GW SCHOOL OF ENGINEERING & APPLIED SCIENCE

Software Tools for Machine Learning & Deep Learning

Shuyue (Frank) Guan

The Medical Imaging & Image Analysis (MIA) Laboratory

Sep. 2021

Hardware

- CPU
  - Speed (GHz) – General ML
  - # of Cores – Parallel computing
- RAM – General ML
  - Size (GB) – Image Processing
  - Speed (MHz) – Loading speed
- GPU – DL
  - Buffer (Memory) – **Crucial**: model scale, input/batch size ($\geq$**6GB**)
  - CUDA Cores – Speed

GEFORCE GTX 1080 Ti

GPU Engine Specs:

NVIDIA CUDA® Cores	3584
Boost Clock (MHz)	1582

Memory Specs:

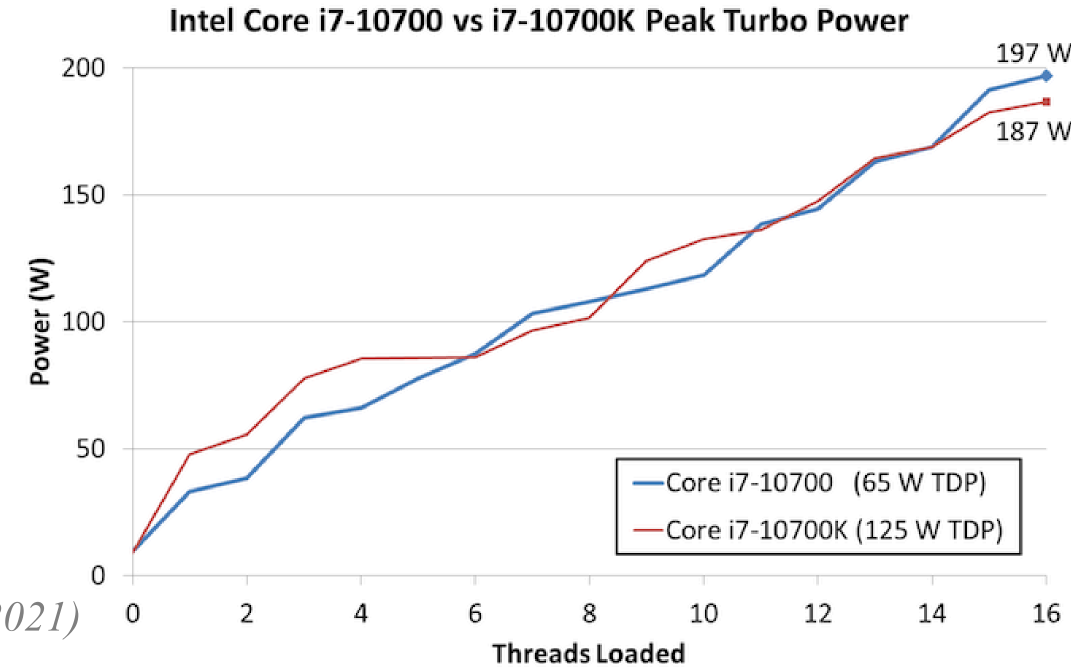
Memory Speed	11 Gbps
Standard Memory Config	11 GB GDDR5X
Memory Interface Width	352-bit

Out of memory (OOM) issues

Hardware (Cont.)

- Solid-state disk (SSD) – read/write data fast
  - Speed (MB/s)
  - Size (GB) – by needs
  - **Tips: SSD (512GB) + HDD (2TB)*

- Cooling system, power supply
  - Multi-CPU
  - Multi-GPU
  - High-speed RAM/SSD

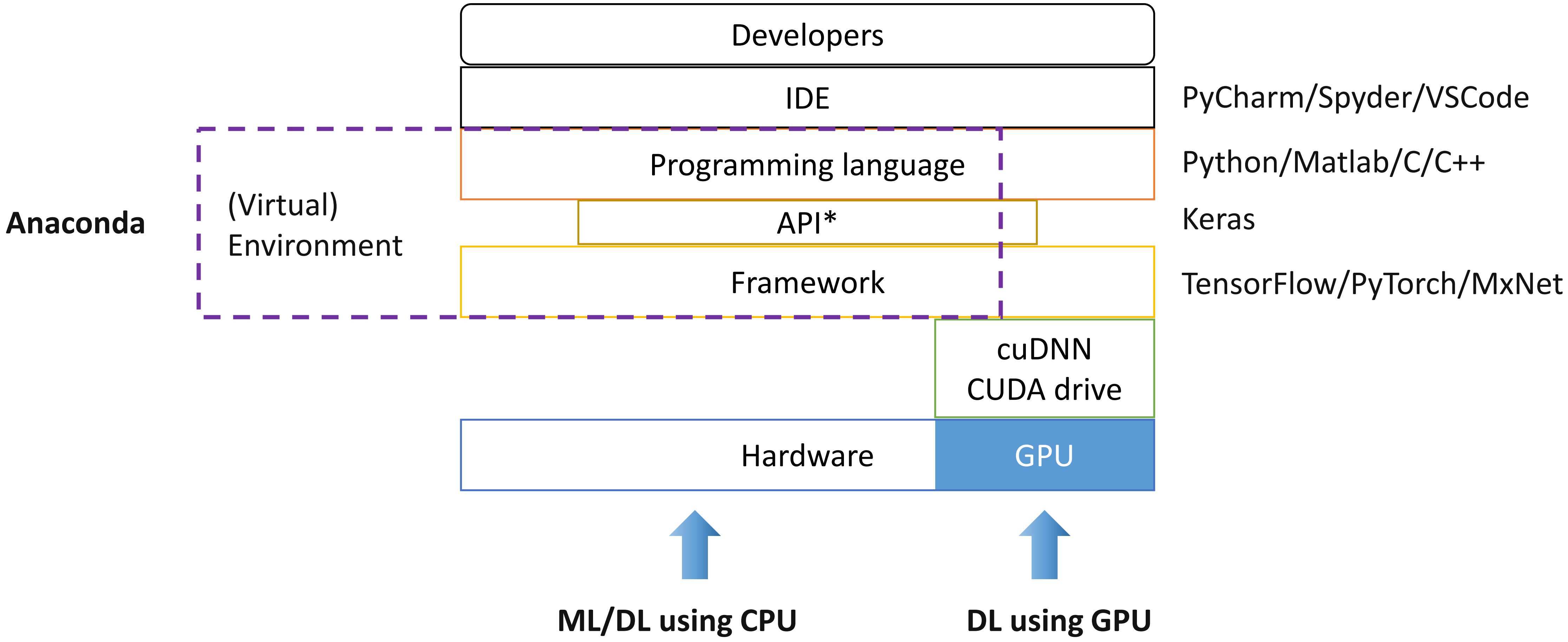


(Ian Cutress, 2021)

Hardware: Tips

- Develop/test on PC, running on workstation
  - Smaller input/batch sizes
  - Smaller model scales
  - Smaller loops
  - …
- HPC at GWU (*discuss details later*)
- Cloud Computing resources
  - AWS
  - Google Cloud
  - …

Software overview



Languages for general ML

- MATLAB
  - **Advantages**: quick to get started and easy to use; visualization; detailed official help documentation; high credibility
  - **Disadvantages**: need to buy; slow official updates; lower flexibility; not many users in the DL field
- Python
  - **Advantages**: free; fast community-based update; higher flexibility; widely used in the NLP/DL field; rich packages
  - **Disadvantages**: difficult to get started; not intuitive enough; mixed help information; may not reliable

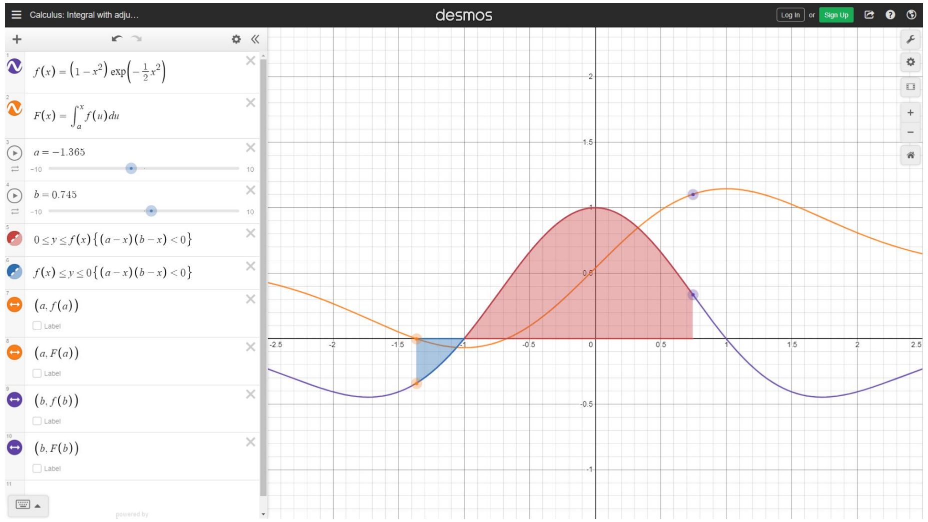
Languages/tools for general ML

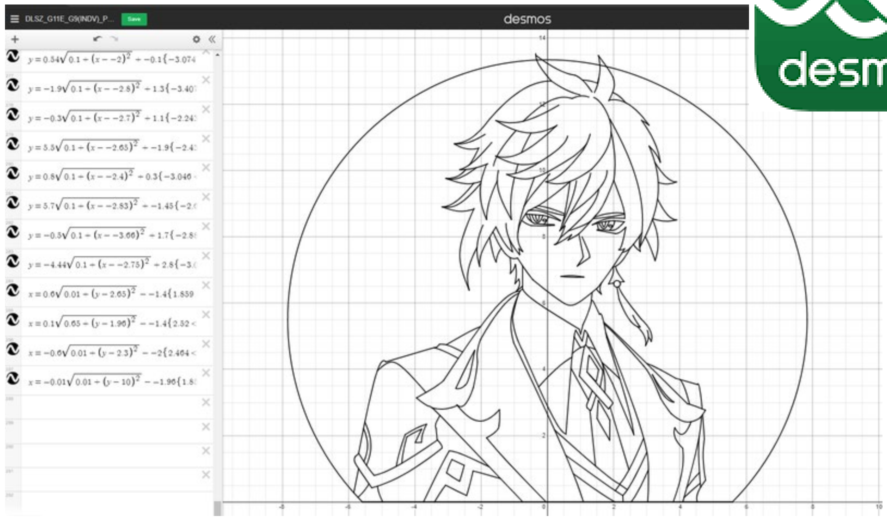
- Other tools
  - **Wolfram Mathematica**: symbolic & numeric computations

In[3]:= DSolve[y''[x] + y[x] == Exp[x], y, x]

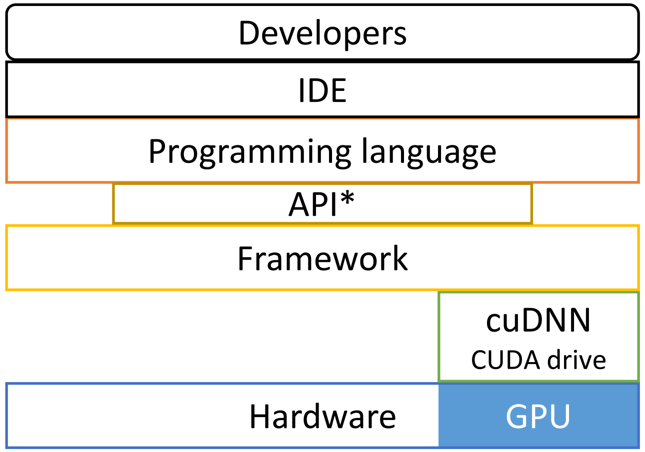
Out[3]= $\{\{y \to \text{Function}[\{x\}, c_1 \text{Cos}[x] + c_2 \text{Sin}[x] + \frac{1}{2} e^x (\text{Cos}[x]^2 + \text{Sin}[x]^2)]\}\}$

  - **Desmos**: graphing calculator, FREE





Tools for DL



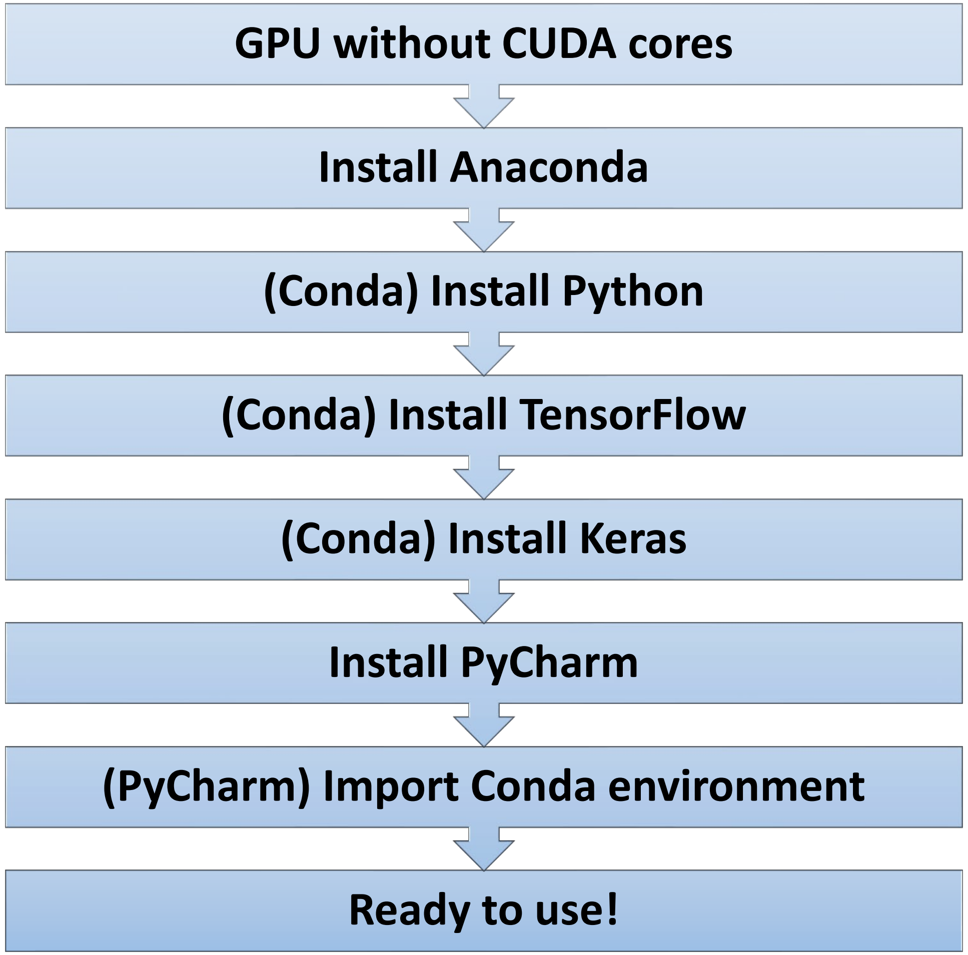
Examples:

- TensorFlow + Keras + Python + PyCharm
- PyTorch + Python + VSCode/PyCharm

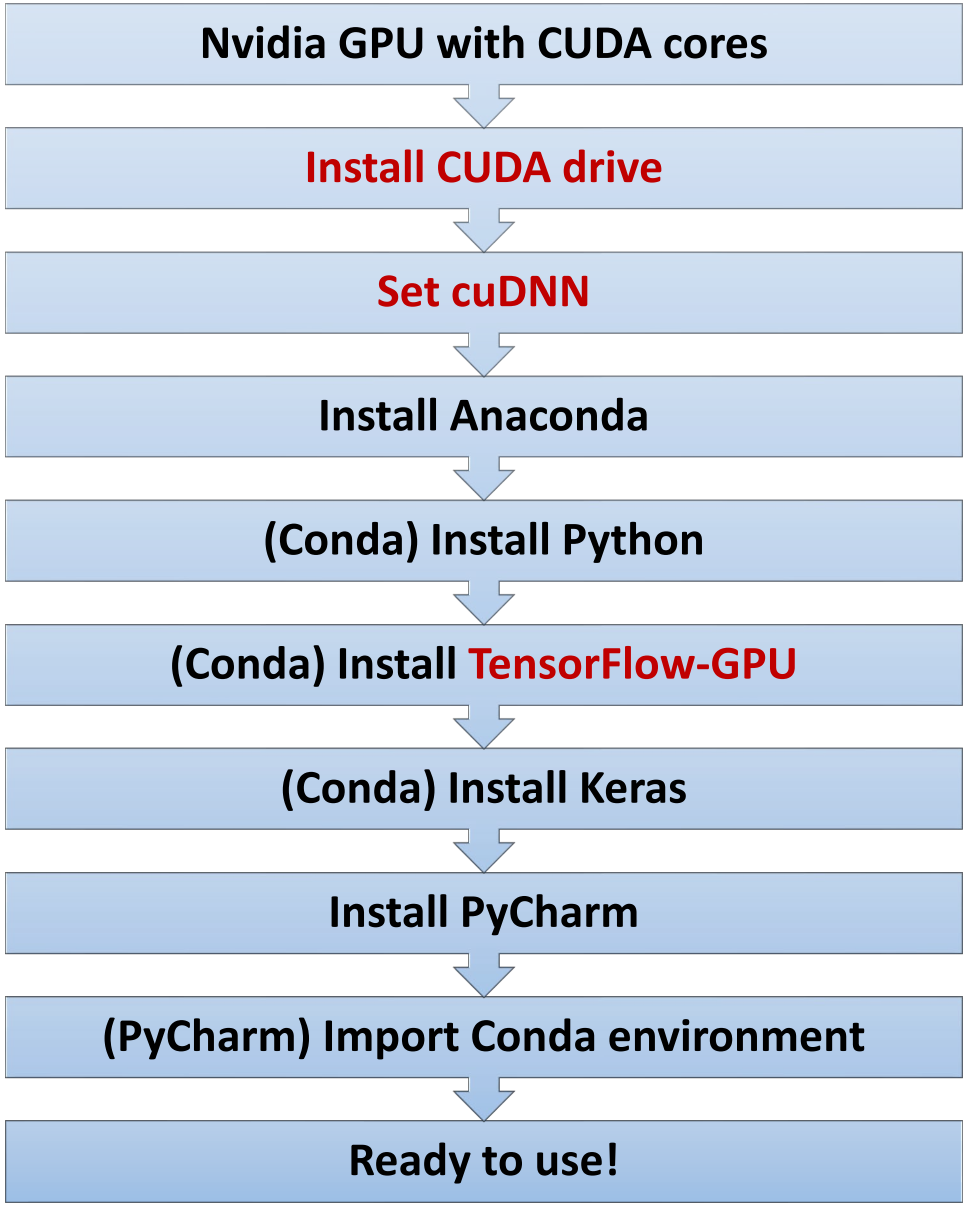
Recommend: **Anaconda**

- **Virtual** environments for development
- Includes/controls: Frameworks + APIs + Languages

Configuration



DL using CPU



DL using GPU

Version problems

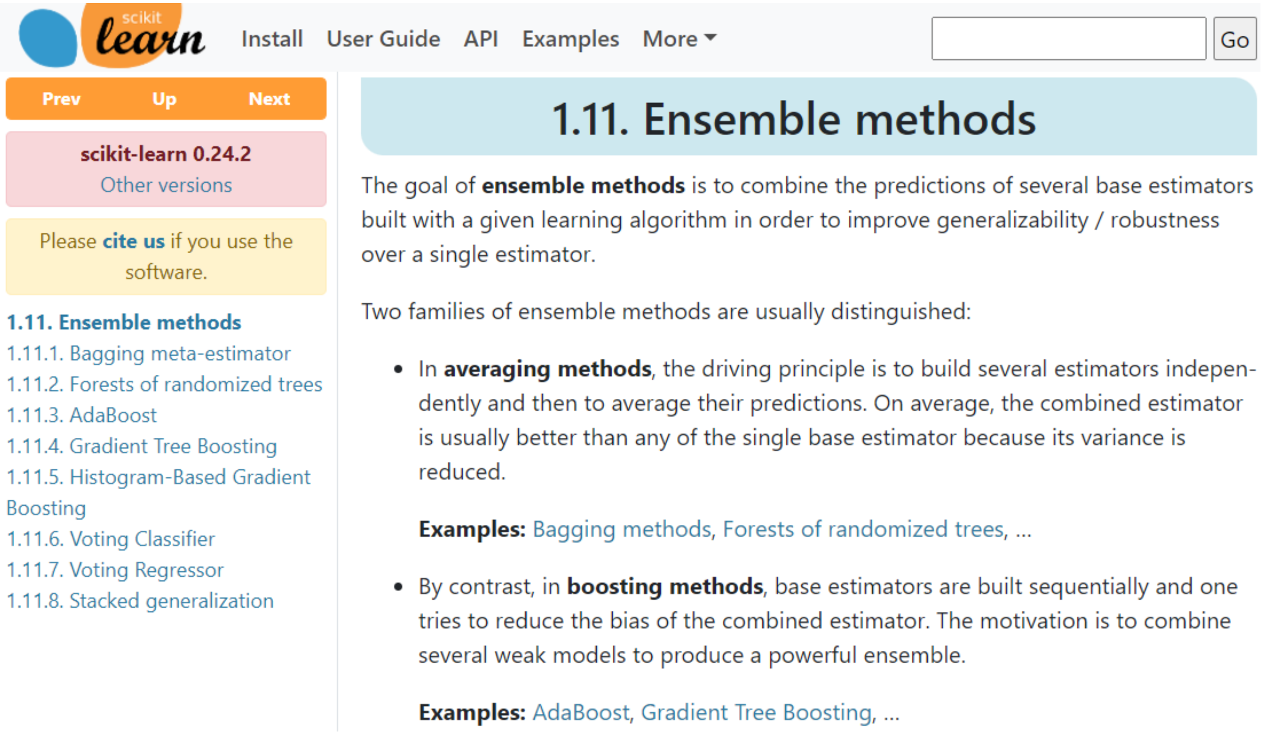
- OS (Win, Linux, macOS)
- CPU/GPU
- CUDA, cuDNN, framework, programming language,…

Version	Python version	cuDNN	CUDA
tensorflow_gpu-2.3.0	3.5-3.8	7.6	10.1
tensorflow_gpu-2.2.0	3.5-3.8	7.6	10.1
tensorflow_gpu-2.1.0	3.5-3.7	7.6	10.1
tensorflow_gpu-2.0.0	3.5-3.7	7.4	10
tensorflow_gpu-1.15.0	3.5-3.7	7.4	10
tensorflow_gpu-1.14.0	3.5-3.7	7.4	10
tensorflow_gpu-1.13.0	3.5-3.7	7.4	10
tensorflow_gpu-1.12.0	3.5-3.6	7	9
tensorflow_gpu-1.11.0	3.5-3.6	7	9
tensorflow_gpu-1.10.0	3.5-3.6	7	9
tensorflow_gpu-1.9.0	3.5-3.6	7	9
tensorflow_gpu-1.8.0	3.5-3.6	7	9
tensorflow_gpu-1.7.0	3.5-3.6	7	9
tensorflow_gpu-1.6.0	3.5-3.6	7	9
tensorflow_gpu-1.5.0	3.5-3.6	7	9
tensorflow_gpu-1.4.0	3.5-3.6	6	8
tensorflow_gpu-1.3.0	3.5-3.6	6	8

Windows GPU

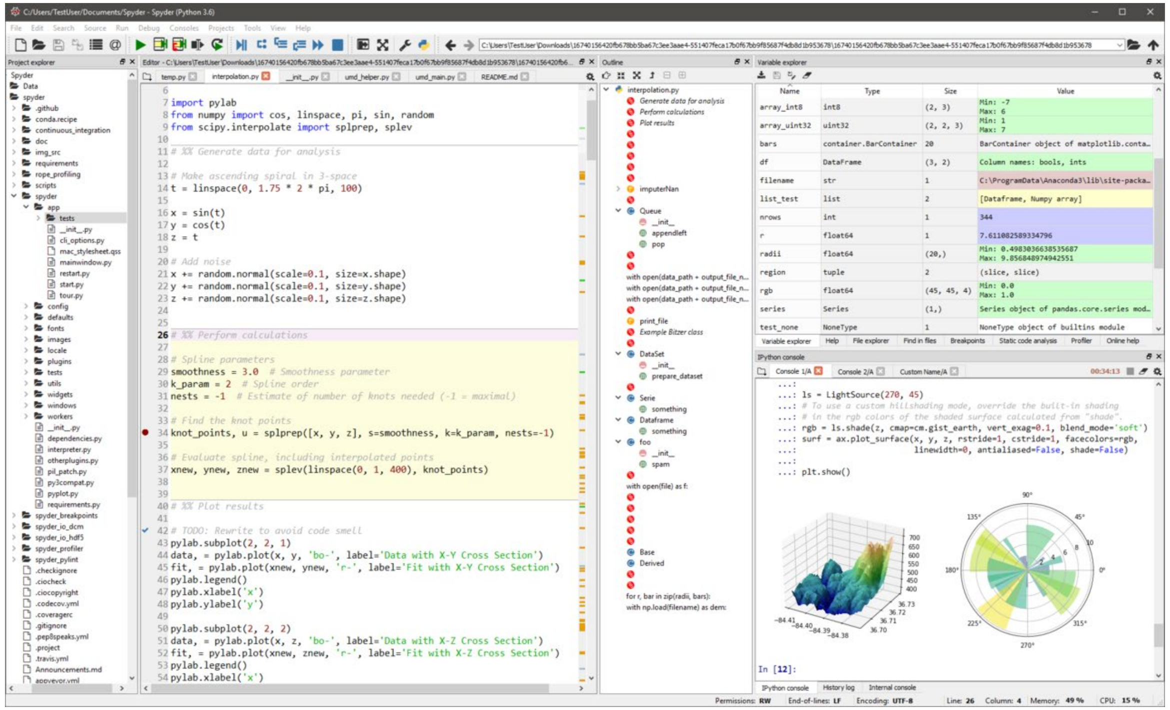
Packages for ML/DL in Python

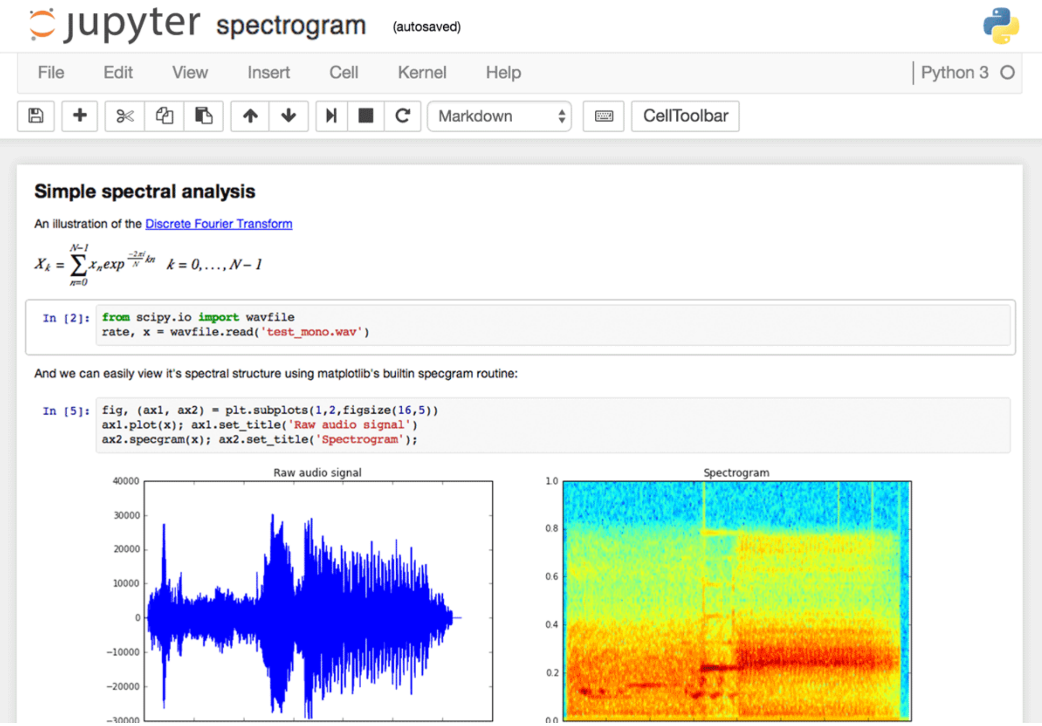
- **NumPy**: basic functions for math and matrix
- **SciPy**: scientific computing
- **Scikit-learn**: machine learning library
- **Matplotlib**: plotting library
- **OpenCV**: computer vision
- **Pandas**: data manipulation and analysis
- …



Integrated development environment (IDE)

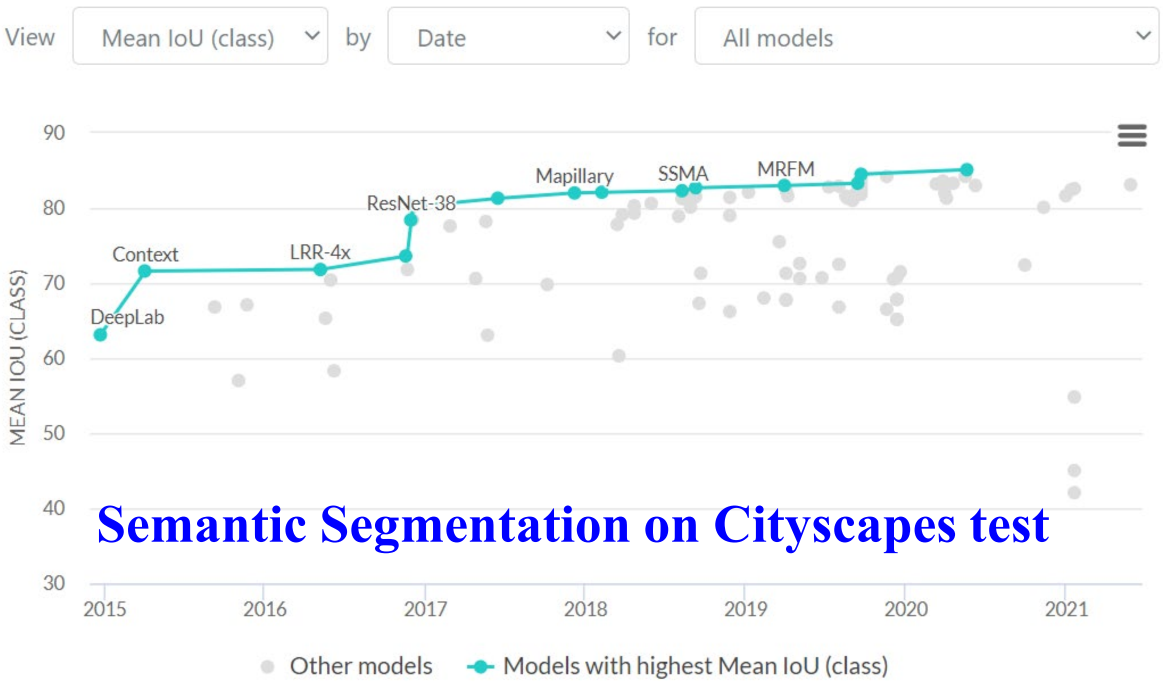
- **PyCharm**: comprehensive IDE, for large projects
- **VSCode**: light-weight IDE, support many languages
- **Jupyter Notebook**: interactive IDE, publish friendly
- **Spyder**: MATLAB-like, Anaconda built-in
- …





Websites for ML/DL

- Guidebooks of each tools/software online – look-up books
- Github – without reinventing the wheel
- **Stack Overflow** – solve problems
- Kaggle – datasets
- **paperswithcode.com** – SOTA performance ranks with papers & codes
- Google – everything!
- …



Semantic Segmentation on Cityscapes test

Rank	Model	Mean IoU (class)	Category mIoU	Extra Training Data	Paper	Code	Result	Year
1	**HRNet-OCR** (Hierarchical Multi-scale Attention)	85.1%		✓	Hierarchical Multi-Scale Attention for Semantic Segmentation			2020
2	**HRNetV2 + OCR +**	84.5%		✓	Segmentation Transformer: Object-Contextual Representations for Semantic Segmentation			2019
3	**EfficientPS**	84.21%		✓	EfficientPS: Efficient Panoptic Segmentation			2020

HPC at GW

- One CPU node
  - Dual 20-Core 3.70GHz Intel Xeon processors
  - 192GB RAM
  - 800 GB SSD
- One GPU node
  - 2 NVIDIA **Tesla V100** GPU (4 for large nodes)
  - Dual 20-Core 3.70GHz Intel Xeon (18-Core Xeon for large nodes)
  - 192GB RAM (384GB for large nodes)
  - 800 GB SSD
- High throughput node, High memory node (3TB RAM!),...

Tesla V100 highlight

640	**Tensor Cores**
5120	**CUDA Cores**
16 GB	**Memory**

How to use the HPC at GW

Overview

1. Request for an account
2. Transfer data & code - Globus
3. VPN Connection (Off-campus) - Cisco AnyConnect
4. Login by **SSH client** - MobaXterm
5. Management by Linux commends
6. Jobs deployment/running by **SLURM**

Workflow

1. Upload data and code files
2. Create a **SLURM script** to assign the job
3. Run **the script**
4. Outputs will be saved
5. Receive states and notifications by Email or check by commands

Epilogue

- The Medical Imaging & Image Analysis Laboratory

  SEH 5290   W: loewlab.seas.gwu.edu

- Detailed instruction of GWU HPC

  https://loewlab.seas.gwu.edu/files/2020/11/mia_GW_HPC_intro.pdf