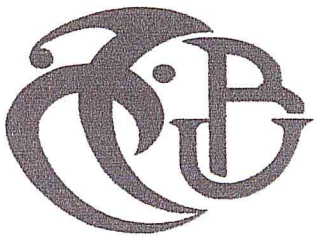
République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

**Université Saad Dahlab Blida**

N° D'ordre :.....



Faculté des sciences

**Département d'informatique**


Mémoire élaboré par :

Lakehal Yacine khelladi Sohaib

**En vue de l'obtention du diplôme de master**

Domaine : Mathématique et informatique

Filière : Informatique
Spécialité : Informatique
Option : Ingénierie de logiciel

**Sujet :** ***Formalisation d'une approche systématique dans le cadre d'un apprentissage supervisé simple de l'exploration des données à la prédiction d'une variable cible.***
***Organisme : GTP***

**Soutenu le :**

| | |
|---|---|
| M.Chrife Zaher | Président |
| M.Zahra Fatimazohra | Examinateur |
| M. Kamache Abdallah Hicham | Promoteur |
| M. Boudaoud Abderrezek | Encadreur |

**Promotion** 2015 / 2016

# *Remerciements*

*Nous tenons à remercier en premier lieu et exprimer notre gratitude à notre Dieu 'Allah'.*

*Nous ne pourrions terminer ces remerciements sans une pensée à l'ensemble de nos enseignants ainsi qu'à nos collègues avec qui nous avons partagé d'agréables moments durant cette formation de Master à l'université de Saad Dahlab Blida.*

## الملخص :

مع التطورات الحديثة في مجال التحاليل التي تعتمد على البيانات، والقدرات المحسنة الناتجة في العمل مع مجموعات البيانات الضخمة، أصبح التخطيط الاستراتيجي أكثر تعقيدا بالنسبة للوحدات التجارية، وفيما بعد لوظيفة الموارد البشرية. وقد اعتمدت معظم وحدات الأعمال على التحليلات التنبؤية لتوجيه عمليات صنع القرار ووضع الاستراتيجيات الخاصة بهم. الفرص الجديدة التي تتيحها التحليلات التنبؤية تنطبق على جميع عمليات الموارد البشرية الأساسية مثل اكتساب الموهبة، إدارة مخاطر الاستنزاف، تحليل مشاعر الموظف، والقدرة على التخطيط.

ويركز هذا الموجز على كيفية مساعدة التحليل لقادة الموارد البشرية لمناقشة المشاكل المتأصلة في هذه العمليات .كما يبرز كيف أن بعض التحديات الرئيسية في مجال الموارد البشرية يمكن معالجته باستخدام التحليلات التنبؤية . التلاعب بأدوات البيانات الكبيرة، وهي منصة Hadoop، سوف يكون متاح.

## Résumé:

Avec les récents progrès dans l'analyse axée sur les données et les capacités améliorées résultantes dans le travail avec d'énormes ensembles de données, la planification stratégique est devenue plus complexe pour les entreprises, et par la suite pour la fonction des ressources humaines. La plupart des entreprises ont déjà adopté l'analyse prédictive pour guider leurs processus de prise de décision et de développement de la stratégie. Les nouvelles possibilités offertes par l'analyse prédictive sont applicables à tous les processus de ressource humaine de base tels que l'acquisition de talents, la gestion des risques d'attrition, l'analyse des sentiments des employés, et la planification des capacités.

A travers ce document, on s'intéresse à reproduire le comportement d'un agent recruteur de GTP. Pour ce faire, on a opté pour une prédiction par régression linéaire tout en utilisant les technologies Hadoop et Hive. Les résultats ont montrés que le système mime le comportement de l'agent recruteur de façon presque identique.



**Mots Clés : Big Data, Analyse prédictive, Hadoop, Hive, Regression Linéaire**



## Abstract:

With recent advances in data-driven analytics, and the resultant improved capabilities in working with huge datasets, strategic planning has become more complex for business units, and subsequently for the human resource function. Most business units have already adopted predictive analytics to guide their decision-making and strategy development processes. The new opportunities offered by predictive analytics are applicable to all core human resource

processes such as talent acquisition, attrition risk management, employee sentiment analysis, and capacity planning.

This paper emphasizes how analytics can help human resource leaders scrutinize the problems inherent to these processes. It also highlights how some key human resource challenges can be addressed by using predictive analytics. A manipulation from big data tools, namely the Hadoop platform, will be available.

## Mots clés:

Big Data, Analyse prédictive, Hadoop, Optimisation, Recrutement.

# Liste des figures

# Liste des tableaux

# Tables des matières

# Introduction générale

# Introduction Générale

La croissance exponentielle des volumes de données entraînent des tensions sur les systèmes d'information. D'où la nécessité de revoir les choses d'une nouvelle façon. Le plus grand défi des décideurs est la maximisation du profit, et pour atteindre ct objectif il faut confronter les différents obstacles au sein de l'entreprise. Les décideurs jouent essentiellement sur les deux contraintes ; temps et couts. La minimisation de ces derniers mène au succès.

L'actualité connait un changement dans les relations humaines. Les réseaux sociaux ont participé à la naissance de la virtualisation qui construit un lien direct entre les décideurs et leurs clients, d'où la multiplicité des informations collectées. Cette multiplicité nous mène à un nouveau concept appelé 'big data'.

La bonne exploitation de l'information joue un rôle précieux dans la durée de vie d'une entreprise, et pour y arriver il faut savoir analyser les données et extraire celles qui aident à la décision.

Les outils classiques ne font pas l'objet de l'analyse dans le concept 'big data', d'où la nécessité de trouver des nouvelles solutions.

Ce document met l'accent sur la façon dont l'analyse peut aider les dirigeants de ressources humaines pour examiner les problèmes inhérents à ces processus. Il met également en évidence la façon dont certains défis clés en matière de ressources humaines peuvent être traités à l'aide de l'analyse prédictive. Une manipulation de certains outils de big data, à savoir la plateforme Hadoop, sera mise à disposition.

Ce mémoire est constitué de deux parties :

Une partie théorique qui contient deux chapitres. Le premiers chapitre englobe tout ce qui est big data ; les différentes définitions, les caractéristiques et un peu d'actualité vis a vis l'entreprise et l'aide à la décision. Le deuxième chapitre est consacré pour l'analyse prédictive ; définition, les méthodes existées, ainsi que l'intégration de l'analyse prédictive dans le big data.

Et une partie pratique qui contient notre objectif du travail, l'étude de l'existant, une méthode d'analyse proposée, l'architecture de notre système, l'environnement du travail y compris les outils du big data utilisés, et enfin des tests qui ont été faits sur des données réelles afin de présélectionner des candidats, qui ont postulé pour un poste de travail, après avoir étudié le comportement du recruteur par une analyse prédictive sur les récents recrutements. Et tout cela pour appliquer ces résultats sur les cas futurs.

# Chapitre 1: Big Data

## 1. Introduction :

L'information est aujourd'hui l'élément le plus important dans la vie des gens en général et des décideurs en particulier, elle vaut de l'or, surtout quand elle est bien exploitée. Qui dit information dit donnée, cette dernière est un composant essentiel de la première, le deuxième composant indispensable est le sens.

L'apparition des réseaux sociaux a changé le monde, par laquelle est née la virtualisation, notamment les marchés virtuels. Le partage d'informations est donc devenu très fréquents. La multiplicité de sources due à la disponibilité et la facilité d'accès à l'internet a contribué à multiplier la quantité des données numériques, à tel point que 90% des données dans le monde ont été créées au cours des deux dernières années seulement, qui a obligé les chercheurs à trouver de nouvelles manières de se comporter envers le monde. On parle ici de la recherche, l'analyse, le stockage et le partage de grands ensembles de données. Ce qui fait apparaitre un nouveau concept qui est le « **Big Data** ». Cette appellation est apparue en octobre 1997.

Dans ce chapitre nous allons définir le concept du big data, afin de clarifier et de simplifier l'environnement de notre travail, ses caractéristiques, l'actualité qui lui concerne, ainsi que son impacts et ses objectifs dans le domaine de l'entreprise et l'aide à la décision.

## 2. Définitions du Big Data :

Multiple définitions existent, cependant, aucune définition précise ou universelle ne peut être donnée au Big Data. Etant un objet complexe polymorphe, sa définition varie selon les communautés. Deux définitions, les plus pertinentes selon nous, ont été mentionnées ci après.

### 2.1. Définition 1 :

Le Big data, littéralement les « grosses données », ou méga-données, parfois appelé données massives, désigne des ensembles de données qui deviennent tellement volumineux qu'ils en deviennent difficiles à travailler avec des outils classiques de gestion de base de données ou de gestion de l'information.

L'explosion quantitative (et souvent redondante) de la donnée numérique contraint à de nouvelles manières de voir et analyser le monde. De nouveaux ordres de grandeur concernent la capture, le stockage, la recherche, le partage, l'analyse et la visualisation des données. Les perspectives du traitement des *Big data* sont énormes et en partie encore insoupçonnées.

Certains supposent que le Big data pourrait aider les entreprises à réduire leurs risques et faciliter la prise de décision, ou créer la différence grâce à l'analyse prédictive et une « expérience client » plus personnalisée et contextualisée.

Divers experts, grandes institutions (comme le MIT 'Massachusetts Institute of Technology' aux États-Unis), administrations et spécialistes sur le terrain des technologies ou des usages considèrent le phénomène Big data comme l'un des grands défis informatiques de la décennie 2010-2020 et en ont fait une de leurs nouvelles priorités de recherche et développement.[1]

## 2.2. Définition 2:

Les Big Data ou méga-données désignent l'ensemble des données numériques produites par l'utilisation des nouvelles technologies à des fins personnelles ou professionnelles.

Cela recoupe les données d'entreprise (courriels, documents, bases de données, historiques de processeurs métiers…) aussi bien que des données issues de capteurs, des contenus publiés sur le web (images, vidéos, sons, textes), des transactions de commerce électronique, des échanges sur les réseaux sociaux, des données transmises par les objets connectés (étiquettes électroniques, compteurs intelligents, smartphones,…etc.). [2]

## 3. Caractéristiques du BIG DATA:

Le Big Data couvre cinq dimensions appelées « les 5 V» : volume, vélocité, variété, véracité et valeur.

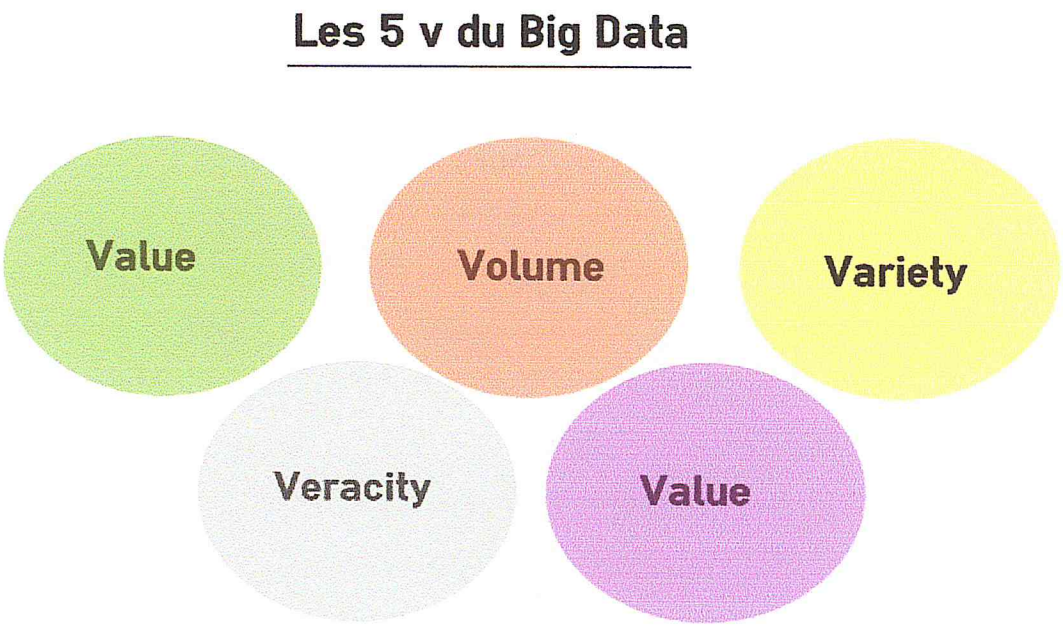



Figure 1 : Les caractéristiques du big data.

## 3.1. Volume :

Les entreprises font face à une augmentation exponentielle des données (Jusqu'à plusieurs milliers de téraoctets):

- Logs,
- Réseaux sociaux,
- e-commerce,
- Catalogue produit,
- Analyse des données,
- Monitoring,...

Les technologies traditionnelles (Business Intelligence, Bases de données) n'ont pas été pensées pour de telles volumétries.

Une des caractéristiques du Big Data est sa capacité à traiter d'énormes quantités de données pour un coût mesuré.

### 3.2. Variété/variabilité :

Les données à traiter dans une entreprise sont de natures multiples.

-Exemple de données structurées:

- Flux RSS 'Rich Site Summary', XML 'Extensible Markup Language'.
- JSON 'JavaScript Object Notation'.
- Bases de données.

Des données non structurées peuvent s'ajouter :

- Mails.
- Pages web.
- MultiMedia (son, image, vidéo, …etc.).

Ces données non structurées peuvent faire l'objet d'une analyse sémantique permettant de mieux les structurer et les classer, entraînant une augmentation du volume de données à stocker.

La solution doit être évolutive car les formats de données ne sont pas tous actuellement connus (voir par exemple comment le format JSON a supplanté XML très rapidement).

### 3.3. Vélocité :

Dans certains cas, l'accès et le partage des données doivent se faire en temps réel (on verra par la suite que ce n'est pas toujours vrai pour Hadoop).

Toutes les entreprises n'ont pas la même échelle des temps entre traitements batch et traitements en temps réel : millisecondes, secondes, minutes, ...etc.

La vitesse de traitement de certaines solutions permet d'offrir des capacités, d'analyse et de traitements, en temps réel et donc un retour utilisateur plus rapide.

### 3.4. Valeur :

C'est un point essentiel du Big Data car il va permettre de monétiser les données d'une entreprise. Ce point n'est pas une notion technique mais économique. On va mesurer le ROI 'Return On Investment' de la mise en œuvre du Big Data et sa capacité à s'auto financer par les gains attendus pour l'entreprise.

### 3.5. Véracité :

C'est la capacité à disposer des données fiables pour le traitement. On va s'intéresser à la provenance des données afin de déterminer s'il s'agit de données de confiance.

En fonction du critère de confiance, on accordera plus ou moins d'importance à la donnée dans les chaînes de traitement. Le critère de confiance ne mesure pas uniquement la méfiance vis à vis de la source mais surtout l'importance que l'on souhaite lui donner.

Toutefois, parmi les données dont il faut éventuellement se méfier on trouve les données des réseaux sociaux dont la provenance et l'objectivité est difficile à évaluer.

## 4. Les objectifs du Big Data :

Cette analyse permet:

- de comprendre les besoins des individus et les contraintes des usagers ;
- d'adapter les infrastructures, réseaux et services (notamment services publics) en fonction de leur utilisation ;
- d'assister la prise de décision des différents acteurs économiques (entreprises, administration) ;
- d'analyser et d'anticiper les comportements des consommateurs (analyse prédictive) ;
- de faciliter l'évaluation des services ;
- d'améliorer l'utilisation des machines et appareils (amélioration des performances, prévention des pannes et maintenance).

## 5. La technologie mise en œuvre sous le Big Data :

Les créations technologiques qui ont facilité la venue et la croissance du Big Data peuvent globalement être catégorisées en deux familles : d'une part, les technologies de stockage, portées particulièrement par le déploiement du Cloud Computing. D'autre part, l'arrivée de technologies de traitement ajustées, spécialement le développement de nouvelles bases de données adaptées aux données non-structurées (Hadoop) et la mise au point de modes de calcul à haute performance (MapReduce).

Il existe plusieurs solutions qui peuvent entrer en jeu pour optimiser les temps de traitement sur des bases de données géantes à savoir les bases de données NoSQL (Not only Structured

Query Language) (comme MongoDB, Cassandra ou Redis), les infrastructures du serveur pour la distribution des traitements sur les nœuds et le stockage des données en mémoire.

La première solution permet d'implémenter les systèmes de stockage considérés comme plus performants que le traditionnel SQL (Structured Query Language) pour l'analyse de données en masse (orienté clé/valeur, document, colonne ou graphe).

La deuxième est aussi appelée le traitement massivement parallèle. Le Framework Hadoop en est un exemple. Celui-ci combine le système de fichiers distribué HDFS 'Hadoop Distributed File System', la base NoSQL HBase et l'algorithme MapReduce.

Quant à la dernière solution, elle accélère le temps de traitement des requêtes.

En outre, les nouveaux challenges auxquels sont confrontées les entreprises sont la vitesse et le coût du traitement de ces données. En plus de ces challenges techniques, il faut ajouter une autre plus value essentielle qui est la valorisation de ces données. C'est cette caractéristique économique qui va justifier la mise en œuvre des technologies du Big Data, elle mesure la plus value pour l'entreprise et le retour sur investissement.

Rapidement il a été constaté que les solutions traditionnelles ne pourraient pas convenir (en tout cas pour un coût mesuré). Il a donc fallu inventer de nouveaux systèmes.

Le fait d'abandonner le mode relationnel des bases de données traditionnelles, pour un mode clé/valeur ou colonnes, a permis de diviser la volumétrie et les temps de traitements.

- **Map Reduce :**

  Au départ, il y eut "Map Reduce", une méthode et une technologie de traitement massivement parallèle issues des laboratoires Google Corp ® avec gestion de la tolérance aux pannes et système de gestion de fichiers spécifiques (Google File System). On parle ici de traitement sur des milliers de machines réparties en grappes (clusters).

- **Hadoop :**

Ensuite, il eut "Hadoop", un framework mis au point par l'Apache Software Fundation afin de mieux généraliser l'usage du stockage et traitement massivement parallèle de Map Reduce et de Google File System. Bien entendu, Hadoop possède ses limites. Quoi qu'il en soit, c'est une solution de Big Data très largement utilisée pour effectuer des analyses sur de très grands nombres de données.

- **Bases No SQL :**

Les bases de données relationnelles ont une philosophie d'organisation des données bien spécifiques, avec notamment le langage d'interrogation SQL, le principe d'intégrité des transactions ACID (atomicité, cohérence, isolation et durabilité), et les lois de normalisation. Bien utiles pour gérer les données qualifiées de l'entreprise, elles ne sont pas du tout adaptées au stockage de très grandes dimensions et au traitement ultra rapide. Les bases NoSQL autorisent la redondance pour mieux servir les besoins en matière de flexibilité, de tolérance aux pannes et d'évolutivité.

- **Stockage "In-Memory" :**

Pour des analyses encore plus rapides, les traitements directement en mémoire sont une solution. Une technologie bien qu'encore trop coûteuse pour être généralisée.

- **Cloud Computing :**

Le Big Data exige une capacité matérielle hors du commun, que ce soit pour le stockage comme pour les ressources processeurs nécessaires au traitement. Nul besoin de s'équiper outre mesure, le "Cloud" est là pour cela. Encore faut-il avoir bien compris le concept pour différencier, le cloud privé du cloud public, l'interne de l'externe et les hybrides combinant plusieurs types de solutions. Ensuite il est aussi prudent de différencier les niveaux de services de chacune des solutions : IAAS 'Infrastructure As A Service', PAAS 'Platform As A Service', SAAS 'Software As A Service'... [3]

## 5.1. HADOOP KERNEL :

Apache Hadoop est un framework qui va permettre le traitement de données massives sur un cluster allant d'une à plusieurs centaines de machines.

Apache Hadoop est un projet open source (Apache v2 licence).

Hadoop est écrit en Java et a été créé par Doug Cutting et Michael Cafarella en 2005 (après avoir créé le moteur de recherche Lucene, Doug travaillait alors pour Yahoo sur son projet de crawler web Nutch).

Apache Hadoop va gérer la distribution des données au cœur des machines du cluster, leurs éventuelles défaillances mais aussi l'agrégation du traitement final.

L'architecture est de type « Share nothing » : aucune donnée n'est traitée par deux nœuds différents même si les données sont réparties sur plusieurs nœuds (principe d'un nœud primaire et de nœuds secondaires).

Apache Hadoop est composé de quatre éléments :

- **Hadoop Common** : Ensemble d'utilitaires communs aux autres éléments.
- **Hadoop Distributed File System (HDFS)** : Un système de fichiers distribué pour le stockage persistant des données.
- **Hadoop YARN** : Un framework de gestion des ressources et de planification des traitements.
- **Hadoop MapReduce v2** : Un framework de traitements distribués basé sur YARN. [4]

### 5.1.1. HDFS (Hadoop Distributed File System):

HDFS est un système de fichiers Java utilisé pour stocker des données structurées ou non sur un ensemble de serveurs distribués.

C'est un système distribué, extensible et portable développé par le créateur d'Hadoop à partir du système développé par Google (GoogleFS).

Écrit en Java, il a été conçu pour stocker de très gros volumes de données sur un grand nombre de machines équipées de disques durs standards.

HDFS s'appuie sur le système de fichiers natif de l'OS 'Operation System' pour présenter un système de stockage unifié reposant sur un ensemble de disques et de systèmes de fichiers hétérogènes.

Un cluster HDFS repose sur deux types de composants majeurs :

**1. NameNode :** Ce composant gère les fichiers et les répertoires du cluster de manière centralisée.

Il est unique mais dispose d'une instance de backup afin d'assurer la continuité du fonctionnement du cluster Hadoop en cas de panne.

**2. DataNode (nœud de données) :** Ce composant stocke et restitue les blocs de données (données primaires) et abrite des copies des autres instances.

Par défaut, les données sont stockées sur trois nœuds différents : dans deux nœuds proches (même machine ou rack) et l'autre sur un nœud plus distant.

Le RAID 'Redundant Arrays of Inexpensive Disks' est par conséquent inutile sur un cluster HDFS.

La consistance des données est basée sur la redondance. Une donnée est stockée sur au moins X volumes différents.

**a.** Node (Master/slave) : Dans une architecture Hadoop chaque membre pouvant traiter des données est appelé Node (Nœud).

**b.** Un seul d'entre eux peut être master même s'il peut changer au cours de la vie du cluster, il s'agit du NameNode.

**c.** Le NameNode est responsable de la localisation des données dans le cluster.

**d.** Le NameNode est donc un SPOF 'Single Point Of Failure' dans un cluster Hadoop. C'est pourquoi il est conseillé d'en démarrer au moins deux.

**e.** Depuis Hadoop 2.0, la "promotion" est automatique en cas de défaillance du NameNode principal.

**f.** Les autres nœuds, stockant les données sont des slaves appelés DataNode.

**3. Rôle des NameNode et des DataNode :** Au sein du cluster, les données sont découpées et distribuées en blocs selon les deux paramètres suivants :

- Blocksize : Taille unitaire de stockage (généralement 64 Mo ou 128 Mo). C'est à dire qu'un fichier de 1 Go (et une taille de bloc de 128 Mo) sera divisé en 8 blocs.

- Replication factor : C'est le nombre de copies d'une donnée devant être réparties sur les différents nœuds du cluster (souvent 3, c'est à dire un primaire et deux secondaires).

**4. Format de stockage :** Le format de stockage est au cœur du système Hadoop car il s'agit du format de sérialisation des données.

Il est utilisé par HDFS pour le stockage des données mais aussi par MapReduce comme format d'échange entre les nœuds.

Les formats supportés sont :

- Format texte de type CSV,
- JSON,
- Binaire (Hadoop Serialization),
- Binaire (Avro, Thrift, Parquet, ...etc.).

Les formats texte et JSON vont présenter l'avantage d'être compréhensibles par des humains mais l'inconvénient d'être peu optimisés pour le stockage et les performances.

Les formats binaires vont au contraire présenter l'avantage d'être plus performants.

### 5.1.2. MapReduce :

C'est un framework de traitements parallélisés, créé par Google pour son moteur de recherche web. Il permet la décomposition d'une requête importante en un ensemble de requêtes plus petites qui vont produire chacune un sous ensemble du résultat final : c'est la fonction Map.

L'ensemble des résultats est traité (agrégation, filtre) : c'est la fonction Reduce.

**1. Les acteurs :** Dans un traitement MapReduce, différents acteurs vont intervenir :

- Workers : Liste de nœuds Hadoop capables de traiter des tâches MapReduce.
- Master : Un worker dédié à la gestion des tâches.

- Client : Lance le traitement MapReduce (souvent nommé driver).

**2. Les différentes phases :**

**a. Initialisation :** Le client/driver charge un/des fichiers dans HDFS et soumet un traitement MapReduce à la grille.

**b. Split :** Les données en entrée sont éventuellement divisées en blocs (16-64Mo).

**c. Affectation :** Le master affecte les tâches (Map et Reduce) aux workers. La configuration définit le nombre de tâches de type Map et Reduce supportées par chacun des nœuds.

**d. Map :** Lecture des splits qui sont transmis à la fonction Map. Les ensembles clé/valeur produits par la fonction sont d'abord stockés en mémoire avant d'être périodiquement écrits localement (pas sur HDFS).

**e. Shuffle :** Les résultats des fonctions Map sont agrégés par la valeur de la clé pour produire une liste de valeurs traitées par le Reducer.

**f. Reduce :** Le master distribue au Reducer la liste des données à traiter. Les résultats sont envoyés au flux de sortie (HDFS, web services, …).

**g. Combiner :** Optimisation, utilise les résultats intermédiaires du Map en entrée pour un traitement qui est généralement équivalent au Reducer (pas de garantie de passage).

**h. Fin :** Le master redonne la main au programme client.

### 5.1.3. Points différenciant :

**1. Vitesse de traitement :**

NoSQL est plus adapté pour des traitements rapides de type temps réel ou des traitements interactifs.

Hadoop intègre une base NoSQL (HBase) qui est souvent utilisée pour ses capacités d'analyse. Le socle Hadoop étant utilisé pour le stockage et la transformation de données.

**2. Framework de traitements :**

Les solutions NoSQL offrent peu ou pas de capacités de traitements natives à la plateforme (MapReduce est parfois possible ainsi que des fonctions d'agrégations) mais on est loin de la variété offerte par la plateforme Hadoop.

En clair NoSQL est une technologie de stockage alors qu'Hadoop est une solution de stockage et de traitement.

**3. Capacités de stockage :**

Même si NoSQL peut stocker des volumétries très importantes, Hadoop conserve un avantage car il dispose d'un système spécifiquement conçu pour ces volumes (HDFS).

De plus, sa scalabilité est linéaire, ce qui n'est pas le cas de toutes les solutions NoSQL dont certaines ont une architecture centralisée.

En réalité, ces deux technologies sont complémentaires et l'on va souvent les trouver associées.

- NoSQL pour les traitements temps réel.
- Hadoop pour les traitements batch et le stockage.
- Eventuellement une consolidation entre les deux systèmes (architecture lambda).

## 5.2. LES EXTENSIONS :

Autour du cœur d'Hadoop, des solutions sont nées afin de couvrir les besoins non pris en compte ou bien pour faire le lien avec les systèmes existants.

Aujourd'hui c'est un écosystème complet qui est proposé, capable de répondre à toutes les problématiques d'une entreprise.

### 5.2.1. Requêtage des données : Hive (Facebook)

Hive est à l'origine un projet Facebook qui permet de faire le lien entre le monde SQL et Hadoop.

Il permet l'exécution de requêtes de type SQL sur un cluster Hadoop en vue d'analyser et d'agréger les données.

Le langage SQL utilisé est nommé HiveQL 'Hive Query Language'. C'est un langage de visualisation.

Dans certains cas, les développeurs doivent faire le mapping entre les structures de données et Hive.

Hive utilise un connecteur JDBC/ODBC.

Il existe une phase de transformation qui va transcrire les requêtes HiveQL en traitements MapReduce ou Tez (framework MapReduce de nouvelle génération par Hortonworks) dans les versions plus récentes.

Afin d'améliorer les performances, il existe des projets de migration de MapReduce vers Spark (présent en version bêta depuis Hive 1.1).

### 5.2.2. Scripting sur les données : Pig (Yahoo)

Pig est à l'origine un projet Yahoo qui permet le requêtage des données Hadoop à partir d'un langage de script.

Contrairement à Hive, Pig est basé sur un langage de haut niveau, PigLatin, qui permet de créer des programmes de type MapReduce ou Tez. Ainsi qu'il ne dispose pas d'interface web.

Afin d'améliorer les performances, il existe des projets de migration afin d'utiliser Spark à la place de MapReduce/Tez.

Ces projets sont directement intégrés au projet Pig mais sont encore en cours de finalisation.

### 5.2.3. Intégration SGBD-R : Sqoop (Cloudera)

Sqoop permet le transfert des données entre un cluster Hadoop et des bases de données relationnelles. C'est un produit développé par Cloudera.

Il permet d'importer/exporter des données depuis une base SQL vers Hadoop (et inversement).

Pour la manipulation des données Sqoop utilise MapReduce et des drivers JDBC. La version Sqoop v2 (2013) apporte quelques améliorations :

- Meilleure utilisation du paradigme MapReduce (avec Sqoop v1 seuls les traitements Map étaient utilisés).
- Meilleure sécurité.

- Déploiement : Sqoop v1 nécessitait de déployer Sqoop ainsi que les drivers sur le client, avec Sqoop2 l'approche est différente puisque tout est installé sur le serveur Sqoop. [6]

## 6. Les applications concrètes du Big Data :

Beaucoup estiment que le Big Data n'est qu'un passage à l'échelle des traitements traditionnels alors que tous les secteurs sont concernés.

Dans la distribution et les télécoms : Le Big Data permet de bien connaître les clients à la fois par leur comportement en boutique, mais aussi en analysant leur activité sur internet, y compris sur les réseaux sociaux. Anticiper leurs besoins pour cibler des offres personnalisées est devenu le «must do» du marketing tiré par les données.

Dans le domaine de la santé, par exemple, le Big Data favorise une médecine préventive et personnalisée. Ainsi, l'analyse des recherches des internautes sur un moteur de recherche a déjà permis de détecter plus rapidement l'arrivée d'une épidémie de grippe. Dans un futur proche, les appareils connectés devraient permettre l'analyse en continu des données biométriques des patients.

Dans le domaine des transports, l'analyse des données du Big Data (données provenant des pass de transport en commun, géolocalisation des personnes et des voitures, etc.) permet de modéliser les déplacements des populations afin d'adapter les infrastructures et les services (horaires et fréquence des trains, par exemple).

De la même manière, l'analyse des données provenant de capteurs sur les avions (données de vol) associées à des données météo permet de modifier les couloirs aériens afin de réaliser des économies de carburant et d'améliorer la conception et la maintenance des avions.

Il existe des utilisations concrètes du Big Data dans de nombreux autres domaines : sciences, développement durable, commerce, éducation, loisirs, sécurité, …etc.

## 7. Le Big data et l'aide à la décision dans l'entreprise :

### 7.1. Les enjeux du Big Data :

Par son ampleur et par ses nombreuses promesses le BIG DATA a rapidement attiré l'attention des entreprises grâce à ses enjeux.

Le Big Data permet de valoriser les Péta-octets de données, ou à explorer la valeur cachée dans l'immensité du contenu non structuré comme les fichiers, les emails, ou les pages web.

- **En marketing :**

C'est tout le secteur qui se trouve renouvelé. Le Big Data permet en effet aux professionnels du secteur de connaître leur client « à 360° », c'est-à-dire à la fois par son parcours internet mais également par ses achats en magasin ou ses préférences affichées sur les réseaux sociaux. Anticiper les besoins de celui-ci et cibler des offres personnalisées ou encore l'analyse de sentiment pour la détection de comportements sur les réseaux sociaux. Le marketing se fait de plus en plus prédictif avec le Big Data, et l'on assiste à une éclosion de nouveaux modèles statistiques davantage inductifs.

- **Dans le domaine du pilotage de l'entreprise :**

Sur cet aspect, les usages sont également nombreux et porteurs d'innovation. En assurant une circulation immédiatement généralisée de l'information sur l'activité, le Big Data laisse entrevoir une optimisation complète des processus et des ressources métiers. Il réduit de facto le temps de réaction face à des erreurs ou des pannes et permet d'ajuster en permanence les équilibres offre-demande et temps-ressource. C'est une promesse importante dans des secteurs comme ceux de l'énergie ou des transports qui sont constamment portés par la logique de flux. Outre une réduction importante des coûts, le Big Data permet ici d'identifier au plus près les moteurs de l'activité, ce qui n'était pas possible avec les indicateurs traditionnels, soumis à des délais de latence bien plus importants.

- **Pour la Recherche :**

Domaine d'application originel du Big Data, l'apport de celui-ci est assez évident. En autorisant le traitement de multitudes de données, le Big Data permet à la science de réaliser des avancées importantes, lorsqu'il s'agit d'explorer l'infiniment petit (ex : exploration

géologique), de croiser des données complexes (ex : imagerie) ou d'effectuer des simulations (ex : domaine spatial). C'est d'ailleurs en génétique que le Big Data a fait ses premières armes car ce secteur réclamait une approche à la fois quantitative et qualitative avancée.

- **Dans le domaine de l'Information :**

Le traitement des Big Data a profondément modifié la donne. Pour une requête donnée, il est désormais possible d'accéder à un croisement d'informations très disparates, issues de sources jusque-là négligées. L'instantanéité des réseaux sociaux est à ce titre une innovation de taille. L'analyse des tweets est devenue une source de renseignements courante pour comprendre les comportements ou les goûts de populations segmentées. De plus, au-delà de la compréhension de phénomènes, ne pas s'intéresser au BIG DATA aujourd'hui, c'est peut être risqué demain de perdre en compétence et d'être en retard sur son marché. Le BIG DATA comme toute avancée technologique, peut comporter des risques, qu'il ne faut surtout pas ignorer. En effet, le BIG DATA repose sur la confiance du consommateur et toute rupture dans cette confiance entrainerait automatiquement un retour en arrière. De la même façon, on craint que le Cloud ne soit pas assez protecteur. Il est donc urgent de maitriser ces risques pour garder la confiance des consommateurs. Cela nécessite d'avoir les compétences nécessaires par le recrutement de personnel qualifié. [5]

## 7.2. Modèles d'organisation « Big Data » dans l'entreprise:

Les modèles d'organisation privilégiés par l'entreprise sont soit centraliser les données ou bien disposer d'architectures réparties au sein des directions métiers.

Trois modes d'organisation sont envisageables :

- **Une option « centralisée » :** Dans laquelle toutes les compétences sont regroupées au sein d'une entité transverse, sorte de Centre de Services Big Data au service des Métiers. En centralisant les ressources, on mutualise les coûts et on évite a priori la duplication des efforts, des données, et des budgets.
- **Une vision « décentralisée »:** Où ce sont les Métiers qui gardent la main en gérant leurs projets, leurs compétences, pour satisfaire au plus près leurs objectifs.
- **Une vision « externalisée » :** Dans laquelle l'entreprise confie à un prestataire spécialisé la gestion des données et des traitements associés.

## 8. Les news du Big Data:

Croissance des volumes des données produites et collectées, baisse des coûts de stockage, éclosion d'outils puissants d'analyse de données hétérogènes ; tout semble prêt désormais pour que le Big Data produise des effets bénéfiques en entreprise. En ligne de mire, l'accès en temps réel aux informations contenues dans d'immenses bases de données.

Google et Facebook ont très vite été confrontées à l'afflux d'information et se sont posées des questions sur la meilleure manière de le gérer. C'est certainement pour cette raison qu'ils sont deux initiateurs des technologies qui structurent aujourd'hui le marché du Big Data. Et en conséquence, ils sont en tête des entreprises qui savent traiter d'importants volumes de données en temps réel (pour cela les spécialistes de la question utilisent la technologie de Base de données In Memory), données en provenance sources multiples (non-structurées type NoSQL, ou structurées en base de données classiques (type SQL).

Reste qu'un traitement si massif de données ne se conçoit qu'à l'aide d'une infrastructure adéquate, et souvent dédiée. Les géants du web utilisent donc des fermes de serveurs à l'architecture massivement parallèle, créant des centaines, voire milliers de nœuds de calcul. Hadoop est une des architectures les plus connues (elle est open source) dans ce domaine. [7]

## 9. L'avenir du Big Data :

Etant une tendance lourde, le Big Data n'est pas une mode. Dans le domaine de l'usage, il satisfait une nécessité de travailler la donnée plus profondément, pour créer de la valeur, conjointement à des aptitudes technologiques qui n'existaient pas dans le passé. Cependant, compte tenu de l'évolution des technologies qui ne semble pas vouloir s'estomper, on ne peut pas alors parler d'une norme véritable ou de standards dans le domaine du Big data.

Beaucoup d'applications du Big Data n'en sont qu'à leurs préludes et on peut s'attendre à voir apparaître des utilisations auxquelles on ne s'attend pas encore aujourd'hui. En quelque sorte, le Big Data est un tournant pour les organisations au moins aussi important qu'internet en son temps. Chaque entreprise doit donc s'y mettre dès maintenant. Dans le cas contraire, il y a un risque qu'elles se rendent comptent d'ici quelques années qu'elles se sont faites dépasser par la concurrence. [8]

## 10. Conclusion :

Nous avons abordé dans ce chapitre l'historique du Big data, son impact sur les entreprises, ses enjeux, et le changement dû à son arrivée. C'est la formalisation de l'évolution des volumes, de la vitesse et de la variété des données, qui crée de la valeur ajoutée.

Jusqu'ici, il n'existe pas encore sur le marché un logiciel Big Data prêt à l'emploi que l'on puisse installer dans une entreprise. Le Big Data est avant tout une démarche stratégique, il faut penser à la stratégie pour donner de la valeur aux données. C'est grâce à cette stratégie que le Big Data fonctionnera à l'intérieur d'une entreprise.

# Chapitre 2 :
# L'analyse prédictive

## 1. Introduction :

Ce deuxième chapitre est, comme le précédent, purement théorique. Il nous présente le concept 'Analyse prédictive'. Deux définitions sont intégrées pour avoir une idée sur ce concept. Nous trouverons ensuite les champs d'application, les différentes méthodes existantes, ainsi que les étapes d'analyse. En fin nous essayons de mettre le lien entre l'analyse prédictive et le big data. Tout cela pour faciliter la compréhension de la méthode utilisée dans notre projet et enrichir les connaissances dans ce domaine.

## 2. Définitions :

### 2.1. Définition 1 :

L'analyse prédictive, considérée comme un type d'exploration de données, est un domaine de l'analyse statistique qui extrait l'information à partir des données pour prédire les tendances futures et les motifs de comportement. Le cœur de l'analyse prédictive se fonde sur la capture des relations entre les variables explicatives et les variables expliquées, ou prédites, issues des occurrences passées, et l'exploitation de ces relations pour prédire les résultats futurs. Il est important de noter, toutefois, que l'exactitude et l'utilité des résultats dépendent grandement du niveau de l'analyse des données et de la qualité des hypothèses. L'analyse prédictive s'occupe exclusivement de la Nécessité, pas du Hasard (Le Hasard et la Nécessité_ J Monod), du "Prédictive future" que J. Derrida appelle "Futur" par opposition à "l'Avenir" que l'on ne peut prédire, car Avenir = Nécessité + Hasard. [9]

### 2.2. Définition2 :

L'analyse prédictive est l'analyse des données historiques et actuelles disponibles sur le client afin de créer des prévisions sur ses comportements, préférences et besoins futurs.
L'analyse prédictive débouche le plus souvent sur la création de scores liés à la probabilité qu'un client ou prospect réalise une action donnée (achat, résiliation, réponse, etc..).
L'analyse prédictive permet d'optimiser le ciblage et le déroulement (moment, canal, offre,..) des actions marketing. [10]

## 3. Champs d'applications :

Bien que l'analyse prédictive puisse être utilisée dans un grand nombre d'applications, quelques exemples où l'analyse prédictive a montré un impact décisif dans les années passées sont présentés ici.

### 3.1. Systèmes d'aide à la décision clinique :

Les experts utilisent l'analyse prédictive dans le domaine de la santé principalement pour déterminer quels sont les patients susceptibles de développer des maladies telles que le diabète, l'asthme, les maladies cardiaques, et d'autres affections potentiellement dangereuses. De plus, les systèmes d'aide à la décision clinique incorporent l'analyse prédictive pour soutenir les décisions médicales. Une définition a été proposée par le Docteur Robert Hayward du Centre des Évidences de Santé : « Les systèmes d'aide à la décision clinique font le lien entre les observations et la connaissance clinique pour influencer les choix des cliniciens afin d'améliorer les services médicaux ».

### 3.2. Recouvrement financier :

Chaque portefeuille contient en son sein un ensemble de clients à risque qui ne remplissent pas leurs obligations à temps. L'institution financière doit entreprendre des actions de recouvrement pour encaisser les sommes dues. Un grand nombre de ressources est gaspillé pour des clients dont les sommes dues sont difficiles voire impossibles à recouvrir. L'analyse prédictive peut aider à optimiser les sommes allouées au recouvrement en identifiant les agences les plus efficaces, les stratégies de contact, les actions judiciaires et autres pour chaque client, afin d'augmenter le taux de recouvrement tout en réduisant les coûts.

### 3.3. Souscription :

Beaucoup de métiers ont à tenir compte de leur exposition aux risques en référence aux services qu'ils offrent et doivent déterminer le coût nécessaire à la couverture des risques. Par exemple les fournisseurs d'assurances automobiles ont besoin d'évaluer le montant de la prime d'assurance pour couvrir le risque couru par l'automobile et le conducteur. Une institution financière a besoin d'évaluer le potentiel et la capacité de remboursement de l'emprunteur avant l'accord de prêt. Pour un assureur santé, l'analyse prédictive peut aider à analyser les données du passé médical sur quelques années, aussi bien que toute autre information en provenance des laboratoires, pharmacies, et autres enregistrements disponibles, pour savoir le cout que l'assuré occasionnera dans le futur. L'analyse prédictive peut aider à la souscription de ces contrats en évaluant les probabilités de maladie, de défaut de paiement, de faillite, etc. L'analyse prédictive peut rationaliser le processus d'acquisition de clients, en évaluant le comportement à risque du client en utilisant les données disponibles. L'analyse prédictive, dans son volet scoring a réduit le temps d'approbation d'une demande de crédit ou

de prêt. Une analyse prédictive adéquate peut mener à des décisions de tarification adéquates qui peuvent aider à alléger les risques futurs de défaut de paiement, de remboursement, etc.

## 4. Méthodes utilisées dans l'analyse prédictive :

### 4.1. Les méthodes de régression :

#### 4.1.1. La régression linéaire :

La régression linéaire est l'une des techniques statistiques les plus utiles et l'une de celles qu'on emploie de plus en plus couramment dans le cas d'une variable dépendante continue. De plus, parce qu'on peut l'étendre au-delà des données bi-variées en l'appliquant à une situation multi-variée, la régression se révèle un outil très utile de la recherche sociale. L'analyse de régression multiple permet de construire une équation explicative d'un phénomène donné. On identifie alors les variables indépendantes les plus significatives, ce qui permet de «prédire» les comportements non mesurés directement.**Erreur ! Source du renvoi introuvable.**

#### 4.1.2. La régression logistique :

La régression logistique permet d'estimer la force de l'association entre une variable qualitative dichotomique (binaire) dépendante et des variables qualitatives ou quantitatives indépendantes. La régression logistique peut être uni-variée mais son intérêt réside dans son utilisation multi-variée. La régression logistique est un outil qui permet de mettre en relation des variables explicatives à une variable réponse dichotomique, c'est-à-dire qui ne peut prendre que deux valeurs, le cas classique étant celui d'une variable réponse (dépendante) binaire. Cette situation est fréquente dans divers champs d'application, particulièrement dans les sciences sociales.

Cette technique est utilisée pour des études ayant pour but de vérifier si des variables indépendantes peuvent prédire une variable dépendante dichotomique.

#### 4.1.3. La méthode des k-plus proches voisins : [11]

C'est une approche très simple et directe. Elle ne nécessite pas d'apprentissage mais simplement le stockage des données d'apprentissage. Son principe est le suivant. Une donnée de classe inconnue est comparée à toutes les données stockées. On choisit pour la nouvelle donnée la classe majoritaire parmi ses K plus proches voisins (Elle peut donc être lourde pour

des grandes bases de données) au sens d'une distance choisie. Afin de trouver les K plus proches d'une donnée à classer, on peut choisir la distance euclidienne**Erreur ! Source du renvoi introuvable.**. Soient deux données représentées par deux vecteurs $x_i$ et $x_j$, la distance entre ces deux données est donnée par:

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{d} (x_{ik} - x_{ik})^2}$$

### 4.1.4. Arbre de décision (Decision Trees) : [12]

Ensemble de règles de classification basant leur décision sur des tests associes aux attributs (ou classes), organisés de manière arborescente. Les éléments à classer sont composés d'attributs et leur valeur cible, Les nœuds de l'arbre peuvent être :

a) des nœuds de décision, dans ces nœuds une seule valeur d'attribut est testée pour déterminer à quelle branche de la sous-arborescence s'applique.
b) Ou nœuds feuilles qui indiquent la valeur de l'attribut cible.

Les Branches de l'arbre correspondent à une valeur d'attribut.

*« Un arbre de décision est un outil pour déterminer l'appartenance d'un objet à une classe en fonction de ses caractéristiques/attributs. »***Erreur ! Source du renvoi introuvable.**

On donne un ensemble X de N dont les éléments sont notés xi et dont les P attributs sont quantitatifs. Chaque élément de X est étiqueté, c'est-à-dire qu'il lui est associé une classe ou un attribut cible que l'on note y appartenant à Y.

A partir de ce qui précède, on construit un arbre dit « de décision » tel que :

- chaque nœud correspond à un test sur la valeur d'un ou plusieurs attributs.
- chaque branche partant d'un nœud correspond à une ou plusieurs valeurs de ce test.

Les arbres de décisions ont pour objectif la classification et la prédiction. Leur fonctionnement est basé sur un enchaînement hiérarchique de règles exprimées en langage courant.

Un arbre de décision est une structure qui permet de déduire un résultat à partir de décisions successives. Pour parcourir un arbre de décision et trouver une solution il faut partir de la racine. Chaque nœud est une décision atomique. Chaque réponse possible est prise en compte et permet de se diriger vers un des fils du nœud. De proche en proche, on descend

dans l'arbre jusqu'à tomber sur une feuille. La feuille représente la réponse qu'apporte l'arbre au cas que l'on vient de tester.

- Débuter à la racine de l'arbre
- Descendre dans l'arbre en passant par les nœuds de test
- La feuille atteinte à la fin permet de classer l'instance testée.

Très souvent on considère qu'un nœud pose une question sur une variable, la valeur de cette variable permet de savoir sur quels fils descendre. Pour les variables énumérées il est parfois possible d'avoir un fils par valeur, on peut aussi décider que plusieurs variables différentes mènent au même sous arbre. Pour les variables continues il n'est pas imaginable de créer un nœud qui aurait potentiellement un nombre de fils infini, on doit discrétiser le domaine continu (arrondis, approximation), donc décider de segmenter le domaine en sous-ensembles. Plus l'arbre est simple, et plus il semble techniquement rapide à utiliser. En fait, il est plus intéressant d'obtenir un arbre qui est adapté aux probabilités des variables à tester. La plupart du temps un arbre équilibré sera un bon résultat. Si un sous arbre ne peut mener qu'à une solution unique, alors tout ce sous-arbre peut être réduit à sa simple conclusion, cela simplifie le traitement et ne change rien au résultat final.**Erreur ! Source du renvoi introuvable.**

Il existe de nombreux algorithmes pour les arbres de décision tel que**Erreur ! Source du renvoi introuvable.** :

- Algorithme de Hunt (1966)
- CART (Classification And Regression Trees) (1984)
- ID3 (1986), C4.5 (1993)
- SLIQ, SPRINT

Nous décrivons brièvement l'algorithme Hunt :

Soit $D_t$ l'ensemble des données qui a été associé au nœud t

- Si $D_t$ contient que des nœuds appartenant à la classe $y_t$, alors le nœud t est une feuille étiquetée $y_t$ ;
- Si $D_t = \emptyset$, alors t est une feuille étiquetée par la classe de défaut $y_d$ ;
- Si $D_t$ contient des données appartenant à plus d'une classe alors :
  a) utilisez un attribut diviseur pour créer des nœuds fils à t. Ces nœuds contiendront les données de t en fonction de la valeur de l'attribut choisi.

b) appliquer les étapes précédentes aux nœuds créés.

Un exemple simple d'arbre de décision :

| Sports fan | Marital Status | Annualincome | Likes Pizza |
|---|---|---|---|
| Yes | Divorced | 90K | Yes |
| No | Single | 125K | No |
| Yes | Married | 100K | No |
| Yes | Married | 60K | No |
| Yes | Married | 75K | No |
| Yes | Single | 105K | No |
| Yes | Single | 85K | Yes |
| Yes | Single | 90K | Yes |
| No | Divorced | 220K | No |
| No | Married | 120K | No |

Tableau 2 : Attributs et attribut cible à partir d'observations.

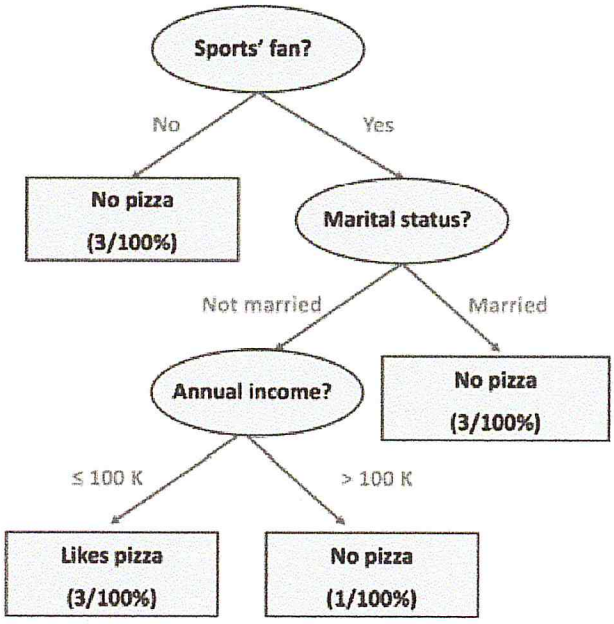



Figure 2 : Exemple d'un arbre de décision pour les données résumées dans 'Tableau 2'.

### 4.1.5. Classificateur bayésien naïf (Réseaux bayésiens) : [13]

Le classificateur bayésien naïf est un outil largement utilisé dans les problèmes de classification supervisée. Il a pour avantage de se montrer efficace pour de nombreux jeux de données réels. Cependant, l'hypothèse naïve d'indépendance des variables peut, dans certains cas, dégrader les performances du classificateur.

Comme son nom l'indique, ce classificateur se base sur le théorème de Bayes permettant de calculer les probabilités conditionnelles. Dans un contexte général, ce théorème fournit une façon de calculer la probabilité conditionnelle d'une cause sachant la présence d'un effet, à partir de la probabilité conditionnelle de l'effet sachant la présence de la cause ainsi que des probabilités a priori de la cause et de l'effet.

### 4.1.6. Machine à vecteurs de support (SVM) : [14]

Les machines à vecteurs de support (Support Vector Machine, SVM) appelés aussi séparateurs à vaste marge sont des techniques d'apprentissage supervisées destinées à résoudre des problèmes de classification. Les machines à vecteurs supports exploitent les concepts relatifs à la théorie de l'apprentissage statistique et à la théorie des bornes de Vapnik et Chervonenkis**Erreur ! Source du renvoi introuvable.**. La justification intuitive de cette méthode d'apprentissage est la suivante :

Si l'échantillon d'apprentissage est linéairement séparable, il semble naturel de séparer parfaitement les éléments des deux classes de telle sorte qu'ils soient le plus loin possibles de la frontière choisie. Ces fameuses machines ont été inventées en 1992 par Boser et al**Erreur ! Source du renvoi introuvable.**, mais leur dénomination par SVM n'est apparue qu'en 1995 avec Cortes et al. Depuis lors, de nombreux développements ont été réalisés pour proposer des variantes traitant le cas non-linéaire. Le succès de cette méthode est justifié par les solides bases théoriques qui la soutiennent. Elles permettent d'aborder des problèmes très divers dont la classification. SVM est une méthode particulièrement bien adaptée pour traiter des données de très haute dimension.

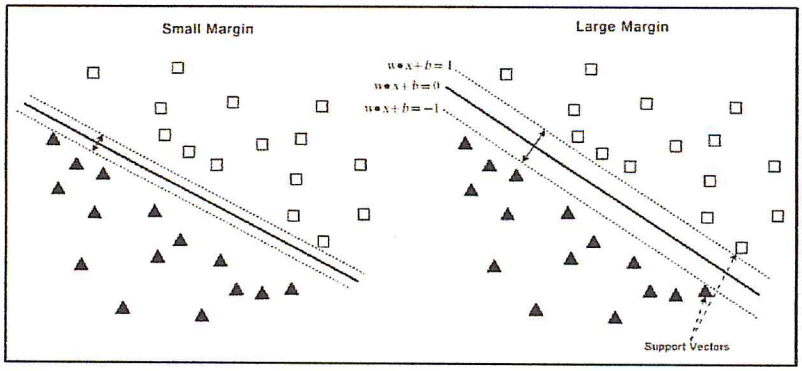



Figure 3 : Machine à vecteurs de support.

### 4.1.7. Les Machines à vecteurs de support dans les Systèmes de recommandation : [15]

Les Machines à vecteurs de support ont récemment gagné en popularité pour leur performance et efficacité dans de nombreux contextes. Les SVMs ont également montré des résultats récents prometteurs dans les systèmes de recommandation.

Kang et Yoo**Erreur ! Source du renvoi introuvable.**, par exemple, Effectuent une étude expérimentale qui vise à sélectionner la meilleure technique de prétraitement pour prédire les valeurs manquantes pour un système de recommandation basé sur SVM. En particulier, ils utilisent SVD (Décomposition en valeurs singulières) et la régression par les machines à vecteurs de support (SVR). Le système de recommandation basé sur SVM est construit d'abord par la binarisation des 80 niveaux de données disponibles sur les préférences d'utilisateur. Ils expérimentent avec plusieurs contextes et rapportent de meilleurs résultats pour un seuil de 32, C'est à dire une valeur de 32 et moins est classé comme 'préférer' et à une valeur plus élevée 'ne préfère pas'. L'id d'utilisateur est utilisé comme étiquette de classe et les valeurs positives et négatives sont exprimées en valeurs préférence 1 et 2.

Xia et al présentent différentes approches à l'aide de SVM pour les systèmes de recommandation dans le cadre de filtrage collaboratif. Ils étudient l'utilisation Machine à vecteurs de support avec lissage (Smooth Support Vector Machine, SSVM). Ils introduisent également une heuristique basée sur SSVM (SSVM-basedheuristic, SSVMBH) pour estimer de manière itérative des éléments manquants dans la matrice user-item. Ils calculent les prévisions en créant un classificateur pour chaque utilisateur. Leurs résultats expérimentaux

indiquent de meilleurs résultats pour le SSVMBH par rapport aux deux SSVM et le filtrage collaboratif traditionnel à base d'item et à base d'utilisateur.

### 4.1.8. Réseau de neurones artificiels (ANN) : [16]

Les réseaux de neurones représentent la technique de data mining la plus utilisée. Pour certains utilisateurs, elle en est même synonyme. C'est une transposition simplifiée des neurones du cerveau humain. Dans leur variante la plus courante, les réseaux de neurones apprennent sur une population d'origine puis sont capables d'exprimer des résultats sur des données inconnues. Ils sont utilisés dans la prédiction et la classification dans le cadre de découverte de connaissances dirigée. Certaines variantes permettent l'exploration des séries temporelles et des analyses non dirigées (réseaux de Kohonen). Le champ d'application est très vaste et l'offre logicielle importante.

Cependant, on leur reproche souvent d'être une "boite noire" : il est difficile de savoir comment les résultats sont produits, ce qui rend les explications délicates, même si les résultats sont bons.

Donc, Utiliser des technologies d'intelligence artificielle afin de découvrir par l'apprentissage du moteur des liens non procéduraux. Ces deux dernières techniques s'appuient sur des algorithmes mathématiques et tentent à travers des méthodes d'apprentissage de constituer des logiques non procéduralesErreur ! Source du renvoi introuvable..

Dans un réseau de neurones, un neurone est simplement une fonction non linéaire, de variables réelles et bornée. Cette fonction est généralement définie comme suit :

$$f(x_1, \ldots, x_k; w_1, \ldots, w_k) = \left| \sum_{i=1}^{k} w_i x_i \right|$$

Où les variables $w_1$, ..., $w_k$ correspondent à des poids à associer aux variables $x_1$, ..., $x_k$, qui sont déterminés à partir d'un corpus d'apprentissage. La fonction tangente hyperbolique est une fonction sigmoïde qui a certaines propriétés particulièrement appropriées pour l'apprentissage de réseaux de neurones. De tels neurones sont associés en réseau selon deux types d'architecture : les réseaux bouclés qui correspondent à des graphes orientés avec circuit et les réseaux non-bouclés qui correspondent à des graphes orientés sans circuit.

Dans le cadre de la recommandation basée sur le contenu, les variables $x_1$, ..., $x_k$ correspondent à la fréquence des termes utilisés pour caractériser les ressources (qui peut être

normalisée par rapport à la longueur du texte). L'architecture la plus fréquemment adoptée est l'architecture en réseaux non bouclés avec une structure de perceptron multicouche. Plus précisément, cette structure consiste en général en k entrées (les k attributs d'une ressource), une couche d'un certain nombre de neurones cachés, et un certain nombre de neurones de sortie. Chaque neurone de sortie indique un score permettant de déterminer si une ressource appartient à la classe du niveau d'appréciation à laquelle il est associé. Un algorithme répandu pour effectuer l'apprentissage des poids est l'algorithme PLA (Perceptron Learning Algorithm). Il consiste à initialiser les variables de façon aléatoire et à les ajuster itérativement de façon à minimiser le nombre de ressources disposées dans de mauvaises classes.

En plus de permettre un apprentissage rapide, l'utilisation de réseaux de neurones a l'avantage de permettre un ajustement particulièrement fin grâce à l'utilisation de la fonction sigmoïde. Selon le domaine d'application il peut s'avérer plus ou moins efficace que ses alternatives.

### Les six étapes clés de l'analyse prédictive : [17]

L'étude met en exergue un cycle de six étapes clés dans l'élaboration de solutions prédictives grâce au Big data :

1. Identifier les données utiles en évaluant diverses sources possibles ;
2. Triturer les data, les agréger, les compléter, …etc. ;
3. Construire un modèle prédictif, à partir d'algorithmes statistiques et de 'machine-learning' ;
4. Evaluer l'efficacité et la précision du modèle prédictif ;
5. Utiliser le modèle prédictif pour orienter des décisions métiers ;
6. Assurer un suivi de l'application et de l'efficacité du modèle prédictif.

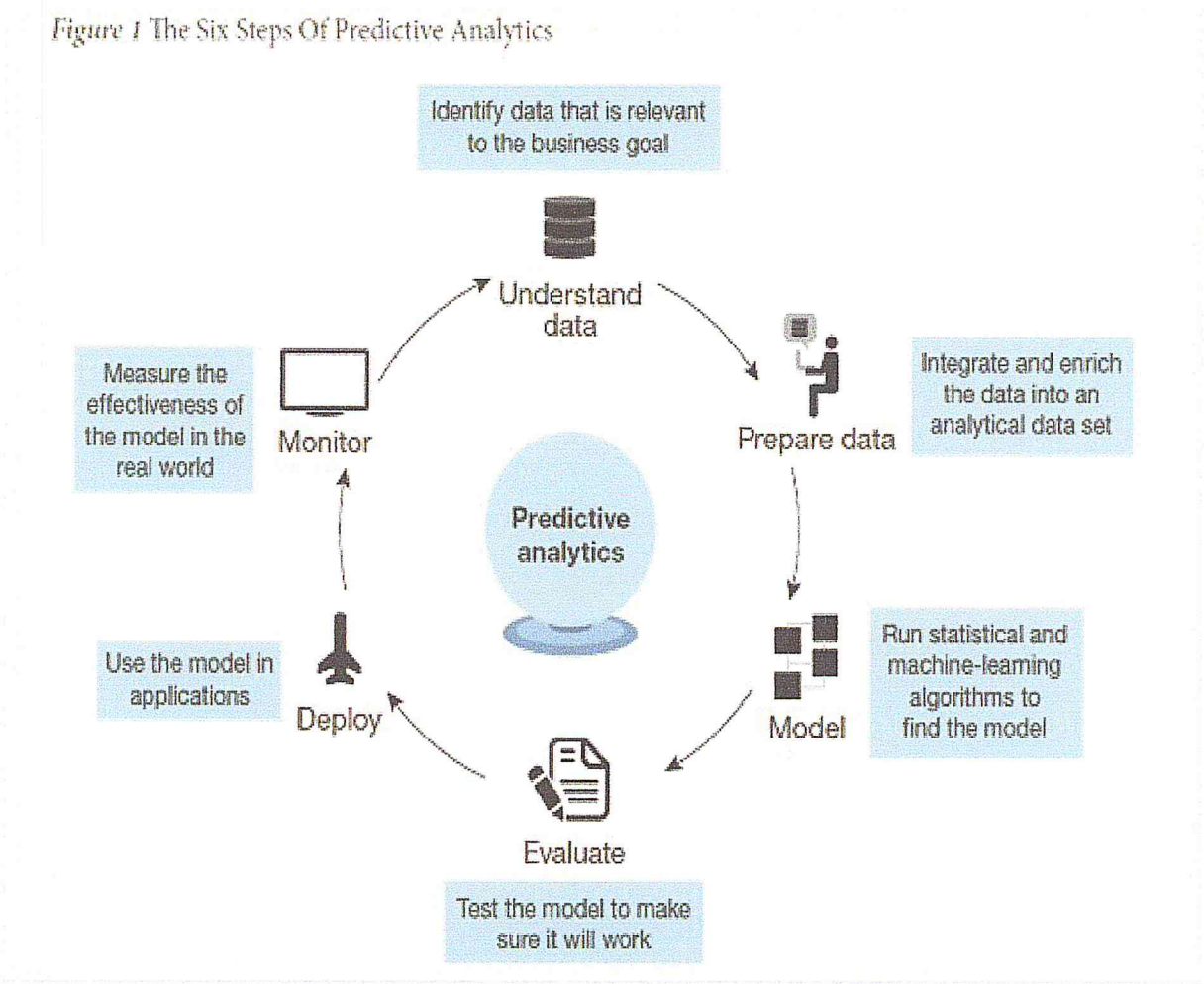



Figure 4 : Les six étapes-clés du Predictive Analytics.

## 5. L'efficacité des modèles prédictifs : [18]

Le Big data et l'imposante puissance de calcul associée, intégrés à divers nouveaux outils, rendent les modèles prédictifs plus efficaces, plus précis. Ces derniers sont également devenus plus accessibles aux entreprises, « y compris chez celles qui disposent de peu de compétences particulières en la matière », constate Forrester.

Les entreprises devraient gagner en capacité prédictive, grâce au Big data, selon trois axes :

- **L'apport de vues et de données nouvelles sur les clients et les processus métiers :** tableaux de bord et reportings deviennent des applications courantes en matière d'analyse prédictive au sein des organisations.

On obtient quantités d'informations sur l'impact possible de certaines situations et sur des projections à court ou moyen terme grâce à des modèles prédictifs simples. Mais il y manque souvent le lien vers le décisionnel métier, l'optimisation de processus ou encore l'expérience utilisateur, constate Forrester.

- **Le rajout d'interactions avec les utilisateurs et les clients** jusque dans les processus métier.

  Si l'organisation n'utilise pas d'outil prédictif pour anticiper l'avenir, elle risque d'employer des spécialistes - des data scientists - à ne rien faire !

  Aujourd'hui, les nouveaux outils de prédiction analytique permettent de déployer des modèles pertinents et d'activer des moteurs d'analyse dans les applications, là où l'on a besoin de perspicacité. « Les organisations tendent à utiliser le prédictif pour améliorer leurs processus métiers, par exemple en permettant de prévenir la fraude sur un service », explique le cabinet d'études.

- **La ré-implication du client ou consommateur** est rendue possible grâce à de nouveaux services numériques. Le recours aux analyses prédictives ouvre de nouveaux horizons dans les entreprises. L'élaboration de nouveaux modèles permet aux développeurs d'applications d'intégrer de nouvelles fonctionnalités et de les déployer rapidement.

## 6. Analyse prédictive et le Big Data : [19]

L'analyse prédictive nécessite d'importantes ressources de calcul. Les solutions récentes du Big data, portées par de nouvelles technologies - le In-memory, Hadoop, Distributed R, etc. - ouvrent des perspectives accessibles à tous. Ou presque.

Dans une récente étude, le cabinet Forrester Research a entrepris de comparer 13 solutions(*). Il montre que, grâce au Big data, les solutions d'analyse prédictive n'ont jamais été aussi pertinentes et faciles à utiliser que maintenant.

Ayant interrogé un panel d'entreprises représentatif du marché, le cabinet d'études révèle deux grandes tendances sur le Big data :

- 89% des dirigeants métier estiment que le Big Data va révolutionner les opérations métier au même titre qu'Internet ;
- 83% déclarent avoir entrepris des projets Big Data afin d'en obtenir un avantage compétitif.

Commentaire du consultant : *« Les réussites les plus visibles du Big Data s'observent dans des entreprises qui ont exprimé le besoin de se libérer de contraintes les empêchant d'être plus réactives vis à vis de leurs clients ou consommateurs ».*

## 7. L'apport des technologies Big data : [20]

Pour mener de tels travaux sur des données à grande échelle, les technologies Big data apportent une contribution indéniable, comme le montrent de récents développements chez HP, par exemple, reposant sur le langage Open source 'R'. L'offre 'Haven Predictive Analytics' utilise la technologie 'Distributed R', une extension qui est le fruit d'une coopération entre les HP Labs et HP Software. Elle tire notamment parti de la distribution de tâches de calculs sur plusieurs nœuds de traitement à grande capacité. Il devient ainsi possible d'élaborer des modèles d'analyse à partir de consoles Open Source 'R' (comme RStudio) capables de travailler sur des milliards d'enregistrements.

En clair, ce serait un changement radical d'échelle, comme en témoigne Cerner Corp. qui a testé la solution. Ce spécialiste IT du secteur de la santé a constaté que cette possibilité d'enrichir ses modèles prédictifs à l'échelle d'un traitement mondial des chiffres existants, permet de réduire considérablement le nombre des faux-positifs dans les diagnostics médicaux – ce qui signifie une réduction significative des interventions médicales inutiles.

## 8. Conclusion:

L'analyse prédictive englobe une variété de techniques issues des statistiques, dont l'objectif est d'associer une probabilité à un évènement futur. Le calcul de cette probabilité étant fondé sur l'observation du passé et toutes les données passées caractérisant le comportement à prédire.

Ce chapitre a été consacré à l'analyse prédictive, les méthodes prédictives et l'apport du big data à l'analyse prédictive. Nous aborderons en ce qui suit une méthode d'analyse

proposée et une architecture hadoop dans laquelle nous ferons des analyses sur des données réelles pour prédire des résultats futurs.

# Chapitre 3 : Conception

# 1. Introduction :

Après avoir initié aux deux premiers chapitres, supposons partie théorique, nous avons eu une idée sur les outils du big data et l'analyse prédictive, et tout cela pour faciliter la compréhension de ce qui vient dans ce chapitre qui contient notre comportement vis-à-vis les données pour arriver à la prédiction qui est notre but dans ce projet.

Nous allons voir dans ce chapitre la conception, du début à la fin, de l'extraction des données brutes aux résultats finaux de cette analyse ainsi que l'application sur des cas futurs, décrivant notre problématique, les objectifs de notre travail, les procédures suivies, l'architecture et la méthode proposée, ainsi que l'implémentation de l'architecture proposée dans les outils du big data choisis.

# 2. Etude de l'existant :

## 2.1. Entreprise et problématique :

Le travail a été réalisé au sein de GTP « Grands travaux pétroliers », une entreprise nationale publique spécialisée dans l'étude et la réalisation des installations industrielles notamment dans les domaines des hydrocarbures, de l'hydraulique, de l'énergie, de l'agroalimentaire, et des industries s'y rapportant à l'intérieur du territoire algérien et à l'étranger, la maintenance d'installations industrielles en exploitation, la formation dans le domaine de soudage, contrôle et activités annexes.

Elle est dotée d'un Conseil d'administration de 07 membres et d'une équipe de cadres dirigeants menée par un Président Directeur Général. Son capital social est actuellement de 6.390.000.000 DA.

Son siège social est fixé à : BP 38, zone industrielle de Réghaia, Alger. Elle est implantée sur le territoire national d'Est en Ouest et du Nord au Sud, notamment à Skikda, Arzew, Hassi R'mel, Hassi Messaoud et InAmenas où elle est représentée par des directions régionales.

Malgré le bon accueil, la facilité d'accès et la disponibilité des moyens physiques, nous n'avons reçu aucun soutien ni orientation, ce qui fait que notre travail été très difficile à réaliser.

Une interview a été faite avec le chargé de recrutement au niveau de département des ressources humaines, d'où nous avons constaté que l'entreprise a besoin d'une optimisation, tel qu'elle dépense de l'argent pour l'entreprise 'Emploitic' pour la présélection des personnes, qui va être mieux clarifiée au cours de ce chapitre.

Notre défi sera de concevoir une plateforme Big data afin de collecter et d'analyser toutes les données qui peuvent aider à une analyse prédictive afin d'optimiser les dépenses de l'entreprise et faciliter la présélection des candidats pour chaque offre d'emploi.

## 2.2. BPMN « Business Process Model and Notation » :

### 2.2.1. Définition du BPMN:

Le Business Process Model and Notation (BPMN) est un modèle de processus métier et une notation graphique standardisée pour modéliser le savoir-faire d'une organisation à travers l'approche processus. La première version était connue sous le nom de Business Process Modeling Notation. [21]

### 2.2.2. But de BPMN :

Le but principal de BPMN est de fournir une notation qui soit réellement compréhensible par tous les utilisateurs de l'entreprise, depuis les analystes métier qui créent les ébauches initiales des processus, jusqu'aux développeurs responsables de mettre en place la technologie qui va exécuter les processus applicatifs correspondants, et finalement, jusqu'aux utilisateurs de l'entreprise qui vont mettre en œuvre ces processus. Ainsi, BPMN crée un pont standardisé pour combler le vide entre la modélisation des processus métier et la mise en place des procédures. Actuellement, il y a des dizaines d'outils de modélisation et de méthodologies pour les procédures d'entreprise.

BPMN améliorera les possibilités des notations traditionnelles des procédures d'entreprise en gérant par nature les concepts de procédures B2B, comme les procédures publiques et privées et les chorégraphies, ainsi que des concepts de modélisation avancée, comme la gestion des exceptions et la compensation des transactions.

BPMN et UML sont deux spécifications de modélisation élaborées par l'OMG qui ne sont pas en compétition mais complémentaires. UML met l'accent sur la conception et le design du système d'information alors que BPMN adopte une approche 'orienté process' pour la modélisation des systèmes.

BPMN va apporter un bénéfice similaire à ce qu'UML a apporté au design et à la conception des logiciels.

BPMN représente une initiative visant à capitaliser et à consolider les bonnes pratiques des différentes méthodes et notations qui existaient dans le domaine de la modélisation de processus, UML activity diagram, IDEF, BPSS, Merise, OSSAD, ebXML et Event-Process Chains (EPCs). [22]

### 2.2.3. La procédure de recrutement de GTP :

Si y a un manque d'effectif dans un département, ce dernier contacte la direction générale. Cette dernière traite la demande et prend une décision selon les besoins et la situation financière de l'entreprise.

Si la direction générale trouve la demande pertinente, elle envoie une lettre à la direction des ressources humaines en lui demandant de faire une annonce. Le département des ressources humaines envoi l'annonce à Emploitic pour qu'elle se charge de la procédure de présélection des candidats.

Emploitic partage l'annonce dans sa plateforme, les candidats doivent être inscrits dans leur site pour qu'ils puissent postuler. Quand le dernier délai est acquis, le charger de recrutement accède à la plateforme d'Emploitic pour qu'il prenne la liste des meilleurs candidats selon les critères définis dans l'annonce.

Le charger de recrutement fait un entretien pour les candidats présélectionnés et sélectionne la ou les personnes dont il a besoin.

Un diagramme est présenté expliquant toute la procédure de recrutement, en mentionnant aussi la phase sur laquelle nous allons travailler.

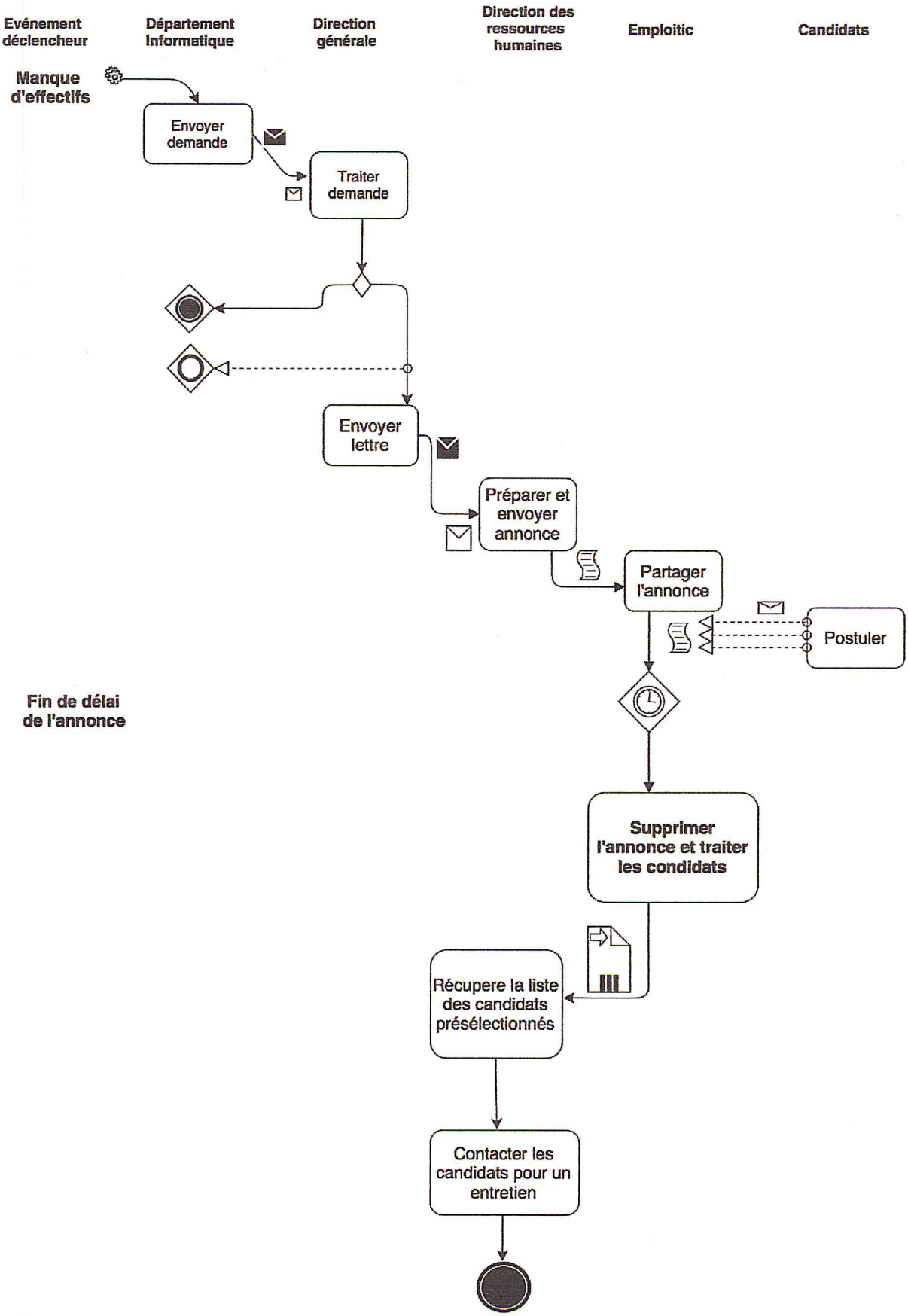



Figure 5 : Processus 'BPMN' pour la procédure de recrutement

## 3. L'architecture proposée :

La figure 7 présente une vue globale de notre système :

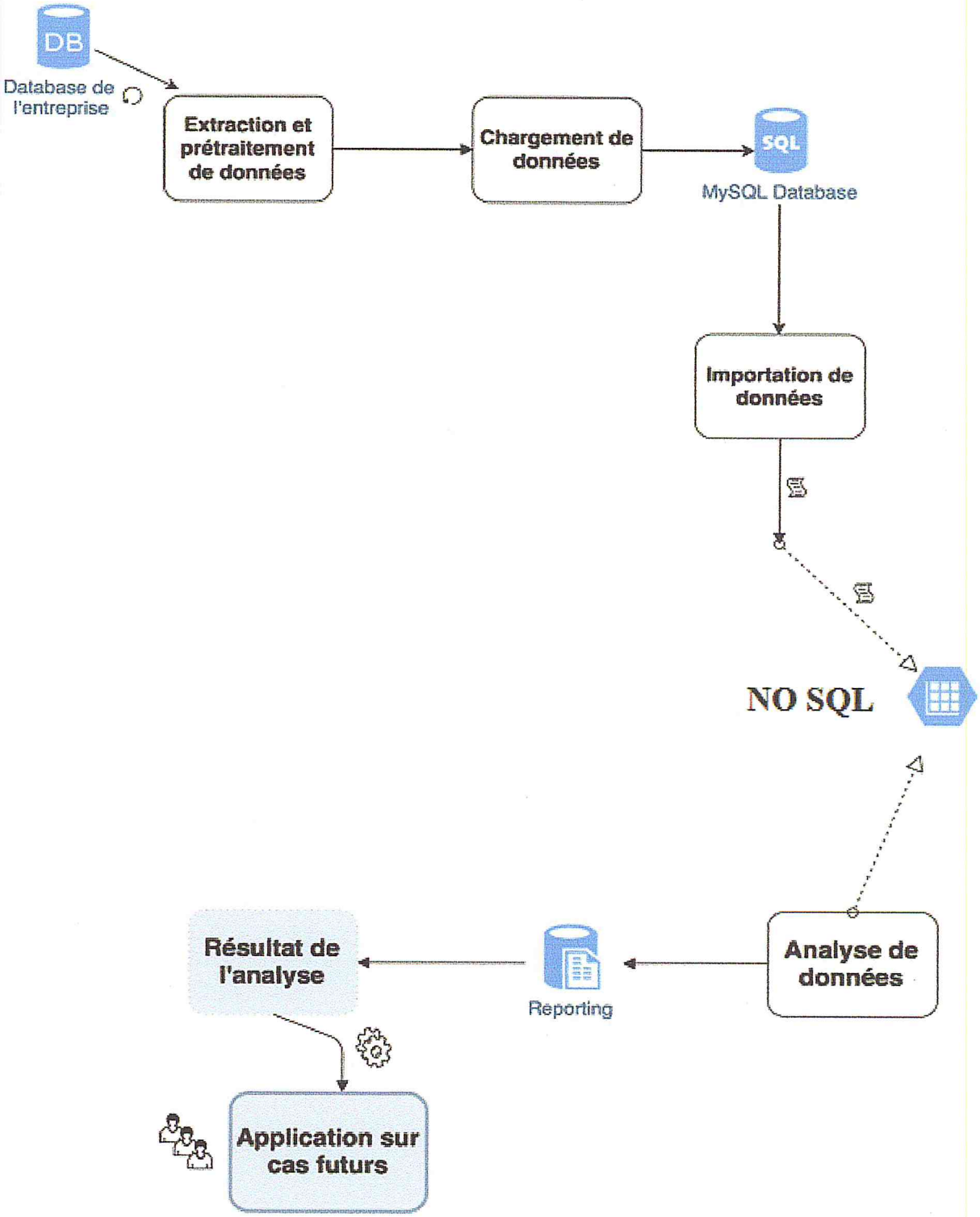



Figure 7 : Architecture proposée

Nous présenterons chaque étape en détail dans la partie 'Révision de l'architecture'.

## 3.1. Importation de données :

Le choix de l'outil de stockage de donnée n'est jamais une chose facile, ça dépend de plusieurs facteurs, à savoir la compatibilité, la quantité et le format des données à traiter.

### 3.1.1. Pourquoi Hive :

En ce qui concerne le stockage Hive a été proposé, vu les avantages dont son utilisateur bénéficie. Parmi ces avantages on trouve :

- Rappel : Apache Hive est une sur-couche à Hadoop qui permet le requêtage, l'analyse de gros volumes de données.
- Hive permet de définir des tables structurées de type SQL et de les alimenter avec des données provenant soit du cluster, soit de sources externes.
- Une fois le schéma des tables définies et les données insérées, il est possible d'utiliser le langage HiveQL pour requêter ces tables.
- Hive fournit un langage déclarative HQL de plus haut niveau pour faciliter le traitement des données à grande échelle.
- Le plus gros avantage de Hive est sa capacité à utiliser une compétence très répandue qu'est la connaissance de SQL rendant les développeurs très rapidement opérationnel pour extraire les données.
- Il traduit automatiquement les requêtes de type SQL en tâches MapReduce exécutées sur Hadoop. Parallèlement, HiveQL prend en charge les scripts MapReduce personnalisés qui se connectent aux requêtes.
- Il autorise également la sérialisation/désérialisation des données, et accroît la flexibilité de la conception de schémas en intégrant un catalogue système appelé Hive-Metastore
- Hive ne propose pas de requêtes en temps réel ou de mises à jour de niveau ligne. Il est optimisé pour les tâches en lots appliquées à des ensembles volumineux de données uniquement cumulatifs.

La figure suivante présente l'architecture de Hive.

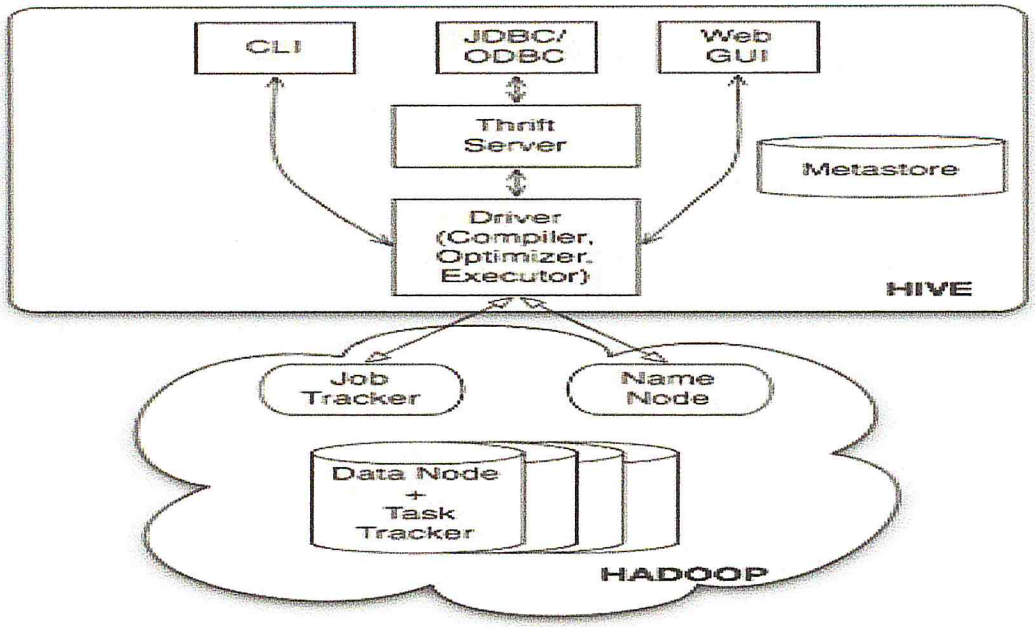



Figure 13 : Architecture de HIVE. [25]

### 3.1.2. Types de données : [26]

Il y a quelques formats de fichiers spécifiques que Hive peut gérer, tels que:

- TEXTFILE
- SEQUENCEFILE
- RCFILE
- AVROFILE
- PARQUETFILE
- ORCFILE

Avant d'aller en profondeur dans les types de formats de fichiers nous clarifions d'abord ce que veut dire un type.

Un format de fichier est une façon dont l'information est stockée ou codée dans un fichier informatique. Dans Hive, il se réfère à la façon dont les enregistrements sont stockés dans le fichier. Comme nous avons affaire à des données structurées, chaque enregistrement doit avoir sa propre structure. La façon dont un fichier est encodé définit le type de ce fichier.

Ces formats de fichiers varient principalement entre le codage de données, le taux de compression, l'utilisation de l'espace et le disque I / O.

Hive ne vérifie pas si les données que vous chargez correspondent au schéma de la table ou non. Cependant, il vérifie si le format de fichier correspond à la définition de la table ou non.

Voyons maintenant les types de formats de fichiers que peut supporter Hive en détail.

- **TEXTFILE :** Format fichier texte est un format d'entrée / sortie célèbre utilisé dans Hadoop. Si nous définissons une table comme fichier texte, Hive peut charger des données allant de CSV (Comma Separated Values), délimité par des tabulations, des espaces et des données JSON (JavaScript Object Notation).

  Par défaut, le format de fichier texte est utilisé, chaque ligne est considérée comme un enregistrement.

  Nous pouvons créer un format TEXT FILE dans Hive comme suit:

  # create table table_name (schema of the table) row format delimited by ',' | stored as TEXTFILE

  À la fin, nous avons besoin de spécifier le type de format de fichier. Si nous ne le spécifions le format TEXT FILE est pris par défaut.

  Les formats d'entrée et de sortie TEXTFILE sont présents dans le package Hadoop comme indiqué ci-dessous:

  # org.apache.hadoop.mapred.TextInputFormat
  # org.apache.hadoop.mapred.TextOutputFormat

- **SEQUENCEFILE :** Nous savons que la performance de Hadoop est tirée lorsque nous travaillons avec un petit nombre de fichiers avec grande taille plutôt que d'un grand nombre de fichiers avec une petite taille. Si la taille d'un fichier est plus petite que la taille du bloc typique dans Hadoop, nous le considérons comme un petit fichier. Pour cette raison, un certain nombre de métadonnées augmente, ce qui deviendra une surcharge au NameNode. Pour résoudre ce problème ce type de format de fichiers est introduit dans Hadoop. Les fichiers de type SEQUENCEFILE agissent en tant que conteneur pour stocker les petits fichiers.

  Les fichiers de ce type sont des fichiers plats constitués de paires clé-valeur binaire. Lorsque Hive convertit les requêtes à des emplois MapReduce, il décide sur les paires clé-valeur appropriées à utiliser pour un enregistrement donné. Ces fichiers sont dans le format binaire qui les rend en mesure de se diviser. L'utilisation principale de ces

fichiers consiste à fusionner deux ou plusieurs fichiers plus petits et les rendre sous forme d'un fichier de ce type.

En pratique, nous pouvons créer un fichier de ce type en spécifiant # STORED AS SEQUENCEFILE à la fin d'une instruction # CREATE TABLE.

Il existe trois types de fichiers de séquence:

-Fichier clé/valeur non compressé.

-Fichier clé/valeur compressé. Que les «valeurs» sont compressées ici.

-Fichier clé/valeur compressé par bloc. Les deux, clés et valeurs, sont collectées séparément en 'blocs' et compressés. La taille du «bloc» est configurable.

Hive a son propre lecteur de SEQUENCEFILE pour lire et écrire dans les fichiers de ce type.

Nous pouvons créer un format SEQUENCEFILE dans Hive comme suit:

# create table table_name (schema of the table) row format delimited by ',' | stored as SEQUENCEFILE

Hive utilise les formats d'entrée et de sortie SEQUENCE FILE des paquets suivants:

# org.apache.hadoop.mapred.SequenceFileInputFormat
# org.apache.hadoop.hive.ql.io.HiveSequenceFileOutputFormat

- **RCFILE : " Record Columnar File "** est un autre type de format de fichier binaire qui offre un taux de compression élevé sur la partie supérieure des rangées.

  RCFILE est utilisé lorsqu'on veut effectuer des opérations sur plusieurs lignes à la fois. Les fichiers de ce type sont des fichiers plats constitués de paires binaires clé/valeur. Ce type partage beaucoup de similitude avec SEQUENCE FILE.

  RCFILE stocke les colonnes d'une table en forme de disque d'une manière colonnaire. D'abord il divise horizontalement les rangées en des groupes de lignes, en suite il divise chaque groupe d'une manière colonnaire. Il stocke les métadonnées de chaque groupe de lignes, comme l'élément ''clé'' d'un enregistrement, et toutes les données ce chaque groupe de lignes comme l'élément ''valeur''. Cela signifie que RCFILE

encourage la colonne orientée vers le stockage plutôt que la ligne orientée vers le stockage.

Facebook utilise ce type comme format de fichier par défaut pour le stockage des données dans leur entrepôt de données et effectue différents types d'analyses à l'aide de Hive.

Dans Hive, nous pouvons créer un format RCFILE comme suit:

# create table table_name (schema of the table) row format delimited by ',' | stored as RCFILE

Hive a ses propres format d'entrée et format de sortie dans son package par défaut:

# org.apache.hadoop.hive.ql.io.RCFileInputFormat
# org.apache.hadoop.hive.ql.io.RCFileOutputFormat

- **ORC** : '' **Optimized Row Columnar** '' Ce type peut stocker des données de manière optimale que les autres formats de fichiers. ORC réduit la taille des données d'origine jusqu'à 75%. En conséquence, la vitesse de traitement de données augmente également.

  Il montre de meilleures performances que les formats TEXTFILES, SEQUENCEFILE et RCFILES.

  Un fichier ORC contient des données rangées dans des groupes appelés ''Stripes along'' avec un pied de page de fichier. Le format ORC améliore les performances lorsque Hive traite les données.

  Dans Hive, nous pouvons créer un fichier de format ORCFILE comme suit:

  # create table table_name (schema of the table) row format delimited by ',' | stored as ORC

  Hive a son propre format ORC d'entrée et de sortie dans son package par défaut :

  # org.apache.hadoop.hive.ql.io.orc

- **PARQUETFILE :** Parquet est un format de stockage colonnaire dans l'écosystème Hadoop. Par rapport à un format orienté ligne traditionnelle, il est beaucoup plus efficace dans le stockage et a une meilleure performance des requêtes. Parquet est largement utilisé dans le monde Hadoop pour l'analyse des charges de travail par de nombreux moteurs de requêtes. Parmi eux sont des moteurs sur le dessus de Hadoop, comme Hive, Impala et les systèmes qui vont au-delà de MapReduce pour améliorer les performances (Spark, Presto).

  Parquet stocke des données binaires d'une manière orientée colonne, où les valeurs de chaque colonne sont organisées de sorte qu'elles sont toutes adjacentes, ce qui permet

une meilleure compression. Il est particulièrement bon pour les requêtes qui lisent des colonnes particulières à partir d'une "large" table (avec beaucoup de colonnes), car que les colonnes nécessaires sont lues et le I/O est réduit au minimum.

Dans Hive, nous pouvons créer un fichier de format PARQUETFILE comme suit:

# create table table_name (schema of the table) row format delimited by ',' | stored as PARQUET

Hive a son propre format PARQUET d'entrée et de sortie dans son package :

# org.apache.hadoop.hive.ql.io.parquet

- **AvroFile :** Avro est un format de stockage des données. Il stocke les données en mettant la définition des données avec les données permettant les fichiers Avro à être lus et interprétés par de nombreux programmes différents. Il stocke toutes les données dans un format binaire qui rend les fichiers plus compacts, et il ajoute aussi des marqueurs pour aider les jobs de mapreduce pour trouver où casser de gros fichiers pour un traitement plus efficace. Il est utilisé en tant que plate-forme de sérialisation. Il est le plus convenable pour les schémas évolutionnaires.

  Dans Hive, nous pouvons créer un fichier de format AVROFILE comme suit:

  # create table table_name (schema of the table) row format delimited by ',' | stored as AVRO

  Hive a son propre format AVRO d'entrée et de sortie dans son package :

  # org.apache.hadoop.hive.ql.io.avro

Vu que notre travail se base sur l'analyse par colonne et après avoir fait des tests sur les données tout en comparant entre les PARQUETFILE et ORCFILE nous sommes convaincus que ce dernier est un choix parfait en termes de stockage et vitesse de traitement.

Le choix du type de données n'est pas suffisant. Ces données ont besoin d'un outil qui va jouer le rôle du transport. Nous présentons SQOOP en ce qui suit.

### 3.1.3. SQOOP :

Sqoop est un outil qui permet l'importation et d'exportation en masse à partir d'une base de données. Il peut être utilisé pour importer des données dans HDFS ou directement dans Hive.

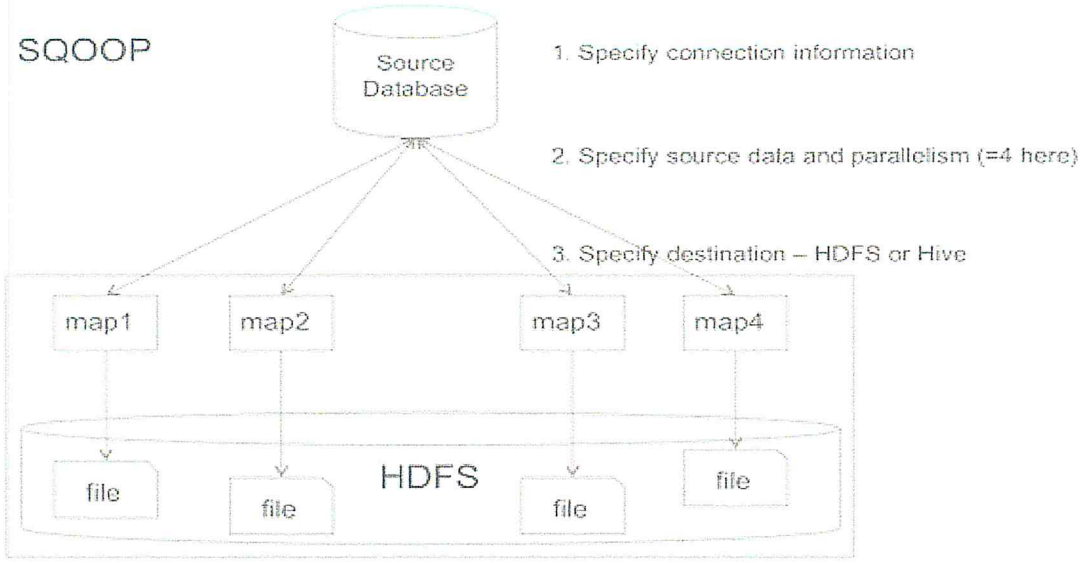



Figure 14 : Architecture de SQOOP. [27]

## 4. Conclusion :

Ce chapitre nous a pris beaucoup de choses. Nous avons vu la bonne exploitation de données, une méthode d'analyse prédictive, une architecture Hadoop, l'importation de données avec SQOOP et la manipulation de ces données dans le fameux HIVE. La partie de programmation est en cours de développement. La partie faite est, selon nous, très bénéfique et suffisante pour un tel travail vu la situation difficile, le manque d'orientation, le manque de référence causé par la nouveauté du thème, et les contraintes de temps.

# Chapitre 4 : implémentation

## 1. L'environnement du travail :

Nous allons mentionner et définir brièvement dans cette partie tous les composants et outils logiciels que nous avons utilisés durant notre projet.

### 1.1. VMware Workstation 10 :

VMware est un logiciel qui permet la création d'une ou plusieurs machines virtuelles simulant plusieurs systèmes d'exploitation au sein d'un même système d'exploitation. Celles-ci, pouvant être reliées au réseau local avec des adresses IP différentes, tout en étant sur la même machine physique qui existe réellement. Il est possible de faire fonctionner plusieurs machines virtuelles en même temps. Workstation maximise les ressources de l'ordinateur de façon à permettre l'exécution des applications les plus gourmandes dans un environnement virtuel. Il permet aussi de construire des machines virtuelles complexes pour l'exécution des applications Big Data sous forme d'un cluster Apache Hadoop.

La figure suivante présente VMware Workstation 10 :

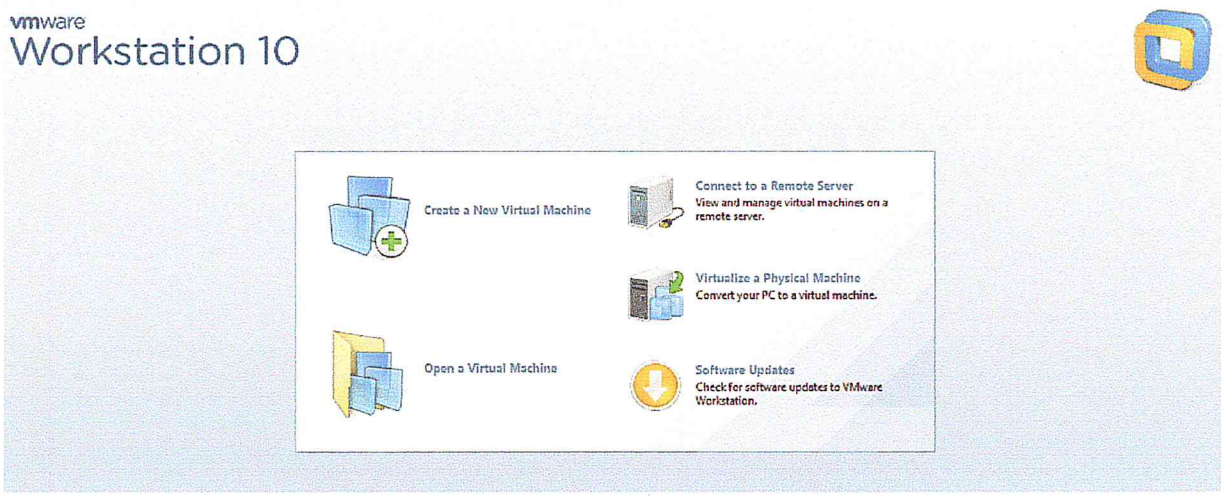



Figure 8: VMware Workstation 10

### 1.2. Linux CentOs-6.7 :

CentOS (Community entreprise Operating System) est une distribution GNU/Linux principalement destinées aux serveurs. Tous ses paquets, à l'exception du logo, sont des

paquets compilés à partir des sources de la distribution RHEL (Red Hat Entreprise Linux), éditée par la société Hat. On peut télécharger CentOs sous la forme de DVD OU CD. [23]


Figure 9 : Logo du système d'exploitation 'CentOS 6'

## 1.3. Cloudera CDH :

CDH est la distribution open source la plus complète et la plus populaire dans le monde. Cloudera est une société de logiciels américaine cofondée en 2008 par le mathématicien Jeff Hammerbach, un ancien de Facebook. Les autres cofondateurs sont Christophe Bisciglia, ex-employé de Google, Amr Awadallah, ex-employé de Yahoo, Mike Olson, PDG de Cloudera. En mars 2009, suivie de Diane Greene, la cofondatrice de VMware, de MartenMickos, l'ex-PDG de MySQL, et de GideonYu, le responsable des finances de Facebook.

La firme Cloudera se consacre au développement de logiciels fondés sur Apache Hadoop, permettant l'exploitation de Big Data, à savoir des bases de données accumulant plusieurs pétaoctets.

CDH contient les principales composantes d'Apache Hadoop pour le stockage évolutif et des calculs distribués. La figure 3.5 présente les composants de la distribution Hadoop de Cloudera.

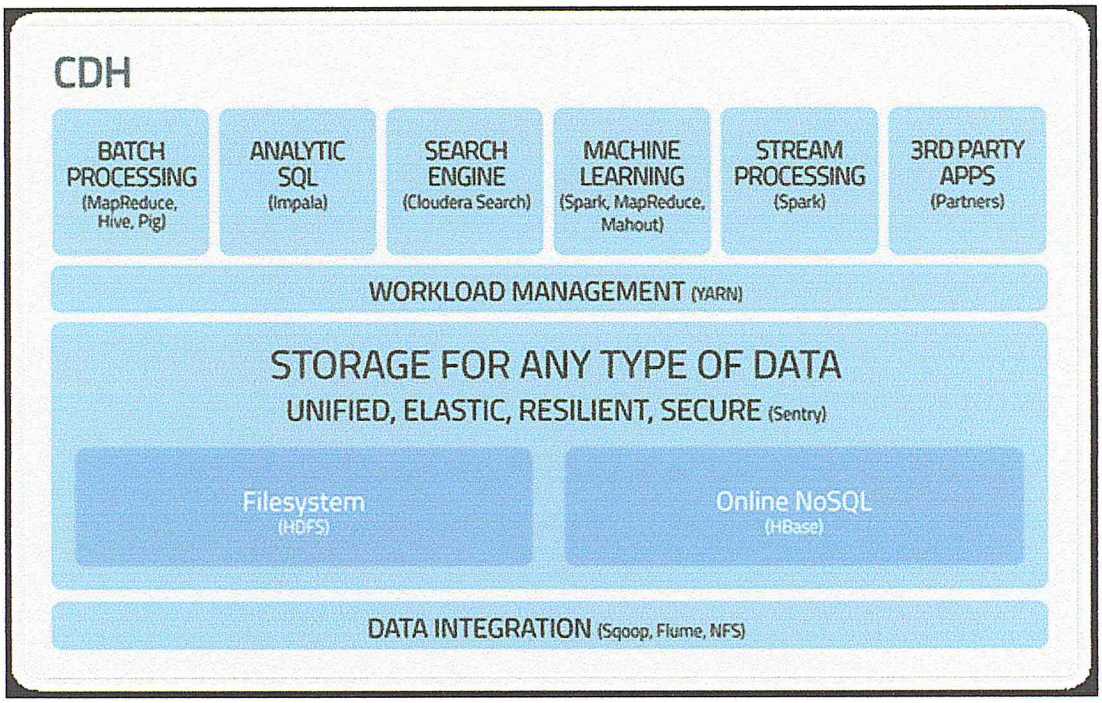



Figure 10 : Composants de la distribution Hadoop de Cloudera [24].

## 1.4. Pourquoi Hadoop :

Nous avons choisit la plateforme Hadoop pour les raisons suivantes :

✓ **Economique** : c'est une solution à moindre coût qui permet aux entreprises de libérer toute la valeur de leurs données en utilisant des serveurs peu onéreux.

✓ **Flexible** : l'utilisation de Hadoop-HDFS permet de stocker d'une manière extensible tout type de donnée structuré et semi-structuré (fichier proxy). Ces données sont stockées dans plusieurs machines afin de les traiter de façon distribuée.

✓ **Analyse performante** : l'utilisation de Hadoop-MapReduce permet d'analyser rapidement les différents types de données et de traiter parallèlement de multiples calculs en distribuant une opération sur plusieurs serveurs.

✓ **Architecture Scale-out** : cette architecture nous permet d'ajouter des ressources (stockage et puissance de traitement) à la demande et aux besoins.

✓ **Sécurité optimale de données** : Hadoop est également tolérant aux pannes puisqu'il réplique chaque donnée dans plusieurs serveurs afin de garantir qu'aucune donnée ne soit perdue. Il permet aussi la détection des pannes des serveurs.

✓ Peu de test sont nécessaires, les librairies mapreduce ont été déjà testées et fonctionnent correctement.

## 2. Révision de l'architecture proposée : (Intégration des logiciels)

Plusieurs outils ont été vus dans le premier chapitre. Le choix de ces outils dépend de plusieurs facteurs. Nous allons présenter tous les outils avec lesquels nous avons travaillés. Une justification est établie pour chaque outil choisi.

La figure 11 présente une vue globale de notre système :

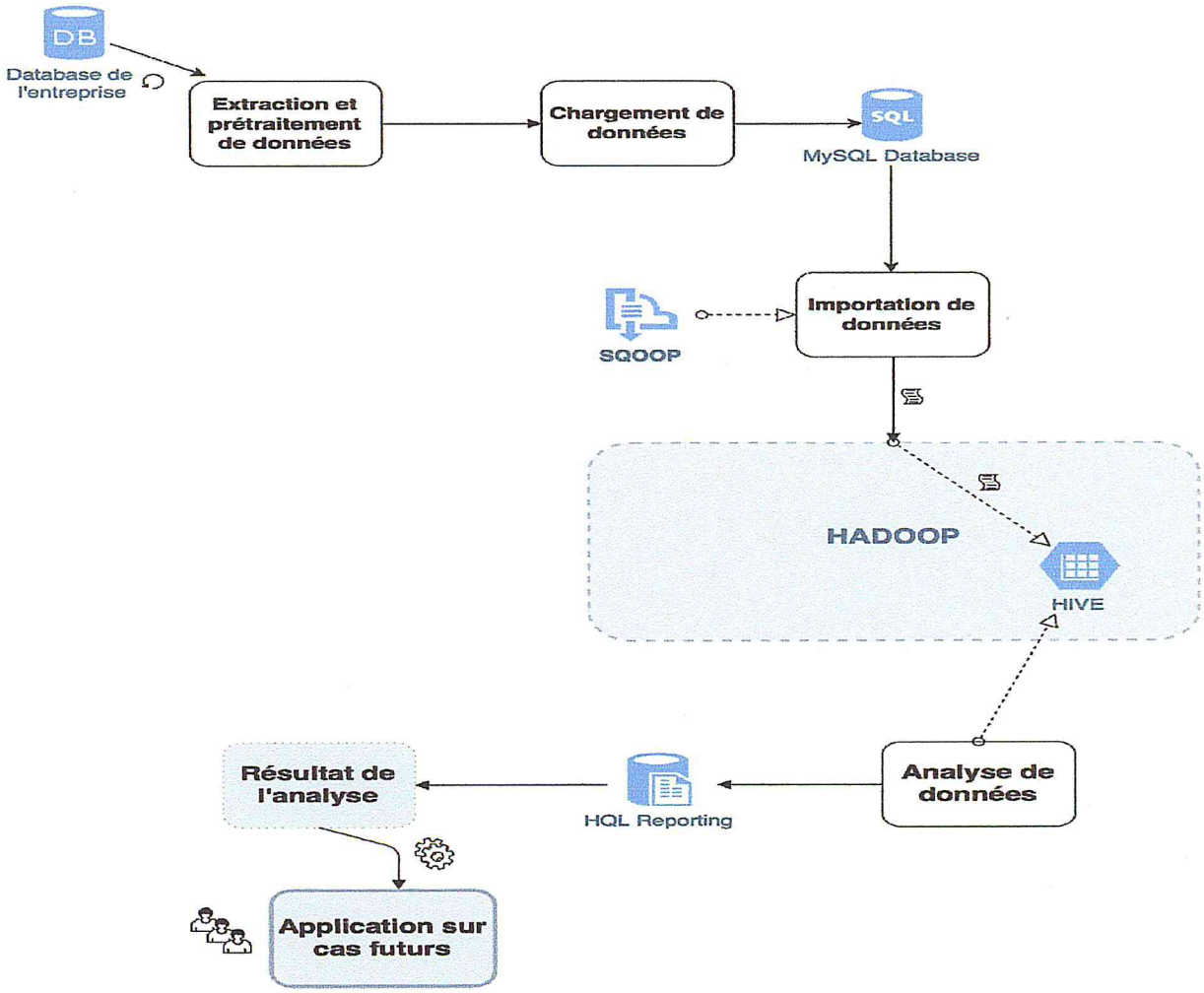



Figure 11 : Architecture du système avec intégration des outils.

Une description est associée, dans ce qui suit, pour chaque étape de notre architecture, ainsi que les méthodes et outils utilisés.

### 2.1. Extraction et prétraitement de données :

Cette étape est le début de notre travail, elle consiste à collecter les données à analyser. Et pour cela un prétraitement est recommandé afin de gagner du temps et d'espace et d'éviter les données dont nous n'avons pas besoin.

- L'analyse a été faite après une étape de prétraitement de données. Nous avons utilisé comme données tout ce qui peut influencer la décision du recruteur. Ces données sont : *Age, *Wilaya, *Service_National, *Niveau_Professionnel, *Spécialité, et *Possibilité du déplacement.
- Le nom et le prénom sont évités tant que l'identifiant est disponible. En réalité, pour cette étape nous n'avons pas besoin de savoir le nom de la personne, ni son identifiant.
- L'âge est acquit par une simple opération de soustraction : l'année actuelle- l'année de naissance du candidat.
- Une transformation a été faite sur chaque wilaya : par exemple : Alger == 16.
- Pour le service national, 5 choix sont possibles : "Compliqué", "Accompli", "Sursitaire", "Néant", "Dispensé".
- Les spécialités disponibles sont : "Travaux Publics", "Chaudronnerie industrielle", "Tuyauterie industrielle", "Mécanique industrielle", "Chantiers navals", "Réparation et de maintenance", "Industrie agricole", "Transport ferroviaire", "Chimie", "Pétrochimie", "Nucléaire", et "Aéronautique".
- Le potentiel de recrutement est ajouté dans cette phase. Ce critère va être bien clarifié dans l'étape d'analyse de données.

## 2.2. Chargement de données :

Notre environnement de travail, à savoir Cloudera, Hadoop, Hive et MySQL, est très spécifique en termes de configuration et manipulation. Pour faciliter la compréhension nous allons décrire l'utilisation de ces outils.

Après avoir fait un prétraitement sur les données pertinentes nous chargerons ces données dans notre base de données MySQL.

Les deux figures suivantes sont deux prises d'écran de deux parties de notre table MySQL sur le framework Cloudera.

- Pour accéder à MySQL depuis le terminal il faut exécuter la commande suivante :

```
[cloudera@quickstart ~]$ mysql -uroot –pcloudera
```

- Notre base de données s'appelle ADB : Analytical DataBase. Et la table concernée s'appelle AT : Analytical Table.

```
mysql> select * from AT ;
```

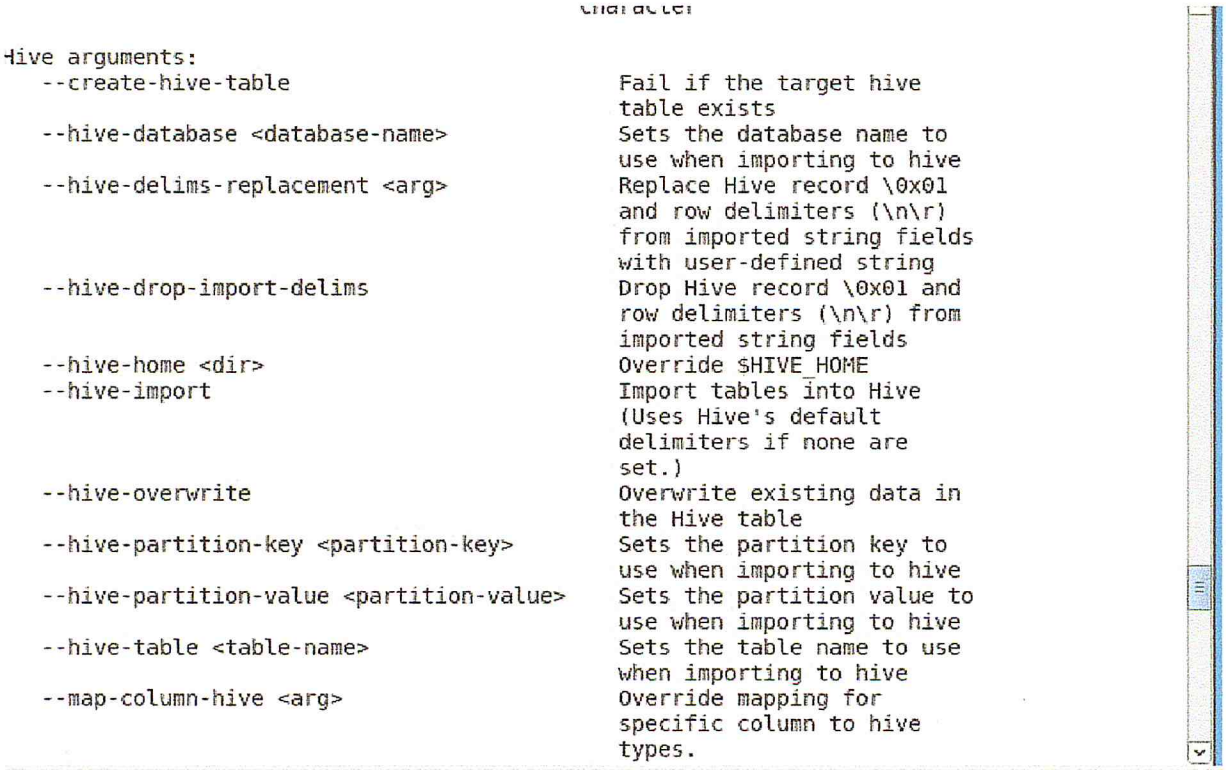
| id | Age | Wilaya | Service_National | Niveau_Profesionnel | Deplacement | Specialite | P_Recrutement |
|---|---|---|---|---|---|---|---|
| 1 | 30 | 16 | Accompli | 3 | Non | Pétrochimie | 1 |
| 2 | 32 | 16 | Accompli | 2 | Oui | Chaudronnerie industrielle | 1 |
| 3 | 35 | 16 | Accompli | 3 | Oui | Pétrochimie | 1 |
| 4 | 26 | 16 | Dispensé | 2 | Oui | Transport ferroviaire | 1 |
| 5 | 33 | 16 | Accompli | 2 | Oui | Chantiers navals | 1 |
| 6 | 26 | 16 | Dispensé | 3 | Oui | Travaux publics | 1 |
| 7 | 25 | 16 | Accompli | 1 | Oui | Pétrochimie | 1 |
| 8 | 21 | 16 | Sursitaire | 1 | Oui | Tuyauterie industrielle | 1 |
| 9 | 33 | 16 | Accompli | 2 | Oui | Mécanique industrielle | 1 |
| 10 | 26 | 16 | Accompli | 2 | Non | Industrie agricole | 1 |
| 11 | 39 | 16 | Accompli | 3 | Oui | Travaux publics | 1 |
| 12 | 27 | 16 | Accompli | 3 | Oui | Réparation et maintenance | 1 |
| 13 | 40 | 16 | Accompli | 1 | Oui | Tuyauterie industrielle | 1 |
| 14 | 22 | 16 | Sursitaire | 2 | Oui | Réparation et maintenance | 1 |
| 15 | 22 | 16 | Dispensé | 2 | Oui | Pétrochimie | 1 |
| 16 | 39 | 16 | Accompli | 2 | Oui | Pétrochimie | 1 |
| 17 | 25 | 16 | Accompli | 2 | Non | Mécanique industrielle | 1 |
| 18 | 27 | 16 | Accompli | 3 | Oui | Chantiers navals | 1 |
| 19 | 22 | 16 | Sursitaire | 2 | Oui | Chantiers navals | 1 |
| 20 | 25 | 16 | Accompli | 2 | Oui | Chantiers navals | 1 |
| 21 | 31 | 16 | Accompli | 2 | Non | Pétrochimie | 1 |
| 22 | 33 | 16 | Accompli | 2 | Oui | Pétrochimie | 1 |
| 23 | 24 | 16 | Accompli | 2 | Oui | Chaudronnerie industrielle | 1 |
| 24 | 40 | 16 | Dispensé | 2 | Non | Mécanique industrielle | 1 |
| 25 | 22 | 16 | Sursitaire | 3 | Oui | Mécanique industrielle | 1 |
| 26 | 35 | 16 | Accompli | 3 | Oui | Pétrochimie | 1 |

Figure 12 : Une partie de la base de données MySQL

Le stockage de Hive est associé avec HDFS. SQOOP peut importer les données dans Hive en générant et en exécutant une instruction # CREATE TABLE pour définir la mise en page des données dans Hive. Sqoop va convertir les données à partir des types de données natifs du stockage externe dans les types correspondants au sein de Hive.

# --hive-import, # --hive-table doivent être ajoutées afin de se distinguer de l'importation dans HDFS qui utilise # --target-dir

La figure suivante présente les commandes permises de Hive dans la plateforme Cloudera.



```
                                        character
Hive arguments:
   --create-hive-table                  Fail if the target hive
                                        table exists
   --hive-database <database-name>      Sets the database name to
                                        use when importing to hive
   --hive-delims-replacement <arg>      Replace Hive record \0x01
                                        and row delimiters (\n\r)
                                        from imported string fields
                                        with user-defined string
   --hive-drop-import-delims            Drop Hive record \0x01 and
                                        row delimiters (\n\r) from
                                        imported string fields
   --hive-home <dir>                    Override $HIVE_HOME
   --hive-import                        Import tables into Hive
                                        (Uses Hive's default
                                        delimiters if none are
                                        set.)
   --hive-overwrite                     Overwrite existing data in
                                        the Hive table
   --hive-partition-key <partition-key>  Sets the partition key to
                                        use when importing to hive
   --hive-partition-value <partition-value>  Sets the partition value to
                                        use when importing to hive
   --hive-table <table-name>            Sets the table name to use
                                        when importing to hive
   --map-column-hive <arg>              Override mapping for
                                        specific column to hive
                                        types.
```

Figure 15 : Liste des commandes permises de HIVE

Pour l'importation du type ORCFILE les étapes suivantes doivent être suivies:

Cependant, nous pouvons utiliser la fonctionnalité d'intégration Sqoop-HCatalog, qui est une abstraction de table.

D'abord, une réservation dans Hive est obligatoire pour le shéma de la table à importer.

# hive> CREATE TABLE HAT (id INT,Age int,Wilaya int,Service_National string,Niveau_Professionnel int,Deplacement string,Specialite string,P_Recrutement int) STORED AS ORCFILE;

Ensuite, nous passons à la partie d'importation, en ajoutant "--driver com.mysql.jdbc.Driver" pour éviter les erreurs de connexion.

# [cloudera@quickstart ~]$ sqoop import-all-tables --num-mappers 4 --connect "jdbc:mysql://quickstart.cloudera:3306/ADB" --username=root --password=cloudera --hcatalog-table HAT --driver com.mysql.jdbc.Driver

La figure suivante nous confirme que SQOOP se base sur le principe de mapreduce

```
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=119156
                Total vcore-seconds taken by all map tasks=119156
                Total megabyte-seconds taken by all map tasks=122015744
        Map-Reduce Framework
                Map input records=1300
                Map output records=1300
                Input split bytes=394
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=3832
                CPU time spent (ms)=25090
                Physical memory (bytes) snapshot=1076649984
                Virtual memory (bytes) snapshot=6304051200
                Total committed heap usage (bytes)=1264582656
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=0
16/09/05 08:14:46 INFO mapreduce.ImportJobBase: Transferred 8.7275 KB in 70.6366
 seconds (126.5208 bytes/sec)
16/09/05 08:14:46 INFO mapreduce.ImportJobBase: Retrieved 1300 records.
16/09/05 08:14:46 INFO hive.metastore: Closed a connection to metastore, current
 connections: 0
[cloudera@quickstart ~]$
```

Figure 16 : Importation de donnée de MYSQL vers HIVE en utilisant SQOOP

## 3. Analyse de données :

Plusieurs méthodes d'analyse prédictive ont été vues au deuxième chapitre. Vu la disponibilité des logiciels des outils d'analyse, comme SPSS pour la régression multiple, nous voulons nous même proposer comme solution une méthode d'analyse qui se base purement sur les statistiques.

Nous allons bien expliquer le principe de cette méthode et les étapes de notre analyse de données, qui sont dans notre cas les informations des candidats, dans ce qui suit.

Notre résultat final sera une classification des meilleurs candidats selon les critères qui ont été extraits des cas précédents selon le comportement du recruteur dans les 'x' derniers recrutements.

L'objectif de cette étape est de faire sortir une équation qui va être expliquée dans ce qui suit.

L'équation : « Y= a X1 + b X2 + c X3 + d X4 + e X5 + f X6 ». Tel que : Les Xi sont les critères étudiés. Pour chaque critère une pondération sera calculée. Cette dernière représente ici les valeurs a, b, c, d, e, et f. Et le Y est le potentiel de recrutement.

Donc nous allons étudier la dépendance des critères Xi par rapport au critère Y qui représente la probabilité du choix de chaque candidats.

Après l'étude de la dépendance entre les Xi et le Y, nous étudierons chaque critère Xi indépendamment des autres, c'est ce qu'on appelle une pondération interne. Nous aborderons les détails de calcul des pondérations internes et externes.

Comme nous avons déjà vu dans la partie 'chargement de données', nous avons ajouté une colonne, dans les données des candidats, qui représente le potentiel de recrutement : 'Y' dans notre cas. La valeur '1' a été affecté pour les personnes recrutées et une valeur de '0' pour le reste. Donc nous étudierons la dépendance de cette variable 'Y' par rapport aux autre critères ou variables Xi.

Pour avoir une idée le tableau suivant donne une perspective du résultat voulu.

| **Critère : Xi** | **Valeur :** | **P. interne :** | **P. externe :** |
|---|---|---|---|
| | j= 1 | | |
| | 2 | | |

| | | | |
|---|---|---|---|
| Critère(i) | . . | | |
| | j= m | | |

Tableau 4 : Représentation d'une perspective des résultats voulus

Nous décrivons maintenant la méthode de calcul pour chaque critère.

Pour chaque valeur j que peut prendre le critère Xi on applique cette fonction pour la pondération interne :

$$P(Xi.j) = \frac{Nij\ (1)}{Nij(T)} * \frac{Nij\ (1)}{N(1.T)}$$

Tel que : i € [1, nombre de critères], j = [1, nombre de valeur (varie selon i)].

- ***P(Xi.j)*** : la pondération interne du critère Xi pour la valeur j.
- ***Nij (1)*** : Le nombre d'occurrences de la valeur j que prend le critère Xi tant que Y=1 (On parle ici des personnes déjà recrutées).
- ***Nij (T)*** : Le nombre d'occurrences de la valeur j que prend le critère Xi dans tout l'échantillon T.
- ***N (1.T)*** : Le nombre de personnes, dans l'échantillon T, qui ont Y=1. En d'autre terme toutes les personnes déjà recrutées parmi l'échantillon T.
- $\frac{Nij\ (1)}{Ni.j\ (T)}$ : L'existence de la valeur j dans l'échantillon (1) par rapport à son existence dans tout l'échantillon T, tel que l'échantillon est composé de personnes qui ont Y=1, les personnes recrutées.
- $\frac{Nij\ (1)}{N(1.T)}$ : L'existence de la valeur j dans l'échantillon (1) par rapport à la taille d'échantillon de toutes les personnes recrutées.

En ce qui concerne la pondération externe de chaque critère une autre méthode est appliquée. Cette dernière sera expliquée dans ce qui suit.

Pour chaque critère Xi on applique cette fonction pour la pondération externe :

$$\mathbf{P(Xi)} = \frac{\boldsymbol{Ni\,(T)}}{\boldsymbol{Ni(1)}} / \boldsymbol{S}$$

Tel que: i € [1,6], S= ∑ P (Xi).

- ***Ni (T) :*** Le nombre de valeurs que peut prendre Xi dans tout l'échantillon.
- ***Ni (1) :*** Le nombre de valeurs que peut prendre Xi dans l'échantillon des personnes recrutées.
- $\frac{\boldsymbol{Ni\,(T)}}{\boldsymbol{Ni(1)}}$ **:** Ce rapport nous reflète sur quelle base le recruteur faisait ses choix. Plus le résultat est grand plus le critère est favorable.
- $\boldsymbol{S}$ représente la somme des rapports. Il est calculé pour équilibrer les poids des critères et pouvoir faire la comparaison entre eux. La somme ∑ P (Xi) doit être égale à 1.

Pour la pondération externe il faut la calculer pour tous les critères afin d'avoir la valeur S qui est demandée dans chaque opération.

Les résultats finaux de notre étude sont stockés dans des tables Hive comme suit :

Pour chaque critère Xi des tables ''PI(i)'', et une table ''PE'' commune pour tous les critères, sont stockées de la manière suivante :

| **Valeur :** | **p :** |
|---|---|
| v1 | p1 |
| v2 | p2 |
| … | … |
| vn | Pn |

Tableau 5 : Table PI(i)

La table PI(i) représente la pondération interne du critère i.

| **X :** | **pe :** |
|---|---|

| | |
|---|---|
| Age | |
| Wilaya | |
| service_national | |
| niveau_professionnel | |
| Déplacement | |
| Spécialité | |

Tableau 6 : Table PE

La table PE représente la pondération externe de chaque critère Xi.

## 4. Les algorithmes proposés :

Cette méthode d'analyse se caractérise par la simplicité et l'efficacité. Pour faciliter l'utilisation de cette méthode, des algorithmes sont en cours de développement. Nous avons osé faire un pas et nous espérons que ça ne s'arrête pas ici.

### 4.1. Phase d'analyse :

L'algorithme suivant nous permet d'appliquer une fonction d'analyse sur des données que nous voulons analyser, cette fonction fait appel à deux autres fonctions.

Les résultats de la fonction 'analysePE' sont une table commune de pondération externe pour tous les critères. Pour le calcul de la pondération interne on applique la fonction 'analysePI' pour chaque critère. Le résultat de cette fonction sera une table contenant toutes les valeurs possibles d'un critère et leurs pondérations internes.

```
** Function analyse (HAT table) {
        analysePE (HAT) ;          // retourne une table qui contient la pondération
                                   // externe de chaque critère.
        For (i==1, i<7, i++) {     // calcul de la pondération interne
            analysePI (HAT,i);     // de chaque valeur.
        }
}

** Function analysePE (HAT) {
    int n = HiveCount (*, HAT);
    HiveCreateTable(PE,X,pe);
    HiveInsert (PE, X, "age", "wilaya", "service_national", "niveau_professionnel",
                "déplacement", "spécialité") ;
    string c ;
    int i=1;
    int k=0;

    for (i==1, i<7, i++) {
        c= getname (i);
        int x = HiveQLCountDinstinct(Hat.c, HAT);
        int y = HiveQLCountDinstinctWhere(HAt.c, HAT, HAT.p_recrutement=1);
        V (i) = x/y;
        k= k + V (i);
    }

    For (i==1, i<7, i++) {
        c= getname (i);
        s= V (i) / k;
        HiveInsertWhere (PE.pe, s, PE.X= c);
    }
return PE;
}

** Function analysePI(HAT, i) {          // retourne la table de pondération
                                          //interne pour le i éme  critère.
    string c = getname(i);
    vecteur w = HiveSelectDistinct (HAT.c, HAT);  // stocker les valeur possible dans le
    t = taille (w);                               // vecteur w
    int x=0; int y=0; int z=0; int m=0;
    HiveCreateTable (Pi(c), valeur, p);   //création d'une table Hive avec des
                                          // valeurs possibles
    for (i==1, i<t+1, i++) {
        x = HiveCountWhere (*, HAT, HAT.c = w(i), HAT.p_recrutement=1);
        y = HiveCountWhere(*, HAT, HAT.c = w(i));
```

```
                z = HiveCountWhere (*, HAT, HAT.p_recrutement=1);
                m= (x*x) / (y*z);
                HiveInsert ( Pi(c) , (valeur, p) , (w(i), m) ) ;
        }
Return PI(c);
}
```

Algorithme 1: Phase d'analyse

## 4.2. Phase de sélection :

Les résultats obtenus de l'analyse prédictive doivent être exploités. Et pour cela la fonction 'selection' sera appliquée sur les cas futurs. Cet algorithme nous retourne un classement des candidats qui ont postulé pour un poste de travail. La fontcion 'getname' retourne le nom du i eme critère.

```
** Function selection (HT table) {        // Cette fonction retourne le classement des candidats

  int n = (count (*) from HT) ;            // n retourne la taille de la table T (nombre de ligne)

  For (i==1, i<n+1, i++)  {
      Read (T.Li);                         // Lire la i éme ligne dans la table T retourne un vecteur
      Y= Note (T.Li);                      // Appel à la fonction Note pour chaque ligne
      HiveAlterAddColumn (HT, p_recrutement int) ;  // Ajouter une colonne
      HiveInsert (HT, p_recrutement, Y) ;  // Insérer le résultat de 'Note' dans la
                                           nouvelle colonne
  }

  HiveQLOrderBy (HT, Y) ;                  // Classement des lignes selon Y
}

** Function Note (V) {                    // La fonction 'Note' calcule Y pour une ligne donnée.

      n=taille (V);                        // n retourne la taille du vecteur. (n=6 dans notre cas).

       Y=0;                                // Initialiser Y.

       For (i=1, i<n+1, i++) do {
          a(i) = (HiveQLWhere (PI(i).p, PI(i), PI(i).valeur=V(i)));  // a(i) récupère la
                                                                     // pondération interne.
          C= getname(i) ;                                  // getname retourne le nom du critère.
          b(i) = HiveQLWhere (PE.pe, PE, PE.X=C);  // b(i) récupére la pondération externe
          Y= Y + (a(i)*b(i));
       }

       Return Y;
}

** Function getname (i) {

 if (i=1) { return "age" else if (i=2) { return "wilaya" else  if (i=3) { return "service_national"

       else if (i=4) { return "niveau_professionnel" else if (i=5) { return "déplacement"

             else if (i=6) { return "spécialité" else return 0

  }}}}}}}

}
```

Algorithme 2 : Phase de présélection.

## 5. Tests :

Notre étude est faite sur une catégorie spécifique. Après avoir interrogé la base de données de l'entreprise nous avons constaté que le plus grand nombre de recrutements est pour la catégorie 'soudeur', et pour cela nous l'avons choisi.

Au niveau national, il existe trois niveaux professionnels dans la soudure. Le niveau est acquit par un stage et une certification, et c'est pour ça le recruteur n'a pas besoin de connaitre le niveau d'étude ni les compétences, spécifiquement pour cette catégorie.

Notre échantillon est constitué de 33550 personnes, dont 260 qui ont été déjà recrutées. L'analyse a été faite sur une période de 5 ans. (2011-2015).

### 5.1. Quelques exemples de calculs :

- **Exemple 1 :**

Le critère WILAYA a été pris comme exemple. Pour la pondération interne les calculs des autres critères se comportent de la même manière.

Prenant comme exemple i = 2 (WILAYA) et calculant la pondération interne pour la valeur '9'.

Pour i=2, j=9 : $P(X2.9) = \frac{N29\,(1)}{N29(T)} * \frac{N29\,(1)}{N(1.T)} = \frac{39}{1362} * \frac{39}{260} = 0.0042$

Les requêtes suivantes sont exécutées sur Hive afin d'avoir les résultats précédents :

**N29 (1):** # hive> select count (*) from hat where hat.wilaya=9 and hat.p_recrutement=1;

**N29 (T):** # hive> select count (*) from hat where hat.wilaya=9;

**N (1.T):** #hive> select count (*) from hat where hat.p_recrutement=1;

- **Exemple 2 :**

Le tableau suivant présente les résultats de pondérations externes de tous les critères.

| Critère : | *Ni (T) :* | *Ni (1) :* | $\frac{Ni\ (T)}{Ni(1)}$ | *S :* | *P(Xi) :* |
|---|---|---|---|---|---|
| **Age** | 29 | 25 | 1.16 | 13.79 | **0.084** |
| **Wilaya** | 48 | 8 | 8 | 13.79 | **0.580** |
| **Service_National** | 4 | 3 | 1.33 | 13.79 | **0.096** |
| **Niveau_professionnel** | 3 | 3 | 1 | 13.79 | **0.072** |
| **Déplacement** | 2 | 2 | 1 | 13.79 | **0.072** |
| **Spécialité** | 13 | 10 | 1.3 | 13.79 | **0.094** |

Tableau 7 : Pondérations externes des critères

Les requêtes suivantes sont exécutées afin d'arriver au résultat de pondération externe du critère 'âge'.

- N1(T) = 29 : #hive > select count (distinct hat.age) from hat ;
- N1(1) = 25 : #hive > select count(distinct hat.age)from hat where hat.p_recrutement==1;

Nous remarquons que le poids le plus grand est celui du critère 'WILAYA'. Par conséquent nous constatons que le recruteur se basait dans le choix sur la wilaya du candidat, beaucoup plus que sur les autres critères.

- **Exemple 3 :**

Le tableau suivant présente les résultats de la pondération interne et externe pour le critère 'niveau_professionnel'.

| Critère : | Valeur : | P. interne : | P. externe : |
|---|---|---|---|
| niveau_professionnel | 1 | 0.0015 | 0.072 |
| | 2 | 0.0038 | |
| | 3 | 0.0027 | |

Tableau 8 : Représentation des résultats de pondération pour le critère niveau_professionnel

$$P(X4.1) = \frac{N41\,(1)}{N41(T)} * \frac{N41\,(1)}{N(1.T)} = \frac{65}{10771} * \frac{65}{260} = 0.0015$$

$$P(X4.2) = \frac{N42\,(1)}{N42(T)} * \frac{N42\,(1)}{N(1.T)} = \frac{106}{11149} * \frac{106}{260} = 0.0038$$

$$P(X4.3) = \frac{N43\,(1)}{N43(T)} * \frac{N43\,(1)}{N(1.T)} = \frac{88}{10872} * \frac{88}{260} = 0.0027$$

## 5.2.Phase d'apprentissage :

Maintenant, et après avoir clarifié en détail la méthode de calcul pour quelques exemples, nous appliquons notre 'algorithme 2' sur un échantillon constitués de 20750 personnes. Nous travaillons sur les 3 dernières années de notre échantillon d'analyse (2013-2015) ou 140 personnes ont été recrutées. Nous calculons ensuite la fiabilité de notre application.

Nous avons stocké le résultat de cette étape dans une table Hive. Une comparaison entre cette table et la table originale, sur laquelle nous avons fait l'analyse, a été faite afin de calculer la fiabilité.

Nous remarquons qu'il existe 90 personnes, qui ont été à l'origine recrutées, parmi les 140 premiers candidats.

Donc la fiabilité = 90/140 = 64.28 %.

# Conclusion générale

# Conclusion Générale

Maintenant, et après avoir vu plein d'outils que ce soit d'analyse prédictive ou bien du big data, nous avons bien compris et nous somme aptes de manipuler ces outils, et tout cela pour mettre de la valeur ajoutée pour notre société.
Le bilan de ce stage est dans l'ensemble positif, les principaux buts du projet étant accomplis. La plus grande partie du travail qui nous a été demandée a été réalisée.

Nous avons réalisé l'objectif attendu qui était la compréhension du nouveau paradigme qui est le Big data et implémenté le plus répandu de ses outils qui est Hadoop. Ce dernier nous a permis de nous familiariser avec le concept du Big data et surtout d'élaborer une architecture d'analyse de données.

Notre travail s'insère dans une vision à long terme dans laquelle des améliorations peuvent avoir lieu comme la finalisation du développement de l'application et la généralisation niveau national.

Pour conclure, on dirait que ce stage nous a été d'un apport indéniable en matière de connaissances acquises sur les technologies du Big Data. C'est une expérience enrichissante qui nous a permis de comprendre les enjeux d'un projet 'big data' et l'utilité des méthodes d'analyse prédictive.

# Bibliographie


[1]: Wikipedia, (2006), BigData, http://fr.Wkipedia.org/WiKi/Big_Data#cite_note-5 (04/04/2016, 16h.15).

[2] : Cf.JO du 24 aout 2014.

[3] : lebigdata.fr, L'avenir du Big Data, (2016) http://www.lebigdata.fr/definition-big-data , (05/04/2016, 15h.30).

[4]: Big Data : La jungle des différentes distributions open source Hadoop,Christophe PARAGEAUD,(Mai2013), http://blog.ippon.fr/2013/05/14/big-data-la-jungle-des-differentes-distributions-Open-source-hadoop/, (24/04/2016, 10h.50).

[5]: Big Data : La jungle des différentes distributions open source Hadoop,Christophe PARAGEAUD,(Mai2013), http://blog.ippon.fr/2013/05/14/big-data-la-jungle-des-differentes-distributions-Open-source-hadoop/, (24/04/2016, 10h.50).

[6]: Guide du Big data 2013-2014, Blandine LAFFARGUE, la societe Corp Events.

[7]: Zdent, (2016), Big Data, Les news Big data, http://www.zdnet.fr/actualites/big-data-4000237198q.html, (08/04/2016, 16h.37).

[8]: Piloter, (2016), Technologie big data, http://www.piloter.org/business-intelligence/technologie-Big- data.html, (08/04/2016, 19h.50).

[9]: Wikipedia, (2016), Analyse predictive, https://fr.wikipedia.org/wiki/Analyse_pr%C3%A9dictive , (07/09/2016, 18h.04).

[10] : docplayer.fr, (2016), analyse prédictive http://docplayer.fr/20357046-Universite-d-ete-gs1-analyse- predictive-appliquee-au-e-commerce.html, (07/09/2016, 18h.15).

[11] : Classification supervisée Les K-plus proches voisins, Faicel Chamroukhi, (P3), (2013).

[12] : Arbres de décision, Cécile Capponi, Université Aix-Marseille, (2014).

[13] : docplayer.fr, (2016), Le classificateur bayésien naïf, http://docplayer.fr/13016027-Optimisation-directe-des-poids-de-modeles-dans-un-predicteur-bayesien-naif-moyenne.html .

[14] : Classifieurs SVM et Réseaux de Neurones, HAMZA CHERIF, (2011).

[15] : De la classification d'opinions à la recommandation, D Poirier, Françoise Fessant, Isabelle Tellier, Orange Labs, Laboratoire d'Informatique Fondamentale d'Orléans, (2010).

[16] : Thèse de Richard-Emmanuel Eastes, Processus d'apprentissage, savoirs complexes et traitement de l'information, Université de Genève, soutenue le 11 juin 2013 à Paris.

[17] : Les six étapes clés de l'analyse prédictive, Forrester Research, (04/2015).

[18] : L'efficacité des modèles prédictifs, Par La rédaction de ZDNet.fr, Jeudi 10 Septembre 2015.

[19] : Big data : quel intérêt pour l'analyse prédictive ? , Par La rédaction de ZDNet.fr, Jeudi 10 Septembre 2015.

[21]: BPMN, http://www.bpms.info/bpmn-cartographie/, (03/05/2016, 19h.12).

[22]: Business Process Model and Notation, https://fr.wikipedia.org/wiki/Business_Process_Model_and_Notation .

[23]:http://pvbookmarks.readthedocs.io/en/master/devel/OS/linux/linux_based/centos/index.html, 2016.

[24]: PARAGEAUD (2014). La jungle des di_érentes distributions open source Hadoop. Http://blog.ippon.fr/2013/05/14/big-data-la-jungle-des-di_erentes-distributionsopen-source-hadoop/ .

[25]: Hadoop Hive Architecture, (2015), http://www.hadooptpoint.com/hadoop-hive-architecture/ .

[26]: Architecture de SQOOP, https://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.3.0/bk_dataintegration/content/figures/1/figures/moving-data-to-hive-with-sqoop.png (05/09/2016 06h.04).