

Mastère Spécialisé Intelligence Artificielle dans la Programmation



Mastère Spécialisé Intelligence Artificielle dans la Programmation

- » Modalité: en ligne
- » Durée: 12 mois
- » Diplôme: TECH Université Technologique
- » Temps estimé: 16 heures/semaine
- » Horaire: à votre rythme
- » Examens: en ligne

Accès au site web: www.techtitute.com/fr/intelligence-artificielle/master/master-intelligence-artificielle-programmation

Sommaire

01

Présentation

page 4

02

Objectifs

page 8

03

Compétences

page 16

04

Direction de la formation

page 20

05

Structure et contenu

page 24

06

Méthodologie

page 42

07

Diplôme

page 50

01

Présentation

À une époque caractérisée par les avancées technologiques, l'Intelligence Artificielle (IA) est devenue un outil essentiel pour les experts en Programmation. Son importance réside dans sa capacité à mécaniser des activités ardues, à prendre des décisions basées sur des données précises et à apprendre à partir de modèles. L'Apprentissage Automatique offre donc aux informaticiens des techniques précieuses pour concevoir des systèmes plus intelligents. Ces techniques vont des algorithmes destinés à rendre les programmes plus complets au développement de systèmes autonomes, utiles pour modifier la manière dont le code est exécuté. C'est pourquoi TECH lance une formation académique qui fournira aux étudiants les dernières avancées dans ce domaine. Le tout dans le cadre d'une méthodologie 100 % en ligne, adaptée à l'agenda des professionnels occupés.



“

Vous serez en mesure de concevoir des expériences utilisateur personnalisées et intuitives grâce à cette formation universitaire 100 % en ligne”

L'Intelligence Informatique aide les institutions à améliorer la productivité dans le développement de *logiciels*. Ses outils sont capables de traiter des données non structurées, de tirer des enseignements des expériences passées et de s'adapter aux changements dans des environnements dynamiques. En outre, l'IA peut prédire les problèmes d'application potentiels avant qu'ils ne se produisent, ce qui permet aux professionnels de prendre des mesures préventives pour éviter des problèmes coûteux à l'avenir. Dans ce contexte, les entreprises informatiques internationales les plus prestigieuses cherchent activement à intégrer des spécialistes de l'Architecture Logicielle pour les QA *Testing*.

C'est pourquoi TECH met en œuvre un programme innovant pour les programmeurs afin de tirer le meilleur parti de l'optimisation et de la gestion des performances des outils d'IA. Conçu par des experts de premier plan, le programme d'études se penchera sur la programmation d'algorithmes pour développer des produits dotés de systèmes intelligents. Il abordera également les extensions essentielles de Visual Studio Code, l'éditeur de code source le plus utilisé aujourd'hui. En outre, le matériel pédagogique traitera de l'intégration de l'IA dans la gestion des bases de données afin de détecter d'éventuelles défaillances et de créer des *test unitaires*. Il s'agit d'une formation universitaire qui dispose d'une diversité de contenus audiovisuels dans de multiples formats et d'un réseau de simulations réelles pour rapprocher le développement du programme de la réalité de la pratique informatique.

Afin d'atteindre les objectifs d'apprentissage proposés, ce programme est enseigné en utilisant une méthodologie d'enseignement en ligne. De cette manière, les professionnels pourront parfaitement combiner leur travail avec leurs études. En outre, vous bénéficierez d'un corps enseignant de premier ordre et de matériel académique multimédia d'une grande rigueur pédagogique, tel que des classes de maître, des résumés interactifs et des exercices pratiques. La seule condition pour accéder au Campus Virtuel est de disposer d'un appareil électronique avec accès à l'Internet et même d'utiliser son téléphone portable.

Ce **Mastère Spécialisé en Intelligence Artificielle dans la Programmation** contient le programme le plus complet et le plus actualisé du marché. Ses caractéristiques sont les suivantes:

- ♦ Le développement d'études de cas présentées par des experts en Intelligence Artificielle dans la Programmation
- ♦ Les contenus graphiques, schématiques et éminemment pratiques avec lesquels ils sont conçus fournissent des informations scientifiques et sanitaires essentielles à la pratique professionnelle
- ♦ Exercices pratiques permettant de réaliser le processus d'auto-évaluation afin d'améliorer l'apprentissage
- ♦ Il met l'accent sur les méthodologies innovantes
- ♦ Cours théoriques, questions à l'expert, forums de discussion sur des sujets controversés et travail de réflexion individuel
- ♦ La possibilité d'accéder aux contenus depuis n'importe quel appareil fixe ou portable doté d'une connexion internet



Vous obtiendrez une perspective holistique sur la façon dont l'Apprentissage Automatique influence et améliore chaque étape du développement de logiciels"

“

Vous cherchez à appliquer les Modèles Transformationnels pour le traitement du langage naturel à votre pratique ? Réalisez-le grâce à ce programme innovant”

Le corps enseignant du programme englobe des spécialistes réputés dans le domaine et qui apportent à ce programme l'expérience de leur travail, ainsi que des spécialistes reconnus dans de grandes sociétés et des universités prestigieuses.

Grâce à son contenu multimédia développé avec les dernières technologies éducatives, les spécialistes bénéficieront d'un apprentissage situé et contextuel, ainsi, ils se formeront dans un environnement simulé qui leur permettra d'apprendre en immersion et de s'entraîner dans des situations réelles.

La conception de ce programme est axée sur l'Apprentissage par les Problèmes, grâce auquel le professionnel doit essayer de résoudre les différentes situations de la pratique professionnelle qui se présentent tout au long du programme académique. Pour ce faire, l'étudiant sera assisté d'un innovant système de vidéos interactives, créé par des experts reconnus.

Vous vous plongerez dans le cycle de vie du Testing, de la création des cas de test à la détection des bogues.

Le Relearning vous permettra d'apprendre avec moins d'efforts et plus de performance, en vous impliquant davantage dans votre spécialisation professionnelle.



02 Objectifs

Cette formation fera des informaticiens des experts en IA appliquée à la Programmation. Les diplômés acquerront une vision globale, combinant les connaissances les plus récentes avec des compétences pratiques qui amélioreront leur prise de décision. Dans le même temps, les professionnels maîtriseront les outils les plus modernes pour le développement de *logiciels* alimentés par l'Apprentissage Automatique. De cette manière, les étudiants concevront des propositions à la fois pour les sites web et les applications mobiles avec adaptabilité. Ils seront hautement spécialisés pour répondre aux exigences actuelles de l'industrie.





“

Vous souhaitez vous spécialiser dans l'Intelligence Artificielle ? Avec ce programme, vous maîtriserez l'optimisation du processus de déploiement et l'intégration de l'Intelligence Artificielle dans le cloud computing"



Objectifs généraux

- ♦ Développer des compétences pour mettre en place et gérer des environnements de développement efficaces, garantissant une base solide pour la mise en œuvre de projets d'IA
- ♦ Acquérir des compétences en matière de planification, d'exécution et d'automatisation des tests de qualité, en intégrant des outils d'IA pour la détection et la correction des *bugs*
- ♦ Comprendre et appliquer les principes de performance, d'évolutivité et de maintenabilité dans la conception de systèmes informatiques à grande échelle
- ♦ Se familiariser avec les modèles de conception les plus importants et les appliquer efficacement dans l'architecture logicielle



Objectifs spécifiques

Module 1. Principes fondamentaux de l'Intelligence Artificielle

- ♦ Analyser l'évolution historique de l'Intelligence Artificielle, de ses débuts à son état actuel, en identifiant les étapes et les développements clés
- ♦ Comprendre le fonctionnement des réseaux neuronaux et leur application dans les modèles d'apprentissage en Intelligence Artificielle
- ♦ Étudier les principes et les applications des algorithmes génétiques, en analysant leur utilité pour résoudre des problèmes complexes
- ♦ Analyser l'importance des thésaurus, des vocabulaires et des taxonomies dans la structuration et le traitement des données pour les systèmes d'IA
- ♦ Explorer le concept de web sémantique et son influence sur l'organisation et la compréhension de l'information dans les environnements numériques

Module 2. Types et cycle de vie des données

- ♦ Comprendre les concepts fondamentaux des statistiques et leur application dans l'analyse des données
- ♦ Identifier et classer les différents types de données statistiques, des données quantitatives aux données qualitatives
- ♦ Analyser le cycle de vie des données, de la génération à l'élimination, en identifiant les étapes clés
- ♦ Explorer les premières étapes du cycle de vie des données, en soulignant l'importance de la planification et de la structure des données.
- ♦ Étudier les processus de collecte de données, y compris la méthodologie, les outils et les canaux de collecte
- ♦ Explorer le concept de *Datawarehouse* en mettant l'accent sur les éléments du *Datawarehouse* et sur sa conception
- ♦ Analyser les aspects réglementaires liés à la gestion des données, en se conformant aux règles de confidentialité et de sécurité, ainsi qu'aux meilleures pratiques

Module 3. Les données en Intelligence Artificielle

- ♦ Maîtriser les fondamentaux de la science des données, couvrant les outils, les types et les sources d'analyse de l'information
- ♦ Explorer le processus de transformation des données en informations à l'aide de techniques d'exploration et de visualisation des données
- ♦ Étudier la structure et les caractéristiques des *datasets*, en comprenant leur importance dans la préparation et l'utilisation des données pour les modèles d'Intelligence Artificielle
- ♦ Analyser les modèles supervisés et non supervisés, y compris les méthodes et la classification
- ♦ Utiliser des outils spécifiques et les meilleures pratiques dans la manipulation et le traitement des données, en assurant l'efficacité et la qualité dans la mise en œuvre de l'Intelligence Artificielle

Module 4. Extraction de données. Sélection, prétraitement et transformation

- ♦ Maîtriser les techniques d'inférence statistique pour comprendre et appliquer les méthodes statistiques dans l'exploration de données
- ♦ Effectuer une analyse exploratoire détaillée des ensembles de données afin d'identifier les modèles, les anomalies et les tendances pertinents
- ♦ Développer des compétences pour la préparation des données, y compris le nettoyage, l'intégration et le formatage des données en vue de leur utilisation dans l'exploration de données
- ♦ Mettre en œuvre des stratégies efficaces pour traiter les valeurs manquantes dans les ensembles de données, en appliquant des méthodes d'imputation ou d'élimination en fonction du contexte
- ♦ Identifier et atténuer le bruit dans les données, en utilisant des techniques de filtrage et de lissage pour améliorer la qualité de l'ensemble de données
- ♦ Aborder le prétraitement des données dans les environnements *Big Data*

Module 5. Algorithmes et complexité en Intelligence Artificielle

- ♦ Introduire les stratégies de conception d'algorithmes, en fournissant une solide compréhension des approches fondamentales de la résolution de problèmes
- ♦ Analyser l'efficacité et la complexité des algorithmes, en appliquant des techniques d'analyse pour évaluer les performances en termes de temps et d'espace
- ♦ Étudier et appliquer des algorithmes de tri, comprendre leur fonctionnement et comparer leur efficacité dans différents contextes
- ♦ Explorer les algorithmes basés sur les arbres, comprendre leur structure et leurs applications
- ♦ Étudier les algorithmes avec *Heaps*, en analysant leur mise en œuvre et leur utilité pour une manipulation efficace des données
- ♦ Analyser les algorithmes basés sur les graphes, en explorant leur application dans la représentation et la résolution de problèmes impliquant des relations complexes
- ♦ Étudier les algorithmes de type *Greedy*, comprendre leur logique et leurs applications dans la résolution de problèmes d'optimisation
- ♦ Étudier et appliquer la technique du *backtracking* pour la résolution systématique de problèmes, en analysant son efficacité dans différents scénarios

Module 6. Systèmes intelligents

- ♦ Explorer la théorie des agents, comprendre les concepts fondamentaux de leur fonctionnement et leur application en Intelligence Artificielle et en génie Logiciel
- ♦ Étudier la représentation des connaissances, y compris l'analyse des ontologies et leur application dans l'organisation de l'information structurée
- ♦ Analyser le concept du web sémantique et son impact sur l'organisation et la recherche d'informations dans les environnements numériques

- ♦ Évaluer et comparer différentes représentations de la connaissance, en les intégrant pour améliorer l'efficacité et la précision des systèmes intelligents
- ♦ Étudier les raisonneurs sémantiques, les systèmes à base de connaissances et les systèmes experts, en comprenant leur fonctionnalité et leurs applications dans la prise de décision intelligente

Module 7. Apprentissage automatique et exploration des données

- ♦ Présenter les processus de découverte des connaissances et les concepts fondamentaux de l'apprentissage automatique
- ♦ Étudier les arbres de décision en tant que modèles d'apprentissage supervisé, comprendre leur structure et leurs applications
- ♦ Évaluer les classificateurs en utilisant des techniques spécifiques pour mesurer leur performance et leur précision dans la classification des données
- ♦ Étudier les réseaux neuronaux, comprendre leur fonctionnement et leur architecture pour résoudre des problèmes complexes d'apprentissage automatique
- ♦ Explorer les méthodes bayésiennes et leur application à l'apprentissage automatique, y compris les réseaux bayésiens et les classificateurs bayésiens
- ♦ Analyser les modèles de régression et de réponse continue pour prédire des valeurs numériques à partir de données
- ♦ Étudier les techniques de *clustering* pour identifier des modèles et des structures dans des ensembles de données non étiquetés
- ♦ Explorer l'exploration de textes et le traitement du langage naturel (NLP), comprendre comment les techniques d'apprentissage automatique sont appliquées pour analyser et comprendre les textes

Module 8. Les Réseaux Neuronaux, la base du *Deep Learning*

- ♦ Maîtriser les fondamentaux du Deep Learning, comprendre son rôle essentiel dans le *Deep Learning*
- ♦ Explorer les opérations fondamentales des réseaux neuronaux et comprendre leur application dans la construction de modèles

- ♦ Analyser les différentes couches utilisées dans les réseaux neuronaux et apprendre à les sélectionner de manière appropriée
- ♦ Comprendre l'enchaînement efficace des couches et des opérations pour concevoir des architectures de réseaux neuronaux complexes et efficaces
- ♦ Utiliser des formateurs et des optimiseurs pour régler et améliorer les performances des réseaux neuronaux
- ♦ Explorer la connexion entre les neurones biologiques et artificiels pour une compréhension plus approfondie de la conception des modèles
- ♦ Réglage fin des hyperparamètres pour le *Fine Tuning* des réseaux neuronaux, optimisant leur performance sur des tâches spécifiques

Module 9. Entraînement de réseaux neuronaux profonds

- ♦ Résoudre les problèmes liés au gradient dans la formation des réseaux neuronaux profonds
- ♦ Explorer et appliquer différents optimiseurs pour améliorer l'efficacité et la convergence du modèle
- ♦ Programmer le taux d'apprentissage pour ajuster dynamiquement le taux de convergence du modèle
- ♦ Comprendre et traiter le surajustement grâce à des stratégies spécifiques pendant la formation
- ♦ Appliquer des lignes directrices pratiques pour assurer une formation efficace et efficiente des réseaux neuronaux profonds
- ♦ Mettre en œuvre le *Transfer Learning* en tant que technique avancée pour améliorer les performances du modèle sur des tâches spécifiques
- ♦ Explorer et appliquer les techniques *Data Augmentation* pour enrichir les ensembles de données et améliorer la généralisation des modèles
- ♦ Développer des applications pratiques utilisant le *Transfer Learning* pour résoudre des problèmes du monde réel
- ♦ Comprendre et appliquer les techniques de régularisation pour améliorer la généralisation et éviter l'overfitting dans les réseaux neuronaux profonds

Module 10. Personnaliser les modèles et l'entraînement avec TensorFlow

- ♦ Maîtriser les fondamentaux de *TensorFlow* et son intégration avec NumPy pour une manipulation efficace des données et des calculs
- ♦ Personnaliser les modèles et les algorithmes de formation en utilisant les capacités avancées de *TensorFlow*
- ♦ Explorer l'API *tf.data* pour gérer et manipuler efficacement les ensembles de données
- ♦ Implémenter le format *TFRecord* pour stocker et accéder à de grands ensembles de données dans *TensorFlow*
- ♦ Utiliser les couches de prétraitement Keras pour faciliter la construction de modèles personnalisés
- ♦ Explorer le projet *TensorFlow Datasets* pour accéder à des ensembles de données prédéfinis et améliorer l'efficacité du développement
- ♦ Développer une application de *Deep Learning* avec *TensorFlow*, en intégrant les connaissances acquises dans le module
- ♦ Appliquer de manière pratique tous les concepts appris dans la construction et l'entraînement de modèles personnalisés avec *TensorFlow* dans des situations réelles

Module 11. Deep Computer Vision avec les Réseaux Neuronaux Convolutifs

- ♦ Comprendre l'architecture du cortex visuel et sa pertinence dans le *Deep Computer Vision*
- ♦ Explorer et appliquer les couches convolutives pour extraire les caractéristiques clés des images
- ♦ Implémenter des couches de clustering et leur utilisation dans les modèles de *Deep Computer Vision* avec Keras
- ♦ Analyser diverses architectures de Réseaux Neuronaux Convolutifs (CNN) et leur applicabilité dans différents contextes
- ♦ Développer et mettre en œuvre un CNN ResNet à l'aide de la bibliothèque Keras afin d'améliorer l'efficacité et les performances du modèle
- ♦ Utiliser des modèles Keras pré-entraînés pour tirer parti de l'apprentissage par transfert pour des tâches spécifiques

- ♦ Appliquer des techniques de classification et de localisation dans des environnements de *Deep Computer Vision*
- ♦ Explorer les stratégies de détection et de suivi d'objets à l'aide de Réseaux Neuronaux Convolutifs
- ♦ Mettre en œuvre des techniques de segmentation sémantique pour comprendre et classer les objets dans les images de manière détaillée

Module 12. Traitement du langage naturel (NLP) avec les Réseaux Récurrents Naturels (RNN) et l'Attention

- ♦ Développer des compétences en génération de texte à l'aide de Réseaux Neuronaux Récurrents (RNN)
- ♦ Appliquer les RNN dans la classification des opinions pour l'analyse des sentiments dans les textes
- ♦ Comprendre et appliquer les mécanismes de l'attention dans les modèles de traitement du langage naturel
- ♦ Analyser et utiliser les modèles *Transformers* dans des tâches NLP spécifiques
- ♦ Explorer l'application des modèles *Transformers* dans le contexte du traitement de l'image et de la vision par ordinateur
- ♦ Se familiariser avec la bibliothèque de *Transformers Hugging Face* pour une mise en œuvre efficace de modèles avancés
- ♦ Comparer différentes bibliothèques de *Transformers* afin d'évaluer leur adéquation à des tâches spécifiques
- ♦ Développer une application pratique du NLP qui intègre les mécanismes de RNN et d'attention pour résoudre des problèmes du monde réel

Module 13. Autoencodeurs, GANs, et Modèles de Diffusion

- ♦ Développer des représentations de données efficaces en utilisant des *Autoencoders*, *GANs* et des Modèles de Diffusion
- ♦ Effectuer une ACP en utilisant un autoencodeur linéaire incomplet pour optimiser la représentation des données

- Mettre en œuvre et comprendre le fonctionnement des autoencodeurs empilés
- Explorer et appliquer les autoencodeurs convolutifs pour des représentations visuelles efficaces des données
- Analyser et appliquer l'efficacité des autoencodeurs clairsemés dans la représentation des données
- Générer des images de mode à partir de l'ensemble de données MNIST à l'aide d'*Autoencoders*
- Comprendre le concept des Réseaux Adversariels Génératifs (*GANs*) et des Modèles de Diffusion
- Implémenter et comparer les performances des Modèles de Diffusion et des *GANs* dans la génération de données

Module 14. Informatique bio-inspirée

- Introduire les concepts fondamentaux de l'informatique bio-inspirée
- Explorer les algorithmes d'adaptation sociale en tant qu'approche clé de l'informatique bio-inspirée
- Analyser les stratégies d'exploration et d'exploitation de l'espace dans les algorithmes génétiques
- Examiner les modèles de calcul évolutif dans le contexte de l'optimisation
- Poursuivre l'analyse détaillée des modèles de calcul évolutif
- Appliquer la programmation évolutive à des problèmes d'apprentissage spécifiques
- Aborder la complexité des problèmes multi-objectifs dans le cadre de l'informatique bio-inspirée
- Explorer l'application des réseaux neuronaux dans le domaine de l'informatique bio-inspirée
- Approfondir la mise en œuvre et l'utilité des réseaux neuronaux dans l'informatique bio-inspirée

Module 15. Intelligence Artificielle: Stratégies et applications

- Développer des stratégies pour la mise en œuvre de l'intelligence artificielle dans les services financiers
- Analyser les implications de l'intelligence artificielle dans la fourniture de services de santé
- Identifier et évaluer les risques associés à l'utilisation de l'IA dans le domaine de la santé
- Évaluer les risques potentiels liés à l'utilisation de l'intelligence artificielle dans l'industrie
- Appliquer des techniques d'intelligence artificielle dans l'industrie pour améliorer la productivité
- Concevoir des solutions d'intelligence artificielle pour optimiser les processus dans l'administration publique
- Évaluer la mise en œuvre des technologies de l'IA dans le secteur de l'éducation
- Appliquer des techniques d'intelligence artificielle dans la sylviculture et l'agriculture pour améliorer la productivité
- Optimiser les processus de ressources humaines par l'utilisation stratégique de l'intelligence artificielle

Module 16. Améliorer la Productivité du Développement de Logiciels grâce à l'IA

- Découvrez la mise en œuvre d'extensions d' IA indispensables dans Visual Studio Code afin d'améliorer la productivité et de faciliter le développement de *logiciels*
- Acquérir une solide compréhension des concepts de base de l'IA et de son application dans le développement de *logiciels*, y compris les algorithmes d'apprentissage automatique, le traitement du langage naturel, les réseaux neuronaux, etc.
- Maîtriser la configuration d'environnements de développement optimisés, garantissant que les étudiants peuvent créer des environnements propices aux projets d'IA
- Appliquer des techniques spécifiques utilisant ChatGPT pour l'identification et la correction automatiques des améliorations possibles du code, en encourageant des pratiques de programmation plus efficaces
- Promouvoir la collaboration entre différents professionnels de la programmation (des programmeurs aux ingénieurs de données en passant par les concepteurs de l'expérience utilisateur) afin de développer des solutions *logicielles* d'IA efficaces et éthiques

Module 17. Architecture du *Logiciel* pour les QA Testing

- ◆ Développer des compétences pour concevoir des plans de test robustes, couvrant différents types de *testing* et garantissant la qualité des logiciels
- ◆ Reconnaître et analyser les différents types de cadres *logiciels*, tels que les cadres monolithiques, microservices ou orientés services
- ◆ Acquérir une vue d'ensemble des principes et des techniques de conception de systèmes informatiques évolutifs et capables de traiter de grands volumes de données
- ◆ Appliquer des compétences avancées dans la mise en œuvre de structures de données alimentées par l'IA afin d'optimiser les performances et l'efficacité des *logiciels*
- ◆ Développer des pratiques de développement sécurisées, en se concentrant sur l'évitement des vulnérabilités pour assurer la sécurité des *logiciels* au niveau architectural

Module 18. Projets Web avec IA

- ◆ Développer des compétences complètes pour la mise en œuvre de projets web, de la conception du *frontend* à l'optimisation du *backend*, avec l'inclusion d'éléments d'IA
- ◆ Optimiser le processus de déploiement des sites web, en incorporant des techniques et des outils pour améliorer la rapidité et l'efficacité
- ◆ Intégrer l'IA dans l'informatique en nuage, permettant aux étudiants de créer des projets web hautement évolutifs et efficaces
- ◆ Acquérir la capacité d'identifier des problèmes et des opportunités spécifiques dans les projets web où l'IA peut être appliquée efficacement, comme dans le traitement de texte, la personnalisation, la recommandation de contenu, etc.
- ◆ Encourager les étudiants à se tenir au courant des dernières tendances et des derniers développements en matière d'IA pour une application adéquate dans les projets web

Module 19. Applications mobiles avec l'IA

- ◆ Appliquer des concepts avancés de *clean architecture*, de *sources de données* et de *référentiels* pour garantir une structure robuste et modulaire dans les applications mobiles avec l'IA
- ◆ Développer des compétences pour concevoir des écrans interactifs, des icônes et des ressources graphiques en utilisant l'IA pour améliorer l'expérience de l'utilisateur dans les applications mobiles
- ◆ Approfondir la configuration du framework d'application mobile et utiliser *GitHub Copilot* pour rationaliser le processus de développement
- ◆ Optimiser les applications mobiles d'IA pour obtenir des performances efficaces, en tenant compte de la gestion des ressources et de l'utilisation des données
- ◆ Effectuer des tests de qualité des applications mobiles d'IA, permettant aux étudiants d'identifier les problèmes et de déboguer les bogues

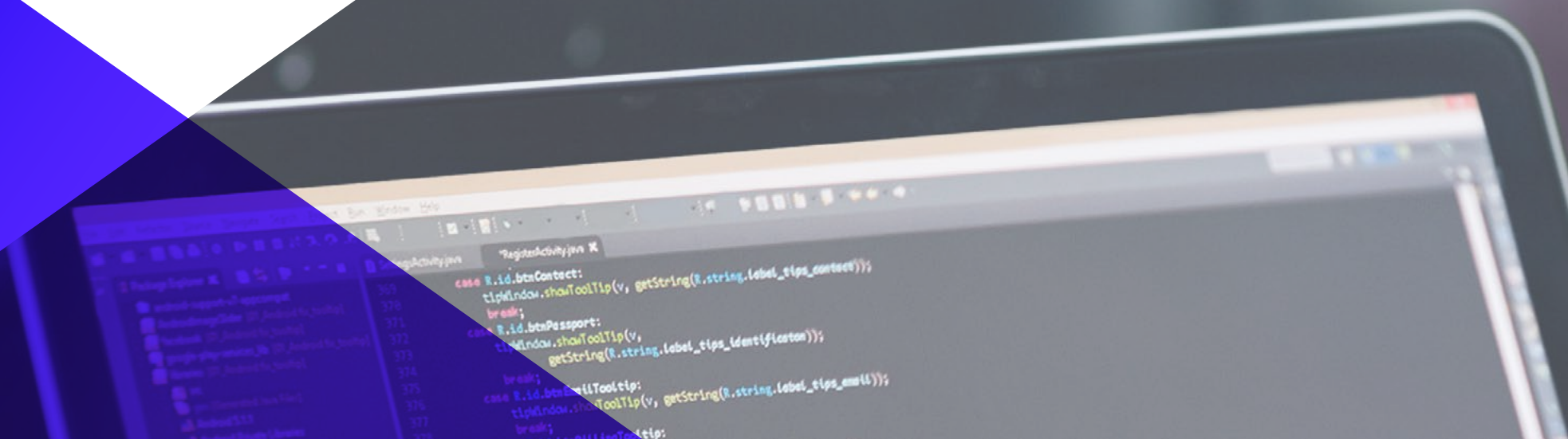
Module 20. L'IA pour les QA Testing

- ◆ Maîtriser les principes et les techniques de conception de systèmes informatiques évolutifs et capables de traiter de grands volumes de données
- ◆ Appliquer des compétences avancées dans la mise en œuvre de structures de données alimentées par l'IA afin d'optimiser les performances et l'efficacité des *logiciels*
- ◆ Comprendre et appliquer des pratiques de développement sécurisées, en mettant l'accent sur la prévention des vulnérabilités telles que l'injection, afin de garantir la sécurité des *logiciels* au niveau de l'architecture
- ◆ Générer des tests automatisés, en particulier dans les environnements web et mobiles, en intégrant des outils d'IA pour améliorer l'efficacité du processus
- ◆ Utiliser des outils avancés de QA alimentés par l'IA pour une détection plus efficace des *bugs* et une amélioration continue des *logiciels*

03

Compétences

Au terme de 12 mois d'apprentissage, les diplômés seront dotés d'un ensemble de compétences qui leur permettront d'élargir leur horizon professionnel dans le secteur des technologies de l'information. Ainsi, les experts seront en mesure de créer et de mettre en œuvre des logiciels avancés et de concevoir des propositions d'IA à grande échelle. Dans le même ordre d'idées, les professionnels s'occuperont de la productivité dans le développement, tout en appliquant les meilleures pratiques en matière de QA Testing. Cette qualification universitaire garantit que les informaticiens sont équipés pour relever avec succès les défis du monde réel et d'exceller dans un domaine qui progresse à pas de géant.



“

Vous serez hautement qualifié pour mettre en œuvre des algorithmes d'Apprentissage Automatique dans vos projets web et applications mobiles"

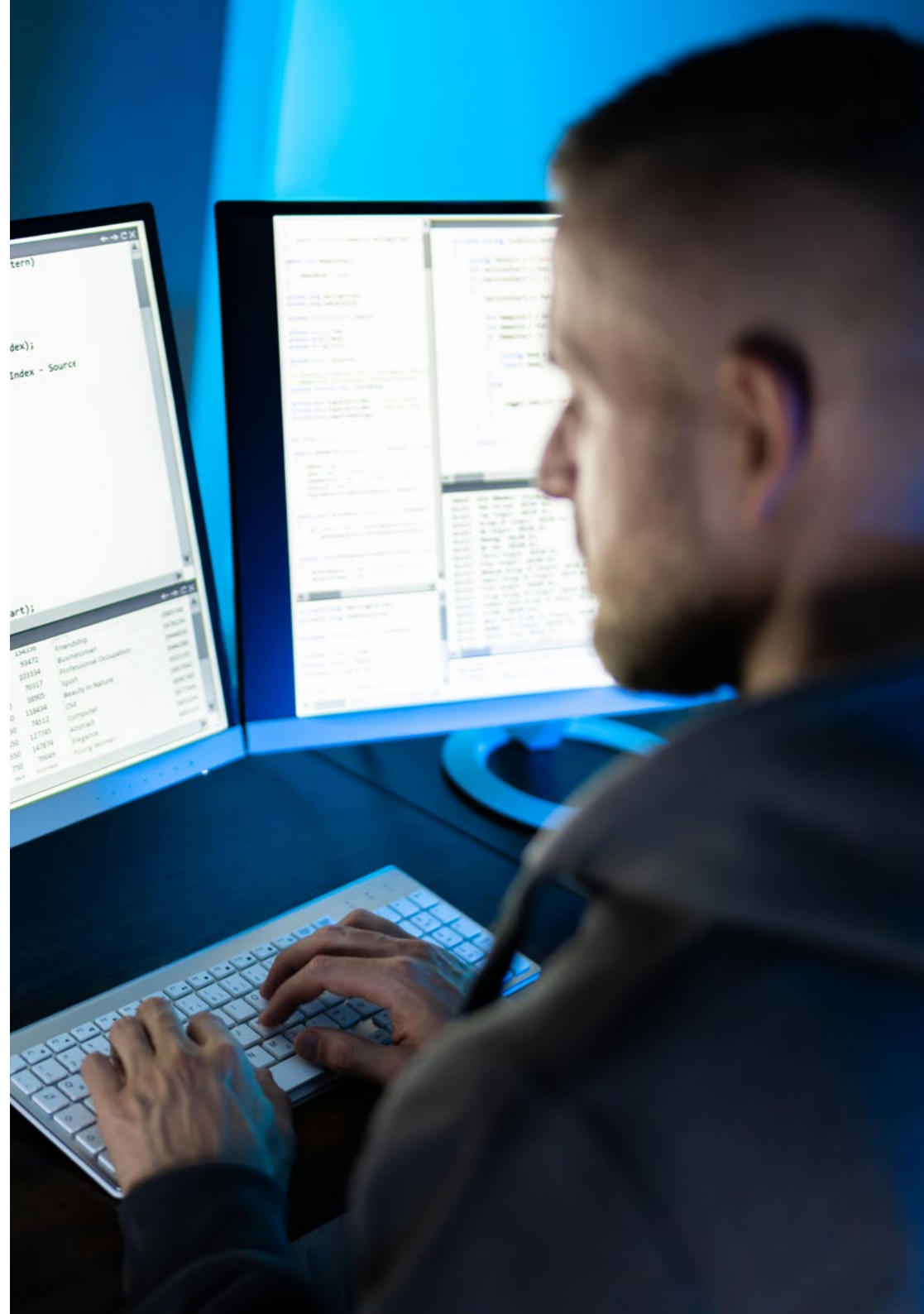


Compétences générales

- ♦ Appliquer les extensions de l'IA dans Visual Studio Code et les techniques de conception *non codées* pour accroître l'efficacité du développement de logiciels
- ♦ Utiliser ChatGPT pour optimiser et améliorer la qualité du code, en appliquant des pratiques de programmation avancées
- ♦ Mettre en œuvre des projets web, de la création de *workspaces* au déploiement, en intégrant l'IA à la fois sur le *frontend* et le *backend*
- ♦ Développer des applications mobiles alimentées par l'IA, de la configuration de l'environnement à la création de fonctionnalités avancées et à la gestion des ressources graphiques
- ♦ Appliquer des concepts de stockage avancés et des structures de données alimentées par l'IA pour améliorer l'efficacité et l'évolutivité des systèmes
- ♦ Inclure des pratiques de développement sécurisées, en évitant les vulnérabilités telles que l'injection, afin de garantir l'intégrité et la sécurité des logiciels développés



Vous maîtriserez les technologies du futur avec cette formation universitaire unique 100% en ligne. Seulement avec TECH!"





Compétences spécifiques

- ♦ Appliquer des techniques et des stratégies d'IA pour améliorer l'efficacité dans le secteur du *retail*
- ♦ Mettre en œuvre des techniques de débruitage à l'aide d'encodeurs automatiques
- ♦ Créer efficacement des ensembles de données d'entraînement pour les tâches de traitement du langage naturel (NLP)
- ♦ Exécuter des couches de clustering et leur utilisation dans des modèles de *Deep Computer Vision* avec Keras
- ♦ Utiliser les fonctions et les graphes de *TensorFlow* pour optimiser les performances des modèles personnalisés
- ♦ Optimiser le développement et l'application de *chatbots* et d'assistants virtuels, en comprenant leur fonctionnement et leurs applications potentielles
- ♦ Maîtriser la réutilisation des couches pré-entraînées pour optimiser et accélérer le processus d'entraînement
- ♦ Construire le premier réseau neuronal, en appliquant les concepts appris dans la pratique
- ♦ Activer le perceptron multicouche (MLP) à l'aide de la bibliothèque Keras
- ♦ Appliquer les techniques d'exploration et de prétraitement des données, en identifiant et en préparant les données pour une utilisation efficace dans les modèles d'apprentissage automatique
- ♦ Étudier les langages et les logiciels pour la création d'ontologies, en utilisant des outils spécifiques pour le développement de modèles sémantiques
- ♦ Développer des techniques de nettoyage des données pour garantir la qualité et l'exactitude des informations et développer les outils nécessaires au développement de modèles sémantiques
- ♦ Maîtriser la configuration d'environnements de développement optimisés, garantissant que les étudiants peuvent créer des environnements propices aux projets d'IA
- ♦ Appliquer des techniques spécifiques utilisant ChatGPT pour l'identification et la correction automatiques des améliorations possibles du code, en encourageant des pratiques de programmation plus efficaces
- ♦ Créer des tests automatisés, en particulier dans les environnements web et mobiles, en intégrant des outils d' IA pour améliorer l'efficacité du processus.
- ♦ Utiliser des outils avancés de QA alimentés par IA pour une détection plus efficace des bugs et une amélioration continue des logiciels
- ♦ Intégrer l'IA dans l'informatique en nuage, permettant aux étudiants de créer des projets web hautement évolutifs et efficaces
- ♦ Configurer le framework pour les applications mobiles et utiliser Github Copilot pour rationaliser le processus de développement

04

Direction de la formation

Fidèle à son engagement d'offrir un enseignement de la plus haute qualité, TECH dispose d'un corps enseignant du plus haut niveau. Pour cette raison, les experts qui composent cette formation universitaire se distinguent par leurs connaissances approfondies de l'IA en Programmation, ainsi que par leurs années d'expérience professionnelle. Grâce à ces aspects, ce parcours académique met à la disposition des étudiants les meilleurs outils et stratégies technologiques, non seulement pour élargir leur compréhension, mais aussi pour acquérir les compétences nécessaires au perfectionnement de leurs procédures informatiques. Les étudiants ont ainsi les garanties nécessaires pour se spécialiser dans un secteur numérique qui offre de nombreuses opportunités.



“

Les meilleurs experts en Apprentissage Automatique en Programmation se sont réunis pour partager avec vous toutes leurs connaissances dans ce secteur”

Direction



Dr Peralta Martín-Palomino, Arturo

- ♦ CEO et CTO de Prometeus Global Solutions
- ♦ CTO chez Korporate Technologies
- ♦ CTO de AI Shephers GmbH
- ♦ Consultant et Conseiller Stratégique auprès d'Alliance Medical
- ♦ Directeur de la Conception et du Développement chez DocPath
- ♦ Docteur en Ingénierie de Informatique de l'Université de Castille - La Manche
- ♦ Doctorat en économie, commerce et finances de l'Université Camilo José Cela
- ♦ Docteur en Psychologie, Université de Castille - la Manche
- ♦ Master en Executive MBA de l'Université Isabel I
- ♦ Master en Business and Marketing Management par l'Université Isabel I
- ♦ Master en Big Data en Formation Hadoop
- ♦ Master en Technologies Avancées de l'Information de l'Université de Castille - la Manche
- ♦ Membre de: Groupe de Recherche SMILE



M. Castellanos Herreros, Ricardo

- ♦ Spécialiste en Ingénierie des Systèmes Informatiques
- ♦ *Chief Technology Officer* chez OWQLO
- ♦ Consultant Technique *Indépendant*
- ♦ Développeur des Applications Mobiles pour eDreams, Fnac, Air Europa, Bankia, Cetelem, Banque Santander, Santillana, Groupón et Groupe Planeta
- ♦ Développeur de Sites Webs pour Openbank et Banque Santander
- ♦ Cours de *Machine Learning Engineer* à Udacity
- ♦ Ingénieur Technique en Systèmes Informatiques de l'Université de Castille la Manche

05

Structure et contenu

Ce Mastère Spécialisé offrira aux diplômés une approche holistique qui leur donnera un avantage significatif dans le développement informatique en les dotant de compétences spécifiques. Pour ce faire, la formation ira de la préparation de l'environnement de développement à l'optimisation des *logiciels* et à la mise en œuvre de l'IA dans des projets réels. Le programme d'études abordera des aspects tels que la conception d'interfaces *non code* l'utilisation de ChatGPT pour optimiser les codes ou l'application de l'Apprentissage Automatique dans les QA *Testing*. Ainsi, les diplômés mettront en œuvre des solutions innovantes de manière efficace dans diverses applications telles que les projets web et mobiles.



“

Actualisez vos connaissances sur l'Intelligence Artificielle dans la Programmation grâce à un contenu multimédia innovant”

Module 1. Principes fondamentaux de l'Intelligence Artificielle

- 1.1. Histoire de l'Intelligence artificielle
 - 1.1.1. Quand avons-nous commencé à parler d'intelligence artificielle ?
 - 1.1.2. Références dans le cinéma
 - 1.1.3. Importance de l'intelligence artificielle
 - 1.1.4. Technologies habilitantes et de soutien pour l'intelligence artificielle
- 1.2. L'Intelligence Artificielle dans les jeux
 - 1.2.1. La théorie des Jeux
 - 1.2.2. *Minimax* et élagage Alpha-Beta
 - 1.2.3. Simulation: Monte Carlo
- 1.3. Réseaux neuronaux
 - 1.3.1. Fondements biologiques
 - 1.3.2. Modèle de calcul
 - 1.3.3. Réseaux neuronaux supervisés et non supervisés
 - 1.3.4. Perceptron simple
 - 1.3.5. Perceptron multicouche
- 1.4. Algorithmes génétiques
 - 1.4.1. Histoire
 - 1.4.2. Base biologique
 - 1.4.3. Codification des problèmes
 - 1.4.4. Génération de la population initiale
 - 1.4.5. Algorithme principal et opérateurs génétiques
 - 1.4.6. Évaluation des individus: Fitness
- 1.5. Thésaurus, vocabulaires, taxonomies
 - 1.5.1. Vocabulaire
 - 1.5.2. Taxonomie
 - 1.5.3. Thésaurus
 - 1.5.4. Ontologies
 - 1.5.5. Représentation des connaissances: web sémantique
- 1.6. Web sémantique
 - 1.6.1. Spécifications: RDF, RDFS et OWL
 - 1.6.2. Inférence/raisonnement
 - 1.6.3. *Linked Data*

- 1.7. Systèmes experts et DSS
 - 1.7.1. Systèmes experts
 - 1.7.2. Systèmes d'aide à la décision
- 1.8. *Chatbots* et assistants virtuels
 - 1.8.1. Types d'assistants: assistants vocaux et textuels
 - 1.8.2. Éléments fondamentaux pour le développement d'un assistant: *Intents*, entités et flux de dialogue
 - 1.8.3. Intégrations: web, *Slack*, Whatsapp, Facebook
 - 1.8.4. Outils d'aide au développement: *Dialog Flow*, *Watson Assistant*
- 1.9. Stratégie de mise en œuvre de l'IA
- 1.10. L'avenir de l'intelligence artificielle
 - 1.10.1. Nous comprenons comment détecter les émotions grâce aux algorithmes
 - 1.10.2. Créer une personnalité: langage, expressions et contenu
 - 1.10.3. Tendances de l'intelligence artificielle
 - 1.10.4. Réflexion

Module 2. Types de données et cycle de vie des données

- 2.1. Statistiques
 - 2.1.1. Statistiques: statistiques descriptives, inférences statistiques
 - 2.1.2. Population, échantillon, individu
 - 2.1.3. Variables: définition, échelles de mesure
- 2.2. Types de données statistiques
 - 2.2.1. Selon le type
 - 2.2.1.1. Quantitatif: données continues et données discrètes
 - 2.2.1.2. Qualitatif: données binomiales, données nominales et données ordinales
 - 2.2.2. Selon la forme
 - 2.2.2.1. Numérique
 - 2.2.2.2. Texte
 - 2.2.2.3. Logique
 - 2.2.3. Selon la source
 - 2.2.3.1. Primaire
 - 2.2.3.2. Secondaire

- 2.3. Cycle de vie des données
 - 2.3.1. Étape de cycle
 - 2.3.2. Les étapes du cycle
 - 2.3.3. Les principes du FAIR
- 2.4. Les premières étapes du cycle
 - 2.4.1. Définition des objectifs
 - 2.4.2. Détermination des besoins en ressources
 - 2.4.3. Diagramme de Gantt
 - 2.4.4. Structure des données
- 2.5. Collecte des données
 - 2.5.1. Méthodologie de collecte
 - 2.5.2. Outils de collecte
 - 2.5.3. Canaux de collecte
- 2.6. Nettoyage des données
 - 2.6.1. Phases du nettoyage des données
 - 2.6.2. Qualité des données
 - 2.6.3. Manipulation des données (avec R)
- 2.7. Analyse des données, interprétations, évaluation des résultats
 - 2.7.1. Mesures statistiques
 - 2.7.2. Indices de ratios
 - 2.7.3. Extraction de données
- 2.8. Entrepôt de données (*Datawarehouse*)
 - 2.8.1. Les éléments qui le composent
 - 2.8.2. Conception
 - 2.8.3. Aspects à prendre en compte
- 2.9. Disponibilité des données
 - 2.9.1. Accès
 - 2.9.2. Utilité
 - 2.9.3. Sécurité
- 2.10. Aspects réglementaires
 - 2.10.1. Loi sur la protection des données
 - 2.10.2. Bonnes pratiques
 - 2.10.3. Autres aspects réglementaires

Module 3. Les données en Intelligence Artificielle

- 3.1. Science des données
 - 3.1.1. La science des données
 - 3.1.2. Outils avancés pour le scientifique des données
- 3.2. Données, informations et connaissances
 - 3.2.1. Données, informations et connaissances
 - 3.2.2. Types de données
 - 3.2.3. Sources des données
- 3.3. Des données aux informations
 - 3.3.1. Analyse des Données
 - 3.3.2. Types d'analyse
 - 3.3.3. Extraction d'informations d'un *dataset*
- 3.4. Extraction d'informations par la visualisation
 - 3.4.1. La visualisation comme outils d'analyse
 - 3.4.2. Méthodes de visualisation
 - 3.4.3. Visualisation d'un ensemble de données
- 3.5. Qualité des données
 - 3.5.1. Données de qualités
 - 3.5.2. Nettoyage des données
 - 3.5.3. Prétraitement de base des données
- 3.6. *Dataset*
 - 3.6.1. Enrichissement du *Dataset*
 - 3.6.2. La malédiction de la dimensionnalité
 - 3.6.3. Modification d'un ensemble de données
- 3.7. Déséquilibre
 - 3.7.1. Déséquilibre des classes
 - 3.7.2. Techniques d'atténuation du déséquilibre
 - 3.7.3. Équilibrer un *Dataset*
- 3.8. Modèles non supervisé
 - 3.8.1. Modèles non supervisé
 - 3.8.2. Méthodes
 - 3.8.3. Classifications avec modèles non supervisé

- 3.9. Modèles supervisés
 - 3.9.1. Modèles supervisé
 - 3.9.2. Méthodes
 - 3.9.3. Classifications avec modèles supervisés
- 3.10. Outils et bonnes pratiques
 - 3.10.1. Bonnes pratiques pour un scientifique des données
 - 3.10.2. Le meilleur modèle
 - 3.10.3. Outils utiles

Module 4. Extraction de données. Sélection, prétraitement et transformation

- 4.1. Inférence statistique
 - 4.1.1. Statistique descriptive vs. Inférence Statistique
 - 4.1.2. Procédures paramétriques
 - 4.1.3. Procédures non paramétriques
- 4.2. Analyse exploratoire
 - 4.2.1. Analyse descriptive
 - 4.2.2. Visualisation
 - 4.2.3. Préparations des données
- 4.3. Préparations des données
 - 4.3.1. Intégration et nettoyage des données
 - 4.3.2. Normalisation des données
 - 4.3.3. Transformer les attributs
- 4.4. Valeurs manquantes
 - 4.4.1. Traitement des valeurs manquantes
 - 4.4.2. Méthodes d'imputation par maximum de vraisemblance
 - 4.4.3. Imputation des valeurs manquantes à l'aide de l'apprentissage automatique
- 4.5. Bruit dans les données
 - 4.5.1. Classes et attributs de bruit
 - 4.5.2. Filtrage du bruit
 - 4.5.3. L'effet du bruit
- 4.6. La malédiction de la dimensionnalité
 - 4.6.1. *Oversampling*
 - 4.6.2. *Undersampling*
 - 4.6.3. Réduction des données multidimensionnelles

- 4.7. Des attributs continus aux attributs discrets
 - 4.7.1. Données continues ou discrètes
 - 4.7.2. Processus de discrétisation
- 4.8. Les données
 - 4.8.1. Sélection des données
 - 4.8.2. Perspectives et critères de sélections
 - 4.8.3. Méthodes de sélection
- 4.9. Sélection des instances
 - 4.9.1. Méthodes de sélection des instances
 - 4.9.2. Sélection des prototypes
 - 4.9.3. Méthodes avancées de sélection des instances
- 4.10. Prétraitement des données dans les environnements Big Data

Module 5. Algorithmes et complexité en Intelligence Artificielle

- 5.1. Introduction aux stratégies de conception d'algorithmes
 - 5.1.1. Récursion
 - 5.1.2. Diviser pour mieux régner
 - 5.1.3. Autres stratégies
- 5.2. Efficacité et analyse des algorithmes
 - 5.2.1. Mesures d'efficacité
 - 5.2.2. Taille de l'entrée de mesure
 - 5.2.3. Mesure du temps d'exécution
 - 5.2.4. Pire, meilleur et moyen cas
 - 5.2.5. Notation asymptotique
 - 5.2.6. Critères d'Analyse mathématique des algorithmes non récursifs
 - 5.2.7. Analyse mathématique des algorithmes récursifs
 - 5.2.8. Analyse empirique des algorithmes
- 5.3. Algorithmes de tri
 - 5.3.1. Concept de tri
 - 5.3.2. Triage des bulles
 - 5.3.3. Tri par sélection
 - 5.3.4. Triage par insertion
 - 5.3.5. Tri fusion (*Merge_Sort*)
 - 5.3.6. Tri rapide (*Quick_Sort*)

- 5.4. Algorithmes avec arbres
 - 5.4.1. Concept d'arbre
 - 5.4.2. Arbres binaires
 - 5.4.3. Allées d'arbres
 - 5.4.4. Représentation des expressions
 - 5.4.5. Arbres binaires ordonnés
 - 5.4.6. Arbres binaires équilibrés
- 5.5. Algorithmes avec *Heaps*
 - 5.5.1. Les *Heaps*
 - 5.5.2. L'algorithme *Heapsort*
 - 5.5.3. Files d'attente prioritaires
- 5.6. Algorithmes graphiques
 - 5.6.1. Représentation
 - 5.6.2. Voyage en largeur
 - 5.6.3. Profondeur de déplacement
 - 5.6.4. Disposition topologique
- 5.7. Algorithmes *Greedy*
 - 5.7.1. La stratégie *Greedy*
 - 5.7.2. Éléments de la stratégie *Greedy*
 - 5.7.3. Change de devises
 - 5.7.4. Le problème du voyageur
 - 5.7.5. Problème de sac à dos
- 5.8. Recherche de chemins minimaux
 - 5.8.1. Le problème du chemin minimal
 - 5.8.2. Arcs et cycles négatifs
 - 5.8.3. Algorithme de Dijkstra
- 5.9. Algorithmes *Greedy* sur les graphes
 - 5.9.1. L'arbre à chevauchement minimal
 - 5.9.2. L'algorithme de Prim
 - 5.9.3. L'algorithme de Kruskal
 - 5.9.4. Analyse de la complexité
- 5.10. *Backtracking*
 - 5.10.1. Le *Backtracking*
 - 5.10.2. Techniques alternatives

Module 6. Systèmes intelligents

- 6.1. Théorie des agents
 - 6.1.1. Histoire du concept
 - 6.1.2. Définition d'agent
 - 6.1.3. Les agents en Intelligence Artificielle
 - 6.1.4. Les agents en génie de Software
- 6.2. Architectures des agents
 - 6.2.1. Le processus de raisonnement d'un agent
 - 6.2.2. Agents réactifs
 - 6.2.3. Agents déductifs
 - 6.2.4. Agents hybrides
 - 6.2.5. Comparaison
- 6.3. Information et connaissance
 - 6.3.1. Distinction entre données, informations et connaissances
 - 6.3.2. Évaluation de la qualité des données
 - 6.3.3. Méthode de capture des données
 - 6.3.4. Méthodes d'acquisition des informations
 - 6.3.5. Méthodes d'acquisition des connaissances
- 6.4. Représentation des connaissances
 - 6.4.1. L'importance de la représentation de la connaissance
 - 6.4.2. Définition de la représentation des connaissances à travers leurs rôles
 - 6.4.3. Caractéristiques de la représentation de la connaissance
- 6.5. Ontologies
 - 6.5.1. Introduction aux Métadonnées
 - 6.5.2. Concept philosophique d'ontologie
 - 6.5.3. Concept informatique d'ontologie
 - 6.5.4. Ontologies de domaine et ontologies de niveau supérieur
 - 6.5.5. Comment construire une ontologie ?
- 6.6. Langages d'ontologie et logiciels de création d'ontologies
 - 6.6.1. Triplés RDF, *Turtle* et N
 - 6.6.2. RDF *Schema*
 - 6.6.3. OWL
 - 6.6.4. SPARQL
 - 6.6.5. Introduction aux différents outils de création d'ontologies
 - 6.6.6. Installation et utilisation du *Protégé*

- 6.7. Le web sémantique
 - 6.7.1. L'état actuel et futur du web sémantique
 - 6.7.2. Applications du web sémantique
- 6.8. Autres modèles de représentation des connaissances
 - 6.8.1. Vocabulaire
 - 6.8.2. Vision globale
 - 6.8.3. Taxonomie
 - 6.8.4. Thésaurus
 - 6.8.5. Folksonomies
 - 6.8.6. Comparaison
 - 6.8.7. Cartes mentales
- 6.9. Évaluation et intégration des représentations des connaissances
 - 6.9.1. Logique d'ordre zéro
 - 6.9.2. Logique de premier ordre
 - 6.9.3. Logique descriptive
 - 6.9.4. Relations entre les différents types de logique
 - 6.9.5. *Prolog*: programmation basée sur la logique du premier ordre
- 6.10. Raisonneurs sémantiques, systèmes à base de connaissances et systèmes experts
 - 6.10.1. Concept de raisonneur
 - 6.10.2. Application d'un raisonneur
 - 6.10.3. Systèmes basés sur la connaissance
 - 6.10.4. MYCIN, histoire des Systèmes Experts
 - 6.10.5. Éléments et Architecture des Systèmes Experts
 - 6.10.6. Création de Systèmes Experts

Module 7. Apprentissage automatique et exploration des données

- 7.1. Introduction à les processus de découverte des connaissances et les concepts de base de l'apprentissage automatique
 - 7.1.1. Concepts clés du processus de découverte de connaissances
 - 7.1.2. Perspective historique du processus de découverte de connaissances
 - 7.1.3. Étapes du processus de découverte de connaissances
 - 7.1.4. Techniques utilisées dans les processus de découverte de connaissances
 - 7.1.5. Caractéristiques des bons modèles d'apprentissage automatique
 - 7.1.6. Types d'informations sur l'apprentissage automatique
 - 7.1.7. Concepts de base de l'apprentissage
 - 7.1.8. Concepts de base de l'apprentissage non supervisé

- 7.2. Exploration et prétraitement des données
 - 7.2.1. Traitement des données
 - 7.2.2. Traitement des données dans le flux d'analyse des données
 - 7.2.3. Types de données
 - 7.2.4. Transformations de données
 - 7.2.5. Affichage et exploration des variables continues
 - 7.2.6. Affichage et exploration des variables catégorielles
 - 7.2.7. Mesures de corrélation
 - 7.2.8. Représentations graphiques les plus courantes
 - 7.2.9. Introduction à l'analyse multivariée et à la réduction des dimensions
- 7.3. Arbres de décision
 - 7.3.1. Algorithme ID
 - 7.3.2. Algorithme C
 - 7.3.3. Surentraînement et taillage
 - 7.3.4. Analyse des résultats
- 7.4. Évaluation des classificateurs
 - 7.4.1. Matrices de confusion
 - 7.4.2. Matrices d'évaluation numérique
 - 7.4.3. Statistique de Kappa
 - 7.4.4. La courbe ROC
- 7.5. Règles de classification
 - 7.5.1. Mesures d'évaluation des règles
 - 7.5.2. Introduction à la représentation graphique
 - 7.5.3. Algorithme de superposition séquentielle
- 7.6. Réseaux neuronaux
 - 7.6.1. Concepts de base
 - 7.6.2. Réseaux neuronaux simples
 - 7.6.3. Algorithme de *Backpropagation*
 - 7.6.4. Introduction aux réseaux neuronaux récurrents
- 7.7. Méthodes bayésiennes
 - 7.7.1. Concepts de base des probabilités
 - 7.7.2. Théorème de Bayes
 - 7.7.3. Naive Bayes
 - 7.7.4. Introduction aux réseaux bayésiens

- 7.8. Modèles de régression et de réponse continue
 - 7.8.1. Régression linéaire simple
 - 7.8.2. Régression linéaire multiple
 - 7.8.3. Régression logistique
 - 7.8.4. Arbres de régression
 - 7.8.5. Introduction aux machines à vecteurs de support (SVM)
 - 7.8.6. Mesures de qualité de l'ajustement
- 7.9. *Clustering*
 - 7.9.1. Concepts de base
 - 7.9.2. *Clustering* hiérarché
 - 7.9.3. Méthodes probabilistes
 - 7.9.4. Algorithme EM
 - 7.9.5. Méthode *B-Cubed*
 - 7.9.6. Méthodes implicites
- 7.10. Exploration de textes et traitement du langage naturel (NLP)
 - 7.10.1. Concepts de base
 - 7.10.2. Création du corpus
 - 7.10.3. Analyse descriptive
 - 7.10.4. Introduction à l'analyse des sentiments

Module 8. Les Réseaux Neuronaux, la base du *Deep Learning*

- 8.1. Apprentissage profond
 - 8.1.1. Types d'apprentissage profond
 - 8.1.2. Applications de l'apprentissage profond
 - 8.1.3. Avantages et Inconvénients de l'apprentissage profond
- 8.2. Opérations
 - 8.2.1. Somme
 - 8.2.2. Produit
 - 8.2.3. Transfert
- 8.3. Couches
 - 8.3.1. Couche d'entrée
 - 8.3.2. Couche cachée
 - 8.3.3. Couche de sortie

- 8.4. Liaison des couches et opérations
 - 8.4.1. Conception des architectures
 - 8.4.2. Connexion entre les couches
 - 8.4.3. Propagation vers l'avant
- 8.5. Construction du premier réseau neuronal
 - 8.5.1. Conception du réseau
 - 8.5.2. Établissement des poids
 - 8.5.3. Entraînement du réseau
- 8.6. Entraîneur et optimiseur
 - 8.6.1. Sélection de l'optimiseur
 - 8.6.2. Établissement d'une fonction de perte
 - 8.6.3. Établissement d'une métrique
- 8.7. Application des principes des réseaux neuronaux
 - 8.7.1. Fonctions d'Activation
 - 8.7.2. Propagation à rebours
 - 8.7.3. Paramétrage
- 8.8. Des neurones biologiques aux neurones artificiels
 - 8.8.1. Fonctionnement d'un neurone biologique
 - 8.8.2. Transfert de connaissances aux neurones artificiels
 - 8.8.3. Établissement de relations entre les deux
- 8.9. Mise en œuvre du MLP (Perceptron Multicouche) avec Keras
 - 8.9.1. Définition de la structure du réseau
 - 8.9.2. Compilation du modèle
 - 8.9.3. Formation au modèle
- 8.10. Hyperparamètres de *Fine tuning* des Réseaux Neuronaux
 - 8.10.1. Sélection de la fonction d'activation
 - 8.10.2. Réglage du *Learning rate*
 - 8.10.3. Réglage des poids

Module 9. Entraînement de réseaux neuronaux profonds

- 9.1. Problèmes de gradient
 - 9.1.1. Techniques d'optimisation du gradient
 - 9.1.2. Gradients stochastiques
 - 9.1.3. Techniques d'initialisation des poids

- 9.2. Réutilisation des couches pré-entraînées
 - 9.2.1. Entraînement par transfert d'apprentissage
 - 9.2.2. Extraction de caractéristiques
 - 9.2.3. Apprentissage profond
- 9.3. Optimisateurs
 - 9.3.1. Optimiseurs stochastiques à descente de gradient
 - 9.3.2. Optimiseurs Adam et *RMSprop*
 - 9.3.3. Optimiseurs de moment
- 9.4. Programmation du taux de d'apprentissage
 - 9.4.1. Contrôle automatique du taux d'apprentissage
 - 9.4.2. Cycles d'apprentissage
 - 9.4.3. Termes de lissage
- 9.5. Surajustement
 - 9.5.1. Validation croisée.
 - 9.5.2. Régularisation
 - 9.5.3. Mesures d'évaluation
- 9.6. Lignes directrices pratiques
 - 9.6.1. Conception de modèles
 - 9.6.2. Sélection des métriques et des paramètres d'évaluation
 - 9.6.3. Tests d'hypothèses
- 9.7. *Transfer Learning*
 - 9.7.1. Entraînement par transfert d'apprentissage
 - 9.7.2. Extraction de caractéristiques
 - 9.7.3. Apprentissage profond
- 9.8. *Data Augmentation*
 - 9.8.1. Transformation d'image
 - 9.8.2. Génération de données synthétiques
 - 9.8.3. Transformation de texte
- 9.9. Application pratique du *Transfer Learning*
 - 9.9.1. Entraînement par transfert d'apprentissage
 - 9.9.2. Extraction de caractéristiques
 - 9.9.3. Apprentissage profond

- 9.10. Régularisation
 - 9.10.1. L et L
 - 9.10.2. Régularisation par entropie maximale
 - 9.10.3. *Dropout*

Module 10. Personnaliser les modèles et l'entraînement avec *TensorFlow*

- 10.1. *TensorFlow*
 - 10.1.1. Utilisation de la bibliothèque *TensorFlow*
 - 10.1.2. Entraînement des modèles avec *TensorFlow*
 - 10.1.3. Opérations avec les graphes dans *TensorFlow*
- 10.2. *TensorFlow* et NumPy
 - 10.2.1. Environnement de calcul NumPy pour *TensorFlow*
 - 10.2.2. Utilisation des tableaux NumPy avec *TensorFlow*
 - 10.2.3. Opérations NumPy pour les graphes *TensorFlow*
- 10.3. Personnalisation des modèles et des algorithmes d'apprentissage
 - 10.3.1. Construire des modèles personnalisés avec *TensorFlow*
 - 10.3.2. Gestion des paramètres d'entraînement
 - 10.3.3. Utilisation de techniques d'optimisation pour l'entraînement
- 10.4. Fonctions et graphiques *TensorFlow*
 - 10.4.1. Fonctions avec *TensorFlow*
 - 10.4.2. Utilisation des graphes pour l'apprentissage des modèles
 - 10.4.3. Optimisation des graphes avec les opérations *TensorFlow*
- 10.5. Chargement des données et prétraitement avec *TensorFlow*
 - 10.5.1. Chargement des données d'ensembles avec *TensorFlow*
 - 10.5.2. Prétraitement des données avec *TensorFlow*
 - 10.5.3. Utilisation des outils *TensorFlow* pour la manipulation des données
- 10.6. L'API *tf.data*
 - 10.6.1. Utilisation de l'API *tf.data* pour le traitement des données
 - 10.6.2. Construction des flux de données avec *tf.data*
 - 10.6.3. Utilisation de l'API *tf.data* pour l'entraînement des modèles
- 10.7. Le format *TFRecord*
 - 10.7.1. Utilisation de l'API *TFRecord* pour la sérialisation des données
 - 10.7.2. Chargement de fichiers *TFRecord* avec *TensorFlow*
 - 10.7.3. Utilisation des fichiers *TFRecord* pour l'entraînement des modèles

- 10.8. Couches de prétraitement Keras
 - 10.8.1. Utilisation de l'API de prétraitement Keras
 - 10.8.2. Construire un prétraitement en *pipeline* avec Keras
 - 10.8.3. Utilisation de l'API de prétraitement Keras pour l'entraînement des modèles
- 10.9. Le projet *TensorFlow Datasets*
 - 10.9.1. Utilisation de *TensorFlow Datasets* pour le chargement des données
 - 10.9.2. Prétraitement des données avec *TensorFlow Datasets*
 - 10.9.3. Utilisation de *TensorFlow Datasets* pour l'entraînement des modèles
- 10.10. Construire une application de Deep Learning avec *TensorFlow*
 - 10.10.1. Application Pratique
 - 10.10.2. Construire une application de Deep Learning avec *TensorFlow*
 - 10.10.3. Entraînement des modèles avec *TensorFlow*
 - 10.10.4. Utilisation de l'application pour la prédiction des résultats

Module 11. Deep Computer Vision avec les Réseaux Neuronaux Convolutifs

- 11.1. L'Architecture *Visual Cortex*
 - 11.1.1. Fonctions du cortex visuel
 - 11.1.2. Théorie de la vision computationnelle
 - 11.1.3. Modèles de traitement des images
- 11.2. Couches convolutives
 - 11.2.1. Réutilisation des poids dans la convolution
 - 11.2.2. Convolution D
 - 11.2.3. Fonctions d'Activation
- 11.3. Couches de regroupement et implémentation des couches de regroupement avec Keras
 - 11.3.1. *Pooling* et *Striding*
 - 11.3.2. *Flattening*
 - 11.3.3. Types de *Pooling*
- 11.4. Architecture du CNN
 - 11.4.1. Architecture du VGG
 - 11.4.2. Architecture *AlexNet*
 - 11.4.3. Architecture *ResNet*
- 11.5. Mise en œuvre d'un CNN *ResNet*- à l'aide de Keras
 - 11.5.1. Initialisation des poids
 - 11.5.2. Définition de la couche d'entrée
 - 11.5.3. Définition de la sortie

- 11.6. Utilisation de modèles Keras pré-entraînés
 - 11.6.1. Caractéristiques des modèles pré-entraînés
 - 11.6.2. Utilisations des modèles pré-entraînés
 - 11.6.3. Avantages des modèles pré-entraînés
- 11.7. Modèles pré-entraînés pour l'apprentissage par transfert
 - 11.7.1. Apprentissage par transfert
 - 11.7.2. Processus d'apprentissage par transfert
 - 11.7.3. Avantages de l'apprentissage par transfert
- 11.8. Classification et Localisation en *Deep Computer Vision*
 - 11.8.1. Classification des images
 - 11.8.2. Localisation d'objets dans les images
 - 11.8.3. Détection d'objets
- 11.9. Détection et suivi d'objets
 - 11.9.1. Méthodes de détection d'objets
 - 11.9.2. Algorithmes de suivi d'objets
 - 11.9.3. Techniques de suivi et de localisation
- 11.10. Segmentation sémantique
 - 11.10.1. Apprentissage profond pour la segmentation sémantique
 - 11.10.2. Détection des bords
 - 11.10.3. Méthodes de segmentation basées sur des règles

Module 12. Traitement du langage naturel (NLP) avec les Réseaux Récurrents Naturels (NNN) et l'Attention

- 12.1. Génération de texte à l'aide de RNN
 - 12.1.1. Formation d'un RNN pour la génération de texte
 - 12.1.2. Génération de langage naturel avec RNN
 - 12.1.3. Applications de génération de texte avec RNN
- 12.2. Création d'ensembles de données d'entraînement
 - 12.2.1. Préparation des données pour l'entraînement des RNN
 - 12.2.2. Stockage de l'ensemble de données de formation
 - 12.2.3. Nettoyage et transformation des données
 - 12.2.4. Analyse des Sentiments
- 12.3. Classement des opinions avec RNN
 - 12.3.1. Détection des problèmes dans les commentaires
 - 12.3.2. Analyse des sentiments à l'aide d'algorithmes d'apprentissage profond

- 12.4. Réseau encodeur-décodeur pour la traduction automatique neuronale
 - 12.4.1. Formation d'un RNN pour la traduction automatique
 - 12.4.2. Utilisation d'un réseau *encoder-decoder* pour la traduction automatique
 - 12.4.3. Améliorer la précision de la traduction automatique avec les RNN
- 12.5. Mécanismes de l'attention
 - 12.5.1. Application de mécanismes de l'attention avec les RNN
 - 12.5.2. Utilisation de mécanismes d'attention pour améliorer la précision des modèles
 - 12.5.3. Avantages des mécanismes d'attention dans les réseaux neuronaux
- 12.6. Modèles *Transformers*
 - 12.6.1. Utilisation des modèles *Transformers* pour le traitement du langage naturel
 - 12.6.2. Application des modèles *Transformers* pour la vision
 - 12.6.3. Avantages des modèles *Transformers*
- 12.7. *Transformers* pour la vision
 - 12.7.1. Utilisation des modèles *Transformers* pour la vision
 - 12.7.2. Prétraitement des données d'imagerie
 - 12.7.3. Entraînement de modèle *Transformers* pour la vision
- 12.8. Bibliothèque de *Transformers* de *Hugging Face*
 - 12.8.1. Utilisation de la bibliothèque de *Transformers* de *Hugging Face*
 - 12.8.2. Application de la bibliothèque de *Transformers* de *Hugging Face*
 - 12.8.3. Avantages de la bibliothèque de *Transformers* de *Hugging Face*
- 12.9. Autres bibliothèques de *Transformers*. Comparaison
 - 12.9.1. Comparaison entre les bibliothèques de *Transformers*
 - 12.9.2. Utilisation de bibliothèques de *Transformers*
 - 12.9.3. Avantages des bibliothèques de *Transformers*
- 12.10. Développement d'une Application NLP avec RNN et Atención Application Pratique
 - 12.10.1. Développer une application du traitement du langage naturel à l'aide de RNN et de l'attention
 - 12.10.2. Utilisation des RNN, des mécanismes de soins et des modèles *Transformers* dans l'application
 - 12.10.3. Évaluation de l'application pratique

Module 13. Autoencodeurs, GANs, et Modèles de Diffusion

- 13.1. Représentation des données efficaces
 - 13.1.1. Réduction de la dimensionnalité
 - 13.1.2. Apprentissage profond
 - 13.1.3. Représentations compactes
- 13.2. Réalisation de PCA avec un codeur automatique linéaire incomplet
 - 13.2.1. Processus d'apprentissage
 - 13.2.2. Implémentation Python
 - 13.2.3. Utilisation des données de test
- 13.3. Codeurs automatiques empilés
 - 13.3.1. Réseaux neuronaux profonds
 - 13.3.2. Construction d'architectures de codage
 - 13.3.3. Utilisation de la régularisation
- 13.4. Auto-encodeurs convolutifs
 - 13.4.1. Conception du modèle convolutionnels
 - 13.4.2. Entraînement de modèles convolutionnels
 - 13.4.3. Évaluation des résultats
- 13.5. Suppression du bruit des codeurs automatiques
 - 13.5.1. Application de filtres
 - 13.5.2. Conception de modèles de codage
 - 13.5.3. Utilisation de techniques de régularisation
- 13.6. Codeurs automatiques dispersés
 - 13.6.1. Augmentation de l'efficacité du codage
 - 13.6.2. Minimiser le nombre de paramètres
 - 13.6.3. Utiliser des techniques de régularisation
- 13.7. Codeurs automatiques variationnels
 - 13.7.1. Utilisation de l'optimisation variationnelle
 - 13.7.2. Apprentissage profond non supervisé
 - 13.7.3. Représentations latentes profondes
- 13.8. Génération d'images MNIST à la mode
 - 13.8.1. Reconnaissance des formes
 - 13.8.2. Génération d'images
 - 13.8.3. Entraînement de réseaux neuronaux profonds

- 13.9. Réseaux adversaires génératifs et modèles de diffusion
 - 13.9.1. Génération de contenu à partir d'images
 - 13.9.2. Modélisation des distributions de données
 - 13.9.3. Utilisation de réseaux contradictoires
- 13.10. Implémentation des modèles
 - 13.10.1. Application Pratique
 - 13.10.2. Implémentation des modèles
 - 13.10.3. Utilisation de données réelles
 - 13.10.4. Évaluation des résultats

Module 14. Informatique bio-inspirée

- 14.1. Introduction à l'informatique bio-inspirée
 - 14.1.1. Introduction à l'informatique bio-inspirée
- 14.2. Algorithmes d'adaptation sociale
 - 14.2.1. Calcul basé sur des colonies de fourmis bio-inspirées
 - 14.2.2. Variantes des algorithmes de colonies de fourmis
 - 14.2.3. Informatique en nuage de particules
- 14.3. Algorithmes génétiques
 - 14.3.1. Structure générale
 - 14.3.2. Implantations des principaux opérateurs
- 14.4. Stratégies d'exploration-exploitation de l'espace pour les algorithmes génétiques
 - 14.4.1. Algorithme CHC
 - 14.4.2. Problèmes multimodaux
- 14.5. Modèles de calcul évolutif (I)
 - 14.5.1. Stratégies évolutives
 - 14.5.2. Programmation évolutive
 - 14.5.3. Algorithmes basés sur l'évolution différentielle
- 14.6. Modèles de calcul évolutif (II)
 - 14.6.1. Modèles d'évolution basés sur l'estimation des distributions (EDA)
 - 14.6.2. Programmation génétique
- 14.7. Programmation évolutive appliquée aux problèmes d'apprentissage
 - 14.7.1. Apprentissage basé sur des règles
 - 14.7.2. Méthodes évolutionnaires dans les problèmes de sélection d'instances

- 14.8. Problèmes multi-objectifs
 - 14.8.1. Concept de dominance
 - 14.8.2. Application des algorithmes évolutionnaires aux problèmes multi-objectifs
- 14.9. Réseaux neuronaux (I)
 - 14.9.1. Introduction aux réseaux neuronaux
 - 14.9.2. Exemple pratique avec les réseaux neuronaux
- 14.10. Réseaux neuronaux (II)
 - 14.10.1. Cas d'utilisation des réseaux neuronaux dans la recherche médicale
 - 14.10.2. Cas d'utilisation des réseaux neuronaux en économie
 - 14.10.3. Cas d'utilisation des réseaux neuronaux en vision artificielle

Module 15. Intelligence Artificielle: Stratégies et applications

- 15.1. Services financiers
 - 15.1.1. Les implications de l'intelligence artificielle (IA) dans les services financiers. Opportunités et défis
 - 15.1.2. Cas d'utilisation
 - 15.1.3. Risques potentiels liés à l'utilisation de l'IA
 - 15.1.4. Développements / utilisations futurs potentiels de l'IA
- 15.2. Implications de l'intelligence artificielle dans les services de santé
 - 15.2.1. Implications de l'IA dans le secteur de la santé. Opportunités et défis
 - 15.2.2. Cas d'utilisation
- 15.3. Risques liés à l'utilisation de l'IA dans les services de santé
 - 15.3.1. Risques potentiels liés à l'utilisation de l'IA
 - 15.3.2. Développements / utilisations futurs potentiels de l'IA
- 15.4. *Retail*
 - 15.4.1. Implications de l'IA dans le commerce de *détail*. Opportunités et défis
 - 15.4.2. Cas d'utilisation
 - 15.4.3. Risques potentiels liés à l'utilisation de l'IA
 - 15.4.4. Développements / utilisations futurs potentiels de l'IA
- 15.5. Industrie
 - 15.5.1. Implications de l'IA dans l'Industrie. Opportunités et défis
 - 15.5.2. Cas d'utilisation

- 15.6. Risques potentiels liés à l'utilisation de l'IA dans l'Industrie
 - 15.6.1. Cas d'utilisation
 - 15.6.2. Risques potentiels liés à l'utilisation de l'IA
 - 15.6.3. Développements / utilisations futurs potentiels de l'IA
- 15.7. Administration publique
 - 15.7.1. Implications de l'IA dans l'Administration Publique. Opportunités et défis
 - 15.7.2. Cas d'utilisation
 - 15.7.3. Risques potentiels liés à l'utilisation de l'IA
 - 15.7.4. Développements / utilisations futurs potentiels de l'IA
- 15.8. Éducation
 - 15.8.1. Implications de l'IA dans l'éducation. Opportunités et défis
 - 15.8.2. Cas d'utilisation
 - 15.8.3. Risques potentiels liés à l'utilisation de l'IA
 - 15.8.4. Développements / utilisations futurs potentiels de l'IA
- 15.9. Sylviculture et agriculture
 - 15.9.1. Implications de l'IA pour la foresterie et l'agriculture. Opportunités et défis
 - 15.9.2. Cas d'utilisation
 - 15.9.3. Risques potentiels liés à l'utilisation de l'IA
 - 15.9.4. Développements / utilisations futurs potentiels de l'IA
- 15.10 Ressources Humaines
 - 15.10.1. Implications de l'IA pour les Ressources Humaines. Opportunités et défis
 - 15.10.2. Cas d'utilisation
 - 15.10.3. Risques potentiels liés à l'utilisation de l'IA
 - 15.10.4. Développements / utilisations futurs potentiels de l'IA

Module 16. Améliorer la Productivité du Développement de Logiciels grâce à l'IA

- 16.1. Préparer un environnement de développement approprié
 - 16.1.1. Sélection des outils essentiels pour le développement avec l'IA
 - 16.1.2. Configuration des outils choisis
 - 16.1.3. Mise en place de pipelines CI/CD adaptés aux projets d'IA
 - 16.1.4. Gestion efficace des dépendances et des versions dans les environnements de développement.
- 16.2. Extensions essentielles de l'IA pour Visual Studio Code
 - 16.2.1. Explorer et sélectionner les extensions d'IA pour Visual Studio Code
 - 16.2.2. Intégration d'outils d'analyse statique et dynamique dans l'IDE
 - 16.2.3. Automatisation des tâches répétitives avec des extensions spécifiques
 - 16.2.4. Personnalisation de l'environnement de développement pour améliorer l'efficacité
- 16.3. Design *Non Codé* de l'Interface Utilisateur avec des Éléments d'IA
 - 16.3.1. Principes de design *Non Codé* et leur application dans les interfaces de l'utilisateur
 - 16.3.2. Incorporation d'éléments d'IA au design de l'interface visuelle
 - 16.3.3. Outils et plates-formes pour la création *Non Codée* d'interfaces intelligentes
 - 16.3.4. Évaluation et amélioration continue des interfaces *Non Codé* grâce à l' IA
- 16.4. Optimisation du code à l'aide de ChatGPT
 - 16.4.1. Identification du code dupliqué
 - 16.4.2. Reformuler
 - 16.4.3. Créer un code lisible
 - 16.4.4. Comprendre ce que fait le code
 - 16.4.5. Améliorer le nom des variables et des fonctions
 - 16.4.6. Créer une documentation automatique
- 16.5. Gestion du Référentiel avec l'IA
 - 16.5.1. Automatisation des processus de contrôle de version avec des techniques d'IA
 - 16.5.2. Détection et résolution automatique des conflits dans les environnements collaboratifs
 - 16.5.3. Analyse prédictive des changements et des tendances dans les dépôts de code
 - 16.5.4. Amélioration de l'organisation et de la catégorisation des dépôts grâce à l'IA
- 16.6. Intégration de l'IA dans la gestion des bases de données
 - 16.6.1. Optimisation des requêtes et des performances à l'aide de techniques d'IA
 - 16.6.2. Analyse prédictive des schémas d'accès aux bases de données
 - 16.6.3. Mise en œuvre de systèmes de recommandation pour optimiser la structure de la base de données
 - 16.6.4. Surveillance et détection proactives des problèmes potentiels de la base de données
- 16.7. Recherche d'erreurs et création de tests unitaires fondés sur l'IA
 - 16.7.1. Génération automatique de cas de test à l'aide de techniques d'IA
 - 16.7.2. Détection précoce des vulnérabilités et des bogues au moyen de l'analyse statique et de l'IA
 - 16.7.3. Amélioration de la couverture des tests grâce à l'identification des zones critiques par l'IA

- 16.8. *Pair Programming* avec GitHub Copilot
 - 16.8.1. Intégration et utilisation efficace de GitHub Copilot dans les sessions de *Pair Programming*
 - 16.8.2. Intégration Amélioration de la communication et de la collaboration entre les développeurs avec GitHub Copilot
 - 16.8.3. Intégration Stratégies pour tirer le meilleur parti des suggestions de code générées par GitHub Copilot
 - 16.8.4. Intégration Études de cas et meilleures pratiques en matière de *Pair Programming* assisté par l' IA
- 16.9. Traduction automatique entre langages de programmation
 - 16.9.1. Outils et services de traduction automatique spécifiques pour les langages de programmation
 - 16.9.2. Adaptation des algorithmes de traduction automatique aux contextes de développement
 - 16.9.3. Améliorer l'interopérabilité entre différentes langues grâce à la traduction automatique
 - 16.9.4. Évaluation et atténuation des difficultés et des limites potentielles de la traduction automatique
- 16.10. Outils d'IA recommandés pour améliorer la productivité
 - 16.10.1. Analyse comparative des outils d'IA pour le développement de logiciels
 - 16.10.2. Intégration des outils d'IA dans les flux de travail
 - 16.10.3. Automatisation des tâches routinières à l'aide d'outils d'IA
 - 16.10.4. Évaluation et sélection des outils en fonction du contexte et des exigences du projet

Module 17. Architecture logicielle avec IA

- 17.1. Optimisation et gestion des performances des outils d'IA
 - 17.1.1. Profilage et analyse des performances dans les outils d'IA
 - 17.1.2. Stratégies d'optimisation des algorithmes et modèles d'IA
 - 17.1.3. Mise en œuvre de techniques de mise en *caching* et de parallélisation pour améliorer les performances
 - 17.1.4. Outils et méthodologies pour le contrôle continu des performances en temps réel
- 17.2. Évolutivité des applications d'IA
 - 17.2.1. Design d'architectures évolutives pour les applications d'IA
 - 17.2.2. Mise en œuvre de techniques de partitionnement et de répartition de la charge
 - 17.2.3. Gestion du flux de travail et de la charge de travail dans les systèmes évolutifs
 - 17.2.4. Stratégies d'expansion horizontale et verticale dans des environnements à demande variable

- 17.3. Maintenabilité des applications IA
 - 17.3.1. Principes de conception pour faciliter la maintenabilité dans les projets d'IA
 - 17.3.2. Stratégies de documentation spécifiques pour les modèles et algorithmes d'IA
 - 17.3.3. Mise en œuvre de tests unitaires et d'intégration pour faciliter la maintenance
 - 17.3.4. Méthodes de remaniement et d'amélioration continue des systèmes comportant des composants d'IA
- 17.4. Design de systèmes à grande échelle
 - 17.4.1. Principes architecturaux pour la conception de systèmes à grande échelle
 - 17.4.2. Décomposition de systèmes complexes en microservices
 - 17.4.3. Mise en œuvre de modèles de conception spécifiques pour les systèmes distribués
 - 17.4.4. Stratégies de gestion de la complexité dans les architectures à grande échelle comportant des composants d'IA
- 17.5. Entreposage de données à grande échelle pour les outils d'IA
 - 17.5.1. Sélection de technologies de stockage de données évolutives
 - 17.5.2. Design de schémas de base de données pour un traitement efficace de grands volumes de données
 - 17.5.3. Stratégies de partitionnement et de réplication dans les environnements de stockage de données de masse
 - 17.5.4. Mise en œuvre de systèmes de gestion des données pour garantir l'intégrité et la disponibilité dans les projets d'IA
- 17.6. Structures des données avec IA
 - 17.6.1. Adaptation des structures de données classiques en vue de leur utilisation dans des algorithmes d'IA
 - 17.6.2. Design et optimisation de structures de données spécifiques pour les modèles d'apprentissage automatique
 - 17.6.3. Intégration de structures de données efficaces dans des systèmes à forte intensité de données
 - 17.6.4. Stratégies de manipulation et de stockage de données en temps réel dans les structures de données d'IA
- 17.7. Algorithmes de programmation pour les produits d'IA
 - 17.7.1. Développement et mise en œuvre d'algorithmes spécifiques pour les applications d'IA
 - 17.7.2. Stratégies de sélection des algorithmes en fonction du type de problème et des exigences du produit
 - 17.7.3. Adaptation d'algorithmes classiques en vue de leur intégration dans des systèmes d'Intelligence Artificielle
 - 17.7.4. Évaluation et comparaison des performances de différents algorithmes dans des contextes de développement de l'IA

- 17.8. Modèles de conception pour le développement avec l'IA
 - 17.8.1. Identification et application de modèles de conception communs dans les projets comportant des composants d'IA
 - 17.8.2. Développement de modèles spécifiques pour l'intégration de modèles et d'algorithmes dans des systèmes existants
 - 17.8.3. Stratégies de mise en œuvre de modèles pour améliorer la réutilisabilité et la maintenabilité dans les projets d'IA
 - 17.8.4. Études de cas et meilleures pratiques dans l'application des modèles de conception dans les architectures d'IA
- 17.9. Mise en œuvre de *Clean Architecture*
 - 17.9.1. Principes et concepts fondamentaux de *Clean Architecture*
 - 17.9.2. Adaptation de *Clean Architecture* aux projets comportant des composants d'IA
 - 17.9.3. Mise en œuvre des couches et des dépendances dans les systèmes à architecture propre
 - 17.9.4. Avantages et défis de la mise en œuvre de *Clean Architecture* dans le développement de logiciels d'IA
- 17.10. Développement de logiciels sécurisés dans les applications web avec l'IA
 - 17.10.1. Principes de sécurité pour le développement de logiciels avec des composants d'IA
 - 17.10.2. Identification et atténuation des vulnérabilités potentielles des modèles et algorithmes d'IA
 - 17.10.3. Mise en œuvre de pratiques de développement sécurisées dans les applications web dotées de fonctionnalités d'Intelligence Artificielle
 - 17.10.4. Stratégies de protection des données sensibles et de prévention des attaques dans les projets d'IA

Module 18. Projets Web avec IA

- 18.1. Préparation de l'Environnement de Travail pour le Développement Web de l'IA
 - 18.1.1. Configuration des environnements de développement web pour les projets faisant appel à l'Intelligence Artificielle
 - 18.1.2. Sélection et préparation des outils essentiels pour le développement web de l'IA
 - 18.1.3. Intégration de bibliothèques et de *frameworks* spécifiques pour les projets web d'Intelligence Artificielle
 - 18.1.4. Mise en œuvre de bonnes pratiques dans la configuration d'environnements de développement collaboratifs
- 18.2. Création des *Workspace* pour les projets d'IA
 - 18.2.1. Conception et organisation efficaces des *workspaces* pour les projets web comportant des éléments d'intelligence artificielle
 - 18.2.2. Utilisation d'outils de gestion de projet et de contrôle de version dans le *workspace*
 - 18.2.3. Stratégies pour une collaboration et une communication efficaces au sein de l'équipe de développement
 - 18.2.4. Adapter du *workspace* aux besoins spécifiques des projets web avec IA
- 18.3. Modèles de Design dans les Produits d'IA
 - 18.3.1. Identification et application de modèles de conception communs dans les interfaces utilisateur avec des éléments d'intelligence artificielle
 - 18.3.2. Développement de modèles spécifiques pour améliorer l'expérience de l'utilisateur dans les projets web avec IA
 - 18.3.3. Intégration de modèles de conception dans l'architecture générale des projets web avec Intelligence Artificielle
 - 18.3.4. Évaluation et sélection des patrons de conception appropriés en fonction du contexte du projet
- 18.4. Développement Frontend avec l'IA
 - 18.4.1. Intégration de modèles d'IA dans la couche de présentation des projets web
 - 18.4.2. Développement d'interfaces utilisateur adaptatives avec des éléments d'Intelligence Artificielle
 - 18.4.3. Mise en œuvre de fonctionnalités de traitement du langage naturel (NLP) dans le Frontend
 - 18.4.4. Stratégies d'optimisation des performances dans le cadre du développement Frontend avec l'IA
- 18.5. Création de la Base de Données
 - 18.5.1. Sélection des technologies de base de données pour les projets web avec Intelligence Artificielle
 - 18.5.2. Conception de schémas de base de données pour le stockage et la gestion des données liées à l'IA
 - 18.5.3. Mise en œuvre de systèmes de stockage efficaces pour les grands volumes de données générés par les modèles d'IA
 - 18.5.4. Stratégies de sécurité et de protection des données sensibles dans les bases de données des projets web d'IA

- 18.6. Développement de Backend avec l'IA
 - 18.6.1. Intégration des services et modèles d'IA dans la logique commerciale du Backend
 - 18.6.2. Développement d'API et de points d'extrémité spécifiques pour la communication entre le Frontend et les composants d'IA
 - 18.6.3. Mise en œuvre de la logique de traitement des données et de prise de décision dans le Backend avec l'IA
 - 18.6.4. Stratégies d'évolutivité et de performance dans le développement du Backend des projets web avec l'IA
- 18.7. Optimiser votre Processus de Déploiement Web
 - 18.7.1. Automatiser le processus de construction et de déploiement des projets web avec l'IA
 - 18.7.2. Mettre en œuvre des pipelines CI/CD adaptés aux applications web avec des composants d'Intelligence Artificielle
 - 18.7.3. Stratégies pour une gestion efficace des versions et des mises à jour dans les déploiements continus
 - 18.7.4. Surveillance et analyse post-déploiement pour l'amélioration continue des processus
- 18.8. L'IA dans l'Informatique en Nuage
 - 18.8.1. Intégration des services d'intelligence artificielle dans les plateformes d'informatique en nuage
 - 18.8.2. Développement de solutions évolutives et distribuées à l'aide de services en nuage fondés sur l'IA
 - 18.8.3. Stratégies de gestion efficace des ressources et des coûts dans les environnements en nuage avec des applications web dotées de capacités d'IA
 - 18.8.4. Évaluation et comparaison des fournisseurs de services en nuage pour les projets web d'Intelligence Artificielle
- 18.9. Création d'un Projet basé sur l'IA pour les Environnements LAMP
 - 18.9.1. Adaptation des projets web basés sur la pile LAMP pour inclure des composants d'Intelligence Artificielle
 - 18.9.2. Intégrer des bibliothèques et des *frameworks* spécifiques à l'IA dans les environnements LAMP
 - 18.9.3. Développement de fonctionnalités d'IA qui complètent l'architecture LAMP traditionnelle
 - 18.9.4. Stratégies d'optimisation et de maintenance dans les projets web avec IA dans les environnements LAMP

- 18.10. Création d'un Projet basé sur l'IA pour les Environnements MEVN
 - 18.10.1. Intégration des technologies et des outils de la pile MEVN avec les composants d'Intelligence Artificielle
 - 18.10.2. Développement d'applications web modernes et évolutives dans les environnements MEVN avec des capacités d'IA
 - 18.10.3. Mise en œuvre de fonctionnalités de traitement des données et d'apprentissage automatique dans les projets MEVN
 - 18.10.4. Stratégies d'amélioration des performances et de la sécurité des applications web avec IA dans les environnements MEVN

Module 19. Applications mobiles avec l'IA

- 19.1. Préparer un Environnement de Travail pour le Développement d'Applications Mobiles avec l'IA
 - 19.1.1. Configuration des environnements de développement mobile pour les projets d'Intelligence Artificielle
 - 19.1.2. Sélection et préparation d'outils spécifiques pour le développement d'applications mobiles d'IA
 - 19.1.3. Intégration de bibliothèques et de *frameworks* d'IA dans les environnements de développement mobile
 - 19.1.4. Configuration d'émulateurs et d'appareils réels pour tester les applications mobiles avec des composants d'intelligence artificielle
- 19.2. Création d'un *Workspace* avec GitHub Copilot
 - 19.2.1. Intégration de GitHub Copilot dans les environnements de développement mobile
 - 19.2.2. Utilisation efficace de GitHub Copilot pour la génération de code dans les projets d'IA
 - 19.2.3. Stratégies de collaboration entre développeurs lors de l'utilisation de GitHub Copilot dans le *workspace*
 - 19.2.4. Bonnes pratiques et limites de l'utilisation de GitHub Copilot dans le développement d'applications mobiles d'IA
- 19.3. Configuration de Firebase
 - 19.3.1. Configuration initiale d'un projet Firebase pour le développement mobile
 - 19.3.2. Intégration de Firebase dans les applications mobiles avec des fonctionnalités d'Intelligence Artificielle
 - 19.3.3. Utilisation des services Firebase tels que la base de données, l'authentification et les notifications dans les projets d'IA
 - 19.3.4. Stratégies de gestion des données et des événements en temps réel dans les applications mobiles utilisant Firebase

- 19.4. Concepts de Clean Architecture, DataSources, Repositories
 - 19.4.1. Principes fondamentaux de Clean Architecture dans le développement mobile avec AI
 - 19.4.2. Mise en œuvre des couches DataSources et Repositories dans les architectures propres
 - 19.4.3. Conception et structuration des composants dans les projets mobiles en mettant l'accent sur l'architecture propre
 - 19.4.4. Avantages et défis de la mise en œuvre de *Clean Architecture* dans les applications mobiles avec IA
 - 19.5. Création d'un Écran d'Authentification
 - 19.5.1. Conception et développement d'interfaces utilisateurs pour les écrans d'authentification dans les applications mobiles avec IA
 - 19.5.2. Intégration des services d'authentification Firebase dans l'écran de connexion
 - 19.5.3. Utilisation de techniques de sécurité et de protection des données dans l'écran d'authentification
 - 19.5.4. Personnalisation de l'expérience utilisateur dans l'écran d'authentification
 - 19.6. Création de *Dashboard* et navigation
 - 19.6.1. Conception et développement de *Dashboards* avec des éléments d'Intelligence Artificielle
 - 19.6.2. Mise en œuvre de systèmes de navigation efficaces dans les applications mobiles grâce à l'IA
 - 19.6.3. Intégration de fonctionnalités d'IA dans le *Dashboard* pour améliorer l'expérience de l'utilisateur
 - 19.7. Création d'un Écran de Listage
 - 19.7.1. Développement d'interfaces utilisateur pour les écrans d'inscription dans les applications mobiles basées sur l'IA
 - 19.7.2. Intégration d'algorithmes de recommandation et de filtrage dans l'écran d'inscription
 - 19.7.3. Utilisation de modèles de conception pour une présentation efficace des données d'inscription
 - 19.7.4. Stratégies pour un chargement efficace en temps réel des données dans l'écran d'inscription
 - 19.8. Création d'un Écran de Détail
 - 19.8.1. Conception et développement d'interfaces utilisateur détaillées pour la présentation d'informations spécifiques
 - 19.8.2. Intégration de fonctionnalités d'IA pour enrichir l'écran de détail
 - 19.8.3. Mise en œuvre d'interactions et d'animations sur l'écran de détail
 - 19.8.4. Stratégies d'optimisation des performances dans l'affichage et le chargement des détails des applications mobiles basées sur l'IA
 - 19.9. Création de l'Écran des *Settings*
 - 19.9.1. Développement d'interfaces utilisateur pour la configuration et les paramètres dans les applications mobiles d'IA
 - 19.9.2. Intégration de paramètres personnalisés liés aux composants de l'Intelligence Artificielle
 - 19.9.3. Mise en œuvre d'options de personnalisation et de préférences dans l'écran de configuration
 - 19.9.4. Stratégies de convivialité et de clarté dans la présentation des options de l'écran de *settings*
 - 19.10. Création d'Icônes, *Splash* et de Ressources Graphiques pour votre Application AI
 - 19.10.1. Concevoir et créer des icônes attrayantes pour représenter l'application mobile IA
 - 19.10.2. Développer des écrans d'accueil (*splash*) avec des visuels percutants
 - 19.10.3. Sélection et adaptation de ressources graphiques qui améliorent l'esthétique de l'application mobile
 - 19.10.4. Stratégies de cohérence et de branding visuel dans les éléments graphiques de l'application avec l'IA
- Module 20. L'IA pour les QA Testing**
- 20.1. Cycle de Vie des *Testing*
 - 20.1.1. Description et compréhension du cycle de vie des *testing* dans le développement de logiciels
 - 20.1.2. Phases du cycle de *testing* et leur importance pour l'assurance qualité
 - 20.1.3. Intégration de l'intelligence artificielle dans les différentes étapes du cycle de vie des *testing*
 - 20.1.4. Stratégies d'amélioration continue du cycle de vie des *testing* par l'utilisation de l'IA
 - 20.2. Test Cases et Détection des Bugs
 - 20.2.1. Conception et rédaction efficaces des cas de test dans le contexte des QA *Testing*
 - 20.2.2. Identification des bugs et des erreurs pendant l'exécution des cas de test
 - 20.2.3. Application de techniques de détection précoce des bugs à l'aide de l'analyse statique
 - 20.2.4. Utilisation d'outils d'intelligence artificielle pour l'identification automatique des bugs dans les cas de test

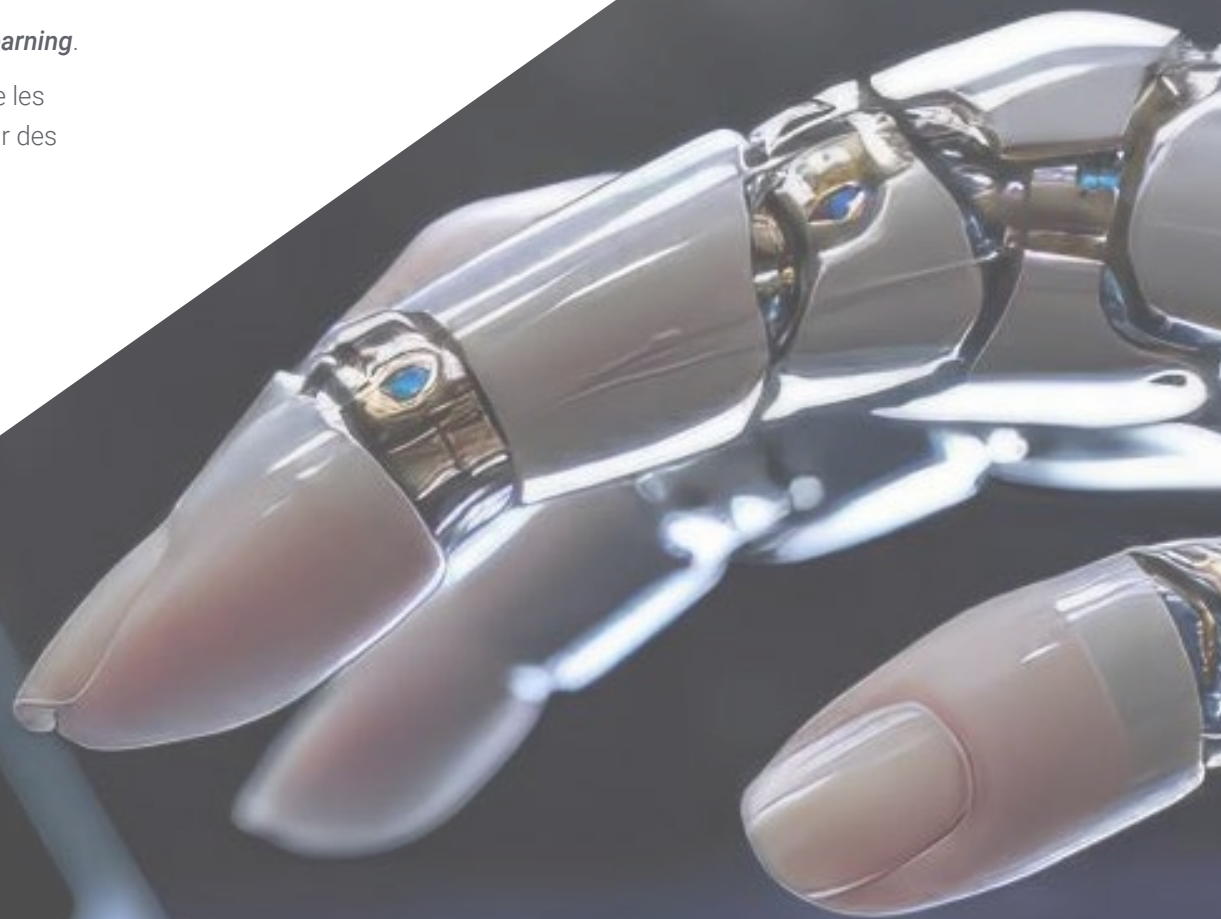
- 20.3. Types de *Testing*
 - 20.3.1. Exploration des différents types de *testing* dans le domaine de l'AQ
 - 20.3.2. Tests unitaires, d'intégration, fonctionnels et d'acceptation: caractéristiques et applications
 - 20.3.3. Stratégies pour la sélection et la combinaison appropriée des types de *testing* dans les projets d'IA
 - 20.3.4. Adaptation des types de *testing* conventionnels aux projets comportant des éléments d'intelligence artificielle
- 20.4. Création d'un Plan de Test
 - 20.4.1. Dessin et structuration d'un plan de test complet
 - 20.4.2. Identification des exigences et des scénarios d'essai dans les projets d'IA
 - 20.4.3. Stratégies de planification des tests manuels et automatisés
 - 20.4.4. Évaluation et ajustement continu du plan de test en fonction de l'évolution du projet
- 20.5. Détection et Signalement des *Bogues* par l' IA
 - 20.5.1. Mise en œuvre de techniques de détection automatique des bogues à l'aide d'algorithmes d'apprentissage automatique
 - 20.5.2. Utilisation d'outils d'intelligence artificielle pour l'analyse dynamique du code à la recherche d'erreurs potentielles
 - 20.5.3. Stratégies de génération automatique de rapports détaillés sur les bogues détectés par l'IA
 - 20.5.4. Collaboration efficace entre les équipes de développement et de QA dans la gestion des bogues identifiés par l'IA
- 20.6. Création de Tests Automatisés avec l'IA
 - 20.6.1. Développement de scripts de tests automatisés pour les projets comportant des composants d'intelligence artificielle
 - 20.6.2. Intégration d'outils d'automatisation des tests basés sur l'IA
 - 20.6.3. Utilisation d'algorithmes de *machine learning* pour la génération dynamique de cas de test automatisés
 - 20.6.4. Stratégies pour l'exécution et la maintenance efficaces des cas de test automatisés dans les projets d'IA
- 20.7. *API Testing*
 - 20.7.1. Concepts fondamentaux des *testing* d'API et leur importance en matière de QA
 - 20.7.2. Développement de tests pour la vérification des API dans des environnements avec des composants d'Intelligence Artificielle
 - 20.7.3. Stratégies de validation des données et des résultats dans les *testing* d'API avec l'IA
 - 20.7.4. Utilisation d'outils spécifiques pour les *testing* d'API dans les projets d'Intelligence Artificielle
- 20.8. Outils d' IA pour les *Web Testing*
 - 20.8.1. Exploration des outils d'intelligence artificielle pour l'automatisation des tests dans les environnements web
 - 20.8.2. Intégration des technologies de reconnaissance d'éléments et d'analyse visuelle dans les *web testing*
 - 20.8.3. Stratégies de détection automatique des changements et des problèmes de performance dans les applications web à l'aide de l'IA
 - 20.8.4. Évaluation d'outils spécifiques pour améliorer l'efficacité des *web testing* grâce à l'IA
- 20.9. *Mobile Testing* Utilisant l'IA
 - 20.9.1. Développement de stratégies de *testing* pour les applications mobiles avec des composants d'intelligence artificielle
 - 20.9.2. Intégration d'outils de *testing* spécifiques pour les plates-formes mobiles fondées sur l'IA
 - 20.9.3. Utilisation d'algorithmes d'apprentissage automatique pour la détection des problèmes de performance dans les applications mobiles
 - 20.9.4. Stratégies de validation d'interfaces et de fonctions spécifiques d'applications mobiles à l'aide de l'IA
- 20.10. Outils de QA avec IA
 - 20.10.1. Exploration des outils et plates-formes d'assurance de la qualité intégrant des capacités d'Intelligence Artificielle
 - 20.10.2. Évaluation d'outils pour une gestion et une exécution efficaces des tests dans les projets d'IA
 - 20.10.3. Utilisation d'algorithmes de *machine learning* pour la génération et l'optimisation des cas de test
 - 20.10.4. Stratégies pour la sélection et l'adoption efficace d'outils d'AQ basés sur l' IA

06

Méthodologie

Ce programme de formation offre une manière différente d'apprendre. Notre méthodologie est développée à travers un mode d'apprentissage cyclique: ***le Relearning***.

Ce système d'enseignement est utilisé, par exemple, dans les écoles de médecine les plus prestigieuses du monde et a été considéré comme l'un des plus efficaces par des publications de premier plan telles que le ***New England Journal of Medicine***.





“

Découvrez Relearning, un système qui renonce à l'apprentissage linéaire conventionnel pour vous emmener à travers des systèmes d'enseignement cycliques: une façon d'apprendre qui s'est avérée extrêmement efficace, en particulier dans les matières qui exigent la mémorisation”

Étude de Cas pour mettre en contexte tout le contenu

Notre programme offre une méthode révolutionnaire de développement des compétences et des connaissances. Notre objectif est de renforcer les compétences dans un contexte changeant, compétitif et hautement exigeant.

“

Avec TECH, vous pouvez expérimenter une manière d'apprendre qui ébranle les fondations des universités traditionnelles du monde entier”



Vous bénéficierez d'un système d'apprentissage basé sur la répétition, avec un enseignement naturel et progressif sur l'ensemble du cursus.



L'étudiant apprendra, par des activités collaboratives et des cas réels, à résoudre des situations complexes dans des environnements commerciaux réels.

Une méthode d'apprentissage innovante et différente

Cette formation TECH est un programme d'enseignement intensif, créé de toutes pièces, qui propose les défis et les décisions les plus exigeants dans ce domaine, tant au niveau national qu'international. Grâce à cette méthodologie, l'épanouissement personnel et professionnel est stimulé, faisant ainsi un pas décisif vers la réussite. La méthode des cas, technique qui constitue la base de ce contenu, permet de suivre la réalité économique, sociale et professionnelle la plus actuelle.

“ Notre programme vous prépare à relever de nouveaux défis dans des environnements incertains et à réussir votre carrière ”

La méthode des cas est le système d'apprentissage le plus largement utilisé dans les meilleures écoles d'informatique du monde depuis qu'elles existent. Développée en 1912 pour que les étudiants en Droit n'apprennent pas seulement le droit sur la base d'un contenu théorique, la méthode des cas consiste à leur présenter des situations réelles complexes afin qu'ils prennent des décisions éclairées et des jugements de valeur sur la manière de les résoudre. En 1924, elle a été établie comme méthode d'enseignement standard à Harvard.

Dans une situation donnée, que doit faire un professionnel? C'est la question à laquelle nous sommes confrontés dans la méthode des cas, une méthode d'apprentissage orientée vers l'action. Tout au long du programme, les étudiants seront confrontés à de multiples cas réels. Ils devront intégrer toutes leurs connaissances, faire des recherches, argumenter et défendre leurs idées et leurs décisions.

Relearning Methodology

TECH combine efficacement la méthodologie des Études de Cas avec un système d'apprentissage 100% en ligne basé sur la répétition, qui associe différents éléments didactiques dans chaque leçon.

Nous enrichissons l'Étude de Cas avec la meilleure méthode d'enseignement 100% en ligne: le Relearning.

En 2019, nous avons obtenu les meilleurs résultats d'apprentissage de toutes les universités en ligne du monde.

À TECH, vous apprendrez avec une méthodologie de pointe conçue pour former les managers du futur. Cette méthode, à la pointe de la pédagogie mondiale, est appelée Relearning.

Notre université est la seule université autorisée à utiliser cette méthode qui a fait ses preuves. En 2019, nous avons réussi à améliorer les niveaux de satisfaction globale de nos étudiants (qualité de l'enseignement, qualité des supports, structure des cours, objectifs...) par rapport aux indicateurs de la meilleure université en ligne.



Dans notre programme, l'apprentissage n'est pas un processus linéaire, mais se déroule en spirale (apprendre, désapprendre, oublier et réapprendre). Par conséquent, chacun de ces éléments est combiné de manière concentrique. Cette méthodologie a permis de former plus de 650.000 diplômés universitaires avec un succès sans précédent dans des domaines aussi divers que la biochimie, la génétique, la chirurgie, le droit international, les compétences en gestion, les sciences du sport, la philosophie, le droit, l'ingénierie, le journalisme, l'histoire, les marchés financiers et les instruments. Tout cela dans un environnement très exigeant, avec un corps étudiant universitaire au profil socio-économique élevé et dont l'âge moyen est de 43,5 ans.

Le Relearning vous permettra d'apprendre avec moins d'efforts et plus de performance, en vous impliquant davantage dans votre formation, en développant un esprit critique, en défendant des arguments et en contrastant les opinions: une équation directe vers le succès.

À partir des dernières preuves scientifiques dans le domaine des neurosciences, non seulement nous savons comment organiser les informations, les idées, les images et les souvenirs, mais nous savons aussi que le lieu et le contexte dans lesquels nous avons appris quelque chose sont fondamentaux pour notre capacité à nous en souvenir et à le stocker dans l'hippocampe, pour le conserver dans notre mémoire à long terme.

De cette manière, et dans ce que l'on appelle Neurocognitive context-dependent e-learning, les différents éléments de notre programme sont reliés au contexte dans lequel le participant développe sa pratique professionnelle.



Ce programme offre le support matériel pédagogique, soigneusement préparé pour les professionnels:



Support d'étude

Tous les contenus didactiques sont créés par les spécialistes qui enseigneront le cours, spécifiquement pour le cours, afin que le développement didactique soit vraiment spécifique et concret.

Ces contenus sont ensuite appliqués au format audiovisuel, pour créer la méthode de travail TECH en ligne. Tout cela, avec les dernières techniques qui offrent des pièces de haute qualité dans chacun des matériaux qui sont mis à la disposition de l'étudiant.



Cours magistraux

Il existe des preuves scientifiques de l'utilité de l'observation par un tiers expert.

La méthode "Learning from an Expert" renforce les connaissances et la mémoire, et donne confiance dans les futures décisions difficiles.



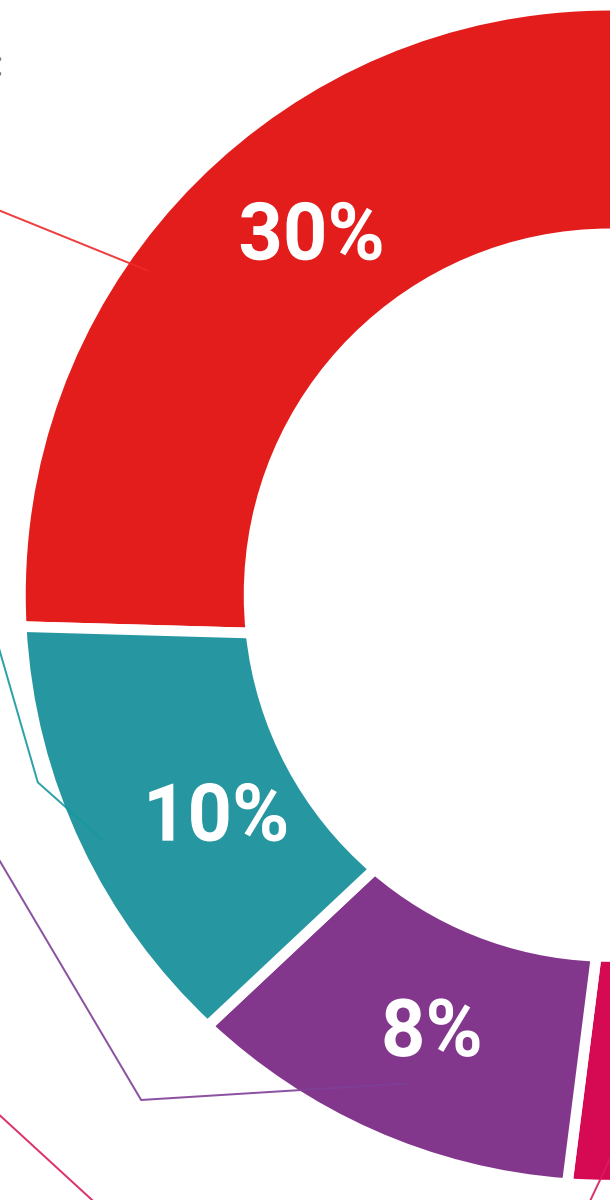
Pratiques en compétences et aptitudes

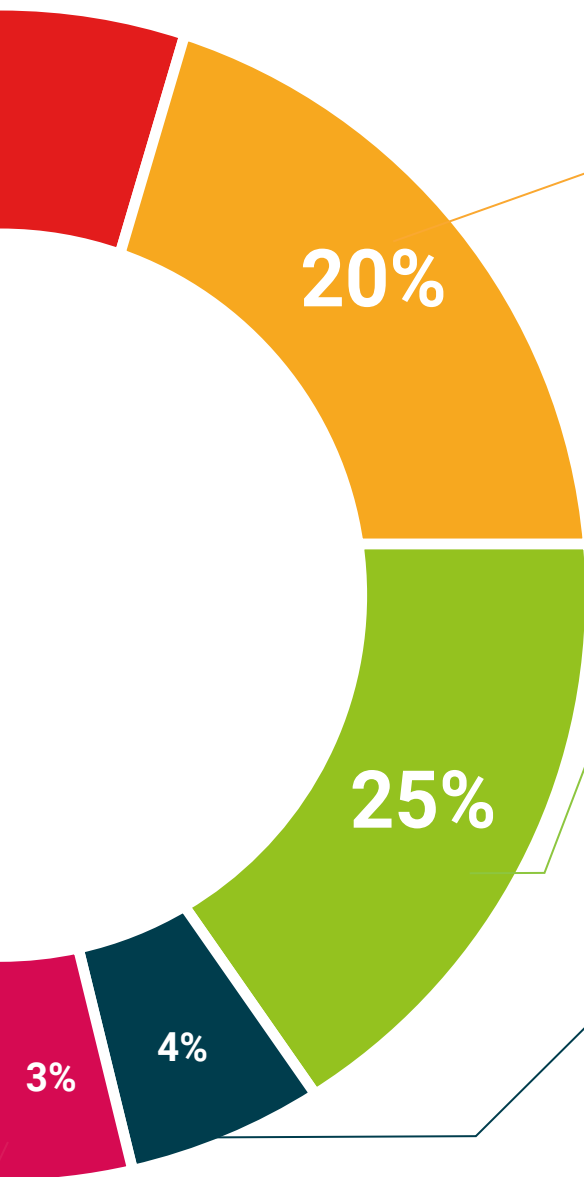
Les étudiants réaliseront des activités visant à développer des compétences et des aptitudes spécifiques dans chaque domaine. Des activités pratiques et dynamiques pour acquérir et développer les compétences et aptitudes qu'un spécialiste doit développer dans le cadre de la mondialisation dans laquelle nous vivons.



Lectures complémentaires

Articles récents, documents de consensus et directives internationales, entre autres. Dans la bibliothèque virtuelle de TECH, l'étudiant aura accès à tout ce dont il a besoin pour compléter sa formation.





Case studies

Ils réaliseront une sélection des meilleures études de cas choisies spécifiquement pour ce diplôme. Des cas présentés, analysés et tutorés par les meilleurs spécialistes de la scène internationale.



Résumés interactifs

L'équipe TECH présente les contenus de manière attrayante et dynamique dans des pilules multimédia comprenant des audios, des vidéos, des images, des diagrammes et des cartes conceptuelles afin de renforcer les connaissances. Ce système éducatif unique pour la présentation de contenu multimédia a été récompensé par Microsoft en tant que "European Success Story".



Testing & Retesting

Les connaissances de l'étudiant sont périodiquement évaluées et réévaluées tout au long du programme, par le biais d'activités et d'exercices d'évaluation et d'auto-évaluation, afin que l'étudiant puisse vérifier comment il atteint ses objectifs.



07 Diplôme

Le Mastère Spécialisé en Intelligence Artificielle dans la Programmation garantit, outre la formation la plus rigoureuse et la plus actualisée, l'accès à un diplôme de Mastère Spécialisé délivré par TECH Université Technologique.



“

*Terminez ce programme avec succès
et recevez votre diplôme sans avoir
à vous soucier des déplacements ou
des formalités administratives”*

Ce **Mastère Spécialisé en Intelligence Artificielle dans la Programmation** contient le programme le plus complet et le plus actualisé du marché

Après avoir passé l'évaluation, l'étudiant recevra par courrier* avec accusé de réception son diplôme de **Mastère Spécialisé** délivrée par **TECH Université Technologique**.

Le diplôme délivré par **TECH Université Technologique** indiquera la note obtenue lors du **Mastère Spécialisé**, et répond aux exigences communément demandées par les bourses d'emploi, les concours et les commissions d'évaluation des carrières professionnelles.

Diplôme: **Mastère Spécialisé en Intelligence Artificielle dans la Programmation**

Heures Officielles **2.250 h.**



*Si l'étudiant souhaite que son diplôme version papier possède l'Apostille de La Haye, TECH EDUCATION fera les démarches nécessaires pour son obtention moyennant un coût supplémentaire.

future
santé confiance personnes
éducation information tuteurs
garantie accréditation enseignement
institutions technologie apprentissage
communauté engagement
service personnalisé innovation
connaissance présent qualité
en ligne formation
développement institutions
classe virtuelle langues



Mastère Spécialisé Intelligence Artificielle dans la Programmation

- » Modalité: en ligne
- » Durée: 12 mois
- » Diplôme: TECH Université Technologique
- » Temps estimé: 16 heures/semaine
- » Horaire: à votre rythme
- » Examens: en ligne

Mastère Spécialisé

Intelligence Artificielle dans la Programmation