PyViz Formation

FinTech

Lesson 6.1


PyViz

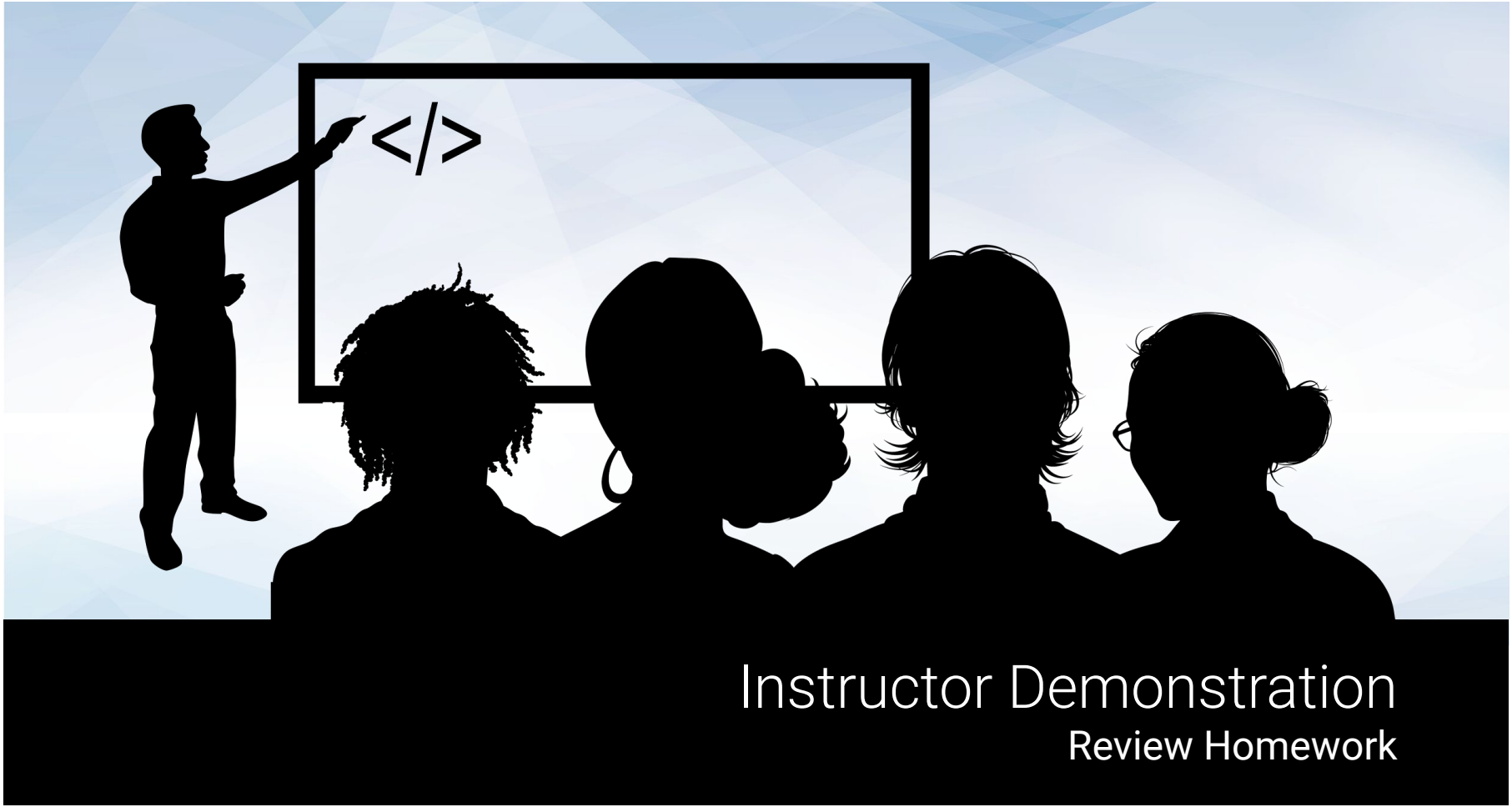
Class Objectives

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

- Comprehend the why, what, and how of Data Visualization.
- Explain the use cases for the different visualization libraries.
- Describe PyViz origin story.
- Set up PyViz ecosystem.
- Create interactive charts using hvPlot.
- Master hvPlot widgets for data exploration.
- Compose and overlay visualizations using hvPlot.
- Customize and Interpret data visualizations.

PyViz aims to provide a single stop-and-shop space for all data visualization needs.

Source: https://miro.medium.com/max/3120/1*6XAf0oi88MRCBbCd_cHcew.png



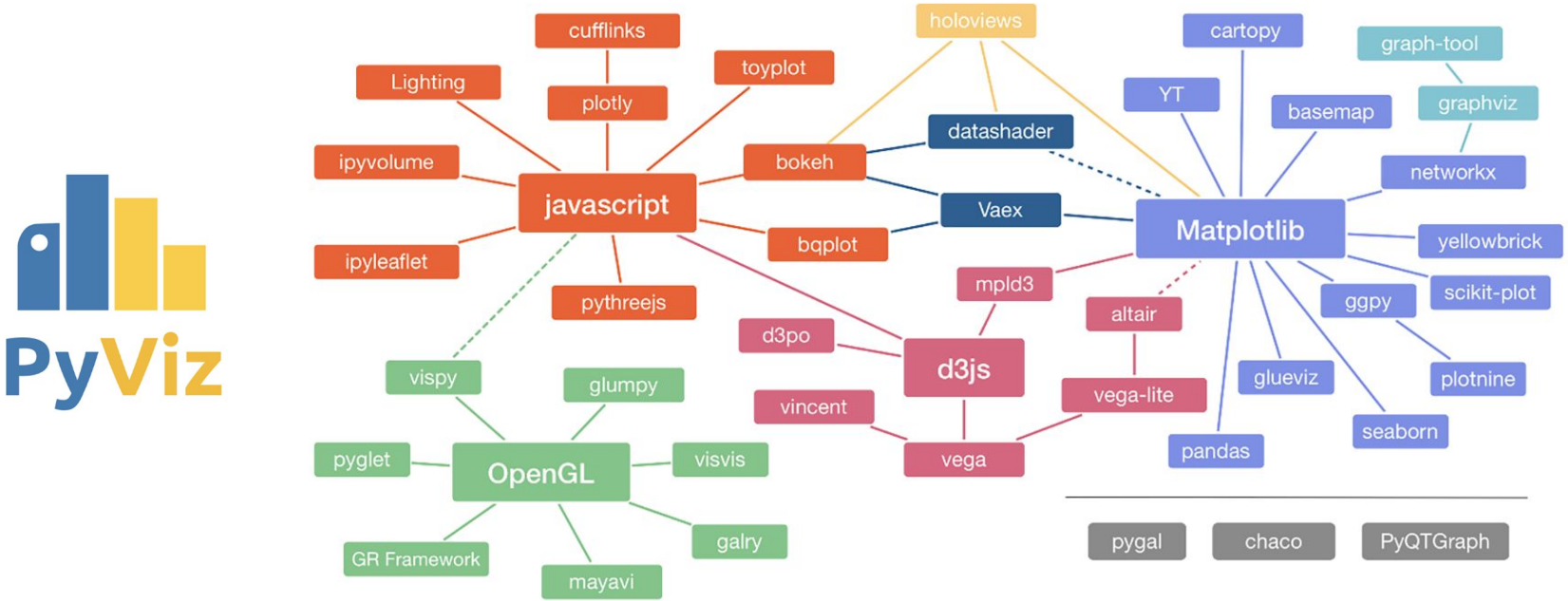
Instructor Demonstration

Review Homework

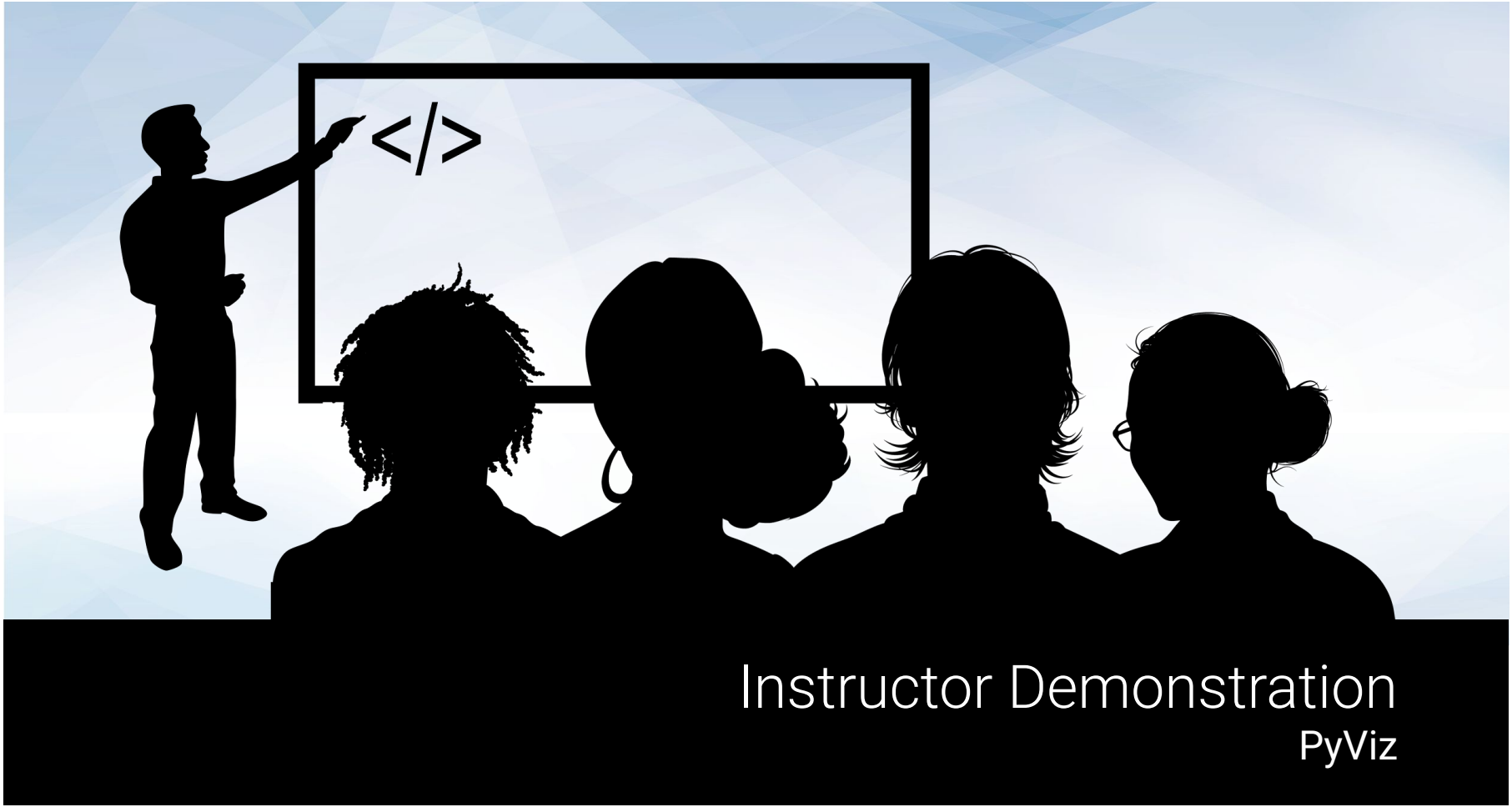
PyViz is a data visualization ecosystem that gives developers an easy way to access multiple data visualization libraries at one time.

PyViz

Each visualization technology in the PyViz ecosystem has the power and features to provide stand-alone visualizations. Each technology also has its strengths and weaknesses, which will be explored later.

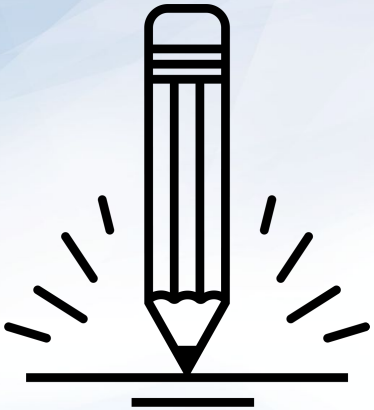


Source: http://blog.pyviz.org/images/pyviz_graph.png



Instructor Demonstration

PyViz



System Check

In this activity, you will install PyViz.

(Instructions sent via Slack.)

Suggested Time:
5 Minutes

hvPlot

hvPlot

hvPlot is a technology that brings plots to life. hvPlot abstracts over Python visualization libraries like Matplotlib, Pandas, and Streamz. The abstraction allows hvPlot to utilize the stand-alone plotting mechanisms of these technologies.

Matplotlib | Pandas | Streamz

hvPlot

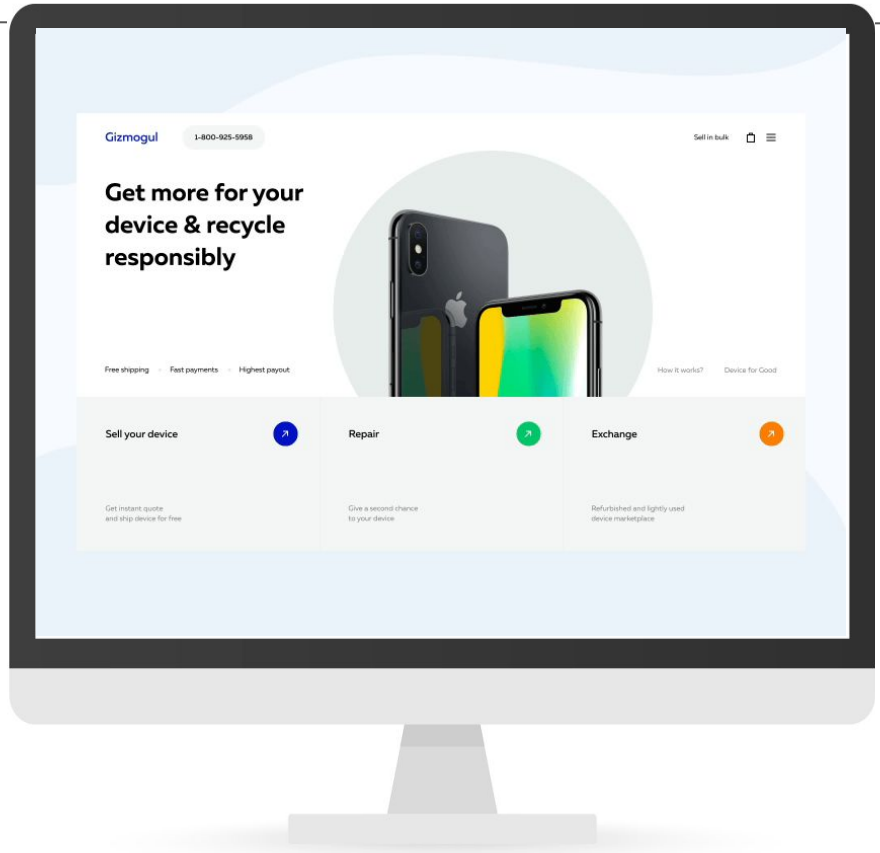
This abstraction also allows hvPlot to transform the static plots (i.e. Matplotlib plots) into interactive sandboxes for data exploration and analysis. hvPlots allow for:

- Panning
- Zooming

hvPlot

This abstraction also allows hvPlot to transform the static plots (i.e. Matplotlib plots) into interactive sandboxes for data exploration and analysis. hvPlots allow for:

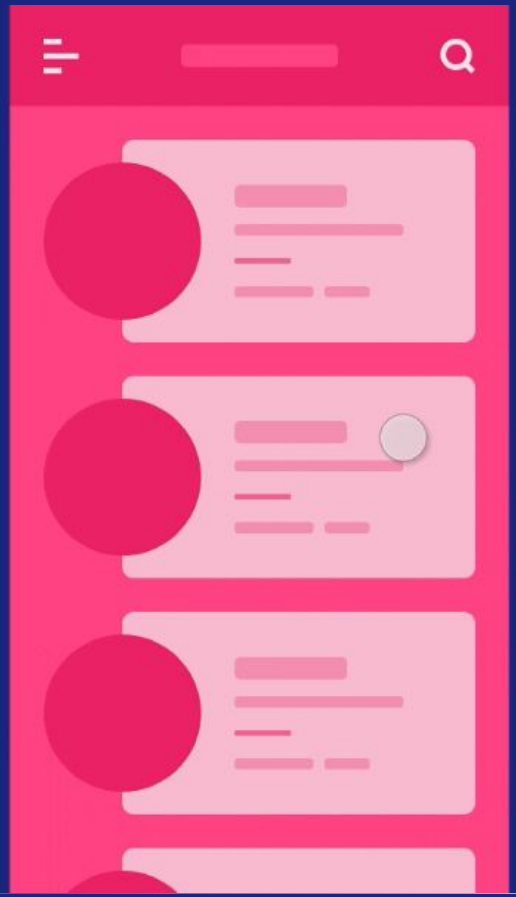
- Hovering

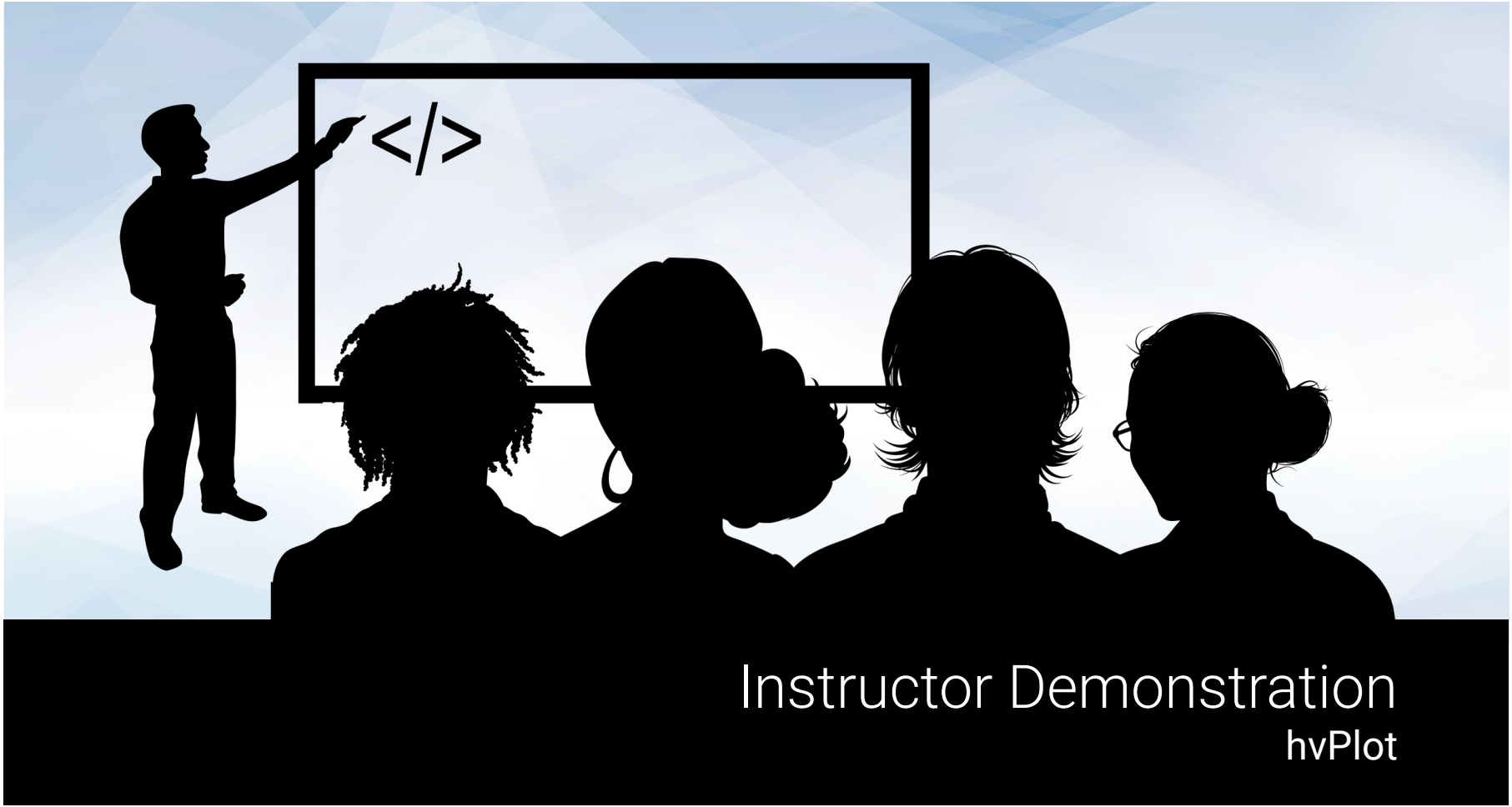


hvPlot

This abstraction also allows hvPlot to transform the static plots (i.e. Matplotlib plots) into interactive sandboxes for data exploration and analysis. hvPlots allow for:

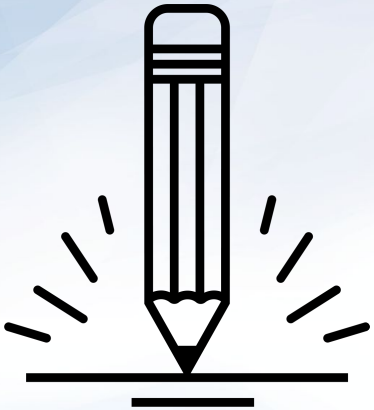
- Clicking
- Filtering





Instructor Demonstration

hvPlot



Activity:

Plotting a Visual Takeover

In this activity, you will re-visit plots you made earlier in the class using Matplotlib and re-create them as hvplots.

(Instructions sent via Slack.)

Suggested Time:
15 Minutes

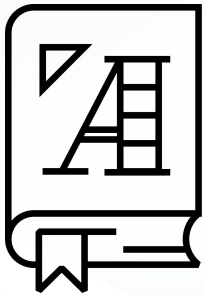
Time’s Up! Let’s Review.

Countdown timer

15:00

(with alarm)

Interactive Visualizations



Interactive visualizations are charts and graphs that can be manipulated by user interaction. Example interactions include clicking, panning, and zooming, all of which come out of the box with hvPlot.

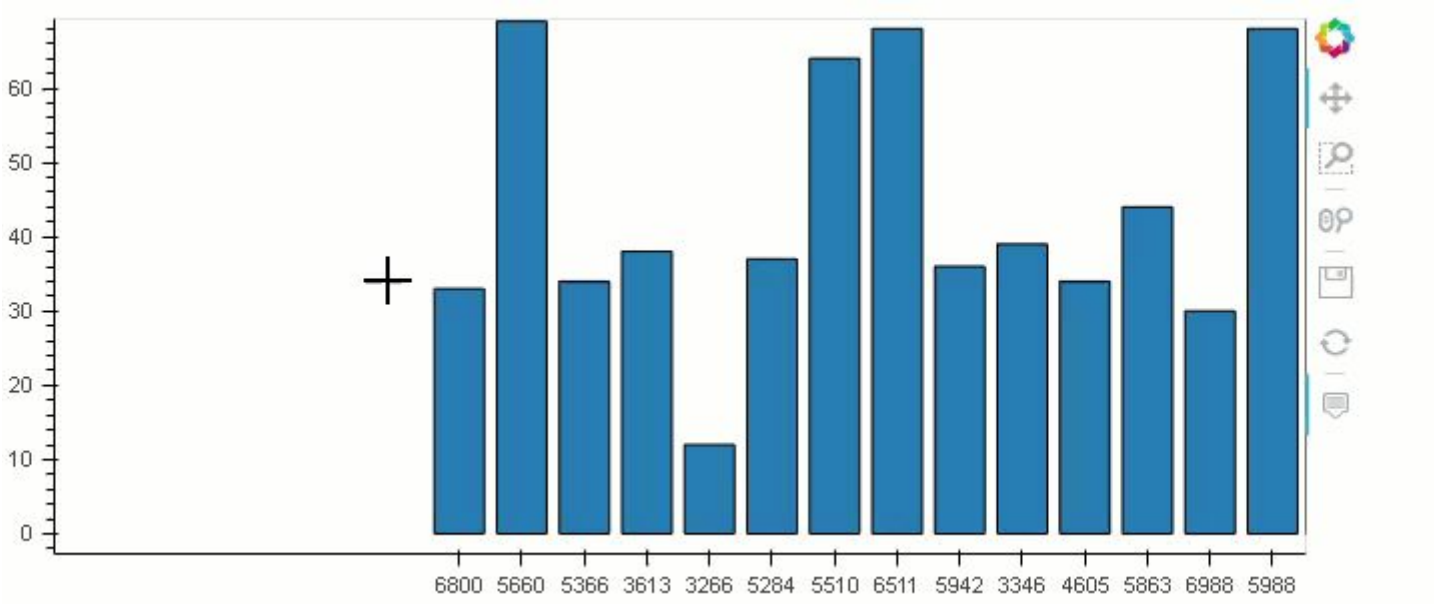
Interactive Visualizations

The widget bar is used to set the mode of interaction. The available modes are:

- Pan 
- Box zoom 
- Save 
- Reset 
- Hover 

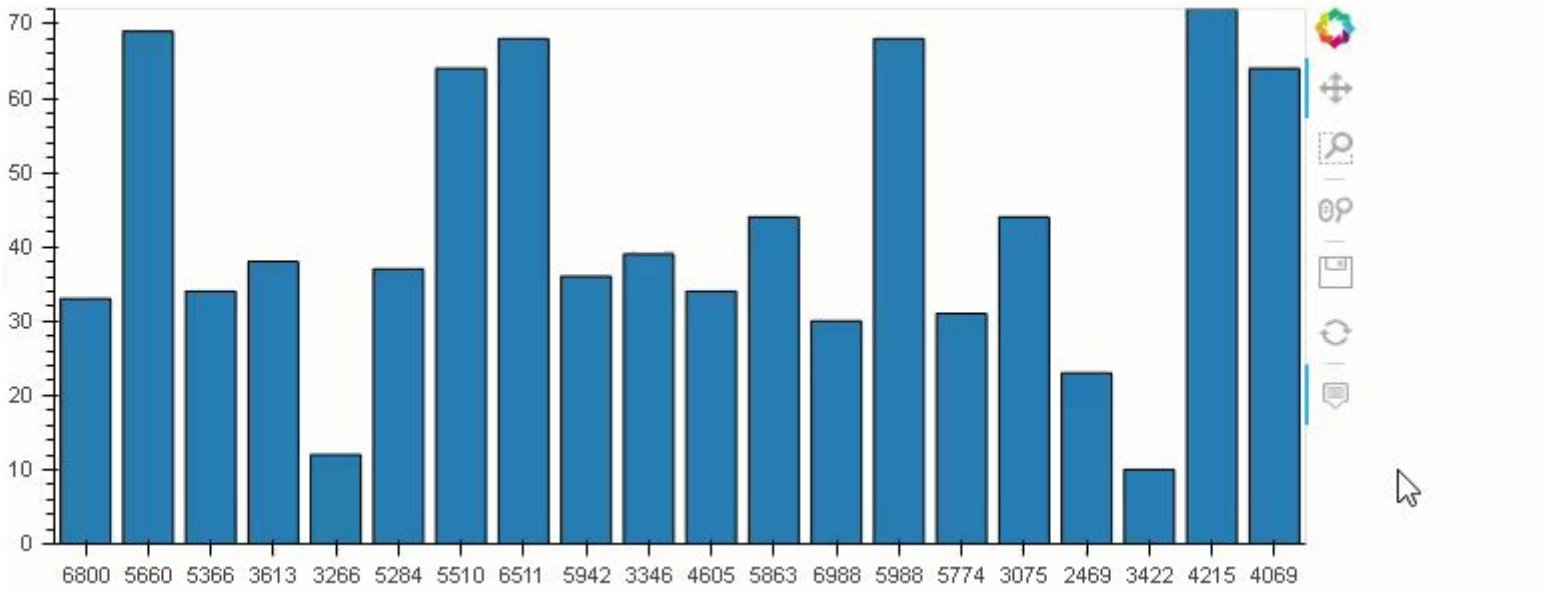
Panning

Panning allows users to move the data on the screen in all directions. This is particularly valuable when trying to view data across time, outlying data points, or even high volume. Instead of having to zoom out to see all data, panning brings data into the forefront of the visualization as needed.



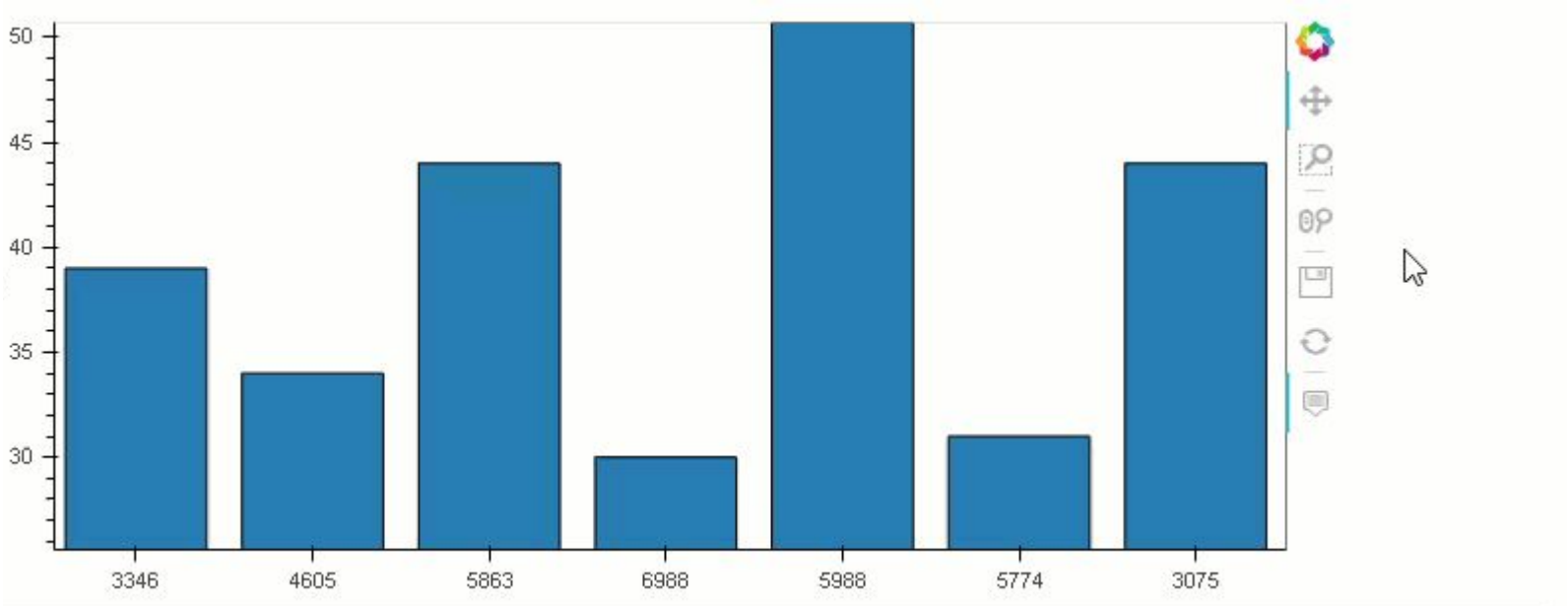
Box Zoom

The box zoom interaction zooms into data points based off of a selection drawn.



Wheel Zoom

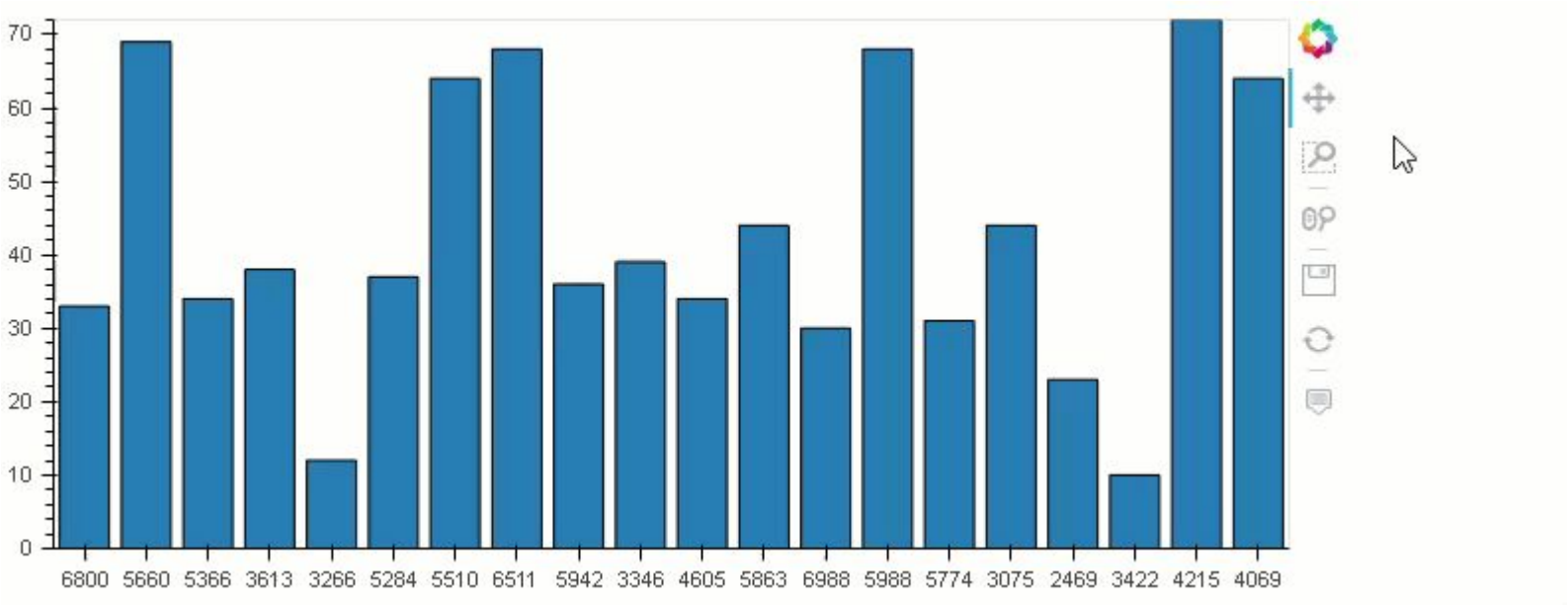
Wheel zoom works similar to box zoom; however, the click wheel is used magnify data points.



Another way to interact with visualizations is to save them. hvPlot allows visualizations to be saved as html documents for later use.

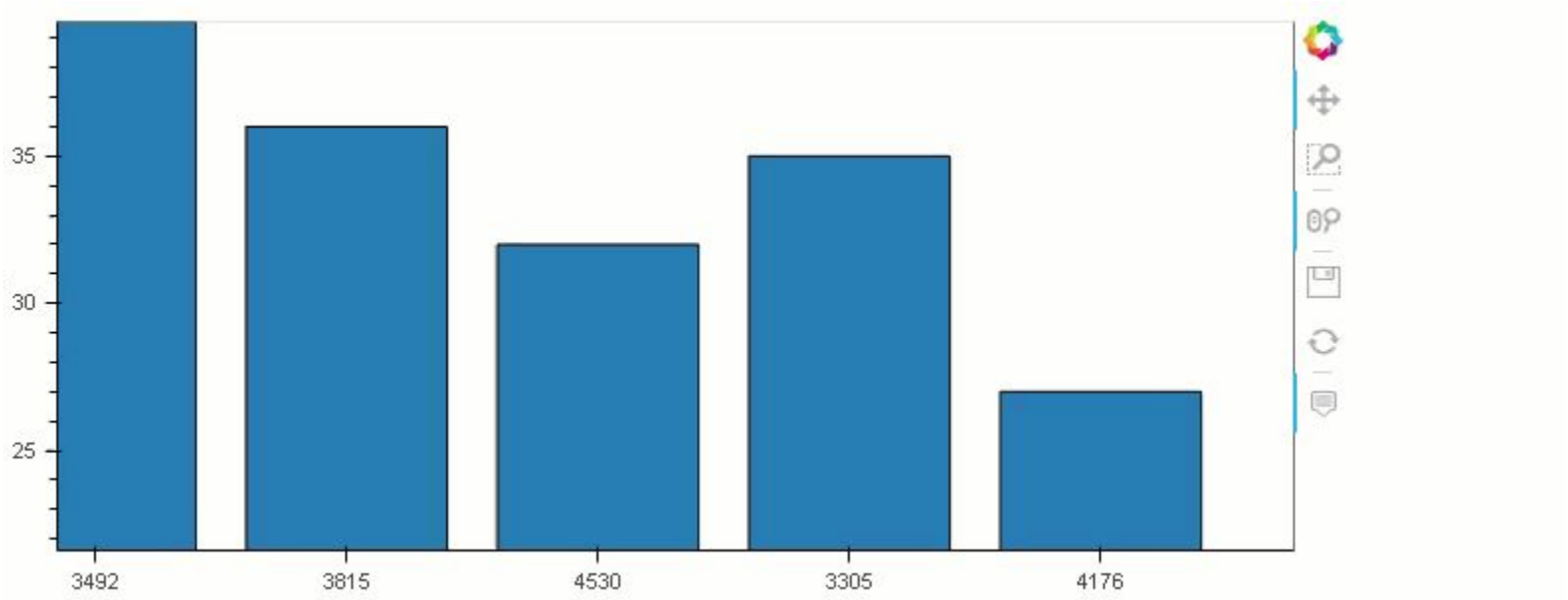
Hovering

Hovering is also a key interaction. Hovering over a value in a visualization shows the actual value for that data point.



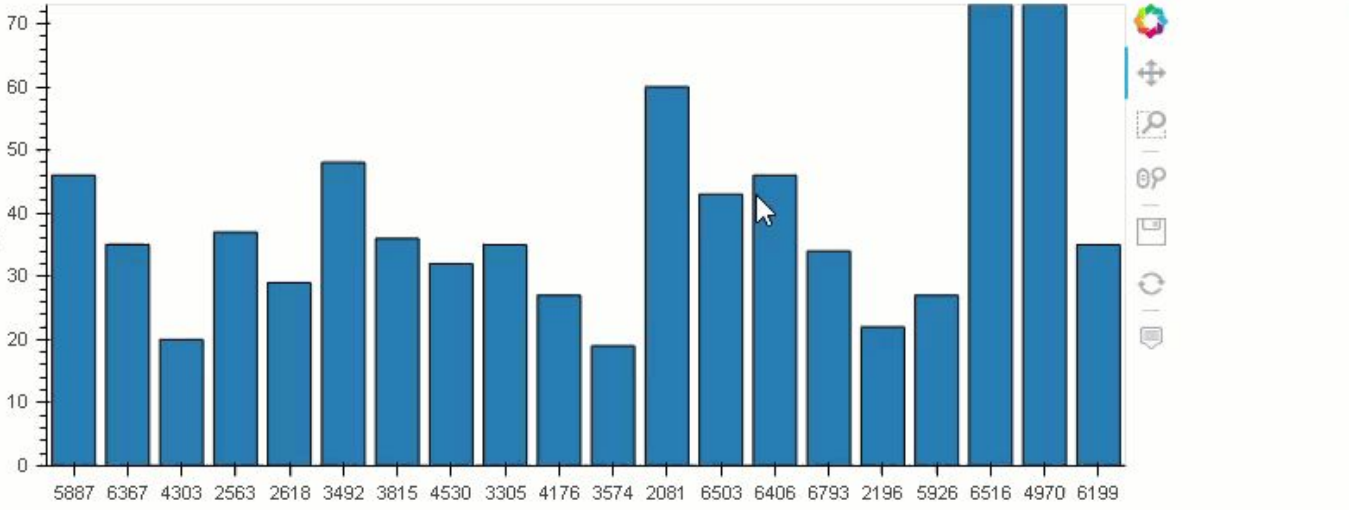
Reset

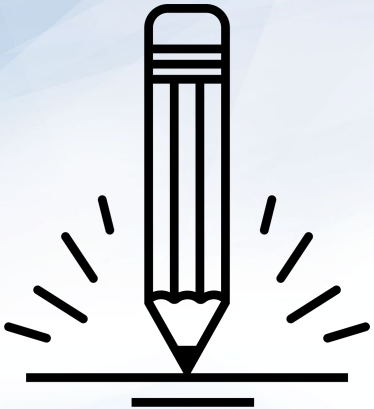
hvPlot also includes a reset widget button, which resets all visualization interactions. If the visualization was previously zoomed in at 110%, the reset will bring the zoom percentage back to 100%.



Combinations

Widget activities can be combined. Clicking pan, wheel, and hover buttons enables users to pan, zoom, and get details by hovering all at the same time. However, certain widgets cannot be used with other widgets. For example, users cannot pan and box zoom at the same time. One action has to occur first, and then the second widget option can be chosen.





Activity: hvPlot Widgets

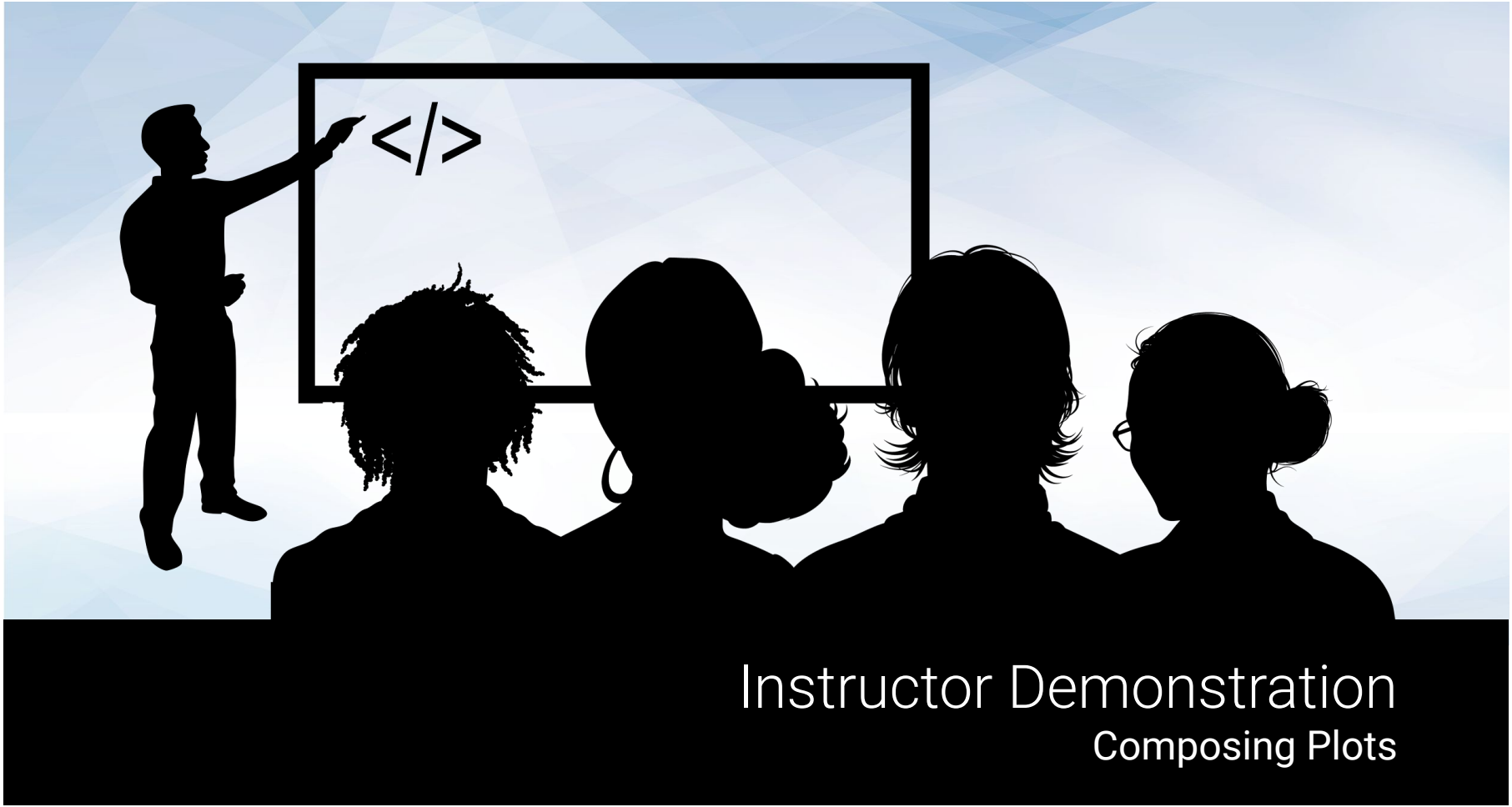
In this activity, you will play around with the hvPlot widgets to get more accustomed to the different types of interactions supported with hvPlots.

(Instructions sent via Slack.)

Suggested Time:
15 Minutes

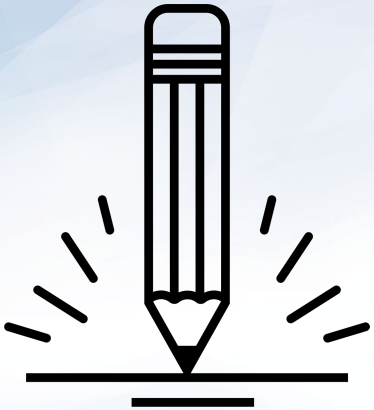
Time’s Up! Let’s Review.

Composing Plots



Instructor Demonstration

Composing Plots



Activity:

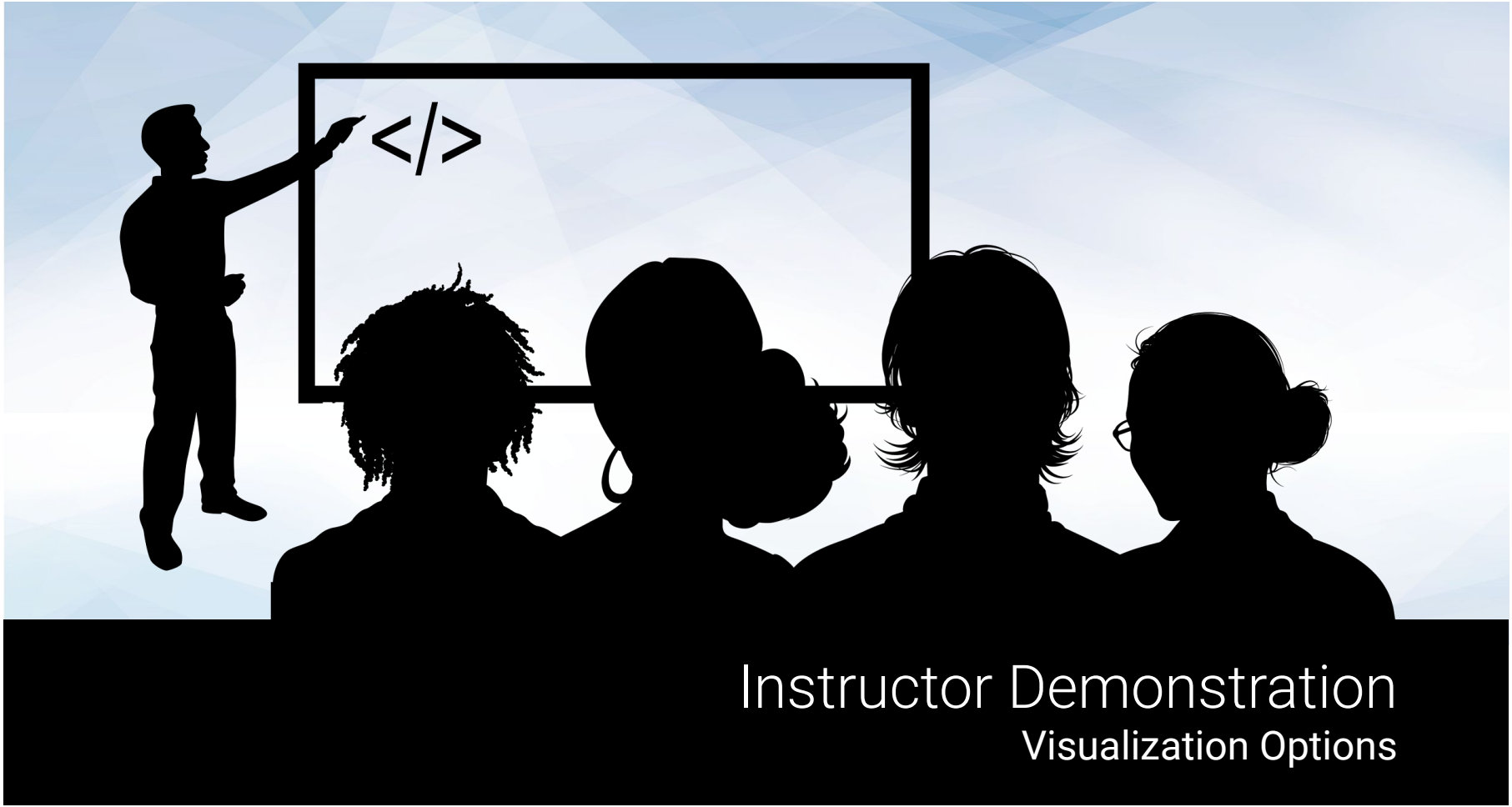
Composing Masterpieces

In this activity, you will use the information learned in your instructor's demo to customize your hvPlots.

(Instructions sent via Slack.)

Suggested Time:
10 Minutes

Time’s Up! Let’s Review.



Instructor Demonstration

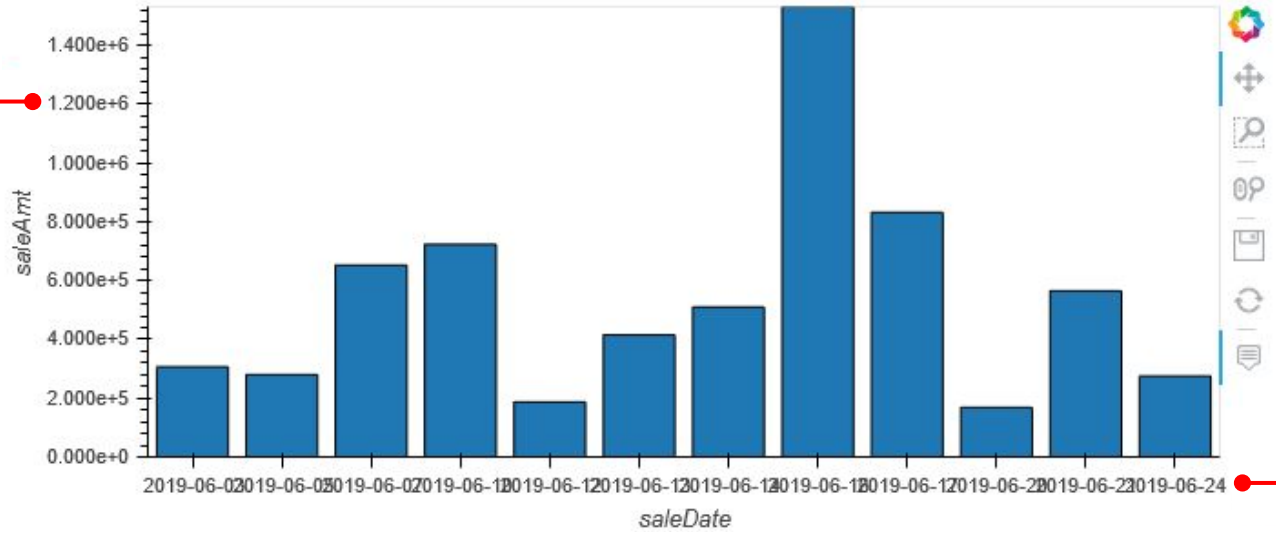
Visualization Options

Visualization Options

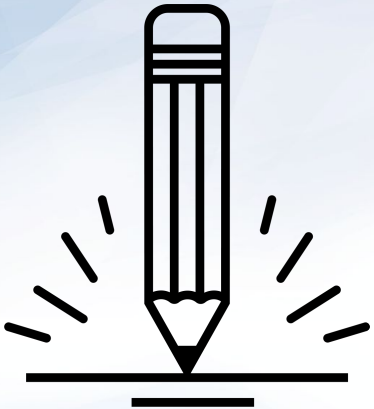
hvPlots do not always come out perfect. Customization options give users the ability to customize the look and feel of their visualizations.

```
# Plot data without rotation
sale_prices_by_year.hvplot.bar(x='saleDate',y='saleAmt')
```

ylabels are not formatted properly



xlabels are hard to read



Activity: Picture Perfect

By the end of this activity, you will have employed hvPlot customization attributes and options to perfect and add finishing touches to your visualizations.

(Instructions sent via Slack.)

Suggested Time:
15 Minutes

Time’s Up! Let’s Review.

Recap & Questions