

WAY TO PARIS!

Eco-Friendly edition 😊

*How to go to Paris with minimal
environmental impact?*

Project by
Gema, Laura, Miqueas

November 2024



Project Goal

Our goal is to know how it is more sustainable to move and discover new places.

We want to compare the carbon footprint of a trip from Madrid to Paris (round trip) by train, plane and in different brands/models of cars.

In addition, we will take into account the hours that we would need to be able to choose the most sustainable and efficient way of doing tourism.

Hypothesis

1. CO₂ emissions: gas car > plane > train > hybrid car > electric car
2. Older cars > newer cars
3. Faster =/ cheaper =/ eco-friendly 😊

Data

We have obtained the data from two APIs:

1. Carbon Interface
2. OpenChargeMap

**Let's go to
Paris!**



APIs | Carbon Interface

Vehicle Estimate Response

```
{
  "data": {
    "id": "6108d711-be04-4dc4-93f9-43d969fd5273",
    "type": "estimate",
    "attributes": {
      "distance_value": 100.0,
      "vehicle_make": "Toyota",
      "vehicle_model": "Corolla",
      "vehicle_year": 1993,
      "vehicle_model_id": "7268a9b7-17e8-4c8d-acca-57059252afe9",
      "distance_unit": "mi",
      "estimated_at": "2021-01-10T15:24:32.568Z",
      "carbon_g": 37029,
      "carbon_lb": 81.64,
      "carbon_kg": 37.03,
      "carbon_mt": 0.04
    }
  }
}
```

Train Estimate Response

```
• v def calculate_train_emissions(distance_km, train_type='electric', passengers=1):
    emission_factors = {'electric': 0.02, 'diesel': 0.14} # in kg CO2 per km per passenger
    factor = emission_factors.get(train_type.lower(), 0.02)
    return distance_km * factor * passengers

# Example usage for a 3,600 km train trip (round trip)
train_distance_km = 3600 # Distance in kilometers
train_emissions = calculate_train_emissions(train_distance_km, train_type='electric', passengers=3)
print(f"Train emissions for {train_distance_km} km: {train_emissions:.2f} kg CO2")
```

Flight Estimate Response

```
{
  "data": {
    "id": "d60edacc-cf6c-4da7-b5de-c538de4ce5ee",
    "type": "estimate",
    "attributes": {
      "passengers": 2,
      "legs": [
        {
          "departure_airport": "SFO",
          "destination_airport": "YYZ"
        },
        {
          "departure_airport": "YYZ",
          "destination_airport": "SFO"
        }
      ],
      "estimated_at": "2020-07-24T02:25:50.837Z",
      "carbon_g": 1077098,
      "carbon_lb": 2374,
      "carbon_kg": 1077,
      "carbon_mt": 1,
      "distance_unit": "km",
      "distance_value": 7454.15
    }
  }
}
```

APIs | Cleaning Data

Carbon Interface

1. From the "Carbon Interface" API, we did the clean-up from the API directly.
2. We only had to indicate the number of passengers, the kilometres to travel and the means of transportation we wanted.
3. We save the information in the data frame (.csv) that we downloaded already had the data and columns that we wanted to work with.



Return trip

By plane



Distance

2600 kilometres



Time

4 hours of travel



CO₂

465.65 kg for 3 people

By car



Distance

2800 kilometres



Time

28 hours of travel



CO₂

From 0 kg (electric)
to 1535.8 kg (gas - supercar)

By train



Distance

3600 kilometres



Time

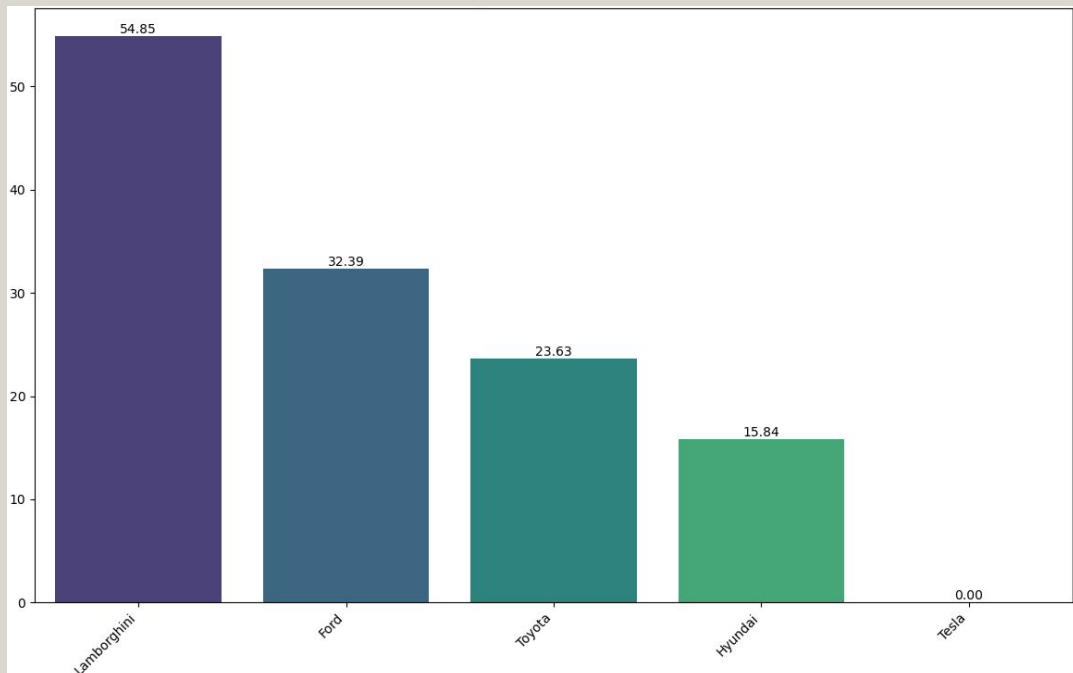
20 hours of travel



CO₂

261 kg for 3 people

Way to Paris by car



We have 5 car brands options to rent, to go to Paris:

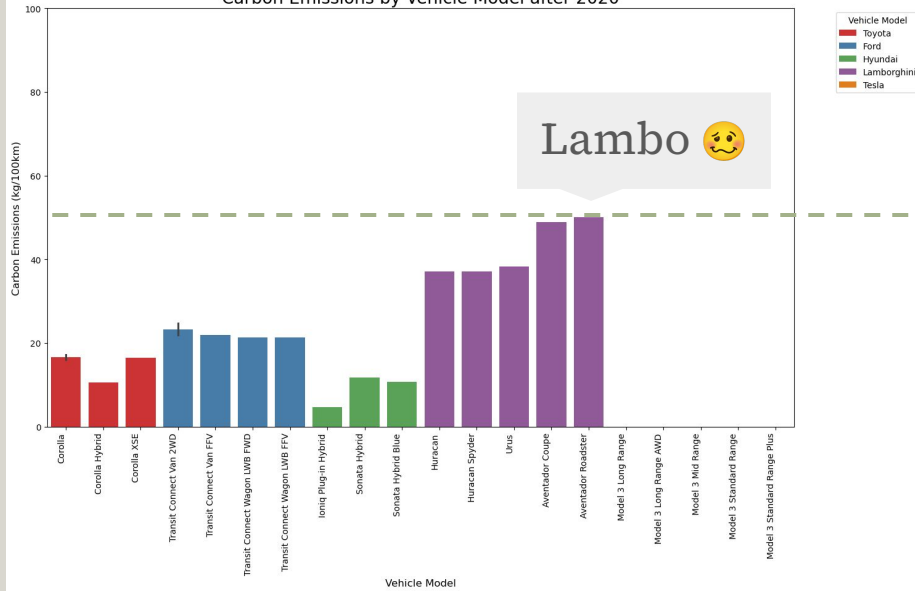
1. Lamborghini (supercar)
2. Ford (regular)
3. Toyota (regular)
4. Hyundai (hybrid)
5. Tesla (electric)

Way to Paris by car

Within the 5 selected brands, we chose cars registered in 2020 (new) or later and before 1990 (old).

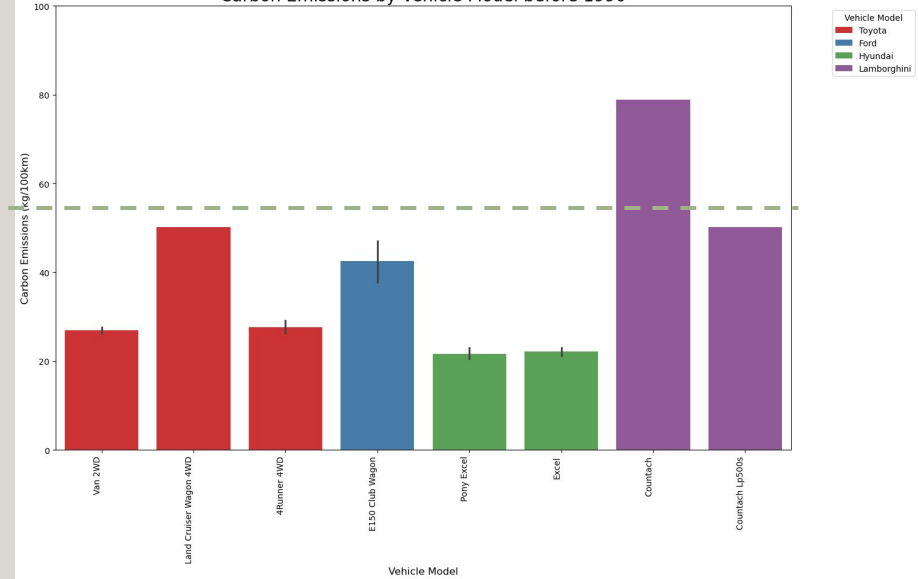
New cars (2020 & after)

Carbon Emissions by Vehicle Model after 2020



Old cars (Before 1990)

Carbon Emissions by Vehicle Model before 1990



APIs | Open Charge Map

This API allows you to see the charging points in different countries, but you cannot generate a route between two points in this beta.

You must clean and classify the data, taking into account the cities you are going to pass through.

```
url = "https://api.openchargemap.io/v3/poi/"

params_base = {
    "output": "json",
    "maxresults": 10000,
    "compact": "true",
    "verbose": "false",
    "key": "d1138da0-c7d4-4f81-a2c6-44de7bbd80a0"
}

country_codes = ["ES", "FR"]

data_country = []

for country in country_codes:
    params = params_base.copy()
    params["countrycode"] = country

    response = requests.get(url, params=params)

    if response.status_code == 200:
        data = response.json()
        data_country.extend(data)
    else:
        print(f"Error en la solicitud para {country}: {response.status_code}")

data_chargesmap = pd.DataFrame(data_country)
```


APIs | Cleaning Data

Open Charge Map

1. From the API "OpenChargeMap", we did the clean-up after downloading the data from the API and creating the data frame (.csv).
2. The raw data set contained many unnecessary columns, including: address info, UU ID, operator ID, status type ID, general comments...
3. To obtain the cities we needed for our route, we had to separate the information I saw in the same column, into six



Way to Paris by Tesla

Tesla is the car that emits the least CO₂ emissions during our trip. However, when travelling in an electric car, we have to take into account:

1. We have to stop to charge the car approximately 7 times.
2. Our trip will last 3 hours more.



Conclusio

By plane



Distance

2600 kilometres



Time

4 hours of travel



CO₂

465.65 kg for 3 people

n

By Tesla



Distance

2800 kilometres



Time

34 hours of travel



CO₂

0 kg

By train



Distance

3600 kilometres



Time

20 hours of travel



CO₂

261 kg for 3 people

Teamwork

Firstly, we spent one day brainstorming to see what data we could focus our project on. After deciding together that we were going to investigate the environment, we divide the tasks and APIs to make it easier to work and more efficient.

Major Obstacles

We had several problems that did not allow us to move forward quickly.

- > The first thing would be to upload the python codes to Git in the different branches. The inexperience resulted in errors and conflicts in the commit.
- > Another problem was getting the data from the "Climatiq" API. This API, which was more efficient than Carbon Interface, did not allow us to extract the data. Therefore, we had to change the API.
- > We try to do Web Scraping, but Google Maps don't let us

Thank You.



Project by
Gema, Laura, Miqueas
November 2024