DATA SCIENCE
FONDAMENTAUX

TABLE DES MATIERES

INTRODUCTION

La data science est l'art de traduire des problèmes industriels, sociaux, scientifiques, ou de toute autre nature, en problèmes de modélisation quantitative, pouvant être résolus par des algorithmes de traitement de données.

Cela passe par une réflexion structurée, devant faire en sorte que se rencontrent problèmes humains, outils techniques/informatiques et méthodes statistiques/algorithmiques.

Chaque projet de data science est une petite aventure, qui nécessite de partir d'un problème opérationnel souvent flou, à une réponse formelle et précise, qui aura des conséquences réelles sur le quotidien d'un nombre plus ou moins important de personnes.

I. DEFINITIONS

La data science est une discipline encore jeune au carrefour de plusieurs domaines.

Il n'existe pas une et unique définition de la data science dans la communauté scientifique et professionnelle.

- Cleveland (2001) la définit comme une extension de l'analyse de données vers d'autres champs techniques, tels que l'informatique et l'expertise métier, qui sont aujourd'hui nécessaires pour récolter, manipuler et exploiter les données disponibles dans nos

environnements professionnels et personnels. Le domaine de l'analyse de données lui-même doit être nuancé, car il englobe des approches différentes.

- Breiman (2001) propose notamment une distinction entre deux cultures : l'analyse statistique classique, dans la lignée des grands penseurs de cette discipline (Gauss, Pearson et tant d'autres), et l'analyse algorithmique. Toujours selon Breiman, la culture classique se base sur la présomption que les données analysées sont générées par des processus stochastiques que l'on cherche à décrire. La culture algorithmique, s'intéresse aux processus sous-jacents aux données se concentrant à en extraire de l'information avec des modèles qui peuvent être des **boîtes noires**.

II. ROLE DU DATA SCIENTIST

Un data scientist doit savoir **naviguer** entre ces différentes disciplines : **statistique**, **algorithmie**, **informatique**, **sans a priori théorique**.

Ce qui prime, c'est sa faculté à **trouver une réponse adéquate à un problème fonctionnel donné**. En ce sens, sa principale qualité sera sa capacité à comprendre son terrain d'action et à trouver la meilleure solution parmi les nombreux choix techniques (plateformes informatiques, logiciels, etc.) et théoriques (méthodes statistiques et algorithmiques) possibles, compte tenu de contraintes de temps et de budget.

Pour cela, le data scientist doit :

1. Comprendre et analyser le besoin métier
2. Collecter les données
3. Analyser les données
4. Choisir le(s) modèle(s) adéquat(s)
5. Evaluer des modèles/algorithmes

Selon la littérature, le data scientist peut répondre à cinq types de questions :

- Quelle quantité ? (Régression)
- Est-ce A ou B ? (Classification)
- Comment les données sont-elles organisées ? (clustering)
- Est-ce étrange ? (Détection d'anomalie)
- Que devons-nous faire ensuite ? (Apprentissage par renforcement)

III. CYCLE DE VIE D'UN PROJET DATA SCIENCE

Le processus de data science commence

Chaque étape décrite ci-dessous ne peut être analysée isolement. Un projet Data Science est un processus itératif et des allers et retours entre les différentes étapes sont la norme. Par exemple c'est en nettoyant les données que l'on peut mettre en lumière des hypothèses et réaliser que d'autres données peuvent être pertinentes à collecter.

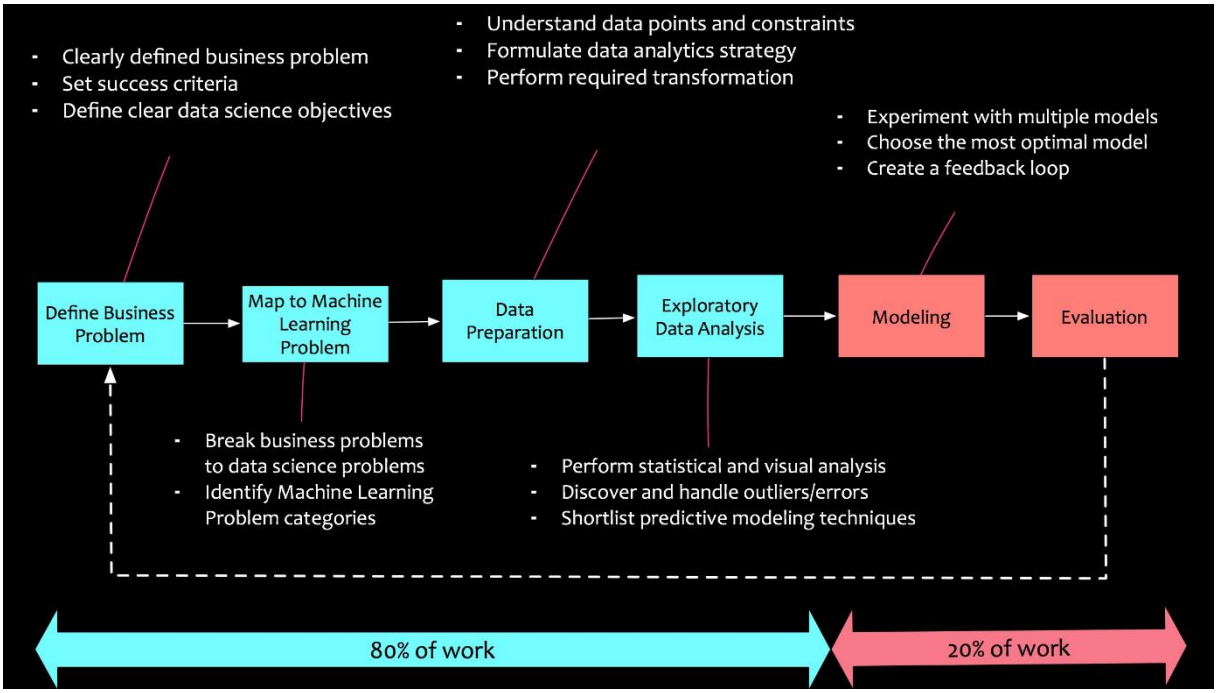


Figure 1 : Cycle de vie d'une projet data science - source : [Data Science Simplified Part I : Principles and Process – Pradeep Menon]

1. COMPREHENSION DU PROBLEME METIER

Le data scientist a pour objectif de s'assurer que l'ensemble des décisions prises dans une entreprise repose sur des données de qualité. En d'autres termes, il a pour ambition de mettre la donnée au cœur des décisions !

Avant de se lancer dans un projet de Data Science, il est fondamental de comprendre l'environnement dans lequel ce professionnel va intervenir (Quelle industrie ? Quel service ? Quels enjeux opérationnels ? Quelle réglementation en vigueur ? etc.) puis de définir la problématique à résoudre.

Lors de cette étape le data scientist doit échanger avec les responsables métiers pour comprendre la problématique à résoudre et identifier les variables à prédire : Prévisions de ventes (régression), profil client (clustering), « Qu'est-ce qui attire le plus les clients : un coupon de 5 € ou une remise de 25 % ? » (Classification), etc.

La compréhension des données et la manière de les exploiter en se posant de bonnes questions est un processus autant essentiel que délicat. Ce processus est, selon nous, plus un art qu'une science. Se poser les « bonnes questions » et « faire le tri entre l'essentiel et l'accessoire » nécessitent beaucoup d'expérience. Une manière d'acquérir cette expérience est d'échanger avec des experts métiers, des data scientists chevronnés et de développer sa sensibilité métier à travers des lectures ou tout autre support.

2. ACQUISITION ET PREPARATION DE DONNEES

Une fois les objectifs du projet bien définis, il est alors temps de collecter les données.

Malheureusement, il est très rare (voire naïf) de penser que l'ensemble des données est stocké à un même endroit et servi sur un plateau d'argent. La plupart du temps, la collecte de données est consommatrice de temps et d'énergie. Ainsi, le data scientist doit avoir une vision claire et exhaustive des données à collecter, identifier les sources où obtenir ces données, savoir y accéder et les stocker.

COLLECTE DE DONNEES

La data science est une démarche empirique qui se base sur des données pour apporter une réponse à des problèmes. Donc, avant toute chose, il faut avoir des données.

PRINCIPAUX TYPES DE DONNEES

On distingue généralement les données quantitatives des données qualitatives.

Les **données quantitatives** sont des valeurs qui décrivent une quantité mesurable, sous la forme de nombres sur lesquels on peut faire des calculs (moyenne, etc.) et des comparaisons (égalité/différence, infériorité/supériorité, etc.). Elles répondent typiquement à des questions du type « combien ». On fait parfois la différence entre :

- Les **données quantitatives continues**, qui peuvent prendre n'importe quelle valeur dans un ensemble de valeurs : la température, le PIB, le taux de chômage, en sont des exemples ;
- Les **données quantitatives discrètes**, qui ne peuvent prendre qu'un nombre limité de valeurs dans un ensemble de valeurs.
  Par exemple : le nombre d'enfants par famille, le nombre de pièces d'un logement, etc.

Les **données qualitatives** décrivent quant à elles des qualités ou des caractéristiques. Elles répondent à des questions de la forme « quel type » ou « quelle catégorie ». Ces valeurs ne sont plus des nombres, mais un ensemble de modalités. On ne peut pas faire de calcul sur ces valeurs, même dans l'éventualité où elles prendraient l'apparence d'une série numérique. Elles peuvent toutefois être comparées entre elles et éventuellement triées. On distingue :

- **Les données qualitatives nominales** (ou catégorielles), dont les modalités ne peuvent être ordonnées. Par exemple : la couleur des yeux (bleu, vert, marron, etc.), le sexe (homme, femme), la région d'appartenance (68, 38, etc.)
- **Les données qualitatives ordinales**, dont les modalités sont ordonnées selon un ordre « logique ».
  Par exemple : les tailles de vêtements (S, M, L, XL), le degré d'accord à un test d'opinion (fortement d'accord, d'accord, pas d'accord, fortement pas d'accord).

PROVENANCE DES DONNEES

La réponse à cette question n'est pas bien difficile : elles viennent de partout !

Il existe trois modes de collecte de données :

- Les **open data**, qui correspondent à la mise à disposition gratuite de données de la société civile, sur des sites tels que : www.data.gouv
- Les **open API**, qui sont des technologies permettant d'accéder à des données sur Internet. Elles vous permettent de récupérer par exemple des données mises à disposition par Google, Twitter, etc. Pour en savoir plus sur les API disponibles, consultez par exemple l'annuaire http://www.programmableweb.com.
- Le Web en tant que tel est lui aussi directement source de données. Pour cela, il faut un minimum d'expertise en programmation pour être capable de faire du **web scraping**, qui consiste à récupérer des données directement à partir des pages des sites Internet.

NETTOYAGE DE DONNEES

Une fois la collecte de données achevée, le data scientist passe à l'étape la plus chronophage du projet (50 à 80% du temps) : le nettoyage et la mise en forme des données.

Outre la volumétrie des données, la dimension chronophage de cette étape s'explique notamment par de nombreux allers - retours entre le data scientist et le métier. En effet la bonne compréhension de l'environnement étudié est primordiale afin de ne pas omettre certaines informations et ainsi ne pas biaiser l'analyse des données.

Les données provenant de différentes sources peuvent avoir des formats différents (csv, json, xml, etc.) et contenir des anomalies ou des valeurs incorrectes. Les problèmes de qualité les plus fréquents sont les suivants :

LES DONNEES ERRONEES

Elles proviennent le plus souvent d'erreurs de saisie ou d'incompatibilités entre la source de données et la base.

LES DONNEES INCOMPLETES :

Il est fréquent que les utilisateurs d'une base de données ne renseignent que les champs obligatoires ou ceux qui les concernent dans leur activité. Des autres données, pourtant pertinentes, passent ainsi à la trappe.

LES DONNEES NON NORMEES

Plusieurs utilisateurs renseignent une donnée identique sous des formats différents. Par exemple, un individu de sexe masculin sera renseigné M., Mr ou Monsieur.

LES DONNEES OBSOLETES

Une entreprise a fermé, a déménagé, ou encore Mr X a remplacé Mr Y, etc. et la qualité de la base se détériore.

LES DOUBLONS

Un même contact se retrouve plusieurs fois dans la base. Et le data scientist s'arrache les cheveux pour retrouver la fiche-source et fusionner les données.

Différents outils et méthodes existent pour nettoyer et restructurer cette donnée. Par exemple, dans le cas des données manquantes, le data scientist peut supprimer les données manquantes ou les remplacer par des valeurs artificielles (on parle d'imputation).

Lors de cette phase, le data scientist a majoritairement recours à **pandas** (pour python) ou **Dplyr**/ **Data.table** (pour R). Ces librairies permettent de manipuler les données (filtrer, trier, regrouper, fusionner, pivoter, etc.).

3. ANALYSE DE DONNEES

En possession d'un jeu de données complet, correctement traité et fidèle à l'activité/au problème à résoudre, le data scientist peut débuter l'analyse des données. Il rentre alors dans une phase de travail qui peut s'associer à un exercice de brainstorming et de statistiques descriptives.

L'objectif est de croiser les différentes natures de données et d'établir des liens de corrélation entre ces dernières. Ces liens doivent se matérialiser par la formulation d'hypothèses. Par exemple, dans le cas d'une estimation d'un prix de vente d'un bien immobilier, un jeu de données pertinemment choisi inclura la localisation, la superficie, le rendement locatif, l'âge et la qualité de la construction, les équipements, etc. Une hypothèse pouvant être établie est la relation prix de vente/ localisation.

Cette analyse de données peut être facilitée par la mise en place d'histogrammes et/ ou de courbes de distribution, diagrammes de dispersion permettant de dégager des tendances. Par ailleurs des outils de restitutions tels que Power BI, R, Dash, ou Qlickview peuvent faciliter ce travail de brainstorming via des visualisations interactives.

Il est à noter que Cette phase très descriptive se fait de façon itérative avec l'étape précédente : c'est en nettoyant les données que l'on s'aperçoit des incohérences.

4. MODELISATION

Les algorithmes sont usuellement classés selon deux composantes :

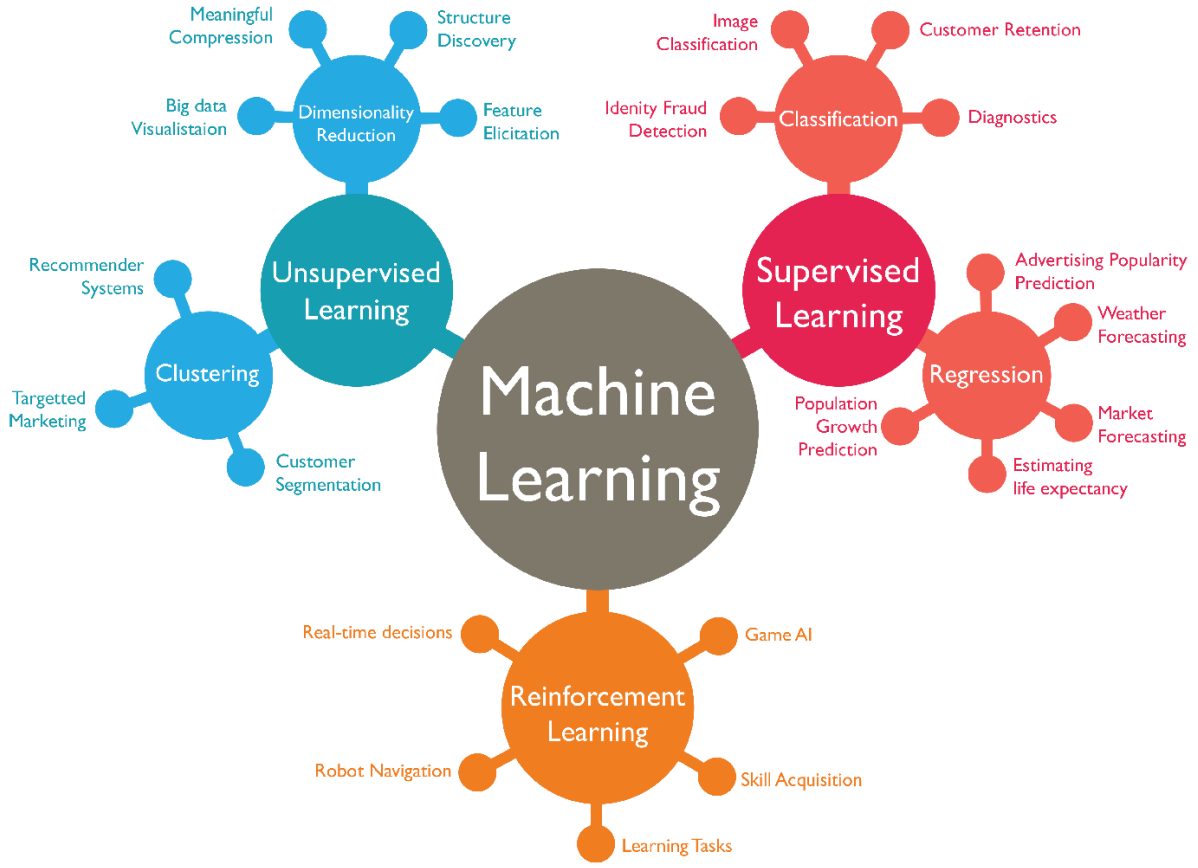


Figure 2 : Classification des algorithme de ML - source :Qu'est ce que le Machine Learning ?

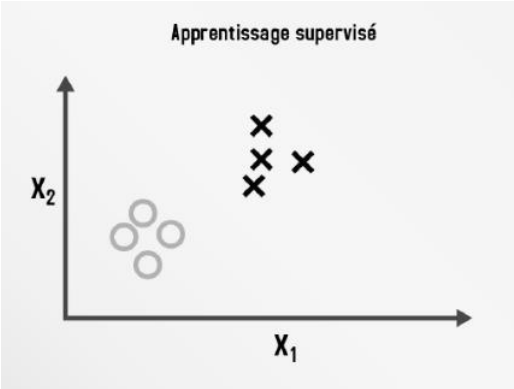
MODE D'APPRENTISSAGE

On distingue les **algorithmes supervisés** et des **algorithmes non supervisés**.

ALGORITHME SUPERVISES

Les algorithmes supervisés extraient de la connaissance à partir d'un ensemble de données contenant des couples entrée-sortie. Ces couples sont déjà « connus », dans le sens où les sorties sont définies a priori.

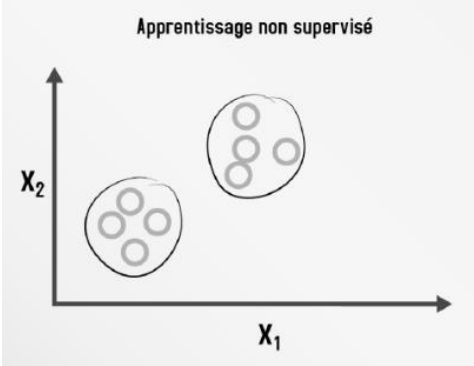
La valeur de sortie peut être une indication fournie par un expert : par exemple, des valeurs de vérité de type OUI/NON ou MALADE/SAIN. Ces algorithmes cherchent à définir une représentation compacte des associations entrée-sortie, par l'intermédiaire d'une fonction de prédiction.



ALGORITHMES NON SUPERVISES

Les algorithmes non supervisés n'intègrent pas la notion d'entrée-sortie. Toutes les données sont équivalentes (on pourrait dire qu'il n'y a que des entrées). Dans ce cas, les algorithmes cherchent à organiser les données en groupes. Chaque groupe doit comprendre des données similaires et les données différentes doivent se retrouver dans des groupes distincts.

Dans ce cas, l'apprentissage ne se fait plus à partir d'une indication qui peut être préalablement fournie par un expert, mais uniquement à partir des fluctuations observables dans les données.



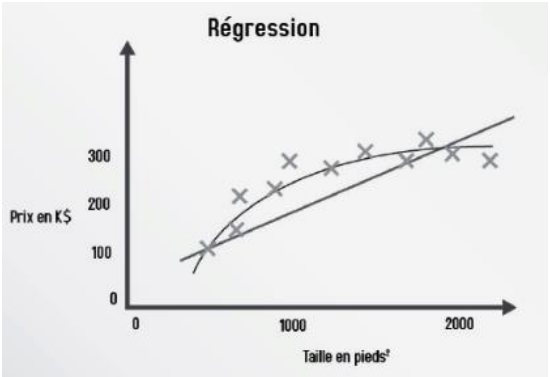
ATTENTION : Généralement, les algorithmes supervisés sont plus performants, mais pour les utiliser, il faut être capable de spécifier une valeur de sortie, et cette information d'expert n'est pas toujours disponible.

TYPE DE PROBLEME A TRAITER

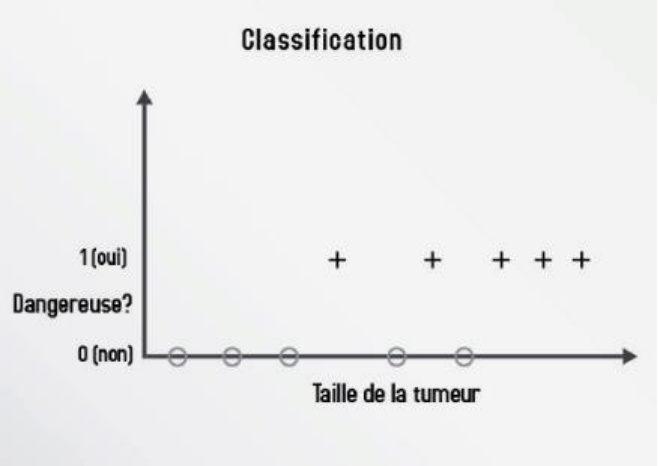
On distingue les algorithmes de **régression** et ceux de **classification**.

La distinction régression/classification se fait au sujet des algorithmes supervisés. Elle distingue deux types de valeurs de sorties qu'on peut chercher à traiter. Dans le cadre d'un problème de régression, Y peut prendre une infinité de valeurs dans l'ensemble continu des réels (noté $Y \in \mathbb{R}$).

Ce peut être des températures, des tailles, des PIB, des taux de chômage, ou tout autre type de mesure n'ayant pas de valeurs finies a priori.



Dans le cadre d'un problème de classification, Y prend un nombre fini k de valeurs ($Y = \{1, \ldots, k\}$). On parle alors d'étiquettes attribuées aux valeurs d'entrée. C'est le cas des valeurs de vérité de type OUI/NON ou MALADE/SAIN évoqués précédemment.



Le tableau ci-dessous récapitule les algorithmes de ML les plus utilisés.

Algorithme	Mode d'apprentissage	Type de problème à traiter
Régression linéaire univariée	Supervisé	Régression
Régression linéaire multivariée	Supervisé	Régression
Régression polynomiale	Supervisé	Régression
Régression régularisée	Supervisé	Régression
Naive Bayes	Supervisé	Classification
Régression logistique	Supervisé	Classification
Clustering hiérarchique	Non supervisé	-
Clustering non hiérarchique	Non supervisé	-
Arbres de décision	Supervisé	Régression ou classification
Random forest	Supervisé	Régression ou classification
Gradient boosting	Supervisé	Régression ou classification
Support Vector Machine	Supervisé	Régression ou classification
Analyse en composantes principales	Non supervisé	-

5. EVALUATION DU MODELE

Il est souvent très facile de construire un modèle qui restitue très bien les données utilisées pour son estimation. Il est néanmoins bien plus difficile de faire en sorte que ce modèle puisse se généraliser, c'est-à-dire qu'il soit capable de prédire de façon satisfaisante de nouvelles observations, non utilisées lors du calcul du modèle. Pour trouver un juste équilibre entre apprentissage du modèle et capacité prédictive, il est indispensable de mettre en place un dispositif qui permette d'évaluer globalement la qualité d'un modèle.

METHODE DE VALIDATION

A partir d'un jeu de données initial, une méthode de validation naïve consiste à diviser les données en trois groupes :

- Un **jeu d'entrainement**, noté $\mathbf{m_{entraînement}}$
- Un **jeu de validation**, noté $\mathbf{m_{validation}}$. Ce jeu va être utilisé pour tester les différents modèles paramétrés sur $m_{entraînement}$.
- Un jeu de test, noté $\mathbf{m_{test}}$. Ce jeu est utilisé à la fin du processus de modélisation pour tester la capacité de généralisation du modèle retenu.

Dans la pratique, on prend souvent 60% des données pour $m_{entraînement}$, 20% pour $m_{validation}$ et 20% pour m_{test}.

VALIDATION CROISEE

Par rapport à la version naïve de la validation présentée juste avant, on a généralement recours à une approche plus exhaustive, visant à ce que les données, à l'exception de celles utilisées pour le test du modèle, soient plusieurs fois utilisées pour faire partie de $m_{entraînement}$ et de $m_{validation}$. On arrive ainsi à mesurer de façon bien plus générale la qualité du modèle. Cette approche est qualifiée de validation croisée. Plusieurs méthodes de validation croisée existent, en voici les principales.

- La méthode **LOOV** (leave-one-out cross-validation) consiste à sortir une observation i de l'ensemble du jeu de données (rappel : à l'exception des données de test) et à calculer le modèle sur les m-1 données restantes. On utilise ce modèle pour prédire i et on calcule l'erreur de prévision. On répète ce processus pour toutes les valeurs de i = 1, …, m. Les m erreurs de prévision peuvent alors être utilisés pour évaluer la performance du modèle en validation croisée ($P_{Xvalidation}$)
- La méthode **LKOV** (leave-k-out cross-validation) fonctionne selon le même principe que la LOOV, sauf que l'on sort non pas une, mais k observations à prédire à chaque étape (donc LOOV est équivalent à LKOV pour k = 1). Le processus est répété de façon à avoir réalisé tous les découpages possibles en données de modélisation/de prévision.
- La méthode **k-fold cross-validation**, les données sont aléatoirement divisées en k sous-échantillons de tailles égales, dont l'un est utilisé pour la prévision et les k-1 restants pour l'estimation du modèle. Contrairement à la LKOV, le processus n'est répété que k fois. À noter que la k-fold cross-validation permet de faire en sorte que la distribution de la variable à prédire soit équivalente dans chacun des sous-échantillons, ce qui est particulièrement intéressant dans le cas des jeux de données déséquilibrées. On parle alors de stratified k-fold cross-validation

METRIQUES DE PERFORMANCE

PROBLEMES DE REGRESSION

Nombreuses sont les mesures disponibles pour évaluer la qualité d'un modèle de régression. Elles se basent toutes sur de savants calculs réalisés à partir de trois grandeurs :

- La valeur observée d'une série à prédire y_i
- La valeur prédite par le modèle pour cette même valeur observée $\hat{y}_i$
- Une prévision naïve de référence, qui est la moyenne de la valeur observée $\bar{y}$[2]

Elles permettent de calculer, pour tout i des m observations :

- **L'erreur de prédiction** du modèle : $y_i - \hat{y}_i$
- **L'erreur de prédiction naïve** : $y_i - \bar{y}$

Tout cela permet de définit des indicateurs de performance du modèle. Les plus connus sont les suivantes :

- L'erreur moyenne absolue (MAE, Mean Absolute Error)
- La racine carrée de la moyenne carrée des erreurs (RMSE, Root Mean Squared Error)
- Le coefficient de détermination R^2

PROBLEMES DE CLASSIFICATION

L'évaluation d'un problème de classification se base sur une matrice de confusion, qui met en regard des données prédites et des données observées.

Tableau 1 : Les termes définis par la matrice de confusion

		Observations		
		+	-	Total
Prédictions	+	Vrais positifs (VP)	Faux positifs (FP)	Total des positifs prédits (VP + FP)
	-	Faux négatifs (FN)	Vrais négatifs (VN)	Total des négatifs prédits (FN + VN)
	Total	Total des vrais positifs observés (VP + FN)	Total des vrais négatifs observés (FP + VN)	Taille totale de l'échantillon (N)

Cette matrice permet de calculer une première mesure très intuitive : le **taux d'erreur**, c'est-à-dire le taux de mauvaise classification défini par (FN+FP)/N.

Tout un ensemble d'autres indicateurs peut être calculé à partir de ces mesures. En général, pour évaluer un modèle, on utilise conjointement les deux indicateurs suivants :

- **Rappel** : taux de vrais positifs VP/(VP+FN)
- **Précision** : VP/(VP+FP)

Le rappel permet de mesurer la proportion de positifs prédits parmi tous les positifs de la population. La précision permet de mesurer la proportion de positifs de la population parmi tous les positifs prédits. Ainsi, un modèle parfait aura un rappel égal à 1 (il prédit la totalité des positifs) et une offre précision égale à 1 (il ne fait aucune erreur : tous les positifs prédits sont des vrais positifs). En pratique, les modèles sont plus ou moins performants suivant ces deux dimensions. Par exemple, on peut avoir un modèle très précis, mais avec un faible rappel : il prédira peu de positifs, mais les positifs prédits seront justes dans la plupart des cas. À l'inverse, un modèle très sensible, mais peu précis va prédire beaucoup de vrais positifs, mais également beaucoup de faux positifs.

Pour comparer plusieurs modèles, on utilise également un indicateur agrégé, composé à partir du rappel et de la précision : le **F_1 score**

$$F_1 = \frac{2(précision * rappel)}{précision + rappel}$$