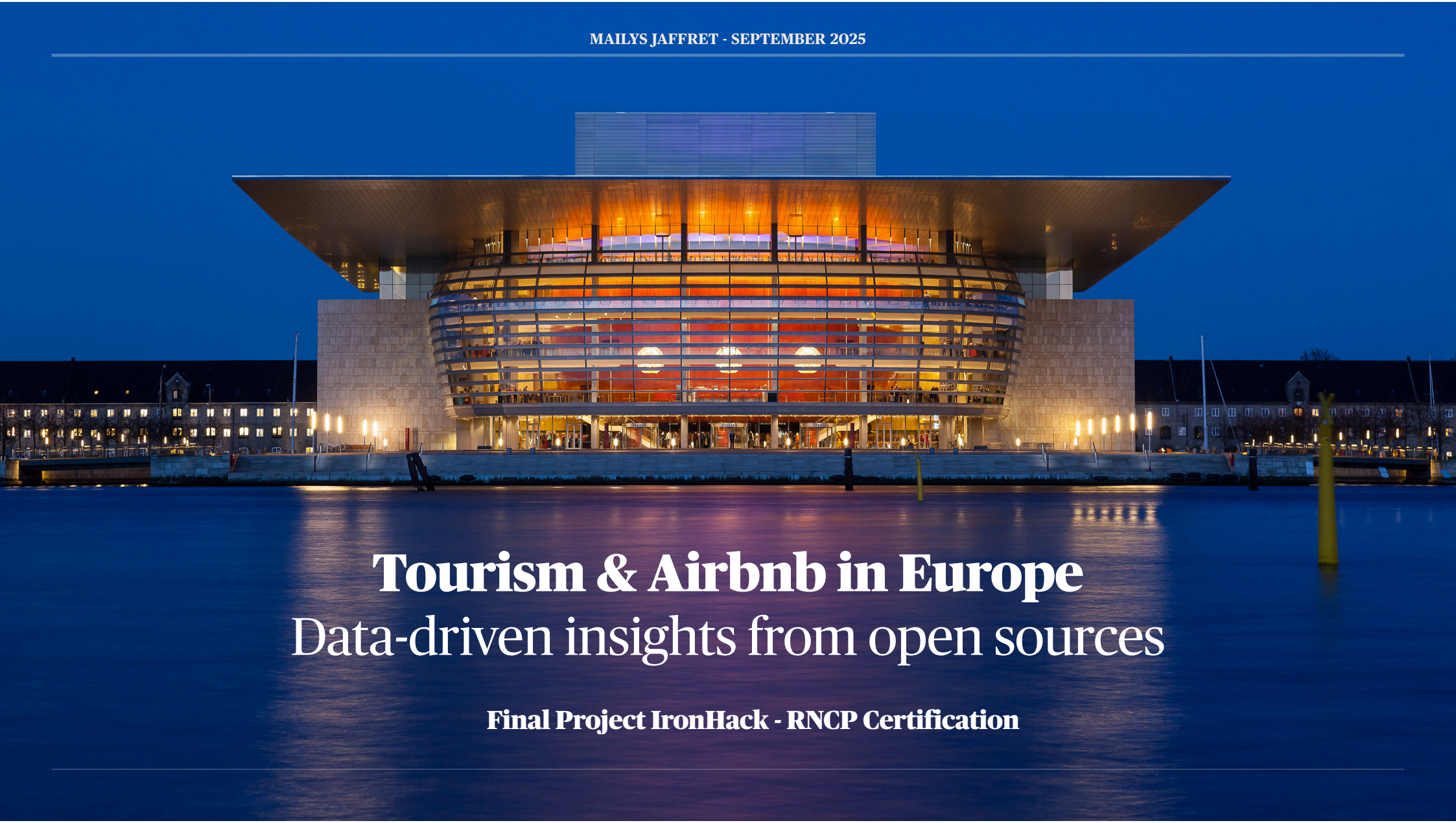
MAILYS JAFFRET - SEPTEMBER 2025



Tourism & Airbnb in Europe

Data-driven insights from open sources

Final Project IronHack - RNCP Certification

Roadmap of the Presentation

- 1- Business Context
- 2- Objectives
- 3- Project Management
- 4- Data sources and data cleaning
- 5- SQL : ERD, Cration of the database, and scripts
- 6- Big Query
- 7- EDA
- 8- Machine Learning
- 9- Demo
- 10- Conclusion

Business Context

1- Tourism is a major sector In Europe

2- Airbnb has changed short-term rentals

3- Challenges: affordability, regulation, competitiveness

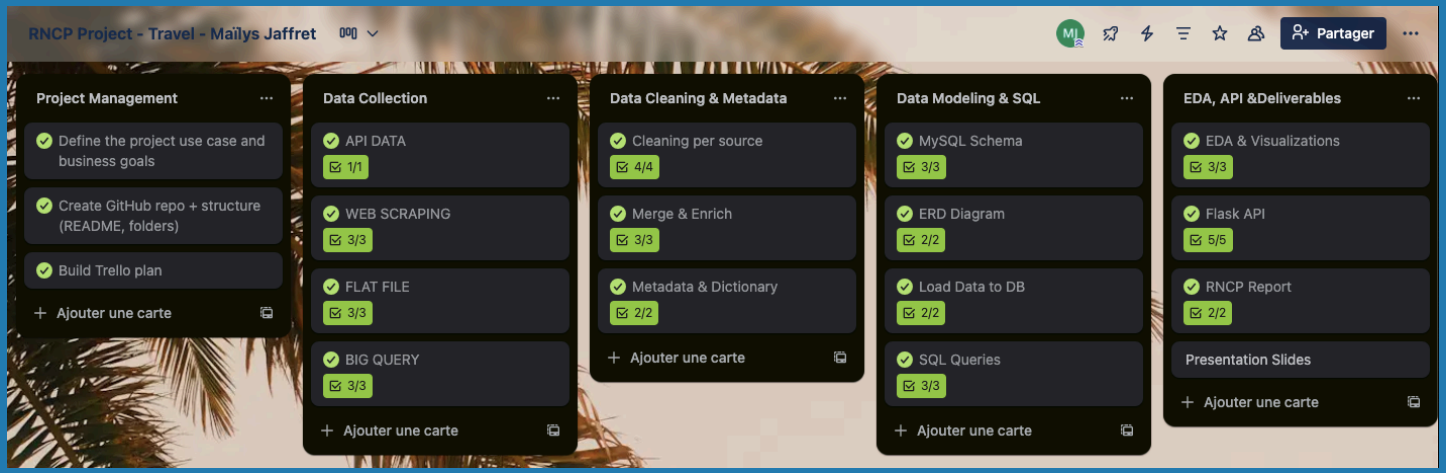
Objectives

1- Identify main factors that influence Airbnb prices

2- Compare tourism behavior across countries

3- Provide insights for: Host & Airbnb, Cities & Policymakers, tourism boards

PROJECT MANAGEMENT



1- Trello Kanban board

2- Tasks tracked with checklists

3- Clear workflow & reproducibility

DATA SOURCES

3 API

from Eurostat:
detailed data on accommodation, transport and destination.

SCOPE :

I identified the top 4 destinations in 2023: France, Germany, Italy, Spain.

Web scraping

from WorldData and Wikipedia:
number of international arrivals and description of each capital city.

Web scraping

from Numbeo:
cost of living data to compare affordability.

4 Flat files

from Airbnb:
detailed data on listings, room types, availability, and prices.

DATA CLEANING

Eurostat

- JSON parsing
- Numeric conversions
- Aggregation by country

WorldData

- Scrap HTML
- Extract relevant tables
- Standardize decimals
- Convert to integers
- Filter las 10 years

Numbeo

- Scrap tables
- Parse numbers with regex
- Normalized labels
- Extract items: meals, cappucino, gasoline, cinema…

Wikipedia

- Scrap tables
- Parse numbers with regex
- Convert columns to numeric

Airbnb

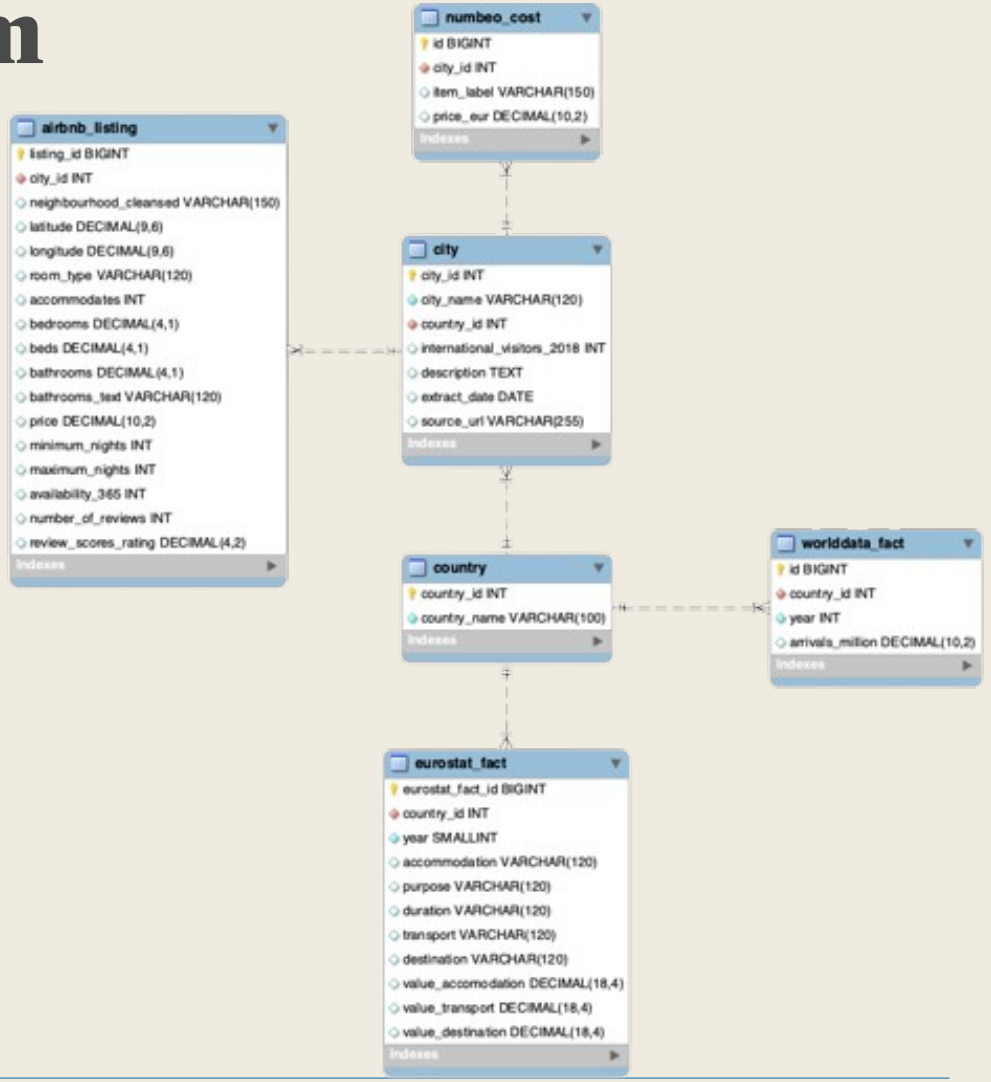
- Merge multiple CSV
- Select useful columns
- Filter invalid rows

General Cleaning Practices

- Missing data —> dropna, conditional filtering
- String manipulation —> .str.strip() / .str.lower() / unicodedata.normalize
- Regex transformations —> re.sub to remove unwanted symbols/units
- Schema harmonization —> renaming columns, reindexing for consistency
- Outlier/aggregate removal —> explicite exclusion lists

Entity Relationship Diagram

- Airbnb listings ➜ linked to cities via **city_id (FK)**
- Cities ➜ linked to countries via **country_id (FK)**
- **City-level data** (Airbnb_listing, Numbeo) joined on **city_id**
- **Country-level data** (Eurostat, WorldData) joined on **country_id**
- PK/FK ensure integrity and consistency



Database Creation - MySQL Workbench

1- Create database tourism_analytics

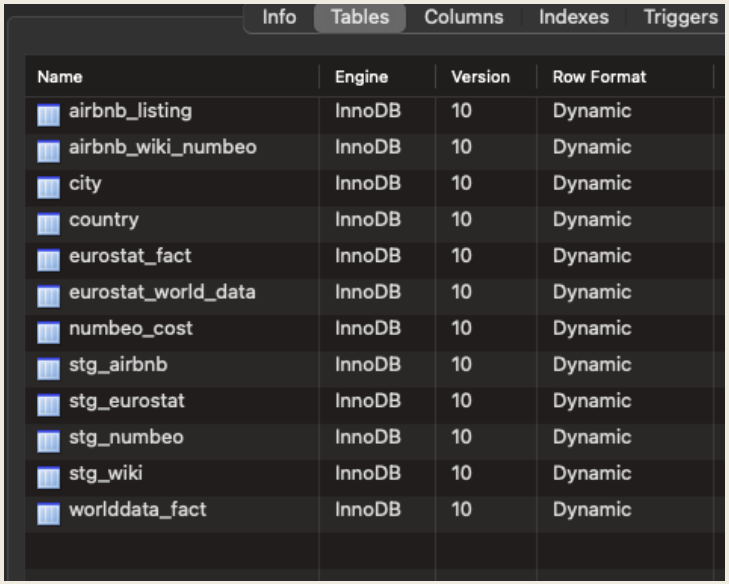
2- Build final tables (Airbnb, Eurostat, WorldData, Numbeo, City, Country)

3- Create staging tables for raw CSVs

4- Load CSVs ➜ staging (LOAD DATA LOCAL INFILE)

5- Insert into final tables with cleaning & casting

6- Run controls to check counts



1- SQL Script: TOP 3 Trip Purposes (2023)

```sql
SELECT t.country_name,
       t.purpose,
       t.destination,
       t.total_trips_million,
       t.rn AS rank_in_country
FROM (
    SELECT c.country_name,
           e.purpose,
           e.destination,
           ROUND(SUM(e.value_destination) / 1000000, 1) AS total_trips_million,
           ROW_NUMBER() OVER (
               PARTITION BY c.country_name
               ORDER BY SUM(e.value_destination) DESC
           ) AS rn
    FROM eurostat_fact e
    JOIN country c ON e.country_id = c.country_id
    WHERE e.purpose IS NOT NULL
      AND e.purpose <> 'Total'
      AND e.destination IS NOT NULL
      AND e.year = 2023
    GROUP BY c.country_name, e.purpose, e.destination
) t
WHERE t.rn <= 3
ORDER BY t.country_name, t.rn;
```

Method: ROW_NUMBER() window function ➜ ranks purposes by country

Results:

🇫🇷 **France ➜** Personal (377M), Family (180M), Business (38M)
➡ Strong dominance of personal trips; family visits also very high

🇩🇪 **Germany ➜** Personal (246M), Family (108M), Business (52M)
➡ More balanced mix; business travel relatively stronger than in South Europe

🇮🇹 **Italy ➜** Personal (65M), Family (16M), Business (6M)
➡ Travel volume much smaller overall; domestic leisure is the main driver

🇪🇸 **Spain ➜** Personal (248M), Family (93M), Business (13M)
➡ High personal travel; business travel is marginal compared to leisure

2- SQL Script: Main transport modes by Country (2023)

```sql
SELECT t.country_name,
       t.transport,
       t.total_trips_million,
       t.rn AS rank_in_country
FROM (
    SELECT c.country_name,
           e.transport,
           ROUND(SUM(e.value_transport) / 1000000, 1) AS total_trips_million,
           ROW_NUMBER() OVER (
               PARTITION BY c.country_name
               ORDER BY SUM(e.value_transport) DESC
           ) AS rn
    FROM eurostat_fact e
    JOIN country c ON e.country_id = c.country_id
    WHERE e.transport IS NOT NULL
      AND e.year = 2023
    GROUP BY c.country_name, e.transport
) t
WHERE t.rn <= 3
ORDER BY t.country_name, t.rn;
```

Method: ROW_NUMBER() window function ➜ ranks purposes by country

Results:

🇫🇷 **France**

1. Car/road vehicles ➜ 164.5M trips
2. Rail ➜ 42.2M trips
3. Air ➜ 24.5M trips

🇩🇪 **Germany**

1. Car/road vehicles ➜ 145.9M trips
2. Rail ➜ 47.8M trips
3. Air ➜ 40.6M trips

🇮🇹 **Italy**

1. Car/road vehicles ➜ 25.8M trips
2. Air ➜ 9.8M trips
3. Rail ➜ 5.3M trips

🇪🇸 **Spain**

1. Car/road vehicles ➜ 108.4M trips
2. Air ➜ 19.6M trips
3. Rail ➜ 9.9M trips

BigQuery for Big Data

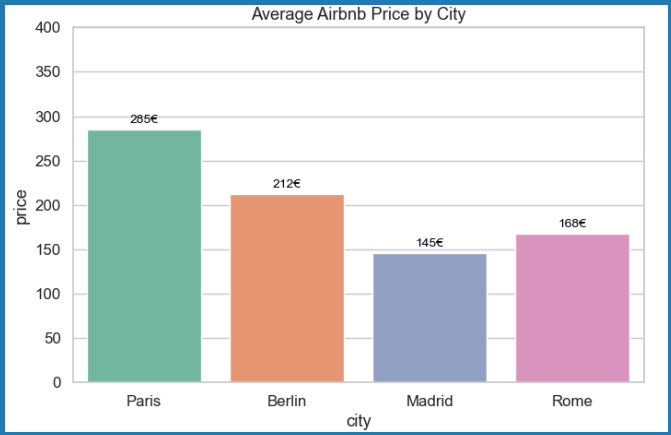
```sql
CREATE OR REPLACE TABLE `tourism-in-europe.tourism_analytics.airbnb_wiki_numbeo_partitioned`
PARTITION BY RANGE_BUCKET(price_int, GENERATE_ARRAY(0, 10000, 50))
CLUSTER BY city_name AS
WITH s AS (
  SELECT
    -- conversion sécurisée du prix en entier
    CAST(ROUND(price) AS INT64) AS price_int,
    CAST(ROUND(price) AS INT64) - MOD(CAST(ROUND(price) AS INT64), 50) AS bucket_low,
    CAST(ROUND(price) AS INT64) - MOD(CAST(ROUND(price) AS INT64), 50) + 49 AS bucket_high, t.*
  FROM `tourism-in-europe.tourism_analytics.airbnb_wiki_numbeo` AS t
  WHERE price IS NOT NULL
)
SELECT
  price_int,
  CASE
    WHEN price_int >= 1000 THEN '1000+'
    ELSE FORMAT ('%d-%d', bucket_low, bucket_high)
  END AS price_range,
  s.* EXCEPT(price_int, bucket_low, bucket_high)
FROM s;
```

- Cloud data warehouse for large-scale analysis
- Airbnb dataset denormalized for cross-city comparison
- Partitioning: price intervals (50 EUR buckets)
- Clustering: by city for faster queries
- Derived field *price_range* (0-49, 50-99, ..., 1000+)
- Python notebook connection ➜ CSV export for Tableau

Exploratory Data Analysis

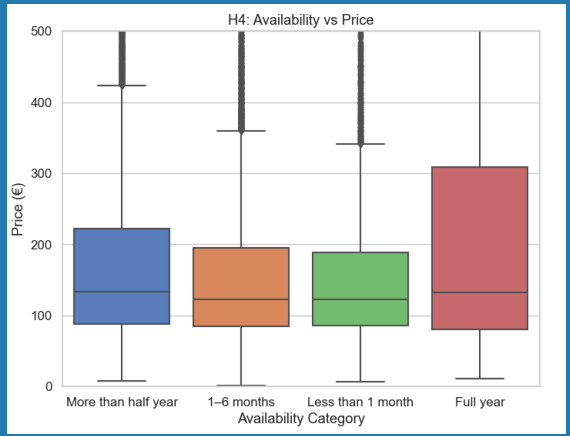
What really drive Airbnb Prices?

Paris is by far the most expensive city



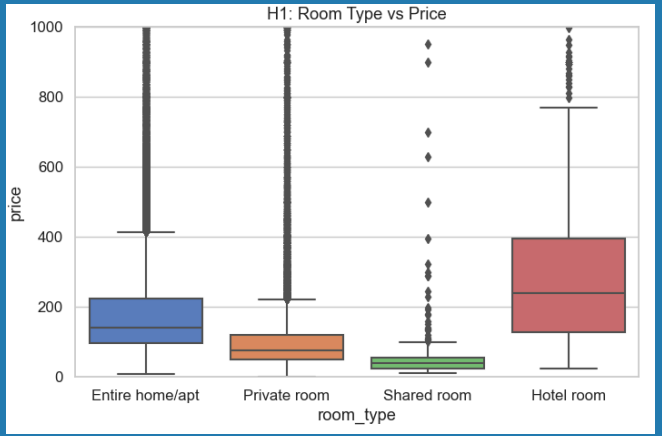

```
ANOVA test (room_type vs price):
F-statistic: 451.1194460271183
p-value: 2.0651529850573897e-291
```

Year-round availability increases price




```
ANOVA test (city vs price):
F-statistic: 267.23132958043436
p-value: 7.5073552375435e-173
```

Entire apartment & hotel rooms drive higher prices



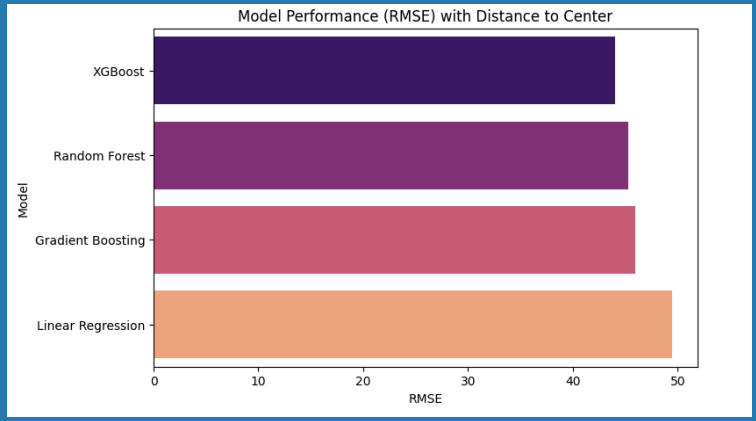

```
ANOVA test (availability vs price):
F-statistic: 146.6613418225656
p-value: 7.346078098034065e-95
```

Machine Learning: Regression results

Predicting Price per Person

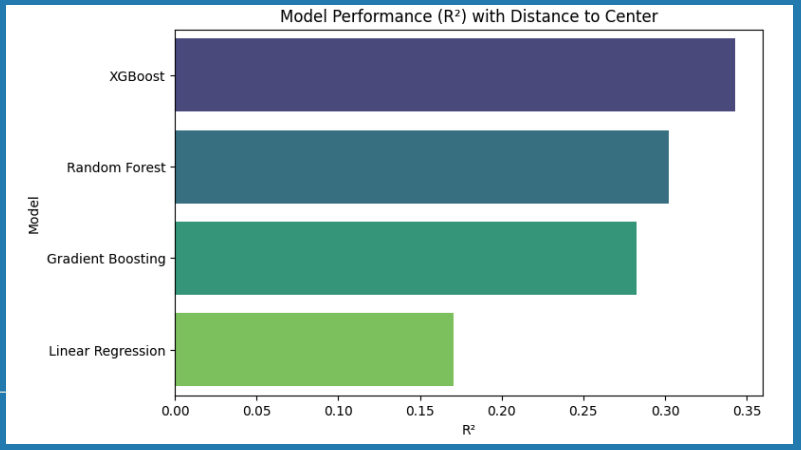
ML Process / General Overview

- **Feature Engineering:** Added price_per_person, encoded categorical vars, created distance_center (geodesic distance to Paris center).
- **Feature Selection:** Numeric (latitude, longitude, accommodates, bedrooms, beds, availability, reviews, distance_center) + Categorical (room_type, neighbourhood).
- **Handling Missing Values:** Median for numerics, most frequent for categoricals.
- **Evaluation Metrics:** RMSE (error), R^2 (explained variance).
- **Models Tested:** Linear Regression, Random Forest, Gradient Boosting, XGBoost.

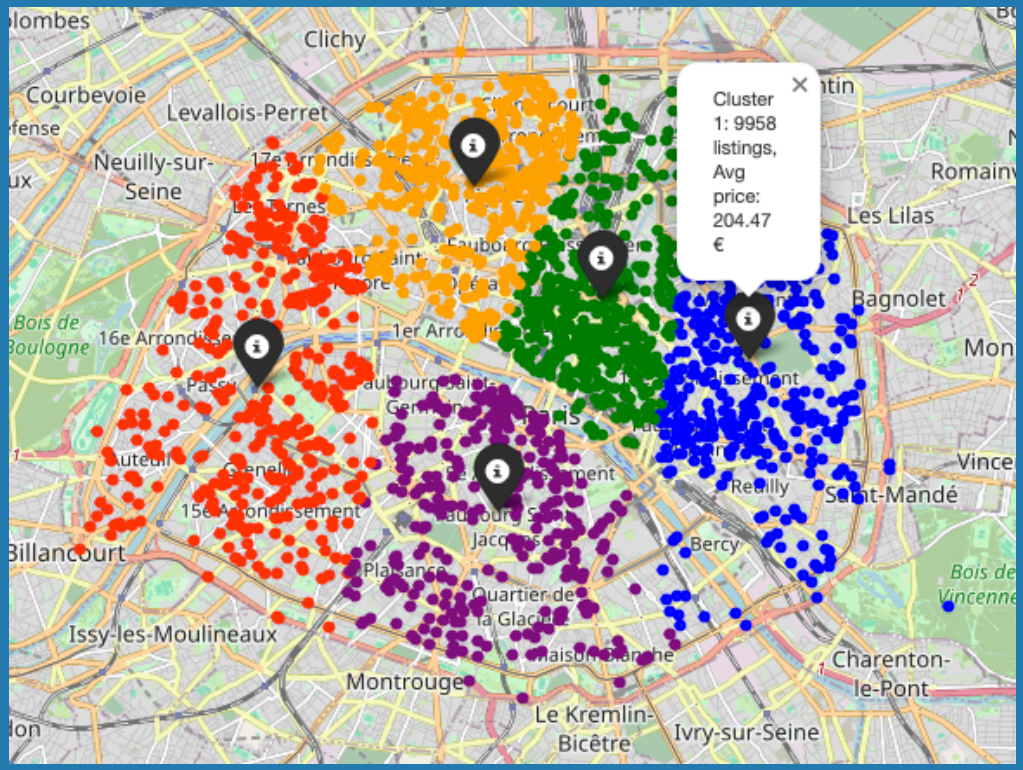


Main Results

- Adding **distance to center** improved performance.
- **Best model: XGBoost (R^2 = 0.34, RMSE ≈ 44) ➜** better at capturing spatial & non-linear effects.
- Random Forest and Gradient Boosting also competitive.
- Linear Regression underperforms (too simple).



Machine Learning: Clustering



ML Process:

- **Feature Engineering:** Used latitude, longitude, price to identify geographical and pricing patterns.
- **Data Cleaning:** Removed extreme prices (>1000€).
- **Hyperparameter Tuning**: tested several cluster numbers (k = 5, 10, 15, 20, 25).
  - Chose k=5 based on silhouette score.
- **Evaluation Metric: Silhouette Score** and visualization on interactive map.

Main Results:

- Best **k = 5 clusters ➜** clear spatial & pricing segmentation.
- Cluster averages (examples):
  - Cluster 0 (red) ➜ High average price: 373€
  - Cluster 1 (blue) ➜ Lower average price: 204€
  - Others around 270-280€.

Conclusion Machine Learning

- **Price Prediction**: location (distance to center) is a strong predictor. Still, prices remain highly variable → other factors (amenities, host reputation) could improve models.
- **Market Segmentation**: clustering shows distinct zones with different price dynamics → useful for hosts (pricing strategy) or platforms (personalized recommendations).
- **Next Steps**:
  - Apply advanced hyperparameter tuning (Grid Search, Bayesian Optimization).
  - Add more features (reviews text, host information, amenities).
  - Combine regression + clustering for deeper market insights.

DASHBOARD

https://public.tableau.com/app/profile/ma.lys.jaffret/viz/tourism-in-europe1/Tableaudebord1

Conclusion

Eurostat → *How and why people travel*

- Transport, destinations, motives, accommodations

Airbnb → *Where and at what price*

- Price drivers
- Price prediction (regression)
- Paris clustering (zones by price & distance to center)

✨ **Combined insight**: Together, they provide a complete picture of tourism, useful for both **public policies** and **tourism businesses**.

Thank you for your attention !

Do you have any questions ?