TP2 BigQuery Projet analyse du mouvement public pendant le Covid

Trainer	Salsabil ZAGHDOUDI

Contexte :

Le but de ce projet est d'analyser les données de mobilité publique pour comprendre comment les déplacements ont changé avant et après le début de la pandémie de COVID-19. Pour cela, nous allons utiliser un dataset public fourni par Google, intitulé "**Google COVID-19 Community Mobility Reports**", disponible sur BigQuery.

Modalités pédagogiques :

Durée : 3 jours

Travail en groupe

Let's practice :

1. **Accéder au Dataset :**
   - Connectez-vous à Google Cloud Platform et ouvrez BigQuery.
   - Assurez-vous que vous avez accès au dataset public `bigquery-public-data.covid19_google_mobility`.
2. **Lister les Colonnes de la Table :**
   Pour comprendre les données disponibles, vous devez d'abord lister les colonnes de la table `mobility_report`.

3. **Explorer les Données :**

- Sélectionnez quelques lignes de la table pour voir à quoi ressemblent les données.

```
SELECT
  *
FROM
  `bigquery-public-data.covid19_google_mobility.mobility_report`
LIMIT 10;
```

4. **Analyser les Tendances de Mobilité :**

- Écrivez une requête pour extraire les tendances de mobilité pour différents types de lieux pour un pays spécifique (la France et l'Allemagne) avant et après mars 2020. Comparez les moyennes de changement de mobilité avant (janvier-février 2020) et après (mars-décembre 2020) le début de la pandémie.

Indications pour la requête :

1. **Sélection des Données** : Sélectionner les données pertinentes (types de lieux) pour un pays spécifique.
2. **Définir les Périodes Avant et Après Mars 2020** : Créer des filtres pour séparer les données avant et après mars 2020.
3. **Calculer les Moyennes des Changements de Mobilité** : Calculer les moyennes des pourcentages de changement pour chaque type de lieu avant et après mars 2020.
4. **Comparer les Résultats** : Comparer les moyennes pour analyser l'impact de la pandémie.

Focus sur France-Allemagne :

Analyser comment les différentes phases de confinement liées à la pandémie de COVID-19 ont affecté les déplacements vers différents types de lieux. En d'autres termes, il faudrait voir si les gens ont moins visité certains endroits (comme les magasins, parcs, lieux de travail, etc.) pendant les périodes de confinement, et identifier les périodes spécifiques où ces changements de mobilité ont été les plus marqués :

Étapes de l'Analyse

1. **Identifier les Phases de Confinement :**
   - Pour la France, les principales phases de confinement étaient :
     - Premier confinement : du 17 mars 2020 au 11 mai 2020.
     - Deuxième confinement : du 30 octobre 2020 au 15 décembre 2020.
     - Troisième confinement (partiel) : du 3 avril 2021 au 3 mai 2021

Hint :

Données pour le 1er confinement :

```
SELECT
  date,
  retail_and_recreation_percent_change_from_baseline AS retail_recreation,
  grocery_and_pharmacy_percent_change_from_baseline AS grocery_pharmacy,
  parks_percent_change_from_baseline AS parks,
  transit_stations_percent_change_from_baseline AS transit_stations,
  workplaces_percent_change_from_baseline AS workplaces,
  residential_percent_change_from_baseline AS residential
FROM
  `bigquery-public-data.covid19_google_mobility.mobility_report`
WHERE
  country_region = "France"
  AND date BETWEEN '2020-03-17' AND '2020-05-11'
ORDER BY
  date;
```

- Faites de même pour la 2ème période de confinement et la 3ème.
- Comparez les résultats obtenus avec les périodes de confinement en Allemagne.

- Calculer les moyennes des changements de mobilité pour chaque type de lieu pendant les périodes de confinement et non-confinement pour la France et analysez les résultats obtenus.

Indications pour la requête :

1. Définit les périodes de confinement dans un Common Table Expression (CTE) (`confinement_periods`).
2. Filtre les données de mobilité pour la France.
3. Attribue chaque date à une période de confinement ou de non-confinement.
4. Calcule la moyenne des changements de mobilité pour chaque type de lieu, en groupant les résultats par période.

2. **Analyse saisonnière :**

- Y a-t-il des variations saisonnières dans les tendances de mobilité ? Examinez les données de mobilité par mois sur plusieurs années. Cela nous permettra de voir si des tendances récurrentes se manifestent à des moments spécifiques de l'année.

Indications pour la requête :

- Extraire les données de mobilité par mois et par année.
- Calculer les moyennes mensuelles des changements de mobilité pour chaque type de lieu.
- Visualiser les données pour identifier les tendances saisonnières **:**
  - Une fois les données extraites, vous pouvez les visualiser pour identifier les tendances saisonnières. Par exemple, en utilisant Python et des bibliothèques telles que Pandas et Matplotlib, nous pouvons créer des graphiques pour chaque type de lieu :

```
import pandas as pd
import matplotlib.pyplot as plt

# Supposons que vous avez exporté les résultats de BigQuery vers un DataFrame
# mobility_data = pd.read_csv('path_to_your_exported_data.csv')

# Exemple de structure de DataFrame attendu
data = {
    'year': [2020, 2020, 2020, 2020, 2021, 2021, 2021, 2021],
    'month': [1, 2, 3, 4, 1, 2, 3, 4],
    'avg_retail_and_recreation': [-10, -20, -30, -40, -15, -25, -35, -45],
    'avg_grocery_and_pharmacy': [5, 10, 15, 20, 6, 12, 18, 24],
    'avg_parks': [0, -5, -10, -15, -2, -7, -12, -17],
    'avg_transit_stations': [-5, -10, -15, -20, -6, -12, -18, -24],
    'avg_workplaces': [-10, -15, -20, -25, -12, -17, -22, -27],
    'avg_residential': [5, 10, 15, 20, 6, 11, 16, 21]
}
mobility_data = pd.DataFrame(data)

# Convertir les colonnes year et month en une colonne de date
mobility_data['date'] = pd.to_datetime(mobility_data[['year', 'month']].assig
```

```
# Définir les types de lieux pour l'analyse
places = ['avg_retail_and_recreation', 'avg_grocery_and_pharmacy', 'avg_parks

# Créer des graphiques pour chaque type de lieu
for place in places:
    plt.figure(figsize=(10, 5))
    plt.plot(mobility_data['date'], mobility_data[place], marker='o', linesty
    plt.title(f'Seasonal Trends in {place.replace("_", " ").title()}')
    plt.xlabel('Date')
    plt.ylabel('Percentage Change from Baseline')
    plt.grid(True)
    plt.show()
```

Remarques : Ces graphiques permettront de visualiser les tendances saisonnières des changements de mobilité pour chaque type de lieu, ce qui pourrait révéler des variations récurrentes à certaines périodes de l'année.

- Comment les déplacements dans les parcs ont-ils varié entre l'été et l'hiver ?

Indications pour la requête :

- Calculer les moyennes des changements de mobilité dans les parcs pour les périodes d'été et d'hiver.
- Générer un graphique pour visualiser ces variations avec Python.

3. **Impact sur le télétravail :**

- Comment la mobilité vers les lieux de travail a-t-elle changé avec l'augmentation du télétravail ? faites une analyse sur les 3 périodes de confinement en France et en Allemagne.
- Visualiser les résultats, nous pouvons utiliser un graphique. Voici un exemple de code Python pour créer un graphique à barres :

```
import matplotlib.pyplot as plt
import pandas as pd

# Exemple de données obtenues de la requête BigQuery
data = {
    'country': ['France', 'France', 'France', 'Germany', 'Germany', 'Germany'
    'period': ['confinement_1', 'confinement_2', 'confinement_3', 'confinemen
    'avg_workplaces_change': [-50, -40, -30, -55, -45, -35]  # Remplacez par
}

df = pd.DataFrame(data)

# Création du graphique
plt.figure(figsize=(12, 6))
```

```
for country in df['country'].unique():
    subset = df[df['country'] == country]
    plt.bar(subset['period'], subset['avg_workplaces_change'], label=country)

plt.xlabel('Confinement Period')
plt.ylabel('Average Change in Workplaces Mobility (%)')
plt.title('Average Change in Workplaces Mobility During Confinement Periods i
plt.legend(title='Country')
plt.show()
```

4. **Analyse des jours de la semaine :**

- Y a-t-il des différences dans la mobilité entre les jours de la semaine et les week-ends avant et après la pandémie ?
  - Avant la pandémie : jusqu'au 15 mars 2020.
  - Après le début de la pandémie : à partir du 16 mars 2020.

  Indications pour la requête :

  1. **Définition des périodes** : Classer les dates en deux périodes, avant et après le début de la pandémie.
  2. **Extraction des jours de la semaine** : Utiliser la fonction `EXTRACT(DAYOFWEEK FROM date)` pour obtenir le jour de la semaine (1 = Dimanche, 2 = Lundi, ..., 7 = Samedi).
  3. **Calcul des moyennes** : Calculer les moyennes des changements de mobilité vers les lieux de travail et les lieux de loisirs pour chaque jour de la semaine, avant et après le début de la pandémie.

- Comment ces différences ont-elles évolué avant et après le début de la pandémie ? (Visualiser les résultats avec Python)

```
import matplotlib.pyplot as plt
import pandas as pd

# Exemple de données obtenues de la requête BigQuery
data = {
    'country_region': ['France', 'France', 'France', 'France', 'France', 'Fra
                       'Germany', 'Germany', 'Germany', 'Germany', 'Germany',
    'day_of_week': [1, 2, 3, 4, 5, 6, 7, 1, 2, 3, 4, 5, 6, 7] * 2,
    'period': ['before_pandemic'] * 7 + ['after_pandemic'] * 7 + ['before_pan
    'avg_workplaces_change': [-10, -5, -5, -5, -5, -5, -10, -50, -40, -40, -4
    'avg_retail_recreation_change': [5, 10, 10, 10, 10, 10, 5, -30, -20, -20,
}

df = pd.DataFrame(data)
```

```
# Séparer les données pour les lieux de travail et les loisirs
workplaces_df = df[['country_region', 'day_of_week', 'period', 'avg_workplace
retail_recreation_df = df[['country_region', 'day_of_week', 'period', 'avg_re

# Création des graphiques
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(15, 10), sharey=True)

for idx, country in enumerate(['France', 'Germany']):
    subset_workplaces = workplaces_df[workplaces_df['country_region'] == coun
    subset_retail = retail_recreation_df[retail_recreation_df['country_region

    axes[0, idx].bar(subset_workplaces[subset_workplaces['period'] == 'before
                     subset_workplaces[subset_workplaces['period'] == 'before
    axes[0, idx].bar(subset_workplaces[subset_workplaces['period'] == 'after_
                     subset_workplaces[subset_workplaces['period'] == 'after_
    axes[0, idx].set_title(f'Workplaces Mobility in {country}')
    axes[0, idx].set_xlabel('Day of the Week')
    axes[0, idx].set_ylabel('Avg Change (%)')
    axes[0, idx].legend()

    axes[1, idx].bar(subset_retail[subset_retail['period'] == 'before_pandemi
                     subset_retail[subset_retail['period'] == 'before_pandemi
    axes[1, idx].bar(subset_retail[subset_retail['period'] == 'after_pandemic
                     subset_retail[subset_retail['period'] == 'after_pandemic
    axes[1, idx].set_title(f'Retail & Recreation Mobility in {country}')
    axes[1, idx].set_xlabel('Day of the Week')
    axes[1, idx].set_ylabel('Avg Change (%)')
    axes[1, idx].legend()

plt.tight_layout()
plt.show()
```

- Interprétez les graphiques obtenus.

5. **Analyse saisonnière :**

- Y a-t-il des variations saisonnières dans les tendances de mobilité ?

Indications pour la requête :

1. **Définir les saisons** :
    - Hiver : décembre, janvier, février
    - Printemps : mars, avril, mai
    - Été : juin, juillet, août
    - Automne : septembre, octobre, novembre
2. **Extraire les données de mobilité pour chaque saison**.
3. **Calculer les moyennes des changements de mobilité pour chaque saison**.

- Visualisez les résultats avec Python :

```
import matplotlib.pyplot as plt
import pandas as pd

# Exemple de données obtenues de la requête BigQuery
data = {
    'country_region': ['France', 'France', 'France', 'France', 'Germany', 'Ge
    'season': ['Winter', 'Spring', 'Summer', 'Autumn', 'Winter', 'Spring', 'S
    'avg_retail_and_recreation_change': [-30, -20, -10, -20, -35, -25, -15, -
    'avg_grocery_and_pharmacy_change': [-5, 0, 5, 0, -10, -5, 0, -5],
    'avg_parks_change': [-15, 10, 30, 10, -20, 5, 25, 5],
    'avg_transit_stations_change': [-40, -30, -20, -30, -45, -35, -25, -35],
    'avg_workplaces_change': [-25, -15, -5, -15, -30, -20, -10, -20],
    'avg_residential_change': [10, 5, 0, 5, 15, 10, 5, 10]
}

df = pd.DataFrame(data)

# Création des graphiques
fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(15, 15), sharey=True)
fig.suptitle('Seasonal Mobility Changes in France and Germany')

metrics = ['avg_retail_and_recreation_change', 'avg_grocery_and_pharmacy_chan
           'avg_transit_stations_change', 'avg_workplaces_change', 'avg_resid

titles = ['Retail & Recreation', 'Grocery & Pharmacy', 'Parks', 'Transit Stat

for i, ax in enumerate(axes.flat):
    for country in df['country_region'].unique():
        subset = df[df['country_region'] == country]
        ax.plot(subset['season'], subset[metrics[i]], label=country)
    ax.set_title(titles[i])
    ax.set_xlabel('Season')
    ax.set_ylabel('Avg Change (%)')
    ax.legend()

plt.tight_layout(rect=[0, 0.03, 1, 0.95])
plt.show()
```

- Interprétez les graphiques obtenus.

Analyse approfondie :

1. **Analyse de la Mobilité par Région :**

- Trouvez comment les tendances de mobilité varient-elles entre les régions d'un même pays en suivant ces étapes :
  - **Sélection des données** : sélectionner les colonnes pertinentes du dataset, y compris la date, la région, et les différents types de changements de mobilité.
  - **Filtrage par pays et région** : filtrer les données pour inclure uniquement la France et l'Allemagne, et s'assurer que les sous-régions sont bien spécifiées.
  - **Calcul des moyennes** : Pour chaque région, calculer les moyennes des changements de mobilité pour les différents types de lieux.
  - **Comparaison** : comparer les moyennes des changements de mobilité entre les régions de chaque pays.

  Ces étapes devraient comparer les changements de mobilité pour différentes régions de la France et de l'Allemagne.

2. **Impact des Jours Fériés :**
   - Les jours fériés ont-ils eu un impact sur la mobilité avant et après la pandémie ? Identifiez les principaux jours fériés en France et en Allemagne et analysez les changements de mobilité autour de ces dates.

   Étapes à suivre pour l'analyse :

   1. Identifier les principaux jours fériés en France et en Allemagne.
   2. Extraire et comparer les changements de mobilité pour ces jours fériés avant et après la pandémie.
   3. Analyser les résultats pour déterminer l'impact des jours fériés sur la mobilité.

3. **Impact des Mesures de Santé Publique :**

   Testez cette requête et expliquez les résultats obtenus

```
WITH public_health_measures AS (
  SELECT 'France' AS country, DATE('2020-03-17') AS start_date, DATE('2020
  SELECT 'France', DATE('2020-10-30'), DATE('2020-12-15'), 'Second Lockdow
  SELECT 'France', DATE('2020-12-15'), DATE('2021-05-03'), 'Curfew' UNION
  SELECT 'Germany', DATE('2020-03-22'), DATE('2020-05-04'), 'First Lockdow
  SELECT 'Germany', DATE('2020-11-02'), DATE('2020-12-20'), 'Partial Lockd
  SELECT 'Germany', DATE('2020-12-16'), DATE('2021-03-28'), 'Strict Lockdo
),
mobility_data AS (
  SELECT
    country_region,
    date,
    retail_and_recreation_percent_change_from_baseline AS retail_recreatio
    grocery_and_pharmacy_percent_change_from_baseline AS grocery_pharmacy,
    parks_percent_change_from_baseline AS parks,
    transit_stations_percent_change_from_baseline AS transit_stations,
```

```
    workplaces_percent_change_from_baseline AS workplaces,
    residential_percent_change_from_baseline AS residential
  FROM
    `bigquery-public-data.covid19_google_mobility.mobility_report`
  WHERE
    country_region IN ('France', 'Germany')
    AND date BETWEEN '2020-01-01' AND '2021-12-31'
),
measure_impact AS (
  SELECT
    m.country,
    m.measure,
    md.date,
    md.retail_recreation,
    md.grocery_pharmacy,
    md.parks,
    md.transit_stations,
    md.workplaces,
    md.residential
  FROM
    public_health_measures AS m
  JOIN
    mobility_data AS md
  ON
    md.country_region = m.country
    AND md.date BETWEEN m.start_date AND m.end_date
)
SELECT
  country,
  measure,
  AVG(retail_recreation) AS avg_retail_recreation,
  AVG(grocery_pharmacy) AS avg_grocery_pharmacy,
  AVG(parks) AS avg_parks,
  AVG(transit_stations) AS avg_transit_stations,
  AVG(workplaces) AS avg_workplaces,
  AVG(residential) AS avg_residential
FROM
  measure_impact
GROUP BY
  country,
  measure
ORDER BY
  country,
  measure;
```

4. **Impact des Variantes du Virus :**

- Comment les apparitions de nouvelles variantes du virus (comme la variante Delta ou Omicron) ont-elles affecté la mobilité ? Analysez les données de mobilité autour des dates clés d'apparition de ces variantes avec ce code :

```
WITH variant_impact_periods AS (
  SELECT 'France' AS country, DATE('2020-12-01') AS start_date, DATE('2021
  SELECT 'France', DATE('2021-12-01'), DATE('2022-01-31'), 'Omicron Varian
  SELECT 'Germany', DATE('2020-12-01'), DATE('2021-01-31'), 'Delta Variant
  SELECT 'Germany', DATE('2021-12-01'), DATE('2022-01-31'), 'Omicron Varia
),
mobility_data AS (
  SELECT
    country_region,
    date,
    retail_and_recreation_percent_change_from_baseline AS retail_recreatio
    grocery_and_pharmacy_percent_change_from_baseline AS grocery_pharmacy,
    parks_percent_change_from_baseline AS parks,
    transit_stations_percent_change_from_baseline AS transit_stations,
    workplaces_percent_change_from_baseline AS workplaces,
    residential_percent_change_from_baseline AS residential
  FROM
    `bigquery-public-data.covid19_google_mobility.mobility_report`
  WHERE
    country_region IN ('France', 'Germany')
    AND date BETWEEN '2020-01-01' AND '2022-01-31'
),
variant_impact AS (
  SELECT
    vip.country,
    vip.variant,
    md.date,
    md.retail_recreation,
    md.grocery_pharmacy,
    md.parks,
    md.transit_stations,
    md.workplaces,
    md.residential
  FROM
    variant_impact_periods AS vip
  JOIN
    mobility_data AS md
  ON
    md.country_region = vip.country
    AND md.date BETWEEN vip.start_date AND vip.end_date
)
SELECT
  country,
  variant,
```

```
  AVG(retail_recreation) AS avg_retail_recreation,
  AVG(grocery_pharmacy) AS avg_grocery_pharmacy,
  AVG(parks) AS avg_parks,
  AVG(transit_stations) AS avg_transit_stations,
  AVG(workplaces) AS avg_workplaces,
  AVG(residential) AS avg_residential
FROM
  variant_impact
GROUP BY
  country,
  variant
ORDER BY
  country,
  variant;
```

5. **Comparaison entre Zones Urbaines et Rurales :**
   - Y a-t-il une différence significative dans les changements de mobilité entre les zones urbaines et rurales ? Utilisez les sous-régions (sub_region_1 et sub_region_2) pour effectuer cette analyse.

   Indications pour la requête :

   - Pour analyser les différences de mobilité entre les zones urbaines et rurales en utilisant les sous-régions (`sub_region_1` et `sub_region_2`), nous devons suivre les étapes suivantes :

     1. **Classification des Zones** : Identifier et classer les sous-régions en zones urbaines et rurales:

     Pour cette analyse, nous devons créer une classification des sous-régions en zones urbaines et rurales. Supposons que nous avons un tableau pré-existant ou que nous pouvons utiliser une logique simple pour classifier ces sous-régions (par exemple, considérer les grandes villes comme urbaines et les autres régions comme rurales).

     Exemple de Classification

     France

     - Zones Urbaines : Île-de-France, Auvergne-Rhône-Alpes
     - Zones Rurales : Bourgogne-Franche-Comté, Centre-Val de Loire

     Allemagne

     - Zones Urbaines : Berlin, Hamburg
     - Zones Rurales : Mecklenburg-Vorpommern, Brandenburg

     1. **Extraction des Données de Mobilité** : Extraire les données de mobilité pour les sous-régions classées.
     2. **Comparaison des Changements de Mobilité** : Comparer les moyennes des changements de mobilité entre les zones urbaines et rurales.

2. Extraction des Données de Mobilité

Testez cette requête et analysez les résultats :

```
WITH region_classification AS (
  SELECT 'France' AS country, 'Île-de-France' AS sub_region_1, 'urban'
  SELECT 'France', 'Auvergne-Rhône-Alpes', 'urban' UNION ALL
  SELECT 'France', 'Bourgogne-Franche-Comté', 'rural' UNION ALL
  SELECT 'France', 'Centre-Val de Loire', 'rural' UNION ALL
  SELECT 'Germany', 'Berlin', 'urban' UNION ALL
  SELECT 'Germany', 'Hamburg', 'urban' UNION ALL
  SELECT 'Germany', 'Mecklenburg-Vorpommern', 'rural' UNION ALL
  SELECT 'Germany', 'Brandenburg', 'rural'
),
mobility_data AS (
  SELECT
    country_region,
    sub_region_1,
    date,
    retail_and_recreation_percent_change_from_baseline AS retail_recrea
    grocery_and_pharmacy_percent_change_from_baseline AS grocery_pharma
    parks_percent_change_from_baseline AS parks,
    transit_stations_percent_change_from_baseline AS transit_stations,
    workplaces_percent_change_from_baseline AS workplaces,
    residential_percent_change_from_baseline AS residential
  FROM
    `bigquery-public-data.covid19_google_mobility.mobility_report`
  WHERE
    country_region IN ('France', 'Germany')
    AND date BETWEEN '2020-01-01' AND '2021-12-31'
),
classified_mobility AS (
  SELECT
    rc.zone,
    md.date,
    md.retail_recreation,
    md.grocery_pharmacy,
    md.parks,
    md.transit_stations,
    md.workplaces,
    md.residential
  FROM
    mobility_data AS md
  JOIN
    region_classification AS rc
  ON
    md.country_region = rc.country
```

```
    AND md.sub_region_1 = rc.sub_region_1
)
SELECT
  zone,
  AVG(retail_recreation) AS avg_retail_recreation,
  AVG(grocery_pharmacy) AS avg_grocery_pharmacy,
  AVG(parks) AS avg_parks,
  AVG(transit_stations) AS avg_transit_stations,
  AVG(workplaces) AS avg_workplaces,
  AVG(residential) AS avg_residential
FROM
  classified_mobility
GROUP BY
  zone
ORDER BY
  zone;
```

Livrables :

1. Requêtes SQL.
2. Graphiques fait sur Python.
3. Analyse des résultats après exécution des requêtes données dans ce brief.