

Nova OnDemand Tutorial: Launching and Using Jupyter for NLP Tasks

This tutorial includes:

- How to launch Jupyter through Nova OnDemand
- A step-by-step demo on using Jupyter to process a basic NLP-related CSV file

Launching Jupyter via Nova OnDemand

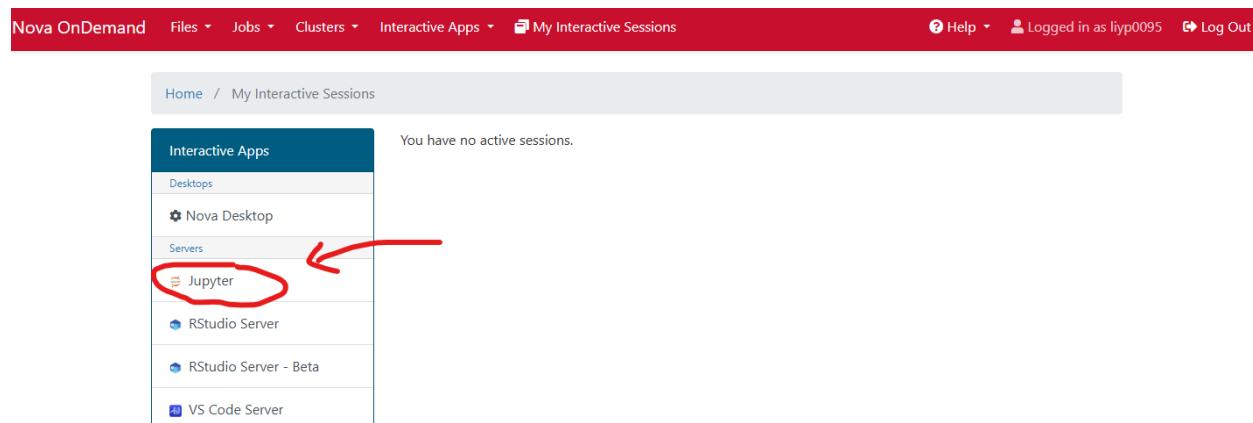
1. Open [https://nova-on-demand.its.iastate.edu/](https://nova-ondemand.its.iastate.edu/)

 If you're off-campus, make sure you're connected to ISU's VPN.



The screenshot shows the Nova OnDemand homepage. At the top, there is a navigation bar with links for Nova OnDemand, Files, Jobs, Clusters, Interactive Apps, and My Interactive Sessions. To the right of the navigation bar, it says "Logged in as liyp0095" and "Log Out". Below the navigation bar is the Nova OnDemand logo, which features a stylized eye-like icon above the text "NOVA OnDemand ACCELERATING SCIENCE". A red arrow points from the top center towards the "Interactive Apps" link in the navigation bar. The main content area below the logo displays a message: "OnDemand provides an integrated, single access point for all of your HPC resources." followed by "Message of the Day" and "Nova OnDemand".

2. Click "My Interactive Sessions", then select **Jupyter** from the available options.



The screenshot shows the "My Interactive Sessions" page. At the top, there is a navigation bar with links for Nova OnDemand, Files, Jobs, Clusters, Interactive Apps, and My Interactive Sessions. To the right of the navigation bar, it says "Logged in as liyp0095" and "Log Out". Below the navigation bar, the page title is "Home / My Interactive Sessions". The main content area shows a list titled "Interactive Apps" under the heading "Desks". It lists several options: "Nova Desktop" (selected), "Servers", "Jupyter" (circled with a red arrow pointing to it), "RStudio Server", "RStudio Server - Beta", and "VS Code Server". To the right of the list, a message says "You have no active sessions."

3. Configure your session with the following settings:
 - a. **Account:** qli-lab
 - b. **Queue:** interactive
 - c. **Number of Hours:** Select based on your need
 - d. **Memory Required:** Select as needed
 - e. **GRES** (for GPU jobs only): If needed, input something like gpu:a100:1 (Format is gpu:TYPE:COUNT, where TYPE is a100 or v100. Example: gpu:a100:1)
 - f. **Jupyter Notebook vs Lab:** Choose **Lab**

Leave other fields as default or modify as needed, then click **Launch**.

4. After successful resource allocation, the status will become **Running**. Click **Connect to Jupyter** to open your Jupyter Lab interface.

The screenshot shows two parts of a user interface for managing Jupyter sessions.

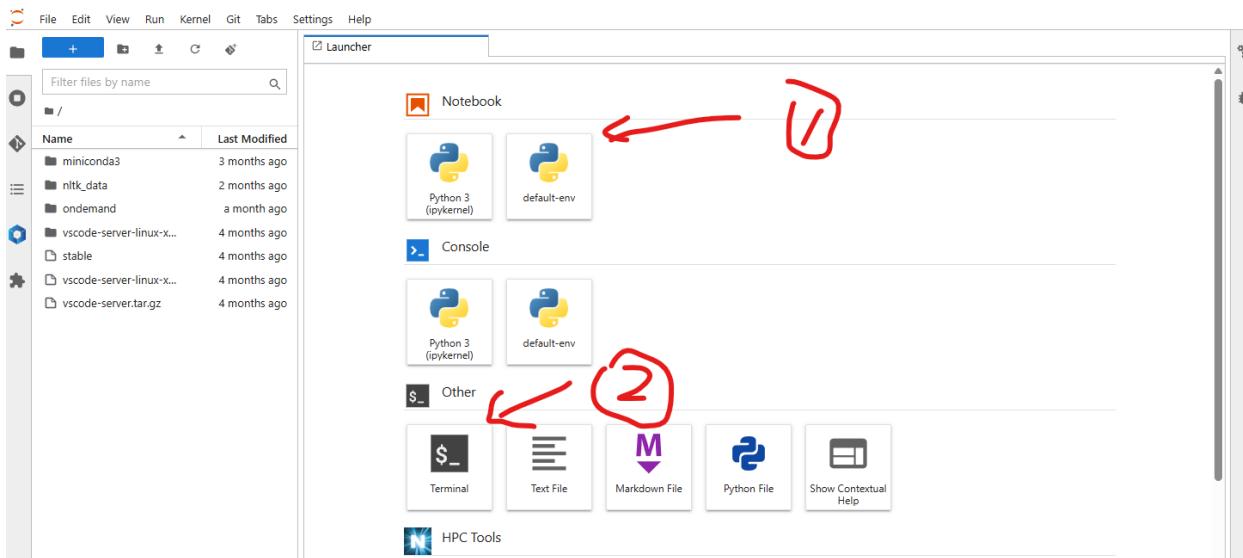
Top Panel (Session Creation):

- A green header bar indicates "Session was successfully created."
- The main area shows "Home / My Interactive Sessions".
- A sidebar titled "Interactive Apps" lists "Desktops" (Nova Desktop, Jupyter), "Servers" (RStudio Server, RStudio Server - Beta), and "VS Code Server".
- The central panel displays a session named "Jupyter (6366697)". It includes details: Host (nova22-gpu-5), Created at (2025-05-16 19:13:56 CDT), Time Remaining (59 minutes), Session ID (a7fe4924-a211-4585-9320-8fbe767d7519), and a "Connect to Jupyter" button.
- Two red arrows point to the "Connect to Jupyter" button and the "Running" status indicator.

Bottom Panel (Jupyter Lab Interface):

- A Jupyter Lab interface window is shown with a toolbar at the top: File, Edit, View, Run, Kernel, Git, Tabs, Settings, Help.
- The left sidebar shows a file tree with items like miniconda3, nltk_data, ondemand, vscode-server-linux-x..., stable, vscode-server-linux-x..., and vscode-server.tar.gz.
- The right sidebar is a "Launcher" pane with sections: Notebook (Python 3 (ipykernel), default-env), Console (Python 3 (ipykernel), default-env), Other (Terminal, Text File, Markdown File, Python File, Show Contextual Help), and HPC Tools.

Jupyter Lab Usage



Choosing a Kernel or Terminal

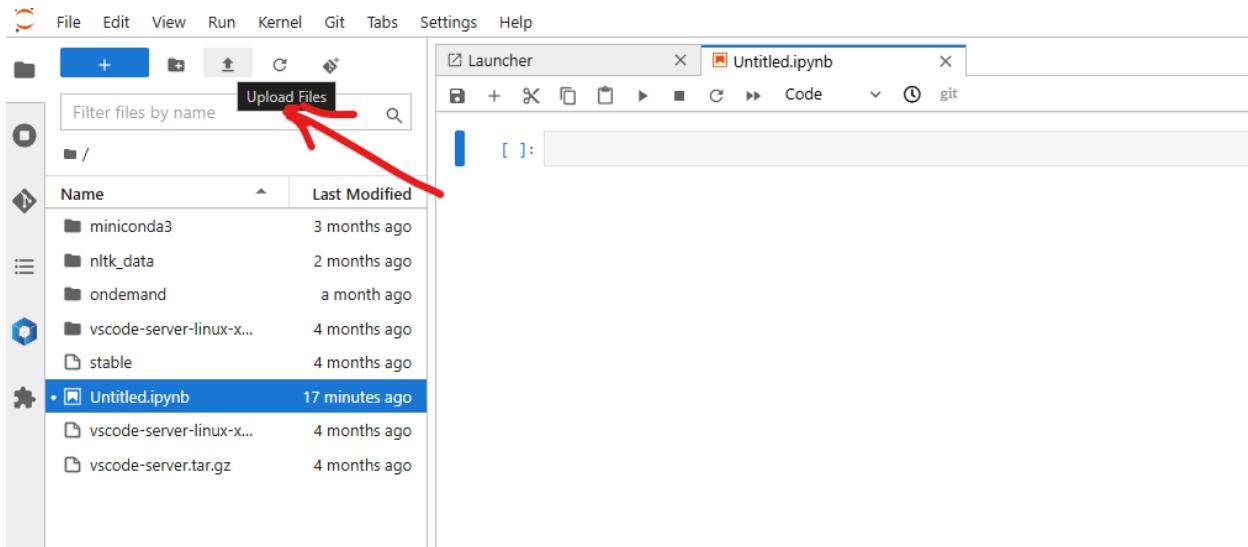
1. To open a Notebook, select Python 3 (ipykernel).
2. To open a Terminal, choose the corresponding icon.

Upload a Demo CSV File

1. Create a file named example.csv on your local machine with the following content:

```
name,age,city
Alice,30,New York
Bob,25,San Francisco
Charlie,35,Chicago
```

2. Use the Jupyter Lab file upload interface to upload example.csv.



3. Run the CSV Processing Code

- Paste the following code into notebook cells and run it:

```
!pip install pandas
!pip install matplotlib
...
import pandas as pd

filename = "example.csv"
df = pd.read_csv(filename)
print(f"Load file {filename}")
display(df.head())

Load file example.csv
   name  age   city
0  Alice  30  New York
1    Bob  25  San Francisco
2  Charlie  35      Chicago

import matplotlib.pyplot as plt

if 'age' in df.columns:
    plt.hist(df['age'].dropna(), bins=10)
    plt.title('Age Distribution')
    plt.xlabel('Age')
    plt.ylabel('Frequency')
    plt.show()
```

The code reads a CSV file named 'example.csv' and displays its first few rows. It then plots a histogram of the 'age' column with 10 bins, titled 'Age Distribution', showing frequency on the y-axis and age on the x-axis. The histogram shows three bars at ages 26, 30, and 34, each reaching a frequency of 1.0.

Code explanation:

```
# Install dependencies
!pip install pandas
!pip install matplotlib

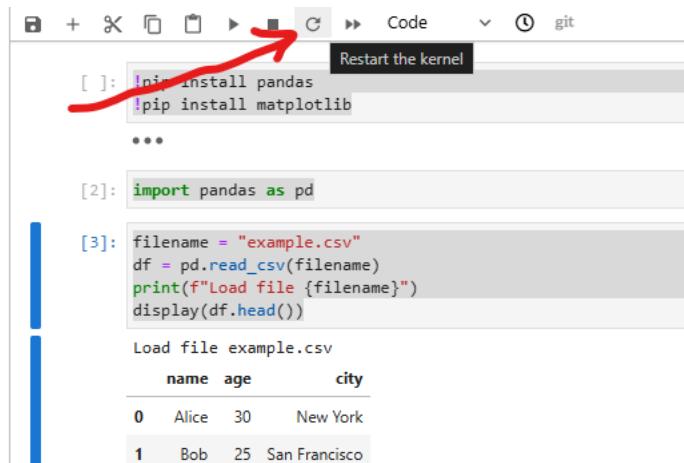
# Import libraries
import pandas as pd
import matplotlib.pyplot as plt

# Load the CSV file
filename = "example.csv"
df = pd.read_csv(filename)
print(f"Load file {filename}")
display(df.head())

# Plot histogram of the 'age' column
if 'age' in df.columns:
    plt.hist(df['age'].dropna(), bins=10)
    plt.title('Age Distribution')
    plt.xlabel('Age')
    plt.ylabel('Frequency')
    plt.show()
```

Troubleshooting

If you still encounter errors like "module not found" after installing packages, try clicking **"Restart Kernel"** and re-running the cells.



The screenshot shows the Jupyter Notebook interface. At the top, there's a toolbar with icons for file operations, a search bar, and a 'Code' button. Below the toolbar, the 'Kernel' menu is open, with the option 'Restart the kernel' highlighted. The main area contains three code cells. Cell [1] contains the command '!pip install pandas' with a red arrow pointing to it. Cell [2] contains the import statement 'import pandas as pd'. Cell [3] contains the code for loading a CSV file and displaying its head. The output of cell [3] shows the first two rows of a DataFrame:

	name	age	city
0	Alice	30	New York
1	Bob	25	San Francisco

Useful Links

Nova OnDemand: <https://www.hpc.iastate.edu/guides/jupyterlab>

JupyterLab Document: <https://jupyterlab.readthedocs.io/en/stable/>

Using a Conda Environment in an OOD Jupyter Notebook:

<https://www.hpc.iastate.edu/guides/virtual-environments/conda-environments#Example3>

Conda environments management

<https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html>