

期末報告事項

2023

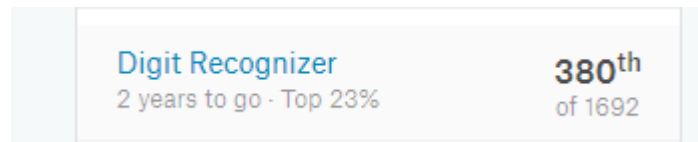
Machine Learning and Deep Learning

評分

- 期中考 – 30% (選擇題20題)
- 期末報告 – 60% (二選一): [5/2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與程式碼)
 - 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, 仍要繳交期末報告與程式碼) (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



House Prices: Advanced Re... 588th of 1727

2 years to go - Top 35%

學號與姓名	題目	是否為 <u>kaggle</u> 比賽	
		Yes	No
A1234 王小明 A1235 陳小華 A1236 王曉英	鐵達尼號生存預測	V	
A1237 林小花	Friday 顧客購物車分析		V

報告繳交

- 報告繳交日期: 2021年6月13 (期末考週)•
- 繳交方式: 上傳ftp server
- 研究報告包括二個
 - 報告: pdf格式儲存, 第一頁為學號, 姓名及題目, 檔名為: 學號.pdf
 - 程式: 檔名為: 學號.ipynb
 - 若為kaggle比賽, 則在報告上附上比賽成績

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>

Google Dataset Search 測試版

搜尋資料集



試用 [boston education data](#) 或 [weather site: noaa.gov](#)

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



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, natheens}@stanford.edu

ABSTRACT

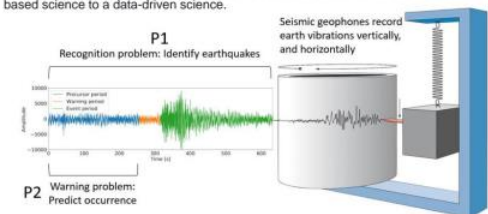
Earthquake prediction is one of the great unsolved problems in the earth sciences. In recent years, the number of seismic monitoring stations has increased, thereby enabling deep learning and other data-driven methods to be applied to this problem. In this study, we test the performance of 1D CNN, 2D 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.



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.

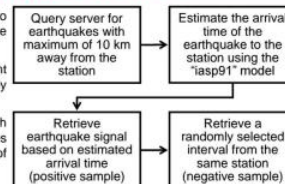


DATASET AND FEATURES

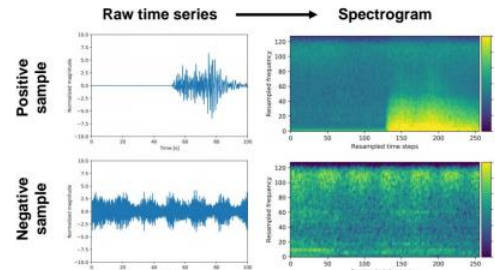
We used the Obspy library [3] to assemble the dataset through the procedure outlined.

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.

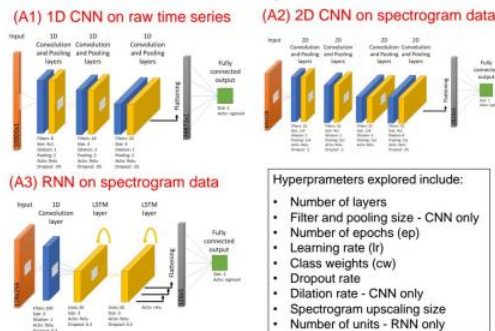


Spectrogram (a representation of energy of the signal at different frequencies) is calculated and used as an input for the 2D CNN and the RNN network architectures.



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.



RESULTS & DISCUSSION

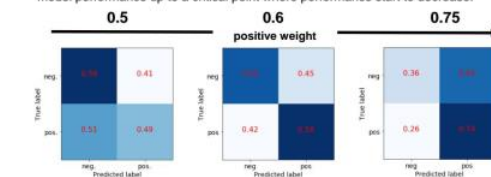
(P1) Earthquake recognition:

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%
	M = 3, lr = 0.001, ep = 100, cw = [0.5, 0.5]	82.5%	54.5%
RNN	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.



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

- Yoon, C.E., O'Reilly, O., Bergen, K.J., and Berzosa, G.C., 2015, Earthquake detection through computationally efficient similarity search: Science Advances, 13 p.
- Geller, R.J., Jackson, D.D., Kagan, Y.Y., and Mulargia, F., 1997, Earthquakes cannot be predicted: Science, vol. 275, 1 p.
- 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.

期末報告
60%

期末報告
格式範例

Sample



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:

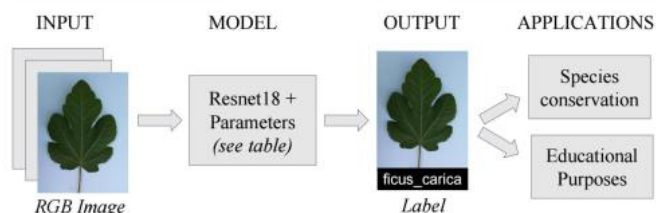
- 224x224 RGB images
- 185 species
- 23,147 lab images (top)
- 7719 phone images (bottom)

Modifications:

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



Model and Results



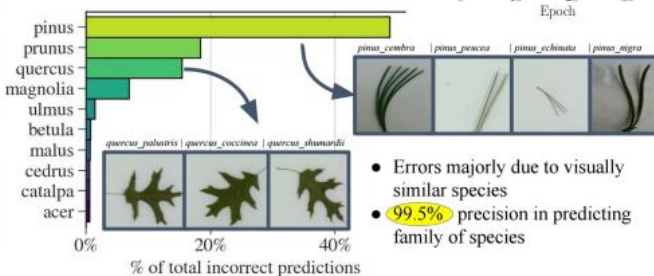
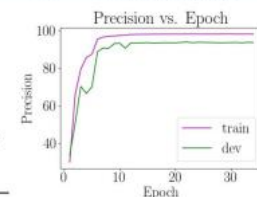
Model	Opt.	Input Size	Epochs	Precision (%)	Comments
Logistic Reg.	SGD	224	35	10.4	Baseline
ResNet18	SGD	16	35	60.5	Low resolution image
ResNet18	SGD	224	35	86.7	Full resolution image
ResNet18	Adam	224	35	41.1	Adam optimizer
ResNet18	SGD	224	35	93.8%	Data augmentation
ResNet50	SGD	224	78	85.4	Loss not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



- 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

The screenshot shows the Kaggle website interface for the 'Digit Recognizer' competition. At the top, there's a navigation bar with the Kaggle logo, a search bar, and links to Competitions, Datasets, Kernels, Discussion, and Learn. The main header area features a grid of handwritten digits (9, 6, 6, 5, 4, 0, 7, 4, 0, 1, 3, 1, 3, 4, 7, 2, 7, 1, 2, 1, 1, 7, 4, 2, 3, 5, 1, 2, 4, 4) and the competition title 'Digit Recognizer'. Below the title, it says 'Learn computer vision fundamentals with the famous MNIST data' and '2,740 teams · Ongoing'. A navigation bar below the header includes links for Overview, Data (which is highlighted), Kernels, Discussion, Leaderboard, Rules, Team, My Submissions, and a prominent blue 'Submit Predictions' button. The 'Competition Data' section lists three files: 'sample_submission.csv', 'test.csv', and 'train.csv'. The 'train.csv' file is selected, showing its size as 73.22 MB and a 'Download' button. At the bottom, there's a terminal-like interface with a command prompt showing the command 'kaggle competitions download -c digit-recognizer' and icons for file operations and help.

kaggle Search kaggle Competitions Datasets Kernels Discussion Learn ...

9 6 6 5 4 0 7 4 0 1
3 1 3 4 7 2 7 1 2 1
1 7 4 2 3 5 1 2 4 4

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.csv	train.csv 73.22 MB Download
test.csv	
train.csv	

```
>_ 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]: 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]: import pandas as pd
train = pd.read_csv('train.csv')
test = pd.read_csv('test.csv')
```

```
In [3]: 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

本機 > Windows (C:) > Users > alung > 2018課程資料 > F_深度學習 > (F2)卷积神经网络CNN > Minist_On_Kaggle

.ipynb_checkpoints

CNN for MNIST-kaggle.ipynb

DR

sample_submission

test

train

```
In [29]: 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

Q

Competitions

Datasets

Kernels

Discussion

Learn

...

9665407401

3134727121

1742351244

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	Submitted	Wait time	Execution time	Score
DR.csv	9 months ago	0 seconds	0 seconds	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



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



test.csv (48.75 MB) Complete 100% 48.75 MB

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](#).

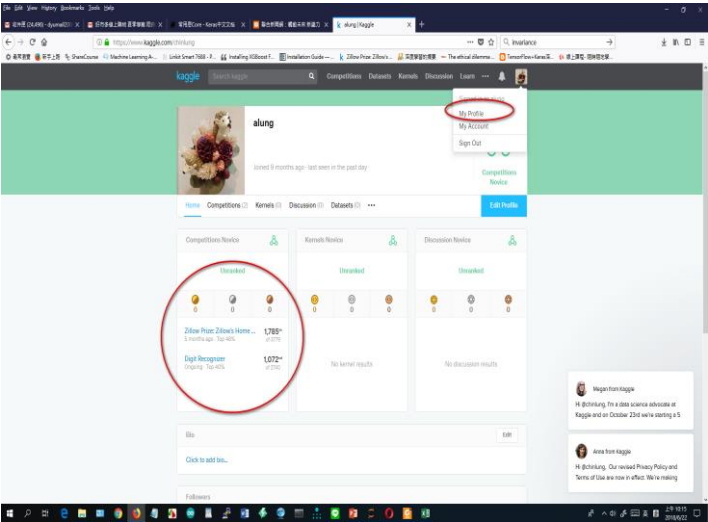
Step 2
Describe submission

B / [Rich Text Editor] [MD] Styling with Markdown supported

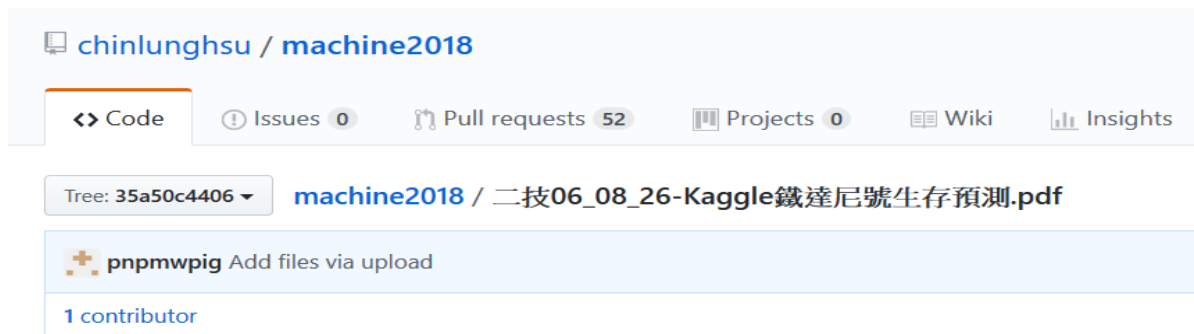
Briefly describe your submission.

Make Submission

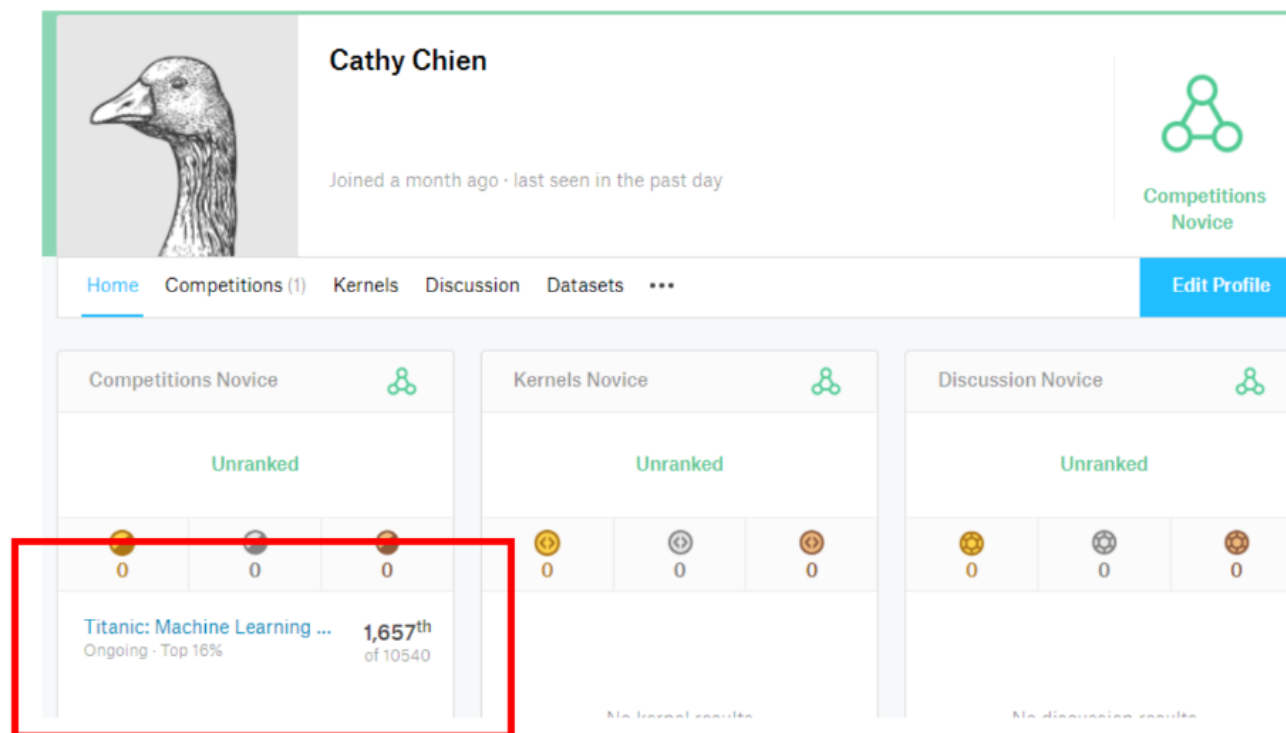
1059	▼ 91	Kristian Møller Schmidt		0.98942	4	1mo
1060	▼ 91	Rajat Bhatt		0.98942	1	1mo
1061	▼ 91	larry141186		0.98942	2	1mo
1062	▼ 91	yufeng1995		0.98942	13	24d
1063	▼ 91	Megaciel		0.98942	17	1mo
1064	▼ 91	younmin		0.98942	1	18d
1065	▼ 91	ploes753		0.98942	1	18d
1066	▼ 91	Shijie Sun		0.98942	1	18d
1067	▼ 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



教學成果01



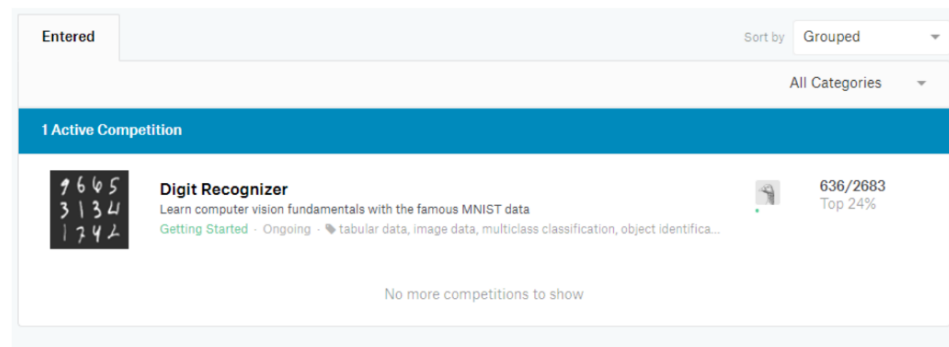
Kaggle 排名如下圖



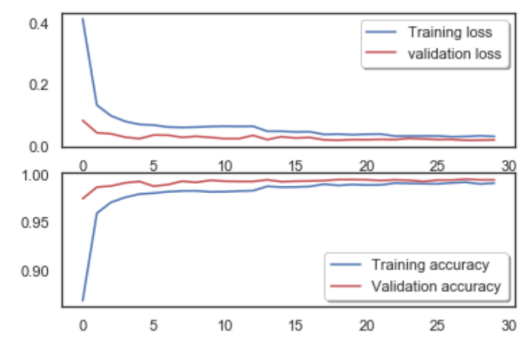
教學成果02

Result

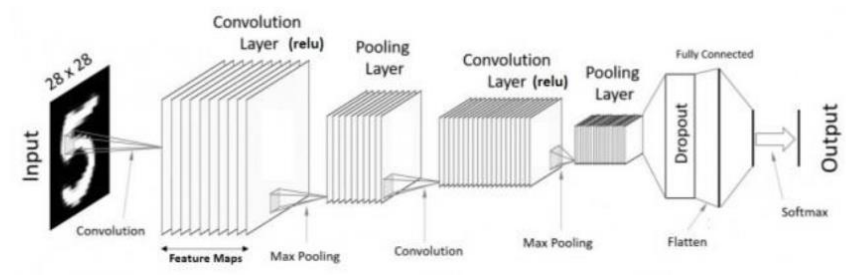
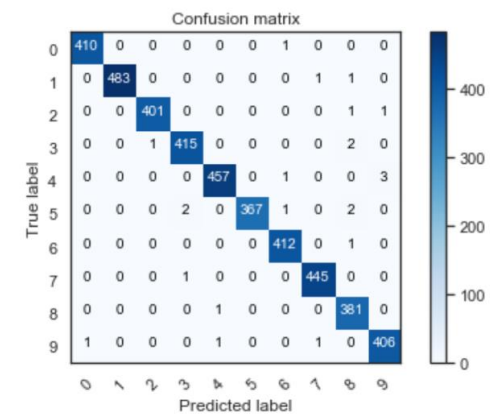
- Competition Ranking
– 636/2683 Top 24%



Train/Validation loss and accuracy



Confusion matrix



教學成果03

CHANGED 2 MONTHS AGO

摘要

在一學期的機器學習與深度學習課程後，於Kaggle平台上選定一資料集進行分析。

介紹

研究目的

本次選定 **Black Friday** 這個資料集，Black Friday 在美國用來指每年感恩節之後的第一天。這一天通常被認為標誌著聖誕購物期的正式開始，被看作是每年零售業聖誕銷售業績的晴雨表，也是一年中各個商家最看重也是最繁忙的日子之一。中文又稱作黑色購物節。

本次期望由 Kaggle 上的 dataset 進行機器學習與深度學習練習，並選定一種方法進行預測與建議。

The dataset here is a sample of the transactions made in a retail store. The store wants to know better the customer purchase behaviour against different products. Specifically, here the problem is a regression problem where we are trying to predict the dependent variable (the amount of purchase) with the help of the information contained in the other variables.

---截自Black Friday中的Description

研究目標

- 分析資訊
 - 主要客群為男或女、結婚與否
- 回歸與分類，並預測
 - 試用回歸法預測購買量
 - 試用分類法預測是否為較高消費群

摘要

介紹

實作解析

研究結果與討論

結論

參考文獻

Expand all

Back to top

Go to bottom

```
1 def show_train_history(train_history, train, validation):
2     plt.plot(train_history.history[train])
3     plt.plot(train_history.history[validation])
4     plt.title('Train History')
5     plt.ylabel(train)
6     plt.xlabel('Epoch')
7     plt.legend(['train', 'validation'], loc='upper left')
8     plt.show()
9 show_train_history(train_history, 'acc', 'val_acc')
10 show_train_history(train_history, 'loss', 'val_loss')
```

