Analyse

Fruits!

Déployez un modèle dans le cloud

Sofiane Mouhab

Octobre 2021

Sommaire

1 - Généralités

1.1 - Problématique

Votre start-up souhaite dans un premier temps se faire connaître en mettant à disposition du grand public une application mobile qui permettrait aux utilisateurs de prendre en photo un fruit et d'obtenir des informations sur ce fruit.

Pour la start-up, 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.

De plus, le développement de l'application mobile permettra de construire une première version de l'architecture Big Data nécessaire.

1.2 - Objectifs

Vous êtes donc chargé de développer dans un environnement Big Data une première chaîne de traitement des données qui comprendra le preprocessing et une étape de réduction de dimension.

Vous devrez tenir compte dans vos développements du fait que le volume de données va augmenter très rapidement après la livraison de ce projet. Vous développerez donc des scripts en Pyspark et utiliserez par exemple le cloud AWS pour profiter d'une architecture Big Data (EC2, S3, IAM), basée sur un serveur EC2 Linux.

La mise en œuvre d'une architecture Big Data sous (par exemple) AWS peut nécessiter une configuration serveur plus puissante que celle proposée gratuitement (EC2 = t2.micro, 1 Go RAM, 8 Go disque serveur).

1.3 - Condition de mise en oeuvre

Pour pouvoir sereinement réaliser ses objectifs, il nous faut donc diverses informations qui pourrait se trouver dans notre base de données.

Il y a donc 3 grandes interrogations :

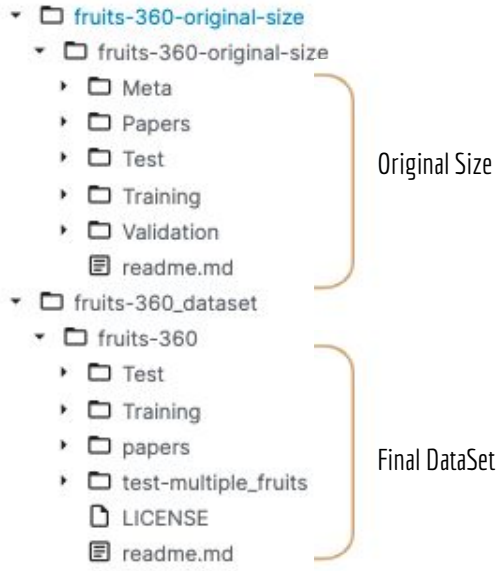
- Quels sont les outils à disposition pour mener cette étude ?
- Quel environnement technique peut fournir la puissance nécessaire
- A quels coûts s'attendre ?

Passons de suite à ce travail, en commençant par rapidement prendre connaissance des données en présence...

2 - Les données

2.1 - Description

Votre collègue Paul vous indique l'existence d'un jeu 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.



Original Size

Final DataSet

Test : 131 catégories

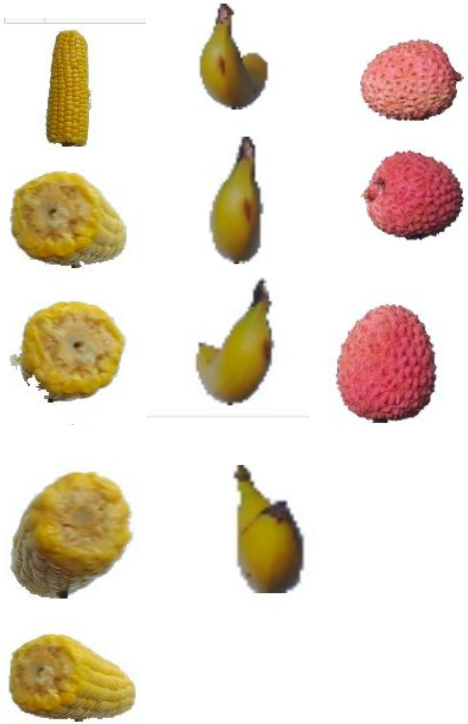
Test : 22688 images

Training : 131 catégories

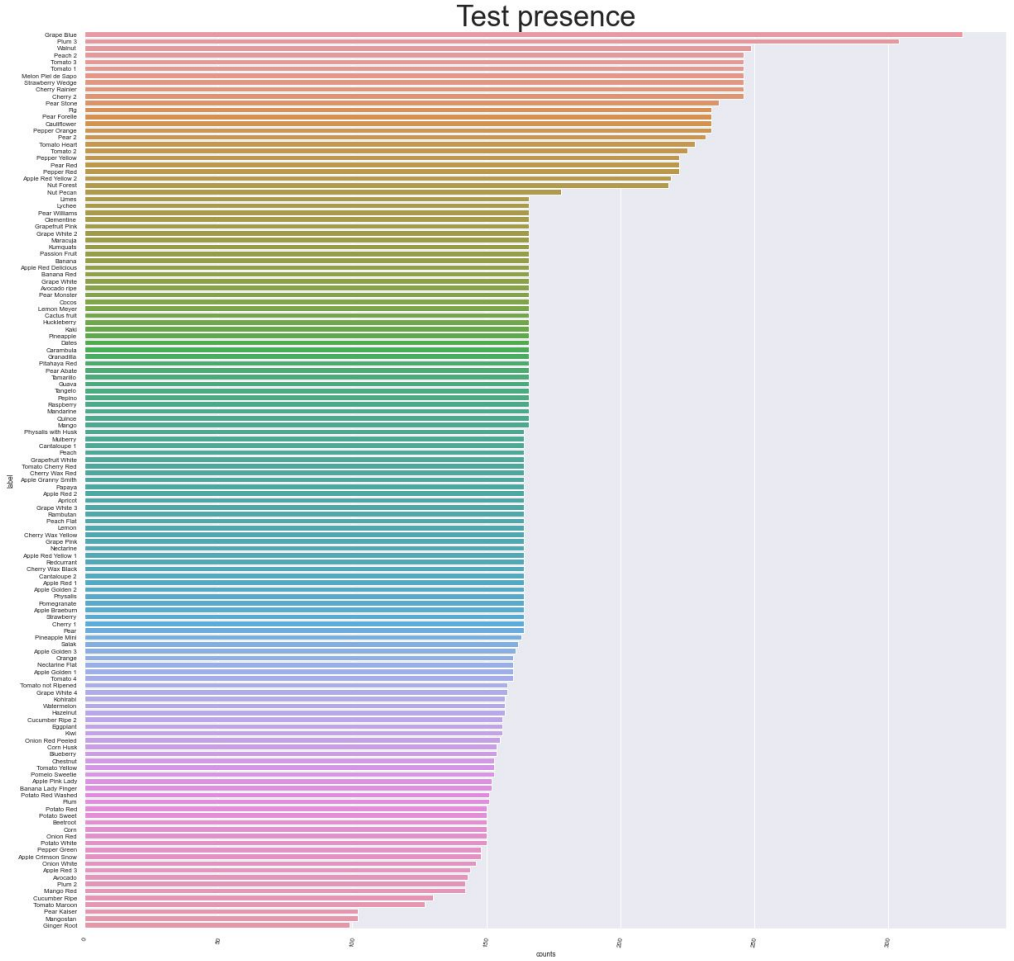
Training : 67692 images

2.2 - Visualisation

Exemple de photo



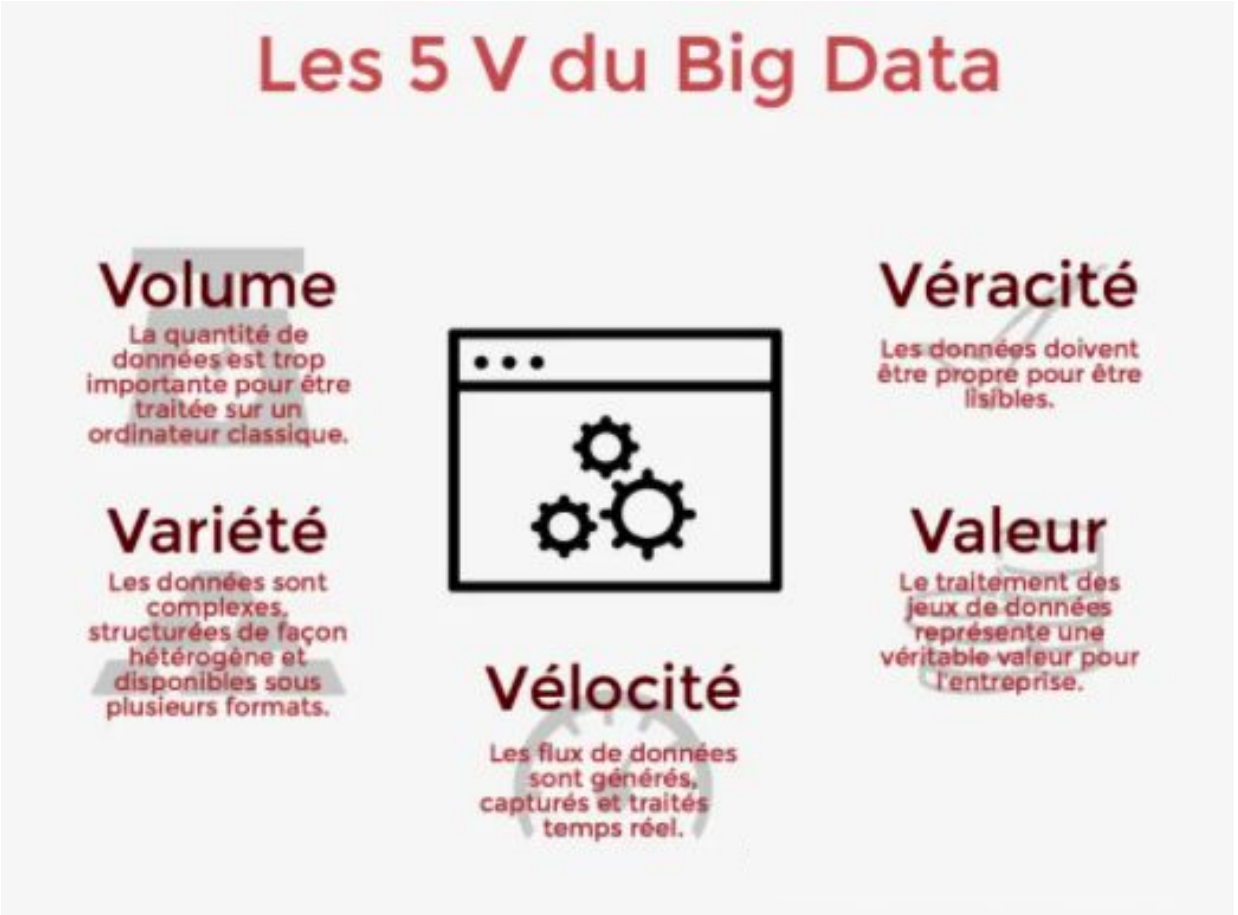
Nombre de photo par catégorie



3 - Le Big Data

3.1 - Présentation

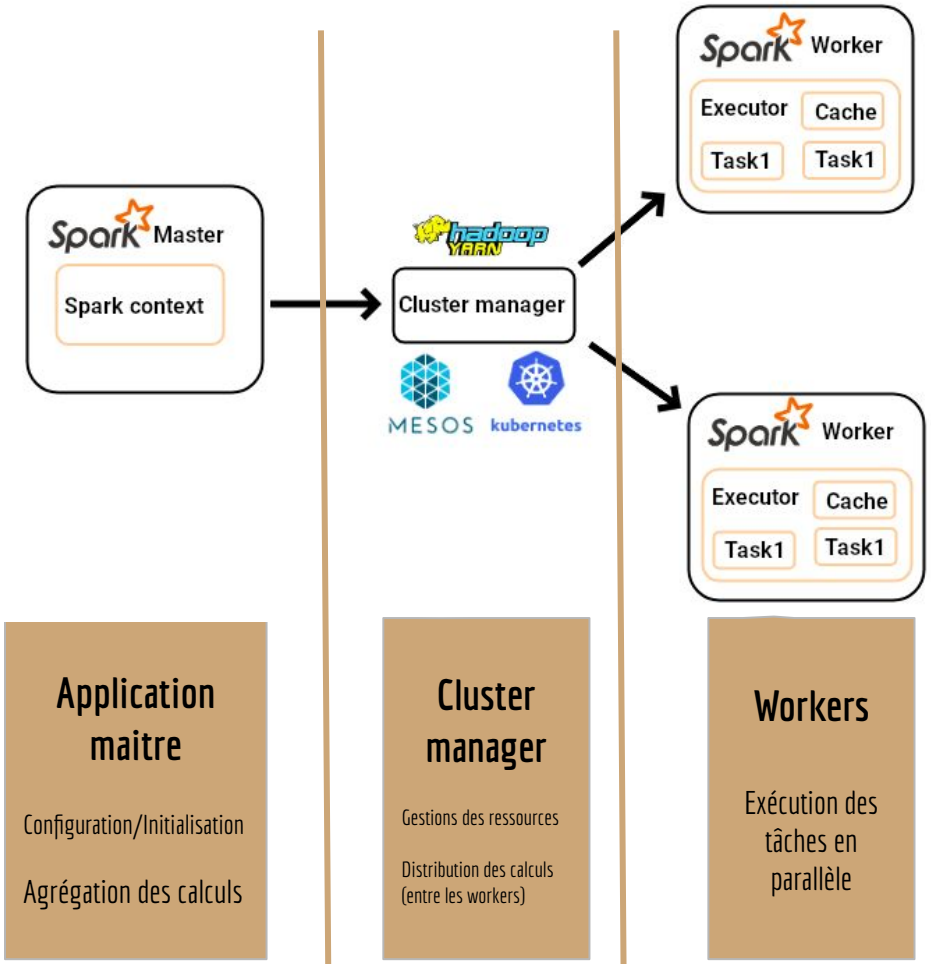
On parle de “Big Data” à partir du moment ou les données à analyser excède les capacités techniques (stockage/analyse) d’une simple machine.



3.2 - En pratique (1)

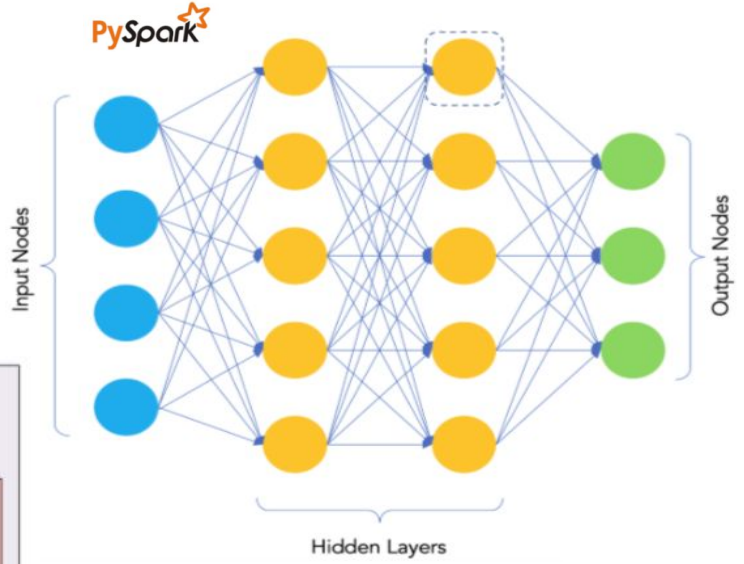
Pour rendre “digeste” le calcul,
on divise les opérations entre
plusieurs machines.

C’est ce qu’on appelle la
parallélisation,
puis on regroupe l’ensemble
sur une machine pour acceder
aux resultats.

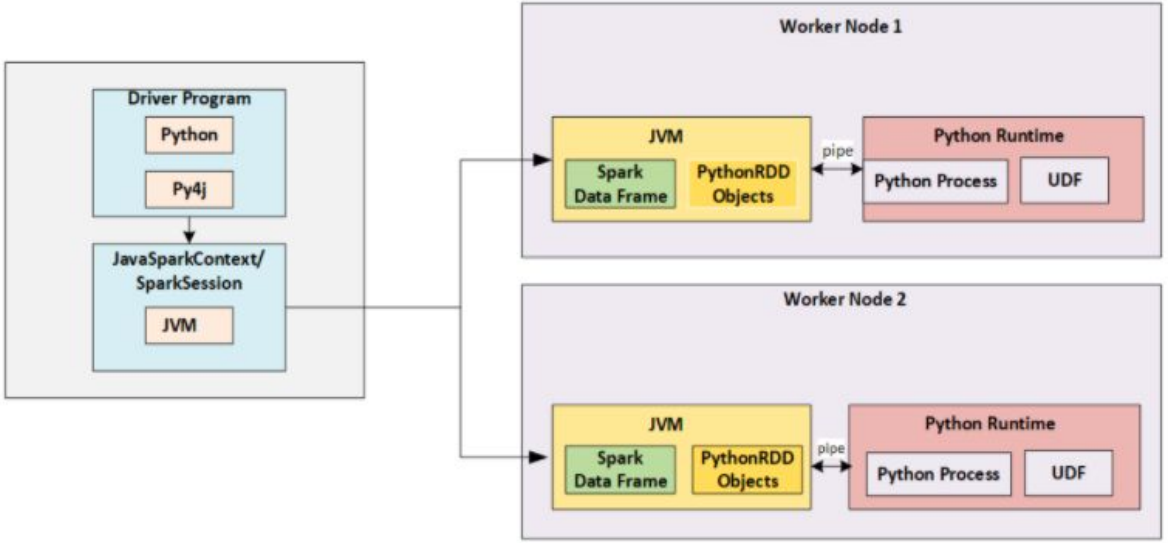


3.3 - En pratique (2)

Schéma du fonctionnement du transfer learning



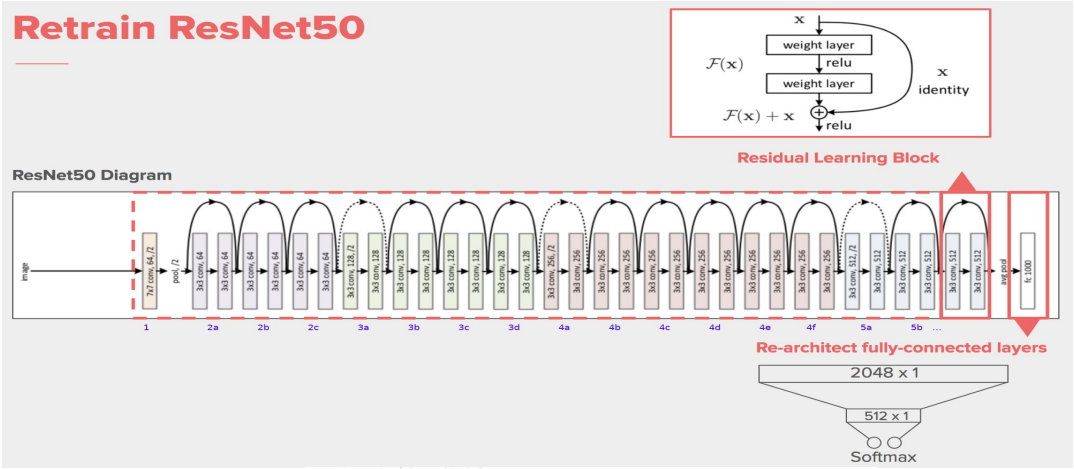
Fonctionnement en détail de PySpark avec Python UDF

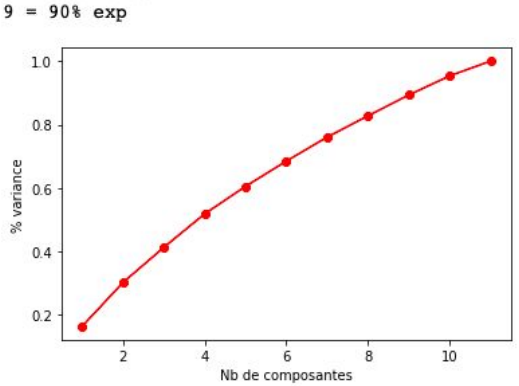


4 - L’architecture

4.1 - Prétraitement

- Stockage des données
- Création Spark Context
- Chargement path
- Isoler la target
- Application du modèle
- Créer UDF → Standard Scaler → PCA → Nettoyage → Export





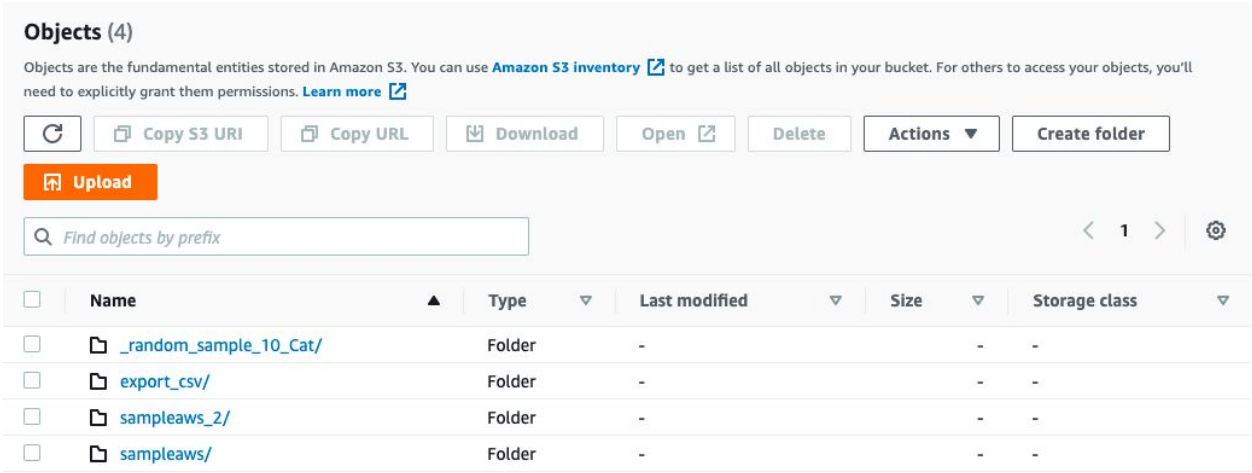
4.3 - AWS : Les services utilisés

4.3.1 - S3

Amazon S3 est un site d'hébergement de fichiers proposé par AWS

Amazon S3 a été conçu pour fournir une disponibilité de 99,99 %

Amazon S3 accepte des fichiers informatiques jusqu'à 5 téraoctets



4.3 - AWS : Les services utilisés

4.3.2 - Ec2

EC2 est un service permettant de louer des serveurs sur lesquels exécuter leurs propres applications web.

Ce service repose sur une infrastructure dite de “cloud” composée de plusieurs dizaines de milliers de serveurs informatiques répartis sur plusieurs sites dans le monde

Plusieurs AMI sont disponibles, parmis lesquelles:
UBUNTU / WINDOWS / REDHAT / DEBIAN / AMAZON LINUX...

Ainsi que des machines de puissance variables à déterminer par l’utilisateur

(image ci joint)

Enfin un stockage est intégré de même à déterminer par l’utilisateur

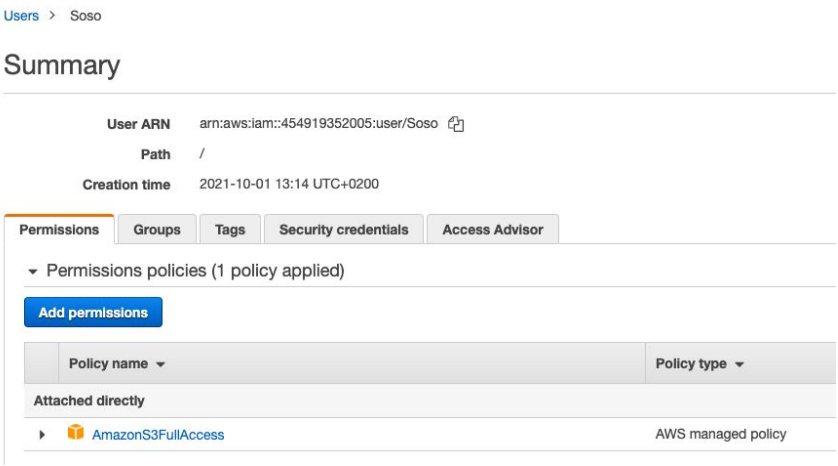
Type	vCPUs	Memory (GiB)
t2.nano	1	0.5
t2.micro Free tier eligible	1	1
t2.small	1	2
t2.medium	2	4
t2.large	2	8
t2.xlarge	4	16
t2.2xlarge	8	32
t3.nano	2	0.5
t3.micro	2	1

4.3 - AWS : Les services utilisés

4.3.3 - IAM

AWS Identity and Access Management (IAM), comme son nom l'indique, est le service de gestion des identités et des accès d'AWS.

En bref, lorsque vous essayer de réaliser une action quelconque sur AWS, vous devez passer par IAM qui vous identifiera, puis autorise l'action selon les droits qui vous ont été accordés par l'administrateur du compte.



5 - La mise en service

5.1 - En pratique (1)

Notebook :

- Installation des pkg
- Mise en place du code
- Creation d'un acces a distance

Stockage:

- Upload via interface Web
- Lecture avec Spark
- Enregistrement des données créés

Machine :

- T2.medium
- OS Ubuntu 18.04
- 30 Go

ubuntu

Configuration :

- Python 3.7
- Java 8
- Hadoop AWS

Service:

- Gestion des autorisations

5.2 - Le passage à l'échelle

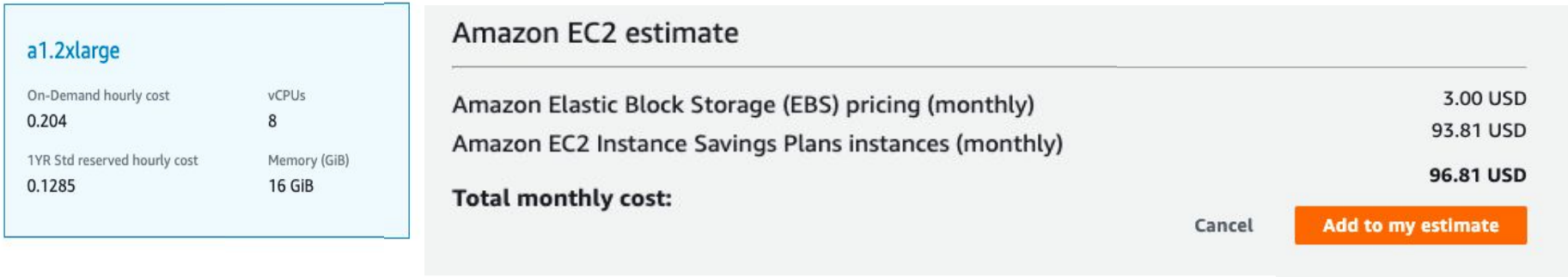
Pour le moment le code fonctionne avec peu d'image, sur une instance Ec2 peu puissante avec quelques photos

Du point de vue fonctionnel, le passage à l'échelle peut se faire aisément avec une instance plus puissante, le stockage S3 semble pouvoir supporter de grand volume de données.

Il faut donc envisager à court terme une machine plus puissante capable de gérer des données massives

5.3 - Les coûts

	Tarification du stockage
S3 Standard - stockage à usage général pour n'importe quel type de données. Cette classe de stockage est généralement utilisée pour les données à accès peu fréquent.	
50 premiers To/mois	0,024 USD par Go
450 To suivants/mois	0,023 USD par Go
Plus de 500 To/mois	0,022 USD par Go



6 - Conclusion

6.1 - Conclusion

Nous avons au cours de cet étude, vu les différentes possibilités qui nous permettent de déployer un modèle dans le cloud en particulier:

- 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

Néanmoins pour parfaire ce travail il nous semble important d'envisager d'autres solutions pour le futur...

6.2 - Perspective

Pour aller plus loin, nous pourrions dans un second temps :

- Etude précise des besoins et donc des coûts pour optimiser la mise à l'échelle
- Envisager d'autres solutions AWS (EMR et SageMaker par exemple)
- Tri des images et optimisation du prétraitement
- Tri des catégories et refactoring de certaines d'entre elles
- Implémenter une solution de monitoring efficace
- Optimiser les versions de chaques librairies pour une utilisation optimal
- Tester d'autres modèles pré-entraînés (InceptionV3 par exemple)

7 - Annexes

Notebook sur Ec2 - Process

0 - Import pkg

```
pyspark==3.0.1
tensorframes==0.2.9
tensorflow==2.4.1
Pillow==5.4.1
pandas==0.24.2
numpy==1.19.5
matplotlib==3.0.3
h5py==2.10.0
findspark==1.4.2
boto3==1.18.52
time: 3.19 ms (started: 2021-10-01 16:00:27 +00:00)
```

1 - Import image

path	modificationTime	length	content
s3a://sosop8/samp...	2021-10-01 12:08:36	5352	[FF D8 FF E0 00 1...]
s3a://sosop8/samp...	2021-10-01 12:08:36	5268	[FF D8 FF E0 00 1...]
s3a://sosop8/samp...	2021-10-01 12:08:36	4985	[FF D8 FF E0 00 1...]

only showing top 3 rows

2 - Ajout target

Ajout Target ======== 2 ========
(12, 5)

path	modificationTime	length	content	target
s3a://sosop8/samp...	2021-10-01 12:08:36	5352	[FF D8 FF E0 00 1...]	Plum
s3a://sosop8/samp...	2021-10-01 12:08:36	5268	[FF D8 FF E0 00 1...]	Plum
s3a://sosop8/samp...	2021-10-01 12:08:36	4985	[FF D8 FF E0 00 1...]	Plum

3 - Vérification target

Isol Target ======== 3 ========

target
Plum
Plum

only showing top 2 rows

4 - Convert > features

Creation UDF ======== 4 ========

path	features	target
s3a://sosop8/samp...	[0.0, 0.0, 0.0, 0...	Banana
s3a://sosop8/samp...	[0.0, 0.0, 0.0, 0...	Tomato

only showing top 2 rows

5 - Convert > Dense

Convert Dense ======== 5 ========

path	features	target	features_dense
s3a://sosop8/samp...	[0.0, 0.0, 0.0, 0...	Banana	[0.0,0.0,0.0,0.0,...
s3a://sosop8/samp...	[0.0, 0.0, 0.0, 0...	Tomato	[0.0,0.0,0.0,0.0,...

only showing top 2 rows

6 - Standard Scaler

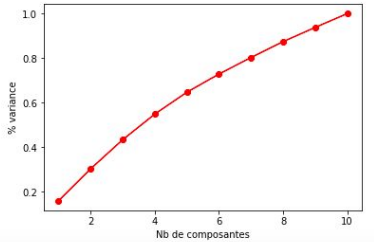
Standard Scal ======== 6 ========

path	features	target	features_dense	feat_scal
s3a://sosop8/samp...	[0.0, 0.0, 0.0, 0...	Banana	[0.0,0.0,0.0,0.0,...	[0.0,0.0,0.0,0.0,...
s3a://sosop8/samp...	[0.0, 0.0, 0.0, 0...	Tomato	[0.0,0.0,0.0,0.0,...	[0.0,0.0,0.0,0.0,...

only showing top 2 rows

7 - Définir la PCA

PCA ======== 7.0 ========
--- reduc
---> variance
8 = 90% exp



Notebook sur Ec2 - Process

7.1 - Création PCA

```
PCA ======== 7.1 ========
8
+--------------------+--------------------+------+--------------------+--------------------+--------------------+
|                path|            features|target|       features_dense|           feat_scal|         feat_reduit|
+--------------------+--------------------+------+--------------------+--------------------+--------------------+
|s3a://sosop8/samp...|[0.0, 0.0, 0.0, 0...|Banana|[0.0,0.0,0.0,0.0,...|[0.0,0.0,0.0,0.0,...|[-56.902647732996...|
|s3a://sosop8/samp...|[0.0, 0.0, 0.0, 0...|Tomato|[0.0,0.0,0.0,0.0,...|[0.0,0.0,0.0,0.0,...|[42.1564435132716...|
+--------------------+--------------------+------+--------------------+--------------------+--------------------+
only showing top 2 rows
```

8 - Nettoyage

```
Select Col ======== 8 ========
+-------+--------------------+--------------------+
| target|         feat_reduit|                path|
+-------+--------------------+--------------------+
| Banana|[-56.902647732996...|s3a://sosop8/samp...|
| Tomato|[42.1564435132716...|s3a://sosop8/samp...|
| Banana|[-106.56696673201...|s3a://sosop8/samp...|
|Avocado|[28.9114397177967...|s3a://sosop8/samp...|
|   Plum|[38.3317382628097...|s3a://sosop8/samp...|
+-------+--------------------+--------------------+
only showing top 5 rows
```

9 - Convert Target > Indexer

```
Indexer======== 9 ========
+------+--------------------+--------------------+----------+
|target|         feat_reduit|                path|target_Idx|
+------+--------------------+--------------------+----------+
|Banana|[-56.902647732996...|s3a://sosop8/samp...|       1.0|
|Tomato|[42.1564435132716...|s3a://sosop8/samp...|       3.0|
+------+--------------------+--------------------+----------+
only showing top 2 rows
```

10 - Nettoyage

```
Select Col - part 2 ======== 10 ========
+----------+--------------------+--------------------+
|target_Idx|         feat_reduit|                path|
+----------+--------------------+--------------------+
|       1.0|[-56.902647732996...|s3a://sosop8/samp...|
|       3.0|[42.1564435132716...|s3a://sosop8/samp...|
|       1.0|[-106.56696673201...|s3a://sosop8/samp...|
|       0.0|[28.9114397177967...|s3a://sosop8/samp...|
|       2.0|[38.3317382628097...|s3a://sosop8/samp...|
+----------+--------------------+--------------------+
only showing top 5 rows
```

11 - Upload sur S3

```
Upload Successful
time: 1.38 s (started: 2021-10-01 16:12:29 +00:00)
```

12 - Vérification sur S3

Name	Type	Last modified	Size
data_to_csv.csv	csv	October 6, 2021, 19:09:29 (UTC+02:00)	13.8 KB

13 - Vérification du CSV

	target_Idx	feat_reduit	path
0	1.0	[-56.902647732996215,-102.05007798009594,-18.3...	s3a://sosop8/sampleaws/Banana/r_322_100.jpg
1	3.0	[42.15644351327161,40.554428359570416,-49.8078...	s3a://sosop8/sampleaws/Tomato/0_100.jpg
2	1.0	[-106.56696673201677,-206.24992662813716,-67.7...	s3a://sosop8/sampleaws/Banana/r_79_100.jpg
3	0.0	[28.911439717796775,2.5067991804536245,64.6365...	s3a://sosop8/sampleaws/Avocado/r_307_100.jpg
4	2.0	[38.33173826280975,7.06829127888658,27.7109173...	s3a://sosop8/sampleaws/Plum/r_309_100.jpg

Linux - Étapes et commandes

Mise en place :

Création clé SSH sur IAM (via AWS)
Téléchargement de la clé SSH
Déplacement dans le dossier .ssh (MacOS)
Téléchargement Anaconda
$ wget https://repo.anaconda.com/archive/Anaconda3-2019.03-Linux-x86_64.sh
Installation Anaconda
$ bash Anaconda3-2019.03-Linux-x86_64.sh
Définir Anaconda en ressource
$ source /.bashrc
Mise à jour Apt
$ sudo apt-get update
Installer Java 8
$ install openjdk-8-jre-headless
Installer Scala
$ sudo apt install scala
Activer Anaconda
$ conda activate
Télécharger Spark/Hadoop
$ wget https://archive.apache.org/dist/spark/spark-3.0.1/spark-3.0.1-bin-hadoop2.7.tgz
Installer Spark/Hadoop
$ sudo tar -zxvf spark-3.0.1-bin-hadoop2.7.tgz
Mise à jour de Pip
$ python -m pip install --upgrade pip
Installation des packages
$ pip install findspark ....
Installation de AwsCli
$ sudo apt install awscli

Connexion à Ec2 :

Se placer dans le dossier de la clé SSH (MacOs)
$ cd .ssh
Connexion à l'instance
$ ssh -i "aws_key_t2_medium.pem"
ubuntu@ec2-13-37-105-123.eu-west-3.compute.amazonaws.com
Lancement du notebook
$ jupyter notebook

Connexion à Jupyter Notebook:

Se placer dans le dossier de la clé SSH (MacOs)
$ cd .ssh
Connexion à l'instance avec un localhost
$ ssh -i "aws_key_t2_medium.pem" -L 8000:localhost:8888
ubuntu@ec2-13-37-105-123.eu-west-3.compute.amazonaws.com

Commande Optionnel:

Option : Convertir de l'espace disque en Ram (Mémoire Virtuelle)
$ sudo dd if=/dev/zero of=/swapfile bs=128 M count=64
$ sudo chmod 600 /swapfile
$ sudo mkswap /swapfile
Option : Afficher la capacité disque Dur disponible
*$ df -hT /dev/xvda1*S

Extrait Spark Jobs (1)

Stages for All Jobs

Completed Stages: 311
Skipped Stages: 139

Completed Stages (311)

Page: 1 2 3 4 >

4 Pages. Jump to 1 . Show 100 items in a page. Go

Stage Id	Description		Submitted	Duration	Tasks: Succeeded/Total	Input	Output	Shuffle Read	Shuffle Write
449	showString at NativeMethodAccessorImpl.java:0	+details	2021/10/11 20:38:54	2 s	3/3			1685.0 KiB	
447	showString at NativeMethodAccessorImpl.java:0	+details	2021/10/11 20:38:53	2 s	1/1			573.5 KiB	
446	showString at NativeMethodAccessorImpl.java:0	+details	2021/10/11 20:38:26	27 s	8/8	38.3 KiB			5.7 MiB
445	collectAsMap at MulticlassMetrics.scala:61	+details	2021/10/11 20:38:26	17 ms	10/10			5.7 KiB	
444	map at MulticlassMetrics.scala:52	+details	2021/10/11 20:38:05	20 s	10/10			5.7 MiB	5.7 KiB
443	rdd at MulticlassClassificationEvaluator.scala:197	+details	2021/10/11 20:37:34	31 s	8/8	38.3 KiB			5.7 MiB
442	treeAggregate at RDDLossFunction.scala:61	+details	2021/10/11 20:37:33	23 ms	2/2			20.1 KiB	
441	treeAggregate at RDDLossFunction.scala:61	+details	2021/10/11 20:37:33	92 ms	10/10	4.4 KiB			20.1 KiB
439	treeAggregate at RDDLossFunction.scala:61	+details	2021/10/11 20:37:33	19 ms	2/2			20.1 KiB	
438	treeAggregate at RDDLossFunction.scala:61	+details	2021/10/11 20:37:33	97 ms	10/10	4.4 KiB			20.1 KiB
436	treeAggregate at RDDLossFunction.scala:61	+details	2021/10/11 20:37:33	33 ms	2/2			20.1 KiB	
435	treeAggregate at RDDLossFunction.scala:61	+details	2021/10/11 20:37:33	81 ms	10/10	4.4 KiB			20.1 KiB
433	treeAggregate at RDDLossFunction.scala:61	+details	2021/10/11 20:37:32	17 ms	2/2			20.1 KiB	

Extrait Spark Jobs (2)