

Tutorial on Datahub

ECE 176

Presenter: Isabella Liu, Credit: Jianglong Ye, Jiarui Xu

Content

1. Datahub Basics
2. Jupyter Notebook Basics
3. SSH to Datahub (Optional)
4. How to Start with Assignment

Datahub

- Link: <https://datahub.ucsd.edu/>
- Log in with our UCSD account
- Multiple types of machines are available:

Select Your Notebook Environment

ECE176_WI25_A00 - Xiaolong Wang [WI25]
ghcr.io/ucsd-ets/scipy-ml-notebook:2025.1-stable (8 CPU, 16G RAM, 1 GPU)

ECE176_WI25_A00 - Xiaolong Wang [WI25]
ghcr.io/ucsd-ets/scipy-ml-notebook:2025.1-stable (8 CPU, 16G RAM)

ECE176_WI25_A00 - Xiaolong Wang [WI25]
ghcr.io/ucsd-ets/datascience-notebook:2025.1-stable (2 CPU, 8G RAM)

Launch Environment

Datahub Interface

- Create
 - Jupyter Notebook
 - Terminal

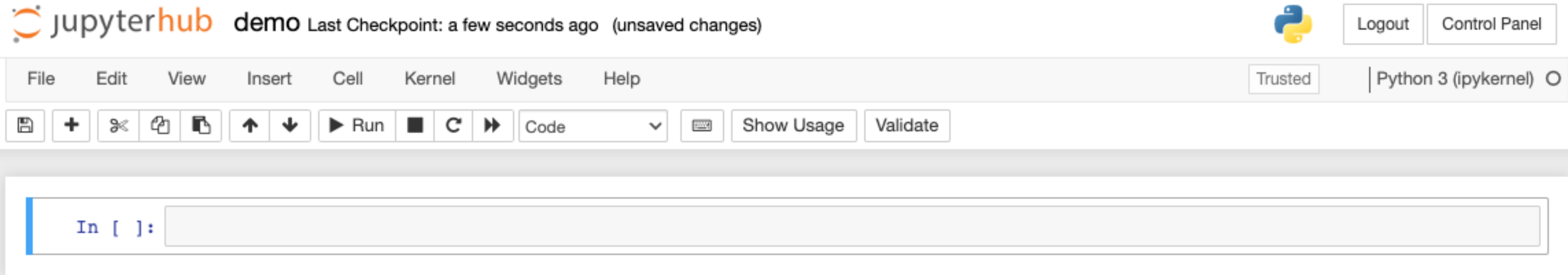
The screenshot displays the JupyterHub interface. At the top left is the JupyterHub logo. On the top right, there are buttons for "Logout" and "Control Panel". Below the logo, a navigation bar contains tabs for "Files", "Running", "Clusters", "Formgrader", "Courses", and "Assignments". The "Files" tab is active. Below the navigation bar, there is a prompt "Select items to perform actions on them." and buttons for "Upload", "New", and a refresh icon. The main content area shows a file browser view with a table of files and folders. The table has columns for "Name", "Last Modified", and "File size". The files listed are: a folder named "_MACOSX" (4 days ago), a folder named "assignment1" (3 minutes ago), a folder named "private" (4 days ago), a folder named "public" (22 days ago), and a file named "assignment1.zip" (4 days ago, 48.6 kB).

<input type="checkbox"/> 0	Name ↓	Last Modified	File size
<input type="checkbox"/>	folder _MACOSX	4 days ago	
<input type="checkbox"/>	folder assignment1	3 minutes ago	
<input type="checkbox"/>	folder private	4 days ago	
<input type="checkbox"/>	folder public	22 days ago	
<input type="checkbox"/>	file assignment1.zip	4 days ago	48.6 kB

Let's try it.

Jupyter Notebook

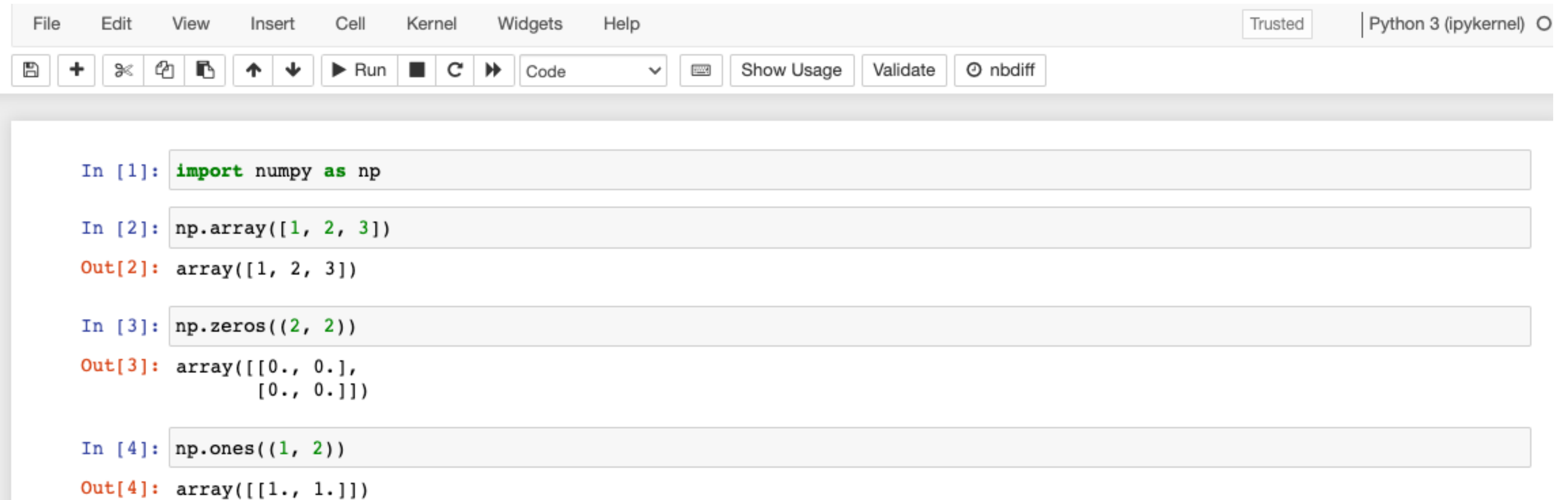
- A powerful tool for interactive python development.
 - It can contain code, output, images, text, and so on.



The screenshot displays the Jupyter Notebook interface. At the top left, the JupyterHub logo is followed by the text "demo" and "Last Checkpoint: a few seconds ago (unsaved changes)". On the top right, there is a Python logo, a "Logout" button, and a "Control Panel" button. Below this is a menu bar with options: File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. To the right of the menu bar, it shows "Trusted" and "Python 3 (ipykernel)". Below the menu bar is a toolbar with icons for saving, adding, deleting, copying, pasting, moving up/down, running, stopping, and refreshing, along with a "Code" dropdown menu, "Show Usage", and "Validate" buttons. The main area contains a code cell with the prompt "In []:" followed by an empty text input field.

Code Block

- Contains python code
 - A block can be run many times.
 - Blocks can be run in any order.



The screenshot shows a Jupyter Notebook interface with a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, running, and validation. The notebook contains four code blocks:

```
In [1]: import numpy as np
```

```
In [2]: np.array([1, 2, 3])
```

```
Out[2]: array([1, 2, 3])
```

```
In [3]: np.zeros((2, 2))
```

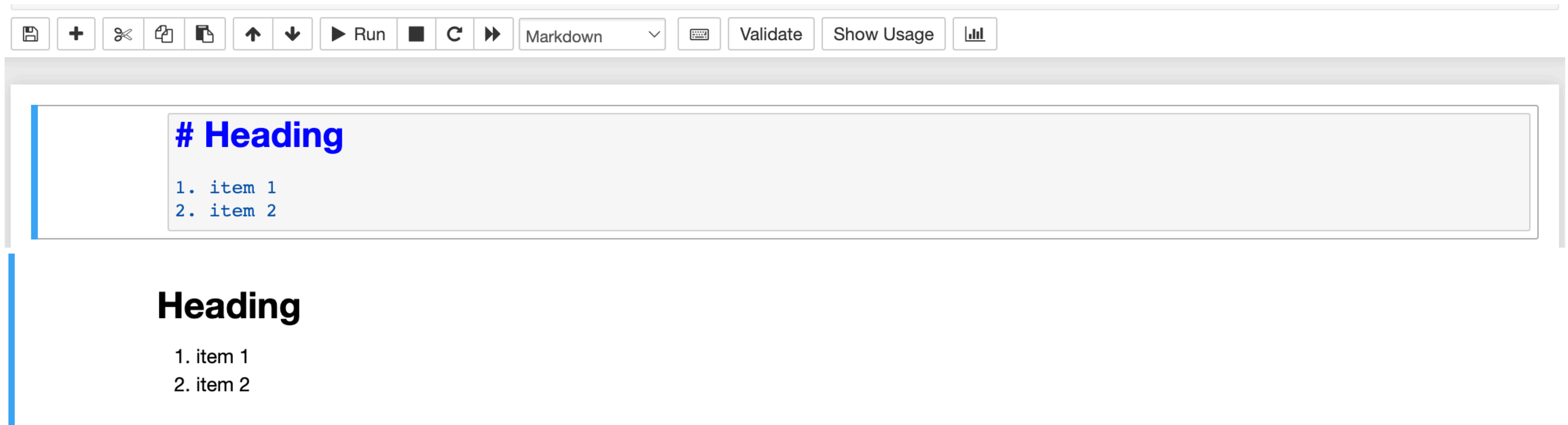
```
Out[3]: array([[0., 0.],
              [0., 0.]])
```

```
In [4]: np.ones((1, 2))
```

```
Out[4]: array([[1., 1.]])
```

Markdown block

- You can write markdown in notebook.



The screenshot shows a notebook interface with a toolbar at the top containing icons for save, add, cut, copy, undo, redo, run, and refresh, along with buttons for 'Markdown', 'Validate', 'Show Usage', and a chart icon. Below the toolbar, a code editor displays the following Markdown text:

```
# Heading  
1. item 1  
2. item 2
```

Below the code editor, the rendered output is shown, featuring a large heading and a list of items:

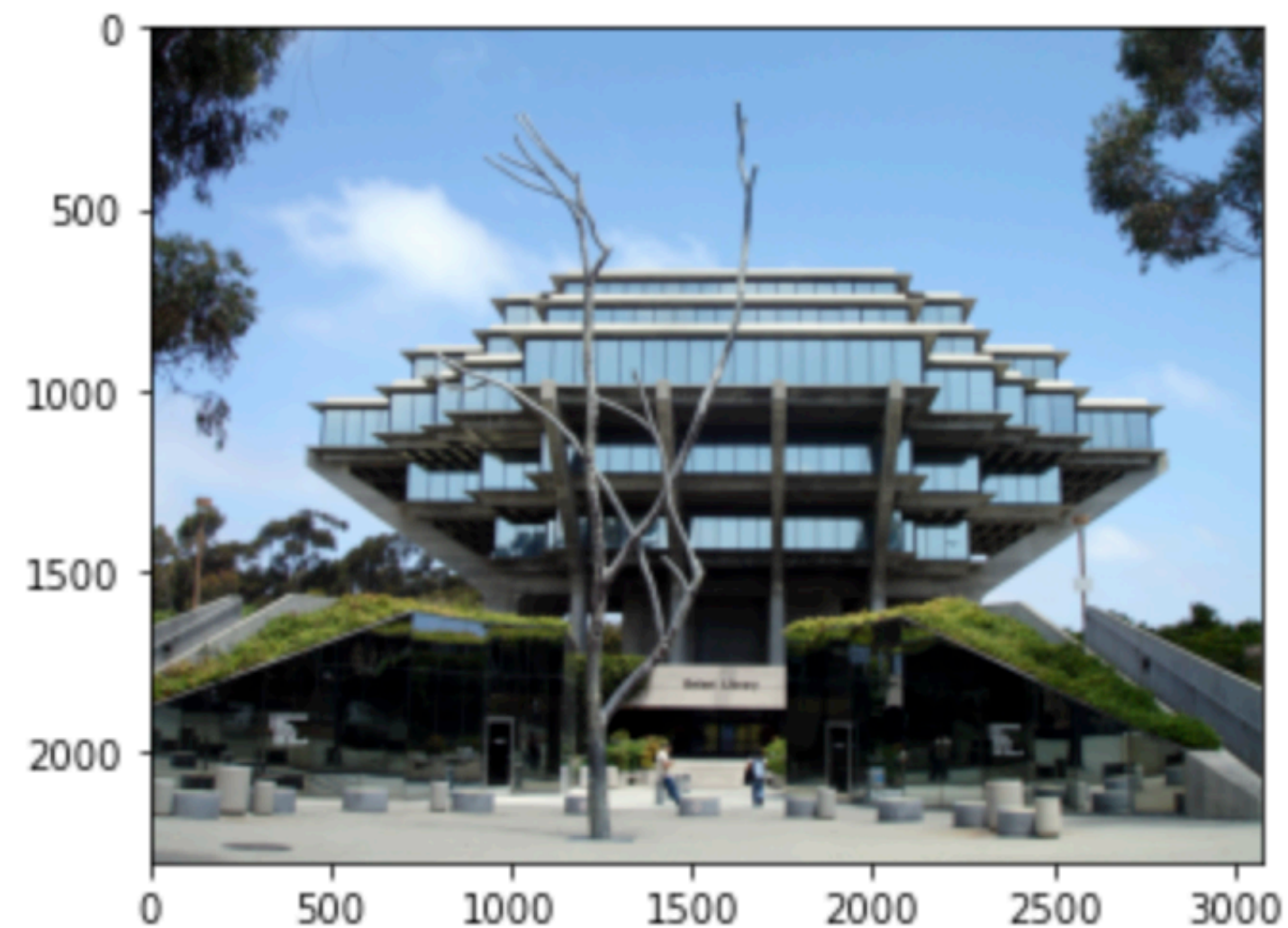
Heading

1. item 1
2. item 2

Visualize Image

- Show image with matplotlib
 - high resolution: `plt.rcParams["figure.dpi"] = 300`

```
In [3]: img = cv2.imread("./library.jpg")  
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)  
plt.imshow(img)  
plt.show()
```



Run Shell Command

- Start with ! (exclamation mark)

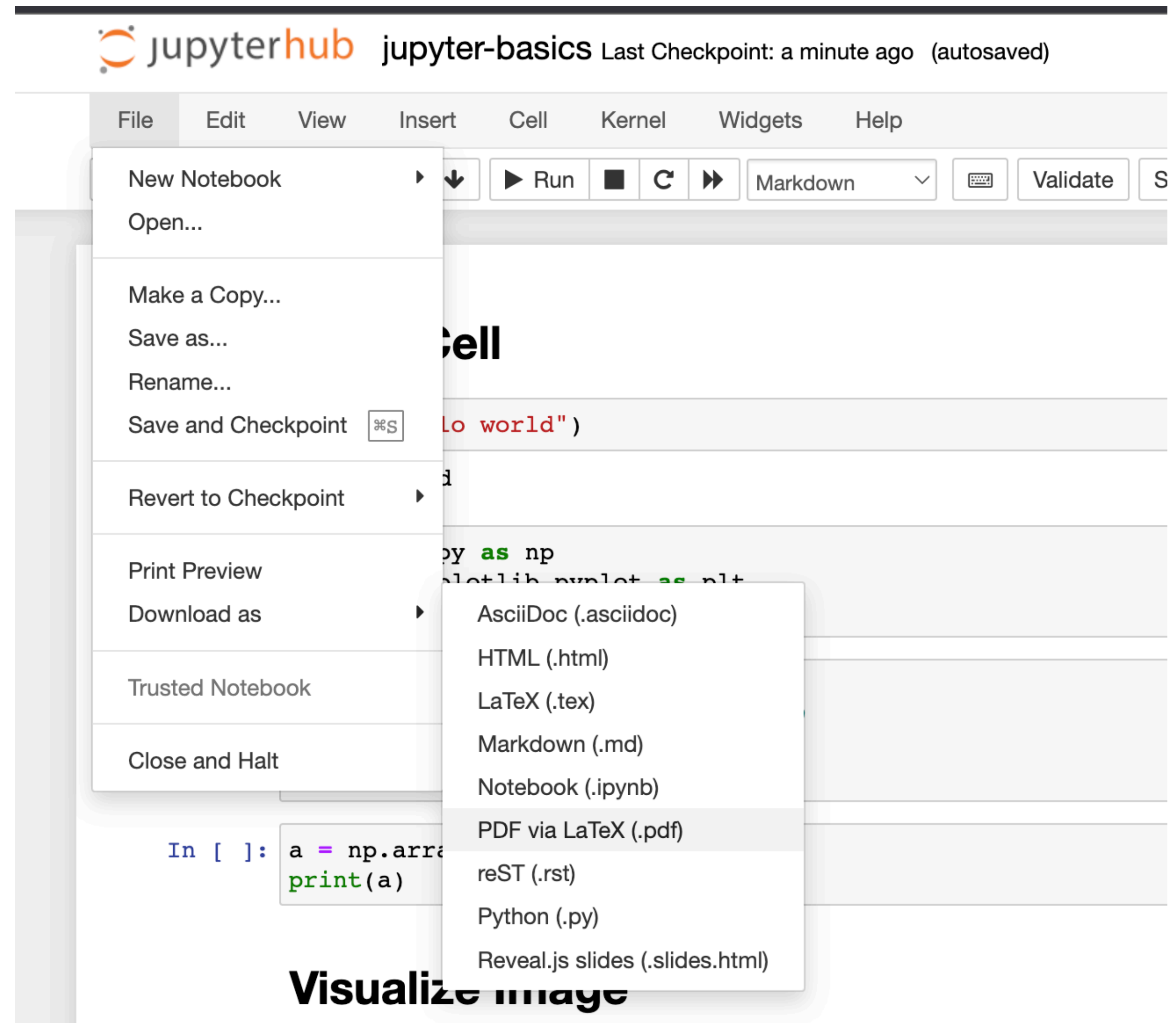
Run Command

```
In [11]: !pip install tqdm
```

```
Defaulting to user installation because normal site-packages is not writeable  
Requirement already satisfied: tqdm in /opt/conda/lib/python3.9/site-packages (4.61.2)
```

Export as PDF

- You need to do this for the assignment
 - Upload generated PDF to Gradescope
- You do not need to install LaTeX package
 - Sometimes it fails to export PDF. Just wait or try later.
 - Export as html and print it as PDF or directly print the notebook as PDF is acceptable as well.
(But the file will be different)



Let's try it.

SSH to datahub

- Command: `ssh user-name@dsmlp-login.ucsd.edu` (password required)
- SSH Key
 - Run `ssh-keygen` in your local PC
 - Upload ssh key to the remote machine
 - `cat ~/.ssh/id_rsa.pub | ssh user-name@dsmlp-login.ucsd.edu "mkdir -p ~/.ssh && chmod 700 ~/.ssh && cat >> ~/.ssh/authorized_keys"`
 - Or edit `~/.ssh/authorized_keys` on remote machine manually
- Reference: [How to: Launching Containers From the Command Line](#)

Container

- We automatically create a GPU container when we start the VSCode and delete it after we close the VSCode
- You may need some kubectl commands:
 - `kubectl get pod`: check running containers
 - `kubectl delete pod vscode-dsmlp`: delete vscode container (if it's not deleted automatically)
 - `kubectl delete pod --all`: delete all running containers

VSCode

- Install Remote-SSH extension
- Add SSH config as follows:

Host vscode-dsmlp

HostName dsmlp-login.ucsd.edu

User user-name

UserKnownHostsFile /dev/null

StrictHostKeyChecking no

ProxyCommand ssh user-name@dsmlp-login.ucsd.edu /opt/launch-sh/bin/launch-scipy-ml.sh -p normal -c 8 -m 16 -g 1 -H -N vscode-dsmlp

- -c means CPU, -m means memory, -g means GPU
- Reference: [How to: Launching Containers From the Command Line](#) (there is some issue in official doc, please follow this instruction)

VSCode

- Install python extensions (full experience)

The screenshot shows the VS Code Jupyter Notebook interface. At the top, the file name is 'jupyter-basic.ipynb'. Below it, the code cell contains the following Python code:

```
import numpy as np
np.zeros
```

The code cell is highlighted in green. Below the code, the output is 'True'. A dropdown menu is open, showing suggestions for the 'np.zeros' attribute:

- zeros
- zeros_like
- trim_zeros

The 'zeros' suggestion is selected, and a tooltip is displayed on the right side of the screen. The tooltip contains the following information:

zeros: (shape: Any, dtype: ...)

zeros(shape, dtype=float, order='C', *, like=None)

Return a new array of given shape and type, filled with zeros.

Parameters

shape : int or tuple of ints
Shape of the new array, e.g., (2, 3) or 2.

dtype : data-type, optional
The desired data-type for the array, e.g.,

VSCoDe (without logging in container)

- More stable but it can only support edit function.
- Add a simpler SSH config as follows:

```
Host vscode-dsmlp-edit-only
```

```
    HostName dsmlp-login.ucsd.edu
```

```
    User user-name
```

- You need to run container and python program manually.

Try the Assignment (Website)

- Upload assignment zip file to the datahub website
- Read README.md carefully
- Download the dataset with our provided script

- Implement algorithms in py files
- Run all blocks in jupyter notebook and answer all questions

- Export notebooks as PDFs
- Upload required codes and PDFs to Gradescope

Try the Assignment (VSCode)

- Upload assignment zip file to dsmlp-login.ucsd.edu
 - `scp -r ./file user-name@dsmlp-login.ucsd.edu:/path` (or drag it to the VSCode)
- Read README.md carefully
- Download the dataset with our provided script

- Implement algorithms in py files
- Run all blocks in jupyter notebook and answer all questions

- Export notebooks as PDFs
- Upload required codes and PDFs to Gradescope

Thanks