

## DATA SCIENTIST

**Durée**

**5 jours**

**Référence Formation**

**4-IT-DAS**

### Objectifs

Savoir mettre en place un DataLake et un DataMart en SQL ou big data

Savoir 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

### Participants

Développeurs, chefs de projets proches du développement, ingénieurs scientifiques 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

Accueil des stagiaires dans une salle dédiée à la formation équipée d'un vidéo projecteur, tableau blanc et paperboard ainsi qu'un ordinateur par participant pour les formations informatiques.

Positionnement préalable oral ou écrit sous forme de tests d'évaluation, feuille de présence signée en demi-journée, évaluation des acquis tout au long de la formation.

En fin de stage : QCM, exercices pratiques ou mises en situation professionnelle, questionnaire de satisfaction, attestation de stage, support de cours remis à chaque participant.

Formateur expert dans son domaine d'intervention

Apports théoriques et exercices pratiques du formateur

Utilisation de cas concrets issus de l'expérience professionnelle des participants

Réflexion de groupe et travail d'échanges avec les participants

Pour les formations à distance : Classe virtuelle organisée principalement avec l'outil ZOOM. Assistance technique et pédagogique : envoi des coordonnées du formateur par mail avant le début de la formation pour accompagner le bénéficiaire dans le déroulement de son parcours à distance.

## PROGRAMME

### 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 DataWarehouse ?
- 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
- 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 distribuée ?
- 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 Mathplotlib

### ScikitLearn

- Régressions polynomiales
- 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 Matplotlib 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