Chapitre 1. Objectif de la data science

L'objectif de la data science est d'utiliser des méthodes pour extraire des informations d'un jeu de données dans le but de prédire, de classer ou de regrouper des objets ou des individus. Les champs d'application sont vastes, notamment dans les domaines concernant l'environnement et l'agriculture où la data science permet de prédire des quantités ayant une importance stratégique pour des entreprises, des scientifiques ou des décideurs comme, par exemple, les prix des matières premières, la production agricole, l'abondance d'une espèce dans un écosystème ou les émissions de gaz à effet de serre et de polluants. Elle permet également de faire émerger des groupes d'objets ou d'individus partageant certaines caractéristiques, par exemple des groupes d'animaux ou de plantes résistant à une maladie et d'autres groupes plus sensibles.

Un des atouts de la data science réside dans sa capacité à pouvoir valoriser des données de natures diverses, aussi bien quantitatives que qualitatives, discrètes ou continues, provenant d'expérimentations ou d'enquêtes, mais également de satellites, de drones, de sites internet, de réseaux sociaux, etc. Le travail du data scientist consiste à utiliser des algorithmes pour établir des liens entre les données dans le but soit de prédire des variables de réponse à partir de variables explicatives, soit de regrouper des objets ou des individus présentant certaines similarités. Pour cela, le data scientist s'appuie sur des algorithmes d'origines diverses issues de la statistique, de l'apprentissage automatique et de l'informatique. Il entraîne ces algorithmes avec des données pour répondre à la question posée de manière aussi précise que possible.

Les algorithmes disponibles sont nombreux et sont généralement classés en deux grandes catégories : les algorithmes supervisés et non-supervisés. Les algorithmes supervisés mobilisent des données de type « entrée-sortie », c'est-à-dire des données contenant des couples (X, Y) où X correspond à des entrées utilisées pour prédire Y. En général, X est une matrice dont les lignes représentent les individus (ou les objets) observés et dont les colonnes représentent différentes variables caractérisant les individus. Y est une autre matrice incluant le même nombre de lignes mais où les colonnes représentent les variables de réponse que l'on souhaite prédire. Souvent, Y n'inclut qu'une seule variable de réponse et présente alors un vecteur. En pratique, le jeu de données utilisé doit inclure suffisamment d'observations de (X, Y) pour permettre à l'algorithme supervisé d'établir une fonction permettant de prédire Y à partir de X avec une précision acceptable. Ce type d'algorithme peut être utilisé, par exemple, pour prédire un rendement agricole (Y) en fonction de la température et des précipitations (X), pour prédire des émissions de gaz à effet de serre (Y) en fonction des usages des sols (X), ou pour prédire la probabilité d'occurrence d'une maladie d'une culture (Y) en fonction d'images des feuilles de cette culture obtenues avec des smartphones (X). Les algorithmes non-supervisés n'incluent pas de couples (X, Y) mais seulement des données X décrivant les caractéristiques d'un ensemble d'objets ou d'individus. Avec ce type d'algorithme, l'objectif est de définir des groupes d'objets ou d'individus (clusters) ayant certaines similarités ou d'extraire des motifs fréquents (patterns) d'après les valeurs des variables constituant les colonnes de X.

Un data scientist travaille en général par projet et chaque projet est conduit selon plusieurs étapes : (i) définition de la question, (ii) création d'une base de données, (iii) entraînement d'algorithmes pour répondre à la question posée à partir des données disponibles, (iv) comparaison des performances des différents algorithmes entrainés, (v) test de l'algorithme sélectionné, (vi) communication des résultats. L'étape i est cruciale car elle détermine non seulement la nature des données qu'il sera nécessaire de collecter à l'étape ii mais également la nature des algorithmes entrainés à l'étape iii (notamment le choix entre algorithmes supervisés et non supervisés) et les critères utilisés aux étapes iv et v pour évaluer et sélectionner les algorithmes les plus performants.

Une des spécificités du data scientist est qu'il mobilise souvent les trois techniques suivantes: les algorithmes ensemblistes, la validation croisée et la

régularisation. Les algorithmes ensemblistes génèrent un grand nombre de modèles à partir de sous-ensembles de données échantillonnés dans le jeu de données initial global. Ces modèles (souvent des centaines, parfois des milliers) sont ensuite combinés pour répondre à la question posée. Cette idée d'entraînement de plusieurs sous-modèles sur des échantillons issus des données initiales, ensuite mis en commun, est l'une des bases de la data science moderne et est mise en œuvre par des méthodes devenues très populaires comme, par exemple, random forest ou gradient boosting.

La validation croisée joue également un rôle central en data science. Cette technique consiste à découper le jeu de données initial en plusieurs sous-parties. Une des sous-parties est mise de côté, et les données restantes sont utilisées pour ajuster (on dira souvent « entraîner ») un ou plusieurs algorithmes. La sous-partie initialement retirée est ensuite utilisée pour évaluer la qualité prédictive des algorithmes entrainés. Cette démarche, répétée successivement avec toutes les sous-parties du jeu de données, permet d'évaluer et de comparer plusieurs algorithmes prédictifs et d'identifier le plus performant pour un objectif donné. Il existe plusieurs façons de découper le jeu de données en sous-parties et la meilleure approche dépend de l'objectif visé, comme nous le verrons dans cet ouvrage.

Enfin, la régularisation est une technique utilisée lors de l'entraînement des algorithmes dans le but d'éviter un surajustement aux données en pénalisant la complexité des algorithmes. Sans régularisation, certains algorithmes peuvent s'ajuster de manière très forte aux données. Ceci conduit à une grande proximité entre les valeurs simulées et observées de la variable de réponse dans le jeu de données utilisé pour l'entraînement de l'algorithme. Cette proximité donne l'illusion que l'algorithme est très performant et peut, en fait, conduire à une forte détérioration de la qualité des prédictions générées pour de nouvelles situations. Pour réduire ce risque, de nombreux algorithmes incluent une procédure de pénalisation qui réduit leur capacité à s'ajuster trop fortement aux données utilisées pour leur entraînement et accroit leur robustesse dans de nouvelles situations.

L'objectif de cet ouvrage est de démocratiser l'usage de la data science pour des applications en lien avec l'agriculture et l'environnement. Ce livre permet aux étudiants, ingénieurs et scientifiques d'acquérir les connaissances de base en data science nécessaires à leur utilisation pratique. L'ouvrage présente des algorithmes supervisés et non-supervisés couramment utilisés en data science

(chapitres 2-7, voir tableau 1), ainsi qu'une explication détaillée des méthodes de validation croisée et de test (chapitre 8). Il comporte à la fois des explications sur le fonctionnement de chaque algorithme, une description des codes informatiques permettant leur utilisation pratique, et des exemples d'applications concrètes dans le domaine des sciences agricoles et environnementales. Les codes informatiques sont proposés à la fois pour les langages R et Python qui sont les langages les plus souvent utilisés en data science. Ces codes peuvent être appliqués avec des logiciels gratuits et permettent à la fois de visualiser les données, d'entraîner différents algorithmes et de les tester. Notre ouvrage ne prétend pas être exhaustif mais constitue une bonne base pour réaliser des projets en data science avec des algorithmes ayant fait leurs preuves dans de nombreux domaines.

Tableau 1. Algorithmes présentés dans le livre.

Chapitre	Nom	Type
2	Régression pénalisée (LASSO, ridge, elastic-net)	Supervisé
3	Régression avec variables d'entrée corrélées (PCR, PLSR)	Supervisé
4	Séparateur à vaste marge (SVM)	Supervisé
5	Arbres et forêts aléatoires	Supervisé
6	Réseaux de neurones et apprentissage profond	Supervisé
7	Partitionnement de données (clustering)	Non-supervisé
7	Extraction de motifs (pattern mining)	Non-supervisé

Chapitre 2. La régression pénalisée

2.1. Les limites de la régression classique

Nous nous plaçons ici dans le contexte d'un modèle reliant une variable dépendante Y avec un certain nombre de variables X explicatives (prédicteurs). Le modèle le plus répandu est la régression linéaire multiple ou plus généralement le modèle linéaire. L'ajustement de ce type de modèle (c'est-à-dire l'estimation des paramètres à partir de données) se fait souvent par les moindres carrés ordinaires (Cornillon et Matzner-Løber, 2011). Cependant, cette méthode conduit parfois à des problèmes d'ajustement (Kuhn et Johnson, 2013) et peut conduire à des valeurs de certains paramètres anormalement élevées ou de signes incohérents, à des estimations imprécises (Fox et Weisberg, 2018), au surajustement d'un modèle trop complexe par rapport aux données disponibles conduisant à des erreurs de prédiction élevées.

Face à ces problèmes, des solutions existent comme par exemple : la sélection d'un sous-ensemble de variables, la régression sur les composantes principales (PCR), la régression par les moindres carrés partiels (PLSR) ou encore la régression pénalisée. Ce chapitre est dédié à cette dernière méthode, appelée également méthode à rétrécisseur[1] ou encore régression régularisée[2]. Elle correspond à un type particulier de modèle linéaire et constitue donc une alternative à la régression linéaire classique. L'objectif recherché est

1 Shrinkage method.

2 Regularized regression.

d'expliquer les variations de Y par une ou plusieurs variables explicatives X, souvent dans le but de prédire Y pour de nouvelles situations. Mais contrairement à la régression classique et aux moindres carrés ordinaires, la régression pénalisée va utiliser une méthode d'estimation des paramètres plus sophistiquée permettant d'obtenir des prédictions souvent plus précises, notamment lorsque le nombre de variables explicatives est grand par rapport au nombre de données disponibles.

L'exemple utilisé ici pour illustrer la régression pénalisée (figure 1) est issu d'une expérimentation avec un piège à spores. La variable Y est une quantité d'ADN d'un champignon capturé sous forme de spores par un piège aérien (échelle logarithmique) ; Y est mesurée quotidiennement sur une période de 40 jours sur un site donné. Les variables explicatives sont la température moyenne du jour (T_jour, en °C), la quantité de précipitation du jour (Pluie_jour, en mm), l'humidité moyenne du jour (Hum_jour, en %), l'interaction humidité*température (hum_temp), et les températures et humidités des jours d'avant (1 à 15 jours ; tmoins1 : température moyenne du jour précédent ; tmoins2 : température moyenne 2 jours avant, etc). Aucune donnée n'est manquante. Le tableau comprend 40 observations (en lignes) et 34 variables explicatives quantitatives (en colonnes). Nous souhaitons construire un modèle prédictif (prédiction pour le jour suivant) de la quantité d'ADN par des variables climatiques. L'exemple est traité dans le logiciel R (R Core Team, 2019); des éléments de code Python sont également fournis.

Figure 1. Extrait du jeu de données « spores ». Seulement une partie des variables explicatives et des observations est présentée ici.

Y	T_jour	Pluie_jour	Hum_jour	hum_temp	tmoins1	tmoins2	Hummoins1	Hummoins2
-7.82	11.05	5.0	90.50	1000.03	10.60	10.05	82.23	86.15
-8.12	12.40	12.6	90.62	1123.75	11.05	10.60	90.50	82.23
-7.83	10.70	15.2	98.06	1049.27	12.40	11.05	90.62	90.50
-7.03	11.55	0.2	88.79	1025.54	10.70	12.40	98.06	90.62
-5.57	11.80	0.2	76.81	906.39	11.55	10.70	88.79	98.06
-5.80	6.30	0.8	86.73	546.39	11.80	11.55	76.81	88.79

2.2. Analyse basée sur le modèle linéaire classique

L'exemple « spores » est d'abord analysé avec la régression linéaire multiple en estimant tous les paramètres associés aux 34 variables explicatives par les moindres carrés ordinaires. Pour cela, nous utilisons la fonction lm de R. Le R^2ajusté du modèle de régression est égale à 0.76. Les estimations des paramètres et leurs erreurs standards sont présentées sur la figure 2. La valeur estimée de l'effet de de T_jour (température moyenne du jour) est égal à -0.586 en régression multiple et à 0.133 en régression simple : il y a une incohérence de signe ; de même, l'erreur standard du paramètre associé à tmoins3 (température moyenne 3 jours avant) vaut environ 43 fois la valeur absolue de l'estimation du paramètre, ce qui montre que ce paramètre est estimé de manière très imprécise.

Nous avons calculé le facteur d'inflation de la variance des paramètres associés aux différentes variables explicatives. Ce facteur mesure l'augmentation de variance de l'estimateur de chaque paramètre (c'est à dire l'augmentation de son imprécision) qui résulte des corrélations entre les variables explicatives (multicolinéarité). Les valeurs de ce facteur s'étendent de 4.5 à 362.1 avec une moyenne à 30.9 (47% des valeurs sont supérieures à 10, seuil au-delà duquel la colinéarité est problématique). Nous avons ensuite calculé la matrice des coefficients de corrélation linéaire des variables explicatives (figure 3) : 2% (soit 11 valeurs) des coefficients sont supérieurs à 0.8 (0.81 à 0.91, en valeur absolue). La colinéarité n'est donc probablement pas la seule cause de l'imprécision des valeurs estimées des paramètres. Ici, le faible nombre de données (40 observations pour 34 variables explicatives) ne permet pas d'estimer précisément tous les paramètres du modèle de régression par les moindres carrés ordinaires.

Enfin, nous avons réalisé une sélection automatique des variables par stepwise avec le critère d'information d'Akaïke (AIC) qui combine une mesure d'ajustement du modèle aux données et une mesure de complexité du modèle liée au nombre de paramètres estimés (Venables et Ripley, 2002). Cette méthode conduit à un modèle avec 22 variables explicatives sélectionnées et des facteurs d'inflation de la variance allant de 2.9 à 25.4. Les paramètres sont donc toujours mal estimés ; la sélection stepwise a atténué un peu le problème mais ne l'a pas réellement résolu.

Figure 2. Valeur estimée (Estim.) et erreur standard (SE) des paramètres (Param.) en régression linéaire ordinaire ; exemple « spores ».

Param.	Estim.	SE	Param.	Estim.	SE	Param.	Estim.	SE
(Intercept)	-46.089	16.766	tmoins8	0.361	0.132	Hummoins5	-0.029	0.037
T_jour	-0.586	0.670	tmoins9	0.009	0.12	Hummoins6	-0.025	0.046
Pluie_jour	-0.058	0.063	tmoins10	0.025	0.116	Hummoins7	0.104	0.04
Hum_jour	-0.013	0.125	tmoins11	-0.069	0.137	Hummoins8	0.091	0.041
hum_temp	0.006	0.008	tmoins12	-0.014	0.17	Hummoins9	0.120	0.051
tmoins1	0.026	0.143	tmoins13	-0.233	0.14	Hummoins10	0.011	0.041
tmoins2	0.335	0.142	tmoins14	-0.301	0.127	Hummoins11	-0.017	0.039
tmoins3	-0.003	0.136	tmoins15	-0.07	0.132	Hummoins12	-0.044	0.039
tmoins4	0.024	0.156	Hummoins1	0.098	0.054	Hummoins13	-0.004	0.035
tmoins5	-0.02	0.146	Hummoins2	0.114	0.04	Hummoins14	-0.047	0.047
tmoins6	0.095	0.129	Hummoins3	0.042	0.047	Hummoins15	0.048	0.045
tmoins7	-0.017	0.134	Hummoins4	0.017	0.036			

Figure 3. Représentation graphique de la matrice des coefficients de corrélation pour l'exemple « spores ». La forme et la couleur des ellipses dépendent du niveau de corrélation.