PYTHON & DATA : API, INTELLIGENCE ARTIFICIELLE

Introduction au machine learning avec Python

Apprenez le machine learning avec scikit-learn : préparation des données, régression, classification, validation croisée, boosting, clustering, pipelines, suivi avec MLflow et déploiement avec FastAPI.

DURÉE	TARIF	NIVEAU	LANGUE	GROUPE	FORMAT
5j (35h)	**2 070 € / participant** Inter-entreprise. Intra : sur devis, au plus 2 070 € / participant	**Avancé**	**Français**	**3-10**	**Formation** Présentiel ou à distance

1 PUBLIC VISÉ

- Développeurs Python qui veulent construire des modèles prédictifs de façon concrète
- Ingénieurs et analystes qui veulent passer de l’analyse de données à la modélisation prédictive

2 PRÉREQUIS

- Écrire des programmes Python avec des fonctions, des classes et des compréhensions de listes
- Calculer et lire des statistiques descriptives : moyenne, écart-type, corrélation
- Aucune connaissance en machine learning n’est requise

3 OBJECTIFS PÉDAGOGIQUES

- Explorer et nettoyer un jeu de données avec numpy et pandas : sélection, valeurs manquantes, regroupements
- Préparer des données pour l’apprentissage : séparation entraînement et test, encodage, normalisation, nouvelles variables
- Entraîner et évaluer des modèles de régression (linéaire, polynomiale, Ridge, Lasso) avec MAE, RMSE et R^2
- Entraîner et évaluer des modèles de classification (régression logistique, arbres, forêts aléatoires, k-NN, SVM) avec la matrice de confusion, la précision, le rappel et le F1
- Comparer des modèles par validation croisée et optimiser leurs hyperparamètres, y compris sur des classes déséquilibrées
- Entraîner un modèle de gradient boosting et expliquer ses prédictions : importance des variables, importance par permutation, dépendance partielle
- Segmenter des données avec k-means et DBSCAN, et les représenter en deux dimensions avec une ACP
- Construire un pipeline scikit-learn reproductible, le sauvegarder et suivre les expériences avec MLflow
- Exposer un modèle entraîné par une API FastAPI validée par Pydantic et conteneurisée avec Docker

4 PROGRAMME DÉTAILLÉ

Jour 1 (7h) : Données et premiers pas en machine learning

Module 1 : numpy et pandas (3h30)

numpy (1h30)

- Tableaux numpy : création, types, dimensions
- Opérations vectorielles et broadcasting ; indexation et filtres
- reshape, transpose, produit matriciel ; fonctions statistiques
- Exercice d'entraînement : construire une matrice de variables à partir de données brutes

pandas (2h)

- Series et DataFrame ; read_csv et read_excel
- loc, iloc, filtres ; valeurs manquantes
- apply, groupby, merge, pivot
- TP évalué : nettoyer et explorer un jeu de données réel : valeurs manquantes, types, regroupements et jointure avec une table de référence

Module 2 : Visualiser et préparer les données (3h30)

Visualiser les données (1h30)

- matplotlib : courbes, histogrammes, nuages de points
- seaborn : carte de corrélation, boîtes à moustaches, pairplot
- Exercice d'entraînement : tableau exploratoire d'un jeu de données

Le machine learning et la préparation des données (2h)

- Apprendre à partir des données ; supervisé et non supervisé ; la place du deep learning
- L'API de scikit-learn : fit, predict, transform
- Séparation entraînement et test, stratification ; éviter la fuite de données
- Encodage (OneHotEncoder, OrdinalEncoder) et normalisation (StandardScaler, MinMaxScaler)
- TP évalué : préparation complète d'un jeu de données de prix immobiliers, prête pour l'entraînement

Jour 2 (7h) : Régression et classification

Module 3 : Régression (3h30)

Régression linéaire (1h30)

- Coefficients, ordonnée à l'origine, hypothèses
- MAE, MSE, RMSE, R^2 ; analyser les résidus
- Exercice d'entraînement : prédire le prix d'un bien

Régression polynomiale et régularisation (2h)

- Régression polynomiale
- Surapprentissage ; Ridge, Lasso, ElasticNet
- TP évalué : comparer régression linéaire, polynomiale, Ridge et Lasso sur un jeu de données bruité

Module 4 : Classification (3h30)

Régression logistique et métriques (1h30)

- La fonction sigmoïde ; classification binaire et multiclasse
- Exactitude, précision, rappel, F1 ; matrice de confusion
- Exercice d'entraînement : clients qui risquent de partir (churn)

Arbres, forêts aléatoires, k-NN et SVM (2h)

- Arbres de décision et leurs hyperparamètres ; forêts aléatoires
- k plus proches voisins ; SVM et noyaux

- TP évalué : classer des e-mails en spam ou non avec quatre modèles, et choisir la métrique selon le coût des erreurs

Jour 3 (7h) : Évaluer, optimiser et expliquer

Module 5 : Validation croisée et optimisation (3h30)

Validation croisée (1h30)

- Score d'entraînement et score de test
- Validation croisée k-fold ; version stratifiée
- Exercice d'entraînement : comparer trois modèles par validation croisée

Métriques avancées et hyperparamètres (2h)

- Courbe ROC et AUC ; courbe précision-rappel
- Classes déséquilibrées : class_weight, sur-échantillonnage
- GridSearchCV et RandomizedSearchCV
- TP évalué : modèle de détection de fraude sur des classes déséquilibrées, optimisé et évalué avec la courbe adaptée

Module 6 : Boosting et interprétabilité (3h30)

Gradient boosting (1h30)

- Le boosting : apprendre des erreurs des modèles précédents
- HistGradientBoosting de scikit-learn ; XGBoost et LightGBM
- Exercice d'entraînement : comparer forêt aléatoire et boosting

Expliquer un modèle (2h)

- Importance des variables ; importance par permutation
- Graphiques de dépendance partielle
- TP évalué : modèle d'octroi de crédit par boosting, et explication de ses décisions

Jour 4 (7h) : Clustering et pipelines

Module 7 : Clustering et réduction de dimension (3h30)

Clustering (1h30)

- k-means : centroïdes et convergence ; méthode du coude et score de silhouette
- DBSCAN : regrouper par densité
- Exercice d'entraînement : segmenter des points simulés

Réduction de dimension (2h)

- Le fléau de la dimension
- ACP : variance expliquée, nombre de composantes
- TP évalué : segmenter des clients par comportement d'achat, justifier le nombre de groupes et les représenter en deux dimensions

Module 8 : Pipelines et suivi des expériences (3h30)

Pipelines scikit-learn (1h30)

- Pipeline : enchaîner préparation et modèle
- ColumnTransformer : une préparation par type de colonne
- Rechercher les hyperparamètres sur un pipeline complet
- Exercice d'entraînement : pipeline du jeu de données immobilier

Sauvegarder et suivre (2h)

- Sauvegarder un pipeline avec joblib
- MLflow : suivre paramètres, métriques et modèles
- TP évalué : pipeline complet sauvegardé, rechargé et suivi dans MLflow sur trois essais

Jour 5 (7h) : Déploiement et projet de synthèse

Module 9 : Déployer un modèle avec FastAPI (3h)

Exposer un modèle par une API (1h30)

- Charger le pipeline au démarrage (lifespan)
- Endpoint /predict ; valider les entrées avec Pydantic
- Renvoyer la prédiction et sa probabilité ; documentation Swagger
- Exercice d'entraînement : API de prédiction de prix

Conteneuriser (1h30)

- Dockerfile avec le modèle embarqué ; Docker Compose
- Journaliser les entrées et les sorties pour repérer une dérive
- TP évalué : API de prédiction conteneurisée avec Docker, entrées validées et documentation Swagger

Module 10 : Projet de synthèse, évalué (4h)

Réalisation (2h30)

- Projet de synthèse, réalisé individuellement : problème de classification ou de régression sur un jeu de données réel : exploration, pipeline, au moins trois modèles comparés par validation croisée, optimisation, explication des prédictions, API FastAPI

Revue et retour (1h30)

- Revue collective : choix des modèles, métriques, interprétation
- Retour individuel du formateur sur le projet rendu

Le contenu peut être ajusté selon le niveau, les prérequis et les besoins des participants.

5 COMPÉTENCES VISÉES

- Préparer des données pour le machine learning
- Entraîner et évaluer des modèles supervisés
- Comparer, optimiser et expliquer des modèles
- Analyser des données sans étiquette
- Industrialiser un modèle : pipeline, suivi des expériences, API

6 MODALITÉS PÉDAGOGIQUES

Formation délivrée en présentiel ou à distance, en classe virtuelle. Le formateur alterne entre méthode démonstrative, méthode interrogative (analyse des résultats des modèles et des choix d'algorithmes) et méthode active (exercices d'entraînement, TP évalués et projet de synthèse).

7 MOYENS PÉDAGOGIQUES ET TECHNIQUES

- Support de cours numérique mis à disposition des apprenants
- Notebooks Jupyter de démonstration avec exercices et corrections par module ; jeux de données réels
- Environnement de développement : Jupyter Lab ou VS Code + Python 3.12 ou plus récent + pandas, scikit-learn, XGBoost, MLflow, FastAPI et Docker
- Pour la classe virtuelle : visioconférence (Microsoft Teams ou équivalent), partage d'écran, chat en direct
- Supports remis par e-mail et sur la plateforme pédagogique LaPolaris (ressources, émargement)

En formation intra-entreprise sur le site du client, celui-ci s'assure et s'engage à disposer de toutes les ressources matérielles nécessaires à son bon déroulement (salle, postes de travail, logiciels, accès à Internet), conformément aux prérequis indiqués dans le programme et aux moyens prévus dans la convention.

8 MODALITÉS D'ÉVALUATION

- Questionnaire de positionnement individuel renseigné par chaque participant avant le démarrage de la formation, analysé par le formateur et donnant lieu, le cas échéant, à un ajustement du déroulé
- Exercices d'entraînement réalisés par chaque participant sur son poste, corrigés en séance
- Évaluation des acquis par les TP évalués, réalisés individuellement sur poste : un par module, puis le projet de synthèse décrit dans le dernier module du programme ; chaque objectif est évalué dans au moins un TP
- Grille de suivi renseignée par le formateur pour chaque participant et pour chacun des objectifs de l'action, au regard des critères d'évaluation ci-dessous
- Questionnaire d'appréciation renseigné par les participants en clôture de formation, et enquête à froid dans un délai de trois à six mois

9 CRITÈRES D'ÉVALUATION

- Données chargées et explorées ; valeurs manquantes traitées ; types corrigés ; regroupements et jointures justes
- Séparation entraînement et test sans fuite de données ; variables catégorielles encodées ; normalisation ajustée sur les seules données d'entraînement
- Modèles de régression entraînés et comparés avec MAE, RMSE et R^2 ; surapprentissage repéré et réduit par la régularisation
- Modèles de classification entraînés ; matrice de confusion lue ; métrique choisie selon le coût des erreurs
- Comparaison par validation croisée ; hyperparamètres optimisés ; classes déséquilibrées traitées et évaluées avec la bonne courbe
- Modèle de boosting comparé aux autres ; variables les plus influentes identifiées et expliquées
- Segments obtenus avec un nombre de groupes justifié (coude ou silhouette) ; projection en deux dimensions lisible
- Pipeline avec ColumnTransformer qui s'entraîne et prédit en une seule commande ; modèle sauvegardé et rechargé ; expériences suivies dans MLflow
- Endpoint /predict qui renvoie la prédiction pour des données validées ; documentation Swagger ; API lancée avec Docker

10 INDICATEURS DE RÉSULTATS

Les indicateurs de résultats de cette formation (taux de satisfaction et taux de réussite) seront publiés sur sa fiche en ligne à l'issue des premières sessions réalisées.

11 MODALITÉS DE VALIDATION

Une attestation de fin de formation est remise à chaque participant à l'issue du parcours. Elle mentionne **l'intitulé** et la **nature de l'action**, sa **durée** en heures, sa **période de réalisation**, **les objectifs de l'action** et le **niveau d'atteinte** de chacun d'eux, exprimé en trois valeurs : **atteint**, **partiellement atteint**, **non atteint**. **Une attestation d'assiduité**, établie à partir des émargements, est délivrée dans les mêmes conditions.

Formation non certifiante : elle ne délivre pas de certification enregistrée au RNCP ou au Répertoire spécifique.

12 SUIVI ET ACCOMPAGNEMENT

- Émargement électronique signé par chaque participant et par le formateur à chaque demi-journée, en présentiel comme à distance
- Traçabilité des activités pédagogiques réalisées
- Attestation d'assiduité délivrée en fin de formation
- Suivi individuel via les QCM de fin de journée, les exercices corrigés et les TP évalués

13 CONDITIONS D'ACCÈS

Formation accessible sans prérequis administratif particulier. Cette formation n'est pas éligible aux financements publics ou mutualisés, notamment au compte personnel de formation et à une prise en charge par un opérateur de compétences.

14 DÉLAIS D'ACCÈS

Inscription possible jusqu'à 15 jours ouvrés avant le début de la session pour permettre la préparation logistique et, si besoin, l'adaptation du contenu au contexte client. Pour toute demande spécifique, nous contacter directement.

ACCESSIBILITÉ · HANDICAP

Nos formations peuvent être accessibles aux personnes en situation de handicap. La question d'un besoin d'aménagement (matériel, temporel ou pédagogique) est posée lors de l'analyse du besoin et du questionnaire de positionnement ; vous pouvez aussi nous le signaler à tout moment, y compris pendant la formation, afin d'étudier les adaptations possibles. Référent handicap : contact@lapolaris.fr

LaPolaris
TÉL. +33762584798
EMAIL contact@lapolaris.fr
WEB lapolaris.fr