

Analyse exhaustive sur les vols aux états-unis sur une jeu de données de 2015

Ghouibi Ghassen

Université Paris 8

ghassen.ghouibi@etud.univ-paris8.fr

20/01/2020

1 Introduction

- Définition
- Problématique
- Sujet

2 Traitement de données

- Hive importer les fichiers
- Hive exporter les fichiers
- Python — hive
- Résultat

3 Exemple de résultat obtenu

4 Conclusion

- Sources
- Questions

Introduction



Introduction

C'est quoi la différence entre une architecture centralisée et une architecture distribuée ?

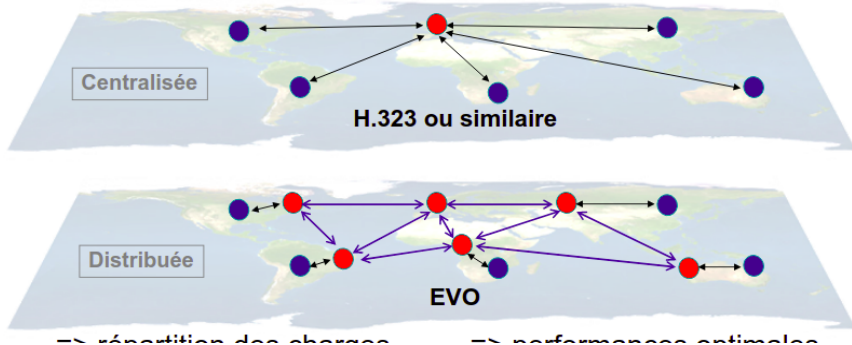


Figure: Les architectures distribuée et les architectures centralisée

- Est ce que l'aéroport peut avoir une influence sur le retard des vols ?

Dans cette présentation on étudier quelques étapes pour la réalisation du projet qui sera comment peut on se fier au aéroport pour prédire les retards.

EMR — Hive importer les fichiers

```
[hadoop@ip-172-31-16-115 ~]$ hive
Logging initialized using configuration in file:/etc/hive/conf.dist/hive-log4j2.properties Async: false
hive> create database flights2015;
OK
Time taken: 0.741 seconds
hive> use flights2015;
OK
Time taken: 0.033 seconds
hive> create external table if not exists flights2015.airlines
> (IATA_CODE string,
> AIRLINE string)
> ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';
OK
Time taken: 0.544 seconds
hive> load data inpath '/user/hadoop/airlines.csv' into table flights2015.airlines;
Loading data to table flights2015.airlines
OK
Time taken: 0.712 seconds
hive> select * from airlines;
OK
IATA_CODE AIRLINE
UA United Air Lines Inc.
AA American Airlines Inc.
US US Airways Inc.
F9 Frontier Airlines Inc.
B6 JetBlue Airways
OO Skywest Airlines Inc.
AS Alaska Airlines Inc.
NK Spirit Air Lines
WN Southwest Airlines Co.
DL Delta Air Lines Inc.
EV Atlantic Southeast Airlines
HA Hawaiian Airlines Inc.
MQ American Eagle Airlines Inc.
VX Virgin America
Time taken: 1.705 seconds, Fetched: 15 row(s)
hive> █
```

Figure: Importer les fichiers et les stocker dans des tables

EMR — Hive exporter les fichiers

```
Time taken: 0.778 seconds
hive> select * from class_departure_airports;
OK
MQ      272650   ORD      63130   PHL      9
HA      70030   HNL      30206   PPG     107
US     198715   CLT     44373   DSM      1
AA     648694   DFW     134270  LIT     10
UA     469829   ORD     59538   AGS      1
UX     56439   SFO     15940   TUL      1
EV     526249   IAH     58330   DAB      2
DL     800329   ATL     221705  LAN      4
WN     1157339  MDW     76350   PNS    962
NK     107171   FLL     11511   CRW     64
AS     158054   SEA     50004   CHS     26
OO     539545   DEN     44633   BWI      1
B6     245135   JFK     40455   ALB     44
F9     82735    DEN     21175   FAT      2
Time taken: 0.067 seconds, Fetched: 14 row(s)
hive>
```

Figure: Table du résultat

```
insert overwrite local directory '/home/hadoop/' row
format delimited fields terminated by ',' select * from
class_departure_airports; hadoop dfs -copyFromLocal
/home/hadoop/ s3://flights2015/input/
```


Python — représentation de table

```
flights={}
airports_best={}
airports_worst={}

with open('file.csv', newline='') as csvfile:
    reader = csv.DictReader(csvfile)
    for row in reader:
        print(row['IATA_CODE'], row['NUMBER_OF_FLIGHTS'])
        flights.update( {row['IATA_CODE'] :int(row['NUMBER_OF_FLIGHTS'])} )
        airports_best.update( {row['MOST LANDED AIRPORT'] :int(row['NB_OF_MLA'])} )
        airports_worst.update( {row['LESS LANDED AIRPORT'] :int(row['NB_OF_LLA'])} )

plt.figure(1)
plt.bar(range(len(flights)), list(flights.values()), align='center')
plt.xticks(range(len(flights)), list(flights.keys()))
plt.xlabel("Nom de compagnies aériennes")
plt.ylabel("Nombre de vol par an")
plt.title("Classement des compagnies aériennes par nombre de vol")

plt.figure(2)
plt.bar(range(len(airports_best)), list(airports_best.values()), align='center')
plt.xticks(range(len(airports_best)), list(airports_best.keys()))
plt.xlabel("Nom de l'aéroport")
plt.ylabel("Nombre de vol par an")
plt.title("Classement des aéroports avec le plus d'influence par nombre de vol")

plt.figure(3)
plt.bar(range(len(airports_worst)), list(airports_worst.values()), align='center')
plt.xticks(range(len(airports_worst)), list(airports_worst.keys()))
plt.xlabel("Nom de l'aéroport")
plt.ylabel("Nombre de vol par an")
plt.title("Classement des aéroports avec le moins d'influence par nombre de vol")

plt.show()
```

Figure: Code python pour la représentation graphique

Data Visualisation

La visualisation des données est un ensemble de méthodes de représentation graphique, en deux ou trois dimensions, utilisant ou non de la couleur et des trames.

Les moyens informatiques permettent de représenter des ensembles complexes de données, de manière plus simple, didactique et pédagogique.

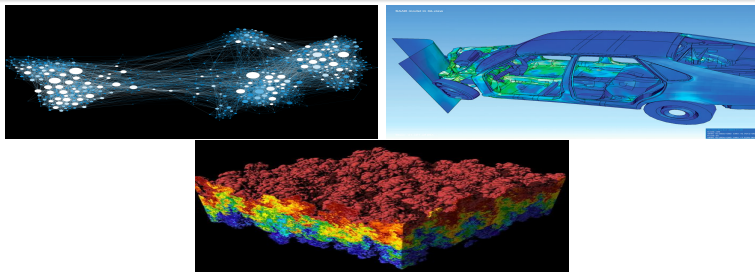


Figure: Exemple différents types de visualisation

Exemple de résultat obtenu

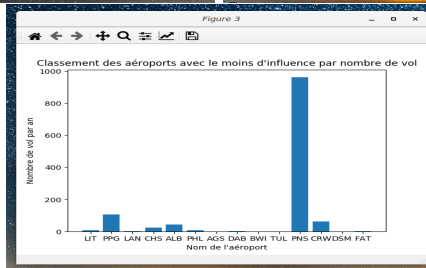
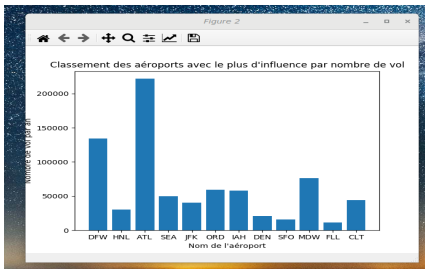
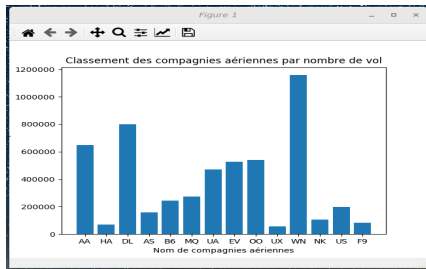


Figure: Résultat obtenu

Conclusion

Les résultats obtenus suite à ce projet montre les classements des aéroports et des compagnies aériennes sur des masses de données d'où l'importance des architecture distribuées pour effectuer ce genre d'opération qui sera un outils incontournable pour les Data Engineer et qu'on ne peut pas produire avec une seule machine.

Conclusion - Sources

-  <https://www.rosettahub.com/welcome>
-  <https://www.kaggle.com/usdot/flight-delays>
-  <https://aws.amazon.com/fr/>
-  <https://spark.apache.org/>
-  <https://www.youtube.com/watch?v=lZvs-YNk4V0>
-  <https://stackoverflow.com/>

Merci pour votre attention !