

ACES: AI TechLab in Jupyter Notebooks

Accelerating AI/ML Workflows on a Composable Cyberinfrastructure

Zhenhua He

09/23/2025



High Performance
Research Computing
DIVISION OF RESEARCH



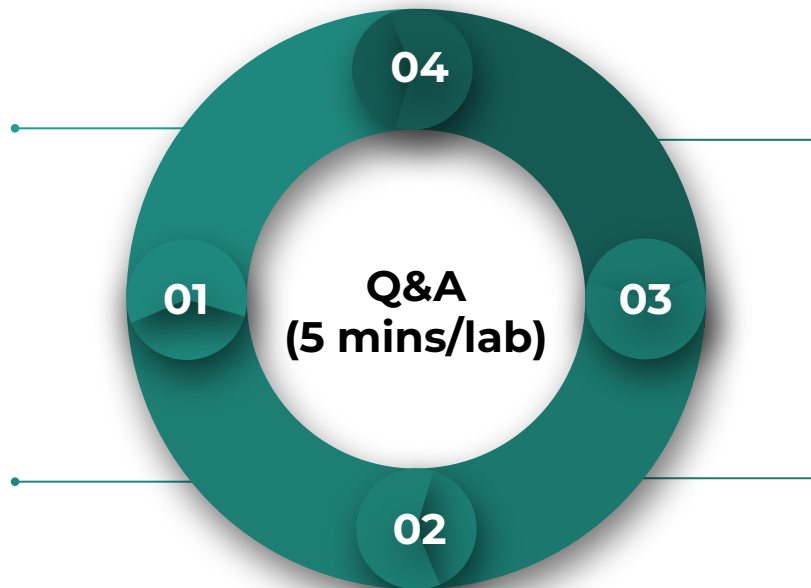
Outline

Lab I. JupyterLab (30 mins)

We will build and activate a ModuLair 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.



Lab IV. Deep Learning (30 minutes)

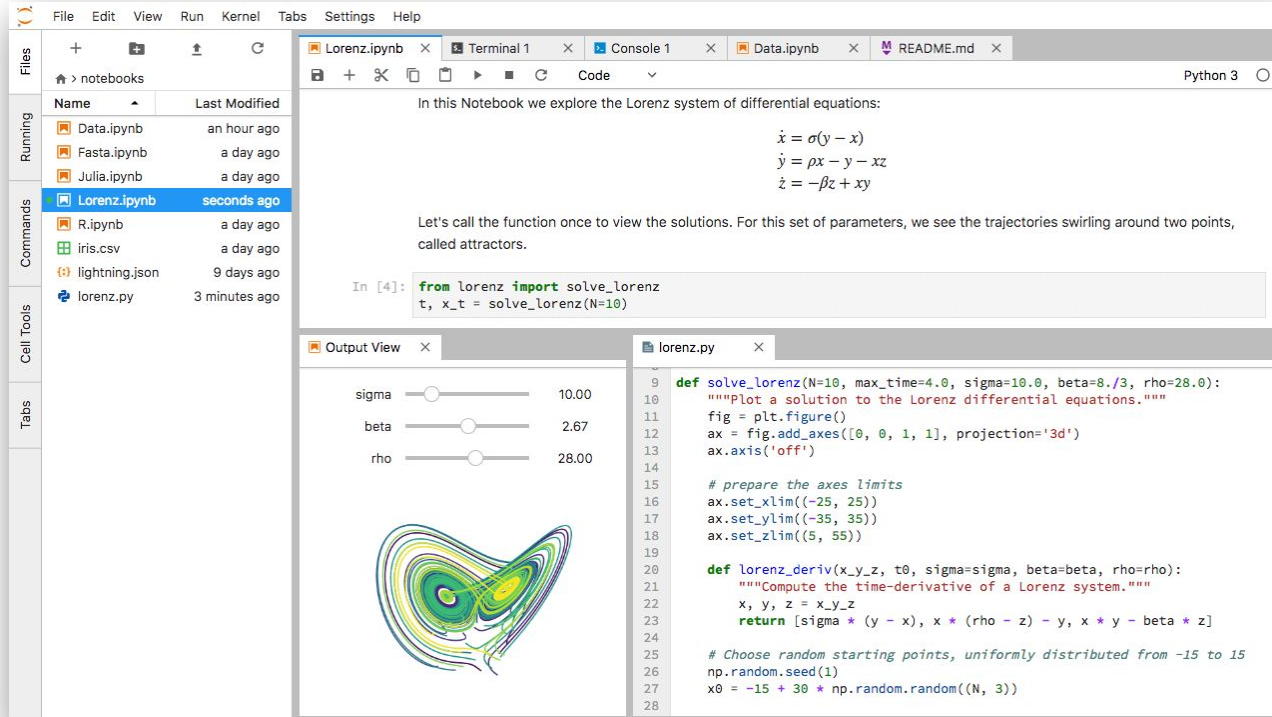
We will learn how to use PyTorch to build and train a simple image classification model with deep neural network (DNN).

Lab III Machine Learning (30 minutes)

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

Figure 1. Structure of the AI TechLab.

Lab I. JupyterLab



The screenshot displays the JupyterLab environment with the following components:

- Files Panel:** A sidebar on the left showing a file browser with notebooks like Data.ipynb, Fasta.ipynb, Julia.ipynb, and the active Lorenz.ipynb.
- Running Panel:** A sidebar showing the status of running notebooks, with Lorenz.ipynb listed as 'seconds ago'.
- Commands Panel:** A sidebar showing available commands, including R.ipynb, iris.csv, lightning.json, and lorenz.py.
- Cell Tools Panel:** A sidebar showing tools for editing cells, including R.ipynb, iris.csv, lightning.json, and lorenz.py.
- Terminal:** A terminal window at the top showing the command 'Code'.
- Console:** A console window at the top showing the command 'Code'.
- Notebook:** The main workspace showing a notebook with the title 'Lorenz.ipynb'. It contains a code cell with the following text:


```
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:** A panel at the bottom showing the output of the code cell. It includes a 3D plot of the Lorenz attractor, which is a complex, swirling shape. The plot is titled 'sigma', 'beta', and 'rho' with sliders for each parameter. The sliders are set to sigma=10.00, beta=2.67, and rho=28.00. Below the sliders is a 3D plot of the Lorenz attractor, showing the trajectories swirling around two points, called attractors.
- lorenz.py:** A file editor showing the source code for the Lorenz system. The code defines a function 'solve_lorenz' and a function 'lorenz_deriv'.

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

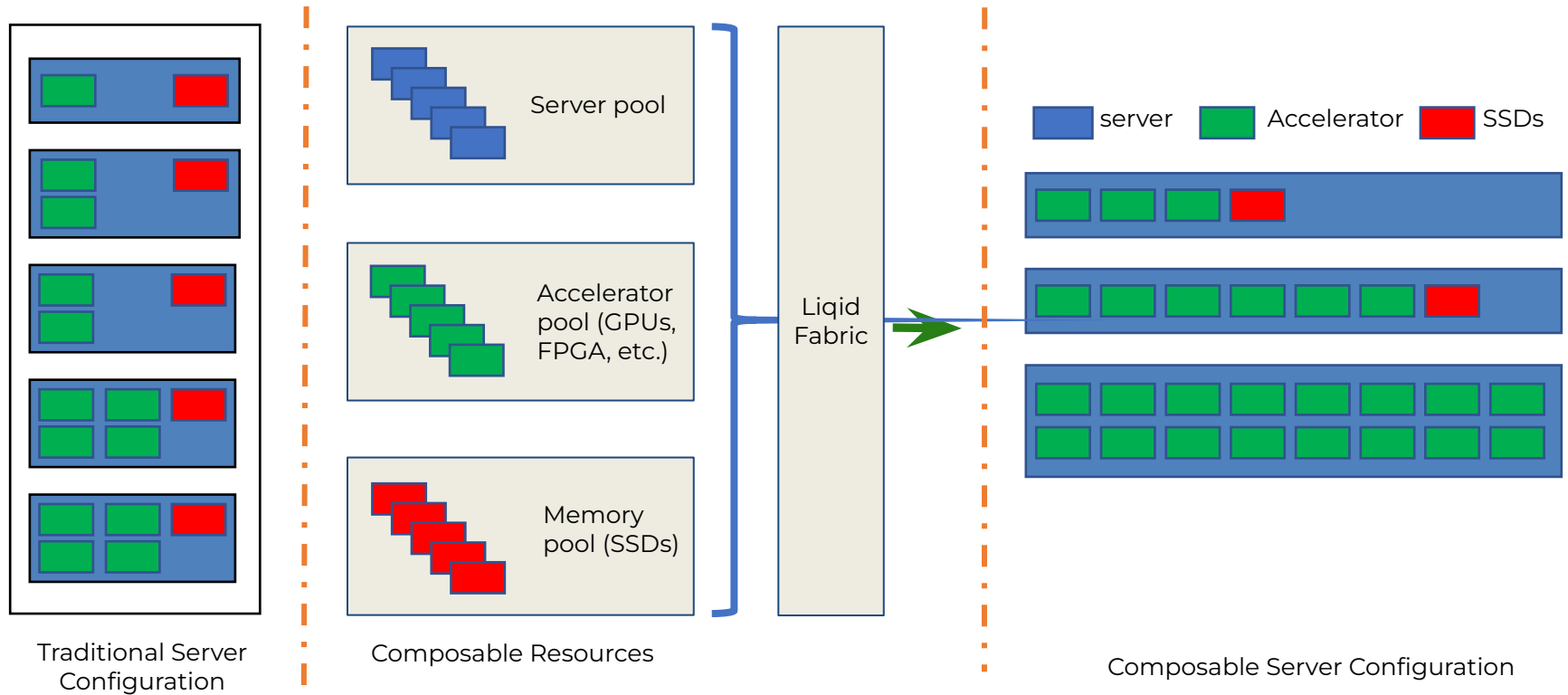
Accelerating Computing for Emerging Sciences

Our Mission:

- Develop an NSF ACSS CI testbed
- 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.



Design: Composability at the Hardware Level



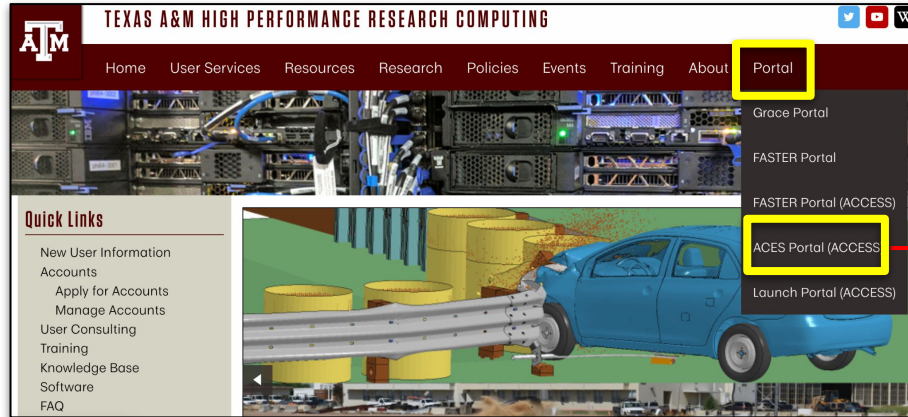
ACES Accelerators/Components

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
<i>FPGAs:</i>		
Intel PAC D5005	2	Accelerator with Intel Stratix 10 GX FPGA and 32 GB DDR4
BittWare IA-840F	3	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 SSDs addressable as memory w/ MemVerge Memory Machine.
<i>NVIDIA GPUs:</i>		
H100	30	For HPC, DL Training, AI Inference
A30	4	For AI Inference and Mainstream Compute
Intel PVC GPUs	120	Intel GPUs for HPC, DL Training, AI Inference

Refer to our **Knowledge Base** for more:

<https://hprc.tamu.edu/kb/User-Guides/ACES/Hardware/>

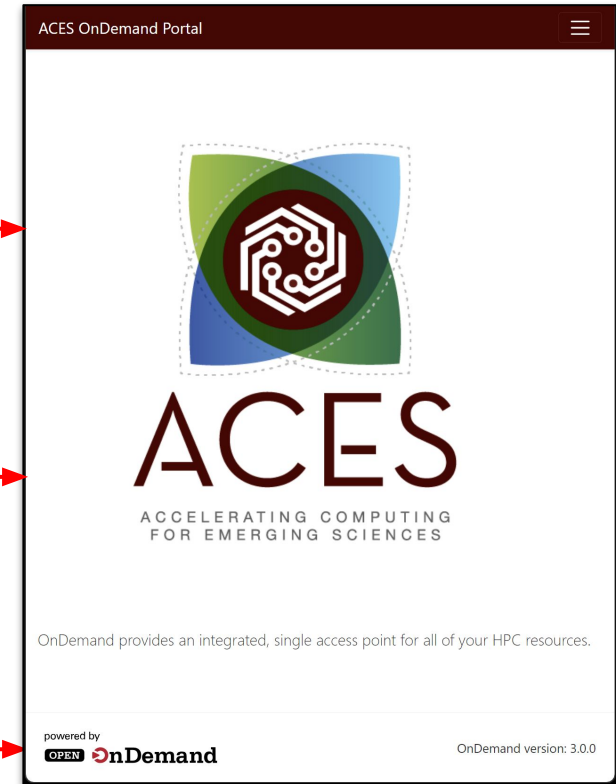
ACES Portal



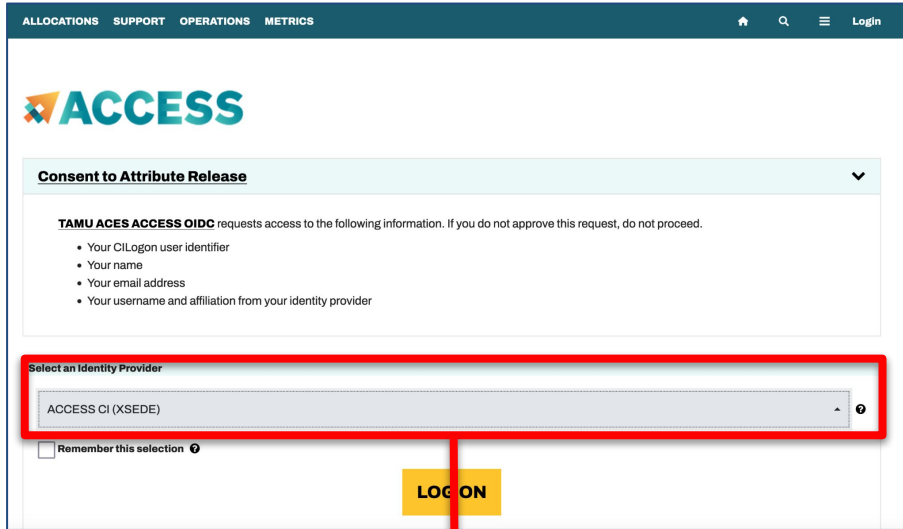
ACES Portal portal-aces.hprc.tamu.edu
is the web-based user interface for the ACES cluster

[HPRC Portal YouTube tutorials](#)

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

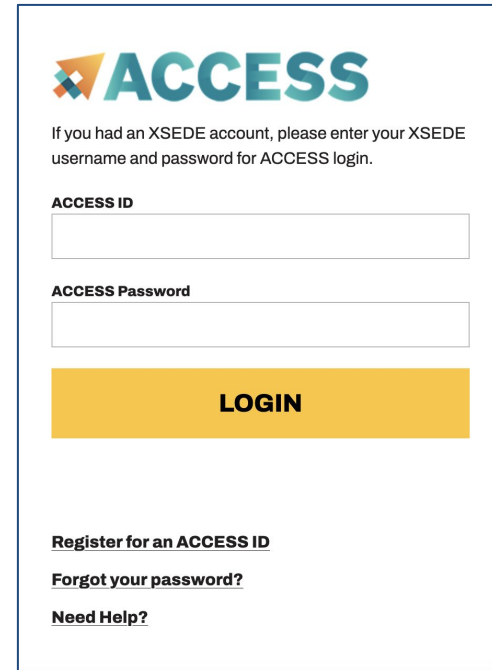


Accessing ACES via the Portal (ACCESS)



The screenshot shows the ACCESS portal interface. At the top is a navigation bar with links: ALLOCATIONS, SUPPORT, OPERATIONS, METRICS, and a Login link. Below the navigation bar is the ACCESS logo. A section titled "Consent to Attribute Release" contains a message from TAMU ACES ACCESS OIDC and a list of requested attributes: CI/Logon user identifier, name, email address, and username/affiliation. Below this is a "Select an Identity Provider" dropdown menu with "ACCESS CI (XSEDE)" selected. A red rectangle highlights this dropdown. Below the dropdown is a "Remember this selection" checkbox and a yellow "LOG ON" button. A red line points from the "LOG ON" button to the text below.

Select the Identity Provider appropriate for your account.

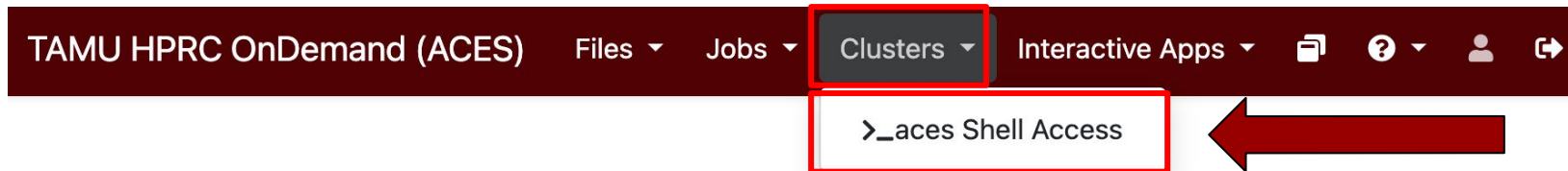


The screenshot shows the ACCESS portal login page. At the top is the ACCESS logo. Below it is a message: "If you had an XSEDE account, please enter your XSEDE username and password for ACCESS login." There are two input fields: "ACCESS ID" and "ACCESS Password". Below these fields is a yellow "LOGIN" button. At the bottom, there are links: "Register for an ACCESS ID", "Forgot your password?", and "Need Help?".

Log in using your ACCESS or institutional credentials.

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
```

(Option 1) Environment Setup: ModuLair

- The ModuLair Framework is designed to streamline the management of Python environments on HPC clusters.

Core Functionalities:

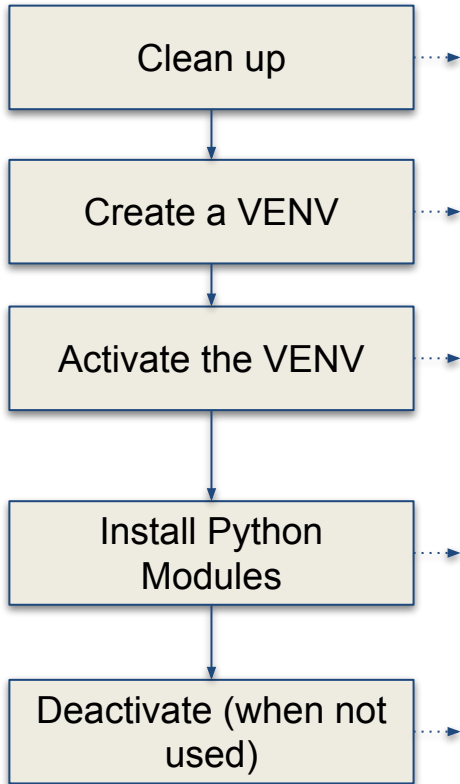
- Creating Environments (`create_venv`)
- Activating Environments (`activate_venv`)
- Listing Environments (`list_envs`)
- Deleting Environments (`delete_venv`)

hprc.tamu.edu/kb/Software/ModuLair

Advantages of ModuLair

- ModuLair builds virtual environments in \$SCRATCH (at \$SCRATCH/virtual_envs) instead of using \$HOME to avoid filling your home directory space quickly.
- ModuLair automatically records the toolchain and Python modules used to create each environment, so you don't need to keep track of them yourself.

ModuLair



```
# clean up
cd $SCRATCH
module purge

# create a ModuLair virtual environment
create_venv ai_labs_env -d "Environment for \
AI TechLab" -t "GCCcore/12.2.0 Python/3.10.8"

# activate the virtual environment
source activate_venv ai_labs_env

# install required packages
pip3 install jupyter
pip3 install pandas matplotlib
pip3 install scikit-learn
pip3 install torch torchvision \
    --index-url https://download.pytorch.org/whl/cu126
pip3 install transformers

# deactivate the virtual environment
# deactivate
```

Go to JupyterLab Page

ACES OnDemand Portal

Files ▾

Jobs ▾

Clusters ▾

Interactive Apps ▾

Affinity Groups ▾

Chatbot ▾

Dashboard ▾

Training ▾

Utilities ▾



OnDemand provides an integrated, sin

Message of the Day

Partial Outage of HPRC Cluste

TAMU Technology Services has scheduled ele Saturday September 20. Some ACES, Launch, maintenance.

There will also be additional electrical mainter down some compute nodes in each cluster du

Job submissions will use the remaining online

IMPORTANT POLICY INFORM

GUI

 NextSilicon VNC

 VNC

Imaging

 ChimeraX

 CryoSPARC

 CryoSPARC 4.2.1

 ImageJ

 Jmol

 Paraview

 cisTEM

Servers

 Jupyter AI Assistant

 Jupyter Notebook

 Jupyter Notebook for Python Training

 JupyterLab

 RStudio

 TensorBoard

sources.

ta Center (WCDC) during 6a-8p
powered down for the electrical

number 28. HPRC will also power

nce windows.

JupyterLab Page

Home / My Interactive Sessions / JupyterLab

Interactive Apps	
GUI	
NextSilicon VNC	
VNC	
Imaging	
ChimeraX	
CryoSPARC	
CryoSPARC 4.2.1	
ImageJ	
Jmol	
Paraview	
cisTEM	
Servers	
Jupyter AI Assistant	
Jupyter Notebook	
Jupyter Notebook	

JupyterLab

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

Type of environment

TAMU ModuLair (Python virtualenv manager)

Select the type of environment in which Jupyter is installed.
[Help me choose](#)

TAMU ModuLair environment (required)

ai_labs_env

Select the name of the TAMU ModuLair environment to be activated.

Your TAMU ModuLair environment is expected to have a Jupyter package installed. Please see [instructions](#)

Node type

CPU only

- [cpuavail](#) [gpuavail](#) select a non-CPU node type only if your software supports the Accelerator

Number of hours (max 72)

Other fields:

- Node Type: CPU
- Number of hours: 3
- Number of cores: 4
- Total memory (GB): 20
- Advanced Params:
Reservation Name: **training**

Option 1: Use own created ModuLair environment

Connect to JupyterLab

[Home](#) / My Interactive Sessions

Interactive Apps

GUI

 NextSilicon VNC

 VNC

Imaging

 ChimeraX

 CryoSPARC

 CryoSPARC 4.2.1

 ImageJ

 Jmol

JupyterLab (1258241)

1 node | 4 cores | Running

Host:  ac098

 Delete

Created at: 2025-09-23 10:04:17 CDT

Time Remaining: 1 hour and 59 minutes

Session ID: [8a71625c-9925-478b-bd4f-9571039efea4](#)

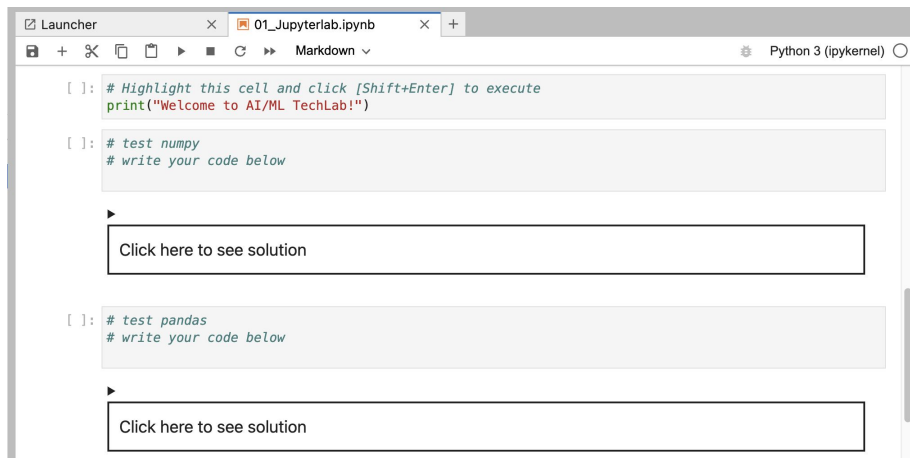
Type of environment: modular

 [Connect to JupyterLab](#)



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.



```
[ ]: # 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) Shared Anaconda Env.

Home / My Interactive Sessions / JupyterLab

Interactive Apps	
GUI	
NextSilicon VNC	
VNC	
Imaging	
ChimeraX	
CryoSPARC	
CryoSPARC 4.2.1	
ImageJ	
Jmol	
Paraview	
cisTEM	
Servers	
Jupyter AI Assistant	
Jupyter Notebook	
Jupyter Notebook	

JupyterLab

This app will launch a JupyterLab server on the ACES cluster.

Type of environment

Anaconda environment

Select the type of environment in which Jupyter is installed.
[Help me choose](#)

Conda/Mamba module to be loaded

Anaconda3/2024.02-1

Select a Conda/Mamba module to load.

Optional conda/mamba environment to be activated

/sw/hprc/sw/Anaconda3/2024.02-1/envs/ai-labs

Enter the name of the conda environment to be activated. This field is optional.

Leave this field blank to use the default conda/mamba environment, which has built-in Python and Jupyter

Your optional environment must have been previously built with conda/mamba, and is expected to have the "jupyter"

Other fields:

- Node Type: CPU
- Number of hours: 3
- Number of cores: 4
- Total memory (GB): 20
- Advanced Params:
Reservation Name: **training**

Option 2: Use a shared environment created by TAMU HPRC for this course

Path to the shared environment:

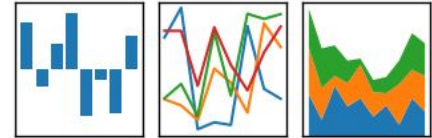
/sw/hprc/sw/Anaconda3/2024.02-1/envs/ai-labs

Lab II. Data Exploration

matplotlib

pandas

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



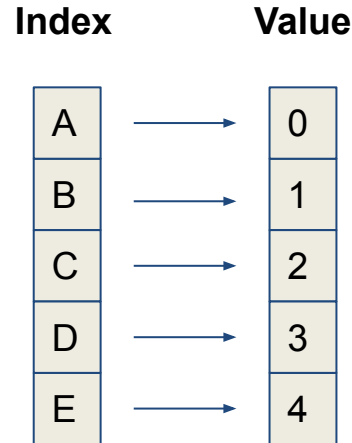
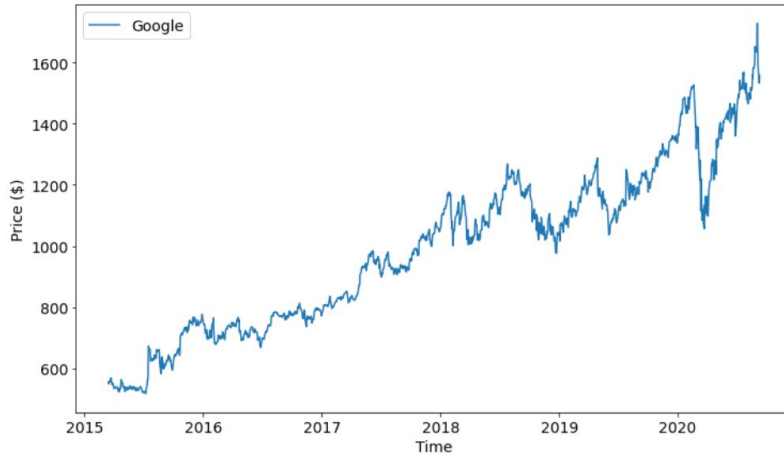
Data Structures

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

		Columns			
Index		C1	C2	C3	C4
A	→	0	x	0.1	True
B	→	1	y	2.4	False
C	→	2	z	1.9	True
D	→	NA	w	8.3	False
E	→	9	a	6.8	False

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**

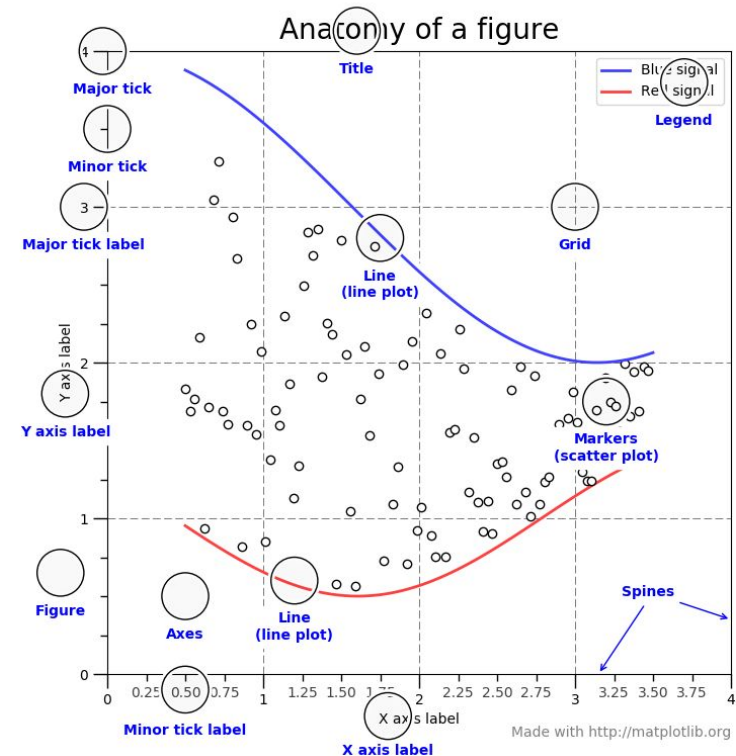
Figure is the object that keeps the whole image output. Adjustable parameters include:

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

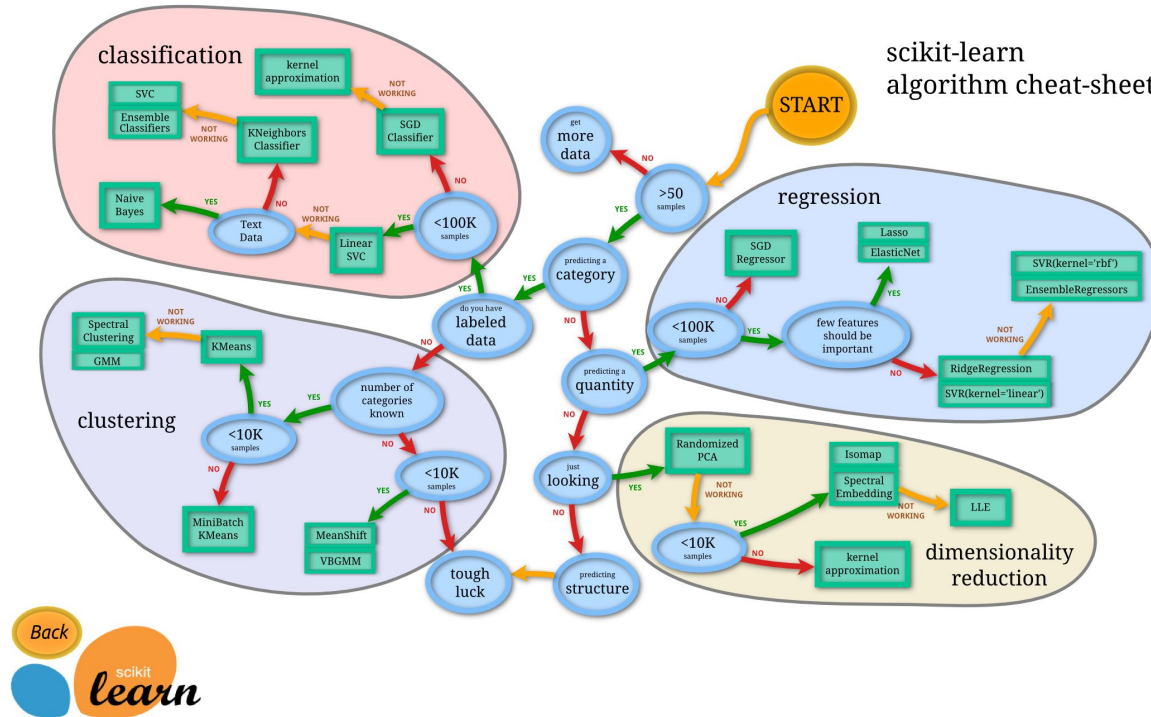
After this lesson, you will know how to create:

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



JupyterLab Exercises

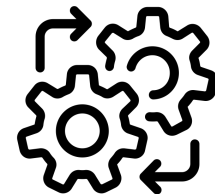
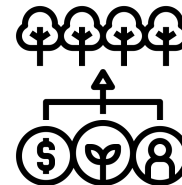
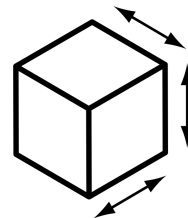
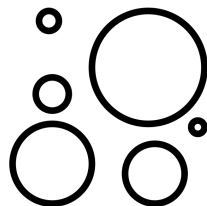
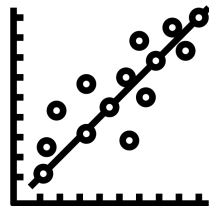
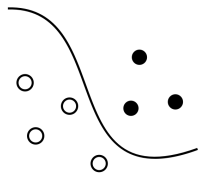
Lab III. Machine Learning



Main Features of scikit-learn



Classification	Regression	Clustering	Dimension Reduction	Model Selection	Preprocessing
Identifying category of an object Applications: Spam detection, image recognition. Algorithms: SVM, nearest neighbors, random forest, and more...	Predicting a attribute for an object Applications: Drug response, Stock prices. Algorithms: SVR, nearest neighbors, random forest, and more...	Grouping similar objects into sets Applications: Customer segmentation, Grouping experiment outcomes Algorithms: k-Means, spectral clustering, mean-shift, and more...	Reducing the number of dimensions Applications: Visualization, Increased efficiency Algorithms: k-Means, feature selection, non-negative matrix factorization, and more...	Selecting models with parameter search Applications: Improved accuracy via parameter tuning Algorithms: grid search, cross validation, metrics, and more...	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

Credit: icons are from [The Noun Project](https://thenounproject.com/) under Creative Commons Licenses

Lab IV. Deep Learning

Deep Learning

by Ian Goodfellow, Yoshua Bengio, and Aaron Courville

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

Animation of Neutron Networks

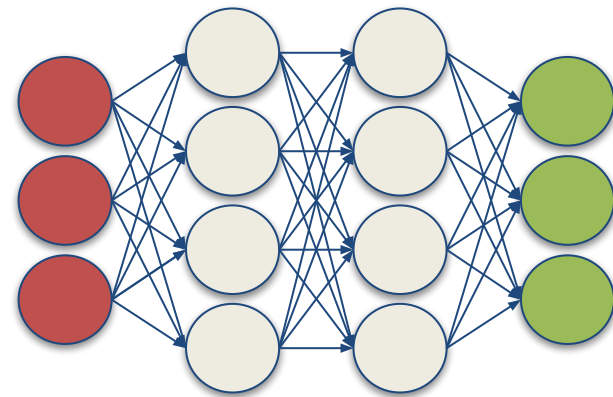
by Grant Sanderson

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

Visualization of CNN

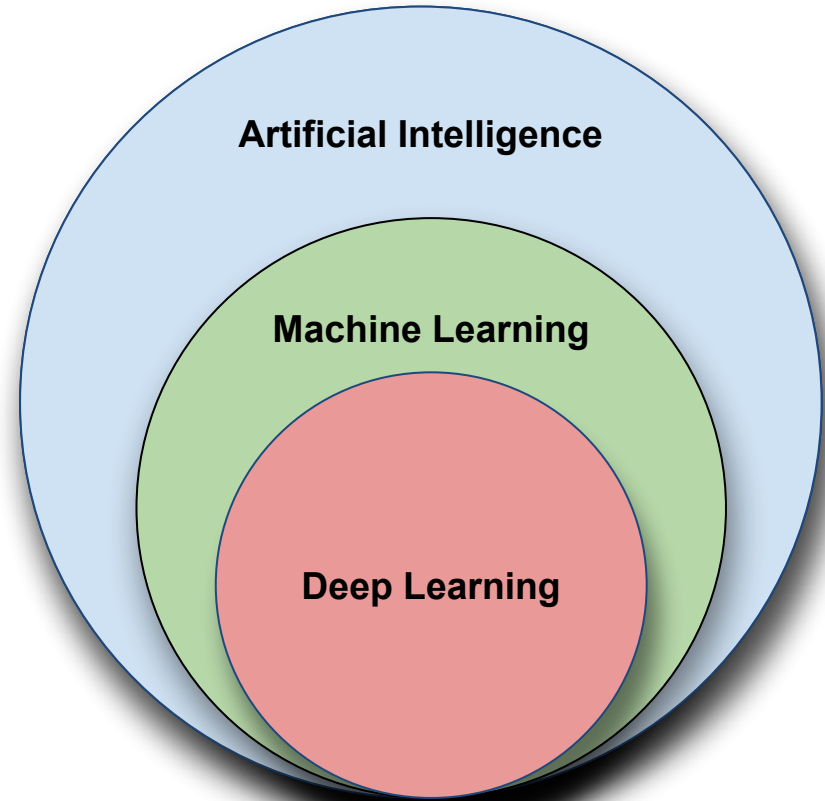
by Adam Harley

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



Introduction to AI/ML/DL

- **AI (Artificial Intelligence):**
Machines simulating human intelligence
- **ML (Machine Learning):**
A method to enable AI through data learning
- **DL (Deep Learning):**
A type of ML using layered neural networks



Types of ML Algorithms

- **Supervised Learning**

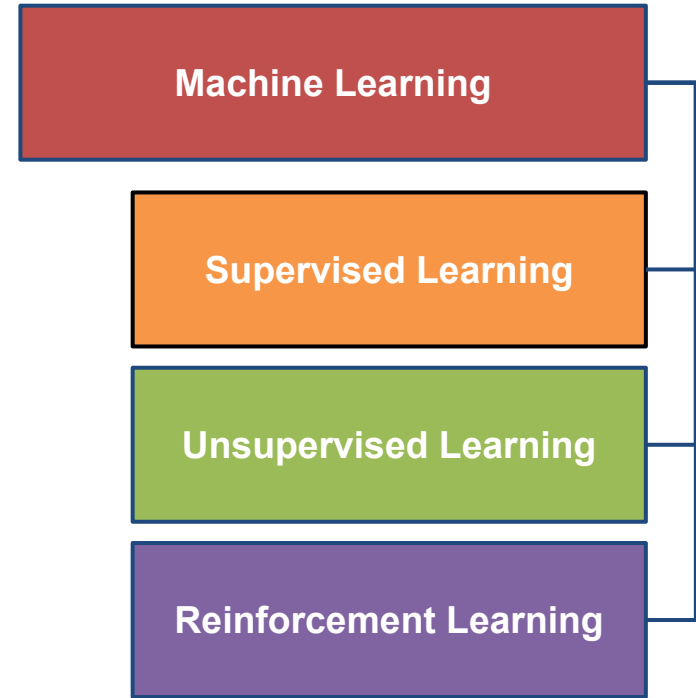
- trained with labeled data; including regression and classification problems

- **Unsupervised Learning**

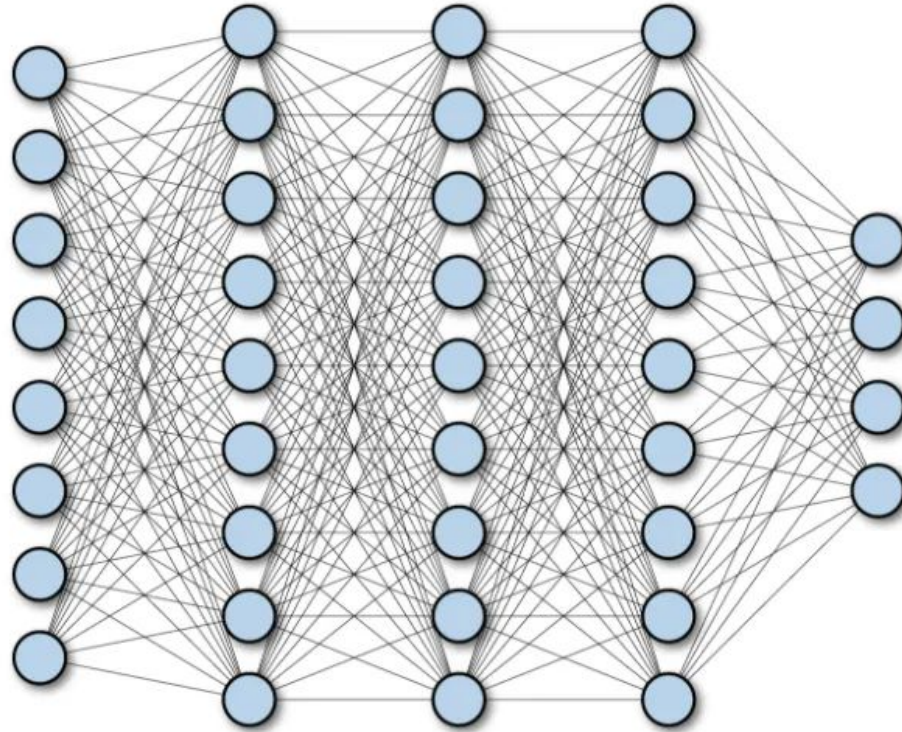
- trained with unlabeled data; clustering and association rule learning problems.

- **Reinforcement Learning**

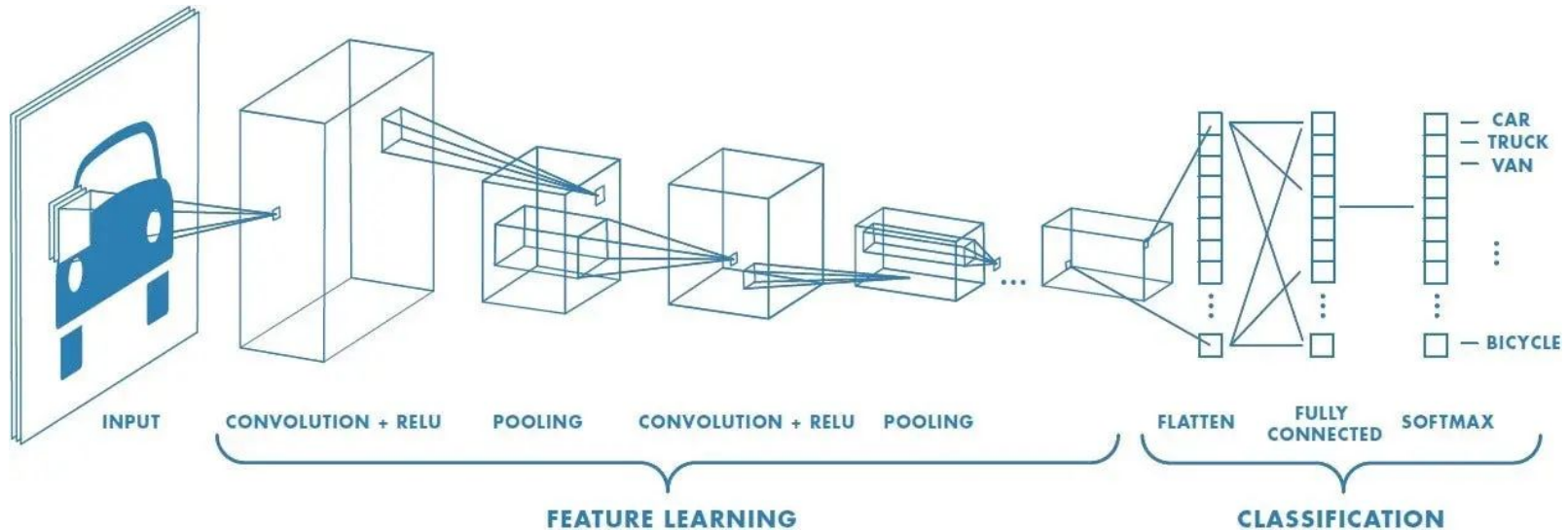
- no training data; stochastic Markov decision process; robotics and business strategy planning.



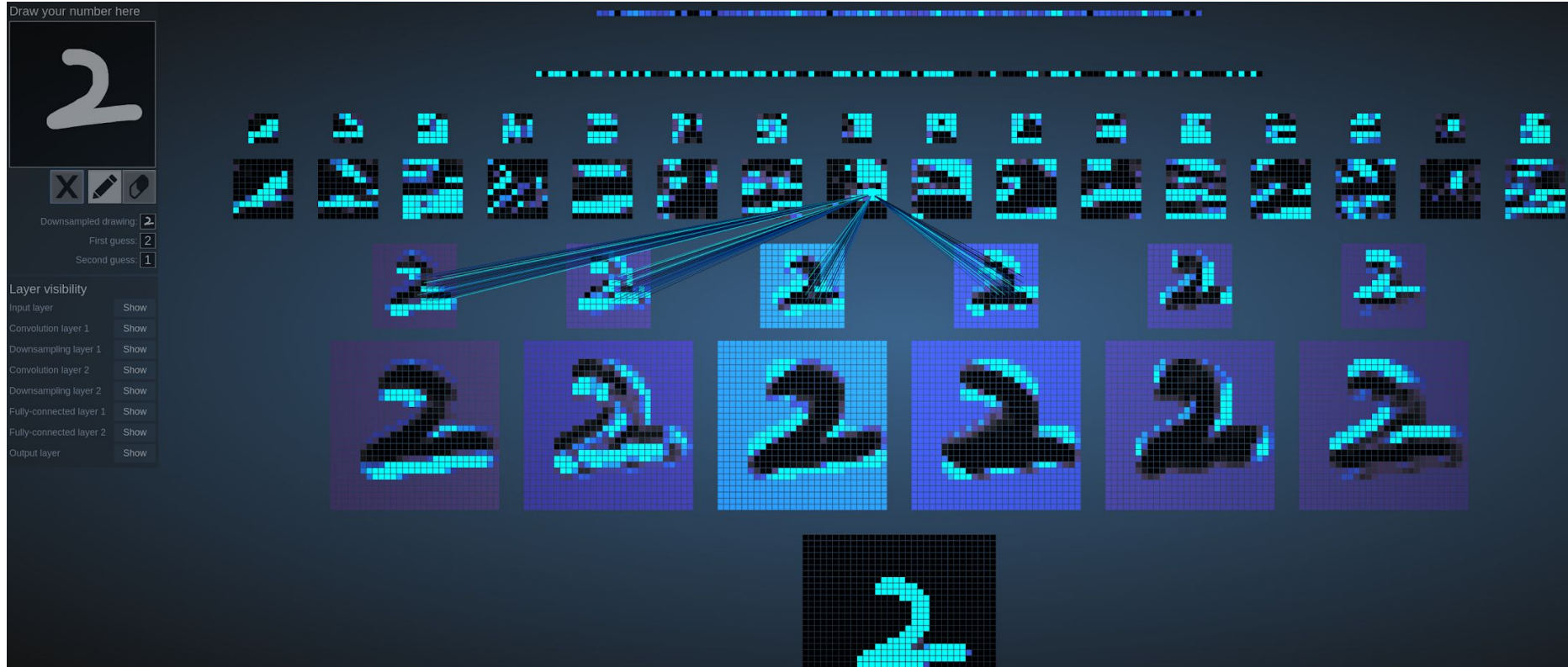
Fully Connected Neural Network



Convolutional Neural Network (CNN)



MNIST - Convolutional Neural Network Visualization



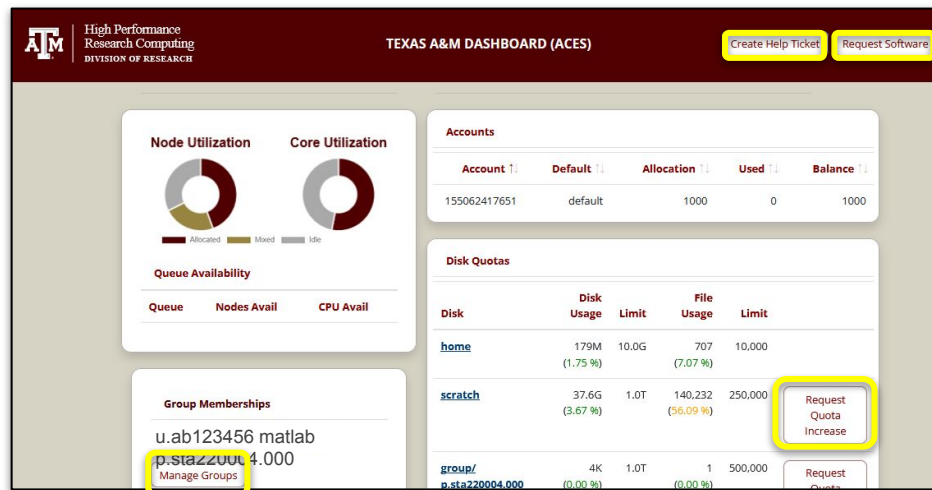
(Image Credit: https://adamharley.com/nn_vis/cnn/3d.html)

Need Help?

First check the FAQ: <https://hprc.tamu.edu/kb/FAQ/Accounts>

- ACES user Guide: <https://hprc.tamu.edu/kb/User-Guides/ACES>
- Email your questions to help@hprc.tamu.edu

Remember the
Dashboard!



Need Help?

Help us help you -- tell us:

- Which cluster
- Username
- Job id(s) if any
- Location of your jobfile, input/output files
- Application used if any
- Module(s) loaded if any
- Error messages
- Steps you have taken, so we can reproduce the problem



**HIGH PERFORMANCE
RESEARCH COMPUTING**
TEXAS A&M UNIVERSITY

<https://hprc.tamu.edu>

HPRC Helpdesk:

help@hprc.tamu.edu

Phone: 979-845-0219

Take our short course survey!



HPRC Survey

https://u.tamu.edu/hprc_shortcourse_survey

Help us help you. Please include details in your request for support, such as, Cluster (ACES, FASTER, Grace, Launch), NetID (UserID), Job information (JobID(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.

