

Sessions garanties disponibles

Formation PyTorch

Réf. SF33829

Devenez opérationnel en Deep Learning avec PyTorch, du tensor au modèle déployable !

3 j Durée	2 550 € HT Inter / participant	4 950 € HT Intra (≤12 pers.)	14 Sessions à venir
--------------	-----------------------------------	---------------------------------	------------------------

Objectifs pédagogiques

- ✓ Configurer un environnement de développement PyTorch et manipuler des tenseurs, Datasets et DataLoader pour préparer des jeux de données
- ✓ Implémenter, entraîner et sauvegarder des architectures de réseaux de neurones (MLP et CNN) avec torch.nn et torch.optim pour des tâches de classification supervisée
- ✓ Appliquer des techniques de régularisation et d'optimisation afin d'améliorer la métrique de validation d'un modèle PyTorch existant
- ✓ Mettre en oeuvre un pipeline complet de transfert d'apprentissage en réutilisant un modèle pré-entraîné PyTorch pour un cas de vision par ordinateur ou de NLP et comparer quantitativement ses performances à un modèle entraîné from scratch
- ✓ Intégrer un modèle PyTorch entraîné dans un script ou notebook réutilisable et le versionner pour pouvoir le rejouer sur un nouveau jeu de données en contexte projet

Public visé & prérequis

Public viséData analysts, Data scientists, Développeurs, Ingénieurs Big Data

PrérequisMaîtrise opérationnelle de Python, notions en machine learning supervisé, connaissances de base en algèbre linéaire et en calcul différentiel, familiarité avec l'écosystème PyData et l'usage de notebooks Jupyter.

Programme détaillé

Prise en main de PyTorch et de l'environnement de développement

Installation de Python, PyTorch et bibliothèques associées (torchvision, torchaudio, numpy, pandas)
Organisation d'un projet : environnement virtuel, structure de répertoires, gestion des dépendances
Notions fondamentales : tenseurs, types, device (cpu/cuda), opérations de base et broadcasting
Présentation de l'auto-grad, du graphe de calcul et de la gestion des gradients
Création de Dataset personnalisés et usage des DataLoader (batchs, shuffle, num_workers)
Présentation des supports pédagogiques : notebooks Jupyter, Colab et dépôt de code fourni

Activités pratiques :

TP : installation guidée et test de l'environnement (import torch, détection du GPU, premier tensor)
Exercice : manipulation de tenseurs (création, reshape, opérations mathématiques, passage CPU/GPU)
Exercice : création d'un Dataset + DataLoader sur un jeu de données tabulaire open source et sur MNIST

Réseaux de neurones fully connected et entraînement de modèles

Rappels sur les réseaux de neurones feed-forward : couches linéaires, fonctions d'activation, fonctions de coût
Architecture d'un MLP dans PyTorch : nn.Linear, nn.ReLU, nn.Sequential vs classe personnalisée
Boucle d'entraînement standard : forward pass, calcul de la loss, backward, step de l'optimiseur
Séparation train/validation/test, gestion des epochs et suivi des métriques
Utilisation de torch.save et torch.load pour sauvegarder et recharger poids et modèles
Structuration d'un notebook d'expérimentation (config, données, modèle, entraînement, résultats)

Activités pratiques :

TP guidé : implémentation d'un MLP pour prédire une classe sur un dataset tabulaire (ex : churn / crédit)

Exercice : ajout de métriques personnalisées, traçage des courbes de perte et d'accuracy
Mini-projet : sauvegarde du meilleur modèle, rechargement et évaluation sur un jeu de test séparé

Convolutional Neural Networks, vision par ordinateur et optimisation des modèles

Rappels sur les convolutions, le pooling et la hiérarchie de caractéristiques en vision par ordinateur
Construction d'un CNN avec `nn.Conv2d`, `nn.MaxPool2d`, `nn.Flatten`, etc. et utilisation de `torchvision.datasets`
Optimisation : choix de la fonction de coût, des optimiseurs (SGD, Adam), learning rate et batch size
Régularisation : dropout, data augmentation, weight decay, early stopping, schedulers de taux d'apprentissage
Suivi et comparaison des expériences (logs, tableaux de résultats, éventuelle intégration avec TensorBoard)
Évaluation avancée : matrice de confusion, erreurs typiques, interprétation qualitative des prédictions

Activités pratiques :

TP : entraînement d'un CNN de base sur MNIST ou Fashion-MNIST avec un jeu de paramètres par défaut
Atelier : application progressive de dropout, data augmentation et scheduler, comparaison des métriques
Exercice : rédaction d'un court rapport d'expérimentation (config, résultats, choix retenus) dans le notebook

Données séquentielles, NLP et introduction aux Transformers

Pipeline NLP : collecte, nettoyage, tokenisation, encodage (index, embeddings) et gestion des longueurs de séquence
Utilisation de `torch.nn.Embedding` et des modules RNN, LSTM ou GRU pour modéliser des séquences
Gestion des `DataLoader` pour des séquences (padding, masks éventuels)
Présentation des mécanismes d'attention et de la structure générale d'un Transformer (sans entrer dans tous les détails mathématiques)
Introduction à Hugging Face Transformers ou équivalent pour l'inférence avec un modèle pré-entraîné (ex : DistilBERT)
Comparaison de performances et de coûts (temps d'entraînement, ressources) entre modèle LSTM maison et modèle pré-entraîné

Activités pratiques :

TP : construction d'un pipeline NLP complet pour de la classification de sentiment (tweets ou reviews) avec un LSTM PyTorch
Exercice : mise en place d'un script d'inférence utilisant un modèle Transformer pré-entraîné pour la même tâche
Comparaison des métriques et discussion des choix d'architecture en groupe (tableau de synthèse dans le notebook)

Transfert d'apprentissage et pipeline d'inférence réutilisable

Principe du transfert d'apprentissage : freeze/unfreeze de couches, réutilisation de features pré-apprises
Utilisation de modèles pré-entraînés de `torchvision` (ex : ResNet) ou d'un modèle NLP pré-entraîné en fine-tuning léger
Construction d'un pipeline d'inférence : chargement du modèle, prétraitement des données, prédictions et post-traitement
Sauvegarde des artefacts : poids du modèle, configuration, encoders, normalisation
Introduction aux bonnes pratiques de versionning (Git, branches simples), tracking des expériences et gestion de `requirements.txt/conda.yaml`
Bonnes pratiques de structuration d'un projet ML en vue d'une future mise en production (séparation entraînement / inférence, scripts CLI ou squelette d'API)

Activités pratiques :

Mini-projet : fine-tuning d'un modèle pré-entraîné (vision ou NLP) sur un petit dataset métier, comparaison avec un modèle from scratch
TP : création d'un notebook ou script d'inférence autonome prenant en entrée un fichier (image ou texte) et renvoyant une prédiction
Exercice de clôture : création d'un dépôt Git local du projet de formation (code, modèles sauvegardés, README de reprise en main)

Prochaines sessions

DATE	LIEU	STATUT
09/11/2026	à distance	Sur confirmation
07/12/2026	à distance	Sur confirmation
27/01/2027	Lyon	Sur confirmation
27/01/2027	Nantes	Sur confirmation
27/01/2027	Toulouse	Sur confirmation
27/01/2027	à distance	Sur confirmation

Pourquoi choisir Sparks Formation ?

20 ans d'expertise IT, des formateurs experts et des sessions garanties contre l'annulation.

20 ans
d'expertise

1 068+
formations

11 178
apprenants formés/an

+67
Net Promoter Score

Avis des participants

4,9/5 — 21 avis

Anthony G. — Formation très bien organisée, très claire Alison explique très clairement et est très à l'écoute, avec toujours une approche pratique essentielle. je n'avais jamais fait de Python et...

François H. — François-Marie est un excellent formateur ! Un grand merci à lui d'avoir été aussi pédagogue et clair dans ses explications du début à la fin de la formation,...

Votre formateur

Cyril V. — Formateur Data Science & Machine Learning

Cyril intervient sur des formations portant sur la Data Science et le Machine Learning. Parmi les formations qui lui sont confiées figurent notamment « MLOps » et « PyTorch »....

Ils nous font confiance

Cette année notre filière Data Science & Machine Learning a accompagné plus de 70 entreprises dont :

SOPRA STERIA RTE AMADEUS SAS CEGEDIM LBC FRANCE SNCF - E.SNCF SOLUTIONS AMADEUS DATA PROCESSING GMBH CSTB GERS SA
ADENTIS +60 autres

Financement & certifications

Cette formation est éligible aux financements OPCO et FNE-Formation. Nos équipes vous accompagnent dans le montage de votre dossier.



Réservez votre place dès maintenant

20 ans d'expertise · 1 000+ formations · NPS +67 · réponse sous 24h ouvrées

0 805 950 800

demande@sparks-formation.com