

PyViz Installation Guide

PyViz is a Python visualization package that provides a single platform to access multiple visualization packages, including Matplotlib, Plotly Express, hvPlot, Panel, D3.js, etc.

Follow the steps below to install and set up PyViz in your Python environment. These steps should be completed outside of class.

Video Guide for installing PyViz: [PyViz Installation Video](#)

NOTE: Make sure that you are using your conda environment that has anaconda installed. Create a new environment for this lesson using:

```
conda update anaconda
conda create -n pyvizenv python=3.7 anaconda -y
conda activate pyvizenv
```

Before installing the PyViz dependencies, you need to install a couple of libraries. First, install the `python-dotenv` library using `pip` to work with environment variables.

```
pip install python-dotenv
```

Next, install the `nb_conda` package that will allow you to switch between virtual environments in Jupyter lab.

```
conda install -c anaconda nb_conda -y
```

Follow the next steps to install PyViz and all its dependencies in your Python virtual environment.

1. Download the PyViz dependencies **nodejs** and **npm** (included in nodejs).

```
conda install -c conda-forge nodejs=12 -y
```

2. Use the `conda install` command to install the following packages. Note: On some of these installs, you may get a message that says that the requested packages are already installed. That is fine. Conda is really good at installing all of the required dependencies between these tools.

```
conda install -c pyviz holoviz -y
conda install -c plotly plotly -y
conda install -c conda-forge jupyterlab=2.2 -y
```

3. Use `pip` to install the correct versions of `matplotlib` and `numpy` using the following commands:

```
pip install numpy==1.19
pip install matplotlib==3.0.3
```

4. PyViz installation also requires the installation of Jupyter Lab extensions. These extensions are used to render PyViz plots in Jupyter Lab. Execute

the below commands to install the necessary Jupyter Lab extensions for PyViz and Plotly Express.

- **IMPORTANT:** *In some installation cases you may encounter the following warning. If you do, **continue with the installations as indicated**, as this warning will **not** affect your code.*

Config option `kernel_spec_manager_class` not recognized by `EnableNBExtensionApp`

- Set NODE_OPTIONS to prevent “JavaScript heap out of memory” errors during extension installation:

```
# (OS X/Linux)
export NODE_OPTIONS=--max-old-space-size=4096
```

```
# (Windows)
set NODE_OPTIONS=--max-old-space-size=4096
```

- Install the Jupyter Lab extensions:

```
jupyter labextension install @jupyter-widgets/jupyterlab-manager --no-build
```

```
jupyter labextension install jupyterlab-plotly --no-build
```

```
jupyter labextension install plotlywidget --no-build
```

```
jupyter labextension install @pyviz/jupyterlab_pyviz --no-build
```

- Build the extensions (This may take a few minutes):

```
jupyter lab build
```

- Using the build and --no-build flags allows the machine to build all four extensions simultaneously, otherwise you will have to wait several minutes in between in each installation.

- After the build, unset the node options that you used above:

```
# Unset NODE_OPTIONS environment variable
# (OS X/Linux)
unset NODE_OPTIONS
```

```
# (Windows)
set NODE_OPTIONS=
```

5. Run the following commands to confirm installation of all PyViz packages. Look for version numbers with at least the following versions.

```
conda list nodejs
conda list holoviz
conda list hvplot
conda list panel
```

```
conda list plotly

nodejs          12.0.0
holoviz         0.11.3
hvplot          0.7.1
panel           0.10.3
plotly          4.14.3
numpy           1.19
matplotlib      3.0.3
```

6. You will also need to register for a public mapbox API key, which can be obtained via [this sign-up link](#). Detailed information on how Mapbox can be used to generate plots can be found on Plotly's documentation page.

Now you're all set to get started creating visual masterpieces using PyViz technologies!

Troubleshooting

If you experience blank plots rendering in your Jupyter Lab preview, try the following steps:

1. First, make sure you are not importing `hvplot.pandas` before a `pn.extension()`. Loading `hvplot.pandas` first initializes a Holoviews extension and causes the Panel extension to fail.
2. Next, clear your browser cache.
 - If using Chrome, you can do this by right clicking and choosing **Inspect** from the drop menu.
 - Next, hold down click on the browser reload button which will cause another drop down menu to appear. From this menu select **Empty Cache and Hard Reload**.
3. Then clear the Kernel cache:
 - Click the **Kernel** drop down menu inside Jupyter Lab. From this menu, click **Restart Kernel and Clear Outputs**.
4. After these steps are completed, re-run your notebook.
5. If your browser is Chrome, and you continue to render blank plots after completing the previous 4 steps, try updating Chrome. Instructions for this can be found [here](#).

If you have issues with PyViz or Jupyter Lab, you may need to update your Conda environment. Follow the steps below to update the environment and then go back to the install guide to complete a fresh installation of PyViz.

1. Deactivate your current Conda environment. This is required in order to update the global Conda environment. Be sure to quit any running applications such as Jupyter prior to deactivating the environment.

```
conda deactivate
```

2. Update Conda.

```
conda update conda
```

3. Create a fresh Conda environment to use with PyViz.

```
conda create -n pyvizenv python=3.7 anaconda
```

4. Activate the new environment.

```
conda activate pyvizenv
```

5. Install the PyViz dependencies following the guide above.

© 2020 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.