09376023

Neural Networks and Deep Learning

神经网络与深度学习

Topic 0

Course Introduction

王小华 Xiaohua Wang

Ph.D. , Associate Professor

Department of Automation

Shanghai University

Before the Class

- I would like to **know each of you**
- Contact me through “super star learning” 超星 APP
- Give me feedbacks if there is any
- Continue using course materials for Senior Design

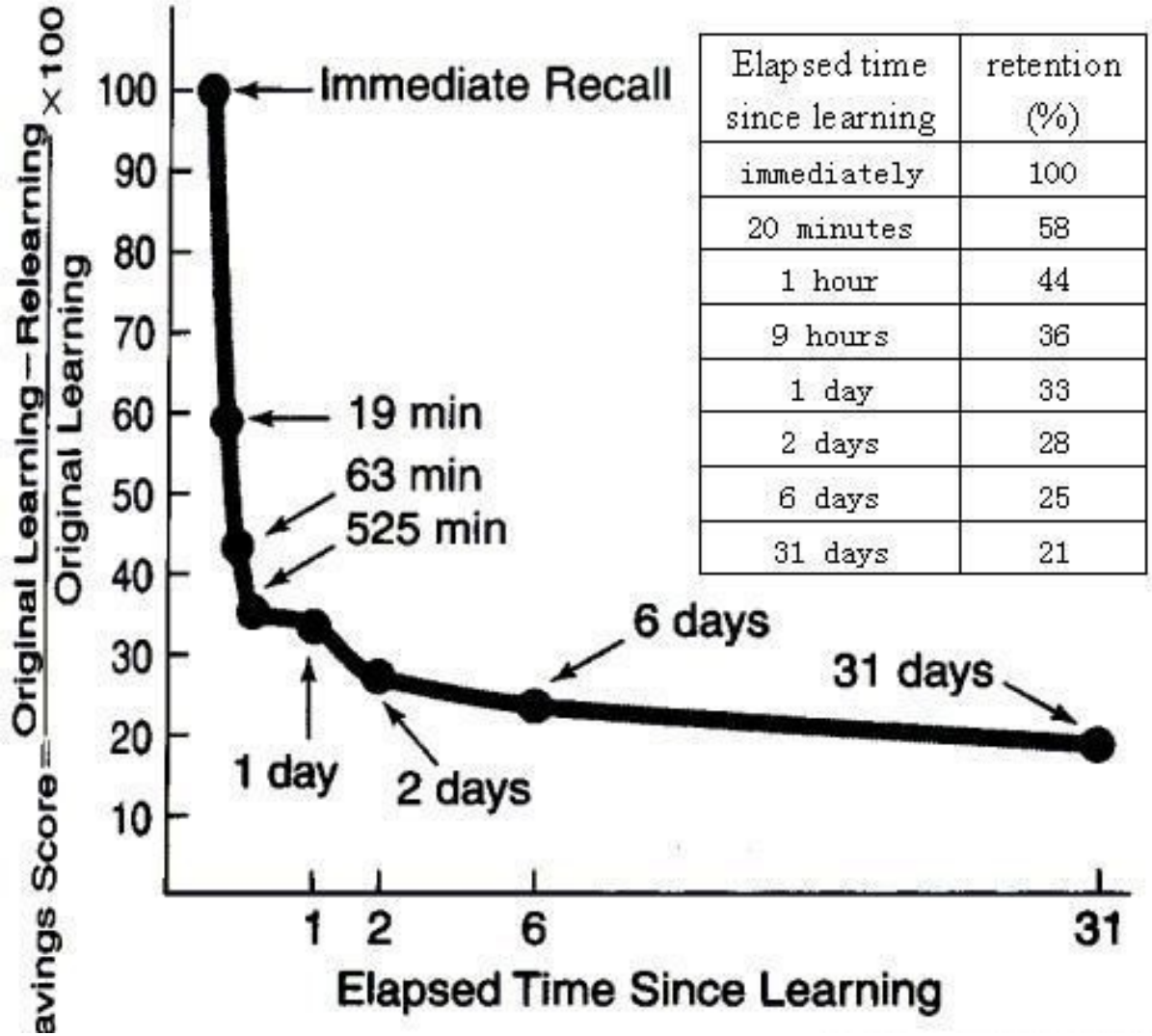
Some Important Notes to this Course

- This is a very important AI related course;

- English will be used in this course, including the lectures, textbooks, homeworks, practical reports, and the final examination;

- Preview and review is very important. Spend half an hour in reading the textbook before and after every class;

Ebbinghaus's Forgetting Curve



Elapsed time since learning	retention (%)
immediately	100
20 minutes	58
1 hour	44
9 hours	36
1 day	33
2 days	28
6 days	25
31 days	21

Outline for Topic 0

- Teaching Staff
- Brief Course Information
- Text Book and Reference
- Classroom Regulations
- Assessment
- Why Neural Network and Deep Learning?

Teaching Staff

上海大学
机电工程与自动化学院自动化系

王小华 副教授

Add: Room 525, Ji-Zi building
Email: x.wang@shu.edu.cn

Course Brief Information

- Course name: Neural Networks and Deep Learning
- Course No.: 09376023
- Credit: 3
- Class and Lab time:
  - » Lab:
    - » **Room 335B**, Ji-Zi Building (Prof Wenbin Guo)
    - » Python, Pytorch
    - » You need to take your own labtop with you
  - » Normal Course(week 1-10):
    - » Theoretical work

Text Book and Reference

Text Book: (a free online book)

Dive into Deep Learning

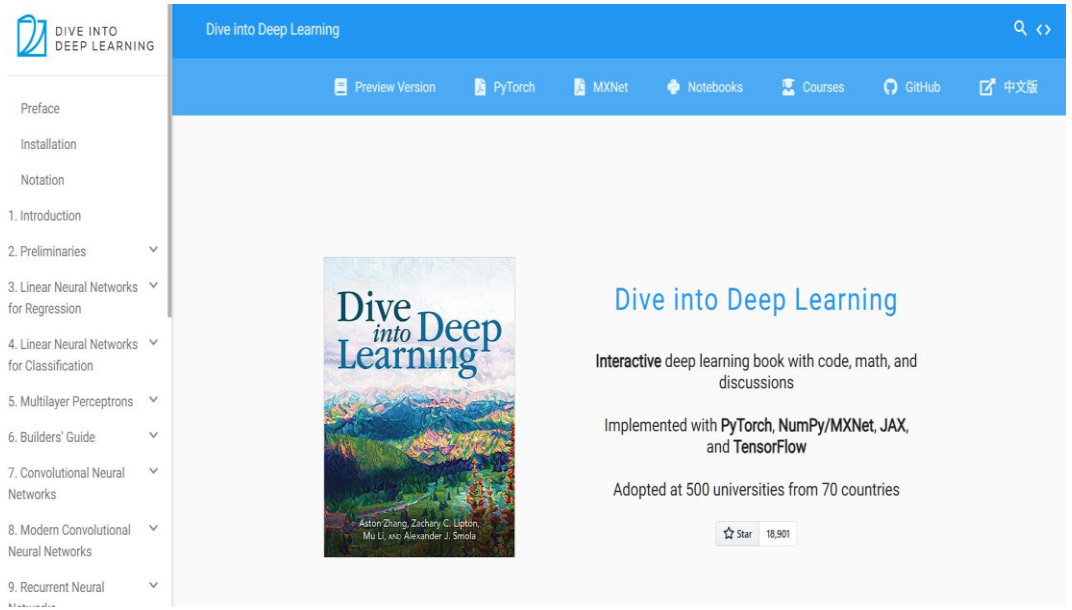
https://d2l.ai

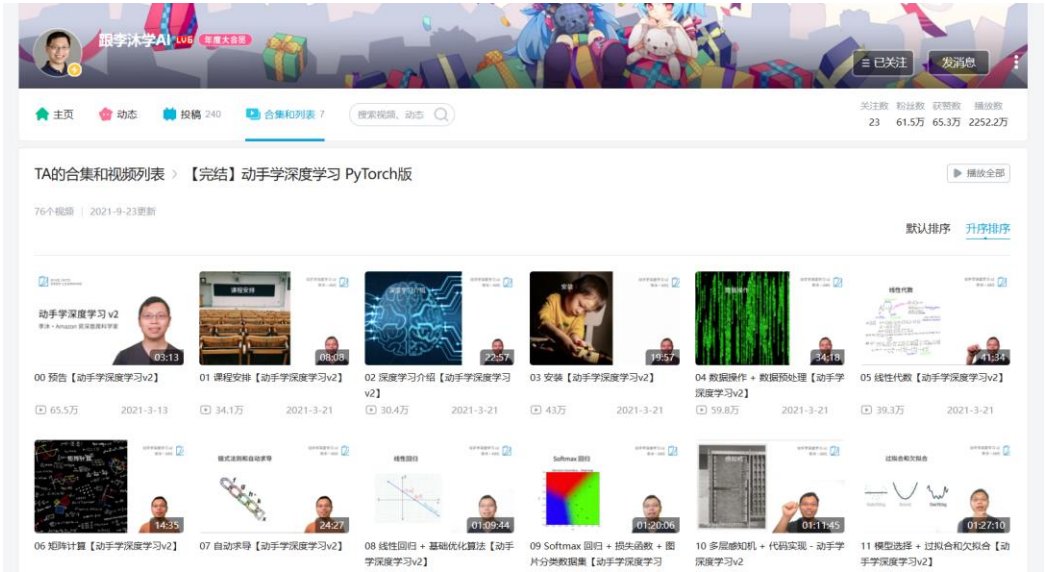
Pytorch version

Free Chinese Version

Explained in Chinese in Bilibili

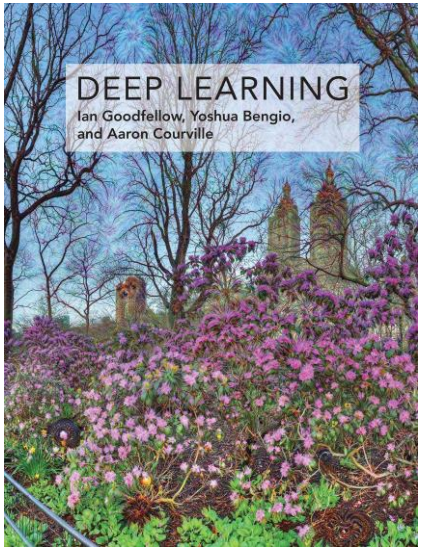
https://space.bilibili.com/1567748478/channel/seriesdetail?sid=358497

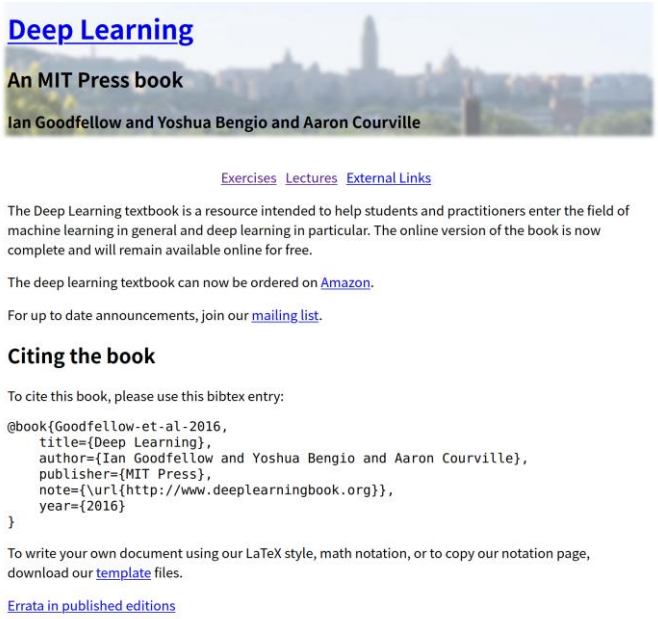




Reference Book

- Deep Learning
  - Ian Goodfellow and Yoshua Bengio and Aaron Courville
  - MIT Press, 2016
  - http://www.deeplearningbook.org





Deep Learning

An MIT Press book

Ian Goodfellow and Yoshua Bengio and Aaron Courville

Exercises Lectures External Links

The Deep Learning textbook is a resource intended to help students and practitioners enter the field of machine learning in general and deep learning in particular. The online version of the book is now complete and will remain available online for free.

The deep learning textbook can now be ordered on Amazon.

For up to date announcements, join our mailing list.

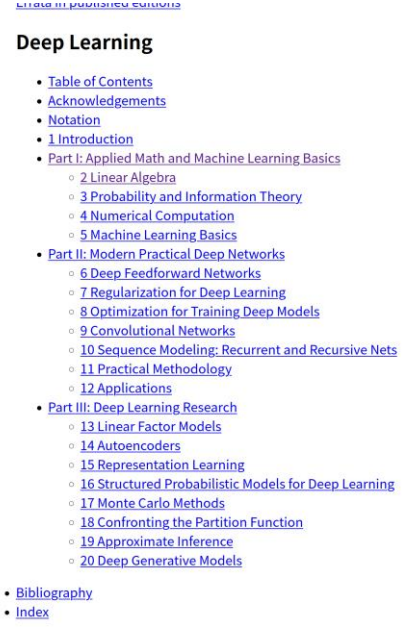
Citing the book

To cite this book, please use this bibtex entry:

```
@book{Goodfellow-et-al-2016,
    title={Deep Learning},
    author={Ian Goodfellow and Yoshua Bengio and Aaron Courville},
    publisher={MIT Press},
    note={\url{http://www.deeplearningbook.org}},
    year={2016}
}
```

To write your own document using our LaTeX style, math notation, or to copy our notation page, download our template files.

Errata in published editions



Errata in published editions

Deep Learning

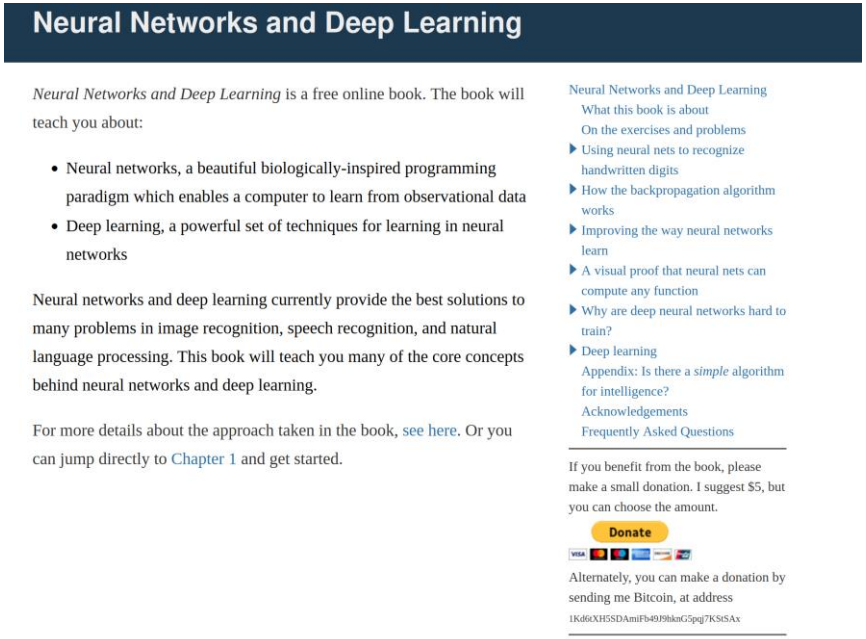
Reference:

- Neural Networks and Deep Learning

  a free online book by Michael Nielsen / Dec 2019

  http://neuralnetworksanddeeplearning.com/

- Chinese Version



Neural Networks and Deep Learning

Neural Networks and Deep Learning is a free online book. The book will teach you about:

- Neural networks, a beautiful biologically-inspired programming paradigm which enables a computer to learn from observational data
- Deep learning, a powerful set of techniques for learning in neural networks

Neural networks and deep learning currently provide the best solutions to many problems in image recognition, speech recognition, and natural language processing. This book will teach you many of the core concepts behind neural networks and deep learning.

For more details about the approach taken in the book, see here. Or you can jump directly to Chapter 1 and get started.

Neural Networks and Deep Learning
- What this book is about
- On the exercises and problems
- Using neural nets to recognize handwritten digits
- How the backpropagation algorithm works
- Improving the way neural networks learn
- A visual proof that neural nets can compute any function
- Why are deep neural networks hard to train?
- Deep learning
- Appendix: Is there a *simple* algorithm for intelligence?
- Acknowledgements
- Frequently Asked Questions

If you benefit from the book, please make a small donation. I suggest $5, but you can choose the amount.

Donate

Alternately, you can make a donation by sending me Bitcoin, at address 1Kd6tXH5SDAmiFb49J9hknG5pqj7KStSAx

Reference Course Website

- **Stanford:** CS231n: Convolutional Neural Networks for Visual Recognition.

  https://cs231n.github.io/

- **MIT:** 6.S191 Deep Learning

  http://introtodeeplearning.com/

  https://www.bilibili.com/video/BV1RS4y1d7Uc?from=search&seid=5722983192539323741&spm_id_from=333.337.0.0

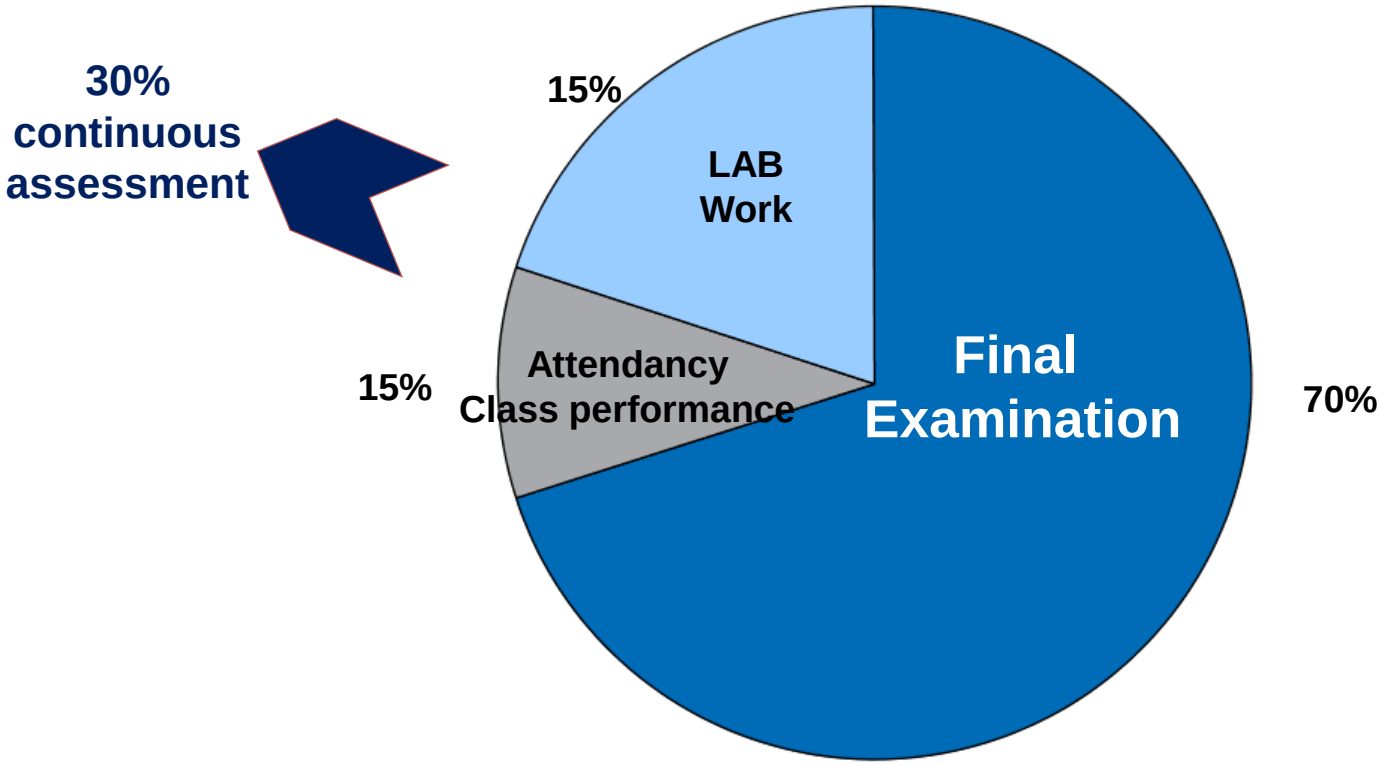
- **Cousera:** Neural Network and Deep Learning

  https://www.coursera.org/learn/neural-networks-deep-learning?

Classroom Regulations

- On time attendance is required for all students. Any absence, late arrival, or early leaving is **NOT allowed.**
  - If you are late, stand outside the classroom and wait for my permission.
- **Switch off** your mobile phone in the class.
- Concentrate on your study in the class. Speaking, eating, sleeping and any irrelevance are **NOT allowed.**
- Your performance will be directly reflected in your final score!

Assessment


30%
continuous
assessment
15%
LAB
Work
15%
Attendancy
Class performance
Final
Examination
70%

Assessment: Final(70')

- Final exam will include both lab material and lecture material
- Open book exam(which is always more difficult than a closed book exam). Please bring in all you have related to this course. No communication is allowed(Internet access is strictly prohibited).

Assessment (Cont'd)

- Attendancy (5%)
  - 90% ≤ attendance rate ≤ 100%: 5
  - 80% ≤ attendance rate < 90%: 4
  - 70% ≤ attendance rate < 80%: 3
  - 60% ≤ attendance rate < 70%: 2
  - 50% ≤ attendance rate < 60%: 1
  - attendance rate < 50%: 0
- Late arrival and early leaving is also considered as being absent from the class.

In order to get a high score, you must:
1) Attend all classes from the beginning to the end!
2) Comply with the classroom regulations!
3) Be an active participant in the in-class discussion!
4) Do your home works and practical report very well and submit them on time with ELearning system!
5) Perform well in your group presentation!

- Lab performance and Lab report(15%)
  - Practical report must be delivered before the deadline **with ELearning System**. Late practical report will not be accepted and marked.

http://www.elearning.shu.edu.cn/portal

Class performance(10%)

- Homework1
- Homework2
- Homework3

Course Outline and Teaching Arrangement

- Outline

Topic	Content	Goals
0	Course Introduction	homework1
1	Mathematics Basics	
2	Neural Network Basics	
3	BP Algorithm	homework2
4	Convolutional Neural Network	homework3
5	Recurrent Neural Network	

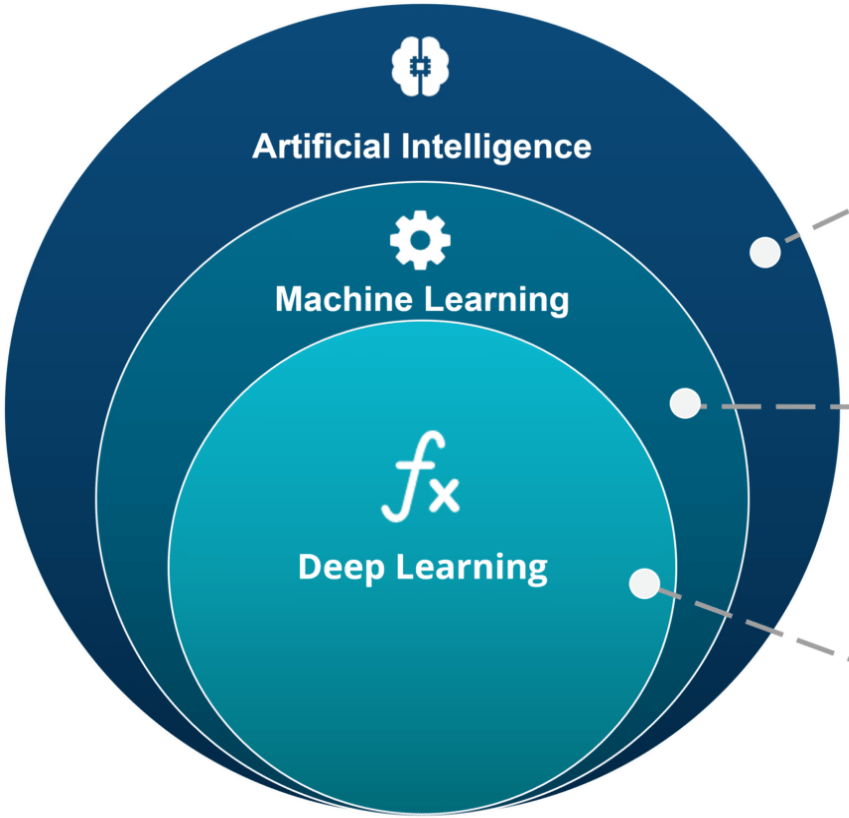
Course Outline and Teaching Arrangement (Cont.)

- Week Arrangement

Week	Lecture	Home work	Lab
1	Topic 0: Course Introduction Topic 1: Mathematics Basics		
2	Topic 2: Neural Network Basics		
3	Topic 3: BP Algorithm		
4，5，6	Topic 4: Convolutional Neural Network		
7，8	Topic 5: Recurrent Neural Network		
9	Other topics in Deep Learning	Lab report	

Any questions?

Why Neural Network and Deep Learning?



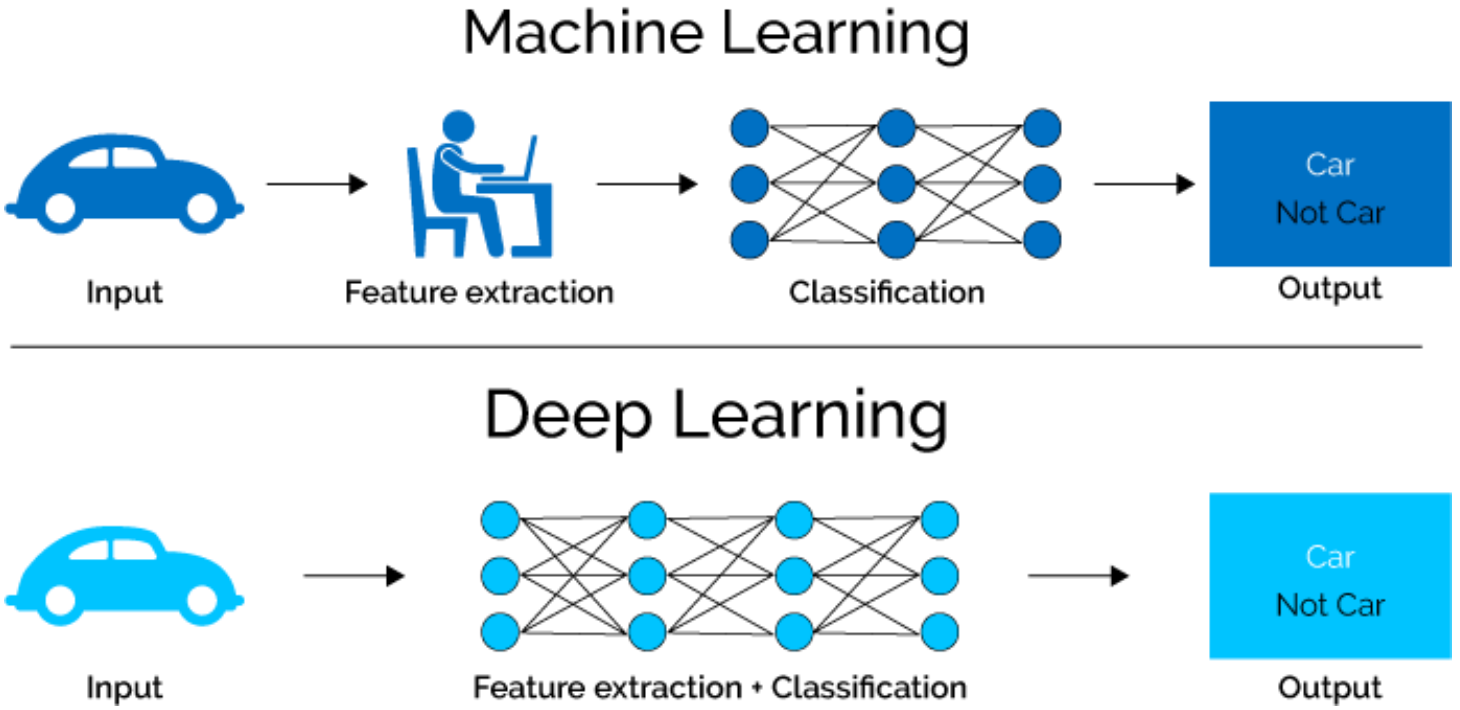
ARTIFICIAL INTELLIGENCE
A technique which enables machines to mimic human behaviour

MACHINE LEARNING
Subset of AI technique which use statistical methods to enable machines to improve with experience

DEEP LEARNING
Subset of ML which make the computation of multi-layer neural network feasible

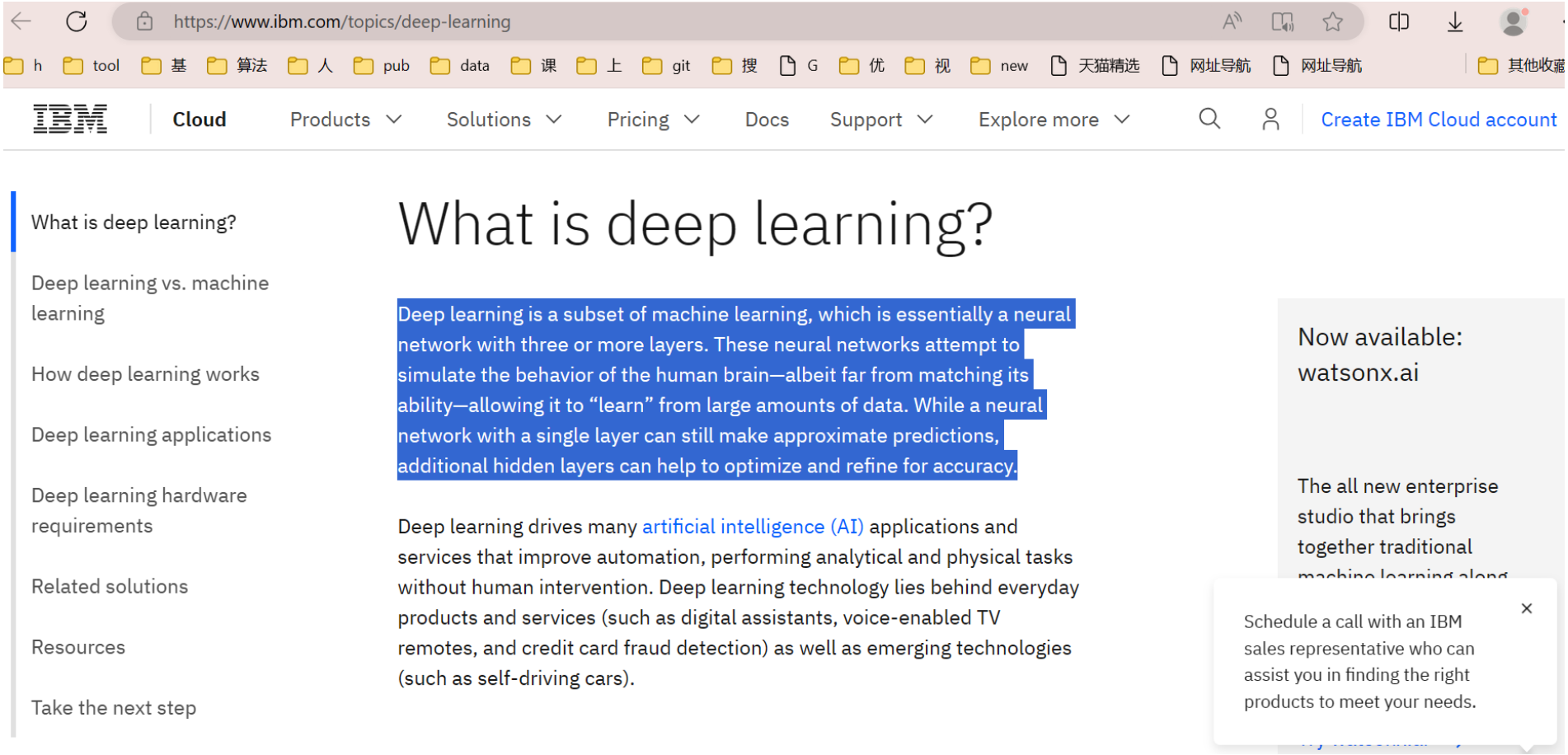
What is Deep Learning (DL) ?

A machine learning subfield of learning **representations** of data. Exceptional effective at **learning patterns**.
Deep learning algorithms attempt to learn (multiple levels of) representation by using a **hierarchy of multiple layers**
If you provide the system **tons of information**, it begins to understand it and respond in useful ways.



https://www.xenonstack.com/blog/static/public/uploads/media/machine-learning-vs-deep-learning.png

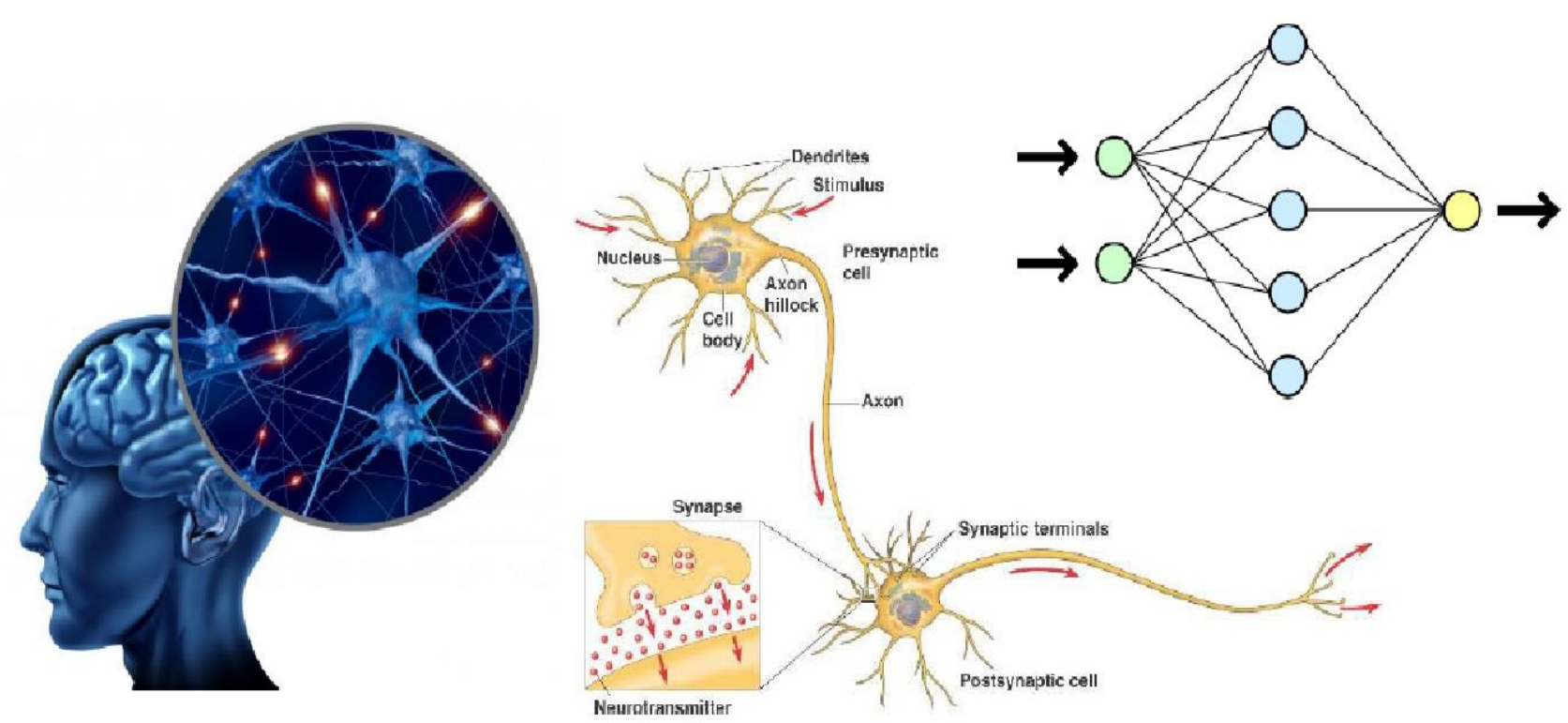
What is Deep Learning



What is Deep Learning(DL)?

- Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers.
- These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to “learn” from large amounts of data.
- While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.

Neural Network: Inspired by Human Brains

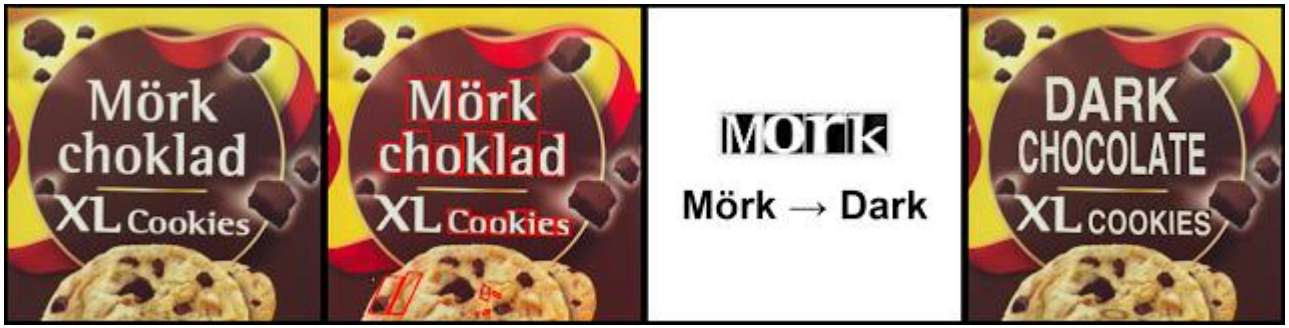


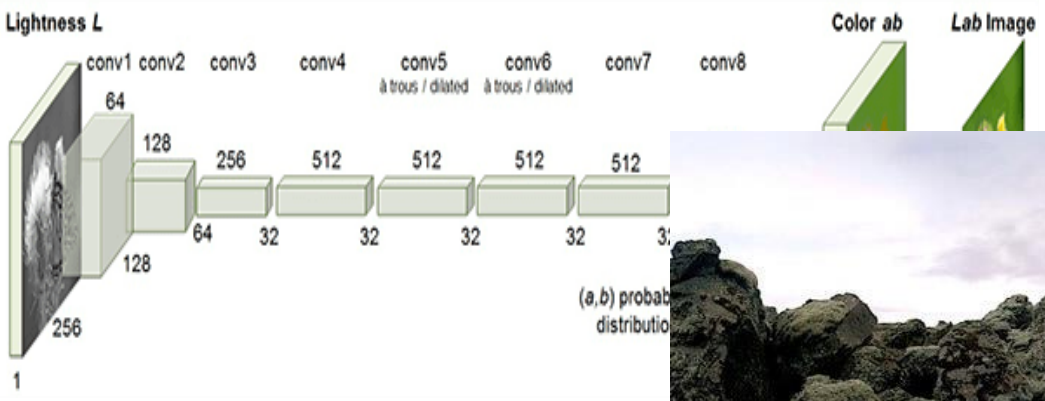
Homework 1: Make a short video(less than 3 minutes) according to this picture And try to explain how exactly human brain functions helped the creation of neural networks

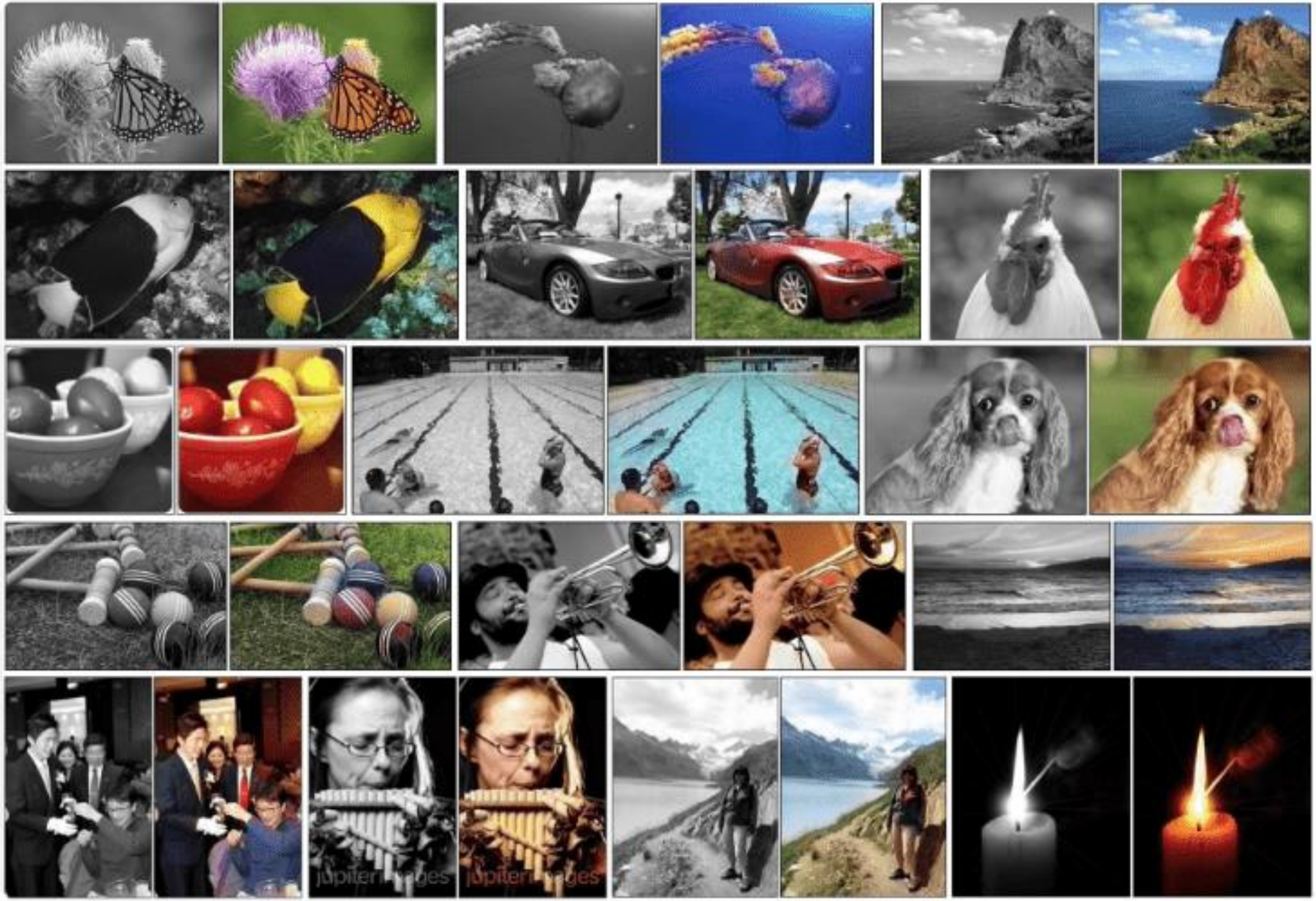
I will select the best 3 and show to the whole class

0 DEEP LEARNING Applications

1. Self Driving Cars
2. Entertainment
3. Visual Recognition
4. Virtual Assistants
5. Fraud Detection
6. Natural Language Processing
7. News Aggregation and Fraud News Detection
8. Detecting Developmental Delay in Children
9. Colourisation of Black and White images
10. Adding sounds to silent movies
11. Healthcare
12. Personalisations
13. Automatic Machine Translation
14. Automatic Handwriting Generation
15. Demographic & Election Predictions
16. Automatic Game Playing
17. Language Translations
18. Pixel Restoration
19. Photo Descriptions
20. Deep Dreaming


Mörk choklad
XL Cookies
Mörk choklad
XL Cookies
Mörk
Mörk → Dark
DARK CHOCOLATE
XL COOKIES


Lightness L
conv1 conv2 conv3 conv4 conv5 conv6 conv7 conv8
à trous / dilated
à trous / dilated
64
128
256
512
512
512
512
64
32
32
32
32
128
256
1
(a,b) probab
distributio
Color ab
Lab Image

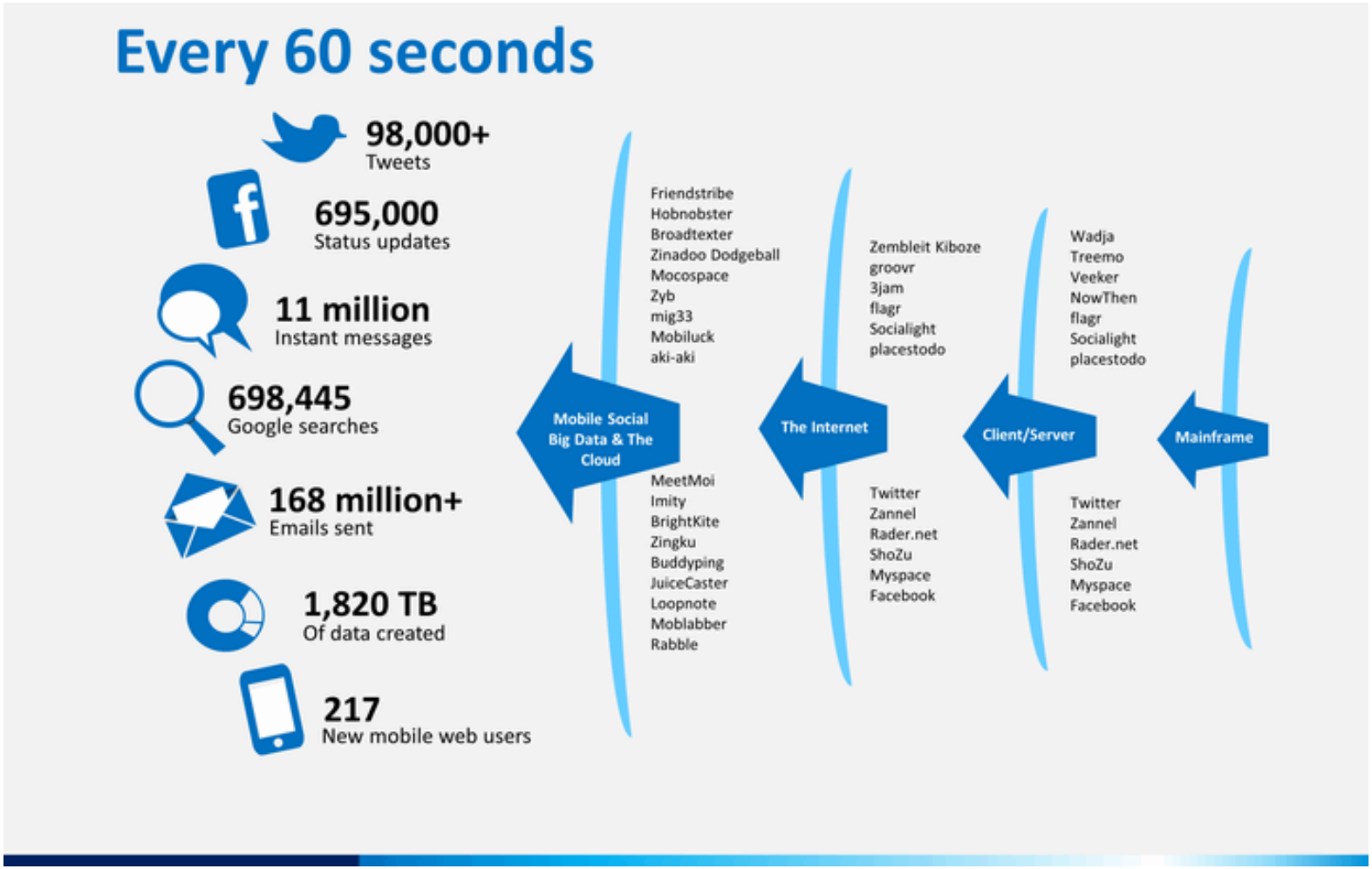


History of Deep Learning

- 1960s: Perceptron (single layer neural network)
- 1969: Perceptron has limitation
- 1980s: Multi-layer perceptron
- 1986: Backpropagation
- 1989: 1 hidden layer is “good enough”, why deep?
- 2006: Restricted Boltzmann Machine
- 2009: GPU
- 2010: breakthrough in Speech Recognition (Dahl et al., 2010)
- 2012: breakthrough in ImageNet (Krizhevsky et al. 2012)
- 2015:“Superhuman” results in Image and Speech Recognition

Reasons Why Deep Learning Woks

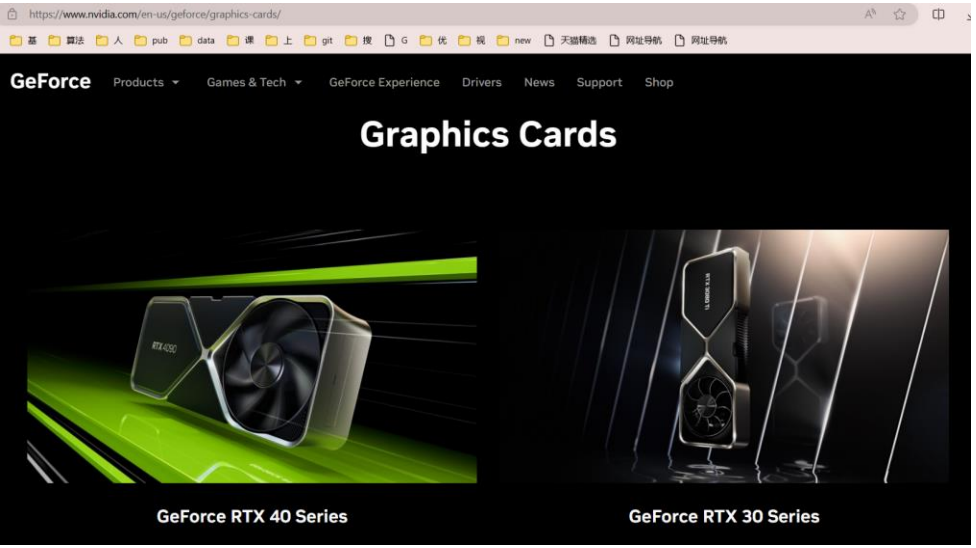
Big Data



Reasons Why Deep Learning Woks

GPU (*graphical processing units*)

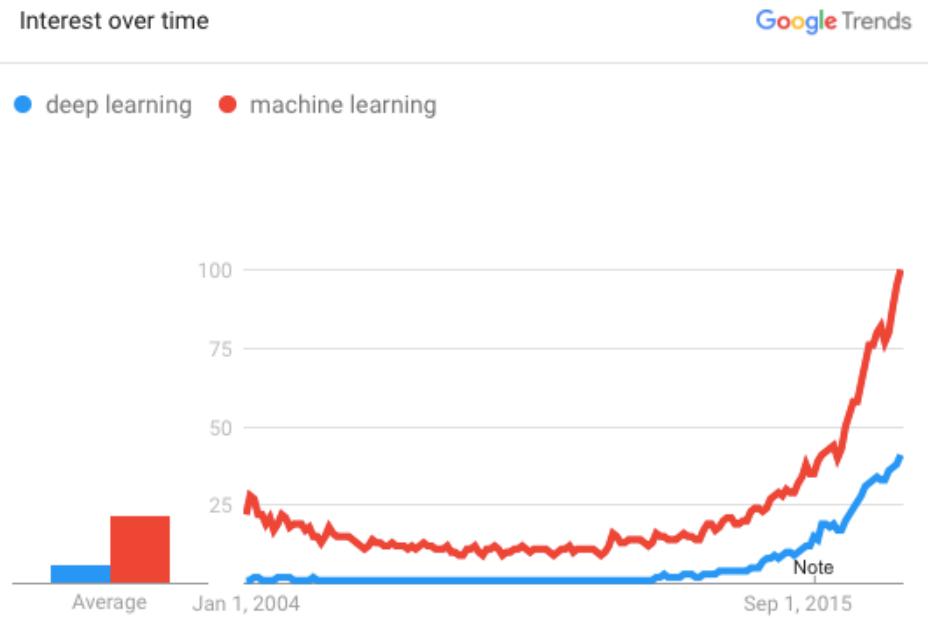
Nvidia Geforce 10-Series GPU Comparison						
	Geforce GTX 1080 (desktop)	Geforce GTX 1080 (notebook)	Geforce GTX 1070 (desktop)	Geforce GTX 1070 (notebook)	Geforce GTX 1060 (desktop)	Geforce GTX 1060 (notebook)
GPU	GP104	GP104	GP104	GP104	GP106	GP106
CUDA Cores	2560	2560	1920	2048	1280	1280
Base Clock	1607MHz	1556MHz	1506MHz	1443MHz	1506MHz	1405MHz
Boost Clock	1733MHz	1733MHz	1683MHz	1645MHz	1709MHz	1670MHz
Memory	8GB GDDR5X	8GB GDDR5X	8GB GDDR5	8GB GDDR5	6GB GDDR5	6GB GDDR5
Release Date	27-Jun-16	15-Aug-16	10-Jun-16	15-Aug-16	19-Jul-16	15-Aug-16



Why is DL useful?

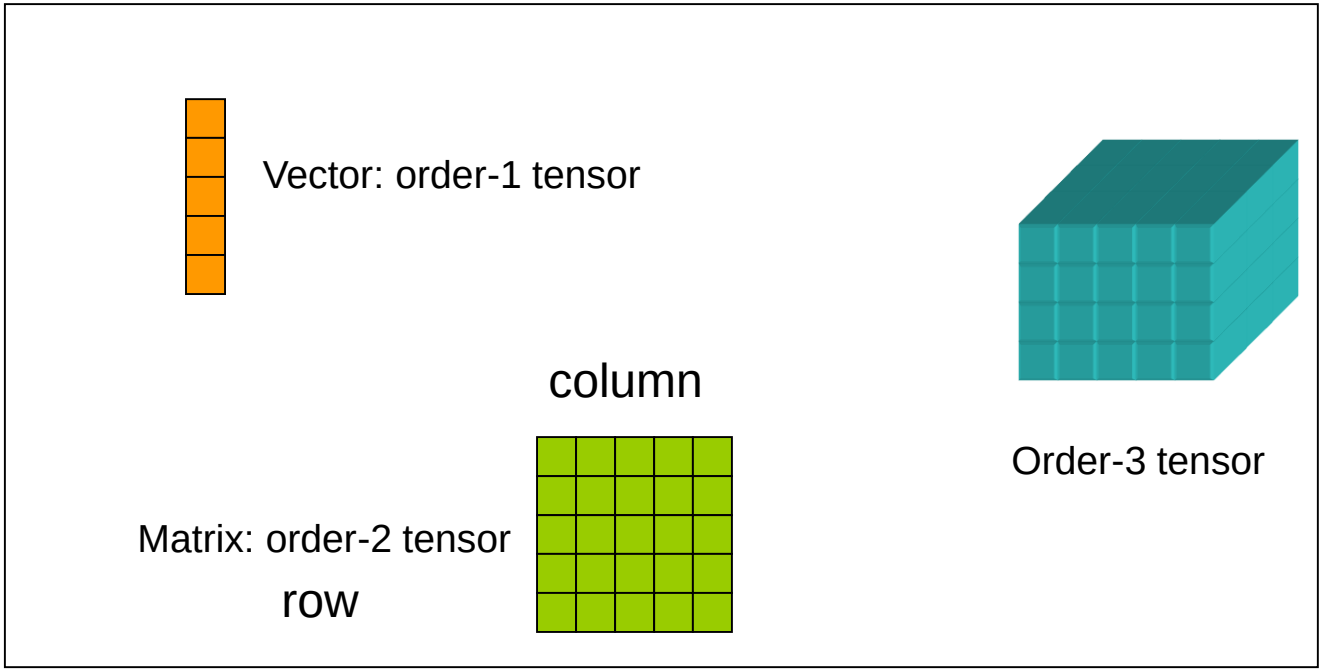
- Manually designed features are often **over-specified**, **incomplete** and take a **long time to design** and validate
- Learned Features are **easy to adapt**, **fast** to learn
- Deep learning provides a very **flexible**, (almost?) **universal**, learnable framework for representing world, visual and linguistic information.
- Can learn both unsupervised and supervised
- Effective **end-to-end** joint system learning
- Utilize large amounts of training data

In ~2010 DL started outperforming other ML techniques
first in speech and vision, then NLP



Mathematics

Tensor: Generalization of an n-dimensional array



What shape should a order-4 tensor be?

What shape should a order-n tensor be?

- Scalar
- Vector
- Matrix
- Tensor

- A = torch.arange(12,dtype=torch.float32)

```
tensor([ 0.,  1.,  2.,  3.,  4.,  5.,  6.,  7.,  8.,  9., 10., 11.])
```

- X1= A.reshape(3,4)

```
tensor([[ 0.,  1.,  2.,  3.],
        [ 4.,  5.,  6.,  7.],
        [ 8.,  9., 10., 11.]])
```

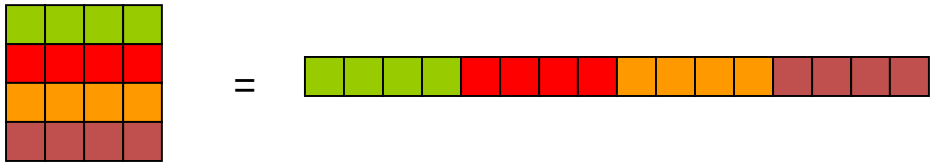
1. First, the tensor class supports automatic differentiation.
2. Second, it leverages GPUs to accelerate numerical computation,

- X2= A.reshape(2,2,3), X3= A.reshape(2,3,2)
- B=torch.tensor([[2, 1, 4, 3], [1, 2, 3, 4], [4, 3, 2, 1]])
- B.shape
- C=tensor([[[0.],[1.],[2.],[3.]],[[4.],[5.],[6.],[7.]])
- C.shape

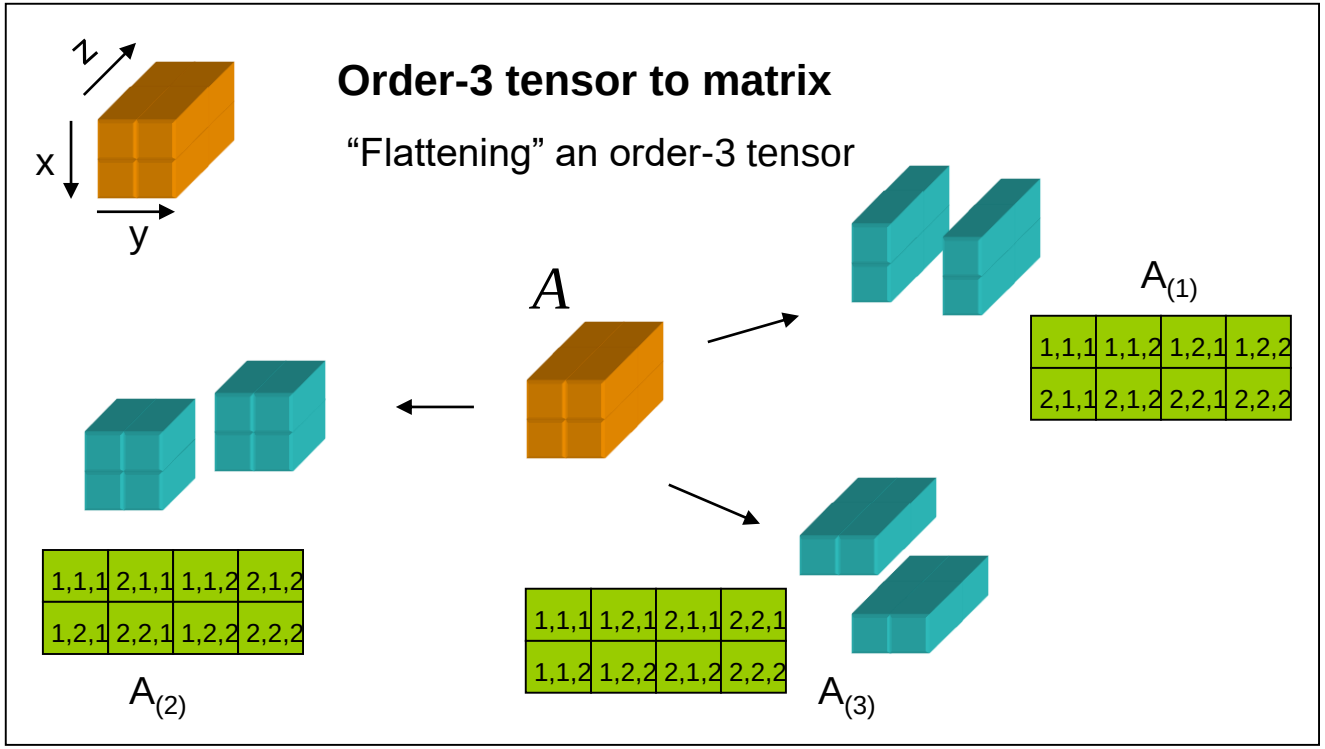
Reshaping Tensors

Matrix to vector

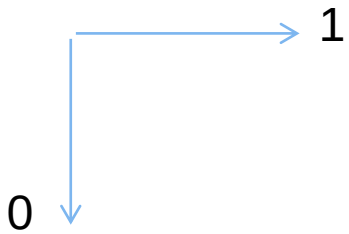
“Vectorizing” a matrix



Reshaping Tensors



Try to reshape differently in your lab time



Tensor Operation

- Elementwise operation

（tensors of same size）

  - Addition
  - Subtraction
  - Production
    - Multiply，Time
  - Exponential Operation
  - Concatenate

- x = torch.tensor([1.0, 2, 4, 8])
- y = torch.tensor([2, 2, 2, 2])
- x + y, x - y, x * y, x / y, x ** y

(tensor([3., 4., 6., 10.]),
tensor([-1., 0., 2., 6.]),
tensor([2., 4., 8., 16.]),
tensor([0.5000, 1.0000, 2.0000, 4.0000]),
tensor([1., 4., 16., 64.]))

- torch.cat((x, y), dim=0)
- torch.cat((x, y), dim=1)

size:(1,4)->(2,4) | size:(1,4)->(1,8)

Tensor Operation

- Comply with Linear Algebra
  - Learn more in your lab

Derivative

- Difference quotients
- $f'(a) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$

- Called the derivative of function f at a
- This computation is called differentiation

Partial Derivative

If $f(x, y)$ is a function of two variables, then $\frac{\partial}{\partial x} f(x, y)$ is defined as the derivative of the function $g(x) = f(x, y)$, where y is considered a constant. It is called **partial derivative** of f with respect to x. The partial derivative with respect to y is defined similarly.

- Gradient
- AutoGrad(automatic differentiation)

$Y=2x^{T}x$

```
x = torch.arange(4.0)
x.requires_grad_(True)
x.grad   # The gradient is None by default
y = 2 * torch.dot(x, x)

#take the gradient of y with respect to x by calling its backward method
y.backward()
x.grad
tensor([ 0., 4., 8., 12.])
x.grad == 4 * x
tensor([True, True, True, True])
```

Definition of Probability

- ***Experiment***: toss a coin twice
- ***Sample space***: possible outcomes of an experiment
  - S = {HH, HT, TH, TT}
- ***Event***: a subset of possible outcomes
  - A={HH}, B={HT, TH}
- ***Probability of an event*** : an number assigned to an event Pr(A)
  - Axiom 1: $Pr(A) \geq 0$
  - Axiom 2: $Pr(S) = 1$
  - Axiom 3: For every sequence of disjoint events, $sum(P_i)=1$

Random Variable and Distribution

- A ***random variable X*** is a numerical outcome of a random experiment
- The ***distribution*** of a random variable is the collection of possible outcomes along with their probabilities:
  - Discrete case: $\Pr(X = x) = p_\theta(x)$
  - Continuous case: $\Pr(a \le X \le b) = \int_a^b p_\theta(x)dx$

Normal (Gaussian) Distribution

- $X \sim N(\mu,\sigma)$

$$p_\theta(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}$$

$$\Pr(a \le X \le b) = \int_a^b p_\theta(x)dx = \int_a^b \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} dx$$

- $E[X] = \mu$, $Var(X) = \sigma^2$

- If $X_1 \sim N(\mu_1,\sigma_1)$ and $X_2 \sim N(\mu_2,\sigma_2)$, $X = X_1 + X_2$?

Independence

- Two events ***A and B are independent*** in case

$$\Pr(AB) = \Pr(A)\Pr(B)$$

Conditioning

- If A and B are events with Pr(A) > 0, the ***conditional probability of B given A*** is

$$\Pr(B \mid A) = \frac{\Pr(AB)}{\Pr(A)}$$

Bayes' Rule

Given two events A and B and suppose that Pr(A) > 0. Then

$$\Pr(B \mid A) = \frac{\Pr(AB)}{\Pr(A)} = \frac{\Pr(A \mid B)\Pr(B)}{\Pr(A)}$$

Example:

Pr(R) = 0.8

Pr(W|R)	R	¬R
W	0.7	0.4
¬W	0.3	0.6

R: It is a rainy day

W: The grass is wet

Pr(R|W) = ?

Home reading

- Textbook chapters
- Whatever material related that interests you