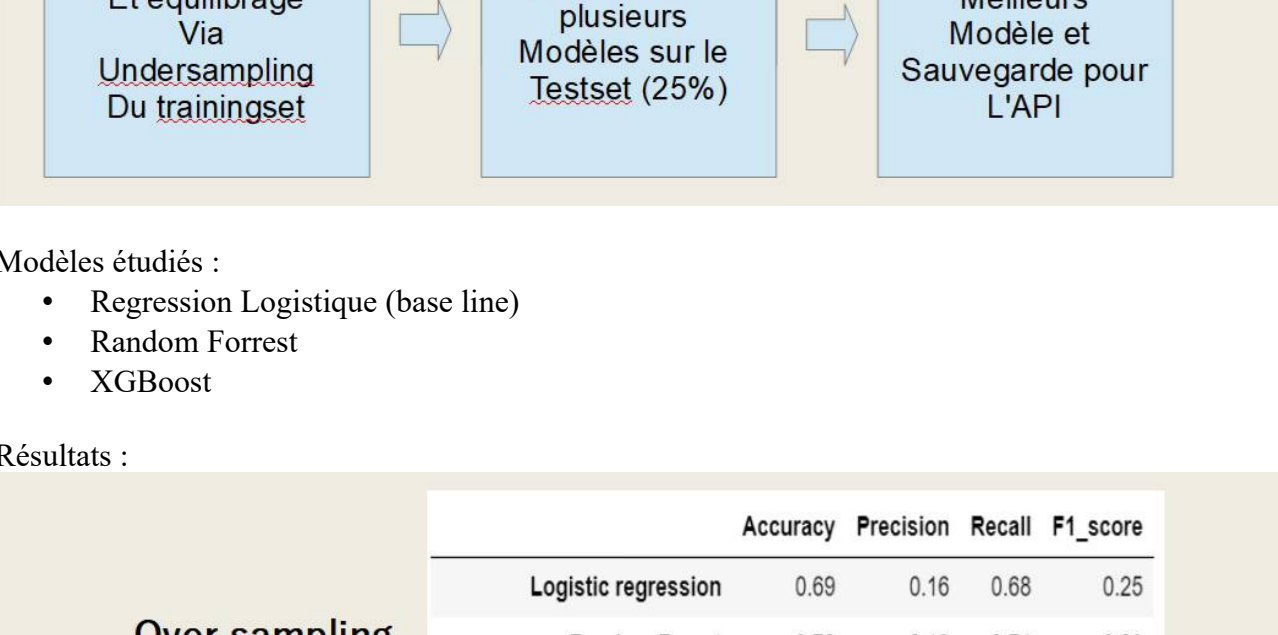


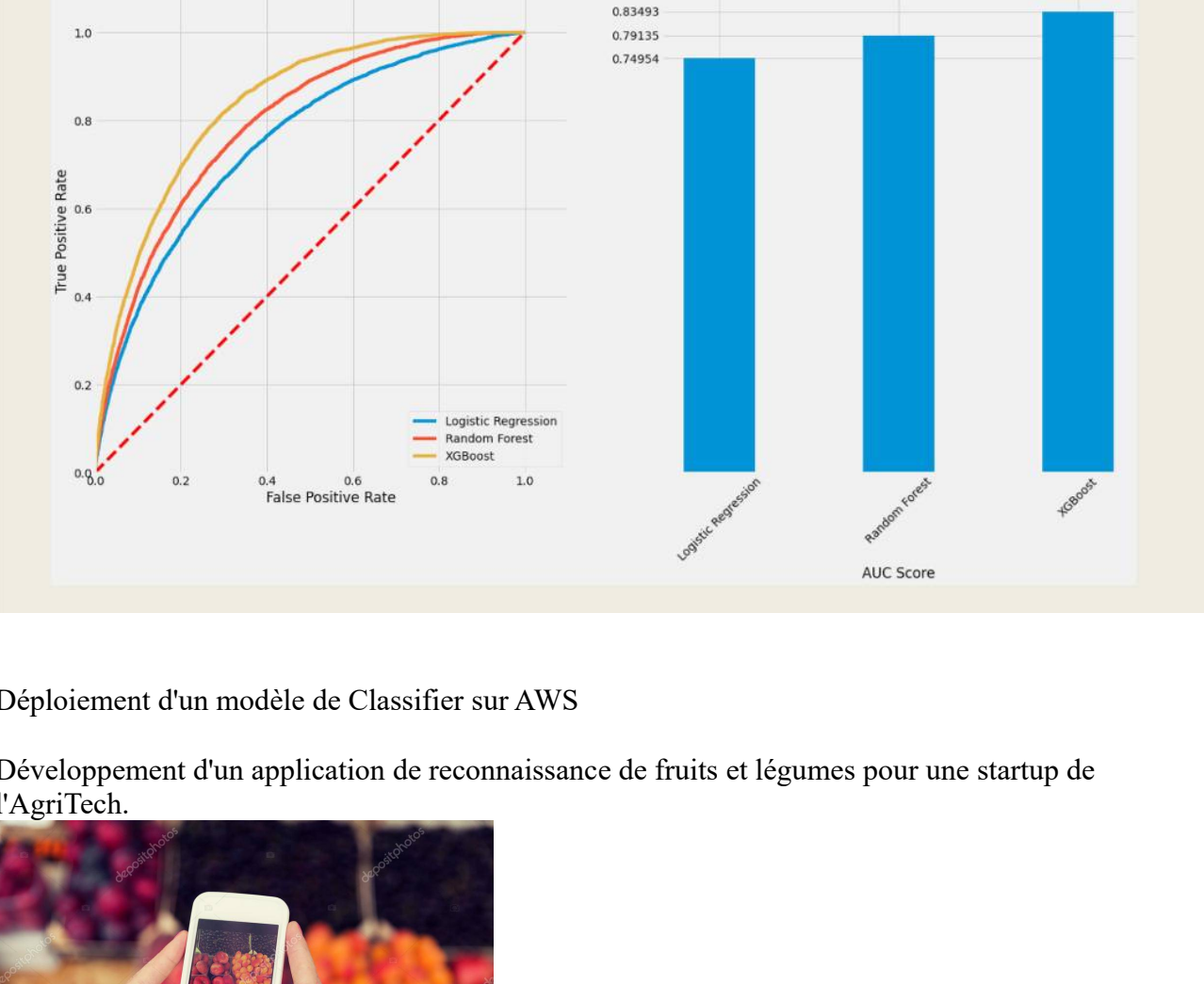
Démarche générale



Modèles étudiés :

- Regression Logistique (base line)
- Random Forrest
- XGBoost

Résultats :



Déploiement d'un modèle de Classifier sur AWS

Développement d'un application de reconnaissance de fruits et légumes pour une startup de l'AgriTech.



Objectifs : Mettre en place une architecture permettant de traiter de grands volume de données (Big Data, préprocessing, réduction de dimension, ...)

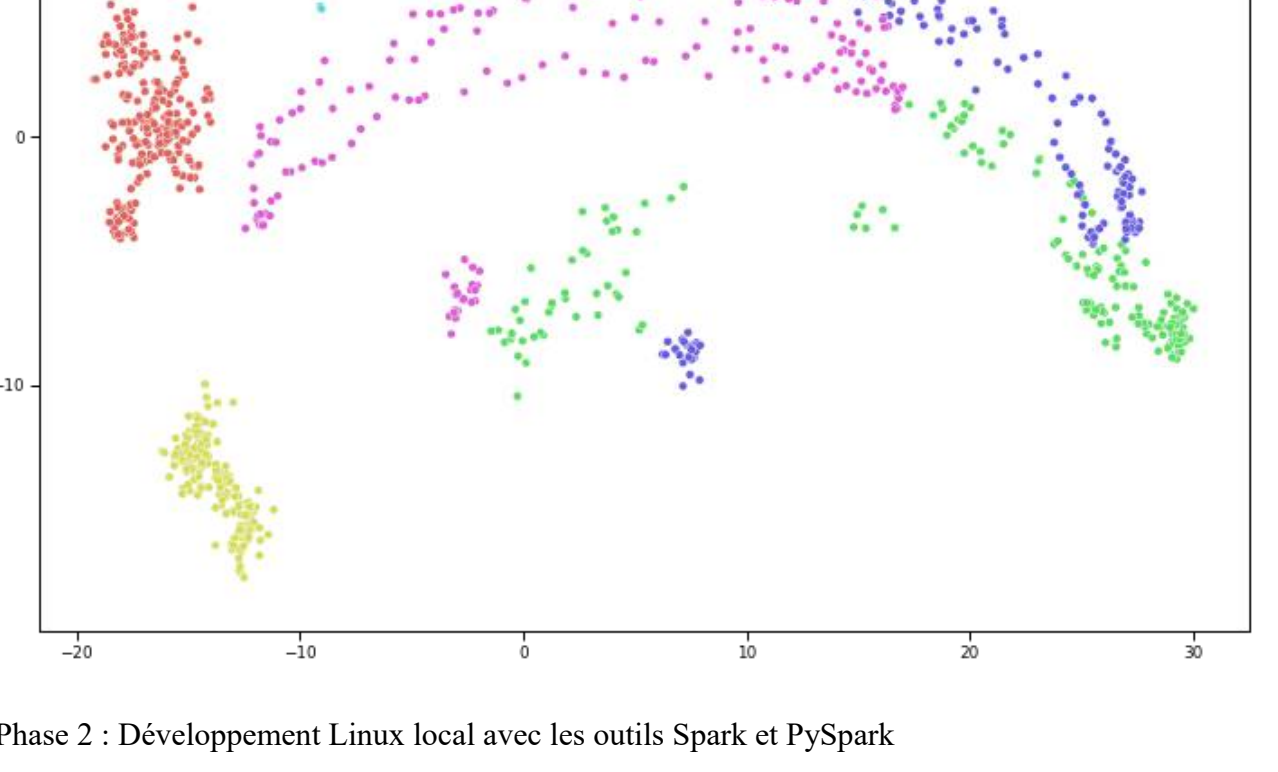
Moyens : Stockage et traitements dans le cloud (AWS)

Nous disposons d'un dataset contenant environ 90000 images de fruits et légumes

Répartition en 131 classes (dossiers)

- un dossier par fruit (classe)
- fruit photographié selon les 3 axes
- 100*100 pixels, .jpg, .rgb

Architecture générale



Implémentation de l'architecture sur Amazon Web Services (AWS)



Phasage du projet

Phase 1 : Développement Linux local avec les outils « small data »

Fabrication de 3 datasets réduits stockés sur S3, nettoyage des noms avec Detox

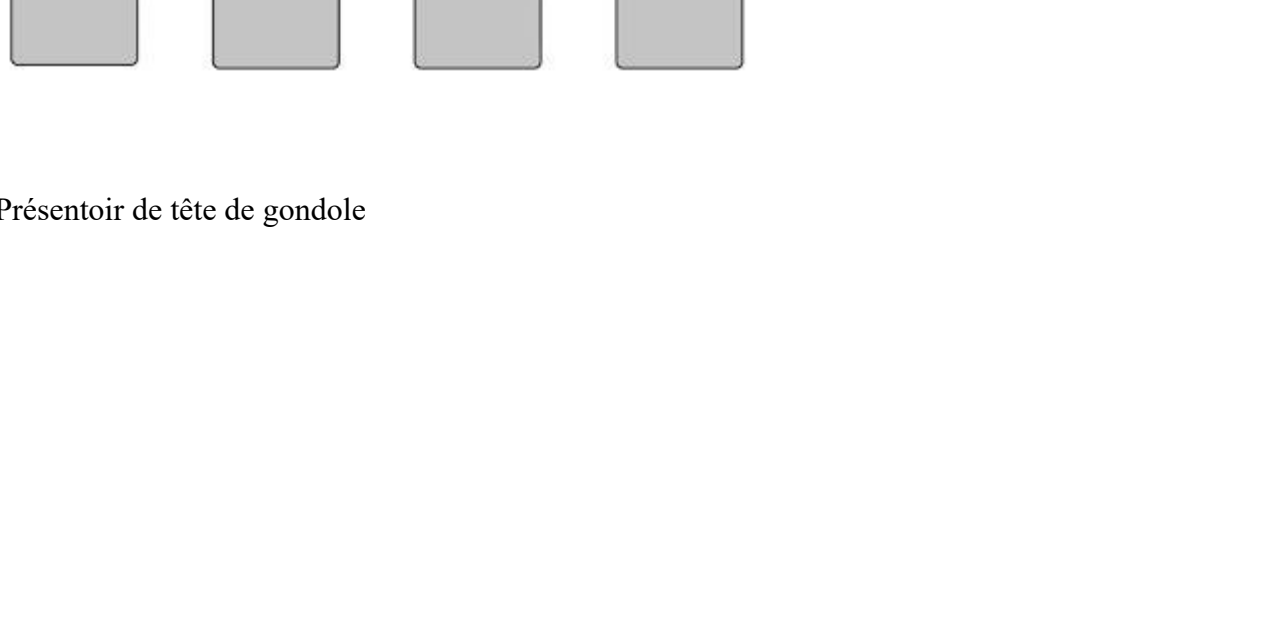
6 catégories, 200 images par catégories

131 catégories, 10 images par catégories

6 catégories, 10 images par catégories

Développement en Python, Pandas, OpenCV, d'un extracteur de features basé sur un CNN VGG16.

Entraînement d'un classifieur XGBoost pour avoir une idée du résultat (très bon). On obtient une très bonne séparation des classes (ici 6 catégories, projection du PCA sur le plan factoriel 0-1)



Phase 2 : Développement Linux local avec les outils Spark et PySpark

On utilise les 3 même datasets réduits stockés sur S3

```
spark=sparkSession.builder ...
```

```
sc=spark.sparkContext
```

```
df_img = spark.read.format ...
```

```
...
```

Phase 3 : Portage sur machine Amazon EC2

Mise en œuvre de Jupyter via SSH

Test sur machine t2.micro : échec

Test sur machine t2.xlarge, 16Go de RAM, 100Go DD : OK



Phase 4 : Cluster EMR

Transformation du Notebook Jupyter en script Python et Pyspark en utilisant au maximum les primitives PySpark

Exécution sur un cluster EMR



Présentoir de tête de gondole