Devenir Data Scientist avec Python

Durée : 5 jours

Prérequis : Maîtriser l’algorithmique, avoir une appétence pour les mathématiques, la connaissance de Python et des statistiques est un plus.

Public visé : Développeurs, chefs de projets proches du développement, ingénieur scientifique sachant coder

Objectifs : Savoir mettre en place un DataLake et un DataMart en SQL ou big data, puis de mettre en place une stratégie de Machine Learning en Python afin de créer le modèle le plus satisfaisant possible en le mesurant et en affichant les résultats, le tout en utilisant des algorithmes performants

Introduction aux Data Sciences

* Qu’est que la data science ?
* Qu’est-ce que Python ?
* Qu’est que le Machine Learning ?
* Apprentissage supervisé vs non supervisé
* Les statistiques
* La randomisation
* La loi normale

Introduction à Python pour les Data Science

* Les bases de Python
* Les listes
* Les tuples
* Les dictionnaires
* Les modules et packages
* L’orienté objet
* Le module math
* Les expressions lambda
* Map, reduce et filter
* Les listes en intention
* Le module CSV
* Les modules DB-API 2
* Anaconda

Introduction aux DataLake, DataMart et DataWharehouse

* Qu’est-ce qu’un DataLake ?
* Les différents types de DataLake
* Le Big Data
* Qu’est-ce qu’un DataWharehouse ?
* Qu’est qu’un DataMart ?
* Mise en place d’un DataMart
* Les fichiers
* Les bases de données SQL
* Les bases de données No-SQL

Python Package Installer

* Utilisation de PIP
* Installation de package PIP
* PyPi

MathPlotLib

* Utilisation de la bibliothèque scientifique de graphes MathPlotLib
* Affichage de données dans un graphique 2D
* Affichages de sous-graphes
* Affichage de polynômes et de sinusoïdales

Machine Learning

* Mise en place d’une machine learning supervisé
* Qu’est qu’un modèle et un dataset
* Qu’est qu’une régression
* Les différents types de régression
* La régression linéaire
* Gestion du risque et des erreurs
* Quarter d’Ascombe
* Trouver le bon modèle
* La classification
* Filtrage
* Loi normale, variance et écart type
* Apprentissage
* Mesure de la performance
* No Fee Lunch

La régression linéaire en Python

* Programmer une régression linéaire en Python
* Utilisation des expressions lambda et des listes en intention
* Afficher la régression avec MathPlotLib
* L’erreur quadratique
* La variance
* Le risque

Le Big Data

* Qu’est-ce que Apache Hadoop ?
* Qu’est-ce que l’informatique distribué ?
* Installation et configuration de Hadoop
* HDFS
* Création d’un datanode
* Création d’un namenode distribué
* Manipulation de HDFS
* Hadoop comme DataLake
* Map Reduce
* Yarn
* Hive
* Hadoop comme DataMart
* Python HDFS

Les bases de données NoSql

* Les bases de données structurées
* SQL avec SQLite et Postgresql
* Les bases de données non ACID
* JSON
* MongoDB
* Cassandra, Redis, CouchDb
* MongoDB sur HDFS
* MongoDB comme DataMart
* PyMongo

Numpy et SciPy

* Les tableaux et les matrices
* L’algèbre linéaire avec Numpy
* La régression linéaire SciPy
* Le produit et la transposée
* L’inversion de matrice
* Les nombres complexes
* L’algèbre complexe
* Les transformées de Fourier
* Numpy et Mathplotlib

ScikitLearn

* Le machine Learning avec SKLearn
* La régression linéaire
* La création du modèle
* L’échantillonnage
* La randomisation
* L’apprentissage avec fit
* La prédiction du modèle
* Les metrics
* Choix du modèle
* Régressions polynomiales
* PreProcessing et Pipeline
* Régressions non polynomiales

Nearest Neighbors

* Algorithme des k plus proches voisins (k-NN)
* Modèle de classification
* K-NN avec SciKitLearn
* Choix du meilleur k
* Sérialisation du modèle
* Variance vs Erreurs
* Autres modèles : SVN, Random Forest

Pandas

* L’analyse des données avec Pandas
* Les Series
* Les DataFrames
* La théorie ensembliste avec Pandas
* L’importation des données CSV
* L’importation de données SQL
* L’importation de données MongoDB
* Pandas et SKLearn

Le Clustering

* Regroupement des données par clusterisation
* Les clusters SKLearn avec k-means
* Autres modèles de clusterisation : AffinityPropagation, MeanShift, …
* L’apprentissage semi-supervisé

Jupyter

* Présentation de Jupyter et IPython
* Installation
* Utilisation de Jupyter avec Mathplotlib et SKlearn

Python Yield

* La programmation efficace en Python
* Le générateurs et itérateurs
* Le Yield return
* Le Yield avec Db-API 2, Pandas et SKlearn

Les réseaux neuronaux

* Le perceptron
* Les réseaux neuronaux
* Les réseaux neuronaux supervisés
* Les réseaux neuronaux semi-supervisés
* Les réseaux neuronaux par Hadoop Yarn
* Les heuristiques
* Le deep learning