DEEP LEARNING assignment 1

Linear Classifer

黃子軒  
 *Institute of Data Science*  
 *National Cheng Kung University*Tainan, Taiwan  
RE6111058@gs.ncku.edu.tw

# Introduction

本次作業內容圖像特徵擷取、建立分類器、比較各分類器性能等三個部分。

# Data

本次作業使用的資料為圖像資料，訓練集中的資料量為99600筆資料，共200個類別，而測試集資料為200筆，共200個類別，由於SVM在計算上會因為資料的維度膨脹導致計算時間過長，因此改以採用類別1~10的資料，因此子訓練集剩下4980筆，共10類別，測試集剩下10筆，共10類別

|  |  |  |
| --- | --- | --- |
| 資料集名稱 | 總影像數 | 包含類別數 |
| Train | 4980 | 10 |
| Test | 10 | 10 |

# feature extration(Haralick)

原始影像為彩色圖像，先使用opencv內建函數將圖像轉為黑白影像之後，再利用套件mahotas將圖片轉為13個haralick特徵，計算步驟與其13個特徵定義如下:

首先須先計算灰階共生矩陣(grey-level co-occurrence matrix,GLCM)，計算方法如下:

假設有一個擁有四個不同level 0~3(Ng=4)的灰階矩陣

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 1 |
| 1 | 3 | 3 |
| 2 | 1 | 2 |

則按照四個方向(水平、垂直，左斜、右斜)計算兩灰階數值發生之頻率，以水平為例，由於第一行有1-2及2-1相鄰一次、第二行有1-3及3-3相鄰一次、第三行有1-2及2-1相鄰一次，計算頻率後可得下列矩陣:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 2 | 1 |
| 2 | 0 | 2 | 0 | 0 |
| 3 | 0 | 0 | 0 | 1 |

令該矩陣為P，定義其他公式如下:

最後定義13個haralick texture feature

## Angular Second Moment(f1)

f1 =

## Contrast(f2)

## Correlation(f3)

*Where 、、 are the means and standard deviations of*

## Variance(f4)

## Inverse Difference Moment(f5)

## Sum Average(f6)

## Sum Variance(f7)

## Sum Entropy(f8)

## Entropy(f9)

## Difference Variance(f10)

## Difference Entropy(f11)

## Information Measures of Correlation(f12)

## Information Measures of Correlation(f13)

由於有四個方向的灰階共生矩陣，因此對應會產生4\*13個haralick feature，為了減低資料的維度，因此將不同方向的featrue以平均的方式進行合併，最後得出13個haralick feature並於之後的分類器使用。

# feature extration(HOG)

原始影像為彩色圖像，先使用opencv內建函數將圖像轉為黑白影像之後，再利用套件skimage將圖片轉為HOG特徵，而經由調整參數後，一共產生1764個特徵，將每筆資料的特徵，其計算方法如下:

特徵維度=影像區塊數量×區塊內細胞數量×每個細胞的直方圖大小

計算結果為:

(2\*2)\*9\*(7\*7)=1764

因次1764個特徵就這樣來的。

# feature extration(LBP)

原始影像為彩色圖像，先使用opencv內建函數將圖像轉為黑白影像之後，再利用套件skimage將圖片轉為LBP特徵，而經由調整參數後，一共產生24個特徵，將每筆資料的特徵，其計算方法如下:

特徵維度=領域的半徑×領域中的點的數量

計算結果為:

3\*8=24

因次24個特徵就這樣來的。

# classification

## 使用KNN、SVM、randomforest、CatBoost等三種機器學習方法來建立分類器，並比較三者的差異，各分類器簡介如下:

## KNN K-nearest neighbors

使用sklearn的KNeighborsClassifier，超參數的部分，

n\_neighbors=5，其餘皆為預設參數值

## SVM Support Vector Mechine

使用sklearn的函式SVC，超參數全部使用預設參數值

## randomforest

## 使用sklearn的函式RandomForestClassifier，超參數按照預設參數值設定

## CatBoost

使用catboost套件的函式CatBoostClassifier，超參數設置如下

loss\_function='MultiClass'

learning\_rate=0.01

iterations=1000

其餘參數使用預設參數

# Experiment

|  |  |  |
| --- | --- | --- |
| Model | Accuracy | F1-score |
| SVM | 0.1 | 0.1 |
| KNN | 0.1 | 0.04 |
| Random forest | 0.2 | 0.17 |
| CatBoost | 0.1 | 0.1 |

可以看到，將所有特徵萃取後放入同一data frame後，

我們的模型皆不是太好，首先是因為測試集的關係，因為每筆資料只有一個測資，所以她的表現很大程度會因為測資的獨特而預測錯誤，再來就是類別數量，即便已經降低維度，只取10個類別，但我們的模型然會因為高維度的資料而導致太多特徵有共線性，如果不是使用神經網路的話，傳統機器學習模型在面對這種資料，都不是表現得很好。

回頭來看表格，在以上的模型中，隨機森林的表現是最好的，這也符合我的預期，因為她的訓練過程會使用子資料以及子特徵，在高共線性的資料下算是可以使用的模型之一，因此他的表現也是裡面最好的。

# GitHub Link

https://github.com/H24061024/DL\_HW\_1