

IOT DATA ANALYSIS (INTRODUCTION TO MACHINE LEARNING)

Assoc Prof Dr Chaodit Aswakul
Wireless Network and Future Internet Research Unit
Chulalongkorn University, Thailand



What is Machine Learning?

[REF: Jake VanderPlas, Python Data Science Handbook
Essential Tools for Working with Data, O'Reilly, 2017]

- Means of building **models** of data
- **Learning** ability by tuning **model parameters**
- **To fit** with previously seen data
- **To predict** and **understand** future unseen data

*“To learn **without** explicitly being **programmed**”*





ML Categories

- **Supervised learning**

Modelling **relationship** between **input features** of data and **output labels** associated with data

- **Unsupervised learning**

Modelling input features of data **without** reference to any **output labels**

- **Reinforcement learning**

Semi-supervised learning with only **incomplete label** available in terms of **state, action, reward**

[REF: Jake VanderPlas, Python Data Science Handbook
Essential Tools for Working with Data, O'Reilly, 2017]



3

IoTcloudServe@TEIN Collaboration Workshop 2



Data: (x, y)
 x is data, y is label

Goal: Learn function to map

$$x \rightarrow y$$

**SUPERVISED
LEARNING**

Apple example:



This thing is an apple.

[REF: ©MIT 6.S191
Introduction to Deep Learning
introtodeeplearning.com]



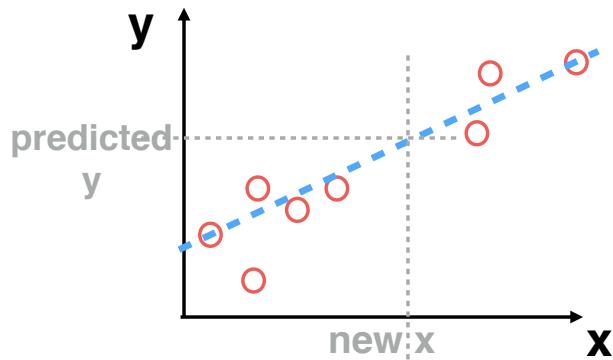
4

IoTcloudServe@TEIN Collaboration Workshop 2



**IF output target is real-valued
THEN “Regression Analysis”**

Example



Wanted to predict

y = house price

from input feature vector

$$\mathbf{x} = (x_1, x_2, x_3)$$

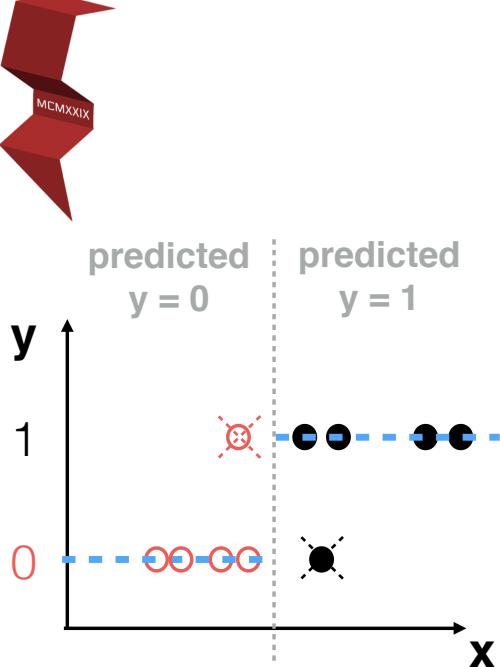
- x_1 = #bed rooms (integers)**
- x_2 = house area (reals)**
- x_3 = location (strings)**

5

IoTcloudServe@TEIN Collaboration Workshop 2



SUPERVISED LEARNING



IF output target belongs to a few discrete set, or *classes*

THEN “Classification Analysis”

Example

Wanted to predict

$y = 1$, system is UP

$y = 0$, system is DOWN

from input features

x_1, x_2, \dots = system measurements



SUPERVISED LEARNING

6

IoTcloudServe@TEIN Collaboration Workshop 2



Data: x

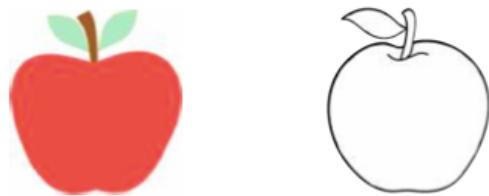
x is data, no labels!

Goal: Learn underlying structure

UNSUPERVISED LEARNING

[REF: ©MIT 6.S191
Introduction to Deep Learning
introtodeeplearning.com]

Apple example:



This thing is like
the other thing.

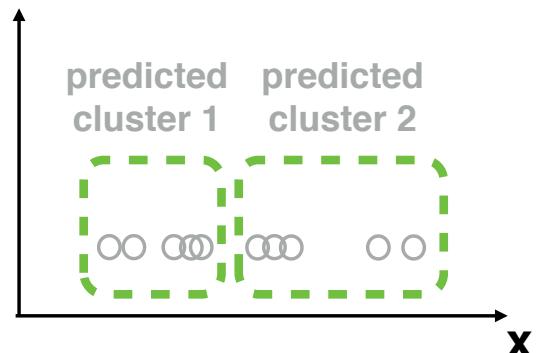
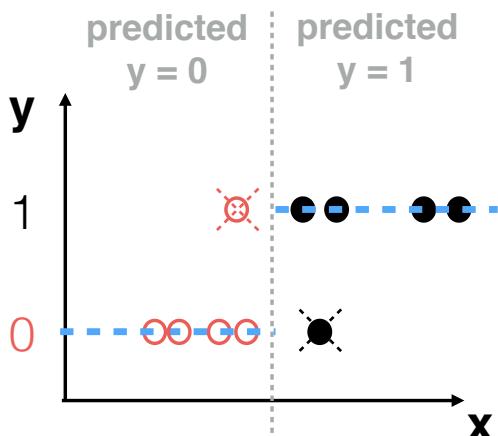


7

IoTcloudServe@TEIN Collaboration Workshop 2

SUPERVISED LEARNING

UNSUPERVISED LEARNING

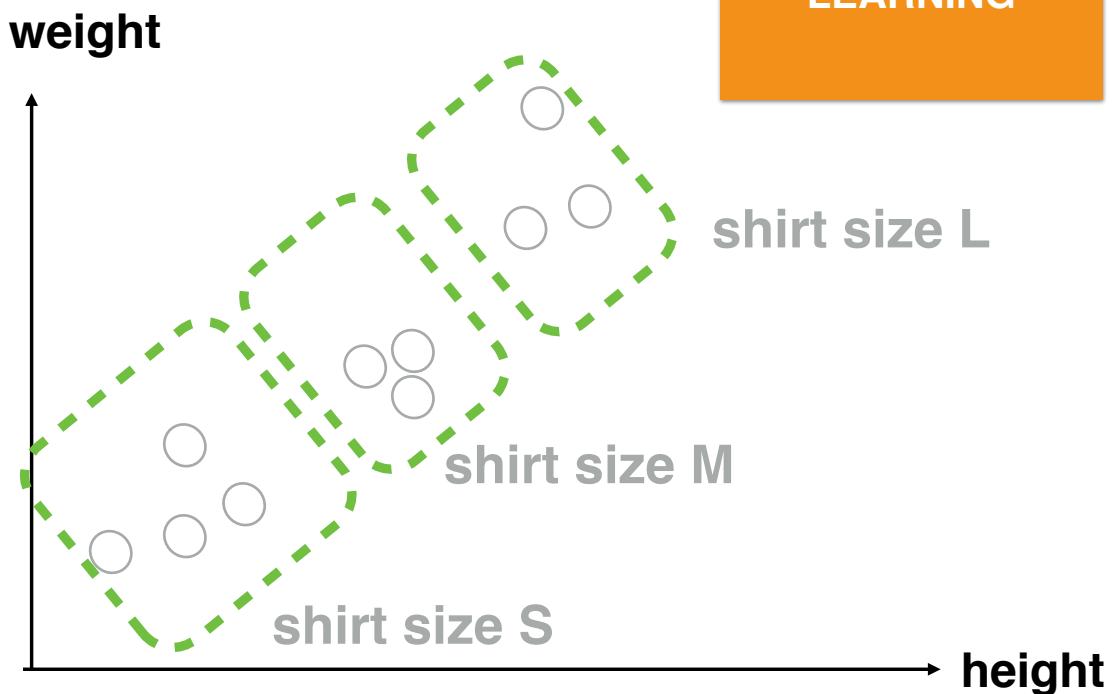


8

IoTcloudServe@TEIN Collaboration Workshop 2



UNSUPERVISED LEARNING



9

IoTcloudServe@TEIN Collaboration Workshop 2



Data: state-action pairs

Goal: Maximize future rewards over many time steps

REINFORCEMENT LEARNING

Apple example:



Eat this thing because it will keep you alive.



[REF: ©MIT 6.S191
Introduction to Deep Learning
introtodeeplearning.com]

10

IoTcloudServe@TEIN Collaboration Workshop 2



Reinforcement Learning



Reward: feedback that measures the success or failure of the agent's action.

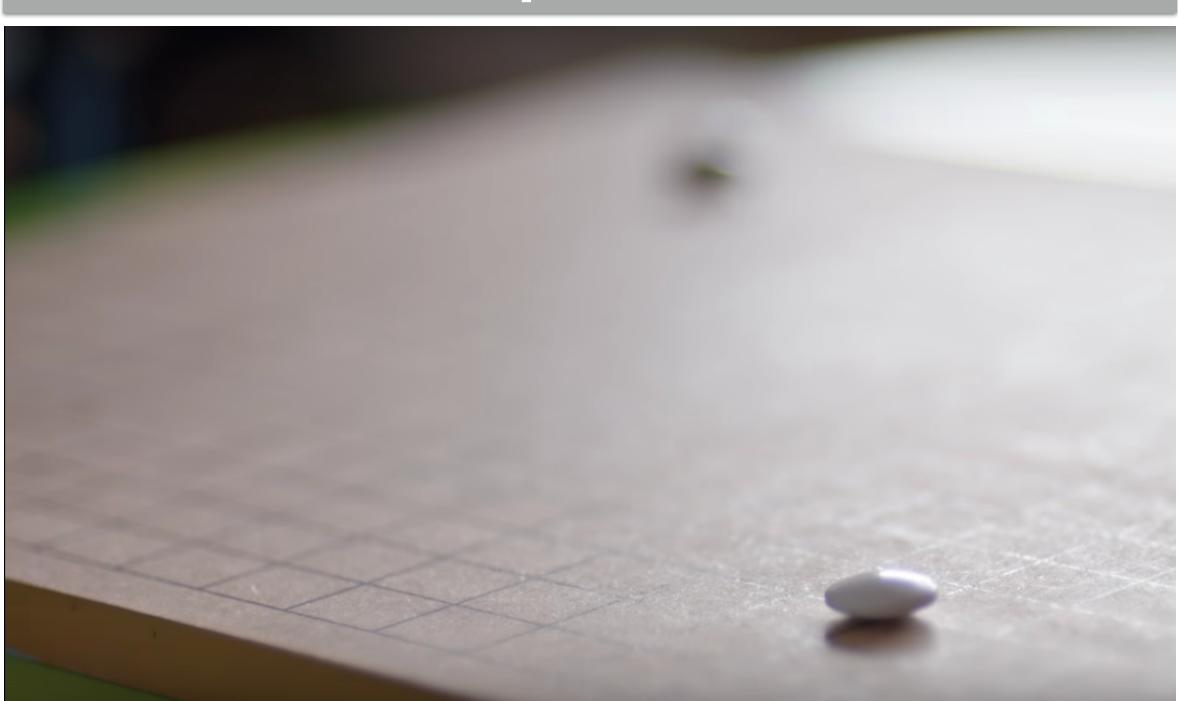
[REF: ©MIT 6.S191
Introduction to Deep Learning
<http://deeplearning.com>]
 cloudserve @TEIN

11

IoTcloudServe@TEIN Collaboration Workshop 2



AlphaGo



https://youtu.be/8tq1C8spV_g Movie trailer 1.30 mins

<https://youtu.be/jGyCsVhtW0M> Documentary 1.30 hrs



12

IoTcloudServe@TEIN Collaboration Workshop 2



AlphaGo

"The IBM chess computer Deep Blue, which famously beat grandmaster Garry Kasparov in 1997, was explicitly programmed to win at the game.

But AlphaGo was not preprogrammed to play Go.

Rather, it learned using a general-purpose algorithm that allowed it to interpret the game's patterns

in a similar way to how a DeepMind program learned to play 49 different arcade games"

REF: Nature 529, 445–446 (28 January 2016)

<https://www.nature.com/news/google-ai-algorithm-masters-ancient-game-of-go-1.19234>



13

IoTcloudServe@TEIN Collaboration Workshop 2



Hands-On Experiment

<https://www.tensorflow.org/js/demos/>

The screenshot shows the TensorFlow.js website with the URL <https://www.tensorflow.org/js/demos/>. The page features a navigation bar with links like Overview, Tutorials, Guide, Models, Demos (which is selected), and API. Below the navigation, there are six demo cards:

- Emoji Scavenger Hunt**: Use your phone's camera to identify emojis in the real world. [Explore demo](#) | [View code](#)
- Webcam Controller**: Play Pac-Man using images trained in your browser. [Explore demo](#) | [View code](#)
- Teachable Machine**: Teach a machine to recognize images and play sounds. [Explore demo](#) | [View code](#)
- Move Mirror**: Explore pictures in a fun new way, just by moving around. [Explore demo](#) | [View code](#)
- Performance RNN**: Enjoy a real-time piano performance by a neural network. [Explore demo](#) | [View code](#)
- Node.js Pitch Prediction**: Train a server-side model to classify baseball pitch types using Node.js. [View code](#)



14

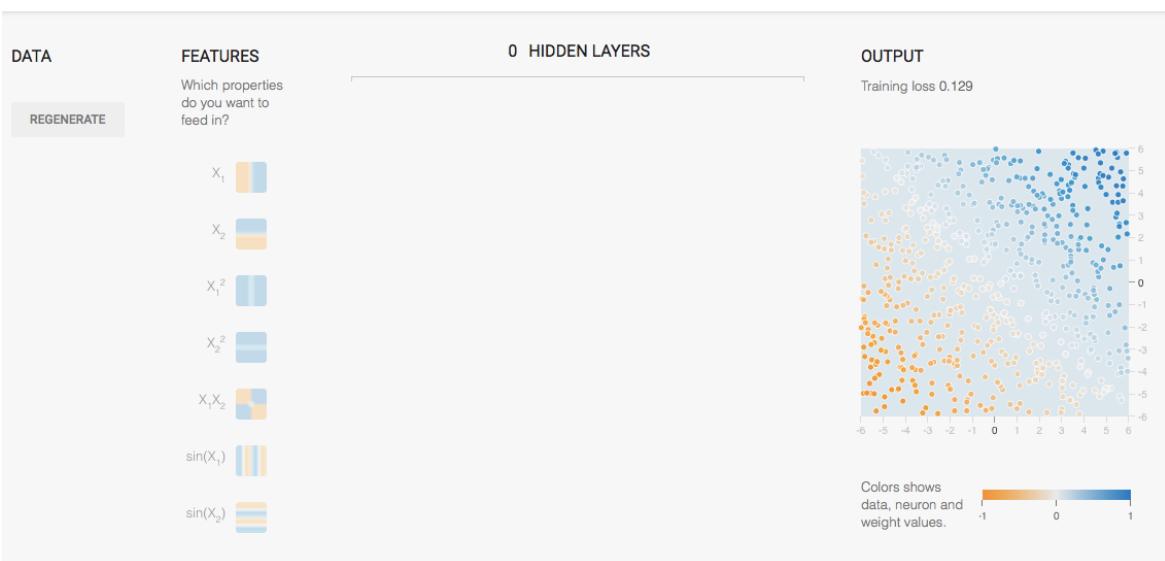
IoTcloudServe@TEIN Collaboration Workshop 2



Linear Regression Experiment



Epoch
000,000



Enter this experiment at bit.ly/2TQYYvf



15

IoTcloudServe@TEIN Collaboration Workshop 2

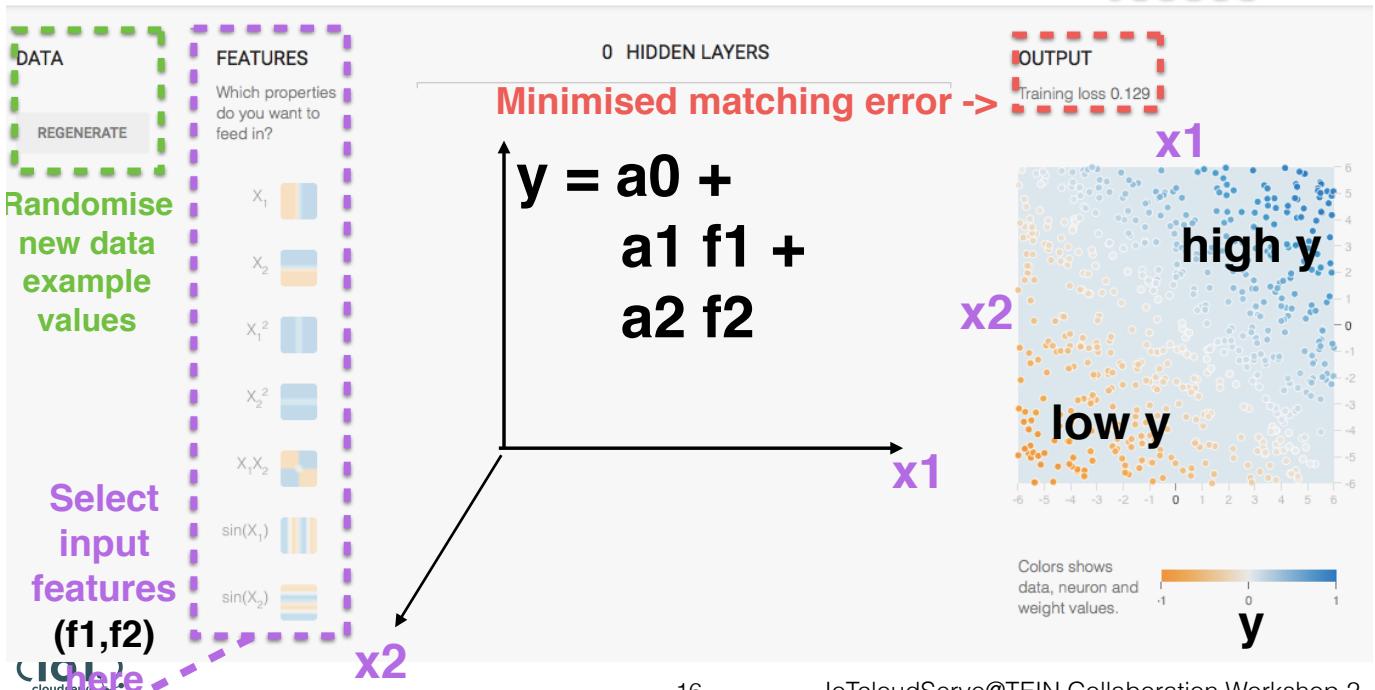


Regression Experiment

Goal: match background colours (predicted values) with dot colours (data example values)

Time step used to minimise matching error ->

Epoch
000,000



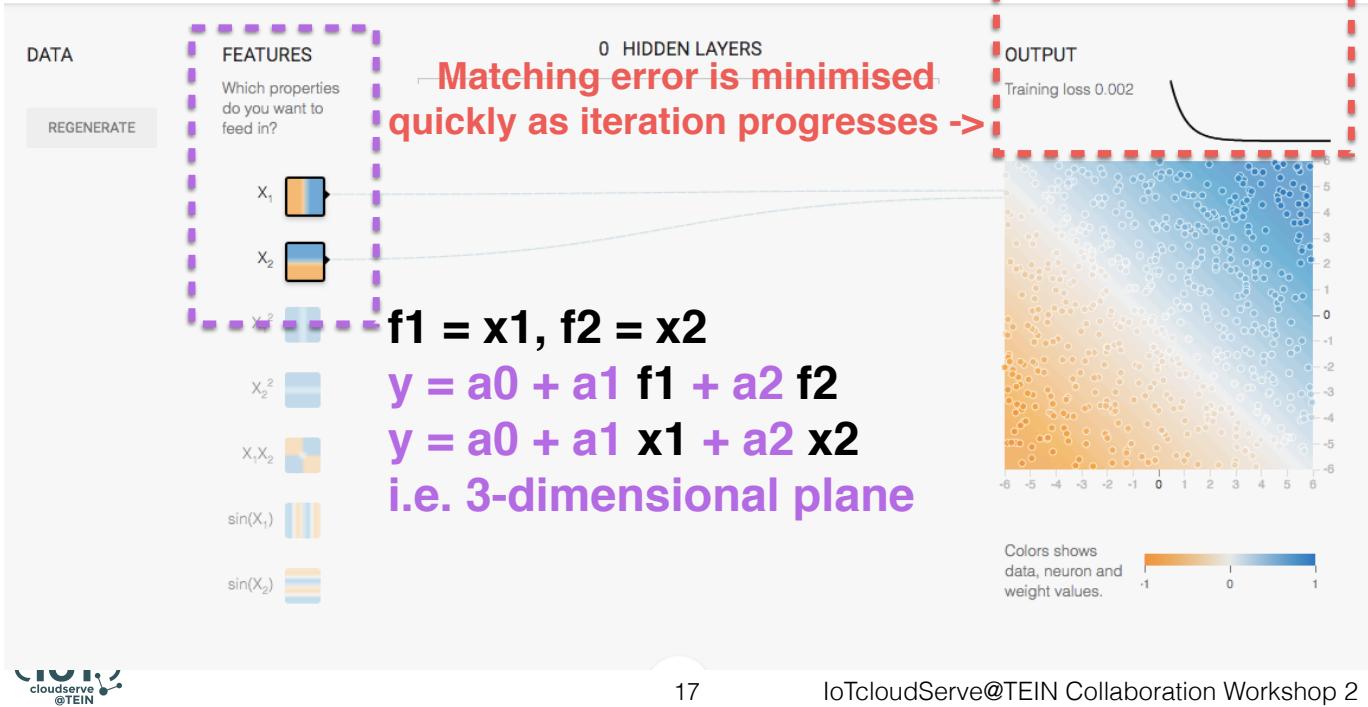
16

IoTcloudServe@TEIN Collaboration Workshop 2



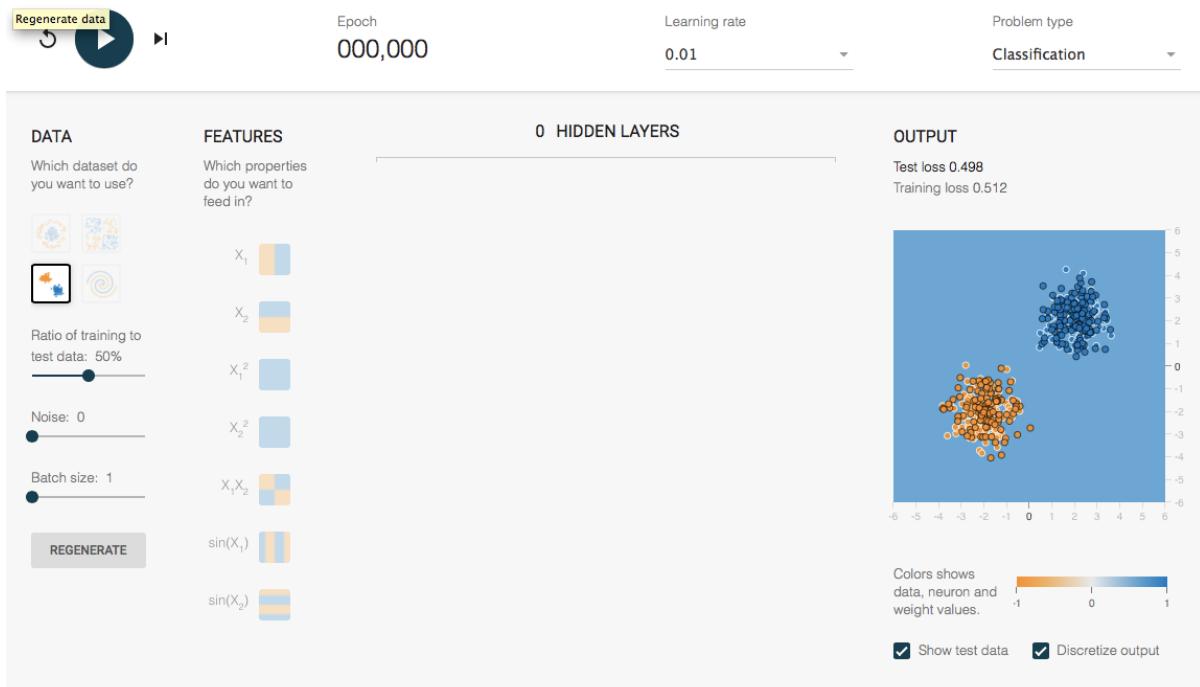
Regression Experiment

Goal: match background colours (predicted values) with point colours (data example values)



Classification Experiment

Goal: match discretised background colours (predicted values) with dot colours (data example values: -1 “orange” or 1 “blue”)

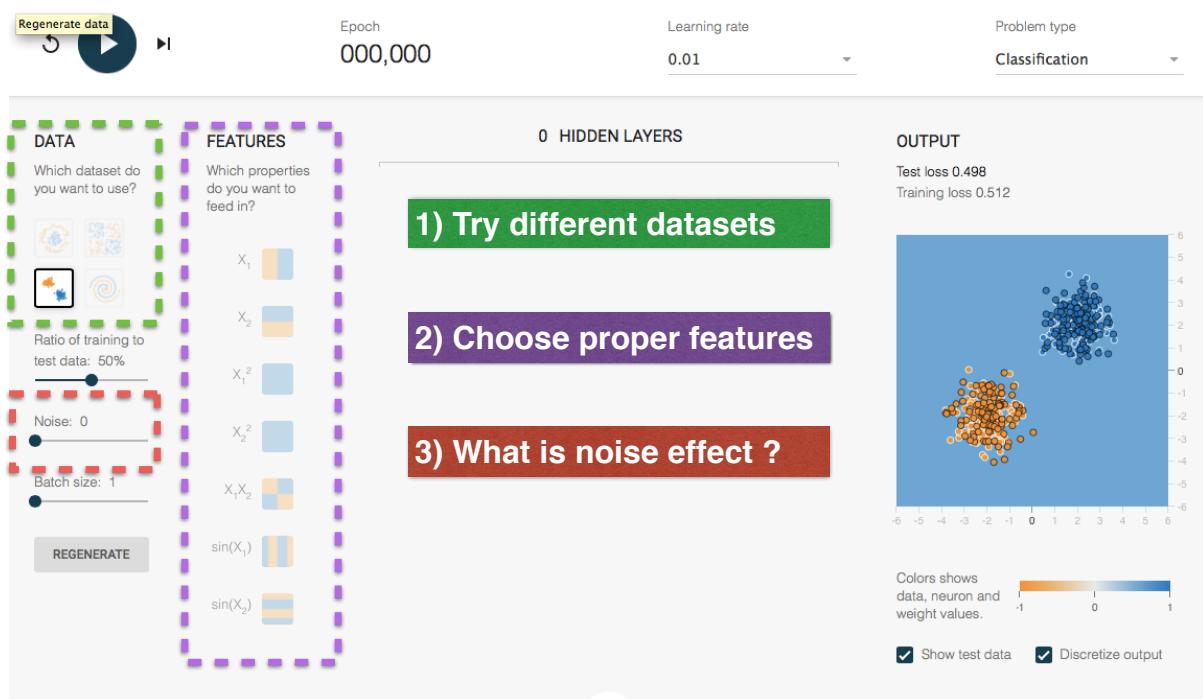


Enter this experiment at bit.ly/2OIT2Ep



Classification Experiment

Goal: match discretised background colours (predicted values) with dot colours (data example values: -1 “orange” or 1 “blue”)



Enter this experiment at bit.ly/2OIT2Ep



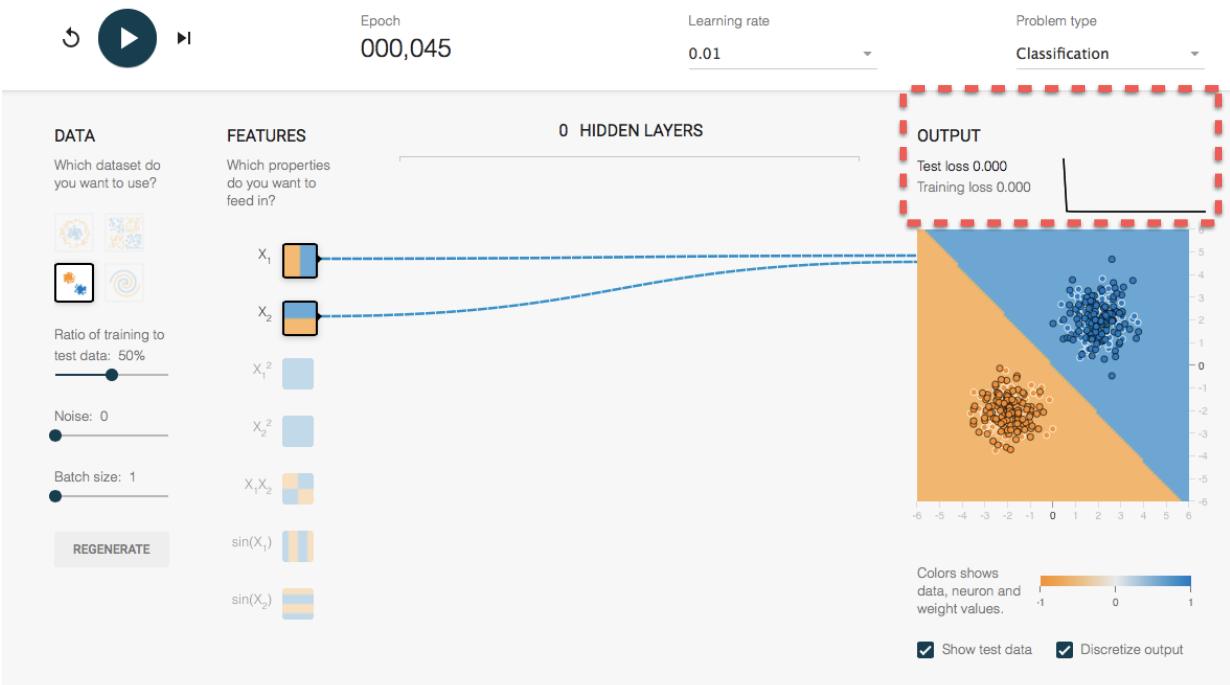
19

IoTcloudServe@TEIN Collaboration Workshop 2



Classification Experiment

Data: Two Gaussian Distributions (No Noise)



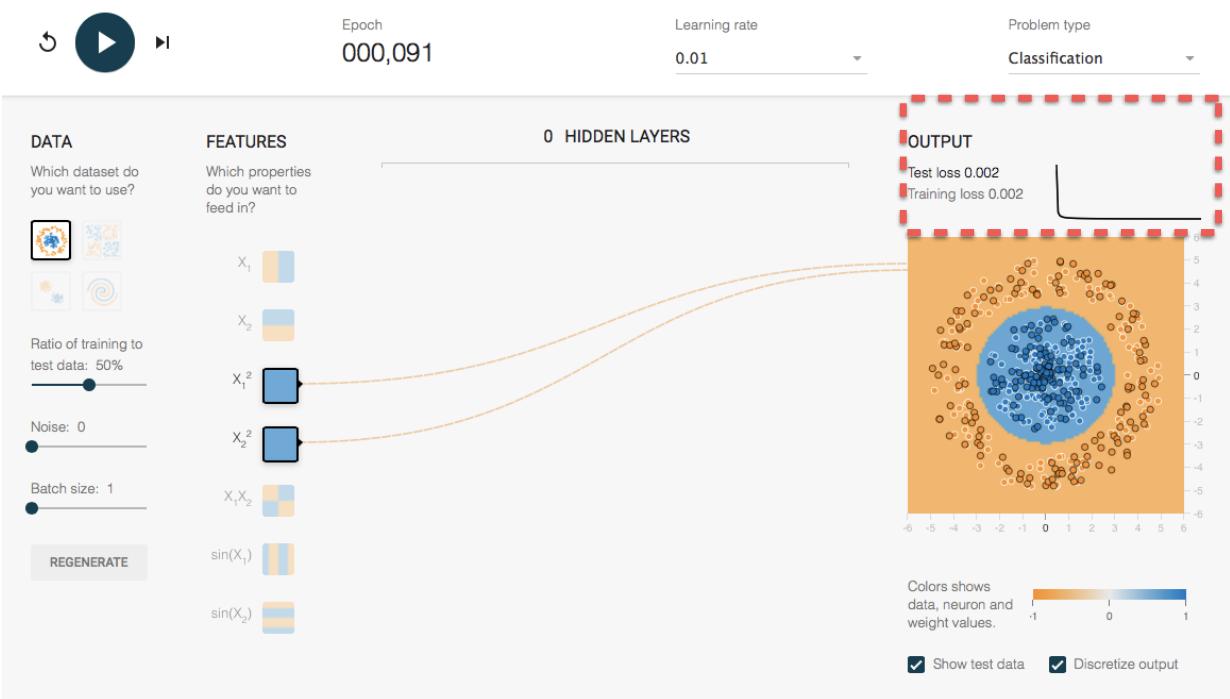
20

IoTcloudServe@TEIN Collaboration Workshop 2



Classification Experiment

Circle Data (No Noise)



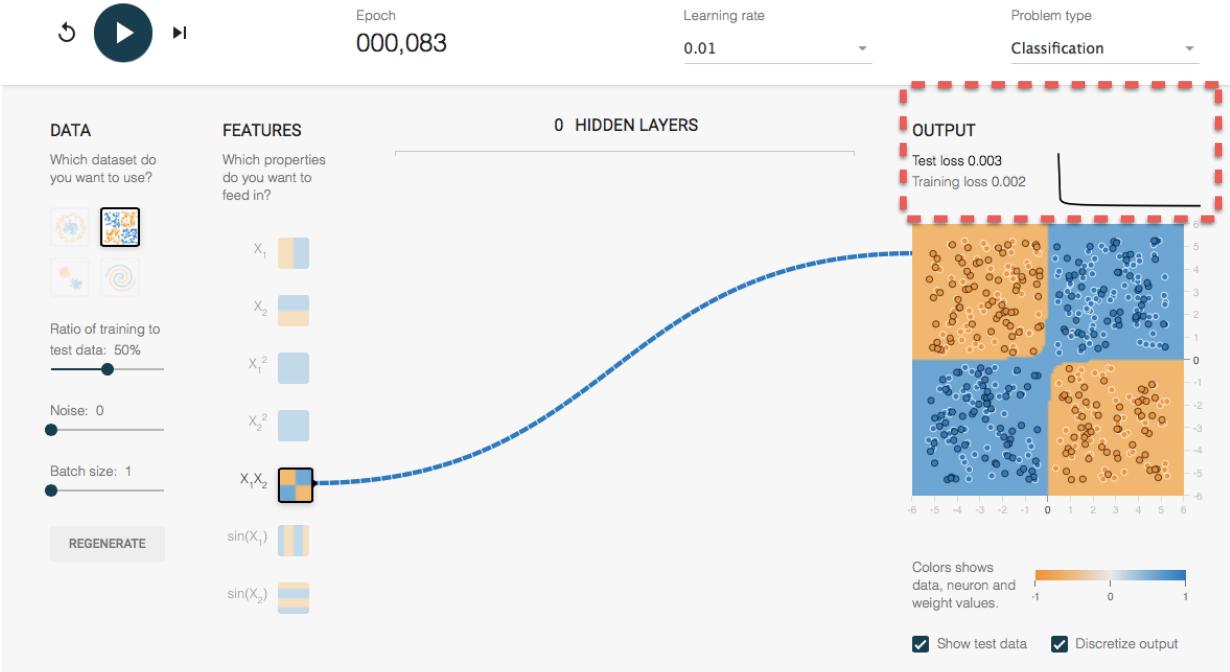
21

IoTcloudServe@TEIN Collaboration Workshop 2



Classification Experiment

XOR Data (No Noise)



22

IoTcloudServe@TEIN Collaboration Workshop 2



Classification Experiment

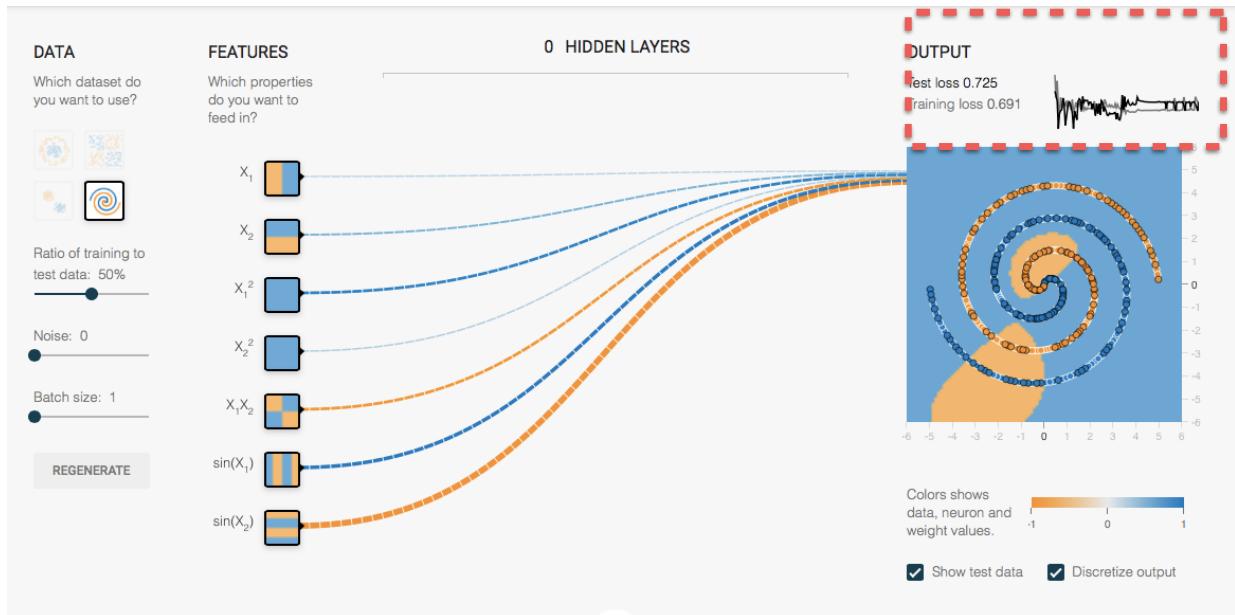
Spiral Data (No Noise)



Epoch
000,124

Learning rate
0.01

Problem type
Classification



23

IoTcloudServe@TEIN Collaboration Workshop 2



Artificial (Deep) Neural Network Experiment



Epoch
000,507

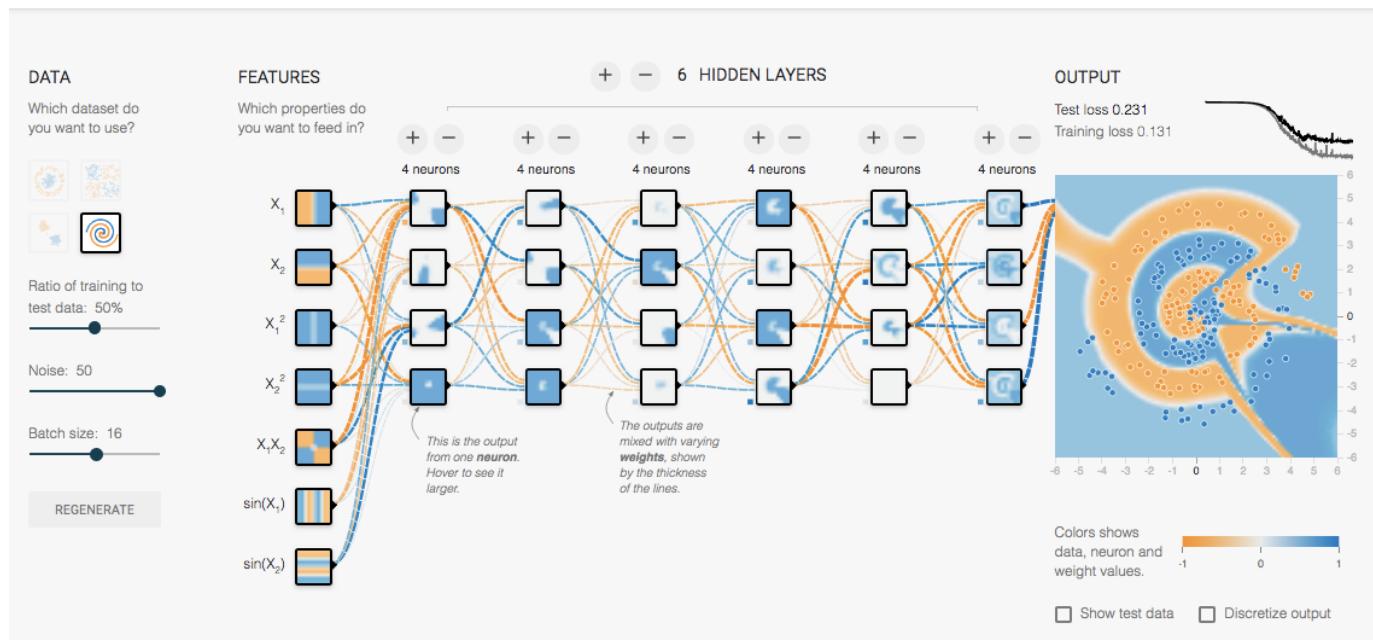
Learning rate
0.03

Activation
ReLU

Regularization
None

Regularization rate
0

Problem type
Classification



24

IoTcloudServe@TEIN Collaboration Workshop 2



NEXT: Scikit-Learn Package Python Code Examples

Import virtual machine into Virtual Box. Turn it on.

Open Jupyter notebook, in terminal, just type

>> jupyter notebook

Browse to two provided folders on Desktop:

- 05.03-Hyperparameters-and-Model-Validation_cak20190327
- 05.11-K-Means_cak20190330

Click to open .ipynb file to load python codes

REF: Jake VanderPlas, Python Data Science Handbook Essential Tools for Working with Data, O'Reilly, 2017
<https://jakevdp.github.io/PythonDataScienceHandbook>



25

IoTcloudServe@TEIN Collaboration Workshop 2

The screenshot shows the Jupyter Notebook interface. At the top, there's a header with the IoT cloudserve @TEIN logo, the title "jupyter notebook", and "Logout" and "Quit" buttons. Below the header, there are tabs for "Files", "Running", and "Clusters". The main area displays a file list titled "Desktop". The list includes a folder named "0" (containing "0" files), a folder named "50.03-Hyperparameters-and-Model-Validation_cak20190327" (last modified 2 hours ago), and a folder named "50.11-K-Means_cak20190330" (last modified 2 hours ago). At the bottom of the interface, there are "Upload", "New", and "Reset" buttons.

Name	Last Modified	File size
0	seconds ago	
50.03-Hyperparameters-and-Model-Validation_cak20190327	2 hours ago	
50.11-K-Means_cak20190330	2 hours ago	



26

IoTcloudServe@TEIN Collaboration Workshop 2



jupyter notebook

File Edit View Insert Cell Kernel Help Trusted Python 3 O

This notebook contains an excerpt from the [Python Data Science Handbook](#) by Jake VanderPlas; the content is available [on GitHub](#).
The text is released under the [CC-BY-NC-ND license](#), and code is released under the [MIT license](#). If you find this content useful, please consider supporting the work by [buying the book!](#)

< [Introducing Scikit-Learn](#) | [Contents](#) | [Feature Engineering](#) >

Open in Colab

Hyperparameters and Model Validation

In the previous section, we saw the basic recipe for applying a supervised machine learning model:

1. Choose a class of model
2. Choose model hyperparameters
3. Fit the model to the training data
4. Use the model to predict labels for new data



27

IoTcloudServe@TEIN Collaboration Workshop 2



jupyter notebook

File Edit View Insert Cell Kernel Help Trusted Python 3 O

This notebook contains an excerpt from the [Python Data Science Handbook](#) by Jake VanderPlas; the content is available [on GitHub](#).
The text is released under the [CC-BY-NC-ND license](#), and code is released under the [MIT license](#). If you find this content useful, please consider supporting the work by [buying the book!](#)

< [In-Depth: Manifold Learning](#) | [Contents](#) | [In Depth: Gaussian Mixture Models](#) >

Open in Colab

In Depth: k-Means Clustering

In the previous few sections, we have explored one category of unsupervised machine learning models: dimensionality reduction. Here we will move on to another class of unsupervised machine learning models: clustering algorithms. Clustering algorithms seek to learn, from the properties of the data, an optimal division or discrete labeling of groups of points.



28

IoTcloudServe@TEIN Collaboration Workshop 2

IOT DATA ANALYSIS

Reinforcement Learning

2102541 IoT Fundamentals (2/2018)

Assoc Prof Dr Chaodit Aswakul

chaodit.a@chula.ac.th



ML Categories

- Supervised learning
 - Regression models (linear regression, neural network)  fitting model with past data to predict unseen cases
 - Classification models (logistic regression, neural network)
- Unsupervised learning
 - K-means clustering  finding structural patterns in data
- Reinforcement learning
 - Q-learning  taking control of system





Data: state-action pairs

REINFORCEMENT LEARNING

[REF: ©MIT 6.S191
Introduction to Deep Learning
introtodeeplearning.com]

Goal: Maximize future rewards over many time steps

Apple example:



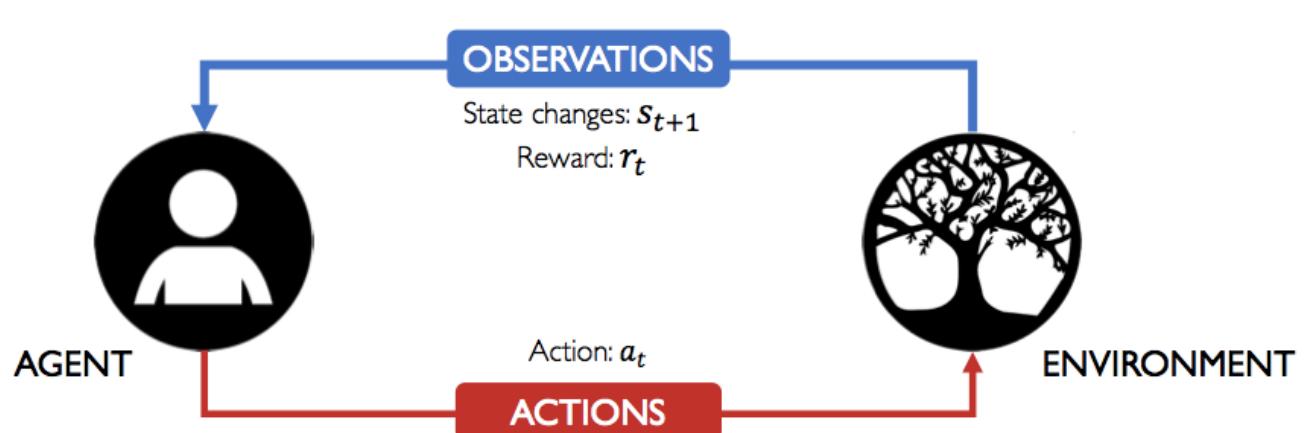
Eat this thing because it will keep you alive.



31

IoTcloudServe@TEIN Collaboration Workshop 2

Reinforcement Learning



Reward: feedback that measures the success or failure of the agent's action.

[REF: ©MIT 6.S191
Introduction to Deep Learning
introtodeeplearning.com]



32

IoTcloudServe@TEIN Collaboration Workshop 2



Classic Reference Textbook

Reinforcement Learning

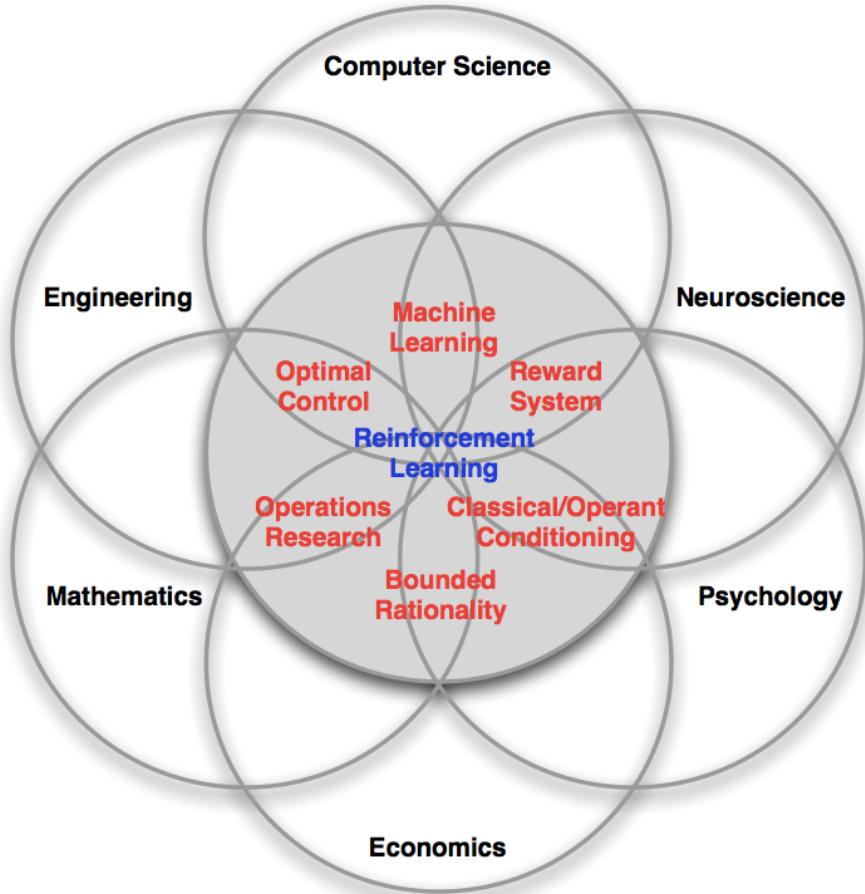
An Introduction
second edition

Richard S. Sutton and Andrew G. Barto



33

IoTcloudServe@TEIN Collaboration Workshop 2



REF: David Silver (2015)

34

IoTcloudServe@TEIN Collaboration Workshop 2



Q-Learning

When system is in state s ...

... agent must choose an action a

Define:

$Q(s,a) =$

**'quality' or 'reward' given to agent after
agent takes action a when system is in state s**



EXAMPLE 1

system of road network

state : congestion level

action : traffic signal light

reward : travel time reduction





EXAMPLE 2

system of network firewall

state : info of captured packets

action : decision to let-go or drop

reward : malicious packet dropped

good packets let-go



EXAMPLE 3

system of Alpha GO

state : play positions on board

action : where to play next

reward : winning chance





TODAY PROBLEM

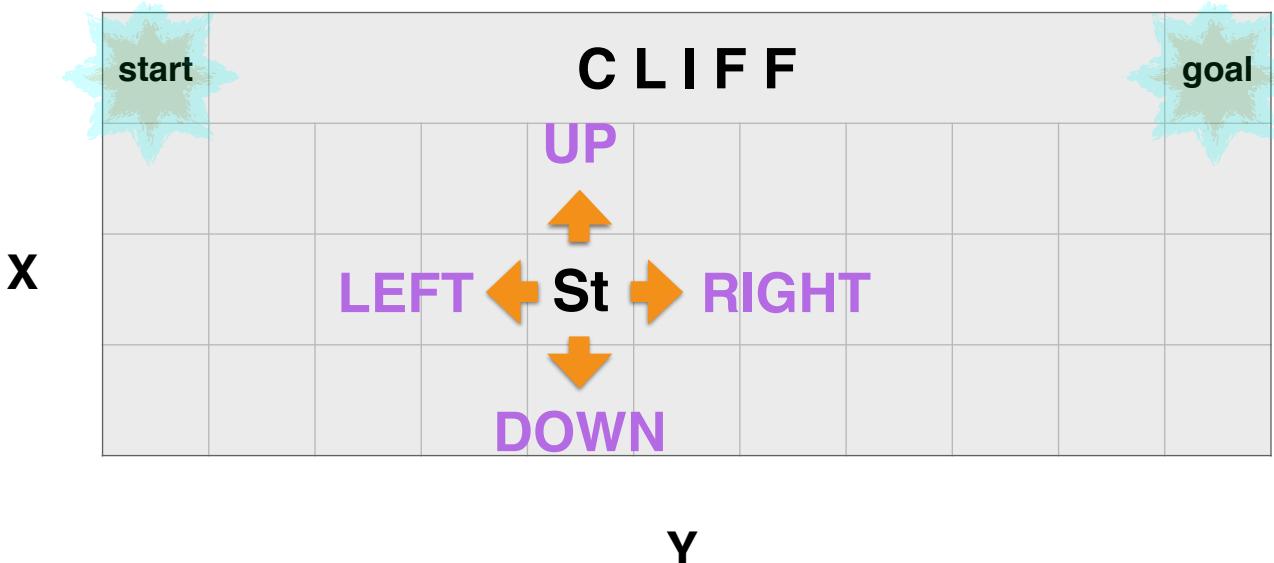


39

IoTcloudServe@TEIN Collaboration Workshop 2



Cliff Walking Scenario



KFR Sutton & Barto, Reinforcement Learning Book

40

IoTcloudServe@TEIN Collaboration Workshop 2



Today's Class : Cliff-Walking

Choose path from START to GOAL

state : current location

action : next movement directions

(**LEFT, RIGHT, UP, DOWN**)

reward : -1 (if still on the way but reaching goal)

-100 (if falling off cliff)

reward recorded after reaching goal
i.e. “completed one learning episode”



REF Sutton & Barto, Reinforcement Learning Book

41

IoTcloudServe@TEIN Collaboration Workshop 2



Q-Learning

- Initialise $Q(s, a)$ for all (s, a) arbitrarily
- Choose actions in any way (based on $Q(s, a)$) such that all actions are taken in all states (infinitely often in the limit)

$\dots S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1}, R_{t+2}, S_{t+2}, \dots$



REF Sutton & Barto, Reinforcement Learning Book

42

IoTcloudServe@TEIN Collaboration Workshop 2



Q-Learning

“epsilon-greedy algorithm”

- with probability = 1 - epsilon choose action that maximises $Q(s, a)$
“exploitation strategy”
- with probability = epsilon choose any other actions randomly
“exploration strategy”



REF: Sutton & Barto, Reinforcement Learning Book

43

IoTcloudServe@TEIN Collaboration Workshop 2



Q-Learning

- On each time step t , update $Q(s, a)$ by

$$Q(S_t, A_t) \leftarrow (1 - \alpha) \underset{\text{old value}}{Q(S_t, A_t)} + \alpha (\underset{\text{immediately gained reward}}{R_{t+1}} + \gamma \underset{\text{future expected maximum reward}}{\max_a Q(S_{t+1}, a)})$$



REF: Sutton & Barto, Reinforcement Learning Book

44

IoTcloudServe@TEIN Collaboration Workshop 2



Hands-on Q-Learning Experiment

File Edit View Insert Cell Kernel Help Trusted | Python 3

Reinforcement Learning (Q-Learning)

Remark: The original file has been simplified and reformatted to Jupyter Notebook for teaching in 2102541 IoT Fundamentals. (C. Aswakul 2 Apr 2019).

```
In [ ]: # REF: github.com/ShangtongZhang/reinforcement-learning-an-introduction  
#           /blob/master/chapter06/cliff_walking.py  
  
#####  
# Copyright (C) #  
# 2016-2018 Shangtong Zhang(zhangshangtong.cpp@gmail.com) #  
# 2016 Kenta Shimada(hyperkentakun@gmail.com) #  
# Permission given to modify the code as long as you keep this #  
# declaration at the top #  
#####
```

The logo consists of a stylized blue cloud shape containing the letters "REF". To the right of the cloud, the word "IoT" is written vertically in large, bold, blue letters. Below "IoT", the words "cloudserve" and "@TEIN" are written in smaller blue text.

 REF: Sutton & Barto, Reinforcement Learning Book

45

IoTcloudServe@TEIN Collaboration Workshop 2



THANK YOU

Assoc Prof Dr Chaodit Aswakul
Wireless Network and Future Internet Research Unit
Chulalongkorn University, Thailand



46

IoTcloudServe@TEIN Collaboration Workshop 2