Deep Learning 1

Deep Learning & Gradient Descent

$$f : X \longmapsto Y$$

$$\updownarrow \mathcal{L}_\theta \quad \mathcal{L} \text{ 與 } \hat{f} \text{ 必須可微}$$

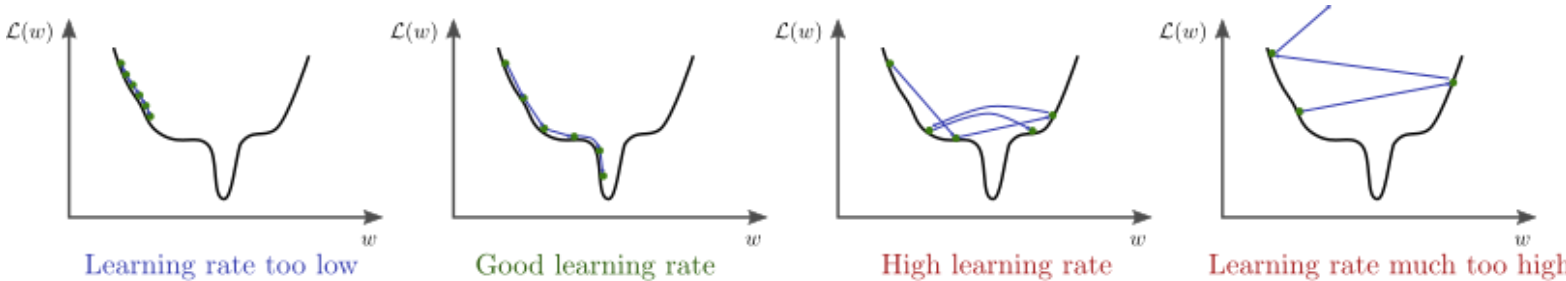
$$\hat{f}_\theta : X \longmapsto \hat{Y}$$

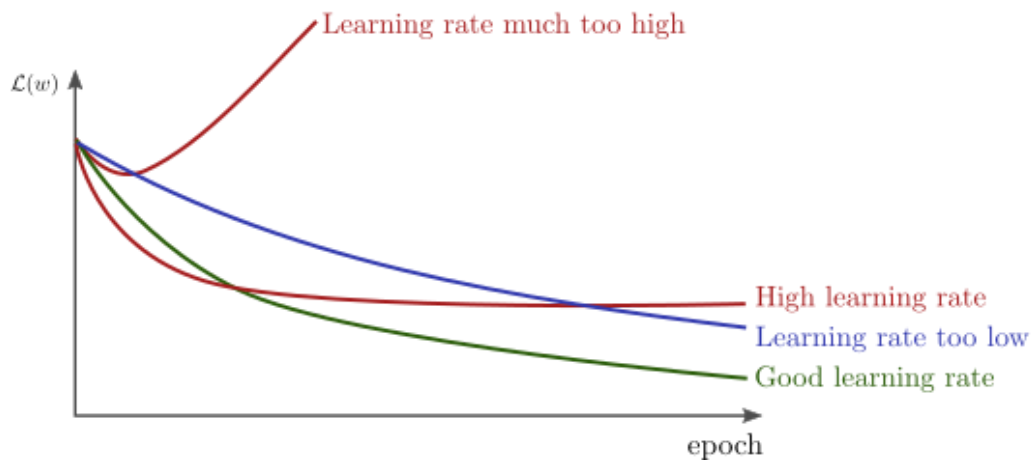
$$\mathcal{L}_\theta(x, y) = ||\hat{f}_\theta(x) - y||$$

$$\theta \leftarrow \theta - \gamma \nabla_\theta \mathcal{L}_\theta(x, y)$$

- $\hat{f}, \theta$: Model & Parameters
- $\mathcal{L}$: Loss Function
- γ: Learning Rate

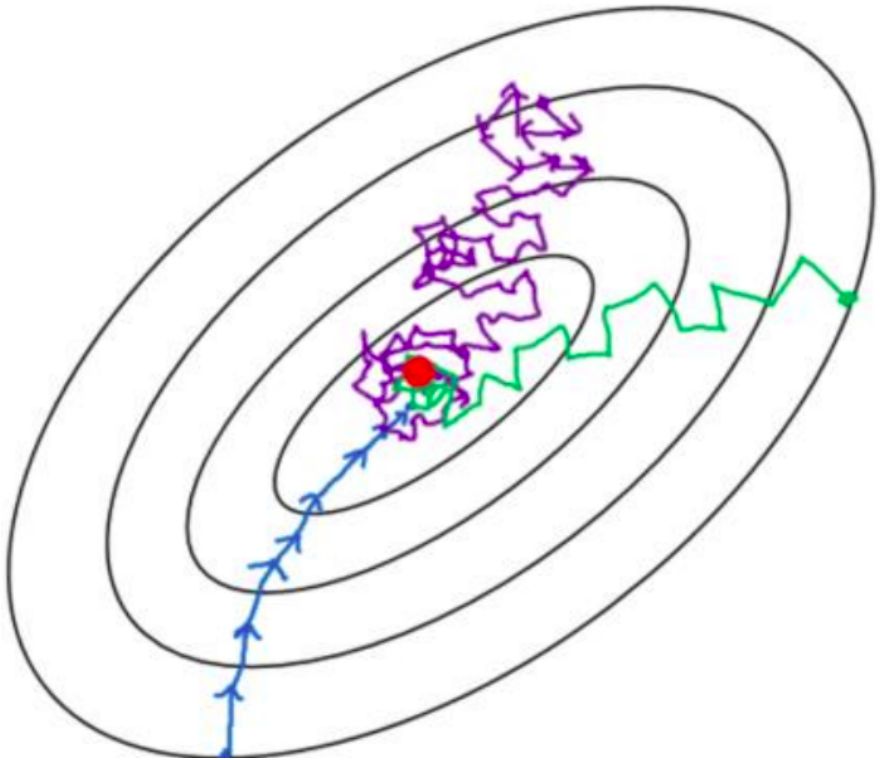
Learning Rate





cite: Stanford cs231

Mini Batch Gradient Descent



Batch gradient decent
Mini-batch GD
Online gradient decent

Batch Size

- 過小容易震盪難以收斂
- 過大需要耗費更多計算資源

cite: Mini-batch performs poorly than Batch gradient descent?

Activation Function

Linear & Nonlinear

$$\hat{f}_\theta(x) = (h_n \circ \cdots \circ h_2 \circ h_1)(x)$$
$$h_i(x) = \alpha(xW_i + b_i)$$
$$\theta = [W_1, b_1, \ldots, W_n, b_n]$$

α is the Nonlinear Activation Function, e.g.,

- $ReLU(x) = max(x, 0)$
- $\sigma(x) = \dfrac{1}{1 + e^{-x}}$

Activation Function

Linear & Nonlinear

$$
\begin{aligned}
\hat{f}_\theta(x) &= (h_n \circ \cdots \circ h_2 \circ h_1)(x) \\
&= xW_1W_2\cdots W_n \\
&\quad b_1W_2\cdots W_n + \cdots + b_n \\
&= xW + b
\end{aligned}
$$

當沒有使用激活函數時，無數線性層相疊等同個單線性層
$\rightarrow$無法解決非線性問題。

Gradient Issues

Vanishing and Exploding

$$\mathcal{L}_\theta = \mathcal{L}((h_n \circ \cdots \circ h_2 \circ h_1)(x), y)$$

$$\nabla_{W_1}\mathcal{L} = \nabla_{h_n}\mathcal{L}\nabla_{h_{n-1}}h_n \cdots \nabla_{h_1}h_2\nabla_{W_1}h_1$$

Gradient Vanishing

當 $\left|\nabla_{h_{i-1}}h_i\right| < 1$ 時，$\nabla_{W_1}\mathcal{L}$ 會依據層數指數縮小，梯度過小導致參數更新緩慢而難以訓練。

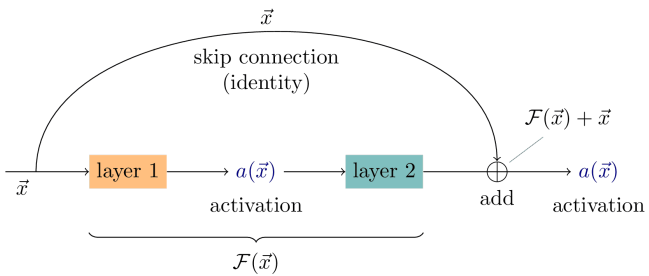
Gradient Exploding

當 $\left|\nabla_{h_{i-1}}h_i\right| > 1$ 時，$\nabla_{W_1}\mathcal{L}$ 會依據層數指數上升，梯度過大使參數無法收斂。

The Solution to the

Gradient Vanishing

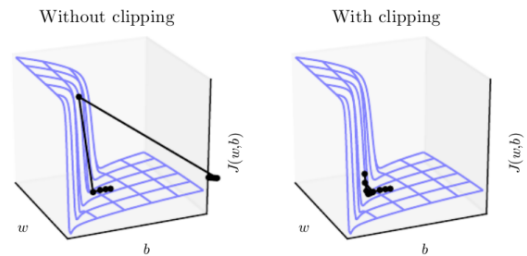
- Skip Connection



cite: Skip Connection

Gradient Exploding

- Gradient Clipping



cite: Understanding Gradient Clipping

- Normalization

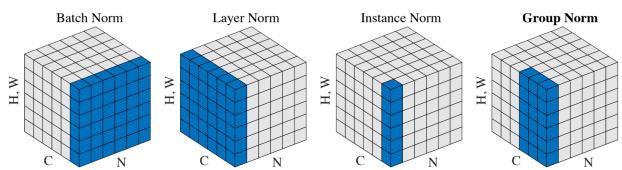
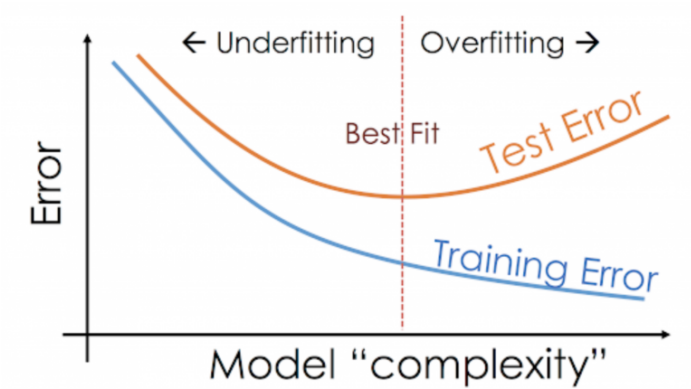


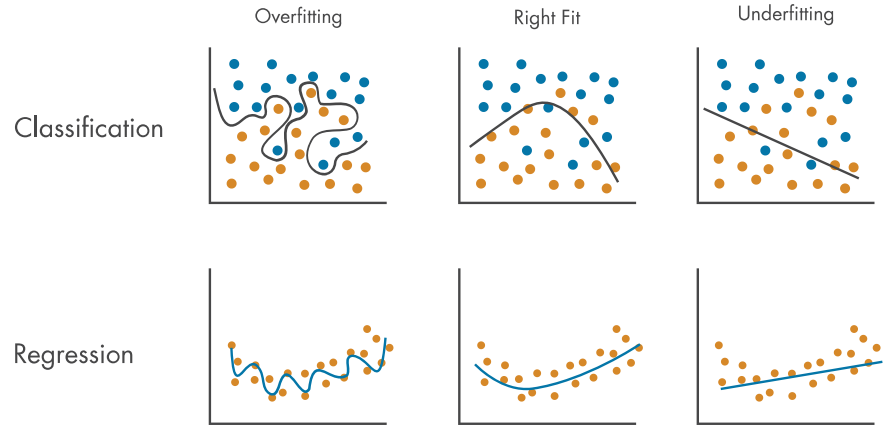
Figure 2. **Normalization methods**. Each subplot shows a feature map tensor, with N as the batch axis, C as the channel axis, and (H, W) as the spatial axes. The pixels in blue are normalized by the same mean and variance, computed by aggregating the values of these pixels.

cite: Group Normalization

Overfitting, Generalization and Robustness



cite: Overfitting And Underfitting in Machine Learning



cite: What Is Overfitting?

- 當模型複雜度超過訓練資料集複雜度時，就容易發生 Overfitting
- 當訓練資料集複雜度超過模型複雜度時，就容易發生 Underfitting

Overfitting, Generalization and Robustness

避免 Overfitting 的方法

- 使用更多訓練資料
- Dropout
- L1/L2 Regularization
- 降低模型複雜度

避免 Underfitting 的方法

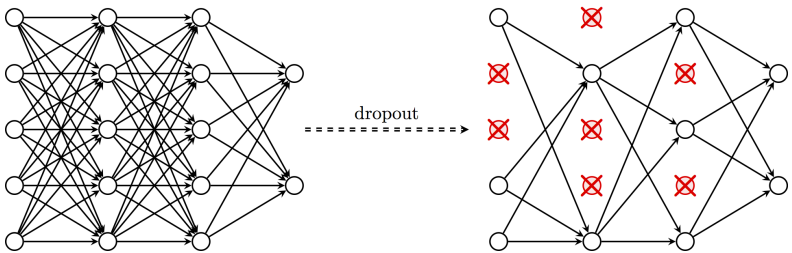
- 大力出奇蹟

  疊更多參數能解決一切
- 錢是萬能的

  用更強的硬體訓練更大的模型

Overfitting, Generalization and Robustness

Dropout



cite: Tikz/Dropout

- 不會過度依賴某些特徵
- 加入雜訊增強穩健性

L1/L2 Regularization

- $\mathcal{L}_\theta + \underbrace{\lambda||\theta||}_{\text{L1}}$ *or* $\underbrace{\lambda||\theta||_2^2}_{\text{L2}}$
- L1: 將大多數貢獻不大的參數歸零，使模型變稀疏
- L2: 使模型不會過度依賴部份參數
- http://playground.tensorflow.org/

Overfitting, Generalization and Robustness

Generalization

- 模型應用到其他資料分佈的能力

> The classic approach towards the assessment of any machine learning model revolves around the evaluation of its generalizability i.e. its performance on unseen test scenarios.

Robustness

- 模型對抗干擾的能力

> Evaluating such models on an available non-overlapping test set is popular, yet significantly limited in its ability to explore the model's resilience to outliers and noisy data / labels (i.e. robustness).

cite: Generalizability vs. Robustness: Adversarial Examples for Medical Imaging , Robustness vs Generalization

Model Architecture

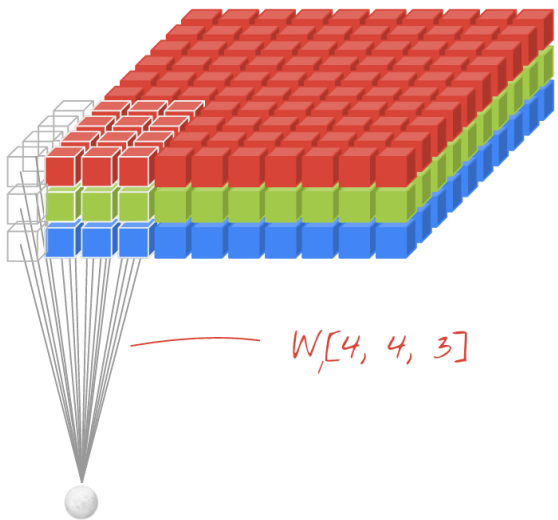
靈魂拷問：「你為什這樣設計？有什麼理論根據嗎？」

$$\left\{\begin{array}{c} CNN \\ RNN \\ Transformer \end{array}\right\} + \left\{\begin{array}{c} Encoder \\ Decoder \end{array}\right\} + \left\{\begin{array}{c} Causal \\ Non\text{-}causal \end{array}\right\}$$

	Receptive Field	Memory Usage	Inductive Bias	Parallelization
CNN	受模型架構限制	$O(L)$	Y	Y
RNN	會遺失太遙遠的資訊	$O(L)$	Y	N
TNN	與模型架構無關	$O(L^2)$	N	Y

Model Architecture

Convolutional Neural Networks, CNN



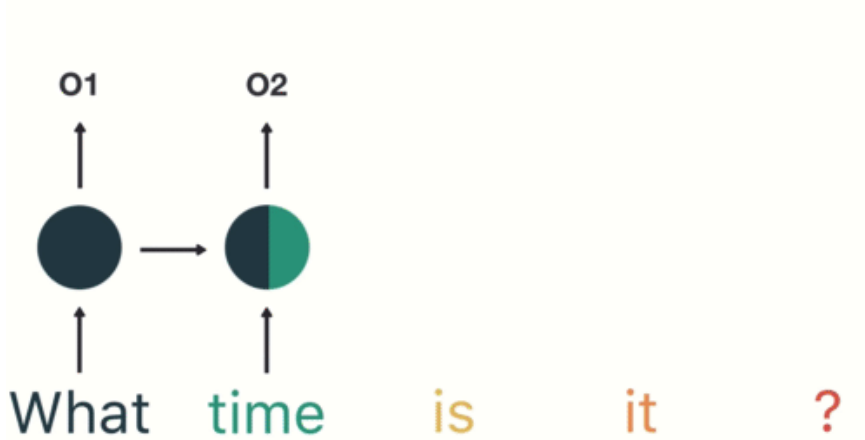
cite: Convolutional Neural Networks am Beispiel eines selbstfahrenden Roboters 0.1 Dokumentation

- Inductive Bias
  - 具備平移不變性
  - 不同位置共享參數
  - 無須龐大資料就能有相對穩定的效能
- 可高度平行化
- Receptive Field 受架構設計所限
- https://poloclub.github.io/cnn-explainer/

Model Architecture

Recurrent Neural Networks, RNN

- Inductive Bias
  - 不同位置共享參數
  - 無須龐大資料就能有相對穩定的效能
- 輸出與過去計算結果相關
  - 訓練時難以有效利用平行化加速



cite: In-Depth Guide to Recurrent Neural Networks (RNNs) in 2023

Model Architecture

Multihead Attention, Transformer

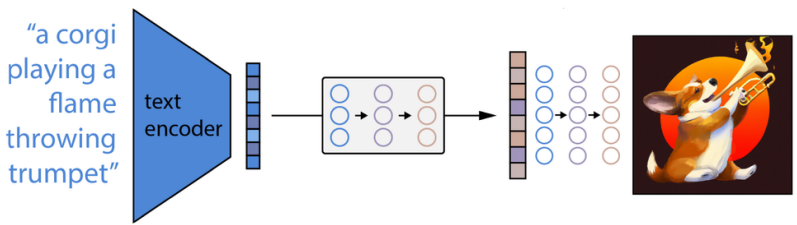
- w/o Inductive Bias
  - 不同位置共享參數
  - Receptive Field 依據訓練時給定的上限決定

    about Length-Extrapolatable
  - 需要額外加入位置資訊
  - 需要依靠龐大資料才能達成良好的性能
- 可高度平行化

cite: Transformer: A Novel Neural Network Architecture for Language Understanding

Model Architecture

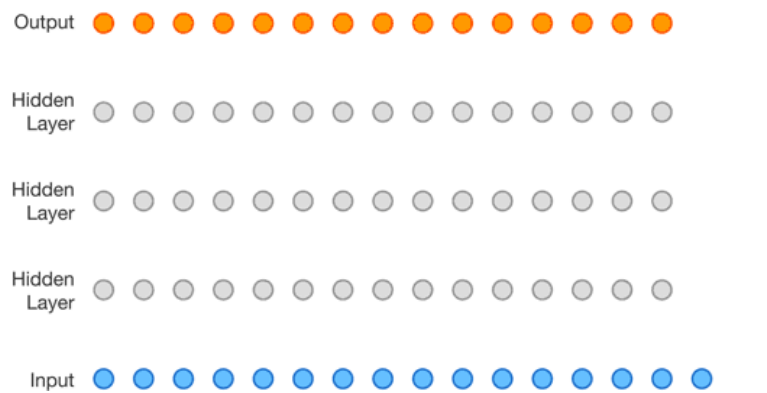
$$Data \underset{Decoder}{\overset{Encoder}{\rightleftharpoons}} Feature$$



cite: How DALL-E 2 Actually Works

Causal $p(y_t|x_{\leq t})$,

Autoregressive $p(y_t|y_{<t})$



cite: WaveNet: A generative model for raw audio

Garbage In, Garbage Out

cite: 盡信資料，不如無資料

- 資料稀缺

  使用預訓練（Pretrain）好的模型微調（fine-tune）
- 缺乏標記

  使用 Self/Semi Supervised Learning
- 標記錯誤 Awesome-Noisy-Labels
- 分佈不平衡

  Classification on imbalanced data

Deep Learning & 調(煉)參(丹)

$$f : X \longmapsto Y$$

$$\mathcal{L}_\theta$$

$$\hat{f}_\theta : X \longmapsto \hat{Y}$$

Training Trick

Data

Learning Algorithm

Model

Loss Function