

ACES: AI/ML TechLab

Accelerating AI/ML Workflows on a Composable Cyberinfrastructure

Zhenhua He

10/03/2023



High Performance
Research Computing
DIVISION OF RESEARCH



Acknowledgements

This short course is sponsored in part by the CSSN Community Engagement Program Travel Award. We gratefully acknowledge their support in providing this community resource.



CSSN Community Engagement Program
CCEP

Please submit by October 1 for SC23

CCEP Travel Rewards Program

CCEP (CSSN Community Engagement Program) gives travel rewards to ANYONE for community engagement, feedback forums, documentation and much more!

Submissions are reviewed once a month. Please submit by October 1 for SC23 or two months prior to the conference that you desire to attend.

AI/ML TechLab

Lab I. JupyterLab (30 mins)

We will load required modules and activate virtual environment and run JupyterLab on HPRC ACES portal.

Lab II. Data Exploration (30 mins)

We will go through some examples with two popular Python libraries: Pandas and Matplotlib for data exploration.

04

Lab IV. Deep Learning (30 minutes)

We will learn how to use PyTorch to build and train a simple image classification model with deep neural network (DNN).

01

Q&A
(5 mins/lab)

03

Lab III Machine Learning (30 minutes)

We will learn to use scikit-learn library for linear regression and classification applications.

02

Figure 1. Structure of the AI/ML TechLab.

Lab I. JupyterLab



File Edit View Run Kernel Tabs Settings Help

Files

- notebooks
- Data.ipynb
- Fasta.ipynb
- Julia.ipynb
- Lorenz.ipynb** (seconds ago)
- R.ipynb
- iris.csv
- lightning.json
- lorenz.py

Running

Commands

Cell Tools

sigma 10.00
beta 2.67
rho 28.00

Code

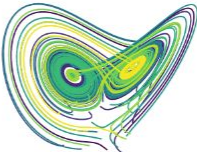
In this Notebook we explore the Lorenz system of differential equations:

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy \end{aligned}$$

Let's call the function once to view the solutions. For this set of parameters, we see the trajectories swirling around two points, called attractors.

```
In [4]: from lorenz import solve_lorenz
t, x_t = solve_lorenz(N=10)
```

Output View



```
9 def solve_lorenz(N=10, max_time=4.0, sigma=10.0, beta=8./3, rho=28.0):
10     """Plot a solution to the Lorenz differential equations."""
11     fig = plt.figure()
12     ax = fig.add_axes([0, 0, 1, 1], projection='3d')
13     ax.axis('off')
14
15     # prepare the axes limits
16     ax.set_xlim((-25, 25))
17     ax.set_ylim((-35, 35))
18     ax.set_zlim((5, 55))
19
20     def lorenz_deriv(x_y_z, t0, sigma=sigma, beta=beta, rho=rho):
21         """Compute the time-derivative of a Lorenz system."""
22         x, y, z = x_y_z
23         return [sigma * (y - x), x * (rho - z) - y, x * y - beta * z]
24
25     # Choose random starting points, uniformly distributed from -15 to 15
26     np.random.seed(1)
27     x0 = -15 + 30 * np.random.random((N, 3))
28
```

L1 - Resources

- Texas A&M High Performance Research Computing (HPRC)
- ACES Quick Start Guide
- ACES Portal (ACCESS)
- ACCESS Documentation
- HPRC YouTube Channel
- help@hprc.tamu.edu

NSF ACES

Accelerating Computing for Emerging Sciences

Our Mission:

- Offer an accelerator testbed for numerical simulations and **AI/ML workloads**
- Provide consulting, technical guidance, and training to researchers
- Collaborate on computational and data-enabled research.



ACES Accelerators

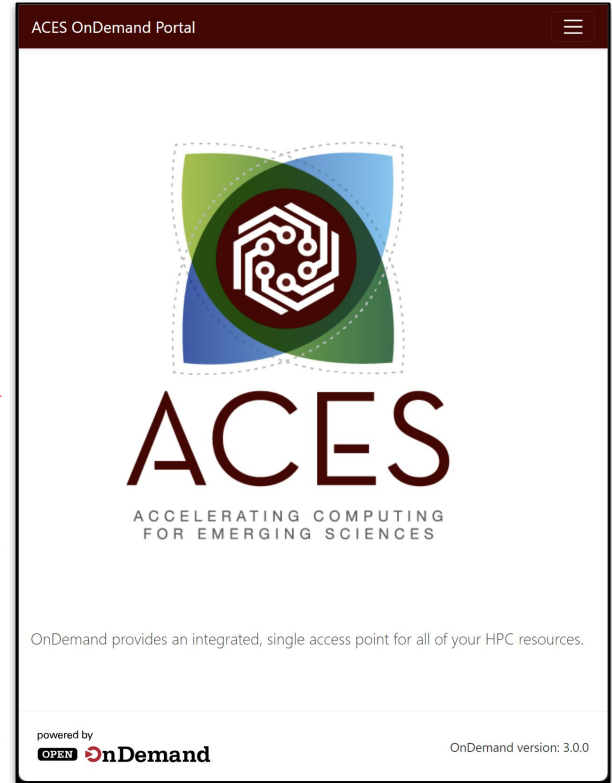
Component	Quantity	Description
Graphcore IPU	32	16 Colossus GC200 IPUs, 16 Bow IPUs. Each IPU group hosted with a CPU server as a POD16 on a 100 GbE RoCE fabric
Intel PAC D5005 FPGA	2	Accelerator with Intel Stratix 10 GX FPGA and 32 GB DDR4
BittWare IA-840F FPGA	2	Accelerator with Agilex AGF027 FPGA and 64 GB of DDR4
NextSilicon Coprocessor	2	Reconfigurable accelerator with an optimizer continuously evaluating application behavior.
NEC Vector Engine	8	Vector computing card (8 cores and HBM2 memory)
Intel Optane SSD	48	18 TB of Intel Optane SSDs addressable as memory w/ MemVerge Memory Machine.
NVIDIA H100 + A30	30 + 4	NVIDIA GPUs for HPC, DL Training, AI Inference
Intel PVC + ATS-P	12 + 22	Software Development Platform for PVC

ACES Portal



ACES Portal portal-aces.hprc.tamu.edu
is the web-based user interface for the ACES cluster

Open OnDemand (OOD) is an advanced web-based
graphical interface framework for HPC users



Authentication via CILogon

Log-in using your ACCESS CI credentials.

The screenshot shows the ACCESS consent screen. At the top left is the ACCESS logo, and at the top right is the 'Powered By CILogon' logo. Below the logo is a teal bar with the text 'Consent to Attribute Release'. Underneath, there is a list of information that TAMU ACES ACCESS OIDC requests access to, including CILogon user identifier, name, email address, and username/affiliation. A red box highlights the 'Select an Identity Provider' section, which contains a dropdown menu with 'ACCESS CI (XSEDE)' selected. Below the dropdown is a 'Remember this selection' checkbox and a 'Log On' button. At the bottom, there is a note about agreeing to the privacy policy by selecting 'Log On'.

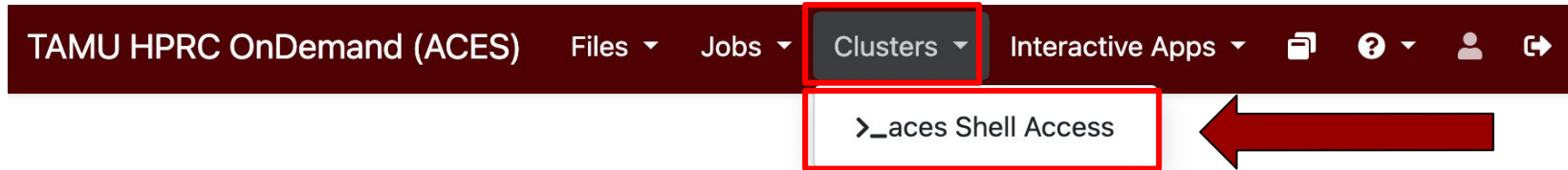
The screenshot shows the ACCESS CILogon login page. At the top left is the ACCESS logo, and at the top right is the CILogon logo. Below the logo is the text 'Login to CILogon'. There are two input fields: 'ACCESS Username' and 'ACCESS Password'. Below the password field is a checkbox for 'Don't Remember Login' and a teal 'Login' button. To the right of the login fields, there is a note about XSEDE accounts and links for 'Register for an ACCESS Account', 'Forgot your password?', and 'Need Help?'. At the bottom, there is a link for 'Click Here for Assistance'.

This is a close-up of the 'Select an Identity Provider' dropdown menu. The dropdown is open, showing 'ACCESS CI (XSEDE)' as the selected option. A red box highlights the entire dropdown area.

Select the Identity Provider appropriate for your account.

Get a Shell on ACES

Click on “Clusters” menu →>_aces Shell Access



Success!

Welcome to the ACES login node.

Check which login node you are on.

```
Host: login.aces Themes: Default
| Consulting: help@hprc.tamu.edu (preferred) or (979) 845-0219 |
| ACES Documentation: https://hprc.tamu.edu/kb/User-Guides/ACES |
| FASTER Documentation: https://hprc.tamu.edu/kb/User-Guides/FASTER |
| Grace Documentation: https://hprc.tamu.edu/kb/User-Guides/Grace |
| Terra Documentation: https://hprc.tamu.edu/kb/User-Guides/Terra |
| YouTube Channel: https://www.youtube.com/texasamhprc |
=====
*****
*                               === IMPORTANT POLICY INFORMATION ===                               *
* - Unauthorized use of HPRC resources is prohibited and subject to criminal prosecution. *
* - Use of HPRC resources in violation of United States export control laws and regulations is prohibited. Current HPRC staff members are US citizens and legal residents. *
* - Sharing HPRC account and password information is in violation of Texas State Law. Any shared accounts will be DISABLED. *
* - Authorized users must also adhere to ALL policies at: https://hprc.tamu.edu/policies/ *
*****

!! WARNING: THERE ARE ONLY NIGHTLY BACKUPS OF USER HOME DIRECTORIES. !!

Please restrict usage to 8_CORES across ALL login nodes.
Users found in violation of this policy will be SUSPENDED.

To see these messages again, run the motd command.
Your current disk quotas are:
Disk          Disk Usage  Limit  File Usage  Limit
/home/u.zh108696      4.0G      10.0G    2361      10000
/scratch/user/u.zh108696 275.4G    1.0T    352057   1000000
Type 'showquota' to view these quotas again.
[u.zh108696@aces-login1 ~]$
```

Commands to copy the materials

- Navigate to your personal scratch directory

```
$ cd $SCRATCH
```

- Files for this course are located at

```
/scratch/training/ai_tech_labs
```

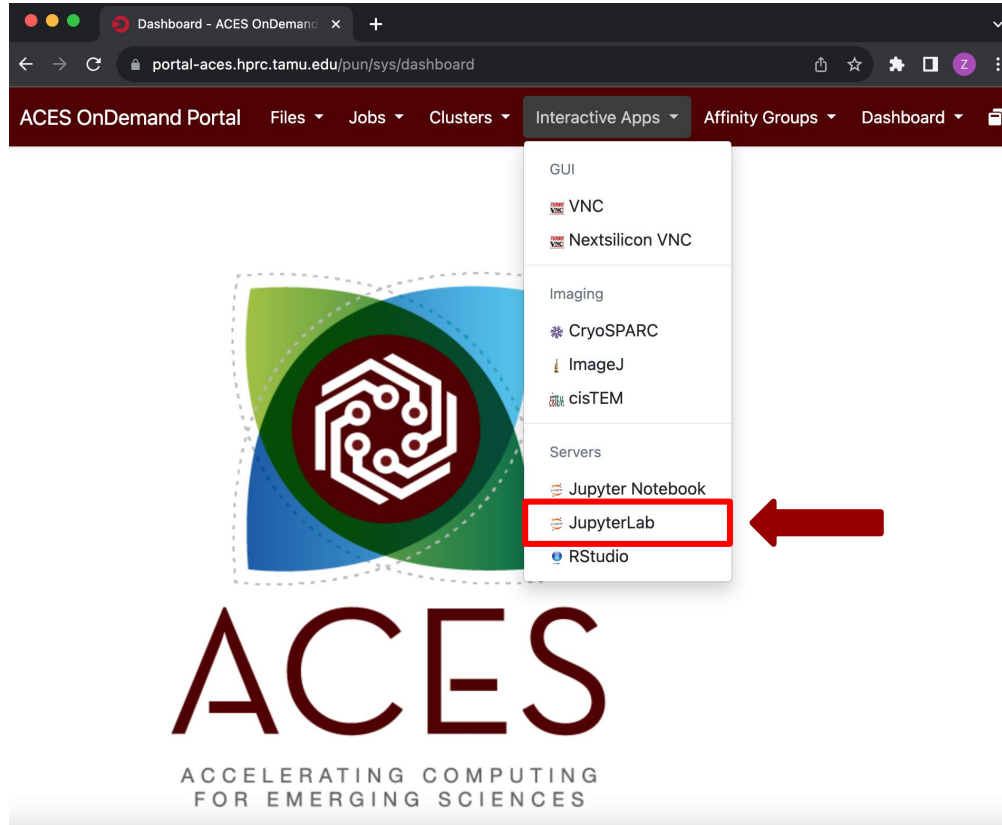
Make a copy in your personal scratch directory

```
$ cp -r /scratch/training/ai_tech_labs $SCRATCH
```

- Enter this directory (your local copy)

```
$ cd ai_tech_labs
```

Go to JupyterLab Page



The image shows a web browser window displaying the ACES OnDemand Portal dashboard. The URL in the address bar is `portal-aces.hprc.tamu.edu/pun/sys/dashboard`. The navigation bar includes links for "ACES OnDemand Portal", "Files", "Jobs", "Clusters", "Interactive Apps", "Affinity Groups", and "Dashboard". The "Interactive Apps" menu is open, listing several options: "GUI", "VNC", "Nextsilicon VNC", "Imaging", "CryoSPARC", "ImageJ", "cisTEM", "Servers", "Jupyter Notebook", "JupyterLab", and "RStudio". The "JupyterLab" option is highlighted with a red box, and a red arrow points to it from the right. Below the menu is the ACES logo, which consists of a stylized circuit board pattern inside a green and blue circular shape, with the text "ACES" and "ACCELERATING COMPUTING FOR EMERGING SCIENCES" below it.

JupyterLab Page

portal-aces.hprc.tamu.edu/pun/sys/dashboard/batch_connect/dev/jupyterlab_shortcourse/ses...

ACES OnDemand Portal Files Jobs Clusters Interactive Apps Affinity Groups Dashboard

Home / My Interactive Sessions / JupyterLab (Short Course)

Interactive Apps

- GUI
- VNC
- Nextsilicon VNC
- Imaging
- CryoSPARC
- ImageJ
- ciSTEM
- Servers
- Jupyter Notebook
- JupyterLab
- RStudio

JupyterLab (Short Course)

This app will launch a [JupyterLab](#) server on the [ACES cluster](#).

Module

Anaconda3/2022.05

Optional Environment to be activated

/sw/hprc/sw/Anaconda3/2022.05/envs/ai-labs

Enter the name of the environment to be activated.

Leave blank to use the [default](#) environment for the selected Module.

Your optional conda environment must have been previously built with one of the Anaconda or Python modules listed in the Module option above. See [instructions](#).

Node type

First available GPU

Other fields:

Node Type: First available GPU

Number of GPUs: 1

Number of hours: 3

Number of cores: 3

Total memory (GB): 5

Option 1: Use a shared environment created by TAMU HPRC for this course

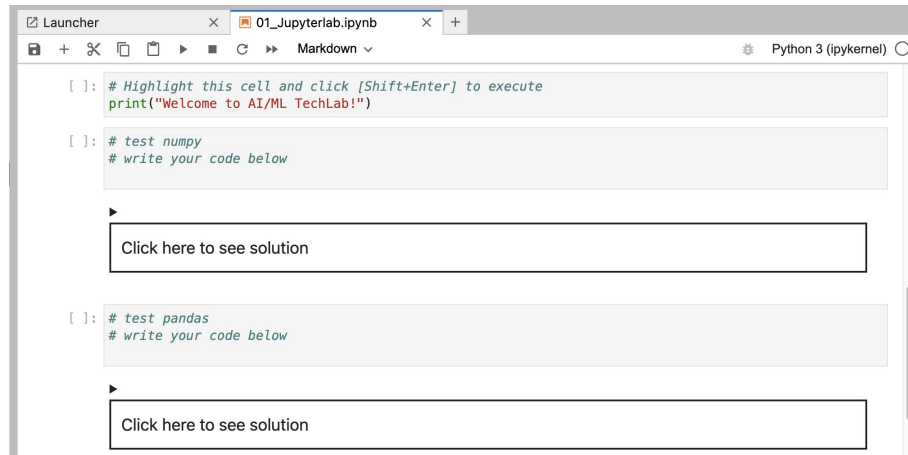
Path to the shared environment:
`/sw/hprc/sw/Anaconda3/2022.05/envs/ai-labs`

Connect to JupyterLab

The screenshot shows a web browser window with the URL `portal-aces.hprc.tamu.edu/pun/sys/dashboard/batch_connect/sessions`. The page title is "My Interactive Sessions - ACE". A navigation bar includes "ACES OnDemand Portal", "Files", "Jobs", "Clusters", "Interactive Apps", "Affinity Groups", and "Dashboard". A green notification bar at the top states "Session was successfully deleted." Below this is a breadcrumb "Home / My Interactive Sessions". On the left, a sidebar titled "Interactive Apps" lists various applications: GUI, VNC, Nextsilicon VNC, Imaging, CryoSPARC, ImageJ, cisTEM, Servers, Jupyter Notebook, JupyterLab, and RStudio. The main content area displays a "JupyterLab (5639)" session with a status of "1 node | 1 core | Running". The session details include: Host: `>_ac036`, Created at: 2023-09-27 10:13:08 CDT, Time Remaining: 57 minutes, and Session ID: 5b705a8e-f469-4e7c-907c-1b856c941774. A red button labeled "Delete" is visible. At the bottom of the session details, a blue button with an eye icon and the text "Connect to JupyterLab" is highlighted with a red box, and a red arrow points to it from the right.

Review and Exercise

- Log into ACES through ACES Portal (ACCESS)
- Copy the training materials to your \$SCRATCH directory
- Launch JupyterLab app
- In the notebook named *01_Jupyterlab.ipynb*, follow the instructions to import the required modules to make sure they have been loaded properly.



The screenshot shows a JupyterLab notebook window titled "01_Jupyterlab.ipynb". The interface includes a toolbar with icons for home, refresh, save, copy, paste, and navigation. The notebook content consists of three code cells:

```
[ ]: # Highlight this cell and click [Shift+Enter] to execute
print("Welcome to AI/ML TechLab!")
```

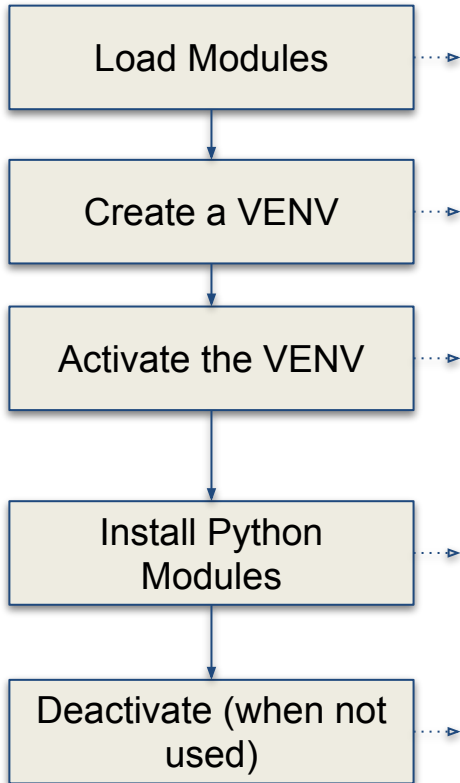
[]: # test numpy
write your code below

Click here to see solution

[]: # test pandas
write your code below

Click here to see solution

Option 2



```
# clean up and load Anaconda  
cd $SCRATCH  
module purge  
module load Anaconda3/2022.05
```

```
# create a Python virtual environment  
conda create -n ai-labs
```

```
# activate the virtual environment  
source activate ai-labs
```

```
# install required package to be used in the portal  
conda install -c anaconda jupyter  
conda install -c anaconda pandas  
conda install -c conda-forge matplotlib  
conda install -c anaconda scikit-learn  
conda install pytorch torchvision torchaudio  
pytorch-cuda=11.8 -c pytorch -c nvidia
```

```
# deactivate the virtual environment  
# source deactivate
```

JupyterLab Page

JupyterLab (Short Course) - A | x +

portal-aces.hprc.tamu.edu/pun/sys/dashboard/batch_connect/dev/jupyterlab_shortcourse/ses...

ACES OnDemand Portal Files Jobs Clusters Interactive Apps Affinity Groups Dashboard

Home / My Interactive Sessions / JupyterLab (Short Course)

Interactive Apps

- GUI
- VNC
- Nextsilicon VNC
- Imaging
- CryoSPARC
- ImageJ
- cisTEM
- Servers
- Jupyter Notebook
- JupyterLab
- RStudio

JupyterLab (Short Course)

This app will launch a [JupyterLab](#) server on the [ACES cluster](#).

Module

Anaconda3/2022.05

Optional Environment to be activated

ai-labs

Enter the name of the environment to be activated.

Leave blank to use the [default](#) environment for the selected Module.

Your optional conda environment must have been previously built with one of the Anaconda or Python modules listed in the Module option above. See [instructions](#).

Node type

First available GPU

Other fields:

Node Type: First available GPU

Number of GPUs: 1

Number of hours: 3

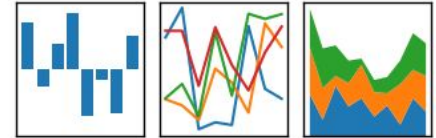
Number of cores: 3

Total memory (GB): 5

Lab II. Data Exploration

matplotlib 

pandas
 $y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$



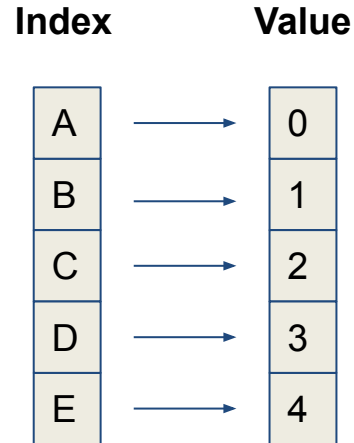
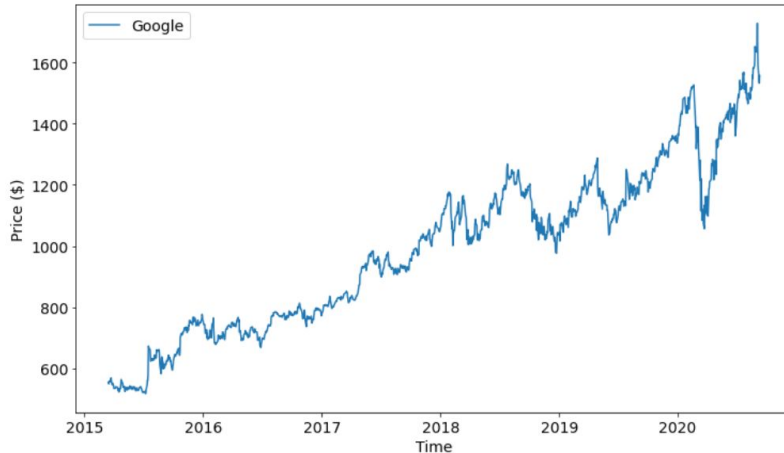
Data Structures

Pandas has two data structures that are descriptive and optimized for data with different dimensions.

- **Series:** 1D labeled array
- **DataFrame:** General 2D labeled, size-mutable tabular structure with potentially heterogeneously-typed columns

Series in pandas

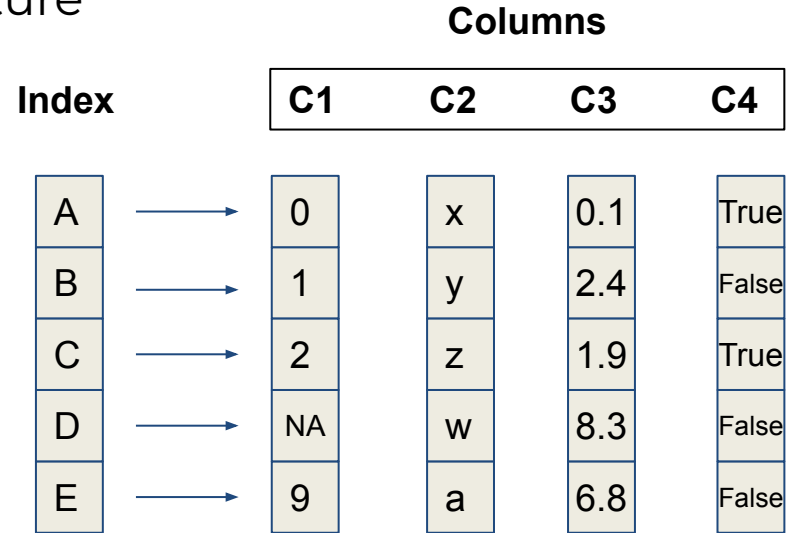
- One-dimensional labeled array
- Capable of holding any data type (integers, strings, floating point numbers, etc.)
- Example: time-series stock price data



DataFrame in pandas

- Primary Pandas data structure
- A dict-like container for Series objects
- Two-dimensional size-mutable
- Heterogeneous tabular data structure

A	B	C	D	E	F	G	H
id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors
7129300520	20141013T00	221900	3	1	1180	5650	1
6414100192	20141209T00	538000	3	2.25	2570	7242	2
5631500400	20150225T00	180000	2	1	770	10000	1
2487200875	20141209T00	604000	4	3	1960	5000	1
1954400510	20150218T00	510000	3	2	1680	8080	1
7237550310	20140512T00	1.23E+06	4	4.5	5420	101930	1
1321400060	20140627T00	257500	3	2.25	1715	6819	2
2008000270	20150115T00	291850	3	1.5	1060	9711	1
2414600126	20150415T00	229500	3	1	1780	7470	1



Pandas Learning Objectives

After this lesson, you will know how to:

- Create a DataFrame
- Retrieve a Row or Column
- Drop Entries
- Index, Select, and Filter data
- Sort data
- Input and Output



[JupyterLab Exercises](#)

Key Plotting Concepts in Matplotlib

- **Matplotlib: Figure**

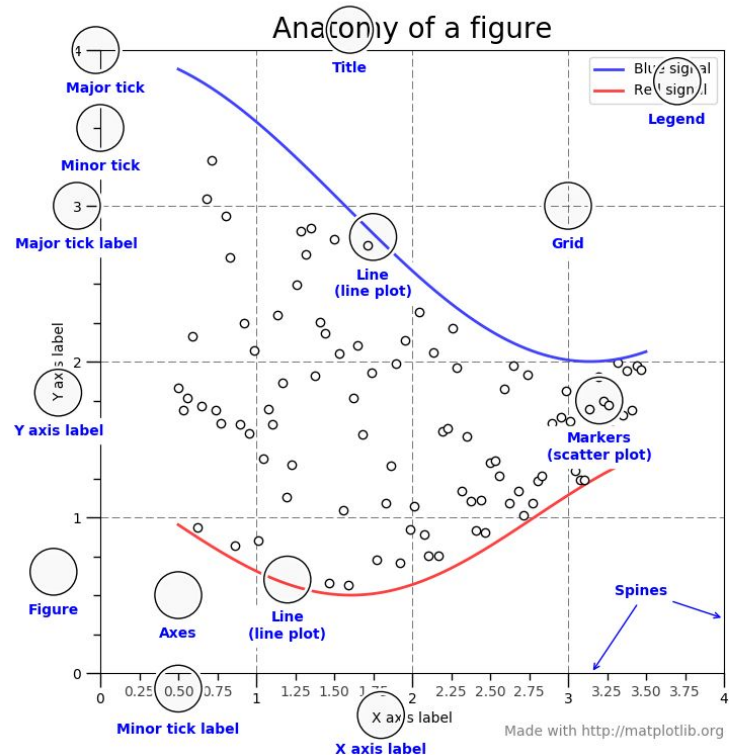
Figure is the object that keeps the whole image output. Adjustable parameters include:

1. Image size (`set_size_inches()`)
2. Whether to use `tight_layout` (`set_tight_layout()`)

- **Matplotlib: Axes**

Axes object represents the pair of axis that contain a single plot (x-axis and y-axis). The Axes object also has more adjustable parameters:

1. The plot frame (`set_frame_on()` or `set_frame_off()`)
2. X-axis and Y-axis limits (`set_xlim()` and `set_ylim()`)
3. X-axis and Y-axis Labels (`set_xlabel()` and `set_ylabel()`)
4. The plot title (`set_title()`)



(Credit: matplotlib.org)

Matplotlib Learning Objectives

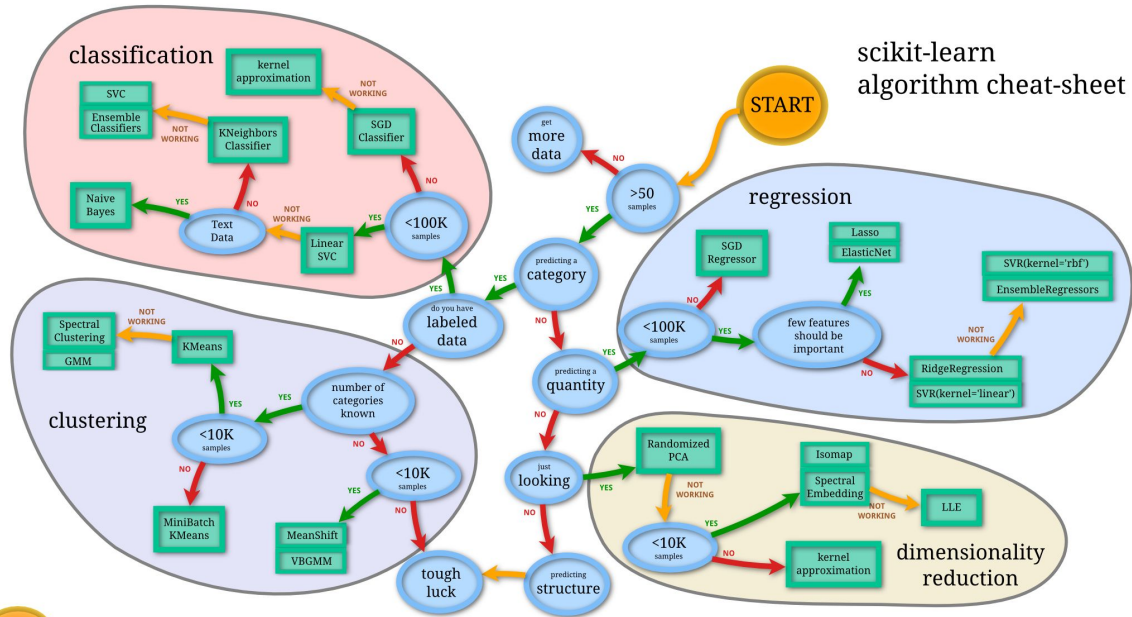
After this lesson, you will know how to:

- Scatter plot and Line plot
- Subplots
- Color map
- Contour figures
- 3D figures
 - Surface plots
 - Wire-frame plot
 - Contour plots with projections



JupyterLab Exercises

Lab III. Machine Learning



Main Features of scikit-learn



Classification

Identifying category of an object

Applications: Spam detection, image recognition.

Algorithms: SVM, nearest neighbors, random forest, and more...

Regression

Predicting a attribute for an object

Applications: Drug response, Stock prices.

Algorithms: SVR, nearest neighbors, random forest, and more...

Clustering

Grouping similar objects into sets

Applications: Customer segmentation, Grouping experiment outcomes

Algorithms: k-Means, spectral clustering, mean-shift, and more...

Dimension Reduction

Reducing the number of dimensions

Applications: Visualization, Increased efficiency
Algorithms: k-Means, feature selection, non-negative matrix factorization, and more...

Model Selection

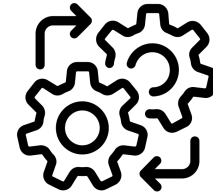
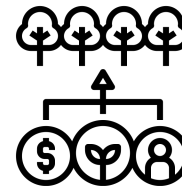
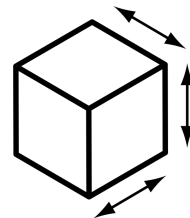
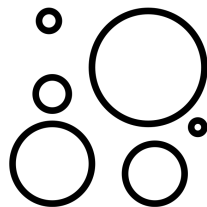
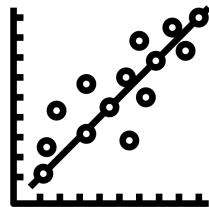
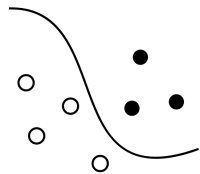
Selecting models with parameter search

Applications: Improved accuracy via parameter tuning
Algorithms: grid search, cross validation, metrics, and more...

Preprocessing

Preprocessing data to prepare for modeling

Applications: Transforming input data such as text for use with machine learning algorithms.
Algorithms: preprocessing, feature extraction, and more...



JupyterLab Exercises

Lab IV. Deep Learning

Deep Learning

by Ian Goodfellow, Yoshua Bengio, and Aaron Courville

<http://www.deeplearningbook.org/>

Animation of Neutron Networks

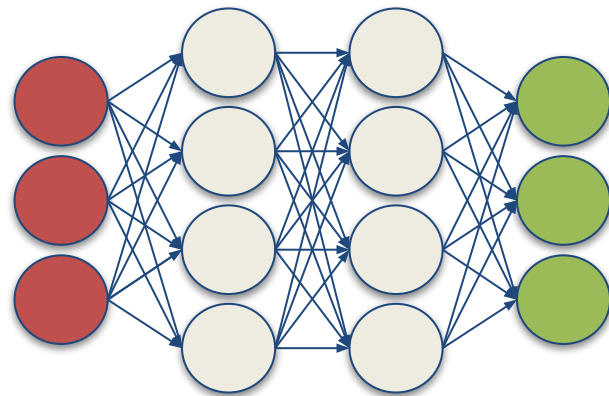
by Grant Sanderson

<https://www.3blue1brown.com/>

Visualization of CNN

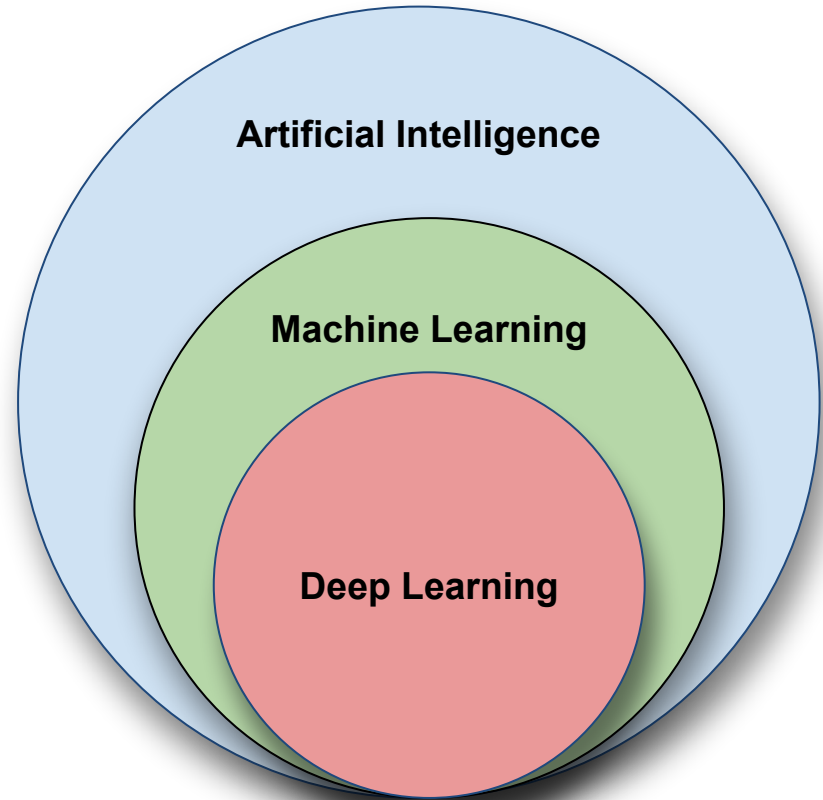
by Adam Harley

https://adamharley.com/nn_vis/cnn/3d.html



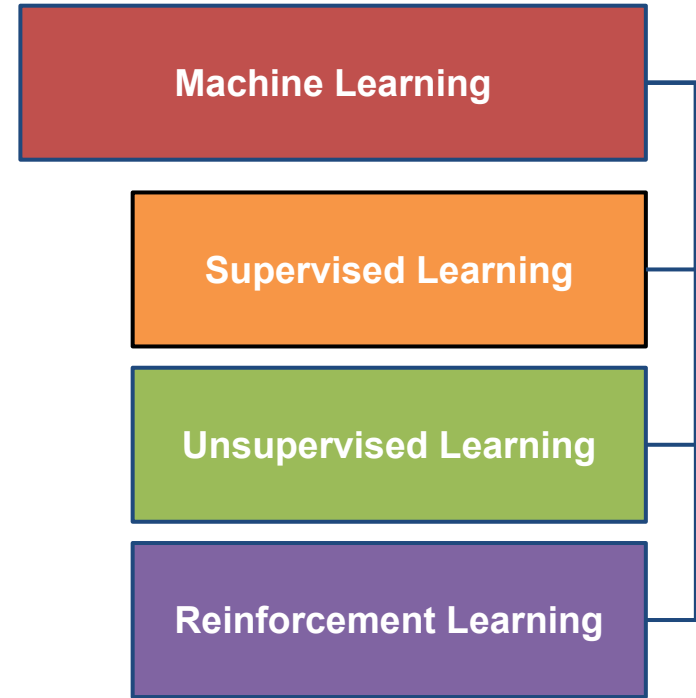
Relationship of AI, ML, and DL

- **Artificial Intelligence (AI)** is anything about man-made intelligence exhibited by machines.
- **Machine Learning (ML)** is an approach to achieve **AI**.
- **Deep Learning (DL)** is one technique to implement **ML**.

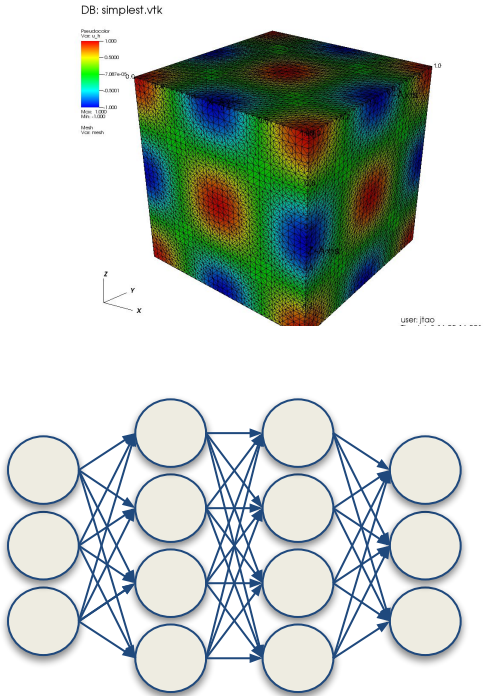


Types of ML Algorithms

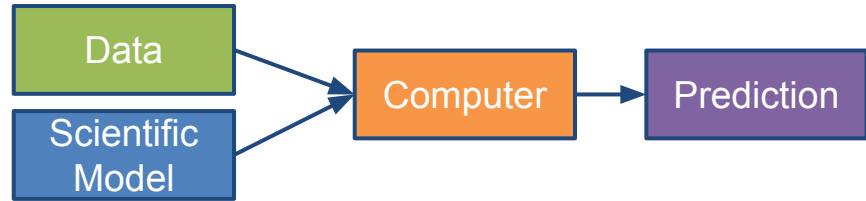
- **Supervised Learning**
 - trained with labeled data; including regression and classification problems
- **Unsupervised Learning**
 - trained with unlabeled data; clustering and association rule learning problems.
- **Reinforcement Learning**
 - no training data; stochastic Markov decision process; robotics and business strategy planning.



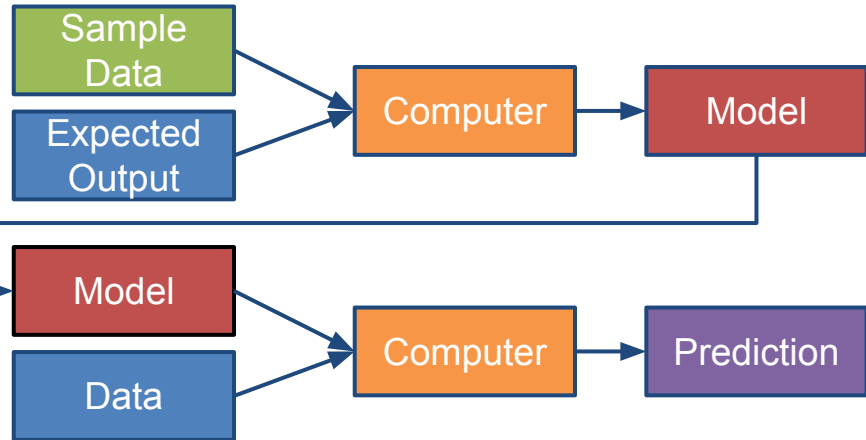
Machine Learning



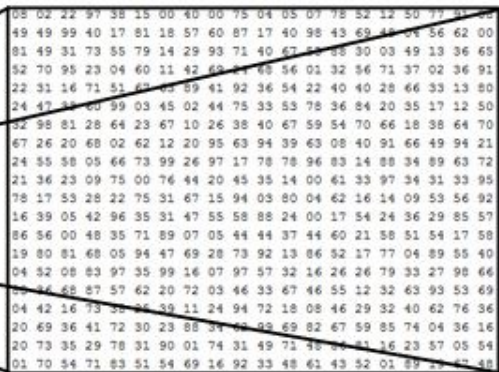
Traditional Modeling



Machine Learning (Supervised Learning)



Inputs and Outputs



What the computer sees

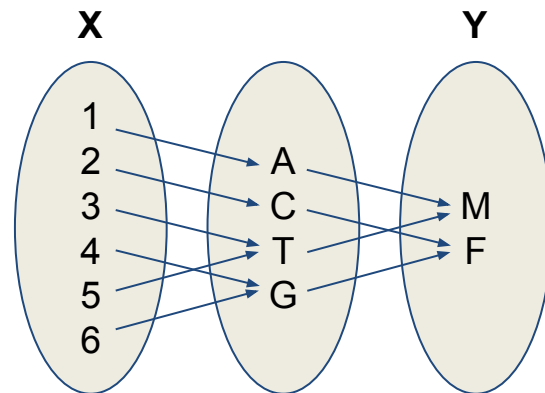
image classification → 82% cat
15% dog
2% hat
1% mug

Image from the [Stanford CS231 Course](#)

256 X 256
Matrix

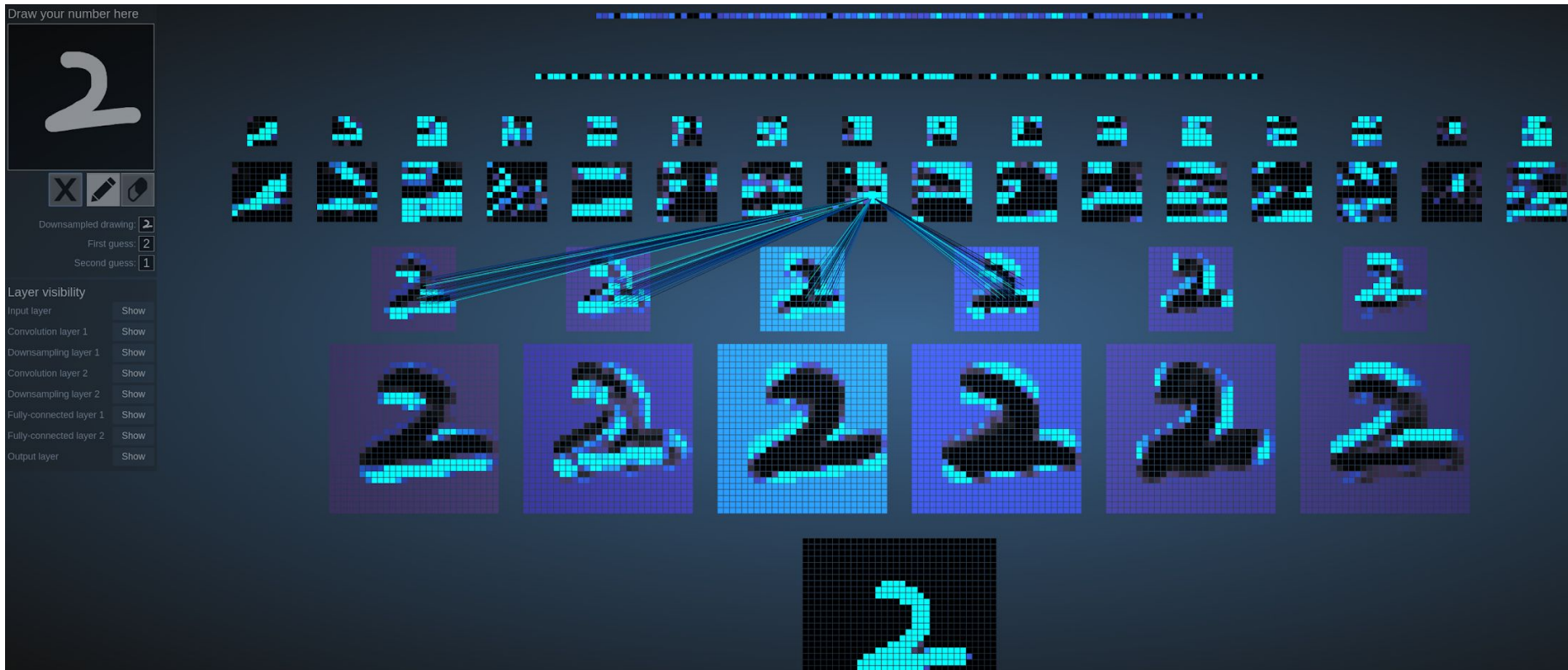
DL model

4-Element Vector



With deep learning, we are searching for a **surjective** (or **onto**) function f from a set X to a set Y .

MNIST - CNN Visualization



(Image Credit: https://adamharley.com/nn_vis/cnn/3d.html)

CNN Explainer

CNN EXPLAINER Learn Convolutional Neural Network (CNN) in your browser!



(Image Credit: <https://poloclub.github.io/cnn-explainer/>)



JupyterLab Exercises



High Performance
Research Computing
DIVISION OF RESEARCH

<https://hprc.tamu.edu>

HPRC Helpdesk:

help@hprc.tamu.edu

Phone: 979-845-0219

Help us help you. Please include details in your request for support, such as, Cluster (Faster, Grace, Terra, ViDaL), NetID (UserID), Job information (Job id(s), Location of your jobfile, input/output files, Application, Module(s) loaded, Error messages, etc), and Steps you have taken, so we can reproduce the problem.