GHINDA CRISTINA-MARIA

VEILLE TECHNOLOGIQUE

Pour Le Titre

EXPERT EN INFORMATIQUE ET SYSTEME D'INFORMATION

3W Academy

2023

LE CLUSTERING EN APPRENTISSAGE AUTOMATIQUE (MACHINE LEARNING)

3W Academy

2023

Table des matières

Le Clustering En Apprentissage Automatique (Machine Learning)

Introduction

Le clustering (ou regroupement en français) est une technique d'analyse de données qui vise à regrouper un ensemble d'objets similaires en sous-groupes, appelés clusters. L'objectif principal du clustering est de découvrir des structures intrinsèques dans les données, en regroupant des éléments similaires et en les séparant des éléments dissimilaires.

Le clustering est largement utilisé dans de nombreux domaines, tels que l'apprentissage automatique (machine learning), l'exploration de données (data mining), la bioinformatique, la reconnaissance de formes, la segmentation d'images, la recommandation de produits, etc.

Il existe plusieurs algorithmes de clustering, chacun avec ses propres caractéristiques et hypothèses. Chaque algorithme a ses propres avantages et inconvénients en fonction du type de données et du problème spécifique à résoudre. Le choix de l'algorithme de clustering dépendra donc des caractéristiques des données et des objectifs de l'analyse.

Dans le cadre de ce rapport, après avoir présenté en grandes lignes l'intelligence artificielle et son sous-domaine le Machine Learning, avec ses différents modèles d'apprentissage (supervisé, non-supervisé, etc.), nous allons nous pencher plus en détail sur le clustering, qui est un type d'algorithme de Machine Learning non-supervisé. Nous examinerons plusieurs types de clustering, tels que le clustering de partitionnement, les algorithmes de clustering basés sur la hiérarchie et les algorithmes de clustering basés sur la distribution et également quelques algorithmes spécifiques tels que le K-means or l'algorithme de clustering DBSCAN.

L'intelligence artificielle

L'intelligence artificielle est une branche de l'informatique qui permet aux systèmes d'apprendre et d'exécuter des tâches normalement associées à l'intelligence humaine, telles que la reconnaissance vocale, la prise de décisions ou la perception visuelle. Elle représente tout outil utilisé par une machine afin de « reproduire des comportements liés aux humains, tels que le raisonnement, la planification et la créativité ». Cette technologie, améliore les performances et la productivité en automatisant des processus ou des tâches qui nécessitaient auparavant des ressources humaines.

Evolution

Dans les années 50, la réalité de l'intelligence artificielle était au mieux hors de portée : les ordinateurs ne pouvaient pas stocker ou exécuter des informations, et les coûts qui y étaient relatifs étaient astronomiques. C'est alors que le mathématicien Alan Turing a posé une question simple, mais révolutionnaire : « les machines peuvent-elles penser ? ». La réponse, un grand oui, a changé le cours de l'histoire.

Entre les années 50 et 70, l'industrie informatique a pu s'appuyer sur des ordinateurs plus rapides, plus accessibles et moins coûteux. Des études mettaient en évidence le fait qu'en seulement trois à cinq ans, les machines auraient bientôt la même intelligence qu'un être humain. Pour cela, des avancées majeures en matière de capacité de stockage et de puissance de calcul étaient nécessaires.

Dans les années 80, deux techniques importantes ont été développées. La première, le deep learning, ou « apprentissage en profondeur », a permis aux ordinateurs d'apprendre par l'expérience. Le second, l'expert system, ou « système expert », imite la capacité de l'homme à prendre des décisions. Les ordinateurs ont commencé à utiliser un raisonnement basé sur des « règles » en recourant principalement à une structure « si-alors » mise en œuvre pour répondre à des questions.

En 1997, Dragon Systems a développé et implémenté une solution de reconnaissance vocale sous Windows. Les années 2000 ont été synonymes de débit et d'options de stockage, comme le cloud, catapultant l'utilisation des ordinateurs auprès du grand public et contribuant à placer l'IA sous les feux de la rampe. Aujourd'hui, l'IA est sur une voie express, en raison de trois améliorations sectorielles majeures.

- **Les unités de traitement graphique (GPU) :** la demande dans le monde de la vidéo et des jeux a entraîné le développement de GPU améliorées et moins coûteuses, un élément nécessaire pour la construction de solutions d'IA.
- **Le Big Data :** les algorithmes utilisés par l'IA sont essentiellement « formés » par la grande quantité d'informations présentes dans le Big Data. Ces algorithmes aident ensuite l'IA à traiter ces informations à un rythme incroyable et à rendre les données accessibles et plus utilisables.
- **Les algorithmes :** les algorithmes permettent l'automatisation de tâches que l'on croyait autrefois uniquement possibles grâce à l'intelligence humaine. Les algorithmes s'améliorent continuellement et deviennent plus complexes grâce à l'utilisation de couches avec des variables cachées, qui trient et optimisent les résultats.

Aujourd'hui, l'intelligence artificielle, ou intelligence machine, et les machines dotées d'IA peuvent accomplir des nombreux actes, par exemple :

- Reconnaissance vocale
- Apprentissage
- Planification
- Résolution de problèmes
- Raisonnement
- Perception
- Possibilité de manipuler et de déplacer des objets

Machine Learning

Machine Learning (**ML**), sous-domaine de l'intelligence artificielle (**IA**), a pour objectif d'analyser et d'interpréter des modèles et des structures de données afin de permettre l'apprentissage, le raisonnement et la prise de décision sans interaction humaine. L'intelligence artificielle est la mère du Machine Learning et de tous les sous-ensembles qui le composent.

Le machine learning existe car il nous permet d'exploiter des données à un niveau qu'aucun humain ne pourrait jamais atteindre. Il répond à des besoins réels dans de nombreux domaines et permet des avancées significatives dans de nombreux domaines, tels que la santé, la finance, la sécurité, l'automobile, la reconnaissance vocale et d'image, et bien d'autres.

Le Machine Learning se décline sous différents types de modèles, qui emploient chacun des techniques algorithmiques différentes.

Selon la nature des données et le résultat souhaité, un des quatre modèles d'apprentissage suivants peut être utilisé :

- **Supervisé** (qui consiste en **à faire apprendre un ordinateur à partir de données étiquetées** ou labellisées. Cela signifie que la machine travaille avec un jeu de données qui ont déjà été identifiées et catégorisées. En conséquence, l'ordinateur connaît en amont les réponses qu'il devrait fournir. Le but du Machine Learning dans ce cas est d'amener la machine à prédire les nouvelles données non étiquetées qui lui seront soumises ensuite.)
- **Non-supervisé** (se manifeste par **l'exécution des tâches par un algorithme sans une aide préalable**. Les données sont adressées à la machine sans les résultats attendus. Les données fournies sont exemptes de réponses. C'est à l'algorithme de classer et d'analyser les données pour aboutir aux résultats modérés.)
- **Semi-supervisé** (qui prend en entrée certaines données annotées et d'autres non. Ce sont des méthodes très intéressantes qui tirent parti des deux mondes (supervised et unsupervised), mais bien sûr apportent leur lot de difficultés.)
- **Reinforcement learning** (qui se base sur un cycle d'expérience / récompense et améliore les performances à chaque itération. Une analogie souvent citée est celle du cycle de dopamine : une "bonne" expérience augmente la dopamine et donc augmente la probabilité que l'agent répète l'expérience)

Dans chacun de ces modèles, une ou plusieurs techniques algorithmiques peuvent être appliquées. Tout dépend des ensembles des données qui seront utilisés et de l'objectif visé au niveau des résultats. Par nature, les algorithmes de Machine Learning sont conçus pour classifier des éléments, repérer des patterns, prévoir des résultats et prendre des décisions éclairées. Les algorithmes peuvent être mis en œuvre individuellement ou en groupe dans le but d'atteindre la plus grande précision possible lorsque les données utilisées sont complexes et imprévisibles.

Non-supervised Learning (L'apprentissage non-supervisé)

Le Machine Learning non-supervisé est une approche où les données d'entrée ne sont pas étiquetées, c'est-à-dire qu'elles n'ont pas de catégories ou de classes prédéfinies. Les algorithmes utilisés pour ce type de Machine Learning sont conçus pour trouver des structures ou des patterns dans les données sans avoir recours à des étiquettes de classification.

Contrairement au Machine Learning supervisé, où les données sont étiquetées et utilisées pour entraîner le modèle, le Machine Learning non-supervisé ne nécessite pas que les données soient étiquetées au préalable. Cependant, les données non-étiquetées peuvent être moins complexes que dans le cas supervisé.

Dans le Machine Learning non-supervisé, toutes les données sont traitées comme des variables aléatoires, ce qui signifie que les algorithmes cherchent à comprendre la distribution statistique des données pour en extraire des informations utiles. Cela permet de trouver des structures sous-jacentes dans les données et de les utiliser pour la segmentation, la classification ou la visualisation.

Pour le Machine Learning non-supervisé, **les données d'entrées sont inconnues et moins complexes** que lorsqu'il s'agit du type supervisé. Toutes les données sont traitées comme des variables aléatoires. Puisque les données ne sont pas étiquetées, il n'est pas possible de calculer des scores de réussite.

Les principaux types d'algorithmes de Machine Learning non-supervisé sont les suivants :

1. Les algorithmes de clustering : ces algorithmes cherchent à regrouper les données en fonction de leur similarité ou de leur dissimilarité. Les algorithmes de clustering les plus couramment utilisés sont K-Means, DBSCAN et hierarchial clustering.
2. Les algorithmes de réduction de dimensionnalité : ces algorithmes cherchent à réduire le nombre de dimensions d'un ensemble de données, tout en conservant autant d'informations que possible. Les algorithmes de réduction de dimensionnalité les plus couramment utilisés

sont PCA (Principal Component Analysis) et t-SNE (t-distributed stochastic neighbor embedding).

3. Les algorithmes de détection d'anomalies : ces algorithmes cherchent à identifier les données qui sont très différentes du reste de l'ensemble de données. Les algorithmes de détection d'anomalies les plus couramment utilisés sont Isolation Forest et LOF (Local Outlier Factor).
4. Les algorithmes de regroupement de texte : ces algorithmes sont utilisés pour regrouper des documents en fonction de leur similitude en termes de contenu. Les algorithmes de regroupement de texte les plus couramment utilisés sont Latent Dirichlet Allocation (LDA) et k-means.
5. Les algorithmes de factorisation de matrice : ces algorithmes cherchent à réduire le nombre de dimensions d'une matrice en trouvant des sous-structures significatives. Les algorithmes de factorisation de matrice les plus couramment utilisés sont NMF (Non-negative Matrix Factorization) et SVD (Singular Value Decomposition).

Dans le cadre de ce travail, nous allons discuter du clustering comme technique d'analyse de données dans le cadre du ML non-supervisé.

Le clustering de données

La difficulté de classifier des données de manière efficace et automatique a conduit à l'apparition du clustering. Le clustering de données (ou regroupement de données) est une technique d'analyse de données qui permet de découvrir des structures cachées dans les données en regroupant les observations similaires dans des ensembles appelés "clusters". Cette méthode est utilisée dans de nombreux domaines, tels que la biologie, la finance, la psychologie, l'informatique, etc. pour résoudre des problèmes de segmentation, de classification, de reconnaissance de formes et d'analyse de données.

Avant l'apparition du clustering, la plupart des méthodes de classification étaient supervisées, c'est-à-dire qu'elles nécessitaient des étiquettes de classe pour chaque exemple de données. Cela signifie que pour classifier les données, il fallait d'abord collecter et étiqueter un grand nombre d'exemples de données.

Le clustering permet aussi de découvrir des groupes de données qui présentent des similarités entre eux, mais qui peuvent être différents des autres groupes et cette méthode est très utile pour explorer les données, pour détecter des anomalies ou pour segmenter des populations. Elle peut également être utilisée pour réduire la complexité des données en les résumant sous forme de clusters et peut aider les scientifiques des données à mieux comprendre les caractéristiques et les relations entre les données.

Les clusters peuvent être créés à partir de différentes mesures de similarité, telles que la distance euclidienne, la corrélation, la similarité cosinus, etc.

Le clustering de données existe pour aider à explorer, comprendre et résumer de grandes quantités de données en identifiant des groupes homogènes de données similaires. Cela peut faciliter l'analyse de données, la prise de décision et la résolution de problèmes dans de nombreux domaines.

Le clustering de données peut potentiellement améliorer la précision des algorithmes, mais cela dépend du contexte et de la manière dont le clustering est utilisé.

Dans certains cas, le clustering peut être utilisé pour pré-traiter les données avant d'appliquer un algorithme de classification ou de régression supervisé. En regroupant les données similaires en clusters, il peut être plus facile pour un algorithme supervisé de trouver des relations entre les données et les étiquettes associées, ce qui peut améliorer la précision du modèle final.

Cependant, dans d'autres cas, le clustering peut être utilisé en tant qu'algorithme autonome pour découvrir des structures dans les données. Dans ce cas, le clustering ne vise pas directement à améliorer la précision d'un algorithme supervisé, mais plutôt à fournir des informations utiles pour l'analyse des données.

Evolution

Le clustering est une technique d'analyse de données qui est apparue dans les années 1950, et les premiers algorithmes de clustering ont été développés pour des applications en biologie et en statistique. L'origine de cette méthode remonte à la recherche opérationnelle et à la classification numérique, où des méthodes de partitionnement ont été utilisées pour diviser des ensembles de données en groupes homogènes.

Le premier algorithme de clustering connu est l'algorithme de classification hiérarchique, proposé par le statisticien américain Joseph Berkson en 1949. L'algorithme de classification hiérarchique est une méthode de clustering qui permet de regrouper les observations en clusters de manière hiérarchique, à partir d'une matrice de distances ou de similarités entre les observations. Cette méthode est souvent utilisée pour représenter graphiquement la structure des données sous forme de dendrogrammes (diagramme fréquemment utilisé pour illustrer l'arrangement de groupes générés par un regroupement hiérarchique ou hiérarchisant). Depuis lors, de nombreux autres algorithmes de clustering ont été développés, chacun avec ses avantages et ses limites en fonction du type de données et des objectifs de la tâche de clustering.

Une autre méthode de clustering développée pour des applications en biologie est la méthode de Ward a été proposée en 1963 pour la classification hiérarchique des données en biologie. La méthode des k-moyennes (ou k-means) a également été proposée à cette époque pour la segmentation de données en groupes distincts.

Depuis lors, le clustering est devenu une technique courante d'analyse de données utilisée dans de nombreux domaines, y compris le marketing, la finance, l'informatique, la géographie, etc. Les algorithmes de clustering ont également été développés pour gérer des données de grande dimension et de grande taille, en utilisant des techniques de calcul distribué et de parallélisation.

Aujourd'hui, le clustering continue d'évoluer avec l'avènement de nouveaux algorithmes et de nouvelles applications, notamment dans le domaine de l'apprentissage automatique et de l'analyse de données non structurées.

Types de clustering

Il existe plusieurs approches de clustering et le choix du type de clustering dépendra des données à traiter et de l'objectif du clustering.

Chaque approche est mieux adaptée à une distribution de données particulière et chaque type de clustering a ses avantages et ses inconvénients.

Nous allons voir par la suite les plus courants types de clustering et des exemples d'algorithmes les plus couramment utilisés.

1. Clustering de partitionnement

Le clustering de partitionnement, ou "Partitioning Clustering" en anglais, est une méthode de clustering qui consiste à diviser un ensemble de données en plusieurs partitions, de telle sorte que les points de données dans chaque partition soient les plus similaires possible entre eux, et les plus différents possible des points de données des autres partitions.

Le clustering de partitionnement est basé sur l'idée que chaque point de données appartient à un seul groupe ou cluster, contrairement aux algorithmes de clustering flous où chaque point de données peut appartenir à plusieurs clusters avec différents degrés d'appartenance.

L'un des algorithmes de clustering de partitionnement les plus populaires est l'algorithme K-means. Cet algorithme commence par initialiser K clusters de manière aléatoire. Ensuite, il attribue chaque point de données au cluster le plus proche en termes de distance euclidienne, et calcule la moyenne de chaque cluster pour obtenir le centre de gravité. Le processus est répété jusqu'à ce que la convergence soit atteinte, c'est-à-dire que les centres des clusters ne bougent plus ou que le nombre maximal d'itérations soit atteint.

Les avantages de l'algorithme K-means et des algorithmes de clustering de partitionnement en général sont qu'ils sont faciles à implémenter, efficaces pour des ensembles de données de taille moyenne, et qu'ils peuvent fournir des résultats interprétables et visuels en termes de partitions. Cependant, ils ont également des inconvénients, tels que leur sensibilité aux valeurs aberrantes, leur dépendance à la valeur initiale de K, et leur capacité limitée à gérer des ensembles de données complexes ou avec des formes irrégulières

2. Les algorithmes de clustering basé sur la hiérarchie

Le clustering basé sur la hiérarchie, ou "Hierarchical Clustering" en anglais, est une méthode de clustering qui organise les données en une structure arborescente ou dendrogramme. Cette méthode peut être divisée en deux types : le clustering hiérarchique agglomératif et le clustering hiérarchique diviseur.

Dans le clustering hiérarchique agglomératif, chaque point de données est considéré comme un cluster à part entière, et les clusters sont fusionnés à chaque étape en fonction de leur similarité. Au début, chaque point de données est considéré comme un cluster. Ensuite, à chaque étape, les deux clusters les plus similaires sont fusionnés pour former un cluster plus grand, jusqu'à ce qu'un seul cluster contenant tous les points de données soit formé.

Dans le clustering hiérarchique diviseur, tous les points de données sont considérés comme un seul cluster au début. À chaque étape, le cluster est divisé en deux clusters plus petits en fonction de leur dissimilarité. Ce processus est répété jusqu'à ce que chaque point de données soit dans son propre cluster.

Les avantages du clustering hiérarchique sont qu'il n'est pas nécessaire de spécifier le nombre de clusters à l'avance, et qu'il peut fournir une représentation visuelle des relations de similarité entre les données sous forme de dendrogramme. Cependant, les inconvénients sont qu'il peut être plus lent que d'autres algorithmes de clustering, surtout lorsque les ensembles de données sont volumineux, et qu'il peut être plus difficile à interpréter que d'autres méthodes de clustering.

Des exemples d'algorithmes de clustering hiérarchique sont le Ward's linkage, le Single linkage, le Complete linkage, et le Average linkage. Chacun de ces algorithmes utilise une méthode différente pour calculer la similarité entre les clusters à chaque étape.

3. Les algorithmes de clustering basés sur la théorie floue

Les algorithmes de clustering basés sur la théorie floue ou "Fuzzy Clustering" en anglais, sont des algorithmes qui permettent de regrouper des données en fonction de leurs similarités et de leurs différences. Contrairement aux algorithmes de clustering classiques qui attribuent chaque point de données à un groupe spécifique, les algorithmes de clustering flous attribuent chaque point de données à chaque groupe avec un certain degré d'appartenance, c'est-à-dire une probabilité que le point de données appartienne à chaque groupe.

Le regroupement flou est utile dans des situations où les frontières entre les groupes ne sont pas nettement définies, ou lorsque les points de données peuvent appartenir à plusieurs groupes à la fois. Par exemple, dans le cas de la segmentation de marché, un client peut appartenir à plusieurs segments en fonction de ses préférences et de son comportement d'achat.

Les algorithmes de clustering flous utilisent des fonctions d'appartenance pour déterminer le degré d'appartenance de chaque point de données à chaque groupe. Les fonctions d'appartenance peuvent être définies de différentes manières, telles que la distance euclidienne ou la similarité cosinus.

Il existe plusieurs algorithmes de clustering flous populaires, tels que le Fuzzy C-means (FCM), le Gustafson-Kessel (GK), et le Possibilistic C-means (PCM). Ces algorithmes sont utilisés dans divers domaines, tels que la reconnaissance de formes, la segmentation d'image, la classification de texte, etc.

4. Les algorithmes de clustering basé sur la distribution

Le clustering basé sur la distribution, ou "Density-Based Clustering" en anglais, est une méthode de clustering qui identifie les zones de haute densité de points de données et les considère comme des clusters, tandis que les zones de basse densité sont considérées comme des points de bruit ou des outliers. Cette méthode est particulièrement utile pour les ensembles de données avec des clusters de formes complexes et de tailles différentes, ainsi que pour les ensembles de données avec des valeurs aberrantes ou du bruit.

L'algorithme de clustering basé sur la distribution le plus populaire est DBSCAN (Density-Based Spatial Clustering of Applications with Noise), qui commence par sélectionner un point de données aléatoire et identifie tous les points de données dans son voisinage, qui sont définis par une distance maximale prédéfinie et le nombre minimal de points de données requis pour former un cluster. Ensuite, DBSCAN étend le cluster en ajoutant tous les points de données dans le voisinage de ces points de données jusqu'à ce que tous les points de données du cluster soient inclus. Les points de données qui ne sont pas inclus dans un cluster sont considérés comme du bruit.

Les avantages de DBSCAN et des algorithmes de clustering basés sur la distribution sont qu'ils sont capables de détecter des clusters de forme arbitraire et de tailles différentes, et qu'ils sont robustes aux valeurs aberrantes et au bruit. Cependant, leur principal inconvénient est que leur performance dépend de la qualité de la définition des paramètres, tels que la distance maximale et le nombre minimal de points de données requis, ce qui peut être difficile à déterminer pour certains ensembles de données.

D'autres exemples d'algorithmes de clustering basés sur la distribution sont le Mean Shift, le OPTICS (Ordering Points To Identify the Clustering Structure), et le HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise).

5. Les algorithmes de clustering basé sur la densité

Le clustering basé sur la densité, ou "Density-Based Clustering" en anglais, est une méthode de clustering qui identifie les zones de haute densité de points de données et les considère comme des clusters, tandis que les zones de basse densité sont considérées comme des points de bruit ou des outliers. Cette méthode est particulièrement utile pour les ensembles de données avec des clusters de formes complexes et de tailles différentes, ainsi que pour les ensembles de données avec des valeurs aberrantes ou du bruit.

L'algorithme de clustering basé sur la densité le plus populaire est DBSCAN (Density-Based Spatial Clustering of Applications with Noise), qui commence par sélectionner un point de données aléatoire et identifie tous les points de données dans son voisinage, qui sont définis par une distance maximale prédéfinie et le nombre minimal de points de données requis pour former un cluster. Ensuite, DBSCAN étend le cluster en ajoutant tous les points de données dans le voisinage de ces points de données jusqu'à ce que tous les points de données du cluster soient inclus. Les points de données qui ne sont pas inclus dans un cluster sont considérés comme du bruit.

Les avantages de DBSCAN et des algorithmes de clustering basés sur la densité sont qu'ils sont capables de détecter des clusters de forme arbitraire et de tailles différentes, et qu'ils sont robustes aux valeurs aberrantes et au bruit. Cependant, leur principal inconvénient est que leur performance dépend de la qualité de la définition des paramètres, tels que la distance maximale et le nombre minimal de points de données requis, ce qui peut être difficile à déterminer pour certains ensembles de données.

D'autres exemples d'algorithmes de clustering basés sur la densité sont le Mean Shift, le OPTICS (Ordering Points To Identify the Clustering Structure), et le HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise).

6. Les algorithmes de clustering basé sur la théorie des graphes

Les algorithmes de clustering basés sur la théorie des graphes sont une classe d'algorithmes de clustering qui utilisent des graphes pour représenter les relations entre les points de données et les regrouper en clusters.

L'un des algorithmes les plus populaires dans cette catégorie est l'algorithme de clustering spectral, qui utilise la théorie des graphes pour extraire les structures de clusters à partir des données. L'algorithme commence par construire un graphe à partir des données d'entrée, où chaque point de données est représenté comme un nœud et les relations entre les points sont représentées par des arêtes. Ensuite, l'algorithme calcule les vecteurs propres associés aux valeurs propres les plus petites de la matrice de Laplacien normalisé du graphe. Ces vecteurs propres sont ensuite utilisés pour projeter les points de données dans un nouvel espace de représentation où la structure de cluster est plus facile à identifier. Les clusters sont ensuite extraits à partir de cette représentation en utilisant des techniques de clustering standard, telles que la méthode de K-moyennes.

Un autre algorithme de clustering basé sur la théorie des graphes est le clustering hiérarchique basé sur le graff, qui construit un arbre de clustering en utilisant un graphe pour représenter les relations entre les points de données. L'algorithme commence par construire un graphe pondéré à partir des données d'entrée, où chaque nœud représente un point de données et les arêtes entre les nœuds sont pondérées en fonction de la similarité entre les points de données correspondants. Ensuite, l'algorithme construit un arbre de clustering en utilisant des méthodes de partitionnement hiérarchique, où chaque nœud de l'arbre représente un cluster de points de données et les feuilles de l'arbre représentent les points de données individuels.

Les avantages des algorithmes de clustering basés sur la théorie des graphes sont qu'ils sont capables de détecter des structures de clusters de forme arbitraire et qu'ils sont robustes aux valeurs aberrantes

et au bruit. Cependant, leur principal inconvénient est qu'ils peuvent être sensibles aux paramètres de construction du graff, tels que le choix de la fonction de similarité ou le seuil de similarité, ce qui peut être difficile à déterminer pour certains ensembles de données.

7. Les algorithmes de clustering basés sur la grille

Les algorithmes de clustering basés sur la grille sont une classe d'algorithmes de clustering qui partitionnent l'espace des données en une grille régulière, puis regroupent les points de données en fonction de leur position dans la grille. Ces algorithmes sont également connus sous le nom de "clustering basé sur la grille" ou "clustering basé sur la quantification de la grille".

L'un des algorithmes les plus populaires dans cette catégorie est l'algorithme de quantification vectorielle de la grille (Grid-Based Vector Quantization ou GBVQ), qui commence par diviser l'espace des données en une grille régulière. Chaque case de la grille est associée à un vecteur de code qui représente le centre de la case. Les points de données sont ensuite attribués au vecteur de code le plus proche. Les vecteurs de code sont ensuite mis à jour en utilisant les points de données associés à chaque vecteur de code. Ce processus est répété jusqu'à ce que les vecteurs de code convergent vers une solution stable.

Un autre algorithme de clustering basé sur la grille est l'algorithme de clustering basé sur la grille auto-organisée (Self-Organizing Grid-Based Clustering ou SOGB), qui est une extension de l'algorithme de quantification vectorielle de la grille. L'algorithme commence par diviser l'espace des données en une grille régulière et initialise les vecteurs de code pour chaque case de la grille à des valeurs aléatoires. Ensuite, l'algorithme attribue chaque point de données au vecteur de code le plus proche et ajuste les vecteurs de code en fonction des points de données associés à chaque vecteur de code. Les vecteurs de code sont ajustés de manière à rapprocher les vecteurs de code similaires et à éloigner les vecteurs de code dissimilaires. Ce processus est répété jusqu'à ce que les vecteurs de code convergent vers une solution stable.

Les avantages des algorithmes de clustering basés sur la grille sont qu'ils sont rapides et efficaces pour les ensembles de données de grande taille. De plus, ils sont capables de détecter des clusters de forme arbitraire et sont robustes aux valeurs aberrantes et au bruit. Cependant, leur principal inconvénient est qu'ils peuvent être sensibles au choix de la taille et de la forme de la grille, ce qui peut être difficile à déterminer pour certains ensembles de données.

Les plus populaires algorithmes de clustering

Il n’y a pas d’algorithme de clustering unique qui soit le plus utilisé en général, car cela dépend souvent du type de données, du domaine d’application, des objectifs de la tâche de clustering et des préférences des utilisateurs. Cependant, certains algorithmes de clustering sont plus populaires et couramment utilisés que d’autres en raison de leur efficacité, leur simplicité, leur robustesse ou leur capacité à traiter des données de grande dimension.

1. K-means

L'algorithme des K-moyennes (K-means) est un algorithme non supervisé très connu en matière de Clustering. C'est un algorithme de clustering de partitionnement.

Cet algorithme a été conçu en 1957 au sein des Laboratoires Bell par Stuart P.Lloyd comme technique de modulation par impulsion et codage(MIC) . Il n'a été présenté au grand publique qu'en 1982. En 1965 Edward W.Forgy avait déjà publié un algorithme quasiment similaire c'est pourquoi le K-means est souvent nommé algorithme de Lloyd-Forgy.

Les champs d'application sont divers : segmentation client, analyse de donnée, segmenter une image, apprentissage semi-supervisé.

Étant donnés des points et un entier k, l'algorithme vise à diviser les points en k groupes, appelés clusters, homogènes et compacts.

2. DBSCAN clustering algorithm

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) est un algorithme de regroupement populaire en apprentissage automatique et en fouille de données. Il est conçu pour découvrir des groupes de formes arbitraires dans un ensemble de données en se basant sur la densité des points de données. Contrairement à certains autres algorithmes de regroupement, DBSCAN ne nécessite pas que le nombre de groupes soit prédéfini, ce qui le rend adapté à l'analyse exploratoire des données.

Voici comment fonctionne DBSCAN :

1. Basé sur la densité : DBSCAN repose sur l'idée que les groupes sont des régions de haute densité séparées par des régions de faible densité. Il définit les groupes comme des régions denses de points de données séparées par des zones moins denses.
2. Points centraux, points de bordure et points de bruit : DBSCAN catégorise les points de données en trois types : les points centraux, les points de bordure et les points de bruit. Un point central est un point de données qui possède un nombre minimal de points voisins (défini par les paramètres min_samples et epsilon) dans un rayon spécifié (epsilon). Un point de bordure a moins de points voisins que le minimum requis, mais se trouve à l'intérieur du rayon d'un point central. Les points de bruit n'ont ni un nombre suffisant de voisins ni ne se trouvent à l'intérieur du rayon d'un point central.
3. Formation des groupes : DBSCAN commence par sélectionner un point central arbitraire et étend le groupe en ajoutant tous les points centraux directement atteignables (et leurs voisins respectifs) au groupe. Ce processus se poursuit de manière récursive jusqu'à ce qu'aucun point central supplémentaire ne puisse être ajouté. Ensuite, un nouveau point central est sélectionné et le processus se répète, créant ainsi un nouveau groupe. Cela continue jusqu'à ce que tous les points de données aient été traités.
4. Points de bordure et points de bruit : Les points de bordure qui ne font partie d'aucun groupe sont considérés comme des points de bruit. Cependant, ils peuvent toujours être connectés à un groupe via d'autres points de bordure.
5. Paramètres de l'algorithme :

- epsilon (ε) : Ce paramètre détermine le rayon dans lequel chercher les points voisins autour de chaque point de données.
- min_samples : Il spécifie le nombre minimal de points voisins (incluant le point de données lui-même) requis pour qu'un point soit considéré comme un point central.

3. Gaussian Mixture Model algorithm

Le modèle de mélange gaussien (Gaussian Mixture Model ou GMM en anglais) est un algorithme de regroupement probabiliste largement utilisé en apprentissage automatique. Il est utilisé pour modéliser des ensembles de données complexes en supposant que les données proviennent d'une combinaison de plusieurs distributions gaussiennes.

Voici comment fonctionne le modèle de mélange gaussien :

1. Initialisation : L'algorithme commence par initialiser les paramètres du modèle de manière aléatoire ou en utilisant une méthode comme l'algorithme de regroupement k-means. Ces paramètres comprennent les moyennes, les matrices de covariance et les poids associés à chaque distribution gaussienne.
2. Expectation-Maximization (EM) : Le modèle de mélange gaussien utilise l'algorithme EM pour estimer les paramètres optimaux du modèle. L'algorithme EM est un processus itératif qui comprend deux étapes : l'étape d'espérance (Expectation) et l'étape de maximisation (Maximization).
    - Étape d'espérance : Dans cette étape, les responsabilités (probabilités d'appartenance) de chaque point de données à chaque distribution gaussienne sont calculées à l'aide de la règle de Bayes. Cela permet d'estimer la probabilité qu'un point de données appartienne à chaque composante du modèle.
    - Étape de maximisation : Dans cette étape, les paramètres du modèle (moyennes, matrices de covariance et poids) sont mis à jour en maximisant la vraisemblance du modèle, en utilisant les responsabilités calculées à l'étape précédente.

    Ces deux étapes (espérance et maximisation) sont répétées jusqu'à ce que la convergence soit atteinte, c'est-à-dire que les paramètres du modèle ne changent que de manière négligeable entre les itérations successives.

3. Attribution des clusters : Une fois que le modèle a convergé, chaque point de données peut être attribué au cluster correspondant à la distribution gaussienne qui a la plus grande probabilité d'appartenance. Cela permet d'obtenir une partition des données en clusters.
4. Prédiction : Une fois le modèle entraîné, il peut également être utilisé pour prédire les clusters de nouveaux points de données qui ne faisaient pas partie de l'ensemble d'apprentissage initial.
5. Avantages du modèle de mélange gaussien :
    - Le modèle de mélange gaussien peut modéliser des ensembles de données complexes avec des formes de clusters arbitraires.
    - Il fournit une estimation probabiliste de l'appartenance à chaque cluster, ce qui permet d'effectuer une analyse plus fine des données.
    - Il est également capable de gérer des données avec des dimensions élevées.
    - Il est largement utilisé dans des applications telles que la classification d'images, la détection d'anomalies et la segmentation d'images médicales.

4. BIRCH algorithm

L'algorithme BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) est un algorithme de regroupement hiérarchique conçu pour traiter de grandes quantités de données. Il est utilisé pour effectuer un regroupement rapide et efficace en construisant une structure d'arbre appelée "Clustering Feature Tree" (CFT).

Voici comment fonctionne l'algorithme BIRCH :

1. Construction du Clustering Feature Tree (CFT) : L'algorithme commence par construire un CFT en utilisant un ensemble de données initial. Le CFT est un arbre hiérarchique qui stocke des informations agrégées sur les clusters et permet une exploration et une mise à jour rapides.
2. Points d'entrée (Entry Points) : Le CFT utilise des points d'entrée pour représenter les clusters. Les points d'entrée sont des vecteurs de caractéristiques compacts qui résument les données d'un cluster spécifique.
3. Partitionnement : L'algorithme BIRCH utilise une stratégie de partitionnement basée sur la densité pour ajouter de nouveaux points de données à l'arbre. Les nouveaux points de données sont insérés dans les sous-clusters appropriés du CFT en se basant sur leur proximité avec les points d'entrée existants.
4. Agrégation : Lors de l'insertion de nouveaux points de données, l'algorithme met à jour les informations agrégées dans le CFT, telles que le nombre de points de données, les sommets et les caractéristiques moyennes.
5. Fusion : Si un sous-cluster dépasse une certaine taille prédéfinie, il peut être fusionné avec d'autres sous-clusters pour maintenir l'efficacité de l'algorithme. La fusion est effectuée en mettant à jour les informations agrégées et en réorganisant les nœuds de l'arbre.
6. Regroupement final : Une fois que tous les points de données ont été ajoutés à l'arbre, l'algorithme peut extraire les clusters finaux en parcourant le CFT et en récupérant les points d'entrée et leurs sous-clusters.
7. Avantages de l'algorithme BIRCH :
    - BIRCH est capable de traiter efficacement de grandes quantités de données, ce qui le rend approprié pour des problèmes avec des ensembles de données massifs.
    - Il utilise une structure d'arbre pour un accès rapide aux informations sur les clusters, ce qui permet une exploration et une mise à jour efficaces.
    - Il peut traiter des données continues et catégorielles.
    - BIRCH est robuste face au bruit et aux valeurs aberrantes.
    - Il est également adapté pour le regroupement en ligne, où de nouveaux points de données peuvent être ajoutés progressivement.
    - Cependant, BIRCH peut ne pas être aussi précis que certains autres algorithmes de regroupement, comme le k-means, en termes de séparation des clusters.

Deux des algorithmes développés dernièrement

L'un des algorithmes de clustering les plus récents est le clustering par densité adaptative (Adaptive Density Clustering, ADC), qui a été proposé en 2021. L'ADC est un algorithme de clustering non paramétrique qui peut être utilisé pour détecter des clusters de formes arbitraires et de densités variables dans des données multidimensionnelles.

Un autre exemple est l'algorithme de clustering par recherche de voisinage (Neighborhood Search Clustering, NSC) qui a également été proposé en 2021. L'NSC est un algorithme de clustering basé sur la recherche de voisinage qui peut être utilisé pour détecter des clusters dans des données de haute dimension.

1. Adaptive Density Clustering

Le Adaptive Density Clustering (ADC) est un algorithme de regroupement qui se base sur la densité adaptative pour identifier des clusters dans un ensemble de données. Contrairement à certains autres algorithmes de regroupement qui supposent une distribution spécifique des données, l'ADC est capable de détecter des clusters de formes arbitraires sans nécessiter de paramètres prédéfinis tels que le nombre de clusters.

Voici comment fonctionne l'Adaptive Density Clustering :

1. Estimation de la densité locale : L'algorithme commence par estimer la densité locale de chaque point de données en utilisant une mesure de la densité adaptative. Cette mesure prend en compte la densité des points voisins de chaque point et permet d'adapter la définition de la densité en fonction de la distribution réelle des données.
2. Identification des centres de cluster : L'ADC identifie les centres de cluster en sélectionnant les points de données avec une densité locale suffisamment élevée. Ces points sont considérés comme les candidats potentiels pour représenter les centres des clusters.
3. Attribution des points aux clusters : L'algorithme attribue ensuite chaque point de données à un cluster en utilisant des critères basés sur la densité et la proximité. Les points de données qui sont suffisamment proches d'un centre de cluster et présentent une densité locale élevée sont attribués à ce cluster.
4. Extension des clusters : L'ADC étend progressivement les clusters en ajoutant des points de données qui sont proches des points déjà attribués à un cluster. Cette extension se fait en utilisant à la fois la proximité spatiale et la densité locale des points.
5. Pruning : Après l'extension des clusters, l'algorithme peut effectuer une étape de "pruning" pour éliminer les points de données qui ont été incorrectement attribués à un cluster. Cette étape vise à améliorer la qualité des clusters formés.
6. Avantages de l'Adaptive Density Clustering :
    - L'ADC est capable de détecter des clusters de formes arbitraires et n'impose pas de contraintes sur la distribution des données.
    - Il ne nécessite pas de paramètres prédéfinis tels que le nombre de clusters, ce qui le rend adaptatif et flexible.
    - L'algorithme est résistant au bruit et peut gérer efficacement des ensembles de données de grande taille.
    - Il est également capable de détecter des clusters de densités différentes au sein d'un même ensemble de données.
    - L'ADC peut être utilisé dans divers domaines tels que l'exploration de données, la reconnaissance de formes et l'analyse de données non structurées.

2. Neighborhood Search Clustering

Neighborhood Search Clustering (NSC) est un algorithme de regroupement qui se base sur la recherche de voisinage pour identifier des clusters dans un ensemble de données. C'est une approche basée sur la densité qui vise à regrouper les points de données qui sont proches les uns des autres dans l'espace.

Voici comment fonctionne l'algorithme Neighborhood Search Clustering :

1. Construction du graphe de voisinage : L'algorithme commence par construire un graphe de voisinage en reliant chaque point de données à ses voisins les plus proches en fonction d'une

mesure de distance prédéfinie. La taille du voisinage est généralement déterminée par un paramètre spécifié par l'utilisateur.

2. Identification des points de départ : NSC identifie les points de départ potentiels pour les clusters en sélectionnant les points qui ont un nombre suffisant de voisins dans leur voisinage défini précédemment. Ces points sont considérés comme des candidats pour représenter les centres des clusters.
3. Extension des clusters : L'algorithme étend progressivement les clusters en ajoutant des points de données qui sont connectés à un cluster existant via des relations de voisinage. Un point est ajouté à un cluster s'il est suffisamment proche d'au moins un point du cluster et s'il satisfait les critères de densité prédéfinis.
4. Attribution des points aux clusters : NSC attribue ensuite chaque point de données à un cluster en utilisant des critères basés sur la densité et la proximité. Les points de données qui satisfont ces critères et sont connectés à un cluster sont attribués à ce cluster.
5. Itérations et fusion des clusters : L'algorithme effectue des itérations jusqu'à ce qu'aucun point de données supplémentaire ne puisse être ajouté à un cluster. Ensuite, les clusters qui se chevauchent ou sont très similaires peuvent être fusionnés pour obtenir une meilleure représentation des groupes.
6. Avantages du Neighborhood Search Clustering :
    - NSC est capable de détecter des clusters de formes arbitraires sans avoir besoin de spécifier le nombre de clusters à l'avance.
    - Il est adaptatif aux densités locales des données et peut trouver des clusters de différentes densités.
    - L'algorithme est robuste face aux bruits et aux valeurs aberrantes, car il se concentre sur les relations de voisinage plutôt que sur des mesures globales de densité.
    - NSC est relativement rapide et peut être efficacement utilisé pour des ensembles de données volumineux.
    - Il est largement utilisé dans des domaines tels que l'exploration de données, la reconnaissance de formes et l'analyse de données non structurées

Méthodes de collecte de flux d'information : pull & push

La méthode de collecte de flux d'information "pull" fait référence à un modèle où les utilisateurs récupèrent activement les informations en demandant explicitement des mises à jour ou en interrogeant des sources spécifiques.

La méthode "pull" donne aux utilisateurs un contrôle plus direct sur les informations auxquelles ils accèdent et leur permet de choisir les sources et les moments de récupération des données. Elle est souvent utilisée dans les cas où les utilisateurs ont des préférences et des intérêts spécifiques et souhaitent obtenir des informations à la demande.

Il est important de noter que bien que la méthode "pull" donne aux utilisateurs un certain niveau de contrôle, elle dépend toujours de la disponibilité des sources d'information et de leur volonté de fournir des mises à jour régulières.

La méthode de collecte de flux d'information "push" est un modèle dans lequel les utilisateurs reçoivent automatiquement des mises à jour ou des notifications d'informations sans avoir à les demander explicitement. Dans ce cas, les informations sont poussées vers les utilisateurs par des sources ou des systèmes.

La méthode "push" permet aux utilisateurs de recevoir des informations pertinentes sans avoir à les rechercher activement. Elle est souvent utilisée pour fournir des actualités, des mises à jour en direct, des notifications d'événements, des rappels, des offres spéciales, etc. Cela peut être particulièrement utile pour les utilisateurs qui souhaitent rester à jour sans avoir à vérifier manuellement les sources d'information.

Il convient de noter que les utilisateurs doivent généralement donner leur consentement et avoir un certain contrôle sur les notifications push qu'ils reçoivent. Ils peuvent également choisir de se désabonner ou de gérer leurs préférences de notification à tout moment.

La méthode "push" nécessite une infrastructure technique pour la diffusion des informations et peut nécessiter un suivi précis des préférences et des consentements des utilisateurs pour se conformer aux réglementations sur la confidentialité et la protection des données.

Dans le cadre de ce rapport, les deux méthodes de collecte de flux d'information décrites précédemment ont été utilisées, à savoir la méthode pull et la méthode push, afin d'obtenir une vue complète des informations pertinentes dans le domaine du Big Data.

Pour la méthode pull, des sites spécifiés dans la bibliographie ont été consultés, sites qui ont permis d'obtenir des mises à jour régulières sur les avancées et les nouvelles recherches dans le domaine. Des fils d'actualités et des newsletters ont été suivis pour recevoir les dernières informations provenant de sources fiables et de confiance. Cette approche a permis de collecter des informations précises et ciblées, en nous assurant de rester à jour avec les développements récents.

En ce qui concerne la méthode push, nous avons créé des comptes sur des plateformes en ligne telles que https://www.datasciencecentral.com/ afin de recevoir des mises à jour automatiques. Ces plateformes envoient des notifications et des e-mails pour informer les utilisateurs des nouvelles publications, des discussions intéressantes et des ressources pertinentes dans le domaine du Big Data. En s’abonnant à certaines chaînes et en personnalisant les préférences, il est possible de recevoir des notifications push et des e-mails ciblés, ce qui permit de rester informés des avancées et des tendances importantes.

L'utilisation combinée des méthodes pull et push a permis d'avoir une couverture étendue de l'information dans le domaine du Big Data.

Cette approche complète de collecte de flux d'information a permis de rassembler des informations de manière efficace et de rester à jour avec les avancements dans le vaste domaine du Big Data.

Bibliographie

https://cran.r-project.org/web/packages/TSclust/TSclust.pdf

https://www.rdocumentation.org/packages/NbClust/versions/3.0.1/topics/NbClust

https://www.netapp.com/fr/artificial-intelligence/what-is-machine-learning/

https://www.sap.com/france/products/artificial-intelligence/what-is-machine-learning.html

https://www.jedha.co/blog/les-differents-types-de-machine-learning

https://datascientest.com/algorithme-des-k-means

https://larevueia.fr/clustering-les-3-methodes-a-connaitre/

https://fr.linedata.com/principaux-algorithmes-dapprentissage-non-supervise

https://www.talend.com/fr/resources/what-is-artificial-intelligence/

https://guidesurvie.com/techniques-survie/analyse-des-clusters-wikipedia/

https://analyticsindiamag.com/all-you-need-to-know-about-time-series-clustering/

https://www.freecodecamp.org/news/8-clustering-algorithms-in-machine-learning-that-all-data-scientists-should-know/

https://brightcape.co/le-clustering-definition-et-implementations/

https://link.springer.com/article/10.1007/s40745-015-0040-1

https://towardsdatascience.com/time-series-clustering-deriving-trends-and-archetypes-from-sequential-data-bb87783312b4

https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

https://www.sciencedirect.com/science/article/abs/pii/S0020025521001596

http://www.iafactory.fr/service-ux/strategie-editoriale/flux-d-information.php

https://www.canal-u.tv/chaines/unit/les-methodes-de-collecte-d-information

https://link.springer.com/article/10.1007/s10489-021-02934-x

https://www.sciencedirect.com/science/article/abs/pii/S0305054821001386