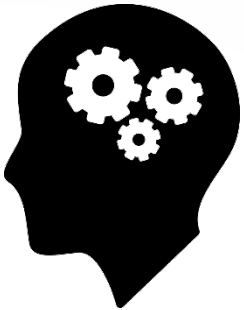
Projet 8 :

Déployez un modèle dans le cloud

Lecerf Defer Amandine

Compétences évaluées

- Paralléliser des opérations de calcul avec Pyspark
- Utiliser les outils du cloud pour manipuler des données dans un environnement Big Data
- Identifier les outils du cloud permettant de mettre en place un environnement Big Data



Plan

I. Contexte
II. Présentation des données
III. Pourquoi un environnement Big Data ?
IV. Traitement des images
V. Conclusion et Recommandations

Problématique + Objectifs

Contexte

- "**Fruits!**" :
  - solution récolte de fruits :
    - traitement adapté à chaque espèce de fruits
    - Robots cueilleurs intelligents
  - application mobile
    - prendre en photo un fruit
    - informations sur ce fruit
    - Classification d'image

Fruits!

Objectifs

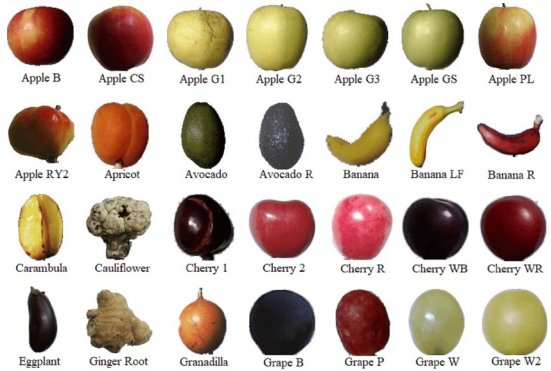
- Classification d'image :
  - Preprocessing images
  - Réduction de dimension

- Mettre en place un environnement Big Data

Présentation des données

Données

- **"Fruit 360"** contenant un total de 90 483 images
  - Training : 67 692 images
  - Test : 22 688 images



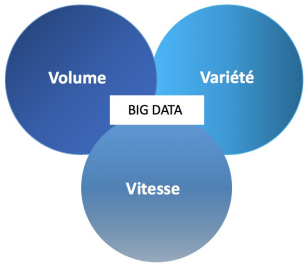
- Répartition images en 131 dossiers
  - 1 dossier = 1 fruit
  - Fruit représenté selon 3 axes
  - Images de 100X100 pixels au format : JPG RGB

- Certains fruits ont plusieurs variétés représentées

Pourquoi un environnement Big Data ?

Pourquoi du Big Data ?

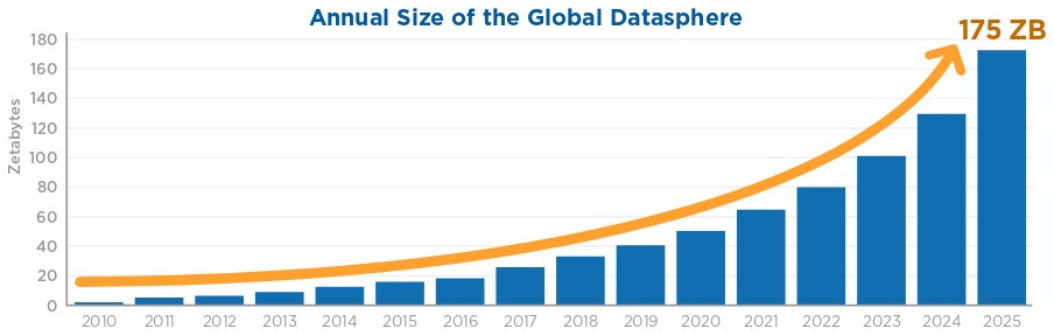
- Très puissant :
  - Analyse d'un grand nombre de données dans un temps acceptable
  - Analyse de données avec ajout progressif de données -> adaptation au besoin dans le temps
- Adaptation outils et méthodes utilisés pour une application small data
- Enjeux des 3V :



Pourquoi du Big Data ?

Volume : stockage données

dépassement capacité RAM + stockage



Exemple pour les données en santé :

Pourquoi du Big Data ?

Vitesse : rapidité production des données

traitement en temps réel sans paralyser le reste

Exemple pour les données en santé :

1 000 000 000 000 000 000 000 000 octets soit 1 Zettaoctet d'informations numériques produites par an

Variété : de plus en plus diversifiée

analyses adaptées à chaque type

Exemple pour les données en santé :

8,9 milliards
de feuilles de soin dans la base de données Sniiram

50000
app santé en 2020

Comment ?

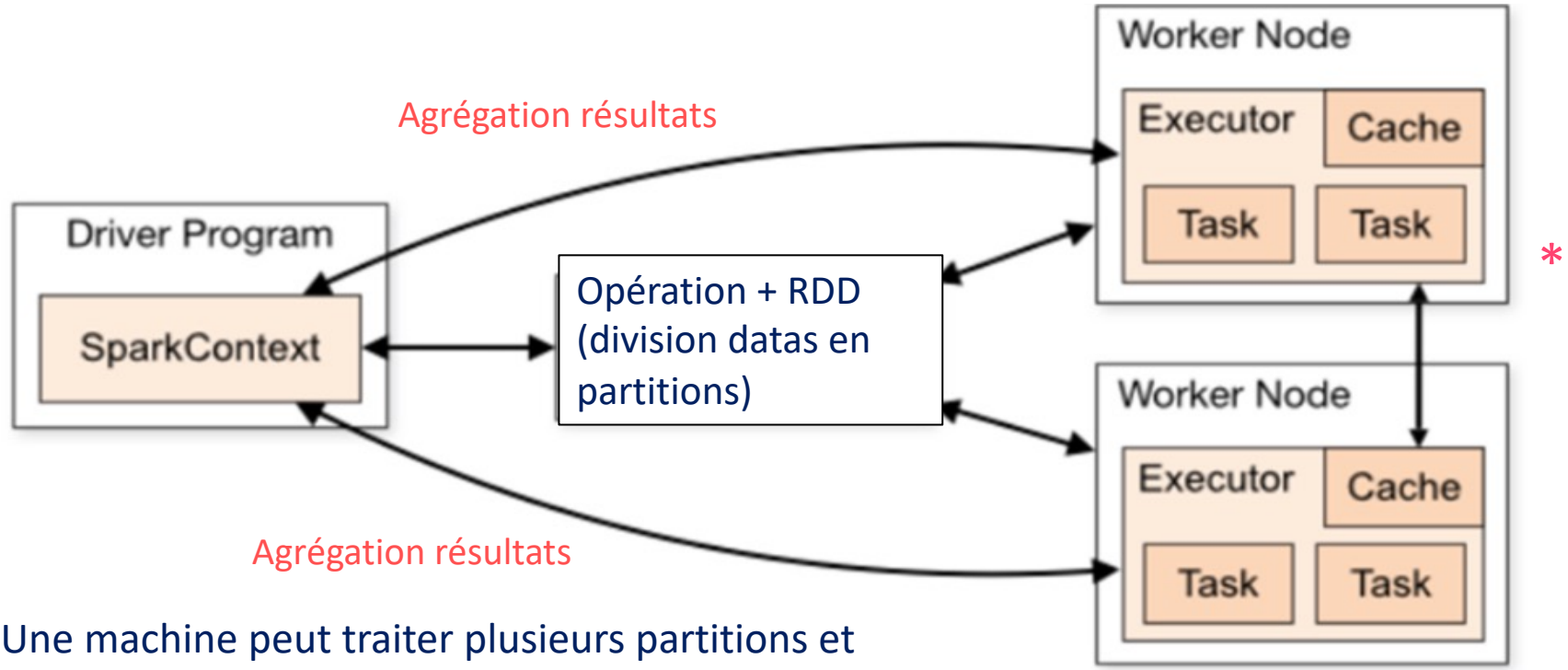
- Capacité de stockage
  - Cloud : Service Amazon
    - Facilite l'accessibilité, le partage et l'intégration entre les différents services

- Capacité RAM
  - Serveur Cloud : Service Amazon
    - Machine plus puissante qu'en local
    - Calculs infinis
    - faible coût

- Enjeux des 3V
  - Traitement des données par calculs distribués (MapReduce)
  - Outils dédiés au Big Data comme Spark (Pyspark)

Calculs Distribués ?

1 partition + opération = résultats



* Une machine peut traiter plusieurs partitions et résultat mis en cache

Chaîne de traitement des images dans le cloud

Plan Général

1. Base de données sur le Cloud

2. Environnement de travail

3. Traitement des images + Extraction Features

4. Réduction de Dimension par ACP

5. CSV des Features sur le Cloud

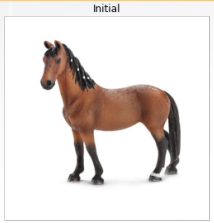
Etapes 1 et 2

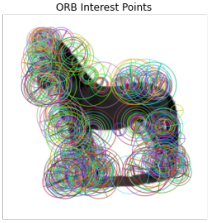
- Base de Données = 1310 images
  - 10 images par fruit + nom dossier : space to "_"
  - Sur le service de stockage illimité S3 d'Amazon

- Environnement de Travail
  - Machine virtuelle : service de calcul EC2 d'Amazon
  - Instance t2.xlarge avec un noyau Ubuntu Server 18.04
  - Clés IAM : lien avec S3
  - Spark 3.0.1 avec Hadoop 3.2 & Pyspark
  - Anaconda (python 3.8 + Jupyter Notebook)
  - Java 8, tensorflow

Comment faire l'extraction ?

- Read image avec boto3 :
- <u>Traitements :</u>
  - Image au niveau de gris
  - Amélioration contraste
  - Débruitage
  - filtre gaussien





- <u>Recherche de features :</u>
  - Par OpenCV (cv2) avec comme détecteur ORB
  - Features local : contour de l'objet
    - Peu précis : détermine moins de 1 000 features / image
    - Nombreuses étapes = technique lente
      - non adapté à du Big Data
      - **De nos jours : CNN (extraction features) = réseau neurone facile à entrainer**

Etape 3 : Spark

Dataframe Spark = dataframe pandas + calculs distribués

Etape 3 : Spark

RDD

Chargement des images depuis le S3 :

- Configuration Hadoop ("s3a" + clé IAM)
- spark.read()
- Forme Binaire
- Image définie par son url

Récupération nom du fruit associé à chaque image :

- Split url images
- Création colonne catégorie : pandas_udf

```
chargement effectué
Nombres d'images : 1310
+--------------------+--------------------+--------------+
|                path|             content|     categorie|
+--------------------+--------------------+--------------+
|s3a://p8-ald/Samp...|[FF D8 FF E0 00 1...|     Raspberry|
|s3a://p8-ald/Samp...|[FF D8 FF E0 00 1...|     Raspberry|
|s3a://p8-ald/Samp...|[FF D8 FF E0 00 1...|Pineapple_Mini|
|s3a://p8-ald/Samp...|[FF D8 FF E0 00 1...|Pineapple_Mini|
|s3a://p8-ald/Samp...|[FF D8 FF E0 00 1...|     Raspberry|
|s3a://p8-ald/Samp...|[FF D8 FF E0 00 1...|     Raspberry|
|s3a://p8-ald/Samp...|[FF D8 FF E0 00 1...|     Raspberry|
|s3a://p8-ald/Samp...|[FF D8 FF E0 00 1...|Pineapple_Mini|
```

Etape 3 : Spark

Traitement des images :

- preprocess_input de VGG16
- Conversion images RGB en BGR
- Traitement similaire à la base de données ImageNet

Extraction features des images:

- Récupération poids d'un modèle VGG16 pré-entrainé sur la base d'images riches (imagenet)
- 4608 features par images dans tableau

path	categorie	image_features
s3a://p8-ald/Samp...	Raspberry	[8.752506, 0.0, 0...
s3a://p8-ald/Samp...	Raspberry	[0.0, 0.0, 0.0, 0...
s3a://p8-ald/Samp...	Pineapple_Mini	[0.0, 0.0, 10.675...
s3a://p8-ald/Samp...	Pineapple_Mini	[0.0, 0.0, 26.388...
s3a://p8-ald/Samp...	Raspberry	[5.615052, 0.0, 2...
s3a://p8-ald/Samp...	Raspberry	[0.0, 0.0, 0.0, 0...
s3a://p8-ald/Samp...	Raspberry	[0.0, 0.0, 0.0, 0...
s3a://p8-ald/Samp...	Pineapple_Mini	[0.0, 0.0, 12.811...

Pourquoi vgg16 ?

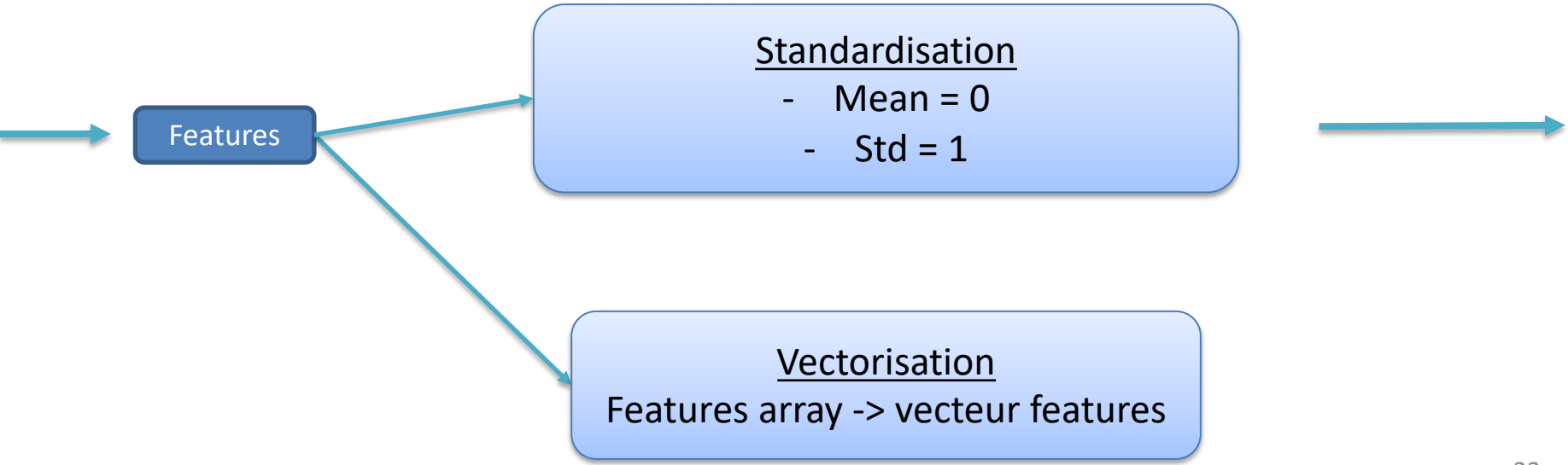
Entrainement long sur de grosses bases + facile accès

Pourquoi Imagenet ?

Bonne base pour classification fruit

Etape 3 : Spark

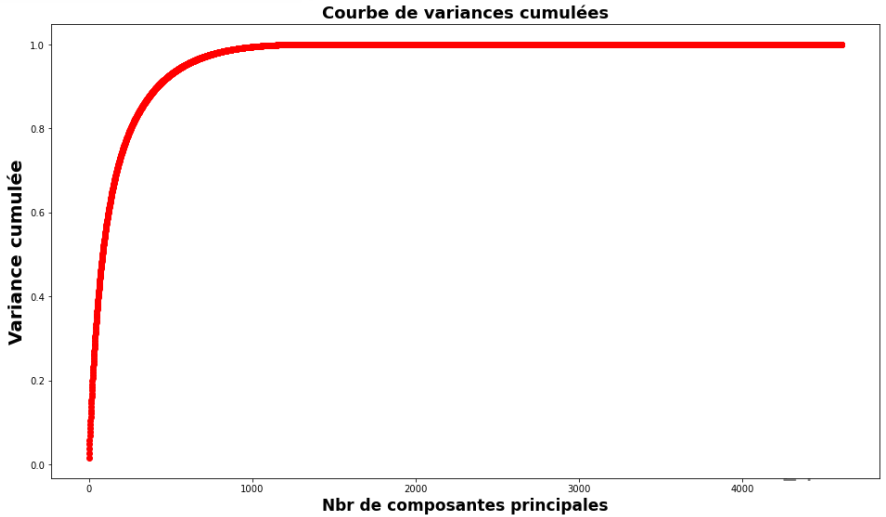
Ici utilisation des features pour une ACP alors qu'il aurait été possible de classifier les fruits.



Etape 4 : ACP (spark)

Réduction de Dimension :

- PCA : 95% de variance expliquée
- Passage de 4608 features à 585 features par image



Etape 5 : Sauvegarde

Sauvegarde dans le Cloud S3:

- Format CSV
- Chargement possible sous forme de dataframe pandas

	path	categorie	image_features_reduit
0	s3a://p8-ald/Sample/Raspberry/59_100.jpg	Raspberry	[-11.739343823504292,-19.282922837703882,0.536...
1	s3a://p8-ald/Sample/Raspberry/17_100.jpg	Raspberry	[-8.932846435115184,-14.729399119040165,-0.140...
2	s3a://p8-ald/Sample/Pineapple_Mini/249_100.jpg	Pineapple_Mini	[-19.489664206140713,-33.34627919533664,1.2217...
3	s3a://p8-ald/Sample/Pineapple_Mini/179_100.jpg	Pineapple_Mini	[-15.807275228826104,-27.638582449608528,0.513...
4	s3a://p8-ald/Sample/Raspberry/134_100.jpg	Raspberry	[-8.853180138955645,-17.935757508158666,0.4905...

Conclusion et Recommandations

Conclusion

- Comment mettre en place un environnement Big Data ?
  - AWS (EC2 , IAM, S3)
  - Spark
  - Administration serveur linux par SSH
- Concernant les 1310 images ?
  - Extraction des features par CNN VGG16 + Imagenet = 4608 features/image
  - ACP : 585 features/image
- Comment passer à l'échelle ?
  - Stockage fichier sur S3 : car stockage illimité
  - Aucune modification à apporter au script en spark/pyspark : adaptation automatique au volume de données à analyser

Recommandations

- Détermination modèle de transfert learning le mieux adapté
- Eviter Saturation RAM :
  - Augmentation nombre images
  - entraînement modèle pour classification
  - Evolution de l'infrastructure :
    - prendre EC2 de plus grande capacité (RAM) pour pouvoir analyser plus de données jusqu'à la totalité des images
    - Remplacement par un cluster Elastic Map Reduce avec plusieurs instances EC2 (1 maître + n esclaves)
    - Cependant, coût plus important

Fin de la présentation

Merci pour votre attention