2018

機器學習與深度學習 Machine Learning & Deep Learning

課程大綱

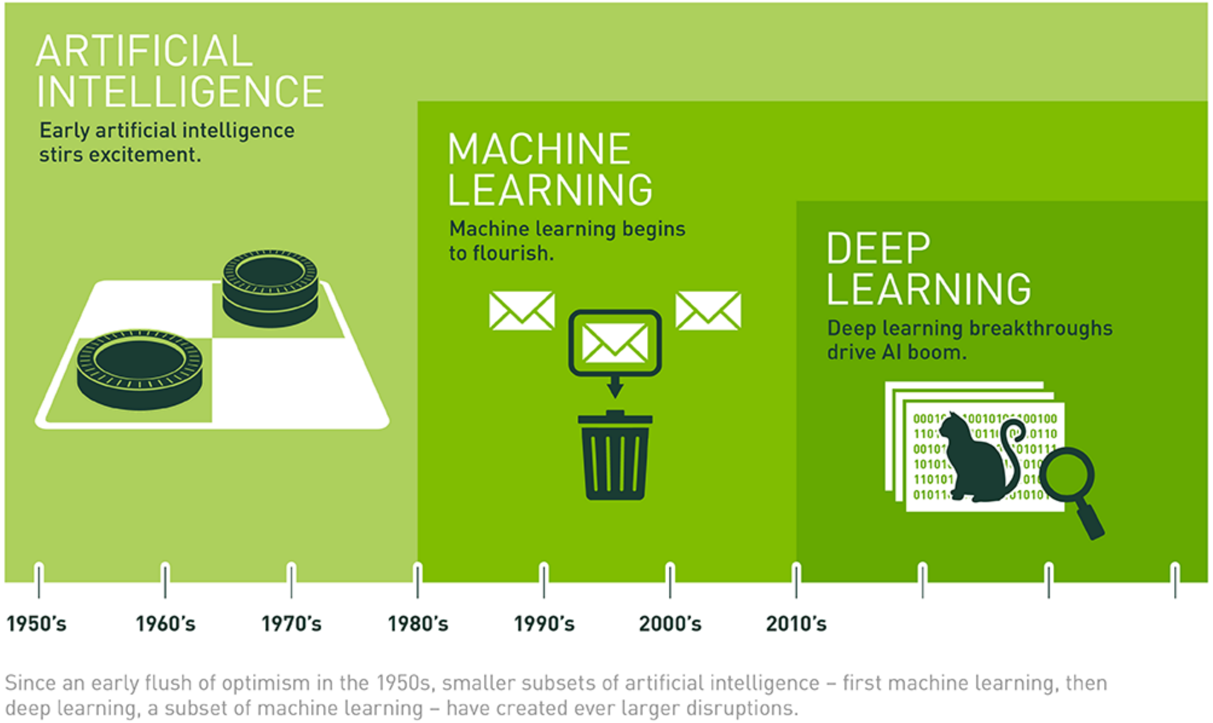
許晉龍

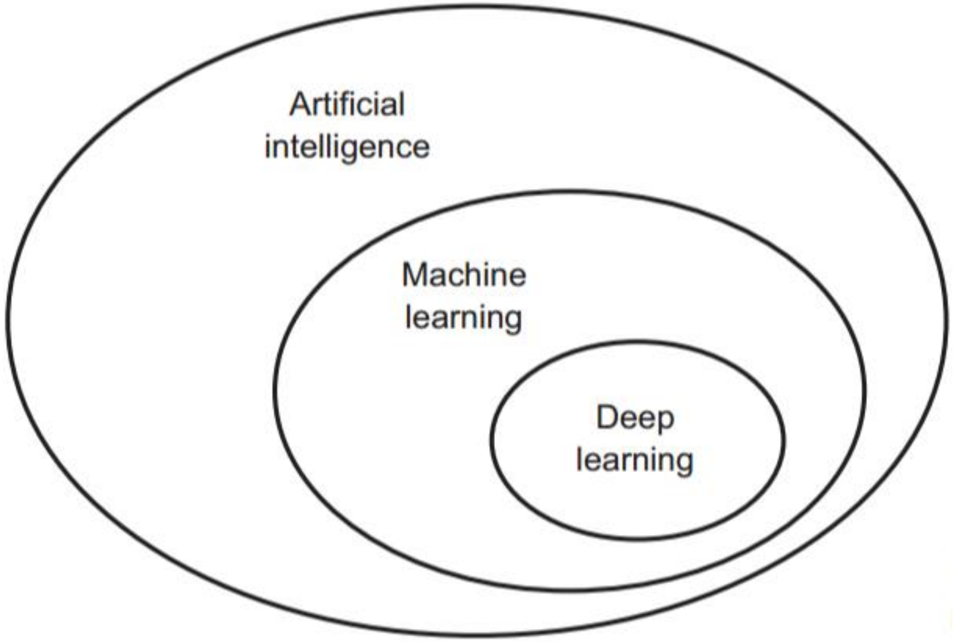
國立臺北商業大學資訊管理系

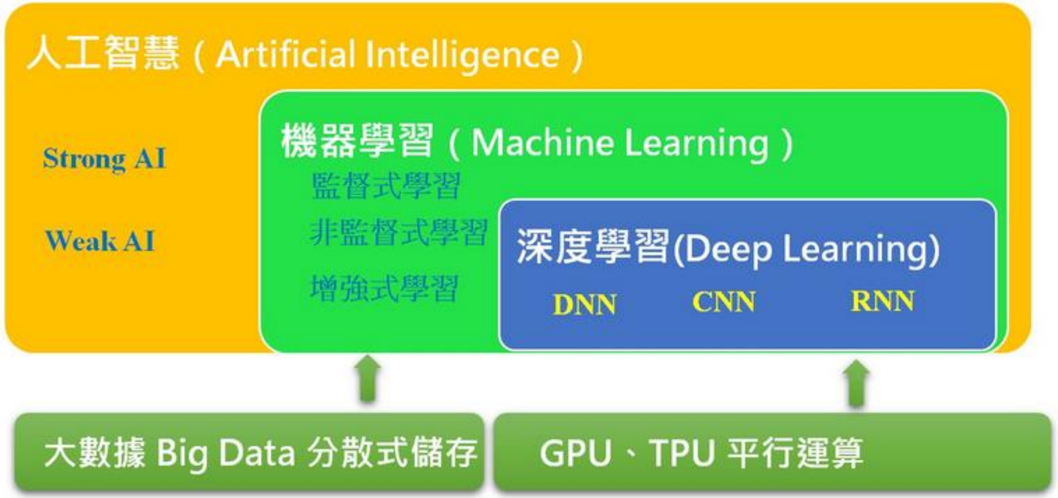
課程目的
Purpose

- 了解機器學習及深度學習基本概念，以ｐｙｔｈｏｎ程式語言實作機器學習及深度學習，實際解決生活上或商業上問題。
- 課程強調python程式語言實作機器學習及深度學習能力養成！

人工智慧
機器學習
深度學習







https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/

林大貴, TensorFlow+Keras深度學習人工智慧實務應用, 博碩, 2017.

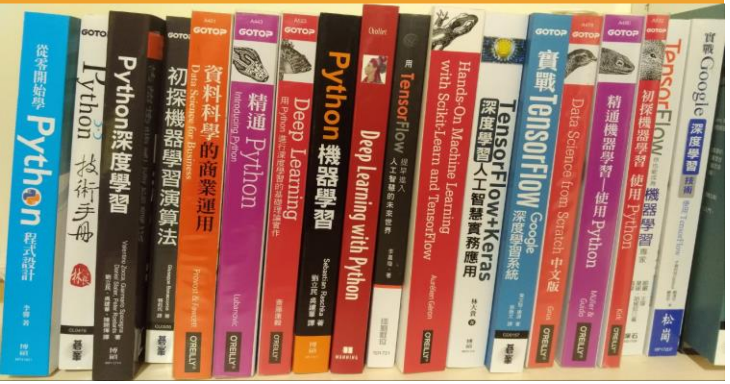
GPU, TPU, VPU

Graphics Processing Unit

Tensor Processing Unit

Vision Processing Unit

課程內容



- 資料科學(Data science)
- 機器學習概論(Basic concept of machine learning)
- Python程式語言實作
- 監督式學習(Supervised learning)
- 非監督式學習(Unsupervised learning)
- 整體學習 (Ensemble learning)
- 深度學習(Deep learning)

A

概論 Concept

- 資料科學(Data science)
- 機器學習概論(Basic concept of machine learning)

B

Python 程式語言實作

B1

- 01Python 資料型態與基本運算.ipynb
- 02Python 字串處理.ipynb
- 03Python 資料結構.ipynb
- 04Python 程式結構.ipynb
- 05Pandas基本處理.ipynb
- 06Numpy 矩陣.ipynb
- 07資料清理與轉換.ipynb
- 08Matplotlib 視覺化繪圖.ipynb
- 09資料聚合.ipynb
- 10樞紐分析.ipynb

B2

綜合練習

PM2.5分析

C

監督式學習 Supervised learning

C1

- 迴歸 (Regression)
- 機器學習流程

C2

- 邏輯迴歸(Logistic Regression)
- 分類模型的評估

C3

- K最近鄰(K Nearest Neighbor, KNN)

C4

- 支持向量機(Support Vector Machine, SVM)

C5

- 決策樹(Decision Tree)

C6

- 單純貝式分析 (Naïve Bayes Classifier)

D

非監督式學習 Unsupervised learning

D1

- 集群分析(Cluster analysis)
  - K-means, K-means++, Hierarchical Clustering, DBSCAN

D2

- 購物籃分析(Market Basket Analysis)
  - Apriori algorithm

D3

- 主成分分析 (Principal Components Analysis, PCA)
  - PCA, Kernel PCA

整體學習 Ensemble Learning

整合多種機器學習演算法

- 投票法 (Voting)
- 裝袋法 (Bagging)
- 隨機森林 (Random Forest)
- 自適應增強(Adaboost)
- StackingClassifier
- XGBoost (Kaggle機器學習競賽神器)

F
深度學習 Deep learning

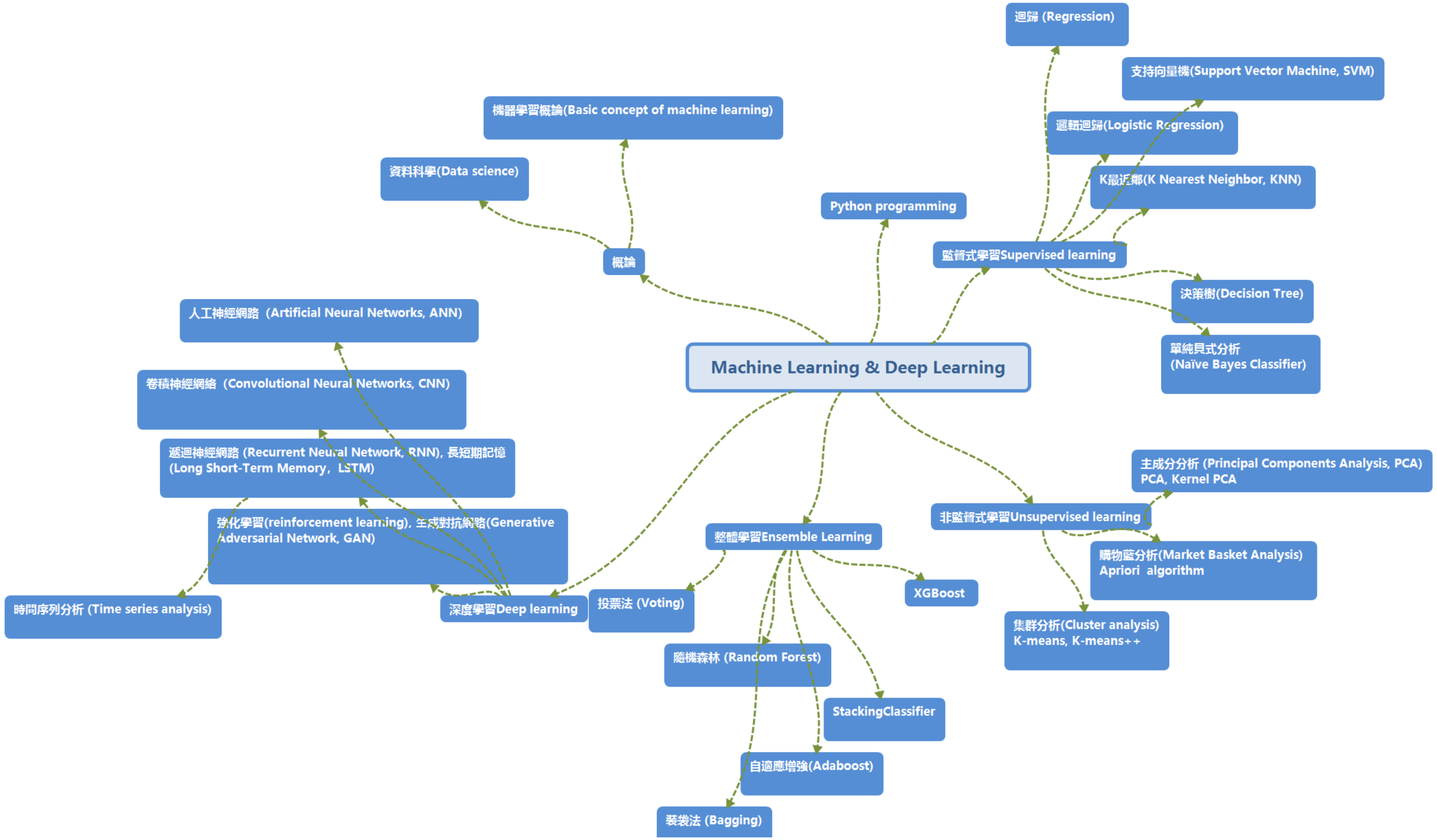
F1
- 人工神經網路（ Artificial Neural Networks, ANN ）

F2
- 卷積神經網絡（ Convolutional Neural Networks, CNN ）

F3
- 遞迴神經網路 (Recurrent Neural Network, RNN), 長短期記憶 (Long Short-Term Memory，LSTM)
  - 時間序列分析 (Time series analysis)

F4
- 強化學習(reinforcement learning), 生成對抗網路(Generative Adversarial Network, GAN)

學習心智圖



Machine Learning & Deep Learning
概論
機器學習概論(Basic concept of machine learning)
資料科學(Data science)
Python programming
監督式學習Supervised learning
迴歸 (Regression)
支持向量機(Support Vector Machine, SVM)
邏輯迴歸(Logistic Regression)
K最近鄰(K Nearest Neighbor, KNN)
決策樹(Decision Tree)
單純貝式分析
(Naïve Bayes Classifier)
非監督式學習Unsupervised learning
主成分分析 (Principal Components Analysis, PCA)
PCA, Kernel PCA
購物籃分析(Market Basket Analysis)
Apriori algorithm
集群分析(Cluster analysis)
K-means, K-means++
整體學習Ensemble Learning
投票法 (Voting)
XGBoost
隨機森林 (Random Forest)
StackingClassifier
自適應增強(Adaboost)
裝袋法 (Bagging)
深度學習Deep learning
人工神經網路 (Artificial Neural Networks, ANN)
卷積神經網絡 (Convolutional Neural Networks, CNN)
遞迴神經網路 (Recurrent Neural Network, RNN), 長短期記憶
(Long Short-Term Memory, LSTM)
強化學習(reinforcement learning), 生成對抗網路(Generative Adversarial Network, GAN)
時間序列分析 (Time series analysis)

個案實作

- Python綜合練習
  PM2.5分析
- 監督式學習
  不動產買賣實價分析
  facebook會員線上購物分析
  鳶尾花數據分析
- 非監督式學習
  購物中心顧客分群
  人口密度分群
  TESCO超市購物車分析
  葡萄酒顧客消費者行為分析
- 整體學習
  人力資源管理數據分析
- 深度學習
  銀行顧客流失預測
  手寫數字辨識
  圖片辨識
  IMDb情緒分析
  台灣百年地表溫度時間序列分析
  強化學習的CartPole
  生成式對抗網路的圖片生成器

Why learn python

為什麼要學習Python?

《哈佛商業評論》說「Data Scientist，數據科學家」是二十一世紀最性感的職業。根據104人力銀行預測2018年，前五大資料經濟職務需求，其中就有三個是資料分析相關職務，包括資料工程師、數據分析師與資料科學家。其求職者需要具備資料處理(ETL)工具開發經驗、熟悉R語言、Python、SQL、建置Hadoop或Spark平台經驗等等。

圖片資料來源：104人力銀行

https://www.tibame.com/offline/python

IEEE Spectrum發布程式語言排名：Python持續強勢、R語言逐漸消退

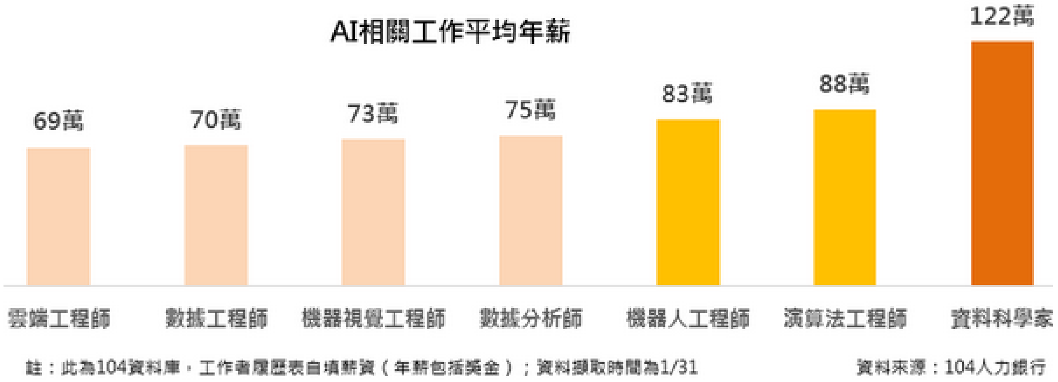
https://www.ithome.com.tw/news/125008

2018 AI元年！

人才缺口 3年翻倍

專業門檻高 薪資行情看俏

104突破傳統分類及年資限制，分析資料庫中，共1,575筆求職會員履歷填寫的AI相關工作經歷，發現AI相關工作平均年薪TOP 3依序是：資料科學家122萬、演算法工程師88萬、機器人工程師83萬。



https://plus.104.com.tw/activity/285e7f0b-0c97-42a3-8feb-bf0750b3abb6

資料科學家

2017全美最棒的工作

排名		職務名稱	年薪中位數（美元）
1	Data Scientist	資料科學家	$ 110,000
2	DevOps Engineer	軟體開發測試維運工程師	$ 110,000
3	Data Engineer	資料工程師	$ 106,000
4	Tax Manager	稅務管理師	$ 110,000
5	Analytics Manager	分析師	$ 112,000
6	HR Manager	人力資源管理師	$ 85,000
7	Database Administrator	資料庫管理員	$ 93,000
8	Strategy Manager	策略管理師	$ 130,000
9	UX Designer	使用者體驗設計師	$ 92,500
10	Solutions Architect	解決方案架構師	$ 125,000

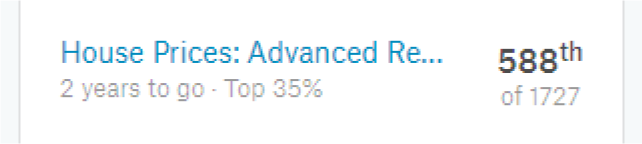
https://life.tw/?app=view&no=731066

評分

- 期中考 – 30% (選擇題20題)
- 期末報告 – 60% (二選一): [期中考後繳交名單]
  - Python程式實作機器學習或深度學習 (不分組, 每位修課同學均要繳交期末報告, 程式碼及資料集於github)
    - 題目可自選
  - 直接參與Kaggle競賽 (題目可於kaggle上自選, 或使用指定題, 仍要繳交期末報告, 程式碼及資料集於github) (可組團, 1 ~ 3人)
- 課堂表現 – 10%

Top	0 – 20%	21 – 30%	31 – 40%	41 – 50%	51% - 60%	61 – 70%	71% -
成績	100	95	90	80	70	60	0

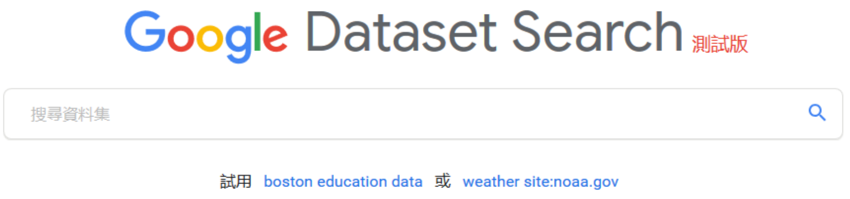
因kaggle上的成績會變動, 請上傳成績最好的top%截圖



期末報告 60%

資料集下載

- Kaggle
  - https://www.kaggle.com/
- • 政府開放資料平台
  - http://data.gov.tw/
- • 美國開放資料平台
  - https://www.data.gov/
- • 加州大學爾灣分校機器學習資料
  - http://archive.ics.uci.edu/ml/
- • Stanford Large Network Dataset Collection
  - https://snap.stanford.edu/data/
- Google Dataset Search
  - https://toolbox.google.com/datasetsearch



期末報告 60%

期末報告格式

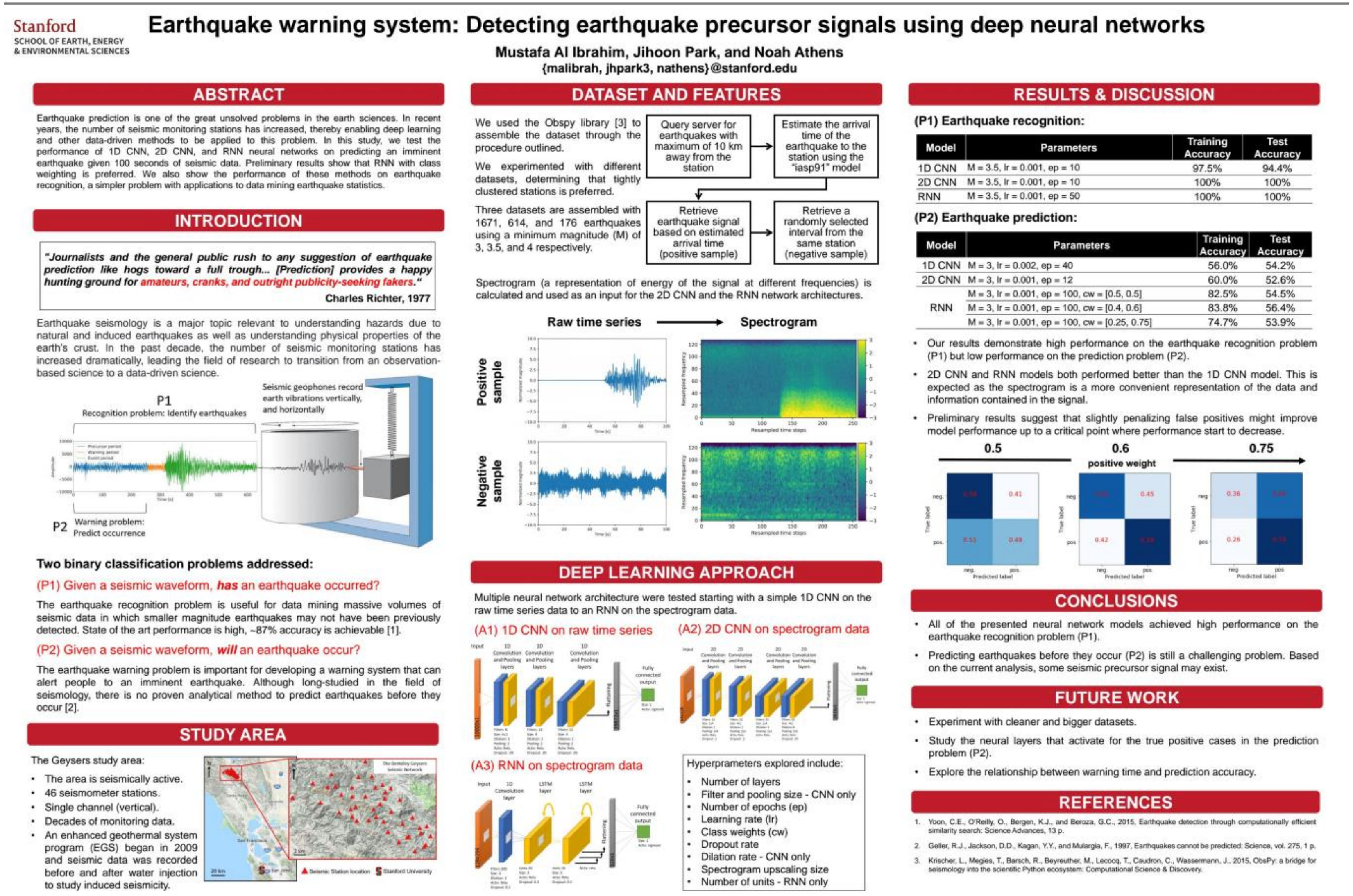
- 摘要
- 介紹 (研究背景及研究目的)
- 資料集介紹(含資料特徵)及資料集來源
- 資料預處理
- 機器學習或深度學習方法 (使用何種方法)
- 研究結果及討論 (含模型評估與改善)
- 結論
- 參考文獻

- 研究報告請以pdf格式上傳至github, 並同時上傳程式碼及資料集

期末報告
60%

期末報告
格式範例

Sample



Stanford
SCHOOL OF EARTH, ENERGY & ENVIRONMENTAL SCIENCES

Earthquake warning system: Detecting earthquake precursor signals using deep neural networks

Mustafa Al Ibrahim, Jihoon Park, and Noah Athens
{malibrah, jhpark3, nathens}@stanford.edu

ABSTRACT

Earthquake prediction is one of the great unsolved problems in the earth sciences. In recent years, the number of seismic monitoring stations has increased, thereby enabling deep learning and other data-driven methods to be applied to this problem. In this study, we test the performance of 1D CNN, 2D CNN, and RNN neural networks on predicting an imminent earthquake given 100 seconds of seismic data. Preliminary results show that RNN with class weighting is preferred. We also show the performance of these methods on earthquake recognition, a simpler problem with applications to data mining earthquake statistics.

INTRODUCTION

"Journalists and the general public rush to any suggestion of earthquake prediction like hogs toward a full trough... [Prediction] provides a happy hunting ground for amateurs, cranks, and outright publicity-seeking fakers."
Charles Richter, 1977

Earthquake seismology is a major topic relevant to understanding hazards due to natural and induced earthquakes as well as understanding physical properties of the earth's crust. In the past decade, the number of seismic monitoring stations has increased dramatically, leading the field of research to transition from an observation-based science to a data-driven science.

Seismic geophones record earth vibrations vertically, and horizontally
P1
Recognition problem: Identify earthquakes
P2 Warning problem: Predict occurrence

Two binary classification problems addressed:

(P1) Given a seismic waveform, *has* an earthquake occurred?

The earthquake recognition problem is useful for data mining massive volumes of seismic data in which smaller magnitude earthquakes may not have been previously detected. State of the art performance is high, ~87% accuracy is achievable [1].

(P2) Given a seismic waveform, *will* an earthquake occur?

The earthquake warning problem is important for developing a warning system that can alert people to an imminent earthquake. Although long-studied in the field of seismology, there is no proven analytical method to predict earthquakes before they occur [2].

STUDY AREA

The Geysers study area:
- The area is seismically active.
- 46 seismometer stations.
- Single channel (vertical).
- Decades of monitoring data.
- An enhanced geothermal system program (EGS) began in 2009 and seismic data was recorded before and after water injection to study induced seismicity.

Seismic Station location Stanford University

DATASET AND FEATURES

We used the Obspy library [3] to assemble the dataset through the procedure outlined.

We experimented with different datasets, determining that tightly clustered stations is preferred.

Three datasets are assembled with 1671, 614, and 176 earthquakes using a minimum magnitude (M) of 3, 3.5, and 4 respectively.

Query server for earthquakes with maximum of 10 km away from the station → Estimate the arrival time of the earthquake to the station using the "iasp91" model → Retrieve earthquake signal based on estimated arrival time (positive sample) → Retrieve a randomly selected interval from the same station (negative sample)

Spectrogram (a representation of energy of the signal at different frequencies) is calculated and used as an input for the 2D CNN and the RNN network architectures.

Raw time series → Spectrogram
Positive sample
Negative sample

DEEP LEARNING APPROACH

Multiple neural network architecture were tested starting with a simple 1D CNN on the raw time series data to an RNN on the spectrogram data.

(A1) 1D CNN on raw time series

(A2) 2D CNN on spectrogram data

(A3) RNN on spectrogram data

Hyperparameters explored include:
- Number of layers
- Filter and pooling size - CNN only
- Number of epochs (ep)
- Learning rate (lr)
- Class weights (cw)
- Dropout rate
- Dilation rate - CNN only
- Spectrogram upscaling size
- Number of units - RNN only

RESULTS & DISCUSSION

(P1) Earthquake recognition:

Model	Parameters	Training Accuracy	Test Accuracy
1D CNN	M = 3.5, lr = 0.001, ep = 10	97.5%	94.4%
2D CNN	M = 3.5, lr = 0.001, ep = 10	100%	100%
RNN	M = 3.5, lr = 0.001, ep = 50	100%	100%

(P2) Earthquake prediction:

Model	Parameters	Training Accuracy	Test Accuracy
1D CNN	M = 3, lr = 0.002, ep = 40	56.0%	54.2%
2D CNN	M = 3, lr = 0.001, ep = 12	60.0%	52.6%
RNN	M = 3, lr = 0.001, ep = 100, cw = [0.5, 0.5]	82.5%	54.5%
	M = 3, lr = 0.001, ep = 100, cw = [0.4, 0.6]	83.8%	56.4%
	M = 3, lr = 0.001, ep = 100, cw = [0.25, 0.75]	74.7%	53.9%

- Our results demonstrate high performance on the earthquake recognition problem (P1) but low performance on the prediction problem (P2).
- 2D CNN and RNN models both performed better than the 1D CNN model. This is expected as the spectrogram is a more convenient representation of the data and information contained in the signal.
- Preliminary results suggest that slightly penalizing false positives might improve model performance up to a critical point where performance start to decrease.

0.5 0.6 0.75
positive weight

CONCLUSIONS

- All of the presented neural network models achieved high performance on the earthquake recognition problem (P1).
- Predicting earthquakes before they occur (P2) is still a challenging problem. Based on the current analysis, some seismic precursor signal may exist.

FUTURE WORK

- Experiment with cleaner and bigger datasets.
- Study the neural layers that activate for the true positive cases in the prediction problem (P2).
- Explore the relationship between warning time and prediction accuracy.

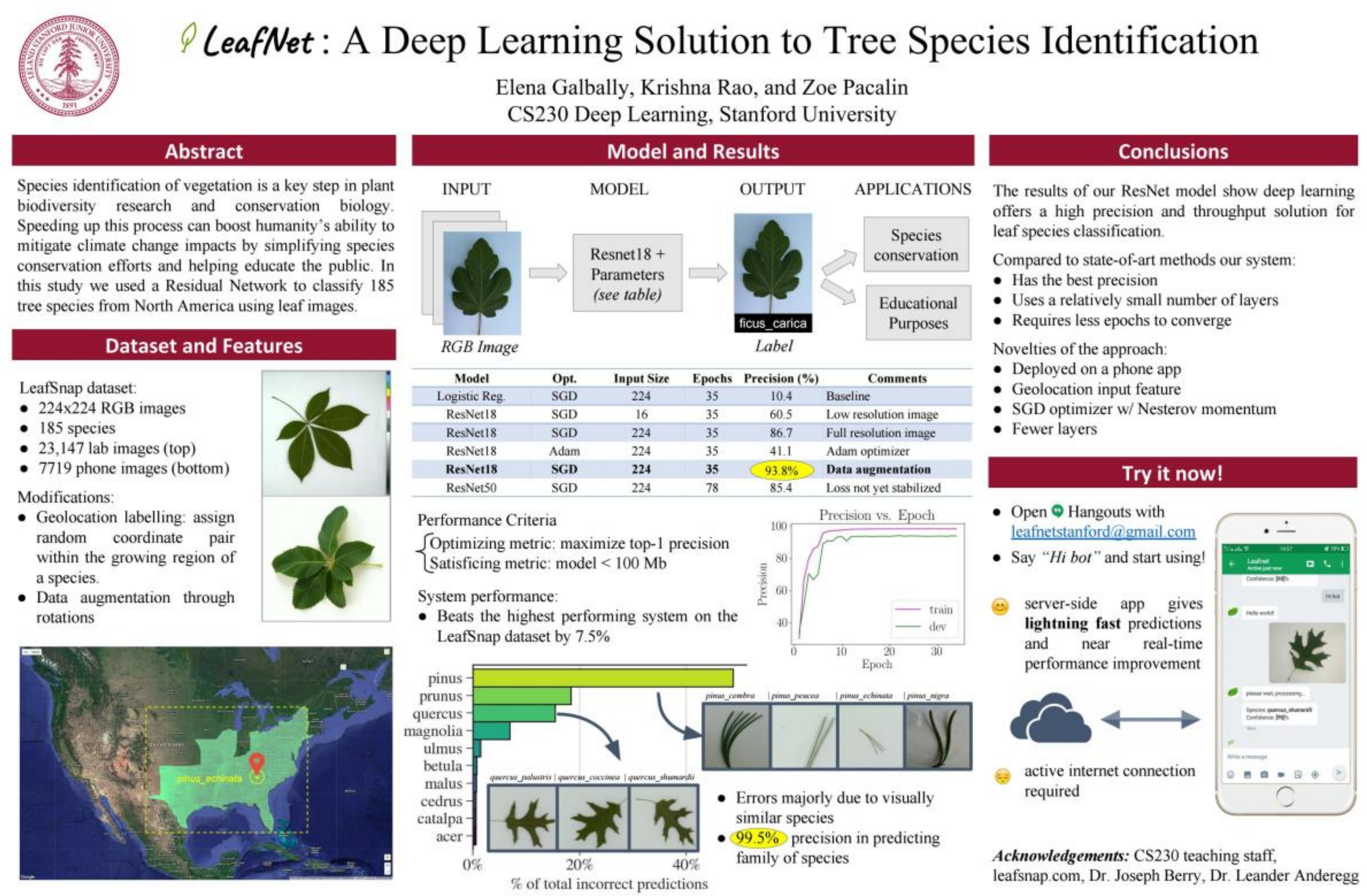
REFERENCES

1. Yoon, C.E., O'Reilly, O., Bergen, K.J., and Beroza, G.C., 2015, Earthquake detection through computationally efficient similarity search: Science Advances, 13 p.
2. Geller, R.J., Jackson, D.D., Kagan, Y.Y., and Mulargia, F., 1997, Earthquakes cannot be predicted: Science, vol. 275, 1 p.
3. Krischer, L., Megies, T., Barsch, R., Beyreuther, M., Lecocq, T., Caudron, C., Wassermann, J., 2015, ObsPy: a bridge for seismology into the scientific Python ecosystem: Computational Science & Discovery.

Sample

期末報告 60%

期末報告格式範例



LeafNet: A Deep Learning Solution to Tree Species Identification

Elena Galbally, Krishna Rao, and Zoe Pacalin
CS230 Deep Learning, Stanford University

Abstract

Species identification of vegetation is a key step in plant biodiversity research and conservation biology. Speeding up this process can boost humanity's ability to mitigate climate change impacts by simplifying species conservation efforts and helping educate the public. In this study we used a Residual Network to classify 185 tree species from North America using leaf images.

Dataset and Features

LeafSnap dataset:
- 224x224 RGB images
- 185 species
- 23,147 lab images (top)
- 7719 phone images (bottom)

Modifications:
- Geolocation labelling: assign random coordinate pair within the growing region of a species.
- Data augmentation through rotations

pinus_echinata

Model and Results

INPUT | MODEL | OUTPUT | APPLICATIONS

RGB Image → Resnet18 + Parameters (see table) → ficus_carica Label → Species conservation / Educational Purposes

Model	Opt.	Input Size	Epochs	Precision (%)	Comments
Logistic Reg.	SGD	224	35	10.4	Baseline
ResNet18	SGD	16	35	60.5	Low resolution image
ResNet18	SGD	224	35	86.7	Full resolution image
ResNet18	Adam	224	35	41.1	Adam optimizer
ResNet18	**SGD**	**224**	**35**	**93.8%**	**Data augmentation**
ResNet50	SGD	224	78	85.4	Loss not yet stabilized

Performance Criteria
- Optimizing metric: maximize top-1 precision
- Satisficing metric: model < 100 Mb

System performance:
- Beats the highest performing system on the LeafSnap dataset by 7.5%

Precision vs. Epoch (Precision; Epoch; train, dev)

pinus, prunus, quercus, magnolia, ulmus, betula, malus, cedrus, catalpa, acer
0% 20% 40%
% of total incorrect predictions

quercus_palustris | quercus_coccinea | quercus_shumardii
pinus_cembra | pinus_peucea | pinus_echinata | pinus_nigra

- Errors majorly due to visually similar species
- 99.5% precision in predicting family of species

Conclusions

The results of our ResNet model show deep learning offers a high precision and throughput solution for leaf species classification.

Compared to state-of-art methods our system:
- Has the best precision
- Uses a relatively small number of layers
- Requires less epochs to converge

Novelties of the approach:
- Deployed on a phone app
- Geolocation input feature
- SGD optimizer w/ Nesterov momentum
- Fewer layers

Try it now!

- Open Hangouts with leafnetstanford@gmail.com
- Say *"Hi bot"* and start using!

server-side app gives **lightning fast** predictions and near real-time performance improvement

active internet connection required

Acknowledgements: CS230 teaching staff, leafsnap.com, Dr. Joseph Berry, Dr. Leander Anderegg

Kaggle競賽

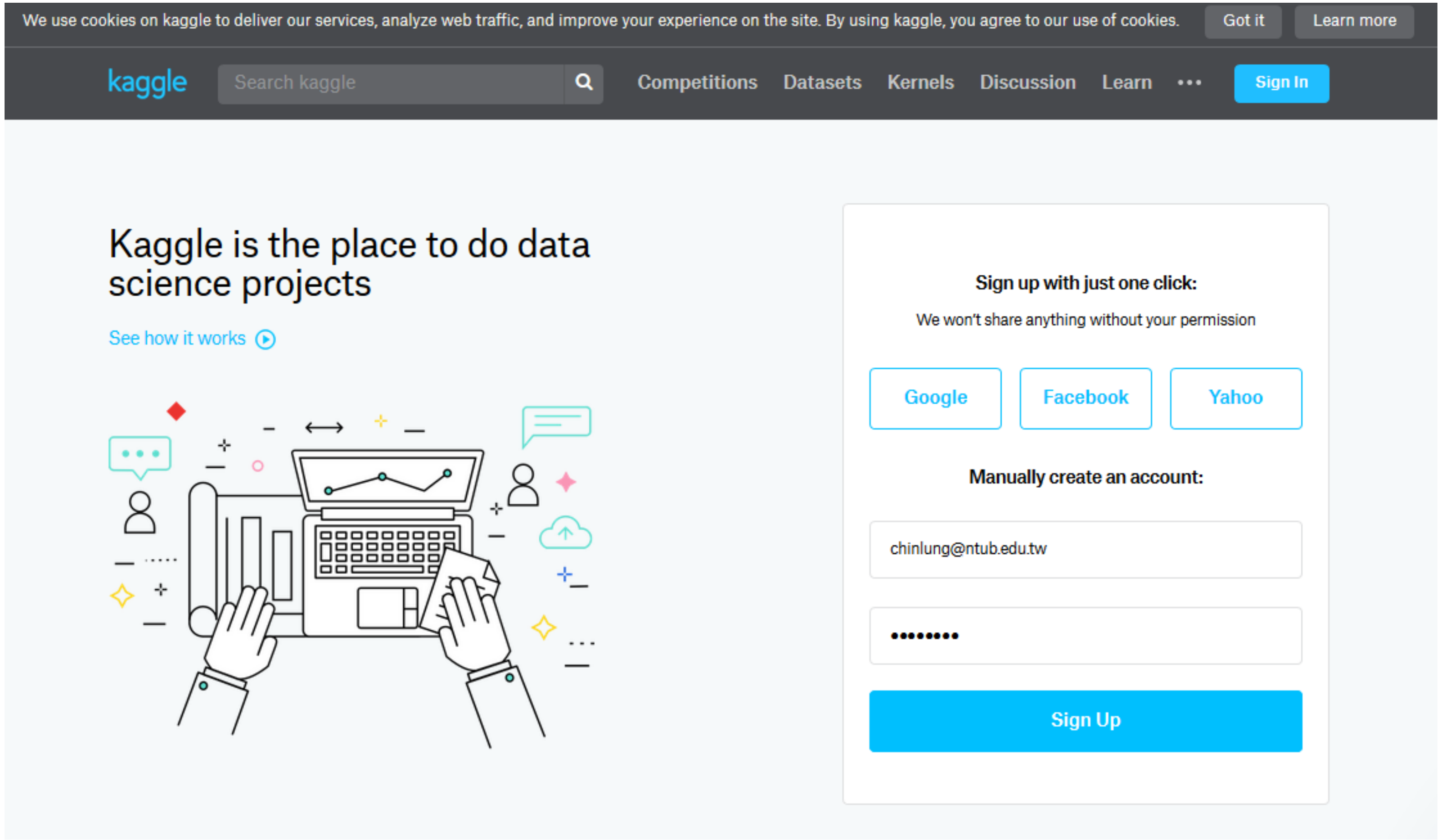
- **三選一**

- 波士頓房價預測
  - https://www.kaggle.com/c/house-prices-advanced-regression-techniques

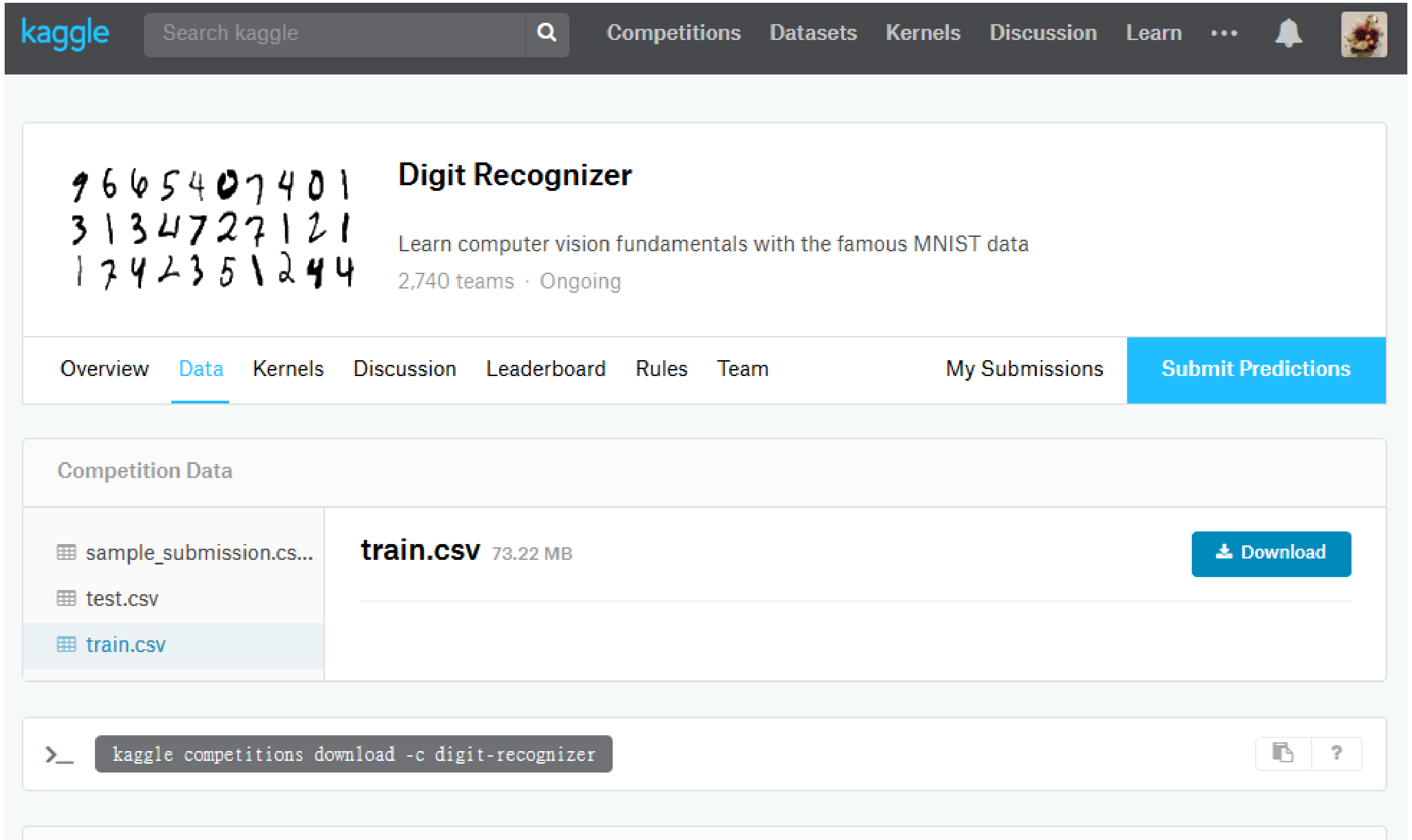
- 鐵達尼號生存預測
  - https://www.kaggle.com/c/titanic

- 自選Competition on Kaggle


Kaggle
企業
kaggle
Kaggle是一個數據建模和數據分析競賽平台。企業和研究者可在其上發布數據，統計學者和數據挖掘專家可在其上進行競賽以產生最好的模型。這一眾包模式依賴於這一事實，即有眾多策略可以用於解決幾乎所有預測建模的問題，而研究者不可能在一開始就了解什麼方法對於特定問題是最為有效的。 維基百科
創辦人： 安東尼·戈德布盧姆
創立於： 2010 年 4 月
執行長： 安東尼·戈德布盧姆 (2010 年 4 月–)
總部： 美國加利福尼亞州舊金山
上級機構： Google


We use cookies on kaggle to deliver our services, analyze web traffic, and improve your experience on the site. By using kaggle, you agree to our use of cookies.
Got it
Learn more
kaggle
Search kaggle
Competitions
Datasets
Kernels
Discussion
Learn
Sign In
Kaggle is the place to do data science projects
See how it works
Sign up with just one click:
We won't share anything without your permission
Google
Facebook
Yahoo
Manually create an account:
chinlung@ntub.edu.tw
Sign Up

Kaggle


kaggle
Search kaggle
Competitions
Datasets
Kernels
Discussion
Learn
Digit Recognizer
Learn computer vision fundamentals with the famous MNIST data
2,740 teams · Ongoing
Overview
Data
Kernels
Discussion
Leaderboard
Rules
Team
My Submissions
Submit Predictions
Competition Data
sample_submission.cs...
test.csv
train.csv
train.csv 73.22 MB
Download
kaggle competitions download -c digit-recognizer

Kaggle

jupyter CNN for MNIST-kaggle Last Checkpoint: 32分鐘前 (autosaved)

File Edit View Insert Cell Kernel Help | Trusted | Python 3

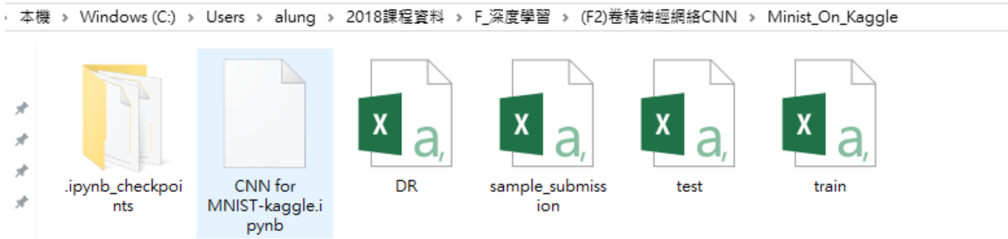
資料匯入與預處理

In [1]:
```python
import numpy as np
from keras.datasets import mnist

from keras.utils import np_utils
from keras.models import Sequential
from keras.layers import Dense, Activation, Conv2D, MaxPooling2D, Dropout, Flatten
```
```
Using TensorFlow backend.
```

In [2]:
```python
import pandas as pd
train = pd.read_csv('train.csv')
test = pd.read_csv('test.csv')
```

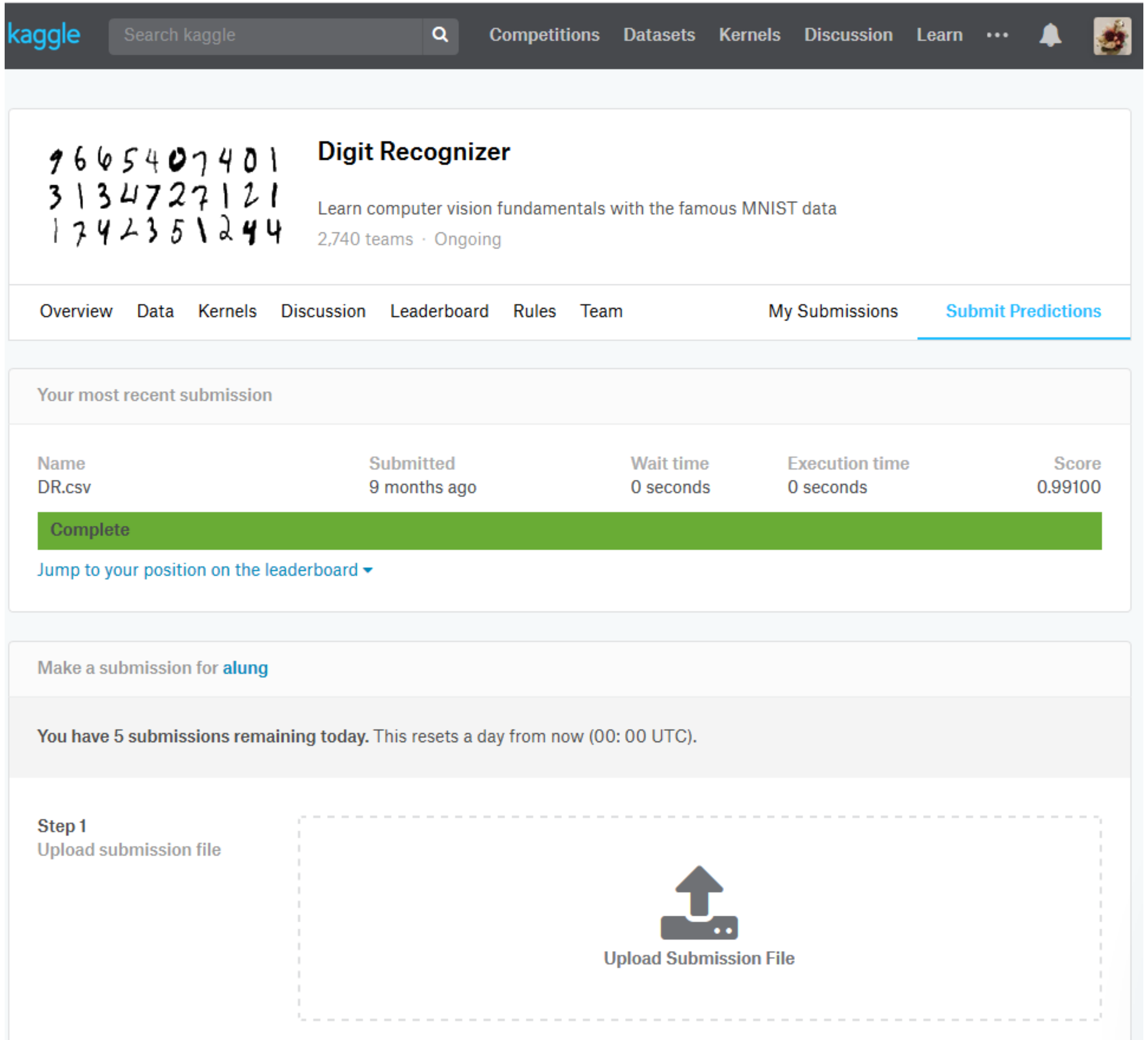
In [3]:
```python
X_train = (train.ix[:,1:].values).astype('float32') # all pixel values
y_train = train.ix[:,0].values.astype('int32') # only labels i.e targets digits
X_test = test.values.astype('float32')
```
```
C:\Users\alung\Anaconda3\envs\gpu\lib\site-packages\ipykernel_launcher.py:1: DeprecationWarning:
.ix is deprecated. Please use
.loc for label based indexing or
.iloc for positional indexing
```



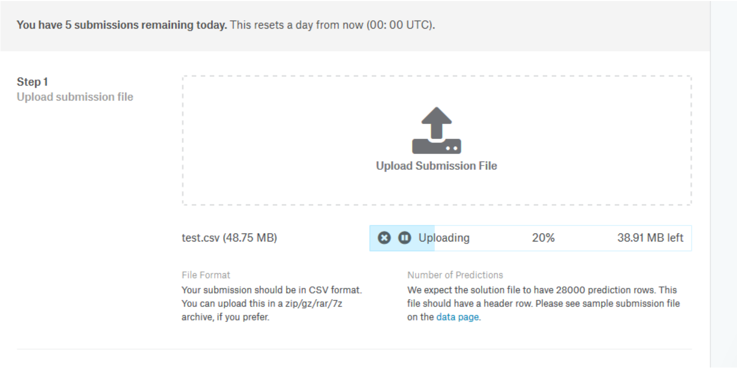
In [29]:
```python
predictions = model.predict_classes(X_test, verbose=0)

submissions=pd.DataFrame({"ImageId": list(range(1,len(predictions)+1)),
                         "Label": predictions})
submissions.to_csv("DR.csv", index=False, header=True)
```

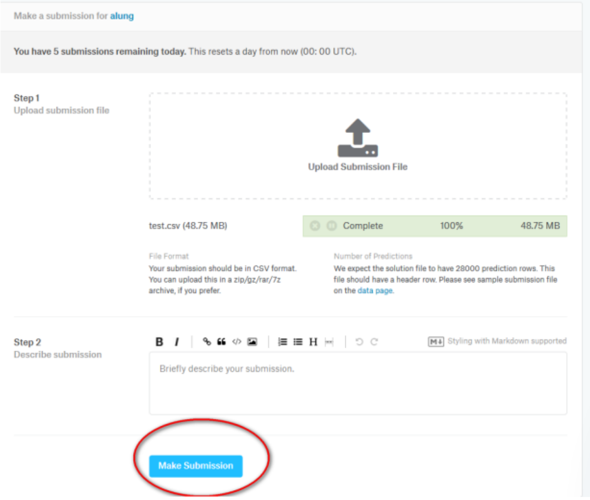
Kaggle


kaggle
Search kaggle
Competitions
Datasets
Kernels
Discussion
Learn
Digit Recognizer
Learn computer vision fundamentals with the famous MNIST data
2,740 teams · Ongoing
Overview
Data
Kernels
Discussion
Leaderboard
Rules
Team
My Submissions
Submit Predictions
Your most recent submission
Name
DR.csv
Submitted
9 months ago
Wait time
0 seconds
Execution time
0 seconds
Score
0.99100
Complete
Jump to your position on the leaderboard
Make a submission for alung
You have 5 submissions remaining today. This resets a day from now (00: 00 UTC).
Step 1
Upload submission file
Upload Submission File

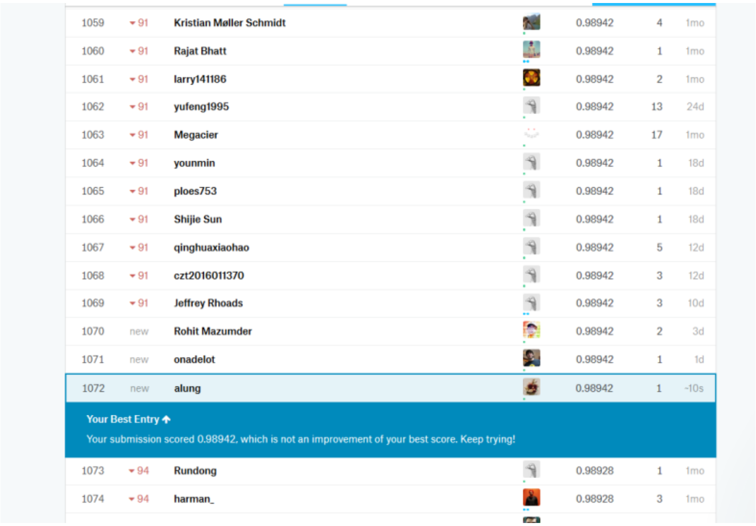
Kaggle



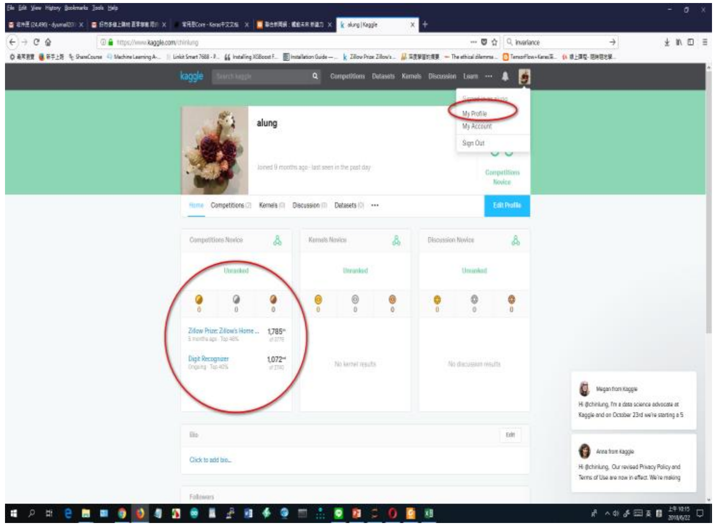
You have 5 submissions remaining today. This resets a day from now (00: 00 UTC).
Step 1
Upload submission file
Upload Submission File
test.csv (48.75 MB)
Uploading
20%
38.91 MB left
File Format
Your submission should be in CSV format. You can upload this in a zip/gz/rar/7z archive, if you prefer.
Number of Predictions
We expect the solution file to have 28000 prediction rows. This file should have a header row. Please see sample submission file on the data page.



Make a submission for alung
You have 5 submissions remaining today. This resets a day from now (00: 00 UTC).
Step 1
Upload submission file
Upload Submission File
test.csv (48.75 MB)
Complete
100%
48.75 MB
Step 2
Describe submission
Briefly describe your submission.
Make Submission



1059 91 Kristian Møller Schmidt 0.98942 4 1mo
1060 91 Rajat Bhatt 0.98942 1 1mo
1061 91 larry141186 0.98942 2 1mo
1062 91 yufeng1995 0.98942 13 24d
1063 91 Megacier 0.98942 17 1mo
1064 91 younmin 0.98942 1 18d
1065 91 ploes753 0.98942 1 18d
1066 91 Shijie Sun 0.98942 1 18d
1067 91 qinghuaxiaohao 0.98942 5 12d
1068 91 czt2016011370 0.98942 3 12d
1069 91 Jeffrey Rhoads 0.98942 3 10d
1070 new Rohit Mazumder 0.98942 2 3d
1071 new onadelot 0.98942 1 1d
1072 new alung 0.98942 1 ~10s
Your Best Entry
Your submission scored 0.98942, which is not an improvement of your best score. Keep trying!
1073 94 Rundong 0.98928 1 1mo
1074 94 harman_ 0.98928 3 1mo

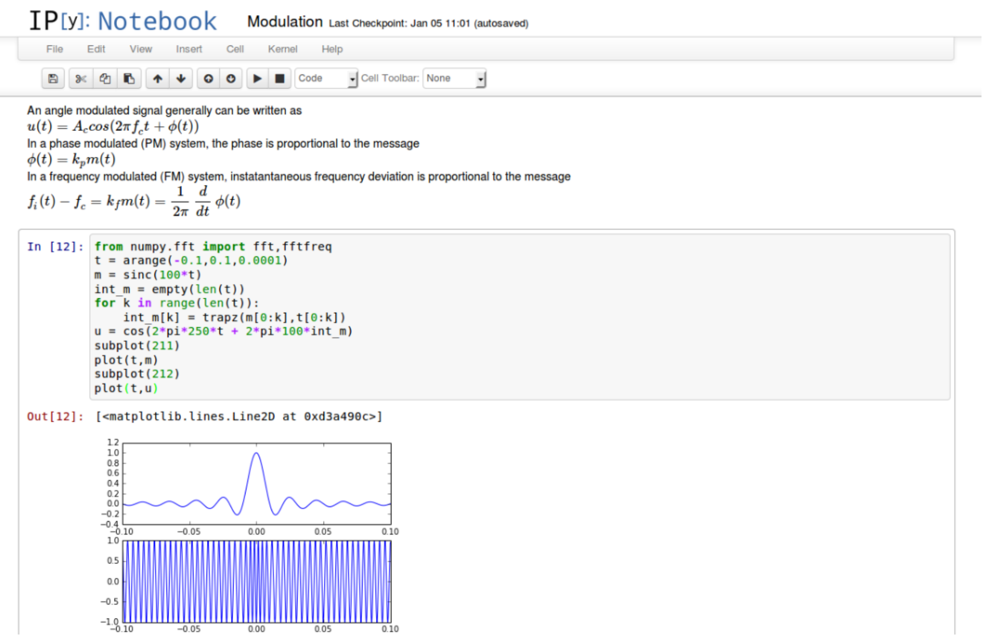


kaggle
alung
Edit Profile

上課平台

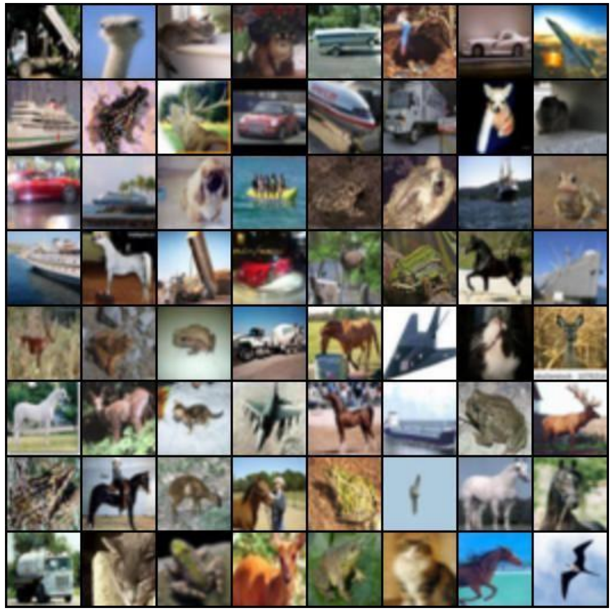
課程將以windows 10為教學作業系統

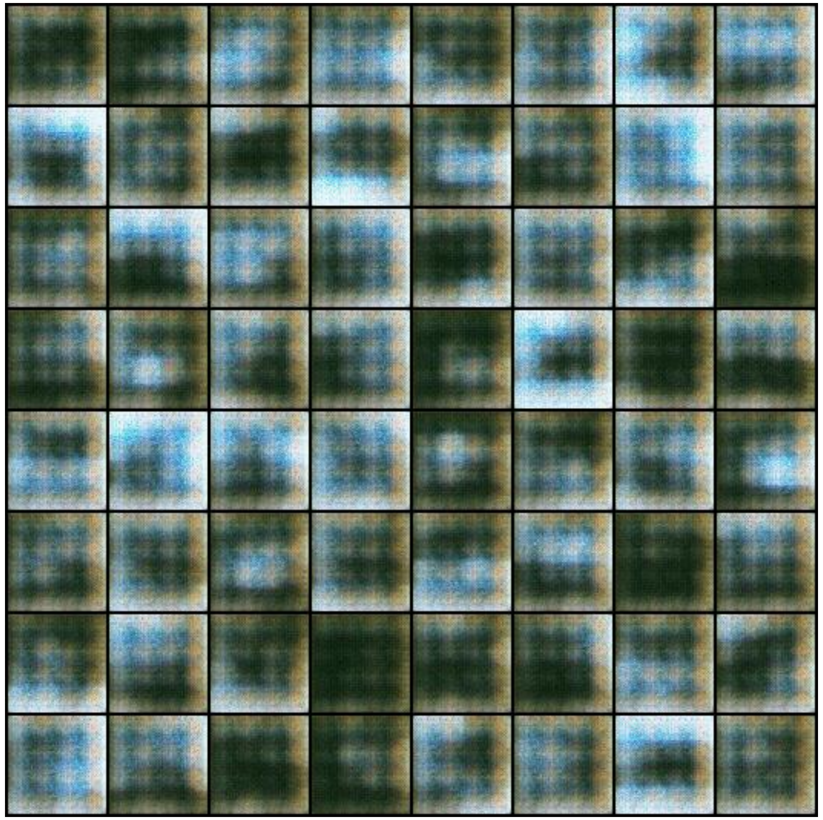
使用Anaconda 之 Jupyter Notebook實作機器學習與深度學習



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

生成式對抗網路(GAN),第一張是真圖, 第二張的生成的假圖



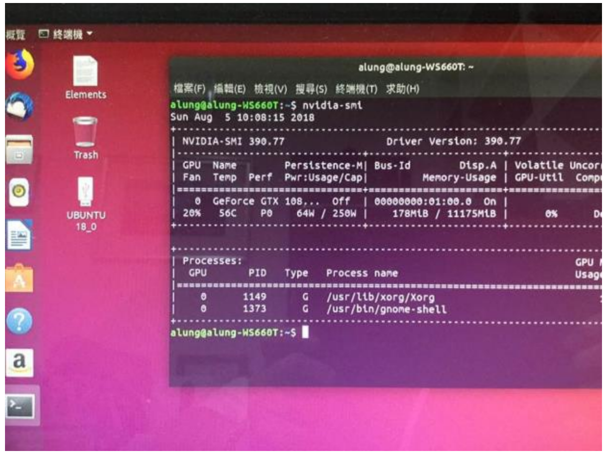


GAN

CPU on windows 10 : about 8 hours +
GPU on windows 10: about 21 mins
GPU on Ubuntu 18.04: about 14 mins

結論: 安裝GPU on linux is a good choice for deep learning.
測試環境

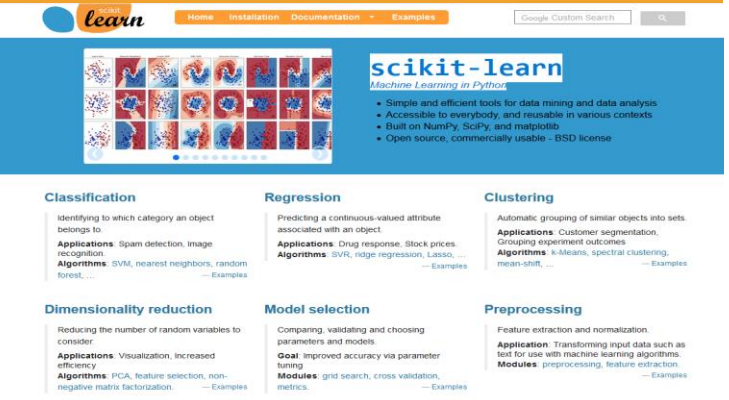
主機 : WS660T
中央處理器 : Intel Xeon E3-1245V6
記憶體 : 32GB DDR4 2400 ECC UDIMM
顯示卡 ： Turbo GTX 1080 Ti （ 3584 CUDA Core , 11GB GDDR5X ）
硬碟 : 512GB M.2 2280 SATA3 SSD



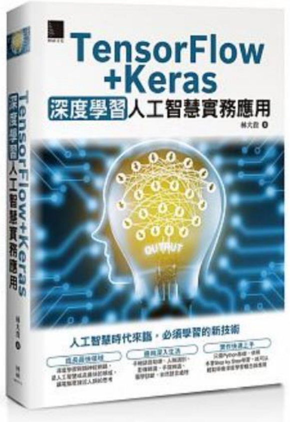
提醒

- 課程講義投影片, python程式碼與數據集皆以RAR壓縮檔上傳至系上ftp server, 需以密碼開啟 (密碼於課堂上公告)
  - 課程講義投影片-.pdf
  - 程式碼 - .ipynb
  - 數據集 - .csv
    - 部分數據集需由網路上下載, 若有亂碼產生, 可google查詢轉換方法.

- 善用ｇｏｏｇｌｅ找資源（教學影片，網站， ｆｂ社團：台灣「人工智慧」社團）



- scikit-learn　Machine Learning in Python（推薦）
  - http://scikit-learn.org/stable/index.html
- TensorFlow+Keras深度學習人工智慧實務應用 （推薦）
  - https://www.books.com.tw/products/0010754327



提醒

請同學攜帶ｕｓｂ隨身碟儲存講義投影片, 程式碼與數據集

勿將課程所使用之講義投影片, 程式碼與數據集儲存在學校公共電腦內。

提醒

- 本講義所使用之圖片, 表格, 文字, 內容, 書籍資料, 引用統計資料與程式碼及數據集資料等, 除自製外，其智慧財產權為原網站, 作者,公司所擁有。
- 講義投影片, 程式碼與數據集僅供教學使用, 請同學勿將課程所使用之講義投影片, 程式碼與數據集放在網路上供人下載及分享, 也請勿做商業用途。

Movie Time

- https://www.youtube.com/watch?v=h2jSSAObA8s&t=7s
- 1:30 -