TP Final Big Data

Groupe de travail

M1 Dev Full Stack (PAR02)
Eleves:
- WU David (20230637)
- LESREL Thomas (20230012)
- HUCHARD Antoine (20230367)
- LOUBAYI MYSSIE Prince Thierry (20241434)

Liens de rendu

Lien vers le dépot GitHub : https://github.com/Orden14/tp-bigdata
Lien vers Databricks :
https://databricks-prod-cloudfront.cloud.databricks.com/public/4027ec902e239c93eaaa8714f173bcfc/1228008949750849/2146465264746755/5015045473982420/latest.html

Datasets

Nous avons décidé de travailler sur les trois datasets suivants :
- Salaire moyen par pays
- Nombre d'arrivées touristiques par pays
- Nombre de départs touristiques par pays

Introduction

Ce notebook est réalisé dans le cadre d'un devoir final pour un cours de Big Data. Il repose sur l'utilisation de Spark et Iceberg et suit un processus structuré autour des zones Bronze, Silver, et Gold pour gérer les données.

Objectif du projet

Le but de ce projet est d'analyser si le salaire moyen d'un pays influence :
- Le nombre de citoyens voyageant à l'étranger.
- Le nombre de touristes étrangers visitant le pays.

En d'autres termes, ce projet vise à évaluer les impacts du niveau de vie sur la mobilité internationale et le tourisme entrant.

Structure et étapes

1. Zone Bronze : Chargement des données brutes

D'abord, nous allons changer les trois datasets :

- Salaires moyens par pays.
- Nombre de départs internationaux.
- Arrivées touristiques.

Puis, nous allons les sauvegarder dans la zone Bronze pour la traçabilité.

```
salaries_path = "/FileStore/tables/dataset_salaire_moyen_par_pays-1.csv"
arrivals_path = "/FileStore/tables/dataset_arrivees_touristes_par_pays-1.csv"
departures_path = "/FileStore/tables/dataset_departs_internationaux_par_pays.csv"

# Chargement des données brutes
df_salaries = spark.read.format("csv").option("header", "true").option("inferSchema", "true").load(salaries_path)
df_arrivals = spark.read.format("csv").option("header", "true").option("inferSchema", "true").load(arrivals_path)
df_departures = spark.read.format("csv").option("header", "true").option("inferSchema", "true").load(departures_path)

# Sauvegarde des données brutes au format Iceberg
df_salaries.writeTo("spark_catalog.bronze.salaries").using("iceberg").createOrReplace()
df_arrivals.writeTo("spark_catalog.bronze.arrivals").using("iceberg").createOrReplace()
df_departures.writeTo("spark_catalog.bronze.departures").using("iceberg").createOrReplace()

df_salaries.printSchema()
df_arrivals.printSchema()
df_departures.printSchema()

display(df_salaries)
display(df_arrivals)
display(df_departures)
```

2. Zone Silver : Nettoyage et transformation

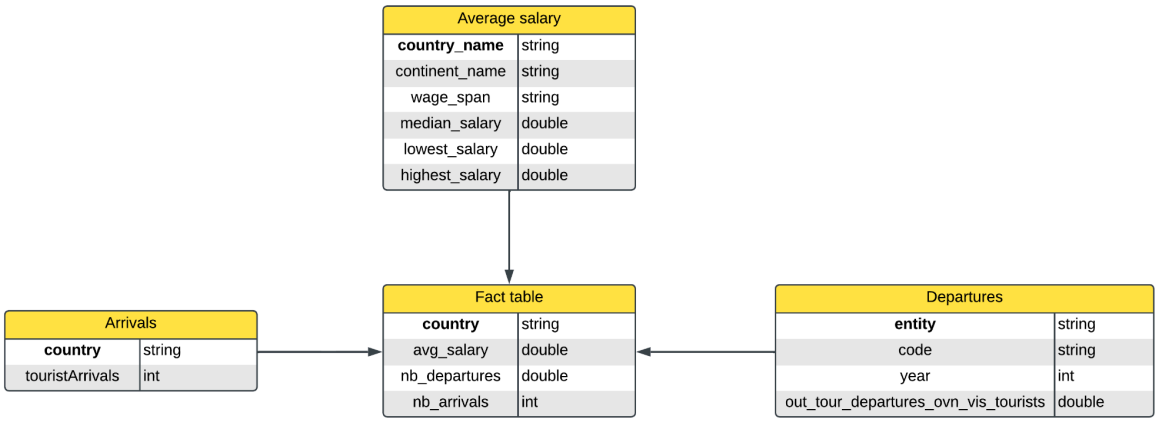
Il s'agit maintenant de filtrer les doublons, les champs non pertinents pour notre analyse, ainsi que de retirer les valeurs nulles non désirées.
Nous allons aussi harmoniser les colonnes puis sauvegarder ces données nettoyées dans la zone Silver.

```
7
```

```
# Nettoyage et harmonisation
df_departures = df_departures.withColumnRenamed("Entity", "country") \
    .withColumnRenamed("out_tour_departures_ovn_vis_tourists", "departures") \
    .drop("Code") \
    .filter(col("Year") == 2021) \
    .filter(col("departures").isNotNull())

df_salaries = df_salaries.withColumnRenamed("country_name", "country") \
    .withColumnRenamed("average_salary", "avg_salary") \
    .drop("continent_name", "wage_span") \
    .filter(col("avg_salary").isNotNull())

df_arrivals = df_arrivals.withColumnRenamed("country", "destination") \
    .withColumnRenamed("touristArrivals", "arrivals") \
    .filter(col("arrivals").isNotNull())

# Sauvegarde des données nettoyées au format Iceberg
df_departures.writeTo("spark_catalog.silver.departures").using("iceberg").createOrReplace()
df_salaries.writeTo("spark_catalog.silver.salaries").using("iceberg").createOrReplace()
df_arrivals.writeTo("spark_catalog.silver.arrivals").using("iceberg").createOrReplace()

df_salaries.printSchema()
df_arrivals.printSchema()
df_departures.printSchema()

display(df_salaries)
display(df_arrivals)
display(df_departures)
```

3. Zone Gold : Modélisation

Nous allons maintenant fusionner les tables nettoyées dans une table de faits (*fact table*), comme illustré dans le schéma MCD suivant.



Ainsi, nous préparons les données pour l'analyse en joignant les informations sur les salaires, départs, et arrivées.

```
# Création de la table des faits
df_fact = df_departures.join(df_salaries, on="country", how="inner") \
    .join(df_arrivals, df_departures["country"] == df_arrivals["destination"], how="left") \
    .select(
        col("country"),
        col("avg_salary"),
        col("departures").alias("nb_departures"),
        col("arrivals").alias("nb_arrivals")
    )

# Sauvegarde en Iceberg
df_fact.writeTo("spark_catalog.gold.fact_table").using("iceberg").createOrReplace()

# Affichage des résultats pour vérification
display(df_fact)
```

Export des résultats

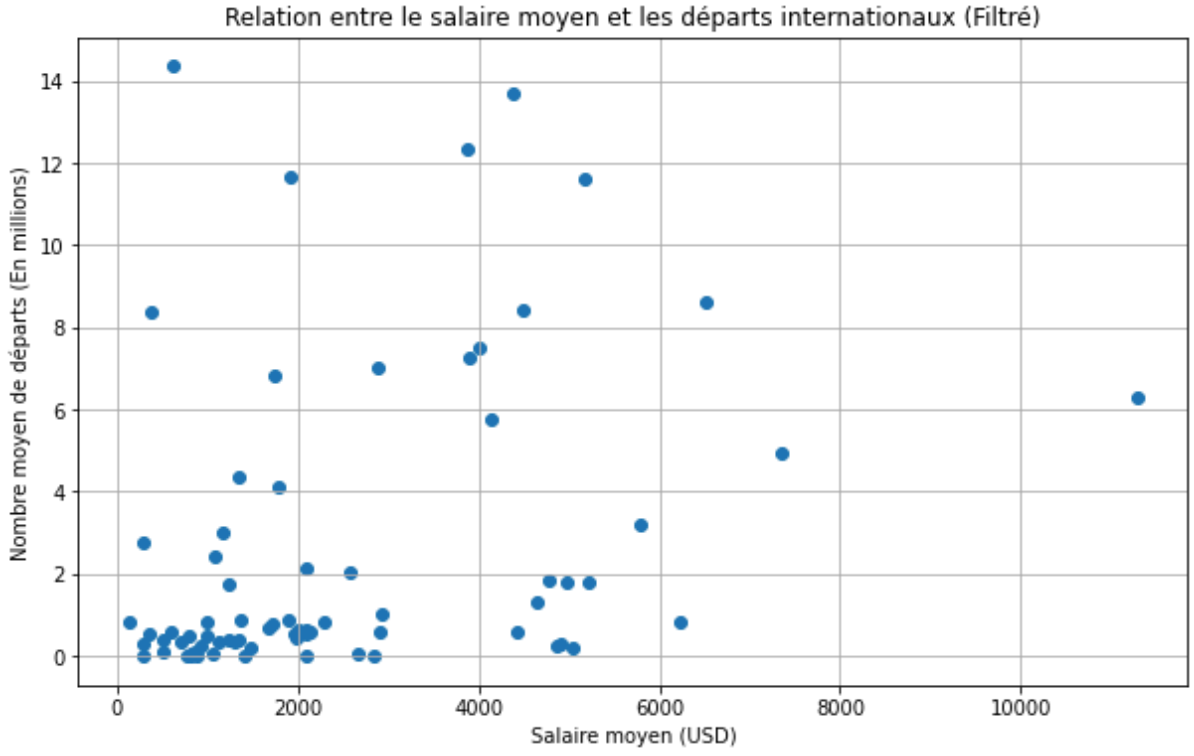
Les données finales sont exportées dans un datalake au format CSV, rendant les résultats accessibles pour une analyse complémentaire ou un partage.

```
# Création du répertoire "exports" dans le datalake (DBFS)
dbutils.fs.mkdirs("dbfs:/mnt/datalake/exports")

# Export des résultats dans un répertoire temporaire
fact_table_path = "/tmp/fact_table_analysis.csv"
df_fact.toPandas().to_csv(fact_table_path, index=False)

# Copie du fichier exporté depuis le répertoire temporaire vers le répertoire cible dans le datalake
dbutils.fs.cp(f"file:{fact_table_path}", "dbfs:/mnt/datalake/exports/fact_table_analysis.csv")

# Vérification que les fichiers ont bien été copiés dans le répertoire cible
display(dbutils.fs.ls("dbfs:/mnt/datalake/exports"))

print("Analyse terminée et résultats exportés dans le datalake.")
```

Conclusion

1. Salaire moyen et départs internationaux

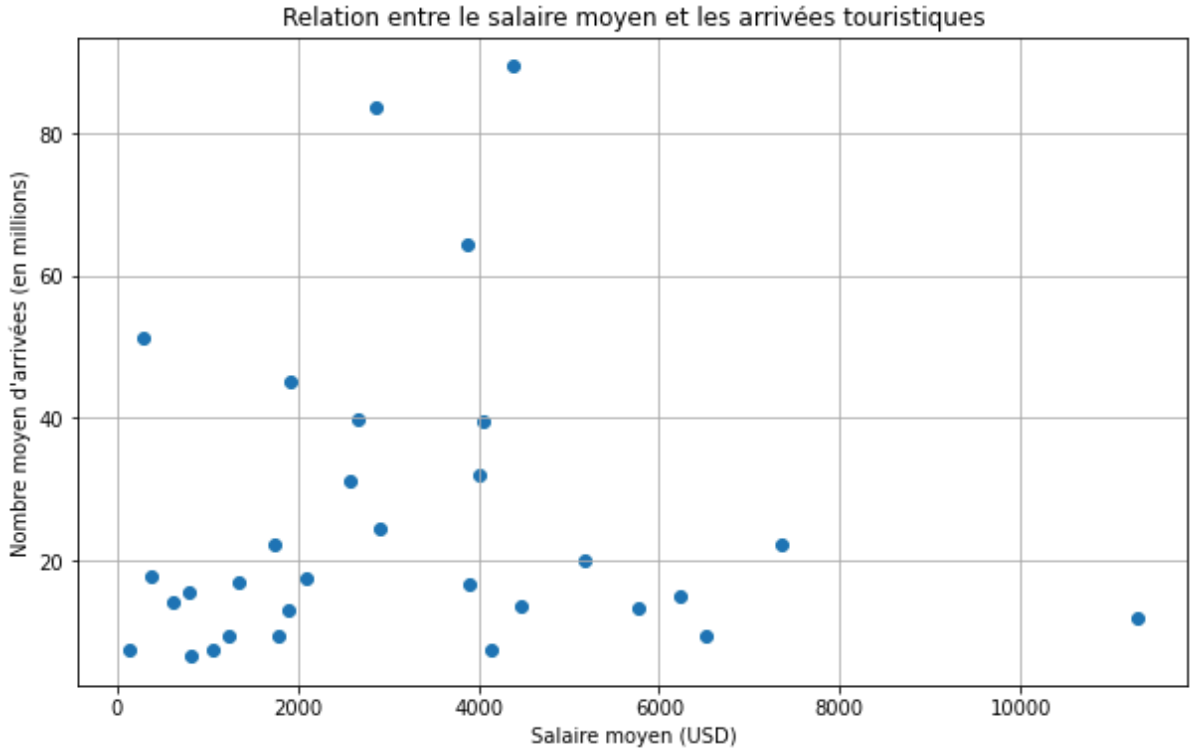
Objectif : Vérifier si un salaire moyen élevé permet à plus de citoyens de voyager.



Analyse : Le graphique montre une corrélation modérée entre un salaire moyen élevé et le nombre de départs internationaux.
En effet, la majorité des pays ayant un salaire moyen inférieur à 4000 USD ont un faible nombre de citoyens partant en voyage à l'étranger (pour la plupart, moins de 2 millions par an), tandis que la moitié des pays avec plus de 4000 USD de salaire moyen ont au moins 4.5 millions de citoyens partant en voyage par an.

2. Salaire moyen et arrivées touristiques

Objectif : Évaluer si les pays avec un haut niveau de vie attirent plus de touristes.



Analyse : Le graphique ne semble pas prouver un quelconque lien entre le niveau d'un pays et son attractivité touristique.

3. Pour aller plus loin

Pour aller plus loin dans l'analyse, il aurait été intéressant d'ajouter une nouvelle dimension à notre jeu de données : le nombre total d'habitants dans chaque pays. Avec cela, nous aurions pu comparer le pourcentage de citoyens de chaque pays partant en voyage et avoir une analyse plus pertinente.