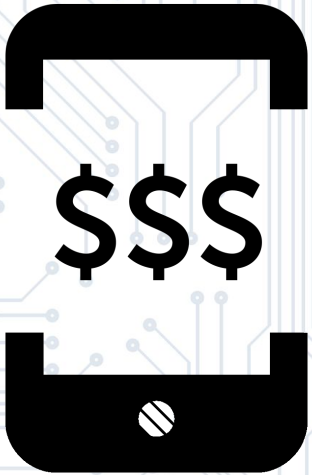


Plotting with Plotly Express

FinTech

Lesson 6.2

Class Objectives

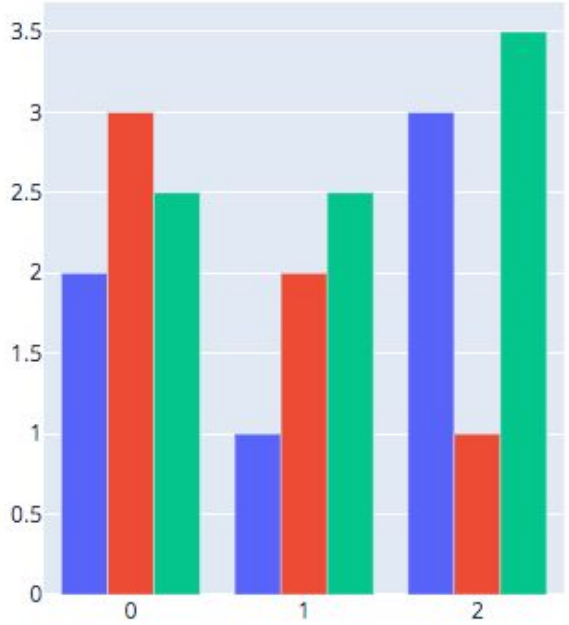
By the end of class, you will be able to:

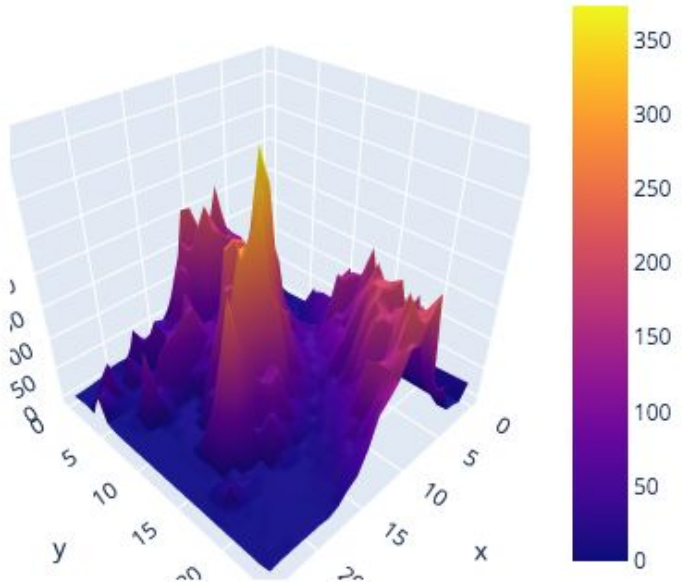
- Define common uses cases for Plotly Express
- Set up Plotly Express Environment
- Complete Plotly Interactive Plots
- Store MapBox API key as environment variable and authenticate
- Integrate MapBox API with Plotly
- Construct Map Plot Visualizations

plotly

Plotly Express

Plotly Express is a package similar to hvPlot, offering many of the same plots as hvPlot (i.e. bar, line, scatter, etc.) but more as well (i.e. Parallel Coordinates and Parallel Categories plots).

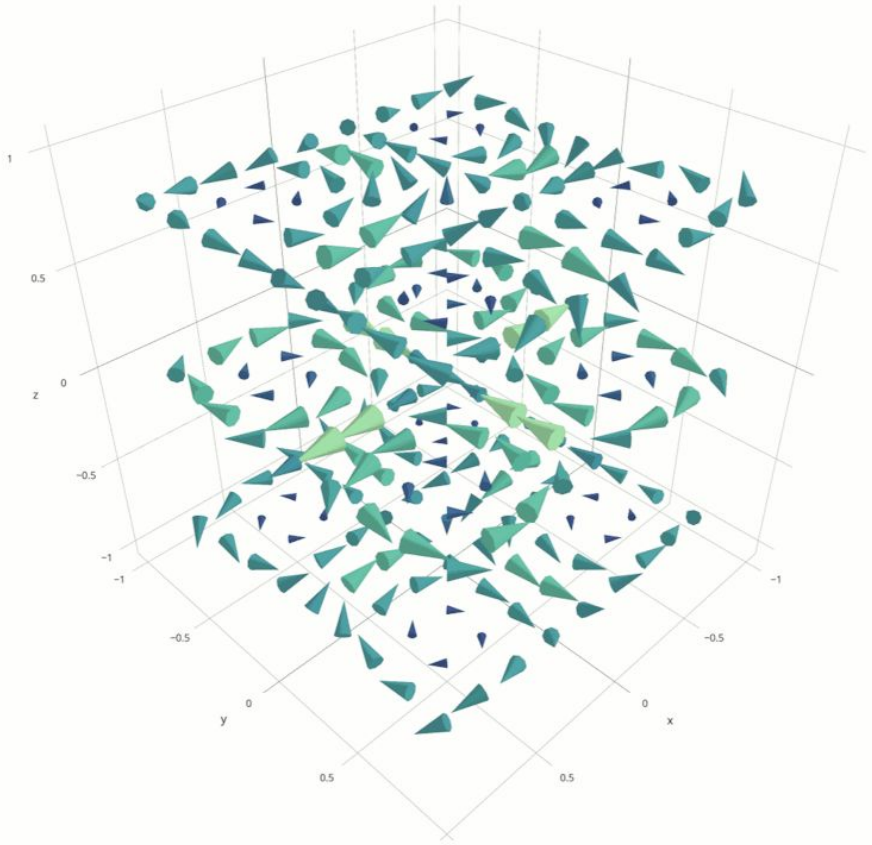




plot.ly/python/v4-migration/

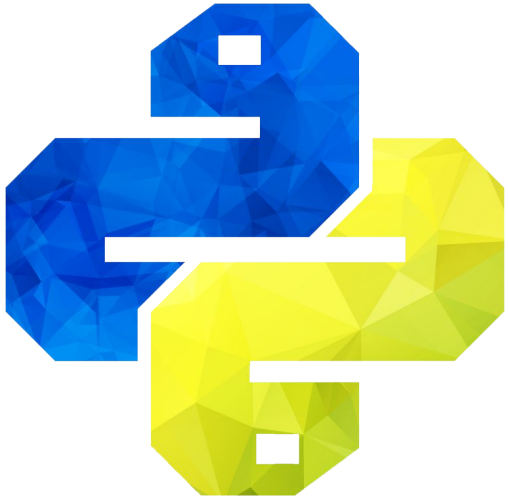
Plotly Express

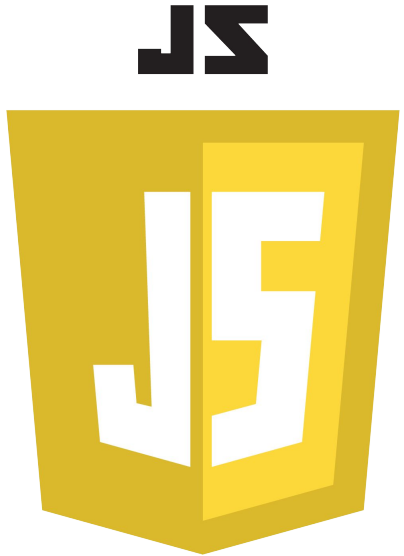
Plotly Express is a favorite among the data science and web-based data visualization communities.



Plotly Express

Plotly Express is a leader in data visualization and supports multiple programming languages, like Python, JavaScript, and R.





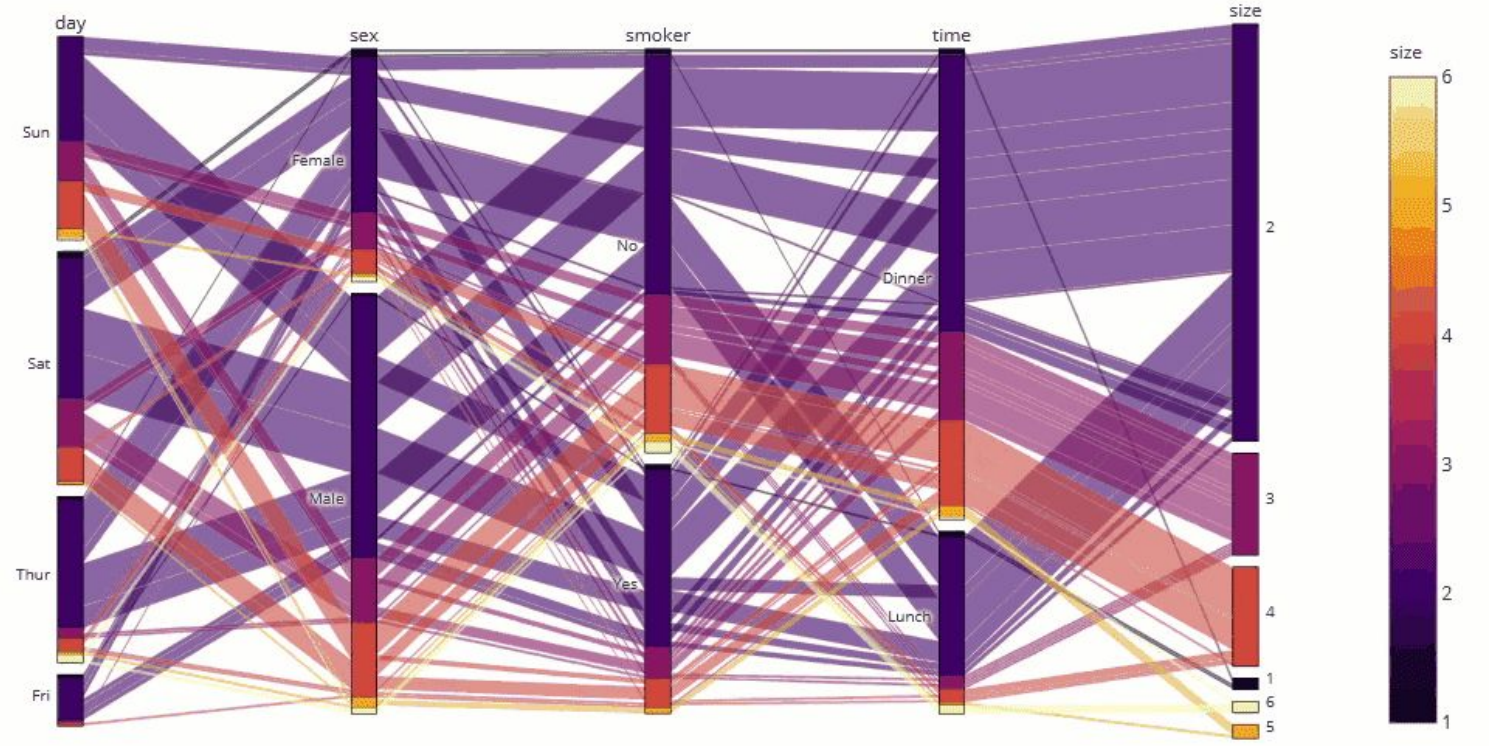
Plotly Express

Plotly Express offers advanced statistical and financial charts that are lacking in technologies like hvPlot, Matplotlib, and Pandas.

plot.ly/python/candlestick-charts/

Plotly Express

Plotly Express works by giving users a simple `plot` based interface that allows developers to create and customize interactive visualizations.



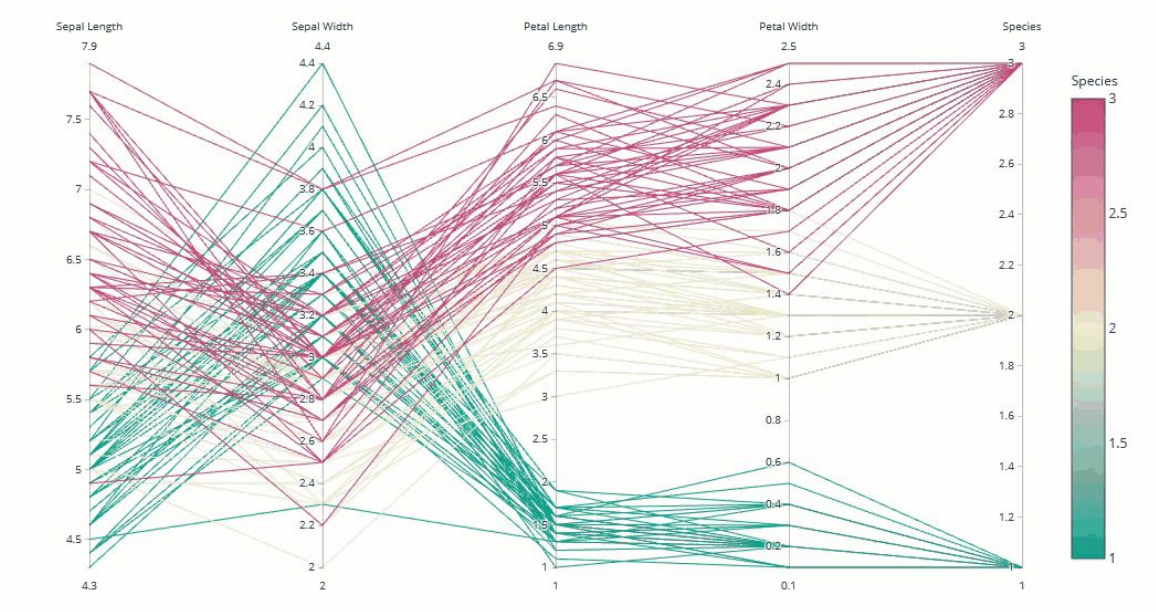
Plotly Express

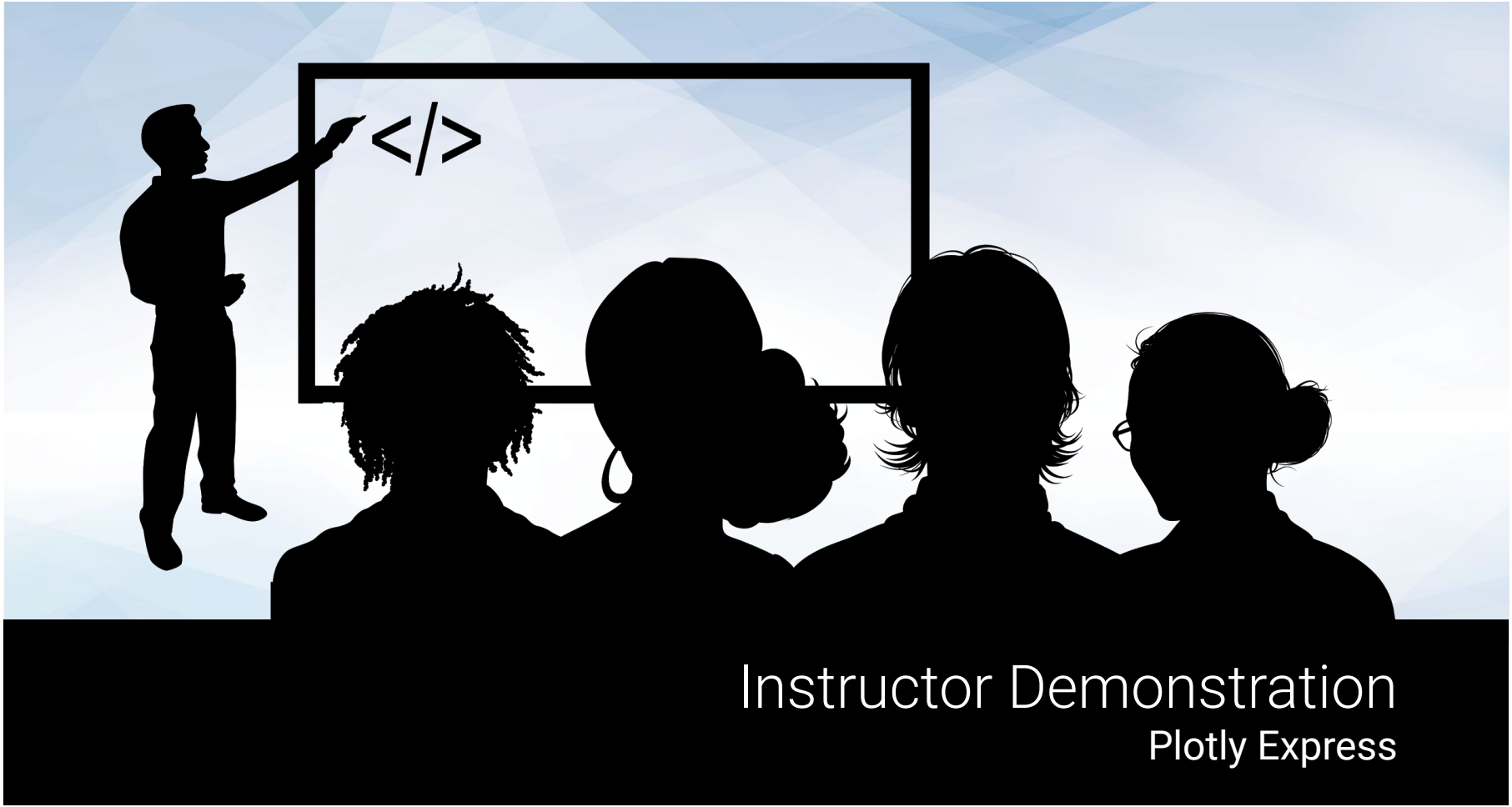
Plotly Express is packaged and powered by the Plotly library, an open source graphing library for Python.

plot.ly/graphing-libraries/

Plotly Express

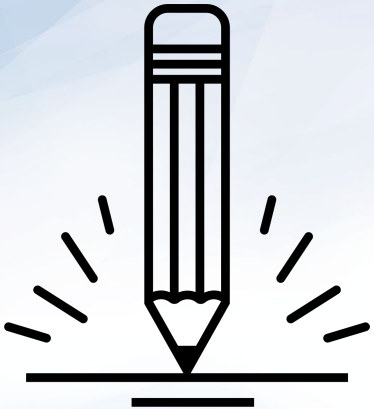
In addition to the chart types we've already seen (i.e. scatter, line, and bar), Plotly Express also includes charting types such as Parallel Coordinates and Parallel Categories: plot types that are useful when visualizing correlations and the relationships between data points.





Instructor Demonstration

Plotly Express



Activity: Plotting with Plotly

Instructions sent via Slack.

Suggested Time:
10 minutes

Time’s Up! Let’s Review.

Parallel Coordinate Plot

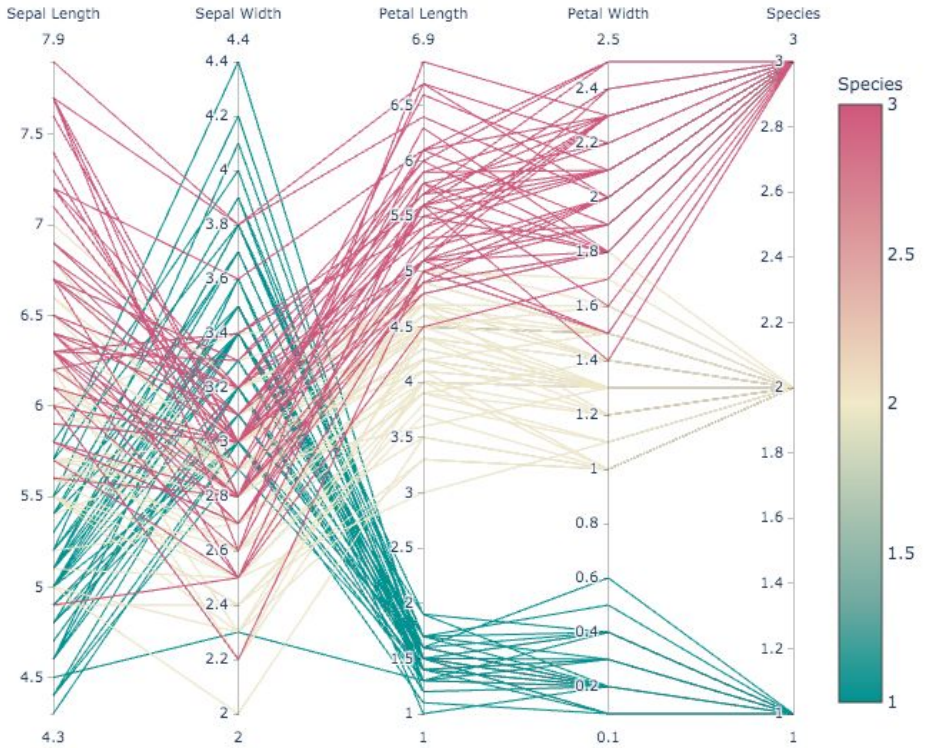
Parallel Coordinate Plots

Parallel Coordinate plots allow for multiple variables to be represented in parallel to one another. This is particularly valuable when tracing the relationships between variables and how each variable relates to/affects the other.

```
import plotly.express as pxiris = px.data.iris()

fig = px.parallel_coordinates(
    iris,
    color="species_id",
    labels={"species_id": "Species",
            "sepal_width": "Sepal Width", "sepal_length": "Sepal Length",
            "petal_width": "Petal Width", "petal_length": "Petal Length", },
    color_continuous_scale=px.colors.diverging.Tealrose,
    color_continuous_midpoint=2)
fig.show()
```

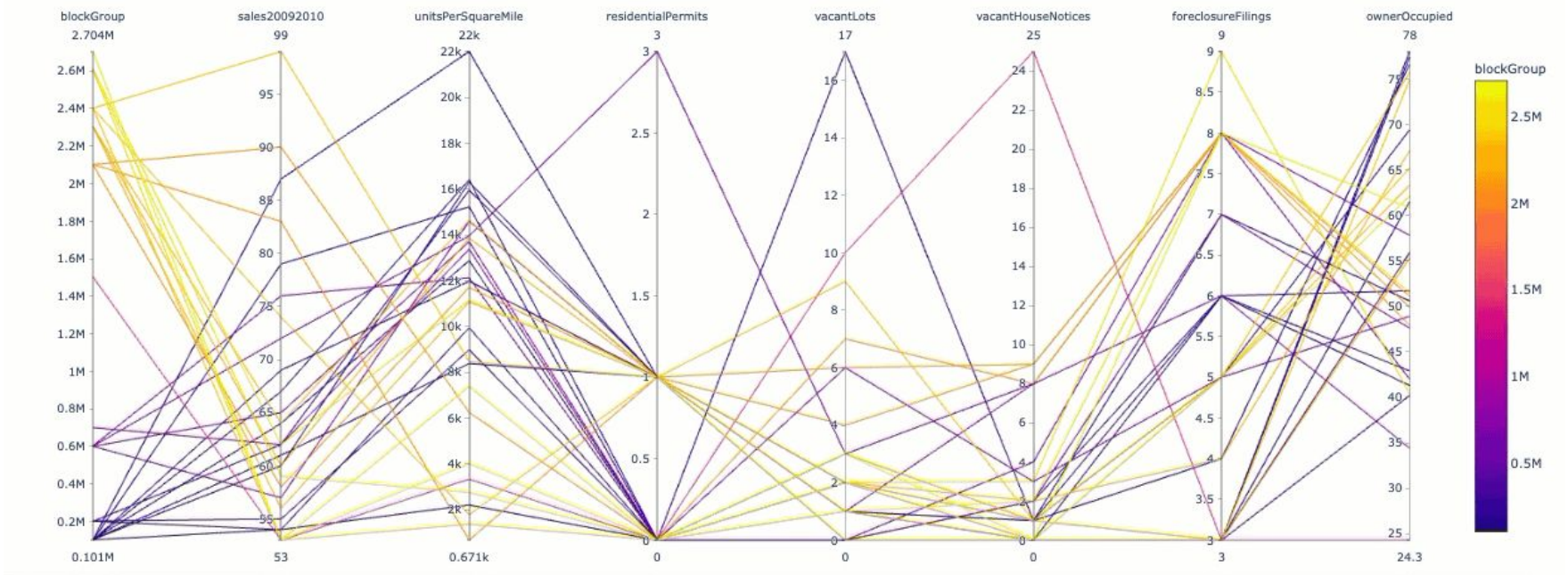
Parallel Coordinate Plots

By sorting the axes and filtering values, analysts can cluster attributes to assess relationships and trends.



Parallel Coordinate Plots

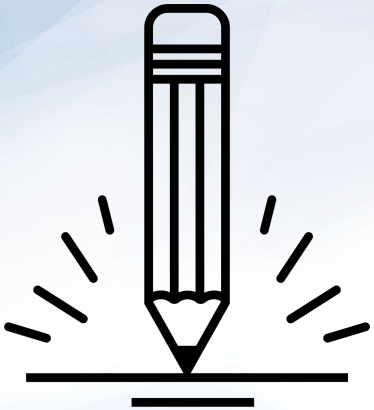
For example, sorting so that vacantLots and sales20092010 are adjacent allows one to see how the number of vacant lots affects the number of sales for that block.



Parallel Coordinate Plots

An assessment of vacantLots, unitsPerSquareMile, and foreclosures reveals that if there are more vacant lots on a block, there will be fewer units per square mile and fewer sales.

```
import plotly.express as px
import pandas as pd
from pathlib import Path

# Read in data
typology =
pd.read_csv(Path('../Resources/housing_market_typology.csv'))[:30].sort_values
('blockGroup')

# Create Parallel Coordinates plot
  px.parallel_coordinates(typology, color='blockGroup')
```



Activity: Plotting in Parallel

Instructions sent via Slack.

Suggested Time:
15 minutes

Time’s Up! Let’s Review.

What's the function used to create a parallel coordinate plot?

ANSWER

```
plotly.express.parallel_coordinates()
```

What's the difference between a scatter plot and parallel coordinate plot?

Scatter Plot and Parallel Coordinate Plot

What's the difference between a scatter plot and parallel coordinate plot?

Scatter Plot

Scatter plots visualize the relationship between two data points as an intersection.

Scatter plots inherently use two axes.

Parallel Coordinate Plot

Parallel coordinate plots visualize the relationship between two data points as parallel axes.

Parallel coordinate are built for multivariate analysis and can have 2+ axes.

In terms of interaction, which plot do you feel allows you to gain more value from interaction?

Scatter Plot and Parallel Coordinate Plot

In terms of interaction, which plot do you feel allows you to gain more value from interaction?

Scatter Plot

The parallel coordinate plot offers limited opportunities for interaction, which makes the scatter plot more fitted for interacting with plots.

Parallel Coordinate Plot

Parallel coordinate plots structurally allow for relationships to be traced more effectively and efficiently.

What is the difference between the types of interactions provided by these different plots?

Scatter Plot and Parallel Coordinate Plot

What is the difference between the types of interactions provided by these different plots?

Scatter Plot

Scatter plots can be zoomed, panned, filtered, etc.

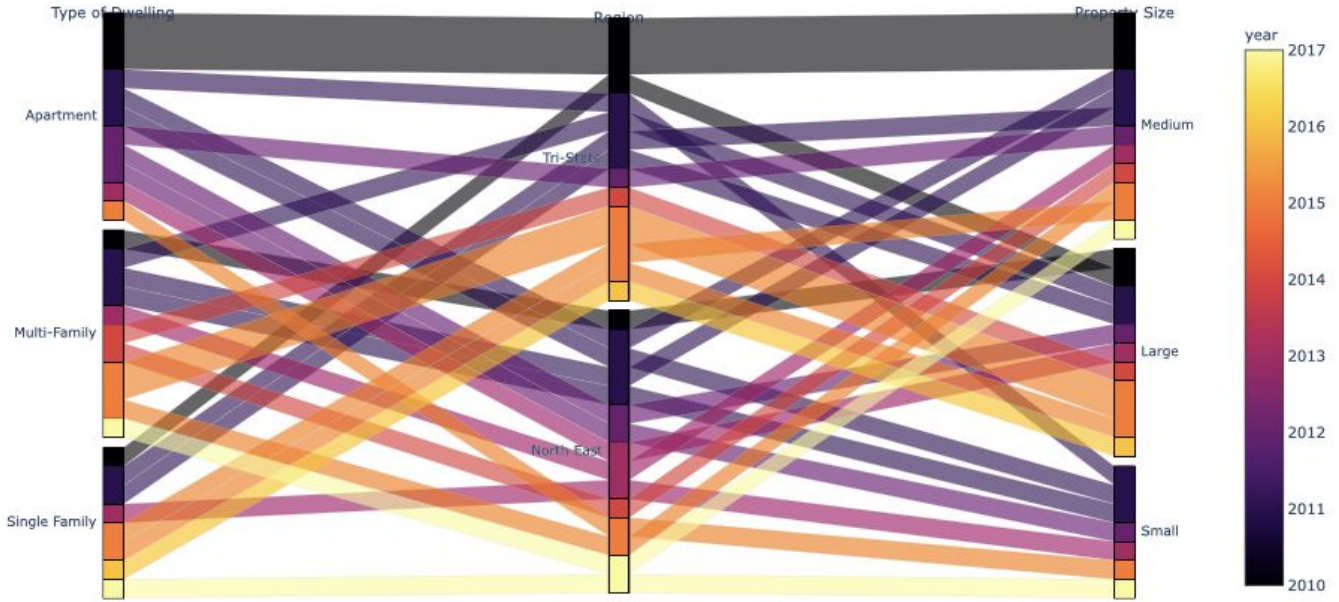
Parallel Coordinate Plot

Parallel coordinate plots can only be sorted and filtered/highlighted.

Parallel Categories

Parallel Categories

Whereas parallel coordinate plots are used for multivariate analysis and mapping relationships between variables, parallel categories plots are used to perform multidimensional analysis.



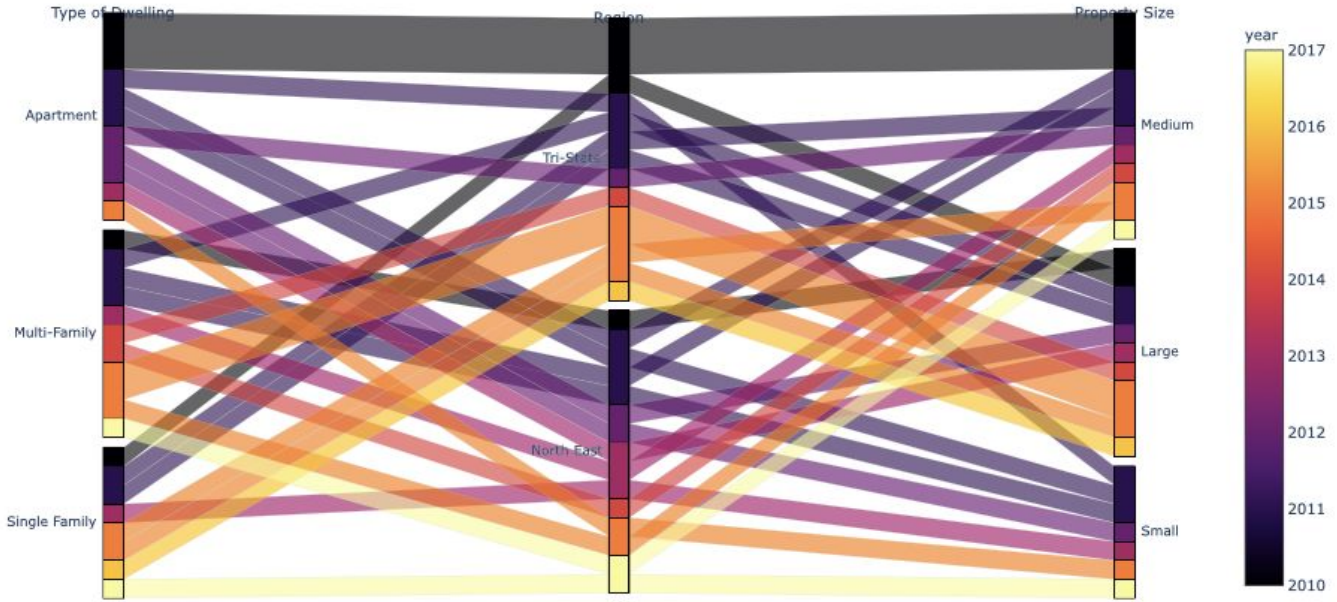
Parallel Categories

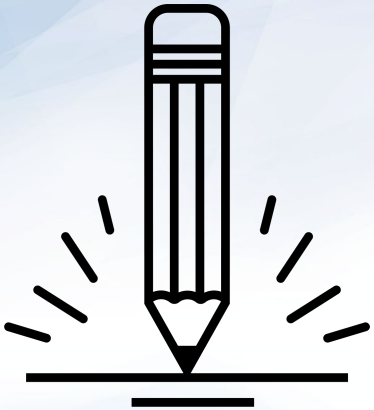
An example of multidimensional analysis would be looking at sales and foreclosures data by housing type, region, and number of units. The dimensions would be housing type, region, and number of units.

```
# Prep Data
housing_type= ['Single Family','Multi-Family','Apartment']
region= ['North East','Tri-State']
prop_size= ['Large','Medium','Small']

df = pd.DataFrame({
    "sold": np.random.randint(999, 1002, 30),
    "year": np.random.randint(2010, 2019, 30),
    "type": np.random.choice(housing_type, 30),
    "region": np.random.choice(region, 30),
    "prop_size": np.random.choice(prop_size, 30)}).sort_values(['year',
                                                                'type',
                                                                'region',
                                                                'prop_size'])

df.head()
```

Parallel Categories

Dimensions are considered to be categories. Parallel categories plots focus on connecting the dots between each category and assessing the nuances per category, as well as the impact of categories on other categories.





Activity: Categorical Property Evaluation Activity Review

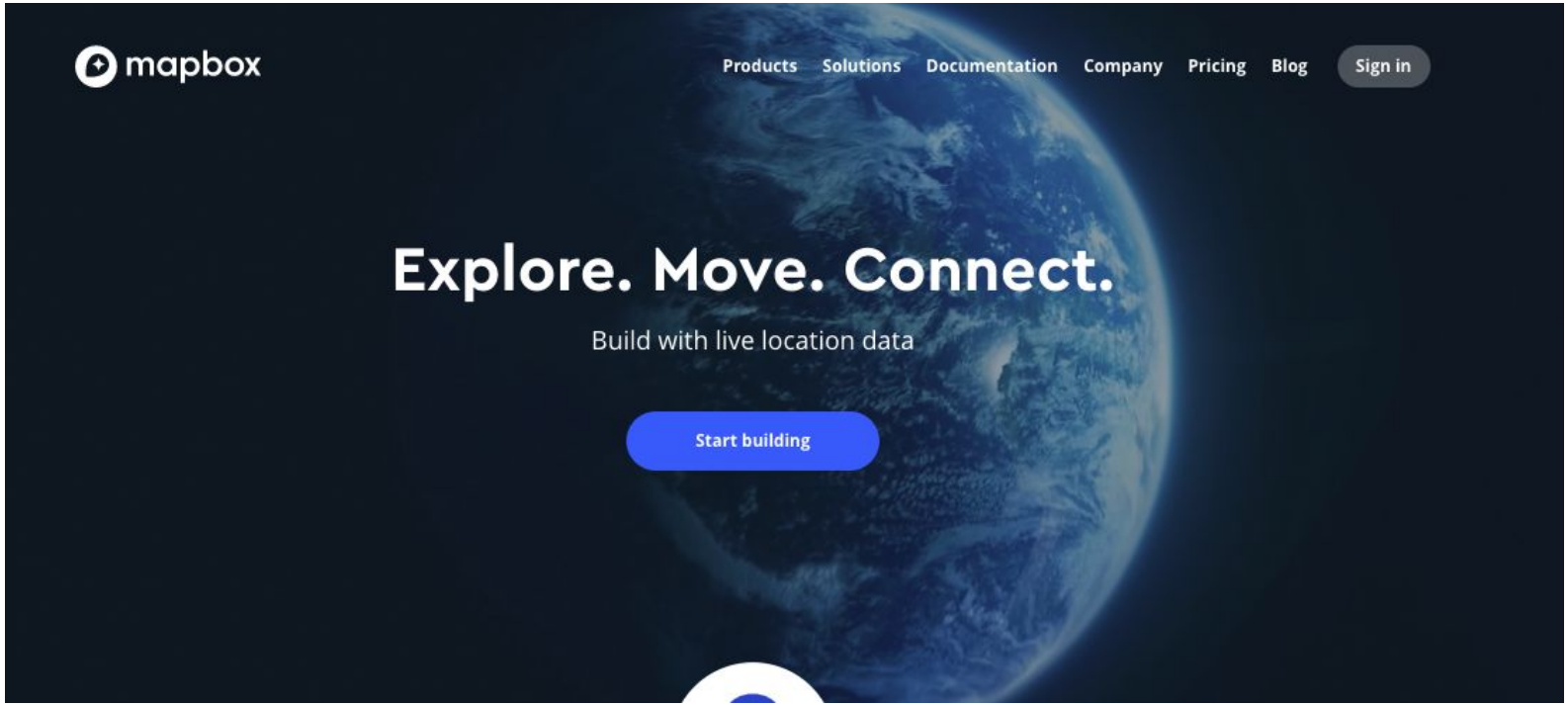
Instructions sent via Slack.

Suggested Time:
15 minutes


mapbox

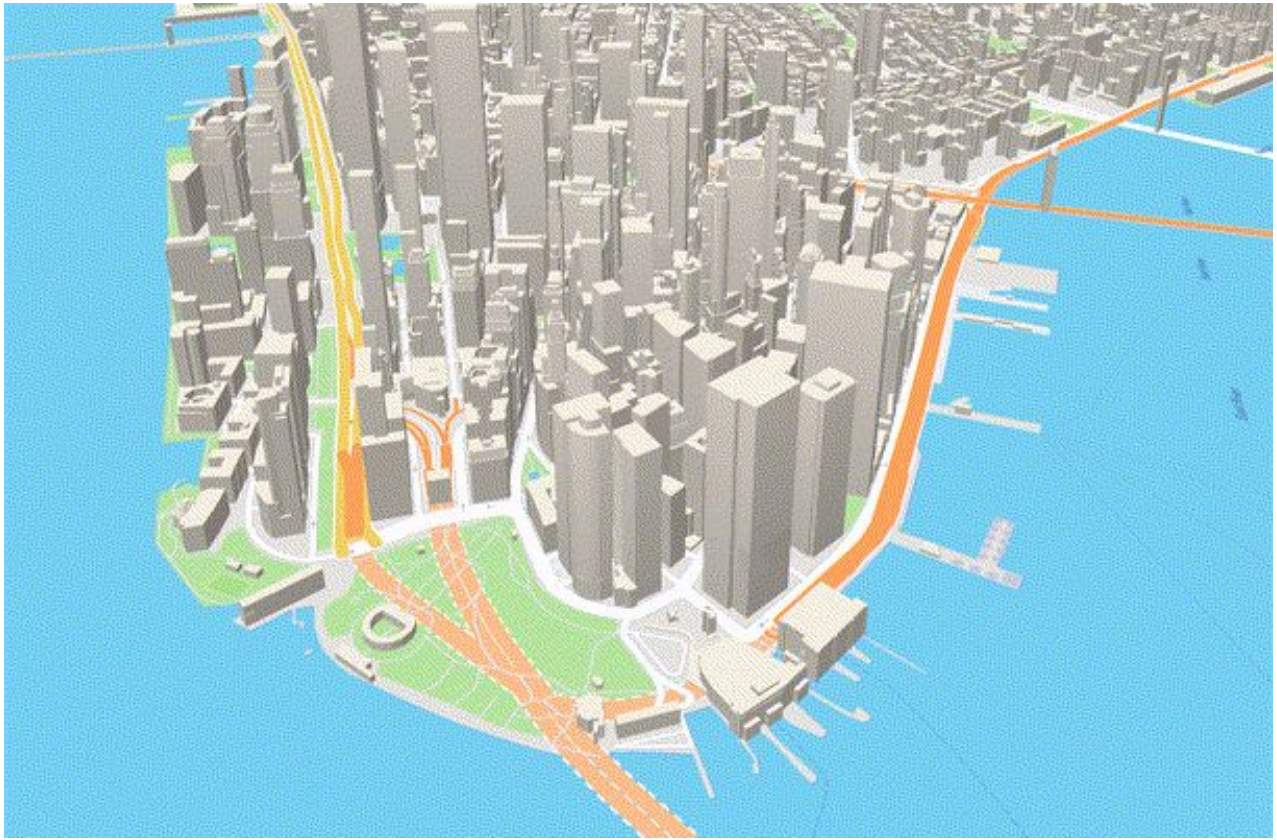
MapBox API

MapBox API is an open source API that gives developers a range of mapping visualizations and functions that enable the creation of interactive map plots.



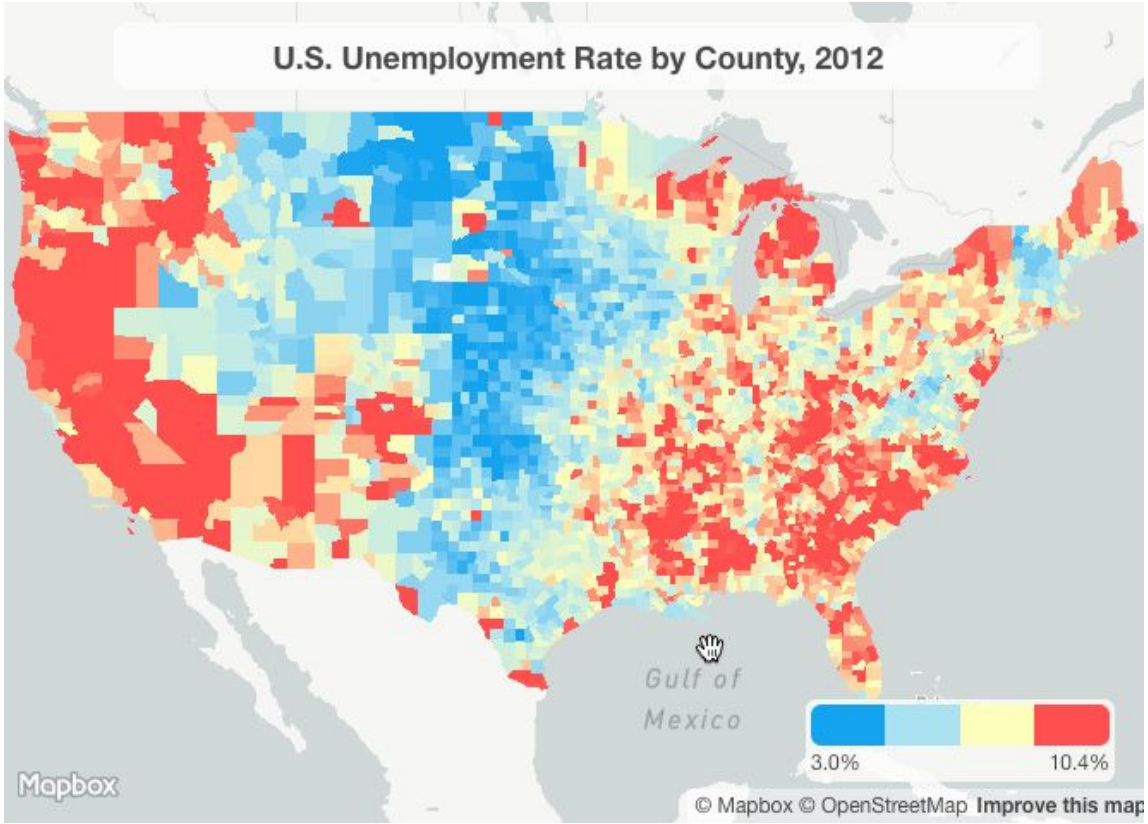
MapBox API

MapBox API is democratizing the map services industry (e.g. navigation and cartography), similar to how Plaid is for FinTech.



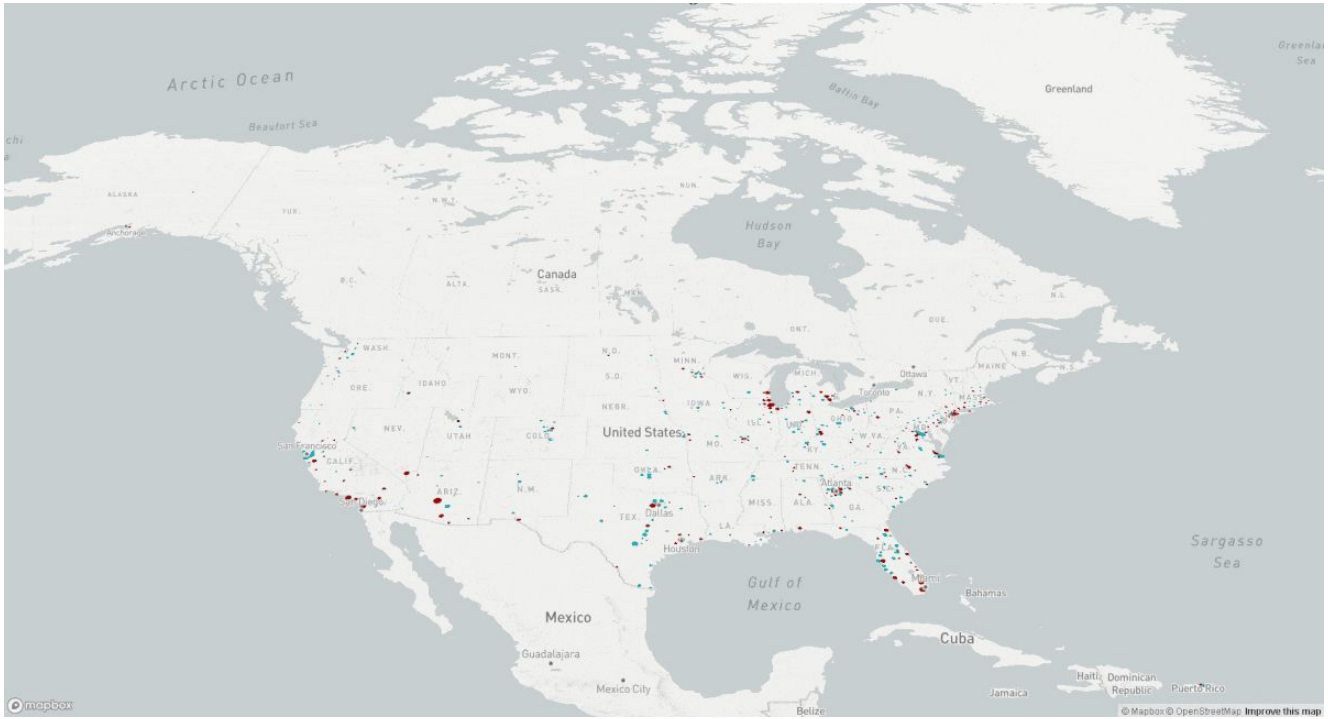
MapBox API

MapBox offers three main services: maps, navigation, and search.



MapBox API

These services come with handy tools, such as map styles and vectors, map images and data sets, and live location.



MapBox API

Plotly Express has an integration endpoint specific for Mapbox API. This allows Plotly to use the Mapbox maps API in order to create interactive map visualizations. Plotly Express has functions designed specifically for interacting with MapBox.

```
import plotly.plotly as py
import plotly.graph_objs as go

# mapbox_access_token = 'ADD_YOUR_TOKEN_HERE'

data = [
    go.Scattermapbox(
        lat=['45.5017'],
        lon=['-73.5673'],
        mode='markers',
        marker=go.scattermapbox.Marker(
            size=14
        ),
        text=['Montreal'],
```

MapBox API

Plotly's integration with Mapbbox makes it extremely convenient to use; no other imports are required. All that is needed is the Plotly Express library.

MapBox API

The Mapbox API uses API keys to monitor API requests. The Mapbox API key needs to be set up as an environment variable. The os.getenv function can then be used to retrieve the key within Python code.

```
import plotly.plotly as py
import plotly.graph_objs as go

# mapbox_access_token = 'ADD_YOUR_TOKEN_HERE'

data = [
    go.Scattermapbox(
        lat=['45.5017'],
        lon=['-73.5673'],
        mode='markers',
        marker=go.scattermapbox.Marker(
            size=14
        ),
        text=['Montreal'],
```

MapBox API

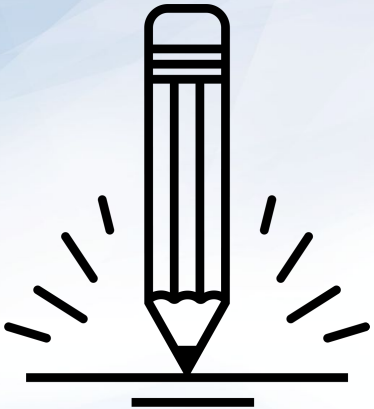
After the token is set with the `set_mapbox_access_token`, the Plotly Express mapbox plot functions can be used to create geographic plots.

```
import plotly.plotly as py
import plotly.graph_objs as go

# mapbox_access_token = 'ADD_YOUR_TOKEN_HERE'

data = [
    go.Scattermapbox(
        lat=['45.5017'],
        lon=['-73.5673'],
        mode='markers',
        marker=go.scattermapbox.Marker(
            size=14
        ),
        text=['Montreal'],
```

MapBox API

The scatter_mapbox function can be used to create a scatter plot that is overlayed on top of a map (provided by Mapbox). This allows for scatter plot data to be analyzed in reference to geographical location.

```
import plotly.plotly as py
import plotly.graph_objs as go

# mapbox_access_token = 'ADD_YOUR_TOKEN_HERE'

data = [
    go.Scattermapbox(
        lat=['45.5017'],
        lon=['-73.5673'],
        mode='markers',
        marker=go.scattermapbox.Marker(
            size=14
        ),
        text=['Montreal'],
```



Activity: Mapping Adventures

Instructions sent via Slack.

Suggested Time:
15 minutes

Time’s Up! Let’s Review.

Challenge:

A Cartographer's Expedition

Instructions sent via Slack.

Suggested Time:
20 minutes

Time’s Up! Let’s Review.