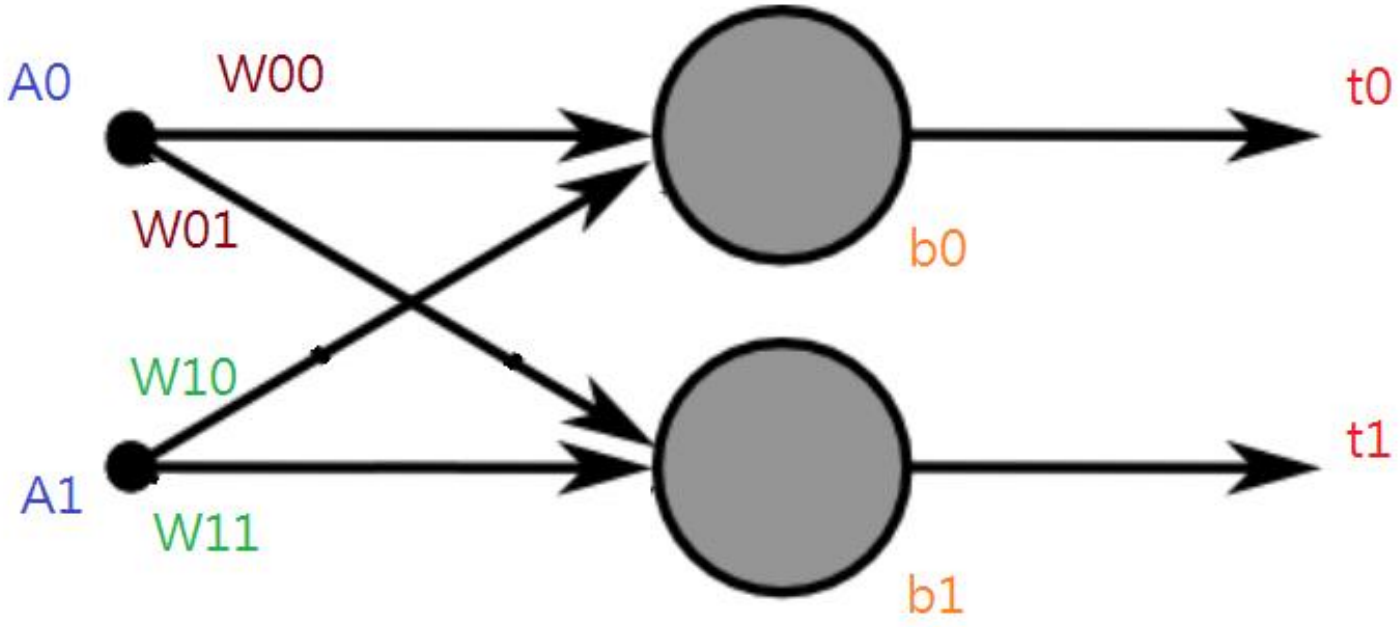
2024 鐵人賽 – 我數學就爛要怎麼來學 DNN 模型安全
Day 05 – DNN 模型基本概念

大綱

- 類神經網路
  - 激勵函數
  - 損失函數
  - 最佳化計算
- 概觀 DNN 訓練過程
- 結論

回憶一下之前的鸚鵡模型

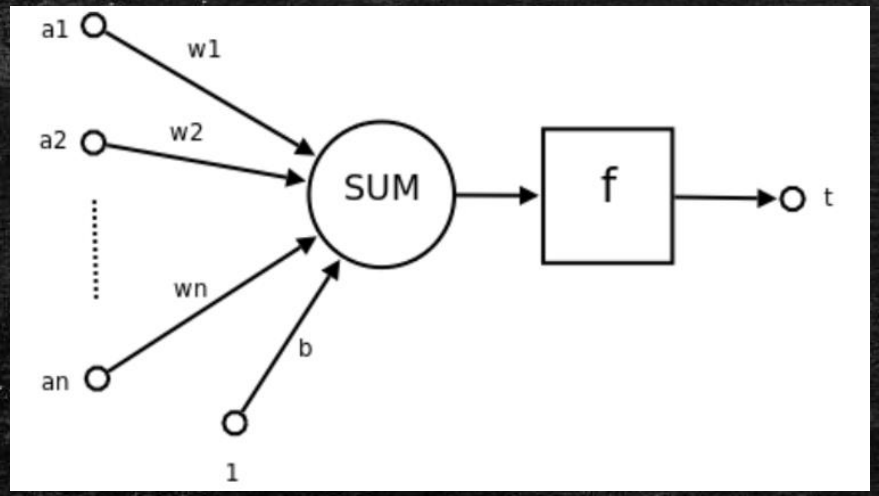
$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} A_0 \\ A_1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} A_0 \\ A_1 \end{bmatrix}$$



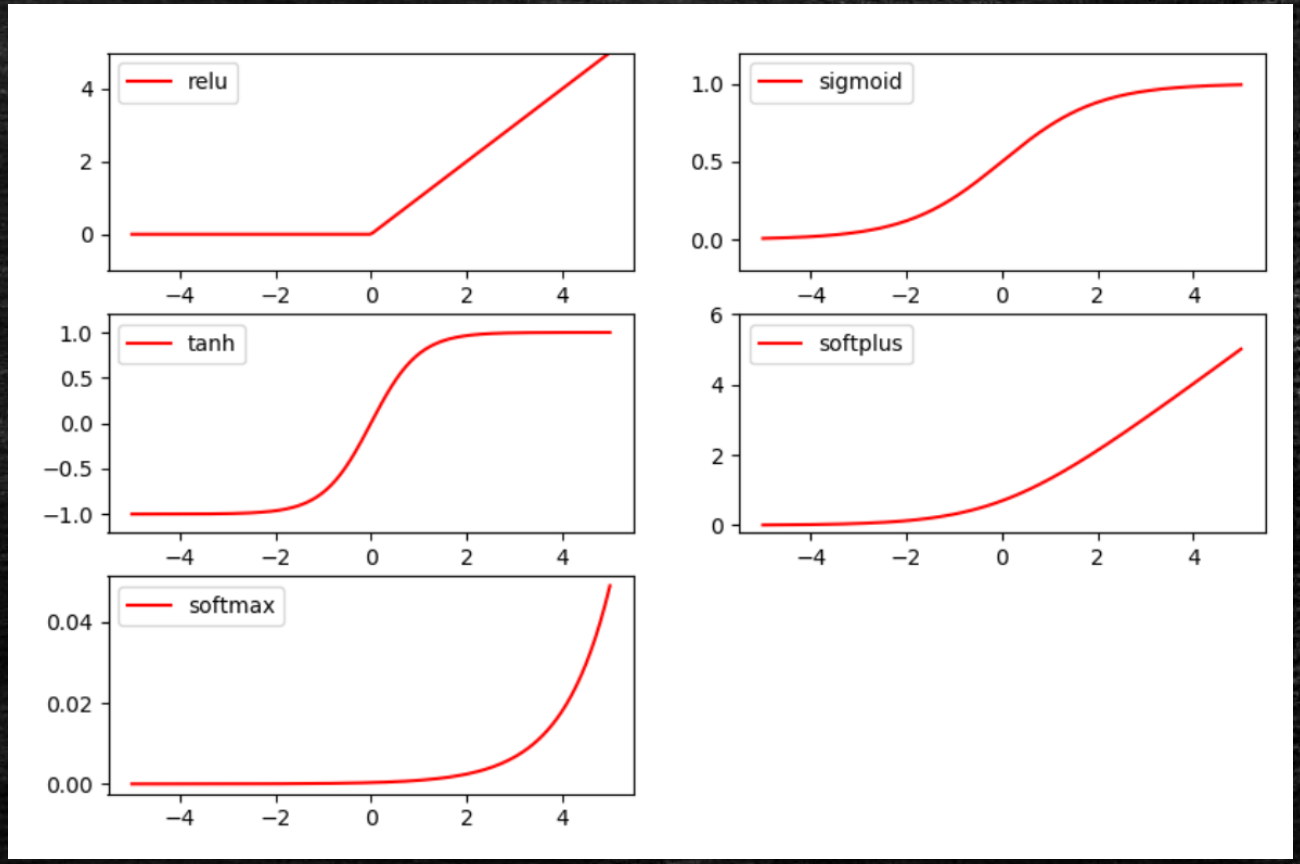
接著要讓機器用數據去學習 w,b 這兩種參數

激勵函數

- 線性的東西組合出來還是線性，所以需要一些函數產生非線性的結果



https://joe1in.github.io/dev_notes/ml/tensorflow/syntax.html#activation



損失函數

https://chih-sheng-huang821.medium.com/%E6%A9%9F%E5%99%A8-%E6%B7%B1%E5%BA%A6%E5%AD%B8%E7%BF%92-%E5%9F%BA%E7%A4%8E%E4%BB%8B%E7%B4%B9-%E6%90%8D%E5%A4%B1%E5%87%BD%E6%95%B8-loss-function-2dcac5ebb6cb

- 定義預測出來的結果和實際數值差距的函式

Loss: $L = \frac{1}{N}\sum_{n} e_n$

$e = |y - \hat{y}|$ L is mean absolute error (MAE)

$e = (y - \hat{y})^2$ L is mean square error (MSE)

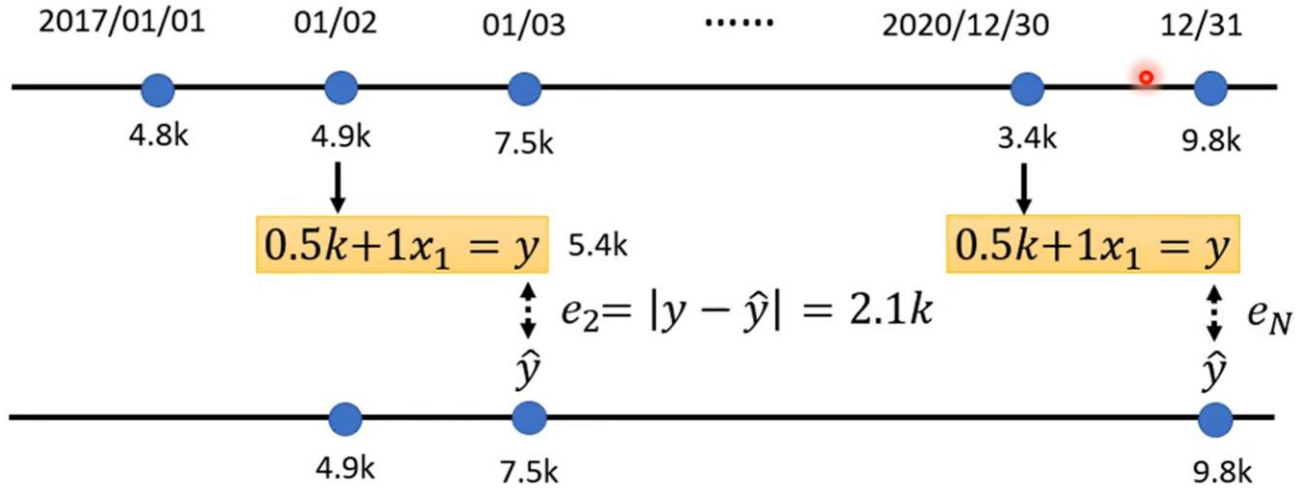
If y and $\hat{y}$ are both probability distributions ⟶ Cross-entropy

2. Define Loss from Training Data

- Loss is a function of parameters $L(b, w)$
- Loss: how good a set of values is.

$L(0.5k, 1)$ $y = b + wx_1$ ⟶ $y = 0.5k + 1x_1$ How good it is?

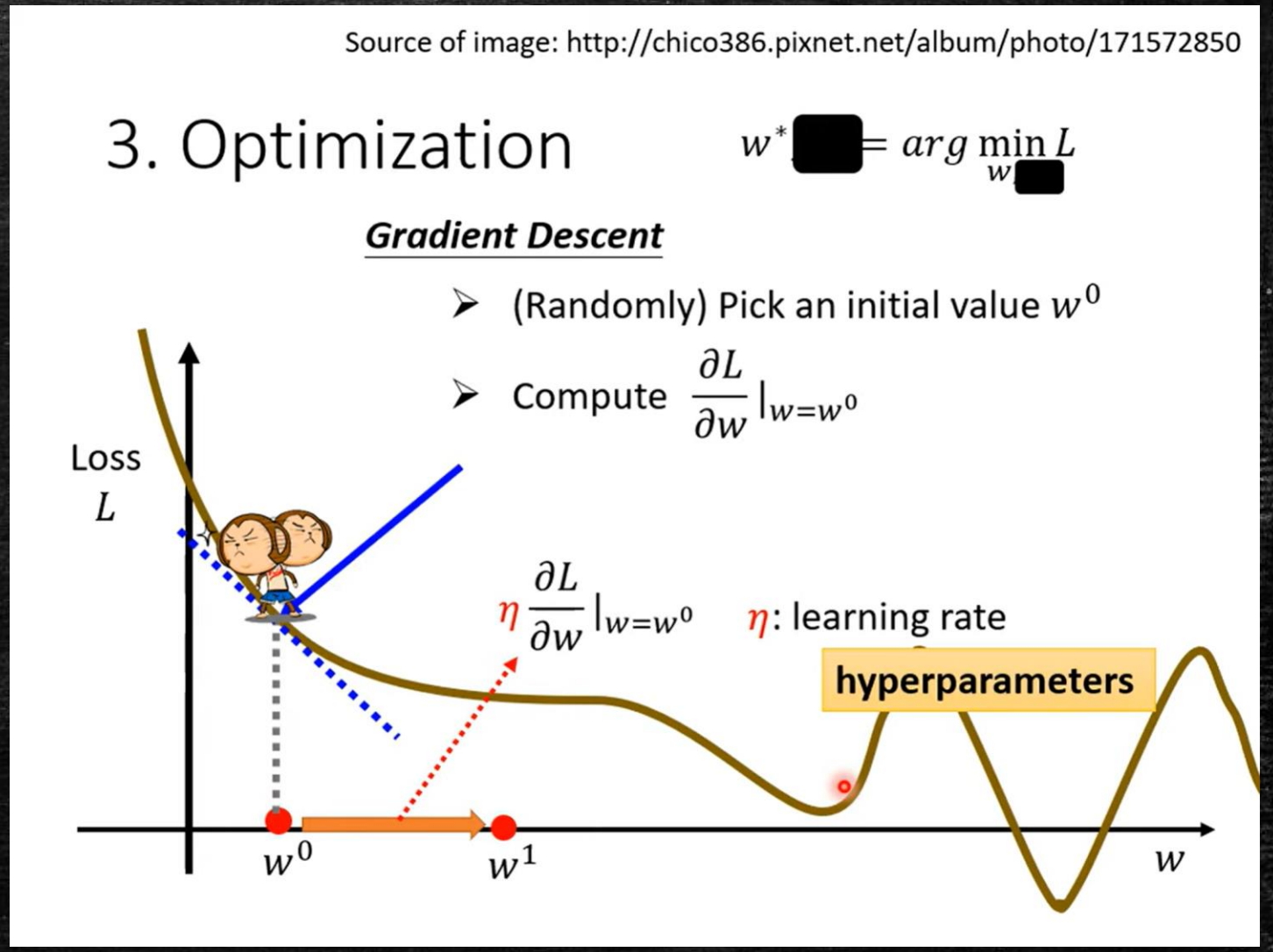
Data from 2017/01/01 – 2020/12/31



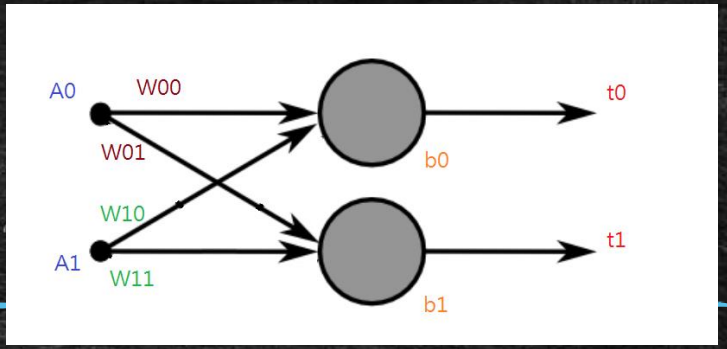
最佳化計算

- 對損失函數針對模型參數做偏微分求出梯度，在依照梯度的反方向做調整

$$w_{new} = w_{old} - \eta \frac{\partial loss}{\partial w_{old}}\Big|_{w=w_{old}}$$



概觀 DNN 模型的訓練過程



1. 定義 DNN 模型的結構，包含輸入層、隱藏層、輸出層的個數，並且初始化模型參數(權重)
2. 傳入訓練資料，針對每筆訓練資料輸入計算出模型輸出結果，並計算該結果跟該筆資料實際輸出的差距，其差距總和之計算訂為**損失函數**
3. 計算**模型內參數**對於**損失函數**的**梯度結果**，該梯度結果代表各參數之變動對損失函數結果的影響
4. 沿著梯度反方向乘上學習比例，並以該數值更新模型參數
5. 觀察結果是否滿意，否則繼續步驟 2

來用類神經網路算數學

- 來寫個程式預設 y = 2x + 1 看看

In [ ]:
```
# 建立屬於自己的 model
model = Sequential()
model.add(Dense(1, input_dim=1, activation='linear'))
```

In [ ]:
```
# 設定一些 function 後進行 compile
model.compile(optimizer = tf.optimizers.Adam(),
              loss='mean_squared_error')
model.summary()
```

In [ ]:
```
# 顯示模型的參數來看看
old_parameter = model.get_weights()
print(model.get_weights())
```

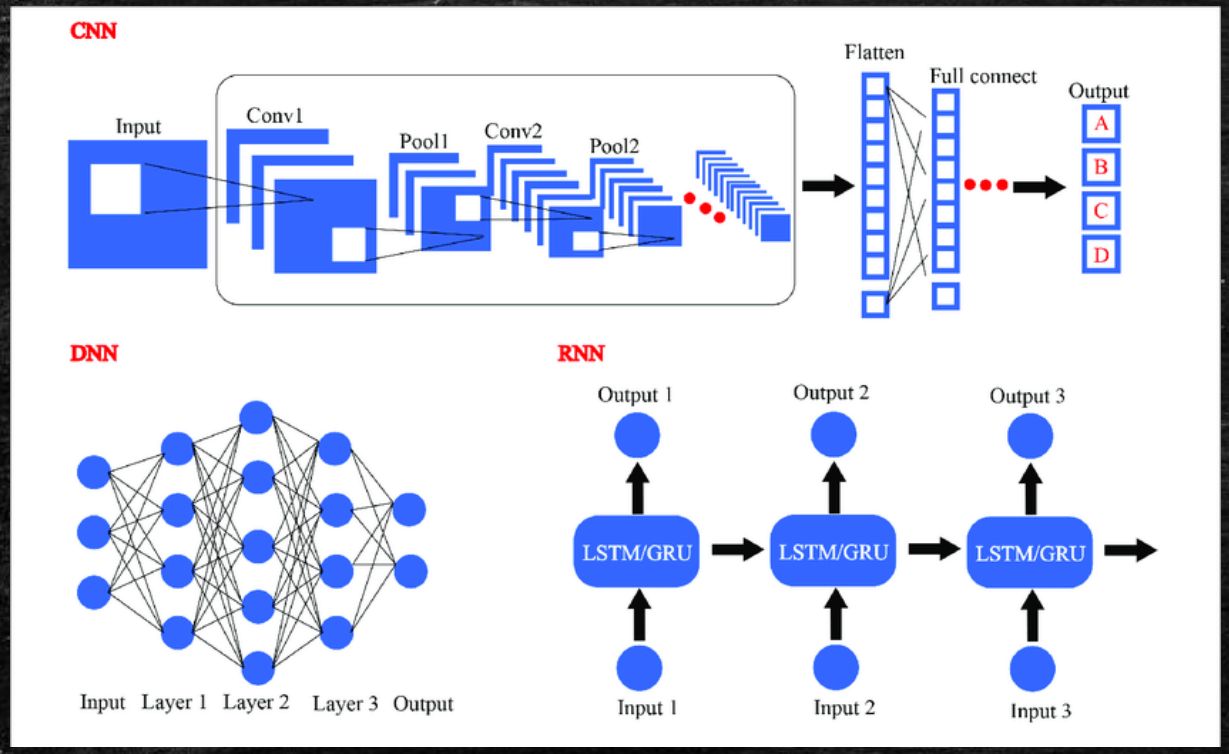
In [ ]:
```
# 開始傳入資料做訓練, epochs 代表要訓練幾回, batch_size 代表一次要讀取的資料作訓練的數量
model.fit(x, y, epochs=15, batch_size=50)
```

In [ ]:
```
# 顯示訓練後模型的參數來看看
print(old_parameter)
print(model.get_weights())
print(model.predict([10, 5, 200, 13]))
```

https://github.com/christianversloot/machine-learning-articles/blob/main/can-neural-networks-approximate-mathematical-functions.md

結論

- 類神經網路 Deep Neural Network (深度神經網路) 算是最基本的應用類型，針對不同應用情境還有 CNN、RNN



https://www.linkedin.com/pulse/comparison-dnn-cnn-lstm-using-tfkeras-joinal-ahmed