au tennis soit vraie sachant que le vent est faible ?Cette probabilité se calcule comme étant égaleà la probabilité que l'on joue au tennis et quele vent soit faible divisée par la probabilitéque le vent soit faible.Dans notre exemple P de tennis sachant vent faible est égalà la probabilité de tennis et vent faible soit 5/divisée par la probabilité de vent faible soit 8/14.Ainsi si l'on sait que le vent est faible on aune probabilité de 5 surpeut estimer prédire la variable de classejouer/pas jouer connaissant la valeur d'une variable.Qu'en est-il lorsque l'information disponible est plus riche,lorsque l'on a plusieurs variables ?Plus on ajoute de variables dans la condition et plusle nombre d'observations satisfaisant ces valeursdiminue. Nos estimations empiriques de probabilités deviennentalors de moins en moins justes.La confiance que l'on a dans une probabilité estiméesur deux observations est beaucoup plus faibleque la confiance que l'on a dans une probabilité estiméesur des centaines d'observations.Cette approche n'est donc pas directementapplicable lorsque l'on a plusieurs variablesà prédire. La méthode Naïve Bayes permet de surmonter ce problèmeen combinant deux éléments : le théorème de Thomas Bayesqui donne une méthode permettant de calculer la probabilitéconditionnelle P(A׀B) à partir de la probabilité duale P(B׀A)et une hypothèse un peu naïve, c'est-à-dire une hypothèsesimplificatrice que l'on sait fausse mais qui en pratiquepermet de calculer P(B׀A) tout en ayant de bons résultatsde classification.Le théorème de Bayes se déduit de la définitionde la probabilité conditionnelle.En effet on a la P(A׀B) = P(A et B) /P(B).De façon similaire, en inversant les rôlesde A et de B on a la P(B׀A) = P (A et B) /(A).Ainsi on a la P(A et B) = P(A׀B) P(B)ou alors c'est également égal à P(B׀A) P(A).On peut donc dériver l'équation P(A׀B) = P(B׀A) P(A)et /P(B) et ça ça constitue le théorème de Bayes.L'hypothèse Naïve suppose que les variablessont indépendantes les unes des autres.Par exemple les occurrences de la modalité vent faiblesont indépendantes des occurrences des modalitésdes autres variables, par exemple temps ensoleillé.C'est clairement une hypothèse simplificatrice qui n'estgénéralement pas satisfaite dans les données.Considérons comment évaluer le numérateur,la P(A, B, C, D׀Y) P(Y).En appliquant la définition du calcul d'une probabilitéconditionnelle, on a la P(A, B, C,D׀Y) P(Y) = P(A, B, C, D, Y).On peut alors appliquer à nouveau la définitionmais cette fois dans l'autre senspour sortir la variable A.On a P(A, B, C, D, Y) = P(A׀B, C, D,Y) P(B, C, D, Y).On répète le procédé pour sortir B et on a P(B,C, D, Y) = P(B׀C, D, Y) P(C, D, Y)et de même on sort C.On obtient alors P(C, D, Y) = P(C׀D, Y) P(D,Y) et enfin on sort D et on obtient P(D,Y) = P(D׀Y) P(Y).Maintenant en utilisant l'hypothèse d'indépendancedes variables A, B, C et D, on simplifiedifférentes probabilités.On a P(A׀B, C, D, Y) qui est simplifiée en P(A׀Y);P(B׀C, D, Y) qui est simplifiée en P(B׀Y) et P(C׀D,Y) qui est simplifiée en P(C׀Y).On a alors P(Y׀A, B, C, D)= P(A׀Y) P(B׀Y) P(C׀Y) P(D׀Y) P(Y) et le tout divisé par P(A,B, C, D).Pour classer une nouvelle observation en fonction deces valeurs sur les attributs A, B, C et D, nous devons calculercette valeur pour les différentes classes à prédireY et affecter l'observation à la classe correspondantà plus forte probabilité.Ainsi on n'a pas besoin de calculer P(A,B, C, D) puisque cette valeur est la même quelle que soitla modalité de la variable de classe.On classifie donc une nouvelle observationen prenant la modalité de la classe Y quimaximise P(A׀Y) P(B׀Y) P (C׀Y) P(D׀Y) P(Y).Les paramètres du modèle à apprendre sont donc lesprobabilités P(Y) pour toutes les classes Y à prédireet P(X׀Y) pour toutes les modalités X des variablesprédictives et les modalités Y de la variable de classe.Ces paramètres sont estimés par leur fréquence dansles données d'apprentissage.Par exemple pour estimer la valeur de la variable de classe lorsquele temps est ensoleillé, que la température est froide,l'humidité est forte et le vent est fort,nous devons compter les observations dansle jeu de données où ces modalités sont associéesà la classe Oui c'est-à-dire jouer tennis et cellesassociées à la classe Non où on ne joue pas au tennis.Multipliées entre elles et avec la probabilité a prioride la variable de classe, on obtient 0,02 pourla classe Non et 0,On prédit alors que l'on ne jouera pas au tennisdans ces conditions.L'algorithme Naïve Bayse est très efficace pourles problèmes d'apprentissage en grande dimension commepar exemple la classification de texte où chaque mot estreprésenté par une variable binaire prenant la valeur1 si le mot apparaît dans l'observation textuellecorrespondante ou 0 sinon.On peut alors réaliser des tâches de catégorisation de texteen fonction de leur sujet : filtrage de spams ou encorede reconnaissance d'opinions.

## Aborder la méthode des k plus proches voisins

La méthode des K plus proches voisins prédit la classed'une nouvelle observation à partir des K observationsqui lui sont les plus proches.La classe prédite est celle qui est majoritaire parmi les Kobservations dans son voisinage.K représente le nombre d'observations d'apprentissagequi se trouvent à proximité de la nouvelle observationet que nous utilisons pour déterminer la classe.Cette méthode est très simple à comprendre età mettre en oeuvre et a été largement appliquée dans de nombreuxdomaines tels que les systèmes de recommandation,la recherche d'information et la détection d'anomalie.Comme pour tout problème d'apprentissage,la première étape consiste à trouver une bonne façonde représenter des observations par un ensemble de descripteurs.Dans le cas de la méthode des K plus proches voisins,ces descripteurs doivent être peu nombreux et de type numérique.Si les caractéristiques observées pour décrirenos objets ne le sont pas,un prétraitement est alors nécessaire.La proximité entre deux observations est évaluéepar un calcul de distance entre les valeurs prisessur chaque descripteur par les deux observations.Par exemple pour la distance euclidienne on prendrala différence des valeurs au carré et on fera la sommepour tous les descripteurs, la racine carrée pouvant êtreignorée puisque seul l'ordre sur les distances est important.C'est donc pour cette raison que l'on a besoin de descripteursde type numérique.Une autre distance beaucoup utilisée est la distance cosinus.Plutôt que de calculer une magnitude,la similarité cosinus utilise la différencede direction entre les deux vecteursreprésentant les valeurs d'attributs.Son utilisation est appropriée lorsqueles attributs sont binaires, par exemple pourles variables muettes encodant des attributs catégoriels.Les différentes mesures de distance deviennentmoins discriminantes lorsque le nombrede descripteurs augmente, c'est-à-dire que plus le nombrede descripteurs est grand et plus les valeurs des distancesvont avoir tendance à être les mêmes pour les différentscouples d'observation, ce qui va dégrader fortementles performances de classification de la méthode.L'utilisation de la méthode des K plus proches voisins requiertdonc d'avoir bien identifié un petit nombre de descripteurspour décrire les observations.Contrairement aux autres méthodes de classification,cette méthode n'apprend pas de modèle.Le processus de généralisation est réalisé durant la phasede classification.On dit que cette méthode est non paramétrique dans le sensoù elle ne consiste pas à estimer les paramètres d'un modèle.En revanche elle réalise la tâche de classificationen se basant sur l'ensemble des données d'apprentissage.Pour prédire la classe d'une nouvelle observation,la méthode est réalisée en trois étapes.Tout d'abord les distances entre la nouvelle observation et chacunedes observations de l'ensemble d'apprentissage sont calculéespuis les K plus petites distances permettent d'identifier les Kplus proches observations.Enfin la classe majoritaire parmi les K observations est affectéeà la nouvelle observation.K n'est pas un paramètre mais un hyper paramètre de la méthode,c'est-à-dire que la valeur de k ne va pas être appriseautomatiquement par l'algorithme à partir des donnéesd'apprentissage. Les hyper paramètres caractérisent l'algorithme d'apprentissage;ils influencent sa complexité, sa rapidité d'exécution et mêmela qualité de l'apprentissage mais ce ne sont pas les donnéesd'apprentissage qui vont permettre de déterminer leur valeur.Celles-ci sont fixées par l'utilisateuravant l'exécution de la méthode.Les techniques dédiées à la détermination dela valeur des hyper paramètres peuvent être utiliséespour fixer la valeur de K.Cette méthode est très particulière dansle champ des méthodes de machine learning car étantune technique non paramétrique elle ne fait pas d'hypothèsesur la distribution des données sous-jacentes.Ainsi les K plus proches voisins est un bon choixpour débuter une tâche de classification lorsqueles connaissances préalables sur les données sont limitées.Un des avantages de la méthode est qu'elle estcapable d'apprendre les fonctions très complexes.L'intégralité du calcul a lieu durant la classificationde nouvelles observations.Il n'y a donc pas besoin de calculer un modèle oude le mettre à jour lorsque la base d'apprentissage évolue.En revanche la méthode est d'autant plus lente quela base d'apprentissage est grande.De plus les performances de la méthode se dégradentpar la présence d'attributs non pertinents.Ainsi cette méthode peut être utiliséelorsque l'on a beaucoup de données d'apprentissageet peu d'attributs, moins de 20.

## Définir les arbres de décision

Le principe sous-jacent aux arbres de décisionest d'identifier des règles du type"si condition alors conclusion" où la condition porte sur desvaleurs des variables et la conclusion indiqueune modalité de la variable de classe.Partant de toutes les observations de l'ensemble d'apprentissage,l'arbre de décision va successivement partitionnercet ensemble en fonction de critères explicites.Prenons l'exemple jouer au tennis.Pour construire l'arbre, on part des quatorze observationsde l'ensemble d'apprentissage.On considère l'attribut temps.Si le temps est couvert, on affecte la classe Jouer = Ouiaux quatre observations qui vérifient la condition.Si le temps est ensoleillé, on a cinq observationsqui vérifient cette condition.On va considérer une deuxième conditionsur ces cinq observations.Est-ce que l'humidité est forte ou faible ?Si elle est forte on affecte les trois observationssatisfaisant ces deux conditions, temps ensoleilléet forte humidité, à la classe Jouer = Non.Les deux observations satisfaisant tempsensoleillé et humidité normale sont affectéesà la classe Jouer = Oui.Quels sont les ingrédients nécessaires pour construireun tel arbre ? Il nous faut 1) une technique pour choisir l'attributque l'on va utiliser pour séparer les observations,2) définir un critère d'arrêt, c'est-à-dire définirquand est-ce que l'on s'arrête de séparer les observationset que l'on affecte une classe.Idée 3, l'algorithme de construction d'arbrede décision le plus répandu utilise la mesure d'entropieet le gain d'information pour réaliser ces étapes.Partant d'un ensemble d'observations,l'arbre de décision va chercher à séparercet ensemble de manière à ce queles observations d'une même classe se retrouventensemble dans une même branche de l'arbre.On utilise la mesure d'entropie pour mesurer ce phénomène.La mesure d'entropie évalue le caractère aléatoire,l'imprévisibilité de la modalité de la variablede classe à attribuer aux observations de l'ensemble.Si toutes les observations d'un ensemble appartiennentà une même classe, alors l'entropie vaudra 0.Au contraire si on a le même nombre d'observationspour chacune des modalités de la variable de classe,alors l'entropie sera maximum et vaudra 1.Sur notre exemple, dans l'ensemble des observationson en a neuf associées à la classe Jouer = Oui et cinqà la classe Jouer = Non.L'entropie mesurée sur ces observations est égale à -9/\* log de 9/14 - 5/14 \* log de 5/14 qui est égal à 0.94.Elle est donc très élevée, proche de 1.Voyons ce qui arrive à l'entropie lorsque nous prenonsnotre première décision sur la base de l'attribut temps.L'attribut temps va générer troisbranches : une associée à ensoleillé,l'autre à couvert et la troisième à pluvieux.Pour la branche ensoleillée cinq observations correspondentà un temps ensoleillé.Parmi ces cinq observations, deux correspondent à Jouer= Oui et trois à Non.L'entropie est égale à 0.97.Pour la modalité couvert, on a quatre observationsavec la valeur Jouer = Oui et zéro avec Jouer = Non.L'entropie est alors égale à zéro.Enfin la modalité pluvieux a trois observations pour Jouer= Oui et deux avec Jouer = Non et l'entropie vaut 0.97.Si on prend l'entropie pondérée sur les trois branches,c'est-à-dire la probabilité d'avoir une observationensoleillée \* l'entropie liée à cette même observation+ la probabilité que le temps soit couvert \* l'entropieliée à la modalité couvert et enfin la probabilitéde pluvieux \* l'entropie de pluvieux,on obtient la valeur 0.69.Le gain d'information est alors la différenceentre l'entropie avant la séparation à l'aidede l'attribut temps et l'entropie après la séparation.On a alors 0.94 - 0.69,un gain d'information égal à 0.246.On calcule le gain obtenu par séparation des observationsà l'aide des différents attributs et on obtient un gain de 0.pour la température et de 0.pour le vent et enfin de 0.On choisit alors l'attribut temps car c'est celui qui induit le gaind'information le plus important.Les arbres de décision peuvent également utiliserdes attributs numériques.Pour cela un point de coupure doit être déterminé surles valeurs prises par l'attribut.La procédure consiste d'abord à trier les valeurs prisespar l'attribut sur les observations considéréespuis à identifier les points de coupure potentiels,c'est-à-dire les points où la variable de classe change,et enfin on choisit le point de coupureparmi ceux identifiés précédemment qui minimisentla mesure de pureté.On peut se demander quand on doit arrêter de découperun ensemble d'observations à l'aide d'attributs.Lorsque l'ensemble est pur, c'est-à-dire qu'il ne contientqu'une seule modalité de classe ?Cela peut conduire à un arbre très grand qui est sur ajustéaux données d'apprentissage et aura ainsi de mauvaisrésultats sur les données tests.Un critère consiste à imposer un nombre minimal d'observationspour pouvoir découper un nœud.Par exemple nous pouvons utiliser un minimum de dix observationset ignorer tous les ensembles de moins de dix observations.Une autre méthode consiste à définir la profondeurmaximale de notre modèle, c'est-à-dire le nombremaximal d'attributs utilisés pour séparer notre ensembled'apprentissage jusqu'à arriver à l'attribution d'une modalitéde la variable de classe.La profondeur maximale se définit égalementpar la longueur du plus long chemin de la racineà une feuille de l'arbre.Pour résumer, l'algorithme fonctionne ainsi :on calcule l'entropie sur l'ensemble des observationsd'un nœud, le nœud originel étantcomposé de tout l'ensemble d'apprentissage.Ensuite on considère chaque attribut indépendamment.On calcule l'entropie liée à chacune des modalitéspuis l'entropie moyenne et enfin le gain d'information.On choisit alors l'attribut de gain maximal.Pour chacune des modalités de cet attribut,on étend l'arbre en créant un nouveau nœud de fils eten répartissant les observations en fonction des modalitésprises sur cet attribut.On répète ce processus sur ce nœud en rappelantla fonction sur cet ensemble d'observations jusqu'àce que le critère d'arrêt soit satisfait.Les avantages des arbres de décision sontqu'ils explicitent les règles de classification.Ainsi ils sont faciles à comprendre et à interpréter.Ils effectuent implicitement un filtrage des attributs,une sélection des variables caractéristiques.Ils peuvent gérer à la fois des données numériques et catégorielles.Les arbres de décision nécessitent relativement peu d'effortsde préparation des données.Les inconvénients de cette méthode sont que :1) les arbres produits peuvent être très complexeset mal se généraliser, c'est ce qu'on appelle lesur apprentissage ou l'over-fitting.Les arbres de décision peuvent être instables car de petitesvariations dans les données peuvent générer un arbrecomplètement différent, c'est ce qu'on appelle la variance.Le calcul de l'arbre de décision repose sur un algorithme gloutonqui ne garantit pas le calcul du meilleur arbre de décisionpossible pour un ensemble d'apprentissage donné.Enfin l'apprentissage d'un arbre de décision peut êtrebiaisé si une classe est beaucoup plus fréquenteque les autres. Il est recommandé d'équilibrer le nombre d'observationspour chacune des classes avant de construireun arbre de décision.

## Utiliser les forêts aléatoires

La méthode des forêts aléatoires appartient au groupedes classifieurs ensemblistes, c'est-à-dire des méthodesqui combinent plusieurs modèles identiquesou différents pour classer de nouvelles observations.En l'occurrence cette méthodecombine plusieurs arbres de décision.C'est un classifieur qui donne de très bons résultats mêmesans réglage fin des hyper paramètres, ce qui le rend très facile à utiliser.Le classifieur par forêt aléatoire crée un ensembled'arbres de décision, une forêt, tout en introduisantde l'aléatoire dans le processus de génération des arbres.L'objectif est d'obtenir une prédiction plus préciseet plus stable. L'aléatoire est introduit à deux endroits dans l'algorithme :dans la construction d'un échantillon d'observation qui sera utilisé commeensemble d'apprentissage et dans la sélectiond'un sous-ensemble d'attributs pouvantêtre utilisés pour partager un nœud.Ainsi, avant de construire un nouvel arbre,un échantillon d'observation est généré de manière aléatoireavec remise c'est-à-dire que, si notre ensemble d'apprentissagecontient n observations, on va effectuer ntirages indépendants, chaque tirage consistantà prendre une observation de manière aléatoire uniformesur l'ensemble d'apprentissage.Ainsi une observation de l'ensemble d'apprentissagepourra être absente de l'échantillon et uneautre pourra être répétée plusieurs fois.Pourquoi construit-on un échantillon avec remise ?Au cours de la construction d'un arbrede décision, le nombre d'observationsconsidérées en chaque nœud diminue fortement.On risque d'avoir rapidement un jeu de données déséquilibréavec peu d'observations par rapport au nombre d'attributs.Dans une telle configuration on a peu de chances d'avoirun nombre suffisamment grand d'observationspour toutes les combinaisons de modalités de variablesfréquemment présentes dans les populations cibles.Les calculs conduisant au choix de l'attributde coupure sont alors biaisés, c'est-à-dire faux,et on va avoir un arbre qui est sur ajustéaux données d'apprentissage.Les forêts aléatoires sont conçues pour réduirela possibilité de créer un modèle spécifiqueà l'ensemble d'apprentissage.L'échantillon avec remise permet de créer de nouvellesdistributions d'observations possibles pour la populationcible à partir de l'ensemble d'apprentissage.En supposant que notre ensemble d'apprentissagesoit de bonne qualité, c'est-à-dire qu'ilsoit représentatif de la population cible,en échantillonnant plusieurs fois avec remisel'échantillonnage avec remise va permettre de construireplusieurs sous-échantillons de même taille ayantdes distributions de variables légèrement différentes.Sans remise on aurait toujours exactement le même échantillonou bien on devrait travailler avec des échantillons de pluspetite taille ce qui comme nous l'avons vu ne permet pasde construire un bon modèle.L'aléatoire est également introduit durant la constructionde l'arbre par le choix des attributs à considérerpour couper un nœud.Au lieu de prendre le meilleur attribut selonle gain d'information sur l'ensemble des attributs,un sous-ensemble d'attributs est d'abord sélectionnéaléatoirement et le meilleur d'entre euxest choisi pour couper le nœud.Cette étape est nécessaire pour éviter que les arbresgénérés soient trop similaires les uns des autres.En effet, si un ou plusieurs attributs ont un pouvoirprédictif important, ils seront sélectionnésdans de nombreux arbres rendant les arbres corrélés.En règle générale, lorsque le nombred'attributs est P, on considérera un sous-ensemblealéatoire de racines P attributs.Un nouvel échantillon d'attributs sera tiré pour chaque nœud de l'arbre.Pour classer une nouvelleobservation, celle-ci est classée par chacundes arbres individuellement.La classe de la nouvelle observation est alors obtenuepar un vote à la majorité sur les classes retournéespar chacun des arbres.On peut également pondérer le vote par l'inverse du taux d'erreurmesuré sur chacun des arbres.Les arbres avec un taux d'erreur élevé ontune valeur de poids plus faible, cela augmente l'impactdans la décision des arbres à faible taux d'erreur.Les forêts aléatoires donnent de très bons résultats carun arbre de décision unique peut être sujet au bruit mais unensemble de plusieurs arbres de décision réduit l'effetdu bruit et donne des résultats plus précis et plus stables.Les forêts aléatoires sont rapides à construire.La construction des arbres peut facilement être parallélisée.Les données n'ont pas besoin d'être pré-traitéesou transformées et la méthode est résistanteaux données aberrantes et aux valeurs manquantes.Par contre les forêts aléatoires sont moins performantes lorsqueles classes sont très déséquilibrées ou lorsqueles modalités d'attributs peu fréquentes sont très corréléesà la variable de classe.De plus les forêts aléatoires sont beaucoup plus difficilesà interpréter que les arbres de décision.

## Étudier les raisons de ces quatre modèles

On peut légitimement se demander en quoi les modèles Naïve Bayse,K plus proches voisins, arbres de décision et forêtsaléatoires sont encore utiles aujourd'hui,à l'heure où des méthodes de type réseau de neuronessuscitent un fort engouement.Tout d'abord leur simplicité les rend très attractifs.Cette simplicité a de nombreux avantages.L'algorithme sous-jacent à la méthode est facilementcompréhensible et un utilisateur peut comprendre de manière assezintuitive comment il fonctionne.Ces modèles sont simples, ce qui les rend plus facilement implémentables.Quelques lignes de code permettent de construire un tel modèle,il est alors assez facile de les mettre en œuvre sur des données.Ces modèles sont simples dans le sens où ils comportent peude paramètres à apprendre et que ceux-ci sont assez intuitifs,ils sont alors facilement explicables et interprétables.Pour le modèle bayésien naïf, chaque caractéristique aune probabilité pour chaque classe et on peut ainsi voir lesquellessont les plus fortement associées à certaines classes.Les arbres de décision sont similaires à un processusde décision humain où l'arbre est une sorte d'organigrammereprésentant des règles de type "si alors" qui fractionnentles observations jusqu'à ce qu'elles proviennent quasimenttoutes d'une même classe.Les petits arbres peuvent être affichés graphiquement.Lorsque l'on prédit une classe à l'aidedes K plus proches voisins,on peut voir exactement quel voisin supporte la prédiction.Enfin ces modèles sont également faciles à calculerou à mettre à jour.Le modèle bayésien naïf est calculé en une passesur les données, une complexité en grand tauxde n \* d où n est le nombre d'exemples et d la dimensiondes caractéristiques.La prédiction de la classe d'une nouvelleobservation est rapide puisqu'elle est baséesur la comparaison des valeurs de l'observationaux probabilités associées à ces valeurs surles différentes classes.Pour les arbres de décision, la construction d'un nœudassocié à un attribut catégoriel nécessite de considérer chaqueattribut et de calculer une mesure d'informationsur la répartition des différentes classes des observationsassociées au fils du nœud, ce qui a une complexitéen grand taux de n \*d pour des attributs catégoriels.Le temps nécessaire à la prédiction est proportionnelà la hauteur de l'arbre.Pour les K plus proches voisins, il n'y a pas réellementde construction de modèle, il est cependant nécessairede garder l'ensemble des observations pour pouvoirfaire une prédiction.Le coût d'une telle prédiction est alors en grand taux de n \* d.De plus les méthodes Bayésien naïve et K plus prochesvoisins sont incrémentales, c'est-à-dire quele modèle peut être mis à jour sans tout recalculer.De plus ces méthodes peuvent combiner facilement des attributsnumériques et symboliques et la complexité du modèlepeut être contrôlée.Un autre intérêt de ces modèles provient de leur compatibilitéavec des exigences légales.En effet de nombreux pays imposent de devoir expliquer une décision,notamment administrative obtenue par un traitement automatique.Qu'est-ce qu'une décision explicable ?Il s'agit d'être capable de rendre compte explicitementdu lien existant entre les observations et la décision.Autrement dit il s'agit de mettre en relation les valeursprises par certaines variables et la prise de décision.Cette exigence est plus forte qu'une exigence d'interprétabilitéoù il s'agit d'identifier les variables qui participentle plus à la décision.Les modèles construits par des arbres de décision,des forêts aléatoires, des classifieurs bayésiensnaïfs lorsqu'ils ne sont pas trop complexessont considérés comme explicables, contrairement aux K plus prochesvoisins ou aux réseaux de neurones qui sont considéréscomme des boîtes noires.Il est seulement possible de construire des indicateursd'importance des variables sur ces modèles.Une décision explicable est interprétablemais pas réciproquement.Enfin, pour des chercheurs tels que Misha Bilenko qui dirigela division Machine Intelligence and Research de Yandex,la méthode des K plus proches voisins est toujourslargement utilisée dans les applications réelles.Selon lui l'avantage majeur des K plus prochesvoisins c'est la capacité de cette méthode à incorporerdes changements de modèles très rapidement.Lorsque l'on a de nouvelles observationsqui sont très différentes des observations considéréesjusque là ou des observations avec de nouvelles étiquettes,la plupart des méthodes d'apprentissage ontdes difficultés à intégrer ces changements et nécessitentde réapprendre le modèle.La méthode des K plus proches voisins quant à ellepeut incorporer rapidement ces nouvelles observationset les prendre en compte pour les prédictions suivantes.Cette capacité à prendre en compte rapidement les changements estune caractéristique essentielle qui rend cette méthode très utilepour mettre à jour rapidement un système en production.Par exemple chez Yandex qui produit un moteurde recherche l'équipe de Misha Bilenko utiliseles K plus proches voisins pour par exemple réagirrapidement lorsqu'il y a des plaintes d'utilisateursou pour corriger des erreurs observées lorsde l'utilisation du système.La méthode des K plus proches voisins qu'ils utilisentne se base pas sur les données brutes mais sur des données transformées.Ces transformations sont de plus en plus faites en utilisantdes réseaux de neurones qui apprennent des transformationsnon linéaires très complexes et permettent ensuite d'obtenirdes mesures de similarité optimales entre les observations.

# Comprendre les réseaux de neurones et l’apprentissage profond

## Découvrir les réseaux de neurones

Les réseaux de neurones artificiels sont une techniqued'apprentissage fondée sur une analogie avec la physiologiedu système cérébral.Ces approches, développées dans un premier temps pour modéliserle fonctionnement du cerveau, ont des applicationsà l'apprentissage de règles de classification.Le cerveau est composé d'environ 100 milliards de cellulesnerveuses, les neurones, chacun reliésen moyenne à 10 000 autres.Un neurone est une cellule excitable qui assurela transmission d'un signal bioélectrique appeléinflux nerveux. Il a deux propriétés : l'excitabilité,c'est-à-dire la capacité de répondre à des stimulationset de convertir celles-ci en impulsions nerveuses,et la conductivité, c'est-à-dire la capacitéde transmettre les impulsions.Un neurone reçoit en entrée des signauxprovenant d'autres neurones.Ces signaux sont des neurotransmetteursqui se lient à des récepteurs.Les récepteurs peuvent être classéscomme excitateurs ou inhibiteurs.Si l'excitation nette reçue par un neurone sur une courte périodede temps est suffisamment grande, le neurone génèreune brève impulsion, appelée potentiel d'actionou influx nerveux, qui est propagée le longde l'axone du neurone.Cet influx nerveux active ou inhibe à sontour d'autres neurones.Certains neurones sont spécialisés.Les neurones sensoriels répondent à un type particulier de stimulus,tels que le toucher, le son ou la lumière,et le convertissent en un signal électrique qui estensuite envoyé à la moelle épinière ou au cerveau.Les neurones moteurs reçoivent des signaux du cerveau,de la moelle épinière qui permettent de contrôlerla contraction musculaire.Par analogie aux systèmesbiologiques, les réseaux de neuronesartificiels modélisent la propagation de l'informationentre les unités élémentaires de calcul appelées neurones formels.Les possibilités de calcul de chaque neurone formelsont très faibles, mais leur interconnexionpermet d'effectuer des calculs complexes.Le premier modèle de calcul basé sur une analogie avecles neurones biologiques a été proposé par Warren McCulloch,un neuroscientifique, et Walter Pitts,un logicien, en 1943.Dans ce modèle, un neurone prend en entréeplusieurs variables binaires, prenons comme valeur 0 ou 1,agrège leur valeur et, en fonction de la valeur agrégée,prend une décision elle aussi binaire.L'agrégation des valeurs d'entrée est réalisée parla somme de celle-ci et la décision est modéliséepar un seuil sur cet agrégat.Ce modèle peut représenter des fonctions booléennescomme le "et" logique ou le "ou".Ainsi, en fixant manuellement le paramètre thêta,le seuil, ce modèle est capable de représenterles fonctions booléennes qui sont séparables linéairement,c'est-à-dire les fonctions booléennes pour lesquellestoutes les entrées qui produisent un 1 se situent d'un côtéde la droite définie par la somme de [inaudible] = thêta,et toutes les entrées qui produisent unse situent de l'autre côté de cette droite.Ce modèle, qui n'est pas un modèle d'apprentissagepuisque ses paramètres, la valeur de thêta,ne sont pas appris à partir des données,est à l'origine du premier modèle d'apprentissagebasé sur un réseau de neurones artificiels,le perceptron, proposé en 1958.

## Définir le perceptron

Le modèle du perceptron, proposé par Frank Rosenblatten 1958 et amélioré et étudié par Minsky et Papert en 1969,est un modèle de calcul plus général que le neuronede McCulloch et Pitts.Il surmonte certaines des limites de ce modèleen introduisant des poids sur les neurones d'entrée,ce qui permet de donner plus d'importanceà certaines des entrées par rapport aux autres,et aussi à pouvoir manipuler des entrées non plus à valeurbooléenne mais à valeur réelle.Mais l'avancée la plus importante est sans douteque l'on peut apprendre la valeur de ces pondérationsà partir d'observations d'apprentissage.Tout en étant très similaire au modèle de McCulloch et Pitts,le perceptron se distingue par le fait qu'il prendune somme pondérée des entrées.La fonction de décision est en revanche identique,elle prend la valeur 1 lorsque la fonction d'agrégationa une valeur supérieure à un seuil thêta et -1 sinon.Cependant, au lieu de fixer manuellement ce paramètre thêta,celui-ci va être appris en même temps que les poids.Le perceptron est donc un classifieur binaire linéairequi calcule un hyperplan, une droite, lorsqu'il y adeux neurones d'entrée, tel que les observationsde la classe +1 se situent d'un côtéde l'hyperplan et les autres de l'autre.Considérons le jeu de données Iris.Chaque observation, un iris, est décrite parla longueur et la largeur de ses pétales et de ses sépales.On a trois sortes d'iris : les setosa, les virginicaet les versicolor.On va choisir seulement deux de leurs variables,la longueur des sépales et la largeur des sépales,afin de pouvoir représenter les observations dans un planavec la longueur en abscisse et la largeur en ordonnée.On affecte la classe supérieure à 0 aux iris setosa et inférieureà 0 aux autres.On voit que ces deux classes sont linéairement séparables.Il existe une droite telle que les iris aux setosa sontà gauche et les autres à droite.Cette droite est définie par trois valeurs :le poids p1 associé à la longueur x1,le poids p2 associé à la largeur x2 et thêta,la valeur constante qui est égale à p1 x x1 + p2 x x2,les poids de la droite.Ces trois paramètres peuvent s'écrire d'une manièreunifiée dans ce modèle en rajoutant un neuronedont la valeur est constante et égale à 1,et le poids p0 est égal à - thêta.L'objectif est alors d'apprendre les trois paramètres p0,p1 et p2, de telle sorte à parfaitementclasser les entrées positives et négativesdans nos données.Voici l'algorithme.Les poids pi sont initialisés avec des valeurs aléatoires.Jusqu'à convergence de l'algorithme,c'est- à-dire jusqu'à ce que les observationssoient bien classées ou qu'il n'y ait pasde modification de poids, on prend une observationde manière aléatoire dans les données.Si cette observation appartient à la classe positive,la somme pondérée des xi x pi devraitêtre supérieure ou égale à 0.C'est l'objectif fixé au perceptron.Ainsi, si ce n'est pas le cas, le perceptronva modifier les poids, il va les augmenterde la valeur des xi.Si l'observation appartient à la classe négative,la somme des xi x pi devrait être inférieure à 0 etsi ce n'est pas le cas, le perceptron soustraitla valeur des xi au poids.Ces modifications vont en effet permettre d'augmenterla valeur prédite pour les observations mal classéesde la classe positive et réduire la valeur préditepour les observations mal classées de la classe négative.Les modifications des poids pi se font le longdes paramètres d'entrée actifs, c'est-à dire pour lesquelsxi est différent de 0.Sinon, le poids n'a pas d'effet sur l'erreurpour cette observation.Dans le cas où l'observation x appartient à la classepositive et est mal classée, on a p x x qui est inférieur à 0.Le vecteur de poids p est alors modifié en un vecteur,appelons-le n pour nouveau, qui est égal à la somme de p et de x.En notation vectorielle, ça donne n = p + x.En transposant le vecteur et en le multipliant par x,on a n t x qui est égal à (p + x)t x x,et si on développe cette expression,on a p t x + x t x.Or, x t x est la normedu vecteur x, c'est-à-dire sa longueur, c'est une valeur positive.Ainsi, on a n t x qui est supérieur à p t x,c'est-à-dire que la nouvelle somme pondérée est plus grandeque la précédente, et donc la somme pondéréedes valeurs de cet exemple tend à être plus juste.Par le même raisonnement, on montre que la modificationdes poids pour les exemples négatives tend à diminuerla valeur prédite et ainsi à la rendre plus juste.On peut donc formellement démontrer la convergencede l'algorithme dans le cas où les deux classes sontséparables par un hyperplan.En revanche, cette convergence est très lente.Il faut généralement considérer plusieurs fois chaque observationpour le faire converger.

## Aborder les réseaux de neurones multicouches

Le perceptron permet de classer des observationslinéairement séparables, c'est-à-dire quel'on peut tracer une droite, ou de manièregénérale un hyperplan, tel queles observations d'une classe se trouvent d'un côtéde celui-ci et les observations de l'autreclasse de l'autre côté.Malheureusement, dans bien des cas,les classes ne sont pas linéairement séparables etle perceptron ne peut donc pas réaliser la tâche souhaitée.Les réseaux de neurones multicouches sont une évolutiondu perceptron qui permet de surmonter cette limiteet de traiter des problèmes non linéairement séparables.En utilisant plusieurs couches de neuroneset une fonction d'activation non linéaire,ils sont potentiellement capables d'apprendre n'importequel type de fonction.Dans un réseau de neurones artificiels multicouches,on distingue généralement trois classes de neurones :les neurones d'entrée, qui sont les unitésqui transmettent chacun une composante du vecteurx décrivant une observation, les neurones de sortiequi indiquent l'étiquette de classe préditepar le modèle, et les neurones cachés qui sonttous les autres neurones.Les neurones sont reliés entre eux par des connexions auxquellessont associés des poids.Ces neurones sont en réalité des fonctions mathématiquesqui sont paramétrées par les poids des connexions en entréeet qui attendent de recevoir les données.Les données arrivent par les connexions d'entrée du neurone,sont transformées par la fonction associée aux neuroneset transmises par les connexions en sortie.L'état d'activation d'un neurone est déterminé par la sommede ses entrées, pondérée par le poids associéà la connexion correspondante.Cet état d'activation est ensuite transformé parla fonction d'activation dont l'objectif estde mettre à la même échelle les données de sortie.La fonction d'activation convertit les valeurs potentiellementgrandes résultant du calcul de la somme pondérée en une valeurdans un intervalle plus restreint.Le choix de la fonction d'activationa une influence importante surle résultat du réseau de neurones.On peut utiliser la fonction signe comme dans le perceptronet affecter la valeur +et la valeur 0 si l'activation est négative.On lui préfère généralement d'autresfonctions non linéaires tellesque la fonction sigmoïde, c'est-à-dire en forme de S,qui est une approximation continue et différentiable,dérivable de la fonction signe.La façon dont les neurones formels sont reliés les uns aux autresest appelée topologie du réseau.Une topologie classique consiste à créer un réseau composéde plusieurs couches superposées.Chaque neurone d'une couche est relié à tous les neurones dela couche suivante et à aucun autre.Dans ce modèle, la couche d'entrée sertà la lecture des données et la couche de sortie à traduirela décision de classification.Un réseau de neurones avec au moins deux couches cachées,chaque couche contenant un grand nombre de neurones,est appelé réseau de neurones profond,deep neuronal network, ce qui a engendré un nouveauchamp d'apprentissage appelé le deep learning.L'apprentissage d'un tel modèle consiste à ajusterla valeur des poids de l'ensemble des connexions de telle sorteque les valeurs des neurones de sortie soient aussi prochesque possible de l'étiquette de classe associéeaux observations dans l'ensemble d'apprentissage.Par exemple, si on doit reconnaître un chiffremanuscrit dans une image, on pourra avoir10 neurones de sortie, chacun correspondant à un chiffre.On utilise une fonction de coût que l'on chercheà minimiser pour déterminer, ajuster la valeur des poids.L'une des fonctions de coût les plus utilisées estla moyenne des erreurs au carré, les erreurs étant la différenceentre la valeur réelle, la target,et la valeur prédite output.Cette valeur est calculée pour l'ensemble des observationsde l'ensemble d'apprentissage.Cette fonction de coût J(w) est minimisée par un algorithmede descente de gradient.C'est un algorithme d'optimisation simple et efficace qui permetde trouver le minimum d'une fonction convexe ouun minimum local lorsque la fonction à optimiser n'est pas convexecomme c'est généralement le cas.Pour illustrer le fonctionnement de la descente de gradient,considérons une fonction de coût convexe avecun seul paramètre poids.Nous recherchons sa valeur minimale par une méthodeitérative qui va nous permettre de nous déplacer sur la courbeen direction du minimum.Comme l'illustre la figure, la descente de gradientconsiste à calculer la droite tangente à la courbe pourle poids w courant et à chaque pas nous prenons la directionopposée au gradient.Le gradient indique comment la valeur dela fonction est modifiée si on se déplace légèrementle long de la courbe .La taille du pas est déterminée par un hyperparamètre,taux, la valeur de la vitesse d'apprentissageainsi que la pente du gradient.Plus on approche du minimum, plus les pas vont être petits.Regardons comment cela se traduit mathématiquement.Comme mentionné précédemment, chaque poids wj est mis à jouren faisant un pas dans la direction opposée du gradient.Il faut donc prendre la dérivée dela fonction J par rapport au poids wj.Nous devons donc calculer la dérivée partiellede la fonction de coût pour chaque poids,dérivée notée δJ /δwj.La dérivée, ou plutôt son opposé, indique la directionà suivre pour descendre le long de la courbe.Maintenant que nous avons la direction,nous devons déterminer la quantité qu'il faut ajouter au poidswj pour réduire l'erreur.Pour cela, nous utilisons le taux d'apprentissage,un hyperparamètre, une valeur fixée lors dela construction du modèle qui peut être considéréecomme un pas effectué dans la bonne direction.Ainsi, la différence à ajouter au poids wj est Δwj qui estégal à -tau x δJ/ δwj, où tau est le tauxd'apprentissage, une constante entre 0 et 1.Ce mécanisme est appelé rétropropagation du gradient.Il consiste à corriger la valeur des poids en propageant l'erreurobservée sur la couche de sortie vers les autres couchesde neurones en amont.Bien que la règle d'apprentissageque nous venons de voir semble identiqueà la règle du perceptron, nous noterons deuxdifférences principales.1, ici la sortie est un nombre réel et nonune étiquette de classe comme dansla règle d'apprentissage du perception,et 2, la mise à jour du poids est calculée sur la basede toutes les observations du jeu d'apprentissage,au lieu de mettre à jour les poids de manière incrémentaleaprès chaque observation.Raison pour laquelle cette approche estégalement appelée descente de gradientpar lot. Une fois les poids du réseau appris,on peut utiliser le réseau pour classer de nouvellesobservations en affectant la valeur de leurs attributsaux neurones d'entrée et en les faisant passer à traversles différentes couches de neurones jusqu'à la couche de sortie.

## Utiliser les réseaux de neurones récurrents

Les réseaux de neurones récurrents sont adaptéspour traiter des données séquentielles.Grâce à leur mémoire interne, ils sont capables dese rappeler des informations importantes sur lesobservations traitées jusque-là, ce qui leur permet d'êtretrès précis dans la prédiction de ce qui va suivre.Ils sont adaptés pour traiter des données telles queles séries chronologiques, la parole, les textes,les données financières, des données audio et vidéo,les données météo, etc.Ce qui caractérise toutes ces données,c'est que la dynamique temporelle qui les relieest plus importante que le contenu même de l'observation.Les réseaux de neurones récurrents sont par exempleutilisés dans l'application de commande vocale Siride Apple et dans Google Translate.Les réseaux de neurones multicouches simples sontappelés Feed-Forward car à travers le réseau,les données ne se déplacent que dans une direction,de la couche d'entrée à la couche de sortie en passantpar les couches cachées.L'information se déplace directement à travers le réseau.De ce fait, la même information ne touchejamais deux fois un même neurone.Les réseaux de neurones Feed-Forward ne mémorisentpas l'observation qu'ils ont reçue précédemment,ils sont donc mauvais pour prédire l'observation suivante.Ce type de réseau ne considère que l'entrée actuelle et n'a pasde notion d'ordre dans le temps.Il ne peut tout simplement passe souvenir de ce qui s'est passé dans le passé,il n'a qu'une vision agrégée des données.Par opposition, un réseau de neurones récurrentest un modèle avec état dans lequel les sortiesdes états précédents alimentent en entréel'état suivant.L'état précédent du réseau de neuronesest l'une des entrées du prochain calcul.Cela signifie que les calculs précédents modifient les résultatsdes calculs futurs.Les calculs intermédiaires sont sauvegardés et utilisésen entrée lorsque l'on considère la donnée suivante.De cette façon, le modèle ajuste ses prévisionsen fonction des données qu'il a traitées récemment.Les réseaux de neurones récurrents ajoutentle passé immédiat au présent.Par conséquent, un réseau neuronalrécurrent a deux entrées : le présent et le passé récent.Ceci est important car la séquence de données contientdes informations cruciales sur ce qui va suivre.Pour cela, le réseau contient des boucles et des cycles.Alors cette astuce permet aux réseaux de neuronesd'apprendre des modèles de séquence de données.Par exemple, un tel réseau a la capacité de prédirele prochain mot le plus probable d'une phrase à partirdes premiers mots de celle-ci.L'apprentissage des poids se fait également par unebackpropagation de l'erreur qui permet d'ajusterla valeur des poids, mais cette fois en prenanten compte le temps.Chaque observation temporelle va être considérée à tour de rôleet être propagée dans le réseau.Pour le calcul des boucles récurrentes,il va être nécessaire de dérouler le réseau.Le déroulement est un outil de visualisation et de conceptionqui permet de comprendre ce qui se passe au sein du réseau.Il permet de visualiser un réseau de neurones récurrent sousla forme d'une séquence de réseaux de neurones entraînésles uns à la suite des autres.Sur la droite du schéma, on a un réseau déroulé.Celui-ci ne contient plus de cycles car lesdifférents pas de temps, ou états du réseau,sont visualisés et les informations sont transmisesles unes après les autres au cours du temps T.La backpropagation temporelle peut être coûteuse en calcullorsque l'on a un grand nombre de pas de temps.Ce type de réseau peut être utilisé pour réaliserde nombreuses tâches, notamment pour faire dela traduction automatique.Pendant de nombreuses années, on a cherché à réaliserla traduction automatique de textes à l'aidede dictionnaires et de probabilités,mais on a obtenu de bien meilleurs résultats en apprenantdes modèles à base de réseaux de neurones récurrents,à partir de très grands corpus de données comportantun texte et sa traduction.L'idée consiste à encoder les mots d'une phrase en un vecteur.Cela nous permet de représenter quelque chose de très compliqué,une phrase, en un vecteur de nombre unique.On utilise un réseau de neurones récurrentafin que l'encodage d'un mot prenne en comptel'encodage des mots précédents de la phrase.Comme une phrase peut être de n'importe quelle longueur,on utilise un symbole d'arrêt, par exemple un point final,qui indique que nous avons atteint la fin de la phrase.Ce symbole d'arrêt est juste un mot supplémentairedans nos données.Maintenant, nous avons un moyen de représenterune phrase entière sous forme d'un ensemblede nombres uniques.Nous ne savons pas ce que chaque nombre dans l'encodage signifie,mais cela n'a pas vraiment d'importance.Tant que chaque phrase est identifiée de manièreunique par son propre ensemble de nombres,nous n'avons pas besoin de savoir exactement commentces nombres ont été générés.Pour la traduction, on utilise un deuxième réseaude neurones récurrents.Il prend en entrée cet encodage et réalise un traitementsimilaire consistant à associer une phrase au codage.Cette phrase peut être la phrase d'origine,mais c'est beaucoup plus intéressantsi c'est une traduction de celle-ci dans une autre langue.Une telle approche nécessite d'entraînerle modèle sur une très grande quantité de données.Mais le modèle est construit de manièrecomplètement automatique sans avoir besoinde connaissances sur les règles de la langue considérée.

## Comprendre les réseaux de neurones convolutifs

Les réseaux de neurones convolutifs forment un typeparticulier de réseau de neurones très performantpour classer des images.Ils sont si performants qu'ils peuvent surpasser l'hommepour identifier des objets dans des images.Ces réseaux ont été développés pour prendreen compte la structure spatiale des imagesdans le réseau. Avec un réseau de neurones classique,une image en niveaux de gris de taille 32 par 32 pixelsest représentée par un vecteur de taille 1024 qui correspondau poids de la couche d'entrée.Ce faisant, la matrice est transformée en un vecteurcomportant par exemple toutes les lignes de valeursles unes à la suite des autres, et la structure spatialede l'image est perdue.Un réseau de neurones classique aura ainsi beaucoup de difficultéà identifier des motifs spatiaux dans cette représentation.Les réseaux de neurones convolutifs sont conçuspour rechercher des motifs spatiaux dans l'image.Pour cela, ils apprennent des représentations internesà l'image en faisant une sorte de moyenne glissante sur l'image.Le type de moyenne réalisée est défini par une matricecarrée appelée "filtre".Ce filtre correspond aux paramètres du neurone.Ce sont les valeurs qui sont apprises au cours de l'apprentissage.Cela permet de reconnaître une forme quel que soit l'endroitoù elle se situe dans l'image.On utilise plusieurs types de matrices carrées,avec des tailles différentes pour permettre l'identificationdes objets à différentes échelles.La moyenne glissante réalisée entre l'image d'origine etle filtre est appelée "convolution".Un filtre est une matrice de petite taille qui contientdes poids dont les valeurs vont être multipliéesavec une partie des valeurs de l'image d'origine, puis sommées.On positionne l'élément central du filtre sur un pixel de l'imageet on calcule la somme des produits des pixelsde l'image avec ceux du filtre.On déplace ensuite le filtre et on calcule la nouvelle convolution.Plus l'image va correspondreau filtre, plus la valeur calculée parla convolution sera élevée.La valeur calculée par la convolution estensuite transformée par une fonction d'activationnon linéaire, généralement la ReLU :Rectified Linear Units.Les valeurs de la convolution calculées sur l'ensemblede l'image source sont enregistrées dansune nouvelle matrice suivant le même arrangementque l'image d'origine.L'idée est ici de ne pas perdre l'arrangementspatial entre les données issues de la convolution.Cette matrice est ensuite sous-échantillonnéeà l'aide d'un algorithme appelé Max Pooling,qui consiste à partitionner la matrice en carrésde taille deux par deux et de garder la valeurmaximale pour chacun des carrés.Cela réduit la taille de la matrice tout en conservantles valeurs les plus importantes.Dans une dernière étape, le tableau réduitest utilisé comme entrée d'un nouveau réseau de neurones.Ce réseau de neurones final décidera si l'imagecontient l'objet recherché.Pour le différencier de l'étape de convolution,ce réseau est appelé "réseau entièrement connecté".Ces étapes peuvent être combinées et empilées plusieurs fois.On peut avoir deux, trois ou même dixcouches de convolution.Les étapes de convolution permettent d'introduire plusde fonctionnalités dans le réseau, et ainsi d'effectuer des tâchesde reconnaissance plus complexes.Ce type de réseau combine deux composantes :une partie "extraction de caractéristiques" etune partie "classification".Les couches de neurones convolution associéesà l'étape de sous-échantillonnage effectuent l'extractionde caractéristiques.Par exemple dans une image, la couche de convolutiondétecte des caractéristiques telles que deux yeux,deux longues oreilles, quatre pattes et une queue courte.Les couches entièrement connectées agissentalors comme un classifieur qui,à partir de ces caractéristiques, attribue à l'imageune probabilité d'appartenance aux différentes classes.Pour réussir cette tâche, les réseaux de neuronesconvolutifs ont besoin de beaucoupd'exemples d'apprentissage et nécessitent égalementde grosses capacités de calcul.

# Apprendre un modèle sur des données

## Sur-apprendre et sous-apprendre

L'apprentissage automatique supervisé peut être vucomme la construction d'un modèle qui approximeune fonction cible "f"."F" associe des variables d'entrée décrivantles observations à une variable de sortie indiquantla modalité de la classe associée à l'observation.Les paramètres du modèle sont ajustés à partirdes observations appartenant aux données d'apprentissage,qui ne sont qu'un échantillon de la population ciblesur laquelle nous souhaitons appliquer notre modèle.À partir d'un même échantillon d'apprentissage,plusieurs modèles peuvent être utilisés,produisant un même taux d'erreurs.Pour autant, ces différents modèles ne vont peut-être pasdonner les mêmes taux d'erreurs sur les données test.Ils ne vont pas avoir la même capacité de généralisation.La généralisation fait référence à la manière dont les conceptsappris par le modèle s'appliquent sur des exemples non vus lorsde la phase d'apprentissage.On dira qu'un modèle a une bonne capacité de généralisations'il se trompe peu sur de nouvelles données ou dansle cas contraire, qu'il souffre de sur ou de sous-ajustement.Le sur-ajustement et le sous-ajustement sont deuxprincipales causes de mauvaises performances d'apprentissage.Le sur-ajustement fait référence à un modèle qui est trop adaptéaux données d'apprentissage.Le sur-ajustement survient lorsqu'unmodèle apprend des détails,des spécificités des données d'apprentissage.Des fluctuations liées à des variations aléatoires,ce qu'on appelle le bruit, sont apprises par le modèleen tant que concept caractéristique des observations.Mais ces concepts ne s'appliquent pas à de nouvelles donnéeset ont un impact négatif sur la capacité du modèleà faire de bonnes prédictions sur de nouvelles données,donc à généraliser.C'est comme si l'on apprenait à identifier les classesen apprenant par cœur la liste des exemples,sans comprendre ce qui permet de les classer ensemble.Le sur-apprentissage est plus probable avec des modèlesqui sont plus flexibles, par exemple les modèlesnon paramétriques ou non linéaires, ce que l'on appelle des modèlesà faible biais comme les k plus proches voisins ou encoreles arbres de décision.De manière plus formelle, on dit d'un modèlequ'il est sur-ajusté s'il a une variance élevéeet un biais faible.Qu'est ce que la variance d'un modèle ?Cela correspond aux variations du modèle lorsque les donnéesd'apprentissage changent.C'est donc la variation de la valeur deces paramètres lorsqu'ils sont appris sur un autreéchantillon d'apprentissage.Dans le cas de sur-apprentissage, comme nous ne faisonsque mémoriser les données d'apprentissage,notre modèle a une grande variance.Il dépend fortement des données d'apprentissage.Le biais est lié au type de modèle choisi.Il représente la force des hypothèses du modèle.Plus les hypothèses sont fortes, plus le biais est fort.Pour séparer deux classes, si je fais l'hypothèse quela frontière est une droite, j'émets une hypothèse forte,plus contraignante que si je suppose que la séparationest n'importe quelle fonction polynomiale.D'ailleurs, l'ensemble des droites possiblesest inclus dans l'ensemble des fonctions polynomiales.Restreindre l'espace des hypothèses possiblesinduit un biais plus important.Un faible biais peut sembler positif :pourquoi voudrions-nous être biaisés envers nos données ?En même temps, nous devons toujours rester sceptiquequant à la capacité de notre échantillon d'apprentissageà représenter l'entièreté de la population cible.Sans biais d'apprentissage, sans contrainte surla forme du modèle, il n'y a pas d'apprentissage possible.Seule une description exhaustive, un apprentissage par cœursans généralisation, est réalisé.En revanche, un modèle avec une faible variance etun fort biais est dit sous-ajusté.Le sous-ajustement fait référence à un modèlequi ne peut ni modéliser les données d'apprentissage,ni se généraliser à de nouvelles données,Les données test.Pour éviter le sur et le sous-ajustement,que pouvons nous faire ?Si nous prenons trop en compte les données,nous apprenons un modèle sur-ajusté.Dans le cas contraire si nous ne considérons pas assez les données,nous obtenons un modèle sous-ajusté.Comment obtenir un bon compromis ?Une solution consiste à utiliser un nouveléchantillon de données appelé "échantillon de validation".Si nous utilisons uniquement un ensemble d'apprentissageet un ensemble de tests, nous ne pouvons pas savoirquelles sont les performances de notre modèle dans le monderéel sur la population cible.Avec un échantillon de validation,nous pouvons faire des pré-tests pour évaluer notre modèleet apporter des améliorations avant le test réelsur l'échantillon test.Ce près-test sur l'ensemble de validation constitueune partie essentielle du développement du modèlepermettant d'évaluer sa variance.Pour résumer dans cette vidéo nous avons vu quatre concepts :le sur-ajustement qui est lié à une trop forte confiancedans les données d'apprentissage ;le sous-ajustement qui indique que l'on n'a pas réussià apprendre la relation entre les données et leurétiquette de classe ;le concept de variance élevée que l'on observelorsque le modèle change de manière significativesur un nouvel échantillon d'apprentissage ;et le concept de biais élevé qui apparaît lorsqueles hypothèses du modèle sont trop simplistes et ne s'appliquentpas aux données d'apprentissage.Le sur-ajustement et le sous-ajustement entraînentune mauvaise généralisation des performancessur l'ensemble des tests.Pour les éviter, on peut utiliser un ouplusieurs ensembles de validation pour évaluerla variance du modèle.

## Mesurer la qualité d'un modèle

L'évaluation d'un modèle d'apprentissageest une étape essentielle qui consiste à mesurerla qualité de la prédiction de l'algorithme surun ensemble de données test, c'est à dire un ensembled'observations pour lesquelles on connaît la valeur dela variable de classe et qui n'ont pas été utiliséespour construire le modèle.La mesure à laquelle on pense spontanémentpour évaluer un modèle est la mesure de justesse,accuracy en anglais.Elle consiste à compter le nombre de fois où l'algorithme donneune bonne réponse sur le nombre de réponses données au total.Cette mesure fonctionne bien si le nombre d'observationsappartenant à chacune des classes est à peu près le même.En revanche, si par exemple on a 98% des observationsqui appartiennent à la classe A et 2% des observationsqui sont de la classe B, alors un modèle prédisanttoujours la classe A aura une exactitude de 98% mais celane sera d'aucune utilité.Il faut donc être vigilant car la mesure de justessepeut nous donner le faux sentiment d'avoir construitun bon modèle. D'autres mesures sont plus robustes lorsqueles classes sont déséquilibrées.Représentons graphiquement les différentes configurationsque l'on peut avoir lorsqu'on compare la valeur dela variable de classe dans le jeu test en ligne,avec la valeur calculée par le modèle en colonnelorsque la variable de classe a deux modalités,0 et 1. Cette matrice dite de confusion croise la vérité des donnéesen ligne avec le résultat du modèle en colonne.À partir de cette matrice, on calculegénéralement deux mesures appelées "précision" et "rappel".Ces mesures sont appropriées pour mesurer la qualitéd'un modèle lorsque la classe 1 correspond aux observationsque l'on cible, par exemple un groupe de malades,la classe minoritaire.La classe étiquetée parlaquelle on s'attend sauf preuve du contraire,l'équivalent à l'hypothèse nulle pour un test statistique.Dans notre exemple, ce peut être le groupede personnes saines.La matrice de confusion nous permet d'identifierquatre groupes d'observation : les vrais positifs,celles pour lesquelles nous avons prédit oui et dont la véritéterrain est également oui ;les vrais négatifs, les observations pour lesquellesnous avons prédit non et dont la vérité terrain est non ;les faux positifs, les cas dans lesquelsnous avons dit oui et la vérité terrain est non ;et enfin les faux négatifs, les observations pour lesquellesnous avons prédit non et dont le résultat réel est oui.On a ainsi deux types d'erreurs, les faux positifs,les fausses alarmes, et les faux négatifs,ceux qu'on a laissé passer dans la classe par défaut alors qu'ilssont dans la classe cible.Cette dernière erreur peut être très préjudiciableselon le contexte.Sur cette matrice, on peut calculer la précisionqui est égale au nombre de vrais positifs classés oui parle modèle et la vérité terrain, divisé parle nombre d'observations classées oui par notre modèle.Par exemple, un modèle de prédiction d'une maladieayant une précision parfaite égale àn'identifie que des personnes qui sontréellement malades, pas de faux positifs.Mais elle peut potentiellement mal identifier des personnesréellement malades.La mesure de rappel correspond quant à elle au nombre de vraispositifs classés oui par le modèle et la vérité terrain,divisé par le nombre d'observationsclassées oui selon la vérité terrain.Un score de rappel parfait ayant une valeur de 1 signifie que tousles malades ont été identifiés correctement par notre modèle,mais ne dit rien sur les personnes saines identifiées comme malades.Les deux mesures sont donc liées.Diminuer l'une signifie augmenter l'autre.On combine généralement ces deux mesures en prenantleur moyenne harmonique qui permet de déterminerle rapport moyen entre les deux ratios :c'est la mesure F1.Les mesures de précision et de rappel peuvent également êtreutilisées lorsque la variable de classe a plusieurs modalités.On peut alors calculer les mesures de précisionet de rappel pour chacune des classes et les combiner.Le dernier cas qu'il est important d'étudierest celui où le modèle appris produit non pas une valeurdiscrète 0 ou 1 pour désigner l'appartenance àune classe ou l'autre, mais une valeur continue entre 0 et 1.C'est le cas de nombreux modèles qui produisent une probabilitéd'appartenance à une classe plutôt que la valeur 0 ou 1.Pour évaluer la qualité d'un tel modèle,il faut déterminer un seuil au-dessus duquel l'observationest affectée à la classe du modèle 1,et en dessous duquel elle est affectée à la classe 0.Comment choisir le seuil ?On va considérer tous les seuils possibles et cherchercelui qui maximise une mesure appelée "AUC ROC",Area Under the ROC Curve.La courbe ROC calcule la probabilité d'un résultat binaire.Il s'agit d'une représentation graphique du taux de fauxpositifs sur l'axe des X par rapport au tauxde vrais positifs sur l'axe des Y pour différentsseuils entre 0 et 1.En d'autres termes, c'est le taux de fausses alarmespar rapport au taux de réussite.Le taux de vrais positifs est calculé en divisant le nombrede vrais positifs par la somme du nombre de vrais positifset du nombre de faux négatifs.Il décrit à quel point le modèle permet de prédirela classe positive lorsque le résultat est positif.Le taux de vrais positifs est exactement identique à la mesurede rappel vue juste avant.Le taux de faux positifs est calculé en divisant le nombrede faux positifs par la somme du nombre de faux positifset du nombre de vrais négatifs.C'est le taux de fausse alerte, car il résume la fréquenceà laquelle une classe positive est prédite lorsque le résultatréel est négatif.La courbe ROC est un outil utile car les courbesde différents modèles peuvent être comparées directementpour différents seuils.Ainsi, l'aire sous la courbe peut être utiliséepour résumer la performance du modèle d'une manièreindépendante du seuil pris pour la classification.Un modèle performant attribuera une probabilitéplus élevée à un événement positif réel choisi au hasardqu'à un événement négatif.Cela se traduit sur la courbe par une aireplus importante au-dessus de la diagonale principale.Un modèle avec une AUC entre 09 et 1 est excellent.Un modèle avec une AUC en dessous de 06 ou 07 est médiocre.Notez bien que les courbes ROC sont appropriéespour évaluer des modèles lorsque les classes sont équilibrées.Lorsque les observations sont déséquilibrées entre les classes,les courbes de rappel et de précision sont plus appropriées.Les mesures de précision et de rappel ne font pas appel aux vraisnégatifs et ne concernent que la bonne prédiction des observationsde la classe minoritaire, la classe 1.Une courbe de rappel précision indique la précisionen ordonnée et le rappel en abscisse pour différentesvaleurs de seuils, de manière assez similaire à ce que l'on faitpour tracer la courbe ROC.La ligne de base est définie par le nombre total de caspositifs divisé par le nombre total de cas positifs et négatifs.Pour un ensemble de données avec un nombre égal de caspositifs et négatifs, il s'agit d'une ligne droite à 05.Les points au-dessus de cette ligne indiquent les valeursde seuils pour lesquels le modèle apporte de la connaissance.On peut, comme dans le cas de la courbe ROC,calculer l'aire sous la courbe pour comparer plusieurs modèles.

## Valider et tester un modèle

Valider un modèle de classification appris à partir d'exemplesest une étape importante dans un projet de machine learning.Dans cette étape, on va chercher à évaluerles performances du modèle tout en déterminant les valeursdes hyperparamètres du modèle, les paramètres qui ne sontpas appris à partir des exemples d'apprentissage,en choisissant le modèle ayant les meilleuresperformances en généralisation.On a donc deux ensembles d'observations :un dit d'apprentissage et un autre servant à évaluer le modèle.L'ensemble servant à l'évaluation du modèledoit être indépendant de celui servant àsa construction. Généralement, il s'avère que le modèle n'est pas aussiperformant sur les données de validation qu'il l'estsur les données d'apprentissage.Le taux d'erreur tend à être surestimé lorsque l'on utiliseun seul ensemble de validation de taille modeste.L'importance de cette différence peut être importante,en particulier lorsque la taille de l'ensemble des donnéesd'apprentissage est petite ou lorsque le nombre de paramètresdu modèle est grand.Maintenant, si je choisis les hyperparamètres du modèleen fonction des performances de celui-ci sur l'ensembled'évaluations, est-ce que mon évaluationest bien indépendante de la construction de mon modèle ?Est-ce que je n'ai pas utiliséde l'information contenue dans les observations servant à l'évaluationpour construire mon modèle ?Si bien sûr, puisqu'on a pris le modèle ayant les meilleuresperformances sur les observations utilisées pour validation.Pour régler les hyperparamètres d'un modèle,par exemple le nombre k de plus proches voisins,la structure d'un réseau de neurones ou bienle nombre minimal d'observations par feuille d'un arbrede décision, on a besoin de trois ensemblesindépendants d'observation : un ensembled'apprentissage sur lequel on apprend les paramètresdu modèle ; un ensemble de validation à partir duquel on choisitles hyperparamètres ;et un ensemble de test sur lequel on peut mesurer les performancesen généralisation de notre modèle.Pour être bien clair, les observations de l'ensembled'apprentissage sont utilisées pour adapter le modèleaux données du problème.Le modèle est construit automatiquement à partirde ces données.Les observations de l'ensemble de validation permettent d'avoirune évaluation non biaisée du taux d'erreur lors du réglagedes hyperparamètres du modèle.On peut ainsi éviter d'avoir un modèle sur-ajustéaux données d'apprentissage.Cependant, l'évaluation devient de plus en plusbiaisée au fur et à mesure qu'on faitdes évaluations sur le jeu de validationpour choisir notre modèle, et ce même si le modèle voitles données de validation mais ne les apprend jamais.On peut dire que l'évaluation des modèles sur l'ensemblede validation affecte le modèle de manière indirecte.Les données de l'ensemble test sont utiliséespour fournir une évaluation non biaisée du modèle.L'ensemble de données de test est l'étalonpermettant d'évaluer le modèle.Il n'est utilisé qu'une fois le modèle complètement formé.On peut observer dans certains projets que le jeu de validationest utilisé comme jeu de test, mais ce n'est pas une bonnepratique puisque cela viole l'hypothèse d'indépendance.L'ensemble de test est généralement bien construitpour contenir des observations bien diversifiées couvrantles différentes classes auxquelles le modèle seraconfronté lorsqu'il sera utilisé en productiondans le monde réel, et ainsi éviter un biais de sélection.Si l'on dispose de nombreuses observations étiquetées,on peut avoir ces trois ensembles distincts.On va séparer les données étiquetées en trois ensemblesde manière aléatoire.En revanche, si le nombre d'observationsdont on dispose est limité, on risque d'être sujetau biais d'échantillon, c'est-à-dire que l'onrisque que nos trois ensembles pris séparémentne soient pas représentatifs de la population cible.L'évaluation des performances est sujette à une varianceplus importante compte tenu de la taille réduite des données.Une technique largement utilisée pour contrerce problème est de procéder à une validation croisée.Celle-ci consiste à utiliser alternativement une partiedes données pour faire l'apprentissage etla partie restante pour réaliser la validation.De manière concrète, les données sont partitionnéesde manière aléatoire uniforme en k groupes de même taille,et le modèle est appris k fois en utilisant à chaque fois undes k sous-échantillons différents comme ensemble de validation,et les k moinscomme données d'apprentissage.À la fin, la mesure de performance utilisée,la mesure de justesse, ROC, est moyennéesur les k évaluations.Comme on peut le constater, chaque donnée figuredans un ensemble de validation exactementune fois et participe à un ensemble d'apprentissagek moins une fois.Cela réduit considérablement les biais car nous utilisonsla majorité des données pour apprendre le modèle,et réduit également la variance de manière significative carles données sont également utilisées dans les ensembles de validation.En règle générale, on fixe K entre 5 et 10,mais on peut prendre n'importe quelle valeur.On peut même fixer k comme étant égal à n le nombre de données,ce qui donne une très bonne estimation du taux d'erreursmais qui est très long à calculer, à moins que l'onparallélise le calcul.L'intérêt de cette approche est qu'elle est exhaustive.Elle considère toutes les manières possibles de diviserl'échantillon d'origine en un échantillon d'apprentissageet un échantillon de validation.Enfin, une fois que la meilleurecombinaison de paramètresa été trouvée, le modèle est réapprissur l'ensemble des données, sur l'ensembledes k échantillons.Les avantages de la k validation croisée sontque l'estimation des performances du modèle est moins sujetteau biais de sélection, car chaque observationva être utilisée pour l'apprentissagedu modèle et aussi pour la validation,mais pas en même temps.Les inconvénients de cette stratégie de validation sontdes coûts de calcul plus élevés.Le modèle est appris k fois durant l'étape de validation,plus une fois pour l'étape de test.Il est important de noter que toutes préparationsde données utilisant la connaissancede la variable de classe, comme des méthodes de sélectionde variables par exemple, réalisées avant l'ajustementdu modèle, ou bien le réglagedes hyperparamètres du modèle,doivent être répétées pour chaque partition des données.Ceci permet d'éviter d'utiliser la connaissance de la variablede classe de l'échantillon de validation lorsde la construction du modèle.Sinon, cela peut aboutir à une surestimationdes performances du modèle.Dans certains cas, il peut exister un déséquilibreimportant entre les modalités de la variable de classe,par exemple en ayant beaucoup plus d'observationscorrespondant à la population des sujetssains que d'observations de la classe des personnesmalades. Dans ce cas, il peut être préférable de ne pas construireles k sous-échantillons de manière uniforme mais des'assurer que les différentes modalités de la variablede classe seront bien présentes dans les différents échantillons.C'est ce qu'on appelle la k validation croisée stratifiée.

## Apprendre un modèle sur des données temporelles

Nous allons voir dans cette vidéo comment validerles performances d'un modèle visant à prédire la valeurd'une série chronologique ou d'une variable de classedans le futur. C'est-à-dire qu'à partir de variablesmesurées dans le passé,on cherche à anticiper la valeur de la variable de classe.L'aspect temporel injecté dans le modèle va perturber les méthodesde validation habituelles.Regardons Tout d'abord comment on construitgénéralement un modèle à partir de sérieschronologiques. On va chercher à décrire la série chronologiqueà l'aide de variables décrivant son comportementsur une fenêtre temporelle.On peut par exemple considérer sa valeurmoyenne, ses valeurs extrêmes ou encorela pente d'une régression linéaire ajustée sur ses données.On réalise ces mesures sur des fenêtres de différentestailles remontant de plus en plus loin dans le passé.Selon la tâche considérée, on affectera à cette observationla valeur à prédire à un jour, une semaine ou un moisselon l'échéance à laquelle on souhaite faire la prédiction.Á partir de ces données, il est alors possibled'utiliser un algorithme d'apprentissage pour ajusterun modèle à ces données, de la même manière qu'on le feraitpour des données non temporelles.Ce qui change surtout, c'est la façon dont on va valideret tester notre modèle.Comment choisir les échantillonssans provoquer de fuites de données,des fuites de données de tests dans les données d'apprentissage,des fuites de connaissance de la prédiction correcteou de la vérité terrain dans les données de tests,en gros des fuites d'informations du futur vers le passé.Comment garantir que nos échantillons d'apprentissage,de validation et de tests sont indépendants ?Le découpage de ces échantillons ne peut pas se faire de manièrealéatoire uniforme.En effet dans les données chronologiques,il faut veiller à construire ces échantillons de sorteà éviter toute fuite de données.Il faut que la prévision se fasse à partir de données similairesà celles qu'utilisera le modèle pour faire ses prévisions.C'est-à-dire que les données de validation et de tests doiventcorrespondre à des événements qui se produisent de manièrechronologique après les événements utilisés pour apprendre le modèle.Les données du jeu de tests viennent chronologiquement aprèsles données d'apprentissage.De même, l'ensemble de validation vient chronologiquement aprèsle sous-ensemble d'apprentissage.Une autre difficulté peut provenir du choix des données de tests,plus ou moins espacées des données d'apprentissage.Ce choix peut entraîner que le taux d'erreur mesurésur l'ensemble de tests soit une mauvaise estimationde l'erreur sur un autre ensemble indépendant.Ce problème peut être surmonté par une technique appeléevalidation croisée imbriquée.Son principe est de faire une succession de découpageschronologiques pour obtenir une estimation robustedes performances du modèle.La méthode repose sur l'utilisationde deux boucles itératives, l'une externe qui faitdifférentes partitions en un ensemble d'apprentissage etun ensemble de tests et l'autre, interne qui, partantde l'échantillon d'apprentissage, le découpe pour obtenirun échantillon de validation sur lequel les hyperparamètresdu modèle sont choisis.Tous les découpages sont faits de sorte à préserverl'ordre chronologique avec l'ensemble d'apprentissagesantérieurs aux échantillons de validation et de tests.La boucle externe divise le jeu de donnéesen plusieurs ensembles différents d'apprentissageet de tests, et l'erreur sur chaquedivision est moyennée afin de calculer une estimationrobuste de l'erreur du modèle.Ceci permet d'avoir une estimationpresque sans erreur de l'erreur réelle.

## Découvrir le bootstrap

Le bootstrap est un outil statistique puissantet flexible qui peut être utilisé pour quantifierl'incertitude associée à un modèle d'apprentissage.Par exemple il peut être utilisé pour estimer la moyenne,l'écart-type ou même un intervalle de confiancepour les paramètres du modèle.Il peut également être utilisé pour estimer les performancesdu modèle sur des observations non incluses dans l'ensembled'apprentissage. La méthode bootstrap est une technique statistiquepermettant d'estimer des valeurs statistiques sur une population,en faisant la moyenne des estimations obtenuessur de nombreux échantillons issus de cette population.Cette méthode consiste à tirer de manière aléatoire uniformedes observations d'un grand échantillon d'observationsl'une après l'autre, et en les remettantdans l'échantillon d'origine une foisqu'elles ont été choisies.Cela permet de choisir une observation plusieursfois et d'avoir un tirage d'une observation qui soitindépendant des tirages réalisés précédemment.Cette méthode d'échantillonnage s'appelleéchantillonnage avec remise.Le processus de création d'un échantillon peut êtrerésumé comme suit.On choisit la taille de l'échantillon (n).Tant que la taille de l'échantillon estinférieure à la taille choisie (n),on va sélectionner au hasard une observation dansle jeu de données et l'ajouter à l'échantillon.Ainsi pour estimer une statistique,la moyenne, les quartiers par exemple d'une variable,on va répéter l'estimation de cette statistiquesur les différents échantillons de taille(n) et prendre la moyenne des statistiques calculées.Cet outil statistique peut être utilisé dans différents contextes,notamment pour quantifier l'incertitude associéeà un estimateur donné ou à un modèle d'apprentissage.Pour cela, le modèle d'apprentissage estajusté sur les observations d'un échantillonet ses performances sont évaluées surles observations n'appartenant pas à cet échantillon.Pour résumer, la méthode bootstrap nécessite de choisirun nombre d'échantillons k, de choisir la taille nde chaque échantillon.Puis pour chaque échantillon, on va tireravec remise un échantillonde taille n, ajuster le modèle d'apprentissagesur l'échantillon, puis estimer les performancesdu modèle sur les observations n'appartenantpas à l'échantillon.Une fois tous les échantillons traités,on peut calculer la moyenne sur les différentséchantillons des mesures de performance du modèle.Il est important de noter que toute préparationde données utilisant la connaissance dela variable de classe, comme des méthodes de sélectionde variables ou bien le réglage des hyperparamètres du modèle,doit être répétée sur chaque échantillon de données,c'est-à-dire à l'intérieur de la boucle.Ceci permet d'éviter les fuites de données qui peuvent aboutirà une estimation optimiste des performances du modèle.La mise en œuvre de la méthode requiert de fixer deux paramètres,la taille de l'échantillon et le nombre de répétitionsde la procédure.Pour estimer les performances d'un modèle d'apprentissage,il est courant de choisir la taille de l'échantilloncomme étant la même que la taille du jeude données d'origine.Comme certaines observations vont être répétées,d'autres n'appartiendront pas à l'échantillon et pourrontêtre utilisées pour l'ensemble de validation.Si l'ensemble de données étiquetées ordonnées de testest énorme et que cela ralentit le temps de calcul du modèle,des échantillons de taille inférieure peuvent être utilisés.Concernant le deuxième paramètre, le nombre d'échantillonsconsidéré doit être suffisamment grand pour queles statistiques soient fiables.Il faut au minimum entre 20 et 30 répétitions.Utiliser un nombre plus petit d'échantillons risque d'accroîtrela variance des statistiques calculées sur l'échantillon.L'avantage de la méthode bootstrap par rapportà d'autres méthodes de validation comme la validation croisée,est qu'elle produit une estimation plus fiable du taux d'erreur.En revanche, comme cette approche implique la répétitionde l'apprentissage du modèle sur de nombreux échantillons,cette méthode est gourmande en ressource informatiqueet nécessite de réaliser beaucoup de calculs.

## Régler les hyperparamètres

La plupart des méthodes de machine learningcomportent des hyperparamètres, c'est-à-dire des paramètresdont la valeur n'est pas apprise à partir des donnéesd'apprentissage, mais est fixéeavant l'apprentissage du modèle.Par exemple un hyperparamètre est le paramètre k dans la méthodedes K-Plus proches voisins.Le réglage des hyperparamètres est délicat,et un bon réglage va permettre d'obtenir un bien meilleur modèle.Le principe consiste à tester différentes configurationsdes hyperparamètres lors de l'apprentissage du modèle,et de retenir celle qui minimise le taux d'erreur.Revenons sur la notion d'hyperparamètres.Les hyperparamètres dirigent le processus d'apprentissage.L'algorithme manipule trois types de donnéespendant la phase d'apprentissage du modèle.Les données d'apprentissage, qui sont l'ensembledes observations étiquetées qui servent d'exemplepour le processus à apprendre, sont utilisées pour ajusterle modèle aux données afin qu'il soit capablede classer avec précision de nouvellesobservations similaires.Les données d'apprentissage permettent d'estimer la valeurdes paramètres du modèle, mais ne sont pas directementintégrées dans le modèle.Ensuite on a les paramètres, qui sont des variables du modèleque la technique d'apprentissage choisie modifie afin quele modèle soit ajusté au problème de classification considéré,c'est-à-dire ajusté aux observations relativesà la tâche à apprendre.Par exemple un réseau de neurones profonds est composé de nœuds,les neurones, reliés entre eux par des connexions.Á chaque connexion est associé un poids,le paramètre de l'opération mathématique réaliséepar le nœud sur les données lorsque celles-ciparcourent le réseau.Ces points sont les paramètres du modèle.Á bien des égards, les paramètres du modèlesont le modèle, ce qui le distinguedes autres modèles du même type ajustés en utilisantd'autres observations.Enfin on a les hyperparamètres, qui sont les variablesqui régissent le processus d'apprentissage lui-même.Par exemple configurer un réseau de neurones profonds consisteà déterminer le nombre de couches cachées etle nombre de neurones par couche.Ces variables ne sont pas directement liées aux donnéesd'apprentissage, ce sont des variablesde configuration.Ce qui distingue les paramètres des hyperparamètres,c'est que les premiers changent durant la phased'apprentissage alors que les seconds restent constants.Alors que ces hyperparamètres ne sont pas apprisà l'aide des données, leur influence surla qualité du modèle est cependant très importante.La question se pose alors de savoir comment régler la valeurdes hyperparamètres, comment déterminer la valeurqui conduit au meilleur modèle.Cela revient à rechercher les valeurs d'hyperparamètresqui optimisent la qualité du modèle,par exemple qui minimisent le taux d'erreur ou touteautre mesure de validation mesurée sur un ensemblede validations indépendant de l'ensemble d'apprentissage.Or cette mesure de qualité du modèle en fonctiond'une combinaison de valeurs des hyperparamètresn'a pas de bonnes propriétés, comme par exemple être convexe,ce qui rend son optimisationdélicate. Une méthode classique d'optimisationdes hyperparamètres est appelée "recherchesur grille" et consiste en une recherche exhaustive,dans un sous-ensemble spécifié manuellement,de l'espace des valeurs possibles des hyperparamètres.Ainsi pour chaque hyperparamètre, on spécifie un ensemblede valeurs possibles.La recherche sur grille va considérer toutesles combinaisons possibles de ces valeurs et retenirla combinaison qui optimise la mesure de qualitédu modèle. Dans notre exemple de réseau de neurones profonds,nous avons deux hyperparamètres, le nombre de couchescachées (C) et le nombre de neuronespar couche (N). Pour effectuer une recherche sur grille,nous devons sélectionner un ensemble fini de valeursà considérer pour chacun des hyperparamètres.Par exemple (C) peut prendre une valeur dans l'ensemble 2,3, 4 et (N) dans l'ensemble 10, 100, 1000.La recherche sur grille va apprendre un modèlepour chaque paire (C-N) de valeurs d'hyperparamètreset évaluer la performance du modèle obtenusur l'ensemble de validation.Finalement les valeurs d'hyperparamètres conduisantau meilleur modèle sont retenues, et le modèle est réapprissur les données d'apprentissage et de validation.La recherche sur grille peut nécessiterbeaucoup de calculs, surtout si le nombrede valeurs considérées par hyperparamètreest grand, mais ces calculs peuventêtre parallélisés puisque les combinaisonsde valeurs d'hyperparamètres sont évaluées de manièreindépendante les unes des autres.Le problème majeur avec l'optimisationdes hyperparamètres est que l'évaluationde la fonction objectif, c'est-à-dire le calculdu modèle de classification, est extrêmement coûteuse.Chaque fois que nous essayons une combinaison d'hyperparamètres,nous devons apprendre un modèle, classer toutes les observationsde l'ensemble de validation, puis calculer la métrique de validation.Si le nombre d'hyperparamètres est important et si le modèleest long à apprendre, ce processus devientdifficile et la méthode exhaustive par recherchesur grille peu efficace.On peut utiliser une approche bayésienne pour déterminerles hyperparamètres.Une telle approche, contrairement àla recherche sur grille, va prendre en compte les résultatsdes évaluations antérieures pour choisir les nouvellesvaleurs de paramètres à tester, et ainsi limiter le nombrede modèles à apprendre, le nombre de combinaisonsd'hyperparamètres à évaluer.On cherche à déterminer la combinaisond'hyperparamètres h\* dans l'espace des hyperparamètres h telle que la mesure de performancef de notre modèle sur les données x soit maximale.La fonction f est inconnue, sauf en un certainnombre fini de points h1,h2, hn. On peut estimer dans quelle mesure une nouvelle combinaisonh pourra donner un meilleur résultat que les combinaisonsdéjà évaluées.Il y a une infinité de fonctions qui passentpar un nombre fini de points.On va chercher à obtenir de l'information surcet ensemble de fonctions aléatoires.En faisant l'hypothèse minimaliste que les valeursproches sur les hyperparamètres doivent donner des valeursproches sur la fonction f, on modélise cet ensemblede fonctions aléatoires par un processus gaussien.Cela nous permet d'estimer la valeur moyenne des fonctionspour une combinaison h, ainsi que leur variance.Un point intéressant est que si f(h) suit un processus gaussien,f(h) connaissant les points h1, hn suit également un processusgaussien pour lequel on peut estimer la moyenne et la variance.Á partir de cette modélisation de l'ensembledes fonctions aléatoires, on va déterminer une nouvellecombinaison à évaluer.Alors différentes stratégies existent et reviennent à trouverun compromis entre l'exploration de l'espace des hyperparamètresH et l'exploitation de la connaissance que l'on a de f,c'est-à-dire choisir un point h de telle sorte à optimiser f.Ces stratégies sont implémentées par une fonction d'acquisitionqui tend à explorer des valeurs de h très différentes de cellesdéjà vues, en prenant des combinaisons hà forte variance et qui exploitent la connaissance que l'on a sur f,en prenant une combinaison h qui a une forte valeur moyenne.Cette approche permet de déterminer de bonnesvaleurs d'hyperparamètres sans devoir testertoutes les combinaisons de valeurs possibles.

## Comprendre les méthodes ensemblistes

Les méthodes ensemblistes consistent à combinerplusieurs modèles d'apprentissage dans l'objectif d'avoirun modèle plus performant ayant des performancesprédictives supérieures à celles de chacun des modèlespris indépendamment.Les méthodes ensemblistes peuvent être divisées en deux groupes.1 Les méthodes ensemblistes séquentielles où les modèlessont appris les uns après les autres.Apprendre les modèles de manière séquentiellepermet d'inciter le modèle suivant à considérer de manièreplus importante les exemples précédemment mal étiquetés,par exemple en leur affectant un poids plus élevé.2 Les méthodes ensemblistes parallèles,quant à elles, apprennent différents modèles en parallèle.L'idée est ici d'exploiter l'indépendance entre les modèles.La plupart des méthodes ensemblistes utilisentun seul type d'algorithme d'apprentissagepour produire des modèles de base homogènes.Cependant on peut aussi avoir des modèles de typesdifférents conduisant à des ensembles hétérogènes.Pour qu'une méthode ensembliste ait de meilleures performancesque chacun des modèles sur lesquels elle repose,les modèles de base doivent être aussi préciset diversifiés que possible.Une première approche s'appelle Bagging,pour Bootstrap aggregating.Le principe est d'utiliser une approche bootstrapqui permet de réduire la variance d'une estimationen faisant la moyenne de plusieurs estimationsfaites sur des échantillons indépendants.La méthode consiste d'abord à échantillonnerplusieurs sous-ensembles avec remise à partirdes données d'apprentissage, c'est-à-dire qu'onréalise un échantillonnage par bootstrapping,puis à ajuster un modèle de classification sur chacunde ces sous-ensembles.Enfin la classification d'une nouvelle observationse fait en prenant la classe majoritaire pour un problèmede classification, ou bien la valeur moyennepour un problème de régression.Par une telle approche, on réduit la variance du modèle.Le boosting quant à lui est une approcheséquentielle dans laquelle chaque nouveau modèleattribue un poids plus importantaux observations précédemment mal classées,forçant ainsi le modèle à se concentrer sur les observationsdifficiles à classer.L'algorithme de boosting le plus connu est AdaBoost.Chaque observation est initialisée avec un poidségal à 1/N et à chaque itération, un nouveau modèle M estajusté de sorte à minimiser l'erreur de classificationpondérée par les poids associés à chaque observation.La quantité ε représente le taux d'erreur pondéré du Classifier.Il vaut 0 si toutes les observationssont bien classées, et 1 si aucune n'est bien classée.Un poids α associé au modèle M est d'autant plus grandque le modèle classe bien.Ce poids est positif si l'erreur ε est inférieure à 1/2.On suppose ici qu'on a un problème de classificationà deux classes les plus probables et donc que l'erreurde décision aléatoire est 1/2.Plus l'erreur du modèle M est faible,plus son coefficient α est grand.L'examen des formules de mise à jour des poidsdes modèles montre que vers la fin de l'apprentissage,le poids des observations difficiles à classerdevient dominant.Á la fin, la règle de classificationd'une nouvelle observation estla somme pondérée des règles associéesà chaque modèle.Cette approche permet de réduire le biais d'apprentissage.Enfin les méthodes de type "empilement de modèles"ou stacking consistent à ajuster séparément plusieurs modèles,puis à apprendre un Meta-Classifier,c'est-à-dire un modèle dans lequel les observationssont maintenant décrites par leurs valeurs de variablesde classe sur chacun des modèles.Dans une première étape, plusieurs modèles de typepotentiellement différents sont ajustés aux observationsde l'ensemble d'apprentissage.Dans une deuxième étape, un nouveau modèle est ajustéà partir de toutes les prédictions des autres algorithmes commedescription des observations.L'empilement donne généralement de meilleures performancesque n'importe lequel des modèles de base et est utilisé dans denombreux challenges.

# Mettre en œuvre les méthodes

## Découvrir et visualiser les données Iris

Dans cette vidéo, nous allons considérer le jeude données Iris qui est le résultat de mesures réaliséesdans les années 30 afin de quantifier les variationsde morphologie des fleurs d'iris issues de trois espèces,les espèces Setosa, Versicolor et Virginica.Donc quatre caractéristiques ont été mesurées sur chaqueobservation, la longueur et la largeurdes sépales et des pétales en cm de chaque iris.Ce jeu de données est un classiquedu machine learning, il est donc présent dansla librairie scikit-learn.On peut directement y accéder en appelant la fonctionload\_iris du package datasets de scikit-learn.On peut maintenant regarder la description de ces données.Alors on voit que les données comportent 150 observations,avec les quatre variables.On a la variable de classe qui prend trois valeursqui correspondent aux trois espèces.Et chacune des classes est équilibrée et contientun tiers des données.On peut maintenant regarder plus précisément les valeursde ces données.Ici on a les valeurs numériques de chaque iris.On peut également regarder le contenude la variable de classe, qui prend des valeursnumériques entre 0 et 2.Avant de construire un modèle de classification,nous allons construire un graphique en deux dimensions,un nuage de points ou scatter plot en anglais.Pour ce faire, on va appeler la librairie matplotlib.On va affecter les observations à la variable X,la variable de classe à la variable Y comme c'est le casgénéralement en machine learning.On a ici deux variables qui vont représenter les attributs quel'on va afficher sur le graphique.Donc la variable abscisse va représenter l'attributaffiché en X, et la variable ordonnée l'attribut affiché en Y.On a ici le nom des attributs qu'on va représentersur le graphique, donc en X et en Y.Ensuite on va parcourir les trois classes du jeu de données,et pour chacune des classes on va sélectionner les observations de la classe.Avec cette instruction [Y==i] on va sélectionner les observationsde la classe i. Et ensuite ici on va projeter ces donnéesen gardant toutes ces lignes et la colonnequi correspond à la variable abscisse,donc celle qui sera représentéesur l'axe des X. De la même façon ici on va sélectionner toutesles observations de la classe i et on va retenirla colonne spécifiée dans la variable ordonnée.On affiche également l'étiquette de la classe,on affiche la légende et le graphique.On obtient le graphique suivant où on peut observerque la classe setosa est très séparée des deux autres,qui sont assez mélangées donc sur ces deux attributsqui sont la longueur du sépale et la largeur du sépale.Si maintenant on regarde ce qui se passe au niveau des pétales,on change ici la valeur des attributs,on recalcule le graphique et on voit qu'il est très facile encoreune fois de séparer les iris de la classe setosa des deux autres.Et les deux autres classes, vericolor et virginica,sont maintenant mieux séparées même si on voitici quelques observations qui sont un peu mélangées.Et peut-être que nos algorithmes de machine learning auront plusde difficultés sur ces points.

## Apprendre des modèles de machine learning sur les données Iris

Nous allons maintenant chercher à ajuster un modèle de machinelearning sur nos données à l'aide de la librairie scikit-learn,et on va voir que quatre lignes de code sont suffisantespour créer ça. Donc dans un premier temps on va construire un classifierde type Naive Bayes.Pour cela on va utiliser la fonction GaussianNB ( )qui permet donc de construire un classifier naive\_bayes,en estimant les probabilités des variables numériquesà l'aide d'une voie de distribution gaussienne.Á l'aide de la fonction fit, on va ajuster les paramètresdu modèle sur nos données X et Y.Et ici on va chercher à regarder les erreurs que commetle classifier sur les données X, en cherchant à prédire la variablede classe pour les données X.On compare ici la différence entre la variable de classeY et la variable Y\_predit qui a été renvoyée par le modèle.Et on compte ainsi les observationsqui ont été mal classées par le classifier.On a ici 6 observations qui sont mal classées.Alors on peut chercher à visualiser où se situentles iris qui ont été mal classés.On va rajouter une petite ligne dans notre graphique ici,pour afficher les observations qui ont été mal prédites en noir.Et sans trop de surprises effectivement,les iris mal classés se situent ici dans la jonction des classesvirginica et versicolor qui sont un peu mélangées.On peut maintenant construire un modèlede type K plus proches voisins,toujours avec scikit-learn.Là, la fonction s'appelle KNeighborsClassifier,elle prend un paramètre.Donc ici on fixe le nombre de voisins à prendre en comptedans la tâche de classification comme étant égal à 7.On ajuste le modèle, on le prédit sur la variableX et on compte les observations mal classées.On a ici 4 observations qui sont mal classées,donc un classifier un petit peu meilleur que le précédent.Et on peut regarder là encore où se situent les erreurs.Les erreurs sont toujoursau même endroit, même s'il y en a moins.On peut maintenant construire un arbre de décision sur nosdonnées toujours Iris, avec la fonctionDecisionTreeClassifier qui ne prend pas de paramètre.On ajuste le modèle avec la fonction fit.On prédit les valeurs associées à la variable Xet on compte les observations qui ont été mal classées.Et là on s'aperçoit que la classification est parfaite,toutes les observations ont été correctement classées.Alors l'intérêt des arbres de décision,c'est également qu'il est possible de les visualiser.C'est ce que l'on va réaliser avec la fonction export\_graphviz.Cette fonction va nous permettre de visualiserl'arbre associé à notre modèle.Ici on va traduire notre modèle dans un fichier de type .dotqui va s'appeler ici 'arbre\_decision.dot'.On va ensuite faire appel à la fonction dot pour convertirnotre fichier .doten .png, en image.Et enfin on va afficher notre image,qui va correspondre à notre arbre de décisions.Donc dans notre arbre, on s'aperçoit qu'il esttrès facile ici de classer les iris de la classe setosa,les 50 Iris de la classe setosa.On doit juste prendre cet attribut longueurdu pétale et faire en sorte qu'il soit inférieur à 2.pour pouvoir spécifier que la classe est de type setosa.En revanche, pour différencier les classes versicoloret virginica, il faut beaucoup plus de critères imbriqués.Mais avec un plus grand nombre de critères imbriqués,on peut arriver à une décision parfaite sans aucune erreur.

## Transformer des données catégorielles

Dans cette vidéo, nous allons apprendreun modèle de machine learning sur des donnéesnécessitant des transformations.Nous allons transformer les attributs pour qu'ilssoient tous de type numérique afin de pouvoir apprendreun modèle de machine learning.Nous allons utiliser les données seismic-bumpsqui proviennent de l'UCI machine learning Repository.Ces données portent sur la prévision d'événementssismiques dangereux dans des mines de charbon.L'objectif principal est de prévoir l'augmentationde l'activité sismique susceptible de provoquerun éclatement minier.Chaque observation est décrite par 18 variablesdécrivant l'activité sismique dans la masserocheuse au cours des huit heures précédant l'observation.Les données sont au format csv, donc nous allons les chargerdans notre environnement Python à l'aide de la fonction read\_csvde la librairie Pandas.Cette fonction renvoie les données sous format d'un dataframe,une sorte de fichier Excel sous Python qu'on peut visualiser ici.Dans ce tableau on peut observer que certaines variables prennentdes valeurs numériques, des chiffres, alors que d'autresprennent des valeurs symboliques représentées par des caractères.Les attributs indiceants 1, 2 et 7 prennent leur valeurdans l'ensemble a,b,c,d et l'attribut indiceant 2 prendses valeurs dans l'ensemble N,W.Avant d'apprendre un modèle de machine learning,nous allons devoir transformer ces attributs.Chaque attribut va être transformé en autantd'attributs binaires qu'il a de valeurs différentes.Ainsi les attributs 0, 1 et 7 vont chacunêtre transformés en quatre attributsbinaires et l'attribut 2 va être transformé,lui, en deux attributs binaires.Avant de binariser les données, nous allons séparerla variable de classe des variables descriptives.Donc on va affecter lesla variable d et la variable de classe à la variable t.Nous allons créer une fonction binarisation qui va êtreappelée sur chacun des attributs devant être binarisés.Donc cette fonction prend en paramètresd'entrée les données, l'indice de l'attribut à binariserainsi qu'un tableau valeurs qui va contenir toutes les valeurspossibles prises par l'attribut.Au début de la fonction, on crée une variable bqui est un tableau qui ne contient que des zéros,et qui a autant de lignes qu'il y a d'observationsdans le tableau d et autant de colonnesqu'il y a de valeurs dans le tableau valeurs.Á l'aide de la boucle indicéepar i, nous allons parcourir toutes leslignes du tableau d et à l'aide de la boucle indicée par v,nous allons parcourir toutes les valeurs du tableau valeurs.Ensuite si dans notre tableau d, sur la ligne i et la colonneindice nous avons la valeur qui se situe à l'indice vdu tableau valeurs, on va affecter la valeur1 dans notre tableau b, puisque la modalité v seraprise pour l'observation i.Et la fonction binarisation retourne ce tableau b.Donc on peut ensuite appeler cette fonction sur nosattributs à binariser.Et enfin à l'aide de la fonction concatenate de la librairie NumPy,on va reconstruire notre tableau avec à la fois les attributsque l'on vient de binariser et les attributs qui étaientdéjà numériques dans notre tableau initial.La dernière ligne consiste juste à changerle type de la variable de classe pour qu'elle soit de type int.Donc si on exécute ici, on peut maintenant visualiserle contenu de nos données.Elles sont toutes de type numérique et on est prêt.On peut maintenant apprendre un modèle de machinelearning sur ces données.

## Gérer des données déséquilibrées

Une fois les données mises en forme,on peut apprendre un modèle de machine learning.Avant de nous lancer dans l'ajustementd'un modèle de forêt aléatoire,regardons Tout d'abord comment les observations se répartissentsur les classes.Nous voyons que le jeu de données est très déséquilibré puisqu'ona beaucoup d'observations, 2414 observations pourla classe 0 et seulement 170 pour la classe 1.Ainsi si on construit un classifier qui prédittoujours la classe 0, celui-ci aura une valeurd'ajustement égale à 93 %.Donc dans 93 % des cas, il prédira correctement la classe.Cependant un tel classifier n'a aucun intérêt,surtout dans le contexte de ce jeu de données où on chercheà prédire des événements dangereux dans la classe 1.Avant de construire notre modèle de machine learning,nous allons d'abord séparer le jeu de données en deux ensembles,un ensemble X\_train sur lequel nous allons apprendre le modèleet un ensemble X\_test sur lequel nous allons tester le modèle,l'ensemble de tests comportant un tiers des données.Nous allons ajuster un modèle de forêt aléatoireavec la fonction scikit-learn RandomForestClassifieret on va demander à construire 10 arbres.On va ajuster le modèle sur l'ensemble X\_trainet prédire les valeurs correspondant à X\_test.On peut mesurer l'ajustement de la variable Y\_predit àla valeur de la variable de classe à l'aide de la fonction accuracy,qui nous indique ici qu'on a une valeur d'accuracy qui est un petitpeu inférieure à celle que l'on avait précédemment lorsque l'onprédisait toujours la classe 0.On peut regarder plus en détail la matrice de confusion ici,pour observer que ce classifier réussit à classer seulementtrois observations de la classe incorrectement.Ce phénomène où le classifier arrive très mal à classerles observations d'une classe est mieux quantifié avecune mesure de type f1.Donc c'est ce qu'on fait ici, on prend la mesure f1\_score,c'est une mesure qui varie entre 0 et 1 et là on s'aperçoitqu'on a une valeur de 0.Alors qu'est-ce que l'on peut faire pour remédierà ce problème d'avoir un modèle qui est surajusté pour apprendrela classe majoritaire, alors que nous on estplus intéressé par la classe minoritaire ?Une technique va consister à construire un sous-échantillonde la classe majoritaire et d'utiliser ce sous-échantillondans l'apprentissage, pour éviter d'avoir ce phénomènede classe déséquilibrée.Donc c'est ce qu'on fait ici.On va utiliser la fonction resample,que l'on va appeler sur les observationsde la classe majoritaire.On va faire un tirage sans remise et on va tirer 170 observations,pour que l'on ait autant d'observationsdans la classe 1 que dans la classe 0.Une fois qu'on a créé cet échantillon,on va le reconcaténer avec les observations de la classeet on va construire la variable de classe correspondante,donc qui contientComme précédemment, on va découper notre jeude données en un jeu\_train et un jeu\_test,ajuster notre modèle sur le jeu\_trainet utiliser notre modèle pour prédire la classede l'ensemble test.On peut maintenant remesurer le f1\_score sur ces donnéeset ce modèle-là.Donc on a un score qui est nettement plus élevé,avec une valeur de 0.alors qu'on était à une valeur de 0.précédemment. Si on regarde plus précisément la matrice de confusion,on s'aperçoit qu'on arrive bien à classer les élémentsdes deux classes.Alors ce score f1 mesuré avec le sous-échantillonnage de la classemajoritaire n'est sans doute pas non plus complètement exact,puisque lorsque l'on va appliquer notre modèlesur l'ensemble des données, on vaavoir beaucoup plus d'observations de la classe majoritaire.Alors pour avoir une mesure plus juste,plus fiable, on va repondérer les observations de la classemajoritaire en fonction du poids qu'on a utilisépour les sous-échantillonnages.C'est-à-dire qu'on va avoir un poids de 2414 divisé par 170.On va construire un vecteur de poids,c'est-à-dire qu'on va affecter ce poids-là à toutesles observations de la classe 0, c'est à dire (1- les valeurs de la classe 1)et on va associer un poids de 1 aux observations de la classe 1.Maintenant, si on calcule le f1\_score avec ce poids,on a une valeur qui est égale à 0.point, donc qui est moins optimiste quecelle que l'on avait précédemment, mais qui va correspondre mieuxà la réalité de nos données.On peut également utiliser ce vecteur de poids dansla matrice de confusion, pour voir un petit peu commentça se représente dans notre matrice de confusion.Donc ici on aurait0 qui seraient bien classées et 38 observations dela classe 1 qui seraient également bien classées.C'est nettement plus que ce que l'on avait dans notre premier modèlesans échantillonnage qui est ici,où on avait seulementla classe 1 qui étaient bien classées.

# Conclusion

## Conclure ce cours sur le machine learning

Merci d'avoir suivi ce cours sur le Machine Learning.Nous avons vu que le Machine Learning a montré ces dernièresannées qu'il était une technique extrêmement performantepour réaliser des tâches cognitives diverses commereconnaître des objets dans des images,traduire automatiquement des textes,jouer au jeu d'échecs ou au Go, et plus généralementclasser des observations à partir d'exemples.Le potentiel de ces techniques est très grand,puisqu'elles permettent d'automatiserla réalisation de tâches complexes qui jusque-lànécessitaient l'expertise humaine.Si ces techniques d'Intelligence Artificielle sontextrêmement performantes, leur mise en œuvrenécessite de maîtriser un ensemble de concepts.Nous avons vu les différentes étapes permettant la miseen œuvre du Machine Learning sur des données réelles :Tout d'abord le recueil et la mise en forme des données,comment sélectionner, prétraiter et transformer les données.Puis nous avons vu les intuitions mathématiques qui permettentde comprendre ce qu'est un modèle d'apprentissage et quels sontles paramètres qui sont appris à partir des données.Différents modèles de classificationont été détaillés, quatre modèles fondamentauxsimples et efficaces pour traiter des données de taille raisonnable,et les réseaux de neurones qui permettent de traiterdes données plus structurées telles que les imagesou du texte. Si les données et les modèles sont les ingrédientsclés du machine learning, l'apprentissage des paramètresd'un modèle à partir des données requiert une méthodologie.Nous avons vu qu'il existe différentes mesures permettantde mesurer la qualité d'un modèle, et des techniques permettantde s'assurer que le modèle est à la fois bien ajusté aux données,tout en n'étant pas trop spécifique à celles-ci afinde pouvoir prédire la classe de nouvelles observations.J'espère vous avoir donné une vision plus claire dece qu'est le Machine Learning, tout en aiguisantvotre curiosité pour aller plus loin .