Experimental Machine Learning with HoloViz and PyTorch in Jupyterlab

PyData LA 2019
Tutorial

Hayley Song
2019-12-03 (T)

Tutorial Sessions — Tuesday Dec. 3, 2019

	Tutorial Track 1	Tutorial Track 2	Tutorial Track 3
8:00 AM	Breakfast & Registration		
9:00 AM	**Computer Vision with PyTorch** Daniel J. Brooks	**Introduction to H2O AutoML with Python** Franklin Velasquez, Pasha Stetsenko	**Introduction to Data Analysis with Python datatable** Ana Castro Salazar, Pasha Stetsenko
10:30 AM	Break		
11:00 AM	**Kedro + MLflow – Reproducible and versioned data pipelines at scale** Tom Goldenberg	**Web Scraping w BeautifulSoup & Yelp's API** Juan S Vasquez	**Git-ting along with others** Paul Anzel
12:30 PM	Lunch		
1:30 PM	**Reinforcement Learning: Pac-Man** Manu Gopinathan, Malte Loller-Andersen	**Experimental Machine Learning with HoloViz and PyTorch in Jupyterlab** Hayley Song	Code Sprints
3:00 PM	Coffee Break		
3:30 PM	**Analyzing genetic networks using neural networks** Manu Flores	**Turn Python Scripts into Beautiful ML Tools** Adrien Treuille	Code Sprints
5:00 PM			

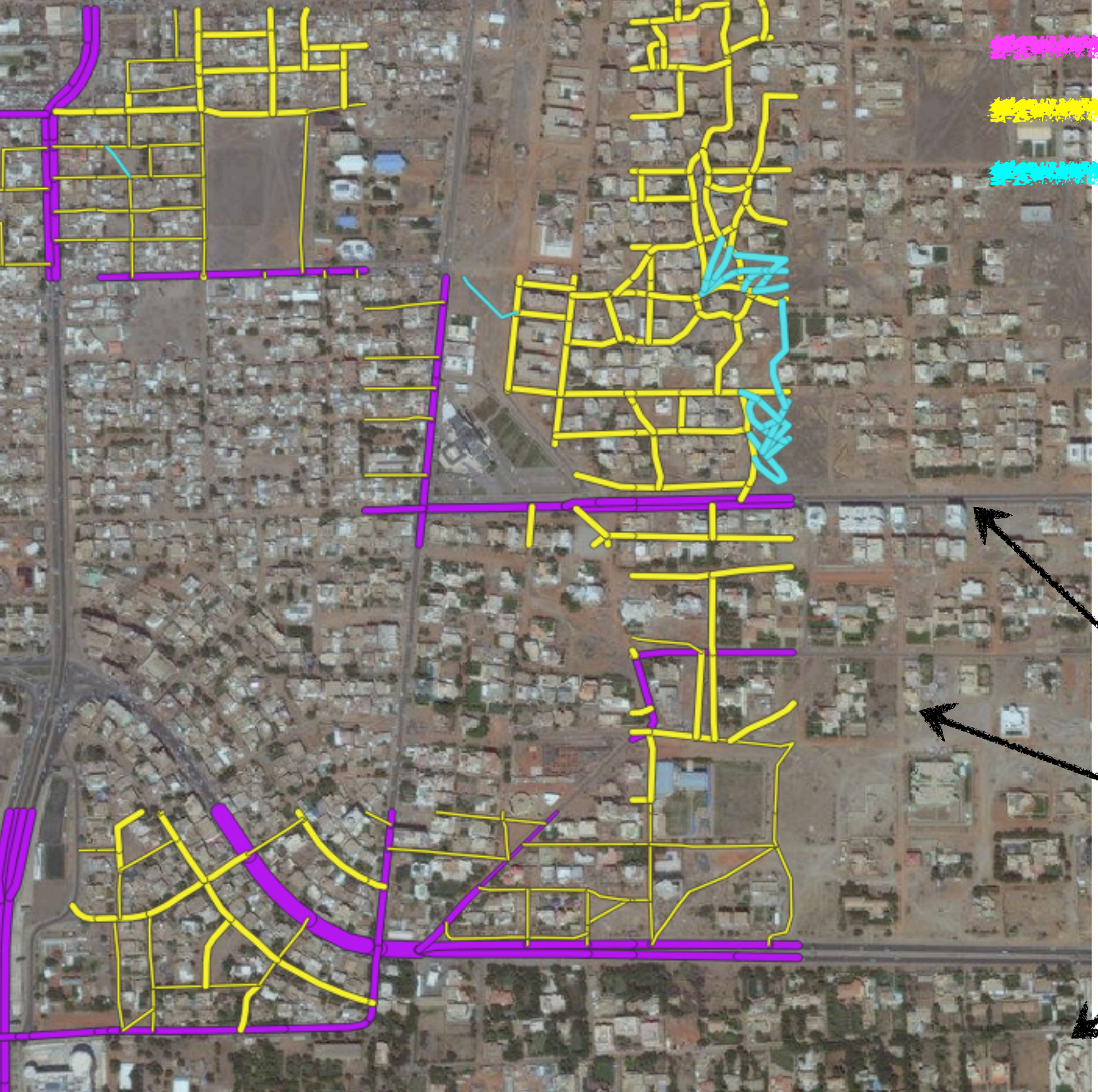
Goal

- Make your data exploration and model building process more interactive and exploratory

- Main tools:
  - JupyterLab/Jupyter Notebook
  - HoloViz (+ param, panel)
  - PyTorch
  - (pandas, geopands, numpy)

Application

- multi-class road detection on satellite images with:
- interactive visualization
- experimental model building



Motorway

Residential

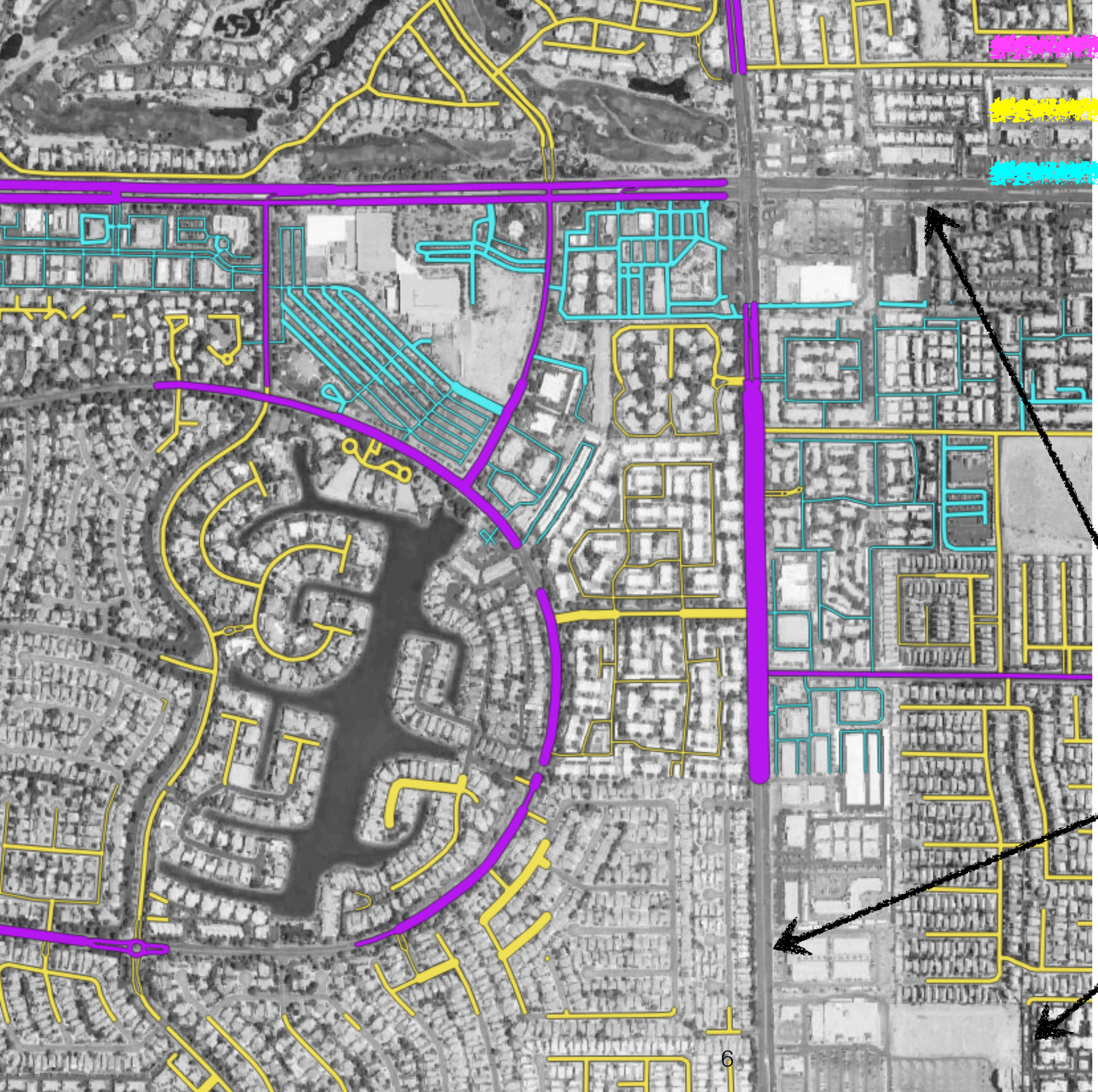
Unpaved

Motorway?

Residential?

Unpaved?

Khartoum, Sudan



Motorway

Residential

Unpaved

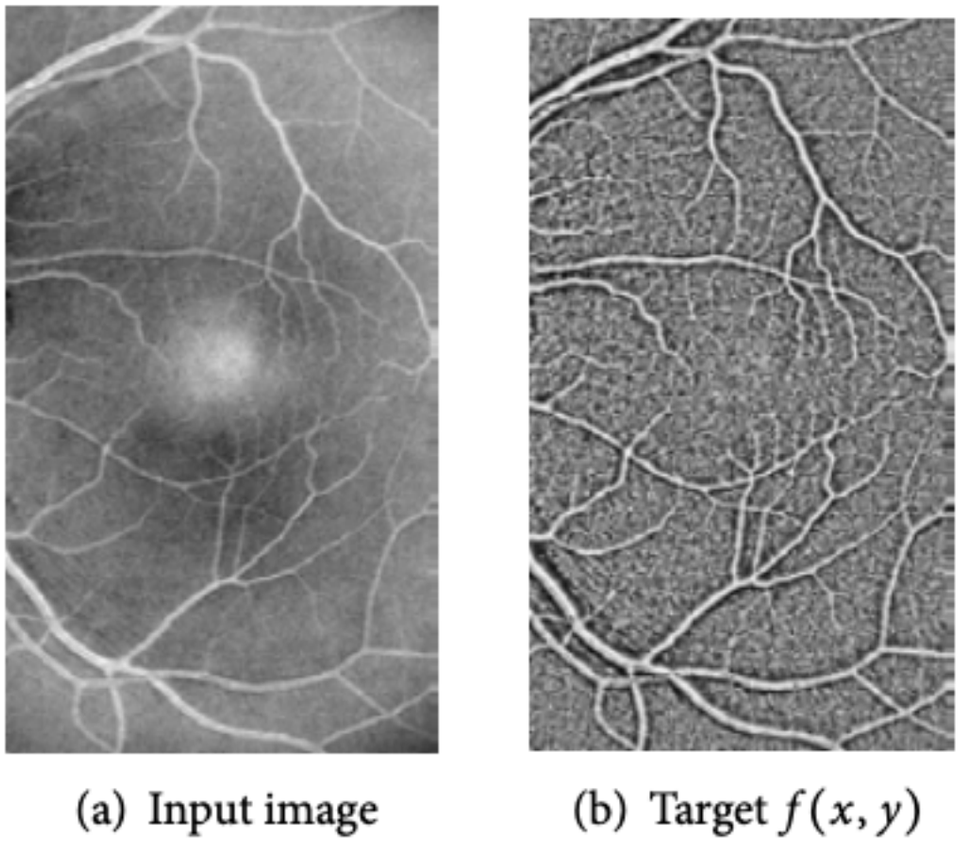
Motorway?

Residential?

Unpaved?

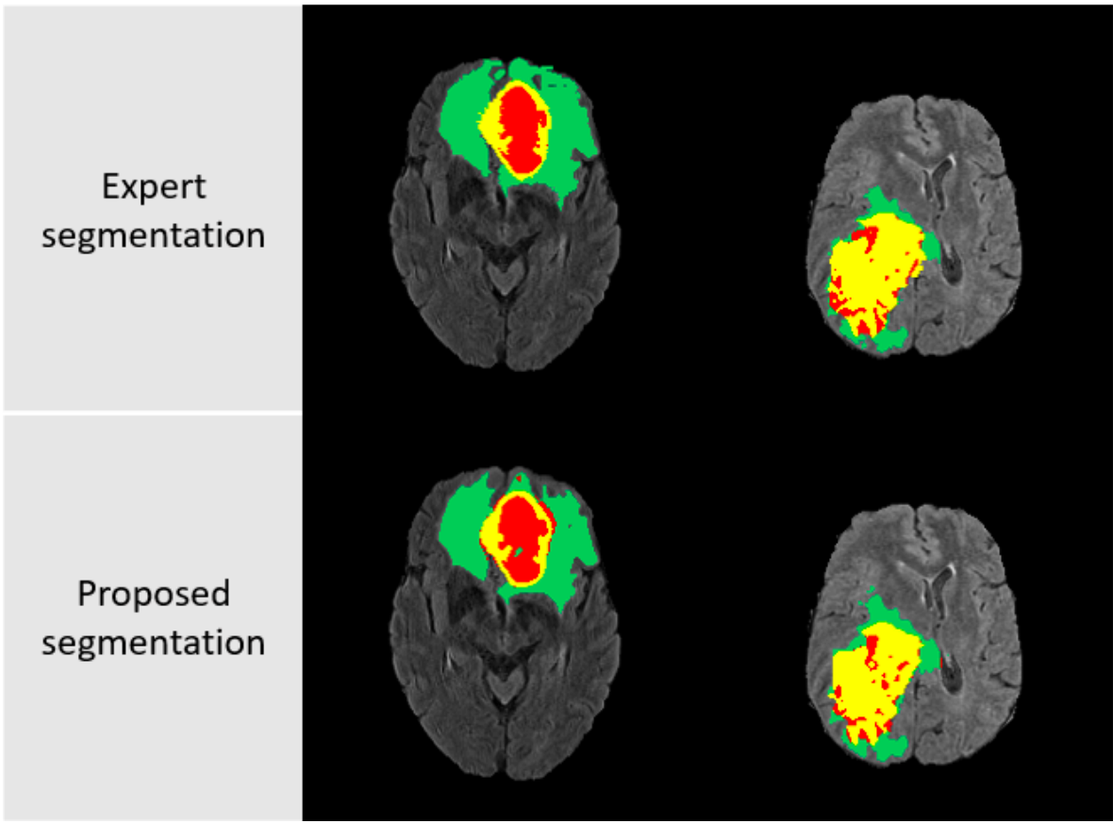
Las Vegas, USA

Similar Segmentation Tasks



(a) Input image (b) Target $f(x, y)$

Level Set Segmentation and Volume Visualization of Vascular Trees
Lathen, 2013

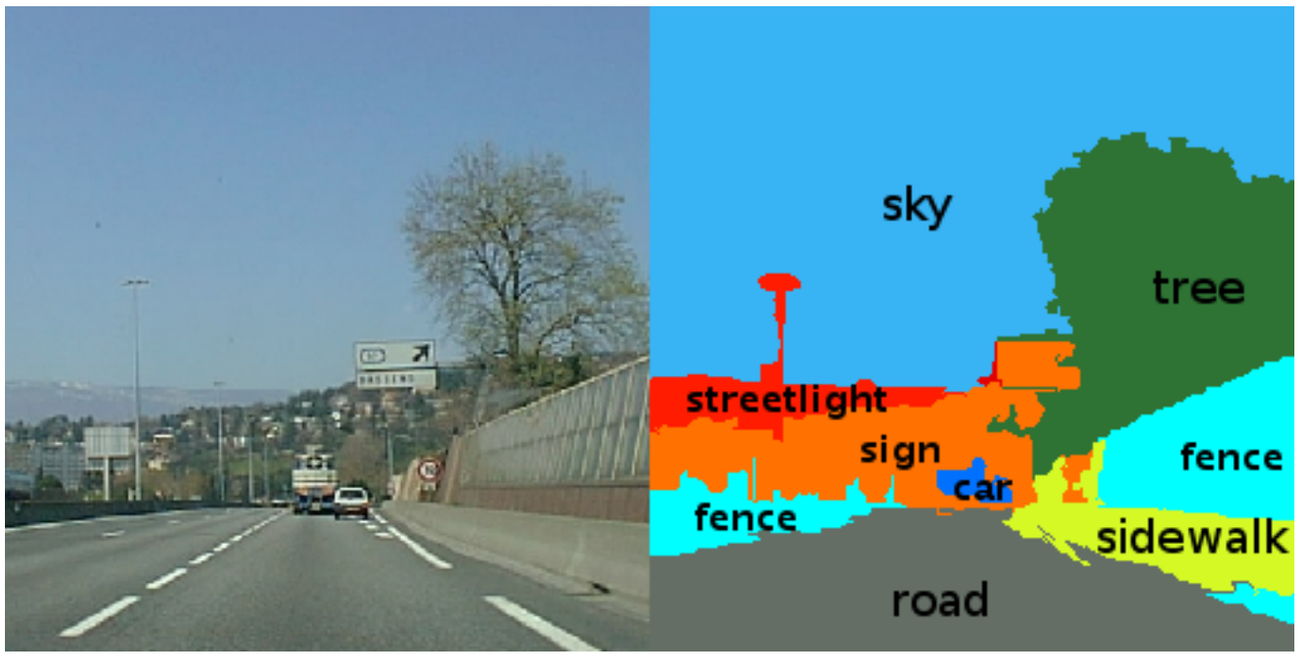


Intel AI: Brain Tumor Segmentation using Fully Convolutional Tiramisu Deep Learning Architecture

Similar Segmentation Tasks

Historical map - weiwei

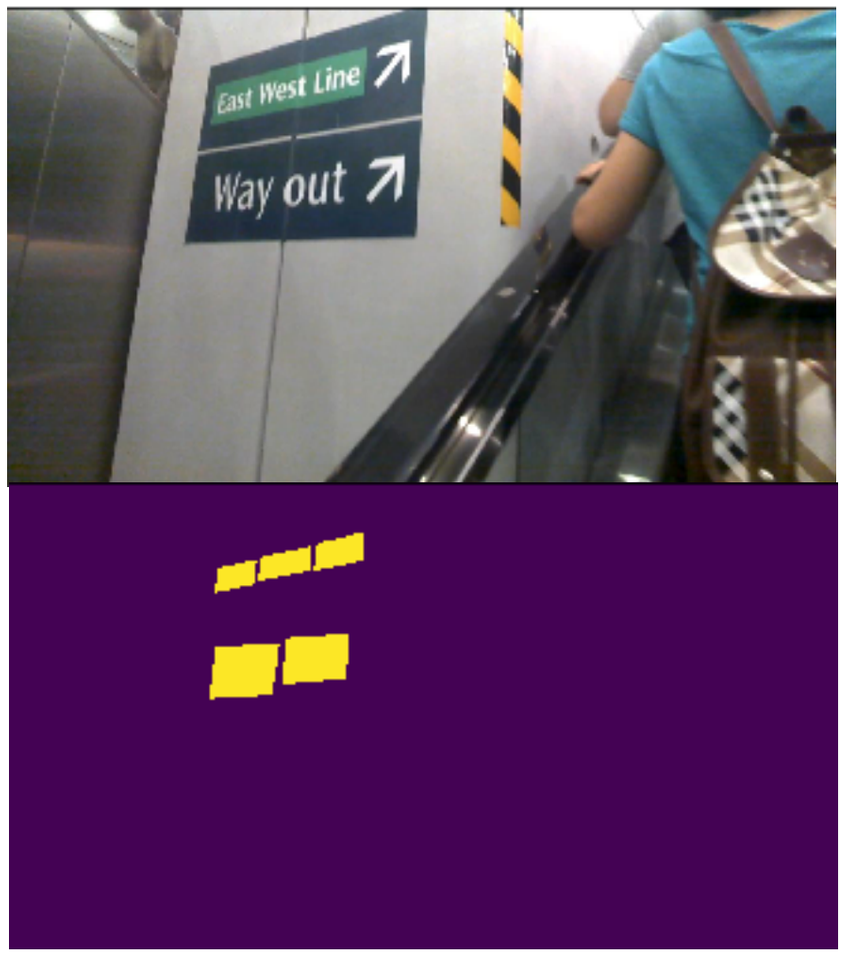
Image source: NVIDIA dev blog



Caesar et al, ECCV 16

Similar Segmentation Tasks

Text detection from historical map and common images



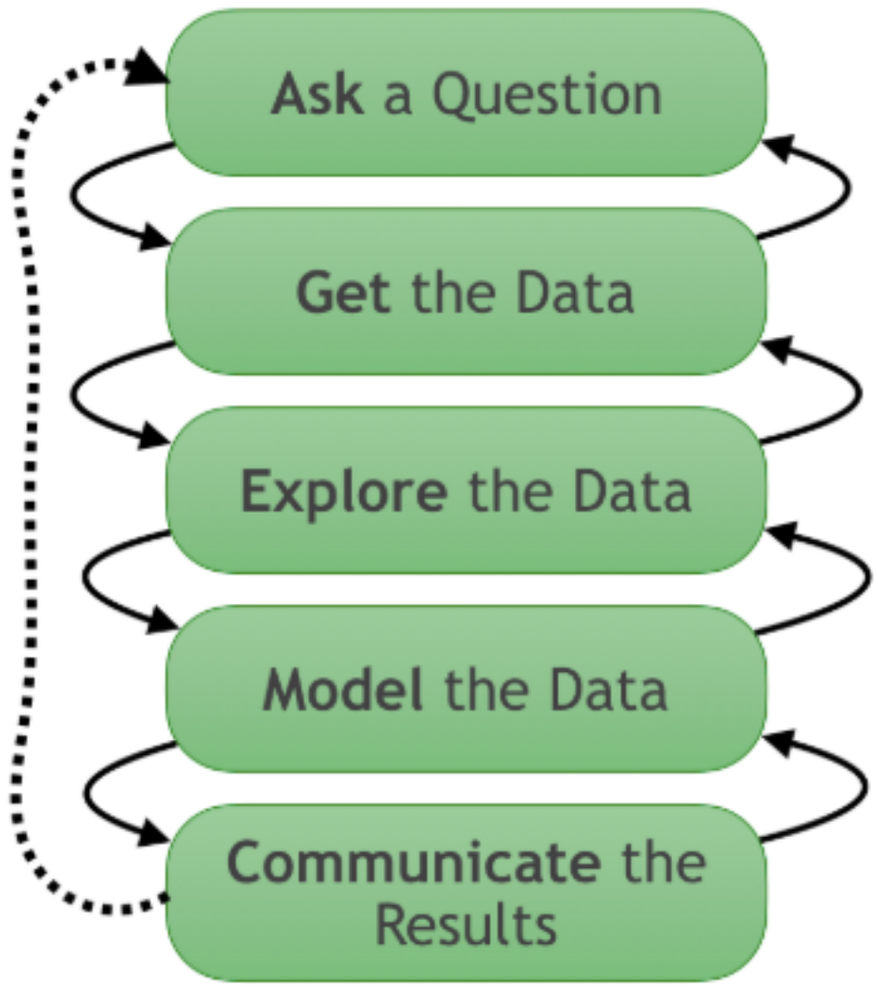
Text detection from historical map and common images

Agenda

- Step 0: Introduction to `Holoviews`/`Geoviews` and `Panel` [15mins]
- Step 1: Explore your dataset with `Holoviews`/`Geoviews` [15mins]
- Step 2: Build an easily-configurable neural network model with `param` [15mins]
- Step 3: Monitor your training process through an interactive GUI [15mins]
- Step 4: Analyze your learned model on new images [15mins]
  - Understand what your model has learned by looking at intermediate feature maps with `Holoviews` and `Panel`
- Q/A [5~10 mins]

Why HoloViz (+friends) for ML?

General ML workflow



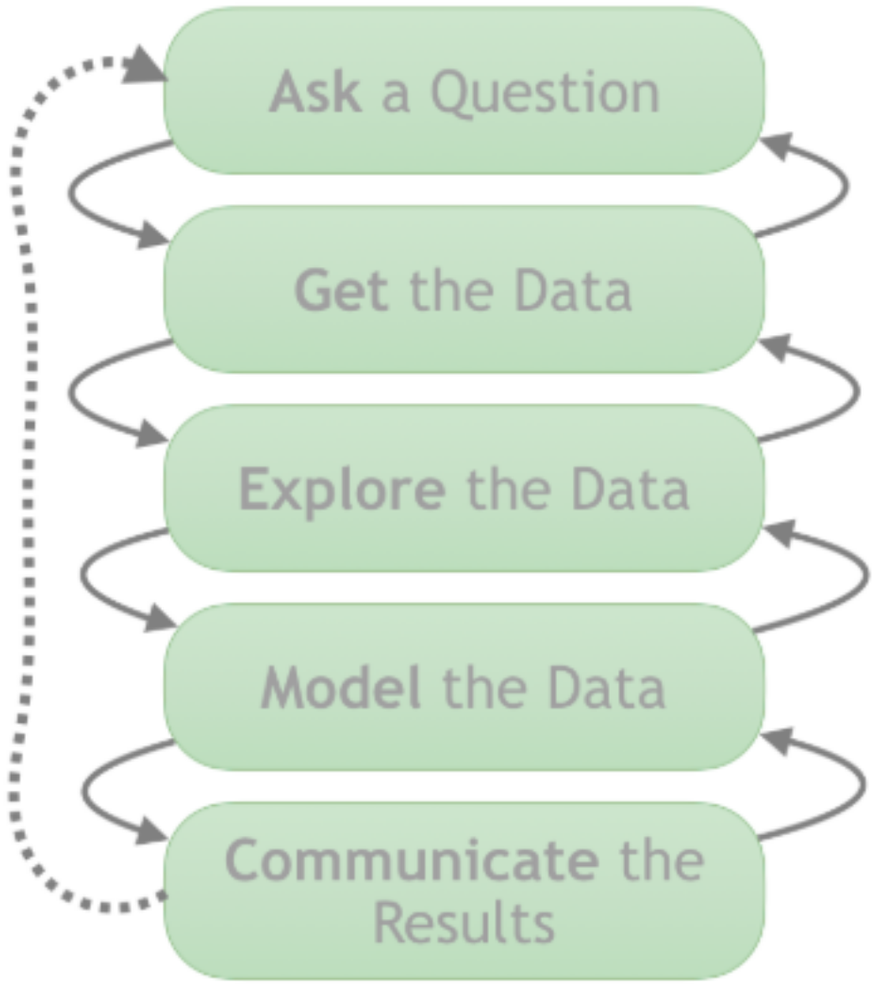
Need to visualize data at **every** step

https://holoviz.org/talks/index.html

Why HoloViz (+friends) for ML?

+ Jupyter

General ML workflow



Need to visualize data at **every** step

https://holoviz.org/talks/index.html

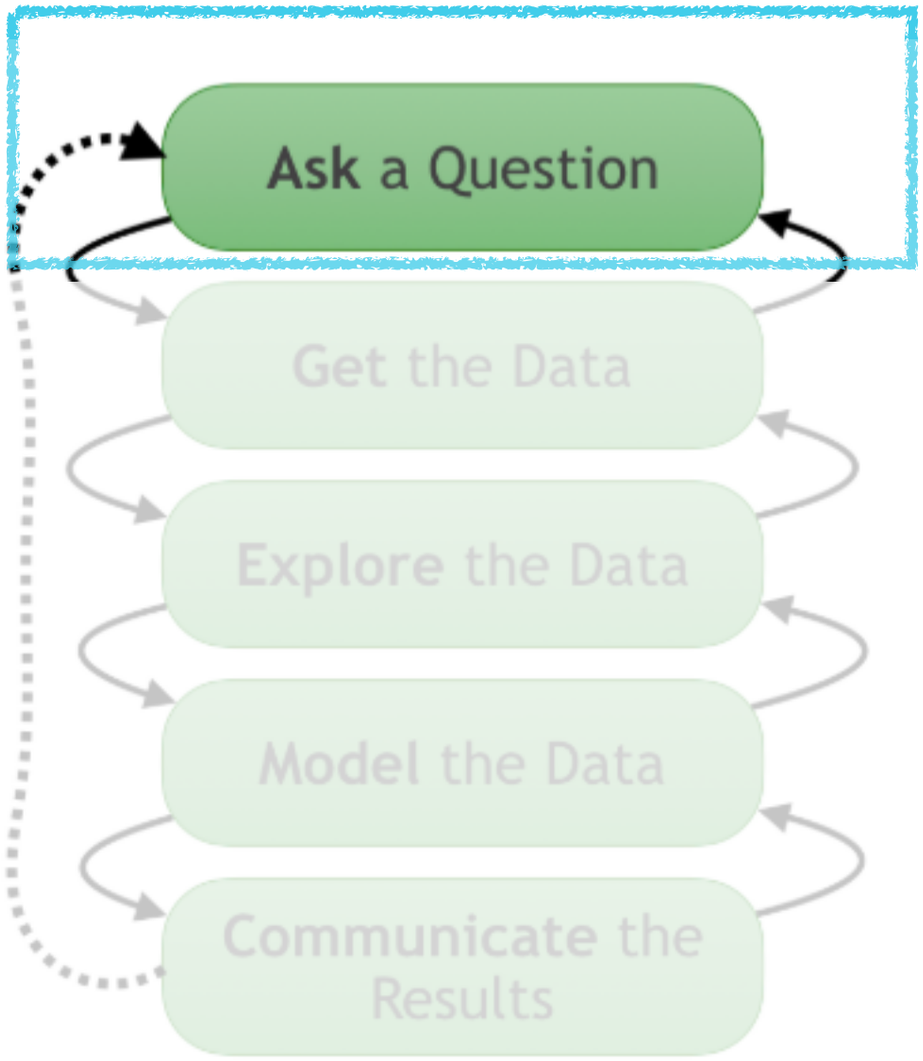
Jupyter Workflow

Initial exploration needs be flexbile. Quickly try different ideas and test hypotheses.

tldr: Jupyter + HoloViz 👍
+ PyTorch 👍👍
+ (Tensorboard)

Go to notebook

For this tutorial…



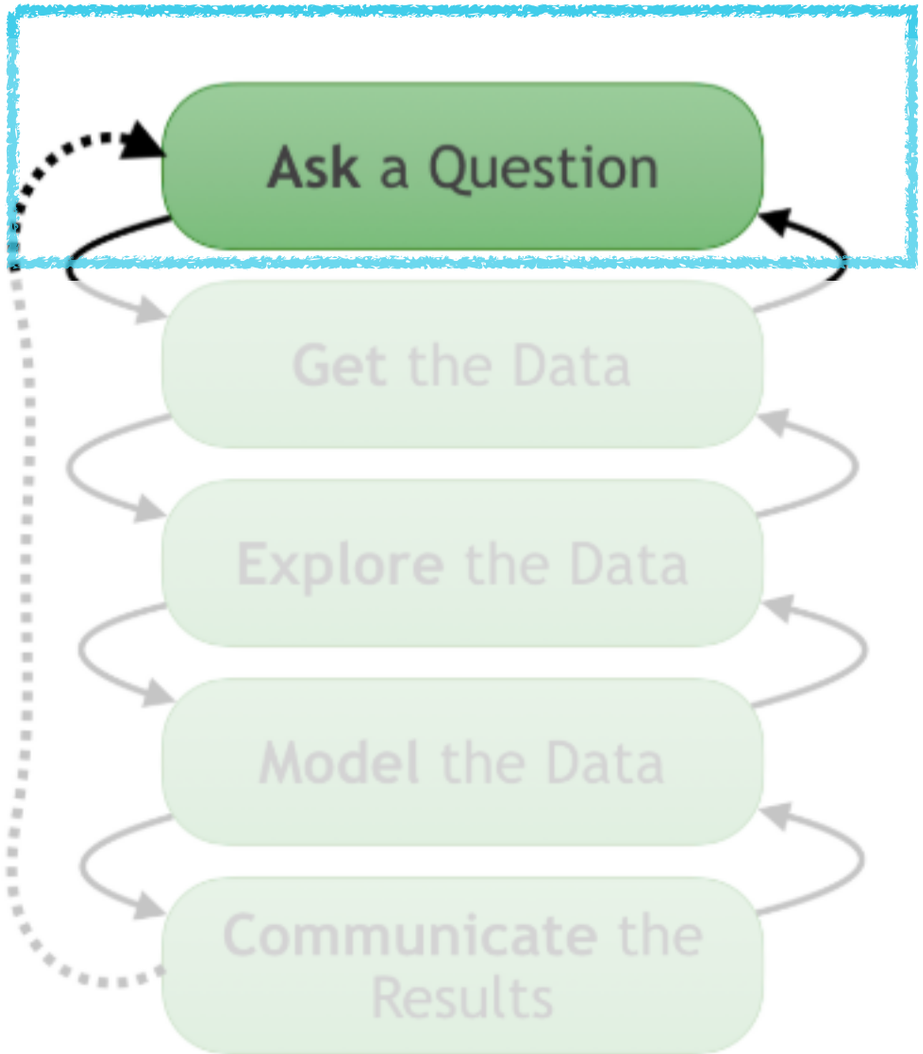
Q1: I want to segment out roads from satellite images

- supervised
- binary classification: road vs. non-road

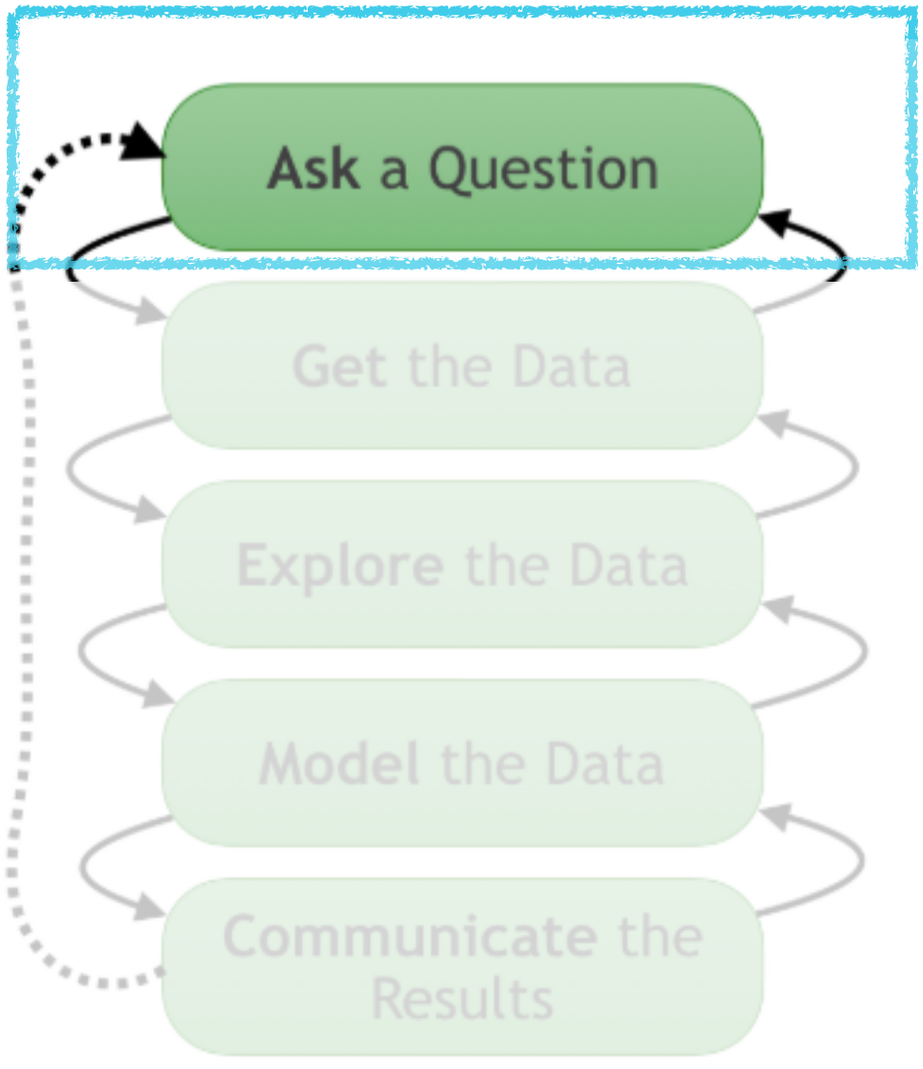
Q2: I want to segment out roads of three different types from satellite images

- supervised: multi-class classification
- highway, residential, footway, non-road

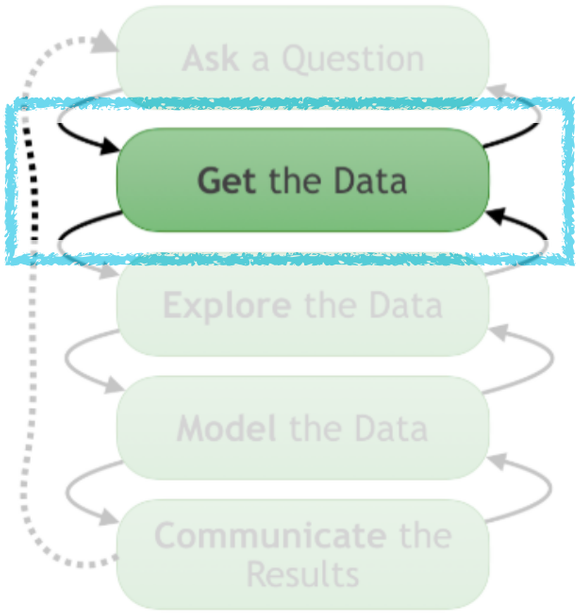
For this tutorial…



For this tutorial…



For this tutorial…




```
[201]:
1 ex = DSExplorer()
2 col = pn.Column()
3 for p in ex.param:
4     col.append(pn.panel(ex.param[p]))
5 pn.Row(col, ex.viewable).servable()
```

```
initialized
Started downloading osm data
OSM data downloaded
```

[201]:

Rgb fn

RGB-PanSharpen-8bits_AOI_2_Vegas_img15:

Mask alpha: **0.30**

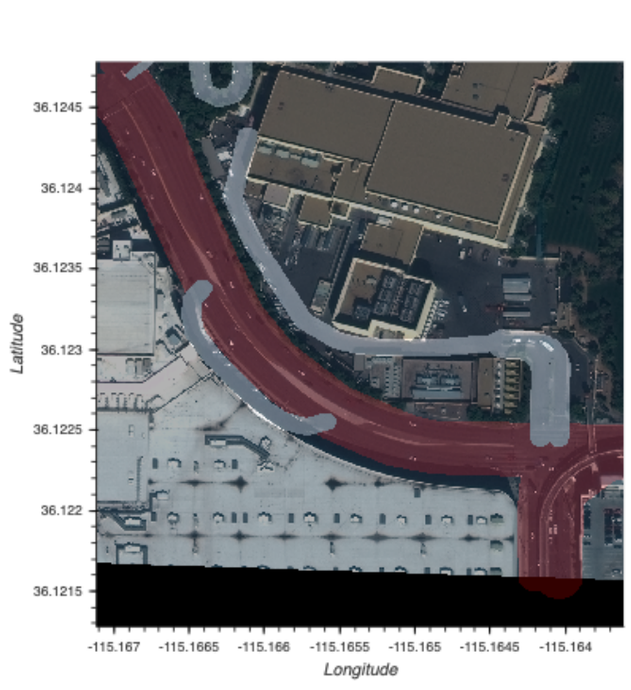
Show legend

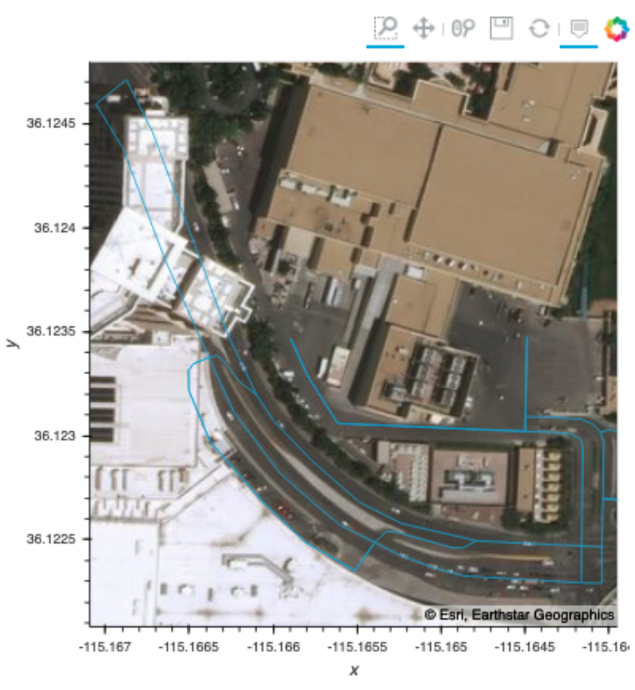
click to download osm

OSM log

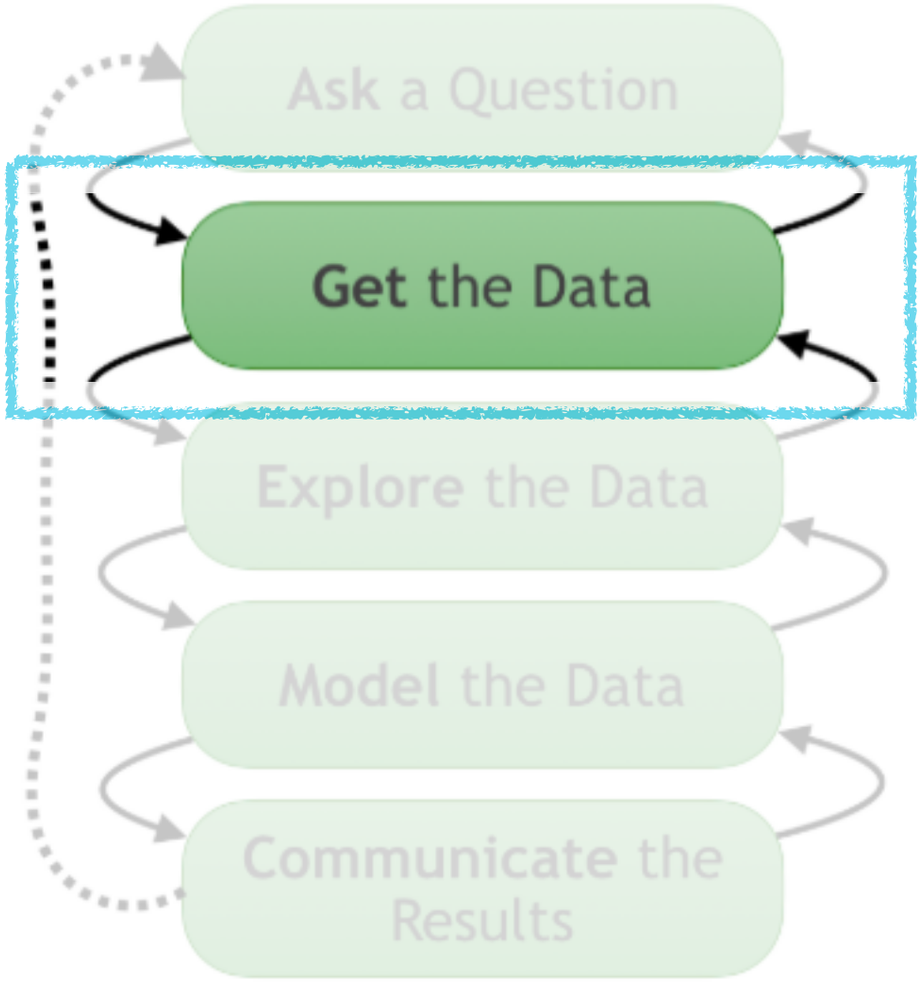
Osm dl count

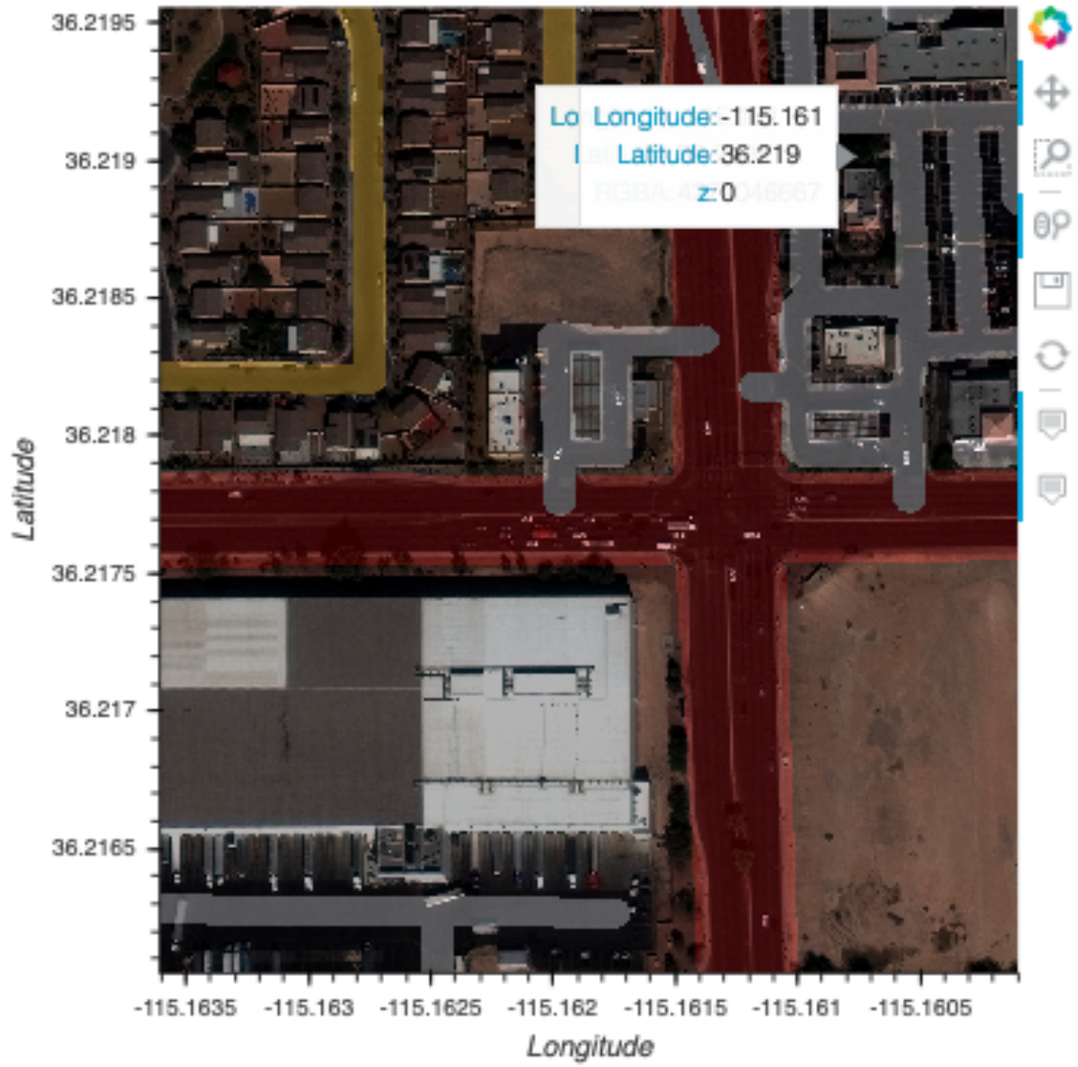
0



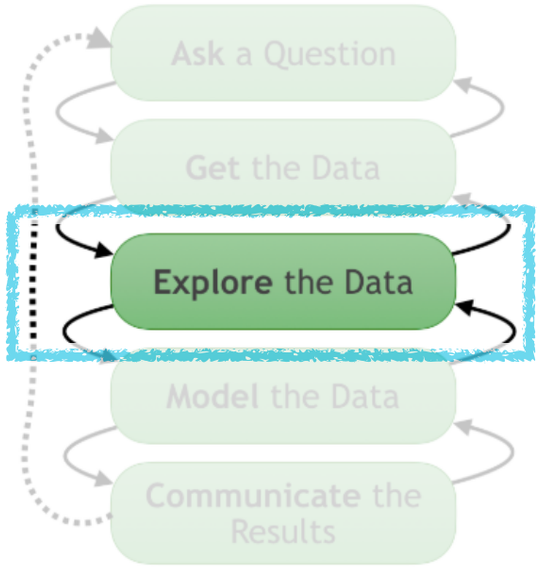


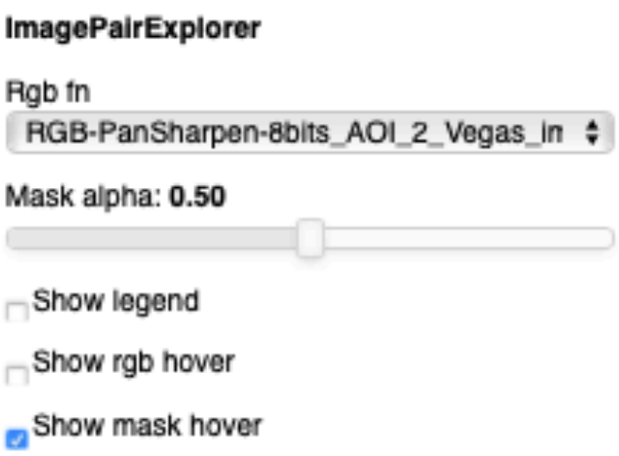
For this tutorial…

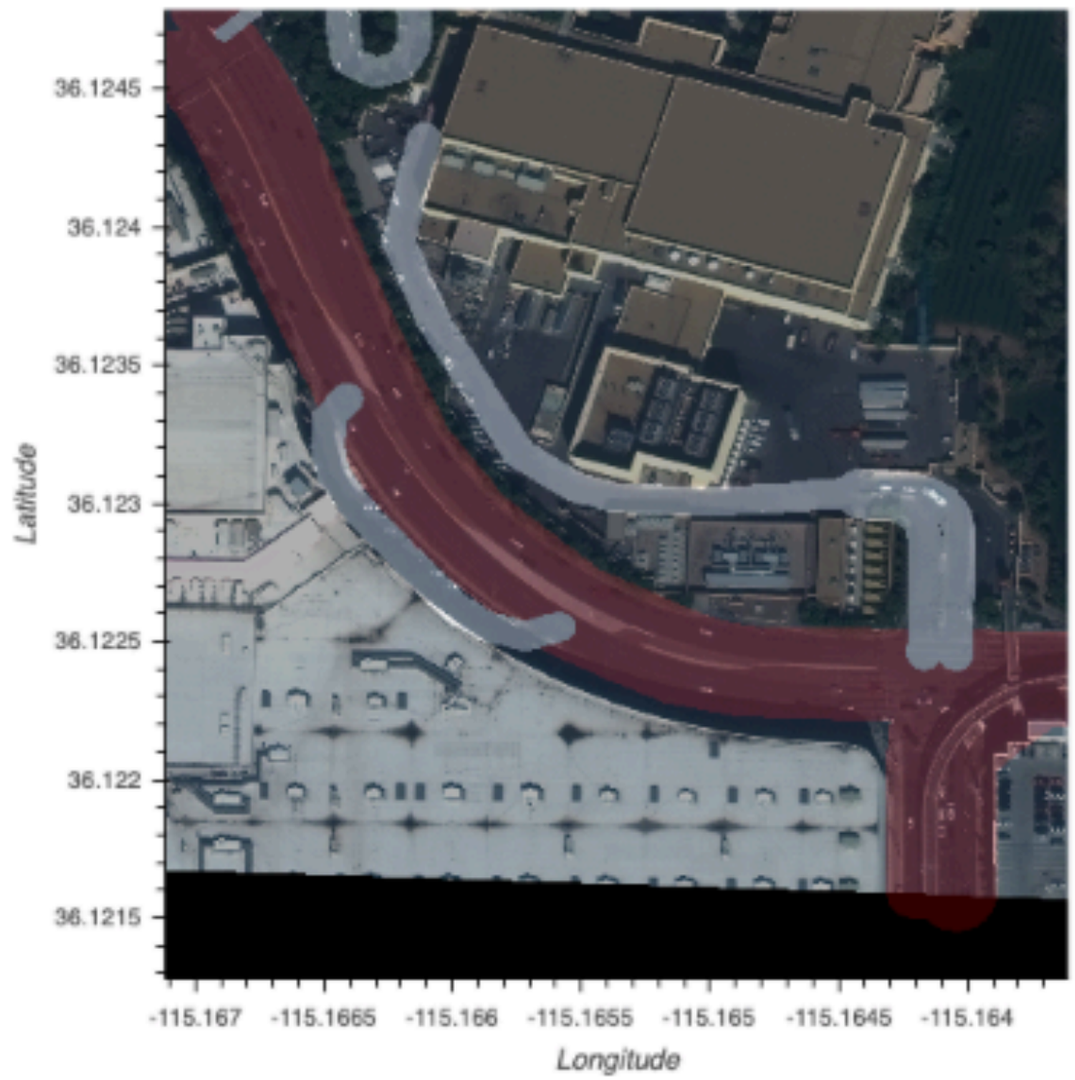




For this tutorial…

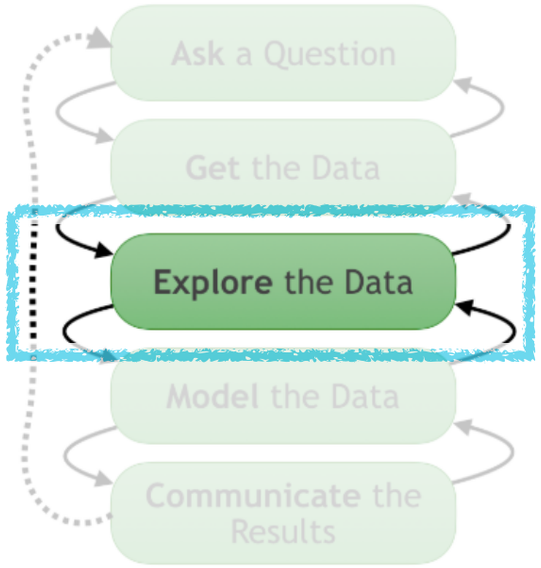


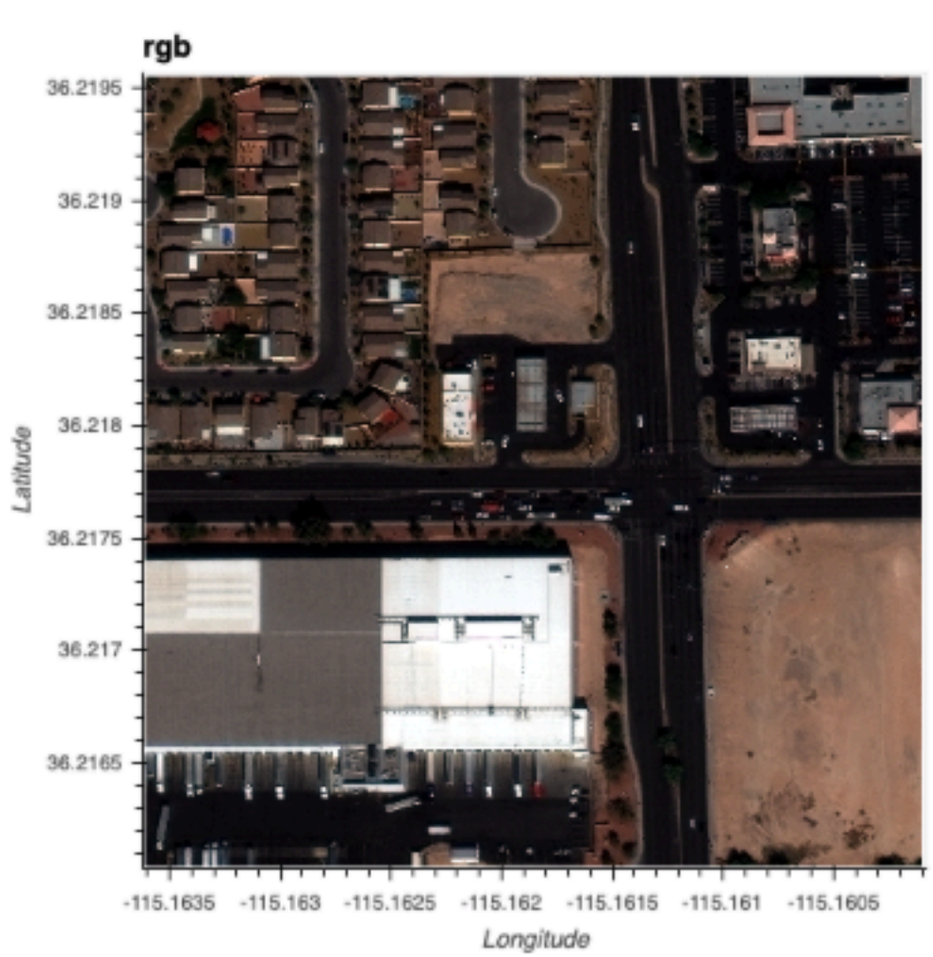


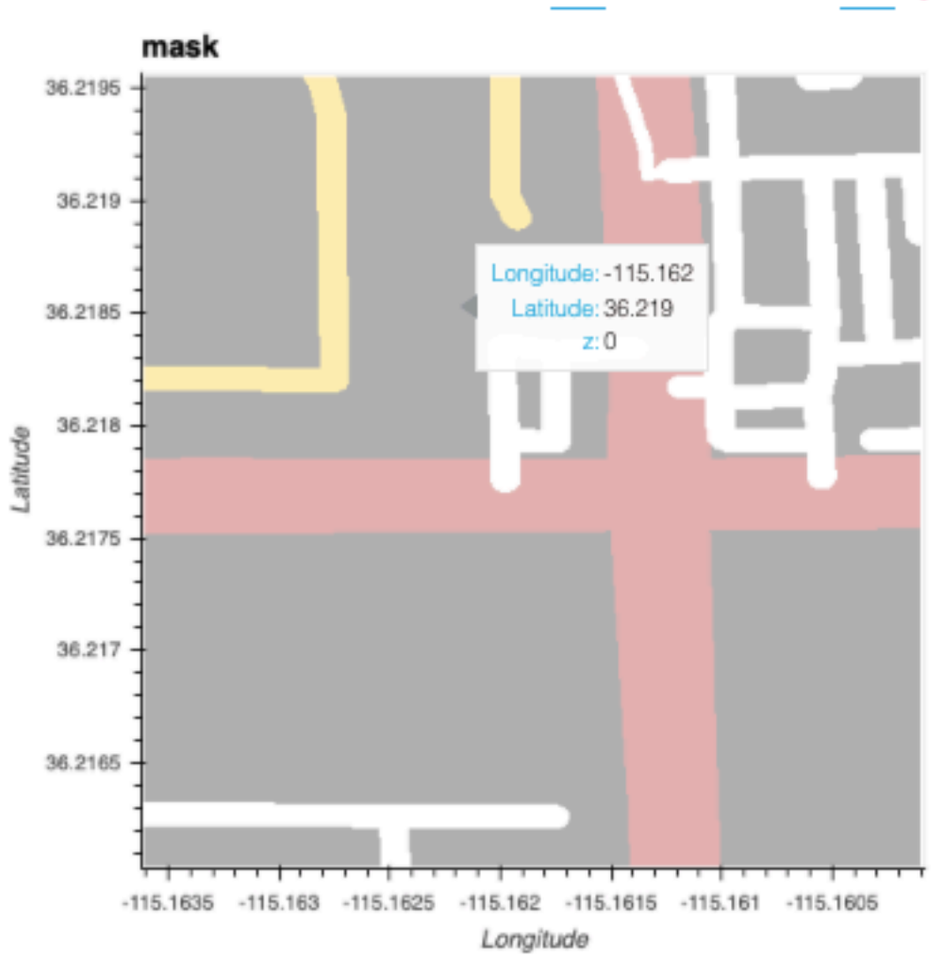


demo-full-workflow.ipynb

For this tutorial…

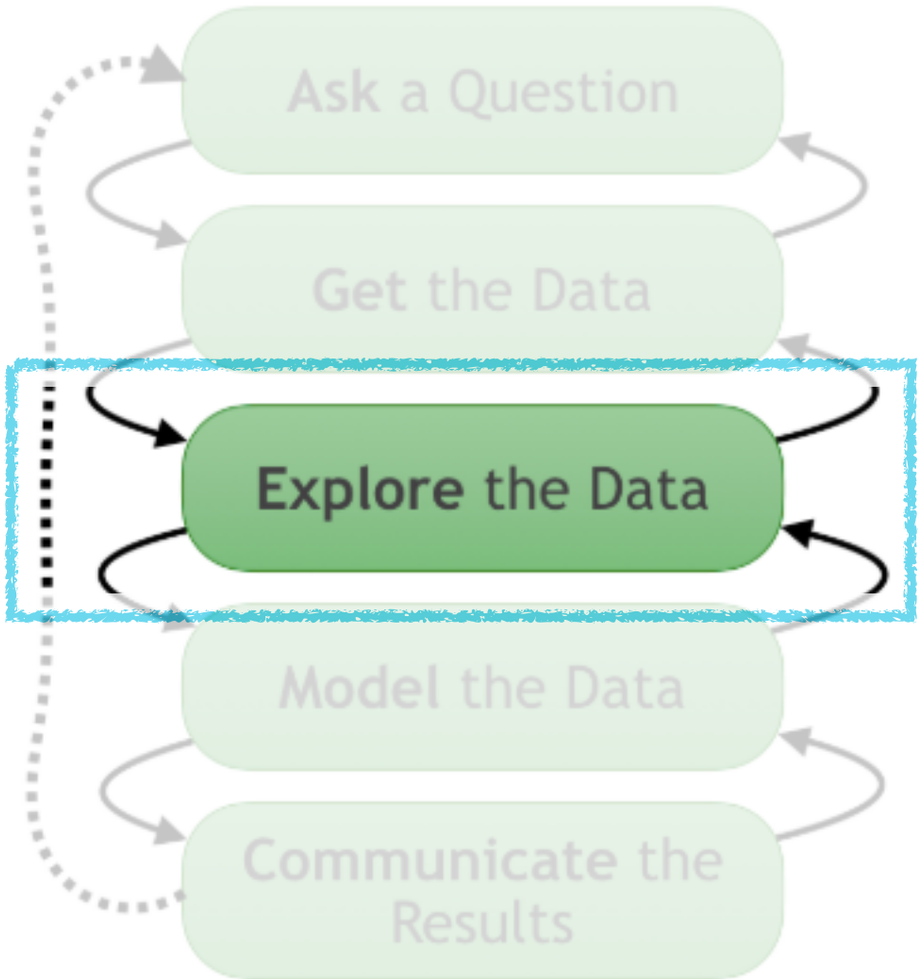


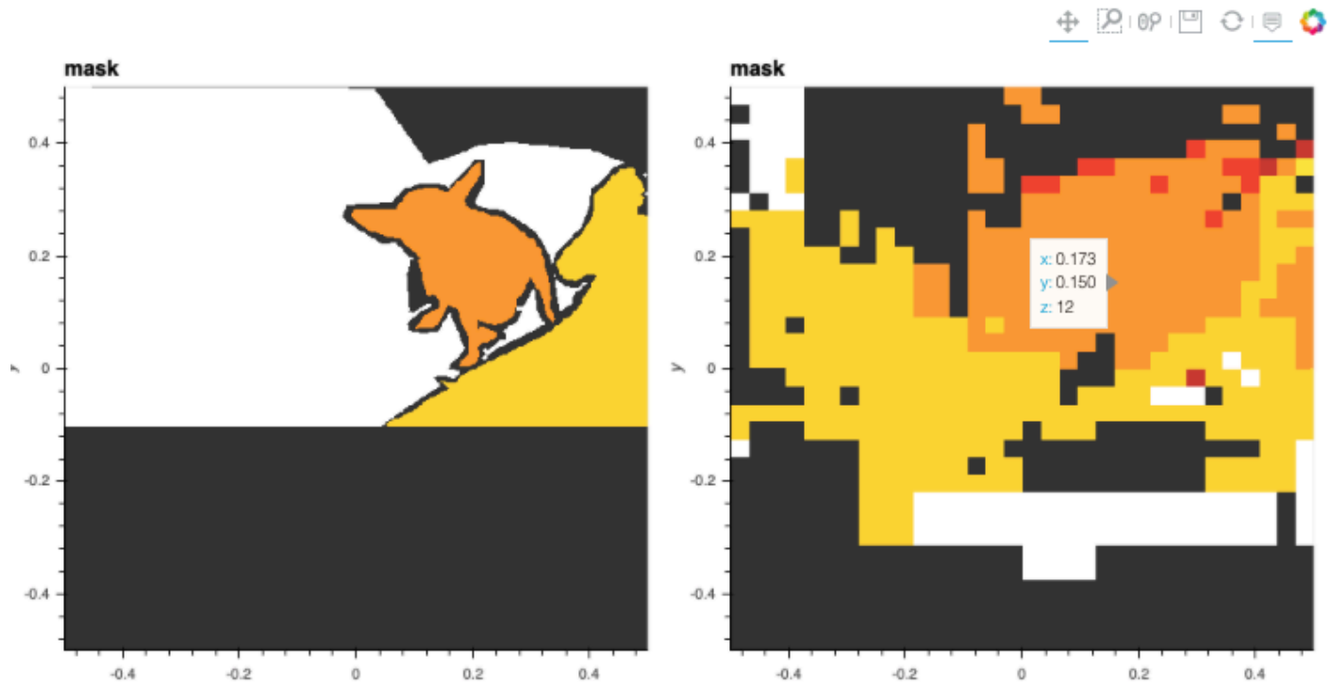




demo-full-workflow.ipynb

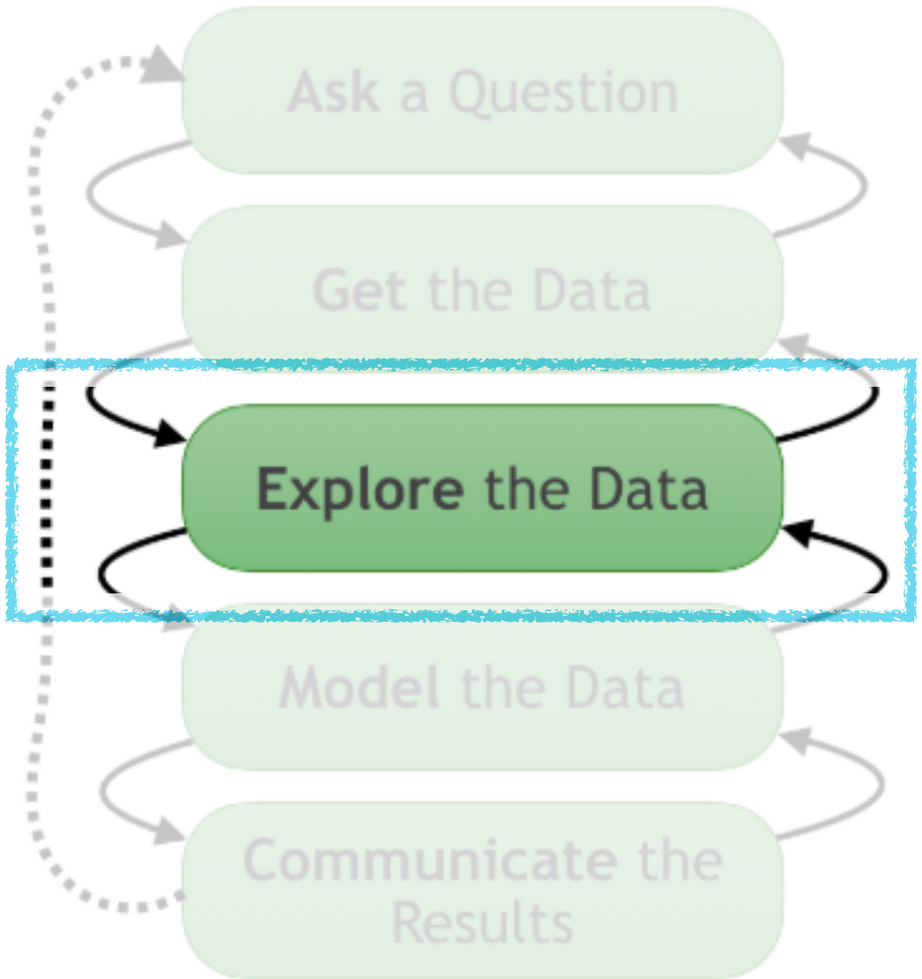
For this tutorial…





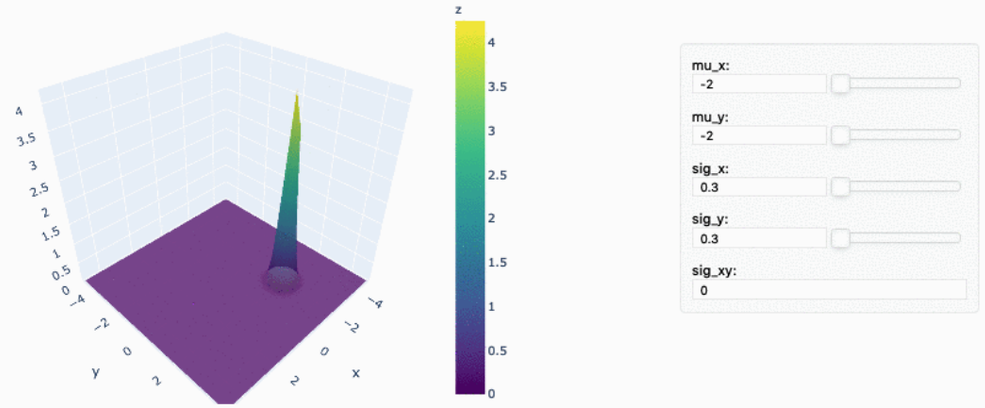
demo-full-workflow.ipynb

For this tutorial…



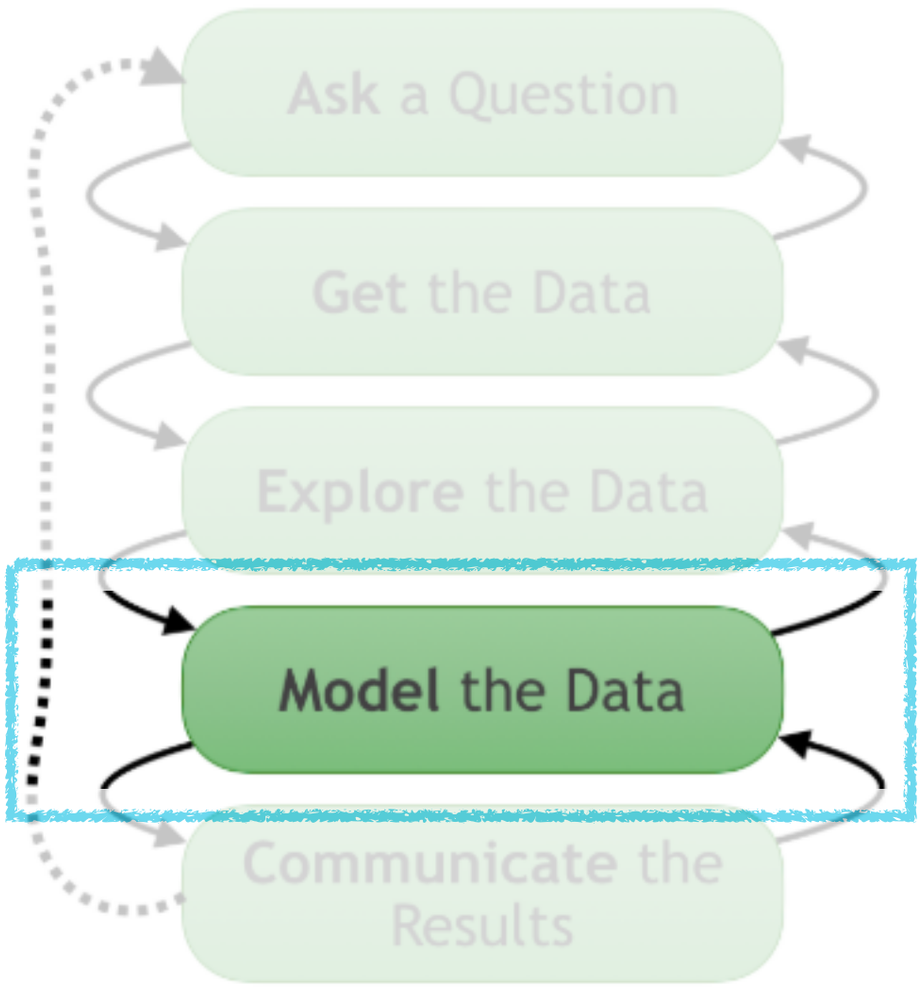

```
%%opts Surface [colorbar=True, width=600, height=600] (alpha=0.9)

dmap_3d.redim.values(mu_x=range(-2,2),
                     mu_y=range(-2,2),
                     sig_x=np.linspace(0.3, 3.0, 10),
                     sig_y=np.linspace(0.3, 3.0, 10),
                     sig_xy=[0.])
```

mu_x: -2, mu_y: -2 sig_x: 0.3, sig_y: 0.3 sig_xy: 0



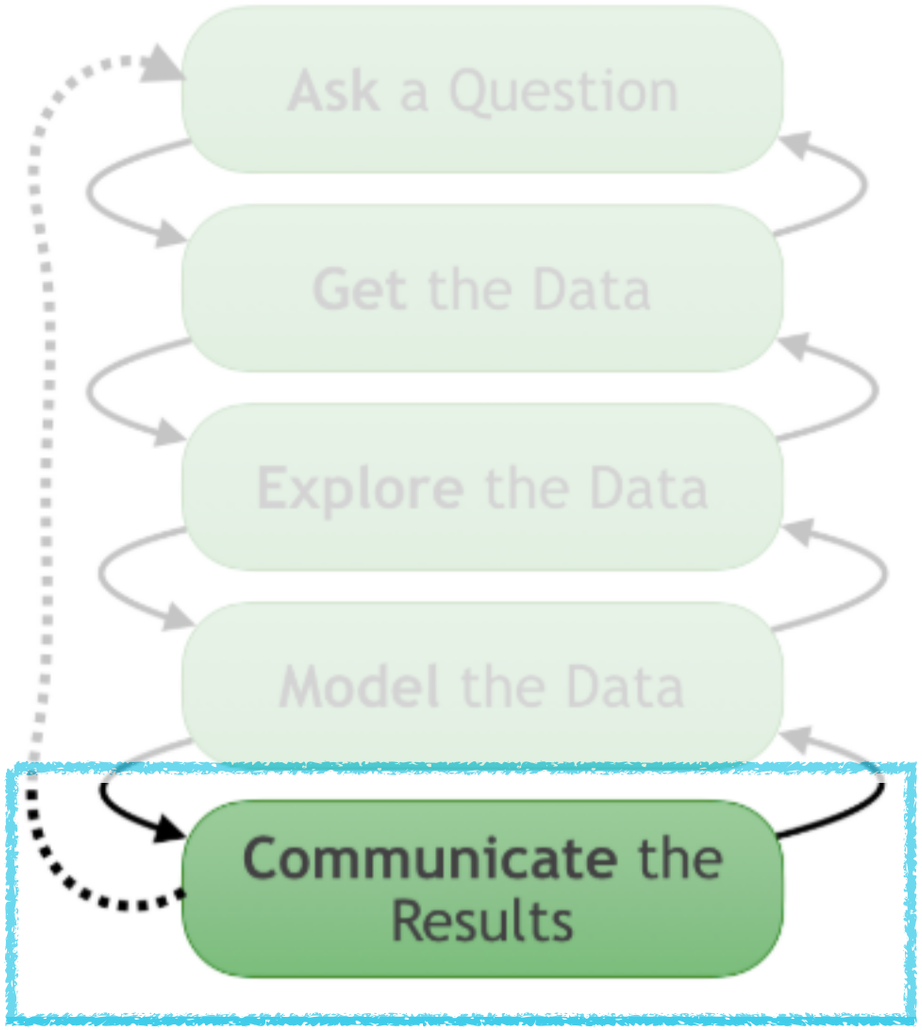
https://recordit.co/2D2kJ8OM6T

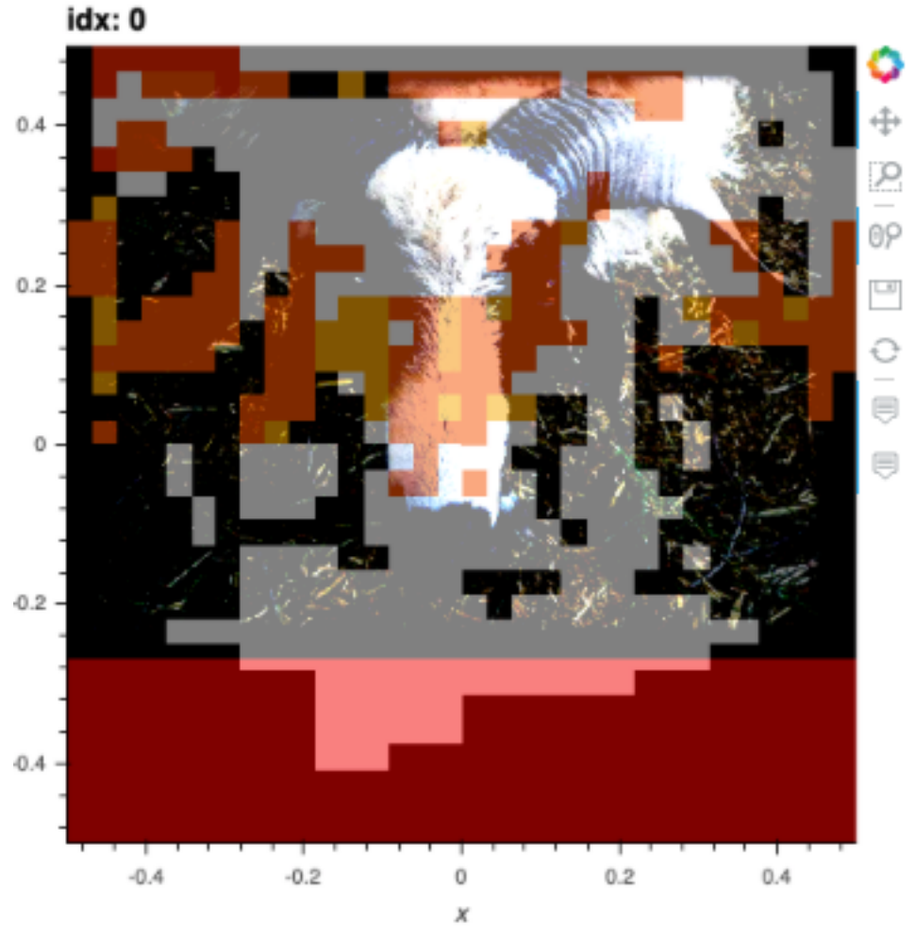
For this tutorial…



!

For this tutorial…





demo-full-workflow.ipynb

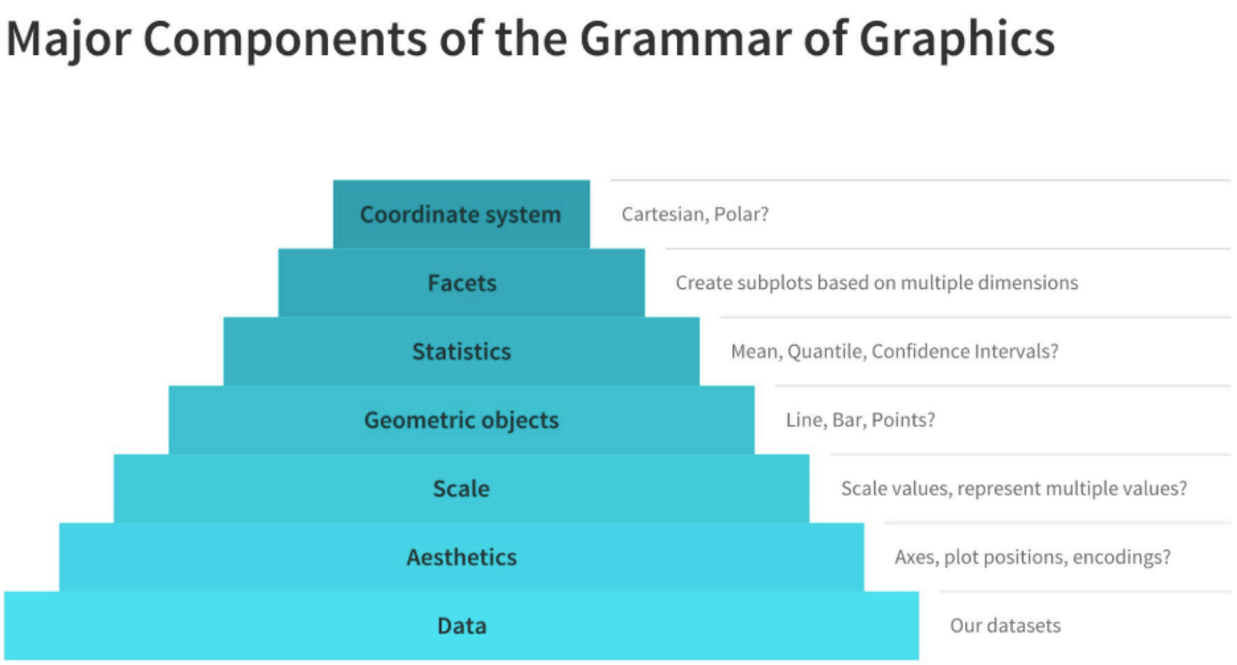
Step 0: Introduction to `HoloViews` libraries

- We will be touching only the tip of the iceberg
- Full tutorial link
- Documentation:
  - Holoviz, Param, Panel
- Userguide
- More examples
- Proceedings

HoloViews: Key Principles

- Declarative, not imperative (eg. matplotlib)
- Data-centered, not plot-centered (eg. gglot2)
- Annotate your data directly semantics
- Separate data/content vs. representation
- Explicitly express relationships between data elements via composition

HoloViews: Grammar of ~~Graphics~~ Data



jbednar on Nov 17, 2018 [-]

Personally, I don't _want_ a grammar of graphics; I want a grammar of data, where the data happens to have a graphical representation. I don't want to spend ages piecing together a fancy plot; I want to spend just a little time annotating my data to declare what it means, and then no matter how I slice and dice my data it will show up in a meaningful way. That way I can explore it to really understand it, which is the point of HoloViews (http://holoviews.org). But people approach plotting in lots of different ways, and some people actually _do_ want to spend their time making plots, so they are welcome to their ggplot2!

▲ gaius on Nov 17, 2018 [-]

I don't _want_ a grammar of graphics; I want a grammar of data, where the data happens to have a graphical representation

The point of having a graphical representation is to communicate with other humans therefore some subjective judgement is required; there's no "one true plot" for each type of data.

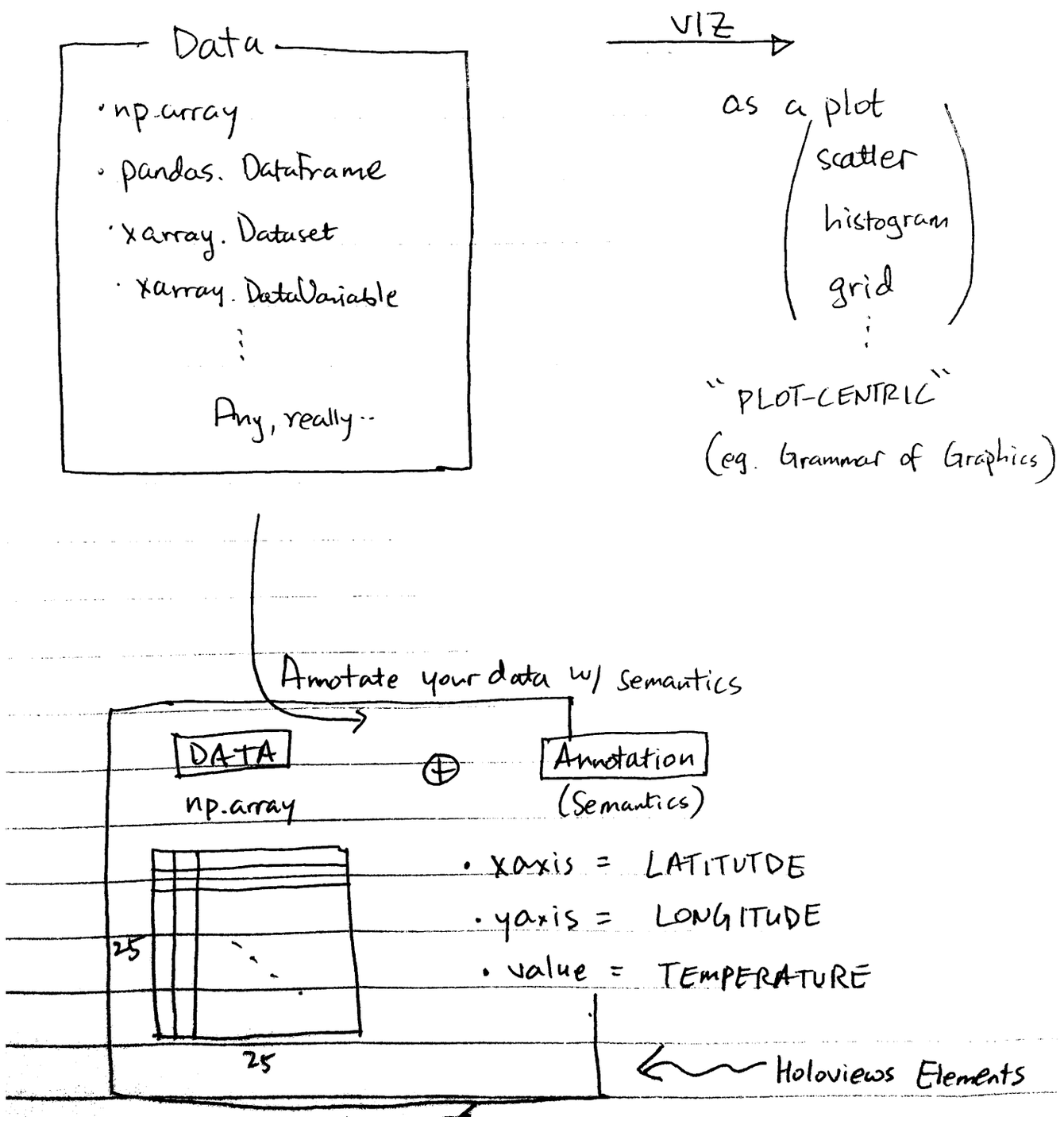
▲ jbednar on Nov 17, 2018 [-]

Of course! HoloViews does allow infinite customizability, to pull out more and more subtle features, show things more clearly, and just to make it look nice or to match your favorite style. But unlike ggplot2, HoloViews does that in a way that can apply to the _data_, rather than having to recapitulate the process every single time you build an individual plot. That way you and your colleagues can together build up whatever style you find most effective, then keep working with it across the full multidimensional landscape of data that you work with in a particular field. HoloViews is a completely different approach, if you really let the ideas sink in (e.g. from our paper about it at http://conference.scipy.org/proceedings/scipy2015/pdfs/jean-...), and is in no way a second-class citizen compared to ggplot2 or any other approach in R...

HoloViz Goals:

- Full functionality in browsers (not desktop)
- Full interactivity (inside and out of plots)
- Focus on Python users, not web programmers
- Start with data, not coding
- Work with data of any size
- Exploit general-purpose SciPy/PyData tools
- Focus on 2D primarily, with some 3D
- Avoid entangling your data, code, and viz:
  - Same viz/analysis code in Jupyter, Python, HPC, ...
  - Widgets/apps in Jupyter, standalone servers, web pages
  - Jupyter as a tool, not part of the results

HoloViews Model



HoloViews Basics

Notebook: 01a-hv-basics.ipynb

- hv.Elements
- hv.Containers
- geoviews