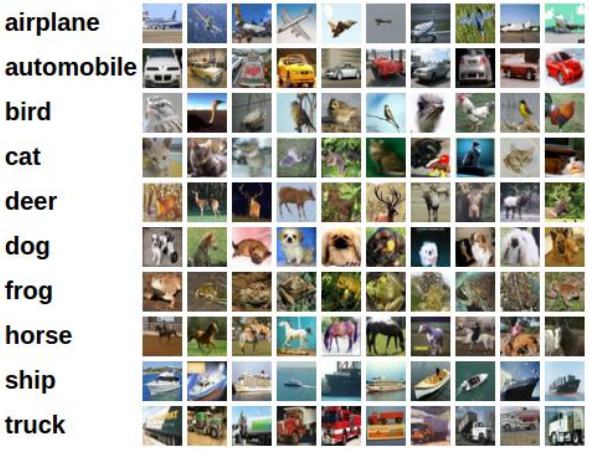
ML HW3

TAs
ml2016ta@gmail.com

Outline

- Introduction
- Data format
- kaggle
- Policy
- Rules
- FAQ

Introduction

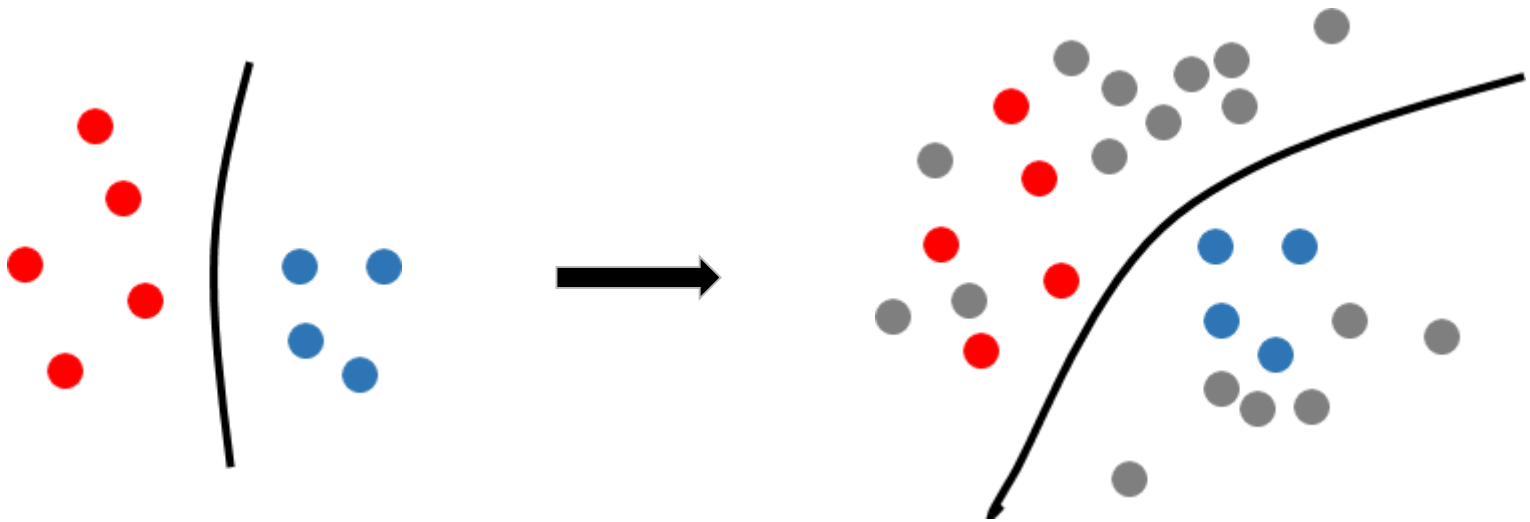


- Image classification
  The dataset comes from cifar-10.
  We will input images, then your model will predict their classes.

- A small portion of labeled data is provided within your dataset.
- How to use unlabeled data to imporve your model?

Introduction

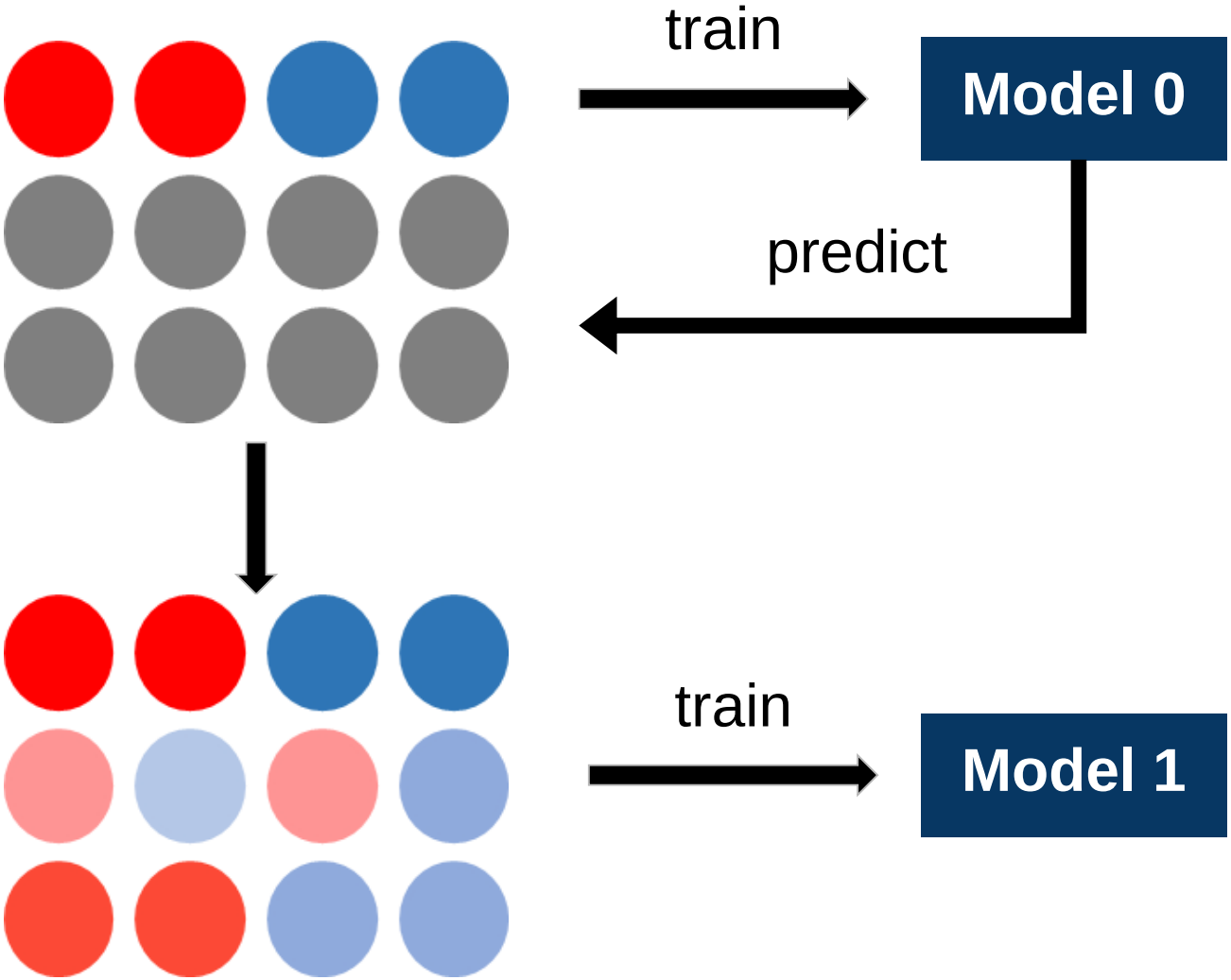
- Semi-supervised learning
  - self-training
  - cluster by autoencoder



Approach 1: Self-training

1. Data have two parts, labeled X_l and unlabeled X_u
2. Train model $f(x)$ on (X_l, Y_l)
3. Use model $f(x)$ to predict $x \in X_u$
4. Add $(x, f(x))$ to labeled data
5. Repeat 2~4

Self-training



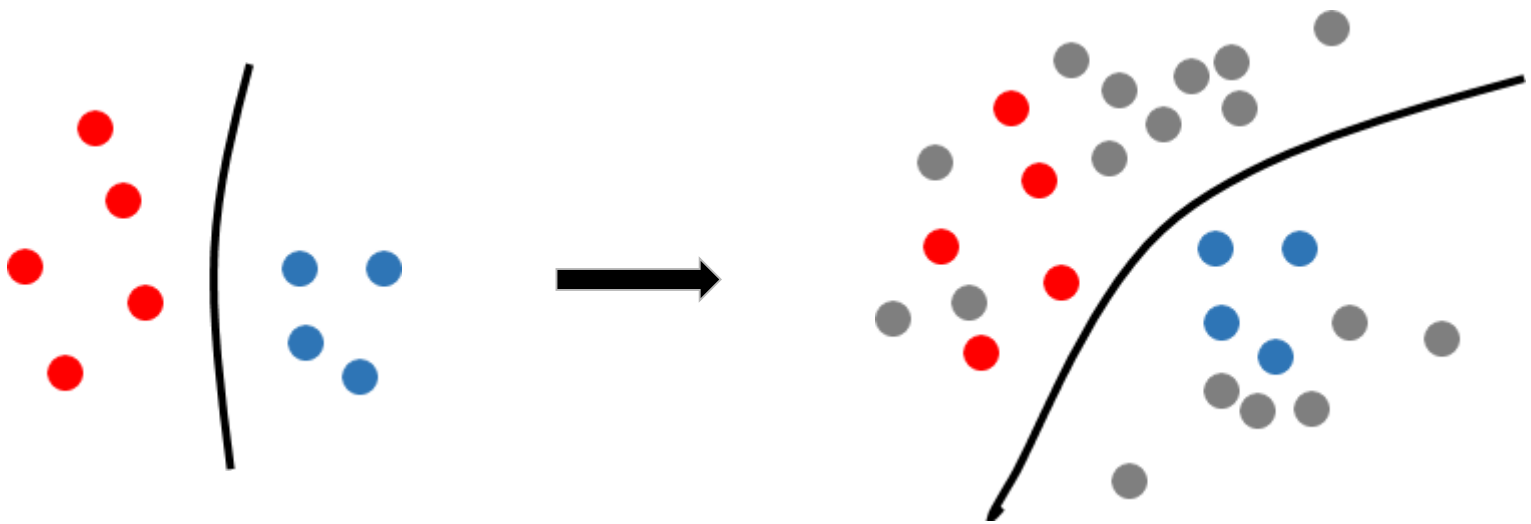
Self-training

1. Data have two parts, labeled X_l and unlabeled X_u
2. Train model $f(x)$ on (X_l, Y_l)
3. Use model to $f(x)$ predict $x \in X_u$
4. Add $(x, f(x))$ to labeled data
5. Repeat 2~4

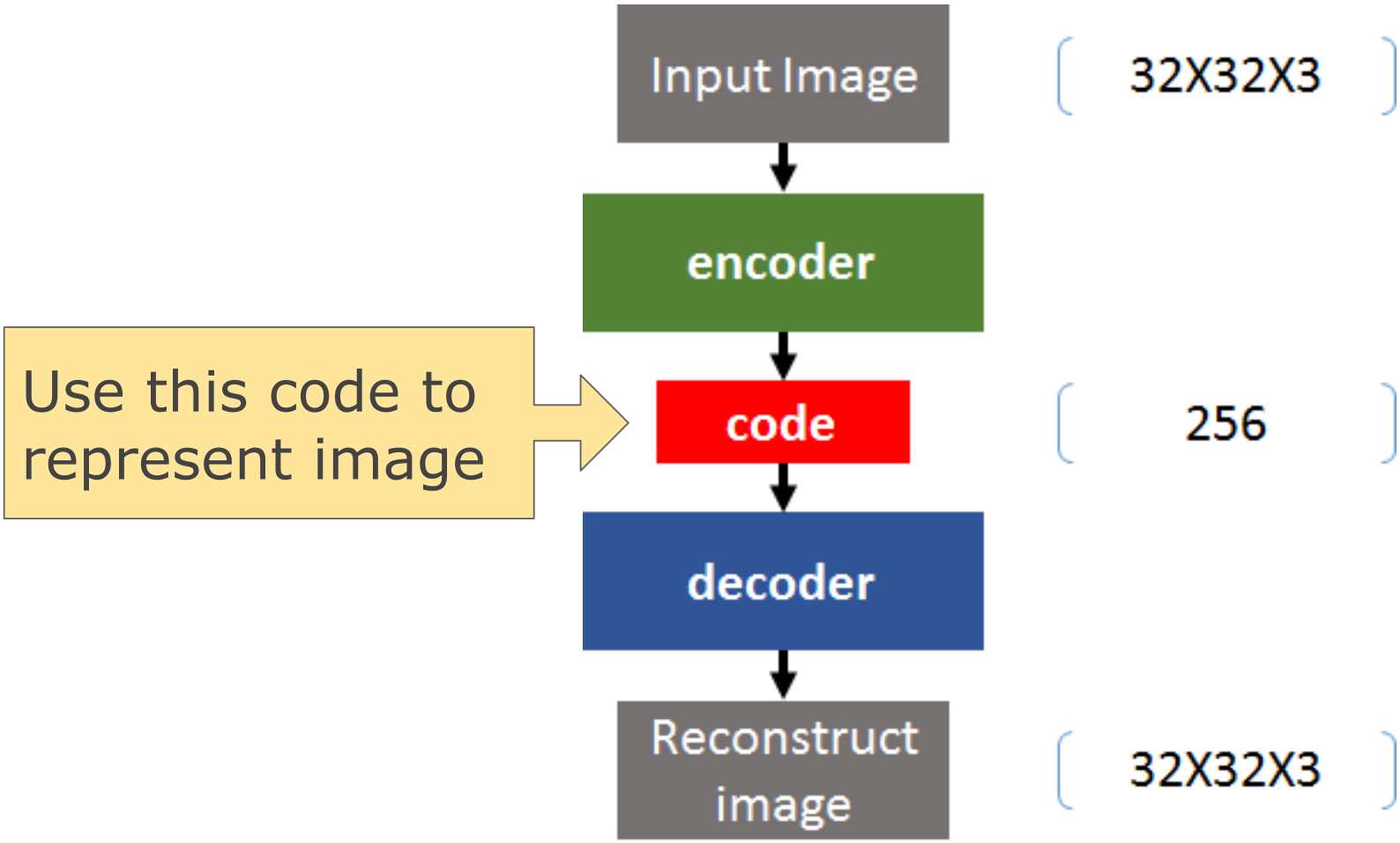
- Add a few most confident $(x, f(x))$
- Add all $(x, f(x))$
- Add all $(x, f(x))$, weighted by confident

Approach 2: clustering

- How to measure the distance of images?
- Directly use raw data (pixel intensity) to represent an image
- Extract other features, for example, extract from autoencoder



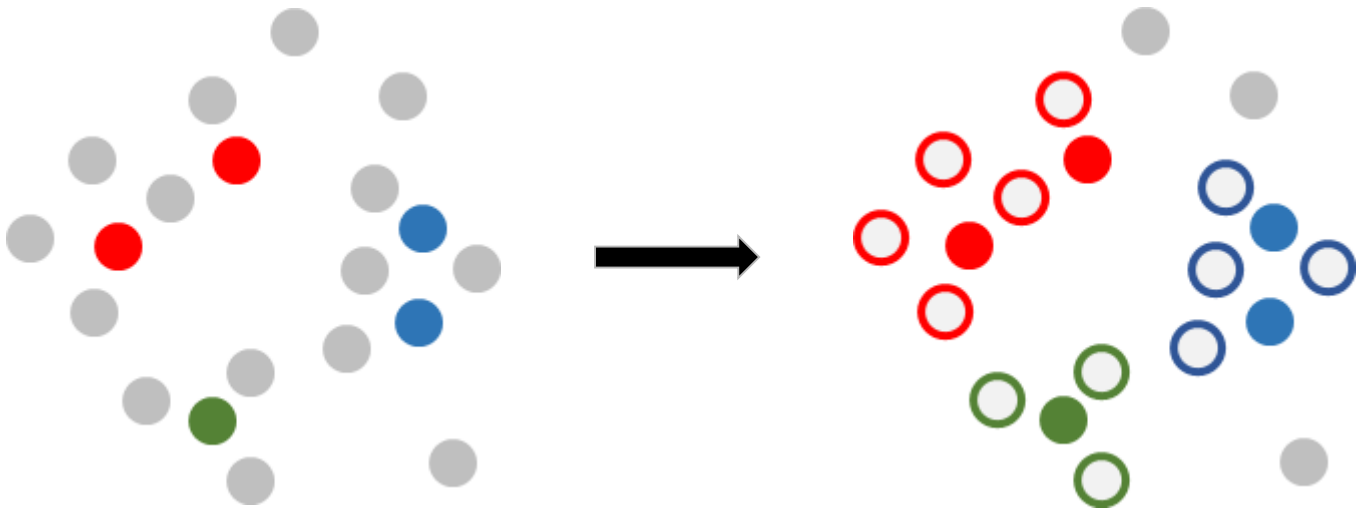
Autoencoder



https://blog.keras.io/building-autoencoders-in-keras.html

How to use autoencoder

- Directly use the code to measure the distance, for example, cosine similarity, ...
- Pretrain the weight of CNN



Data format

Three files: all_label.p, all_unlabel.p, test.p

Please use python package **pickle** to decode the files.

all_label.p contains ten classes (0-9), each class has 500 images.

Ex: all_label = pickle.load(open('all_label.p','rb'))

all_unlabel.p contains 45000 images

test.p contains 10000 images

Data format

Each image has (channels, height, width) = (3, 32, 32)

all_label[class_id (0-9)][image_id (0-499)] contains:

[3072] = [1024, 1024, 1024]

(each stands for different channel)

all_unlabel[image_id(0-44999)] = [3072]

test['ID'][i (0-9999)] = image_id

test['data'][i (0-9999)] = [3072]

Data format

Submission format: csv

第一行需為 ID,class

第二行開始為預測之 ID, 及其所屬之 class

Evaluation Function: Accuracy

```
ID,class
0,5
1,5
2,5
3,5
4,5
5,5
6,5
7,5
8,5
9,5
10,5
11,5
```

kaggle

Link: 請按此

請用 NTU 信箱登入

個人為單位

隊名：學號_任意隊名 (有修課之同學) ，旁聽請勿用學號

每日上傳 5 次為限

Public Set 5000 筆 ， Private Set 5000 筆

最後計分以 Private Set Score 為準

Kaggle Deadline: 2016/11/18 09:00:00 (GMT+8)

Github + Report Deadline: 2016/11/18 21:00:00 (GMT+8)

Github - Branch on another Method!

Branch: **master** ML2016 / **hw3** /

Create new file | Upload files | Find file | History

hoaoh add hw3 — Latest commit `744f99a` 40 seconds ago

..		
cnn.py	add hw3	40 seconds ago
test.sh	add hw3	40 seconds ago
train.sh	add hw3	40 seconds ago

Branch: **method2** ML2016 / **hw3** /

Create new file | Upload files | Find file | History

This branch is 2 commits ahead of master. — Pull request | Compare

hoaoh remove and add — Latest commit `333e1df` 16 hours ago

..		
method2.py	add method2.py, remove cnn.py	16 hours ago
test.sh	add hw3	16 hours ago
train.sh	add hw3	16 hours ago

Github -- What it should contain?

Don’t upload your own corpus!!!

Directory “ML2016/hw3” should contain at least:
Report.pdf (Master only), train.sh, test.sh, “trained_model”

Training time should be in 24 hours.
Testing time should be in 5 minutes.

Usage:
./train.sh $1 $2
$1: directory path contains (all_label.p, all_unlabel.p, test.p)
$2: output_model

./test.sh $1 $2 $3
$1: directory path contains (all_label.p, all_unlabel.p, test.p)
$2: input_model
$3: prediction.csv

Policy

1. **kaggle rank (4%)**

   top 10%: 4, top 20%: 3, top 50%: 2, beyond baseline: 1.

2. **report (4%)**

   filename: Report.pdf

   4 questions (on the next page)

   2 pages

3. **format/github error (2%)**

   If you hand in with any wrong script, you get 0.

   If you hand in with any wrong format, and you come to TAs and fix it, you get (0.5*format part score)

Report

1. **(1%) Supervised learning**:
   Use only labeled data to train a model, record its performance, and describe your method
2. **(1%) Semi-supervised learning (1)**:
   Use whole data to train a model, record its performance, and describe your method
3. **(1%) Semi-supervised learning (2)**:
   Use another method, record its performance, and describe your method
4. **(1%) Compare and analyze your results**

Rules

1. **可以**使用 package, 例如keras, tensorflow, ...
2. **不能**使用額外的資料 (包含原始的 cifar-10)
3. Only Python and C/C++
4. 任何形式的作弊、抄襲都是不被允許的

FAQ

1. 為什麼每一筆資料的維度是3*32*32

這次主題為圖片辨識，每一張圖的大小是32*32，有三原色(RGB)，因此一筆資料(一張圖)的維度為3*32*32。

2. 我可以使用其他訓練好的模型的參數來初始化模型嗎？

不行。

此外，蒐集unlabel data的label、test data的label都是不允許的。

3. 助教會跑我們的程式嗎？

會。請大家確認自己的程式是能夠執行的。

4. 程式跑很久是正常的嗎？

不同的運算資源、硬體設備會有不同的狀況，還請同學及早開始。

(助教的程式執行的總時間在5分鐘以內，同學們不用太擔心。)

FAQ

5. 助教的運算資源是什麼呢？

CPU: I7-4790K
Memory: 16GB RAM
GPU: GTX960/2GB

6. 我可以遲交嗎？

可以， Decay rate = 0.7/day
超過兩天不接受遲交
遲交表單：https://goo.gl/forms/jtiZRJZIYaaNRQav1

QAQ

Questions and Questions?

Thank You