Etat de l’art en Intelligence Artificielle et Machine Learning appliqué à l’analyse statistique

12 FEVRIER 2021

Université Bretagne Sud
Crée par : Groupe CMI (Licence, Master)

Introduction à l'IA et au ML

Ce rapport a pour but de faire la synthèse sur les pratiques et méthodes utilisées dans le milieu de l'intelligence artificielle et du machine learning appliqué à l'analyse statistique. Nous proposerons d'abord une définition de l'IA et du ML ensuite nous étudierons les algorithmes les plus utilisés dans le milieu professionnel à l'aide des études de benchmark fournies par les entreprises, enfin nous parlerons des méthodes d'évaluation des modèles statistiques les plus courantes.

L'intelligence artificielle

L'Intelligence artificielle est définie comme étant « l'ensemble des techniques et théories mises en œuvre en vue de réaliser des machines capables de simuler l'intelligence ». Elle correspond, entre autre à un ensemble de concepts et de technologies simulant la cognition humaine plus qu'à une discipline autonome constituée.

Le machine learning

L'apprentissage automatique ou apprentissage statistique (machine learning en anglais), est un champ d'étude de l'intelligence artificielle qui concerne la conception, l'analyse, le développement et l'implémentation de méthodes permettant à une machine (au sens large) d'évoluer par un processus basé sur l'apprentissage sur un ou plusieurs types de données plutôt que par des algorithmes classiques.

Le machine learning peut être décliné en plusieurs types d'apprentissage :

1) L'apprentissage supervisé
2) L'apprentissage semi-supervisé
3) L'apprentissage renforcé

Chacun de ses types d'apprentissage ont évolué avec les besoins croissant en automatisation du traitement de données massives et des avancées technologiques (computer vision, natural language processing, emotion processing etc.)

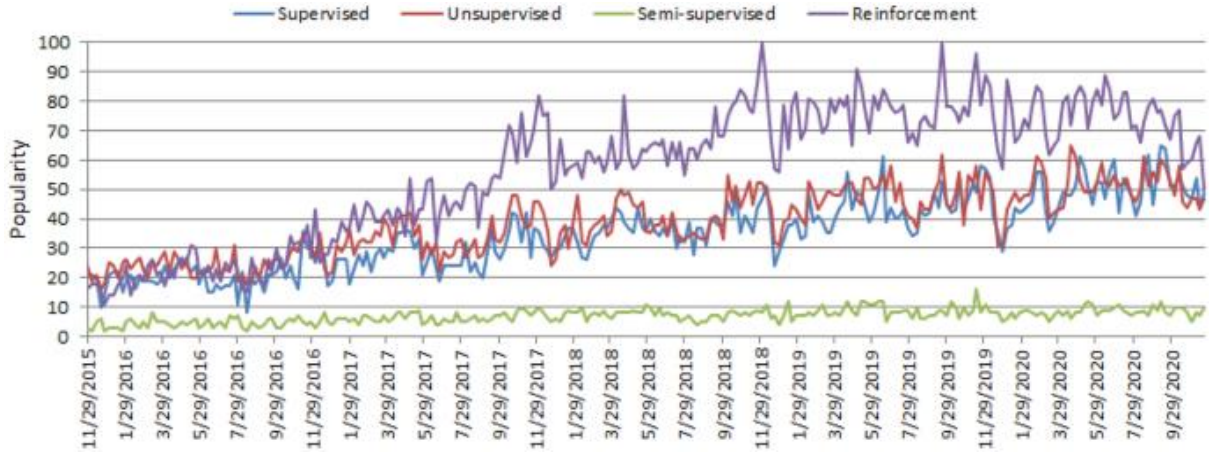

Figure 1 Evolution de l'utilisation des méthodes de ML au cours du temps

L'apprentissage supervisé consiste à développer une fonction qui associe des données d'entrée à des étiquettes (labels) cibles. Il est fourni avec un ensemble de données d'apprentissage labélisées par des

méthodes de séparation des donnés (cross-validation) ainsi qu'un ensemble de données de test. Lorsque les variables cibles sont des valeurs réelles continues, les tâches d'apprentissage supervisé sont connues comme des problèmes de régression, et lorsque les variables cibles sont des variables catégoriques, les tâches sont connues comme des problèmes de classification.
Les algorithmes d'apprentissage supervisé courants comprennent :

1) La régression linéaire
2) La méthode Naive Bayes (NB)
3) L'analyse discriminante linéaire (LDA)
4) La régression logistique (RL) -> Simon
5) L'arbre de décision
6) La forêt aléatoire (RF) -> Titouan
7) La machine à vecteurs de support (SVM)
8) Les K-voisins les plus proches (KNN) -> Ewen
9) Le réseau neuronal artificiel (ANN) -> Pierre + CNN + RN

Les algorithmes et modèles de machine learning

L'apprentissage automatique implique l'utilisation d'algorithmes et de modèles de machine learning. Une erreur commune à la plupart des personnes débutant dans ce domaine est le fait de confondre la notion « algorithme d'apprentissage automatique » et celle de « modèle d'apprentissage automatique ». Ce point permet donc de lever la différence entre un algorithme et un modèle de machine learning avant de les étudier.

Qu'est-ce qu'un algorithme de machine learning ?

Dans le domaine de l'apprentissage automatique, un « algorithme » est une procédure qui est exécutée sur des données pour créer un « modèle » d'apprentissage automatique. Les algorithmes simulent les fonctions principales de la cognition humaine comme la reconnaissance de formes (pattern recognition), ils « apprennent » à partir de données, ou sont « adaptés » à un ensemble de données.

En tant que tels, les algorithmes de machine learning ont un certain nombre de propriétés qui permet de les aborder de différentes manières :

- Les algorithmes de ML peuvent être décrits à l'aide de mathématiques et de pseudo-code.
- L'efficacité des algorithmes de ML peut être analysée et décrite.
- Les algorithmes de ML peuvent être mis en œuvre avec la plupart des langages de programmation modernes.

Qu'est-ce qu'un modèle de machine learning ?

Un « modèle » en ML est le résultat d'un algorithme d'apprentissage automatique exécuté des données.
Un modèle représente ce qui à été appris par un algorithme d'apprentissage automatique. Le modèle est la « chose » qui est sauvegardée après l'exécution d'un algorithme de ML sur des données d'apprentissage et représente les règles, les chiffres et toute autre structure de données spécifiques à l'algorithme nécessaire pour faire des prédictions.

Voici quelques exemples :

- L'algorithme de régression linéaire aboutit à un modèle d'un vecteur de coefficients avec des valeurs spécifiques
- L'algorithme de l'arbre de décision est composé d'un arbre d'instructions « si-alors » avec des valeurs menant à une décision.
- Les algorithmes de réseau neuronaux donnent un modèle composé d'une structure graphique avec des vecteurs ou des matrices de poids menant à une décision.

La meilleure approche pour comprendre un modèle de ML consiste à considérer le modèle comme un « programme ». Ce programme, comprends à la fois des données et une procédure d'utilisation des données pour les classifier.

Analyse des algorithmes de ML pour la classification

Cette section a pour but d'analyser les utilisations et le fonctionnement des algorithmes de machine learning les plus utilisés dans l'analyse statistique en milieu professionnel.

Les réseaux de neurones artificiels

Concept des neurones artificiels

Les réseaux neuronaux reflètent le comportement du cerveau humain, permettant aux programmes informatiques de reconnaître des modèles et de résoudre des problèmes courants dans les domaines de l'IA, de l'apprentissage automatique (ML) et de l'apprentissage profond (DL).

Les réseaux neuronaux artificiels (ANN) sont constitués de couches de nœuds, contenant une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. Chaque nœud, ou neurone artificiel, se connecte à un autre et possède un poids et un seuil associés. Si la sortie d'un nœud individuel est supérieure à la valeur seuil spécifiée, ce nœud est activé, envoyant des données à la couche suivante du réseau. Dans le cas contraire, aucune donnée n'est transmise à la couche suivante du réseau.

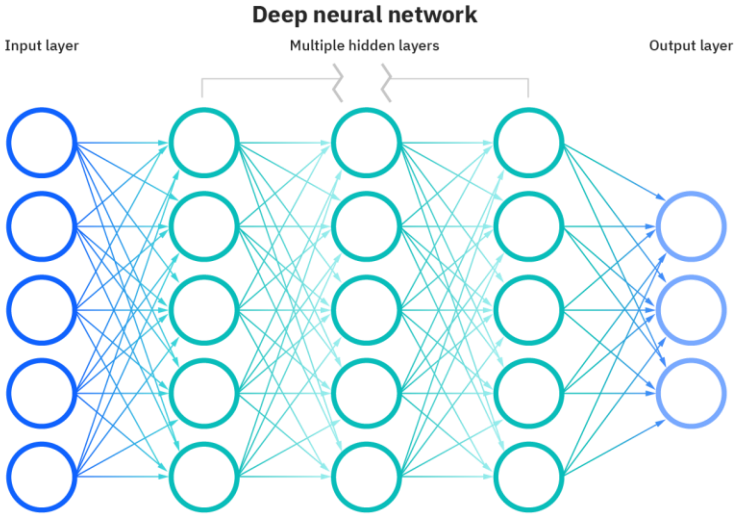

Figure 2: Illustration d'un réseau de neurones artificiel

Les réseaux neuronaux reposent sur des données d'apprentissage pour apprendre et améliorer leur précision au fil du temps. Cependant, une fois que ces algorithmes d'apprentissage sont réglés avec précision, ils constituent des outils puissants en informatique et en intelligence artificielle, nous permettant de classer et de regrouper des données à une vitesse élevée. Les tâches de reconnaissance vocale ou de reconnaissance d'images peuvent prendre quelques minutes au lieu de quelques heures, par rapport à l'identification manuelle par des experts humains. L'un des réseaux neuronaux les plus connus est l'algorithme de recherche de Google.

Types de réseaux de neurones

Les réseaux neuronaux peuvent être classés en différents types, qui sont utilisés à des fins différentes. Bien qu'il ne s'agisse pas d'une liste exhaustive, les types ci-dessous sont représentatifs des types les plus courants de réseaux neuronaux que vous rencontrerez dans les cas d'utilisation les plus courants :
Le perceptron est le plus ancien réseau neuronal, créé par Frank Rosenblatt en 1958. Il possède un seul neurone et constitue la forme la plus simple de réseau neuronal.

Les réseaux neuronaux à action directe, ou perceptrons multicouches (MLP), sont ceux sur lesquels nous nous sommes principalement concentrés dans cet article. Ils se composent d'une couche d'entrée, d'une ou plusieurs couches cachées et d'une couche de sortie. Bien que ces réseaux neuronaux soient aussi communément appelés MLP, il est important de noter qu'ils sont en fait constitués de neurones sigmoïdes, et non de perceptrons, car la plupart des problèmes du monde réel sont non linéaires. Les données sont généralement introduites dans ces modèles pour les former, et ils constituent la base de la vision par ordinateur, du traitement du langage naturel et d'autres réseaux neuronaux.

Les réseaux neuronaux convolutifs (CNN) sont similaires aux réseaux à anticipation, mais ils sont généralement utilisés pour la reconnaissance des images, la reconnaissance des formes et/ou la vision par ordinateur. Ces réseaux exploitent les principes de l'algèbre linéaire, en particulier la multiplication des matrices, pour identifier des motifs dans une image.
Les réseaux neuronaux récurrents (RNN) sont identifiés par leurs boucles de rétroaction. Ces algorithmes d'apprentissage sont principalement exploités lors de l'utilisation de données de séries temporelles pour faire des prédictions sur des résultats futurs, comme les prédictions boursières ou les prévisions de ventes.

Cas d’utilisation en entreprise

Depuis des décennies, IBM est un pionnier dans le développement des technologies d'IA et des réseaux neuronaux, mis en évidence par le développement et l'évolution d'IBM Watson. Watson est désormais une solution de confiance pour les entreprises qui cherchent à appliquer des techniques avancées de traitement du langage naturel et d'apprentissage profond à leurs systèmes en utilisant une approche par paliers éprouvée pour l'adoption et la mise en œuvre de l'IA.

Watson utilise le cadre Apache Unstructured Information Management Architecture (UIMA) et le logiciel DeepQA d'IBM pour mettre à la disposition des applications de puissantes capacités d'apprentissage profond. Grâce à des outils comme IBM Watson Studio, votre entreprise peut mettre en production des projets d'IA open source de manière transparente, tout en déployant et en exécutant des modèles sur n'importe quel cloud.

La méthode Naive Bayes

Concept de la méthode

La méthode de Naïve Bayes est une méthode utilisant un classificateur Bayésien naïf. Comme son nom l'indique, ce classificateur repose sur le théorème suivant :

$$P(VC \mid VE) = (P(VE \mid VC) * P(VC)) / P(VE)$$

Avec VC étant la variable cible et VE étant les variables explicatives

Ce classificateur utilise une hypothèse naïve étant que chacune des variables explicatives sont supposées indépendantes. Bien que cette hypothèse soit rarement vérifiée, les estimations obtenus grâce au classificateur Bayésien naïf n'en reste pas moins très bonnes. Pour l'utiliser, il suffit de connaître l'estimation des probabilités conditionnelles et les probabilités à posteriori.

Avantage de la méthode

Le principal avantage de ce classificateur est sa vitesse d'apprentissage et ses prédictions. En effet, admettre que les variables explicatives sont toutes indépendantes engendre de forte réduction des calculs à réaliser. Pour des jeux de données de faible taille, ce classificateur se montre très efficace Cette qualité amène logiquement à l'utilisation de ce classificateur en combinaison avec d'autres algorithmes (Arbres de décision).
Une autre qualité de ce classificateur est que celui-ci est facilement incrémentale. Dans son cas, il peut être rapidement mis à jour sans nécessiter de refaire chaque calcul. Il suffit de mettre à jour les probabilités conditionnelles univariées des variables.

Classificateur Bayésien Naïf en ligne

Comme vu précédemment, l'idée de départ de ce classificateur est de calculer la probabilité conditionnelle :

$$P(VC \mid VE) = (P(VE \mid VC) * P(VC)) / P(VE)$$

Cependant, la probabilité conditionnelle P(VE | VC) n'est pas facilement estimable, il faudra faire alors appel à la version naïve de ce classificateur :

$$PNB(VC \mid VE) = P(VC)*sum(P(VE \mid VC) / P(VE)$$

Pour calculer ce classificateur, il suffit d'avoir en paramètre P(VC). Par contre, la probabilité P(VE | VC) est difficile à calculer car il faudra sauvegarder chaque instance.

Classificateur Bayésien Naïf Moyenné-en-ligne

L'amélioration du classificateur bayésien peut avoir lieu de deux principales méthodes. La première consiste à sélectionner des variables et la seconde consiste à pondérer les variables. La sélection des variables consiste à ne sélectionner que certaines variables. On utilisera alors le terme SNB « Selective Naïve Bayes ». Mais il serait trop simple de sélectionner aléatoirement certaines variables. La solution serait de supprimer celles qui ne sont pas informatives (dont la loi a priori ne serait pas informative). Pondérer chaque variable permet également d'améliorer les prédictions du classificateur. Cette approche nous amène donc à moyenner chaque variable. Le moyennage consiste à combiner la prédiction de différents classificateurs de façon à améliorer les capacités prédictives.

Le classificateur Bayésien naïf moyenné procède de la même manière que le classificateur Bayésien naïf à la différence près qu'il ajoute une pondération par variable. Cette pondération a pour but de limiter le biais enclenché par l'hypothèse initiale du classificateur qui consiste à admettre que chaque variable explicative est indépendante aux autres.
On peut logiquement deviner que le classificateur Bayésien Naïf moyenné sera plus précis dans ses prédictions que le classificateur initial. Cette différence sera de plus en plus évidente lorsque le jeu de donnée sera grand.

Exemple d'utilisation

Il y a de nombreuses utilisations de la méthode de naïve Bayes afin de réaliser des prédictions sur un échantillon donné. On peut notamment l'utiliser dans plusieurs domaines comme les finances, la médecine, le sport, les prédictions météorologiques, …
Dans le cas d'une étude réalisée par un étudiant en Master 1 Mathématiques et Applications Spécialité Statistique de l'université de Strasbourg, L'algorithme Naïve Bayes est utilisé pour prédire l'apparition de séisme en fonction de plusieurs paramètres comme la localisation, la magnitude etc. Cette étude fait également appel aux algorithmes SVM et k plus proches voisins.

Les Forêts d'arbres décisionnel

Concept de l'arbre de décision

Un arbre de décision est un outil d'aide à la décision représentant un ensemble de choix sous la forme graphique d'un arbre. Il s'agit d'un algorithme classique d'apprentissage supervisé. L'objectif d'un arbre de décision est de construire pas à pas des « segments » de population les plus « pure » possible. Cet algorithme peut être utilisé pour des problèmes de régression et de classification grâce à des variables explicatives qui peuvent être quantitatives et/ou qualitatives.

Choix de la variable de segmentation

Afin de décider comment segmenter la population initiale, on teste toutes les variables et on choisit la variable X qui présente la plus forte liaison avec Y. Afin de quantifier cette liaison on utilise la quantité du χ^2 calculée sur le tableau de contingence (croisement de Y avec Xi). Les prochaines divisions tiennent compte de la nouvelle population afin de renouveler la segmentation. Cette segmentation en cascade forme un arbre de décision avec des segments de plus en plus pure.
Dans le cas d'une variable quantitative, toutes les valeurs seront testées et la segmentation sera effectuée pour la valeur séparant au mieux la population identifiée précédemment.

Arrêt de la segmentation

L'objectif de l'arrêt de la segmentation est de conserver une capacité de généralisation du modèle. En effet si l'on poursuit la segmentation même sur de très faible effectif l'on risque que notre modèle soit sur-ajuster aux données. Afin d'éviter cela, l'on peut définir des règles d'élagage sur les effectifs, sur la significativité des segmentations ou sur l'homogénéité des segments.

Règles de décision

Un segment terminal est affecté à la classe à (k de la variable Y) la plus représentée. Mais cette règle est bonne si la variable à expliquer Y présente des modalités équilibrées en proportion. Dans le cas contraire, si les modalités

sont déséquilibrées, il est plus judicieux d'affecter le segment terminal à la modalité surreprésentée par rapport à la distribution d'origine

Exemple d'utilisation

Cet exemple illustre le cas où l'on cherche à prédire si des sportifs vont disputer ou non un match en fonction de données météorologiques.

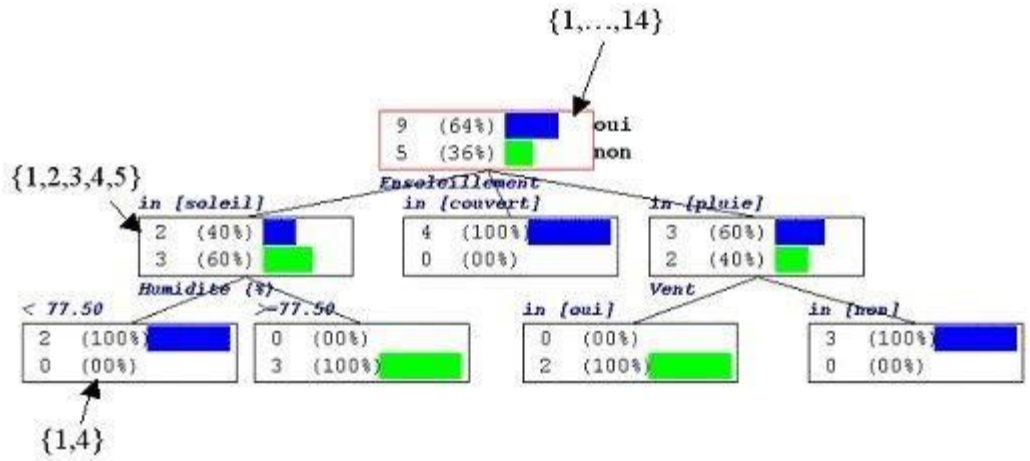

Figure 3 : Illustration d'un arbre de décision

Avantages et inconvénients

L'arbre de décision possède de nombreux points fort :

- Il ne nécessite pas d'hypothèse sur les données
- Les variables explicatives peuvent être de nature qualitative et quantitative
- Il est idéal pour trouver les seuils de coupure optimaux pour les variables continues explicatives et il est robuste aux données aberrantes.
- De plus ce modèle est de type « boîte blanche » en effet il est simple d'expliquer les sorties du modèle, notamment grâce au côté visuelle de la méthode.

Également il est facile à implémenter grâce à de simple « if » « else if ». A l'inverse, ce modèle a certains défauts :

- Il est nécessaire d'avoir un effectif important car sur un petit échantillon le modèle peut s'avérer instable.
- Si une variable à plus de 2 groupes la classification peut être difficile.
- Il faut aussi faire attention à ce qu'une variable explicative n'en cache pas une autre.

Random Forest

La Random Forest est une méthode de Machine Learning qui repose sur l'utilisation des arbres de décisions. La méthode consiste à créer un grand nombre d'échantillons d'apprentissage N grâce à une méthode de tirage avec remise (Bootstrap). Ensuite, sur ces N échantillons on va construire N arbres de décisions. Chaque arbre affecte une réponse et par un système de « vote », la réponse est définie grâce à la majorité des arbres. Il est également possible de pondérer le vote d'un arbre en fonction des performances de ces prédictions individuelles.

Le principal défaut de cette méthode est qu'il est de type « boite noire » en effet, il n'est plus possible d'expliquer aussi facilement les résultats de ce modèle qu'avec un arbre de décision seule. De plus, entraîner un modèle de random forest est bien plus exigeant en termes de puissance de calcul.

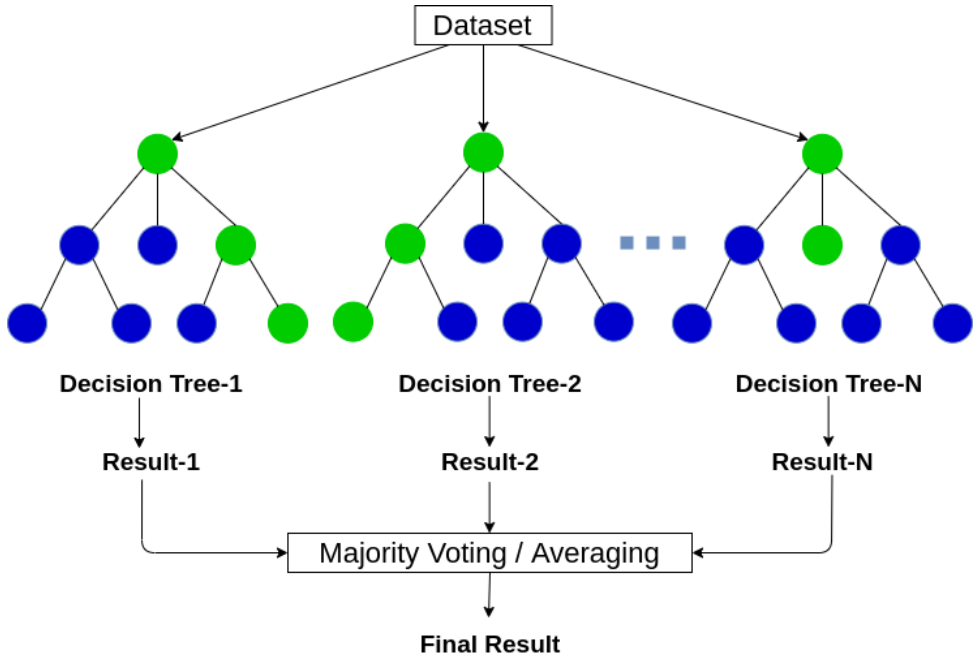

Figure 4: illustration d'une Forêt d'arbres aléatoires