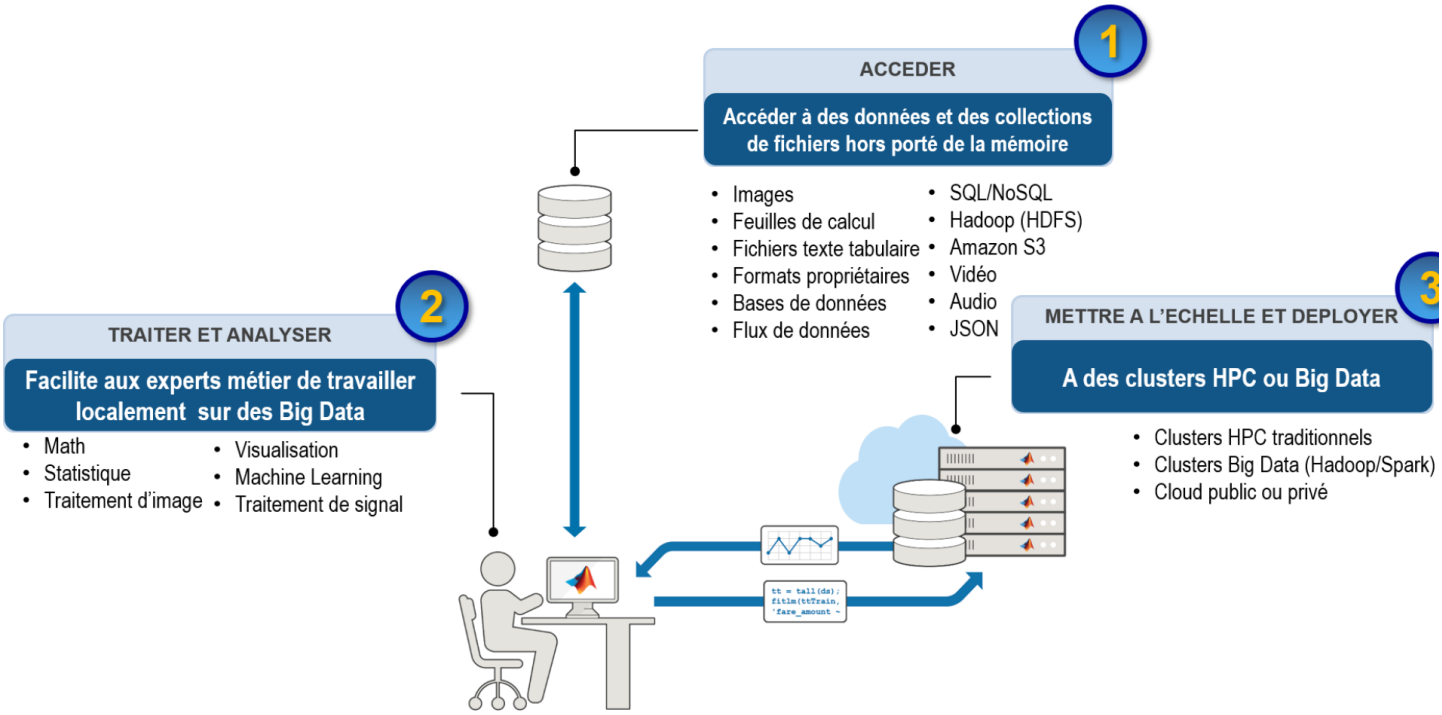


Peut-on rendre la création des applications Big Data accessible à tous ?

Publié le 28 juin 2017

Lyamine Hedjazi | Suivre
Application Engineer in Data Science chez MathWorks

Avec un nombre croissant de données collectées à partir d'objets connectés, de systèmes industriels et d'outils de surveillance environnementale, les entreprises exploitent des téraoctets de données pour optimiser le fonctionnement de leurs équipements et se démarquer par leur offre de produits et services.

Accéder à ces données et les exploiter peut sembler, au premier abord, une tâche colossale. Jusqu'à récemment, pour appliquer des techniques avancées telles que l'apprentissage automatique (machine learning) à des jeux de données très volumineux, les ingénieurs et scientifiques devaient s'associer avec des informaticiens expérimentés en systèmes IT et en développement d'algorithmes exécutés sur des clusters informatiques. Cette équipe devait mettre en place un workflow comprenant les étapes suivantes:

1. *Accéder à de vastes jeux de données dans des fichiers, bases de données ou systèmes de fichiers distribués Hadoop*

2. *Explorer, traiter et analyser ces données*

algorithmes au sein des applications d'entreprise

Aujourd'hui, les entreprises n'ont plus besoin de constituer une équipe d'informaticiens pour accompagner leurs ingénieurs et scientifiques, ni de se mettre à la recherche du légendaire 'Data Scientist'. Les outils auxquels les ingénieurs et scientifiques en sont venus à se fier leur permettent de pratiquer l'art du Big Data.

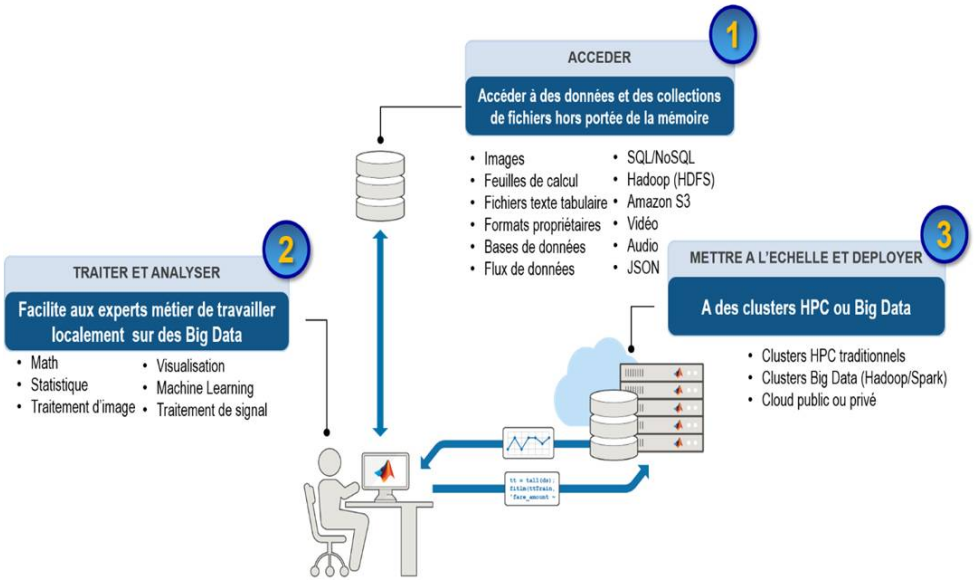


Figure1. Workflow Big Data

Dans cet article, nous aborderons tout d'abord les principaux systèmes utilisés pour stocker de vastes jeux de données techniques et scientifiques, et les aspects que nous devons connaître pour accéder à ces données et les traiter sur notre station de travail. Nous parlerons ensuite plus en détail des méthodes de traitement de ces vastes jeux de données, puis nous essaierons de démystifier Hadoop et Spark et expliquerons comment intégrer ces outils à votre workflow de Big Data (Figure 1).

Accéder aux données

Lorsque l'on travaille avec le Big Data, la première difficulté consiste à accéder à ces vastes jeux de données, qui se présentent sous de nombreuses formes et sont stockées sur différents types de systèmes :

- **Fichiers :** ils se trouvent généralement dans un ou plusieurs répertoires sur un disque partagé, et se présentent sous forme de textes délimités, feuilles de calcul, images, vidéos et divers formats propriétaires.

- **Bases de données** : il en existe de nombreux types, utilisés pour stocker et gérer les données. Parmi les plus fréquemment rencontrés, citons les systèmes relationnels (SQL), non-relationnels (NoSQL), d'historisation des données et d'agrégation de données IoT. L'un des points communs à tous ces systèmes de bases de données est qu'ils sont en mesure de gérer un volume de données dépassant les capacités de la mémoire d'un ordinateur.

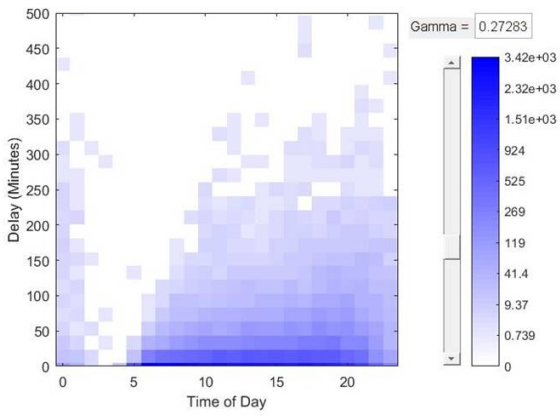
base sur le principe de la distribution du calcul et du système de fichiers. Les données sont stockées dans un système de fichiers tolérant aux pannes appelé Hadoop Distributed File System (HDFS). Le processus d'accès et de traitement des données stockées sur HDFS est ensuite géré par le gestionnaire de clusters YARN (Yet Another Resource Navigator) en se basant sur plusieurs technologies, les plus courantes étant MapReduce et Spark.

Nous constatons que les entreprises adoptent au moins deux de ces plates-formes de stockage pour conserver et gérer leurs données techniques et scientifiques. Il est donc important de disposer d'un outil nous permettant d'accéder facilement aux différents types de données et systèmes de stockage depuis notre station de travail. Les objets datastore de MATLAB permettent d'accéder différents formats de données quelque soit leurs lieu de stockage.

Explorer, traiter et analyser ces données

La possibilité, dans un premier temps, d'opérer sur notre station de travail peut grandement améliorer notre efficacité lorsqu'il s'agit de manipuler de vastes jeux de données. Des fonctionnalités telles que les conteneurs de données « tall arrays » dans MATLAB comprennent des outils mathématiques, statistiques et de nettoyage de données, ainsi que d'autres fonctions conçues pour traiter des jeux de données trop volumineux pour être gérés intégralement par la mémoire. Nous pouvons ainsi traiter ces données volumineux sur notre station de travail standard à l'aide d'un langage de programmation intuitif. Nous savons maintenant que nous pouvons utiliser notre station de travail, mais nous n'avons toujours pas répondu à la question « par où commencer ? ».

- **Visualisation :** pour comprendre la nature de nos données, il est intéressant de commencer par la visualisation. Les outils de visualisation de synthèse, comme la fonction binScatterPlot illustrée dans la Figure 2, nous permettent de mieux appréhender ces jeux de données en utilisant des tall arrays. binScatterPlot est une fonction de visualisation de synthèse qui illustre, en jouant sur l'intensité ajustable de la couleur, les zones de forte concentration de points de données. En ajustant l'intensité de la couleur à l'aide du curseur, nous pouvons nous livrer à une exploration visuelle rapide et obtenir une meilleure connaissance de la structure des données.



- **Large assortiment de fonctions flexibles :** si nous utilisons de prime abord l'intégralité de notre jeu de données pour l'exploration et le prototypage, nous nous exposons à de très longues heures de frustration et d'inefficacité. Avec les fonctions conçues pour gérer des jeux données hors porté de la mémoire, nous pouvons traiter et analyser un sous-ensemble de nos données, puis étendre nos algorithmes à l'intégralité des données sans les modifier.

- **Machine Learning :** vous pouvez développer dans le même contexte des modèles prédictifs à l'aide d'une large palette de techniques de machine learning, comme illustré dans la Figure 3.

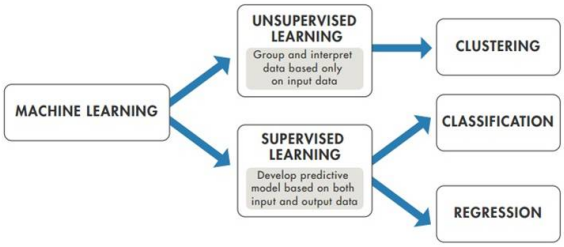


Figure 3. Techniques de Machine Learning

- **Infrastructure de traitement efficace :** lorsque nous travaillons avec des jeux de données sur notre ordinateur, l'efficacité de l'infrastructure de traitement est un aspect fondamental. Avec une infrastructure capable de différer l'exécution de commandes spécifiques et d'optimiser un ensemble de commandes de manière à minimiser le nombre de fois où les données sont parcourues, il est possible de traiter de vastes jeux de données, même sur une station de travail.

Évoluer vers des plateformes Big Data et le déploiement d'applications

Une fois que nous avons prototypé notre algorithme sur notre ordinateur, nous n'avons qu'une hâte : déporter notre application sur des plateformes Big Data comme Hadoop pour traiter de vastes jeux de données ou la déployer comme une application entreprise.

- **Mise à l'échelle sur des plateformes Big Data :** il existe un certain nombre d'applications qui sont exécutées sur Hadoop, mais MapReduce et Spark, ainsi que certaines interfaces de type SQL, comptent parmi les plus utilisées.

- MapReduce : il s'agit de la première technique de traitement rendue disponible sur Hadoop. Elle est généralement utilisée pour les transformations de très vastes jeux de données.

- Apache Spark : cette plateforme, plus récente, offre un modèle de programmation plus générique, ainsi qu'un outil de mise en cache et une infrastructure d'évaluation différée. Elle est donc idéale pour l'analyse statistique et l'apprentissage automatique.

pour accéder à vos données et y appliquer des algorithmes d'analyse limités.

Nous pouvons avoir l'impression dans un premier temps que le processus d'accès et de gestion de données stockées dans Hadoop à l'aide de tels systèmes requiert des outils et des compétences spécialisées. La bonne nouvelle, c'est que nous pouvons là encore appliquer le processus décrit dans les deux parties précédentes de cet article. Les outils d'analyse comme MATLAB s'appuient sur ces environnements de traitement, et permettent aux codes basés sur MATLAB d'accéder aux données dans HDFS et de les traiter sur un cluster Hadoop. A ce titre, les algorithmes MapReduce de MATLAB peuvent etre déporté sur un cluster Hadoop avec des modifications mineurs. Les tall arrays fournie aussi un interfaçage automatique avec Spark.

Pour aller plus loin, un certain nombre d'organisations doivent déployer leurs algorithmes Big Data avant de les intégrer à une application d'entreprise, tout en conservant la propriété de leurs idées et de leurs algorithmes.

- **Déploiement d'applications :**le déploiement de vos applications Big Data chez vos clients apporte une valeur ajoutée à votre entreprise. Il existe différents moyens de déployer des composants pouvant être exécutés sur les clusters Hadoop pour des applications Big Data.

- Déployer des applications MapReduce sur un cluster Hadoop : vous packagez vos algorithmes MapReduce développés sur ordinateur afin de les exécuter sur un cluster Hadoop.

- Déployer des applications sur un cluster combinant Hadoop et Spark : vous créez et packagez une application autonome, exécutable sur un cluster combinant Hadoop et Spark.

- Générer du code à partir de modèles Big Data : vous générez des modèles prédictifs sous forme de code embarqué pouvant être intégré à un appareil, véhicule, système IT ou service Web.

Conclusion

L'exploitation du Big Data ne devrait pas impliquer la maitrise de nouveaux paradigmes et outils de programmation, ni l'utilisation de systèmes spécialisés. Pour devenir votre propre Big Data Scientist, vous devez en revanche obligatoirement disposer d'outils capables à la fois de mettre en œuvre un workflow Big Data utilisant les techniques de programmation auxquels vous êtes habitués et de s'adapter aux nouvelles plateformes telles que Hadoop et Spark.

Lyamine Hedjazi
Application Engineer in Data Science chez MathWorks
1 article

Vous souhaitez suivre d'autres titres d'actualité sur LinkedIn ?

Découvrir d'autres articles