Projet BigData NoSQL

Sujet

Le jeu de données représente des données de clients cherchant à obtenir un prêt immobilier. Chaque client a des informations sur son emploi, son statut, le type de bien qu'il souhaite acheter... Chaque client possède aussi un attribute « TARGET » qui vaut soit 0 (la personne a remboursé son prêt dans les temps) ou 1 (la personne a eu des difficultés pour rembourser son prêt). L'objectif de l'analyse est d'être capable de prédire si une personne pourra ou ne pourra pas rembourser son prêt, étant en connaissance d'informations sur elle. Les données sont téléchargeables ici.

Etape 0

Les données doivent être installées sur HDFS, sur une VM Hadoop (ex : HortonWorks Data Platform 2.6) téléchargée et installée sur un Virtual Box local.

Etape 1

Ces données sont rapatriées en local, via un script dédié, qui les récupère depuis HDFS.

Etape 2

Ces données sont poussées sur une VM dans le cloud AWS. La nature des données étant sensible, vous devez au préalable vous posez les questions de sécurité à mettre en œuvre sur la VM. Par ailleurs, les données poussées sur la VM doivent être chiffrées en amont.

Etape 3

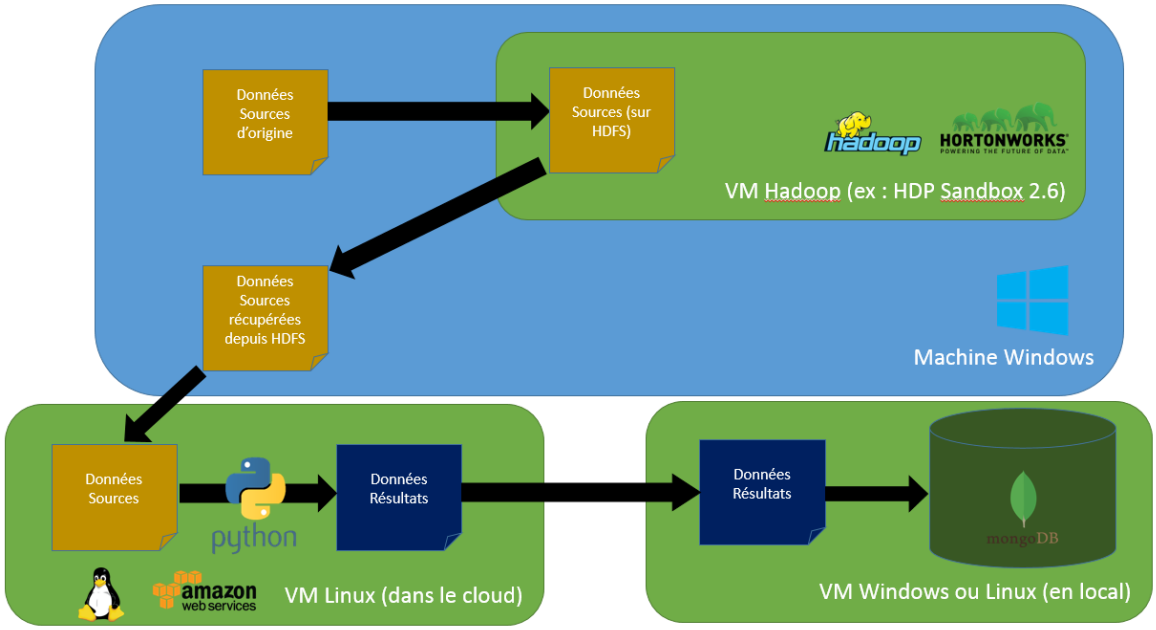
Un modèle d'apprentissage est appris à l'aide d'un des algorithmes de machine learning vu en cours ainsi que des données fournies. L'algorithme est exécuté sur cette VM dans le cloud AWS. Le choix de l'algorithme utilisé devra être justifié.

Etape 4

Vous utiliserez une partie des données pour créer un fichier predict.csv contenant uniquement les informations clients, c'est-à-dire sans le champ « TARGET ». Une taille d'environ 10k prédictions est suffisante. Vous exécuterez sur la VM AWS votre modèle appris afin de prédire si les contractants du fichier predict.csv sont en mesure de rembourser leur prêt. Un fichier de résultats sera créé en concaténant chaque ligne de predict.csv avec la prédiction de votre modèle. Ce fichier sera sauvegardé sur le FileSystem de la VM AWS, au format CSV.

Etape 5

Les résultats de l'étape 4 sont alors récupérés et chargés dans une base NoSQL (par ex. MongoDB, choix le plus approprié à faire par l'équipe) s'exécutant en local (sur une VM ou sur l'OS local), via un script dédié (langage ou outil à déterminer par l'équipe).



Modalités

Equipes

Le projet se fait par équipes de 3 (5 équipes de 3, une de 4).

Lors des soutenances, les questions pourront être posées indifféremment aux différents membres du groupe. Cela signifie que chacun doit avoir une maîtrise a minima des différentes parties du projet.

Livrables

Rapports intermédiaires

Le projet doit faire l'objet de 2 livraisons de rapports intermédiaires :

- 18/01/2019 : Rapport de reformulation du sujet
- 25/01/2019 (avant la soutenance intermédiaire qui a lieu l'après-midi) : Rapport intermédiaire : statut de l'avancement

Le rapport de reformulation est un document « word » qui fait état de la bonne compréhension du sujet.

Le rapport intermédiaire est un document « word » qui fait état de l'avancement de l'équipe (tâches réalisées, reste à faire, choix effectués avec leur justification…)

Soutenance intermédiaire

Le 25/01 (après-midi) auront lieu les soutenances intermédiaires : présentation par l'équipe du travail réalisé devant le jury des évaluateurs. 20 min (10 de présentatoin +10 de questions) par soutenance.

Soutenances finales

Le 14/02 auront lieu les soutenances finales : présentation par l'équipe du travail réalisé devant le jury des évaluateurs. 30 min (20 de présentation + 10 de questions) par soutenance.

Livrables finaux

Le 14/02 (avec la soutenance finale) devront être remis les livrables finaux :

- Document de soutenance
- Repository Github avec les codes sources réalisés pendant le projet (scripts, notebooks…)
- ReadMe file (dans le repo Github) expliquant le projet

Notation

- Rapport de reformulation (sur 10)
- Rapport intermédiaire (sur 10)
- Soutenance intermédiaire (sur 20)
- Soutenance finale (sur 20)
- Livrables finaux (sur 20)