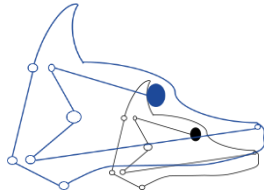
Short Intro for this workshop

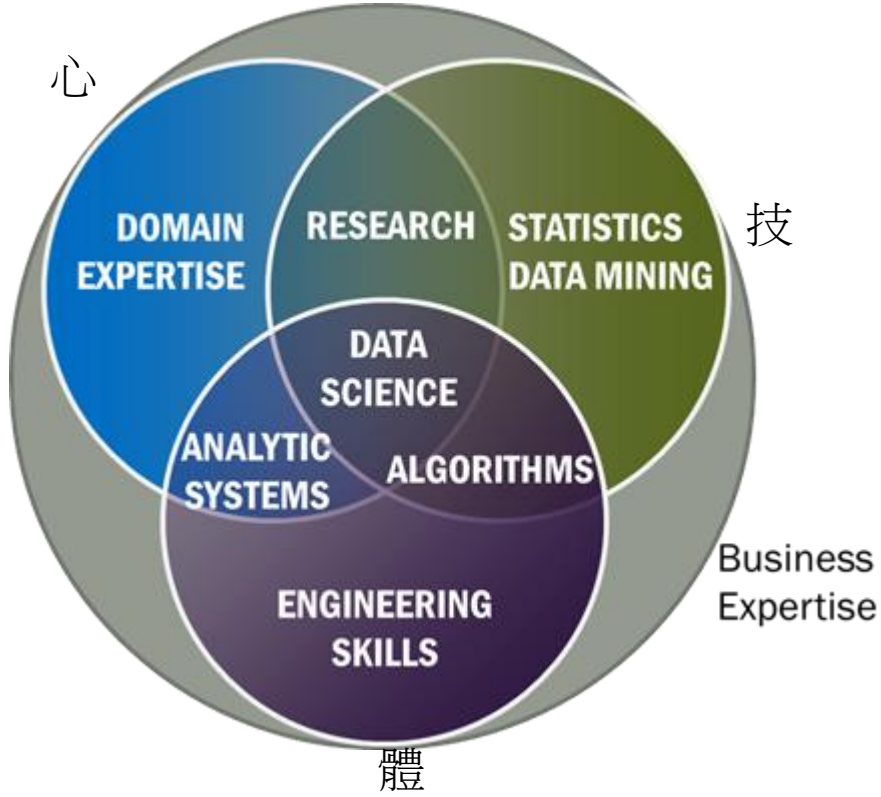
Yi-Fan Liou



Overture


城市風雲兒24
你到底怎麼贏的？
嗯…
鬼丸說我的鬥劍變成一個…
但怎麼會這樣？
因爲你心技體合一了。
認定八一八八號為織田信長御前賽之冠軍!!

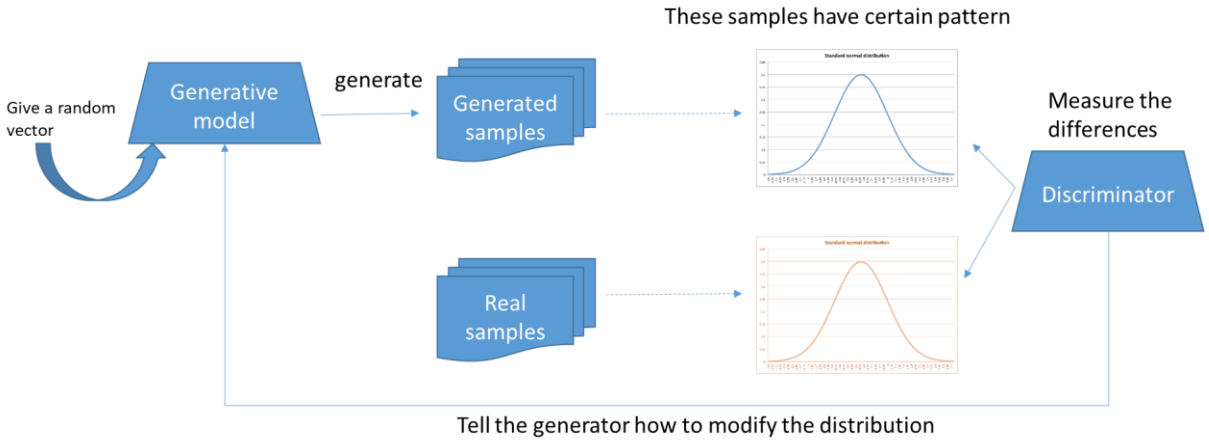
Mind, Skills and Body



http://semanticommunity.info/Data_Science/Data_Science_for_NIST_Big_Data_Framework

The aims of this workshop

- Enhancing the basic concepts of the math
  - Avoiding the “math signs”
  - “Math” is a represent form. But the real math could be found in the daily life
- Using the source codes for better understanding the concepts
  - Because many members of this workshop would be engineers. Using source codes would be more easy to communicate with them



The strong discriminator cannot improve the generators’ performances

About the speaker

- Education
  - Bioinformatics and Systems Biology
  - Biotechnology
  - Life Science

- Experiences
  - Postdoc fellow, NCKU
  - Data Scientist, Light Up Biotech. Corp.
  - Machine learning consultant, Bcondux Corp.
  - Algorithm Engineer, 京悅投資開發股份有限公司
  - Postdoc fellow, NCTU
  - Research Assistant, NCTU
  - 桃園市106年資訊組長初階及進階研習計畫 (Docker 助教)

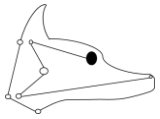
The agenda of this workshop

Day 1.

1. Basic usage of container
2. Installing Tensorflow
3. Essential operators
4. (The things of optimization)

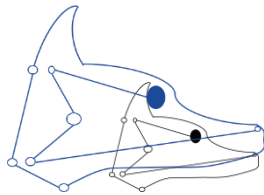
Day2

4. (The things of optimization)
5. Logistic regression
6. Multi-layer perceptron learning
7. Convolutional Neural network



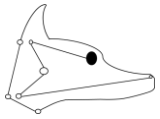
Bcondux

Docker



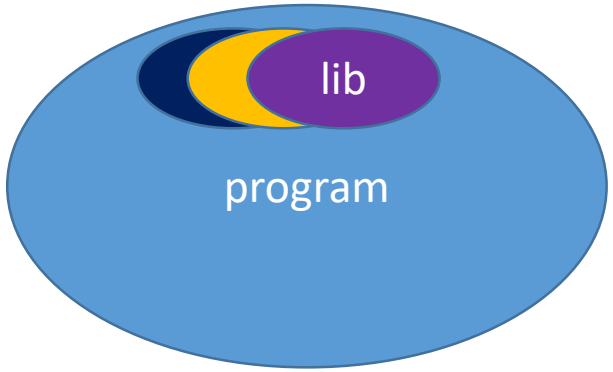
Why open this section with docker?

- Container becomes more and more important in Soft engineering.
  - CUDA and Tensorflow change very fast
- The concept of container are generally used in large scale machine learning
  - Data science
  - Data engineering



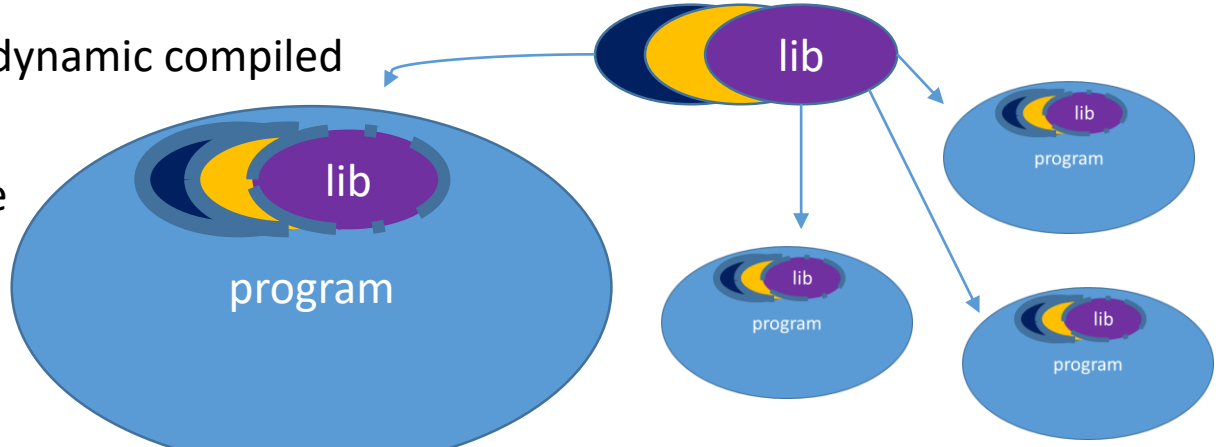
The concepts of containers

Static compiled

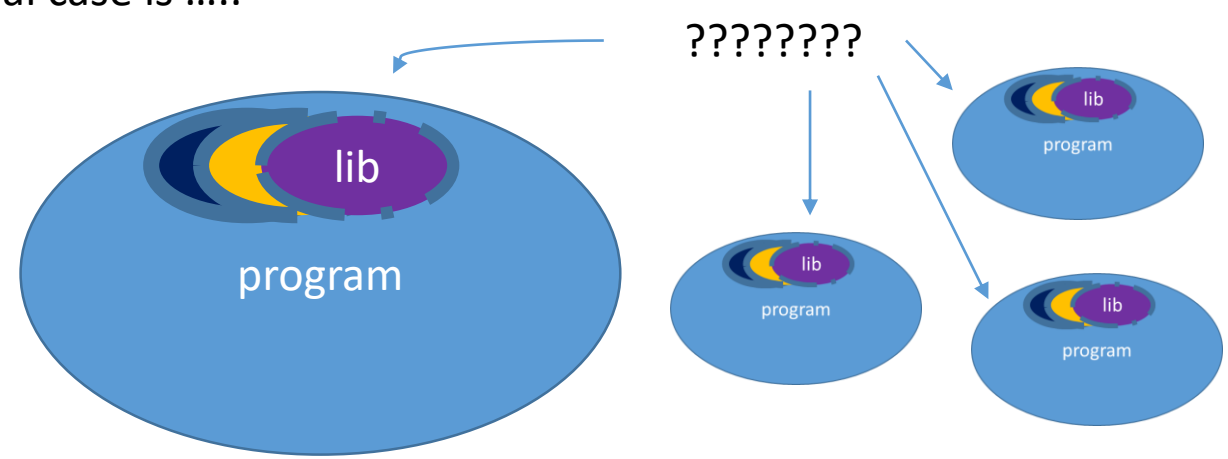


1. To save storage space
2. To reduce the cost of mantainace

dynamic compiled



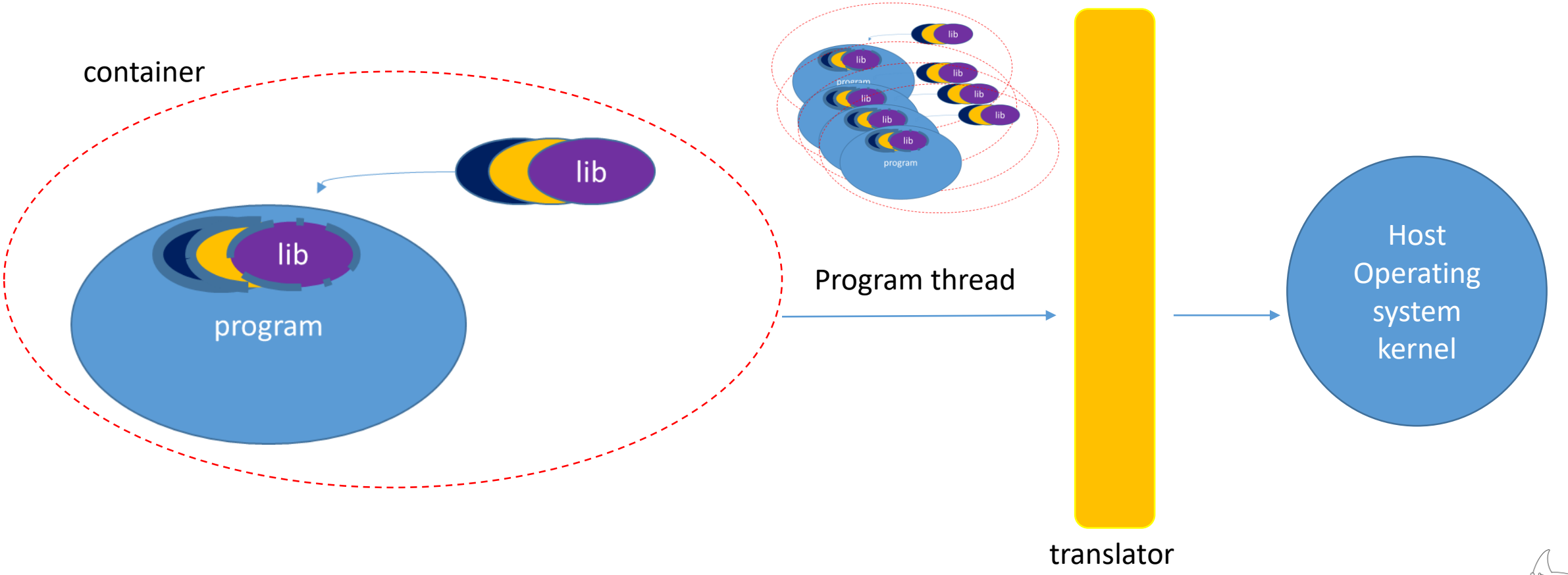
The real case is …..



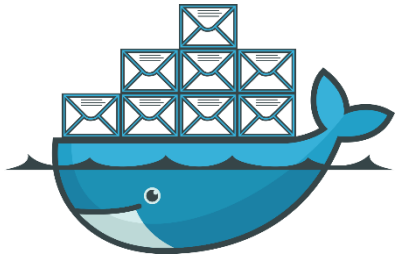
This can be more seen when use difference distro of Linux

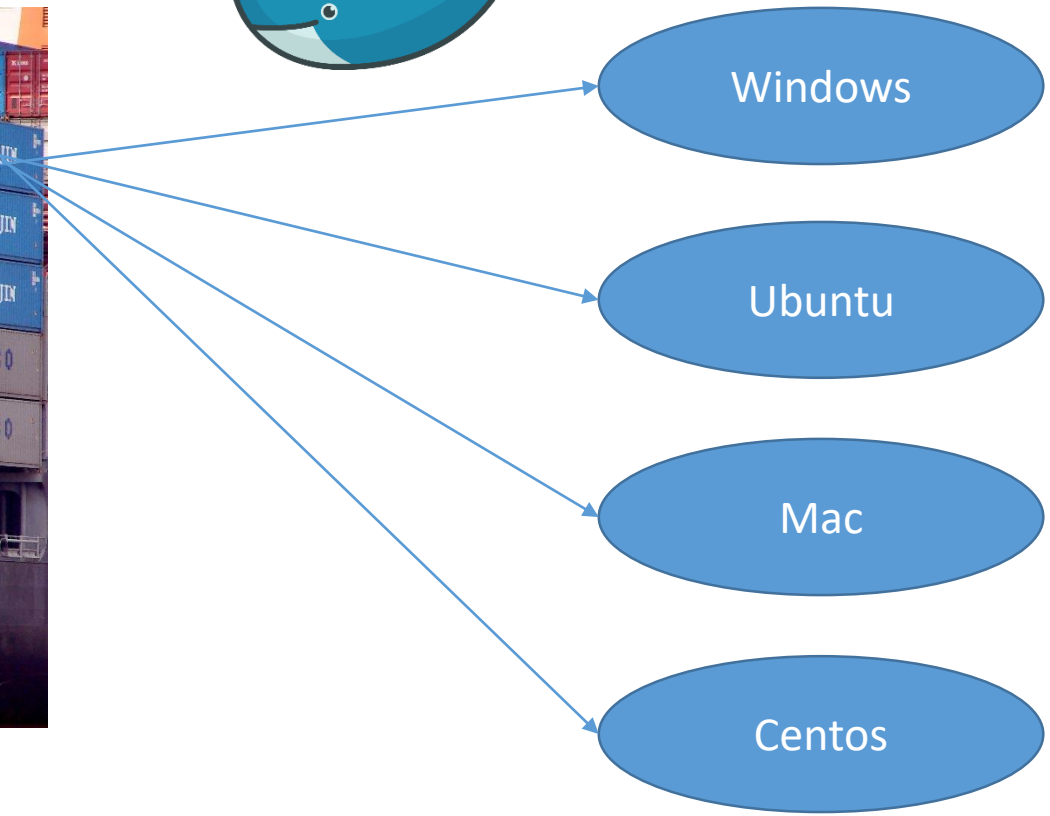
How about collect all the essential libraries?

- We can just collect the essential libraries for our program
- We abandon the needless library of the programs

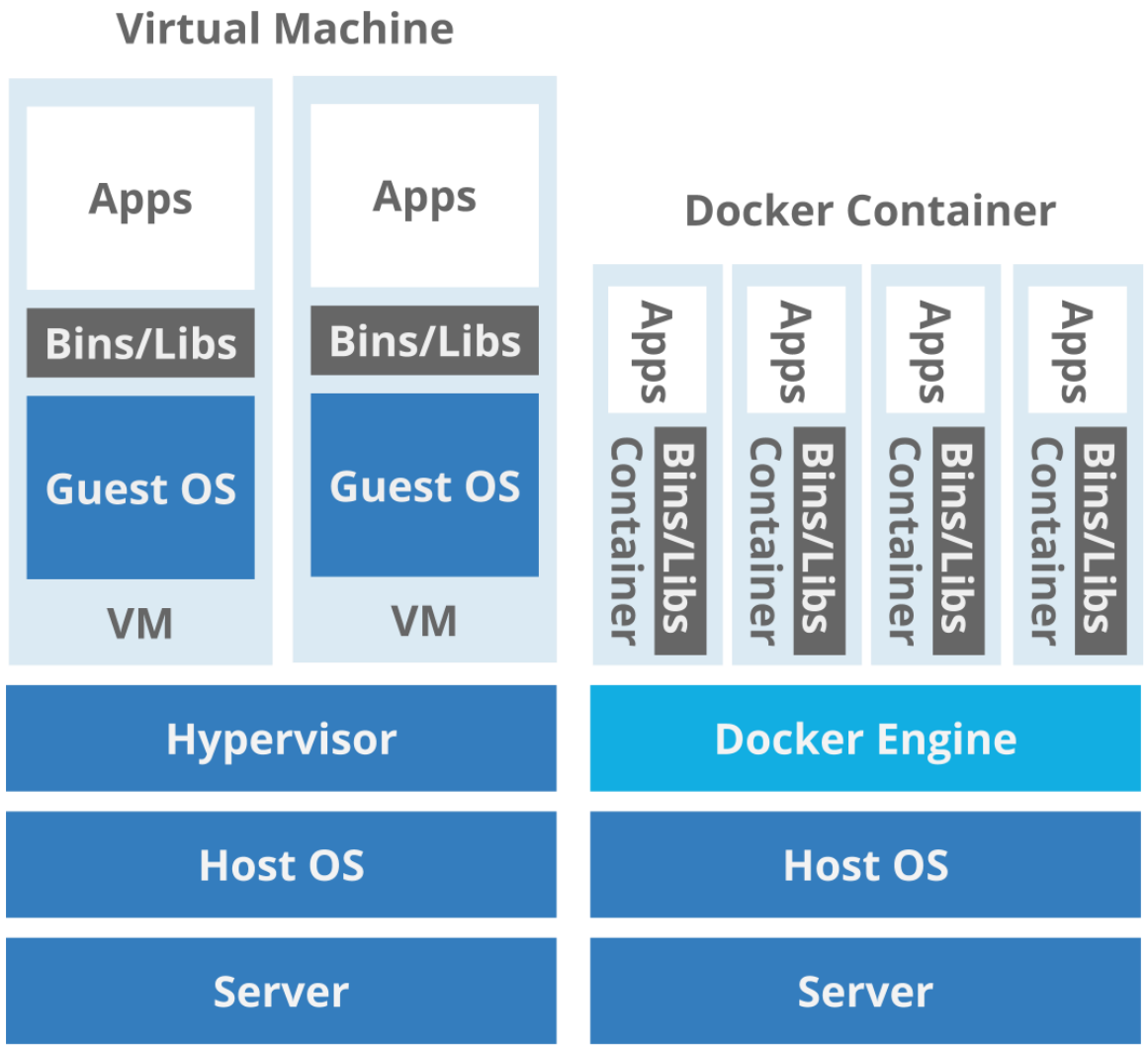


You can deliver your program regardless the operating system





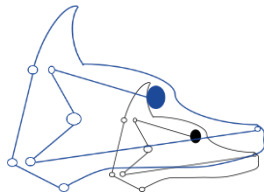
The concept of Docker



http://dockerdocs.org/what-is-docker/

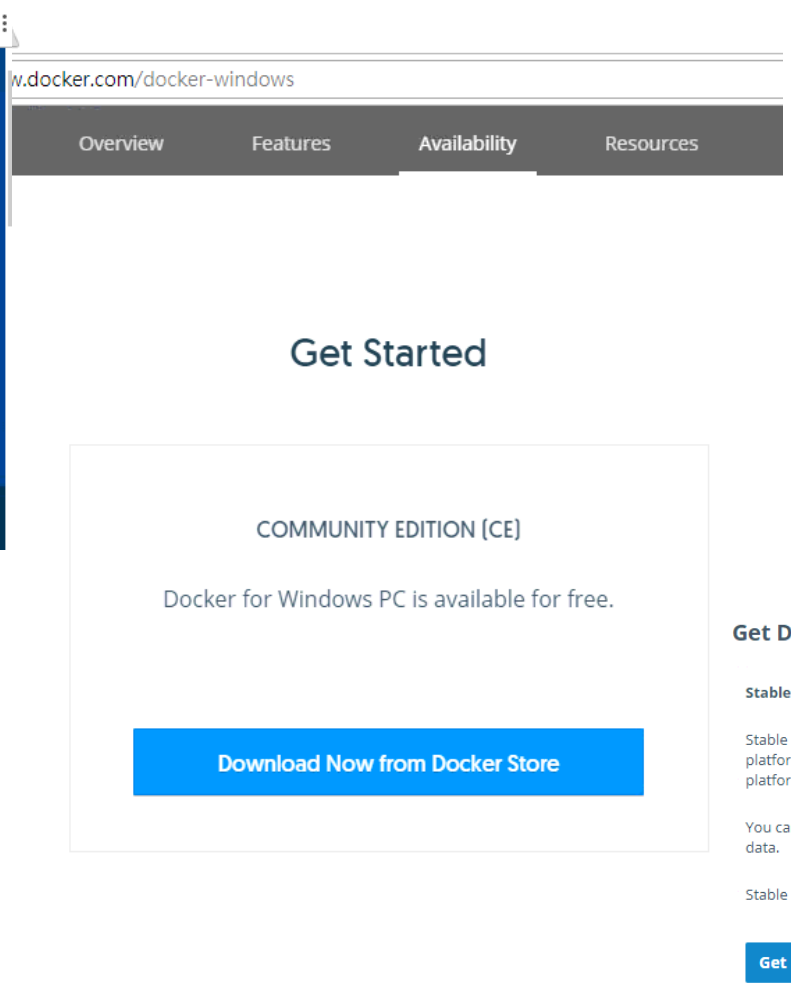
Bcondux

Docker on Windows



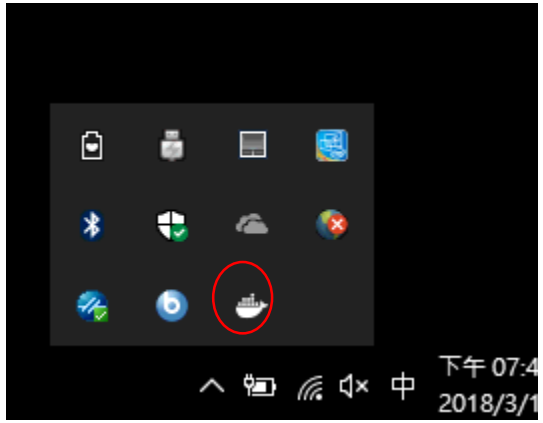
下載Docker-CE (windows version)

- https://www.docker.com/docker-windows



確認安裝docker

- Install the docker on windows
  - Only need click “next step”




```
命令提示字元
Microsoft Windows [版本 10.0.15063]
(c) 2017 Microsoft Corporation. 著作權所有，並保留一切權利。

C:\Users\User>docker version
Client:
 Version:       17.12.0-ce
 API version:   1.35
 Go version:    go1.9.2
 Git commit:    c97c6d6
 Built: Wed Dec 27 20:05:22 2017
 OS/Arch:       windows/amd64

Server:
 Engine:
  Version:      17.12.0-ce
  API version:  1.35 (minimum version 1.12)
  Go version:   go1.9.2
  Git commit:   c97c6d6
  Built:        Wed Dec 27 20:12:29 2017
  OS/Arch:      linux/amd64
  Experimental: true

C:\Users\User>
```

試跑一下Tensorflow in docker

- docker run -it gcr.io/tensorflow/tensorflow bash

```
C:\Users\User>docker run -it gcr.io/tensorflow/tensorflow bash
Unable to find image 'gcr.io/tensorflow/tensorflow:latest' locally
latest: Pulling from tensorflow/tensorflow
1be7f2b886e8: Pull complete
6fbc4a21b806: Pull complete
c71a6f8e1378: Pull complete
4be3072e5a37: Pull complete
06c6d2f59700: Pull complete
20a601a42386: Pull complete
d8967df06d5c: Pull complete
adbe4dda11a0: Pull complete
eeb8b3ca49ee: Pull complete
ab96b2cecaaa: Pull complete
e61c2ef48dde: Pull complete
50042b70c2f5: Pull complete
Digest: sha256:188bcda72801c3b756d483e3110a994567f7e3d5f197860279ae68cd2a94f97c
Status: Downloaded newer image for gcr.io/tensorflow/tensorflow:latest
root@f19867edaa26:/notebooks# python3
Python 3.5.2 (default, Nov 23 2017, 16:37:01)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow as tf
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named 'tensorflow'
>>> exit()
root@f19867edaa26:/notebooks# python
Python 2.7.12 (default, Dec  4 2017, 14:50:18)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow as tf
/usr/local/lib/python2.7/dist-packages/h5py/__init__.py:36: FutureWarning: Conversion of the second argument of issubdty
pe from `float` to `np.floating` is deprecated. In future, it will be treated as `np.float64 == np.dtype(float).type`.
  from ._conv import register_converters as _register_converters
>>> tf.__version__
'1.6.0'
>>>
```

There are still several versions of Tensorflow in container

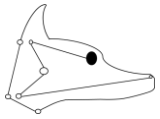
- *`TensorFlowCPUImage`* is required. It identifies the Docker container. Specify one of the following values:
  - `gcr.io/tensorflow/tensorflow`, which is the TensorFlow CPU binary image.
  - `gcr.io/tensorflow/tensorflow:latest-devel`, which is the latest TensorFlow CPU Binary image plus source code.
  - `gcr.io/tensorflow/tensorflow:`*`version`*, which is the specified version (for example, 1.1.0rc1) of TensorFlow CPU binary image.
  - `gcr.io/tensorflow/tensorflow:`*`version`*`-devel`, which is the specified version (for example, 1.1.0rc1) of the TensorFlow GPU binary image plus source code.

`gcr.io` is the Google Container Registry. Note that some TensorFlow images are also available at dockerhub.

The pre-build Tensorflow containers (Bondux)

- Tensorflow 1.6 with python3
  - The original version from Google use python2
  - GPU
    - bcondux/python3-tensorflow:gpu
  - CPU
    - bcondux/python3-tensorflow:gpu

- C:\Users\User>docker run -it -v C:/Users/User/Desktop:/workspace bcondux/python3-tensorflow:cpu bash
  - Remember to set the “shared folder”



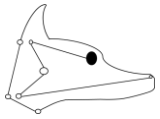
More information

- Philipz
  - https://www.gitbook.com/book/philipzheng/docker_practice/details
  - https://www.slideshare.net/philipzh/docker-77631136

DGX series and NVIDA for deep learning

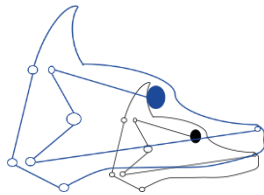
- If you use the DGX system or running with some NVIDA products, this sections is for you
- A high efficiency computing task need two stuffs
  - Hardware
  - Software

- Since you have NVIDA GPUs (sometimes, you did not have any choice about this…)
  - Are you sure you already use the “best performance” of you NVIDIA GPU?



Bcondux

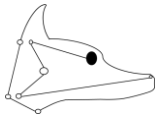
The Basic of Tensorflow



Bcondux

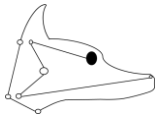
Topics

- How to install Tensorflow in Windows
  - Without GPU – installing cuda in windows usually have trouble
  - The CUDA install in Windows will just use the “Note”
- The basic concepts of how to use Tensorflow



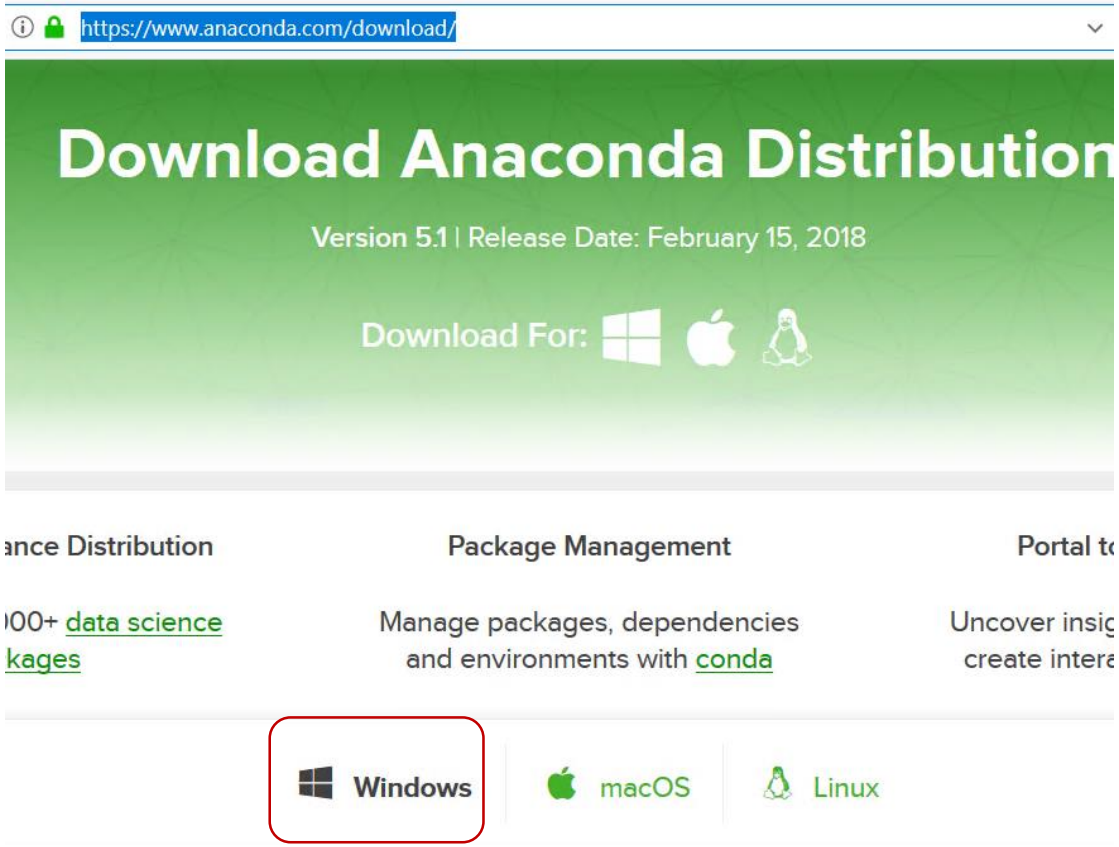
CUDA install in Windows

- Essenstial
  - Visual studio 2015 (community version)
    - Tip1: VS2015 only!! 2017 will be out of work
    - “community version” is good, and FREE!!
  - CUDA 9
    - Please notice “where the SKD will be installed”
  - CUDNN 8
    - You just need to “copy and paste”
    - According to where is the CUDA be installed



Installing Tensorflow - Anaconda

- Anaconda is a scientific tool package
  - Including Numpy, matplotlib etc.
  - Several requests are satisfied by only one click !
  - https://www.anaconda.com/download/



Anaconda 5.1 For Windows Installer

Python 3.6 version *

Download

64-Bit Graphical Installer (537 MB)

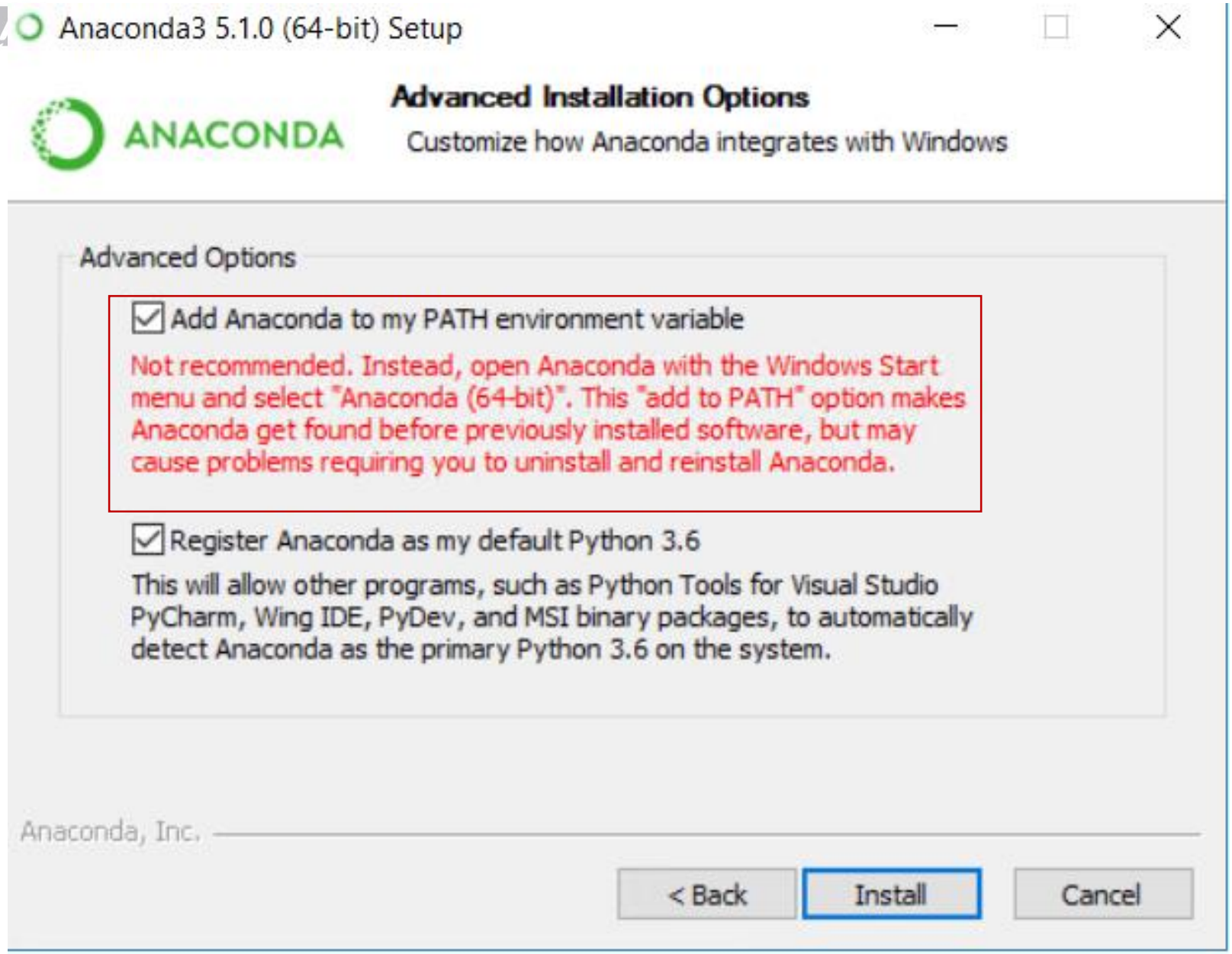
32-Bit Graphical Installer (436 MB)

Python 2.7 version *

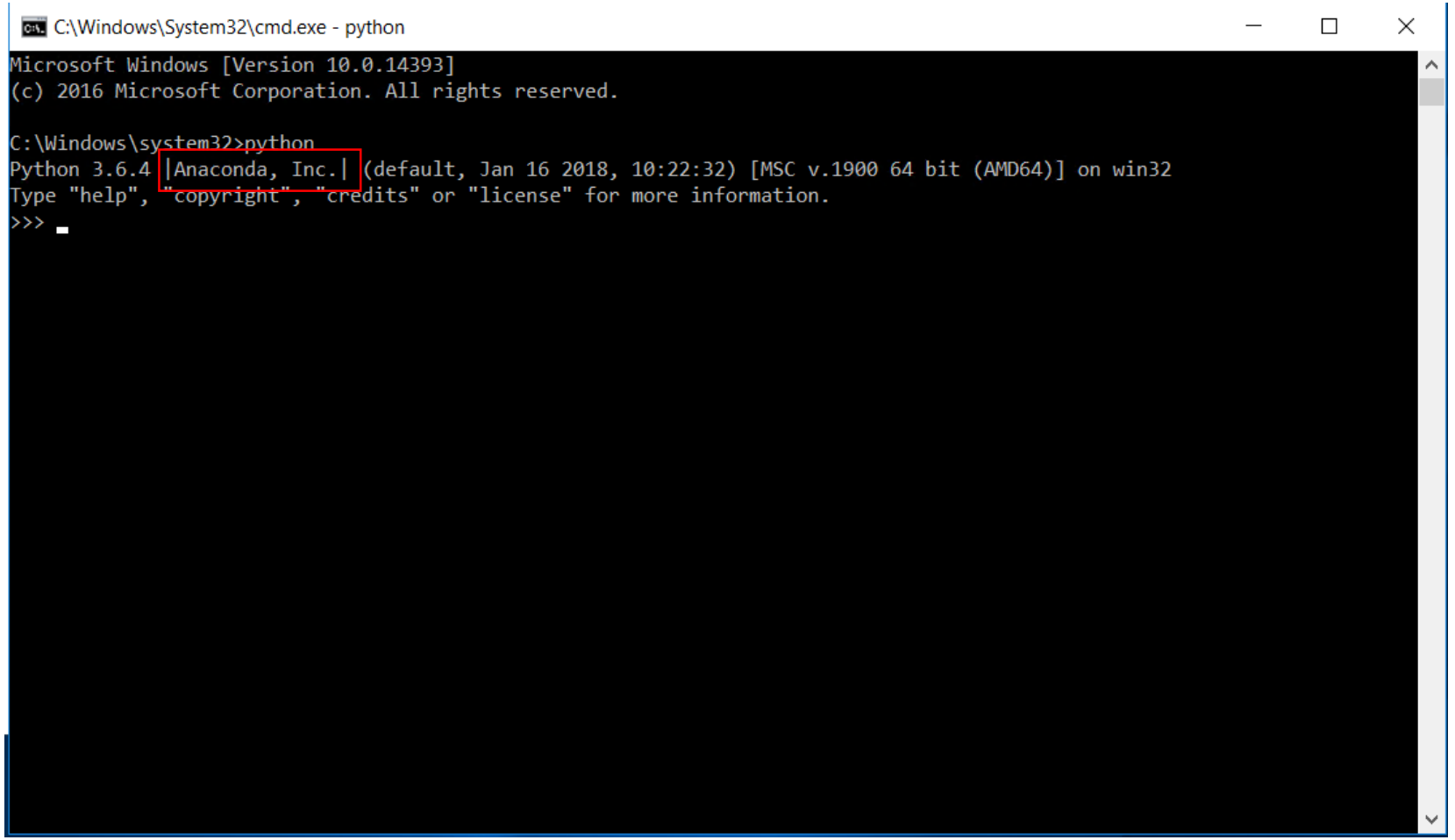
Download

64-Bit Graphical Installer (523 MB)

32-Bit Graphical Installer (420 MB)


Anaconda3 5.1.0 (64-bit) Setup
ANACONDA
Advanced Installation Options
Customize how Anaconda integrates with Windows
Advanced Options
Add Anaconda to my PATH environment variable
Not recommended. Instead, open Anaconda with the Windows Start menu and select "Anaconda (64-bit)". This "add to PATH" option makes Anaconda get found before previously installed software, but may cause problems requiring you to uninstall and reinstall Anaconda.
Register Anaconda as my default Python 3.6
This will allow other programs, such as Python Tools for Visual Studio PyCharm, Wing IDE, PyDev, and MSI binary packages, to automatically detect Anaconda as the primary Python 3.6 on the system.
Anaconda, Inc.
< Back
Install
Cancel

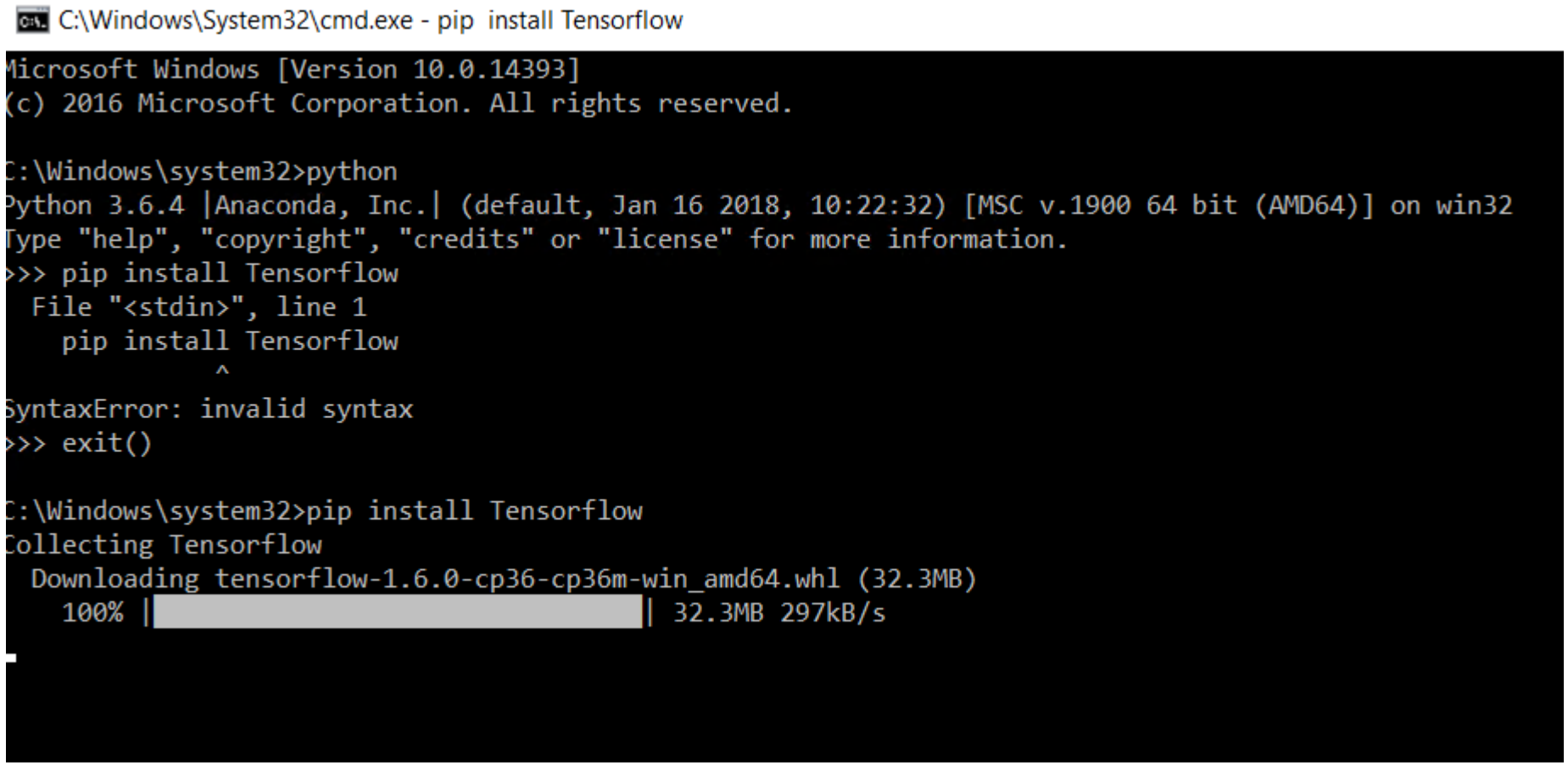
Test if you Anaconda is successfully installed



Please make sure “Anaconda” is shown on the CMD window

Installing Tensorflow – Tensorflow framework

- C:\pip install Tensorflow



Test if the Tensorflow is installed successfully

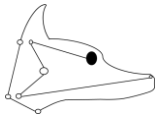
- Open the python interactive interface
  - Import tensorflow as tf
  - tf.__version__

```
Python 3.6.4 |Anaconda, Inc.| (default, Jan 16 2018, 10:22:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import  tensorflow as tf
C:\Users\yifanliou\Anaconda3\lib\site-packages\h5py\__init__.py:36: FutureWarning: Conversion of the second argument of
issubdtype from `float` to `np.floating` is deprecated. In future, it will be treated as `np.float64 == np.dtype(float).
type`.
  from ._conv import register_converters as _register_converters
>>> tf.__version__
'1.6.0'
>>>
```

There should be no error messages showing!!

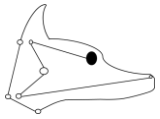
Bcondux

Basic operation of Tensorflow

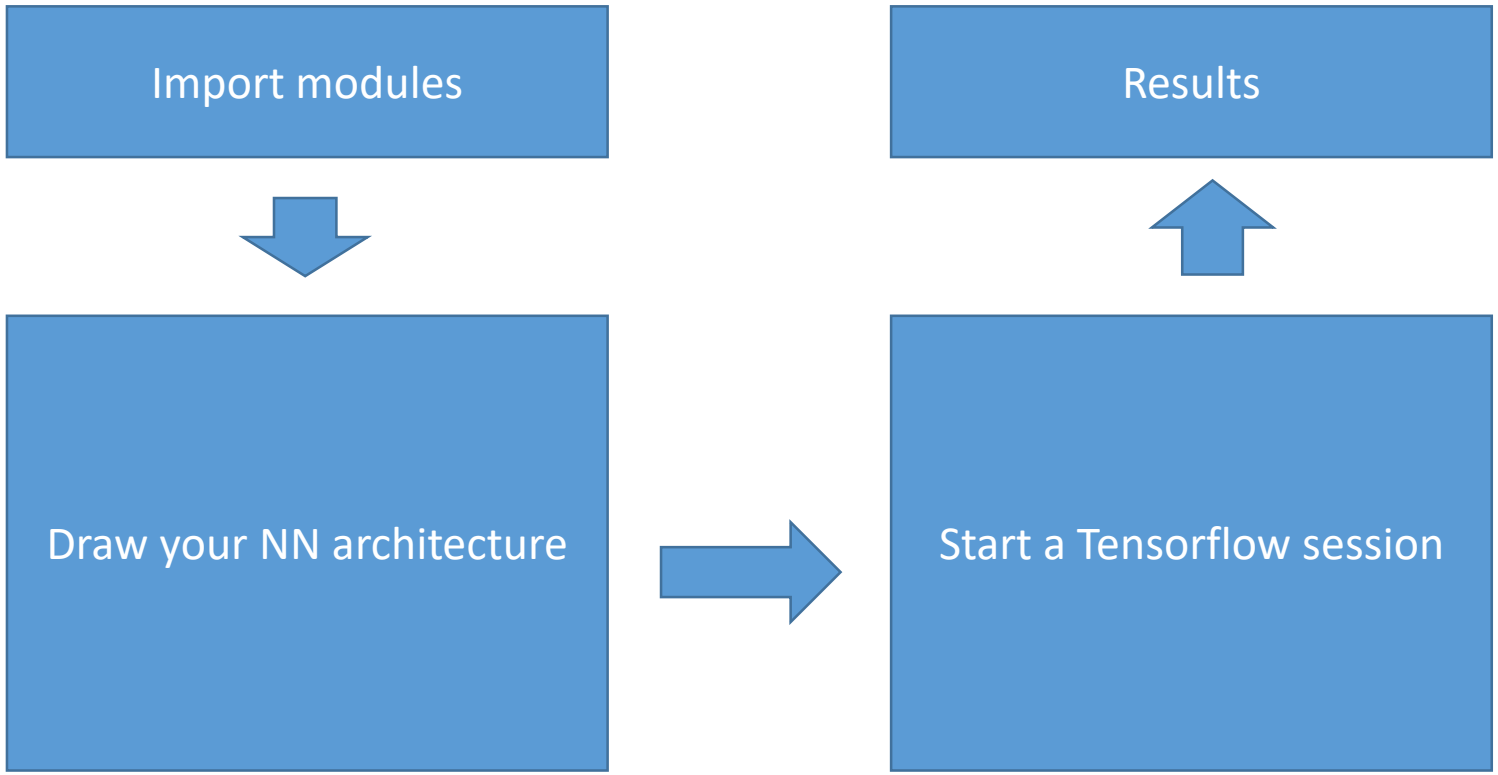


The basic concept of Tensorflow

- Static computation graph
  - You need to define the whole graph before you run a complete process
  - The framework will deploy your job to the corresponding computing nodes
- By the way
  - Dynamic computational graph
    - pyTorch
    - Eager execution



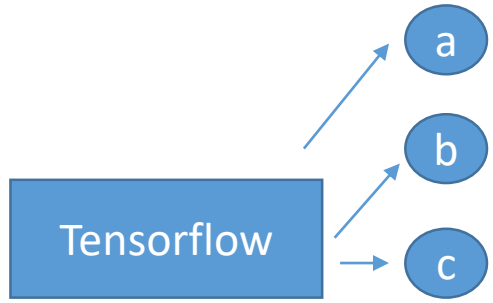
Process diagram



Session is a “canvas”

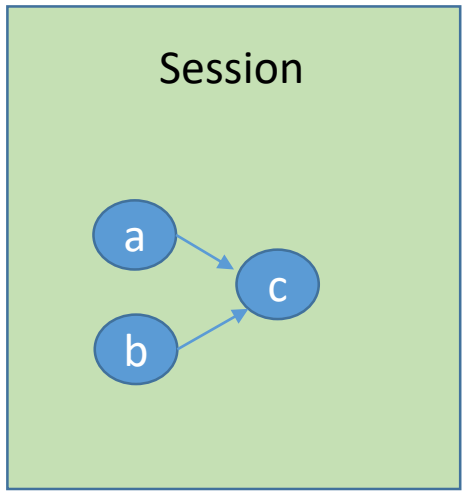
Import tensorflow as tf

a = tf.Variable([1,2,3])
b = tf.Variable([4,5,6])
c = a + b



They are just occupy in the python

sess = tf.session()
sess.run(c)
sess.close()



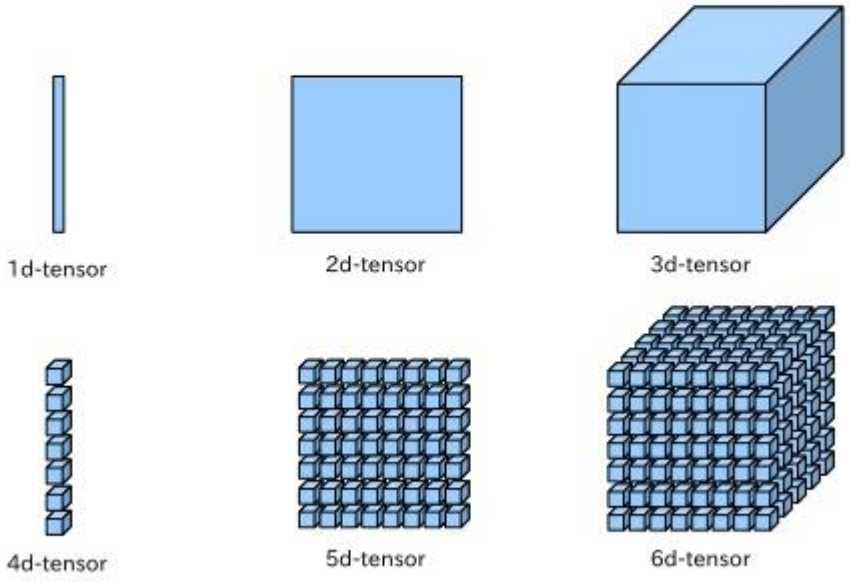
Tip:

1. Once the Session announced, the architecture fixed.
2. Once the operators announced, it will live until you close Python (advance)
3. Since the architecture is fixed, we usually use ‘ delivery pores’ for feeding the data.

What is “tensor” ?

What's Tensor

Tensor is a general name of multi-way array data. For example, 1d-tensor is a vector, 2d-tensor is a matrix and 3d-tensor is a cube. We can image 4d-tensor as a vector of cubes. In similar way, 5d-tensor is a matrix of cubes, and 6d-tensor is a cube of cubes.



February 2, 2012 7/26

“A tensor is a generalization of vectors and matrices to potentially higher dimension”

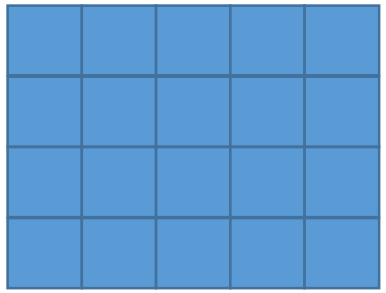
--https://www.tensorflow.org/programmers_guide/tensors

https://stackoverflow.com/questions/37849322/how-to-understand-the-term-tensor-in-tensorflow

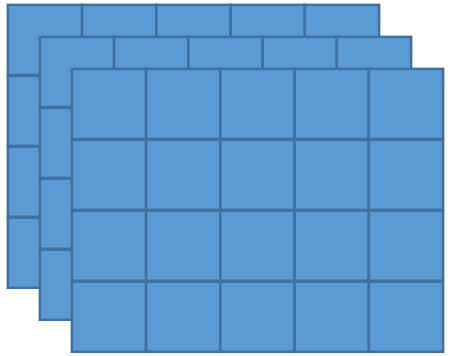
The shape of tensor

[5]

[4,5]



[3,4,5]



The basic operation in tensor calculating

We will compare the basic operator between python and Tensorflow

- Since python did not have an exact ‘array’, we use the “list” to instead

add

tf.add(x,y)

multiply

tf.matmul(x,y)

tf multiply(x,y)??

How about “weighted sum” ?

tf.reduce_sum(tf.matmul(x,y))

Example 1.

Task : we use Tensorflow to complete some simple operation

1. Add 2 tensors
2. Multiply 2 tensors
3. Division 2 tensors

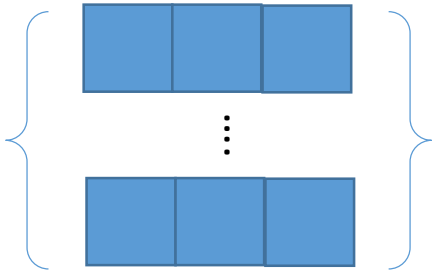
Practice 1. Weighted Sum

Assume we have a simple linear function:

$$F(x) = 2.5\ X_1 + 1.4\ X_2 + 0.2\ X_3 + 4.5\ X_4$$

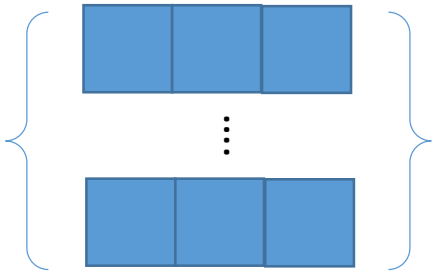
Placeholder and feed dictionaries

p2 = tf.placeholder(dtype=tf.float32, shape=[800,3])



800

p1 = tf.placeholder(dtype=tf.float32, shape=[None,3])



dynamic

https://cs224d.stanford.edu/lectures/CS224d-Lecture7.pdf

Than you for your attention

Day 2

The agenda of this workshop

Day 1.

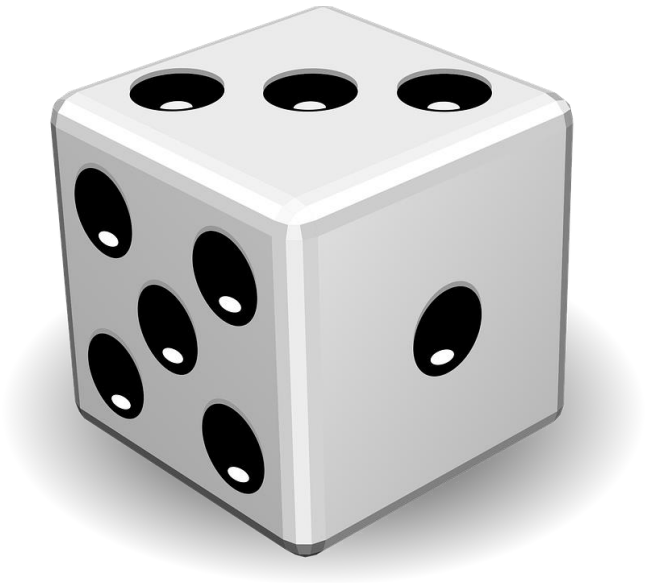
1. Basic usage of container
2. Installing Tensorflow
3. Essential operators
4. (The things of optimization)

Day2

4. (The things of optimization)
5. Logistic regression
6. Multi-layer perceptron learning
7. Convolutional Neural network

The small story of probability

A general example of Dice



If this is a fair dice:

1 -> 1/6
2 -> 1/6
3 -> 1/6
4 -> 1/6
5 -> 1/6
6 -> 1/6

Status 1:
I want to get (1)
=> 1/6

Status 2:
I want to get (1,1)
=> 1/6 * 1/6 = 1/36

Status 3:
I want to get (1,2)
=> 1/6 * 1/6 = 1/36

So…
You just need to multiply the probability, everything would be solve

How about the commuters

Suppose : you have two ways for commuting

1. Walk
2. Bus

You won't use those methods equally

	Walk	Bus
Probability	3/5	2/5

If you want to have a list that the probabilities of all ways:

All you need is multiply this small table

	Walk	Bus
Probability	3/5	2/5

X

	Walk	Bus
Probability	3/5	2/5

	Walk	Bus
Walk	9/25	6/25
Bus	6/25	4/25

If the relations of commuting and weathers are recorded

Suppose : you have two weathers for commuting

1. Rainy
2. Sunny

	Rainy	Sunny
walk	2/5	3/5
bus	4/7	3/7

A small story

I have a nerd friend, he is good at investigating things. He always asks me to buy food for him.
When he get the food, he also ask me what's the weather and how do I buy the food.

One day, I told him I buy food by walking, but I do not tell him the weather.
So…

		Rainy	Sunny
1	walk	2/5	3/5
0	bus	4/7	3/7

He directly guess the weather is good!!

Then, I am angry….

The angry nerd

another day, I told him it is raining. but I do not tell him the how I buy food.
So…

	1	0
	Rainy	Sunny
walk	2/5	3/5
bus	4/7	3/7

He directly guess I am by bus.

To be a real nerd. This time I told him I have 2/3 change by walking
So…

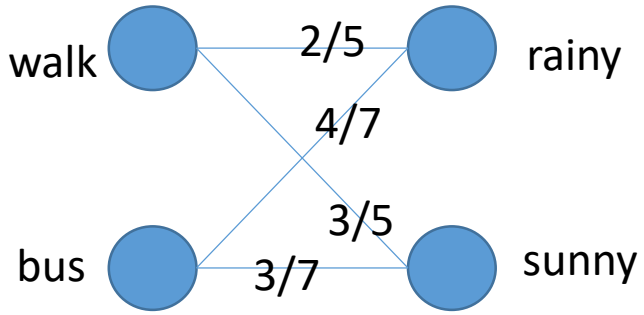
		0.46	0.54
		Rainy	Sunny
2/3	walk	2/5	3/5
1/3	bus	4/7	3/7

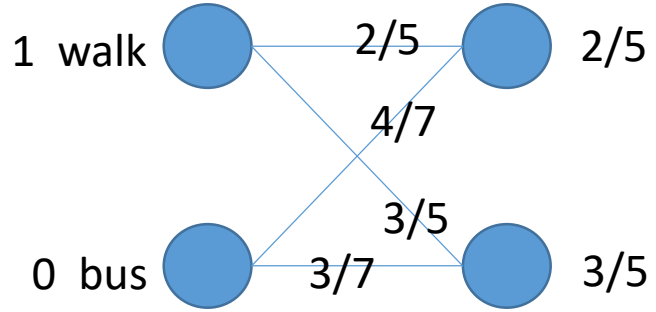
He directly guess it’s sunny day

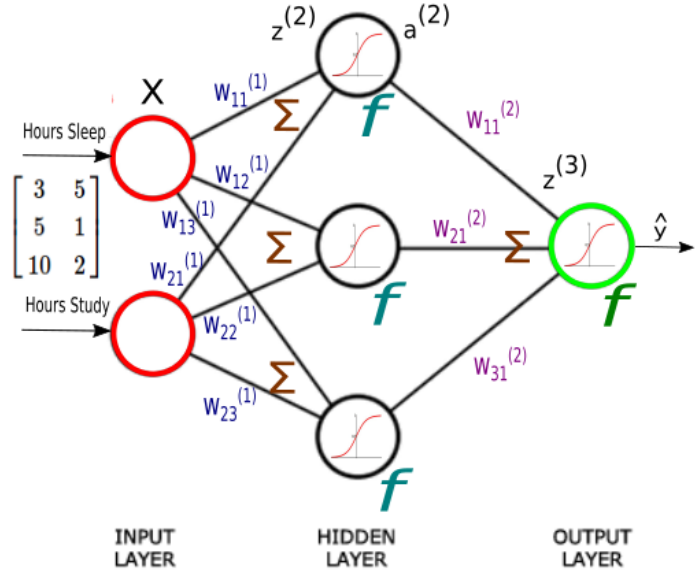
From the boring story, we had a small conclusion that …

This table have “translating” ability.

The representing using graph

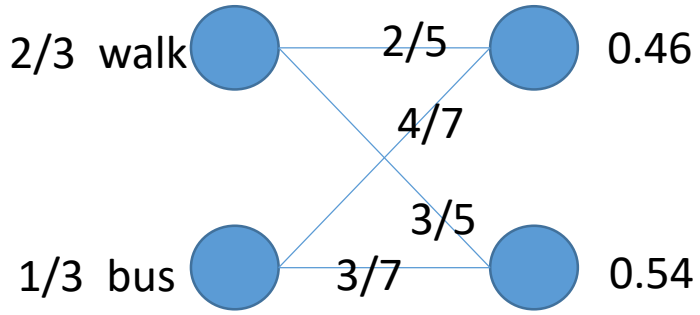






http://www.bogotobogo.com/python/scikit-learn/Artificial-Neural-Network-ANN-1-Introduction.php

	Rainy	Sunny
walk	2/5	3/5
bus	4/7	3/7



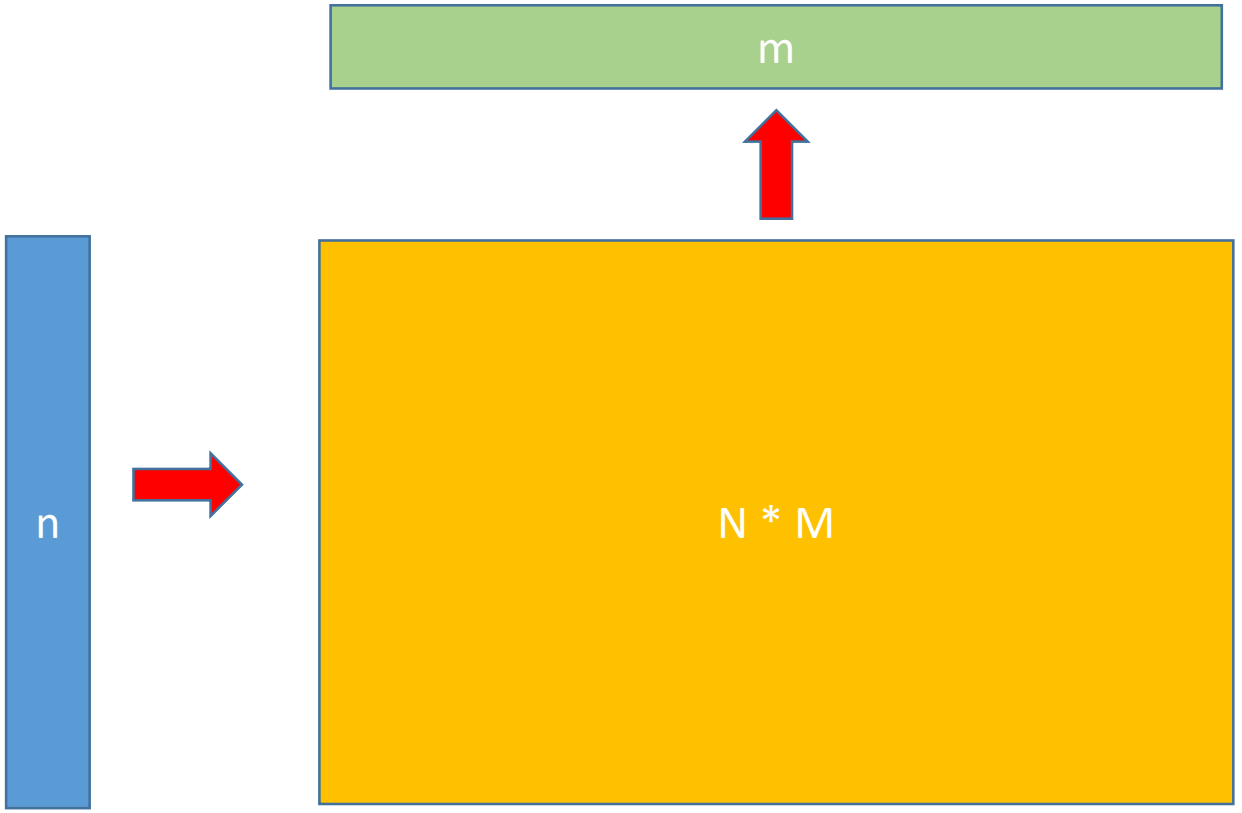
If you have 2 input nodes and 3 output nodes

⇒ You need 2*3 probability (or weights)

If you have 3 input nodes and 1 out node

=> You need 3*1 probability (or weights)

The simple illustration



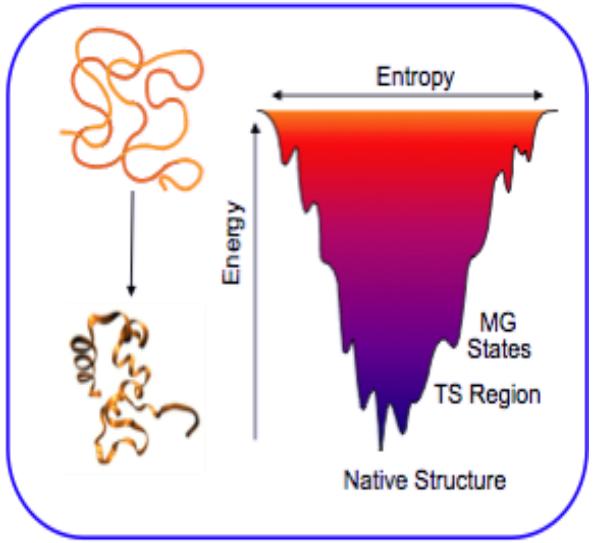
The weight can be approximate using optimization

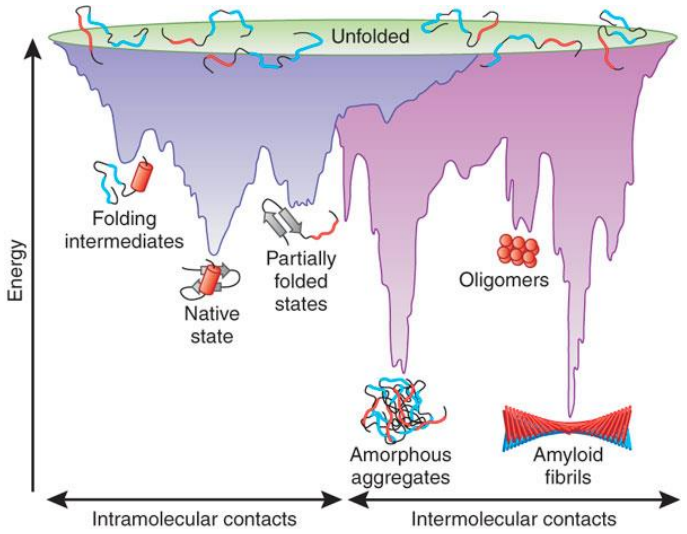
Simple things about optimization

Yi-Fan Liou

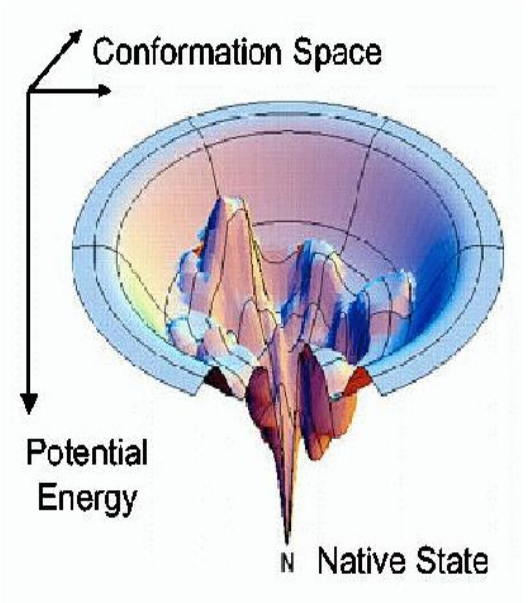
Optimized solution search landscape

Use protein folding as example

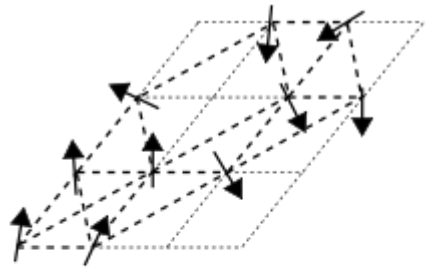




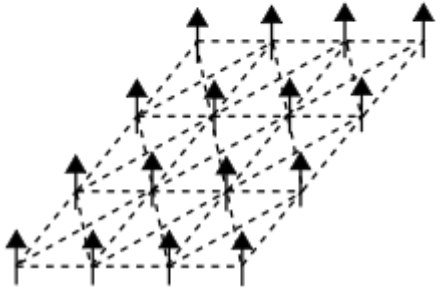
http://www.nature.com/nsmb/journal/v16/n6/fig_tab/nsmb.1591_F1.html



https://parasol.tamu.edu/groups/amatogroup/research/computationalBio/slide/EnergyLandscape.gif

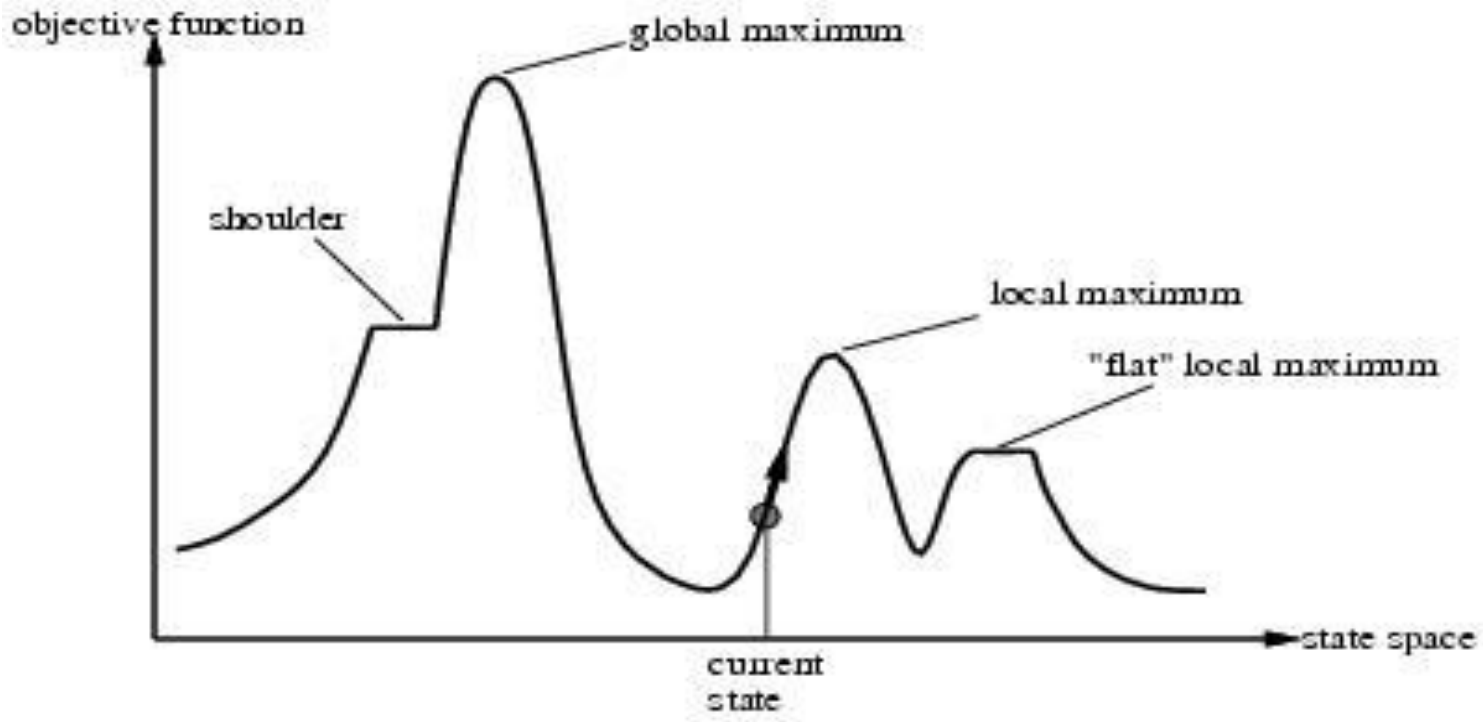


Spin glass



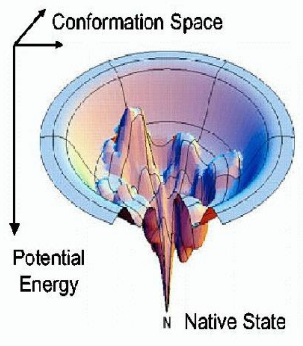
https://en.wikipedia.org/wiki/Spin_glass

“Landscape” of search



The problems to look for solutions

- The only way to get the best solution is to scan all the space.
  - This will take long time.
- If we cannot find the best solution, the acceptable solution would be desired.
  - Traditional method (numerical analysis based)
  - Heuristic algorithm (random based)



Gradient Descent

Assume we have some cost-function: $C(x_1,\ldots,x_n)$
and we want minimize over continuous variables X1,X2,..,Xn

1. Compute the *gradient :* $\frac{\partial}{\partial x_i} C(x_1,\ldots,x_n) \qquad \forall i$

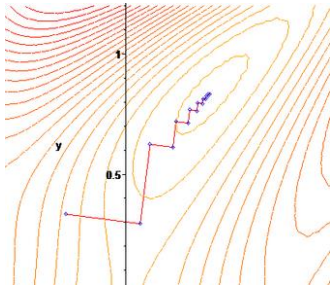
2. Take a small step downhill in the direction of the gradient:

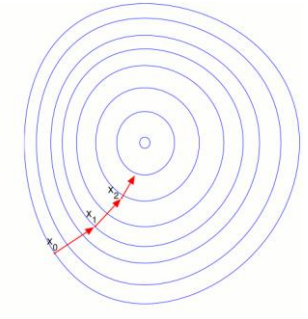
$$x_i \to x'_i = x_i - \lambda \frac{\partial}{\partial x_i} C(x_1,\ldots,x_n) \qquad \forall i$$

3. Check if $C(x_1,..,x'_i,..,x_n) < C(x_1,..,x_i,..,x_n)$

4. If true then accept move, if not reject.

5. Repeat.

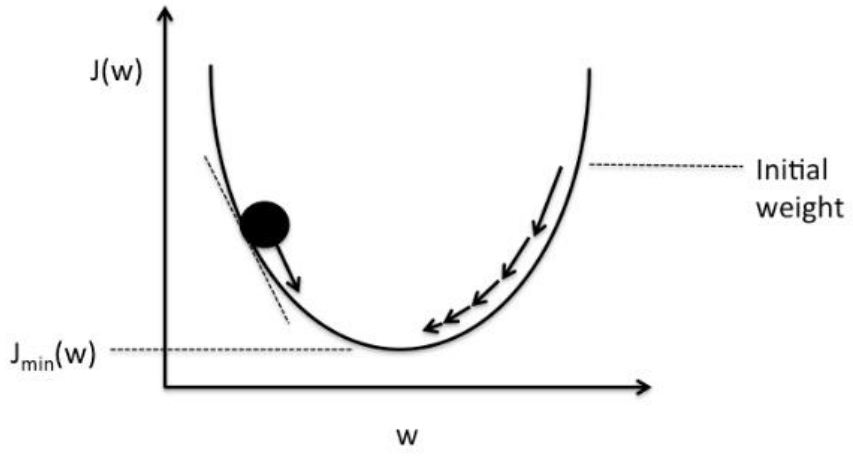




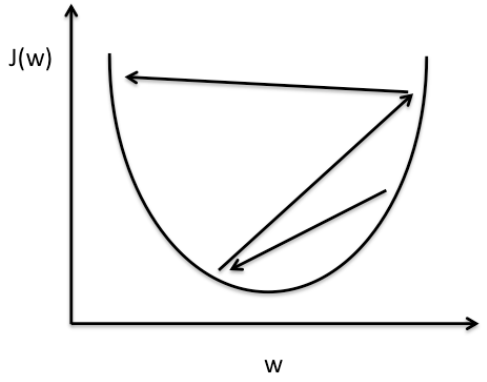
https://www.ics.uci.edu/~welling/teaching/.../LocalSearch271f09.pdf

Problems

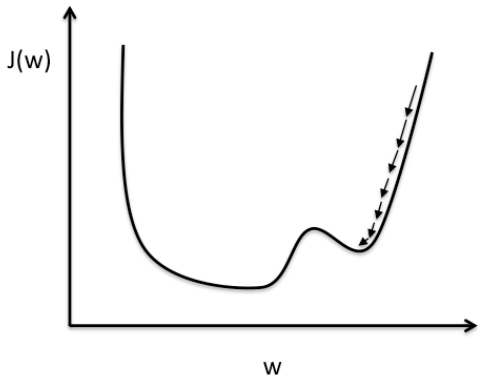
- Gradient decent and learning rate



Schematic of gradient descent.



Large learning rate: Overshooting.



Small learning rate: Many iterations until convergence and trapping in local minima.

http://sebastianraschka.com/Articles/2015_singlelayer_neurons.html

Basic operating the Tensorflow using simple linear regression

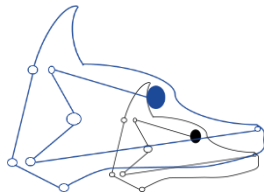
Task : find a good parameters to estimate the data point

Assignment:

1. Make a function and sample some points of this functions
2. Add a little noise to these points
3. Using gradient decent method to approximate this function from these data points with noise

Bcondux

Logistic Regression



Why

Some concepts in machine learning is also from the statistics.

They are quite similar!!

You can find the logit models in:

1. Traditional statistics – ex. Survival analysis
2. Machine learning

Logistic regression – from statistic

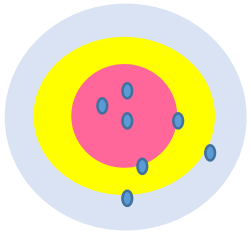
Set: All the dependent variables are linear

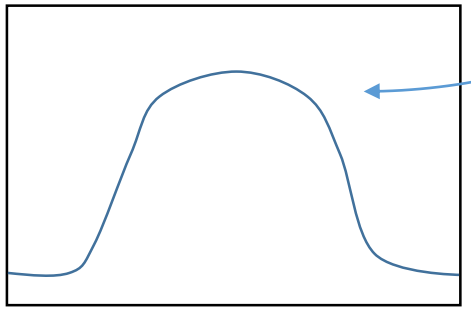
continuous : least square errors

categorical : continuous **with some errors**

Y = a1x1 + a2x2 + …. + anxn + b

Ideally



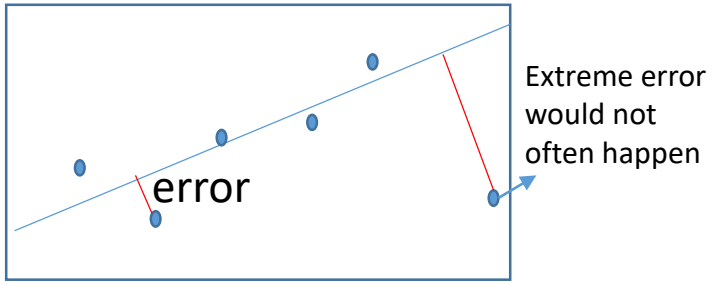


Normal distribution

Y = a1x1 + a2x2 + …. + anxn + b

Never forget : the least square assume the **normal distribution**

This concept make a straight line



Logistic regression – from statistic

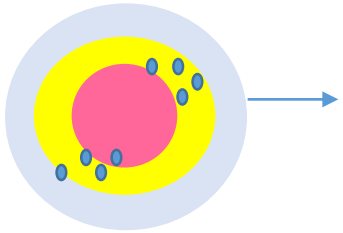
Set: All the dependent variables are linear

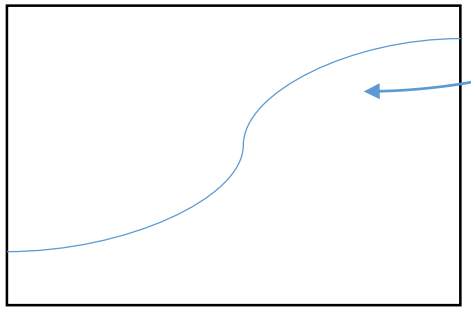
continuous : least square errors

categorical : continuous **with some errors**

$Y = a1x1 + a2x2 + ... + anxn + b$

Classification case





NOT Normal distribution

$Y = a1x1 + a2x2 + ... + anxn + b$

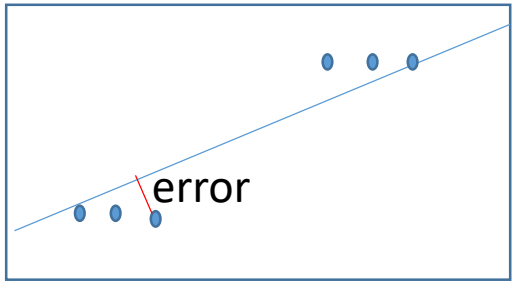
If you want to fit this line ....

This concept also make a straight line

It will be like this ....

It is obviously : the errors between the line and data

won't be normal distribution



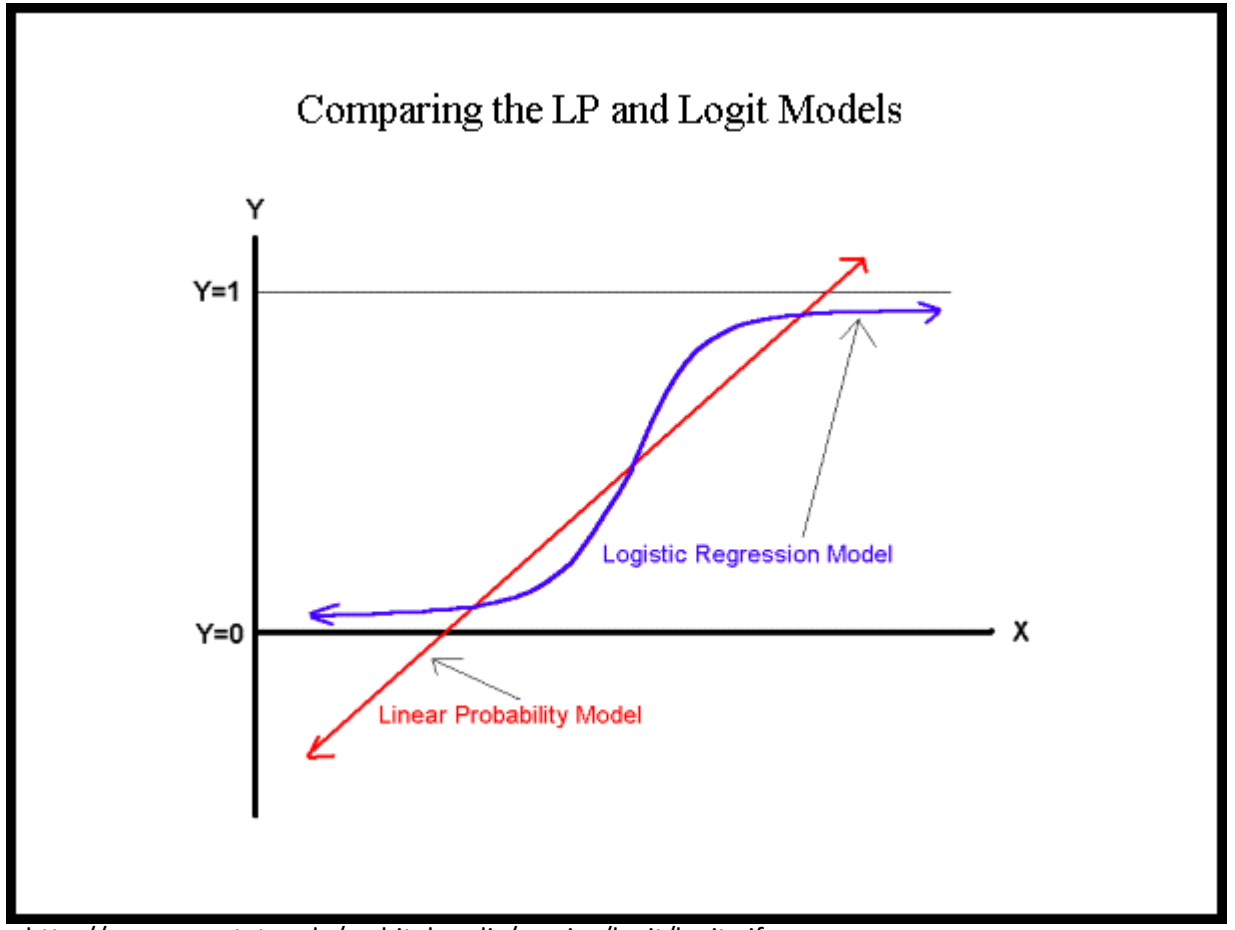
That's why the classification problem never use MSE as loss

(you can use MSE, but it will make a tragedy)

Logistic regression – from statistic

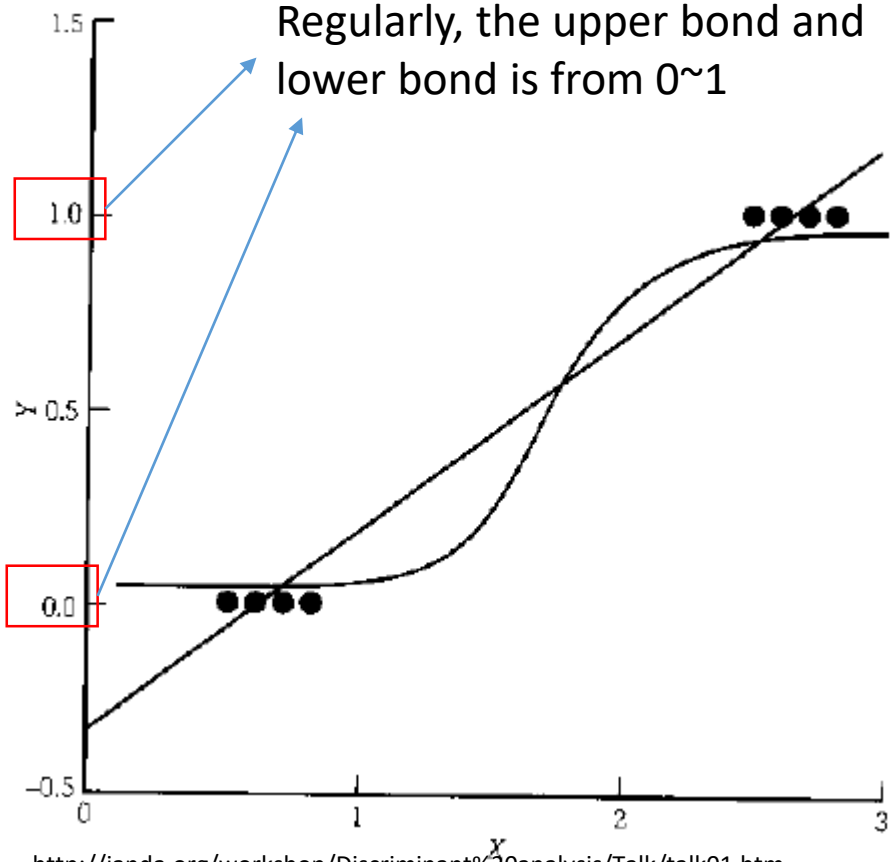
To solve this problem, we already know the distribution of data will not be normal distribution…

SO……



http://www.appstate.edu/~whiteheadjc/service/logit/logit.gif

Logistic regression – from statistic



http://janda.org/workshop/Discriminant%20analysis/Talk/talk01.htm

Regularly, the upper bond and lower bond is from 0~1

Why not use “ratio”?

$$log(P/(1-P)) = \sum_{k=i}^{n} a_i x_i + b$$

case

Control
(total samples except the cases)

Odds ratio

According to “maximum likelihood”, using the optimization method can get the odds ratio

1. Newton
2. Gradient decent

Maximum likelihood V.S. cross entropy

- In machine learning, the loss usually use **cross entropy**
- In statistics, the gain usually use **maximum likelihood**
- But Don’t worry, they are similar …

Alarm !!!!! Math time~~

The relation between BCE and ML

- BCE = binary cross entropy ML = maximum likelihood
- Set : the problem is simple as **bi-classification**
  - The ML can applied as Bernoulli

Bernoulli

$$p(y|\theta) = \prod_{i=1}^{n} \theta_i^{yi}(1-\theta_i)^{1-yi}$$

This is a distribution from model, so..
Let p(x|θ') denote the training model θ

$$p(y|x,\theta') = \prod_{i=1}^{n} p\theta'(y|x_i)^{yi}(1-p\theta'(y|x_i))^{1-yi}$$

How about give a 'log' ?

$$f(\theta;x,y) = \sum_{i=1}^{n} y_i \, log \, p_{\theta'}(y|xi) + (1-y_i)\log(1-p_{\theta'}(y|xi))$$

$$\mathcal{L}(\theta) = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(p_i) + (1-y_i)\log(1-p_i)]$$

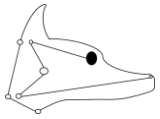
BCE

https://stats.stackexchange.com/questions/260505/machine-learning-should-i-use-a-categorical-cross-entropy-or-binary-cross-entro

The math time is over ~~~

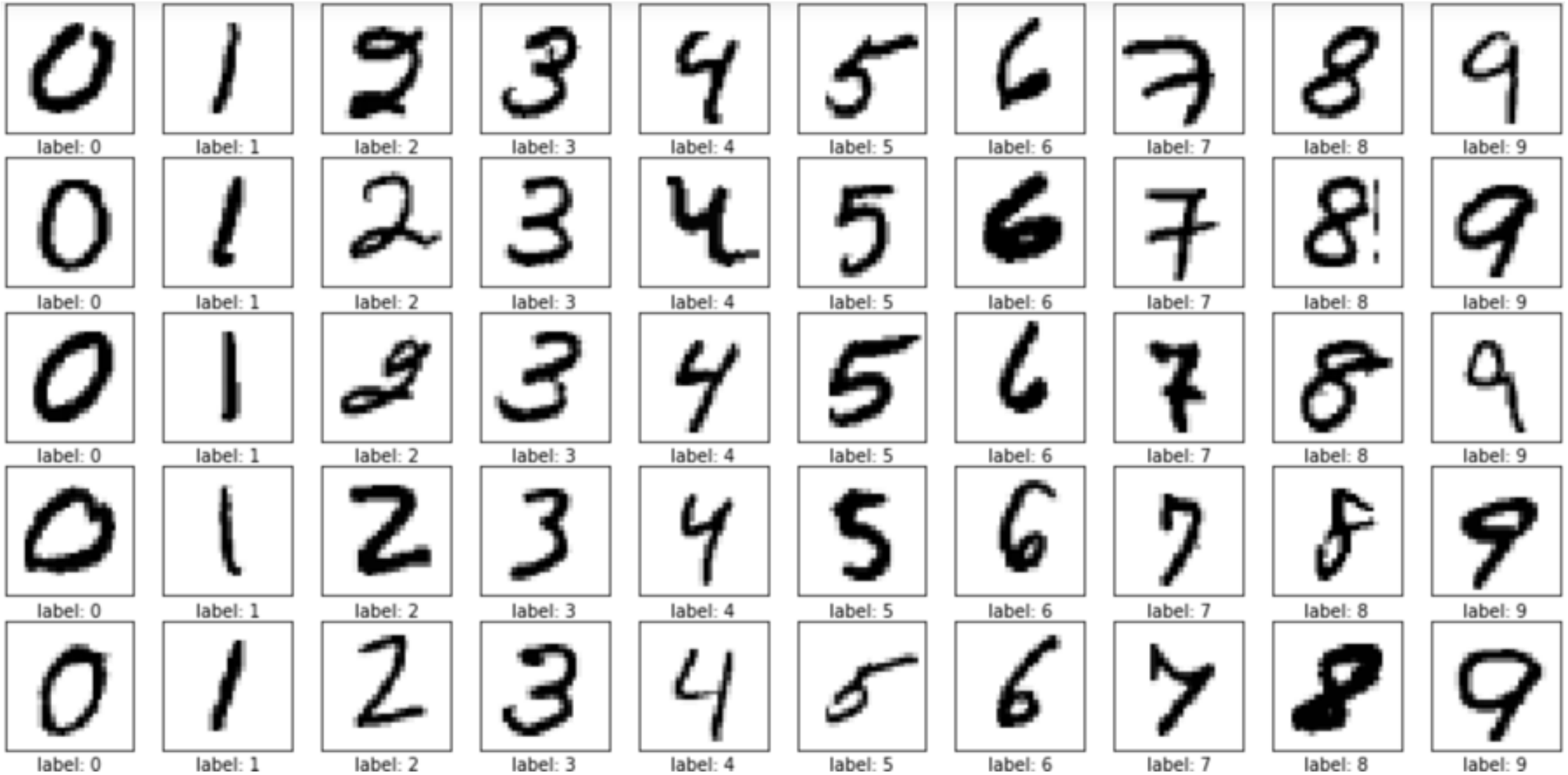
- Alarm release….
- The conclusion is that
  - Using BCE is similar to use ML
  - Most often … they are the same
- The assignments
  - Use the MNIST dataset as example. Make a logit model for prediction the

$$z = w_0 x_0 + w_1 x_1 + \ldots + w_m x_m = \sum_{i=0}^{m} w_i x_i = \mathbf{w}^T \mathbf{x}.$$

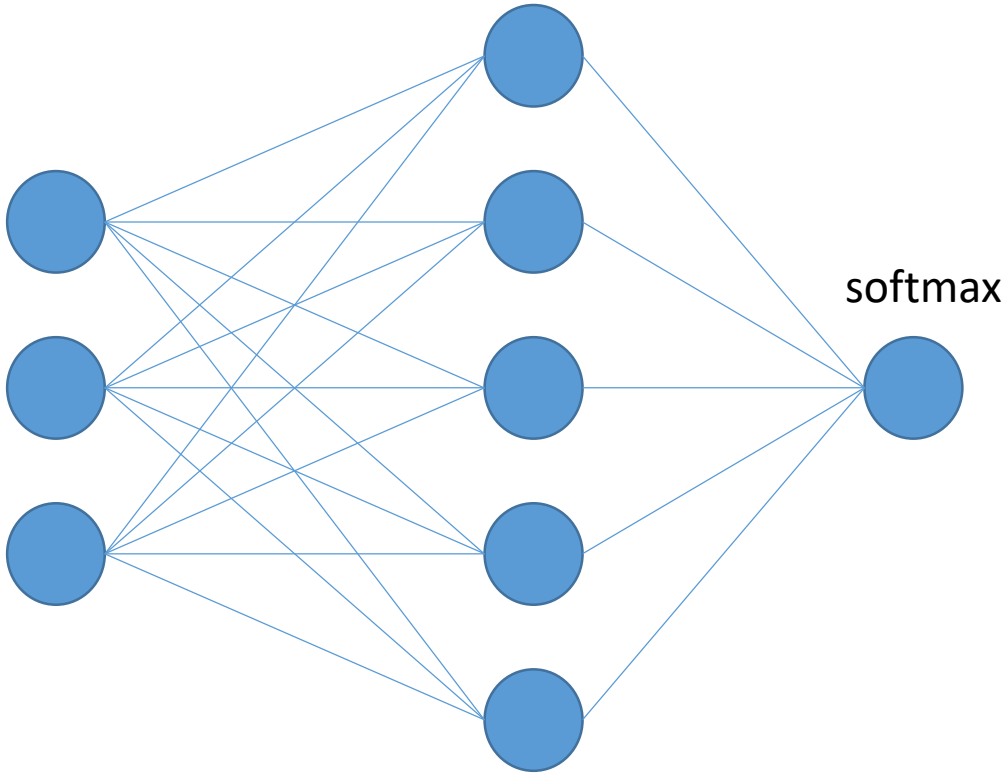


Bcondux

MNIST



The architecture of Logit model

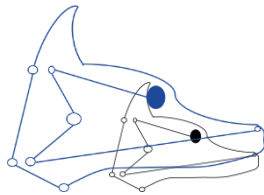


$$z = w_0 x_0 + w_1 x_1 + \ldots + w_m x_m = \sum_{l=0}^{m} w_l x_l = \mathbf{w}^T \mathbf{x}.$$

Bcondux

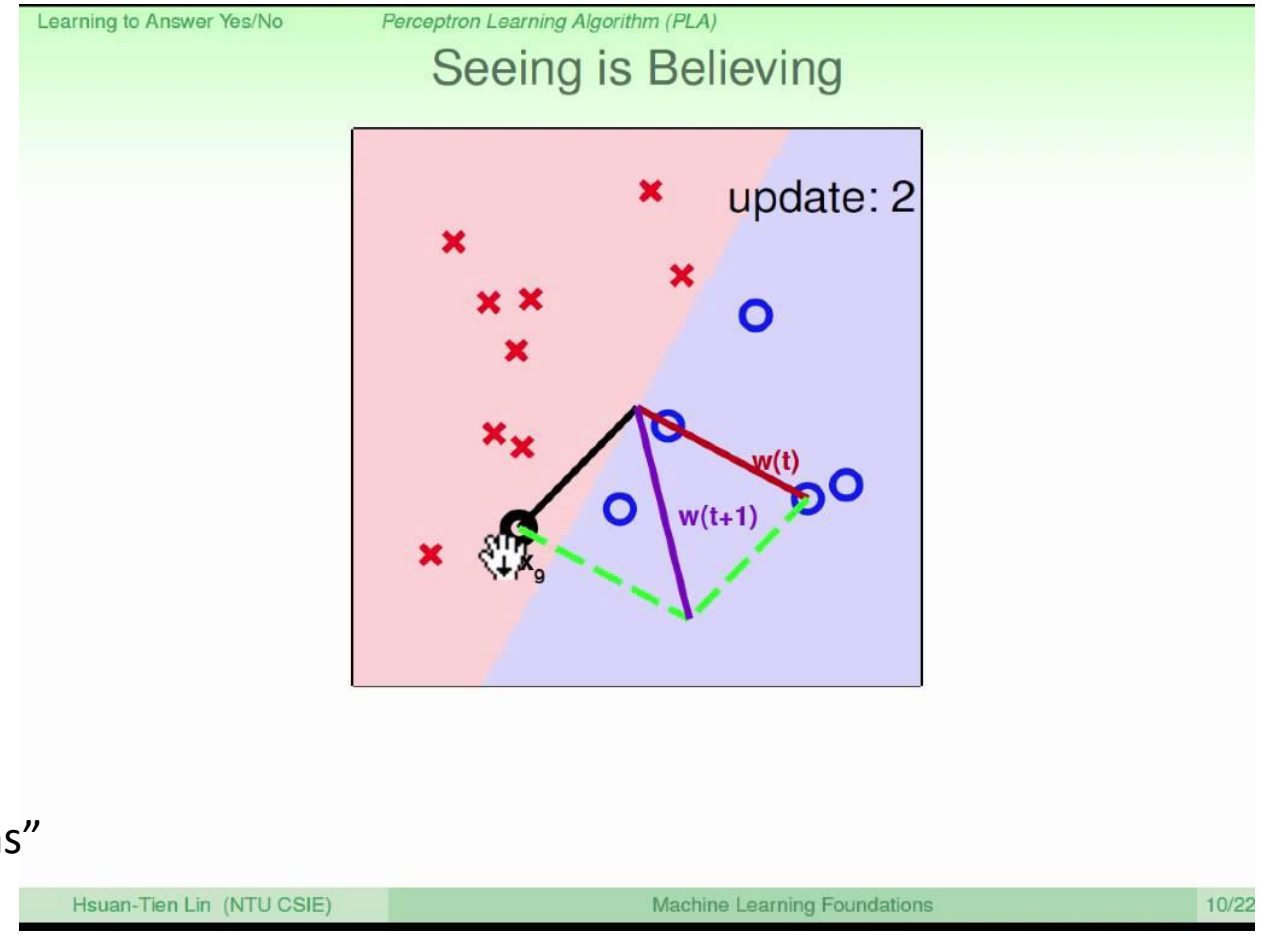
Multilayer perceptron

Yi-Fan Liou



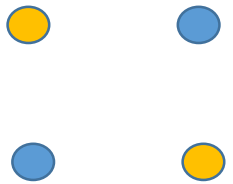
Small story – Perceptron learning

Target: We want to use a simple line as decision boundary



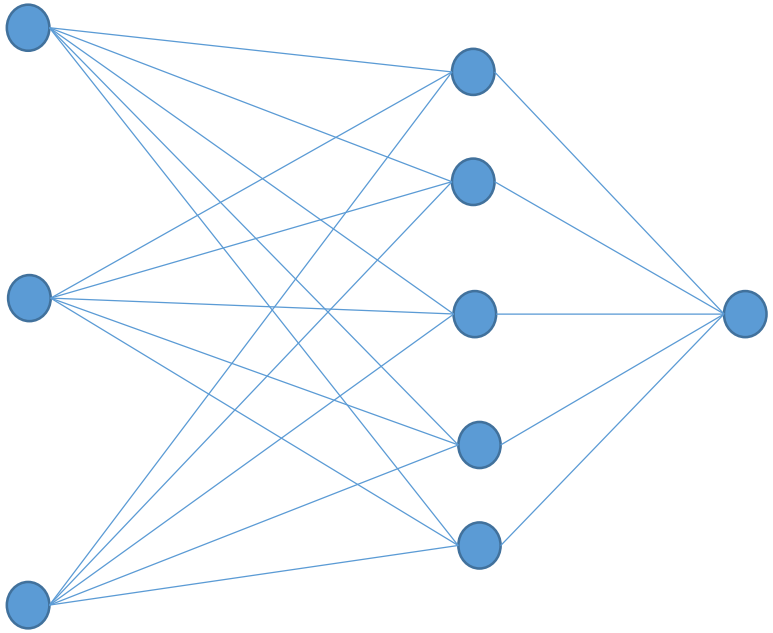
H.T. Lin, “Machine Learning Foundations”

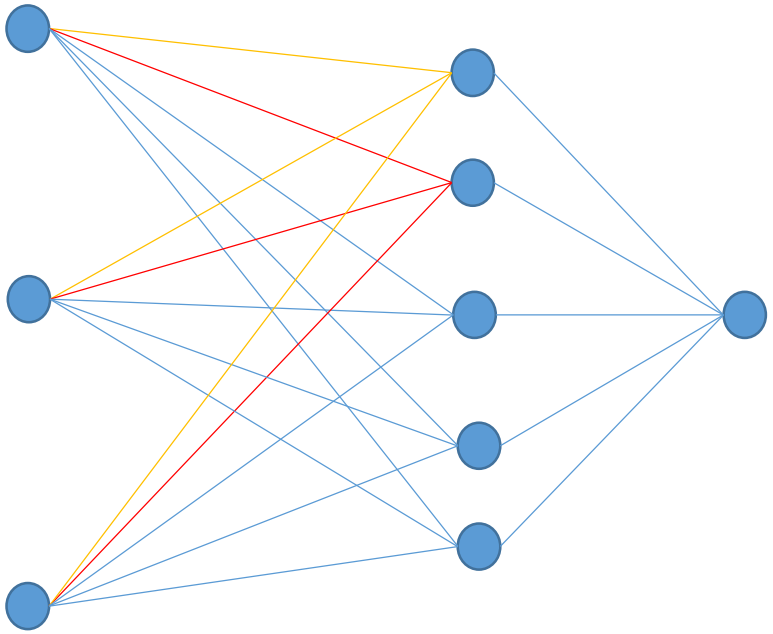
XOR problem



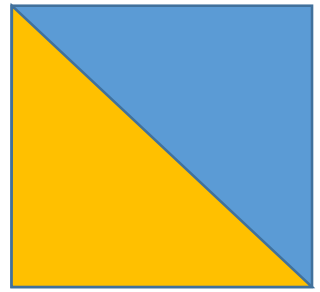
"Perceptrons: An Introduction to Computational Geometry", Minsky and Papert

Multi-layer perceptron

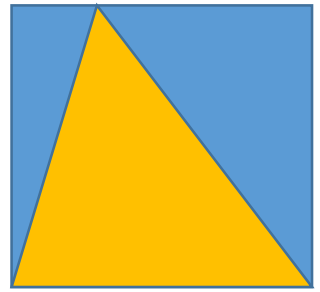




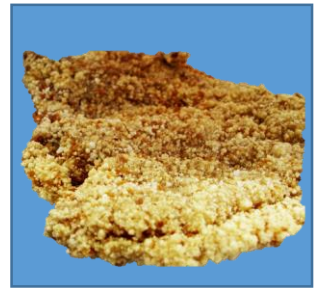
Weighted sum



N = 1



N = 2



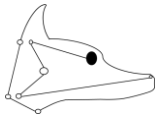
N = m

Bcondux

The assignment

Use MNIST as example.

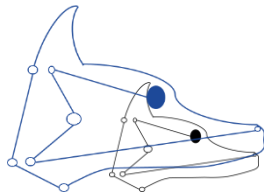
Make a MLP network and optimize it using gradient decent



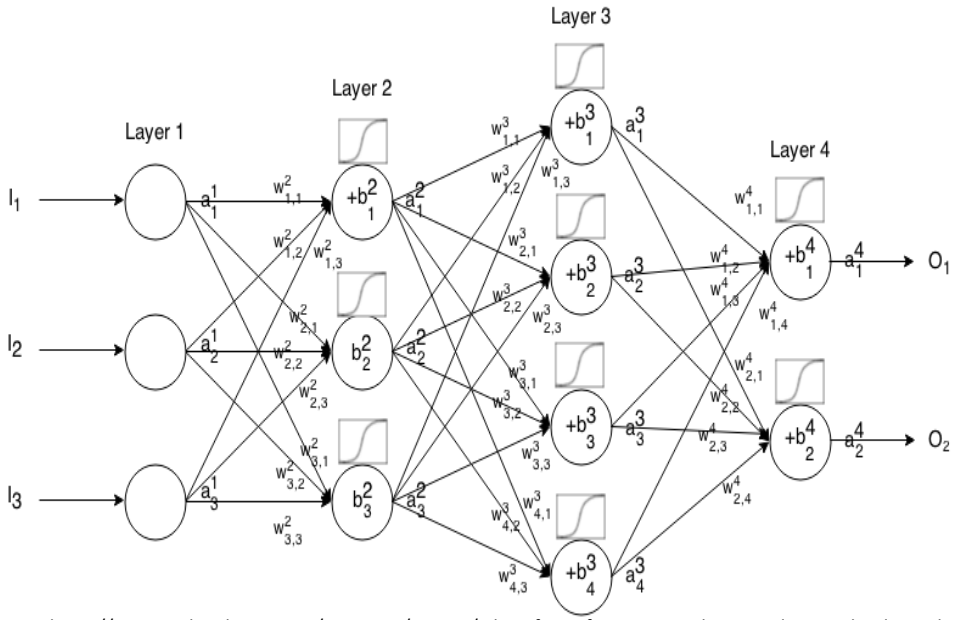
Bcondux

Convolutional Neural Network

Yi-Fan Liou



Neural network

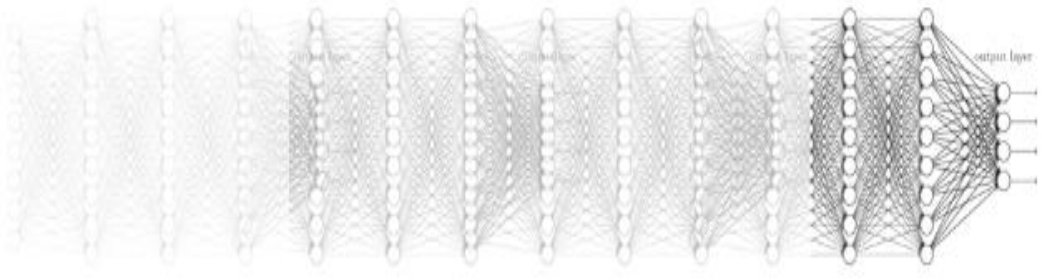


https://stats.stackexchange.com/questions/154879/a-list-of-cost-functions-used-in-neural-networks-alongside-applications

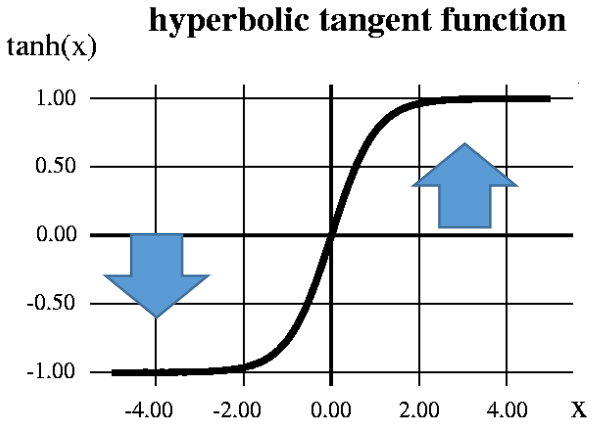
- What's the problem to NN?
  - Computing resource absence
    - Is the computing resource the only problem ?
    - **MPI was born in 1991 while the "AI winter" is 1986~2006**
  - Gradient Vanishing

Gradient vanishing

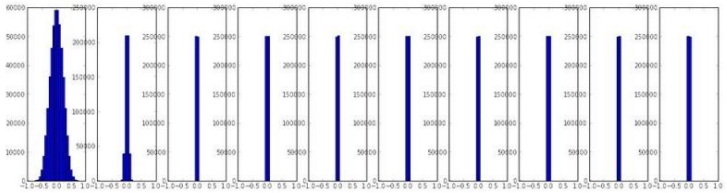
Vanishing gradient (NN winter2: 1986-2006)



http://cswithjames.com/keras-6-vanishing-gradient-problem-relu/



https://towardsdatascience.com/activation-functions-and-its-types-which-is-better-a9a5310cc8f



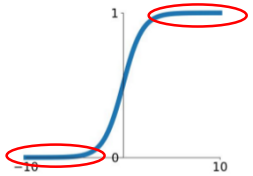
http://blog.csdn.net/zjucor/article/details/7815

Activation functions

Activation Functions

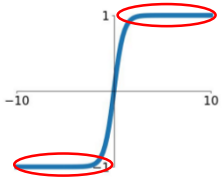
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



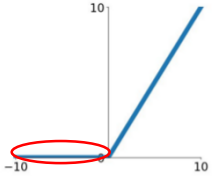
tanh

$$\tanh(x)$$



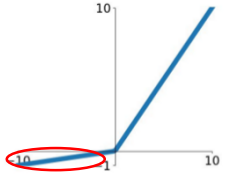
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

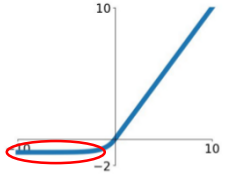


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

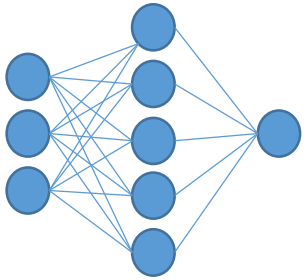
$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



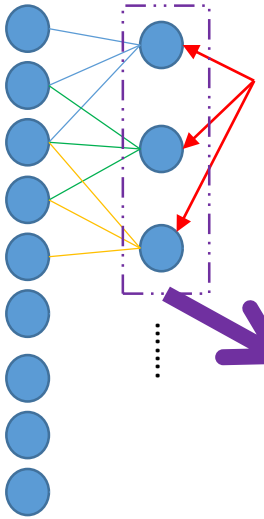
https://medium.com/machine-learning-world/how-to-debug-neural-networks-manual-dc2a200f10f2

- The activation function help the NN
  - Now, the problem is mainly the computing resources

Considering the transform of the NN



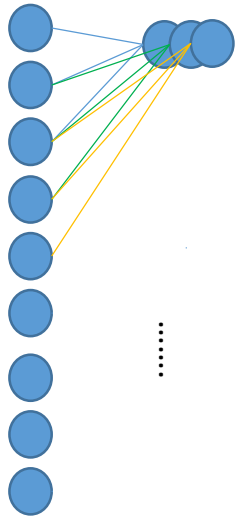
Basic architecture



Weighted sum

How about…
If we already know the “triplet pattern” appears anywhere of the input?

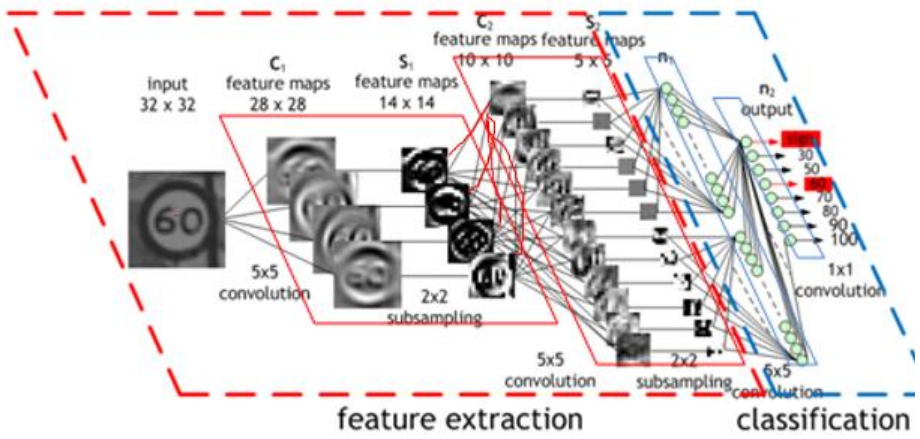
=> Yes, we can use the same weights for weighted sum



- Assume we already know the input data have “triplet pattern”
  - receptive field

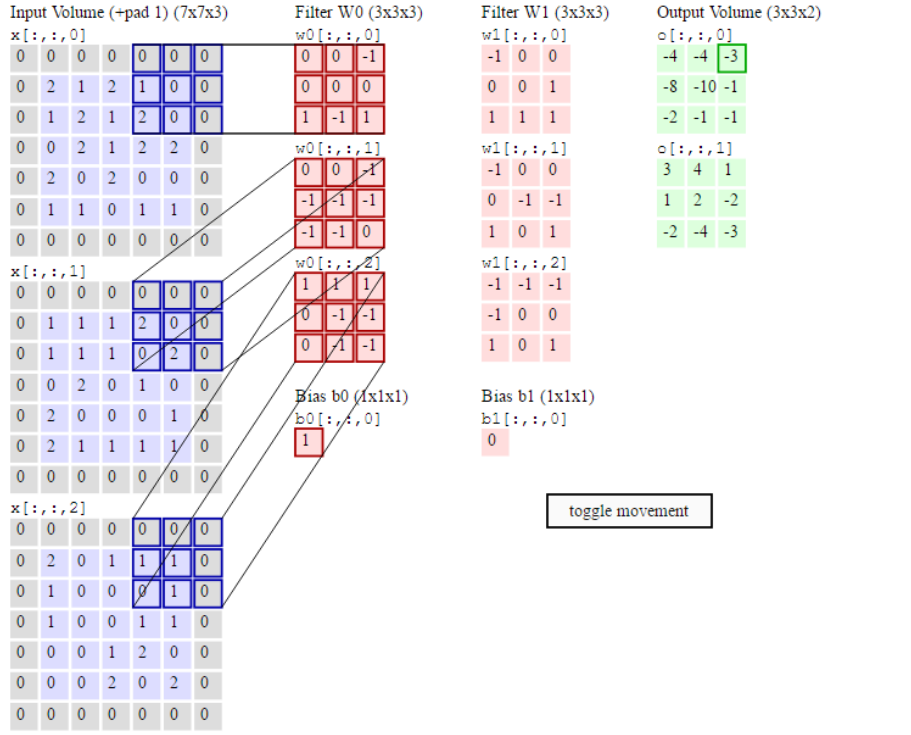
- We use the same weight for weighted-sum operation
  - Weight sharing
  - Save more variables
  - Regulating the weights

How about convolutional neural network



http://www.cnblogs.com/walccott/p/4957076.html

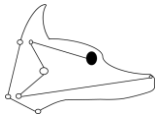
- CNN is also a kind of NNs, but the hidden nodes have their own receptive fields and sharing their weights
- Each “feature map” is a node.
- The ability for classification is still from the fully connected NN.
  - This also means, appending deeper layer here would cause gradient vanishment



https://www.quora.com/In-a-convolutional-neural-network-how-is-convolution-defined-with-an-RGB-image

Assignment

- Use MNIST as example
- Make a CNN for predict hand writing pic.



Bcondux

Thanks