

Préface

Introduction générale

Avant-propos

| | |
|---|----|
| 1. Pourquoi ce livre ? | 23 |
| 2. À qui s'adresse ce livre ? | 25 |
| 3. Comment est organisé ce livre ? | 27 |
| 4. Comment lire ce livre ? | 28 |
| 5. Quels sont les prérequis pour la lecture de ce livre ? | 29 |
| 6. Qui est l'auteur ? | 29 |
| 7. Remerciements | 31 |

Partie 1 : La Data Science - Concepts généraux

Chapitre 1-1

La Data Science

| | |
|---|----|
| 1. Objectif du chapitre | 33 |
| 2. L'objectif recherché en Machine Learning | 34 |
| 3. Une expérimentation Machine Learning | 38 |
| 3.1 Types de données | 46 |
| 3.2 Préparation des données | 48 |
| 4. Cycle de vie d'un projet Data Science | 51 |
| 5. Les algorithmes du Machine Learning | 54 |
| 6. Le problème de surapprentissage | 56 |
| 7. Les paramètres et les hyperparamètres | 57 |
| 8. Validation croisée | 59 |

2 ————— Le Machine Learning

et l'IA générative avec Python

| | |
|---|----|
| 9. Données d'entraînement, de validation et de test | 63 |
| 10. Métriques de performance | 64 |
| 10.1 Métriques pour les problèmes de régression | 66 |
| 10.2 Métriques pour la classification | 69 |
| 10.2.1 Matrice de confusion binaire | 69 |
| 10.2.2 Matrice de confusion générale | 71 |
| 10.2.3 Exemple de matrice de confusion | 72 |
| 10.2.4 La courbe ROC | 74 |
| 10.3 Métriques pour le clustering | 75 |
| 11. Conclusion | 75 |

Partie 2 : Outils techniques de la Data Science - Python, Numpy, Pandas et Jupyter

Chapitre 2-1

Le langage Python

| | |
|---|-----|
| 1. Objectif du chapitre | 77 |
| 2. Python en deux mots | 78 |
| 3. Installer l'interpréteur Python | 78 |
| 4. Les bases de la programmation Python | 81 |
| 4.1 Hello world avec Python | 81 |
| 4.1.1 La fonction print | 81 |
| 4.1.2 La fonction input | 85 |
| 4.2 Les structures de données | 85 |
| 4.2.1 Les variables numériques | 86 |
| 4.2.2 Les chaînes de caractères | 91 |
| 4.2.3 Le type booléen | 93 |
| 4.2.4 Les listes | 99 |
| 4.2.5 Les tuples | 102 |
| 4.2.6 Les dictionnaires | 103 |
| 4.2.7 Les ensembles | 105 |

| | | |
|-------|---|-----|
| 4.2.8 | Liste vs tuple vs dictionnaire vs ensemble | 109 |
| 4.3 | Structurer un code Python. | 110 |
| 4.3.1 | L'indentation et les blocs de code | 110 |
| 4.3.2 | Écrire une instruction sur plusieurs lignes | 111 |
| 4.3.3 | Écrire plusieurs instructions sur une ligne | 113 |
| 4.3.4 | Les commentaires en Python. | 113 |
| 4.4 | Les instructions conditionnelles | 113 |
| 4.4.1 | Les conditions avec la structure if | 114 |
| 4.4.2 | Les conditions avec la structure if-else | 115 |
| 4.4.3 | Les conditions avec la structure if-elif-else | 116 |
| 4.5 | Les boucles. | 119 |
| 4.5.1 | La boucle for | 119 |
| 4.5.2 | La boucle for et la fonction zip | 123 |
| 4.5.3 | La boucle while | 129 |
| 4.5.4 | Contrôler les boucles avec break | 131 |
| 4.5.5 | Contrôler les boucles avec continue | 132 |
| 4.6 | Les fonctions. | 133 |
| 4.6.1 | Définir et utiliser une fonction sans paramètre | 134 |
| 4.6.2 | Les fonctions avec paramètres | 136 |
| 4.6.3 | Les valeurs par défaut des paramètres. | 137 |
| 4.6.4 | Renvoi de résultats | 141 |
| 4.6.5 | La portée des variables | 142 |
| 4.6.6 | Passage d'arguments à une fonction | 145 |
| 4.6.7 | Les fonctions récursives | 148 |
| 4.7 | Les listes en compréhension. | 153 |
| 4.7.1 | Les origines des listes en compréhension | 153 |
| 4.7.2 | Construire une liste avec les listes en compréhension | 154 |
| 4.7.3 | Application de fonction avec une liste en compréhension. | 155 |
| 4.7.4 | Utiliser if-else avec les listes en compréhension | 156 |
| 4.7.5 | Filtrer avec les listes en compréhension | 157 |

4 — Le Machine Learning

et l'IA générative avec Python

| | | |
|-------|---|-----|
| 4.8 | Les expressions régulières | 157 |
| 4.8.1 | Regex sans caractères spéciaux | 160 |
| 4.8.2 | Regex avec caractères spéciaux | 162 |
| 4.8.3 | Regex avec les multiplicateurs | 164 |
| 4.8.4 | Regex avec un nombre d'occurrences limité | 167 |
| 4.8.5 | Regex avec groupage des résultats | 168 |
| 4.8.6 | Taille des motifs | 169 |
| 4.8.7 | Aller plus loin avec les expressions régulières | 174 |
| 4.9 | Gestion des exceptions | 175 |
| 4.9.1 | La levée des exceptions | 175 |
| 4.9.2 | Utiliser le bloc try-except | 177 |
| 4.9.3 | Gérer plusieurs exceptions | 178 |
| 4.9.4 | Utiliser la clause finally | 180 |
| 4.9.5 | Utiliser la structure try-except-finally-else | 181 |
| 4.9.6 | Lever une exception avec raise | 183 |
| 5. | Conclusion | 185 |

Chapitre 2-2

La bibliothèque NumPy

| | | |
|-------|--|-----|
| 1. | Objectif du chapitre | 187 |
| 2. | NumPy en deux mots | 188 |
| 3. | Les tableaux NumPy | 188 |
| 3.1 | Création de tableaux NumPy | 188 |
| 3.1.1 | Créer un tableau à une dimension | 189 |
| 3.1.2 | Créer un tableau à plusieurs dimensions | 189 |
| 3.2 | Les dimensions d'un tableau NumPy | 191 |
| 3.3 | Le type et la taille d'un tableau NumPy | 193 |
| 3.4 | Fonction d'initialisation de tableaux NumPy | 195 |
| 4. | Accéder aux données d'un tableau NumPy | 197 |
| 4.1 | Accès aux données d'un tableau à une dimension | 197 |
| 4.2 | Accès aux données d'un tableau à deux dimensions | 199 |

| | | |
|--------|---|-----|
| 4.3 | Accès aux données d'un tableau à trois dimensions | 201 |
| 5. | Modifier les données d'un tableau NumPy | 202 |
| 6. | Copier un tableau NumPy dans un autre tableau NumPy | 203 |
| 7. | Algèbre linéaire avec NumPy | 205 |
| 7.1 | Opérations mathématiques de base avec NumPy | 205 |
| 7.2 | Opérations sur les matrices avec NumPy | 207 |
| 8. | Réorganiser des tableaux NumPy | 208 |
| 8.1 | Restructurer un tableau NumPy | 208 |
| 8.2 | Superposer des tableaux NumPy | 210 |
| 9. | Statistiques descriptives avec NumPy | 211 |
| 10. | Lire des données NumPy à partir d'un fichier | 213 |
| 11. | Les masques booléens avec NumPy | 214 |
| 11.1 | Créer et utiliser un masque booléen | 214 |
| 11.2 | Un masque avec plusieurs conditions | 216 |
| 11.3 | Les fonctions numpy.any et numpy.all | 217 |
| 12. | Tableaux NumPy versus listes Python | 219 |
| 12.1 | Comparaison des tailles en mémoire | 220 |
| 12.2 | Comparaison des temps de calcul | 221 |
| 12.2.1 | Temps de calcul sur une liste Python | 222 |
| 12.2.2 | Temps de calcul sur un tableau NumPy | 223 |
| 13. | Conclusion | 224 |

Chapitre 2-3

La bibliothèque Pandas

| | | |
|----|----------------------------------|-----|
| 1. | Objectif du chapitre | 225 |
| 2. | C'est quoi, Pandas ? | 226 |
| 3. | Installation de Pandas | 227 |

6 — Le Machine Learning

et l'IA générative avec Python

| | |
|--|-----|
| 4. DataFrame Pandas | 228 |
| 4.1 Création d'un DataFrame à partir d'un dictionnaire | 229 |
| 4.2 Création d'un DataFrame à partir d'un tableau NumPy | 231 |
| 4.3 Chargement des données à partir de fichiers | 232 |
| 4.3.1 Lecture des données d'un fichier CSV | 233 |
| 4.3.2 Lecture d'un fichier texte | 234 |
| 5. Accès aux données d'un DataFrame | 235 |
| 5.1 Lire les lignes d'un DataFrame | 236 |
| 5.1.1 Lire une ligne d'un DataFrame | 236 |
| 5.1.2 Lire plusieurs lignes d'un DataFrame | 236 |
| 5.1.3 Parcourir les lignes d'un DataFrame | 237 |
| 5.1.4 Filtrer les lignes avec une condition | 238 |
| 5.1.5 Filtrer les lignes avec plusieurs conditions | 238 |
| 5.1.6 Filtrage avec des critères textuels | 239 |
| 5.1.7 Réinitialiser les index | 240 |
| 5.1.8 Filtrer avec les valeurs uniques | 242 |
| 5.1.9 Filtrer avec une expression régulière | 242 |
| 5.2 Accéder aux variables d'un DataFrame | 243 |
| 5.2.1 Liste des variables d'un DataFrame | 243 |
| 5.2.2 Accès aux valeurs d'une colonne | 244 |
| 5.2.3 Accès à plusieurs colonnes | 244 |
| 5.3 Lire une cellule spécifique avec les index | 245 |
| 6. Modifier un DataFrame | 245 |
| 6.1 Modifier les valeurs dans un DataFrame | 245 |
| 6.2 Modifier la structure d'un DataFrame | 246 |
| 6.2.1 Ajouter une variable à un DataFrame | 246 |
| 6.2.2 Réordonner les variables d'un DataFrame | 249 |
| 6.2.3 Supprimer une variable d'un DataFrame | 250 |
| 6.2.4 Utiliser la méthode melt pour diminuer le nombre de variables | 251 |
| 6.3 Appliquer une fonction sur une variable avec la méthode apply | 253 |
| 6.4 Modification avec conditions | 255 |

| | | |
|-----|---|-----|
| 6.5 | Ajouter des lignes dans un DataFrame | 256 |
| 7. | Tri sur les données d'un DataFrame | 257 |
| 7.1 | Tri avec un seul critère | 257 |
| 7.2 | Tri avec plusieurs critères | 259 |
| 8. | Sauvegarder les données d'un DataFrame | 260 |
| 9. | Faire des statistiques sur un DataFrame | 261 |
| 9.1 | Faire un résumé direct | 261 |
| 9.2 | Faire un résumé par agrégation | 262 |
| 9.3 | Agrégation avec plusieurs paramètres | 264 |
| 10. | Lecture des fichiers de grande taille | 265 |
| 11. | Conclusion | 267 |

Chapitre 2-4

Travailler avec Jupyter

| | | |
|-------|---|-----|
| 1. | Objectif du chapitre | 269 |
| 2. | Installation de l'environnement Anaconda et Jupyter | 270 |
| 3. | Travailler avec Jupyter | 276 |
| 3.1 | Les documents dans Jupyter | 278 |
| 3.1.1 | Créer un dossier | 279 |
| 3.1.2 | Renommer un dossier | 280 |
| 3.1.3 | Déplacer un dossier | 281 |
| 3.1.4 | Charger des documents | 282 |
| 3.1.5 | Supprimer des éléments | 283 |
| 3.1.6 | Navigation dans l'arborescence des dossiers | 284 |
| 3.1.7 | Créer un notebook | 285 |
| 3.2 | Utiliser un notebook Jupyter | 286 |
| 3.2.1 | Renommer un notebook | 286 |
| 3.2.2 | Les cellules Jupyter | 287 |
| 3.2.3 | Les fonctionnalités d'un notebook | 291 |

8 ————— Le Machine Learning

et l'IA générative avec Python

| | | |
|-------|---|-----|
| 3.3 | Utiliser les widgets Jupyter | 297 |
| 3.3.1 | Le widget FloatSlider | 297 |
| 3.3.2 | Associer une fonction à un slider | 298 |
| 3.3.3 | Le widget interact | 300 |
| 3.3.4 | Le widget Image | 302 |
| 3.3.5 | Le widget DatePicker | 302 |
| 4. | Conclusion | 303 |

Partie 3 : Les statistiques

Chapitre 3-1 Statistiques

| | | |
|-------|---|-----|
| 1. | Objectif du chapitre | 305 |
| 2. | Les statistiques descriptives | 306 |
| 2.1 | Paramètres de position | 306 |
| 2.1.1 | La moyenne | 306 |
| 2.1.2 | Le mode | 307 |
| 2.1.3 | La médiane | 307 |
| 2.1.4 | Les quartiles | 310 |
| 2.2 | Paramètres de dispersion | 310 |
| 2.2.1 | La variance | 311 |
| 2.2.2 | Calcul de la variance avec la formule de Koenig | 311 |
| 2.2.3 | L'écart-type | 311 |
| 2.2.4 | L'écart interquartile | 312 |
| 3. | Les lois de probabilité | 312 |
| 4. | La loi normale | 314 |
| 5. | L'échantillonnage | 318 |
| 5.1 | Principe de l'échantillonnage | 318 |
| 5.2 | Résultats sur la distribution des moyennes | 319 |
| 5.3 | Résultats sur la distribution des proportions | 328 |
| 5.4 | Théorème central limite | 332 |

| | |
|--|-----|
| 6. Les statistiques inférentielles | 333 |
| 6.1 Estimation ponctuelle | 334 |
| 6.2 Estimation de la moyenne par intervalle de confiance. | 338 |
| 6.3 Estimation d'une proportion par intervalle de confiance. | 342 |
| 6.4 Test d'hypothèse. | 346 |
| 6.4.1 Tests paramétriques | 347 |
| 6.4.2 Tests non paramétriques | 347 |
| 6.4.3 Construire un test d'hypothèse | 348 |
| 6.5 Types de tests d'hypothèse | 351 |
| 6.5.1 Test de conformité | 351 |
| 6.5.2 Test d'adéquation | 352 |
| 6.5.3 Tests d'homogénéité. | 353 |
| 6.5.4 Test d'indépendance de variables | 354 |
| 6.6 Exemple numérique de test de conformité d'une moyenne. . . | 355 |
| 6.7 Le paradoxe de Simpson. | 358 |
| 7. Les séries temporelles | 361 |
| 7.1 Techniques d'analyse des séries temporelles. | 362 |
| 7.1.1 La décomposition des séries temporelles. | 362 |
| 7.1.2 Lissage des données | 362 |
| 7.1.3 Modèles de prévision | 362 |
| 7.2 Stationnarité des séries temporelles | 363 |
| 7.2.1 Tests de stationnarité. | 363 |
| 7.2.2 Transformation pour rendre une série stationnaire . . . | 363 |
| 8. Conclusion | 364 |

Partie 4 : Les grands algorithmes du Machine Learning

Chapitre 4-1

La régression linéaire et polynomiale

| | |
|---|-----|
| 1. Objectif du chapitre | 365 |
| 2. La régression linéaire simple | 366 |
| 2.1 La régression linéaire simple de point de vue géométrique . . . | 367 |
| 2.2 La régression linéaire simple de point de vue analytique . . . | 368 |
| 2.2.1 La méthode des moindres carrés | 368 |
| 2.2.2 Quelques considérations statistiques sur les données . . | 370 |
| 3. La régression linéaire multiple | 371 |
| 3.1 La méthode des moindres carrés pour la régression multiple . | 372 |
| 3.2 La méthode de la descente de gradient | 373 |
| 3.3 Exemple de régression linéaire multiple | 374 |
| 3.3.1 Définition du jeu de données utilisées | 374 |
| 3.3.2 Régression linéaire multiple avec Scikit-learn | 375 |
| 3.3.3 Importer les modules Scikit-learn | 376 |
| 3.3.4 Lecture des données dans un DataFrame | 377 |
| 3.3.5 Normalisation des données | 378 |
| 3.3.6 Construction d'un modèle linéaire | 381 |
| 3.3.7 Évaluation d'un modèle linéaire. | 383 |
| 3.3.8 Évaluer le futur comportement d'un modèle | 387 |
| 3.3.9 Cross-validation avec KFold. | 390 |
| 4. La régression polynomiale | 398 |
| 4.1 Exemple de régression polynomiale. | 399 |
| 4.1.1 Construction d'un modèle polynomial. | 399 |
| 4.1.2 Le coefficient de détermination R^2 | 406 |
| 4.1.3 R^2 et les valeurs extrêmes | 408 |
| 4.1.4 Modèle polynomial et surapprentissage | 408 |
| 5. Aller plus loin avec les modèles de régression. | 414 |
| 5.1 La régularisation Lasso | 414 |
| 5.2 La régularisation Ridge. | 415 |

| | |
|---------------------|-----|
| 6. Conclusion | 415 |
|---------------------|-----|

Chapitre 4-2

La régression logistique

| | |
|--|-----|
| 1. Objectif du chapitre | 417 |
| 2. La régression logistique | 418 |
| 3. Prédire les survivants du Titanic | 422 |
| 3.1 Définition du jeu de données Titanic | 422 |
| 3.2 Réalisation du modèle de régression logistique | 423 |
| 3.2.1 Chargement des modules Scikit-learn | 423 |
| 3.2.2 Lecture des données | 424 |
| 3.2.3 Traitement des valeurs manquantes | 425 |
| 3.2.4 Transformation de variables | 426 |
| 3.2.5 Sélection des variables | 428 |
| 3.2.6 Traitement des variables catégorielles | 430 |
| 3.2.7 Entraînement du modèle logistique | 431 |
| 3.2.8 Le seuil de décision | 432 |
| 4. L'algorithme One-vs-All | 436 |
| 5. Conclusion | 436 |

Chapitre 4-3

Arbres de décision et Random Forest

| | |
|--|-----|
| 1. Objectif du chapitre | 437 |
| 1.1 Construction d'un arbre de décision | 438 |
| 1.2 Prédire la classe d'appartenance avec un arbre de décision ... | 441 |
| 1.3 Considérations théoriques sur les arbres de décision | 442 |
| 1.3.1 Choix de la variable de segmentation | 443 |
| 1.3.2 Profondeur d'un arbre de décision | 445 |
| 2. Problème de surapprentissage avec un arbre de décision | 449 |
| 3. Random Forest | 449 |

12 ————— Le Machine Learning

et l'IA générative avec Python

| | |
|---|-----|
| 4. Exemple de Random Forest avec Scikit-learn | 450 |
| 5. Conclusion | 456 |

Chapitre 4-4

L'algorithme k-means

| | |
|---|-----|
| 1. Objectif du chapitre | 457 |
| 2. k-means du point de vue géométrique | 458 |
| 3. k-means du point de vue algorithmique | 465 |
| 4. Application de k-means avec Scikit-learn | 467 |
| 5. L'algorithme k-means et les valeurs extrêmes. | 474 |
| 6. Choisir le k de k-means | 480 |
| 6.1 Déterminer k avec la méthode Elbow | 483 |
| 6.2 Déterminer k avec le coefficient de silhouette | 489 |
| 7. Les limites de k-means | 495 |
| 8. Avantages et inconvénients de l'algorithme k-means | 500 |
| 9. Quelques versions de l'algorithme k-means | 501 |
| 10. Conclusion | 501 |

Chapitre 4-5

Support Vector Machine

| | |
|---|-----|
| 1. Objectif du chapitre | 503 |
| 2. Le SVM du point de vue géométrique. | 504 |
| 3. Le SVM du point de vue analytique | 509 |
| 4. Données non linéairement séparables. | 513 |
| 4.1 Le Kernel Trick | 515 |
| 4.2 La condition de Mercer. | 517 |
| 4.3 Exemple de fonction noyau | 517 |

| | | |
|-------|--|-----|
| 5. | Détecter les fraudes de cartes de crédit | 518 |
| 5.1 | Les données des transactions de cartes de crédit | 518 |
| 5.2 | Application de l'algorithme SVM pour la détection des transactions bancaires frauduleuses | 519 |
| 5.2.1 | Application de l'algorithme SVM sur les données creditcard.csv | 519 |
| 5.2.2 | Application du SVM sur un sous-ensemble de creditcard.csv | 526 |
| 5.2.3 | Application du SVM sur des données normalisées. | 528 |
| 5.3 | Les paramètres de l'algorithme SVM. | 533 |
| 5.3.1 | Le paramètre Kernel pour la variation de la fonction noyau. | 534 |
| 5.3.2 | Le paramètre C | 535 |
| 5.3.3 | Le paramètre Gamma. | 539 |
| 5.3.4 | Le paramètre C versus le paramètre Gamma | 541 |
| 5.3.5 | Tuning des hyperparamètres d'un SVM avec GridSearchCV. | 541 |
| 6. | Conclusion | 545 |

Chapitre 4-6

Analyse en composantes principales

| | | |
|-------|--|-----|
| 1. | Objectif du chapitre | 547 |
| 2. | Pourquoi l'ACP ? | 548 |
| 3. | L'ACP du point de vue géométrique | 550 |
| 4. | L'ACP du point de vue analytique. | 552 |
| 5. | Indicateurs de la qualité de la représentation des données | 555 |
| 5.1 | Indicateurs liés aux individus | 555 |
| 5.1.1 | Score des individus | 556 |
| 5.1.2 | Qualité de la représentation des individus | 556 |
| 5.1.3 | Contribution des individus | 556 |

14 — Le Machine Learning

et l'IA générative avec Python

| | | |
|-------|---|-----|
| 5.2 | Indicateurs liés aux variables | 557 |
| 5.2.1 | Le cercle des corrélations | 557 |
| 5.2.2 | Qualité de la représentation des variables. | 558 |
| 5.2.3 | Contribution des variables. | 559 |
| 6. | Exemple d'ACP avec Python | 559 |
| 6.1 | Déterminer le nombre de facteurs pertinents. | 564 |
| 6.2 | Interprétation des résultats sur les individus | 569 |
| 6.2.1 | Représentation des individus. | 570 |
| 6.2.2 | Calcul de la qualité de la représentation des individus . | 574 |
| 6.2.3 | Calcul de la contribution des individus. | 575 |
| 6.3 | Interprétation des résultats sur les variables | 576 |
| 6.3.1 | Tracer un cercle des corrélations | 576 |
| 6.3.2 | Calcul de la qualité de la représentation des variables . | 579 |
| 6.3.3 | Calcul des contributions des variables | 580 |
| 6.4 | Représentation de nouveaux individus. | 581 |
| 7. | Conclusion | 583 |

Chapitre 4-7

Les réseaux de neurones

| | | |
|-----|---|-----|
| 1. | Objectif du chapitre | 585 |
| 2. | Modélisation d'un neurone artificiel | 586 |
| 2.1 | Le neurone biologique | 587 |
| 2.2 | Le neurone artificiel | 588 |
| 3. | Architecture d'un réseau de neurones | 590 |
| 4. | L'algorithme de rétropropagation | 593 |
| 5. | Exemple d'un réseau de neurones avec Scikit-learn | 604 |
| 6. | Conclusion | 611 |

Partie 5 : Le Deep Learning et le traitement automatique du langage

Chapitre 5-1

Le Deep Learning avec TensorFlow

| | |
|--|-----|
| 1. Objectif du chapitre | 613 |
| 2. Le Deep Learning : notions générales | 614 |
| 2.1 Réseau de neurones avec plusieurs couches d'entrée | 617 |
| 2.2 Réseau de neurones avec plusieurs couches de sortie | 618 |
| 2.3 Réseau de neurones avec des branchements conditionnels | 619 |
| 2.4 Réseau de neurones avec de la récurrence RNN | 620 |
| 2.5 Réseau de neurones avec couches de convolution CNN | 621 |
| 2.6 Éviter le surapprentissage avec les couches Dropout | 623 |
| 2.7 Le Transfer Learning | 625 |
| 3. Introduction à TensorFlow | 629 |
| 3.1 Installer TensorFlow | 629 |
| 3.1.1 Créer un environnement virtuel | 630 |
| 3.1.2 Installer des bibliothèques dans un environnement virtuel avec Anaconda | 634 |
| 3.1.3 Installer la bibliothèque TensorFlow | 636 |
| 3.1.4 Tester TensorFlow | 637 |
| 3.2 Opérations élémentaires avec les tensors | 638 |
| 3.2.1 Travailler avec les tensors | 639 |
| 3.2.2 Les tensors variables | 643 |
| 3.2.3 Initialiser les tensors | 643 |
| 3.2.4 Opérations algébriques avec les tensors | 644 |
| 4. Les réseaux de neurones avec Sequential API | 645 |
| 4.1 Charger les données | 646 |
| 4.2 Définir un MLP avec Sequential API | 651 |
| 4.3 Accéder aux éléments d'un réseau de neurones | 653 |
| 4.4 Initialisation des poids et des biais d'un réseau de neurones .. | 655 |
| 4.5 Compiler un réseau de neurones | 657 |

16 — Le Machine Learning

et l'IA générative avec Python

| | | |
|-------|---|-----|
| 4.6 | Entraîner un réseau de neurones | 658 |
| 4.7 | Analyser les résultats de l'entraînement d'un réseau de neurones | 660 |
| 4.8 | Évaluer un réseau de neurones. | 662 |
| 4.9 | Prédire avec un réseau de neurones pour la classification | 662 |
| 5. | Utiliser Functional API. | 664 |
| 5.1 | Un modèle Functional API avec plusieurs couches d'entrée . . | 665 |
| 5.2 | Un modèle Functional API avec plusieurs couches de sortie . . | 668 |
| 6. | Opérations avancées sur les réseaux de neurones. | 671 |
| 6.1 | Monitorer un réseau de neurones | 671 |
| 6.1.1 | Contrôler les critères d'arrêt avec les callbacks | 671 |
| 6.1.2 | TensorBoard | 674 |
| 6.1.3 | Sauvegarder un réseau de neurones. | 678 |
| 6.1.4 | Charger et utiliser un réseau de neurones. | 678 |
| 6.2 | Réseaux de neurones de convolution | 680 |
| 6.3 | Réutiliser un réseau de neurones | 684 |
| 6.4 | Le Transfer Learning. | 686 |
| 6.4.1 | Chargement des données locales | 688 |
| 6.4.2 | Chargement du modèle VGG16 | 689 |
| 6.4.3 | Extraction des features. | 689 |
| 6.4.4 | Étendre un modèle | 690 |
| 6.4.5 | Chargement des données de test pour le Transfer Learning. | 691 |
| 7. | Aller plus loin avec le Deep Learning et TensorFlow | 694 |
| 8. | Conclusion | 695 |

Chapitre 5-2

Le Deep Learning avec OpenCV

| | |
|--|-----|
| 1. Objectif du chapitre | 697 |
| 2. Introduction à OpenCV | 698 |
| 2.1 Qui utilise OpenCV ? | 698 |
| 2.2 Exemples de cas d'utilisation d'OpenCV | 699 |
| 3. L'architecture d'OpenCV | 700 |
| 4. Installer et tester OpenCV | 703 |
| 5. Manipuler les images avec OpenCV | 704 |
| 5.1 Charger une image | 704 |
| 5.2 Capturer le flux d'une vidéo | 706 |
| 5.3 Ajouter des objets dans une image | 708 |
| 5.4 Ajouter des objets dans une vidéo | 710 |
| 5.5 Gérer les clics de la souris sur une vidéo | 712 |
| 5.6 Comprendre la structure d'une image | 714 |
| 5.7 Modifier les pixels d'une image | 716 |
| 5.8 Flouter une image | 717 |
| 6. La détection et la reconnaissance d'objets | 719 |
| 6.1 La détection faciale sur une image | 719 |
| 6.2 La détection faciale sur une vidéo | 721 |
| 6.3 Traquer les mouvements | 723 |
| 6.4 Détecter des objets avec YOLO | 725 |
| 7. Conclusion | 729 |

Chapitre 5-3

Les réseaux de neurones antagonistes génératifs

| | |
|--|-----|
| 1. Objectif du chapitre | 731 |
| 2. Introduction au GAN | 732 |
| 2.1 Comprendre les différents types de modèles d'IA générative | 733 |
| 2.2 Définition et origine des GAN | 733 |

18 — Le Machine Learning

et l'IA générative avec Python

| | | |
|-------|--|-----|
| 2.3 | Importance des GAN dans l'apprentissage profond | 734 |
| 2.4 | Les différents types de GAN | 735 |
| 2.5 | Quelques exemples de GAN | 735 |
| 2.6 | Avantages et inconvénients des GAN | 736 |
| 2.6.1 | Avantages | 736 |
| 2.6.2 | Inconvénients | 736 |
| 3. | Fonctionnement des GAN | 737 |
| 4. | Mes premiers pas avec PyTorch | 744 |
| 4.1 | Tester PyTorch dans Google Colab | 745 |
| 4.2 | Transformer une image en tensors | 748 |
| 4.3 | Appliquer des filtres sur les images | 749 |
| 4.3.1 | Accentuer les bords dans une image | 749 |
| 4.3.2 | Détecter des bords verticaux | 751 |
| 4.3.3 | Détecter des bords horizontaux | 753 |
| 4.3.4 | Appliquer un noyau gaussien | 755 |
| 4.3.5 | Donner un effet de gravure à une image | 755 |
| 5. | Développer des réseaux de neurones avec PyTorch | 756 |
| 5.1 | Entraîner un modèle à rejeter les mauvaises images | 758 |
| 5.2 | Entraîner un modèle à accepter les images réalistes | 768 |
| 6. | Générer des images réalistes avec un GAN | 769 |
| 6.1 | Chargement des vraies images de référence | 771 |
| 6.2 | L'entrée du Generator ou le vecteur latent | 775 |
| 6.2.1 | Pourquoi le vecteur latent est-il important ? | 775 |
| 6.2.2 | Quel est l'effet de modifier la taille du vecteur latent ? | 776 |
| 6.3 | Définition du réseau de neurones Generator | 777 |
| 6.4 | Définition d'un réseau de neurones Discriminator | 779 |
| 6.5 | Tester le Generator et le Discriminator avant la boucle d'apprentissage | 781 |
| 6.6 | Implémentation de la boucle d'apprentissage du GAN | 784 |
| 6.7 | Tester le Generator et le Discriminator | 791 |
| 7. | Conclusion | 794 |

Chapitre 5-4

Le traitement automatique du langage

| | |
|--|-----|
| 1. Objectif du chapitre | 795 |
| 2. NLP : concepts généraux | 796 |
| 2.1 Le nettoyage des données textuelles | 798 |
| 2.1.1 Suppression des stopwords | 798 |
| 2.1.2 Appliquer le Stemming sur un texte | 800 |
| 2.1.3 Appliquer la Lemmatization sur un texte | 800 |
| 2.1.4 Stemming versus Lemmatization | 800 |
| 2.2 Vectorisation des données textuelles | 801 |
| 2.2.1 La vectorisation par comptage d'occurrences des mots | 802 |
| 2.2.2 La vectorisation avec TF-IDF | 804 |
| 2.2.3 La vectorisation avec N-Gram | 806 |
| 2.2.4 Feature Engineering sur des documents | 807 |
| 3. Exemple complet pour la détection des spams | 808 |
| 3.1 Installation de la NLTK | 809 |
| 3.2 Modèle de détection de spams | 810 |
| 4. Conclusion | 817 |

Chapitre 5-5

Le prompt engineering

| | |
|---|-----|
| 1. Objectif du chapitre | 819 |
| 2. Le prompt engineering | 820 |
| 2.1 Concepts généraux | 820 |
| 2.2 Les tokens | 820 |
| 2.3 Comment écrire un prompt efficace et précis ? | 822 |
| 3. Exemples de prompts | 822 |
| 3.1 Les prompts simples et naïfs | 823 |
| 3.2 Les prompts zero-shot | 823 |
| 3.3 Les prompts few-shot | 827 |
| 3.4 Les prompts Chain-of-Thought | 830 |

20 ————— Le Machine Learning

et l'IA générative avec Python

| | | |
|------|--|-----|
| 3.5 | Les prompts de type Generated-Knowledge-Prompting | 835 |
| 3.6 | Les prompts Directional-stimulus-prompting | 836 |
| 3.7 | Les prompts OPRO. | 837 |
| 3.8 | Résoudre les problèmes de logique avec des prompts | 840 |
| 3.9 | Faire des résumés avec Chain-of-Density | 842 |
| 3.10 | Générer du code avec les LLM | 844 |
| 4. | Conclusion | 853 |

Annexe

La programmation orientée objet avec Python

| | | |
|-------|---|-----|
| 1. | Programmation orientée objet avec Python | 855 |
| 1.1 | Pourquoi la programmation orientée objet? | 855 |
| 1.2 | Classes et objets | 857 |
| 1.2.1 | Définir une classe | 857 |
| 1.2.2 | La fonction <code>__init__</code> | 858 |
| 1.2.3 | Instanciation d'un objet | 860 |
| 1.2.4 | Les attributs d'un objet | 861 |
| 1.2.5 | Les méthodes d'objet | 863 |
| 1.2.6 | Les attributs de classe | 865 |
| 1.2.7 | Les méthodes de classe | 866 |
| 1.2.8 | Les méthodes statiques | 868 |
| 1.2.9 | Sécuriser les attributs | 869 |
| 1.3 | L'héritage | 873 |
| 1.3.1 | L'héritage simple | 873 |
| 1.3.2 | L'héritage multiple | 877 |
| 1.4 | Les classes abstraites | 879 |
| 1.5 | Les interfaces | 881 |

| | |
|---|-----|
| 1.6 Les méthodes spéciales | 885 |
| 1.6.1 Afficher un objet avec la fonction <code>print()</code> | 885 |
| 1.6.2 Personnaliser les accès aux attributs d'une classe | 888 |
| 1.6.3 Vérifier la validité d'un attribut | 889 |
| 1.6.4 Comparer deux objets | 891 |
| 1.6.5 Rendre les objets callable | 892 |
| 2. Les modules | 893 |
| 2.1 Importer des modules | 894 |
| 2.2 Le module principal | 897 |
| 3. Pour aller plus loin avec Python | 900 |
| Index | 901 |

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

Avant-propos

Chapitre 1
Introduction

| | |
|--|----|
| 1. Des données partout. | 15 |
| 1.1 Provenance des données. | 16 |
| 1.1.1 Le Web. | 16 |
| 1.1.2 Les données privées. | 17 |
| 1.1.3 Créons nos propres données | 18 |
| 1.2 Forme des données | 19 |
| 1.3 Volumétrie | 20 |
| 2. La data science | 21 |
| 2.1 Feature engineering | 21 |
| 2.1.1 La collecte des données. | 22 |
| 2.1.2 Le nettoyage | 22 |
| 2.1.3 L'exploration. | 23 |
| 2.1.4 L'analyse | 24 |
| 2.2 La modélisation. | 25 |
| 2.2.1 La sélection et la préparation des données | 25 |
| 2.2.2 La séparation des données | 26 |
| 2.2.3 La phase d'expérimentation et d'évaluation. | 27 |
| 2.2.4 La finalisation | 28 |
| 2.2.5 La présentation des résultats | 28 |
| 2.2.6 La maintenance. | 28 |

2 ————— Maîtrisez la Data Science

avec Python

| | |
|--|----|
| 3. Python | 29 |
| 3.1 Les atouts naturels de Python | 29 |
| 3.2 Les bibliothèques spécialisées | 30 |
| 3.3 Plus encore | 31 |

Chapitre 2

Bases de Python et environnements

| | |
|--|----|
| 1. Les notebooks | 33 |
| 1.1 Principe du notebook | 33 |
| 1.1.1 Fonctionnement par cellule | 34 |
| 1.1.2 Possibilité d’annoter le code | 34 |
| 1.1.3 Affichage de contenu interactif | 34 |
| 1.2 Comment créer un notebook | 36 |
| 1.2.1 Installation directe du module Jupyter | 36 |
| 1.2.2 Installation de la suite Anaconda | 36 |
| 1.2.3 Google Colaboratory | 37 |
| 2. Commandes de base | 39 |
| 2.1 Acquisition des données | 39 |
| 2.1.1 Définition du dossier de travail | 40 |
| 2.1.2 Accès aux données | 40 |
| 2.2 Définition des données | 42 |
| 2.2.1 Changement du type | 42 |
| 2.2.2 Gestion des dates | 43 |
| 2.2.3 Taille du stockage par type | 44 |
| 2.3 Structuration du code | 46 |
| 2.3.1 PEP8 | 46 |
| 2.3.2 Optimisation du code | 48 |
| 3. Utilisation avancée | 49 |
| 3.1 Gestion des bibliothèques | 49 |
| 3.1.1 Installation | 50 |
| 3.1.2 Mise à jour | 50 |
| 3.1.3 Suppression | 50 |

| | | |
|-------|--|----|
| 3.2 | L'environnement virtuel | 51 |
| 3.2.1 | Déploiement d'un environnement virtuel | 51 |
| 3.2.2 | Utilisation d'un environnement virtuel dans un notebook | 52 |
| 3.3 | Les notions utiles pour la data science | 53 |
| 3.3.1 | Le pipeline | 54 |
| 3.3.2 | La programmation orientée objet (POO) | 55 |
| 3.3.3 | Les décorateurs | 56 |
| 3.3.4 | La gestion des erreurs | 58 |

Chapitre 3

Préparer les données avec Pandas et Numpy

| | | |
|-------|---|----|
| 1. | Pandas, la bibliothèque Python incontournable pour manipuler les données | 61 |
| 1.1 | Installation | 61 |
| 1.2 | Structure et type de données | 62 |
| 1.3 | Possibilités offertes | 63 |
| 2. | Numpy, le pilier du calcul numérique | 64 |
| 2.1 | La structure ndarray | 64 |
| 2.1.1 | Une structure homogène | 65 |
| 2.1.2 | L'indexation | 68 |
| 2.1.3 | La modification des structures | 69 |
| 2.1.4 | La vectorisation | 73 |
| 2.2 | La puissance au service du calcul scientifique | 74 |
| 2.3 | Les possibilités offertes par Numpy | 75 |
| 2.3.1 | Opérations mathématiques de base | 75 |
| 2.3.2 | Algèbre linéaire et calculs statistiques | 76 |
| 2.3.3 | Création d'images | 78 |
| 3. | Collecte des données | 79 |
| 3.1 | Acquisition et contrôle des données | 81 |
| 3.1.1 | Les formats classiques des fichiers de données | 81 |
| 3.1.2 | L'acquisition de données en pratique | 82 |

4 _____ Maîtrisez la Data Science

avec Python

| | | |
|-------|---|-----|
| 3.2 | Manipulations avancées des données | 87 |
| 3.2.1 | Concaténation | 87 |
| 3.2.2 | Fusion | 89 |
| 3.2.3 | Agrégation | 90 |
| 3.2.4 | Export des données | 93 |
| 4. | Nettoyage des données | 96 |
| 4.1 | Sélection des données | 97 |
| 4.2 | Contrôle de la qualité des données | 99 |
| 4.2.1 | Définition du bon type de données | 99 |
| 4.2.2 | Gestion des problèmes d'encodage | 100 |
| 4.3 | Identification des valeurs atypiques ou aberrantes | 100 |
| 4.3.1 | Z-score et méthode des quartiles | 101 |
| 4.3.2 | Local Outlier Factor | 104 |
| 4.4 | Gestion des outliers | 106 |
| 4.4.1 | Suppression des valeurs | 106 |
| 4.4.2 | Changement de la distribution | 107 |
| 4.4.3 | Conservation des valeurs aberrantes | 107 |
| 4.5 | Imputations | 108 |
| 4.5.1 | Imputation par la valeur la plus fréquente (modale) | 108 |
| 4.5.2 | Imputation par la moyenne ou la médiane | 109 |
| 4.5.3 | Imputation par régression | 110 |
| 4.5.4 | Imputation basée sur les plus proches voisins (KNN) | 111 |
| 4.5.5 | Autres types d'imputations | 112 |

Chapitre 4

DataViz avec Matplotlib, Seaborn, Plotly

| | | |
|-------|---|-----|
| 1. | Introduction à la visualisation des données | 113 |
| 1.1 | La visualisation au service de la compréhension | 114 |
| 1.2 | La méthodologie | 114 |
| 1.2.1 | Contextualisation des recherches | 114 |
| 1.2.2 | Public concerné | 115 |
| 1.2.3 | Les nombreuses possibilités de graphiques | 115 |

| | |
|---|-----|
| 1.2.4 Règles à respecter concernant les graphiques | 116 |
| 2. Les principales bibliothèques pour la visualisation : | |
| Matplotlib, Seaborn et Plotly-Express | 117 |
| 2.1 Matplotlib | 117 |
| 2.1.1 Présentation de Matplotlib | 117 |
| 2.1.2 Premiers pas avec Matplotlib | 118 |
| 2.1.3 Personnalisation et options avancées | 120 |
| 2.2 Seaborn | 124 |
| 2.2.1 Présentation de Seaborn | 124 |
| 2.2.2 Simplification de l'exploration des relations complexes | 124 |
| 2.3 Plotly.express | 127 |
| 2.3.1 La version simplifiée de Plotly | 127 |
| 2.3.2 L'interactivité de Plotly-Express | 128 |
| 2.3.3 L'avenir de Plotly-Express | 129 |
| 3. Les différents types de graphiques | 129 |
| 3.1 Les enjeux | 129 |
| 3.1.1 Le cheminement vers le bon graphique | 129 |
| 3.1.2 Les postes importants | 130 |
| 3.1.3 Les contraintes | 130 |
| 3.2 Les graphiques univariés | 133 |
| 3.2.1 Graphiques univariés pour les données numériques . . . | 133 |
| 3.2.2 Graphiques univariés pour les données catégorielles . . | 140 |
| 3.2.3 Récapitulatif | 152 |
| 3.3 Les graphiques bivariés et multivariés | 152 |
| 3.3.1 Graphiques bivariés portant sur des variables de même nature | 153 |
| 3.3.2 Graphiques bivariés portant sur des variables de natures différentes | 159 |
| 3.3.3 Graphiques multivariés | 166 |
| 3.4 Les autres types de graphiques | 172 |
| 3.4.1 La cartographie | 172 |
| 3.4.2 Les données temporelles | 178 |
| 3.4.3 Les autres solutions graphiques | 182 |

6 ————— Maîtrisez la Data Science

avec Python

Chapitre 5 Analyse des données

| | |
|---|-----|
| 1. Introduction à l'analyse des données | 185 |
| 1.1 Définition et rôle de l'analyse de données | 186 |
| 1.2 Enjeux | 186 |
| 1.2.1 Innovation et créativité | 187 |
| 1.2.2 Prise de conscience des contraintes spécifiques | 188 |
| 1.2.3 Amélioration de la prise de décision | 189 |
| 2. Statistiques descriptives et inférentielles | 191 |
| 2.1 Description des variables quantitatives | 192 |
| 2.1.1 Mesures de tendance centrale | 192 |
| 2.1.2 Mesures de dispersion | 198 |
| 2.1.3 La distribution | 203 |
| 2.2 Description des variables catégorielles | 207 |
| 2.2.1 Fréquence, proportion et gestion des modalités rares .. | 207 |
| 2.2.2 Tableau de contingence | 209 |
| 2.2.3 Indices de diversité | 210 |
| 2.3 Statistiques inférentielles | 215 |
| 2.3.1 Concepts de base | 215 |
| 2.3.2 Hypothèses nulles et alternatives | 215 |
| 2.3.3 P-value | 216 |
| 2.3.4 Significativité | 216 |
| 2.3.5 Marge d'erreur et impact des effectifs sur l'intervalle de confiance | 217 |
| 3. Modules Python pour l'analyse de données | 219 |
| 3.1 Les capacités limitées des modules classiques | 219 |
| 3.2 Les modules spécialisés en statistiques | 220 |
| 3.2.1 Scipy | 220 |
| 3.2.2 Statmodels | 221 |
| 4. Tests statistiques de normalité | 221 |
| 4.1 Contexte et objectif | 221 |
| 4.2 Les Q-Q plots | 222 |

| | | |
|-------|--|-----|
| 4.2.1 | Définition et tracé du graphique | 222 |
| 4.2.2 | Interprétation | 223 |
| 4.3 | Principe de fonctionnement général des tests de normalité . . | 224 |
| 4.3.1 | Principe de fonctionnement. | 224 |
| 4.3.2 | Les différents tests de normalité | 225 |
| 5. | Tests statistiques bivariés | 228 |
| 5.1 | Tests bivariés entre des variables de même nature. | 229 |
| 5.1.1 | Corrélations entre variables numériques | 229 |
| 5.1.2 | Tests d'indépendance entre variables catégorielles . . . | 235 |
| 5.2 | Tests bivariés entre des variables de nature différente. | 241 |
| 5.2.1 | Tests de comparaison à deux modalités | 241 |
| 5.2.2 | Tests de comparaison à trois modalités ou plus. | 243 |
| 5.2.3 | Conclusions sur les tests bivariés | 249 |
| 6. | Analyse multivariée | 249 |
| 6.1 | Analyse de la variance multivariée (MANOVA) | 250 |
| 6.1.1 | Présentation et champs d'applications | 250 |
| 6.1.2 | Cas pratique d'utilisation. | 250 |
| 6.2 | Analyse en composantes multiples (ACM) | 252 |
| 6.3 | Analyse en composantes principales (ACP) | 255 |
| 6.3.1 | Un des piliers de la data science. | 255 |
| 6.3.2 | Utilisation sur un cas pratique | 256 |
| 6.3.3 | L'écroulis des valeurs propres | 257 |
| 6.3.4 | Le cercle des corrélations | 258 |
| 6.3.5 | Le graphique des individus. | 259 |

Chapitre 6

Le Machine Learning avec Scikit-Learn

| | | |
|-------|---|-----|
| 1. | Introduction au Machine Learning : concepts et types de modèles. | 263 |
| 1.1 | L'apprentissage non supervisé | 264 |
| 1.1.1 | Définition | 264 |
| 1.1.2 | La réduction dimensionnelle | 265 |

8 — Maîtrisez la Data Science

avec Python

| | | |
|-------|---|-----|
| 1.1.3 | Le clustering | 267 |
| 1.2 | L'apprentissage supervisé | 269 |
| 1.2.1 | Introduction | 269 |
| 1.2.2 | Régression | 270 |
| 1.2.3 | Classification | 271 |
| 1.3 | Le texte et l'image | 273 |
| 1.3.1 | Définitions des concepts | 273 |
| 1.3.2 | Le texte et le NLP | 273 |
| 1.3.3 | Le traitement des images | 274 |
| 2. | Présentation de Scikit-Learn, la bibliothèque Python pour la data science | 276 |
| 2.1 | Une offre simple et complète de fonctionnalités | 276 |
| 2.2 | Des méthodes communes aux différentes fonctions | 277 |
| 2.2.1 | La méthode fit() | 278 |
| 2.2.2 | Les méthodes transform et fit_transform | 279 |
| 2.2.3 | La méthode predict | 280 |
| 2.2.4 | La méthode score() | 280 |
| 2.2.5 | Les méthodes get_params et set_params | 281 |
| 2.3 | Le soutien de la licence BSD et d'une communauté active | 282 |
| 3. | Les grandes étapes d'un projet de Machine Learning | 282 |
| 3.1 | La préparation des données | 282 |
| 3.1.1 | La séparation des variables explicatives de la variable cible | 282 |
| 3.1.2 | La séparation entre données d'entraînement et données de test | 283 |
| 3.1.3 | Les transformations des variables | 284 |
| 3.1.4 | La mise en œuvre ciblée des transformations | 287 |
| 3.1.5 | Finalisation de la préparation des données | 290 |
| 3.2 | L'expérimentation | 291 |
| 3.2.1 | Définition des métriques pour l'évaluation | 292 |
| 3.2.2 | Les algorithmes d'optimisation d'hyperparamètres | 295 |
| 3.2.3 | Le modèle de base (DummyRegressor et DummyClassifier) | 295 |

| | |
|--|-----|
| 3.2.4 Tests des divers algorithmes avec différentes combinaisons de paramètres. | 297 |
| 3.2.5 L'évaluation et le choix final | 299 |
| 4. Conclusions sur la modélisation | 301 |

Chapitre 7 L'apprentissage supervisé

| | |
|--|-----|
| 1. Introduction | 303 |
| 2. Les familles d'algorithmes | 303 |
| 2.1 Les algorithmes linéaires | 304 |
| 2.1.1 Les régressions. | 304 |
| 2.1.2 Les régressions régularisées | 307 |
| 2.1.3 Les machines à vecteur de support (SVM) | 310 |
| 2.2 Les algorithmes semi-linéaires (modèles à noyau) | 313 |
| 2.3 Les algorithmes non linéaires. | 317 |
| 2.3.1 Les plus proches voisins (KNN). | 317 |
| 2.3.2 L'arbre de décision | 319 |
| 2.3.3 Les méthodes ensemblistes | 321 |
| 2.3.4 Les réseaux de neurones | 327 |
| 3. La régression en pratique | 330 |
| 3.1 Préparation des données. | 331 |
| 3.1.1 Import des données | 331 |
| 3.1.2 Séparation des variables explicatives de la variable cible. | 332 |
| 3.1.3 Séparation entre données d'entraînement et de test | 332 |
| 3.1.4 Les transformations des variables | 333 |
| 3.1.5 Finalisation de la préparation des données | 333 |
| 3.2 Fonction de calcul et d'affichage des régressions | 335 |
| 3.3 La modélisation d'une régression. | 337 |
| 3.3.1 Modèle de base (DummyRegressor) | 337 |
| 3.3.2 Test des algorithmes concurrents | 338 |
| 3.3.3 Le pipeline | 343 |

10 _____ Maîtrisez la Data Science

avec Python

| | |
|--|-----|
| 4. La classification en pratique. | 347 |
| 4.1 Préparation des données. | 347 |
| 4.1.1 Import des données | 347 |
| 4.1.2 Séparation entre les variables explicatives et la variable cible | 347 |
| 4.1.3 Séparation entre données d'entraînement et de test ... | 347 |
| 4.1.4 Transformation des colonnes | 348 |
| 4.1.5 Remise en forme des noms | 348 |
| 4.1.6 Ajustement du type des variables | 349 |
| 4.2 Fonction de calcul et d'affichage des classifications. | 349 |
| 4.3 Expérimentations | 352 |
| 4.3.1 Modèle de base (DummyClassifier) | 352 |
| 4.3.2 Algorithmes concurrents | 354 |
| 5. Conclusion | 359 |

Chapitre 8

L'apprentissage non supervisé

| | |
|---|-----|
| 1. Introduction | 363 |
| 2. La réduction dimensionnelle | 364 |
| 2.1 L'ACP en pratique pour analyser. | 364 |
| 2.1.1 Préparation des données. | 364 |
| 2.1.2 L'ébouillissement des valeurs propres | 367 |
| 2.1.3 Le cercle des corrélations | 370 |
| 2.1.4 Le graphique des individus. | 373 |
| 2.2 L'ACP en pratique pour modéliser. | 376 |
| 2.3 Les autres algorithmes de réduction dimensionnelle | 378 |
| 3. Le clustering | 383 |
| 3.1 La pratique du clustering avec le K-means | 383 |
| 3.1.1 Acquisition et préparation des données | 383 |
| 3.1.2 Les tests pour déterminer le nombre de clusters | 386 |
| 3.1.3 Choix du clustering | 389 |
| 3.1.4 Le score ARI | 391 |

| | | |
|-------|--|-----|
| 3.2 | Les autres algorithmes de clustering | 392 |
| 3.2.1 | GMM. | 392 |
| 3.2.2 | Meanshift | 394 |
| 3.2.3 | DBSCAN | 396 |

Chapitre 9 Modéliser le texte et l'image

| | | |
|-------|--|-----|
| 1. | La modélisation du texte | 401 |
| 1.1 | Les modules du NLP | 402 |
| 1.1.1 | NLTK. | 402 |
| 1.1.2 | TextBlob | 404 |
| 1.1.3 | spaCy. | 405 |
| 1.2 | Mise en pratique de la NLP | 407 |
| 1.2.1 | Prétraitement des données. | 407 |
| 1.2.2 | Les extracteurs de caractéristiques | 411 |
| 1.2.3 | La modélisation. | 412 |
| 1.3 | Introduction aux modèles avancés en NLP. | 418 |
| 1.3.1 | Les représentations de mots. | 418 |
| 1.3.2 | L'encodage des phrases. | 420 |
| 1.3.3 | Transformers et modèles contextuels | 420 |
| 1.3.4 | Les Larges Languages Models (LLM) | 421 |
| 2. | La modélisation des images | 421 |
| 2.1 | Les solutions de Machine Learning destinées aux images | 422 |
| 2.1.1 | Pillow pour s'initier au prétraitement. | 422 |
| 2.1.2 | Scikit-image. | 426 |
| 2.1.3 | OpenCV | 431 |
| 2.2 | Méthodes de modélisation des images | 433 |
| 2.2.1 | Segmenter | 434 |
| 2.2.2 | Détecter. | 438 |
| 2.2.3 | Classifier | 441 |

12 _____ Maîtrisez la Data Science

avec Python

| | | |
|-------|-----------------------------------|-----|
| 2.3 | Aller plus loin avec les CNN | 443 |
| 2.3.1 | Principe de fonctionnement du CNN | 443 |
| 2.3.2 | Transfer learning | 444 |
| 2.3.3 | Initiation à Tensorflow et Keras | 445 |
| 2.3.4 | Exemples d'utilisation des CNN | 446 |

Chapitre 10

Mener un projet de data science avec Python

| | | |
|-------|--|-----|
| 1. | Introduction | 455 |
| 2. | Le sujet : déterminer le prix des véhicules d'occasion | 455 |
| 2.1 | Les données | 455 |
| 2.2 | Les étapes du projet | 456 |
| 2.2.1 | Le notebook de l'EDA | 456 |
| 2.2.2 | Le notebook de modélisation | 456 |
| 2.2.3 | Les aléas des données | 457 |
| 3. | La modélisation en pratique | 457 |
| 3.1 | Notebook 1 : EDA | 457 |
| 3.1.1 | Acquisition et premiers contrôles des données | 457 |
| 3.1.2 | Nettoyage des données | 460 |
| 3.1.3 | Exploration et analyse | 467 |
| 3.2 | Notebook 2 : modélisation simple | 480 |
| 3.2.1 | Acquisition et sélection des données | 480 |
| 3.2.2 | Modélisation | 482 |
| 3.2.3 | Résultats | 484 |
| 3.3 | Notebook 3 : modélisation mixte | 491 |
| 3.3.1 | Acquisition et sélection des données | 491 |
| 3.3.2 | Modélisation | 493 |
| 3.3.3 | Résultats | 494 |
| 4. | Conclusion | 496 |

Conclusion

1. Le rôle central des données et de leur compréhension 497
2. Des évolutions qui transforment et accélèrent tout 498
 - 2.1 L'évolution du matériel technologique 498
 - 2.2 L'amélioration des modèles 499
 - 2.3 La diffusion dans le grand public et la prise en compte progressive des enjeux 499
3. Importance de la théorie et invitation à l'exploration 500

Index 501