Introduction to Deep Learning for Scientists and Engineers Part-II

Abhijat Vatsyayan [1]

October 12, 2020

Summary

- Part I
  - Problem definition (rely on supervised learning)
  - Compute graph and gradients
  - A little about deep learning libraries.
- Part II
  - Need for different architectures
  - Convolution networks
  - Recurrent networks
- Not covered
  - Diagram of neurons
  - History
  - Recent advances and business context
  - Tutorial on pytorch or keras
- Expectation and whats next

Need for different kind of functions

Discussed previously

- Simple, linear layers can be connected together to form deep networks.
- Linear layers should be separated using non-linear functions (layers) - also referred to as activations, e.g., $RelU(x)$, $\sigma(x)$.
- Mathematically, learning is possible. In reality, people struggled to make deep networks learn.
  - Vanishing gradients
  - Compute capacity
  - Availability of data

New developments

- Activation functions
- Regularization techniques (drop off, batch normalization)
- Data (Google, Facebook, . . .), standard datasets and competitions
  - Data collected by internet and social media companies, digital consumer products like Cameras and Phones.
  - Dataset and benchmarks created by research labs and universities [1]
  - Competitions and conferences organized around some of the datasets and benchmarks
- CPUs, GPUs, nVidia
- *"New" functions*

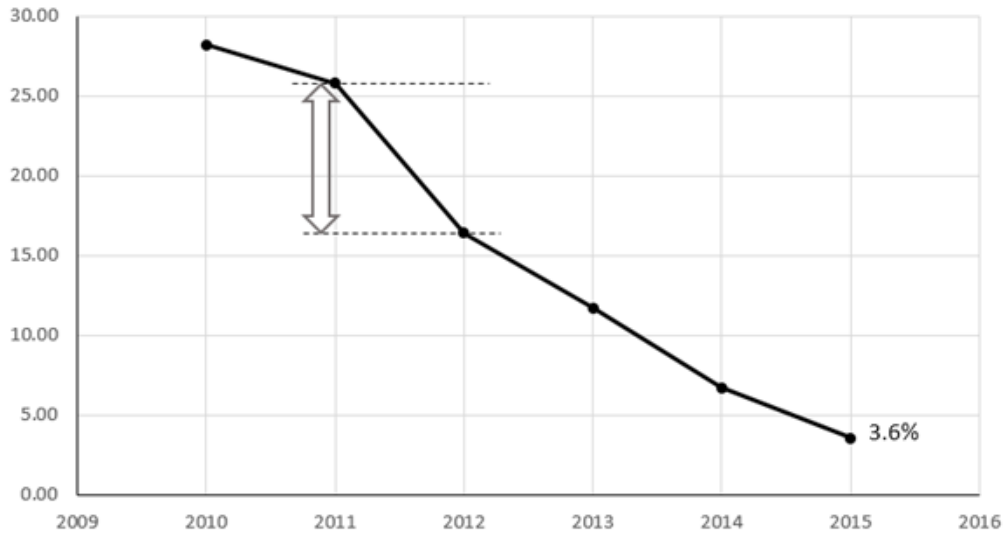
[1] See Russakovsky et al. 2015 for an example

Datasets

- Modified National Institute of Standards and Technology - MNIST (60k/10k)
- Canadian Institute For Advanced Research - CIFAR-10 (50k/10k) and CIFAR-100 (2 level, 500/100)
- Pascal Visual Object Classes (VOC) - 22k images, 20 classes
- ...
- ImageNet

ImageNet Large Scale Visual Recognition Challenge

- Publicly available dataset - ImageNet (14M+, 22k categories)
- Annual competition
  - Image classification
  - Object detection and localization
- Increasing depth
  - 7 layers AlexNet
  - 19 layers GoogLeNet
  - 152 layers ResNet



Using linear layer

Image as 2D Tensor(Matrix)

1	2	3	4	5
6	7	8	9	10
5	4	3	2	1
10	9	8	7	6

2D $\rightarrow$ 1D

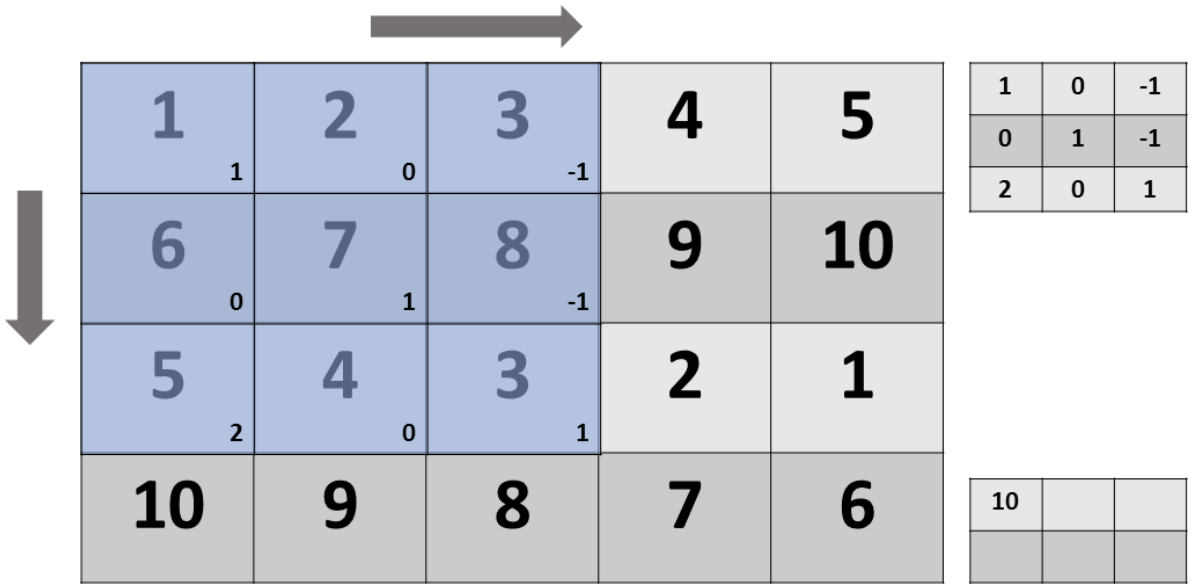
$$\mathbb{W}_{m\times 20}\begin{bmatrix}1\\2\\3\\4\\5\\6\\\cdots\\8\\7\\6\end{bmatrix}+\mathbb{B}_{m\times 1}$$

2D Convolution Example

1	2	3	4	5
6	7	8	9	10
5	4	3	2	1
10	9	8	7	6

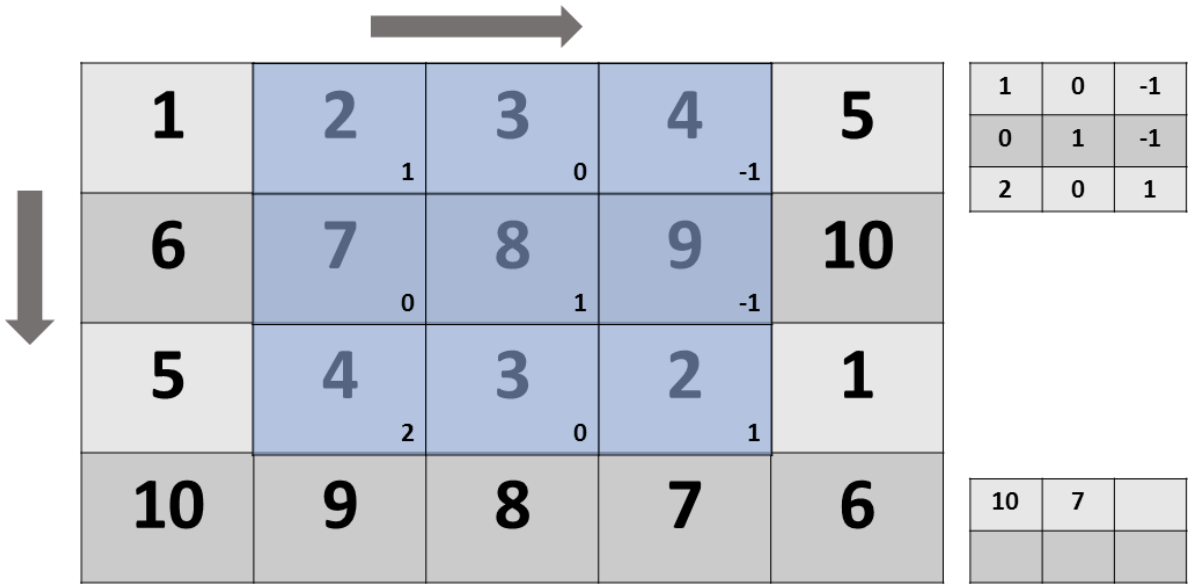
1	0	-1
0	1	-1
2	0	1

2D Convolution Example

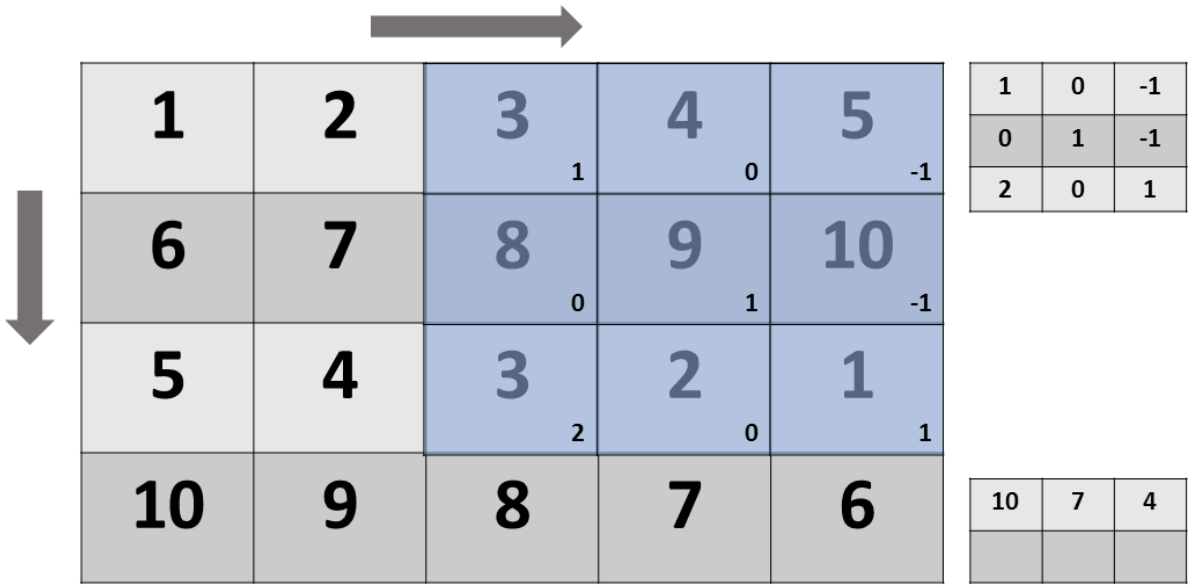


$$(1 \times 1) + (2 \times 0) + (3 \times -1) +$$
$$(6 \times 0) + (7 \times 1) + (8 \times -1) +$$
$$(5 \times 2) + (4 \times 0) + (3 \times 1)$$

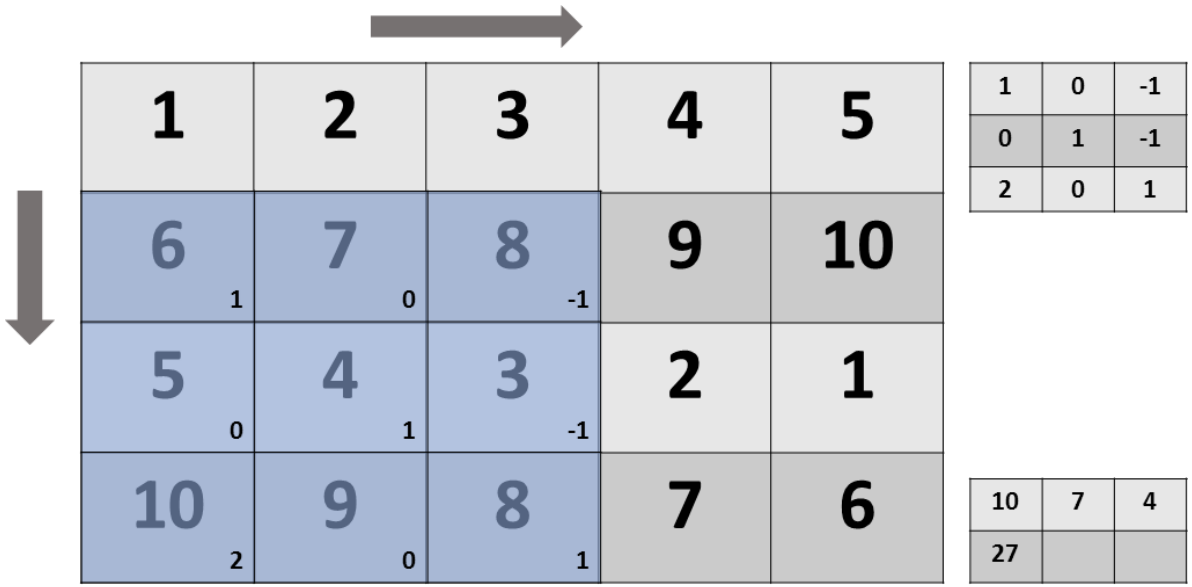
2D Convolution Example



2D Convolution Example



2D Convolution Example



2D Convolution

- Bias
- Stride
- Padding
- Layers or channels

Sequences

- Natural language tasks
- Event processing
- Statefull systems in general

- Entire sequence is known ahead of time.
- Constant length sequences.
- Variable length sequences revealed one element at a time.

Recurrent function

Functions of form

$$y^t = f(y^{t-1}, x^t; \theta) \qquad (1)$$

Hello

References

 Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, Nov 1998.