# D14：CNN分類器架構：Batch Normalization

簡報閱讀 範例與作業 問題討論


## 深度學習理論與實作

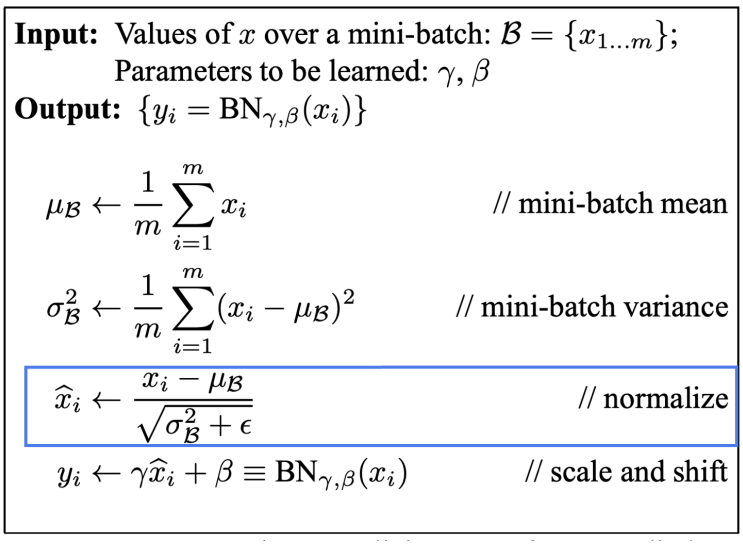
●理解 Batch Normalization 原理
●Batch Normalization 用來解決什麼問題

## BN ( Batch Normalization )

Batch Normalization 是 2015 年 Google 研究員在論文《Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift》一文中提出的，並同時將 BN 應用於 Inception-v2 的框架中。

## BN算法

首先計算輸入 Batch 的平均值與標準差



**Input:** Values of $x$ over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: $\gamma$, $\beta$
**Output:** $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

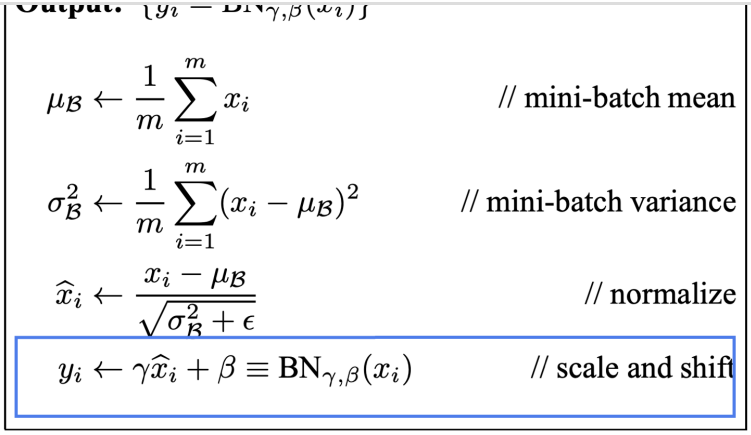
$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

透過學習 Gamma 與 Beta 做縮放與平移， Gamma 與 Beta 為 BN 層內唯二需要學習的參數



$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma \widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

## 資料分佈

一般來說我們都是以 **Mini Batch的方式訓練資料**，然而每一個 Batch 間的**資料分佈**可能不太相同，而輸入每一層神經元的資訊分布也都可能會改變，造成收斂上的困難。

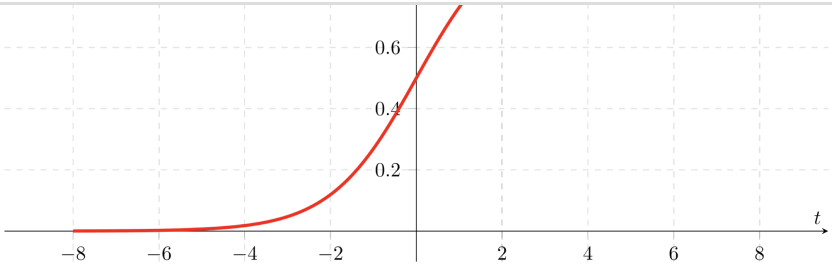
透過 BN，將每一層輸入資料的分佈歸一化為**平均值為0，方差為1**，確保資料分佈的穩定性。

然而 Normalize 改變資料的分佈，可能會造成**上一層學到的特徵消失**，因此BN 的最後一步透過學習 Beta、Gamma，去微調 Normalize 後資料的分佈 。

參考資料：

> Understanding the backward
>
> At the moment there is a wonderful course running at Standford University,
>
> kratzert.github.io

## 梯度消失




參考來源：Derivative of the Sigmoid function

Sigmoid會將數值較大與較小的值通通壓在一起，並且由於其導函數最大值為0.25，容易發生**梯度消失**的情形，透過BN，我們將資料分布歸一化，能有效降低梯度消失的可能性。

## 推薦延伸閱讀

### Covariate Shift 解釋

A Simple Tutorial on Covariance Shift!!!!! |

Explore and run machine learning code with Kaggle Notebooks | Using data from multiple data sources

www.kaggle.com

Tutorial - Covariate Shift -Sberbank Housing EDA

Introduction

You may have heard from various people that data science competitions are a good way to learn data science, but they are not as useful in solving real world data science problems. Why do you think this is the case?

One of the differences lies in the quality of data that has been provided. In Data Science Competitions, the datasets are carefully curated. Usually, a single large dataset is split into train and test file. So, most of the times the train and test have been generated from the same distribution.

But this is not the case when dealing with real world problems, especially when the data has been collected over a long period of time. In such cases, there may be multiple variables / environment changes might have happened during that period. If proper care is not taken then, the training dataset cannot be used to predict anything about the test dataset in a usable manner.

### Batch Normalization 的用途

深度學習中Batch

用mxnet 做了實驗，用不用bn簡直

個世界，請問大俠們，為什麼BN這

www.zhihu.com

度）的均值為0，方差為1.而最後的“ scale and shift”操作則是為了讓因訓練所需而“刻意”加入的BN能夠有可能還原最初的輸入（即當 $\gamma^{(k)} = \sqrt{Var[x^{(k)}]}, \beta^{(k)} = E[x^{(k)}]$ ），從而保證整個network的capacity。（有關capacity的解釋：實際上BN可以看作是在原模型上加入的“新操作”，這個新操作很大可能會改變某層原來的輸入。當然也可能不改變，不改變的時候就是“還原原來輸入”。如此一來，既可以改變同時也可以保持原輸入，那麼模型的容納能力（capacity）就提升了。）

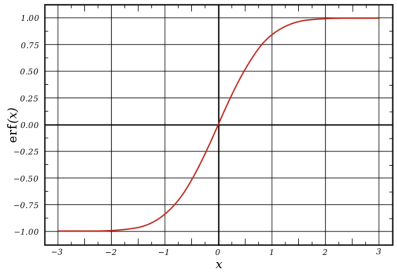
## Sigmoid梯度消失原因與解決

**梯度消失 Gradient Vanish**

「類似」於 Sigmoid function 的激勵函數，普遍帶有梯度消失（ Gradient Vanish ）的隱憂，那究竟什麼是梯度消失?

$$Sigmoid\ function = \theta(s) = \frac{1}{1+e^{-s}}$$

此函數圖形為






## 解題時間


下一步：閱讀範例與完成作業