

New

Analyse de données avec Python - Niveau Intermédiaire

Manipuler, visualiser, nettoyer et analyser statistiquement des données avec Python

 Présentiel ou en classe à distance



3 jours (21 h)

Prix inter : 2.490,00 € HT
Forfait intra : 7.690,00 € HT

Réf.: BD556

La formation **Analyse de données avec Python - Niveau Intermédiaire** permet de maîtriser les fondamentaux de l'**analyse de données** avec l'écosystème Python. Elle couvre la manipulation de données avec **pandas**, le nettoyage et la préparation des jeux de données, la création de visualisations avec **Matplotlib** et **Seaborn**, ainsi que les principales méthodes d'**analyse statistique**. Une formation essentielle pour réaliser des analyses exploratoires fiables, produire des visualisations pertinentes et exploiter efficacement les données en environnement professionnel.

A qui s'adresse cette formation ?



Pour qui

- Analystes, data scientists



Prérequis

- Connaissances des bases de Python et Excel
- **Disposez-vous des connaissances nécessaires pour suivre cette formation ? Testez-vous !**

Programme

1 - Environnement Python pour la data et son écosystème

- Présentation de l'écosystème Python dédié à l'analyse de données : Pandas, NumPy, Matplotlib, Seaborn, SciPy et Jupyter
- Prise en main de Jupyter Notebook et JupyterLab : cellules, kernels, commandes magiques, exécution et raccourcis
- Utilisation de NumPy : tableaux multidimensionnels ndarray, opérations vectorisées et broadcasting
- Bonnes pratiques pour structurer un notebook d'analyse de données

2 - Manipulation de DataFrames avec Pandas

- Création d'un DataFrame à partir d'un dictionnaire, d'une liste ou de fichiers CSV, Excel et JSON
 - Exploration des données avec les méthodes head, tail, info et describe, ainsi qu'avec les attributs shape et dtypes
 - Sélection des données par labels avec loc, par positions avec iloc et à l'aide de masques booléens
 - Filtrage, tri conditionnel, classement croissant ou décroissant et tri multicritère
 - Ajout et modification de colonnes : affectations directes, méthode apply et calculs vectorisés
 - Création et manipulation d'index hiérarchiques avec MultiIndex
- Atelier

Charger un fichier CSV de transactions commerciales dans un DataFrame Pandas

Explorer le jeu de données avec info, describe, head et value_counts afin d'identifier les colonnes et leurs types

Filtrer les transactions selon plusieurs conditions date, montant et catégorie avec des masques booléens

Créer une colonne calculée correspondant au taux de marge en utilisant une opération vectorisée

3 - Agrégations, jointures et restructuration avec Pandas

- Agrégation de données avec groupby : sum, mean, count, min, max et agrégations multiples avec agg
- Réalisation de jointures avec merge : inner, left, right et outer sur des clés simples ou composées
- Concaténation verticale ou horizontale de DataFrames avec concat et axis=0 ou axis=1
- Restructuration des données avec pivot, pivot_table, melt, stack et unstack
- Manipulation des dates : conversion au format datetime et extraction de l'année, du mois, du jour ou de la semaine
- Rééchantillonnage de séries temporelles avec resample par jour, semaine, mois ou année

Atelier

Charger deux DataFrames contenant des données de ventes et de clients

Fusionner les DataFrames avec merge en utilisant la clé client_id

Calculer le chiffre d'affaires par catégorie et par mois avec groupby et plusieurs fonctions d'agrégation

Présenter les résultats sous la forme d'un tableau croisé avec pivot_table

Rééchantillonner les ventes quotidiennes afin d'obtenir une agrégation hebdomadaire avec resample

4 - Nettoyage et préparation des jeux de données

- Détection des valeurs manquantes avec isna, isnull et notna, par colonne ou par ligne
- Traitement des valeurs manquantes avec dropna et fillna : constante, moyenne, médiane, propagation avant ffill ou arrière bfill
- Détection et suppression des doublons avec duplicated et drop_duplicates
- Conversion des types avec astype, to_datetime et to_numeric, en utilisant notamment le paramètre errors="coerce"
- Détection et traitement des valeurs aberrantes avec la méthode du z-score et la méthode de l'intervalle interquartile, avec visualisation par boxplot
- Encodage des variables catégorielles avec map, replace et get_dummies

Atelier

Auditer le jeu de données en recherchant les valeurs manquantes, les doublons et les incohérences de types

Produire un rapport synthétique sur la qualité des données

Imputer les valeurs manquantes selon une stratégie adaptée à chaque colonne : médiane pour une variable numérique et modalité majoritaire pour une variable catégorielle

Détecter et traiter les valeurs aberrantes de deux colonnes numériques avec la méthode de l'IQR

Encoder les variables catégorielles avec get_dummies

Exporter le DataFrame nettoyé au format CSV

5 - Visualisation des données avec Matplotlib

- Architecture de Matplotlib : Figure, Axes et Axis, et comparaison entre l'approche orientée objet et l'interface Pyplot
- Création de graphiques fondamentaux : courbes, nuages de points, diagrammes en barres et histogrammes
- Personnalisation des visualisations : titres, libellés, légendes, grilles, limites des axes et graduations
- Utilisation des couleurs, des styles et des palettes : couleurs nommées, codes hexadécimaux et colormaps
- Création de plusieurs graphiques avec plt.subplots et gestion de la taille avec figsize
- Export des figures aux formats PNG, PDF et SVG avec savefig et gestion de la résolution DPI

Atelier

Représenter l'évolution mensuelle du chiffre d'affaires avec une série temporelle et personnaliser le titre, les axes et la grille

Créer un histogramme et un boxplot afin d'analyser la distribution des montants

Construire un nuage de points représentant la relation entre le montant et la marge, avec une couleur par catégorie

Regrouper les graphiques dans une figure composée de quatre sous-graphiques organisés en grille 2 x 2

Exporter la figure obtenue au format PNG

6 - Analyse visuelle des données avec Seaborn

- Présentation de Seaborn : positionnement par rapport à Matplotlib et syntaxe orientée DataFrame
- Création de graphiques univariés avec histplot, kdeplot, boxplot et violinplot
- Création de graphiques bivariés avec scatterplot, regplot et jointplot
- Visualisation des variables catégorielles avec barplot, countplot et swarmplot
- Création de heatmaps pour représenter les matrices de corrélation et les tableaux croisés
- Exploration multivariée avec les grilles de graphiques FacetGrid et pairplot

Atelier

Calculer une matrice de corrélation entre les variables numériques et la représenter sous la forme d'une heatmap

Produire un pairplot des principales variables numériques en colorant les observations par catégorie

Comparer la distribution d'une variable numérique entre plusieurs catégories avec boxplot et violinplot

Visualiser la relation entre deux variables numériques avec regplot et représenter la régression linéaire

7 - Analyses statistiques fondamentales avec Python

- Calcul des statistiques descriptives : moyenne, médiane, écart-type, quantiles, skewness et kurtosis
 - Présentation des principales distributions : normale, uniforme et binomiale, avec identification visuelle et tests de normalité
 - Mesure des associations entre variables avec les coefficients de corrélation de Pearson, Spearman et Kendall
 - Principes des tests d'hypothèses : hypothèse nulle, p-value et seuil de signification
 - Test t de Student pour comparer les moyennes de deux échantillons
 - Test du khi-deux pour analyser l'indépendance entre deux variables catégorielles
- Atelier

Calculer les statistiques descriptives des variables numériques avec describe, skewness et kurtosis

Construire une matrice de corrélation et identifier les paires de variables fortement corrélées

Tester l'hypothèse d'une différence de moyenne entre deux groupes avec scipy.stats.ttest_ind

Tester l'indépendance entre deux variables catégorielles avec scipy.stats.chi2_contingency



Les objectifs de la formation

- Manipuler et transformer des données avec pandas
- Nettoyer et préparer des jeux de données pour l'analyse
- Réaliser des analyses statistiques descriptives avec Python
- Concevoir des visualisations de données avec Matplotlib et Seaborn.



Evaluation

- Pendant la formation, le formateur évalue la progression pédagogique des participants via des QCM, des mises en situation et des travaux pratiques. Les participants passent un test de positionnement avant et après la formation pour valider leurs compétences acquises.



Les points forts de la formation

- Une approche pédagogique fondée sur un cas d'étude fil rouge, permettant d'explorer un jeu de données réaliste, de son chargement jusqu'à la réalisation d'analyses statistiques et à l'interprétation des résultats
- Une formation couvrant l'ensemble des étapes d'un projet d'analyse de données avec Python : chargement, préparation, exploration, nettoyage, visualisation et analyses statistiques
- Des ateliers pratiques mettant en oeuvre les bibliothèques de référence de l'écosystème Python (pandas, NumPy, Matplotlib, Seaborn et SciPy) selon les bonnes pratiques actuelles de la data science



Dates et villes 2026 - Référence BD556



Dernières places disponibles



Session garantie

A distance

du 26 oct. au 28 oct.

du 30 nov. au 2 déc.

Paris

du 26 oct. au 28 oct.

du 30 nov. au 2 déc.