

# Python, Data Scientist avec Python

5 j (35 heures)

Ref : PYTS

## Public

Développeur, chef de projets proche du développement, ingénieur scientifique sachant coder

## Pré-requis

Maîtriser l'algorithmique,  
Avoir une appétence pour les mathématiques,  
La connaissance de Python et des statistiques est un plus

## Moyens pédagogiques

Formation réalisée en présentiel ou à distance selon la formule retenue  
Exposés, cas pratiques, synthèse, assistance post-formation pendant trois mois  
Un poste par stagiaire, vidéoprojecteur, support de cours fourni à chaque stagiaire

## Modalités de suivi et d'évaluation

Feuille de présence émargée par demi-journée par les stagiaires et le formateur  
Exercices de mise en pratique ou quiz de connaissances tout au long de la formation permettant de mesurer la progression des stagiaires  
Questionnaire d'évaluation de la satisfaction en fin de stage  
Auto-évaluation des acquis de la formation par les stagiaires  
Attestation de fin de formation

## Objectifs

Savoir mettre en place un DataLake et un DataMart en SQL ou big data  
Définir une stratégie de Machine Learning en Python  
Créer le modèle le plus satisfaisant possible en le mesurant et en affichant les résultats  
Développer des algorithmes performants

## Programme détaillé

### 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
- Le module CSV
- Les modules DB-API 2
- Anaconda

## **INTRODUCTION AUX DATALAKE, DATAMART ET DATAWAREHOUSE**

---

- 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

## **MATPLOTLIB**

---

- Utilisation de la bibliothèque scientifique de graphes MatPlotLib
- 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
- 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 Matplotlib
- 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
- 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 Matplotlib

## **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  
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

---