Outline

- Introduction to Deep Learning Framework

- PyTorch

- Tensor

- Networks, Forward, and Optimize

- MNIST Classification: a Toy Example

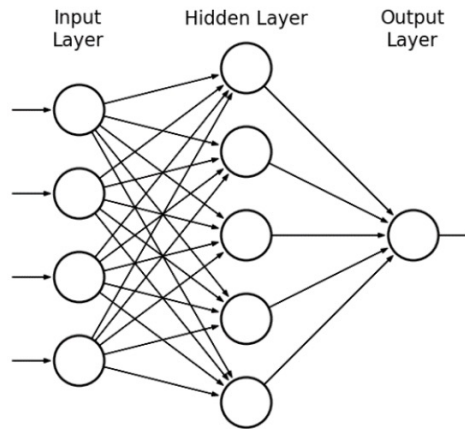- Google Cloud Platform

Deep Learning

- Learn underlying features in data using neural networks

Input:

    Images,
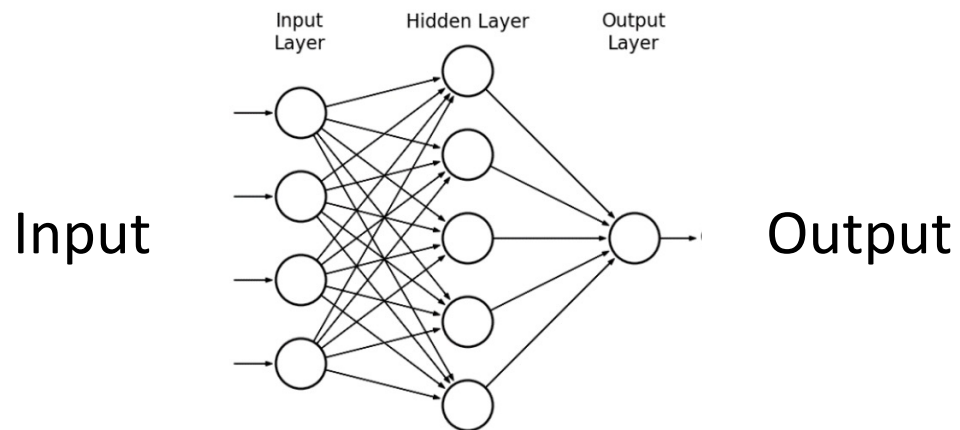   Natural language,
    Graphs,
    Signals,
    Etc.

Output:

    Image categories,
    Translated Language,
    Link prediction,
    Signal labels,
    Etc.

Input Layer   Hidden Layer   Output Layer

Multi-layer Perceptron Neural Network

Deep Learning

Input   Output

1. **Forward**: compute the output using current model and input data
2. **Compute the loss**: the error/gap between the output and ground-truth label
3. **Backward**: back propagate the loss, compute the gradient, and update the model

Deep Learning Framework

- The framework for automatic differentiation

- APIs for building neural networks, loading data, training, inference, and offering official pre-trained models

PyTorch

- A Python-based scientific computing package
  - Use the power of GPUs and other accelerators
  - Implement neural networks
  - Compute gradient in an automatic way

- Installation
  - [Anaconda](#) (Recommend): a distribution of the Python for scientific computing which makes python package installation more convenient.
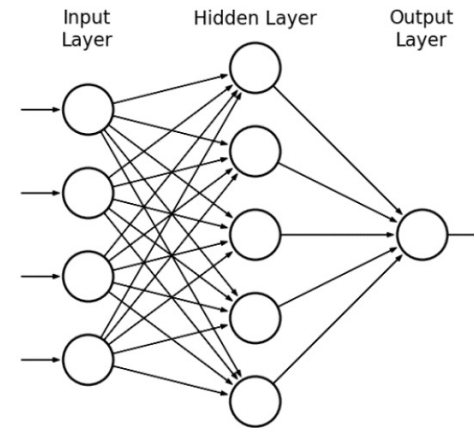
Tensor

- Tensor
  - A specialized data structure
  - Similar to NumPy's ndarrays
  - Can run on GPUs to accelerate computing

- Basic Tensor Operations
  - Create
  - Convert
  - CPU/GPU
  - Matrix Operations
  - Math Operations
  - …

Neural Networks

1. Building neural networks

2. Forward

3. Gradient (equal to partial derivatives) computation

4. Update the model weights

Neural networks: Multi-layer Perceptron, Convolutional Neural Networks, Recurrent Neural Networks, Graph Neural Networks, BERT, …
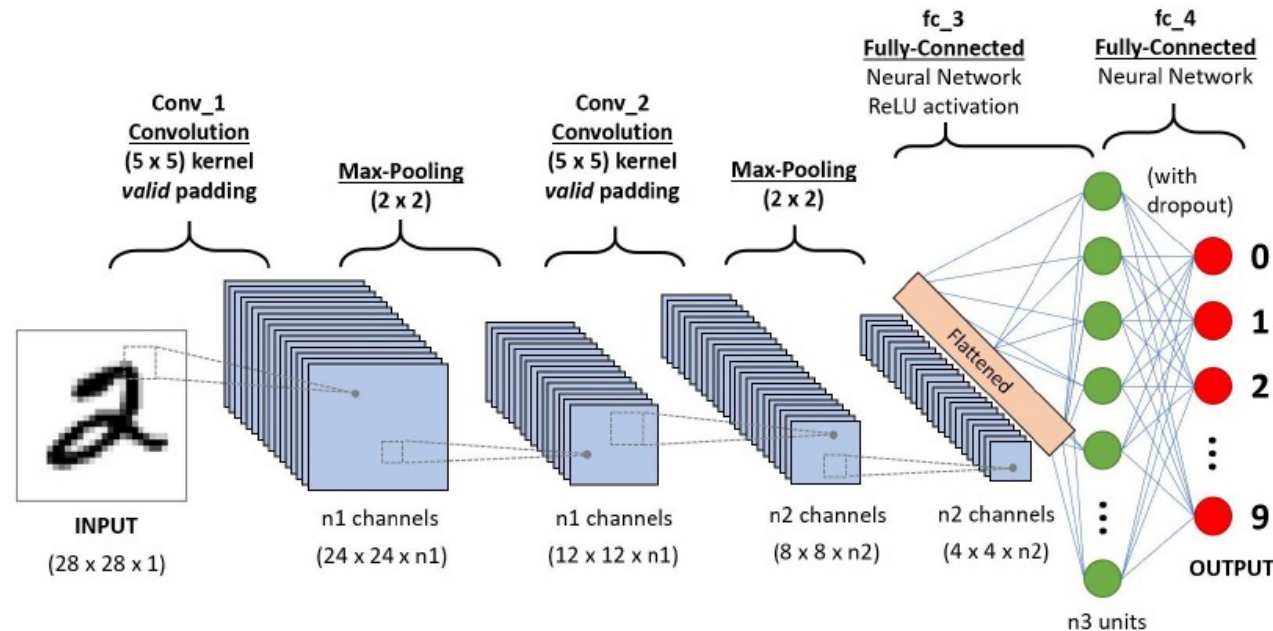
MNIST Classification: a Toy Example

- MNIST
  - A database of handwritten digits
  - Image classification

MNIST Classification: a Toy Example

- MNIST
  - A database of handwritten digits
  - Image classification

Google Colab – A *free* and *easy* way to access GPU

1. https://colab.research.google.com

2. tensor_tutorial.ipynb - Colaboratory (google.com)

3. autograd_tutorial.ipynb - Colaboratory (google.com)

4. cifar10_tutorial.ipynb - Colaboratory (google.com)

Tutorial material

1. tensor_tutorial.ipynb - Colaboratory (google.com)

2. autograd_tutorial.ipynb - Colaboratory (google.com)

3. neural_networks_tutorial.ipynb - Colaboratory (google.com)

4. cifar10_tutorial.ipynb - Colaboratory (google.com)