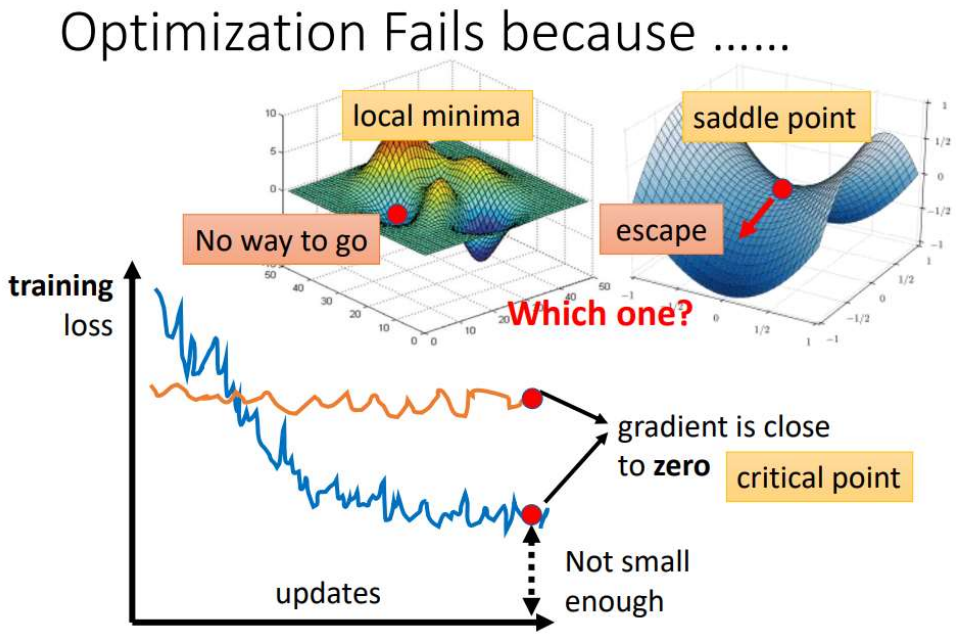
類神經網路訓練不起來怎麼辦 (一)： 局部最小值 (local minima) 與鞍點 (saddle point)

Create at 2022/06/08

- 上課資源：
  1. 類神經網路訓練不起來怎麼辦 (一)： 局部最小值 (local minima) 與鞍點 (saddle point) (https://www.youtube.com/watch?v=QW6uINn7uGk)

Optimization 時怎麼把 gradient descent 做得更好

為甚麼 optimization 會失敗呢？



- 隨著參數不斷 update，training loss 不會再下降
- 但是對於這個 loss 仍然不滿意
- 有時候會發現 model train 不起來，不管怎麼 update 參數 loss 都掉不下去
  - 猜想：走到一個點，這個點對 loss 的微分為 0，當對 loss 的微分為 0 時，gradient descent 就沒辦法再 update 參數，此時 training 就停下來了，參數不再 update，loss 就不會再下降了
- 不是只有 local minima 的 gradient 為 0
- saddle point 的 gradient 是 0，但不是 local minima 也不是 local maxima
- gradient 為 0 的點，統稱為 critical point
- 可以說 loss 沒辦法再下降，可能是卡在 critical point

Warning of Math

可以跳過沒關係

Tayler Series Approximation

$L(\boldsymbol{\theta})$ around $\boldsymbol{\theta} = \boldsymbol{\theta}'$ can be approximated below

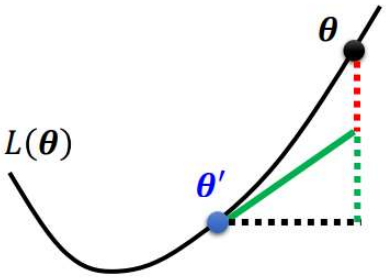
$$L(\boldsymbol{\theta}) \approx L(\boldsymbol{\theta}') + (\boldsymbol{\theta} - \boldsymbol{\theta}')^T \boldsymbol{g} + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}')^T H (\boldsymbol{\theta} - \boldsymbol{\theta}')$$

Gradient $\boldsymbol{g}$ is a *vector*

$$\boldsymbol{g} = \nabla L(\boldsymbol{\theta}') \qquad \boldsymbol{g}_i = \frac{\partial L(\boldsymbol{\theta}')}{\partial \boldsymbol{\theta}_i}$$

Hessian H is a *matrix*

$$H_{ij} = \frac{\partial^2}{\partial \boldsymbol{\theta}_i \partial \boldsymbol{\theta}_j} L(\boldsymbol{\theta}')$$



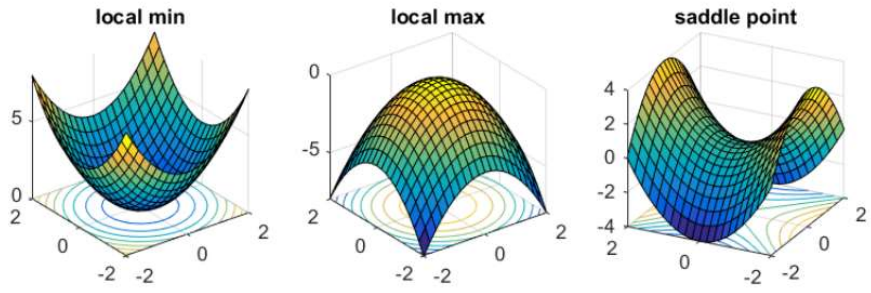
Hessian

$L(\boldsymbol{\theta})$ around $\boldsymbol{\theta} = \boldsymbol{\theta}'$ can be approximated below

$$L(\boldsymbol{\theta}) \approx L(\boldsymbol{\theta}') + \cancel{(\boldsymbol{\theta} - \boldsymbol{\theta}')^T \boldsymbol{g}} + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}')^T H (\boldsymbol{\theta} - \boldsymbol{\theta}')$$

At critical point

telling the properties of critical points



Hessian

At critical point:

$$L(\boldsymbol{\theta}) \approx L(\boldsymbol{\theta}') + \frac{1}{2}(\boldsymbol{\theta}-\boldsymbol{\theta}')^T H(\boldsymbol{\theta}-\boldsymbol{\theta}') \qquad \boldsymbol{v}^T H \boldsymbol{v}$$

For all $\boldsymbol{v}$

$\boldsymbol{v}^T H \boldsymbol{v} > 0$ ➡ Around $\boldsymbol{\theta}'$: $L(\boldsymbol{\theta}) > L(\boldsymbol{\theta}')$ ➡ **Local minima**

= H is positive definite = All eigen values are positive. ⬆

For all $\boldsymbol{v}$

$\boldsymbol{v}^T H \boldsymbol{v} < 0$ ➡ Around $\boldsymbol{\theta}'$: $L(\boldsymbol{\theta}) < L(\boldsymbol{\theta}')$ ➡ **Local maxima**

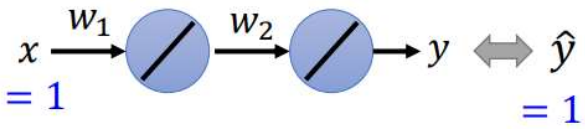
= H is negative definite = All eigen values are negative. ⬆

Sometimes $\boldsymbol{v}^T H \boldsymbol{v} > 0$, sometimes $\boldsymbol{v}^T H \boldsymbol{v} < 0$ ➡ **Saddle point**

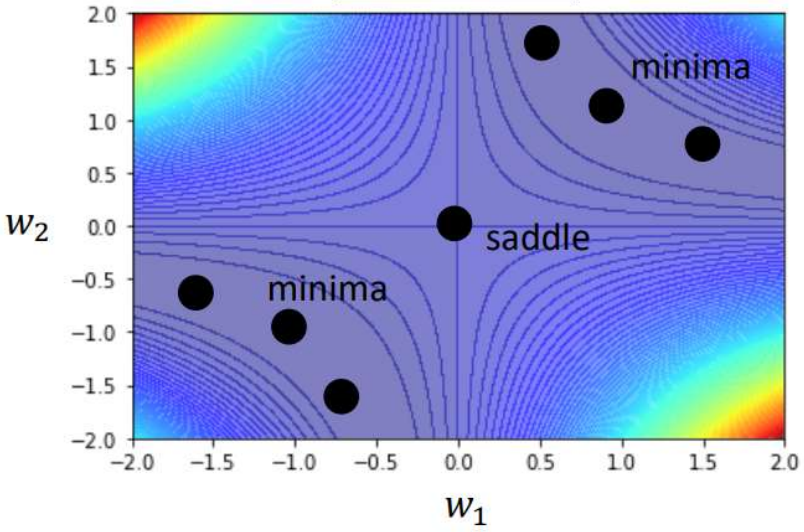
Some eigen values are positive, and some are negative. ⬆

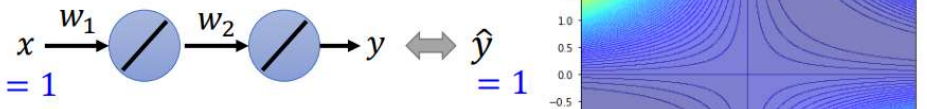
Example

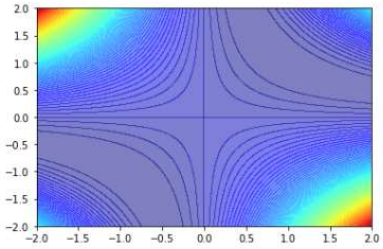
$y = w_1 w_2 x$



Error Surface







$$L = (\hat{y} - w_1 w_2 x)^2 = (1 - w_1 w_2)^2$$

$$g = \begin{bmatrix} \frac{\partial L}{\partial w_1} \\ \frac{\partial L}{\partial w_2} \end{bmatrix} = \begin{bmatrix} 2(1 - w_1 w_2)(-w_2) \\ 2(1 - w_1 w_2)(-w_1) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Critical point: $w_1 = 0, w_2 = 0$

$$H = \begin{bmatrix} 0 & -2 \\ -2 & 0 \end{bmatrix} \quad \lambda_1 = 2, \lambda_2 = -2$$

Saddle point

$$H = \begin{bmatrix} \frac{\partial^2 L}{\partial w_1^2} & \frac{\partial^2 L}{\partial w_1 \partial w_2} \\ \frac{\partial^2 L}{\partial w_2 \partial w_1} & \frac{\partial^2 L}{\partial w_2^2} \end{bmatrix} = \begin{bmatrix} 2(-w_2)(-w_2) & -2 + 4w_1 w_2 \\ -2 + 4w_1 w_2 & 2(-w_1)(-w_1) \end{bmatrix} = \begin{bmatrix} 0 & -2 \\ -2 & 0 \end{bmatrix}$$

Don't afraid of saddle point?

At critical point: $L(\boldsymbol{\theta}) \approx L(\boldsymbol{\theta}') + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}')^T H (\boldsymbol{\theta} - \boldsymbol{\theta}')$, where $\boldsymbol{v}^T H \boldsymbol{v}$ is the quadratic term

Sometimes $\boldsymbol{v}^T H \boldsymbol{v} > 0$, sometimes $\boldsymbol{v}^T H \boldsymbol{v} < 0$ ➡ Saddle point

H may tell us parameter update direction!

$\boldsymbol{u}$ is an eigen vector of H
λ is the eigen value of $\boldsymbol{u}$
$\lambda < 0$

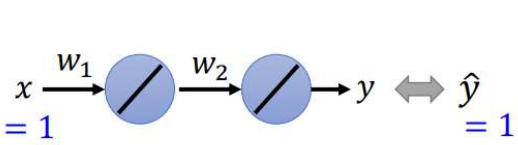
$$\boldsymbol{u}^T H \boldsymbol{u} = \boldsymbol{u}^T (\lambda \boldsymbol{u}) = \lambda \|\boldsymbol{u}\|^2$$
$$< 0 \qquad < 0$$

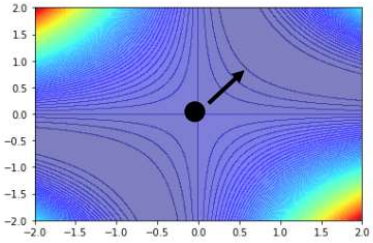
$$L(\boldsymbol{\theta}) \approx L(\boldsymbol{\theta}') + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}')^T H (\boldsymbol{\theta} - \boldsymbol{\theta}') \Rightarrow L(\boldsymbol{\theta}) < L(\boldsymbol{\theta}')$$

$\boldsymbol{u}$ $\boldsymbol{u}$

$$\boldsymbol{\theta} - \boldsymbol{\theta}' = \boldsymbol{u} \qquad \boldsymbol{\theta} = \boldsymbol{\theta}' + \boldsymbol{u}$$

Decrease L





$$L = (\hat{y} - w_1 w_2 x)^2 = (1 - w_1 w_2)^2$$

$$\frac{\partial L}{\partial w_1} = 2(1 - w_1 w_2)(-w_2)$$

$$\frac{\partial L}{\partial w_2} = 2(1 - w_1 w_2)(-w_1)$$

Critical point: $w_1 = 0, w_2 = 0$

$$H = \begin{bmatrix} 0 & -2 \\ -2 & 0 \end{bmatrix} \quad \lambda_1 = 2, \lambda_2 = -2$$

Saddle point

$\lambda_2 = -2$ Has eigenvector $\boldsymbol{u} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

Update the parameter along the direction of $\boldsymbol{u}$

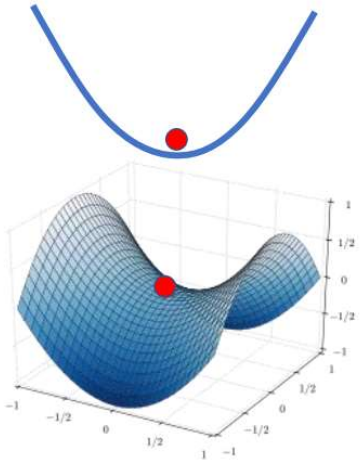
You can escape the saddle point and decrease the loss.

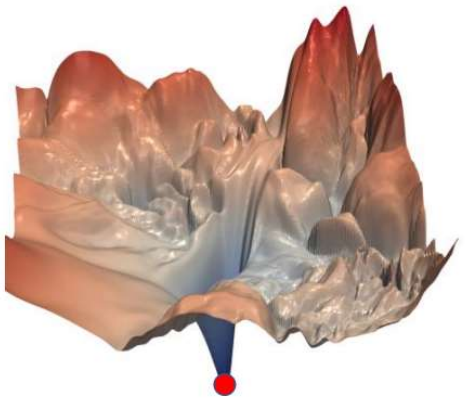
(this method is seldom used in practice)

End of Warning

Saddle Point 跟 Local Minima 誰比較常見？

Saddle Point v.s. Local Minima



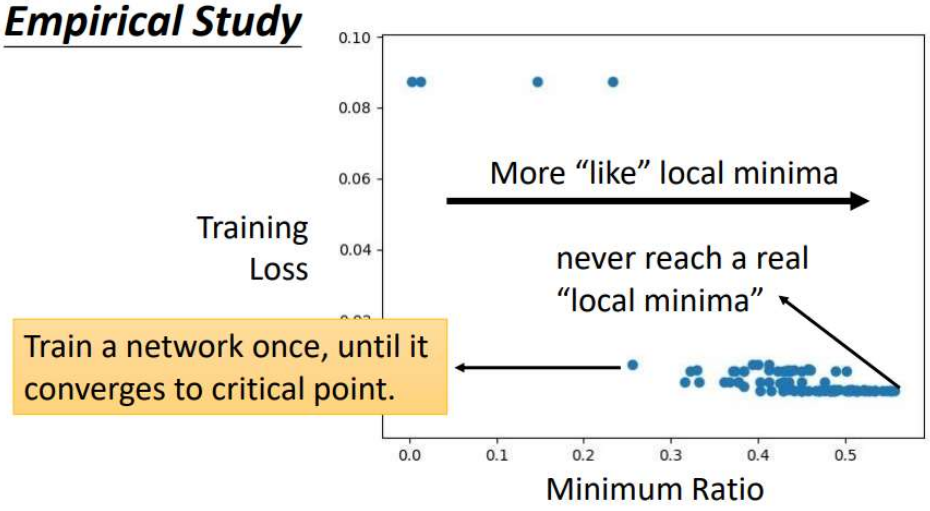


Saddle point in higher dimension?

When you have lots of parameters, perhaps local minima is rare?

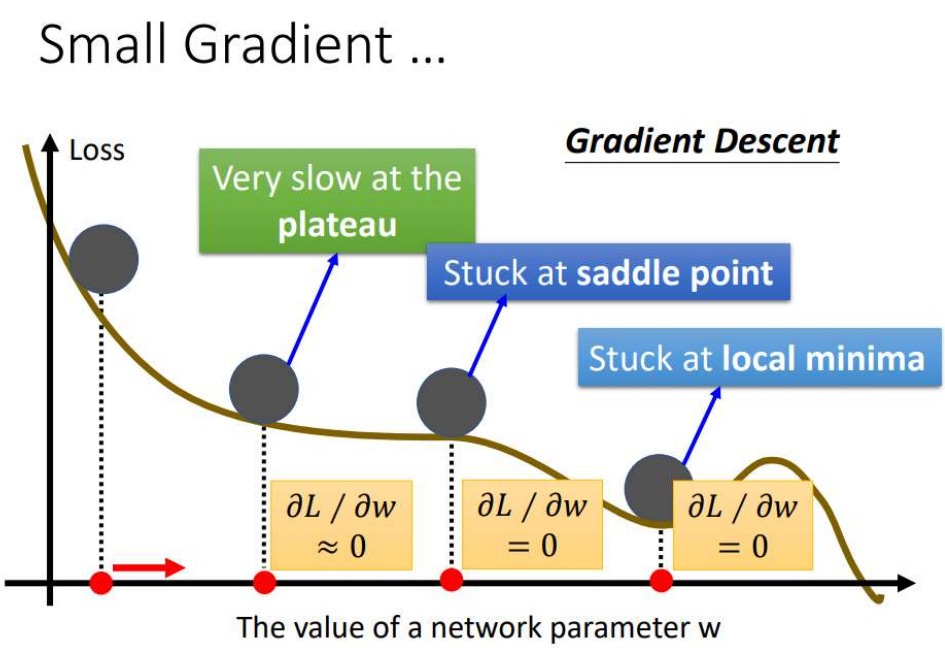
- 維度越高，可能可以走的路越多
- 所以在訓練 network 時，參數往往會很多，所以 error surface 其實是在一個非常高的維度中
- 參數有多少代表 error surface 的維度有多少

Empirical Study



$$\text{Minimum ratio} = \frac{\text{Number of } \mathbf{Positive} \text{ Eigen values}}{\text{Number of Eigen values}}$$

- 每一個點都代表訓練 network 訓練完之後，把它的 Hessian 拿出來進行計算，訓練到 gradient 很小，卡在 critical point，把那組參數拿出來分析，看它比較像是 saddle point 還是比較像 local minima
- 縱軸代表 training 時的 loss，loss 沒辦法再下降時
- 橫軸是 minimum ratio
- 如果所有的 eigen value 都是正的，代表 critical point 是 local minima
- 如果有正有負代表是 saddle point
- local minima 不常見



Gradient 非常小的時候，有甚麼可能的解決辦法

課程網頁 (https://speech.ee.ntu.edu.tw/~hylee/ml/2022-spring.php)

tags: **2022 李宏毅_機器學習**