

CART●

CARTOframes

A Python Library
for Spatial Data Science



London | Nov. 19 - Nov. 22 2019

Álvaro Arredondo
Data Scientist
aarredondo@carto.com

Follow along at
<https://github.com/arredond/odsc-e19>

“Spatial is special”

...is it really?

HOW IT WORKS

CARTO turns your Location Data Into Business Outcomes

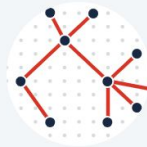
Whether it's more efficient delivery routes, strategic store placements or targeted geomarketing campaigns - CARTO makes it simple in 5 key steps:



**Data
Ingestion**



**Data
Enrichment**



Analysis



**Solutions &
Visualization**



Integrations

Follow along at
<https://github.com/arredond/odsc-e19>

```
pip install --pre cartoframes
```

Follow along at
<https://github.com/arredond/odsc-e19>

```
import cartoframes  
cartoframes.__version__
```

```
'1.0b5'
```

<https://github.com/CartoDB/cartoframes>

Follow along at
<https://github.com/arredond/odsc-e19>

The screenshot shows the CARTOframes documentation page. The header includes the CARTO logo and navigation links: Developers, Fundamentals, Libraries, APIs, Others, and a search icon. The main heading is "CARTOframes" with a "BETA RELEASE" badge and version "v1.0b4". Below this is the tagline: "A Python package for integrating CARTO maps, analysis, and data services into data science workflows." The navigation menu includes Overview, Guides, Reference, Examples (which is underlined), and Support. On the left, there are two sidebar sections: "Data Workflows" with links like "Create Dataset from CSV", "Create Dataset from JSON", "Create Dataset from GeoJSON", "Create Dataset from SQL Query", "Download Dataset from CARTO", "Upload Dataset to CARTO", and "Change Dataset Privacy"; and "Layers" with links like "Add Layer", "Add Default Widget", "Add Default Legend", "Add Popup on Click", "Add Popup on Hover", and "Add Basic Style". The main content area is titled "Create a Dataset from a CSV file" with a Python icon. It states: "This example illustrates how to create a Dataset from a CSV file using pandas". Below this is a code block with the following Python code:

```
import pandas

remote_file_path = 'http://data.sfgov.org/resource/wg3w-h783.csv'

df = pandas.read_csv(remote_file_path)

# Clean latitude and longitude values that are NaN
df = df[df.longitude == df.longitude]
df = df[df.latitude == df.latitude]

df.head()
```

At the bottom of the code block, there is a table with 10 columns: incident_datetime, incident_date, incident_time, incident_year, incident_day_of_week, report_datetime, row_id, incident_id, incident_num, and incident_name. A green chat bubble icon is located at the bottom right of the code block.

<https://carto.com/developers/cartoframes/examples/>

CART0frames: only visualization?

Map().publish()

Communicating Results

```
map_viz.publish(  
    name='sustainable_palm_oil_production_mills_map',  
    password='112358'  
)  
  
{  
    'id': 'cd919833-5bcd-47a5-a1b5-f66c5d390304',  
    'url': 'https://team.carto.com/u/johnsmith/kuviz/cd919833-5bcd-f66c5304',  
    'name': 'sustainable_palm_oil_production_mills_map',  
    'privacy': 'private'  
}
```

Data Enrichment



A one-stop shop for spatial data

HOW IT WORKS

CARTO offers a wide range of datasets from around the globe accessible through Data Observatory, our spatial data repository. Select the data category and country you're interested in and you'll see what we've got available.



Financial



Human Mobility



Demographics



Housing



Road Traffic



Points of Interest



Environmental



Global Boundaries

Data Enrichment

```
from cartoframes.data.observatory import CatalogDataset
from cartoframes.data.observatory import Enrichment
from geopandas as gpd

original_df = gpd.read_file(file) # Point Data

# Discovery

dataset =
CatalogDataset.get('carto-do-public-data.usa_acs.demographics_acs_usa_censust
ractclipped_2015_5yrs_20132017')

dataset.is_public_data
# True

dataset.variables.get('no_cars_d19dfd10') # dataset.variables
# <Variable('no_cars_d19dfd10', 'The number of households without car')>
```

Data Enrichment

```
from cartoframes.data.observatory import CatalogDataset
from cartoframes.data.observatory import Enrichment
from geopandas as gpd
```

```
# Enrichment
```

```
enrichment = Enrichment()
enriched_dataset_df = enrichment.enrich_points(
    original_df,
    variables=['no_cars_d19dfd10']
)
```

Location Based Services

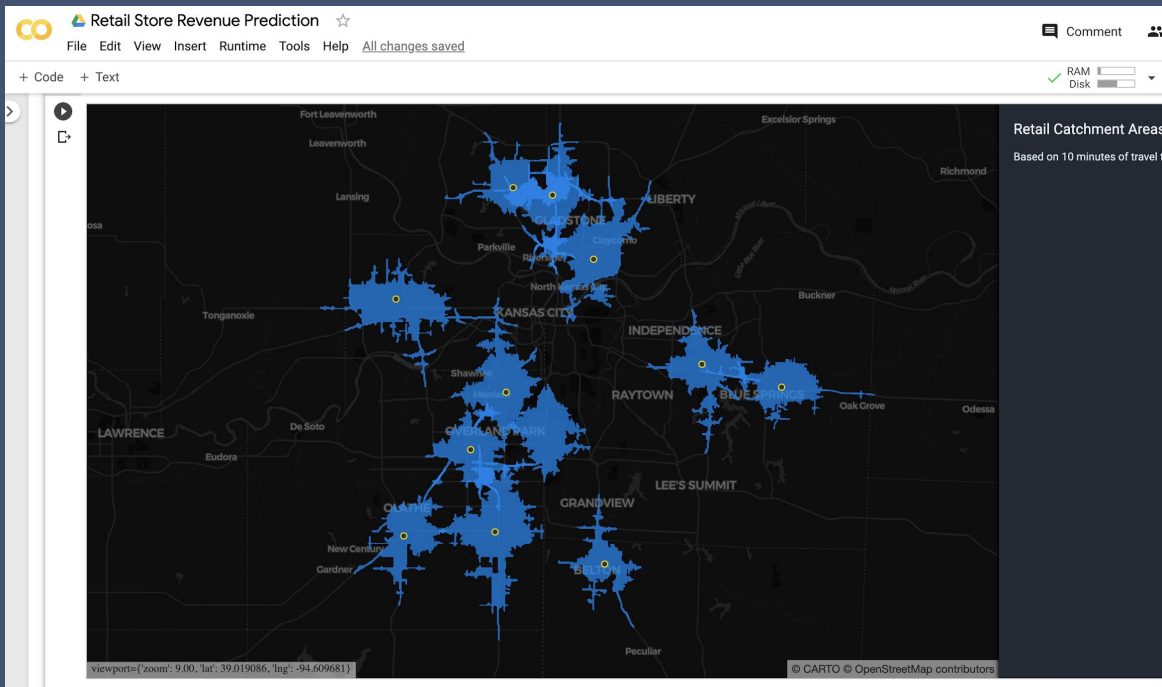
Location Based Services

Geocoding

```
gc = Geocoding()  
geocoded_dataframe, info =  
gc.geocode(df,  
            street='address',  
            city='city',))
```

Isochrone (second)

```
iso_service = Isolines()  
isochrones =  
iso_service.isochrones(  
    geocoded_dataframe,  
    [600, 900, 1200, 1500, 1800],  
    mode='car')
```



Follow along at
<https://github.com/arredond/odsc-e19>

Thank you!

Feedback form:

<http://bit.ly/CF-feedback>

Sign up: <https://carto.com/signup>

Student?

<https://education.github.com/pack>