



Ressourcesinformatiques

 + QUIZ

Version en ligne
OFFERTE !
pendant 1 an

Intelligence artificielle vulgarisée

Le Machine Learning
et le Deep Learning par la pratique

En téléchargement



le code source des
différents cas pratiques

Aurélien VANNIEUWENHUYZE





Les éléments à télécharger sont disponibles à l'adresse suivante :
<http://www.editions-eni.fr>
Saisissez la référence ENI de l'ouvrage **RIIAVUL** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Avant-propos

- 1. Un souhait de vulgarisation des concepts liés à l'intelligence artificielle 17
- 2. Un mot sur l'auteur 18
- 3. À qui s'adresse cet ouvrage ? 19
- 4. Structure du livre 20
- 5. Aspects pratiques 21
- 6. Remerciements 22

Chapitre 1
Vous avez dit intelligence artificielle ?

- 1. Ce que nous allons découvrir et les prérequis 23
- 2. L'intelligence artificielle, ce n'est pas nouveau ! 23
- 3. Quelques dates et périodes clés 25
- 4. Mais qu'est-ce que l'intelligence artificielle ? 27
- 5. Intelligence artificielle, Machine Learning et Deep Learning. 28
- 6. Les différents types d'apprentissage 29
- 7. L'intelligence artificielle fait peur. 30
 - 7.1 La singularité technologique 30
 - 7.2 Des emplois menacés 31
 - 7.3 Des intelligences artificielles détournées. 31
 - 7.4 Des boîtes noires qui font peur 32
 - 7.5 Et la vie privée dans tout ça ? 32

2 — Intelligence Artificielle Vulgarisée

Le Machine Learning et le Deep Learning par la pratique

- 8. Créer une intelligence artificielle chez soi c'est possible! 34

Chapitre 2

Les fondamentaux du langage Python

1. Ce que nous allons découvrir et les prérequis	35
2. Pourquoi Python?	36
3. Installation de Python	36
4. Une rapide découverte du langage Python	39
4.1 Python, un langage interprété	40
4.2 Les opérations de base	40
4.2.1 Affectation et affichage d'une variable	40
4.2.2 Affectations et affichage de plusieurs variables et éventuellement de types différents	41
4.2.3 Création de plusieurs variables de même type et de même valeur	41
4.3 Manipulation de chaînes de caractères	42
4.3.1 Création d'une chaîne de caractères	42
4.3.2 Les concaténations	42
4.3.3 Accès aux caractères d'une chaîne	43
4.3.4 Quelques fonctions utiles	44
4.4 Utilisation des listes	45
4.4.1 L'initialisation	45
4.4.2 Les fonctions de manipulation des listes	45
4.5 Les tuples et les dictionnaires	46
4.6 Les structures conditionnelles et les boucles	47
4.6.1 Les structures conditionnelles	47
4.6.2 Les boucles "tant que"	48
4.6.3 Les boucles "Pour..."	49
5. Installation de PyCharm	50

6. Votre premier script en Python	55
6.1 Création du projet	58
6.2 Création du fichier de script principal	59
6.3 Nos premières lignes de code	60
6.3.1 Un tuple pour le paramétrage	61
6.3.2 Création des zones à l'aide de dictionnaires	61
6.3.3 Regroupement des zones dans une liste	62
6.3.4 Une fonction pour calculer la surface à nettoyer	62
6.3.5 Une deuxième fonction pour coder le temps de nettoyage	64
6.3.6 Le script dans son ensemble	65
7. Conclusion	67

Chapitre 3

Des statistiques pour comprendre les données

1. Ce que nous allons découvrir et les prérequis	69
2. Les statistiques, un outil d'aide à la compréhension des données ..	70
3. Une série de notes en guise d'étude de cas	71
4. Petites notions de vocabulaire avant de commencer	72
4.1 Observations et features	72
4.2 Les types de données	72
5. Et Python dans tout ça ?	73
5.1 Des modules dédiés	73
5.2 Une représentation un peu particulière de notre étude de cas ..	74
5.3 Pas de Python, mais Excel en guise d'outil	74
6. Mesure de tendance centrale	75
6.1 Connaître le nombre d'observations et de features	76
6.2 Les valeurs minimales et maximales	76
6.3 La moyenne arithmétique	78

4 — Intelligence Artificielle Vulgarisée

Le Machine Learning et le Deep Learning par la pratique

6.4	La médiane	80
6.4.1	Cas d'un nombre d'observations impair	80
6.4.2	Cas d'un nombre d'observations pair	81
6.4.3	Retour à notre exemple	81
6.5	Le mode	84
7.	Premières déductions	85
8.	La dispersion	86
8.1	L'étendue	86
8.2	L'écart type (Standard déviation)	87
8.2.1	Calcul de la variance	88
8.2.2	Calcul de l'écart type	89
8.2.3	Interprétation de l'écart type	90
8.3	Les quartiles et interquartile	90
8.3.1	Les quartiles	90
8.3.2	L'interquartile	94
9.	Détection de valeurs extrêmes (outliers en anglais)	94
10.	Traitement des valeurs extrêmes	96
11.	Un peu de visualisation graphique	97
12.	Conclusion sur les données	99
13.	Distribution gaussienne et loi normale	99
13.1	Un exemple pour faire connaissance	100
13.2	Un peu de probabilités	104
14.	Une classe Python pour vous aider à analyser vos données	105
15.	Combien d'observations sont nécessaires pour un bon apprentissage?	107

Chapitre 4

Principaux algorithmes du Machine Learning

1. Ce que nous allons découvrir et les prérequis	109
2. Supervisé ou non supervisé? Régression ou classification?	110
3. Les algorithmes d'apprentissage supervisés pour la régression (prédiction de valeurs)	110
3.1 La régression linéaire univariée (linear regression)	110
3.2 La régression linéaire multiple (Multiple Linear Regression-MLR)	111
3.3 La méthode de descente de gradient	112
3.4 Régression polynomiale (polynomial regression)	113
3.4.1 Monôme et polynôme	114
3.5 Régression logistique	114
3.6 Arbre de décision (decision tree)	115
3.7 Forêts aléatoires (Random Forest)	116
3.8 Agrégation de modèle : le bagging, le boosting et le Gradient boosting	117
3.8.1 Le bagging	117
3.8.2 Le boosting	118
3.8.3 Gradient Boosting (GBoost) et XGBoost	118
3.9 Machine à vecteurs de support (SVM)	118
3.10 KNN (K-Nearest Neighbours)	120
3.11 Naive Bayes	121
4. Les algorithmes pour les apprentissages non supervisés	123
4.1 K-Moyennes (KMeans)	123
4.2 Mean-shift	124
4.3 DBSCAN (Density Based Spatial Clustering of Application with Noise)	125
4.4 Mélange gaussien (Gaussian Mixture Models (GMM))	127
5. Et c'est tout ?	129

6 — Intelligence Artificielle Vulgarisée

Le Machine Learning et le Deep Learning par la pratique

Chapitre 5

Machine Learning et Pokémons : première partie

1. Ce que nous allons découvrir et les prérequis	131
2. L'univers des Pokémons	132
3. Notre mission : choisir le bon Pokémon!	133
4. Des données pour un apprentissage supervisé	133
4.1 Des données basées sur l'expérience	133
4.2 Disposer d'un grand nombre de données d'apprentissage	133
4.3 Des données d'apprentissage et des données de tests.	133
5. Les étapes à réaliser pour mener à bien un projet de Machine Learning	134
5.1 Création et configuration d'un nouveau projet Python.	134
5.1.1 Installation de modules	135
5.1.2 Utilisation des modules dans un script Python	138
5.1.3 Référencement des fichiers de données dans notre projet	139
6. Étape 1 : définir le problème à résoudre	140
7. Étape 2 : acquérir des données d'apprentissage et de tests.	140
8. Étape 3 : préparation des données	141
8.1 De quelles données disposons-nous?	141
8.2 Affichage des dix premières lignes de nos données	144
8.3 Quelles sont les features de catégorisation?	146
8.4 Quelles données sont de type numérique?	147
8.5 Que faut-il penser de la feature LEGENDAIRE?	148
8.6 Manque-t-il des données?	149
8.7 À la recherche des features manquantes.	150
8.8 Place aux observations des combats	153
8.9 Assemblage des observations.	154
8.9.1 Nombre de combats menés	154
8.9.2 Nombre de combats gagnés.	157
8.9.3 Agrégation des données avec le Pokédex.	158

9. Une petite pause s'impose	161
--	-----

Chapitre 6

Machine Learning et Pokémons : seconde partie

1. Ce que nous allons découvrir et les prérequis	163
2. Un peu de statistiques	164
2.1 Le nombre de données (count)	164
2.2 La moyenne (mean)	165
2.3 L'écart type (Std pour Standard Deviation)	165
2.4 Les valeurs minimales et maximales	165
2.5 Les quartiles	166
2.6 Description de notre jeu d'observations	167
3. Quels sont les types de Pokémons qu'un dresseur doit posséder? .	168
4. Les types de Pokémons gagnants et perdants	170
5. Essayons de trouver une corrélation entre les données	172
6. Résumé de nos observations	174
7. Vérifions nos hypothèses	175
8. Passons à la phase d'apprentissage	176
8.1 Découpage des observations en jeu d'apprentissage et jeu de tests	177
8.2 Algorithme de régression linéaire	181
8.3 L'arbre de décision appliqué à la régression	184
8.4 La random forest	185
8.5 Sauvegarde du modèle d'apprentissage	186
9. Phénomènes de surapprentissage (overfitting) et de sous-apprentissage (underfitting)	186
10. Utiliser le modèle d'apprentissage dans une application	187
11. Fin du cas d'étude	191

8 — Intelligence Artificielle Vulgarisée

Le Machine Learning et le Deep Learning par la pratique

Chapitre 7

Bien classifier n'est pas une option

1. Ce que nous allons découvrir et prérequis	193
2. Origines et source du jeu d'observations.	194
3. Un problème de classification et algorithmes de prédiction associés	195
4. Démarche de résolution du problème	195
4.1 Définition du problème à résoudre	195
4.2 Acquisition des données d'apprentissage	196
4.3 Préparer et nettoyer les données	196
4.3.1 De quelles données disposons-nous?	196
4.3.2 De combien de données disposons-nous?	197
4.3.3 Affichage des 10 premières observations	198
4.3.4 Transformation de la feature OBJET	200
4.3.5 Manque-t-il des données?	201
4.4 Analyser et explorer les données	203
4.4.1 Combien de mines et combien de rochers?	203
4.4.2 Moyenne, écart type, min, max et quartiles.	203
4.4.3 À la recherche des valeurs extrêmes	204
4.4.4 Traitement des valeurs extrêmes.	209
4.5 Choix d'un modèle de prédiction et résolution du problème.	209
4.5.1 Des données d'apprentissage et des données de tests.	209
4.5.2 Test des algorithmes	211
4.5.3 Optimisation	213
4.5.4 Et si on boostait un peu tout ça?	216
4.5.5 Que faire des données extrêmes?	216
5. En résumé	220

Chapitre 8

Opinions et classification de textes

1. Ce que nous allons découvrir et les prérequis	223
2. Le traitement automatique du langage naturel (TALN)	224
3. Naive Bayes appliqué au TALN	224
3.1 Le théorème	225
3.2 Un exemple : quels mots-clés choisir ?	225
3.2.1 Détermination des probabilités	226
3.2.2 Conclusion	229
4. Naive Bayes pour l'analyse d'opinion	229
4.1 Étape 1 : normalisation des données	230
4.2 Étape 2 : suppression des stops words	231
4.3 Étape 3 : le stemming	231
4.4 Étape 4 : la lemmatisation	232
4.5 Étape 5 : déterminer le nombre d'occurrences de chaque mot .	232
4.6 Étape 6 : déterminer les probabilités pour l'opinion positive .	233
4.7 Étape 7 : déterminer les probabilités pour le sentiment positif	235
4.8 Étape 8 : déterminer le sentiment d'une nouvelle phrase	236
5. Cas pratique : croyez-vous au réchauffement climatique?	237
5.1 Comment obtenir des données?	237
5.2 Création d'un projet Python	238
5.3 Acquisition des données et préparation des données	239
5.3.1 Chargement du fichier	239
5.3.2 Normalisation	241
5.3.3 Suppression des stop words	242
5.3.4 La stemming	242
5.3.5 La lemmatisation	243

10 — Intelligence Artificielle Vulgarisée

Le Machine Learning et le Deep Learning par la pratique

6. Phases d'apprentissage et de prédiction.	243
6.1 Découpage en jeux de tests et d'apprentissage	243
6.2 Création d'un pipeline d'apprentissage	244
6.3 Apprentissage et analyse des résultats	245
6.4 Classification d'un nouveau message	246
7. L'algorithme SVM (Machine à vecteurs de supports) pour le classement de texte	247
8. L'algorithme SVM plus performant que Naive Bayes?	249

Chapitre 9

Abricots, cerises et clustering

1. Une machine qui apprend seule.	251
2. Acquisition de données d'apprentissage	252
3. Algorithme des K-Means (K-Moyennes).	255
4. Visualiser les données.	256
5. Laisser la machine classifier seule	258
6. Réaliser des classifications	260
7. Des erreurs de classifications	261
8. Algorithme de mélanges gaussiens ou Gaussian Mixture Model (GMM)	263
9. Pour conclure	265

Chapitre 10

Un neurone pour prédire

1. Ce que nous allons découvrir et les prérequis.	267
2. 1957 - Le perceptron	267
2.1 Un peu de biologie	268
2.2 La biologie appliquée au machine learning	270
3. Des données linéairement séparables	271

4. Fonctions d'activation, rétropropagation et descente de gradient	274
4.1 La fonction d'activation	274
4.1.1 La fonction de seuil binaire	274
4.1.2 La fonction sigmoïde	275
4.1.3 La fonction tangente hyperbolique (tanH)	276
4.1.4 La fonction ReLU (Rectified Linear Unit, unité de rectification linéaire)	277
4.1.5 La fonction softMax	278
5. La rétropropagation de l'erreur	279
6. Les fonctions de perte (Loss function)	279
6.1 L'erreur linéaire ou erreur locale	280
6.2 Erreur moyenne quadratique MSE ou erreur globale	280
7. La descente de gradient	281
8. Le biais, un neurone particulier	284
9. Un cas pratique pour comprendre le perceptron	286
9.1 Initialisation du perceptron	287
9.2 Les étapes d'apprentissage	288
9.2.1 Étape 1 : initialisation des poids	288
9.2.2 Étape 2 : chargement des données de la première observation	289
9.2.3 Étape 3 : préactivation	290
9.2.4 Étape 4 : utilisation d'une fonction d'activation	291
9.2.5 Étape 5 : calcul de l'erreur linéaire commise lors de l'apprentissage	292
9.2.6 Étape 6 : ajustement des poids synaptiques	292
10. Codons notre premier neurone formel "From Scratch"	294
10.1 Les données d'apprentissage	294
10.2 Définition des poids	295
10.3 Gestion des hyperparamètres	295
10.4 Codage de fonctions utiles	295
10.5 Passons à l'apprentissage!	297
10.6 À la recherche du point de convergence	300

12 — Intelligence Artificielle Vulgarisée

Le Machine Learning et le Deep Learning par la pratique

10.7 Tests de prédictions	302
11. Un neurone artificiel avec TensorFlow	304
11.1 Un petit mot sur TensorFlow	305
11.2 Données d'apprentissage et de tests	306
11.3 Paramétrage du neurone	306
11.4 L'apprentissage	308
11.5 Tests de prédictions	310
12. Un premier pas vers le Deep Learning	311

Chapitre 11

Utilisation de plusieurs couches de neurones

1. Ce que nous allons découvrir et les prérequis	313
2. Fonctionnement des réseaux de neurones multicouches	313
3. Le cas du Ou exclusif (XOR)	314
3.1 De combien de couches et de neurones avons-nous besoin?	315
3.2 Un exemple chiffré	316
3.2.1 Les données d'apprentissage	316
3.2.2 Initialisation des poids	317
3.2.3 Chargement des données d'entrée	317
3.2.4 Calcul de la préactivation du neurone de sortie	318
3.2.5 Calcul de l'activation	318
3.2.6 Calcul de l'erreur	318
3.2.7 Mise à jour des poids	319
3.3 Place au code avec TensorFlow!	321
4. Le retour des mines et des rochers	326
4.1 De meilleures performances avec plus de neurones sur la couche cachée?	327
4.1.1 Chargement des données d'apprentissage	327
4.1.2 Création des jeux d'apprentissage et de tests	328
4.1.3 Paramétrage du réseau de neurones avec une couche cachée de 24 neurones	328

4.1.4	Réalisation de l'apprentissage	331
4.1.5	Calcul de la précision de l'apprentissage	332
4.1.6	De meilleurs résultats avec une couche cachée composée de 24 neurones?	335
4.1.7	Pouvons-nous obtenir de meilleurs résultats?	337
5.	Conclusion	339

Chapitre 12

La classification d'images

1.	Ce que nous allons découvrir et les prérequis	341
2.	Différence entre détection et classification d'images	341
3.	Des réseaux de neurones convolutifs pour classifier des images	342
3.1	De nombreuses données d'apprentissage nécessaires	342
3.2	Un outil pour illustrer nos propos	343
3.3	L'image d'entrée	344
3.4	Les caractéristiques	345
3.5	La convolution	346
3.6	Pooling	351
3.7	Plusieurs couches de convolutions	353
3.8	Mise à plat (Flatten)	354
3.9	L'apprentissage	355
3.10	Un schéma global qui résume tout	355
4.	Un cas pratique autour de la mode	356
4.1	Présentation de Kaggle	356
4.2	Parlons un peu de Keras	357
4.3	Classifier des robes, pulls et chaussures?	357
4.4	De quelles données disposons-nous?	358
4.5	Préparation des données d'apprentissage	361
4.6	Préparation des données de tests	363

14 — Intelligence Artificielle Vulgarisée

Le Machine Learning et le Deep Learning par la pratique

4.7	Un réseau avec une seule couche de convolution	363
4.7.1	Configuration	363
4.7.2	Compilation, apprentissage et test	365
4.7.3	Conclusion sur l'apprentissage	367
4.7.4	Augmentation du nombre de données	369
4.7.5	Sauvegarde du modèle	373
4.8	Un modèle plus performant	374
5.	Utilisation du modèle avec de nouvelles images	375
6.	Pour conclure ce chapitre	381

Chapitre 13

Votre ordinateur sait lire!

1.	Ce que nous allons découvrir et les prérequis	383
2.	Votre mission	384
2.1	Question n°1 : de quelles données avez-vous besoin?	384
2.2	Question n°2 : comment utiliser le module Python-Mnist ?	385
2.3	Question n°3 : de quelles données disposez-vous à présent?	387
2.4	Question n°4 : est-ce un problème de régression ou de classification?	389
2.5	Question n°5 : quel algorithme allez-vous utiliser?	389
2.6	Question n°6 : comment allez-vous créer vos jeux d'apprentissage et de tests?	390
2.7	Question n°7 : les images sont elles au bon format?	390
2.8	Question n°8 : qu'est-ce que la catégorisation des libellés en One-Hot et comment procéder pour la réaliser?	392
2.9	Question n°9 : avez-vous une petite idée des paramètres à utiliser pour créer le réseau de neurones?	392
2.10	Question n°10 : trouvez-vous le résultat satisfaisant?	395
2.11	Mission accomplie!	395

3. La reconnaissance de lettres sur une vidéo	398
3.1 Une ardoise en guise de support	398
3.2 OpenCV, un module de traitement d'images	399
3.2.1 Utiliser la webcam	400
3.2.2 Détecter les formes rectangulaires.	402
3.2.3 Détecter la zone d'écriture	405
3.2.4 Détecter et extraire la lettre écrite.	406
3.2.5 Reconnaître la lettre écrite et la faire lire	
à votre ordinateur	409
4. Et voilà!	412

Chapitre 14

Hommage au premier ChatBot

1. Introduction	413
2. Eliza	414
2.1 Comment fonctionne Eliza?	414
2.2 Le code d'Eliza	415
3. D'autres ChatBots!	421
4. C'est déjà la fin!	422
Index	425



Chapitre 4

Principaux algorithmes du Machine Learning

1. Ce que nous allons découvrir et les prérequis

Dans le chapitre précédent, nous avons découvert ou redécouvert les fondamentaux de l'analyse statistique descriptive qui, nous le verrons par la pratique, nous permettront de comprendre et de préparer nos données avant l'apprentissage. Nous allons à présent faire connaissance avec les principaux algorithmes du Machine Learning qui vont nous permettre de réaliser cet apprentissage.

Attention, notre objectif en écrivant cet ouvrage est de vulgariser les concepts de l'intelligence artificielle. Par conséquent, **nous n'y aborderons pas les explications théoriques et mathématiques de chaque algorithme d'apprentissage.**

Nous nous contenterons d'une explication la plus explicite possible illustrée par un ou plusieurs exemples le cas échéant. Si nous devons faire un parallèle avec le monde du bricolage, nous allons vous présenter les différents outils à utiliser en fonction du travail à réaliser, mais nous ne vous expliquerons pas comment ils ont été fabriqués.

Nous vous conseillons de considérer ce chapitre comme un aide-mémoire dans lequel vous pourrez venir vous référer au fur et à mesure de votre lecture afin de comprendre pourquoi nous utilisons tel ou tel algorithme et en comprendre son fonctionnement dans les grandes lignes.

■ Remarque

Prérequis nécessaires pour bien aborder ce chapitre : avoir lu le chapitre Des statistiques pour comprendre les données

2. Supervisé ou non supervisé ? Régression ou classification ?

Réaliser un apprentissage supervisé consiste à fournir à la machine des données étiquetées (labellisées) et propices à l'apprentissage. C'est-à-dire que nous allons analyser et préparer les données et leur donner une signification. C'est à partir de cette signification que la machine va réaliser son apprentissage. L'objectif étant d'indiquer à la machine que pour une série de données et pour une observation précise, la valeur à prédire est un chat, un chien ou bien une autre valeur.

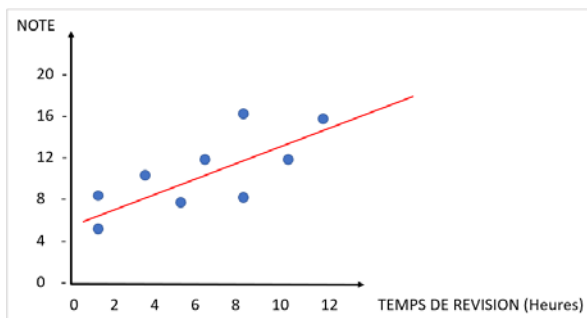
Lorsqu'il s'agit de prédire une valeur, nous parlerons alors de **régression**, dans le cas contraire, nous parlerons de **classification**. Prédire le pourcentage de réussite d'une équipe de football lors d'un match est une régression, prédire que la photo affichée est un chat ou un chien est une classification.

3. Les algorithmes d'apprentissage supervisés pour la régression (prédiction de valeurs)

3.1 La régression linéaire univariée (linear regression)

Cet algorithme cherche à établir, sous forme d'une droite, une relation entre une variable expliquée et une variable explicative. Par exemple, prédire une note à un examen (variable expliquée) en fonction du nombre d'heures de révisions (variable explicative).

En d'autres termes, les données d'une série d'observations sont représentées sous forme d'un nuage de points et l'on cherche à trouver une droite passant au plus près de ces points.



Régression linéaire univariée

Ainsi, connaissant le nombre d'heures de révisions, il nous est possible de prédire approximativement la note que l'on obtiendra au prochain examen.

3.2 La régression linéaire multiple (Multiple Linear Regression-MLR)

Là où nous utilisons une seule variable explicative pour expliquer une autre variable (une note en fonction d'un temps de révision), dans la régression linéaire multivariée nous allons utiliser plusieurs variables explicatives.

Par exemple, nous allons chercher à prédire le temps que va mettre un cycliste pour remporter une étape du tour de France, en fonction de son âge, du temps qu'il a réalisé à la précédente étape, de son classement dans le peloton...

Une étape importante lors de l'utilisation de multiples variables explicatives est leur **normalisation** (mise à l'échelle). Dans notre exemple, le temps réalisé en minutes lors de la précédente étape peut éventuellement varier entre 160 à 200, la position dans le peloton entre 1 et 80 en fonction du nombre de participants au tour de France. Nous ne sommes donc pas sur la même échelle pour chacune des variables explicatives (160 à 200 vs 1 à 80).

La mise à l'échelle (*scaling*) va donc consister à faire en sorte que la moyenne de chaque série d'observations soit égale à 0, que la variance et l'écart-type soient égaux à 1. Cette méthode est également appelée centrage de réduction.

112 — Intelligence Artificielle Vulgarisée

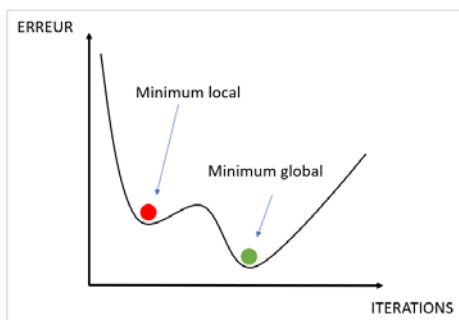
Le Machine Learning et le Deep Learning par la pratique

Une fois cette étape réalisée, nous pouvons passer à la prédiction grâce à **la méthode de descente de gradient** ou bien encore **la méthode des moindres carrés**. Ces deux méthodes prenant en compte les différentes variables explicatives mises à l'échelle dans le but de prédire la variable expliquée.

3.3 La méthode de descente de gradient

Cette notion est essentielle, car elle est appliquée dans divers algorithmes d'apprentissage du Machine Learning et du Deep Learning que nous verrons un peu plus loin dans cet ouvrage.

Lorsqu'un système est en phase d'apprentissage, il commet des erreurs. Le taux d'erreur diminue au fur et à mesure de l'apprentissage, mais il se peut qu'à un moment donné l'erreur augmente pour à nouveau rediminuer et atteindre un niveau d'erreur plus bas que le précédent qui est le niveau optimal d'apprentissage.



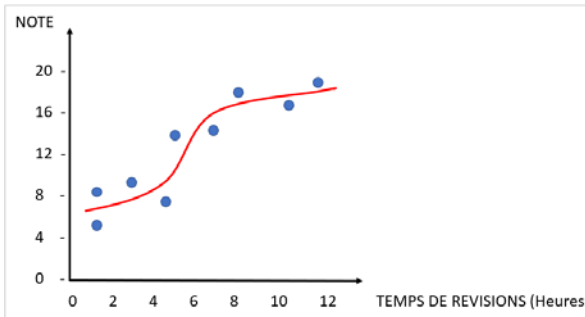
La descente de gradient

Sur la figure précédente, on constate qu'en début d'apprentissage, l'erreur diminue progressivement pour ensuite remonter. Nous aurions donc tendance à dire que le niveau optimal d'apprentissage a été atteint puisque de nouvelles erreurs apparaissent. Cependant, on peut s'apercevoir qu'après de nouvelles itérations d'apprentissage, l'erreur continue de diminuer pour atteindre un niveau plus bas que le précédent appelé minimum global ! Le niveau optimal d'apprentissage n'était donc pas atteint.

L'algorithme du gradient consiste donc à trouver par itérations successives le minimum global de la fonction de coût (erreur). Par analogie souvent reprise dans la littérature, imaginez-vous en haut d'une montagne avec pour objectif d'atteindre la plaine en contre bas. À chaque pas, vous analysez votre situation et décidez d'avancer de quelques pas, quitte à remonter pour prendre le chemin qui mène au but. Le gradient correspondant à la pente du sol que vous êtes en train de parcourir. Le "pas" porte également le nom de taux d'apprentissage dont nous verrons la mise en pratique dans le chapitre Un neurone pour prédire.

3.4 Régression polynomiale (polynomial regression)

Il est parfois difficile de trouver une droite pouvant passer parmi les points de la série d'observations de façon optimale. Cependant, il est parfois possible de trouver un lien entre les variables à l'aide d'une courbe. C'est ce que permet la régression polynomiale en ajoutant des plis à la courbe à l'aide d'éléments appelés polynômes.



Régression polynomiale

114 — Intelligence Artificielle Vulgarisée

Le Machine Learning et le Deep Learning par la pratique

3.4.1 Monôme et polynôme

Un monôme est une expression mathématique s'exprimant sous cette forme :

$$\alpha x^n$$

Où

- α (alpha) est un nombre réel ou complexe appelé coefficient du monôme.
- n est un entier naturel représentant le degré du monôme.

Ainsi, $5x^2$ est un monôme de coefficient 5 et de degré 2.

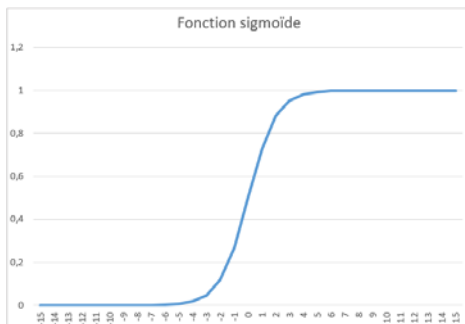
Un polynôme est une somme de monômes. On peut donc dire que $5x^2 + 2x$ est un polynôme.

3.5 Régression logistique

Comme nous venons de le voir, lorsque les données ne sont pas linéairement séparables, il est possible d'utiliser des polynômes pour donner à notre droite la possibilité de réaliser des virages afin de séparer nos observations.

La régression logistique utilise, quant à elle, une fonction logistique encore appelée sigmoïde ou courbe en S. Ce type d'algorithme est à appliquer dans des problèmes de classification.

À noter que nous croiserons plus en détail la fonction sigmoïde lorsque nous traiterons en pratique les réseaux de neurones.



Fonction sigmoïde ou courbe en S