DÉPLOYEZ UN MODÉLE DANS LE CLOUD

Présenté par Souleymane Camara

Contexte de l'étude

L'ENTREPRISE AGRITECH, NOMMÉE "FRUITS!"

- cherche à proposer des solutions innovantes pour la récolte des fruits
- La volonté de l'entreprise est de préserver la biodiversité des fruits en permettant des traitements spécifiques pour chaque espèce de fruits en développant des robots cueilleurs intelligents.
- L'entreprise cherche à se faire connaître en mettant à disposition au grand public une application mobile qui permettrait aux utilisateurs de prendre en photo un fruit et d'obtenir des informations sur ce fruit.
- Cette application permettrait de sensibiliser le grand public à la biodiversité des fruits et de mettre en place une première version du moteur de classification des images de fruits, il permettra aussi de construire une première version de l'architecture Big Data nécessaire.
- Pour réaliser cette étude nous avons une table de données constitué des images de fruits et des labels associés, qui pourra servir de point de départ pour construire une partie de la chaîne de traitement des données
- On doit développer dans un environnement Big Data une première chaîne de traitement des données qui comprendra le preprocessing et une étape de réduction de dimension.

Contexte de l'étude

L'ENTREPRISE AGRITECH, NOMMÉE "FRUITS!"

- Pour faire ce passage à l'échelle on doit utiliser des scripts pyspark et travailler sur le cloud AWS



Présentation des données

NOUS AVONS DES DONNÉES DE TRAIN ET DE TEST:

Origine:

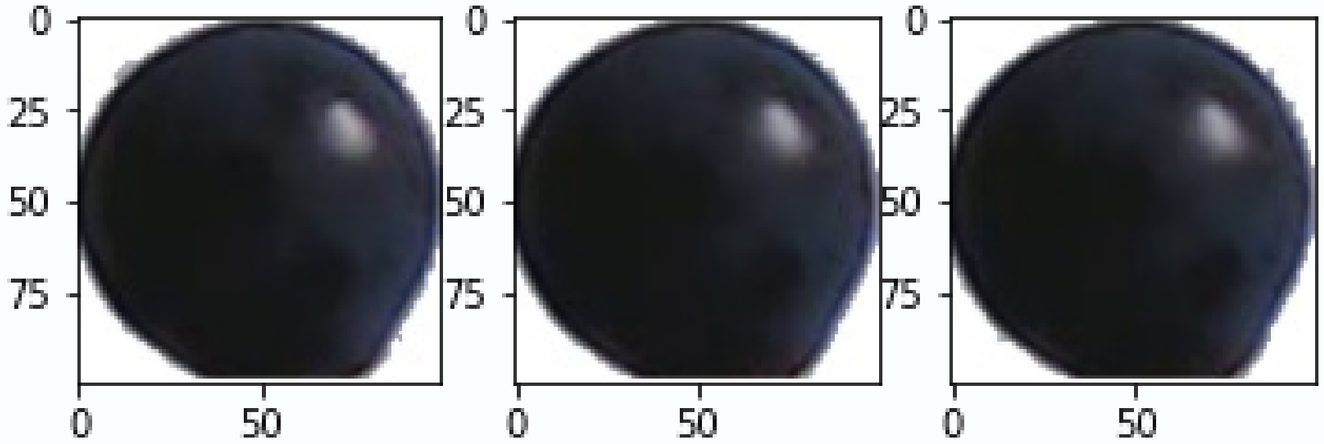
- Images de fruits et labels associés (Fruits 360, Mihai Oltean)
- 131 variétés de fruits différents (un dossier par variété)
- Plusieurs variétés du même fruit (exemple : pomme « red » et « golden »)

Caractéristiques :

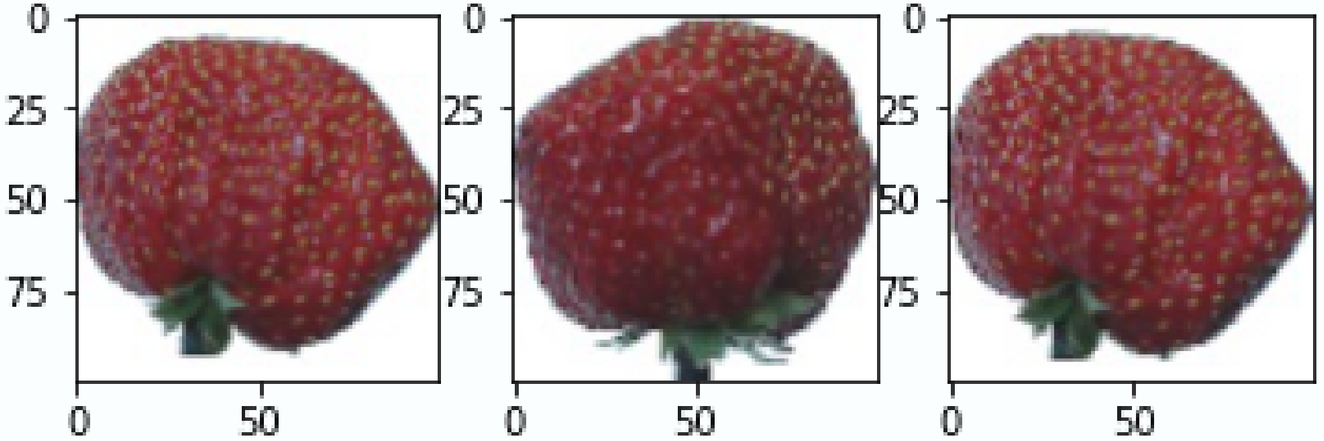
- Images 100x100 JPEG RGB
- Photos studio sur fond blanc de fruits centrée sur le fruit
- Photos sous tous les angles (timelapse + rotation 3 axes)

Jeu d'entraînement : 67702 images

Exemples d'images



Grap Blue

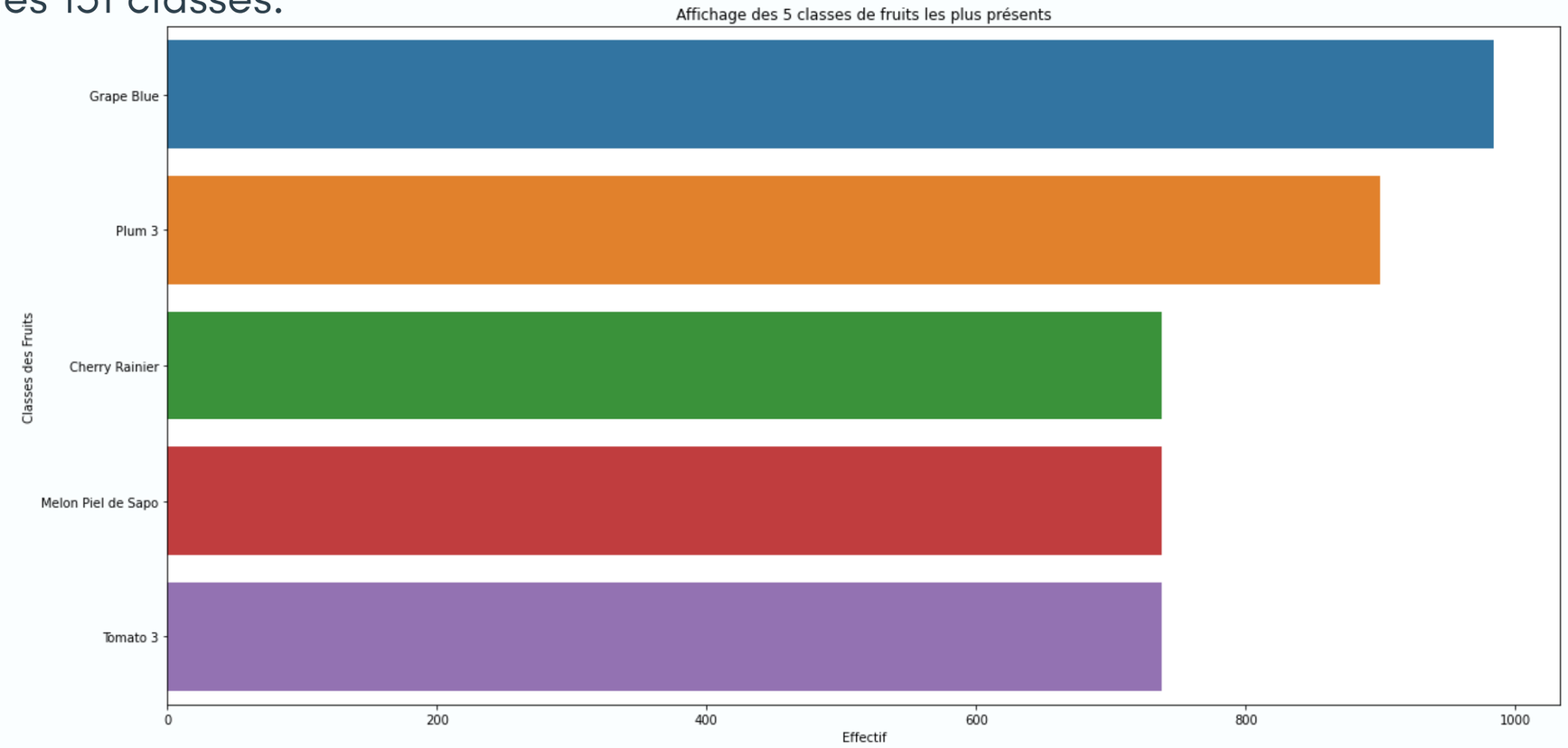


StrawBerry

Catégories de fruits

LES CATÉGORIES DE FRUITS LES PLUS REPRÉSENTÉS

Nous allons travailler sur les 5 catégories de fruits les plus représentés sur les 131 classes.



Contexte du Big Data

POURQUOI LE BIG DATA ?

Volume : trop important pour être stocké et/ou traité sur une seule machine avec des performances acceptables.

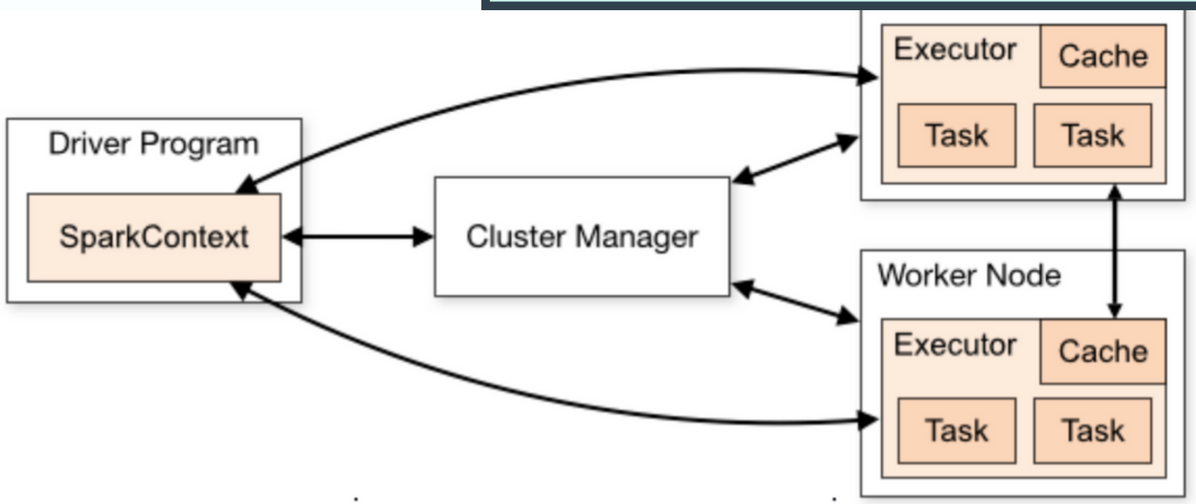
- Dépassement de la capacité de RAM
- Dépassement des capacités de stockage
- Vitesse à laquelle les données sont produites
- Large Variété de types de données

COMMENT RÉPONDRE À CES ENJEUX DU BIG DATA ?

Capacités de calcul : Traitement par calculs distribués (MapReduce)

- Diviser les opérations en micro opérations distribuables entre différentes machines, réalisables en parallèle
- Agréger les résultats sur une même machine

Répondre aux enjeux du Big Data



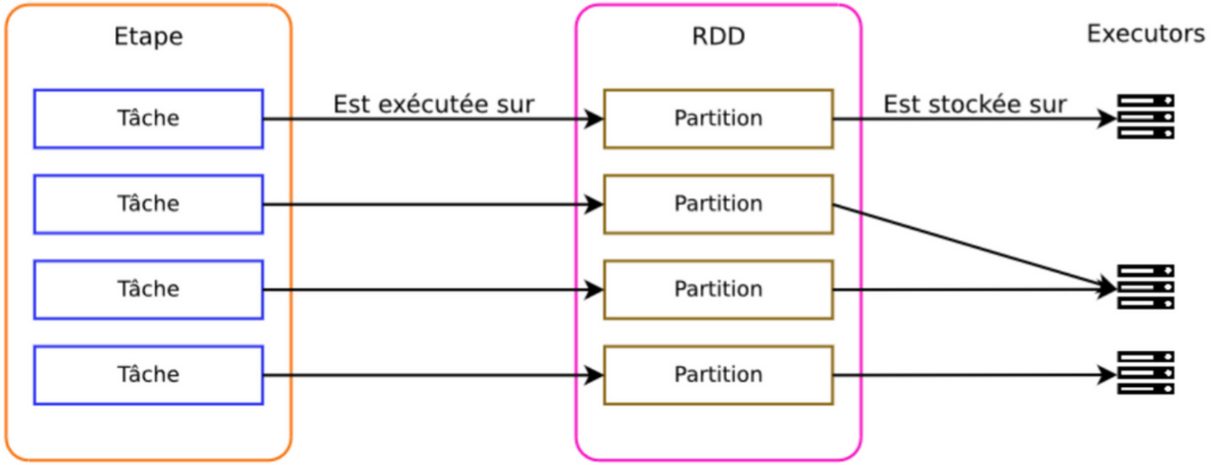
Application maître :
Configuration / Initialisation
Aggrégation des calculs

Cluster Manager :
Gestion des ressources
Distribution des calculs entre les workers

Workers :
Exécution des tâches en parallèle

Stockage : système de fichier distribué (ex : HDFS)

- Tolérance aux pannes
- Utilisation de Resilient Distributed Datasets (RDD)
- Division des données en partitions
- Duplication des données (3 machines par défaut)



Répondre aux enjeux du big data

OUTILS POUR RÉPONDRE À CES ENJEUX

- Nous allons utiliser le AWS le leader du cloud. Il s'agit du service cloud le plus utilisé et de loin. C'est aussi le premier à avoir lancé des services de cloud, dont il est en quelque sorte l'inventeur.
- A la base, AWS permet d'héberger des sites web sur des serveurs. Mais aujourd'hui, ça va plus loin : AWS nous aide à créer nos applications.
- Nous allons nous intéresser sur les deux services d'AWS qui permettent de répondre à notre problématique.
- **EC2 : Elastic Compute Cloud**

Ce service permet de gérer des serveurs sous forme de machines virtuelles dans le cloud. En gros, vous pouvez lancer des serveurs et faire ce que vous voulez avec. Vous avez accès à la ligne de commande, donc vous pouvez les piloter à distance.

- **S3 : Simple Storage Service**

Amazon S3 (Simple Storage Service) est un service de stockage et de distribution de fichiers. C'est une sorte de gros FTP (même s'il n'est pas basé sur FTP). Utilisez pour télécharger des fichiers ou pour y stocker des images.

Répondre aux enjeux du big data

CONFIGURATION DES SERVICES EC2

- EC2 est un service de calcul élastique dans le cloud.
- On dit "élastique" car il est possible d'en ajouter et d'en enlever en fonction de vos besoins. Si vous avez beaucoup de trafic, vous pouvez passer de 1 serveur à 2 serveurs par exemple, afin de mieux gérer ce nouveau trafic.
- Nous allons créer une instance, un serveur t2.medium, (CPU 2), t2.medium utilise un processeur évolutif Intel allant jusqu'à 3,3 GHz, avec un stockage de 30 G
- Même si vous souhaitez démarrer avec un serveur vierge, vous avez besoin d'un système d'exploitation installé au départ, nous avons utiliser le système d'exploitation Ubuntu Bionic version 18.04.

SE CONNECTER À MON SERVEUR

- On va utiliser le logiciel PuTTY pour se connecter en SSH.
- Utiliser Bash (la ligne de commande Linux) qui est désormais disponible aussi sous Windows
- De l'adresse de votre serveur (DNS ou IP)
- De la clé que vous avez téléchargée lors de la création du serveur
- Aprés avoir transformer la clé en .ppk à l'aide du logiciel "PuttyGen" fourni avec PuTTY.
- Pour pouvoir mettre des fichiers sur notre serveur EC2 nous allons utiliser le logiciel FileZilla.

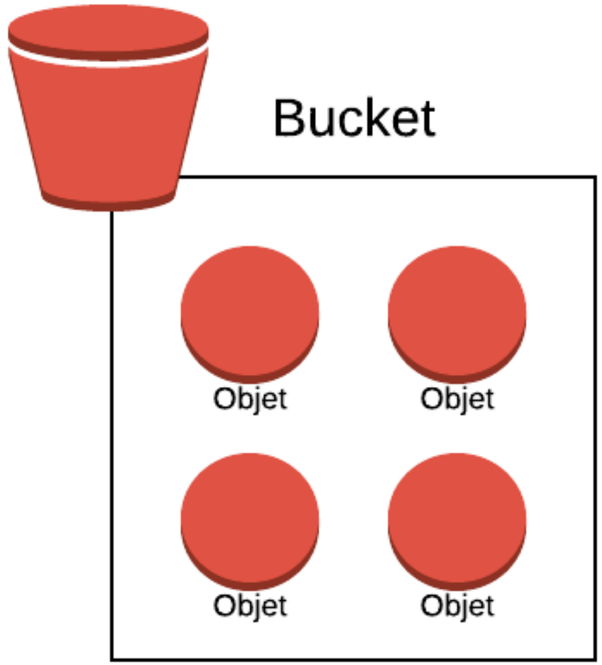
INSTALLATION DES DIFFÉRENTS LOGICIELS

- Pour pouvoir travailler en Python nous allons installer ANACONDA 3
- Nous allons installer le logiciel PySpark pour celà nous devons installer quelques logiciels avant de pouvoir l'installer
- Tout d'abord nous allons installer java-sdk 8
- Installation de Scala 2.11.12
- Installation de py4j qui permet aux programmes Python de s'exécuter dans un interpréteur Python d'accéder dynamiquement aux objets Java dans une machine virtuelle Java.
- Aprés toutes ces installations nous allons installer pyspark 3.0.3 hadoop 2.7, attention aux versions récentes certaines ne sont pas compatibles avec AWS et Java d'ou le choix de Java 8.

Répondre aux enjeux du big data

CONFIGURATION DES SERVICES S3

- Amazon Simple Storage Service (abrégé S3) est un service de stockage de données. En fait, il s'agit tout bêtement d'un moyen de stocker des fichiers sur Internet, qui est devenu très populaire.
- L'avantage de S3 on peut configurer facilement les droits d'accès pour chaque fichier. Qui peut lire, modifier et supprimer chaque fichier.
- Vos fichiers peuvent être versionnés : vous pouvez revenir à une version précédente à tout moment.



Répondre aux enjeux du big data

CONFIGURATION DES SERVICES S3

- Pour stocker et accéder à des fichiers sur S3, on peut y accéder en déposant directement des fichiers en utilisant l'option Charger
- Pour accéder à S3, on peut utiliser un SDK d'AWS, qui simplifie l'accès à AWS depuis votre code source.
- Le SDK pour python est boto3.
- Pour utiliser le SDK, il faudra se connecter avec une clé API. Pour obtenir cette clé, il faut impérativement créer un utilisateur avec le service IAM.
- Pour pouvoir transférer des fichiers de notre ordinateur à S3, nous allons installer AWSCLI, dans notre invité de commande nous tapons AWS CONFIG nous mettons les clés de l'utilisateur.
- Puis nous allons dans PyCharm pour mettre notre code python et envoyer nos fichiers qu'on veut stocker dans le S3.

Objectif du projet

Objectif : préparer les images pour le learning

Réduction de dimensions

Extraction d'information des images

Solutions envisageables

Egalisation histogramme et redimensionnement

Traitement d'image + extraction de features

ORB, SURF, SIFT, etc.

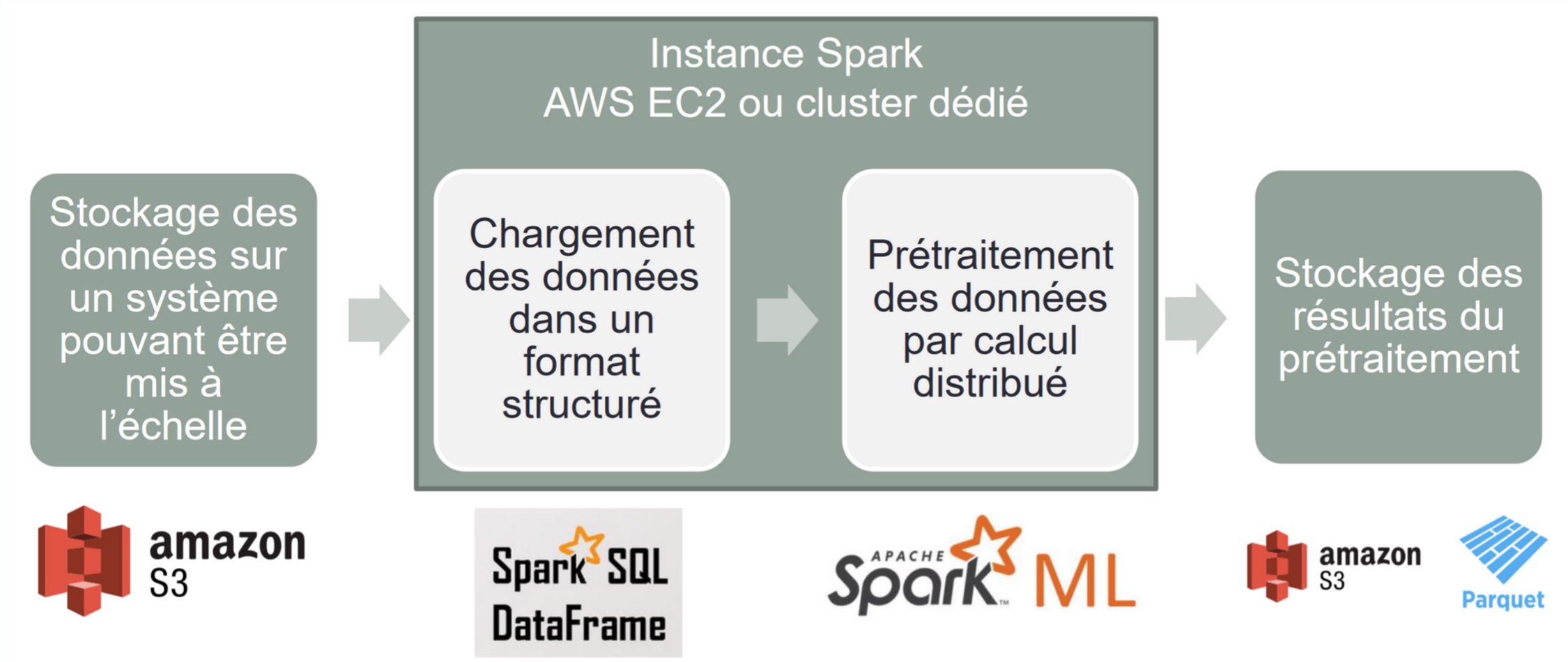
Algorithme préentraînés (Transfer Learning)

Traitement des images

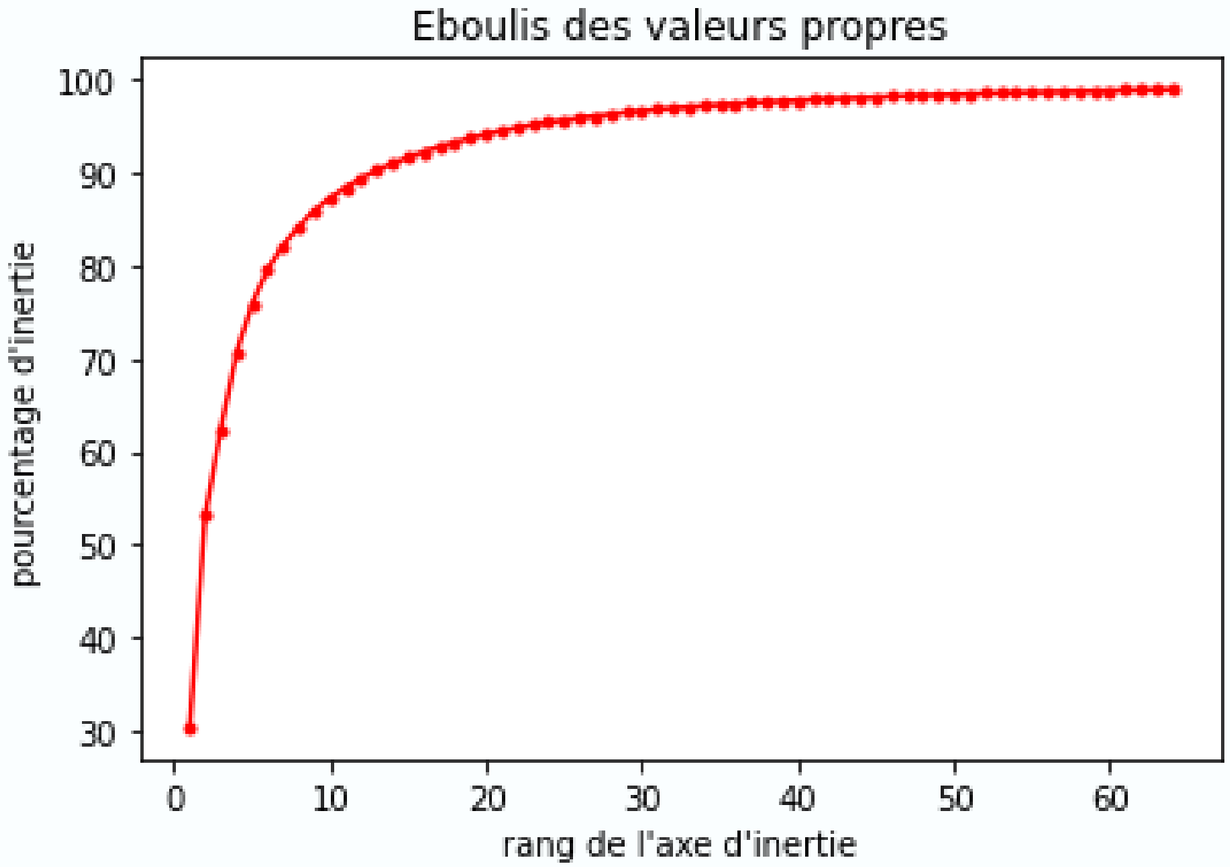
TRANSFORMATION DES IMAGES

- Nous allons redimensionner les images en les mettant au format (224,224,3)
- Les images étaient de la forme (100,100,3)
- Nous allons chercher les descripteurs de nos images en utilisant le transfert learning avec un modéle déjà entrainé sur d'autres images VGG16
- Nous allons enlever la dernière couche puis nous réalisons une prédiction en vue de trouver nos descripteurs.
- On trouve 25088 descripteurs pour une image, nous avons 4098 images

Décomposition de la problématique



Réalisation de l'ACP

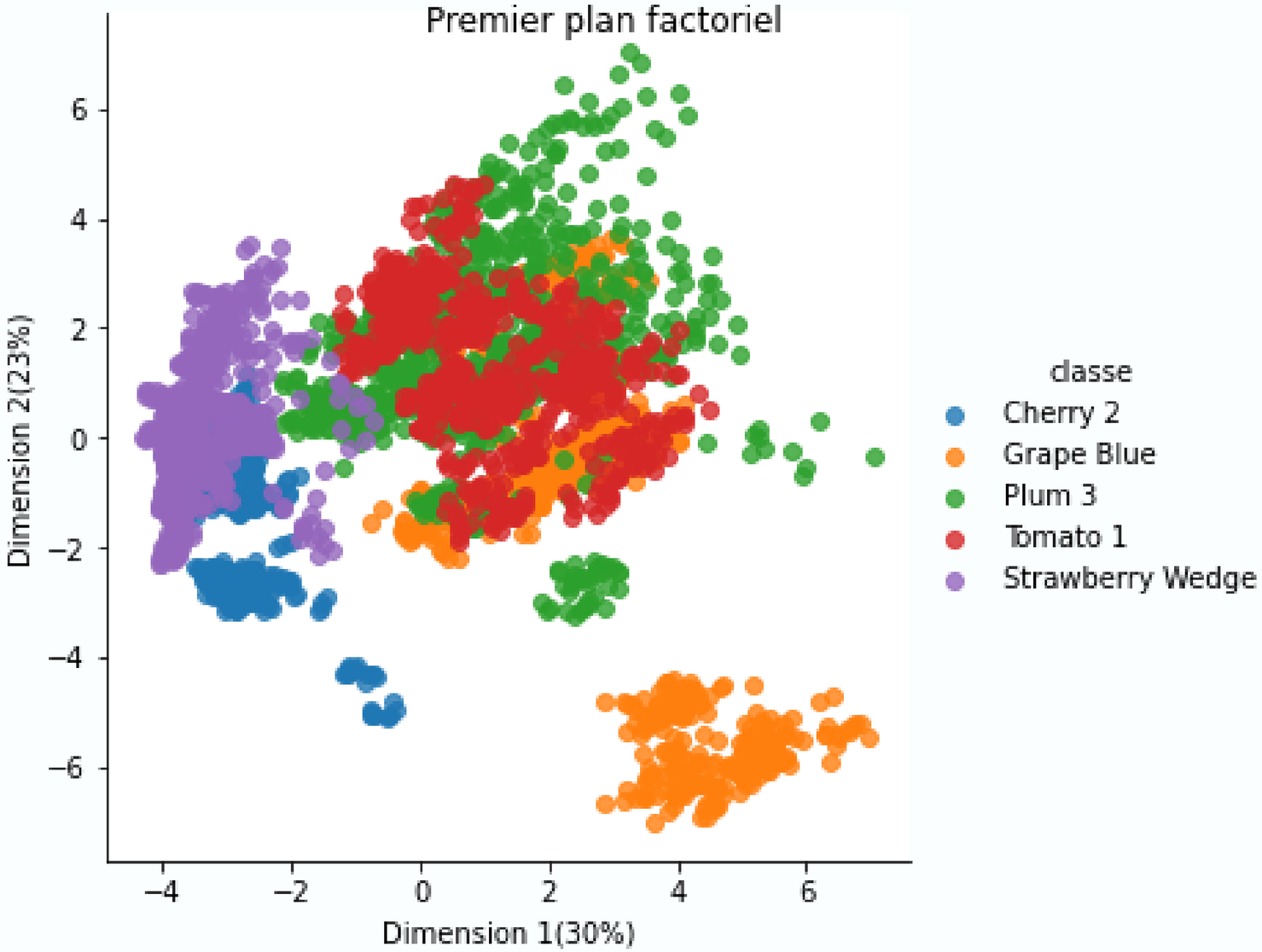


RÉALISATION DE L'ACP

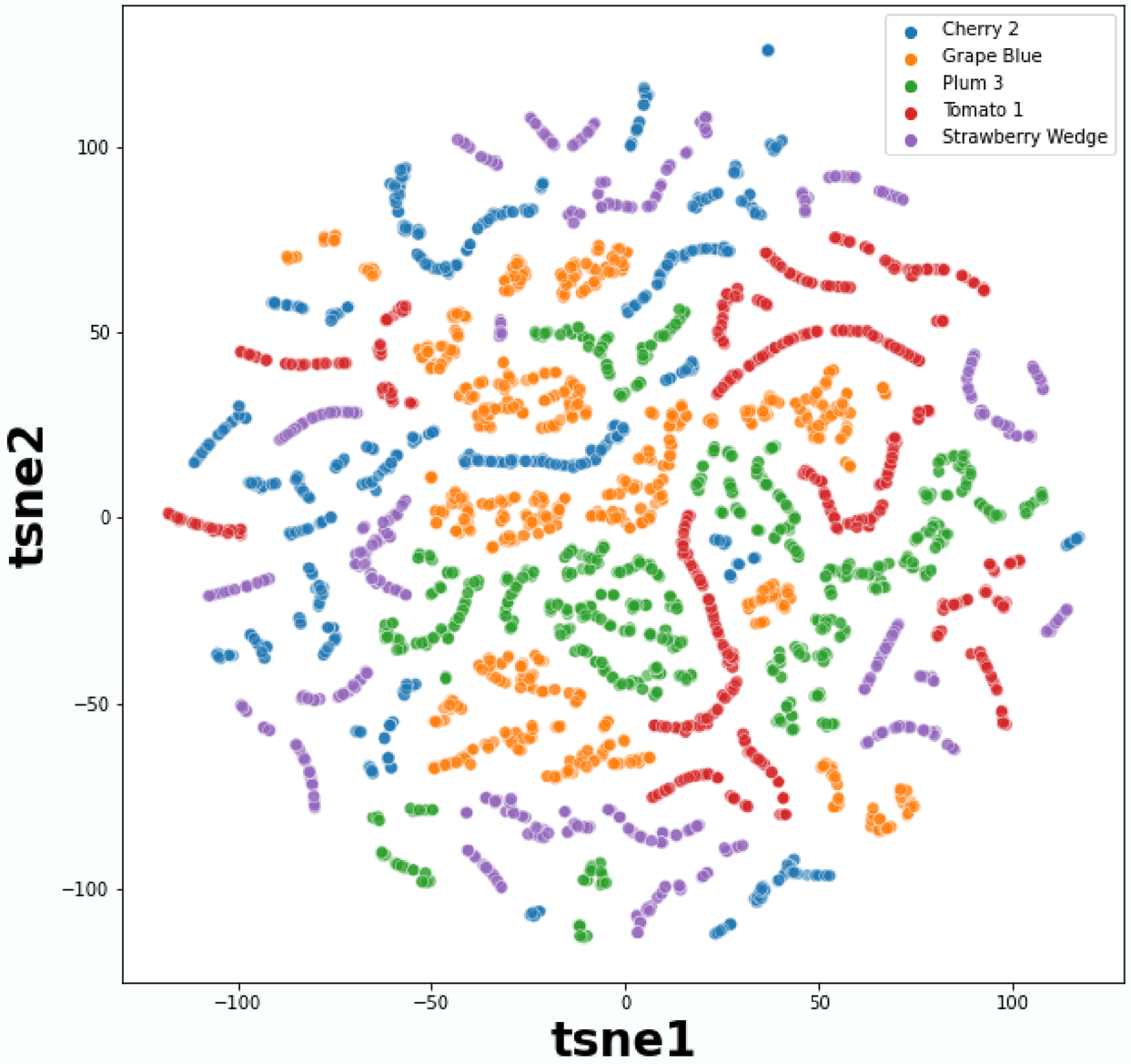
- Nous passons de 25088

à 64 features en gardant 99 % de variance expliquée.

Résultats de l'ACP en prenant cinq classes



Résultats de la TSNE en prenant cinq classes



Comment passer à l'échelle ?

PRÉSENCE DE PLUSIEURS IMAGES

- Aucune modification du code Spark/Python à apporter
- Le stockage des fichiers se fera sur S3
- Nous pouvons prendre une instance EC2 de plus grande capacité RAM/Processeur
- On peut aussi utiliser plusieurs instances et fixer une instance comme le principal

Conclusions

ENSEIGNEMENTS ET DIFFICULTÉS RENCONTRÉS

- Prise en main de Pyspark
- Découverte du format distribué
- Découverte de l'écosystème AWS
- Administration d'un serveur Linux par SSH
- Nombreuses problémes techniques sur les différentes versions de logiciels qui sont compatibles entre Spark, Java

FINAL WORDS

MERCI POUR VOTRE ÉCOUTE

SOULEYMANE CAMARA ETUDIANT
DATA SCIENTIST