

# Deep Learning in the Cloud and at the Edge

---

## **Course Description:**

This hands-on course introduces data scientists to technologies related to building and operating live, high throughput Deep Learning applications running on powerful servers in the Cloud as well on smaller and lower power devices at the Edge of the Network.

To participate, each student will need to order an Nvidia Jetson TX2 - a System on a Chip (SOC) low power Edge GPU-equipped developer kit (one of the most powerful Edge AI platforms out there) along with a few additional peripherals. Each student will also receive a free account in the IBM SoftLayer Cloud with a \$2,000 / month allowance.

As students progress through the class, they learn to apply the concepts covered by completing hands-on homeworks and labs on the Jetson TX2 as well as in the cloud, culminating in the final project.

The material of the class is a set of practical approaches, code recipes, and lessons learned. It is based on the latest developments in the Industry and industry use cases as opposed to pure theory. It is taught by professionals with decades of industry experience.

While this class is an advanced elective and we generally assume familiarity with the basics of machine learning, operating systems, big data, and infrastructure, we fill in the blanks when it comes to some practical tooling as well as the latest developments in Deep Learning.

Every asynchronous segment is followed by a hands-on assignment as well as a synchronous hands-on lab, where students get to explore the concepts and technologies covered in the lecture. By the time you complete the course, you should be able to name the Deep Learning problem you are facing, select proper tooling, and know enough to put together an end to end application around it.

We begin the class with some real life examples of how Deep Learning is used, an overview of the state of the art advances across industries, along with a review of prominent data sets. We will review the Nvidia Jetson TX2 Edge platform, unbox it, and get it to work. There's no Deep Learning without Cloud today, so we dive into the Cloud stack for Deep Learning next. We also learn the basics of the environment that will host our labs, home assignments and the final project. Internet of Things (IoT) is the largest source of continuous data for Deep Learning, so we will discuss it next. The fourth lecture covers the fundamentals of Deep Learning such as artificial neurons, activation functions, loss functions,

backpropagation, regularization, and dropout. Next on the agenda is an in-depth overview of leading Deep Learning frameworks that provide runtimes for DL applications. We cover hardware for Deep Learning next, which includes GPUs, CPUs, and TPUs. A second lecture on Deep Learning fundamentals follows, where we review Convolutional and Recurrent Neural Networks. We move on to the tooling required to prepare datasets for Deep Learning, such as classification of images, placement of bounding boxes, semantic segmentation, and data augmentation. We next cover distributed Deep Learning, starting with the basics of High Performance Computing, Message Passing Interface, and learn how to train models on clusters of hardware. Real time Streams and Deep Learning on CPUs is next, where we learn to apply Deep Learning to heavy binary streams such as video and audio. A lecture on robotics and Deep Reinforcement Learning follows. We discuss distributed storage and data transfer aspects next. We conclude the class by reviewing several industry specific DL use cases and an update on commercial Deep Learning Cloud APIs and platforms that Cloud providers have available for academic and commercial purposes. **(3 units)**.

## **Course Format:**

The course is organized as an online inverted classroom. During each week, students first work through a set of asynchronous materials, including video lectures, readings, and other activities. Once a week, students meet for a 90-minute live session, in which they connect with an instructor and other students in an online classroom. A functioning webcam and an audio headset are required to participate in the live sessions. Students must complete all assigned asynchronous material before the scheduled live session begins.

## **Course Objectives:**

After completing this course, students will:

- Gain a basic understanding of key Deep Learning fundamental theoretical concepts.
- Achieve hands-on experience with key Deep Learning tooling and frameworks
- Learn about the classic Deep Learning datasets, such as KITTI, ImageNet, MS Coc, LibriSpeech
- Be able to code in at least one DL framework, such as PyTorch
- Develop a understanding of designing and implementing end to end Deep Learning applications on the Edge and in the Cloud
- Develop fluency with the Nvidia Jetson Edge Platform
- Understand some of the Deep Learning Industry use cases
- Understand the Infrastructure required to power Deep Learning applications, such as on-premise HPC, Big Data, and Cloud environments
- Apply Deep Learning to large scale Big Data problems

## **Prerequisites:**

Before enrolling in this course, students must have completed:

W201: Research Design and Application for Data and Analysis,

W203: Statistics for Data Science, and

W205: Fundamentals of Data Engineering

Students should be able to program in Python, Java and / or be able to pick up a new programming language on the fly. A degree of fluency is expected with the basics of operating systems (e.g. Linux) and Internet Technologies.

## **Course Evaluation:**

- Graded homeworks: 40% (homeworks 3, 6, 9, and 11 at 10% each)
- Credit/No-Credit homeworks: 10% (homeworks 1, 2, 4, 5, 7, 8, 10, 12, and 13 at 1.11% each)
- Participation: 10%
- Final Project: performing an analysis on a large dataset: 40%. The students will be required to organize into groups of 4-5 and prepare a final presentation (slides) + live demo or video (10 min)

## **List of Topics by Week:**

**Week 1:** General Course Overview. Cloud Computing. Big Data. Artificial Intelligence. Deep Learning Frameworks and Hardware. Datasets. Edge Computing. Course Project overview and sample.

Reading:

For most up-to-date reading list, please visit <https://github.com/MIDS-scaling-up/v2>

- Jetson family introduction
  - <https://developer.nvidia.com/embedded/twodaysstoadeemo>
- Introduction to SSH
  - <https://www.digitalocean.com/community/tutorials/ssh-essentials-working-with-ssh-servers-clients-and-keys>

**Week 2:** Introduction to Cloud Computing and Cloud AI. Defining the Cloud. How Clouds are used. Hypervisors in a nutshell. Types of Clouds. Cloud Services. Cloud Storage. AI as a Service. Deep Learning as a Service

Reading:

- What is Cloud Computing?: <http://www.ibm.com/cloud-computing/us/en/what-is-cloud-computing.html>

- Linux as a Hypervisor: <http://www.linuxplanet.com/linuxplanet/reports/6503/1>
- Types of clouds: <http://it.toolbox.com/blogs/storage-and-io/cloud-virtual-and-storage-networking-conversations-part-iv-49637>
- Cloud Service Types: <http://blog.appcore.com/blog/bid/168247/3-Types-of-Cloud-Service-Models>
- SoftLayer API: <http://sldn.softlayer.com/article/SoftLayer-API-Overview>
- Introduction to cloud technologies
  - <https://kubernetes.io/docs/tutorials/kubernetes-basics/>
  - <https://docs.saltstack.com/en/latest/topics/index.html>
  - <https://docs.openstack.org/security-guide/introduction/introduction-to-openstack.html>
- Getting Started with SoftLayer:
  - <http://knowledgelayer.softlayer.com/gettingstarted/meet-softlayer>
  - <http://knowledgelayer.softlayer.com/gettingstarted/how-to>
  - <http://knowledgelayer.softlayer.com/gettingstarted/how-to/set-up-your-account>

**Week 3:** Internet of Things. How has it changed in the past few years? What is the Edge and how it is different from the Cloud? Data of the Internet of Things [that usually does not make it to the cloud]. Edge devices and gateways, sensors. The Nvidia Jetson family, the Raspberry Pi. Smart Speakers such as Google Home and Amazon Alexa. IBM Blue Horizon and AWS Greengrass.

Reading:

- Fog Computing and Edge Computing: <https://www.cisco.com/c/en/us/solutions/enterprise-networks/edge-computing.html>
- Amazon IoT Portal: <https://aws.amazon.com/iot/>
- AWS GreenGrass: <https://aws.amazon.com/greengrass/>
- Google Android Things: <https://developer.android.com/things/>
- Raspberry Pi Foundation: <https://www.raspberrypi.org/>
- Amazon Lambda: <https://aws.amazon.com/lambda/>

**Week 4:** Deep Learning 101. The definition of Deep Learning: how is it different from AI and ML? Artificial Neurons, Neural Layers. Feed Forward networks. Multi-layer Perceptron. Back propagation. Normalization. Regularization. Convolutional Layers. Attention. Introduction to RNNs.

Reading:

- ImageNet: <http://image-net.org/about-overview>
- Microsoft Coco: <http://cocodataset.org/#home>
- Pascal VOC: <http://host.robots.ox.ac.uk/pascal/VOC/index.html>
- Nvidia DIGITS: <https://developer.nvidia.com/digits>
- Mozilla Deep Speech: <https://github.com/mozilla/DeepSpeech>

- ConvNetJs: <https://cs.stanford.edu/people/karpathy/convnetjs/>

**Week 5:** Deep Learning Frameworks -- What is a Deep Learning Framework? Why are there so many different ones? Which one is the best? What is a tensor? We will cover these topics as well as look at recent developments and strengths/weaknesses of the various frameworks.

Reading:

- Caffe / Caffe2: <https://caffe2.ai/docs/caffe-migration.html>
- Tensorflow: <https://www.tensorflow.org>
- PyTorch: <https://pytorch.org>
- MXNet: [https://mxnet.incubator.apache.org/faq/why\\_mxnet.html](https://mxnet.incubator.apache.org/faq/why_mxnet.html)
- Darknet YOLO: <https://wwwpjreddie.com/darknet/yolo>
- Compare all the frameworks:  
[https://en.wikipedia.org/wiki/Comparison\\_of\\_deep\\_learning\\_software](https://en.wikipedia.org/wiki/Comparison_of_deep_learning_software)

**Week 6:** Hardware for Deep Learning -- GPUs, created to accelerate computer graphics, have revolutionized Deep Learning. This section looks at why Deep Learning at scale requires special hardware, like GPUs. We will cover different chip architectures commonly used to increase performance of different frameworks.

Reading:

- GPU: <https://developer.nvidia.com/deep-learning>
- FPGA: <https://www.xilinx.com/products/silicon-devices/fpga/what-is-an-fpga.html>
- SoC: <https://www.techradar.com/news/computing/pc/system-on-a-chip-what-you-need-to-know-about-socs-1147235>
- TPU: <https://cloud.google.com/blog/products/gcp/an-in-depth-look-at-googles-first-tensor-processing-unit-tpu>

**Week 7:** Deep Learning 201. Weight Initialization. Optimizers. Adjusting the learning rate. Loss Functions. Transfer Learning. Auto-encoders. Embeddings. Inference vs training vs model design. Recurrent Neural Networks.

Reading:

- Attention is all you need: <https://hub.packtpub.com/paper-in-two-minutes-attention-is-all-you-need/>
- Yolo: <https://pjreddie.com/darknet/yolo/>
- The unreasonable effectiveness of RNNs: <http://karpathy.github.io/2015/05/21/rnn-effectiveness/>

- Black Box Adversarial Queries with Limited Queries and Information:  
<https://arxiv.org/abs/1804.08598>
- Google Tacotron: <https://google.github.io/tacotron/publications/tacotron2/>
- Creating a Driver's license test for self-driving cars:  
<https://blogs.nvidia.com/blog/2018/10/10/tuv-sud-safety-autonomous-vehicle-standards/>

**Week 8:** Dataset Collection and Preparation. Image Classification and the importance of data. Data Preparation. Data Augmentation. Sample Datasets.

Reading:

- ImageNet
  - [http://www.image-net.org/papers/imagenet\\_cvpr09.pdf](http://www.image-net.org/papers/imagenet_cvpr09.pdf)
  - <https://arxiv.org/pdf/1409.0575.pdf>
  - [http://www.image-net.org/papers/ImageNet\\_2010.pdf](http://www.image-net.org/papers/ImageNet_2010.pdf)
- Data Augmentation
  - <https://ai.googleblog.com/2018/06/improving-deep-learning-performance.html>
  - <https://arxiv.org/pdf/1708.06020.pdf>
  - <https://medium.com/ai-society/gans-from-scratch-1-a-deep-introduction-with-code-in-pytorch-and-tensorflow-cb03cdcd8a0f>

**Week 9:** HPC, MPI, Distributed Training. An overview and a bit of history. HPC vs HTC/Big Data. Typical HPC problems. Architecture of supercomputers. Interconnect topologies: fat tree, torus. FLOPs, Top500. Scaling: strong, weak, Amdahl's law. Programming for HPC systems. OpenMPI overview. Uber Horovod. Distributed Deep Learning Model training. Applying distributed training: Generative Adversarial Networks (GANs).

Reading:

- What is Open MPI? <https://www.open-mpi.org/faq/?category=general>
- Top 500 Supercomputers list: <https://www.top500.org/lists/top500/>
- IBM Summit: [https://en.wikipedia.org/wiki/Summit\\_\(supercomputer\)](https://en.wikipedia.org/wiki/Summit_(supercomputer))
- Meet Uber Horovod: <https://eng.uber.com/horovod/>
- Progressive Growing of GANs: [https://github.com/tkarras/progressive\\_growing\\_of\\_gans](https://github.com/tkarras/progressive_growing_of_gans)
- Nvidia GPU Direct: <https://developer.nvidia.com/gpudirect>

**Week 10:** Scalable Streams Processing. Concepts on Stream processing, Apache Spark as a distributed system, Running Deep learning workloads on distributed systems with Caffe, TensorFlow and other examples.

Reading:

- Deep learning on Databricks: <https://databricks.com/blog/2016/12/21/deep-learning-on-databricks.html>
- TensorFlow on Spark: <https://conferences.oreilly.com/strata/strata-eu-2018/public/schedule/detail/65059>

**Week 11:** Robotics and Deep Reinforcement Learning. Nvidia Isaac platform. Gazebo. OpenAI Gym. UC Berkeley Ray framework. We will briefly touch on the history of robotics, then dive into the world of "smart" robots.

Reading:

- OpenAI Gym: <https://gym.openai.com>
- Berkeley AI Research Ray: <https://bair.berkeley.edu/blog/2018/01/09/ray/>
- NVIDIA Isaac: <https://www.nvidia.com/en-us/deep-learning-ai/industries/robotics/>
- Deep Reinforcement Learning: <https://venturebeat.com/2018/04/05/whats-hot-in-ai-deep-reinforcement-learning/>

**Week 12:** Overview of different types of Cloud storage applicable for Big data, distributed compute and Deep Learning platforms, overview of Hadoop Distributed File System, overview of General parallel File system and overview of Openstack Object Storage Swift

Reading:

- Cloud storage defined: <https://aws.amazon.com/what-is-cloud-storage/>
- Apache HDFS: [https://hadoop.apache.org/docs/r1.2.1/hdfs\\_design.html](https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html)
- IBM GPFS:  
[https://www.ibm.com/support/knowledgecenter/en/SSFKCN\\_4.1.0/com.ibm.cluster.gpfs.v4r1.gpfs300.doc/bl1ins\\_intro.htm](https://www.ibm.com/support/knowledgecenter/en/SSFKCN_4.1.0/com.ibm.cluster.gpfs.v4r1.gpfs300.doc/bl1ins_intro.htm)
- Openstack Object storage: [https://docs.openstack.org/swift/latest/overview\\_architecture.html](https://docs.openstack.org/swift/latest/overview_architecture.html)

**Week 13:** Industry Use cases for Deep Learning. Healthcare use cases. Predicting diseases, analyzing motion data sets. What is Computational Genomics? The sequencing pipeline. Analyzing the assembled genome - the genome and the transcriptome. Use cases related to Computational Genomics. Use cases for the low power GPUs: AI Cities and the Nvidia Metropolis platform.

Reading:

- Building your AI strategy in Healthcare: <https://aibusiness.com/building-ai-healthcare-strategy/>
- Nvidia Project Clara: <https://blogs.nvidia.com/blog/2018/03/28/ai-healthcare-gtc/>
- Genome Analysis Toolkit: <https://software.broadinstitute.org/gatk/>
- Stanford GragoNN: <http://kundajelab.github.io/dragonn/>
- Nvidia Metropolis: <https://www.nvidia.com/en-us/deep-learning-ai/industries/ai-cities/>
- DeepMind Health: <https://deepmind.com/applied/deepmind-health/>

**Week 14:** Deep Learning Services and APIs, including Deep Learning as a service IBM Watson Data platform, AWS Deep Learning AMI's, Google Cloud AI and TPU's and an overview of the NVIDIA Deep learning institute platform and resources.

Reading:

- Watson studio: <https://www.ibm.com/cloud/watson-studio>
- IBM AI products and services: <https://www.ibm.com/watson/products-services/>
- AWS Deep learning: <https://docs.aws.amazon.com/dlami/latest/devguide/what-is-dlami.html>
- Google Cloud platform: <https://cloud.google.com/docs/>
- Azure Cognitive Services: <https://azure.microsoft.com/en-us/services/cognitive-services/>
- NVIDIA Deep learning institute: <https://www.nvidia.com/en-us/deep-learning-ai/education/>