Cycle certifiant Data scientist

DS040

Durée: 20 jours

11 020 €

Public :

Chefs de projet, data scientists, statisticiens, développeurs.

Objectifs :

Savoir identifier et mettre en oeuvre les outils adaptés à l'analyse de données. Savoir définir les étapes de préparation des données, connaître les algorithmes de Machine Learning, les mettre en oeuvre avec des outils comme scikit-learn ou Spark ML. Savoir mettre en oeuvre TensorFlow pour de l'apprentissage machine, connaitre les APIs disponibles pour réaliser des modèles fiables et efficaces. Comprendre les apports du deep learning et de l'IA, l'architecture et les différents types de réseaux de neurones et les mettre en pratique avec keras.

Connaissances préalables nécessaires :

Connaissance des bases des systèmes d'information, et notions de calculs statistiques.

Programme :

IA Etat de l'art

IA Etat de l'art

IA Etat de l'art

Définitions et positionnement IA, deep learning et Machine Learning
Les apports du deep learning, état de l'art.
Outils disponibles. Exemple de projets.
Exemples, domaines d'application. Présentation de deepmind.
Démonstrations avec OpenAI (ChatGPT), Google Gemini, AWS

Connaître les principales solutions, outils et technologies déployés dans un projet d’IA

Outils DeepLearning de haut niveau distribués : Keras/TensorFlow.
Non distribués : PyTorch, Lasagne

Identifier les clés de réussite d’une solution d’Intelligence Artificielle

Préparation des données, régularisation, normalisation, extraction des caractéristiques.
Optimisation de la politique d'apprentissage.
Exploitation des modèles, mise en production. TensorFlow Hub. Serving.
Visualiser les reconstructions.

Atelier : mise en place d'un serveur de modèles et d'une application tf-lite

Mise en évidence des problèmes de convergence et du vanishing gradient.
Les erreurs d'architecture. Comment distribuer un réseau de neurones.
Les limites du DeepLearning : imiter/créer. Cas concrets d'utilisation.

Appréhender les enjeux juridiques et éthiques de l'IA

Propriété de la donnée, environnement juridique du traitement, sécurité.
Notion de loi extra-territoriales, champs d'application.
Impact des choix technologiques en matière d'analyse et de stockage de données.

Comprendre les applications de l'IA à différents domaines

Santé, industrie, finance.
Prévision : prospectives, gestion des stocks, négociations
Transformation des métiers : automatisation de tâches, robotique, refonte des modes de fabrication
L'IA au service de la protection des données

Atelier : Mise en oeuvre sur cloud AutoML : langages naturels, traduction, reconnaissance d'images, ...

Appréhender les concepts de Machine Learning et Deep Learning

Les réseaux de neurones : principe, différents types de réseaux de neurones (artificiels, convolutifs, récurrents, ...)
Fonctionnement d'un réseau de neurones. Comprendre le fonctionnement de l'apprentissage d'un réseau de neurones.
Apprendre à lire une courbe d'apprentissage.

Atelier : Comparaison de courbes d'apprentissage avec TensorFlow sur plusieurs paramètres.

IA Deep Learning

IA Deep Learning

IA Deep Learning

Définitions et positionnement IA, deep learning et Machine Learning
Les apports du deep learning, état de l'art.
Outils disponibles. Exemple de projets.
Exemples, domaines d'application. Présentation de deepmind
Outils DeepLearning de haut niveau : Keras/TensorFlow, Caffe/PyTorch, Lasagne.

Atelier : Mise en oeuvre sur cloud AutoML : langages naturels, traduction, reconnaissance d'images, ...

Appréhender les bases théoriques et pratiques d'architecture et de convergence de réseaux de neurones

Fonctionnement d'un réseau de neurones. Comprendre le fonctionnement de l'apprentissage d'un réseau de neurones.
Comprendre la rétro-propagation de l'erreur et la convergence.
Comprendre la descente de gradient. Les fonctions d'erreur : MSE, BinaryCrossentropy, et les optimiseurs SGD, RMSprop, Adam.
Définitions : couche, epochs, batch size, itérations, loss, learning rate, momentum.
Optimiser un entraînement par découpage d'entraînements peu profonds.
Comprendre le principe des hyper-paramètres. Choix des hyper-paramètres.

Atelier : construire un réseau capable de reconnaître une courbe

Connaitre les briques de base du Deep Learning : réseaux de neurones simples, convolutifs et récursifs

Les réseaux de neurones : principe, différents types de réseaux de neurones (artificiels, convolutifs, récurrents, ...)
Les différentes formes de réseaux : MultiLayer Perceptron FNN/MLP, CNN.
Couches d'entrée, de sortie, de calcul.
Fonctionnement d'une couche de convolution. Définitions : kernel, padding, stride. Fonctionnement d'une couche de Pooling.
APIs standard, modèles d'apprentissage
Apprendre à lire une courbe d'apprentissage.

Atelier : Comparaison de courbes d'apprentissage avec TensorFlow sur plusieurs paramètres.

Les modèles de DeepLearning pour Keras : Xception, Inception, ResNet, VGG, LeNet.

Atelier : Construction d'un réseau de neurones de reconnaissance d'images

Appréhender les modèles plus avancés : auto-encodeurs, gans, apprentissage par renforcement

Représentations des données. Bruits. Couches d'encodage : codage entier, One-hot, embedding layer.
Notion d'autoencodeur. Autoencodeurs empilés, convolutifs, récurrents.
Comprendre les réseaux antagonistes génératifs (GANS) et leur limites de convergences. Apprentissage par transfert.
Comment optimiser les récompenses?
Évolutions vers les GRU (Gated Recurrent Units) et LSTM (Long Short Term Memory).
Traitement NLP : encodage des caractères et des mots, traduction.

Atelier : entraînement d'un autoencodeur variationnel sur un jeu d'images

Comprendre les méthodologies de mise en place de réseaux de neurones

Préparation des données, régularisation, normalisation, extraction des caractéristiques.
Optimisation de la politique d'apprentissage.
Exploitation des modèles, mise en production. TensorFlow Hub. Serving.
Visualiser les reconstructions.

Atelier : mise en place d'un serveur de modèles et d'une application tf-lite

Comprendre les points forts et les limites de ces outils

Mise en évidence des problèmes de convergence et du vanishing gradient.
Les erreurs d'architecture. Comment distribuer un réseau de neurones.
Les limites du DeepLearning : imiter/créer. Cas concrets d'utilisation.
Introduction aux machines quantiques.

DataScience

DataScience

DataScience

Définition. De la statistique à l'apprentissage automatique.
Apprentissage automatique : comprendre ou prédire?
Besoin en puissance de calcul et de stockage.
Intégration de l'apprentissage automatique dans les fermes de Big Data.
Les valeurs d'observation, et les variables cibles.
Ingénierie des variables.

Appréhender les enjeux de l'utilisation du Machine Learning, incluant les bénéfices attendus et des exemples d'usage

Comment automatiser les processus métier. Attentes. Création de valeur à partir de la donnée. Problème du ratio pertinence/volume.
Les risques et écueils. Importance de la préparation des données. L'écueil du "surapprentissage".
Les erreurs d'architecture à éviter.

Atelier : mise en évidence d'erreurs d'apprentissage sur des données non qualifiées.

Modélisation automatique. Le rôle du data scientist.

Atelier : démonstration de reconnaissance d'images.

Identifier le positionnement du Machine Learning dans la chaîne de traitement de la donnée

Le pattern MapReduce. Exemple d'utilisation.
Gouvernance des données. Qualité des données.
Transformation de l'information en donnée. Qualification et enrichissement.
Sécurisation et étanchéité des lacs de données.
Flux de données et organisation dans l'entreprise. De la donnée maître à la donnée de travail. MDM.
Mise en oeuvre pratique des différentes phases :
nettoyage,enrichissement,organisation des données.
Zoom sur les données : format, volumes, structures.
Zoom sur les requêtes, attentes des utilisateurs.
Etapes de la préparation des données.
Définitions, présentation du data munging

Connaitre les outils et les acteurs leaders du marché

Comparatifs des outils d'apprentissage automatique. Les outils en mode local, en mode distribué.
Les acteurs. Leurs outils.

Atelier : utilisation de scikit learn et de SparkML. Comparatif.

Apprentissage profond : introduction aux réseaux de neurones.
Réseaux de neurones à convolution. Modèles de CNN.
L'approche du Deep Learning. Deeplearning4j sur Spark. TensorFlow sur rig, sur Spark.

Atelier : mise en oeuvre d'une reconnaissance automatique avec TensorFlow

Découvrir les principaux algorithmes et la démarche projet à appliquer selon les cas d'usages en entreprise

Apprentissage supervisé/non supervisé, classification ou régression.
Algorithme paramétrique ou non-paramétrique, linéaire ou non-linéaire.
Les méthodes : apprentissage supervisé et non supervisé
Classification des données,
Algorithmes : régression linéaire, k-moyennes, k-voisins, classification naïve bayésienne, arbres de décision, forêts aléatoires, ...

Atelier : classification automatique d'un jeu de données à partir d'une régression logistique

Création de jeux d'essai, entraînement et construction de modèles.
Prévisions à partir de données réelles. Mesure de l'efficacité des algorithmes. Courbes ROC.
Parallélisation des algorithmes. Choix automatique.

Atelier : Mise en évidence des erreurs d'apprentissage en fonction des hyper-paramètres

Identifier les clés de réussite d'un projet intégrant du Machine Learning

Choix des architecture. Comment définir le besoin métier?
Extraction et organisation des classes de données.
Applications aux fermes de calculs distribués. Problématiques induites. Approximations. Précision des estimations.
Analyse factorielle.
Visualisation des données. L'intérêt de la visualisation. Outils disponibles.

Python pour la DataScience

Python pour la DataScience

Positionnement Python

Besoins des data-scientists : calculs, analyse d'images, machine learning, interface avec les bases de données
Apports de python : grande variété d'outils, expertise dans le domaine du calcul scientifique
Tour d'horizon des outils:
pandas, pyarrow, agate, bokeh, scikit-learn, pybrain, tensorflow, keras, mxnet, caffe, Pytorch

Calculs et graphiques

NumPy : Base du calcul sur des tableaux
SciPy : Scientific Tools for Python, couche scientifique
Manipulation de tableaux, fonctions mathématiques.
Représentation graphique avec basemap et matplotlib.

Atelier : Mise en oeuvre de SciPy/NumPy : manipulation d'images, détection de contours

Être capable d'extraire des données d'un fichier

Pandas : manipulation de tables de données. Notion de dataframe.
Manipulation de données relationnelles
Tableaux avec Pandas: indexation, opérations, algèbre relationnelle
Stockage dans des fichiers: CSV, JSon
Comparaison et performances Pandas / pyarrow / NumPy

Atelier : construction d'ETL de base entre json et csvkagglt.com,

Comprendre les mécanismes d'interconnexion aux bases de données

Définitions : pilotes, connexions, curseurs, CRUD, transactions
Les pilotes : postgresql, mysql, mariadb, ... Présentation de sql-alchemy
Opérations : gestion du curseur, chargement de données, insertion et modification d'enregistrements

Atelier : mise en oeuvre avec postgresql. Construction d'ETL SQL/json

Comprendre les principaux outils de traitement et d'analyse de données pour Python

Présentation des outils d'apprentissage Python : scikit-learn, pybrain, TensorFlow/keras, mxnet, caffe

Atelier : mise en oeuvre de scikit-learn

Créer des sélections et des classements dans de grands volumes de données pour dégager des tendances

Présentation de pyspark
Machine learning et deep learning : les solutions Python,
TensorFlow : principe de fonctionnement, plateformes supportées, distribution

Sites de références data-sciences

Ressources d'apprentissage, datasets, modèles de données pré-entrainés, etc ..
Présentation de : kaggle.com, data-puzzles.com, huggingface.co

Optimisation des développements

Tour d'horizon des outils actuels et futurs:
Jupyter notebook, Aide à la vérification de code, respect des recommandations PEP8 :
exemples avec pydecodestyle, Pylint, Black
Analyse et production de code informatique avec une IA.
Génération de code avec OpenAI : démonstrations ChatGPT, apports, bonnes pratiques.

Atelier : utilisation de la génération de code et de snippets Python avec ChatGPT

Spark Machine Learning

Spark Machine Learning

Spark Machine Learning

Rappels sur Spark : principe de fonctionnement, langages supportés.

DataFrames

Objectifs : traitement de données structurées. L'API Dataset et DataFrames
Optimisation des requêtes. Mise en oeuvre des Dataframes et DataSet.
Chargement de données, pré-traitement : standardisation, transformations non linéaires, discrétisation
Génération de données.

Traitements statistiques de base

Introduction aux calculs statistiques. Paramétrisation des fonctions.
Applications aux fermes de calculs distribués. Problématiques induites. Approximations. Précision des estimations.
Exemples sur Spark : calculs distribués de base : moyennes, variances, écart-type, asymétrie et aplatissement (skewness/kurtosis)

Machine Learning

Apprentissage automatique : définition, les attentes par rapport au Machine Learning
Les valeurs d'observation, et les variables cibles. Ingénierie des variables.
Les méthodes : apprentissage supervisé et non supervisé. Classification, régression.
Fonctionnalités : Machine Learning avec Spark, algorithmes standards, gestion de la persistence, statistiques.

Mise en oeuvre sur Spark

Mise en oeuvre avec les DataFrames.
Algorithmes : régression linéaire, k-moyennes, k-voisins, classification naïve bayésienne, arbres de décision, forêts aléatoires, etc ...
Création de jeux d'essai, entraînement et construction de modèles.
Prévisions à partir de données réelles.

Atelier : régression logistiques, forêts aléatoires, k-moyennes.

Recommandations, recommendForAllUsers(), recommendForAllItems();

Modèles

Chargement et enregistrement de modèles.
Mesure de l'efficacité des algorithmes. Courbes ROC. MulticlassClassificationEvaluator().
Mesures de performance. Descente de gradient.
Modification des hyper-paramètres.
Application pratique avec les courbes d'évaluations.

Spark/GraphX

Gestion de graphes orientés sur Spark
Fourniture d'algorithmes, d'opérateurs simples pour des calculs statistiques sur les graphes

Atelier : exemples d'opérations sur les graphes.

IA

Introduction aux réseaux de neurones.
Les types de couches : convolution, pooling et pertes.
L'approche du Deep Learning avec Spark. Deeplearning4j sur Spark.

Tensorflow/Keras

Tensorflow/Keras

TensorFlow

Introducion au traitement d'images et à l'apprentissage automatique.
Présentation de Keras, PyTorch et OpenCV.

Le projet Tensorflow

Historique , fonctionnalités
Architecture distribuée, plateformes supportées

Premiers pas avec TensorFlow

Principe des tenseurs, caractéristiques d'un tenseur: type de données, dimensions
Définition de tenseurs simples,
Gestion de variables et persistance,
Représentation des calculs et des dépendances entre opérations par des graphes

Optimisation des calculs

Calculs distribués : différents types de stratégies (synchrone ou asynchrone),
avec stockage centralisé des données ou dupliqué sur différents cpu
Distribution sur des GPUs
Utilisation de TPUs
Travaux pratiques sur une plateforme multi-GPU (RIG)

Présentation des RN

Principe des réseaux de neurones
Différents types de couches: denses, convolutions, activations
Fonctionnement des réseaux de neurones convolutifs (CNN).
Descente de gradient
Multi-Layer Perceptron

Mise en oeuvre avec Keras

Conception d'un réseau de neurones
Création et entraînement d'un modèle CNN simple avec Keras.
Classification d'images avec Keras
Notion de classification, cas d'usage
Architectures des réseaux convolutifs, réseaux ImageNet
RCNN et SSD
Démonstrations sur les convolutions

Optimisation d'un modèle

Visualisation avec Tensorboard
Optimisation des couches de convolutions
Choix des hyper-paramètres avec Keras et Keras Tuner
Utilisation de checkpoints

Détection d'Objets avec OpenCV et IA

Principes de la détection d'objets.
Les différents types de modèles de détection d'objets (classificateurs en cascade, YOLO, SSD, Faster R-CNN, etc.).
Utilisation d'OpenCV pour la détection d'objets.
Présentation approfondie de la bibliothèque OpenCV pour la vision par ordinateur.
Configuration de l'environnement OpenCV.
Charger et afficher des images dans OpenCV.
Introduction aux classificateurs en cascade d'OpenCV pour la détection d'objets.
Présentation des modèles IA pré-entraînés pour la détection d'objets.
Comparaison des différents modèles disponibles (YOLO, SSD, Faster R-CNN, etc.).
Choix du modèle en fonction des besoins de l'application.

Segmentation d'Images avec PyTorch

Comprendre la segmentation d'images.
Création d'un modèle de segmentation convolutif avec PyTorch.
Préparation des données d'entraînement pour la segmentation.
Entraînement et évaluation des performances du modèle.

Génération d'Images avec les GAN

Introduction aux réseaux génératifs adverses (GAN).
Création d'un modèle GAN simple avec PyTorch.

NLP Introduction

Objectifs : comprendre le langage humain, et savoir générer des réponses
Différentes étapes : reconnaissance de caractères, ou de la voix, conversion des données en texte, décomposition en éléments de phrase, nettoyage des données, traitement de l'ambiguité d'un mot, reconnaissance d'une entité nommée (NEM), traitement des multiples références pour une entité, extraction des informations subjectives, etc ...
Les outils de NLP et historique :
outils statistiques, de machine learning, de deep learning,
Watson NLU, Python et le NLTK.
Applications actuelles : solutions de détection de spam, traduction automatique, assistants virtuels, chatbots, analyses d'opinions, de sentiments, etc ...

Python et le NLTK

Introduction : plate-formes supportées, versions de Python,
Présentation des textes et modèles fournis avec le NLTK

Atelier : installation du package NLTK et des datasets

Traitements de textes

Etude des différentes fonctions fournies par le NLTK.
Découpage d'un texte en mots ou en phrases,avec nltk.tokenize(),
nettoyage de textes avec le filtrage de mots,
stemming avec nltk.stem, alertes sur les risques d'un mauvais usage,
étiquetage des différentes parties d'un texte avec nltk.pos-tag(),
lemmatisation, pour identifier les formes canoniques des mots,
identification de phrases avec le chunking

Atelier : réalisation d'exemples sur des datasets simples

Analyses de textes

Description de nltk.ne_chunk() pour la reconnaissance d'identités nommées,
Présentation des fonctions concordance(), dispersion_plot(), FreqDist

Atelier : Import de corpus de textes, analyse, mise en évidence de l'utilisation de termes caractéristiques

Etude de cas

Analyse de sentiments avec nltk.sentiment
Présentation des fonctions disponibles

Atelier : mise en oeuvre sur un corpus. Utilisation de polarity.scores()

Intégration de scikit-learn

Import des algorithmes de classification de scikit-learn

Atelier : exemple d'utilisation des aglorithmes de scikit-learn depuis nltk

Introduction

Définition IA conversationnelle, historique, évolutions.
Présentation des IA les plus utilisées, caractéristiques, points forts, limites
Domaines d'application, exemples.
Aspects techniques, éthiques, économiques.

ChatGPT

ChatBot d'OpenAI basé sur les modèles GPT-3 (gratuit) ou GPT-4 (payant)
Principe des modèles GPT-x (Generative Pre-trained Transformer)
Différents modèles disponibles.
Apports de GPT-4 par rapport à GPT-3.
Notions de pulgins.

Atelier : exemples de conversations avec ChatGPT

Méthodes d'interrogation, bonnes pratiques

Google Gemini

ChatBot basé sur LaMDA (Language Model for Dialogue Applications)
Fonctionnalités :
Réponse aux requêtes en langage naturel,
facilité de réédition des prompts,
proposition de plusieurs réponses,
connexion au web,
interface vocale

Atelier : démonstrations avec Gemini

Utilisation de l'interface vocale
Comparaison des résultats de requêtes avec ChatGPT

Création d'un ChatBot

Différentes utilisations : assistant virtuel clients, chatbot entreprise, ...
Implémentation de différentes fonctionnalités :
conversation en langage naturel, Intelligence artificielle, connexion au web, utilisation de données de l'entreprise, etc.
Technologies utilisées : NLP (Natural Language Processing), NLU (Natural Language Understanding, NLG (Natural Language Generation), machine-learning, deep-learning pour l'IA
Exemples d'outils et plate-formes :
AWS lex, (base d'Alexia) :
service AWS permettant la création d'interfaces conversationnelles
en langage naturel,
connexion à l'environnement et aux services AWS,
plate-forme de déploiement de Bots,
déploiement pour des applications mobiles,
gestion de la sécurité par AWS lex.
Golem, BotMan, Tock, Botpress, Wit.ai

Atelier : création d'un assistant virtuel hébergé en interne

Evolutions

Nouvelles versions en développement, acteurs.
Apports de l'IA quantique

Serious game : Création assistant virtuel

Présentation des assistants IA.
Principe : traitement du langage naturel, reconnaissance et synthèse vocale, réalisation d'actions et requêtes internet.

La méthode

Simulation d'un projet collaboratif ayant pour objectif la création d'un assistant IA.
Épreuves personnelles et épreuves en commun vont permettre de contrôler les connaissances et d'échanger entre stagiaires, tout en bénéficiant du soutien et des explications complémentaires du formateur sur les thèmes proposés

Les jeux

Battle d'architecture, la techno mystère, l'intrus, les points de faiblesse, etc...

Le debrief

Retour des travaux, bilan des points individuels et classement des joueurs.
Retour d'expérience des participants