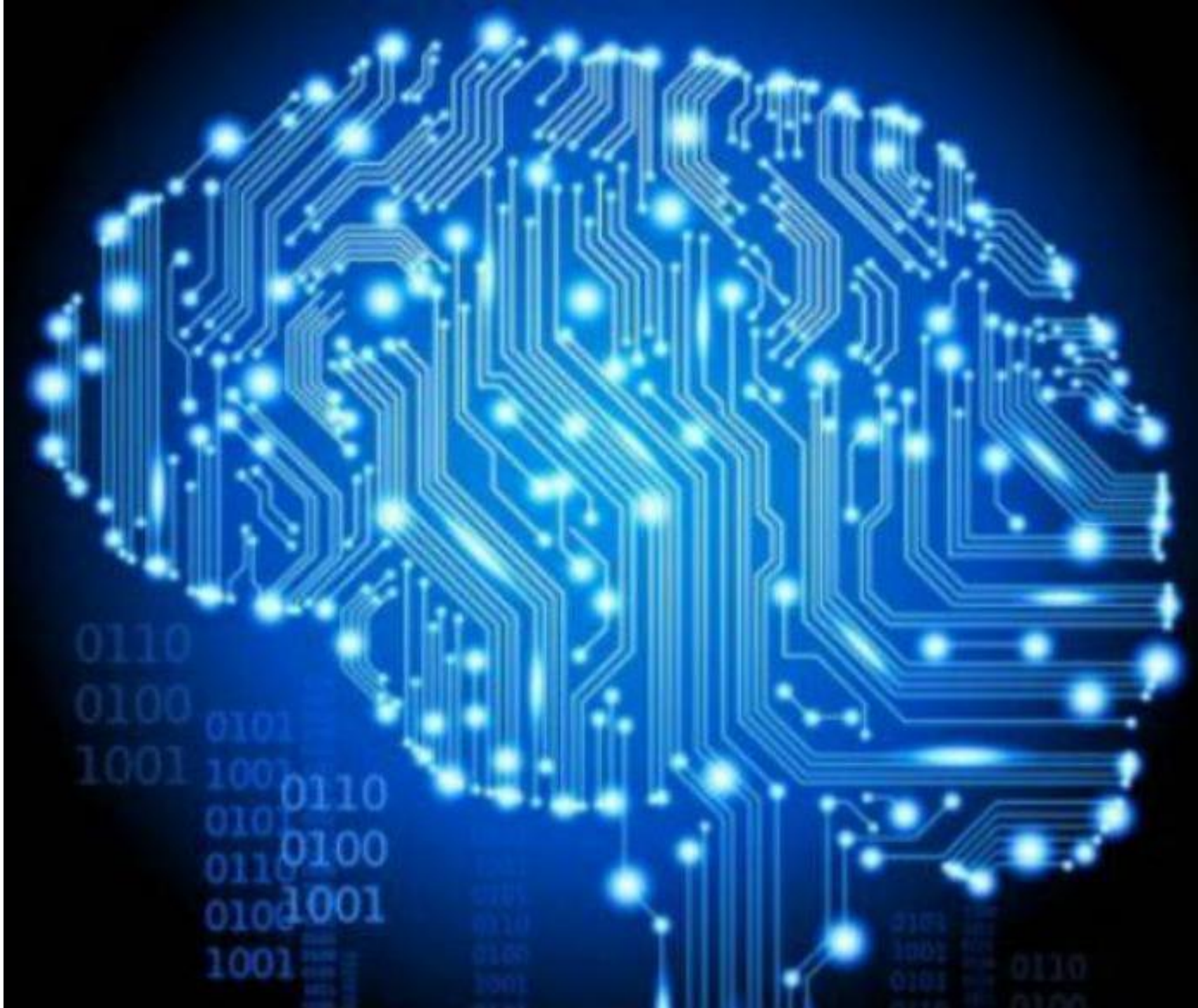
The Cluster and I



The Cluster and I

Frameworks
MinoTauro Accounts
Connecting to MinoTauro
MinoTauro jobs and params

- Many alternatives
  - Caffe2 (by Berkeley)
  - Microsoft Cognitive Toolkit (aka CNTK)
  - MXNet (by Apache)
  - PyTorch (by Facebook)
  - TensorFlow (by Google)
  - PaddlePaddle (by Baidu)
  - **Keras** (on top of TF, Theano, CNTK)
  - ...

The Cluster and I

Frameworks
MinoTauro Accounts
Connecting to MinoTauro
MinoTauro jobs and params

- One account per **registered** student
- Each student is responsible of its own account
- BSC clusters sometimes have down time
  - We will inform you promptly
  - Deadlines will be addapted if needed
- Cluster priorities
  - Based on resources and time
  - Don't leave everything for the last 3 days

The Cluster and I

Frameworks
MinoTauro Accounts
Connecting to MinoTauro
MinoTauro jobs and params

- Connect through SSH
  - `ssh username@mt1.bsc.es`
- Work on your home directory
  - `/home/nct01/username`
- To change your password, use the dt node
  - `ssh username@dt01.bsc.es`
  - `passwd`
- Use dt also for moving data

```
(DW) scp username@dt01.bsc.es:gpfs_path  local_path
(UP) scp local_path username@dt01.bsc.es:gpfs_path
```

- MT has no outside access!
- Check the MinoTauro user manual online

The Cluster and I

Frameworks
MinoTauro Accounts
Connecting to MinoTauro
MinoTauro jobs and params

- To execute you must submit a job to a queue
  - bsc_cs: max 48h
  - debug: max 1h (faster to get in)

  `#SBATCH--partition=debug`
  - Time to get enqueued depends on resources and time asked, and quota

- Giving a maximum execution time (hard limit!)

  `#SBATCH--time=HH:MM:SS`

- Initial execution path

  `#SBATCH--workdir=pathname`

- Error and log file

```
#SBATCH--error=file_name_%j.err
#SBATCH--output=file_name_%j.out
```

The Cluster and I

Frameworks
MinoTauro Accounts
Connecting to MinoTauro
MinoTauro jobs and params

- Define resources to use
  - Just one CPU

```
#SBATCH --ntasks=1
```

  - Just one GPU (unless you parallelize)

```
#SBATCH --gres gpu:1
```

- Define software needed (modules to load)
  - This includes Python 3.6 + Keras + TF

```
module purge; module load K80/default
impi/2018.1 mkl/2018.1 cuda/8.0
CUDNN/7.0.3 python/3.6.3_ML
```

The Cluster and I

Frameworks
MinoTauro Accounts
Connecting to MinoTauro
MinoTauro jobs and params

- Example of launcher file

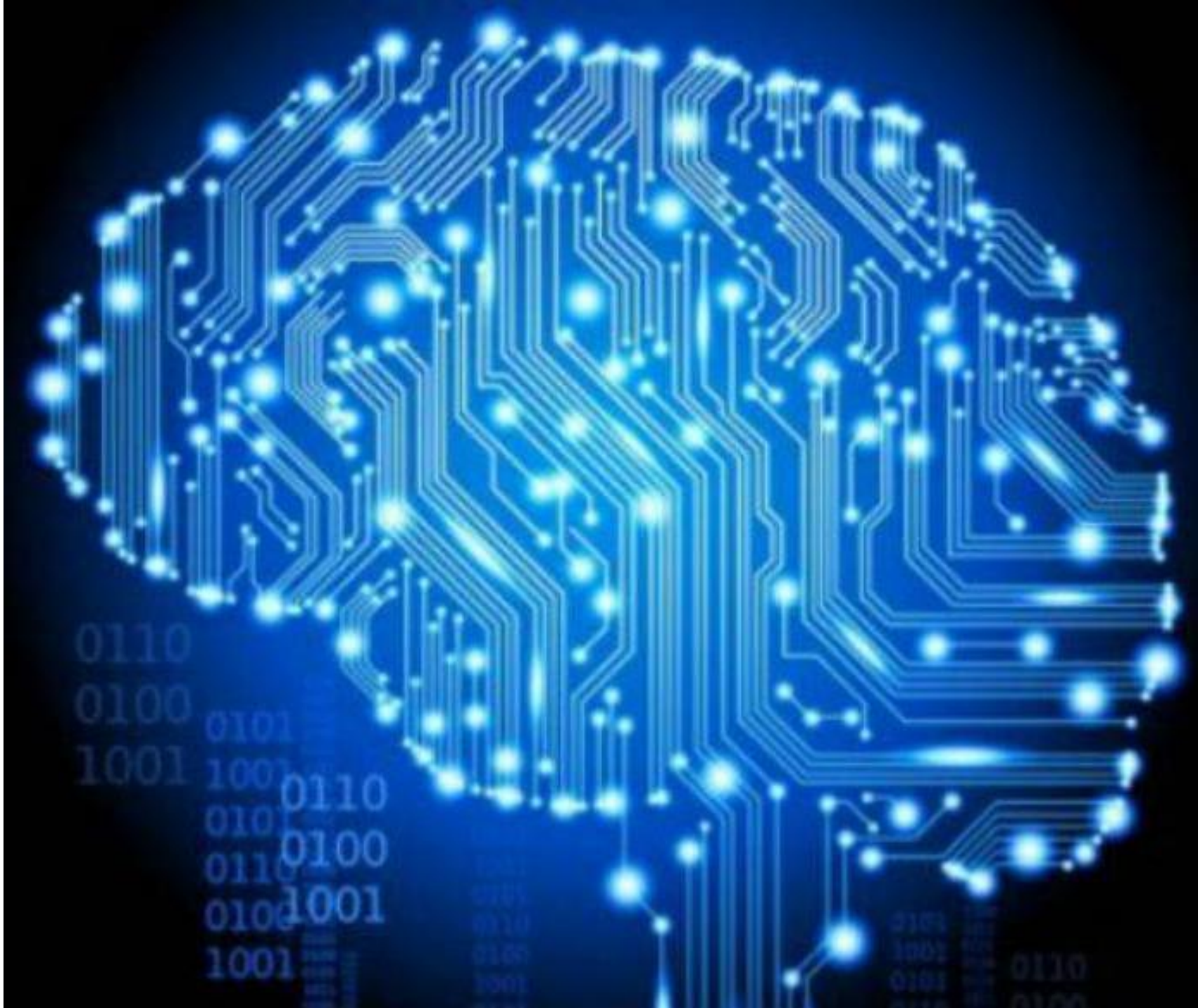
```
#!/bin/bash
#SBATCH --job-name="test_job"
#SBATCH --workdir=.
#SBATCH --output=mnist_%j.out
#SBATCH --error=mnist_%j.err
#SBATCH --ntasks=1
#SBATCH --gres gpu:1
#SBATCH --time=00:02:00

module purge; module load K80/default impi/2018.1
mkl/2018.1 cuda/8.0 CUDNN/7.0.3 python/3.6.3_ML

python some_code.py
```

The Cluster and I

Frameworks
MinoTauro Accounts
Connecting to MinoTauro
MinoTauro jobs and params

- Launching jobs
  `sbatch launcher_file.cmd`
- Checking the status of jobs
  `squeue`
- Killing jobs
  `scancel jobId`
- Checking a job "live"
  `squeue` (get computing node)
  `ssh node` (from within login node)

MNIST example



MNIST example

- Lets get the data

```
wget https://s3.amazonaws.com/img-datasets/mnist.npz
```

Upload it to the cluster

```
scp mnist.npz nct01058@dt01.bsc.es:/home/nct01/nct01058XX/.keras/datasets
```

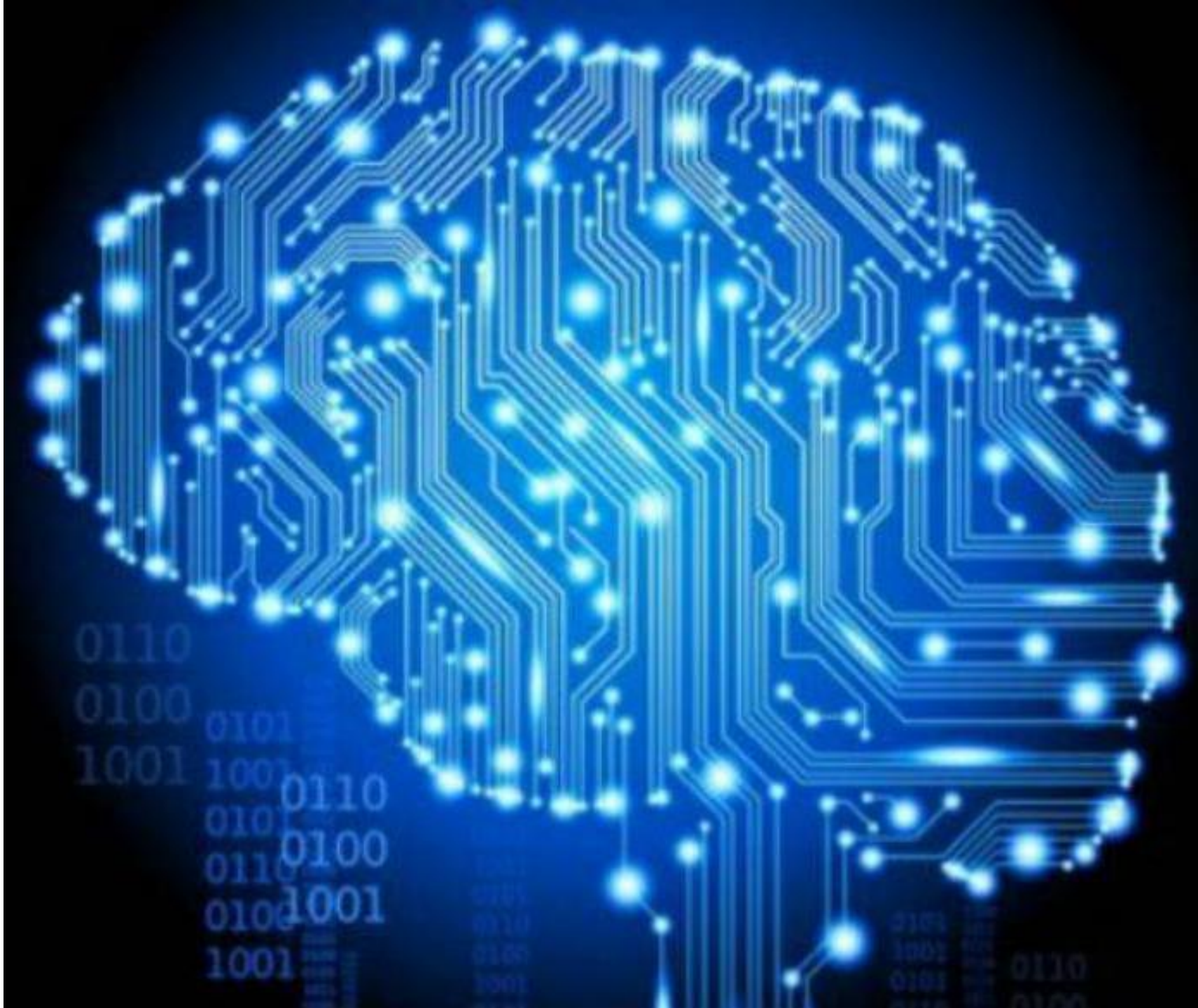
(you will need to create that directory first)

MNIST example

https://github.com/UPC-MAI-DL/UPC-MAI-DL.github.io/blob/master/_codes/1.FNN-CNN/mnist_fnn_example.py

https://github.com/UPC-MAI-DL/UPC-MAI-DL.github.io/blob/master/_codes/1.FNN-CNN/mnist_cnn_example.py

More experiments



More experiments

- You can start with CIFAR10. Its relatively hard but fast to compute. Also, in color (channels!).

```
wget http://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz
```

- You may need to rename the file, once uploaded to `.keras/datasets` to `cifar-10-batches-py.tar.gz`

- If you have problems, run the command locally, to get the files, and then upload them to MT.

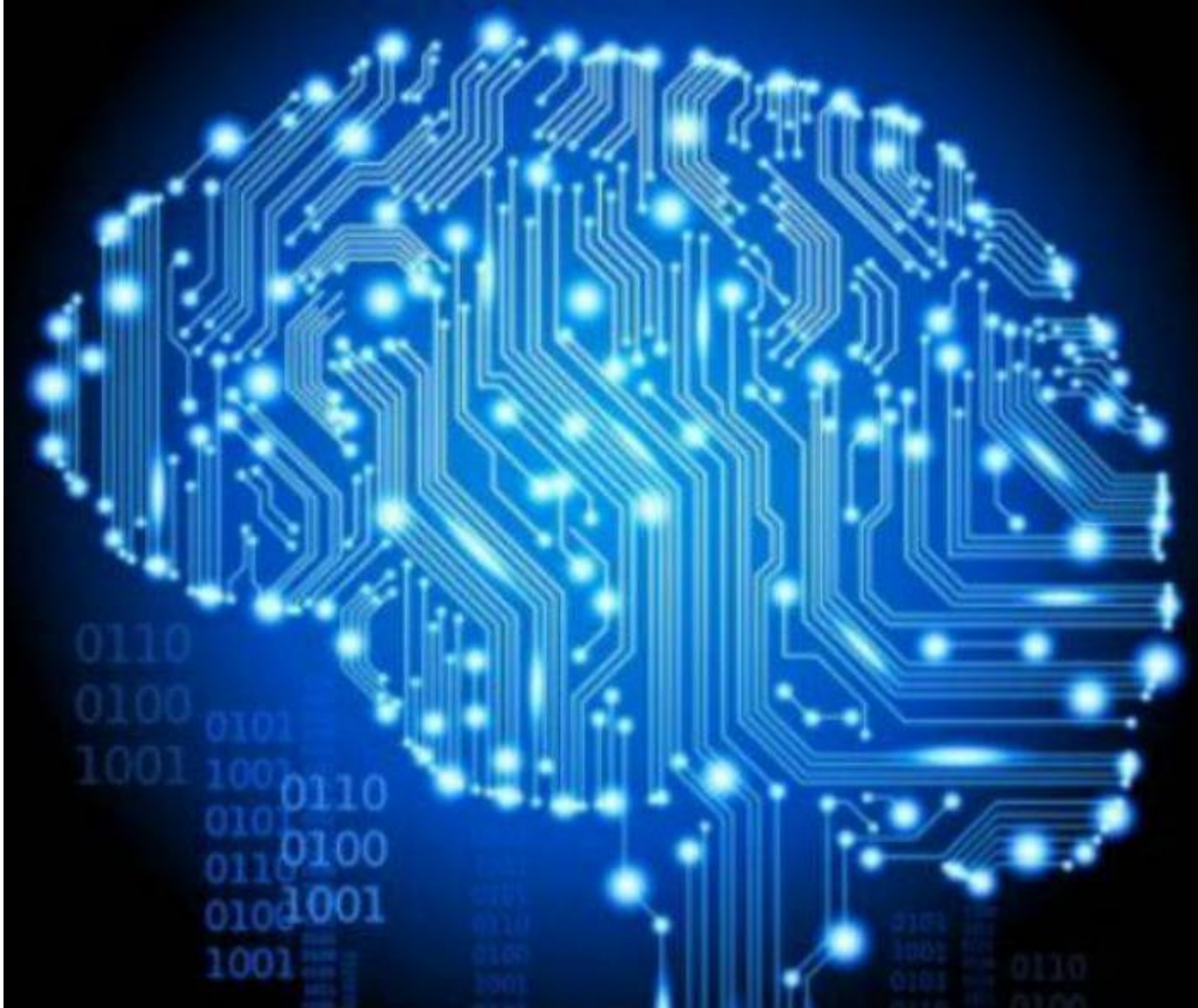
```
import keras
from keras.datasets import cifar10
(x_train, y_train), (x_test, y_test) =
                                    cifar10.load_data()
```

More experiments

- MNIST and CIFAR datasets come pre-processed for Keras (from keras.datasets import cifar10). This is rarely the case.
- Start by looking for a new dataset, and figuring out how to load it into keras (e.g., [1])

[1] https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html

Autonomous lab



Autonomous lab

- Next week: Time for your autonomous lab.
- You are supposed to work on your reports. We will be around to answer questions.
- Delivery dates on the web.
- PDF document, 8 pag. (soft restriction). Delivered through Raco. Upload the code to a public repo (github or similar) and reference in the document.

Autonomous lab

- **Goal**: The student must make coherent use of deep learning techniques, making proper analysis of the results obtained, and taking reasoned decisions.

- The topic of the report is open to your interests and curiosity.

- Simple options:
  - Choose a problem and train a network to solve it. Face the problems that will arise and use methods coherently.
  - Choose a technique or algorithm and analyze its impact and behavior in depth.

Autonomous lab

- **What will be valued**
  - Results obtained are properly assessed, leading to well argued experiments
    - No random experiments!
  - Report is written in a clear, scientific manner
    - Support your claims!
  - Show plots/tables to illustrate your findings
    - Accuracy/Loss at least!
    - Captions, legends, ...
  - Originality
    - NO MNIST please

Autonomous lab

- **What will NOT be valued**
  - Too many experiments on too many aspects without getting into any depth
    - No mindless/random experiments!
  - Questions left open or unanswered, to perform experiments on a different unrelated topic.
  - Excuses regarding the cluster.
  - Reports way over the page limit.
  - Repeating stuff said in class or introducing basic concepts.

Autonomous lab

- **TIPS**
  - Introduce the data you work with. Understand its particularities (size, resolution, sample distribution, variance, etc)
  - Specify the data splits
  - Always train your model until overfit.
  - After generating a plot, stare at it for 5 min. Make yourself questions and act in consequence. Show this reasoning process in the report.
  - You don’t need to plan ahead. Do an experiment and see what questions it raises. Follow your curiosity.

Autonomous lab

- **An very incomplete list of datasets**
  - http://web.mit.edu/torralba/www/indoor.html (MIT67)
  - http://www.vision.caltech.edu/visipedia/CUB-200.html (CUB200)
  - http://www.robots.ox.ac.uk/~vgg/data/flowers/102/ (FLOWERS102)
  - https://upc-mai-dl.github.io/mlp-convnets-lab-autonomous/vision.stanford.edu/aditya86/StanfordDogs/ (SDOGS)
  - https://upc-mai-dl.github.io/mlp-convnets-lab-autonomous/www.vision.caltech.edu/Image_Datasets/Caltech101/ (CALTECH101)
  - https://upc-mai-dl.github.io/mlp-convnets-lab-autonomous/www.vision.ee.ethz.ch/datasets_extra/food-101 (FOOD101)
  - https://www.robots.ox.ac.uk/~vgg/data/dtd/ (TEXTURES)
  - https://upc-mai-dl.github.io/mlp-convnets-lab-autonomous/www.ee.oulu.fi/~olli/Projects/Lumber.Grading.html (WOOD)