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 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.

Big Data

Big Data

Comprendre les principaux concepts du Big Data ainsi que l'écosystème technologique d'un projet Big Data

L'essentiel du BigData : calcul distribué, données non structurées. Besoins fonctionnels et caractéristiques techniques des projets. La valorisation des données. Le positionnement respectif des technologies de cloud, BigData et noSQL, et les liens, implications.
Concepts clés : ETL, Extract Transform Load, CAP, 3V, 4V, données non structurées, prédictif, Machine Learning.
L'écosystème du BigData : les acteurs, les produits, état de l'art. Cycle de vie des projets BigData.

Atelier : Démonstration d'un prédiction Machine Learning avec Dataiku DSS

Savoir analyser les difficultés propres à un projet Big Data

Rôle de la DSI dans la démarche BigData. Gouvernance des données: importance de la qualité des données, fiabilité, durée de validité, sécurité des données
Emergence de nouveaux métiers : Data-scientists, Data labs, Hadoop scientists, CDO, ...
Intégration avec les outils statistiques présents et les outils BigData futurs.

Déterminer la nature des données manipulées

Les différents modes et formats de stockage.
Les types de bases de données : clé/valeur, document, colonne, graphe. Besoin de distribution. Définition de la notion d'élasticité. Principe du stockage réparti.
Données structurées et non structurées, documents, images, fichiers XML, JSON, CSV, ...

Atelier : démonstrations avec une base MongoDB et une base Cassandra sur des données de différents types.

Appréhender les éléments de sécurité, d'éthique et les enjeux juridiques

Les risques et points à sécuriser dans un système distribué.
Aspects législatifs et éthiques: sur le stockage, la conservation de données, ..., sur les traitements, la commercialisation des données, des résultats

Atelier : mise en évidence des problèmes liés à la réplication inter-régions et concernant les aspects juridiques des données : droits d'exploitation, propriété intellectuelle, ...

Etude des failles de sécurité sur une infrastructure Hadoop.

Exploiter les architectures Big Data

Les objectifs de la supervision, les techniques disponibles. La supervision d'une ferme BigData.
Objets supervisés. Les services et ressources. Protocoles d'accès. Exporteurs distribués de données.
Définition des ressources à surveiller. Journaux et métriques.
Application aux fermes BigData : Hadoop, Cassandra, HBase, MongoDB
Besoin de base de données avec agents distribués, de stockage temporel (timeseriesDB)
Produits : Prometheus, Graphite, ElasticSearch.
Présentation, architectures.
Les sur-couches : Kibana, Grafana.

Atelier : mise en oeuvre de prometheus pour la supervision d'une ferme Cassandra sur une infrastructure distribuée multi-noeuds.

Mettre en place des socles techniques complets pour des projets Big Data.

Etude des différents composants d'une infrastructure BigData :
Datalake : collecte des différents types de données
Stockage distribué : réplication, sharding, gossip, hachage,
Principe du schemaless, schéma de stockage, clé de distribution, clé de hachage
Systèmes de fichiers distribués : GFS, HDFS, Ceph. Les bases de données : Redis, Cassandra, DynamoDB, Accumulo, HBase, MongoDB, BigTable, Neo4j, ...
Calcul et restitution :Apport des outils de calculs statistiques
Langages adaptés aux statistiques, liens avec les outils BigData.
Outils de calcul et visualisation :R, SAS, Spark, Tableau, QlikView, ...
Caractéristiques et points forts des différentes solutions.

Atelier : mise en oeuvre du sharding avec une base de données MongoDB sur une infrastructure distribuée

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.

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.

Réseaux de neurones

Réseaux de neurones

Serious game : Implémentations Réseaux de neurones

Identifier les oiseaux qui viennent gazouiller aux fenêtres selon les saisons et les régions, et améliorer la reconnaissance des différentes espèces.

La méthode

Simulation d'un cas d'étude, avec un travail collaboratif sur des données réelles, accessibles en opendata, et des labs techniques (Keras/Tensorflow, Pytorch/Caffe, Lasagne, ..)
Épreuves personnelles et épreuves en commun vont permettre de contrôler les connaissances et d'échanger entre participants, 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

Tensorflow/Keras

Tensorflow/Keras

TensorFlow

Introducion au traitement d'images et à l'apprentissage automatique, les apports de l'IA.
Cas d'applications : analyse, tri d'images, détection d'objets, reconnaissance faciale, génération d'images, etc.
Présentation de Keras, PyTorch et OpenCV : principes de fonctionnement, caractéristiques, points forts.

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

Le projet Tensorflow et Keras

Historique , fonctionnalités
Architecture distribuée, plateformes supportées
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

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

Optimisation d'un modèle

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

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.

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.).
Mise en oeuvre d'OpenCV pour la détection d'objets.
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.

Génération d'Images avec les GAN

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

Spark

Spark

Spark

Présentation Spark, origine du projet,apports, principe de fonctionnement. Langages supportés.
Modes de fonctionnement : batch/Streaming.
Bibliothèques : Machine Learning, IA
Mise en oeuvre sur une architecture distribuée. Architecture : clusterManager, driver, worker, ...
Architecture : SparkContext, SparkSession, Cluster Manager, Executor sur chaque noeud. Définitions : Driver program, Cluster manager, deploy mode, Executor, Task, Job

Savoir intégrer Spark dans un environnement Hadoop

Intégration de Spark avec HDFS, HBase,
Création et exploitation d'un cluster Spark/YARN. Intégration de données sqoop, kafka, flume vers une architecture Hadoop et traitements par Spark.
Intégration de données AWS S3.
Différents cluster managers : Spark interne, avec Mesos, avec Yarn, avec Amazon EC2

Atelier : Mise en oeuvre avec Spark sur Hadoop HDFS et Yarn. Soumission de jobs, supervision depuis l'interface web

Développer des applications d’analyse en temps réel avec Spark Structured Streaming

Objectifs , principe de fonctionnement: stream processing. Source de données : HDFS, Flume, Kafka, ...
Notion de StreamingContext, DStreams, démonstrations.

Atelier : traitement de flux DStreams en Scala. Watermarking. Gestion des micro-batches.

Intégration de Spark Structured Streaming avec Kafka

Atelier : mise en oeuvre d'une chaîne de gestion de données en flux tendu : IoT, Kafka, Spark Structured Streaming, Spark. Analyse des données au fil de l'eau.

Faire de la programmation parallèle avec Spark sur un cluster

Utilisation du shell Spark avec Scala ou Python. Modes de fonctionnement. Interprété, compilé.
Utilisation des outils de construction. Gestion des versions de bibliothèques.

Atelier : Mise en pratique en Java, Scala et Python. Notion de contexte Spark. Extension aux sessions Spark.

Manipuler des données avec Spark SQL

Spark et SQL
Traitement de données structurées. L'API Dataset et DataFrames
Jointures. Filtrage de données, enrichissement. Calculs distribués de base. Introduction aux traitements de données avec map/reduce.
Lecture/écriture de données : Texte, JSon, Parquet, HDFS, fichiers séquentiels.
Optimisation des requêtes. Mise en oeuvre des Dataframes et DataSet. Compatibilité Hive

Atelier : écriture d'un ETL entre HDFS et HBase

Atelier : extraction, modification de données dans une base distribuée.
Collections de données distribuées. Exemples.

Support Cassandra

Description rapide de l'architecture Cassandra. Mise en oeuvre depuis Spark. Exécution de travaux Spark s'appuyant sur une grappe Cassandra.

Spark GraphX

Fourniture d'algorithmes, d'opérateurs simples pour des calculs statistiques sur les graphes

Atelier : exemples d'opérations sur les graphes.

Avoir une première approche du Machine Learning

Machine Learning avec Spark, algorithmes standards supervisés et non-supervisés (RandomForest, LogisticRegression, KMeans, ...)
Gestion de la persistance, statistiques.
Mise en oeuvre avec les DataFrames.

Atelier : mise en oeuvre d'une régression logistique sur Spark

Python pour la DataScience

Python pour la DataScience

Positionnement Python

Les valeurs d'observation, et les variables cibles.
Ingénierie des variables.
Analyses statistiques,
Classification des données, rapprochements,
Production de recommandations. Evolutions des outils statistiques classiques vers l'apprentissage automatique.

Atelier : exercices sur les outils statistiques de base

Savoir utiliser les principaux outils de traitement et d'analyse de données pour 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
Présentation des outils d'apprentissage Python : scikit-learn, pybrain, TensorFlow/keras, mxnet, caffe

Atelier : mise en oeuvre de scikit-learn et génération de jeux de données.

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

Atelier : construction d'ETL de base entre json et csv

Savoir appliquer les pratiques optimales en matière de nettoyage et de préparation des données avant l'analyse

Encodeurs
Filtres et ETL
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.

Atelier : construction d'un système de détection de contours

Apprendre à mettre en place un modèle d'apprentissage simple

Les différentes méthodes : apprentissage supervisé, apprentissage automatique.
Algorithmes : régression linéaire, k-voisins,classification naïve bayésienne, arbres de décision, ...

Atelier : classifieurs. scoring

APIs fournies en standard, modèles d'apprentissage
Projet scikit-learn : classification, régression, validation de modèles prédictifs.
Démonstrations avec les modèles fournis par scikit-learn
Positionnement et comparaison avec Keras, mxnet, caffe

Atelier : codage d'une reconnaissance d'animaux avec une forêt aléatoire

Choisir entre la régression et la classification en fonction du type de données

Présentation des types de données en entrées : données discrètes, données continues. Labelisation, mapping par fonction.
Comprendre les algorithmes : régression linéaire, k-moyennes, k-voisins, classification naïve bayésienne, arbres de décision, forêts aléatoires, ...
Critères de choix des algorithmes.

Atelier : construction d'un système décisionnel fondé sur des forêts aléatoires

Évaluer les performances prédictives d'un algorithme

Les courbes d'apprentissage. Définitions : AUC, courbes ROC.
Comprendre le principe des hyper-paramètres. Choix des hyper-paramètres.

Atelier : calcul et visualisation d'une matrice de confusion

Atelier : Visualisation de courbes d'apprentissage fonction des hyper-paramètres

Atelier : Visualisation d'une mise en sur-apprentissage

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

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