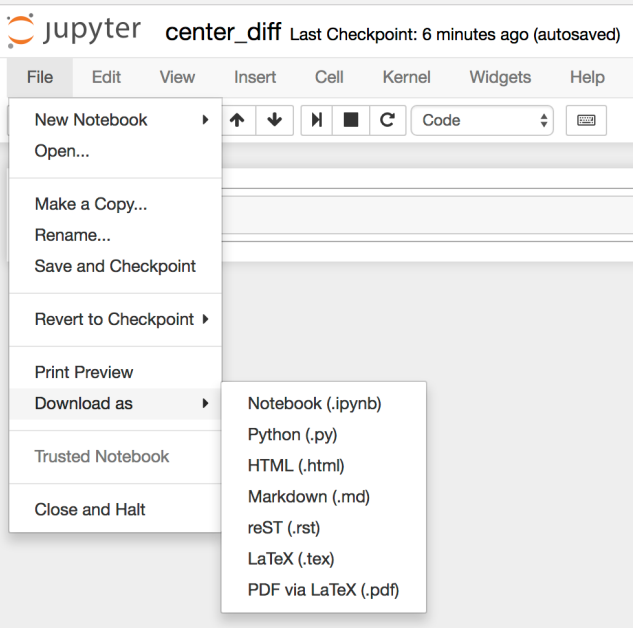
Homework 3
Concept, Derivation and Programming, Due 9:00, Tuesday, November 8, 2022

Late submission within 24 hours: score*0.9;
Late submission before post of solution: score*0.8 (the solution will usually be posted within a week); no late submission after the post of solution)

HW Submission Procedure (請仔細閱讀)

1. For concept and derivation, please write them in a professional format and submit a pdf file. Name your pdf file `YourID_HW3.pdf`, for example, `n96081494_HW3.pdf`

2. You should submit your Jupyter notebook and Python script (`*.py`, in Jupyter, click File, Download as, Python (`*.py`)).



3. Name a folder using your student id and HW number (e.g., n96081494_HW3), put the pdf and all the Jupyter notebooks and python scripts into the folder and zip the folder (e.g., n96081494_HW3.zip).

4. Submit your HW directly through the course website.

Total 100%

Concept and Derivation (Name your pdf file `YourID_HW3.pdf`, for example,`n96081494_HW3.pdf`)

1. (**50%**) Consider the simple network example with a single input $x = 2$ and a single output $y = 1$ shown in Figure 1 below.

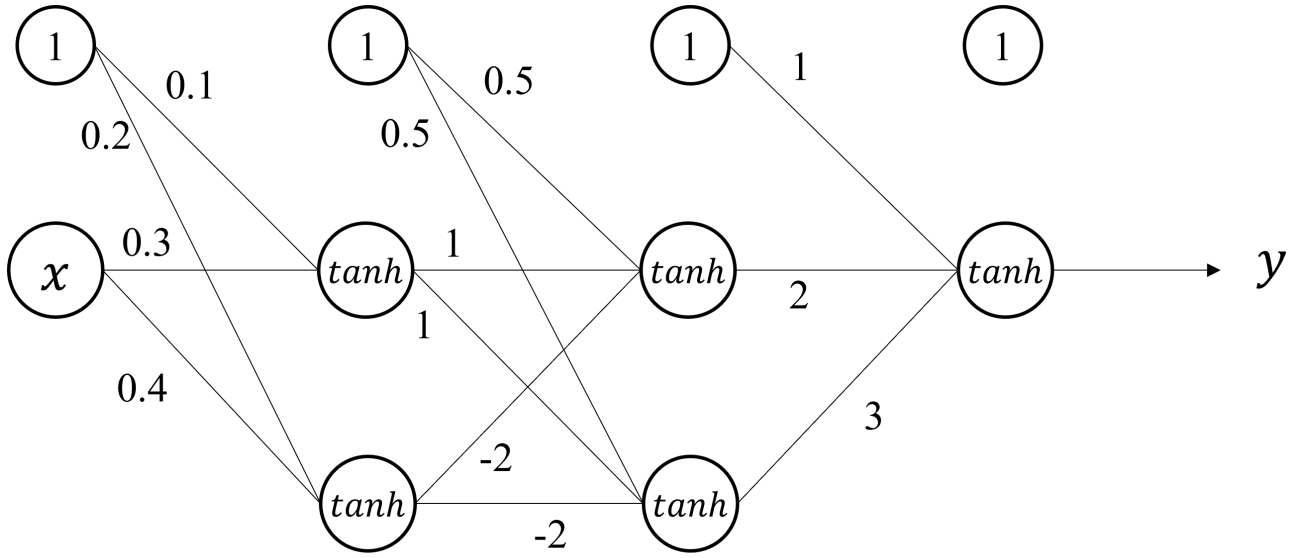


Figure 1

The weight matrices are:

$$\mathbf{W}^{(1)} = \begin{bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{bmatrix};\ \mathbf{W}^{(2)} = \begin{bmatrix} 0.5 & 0.5 \\ 1 & 1 \\ -2 & -2 \end{bmatrix};\ \mathbf{W}^{(3)} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

and the summation of weighted nodes for layer 1 can be expressed as $\mathbf{u}^{(1)} = (\mathbf{W}^{(1)})^T \mathbf{x}^{(0)}$; you can perform similar operation for other layers.

(a) (**10%**) Derive and compute $\mathbf{u}^{(1)}$, $\mathbf{z}^{(1)}$, $\mathbf{u}^{(2)}$, $\mathbf{z}^{(2)}$, and $\mathbf{y}^{(3)}$.

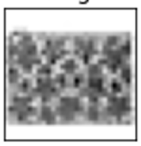
(b) (**15%**) Using the half of the sum square as our error function, derive and compute $\boldsymbol{\delta}^{(3)}$, $\boldsymbol{\delta}^{(2)}$, $\boldsymbol{\delta}^{(1)}$.

(c) (**10%**) Compute $\frac{\partial E_n}{\partial \mathbf{W}^{(1)}}$, $\frac{\partial E_n}{\partial \mathbf{W}^{(2)}}$, $\frac{\partial E_n}{\partial \mathbf{W}^{(3)}}$.

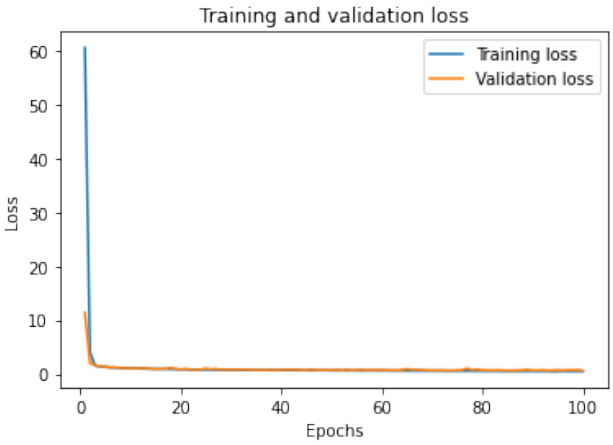
(d) (**15%**) Update the weight matrices using learning rate $\eta = 0.5$, repeat the forward propagation and compute $\mathbf{u}^{(1)}$, $\mathbf{z}^{(1)}$, $\mathbf{u}^{(2)}$, $\mathbf{z}^{(2)}$, and $\mathbf{y}^{(3)}$.

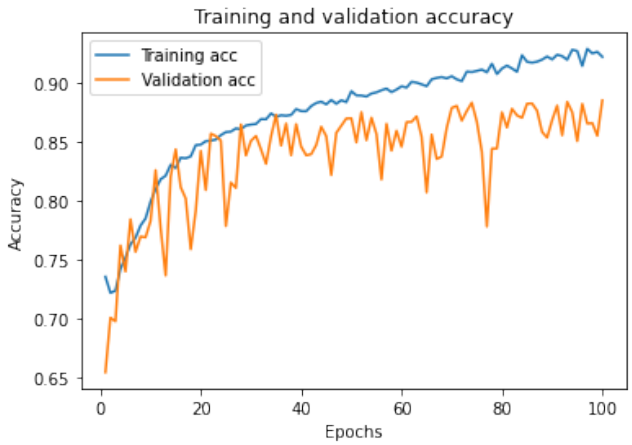
2. (**50%**)Name your Jupyter notebook `MLP` and Python script `MLP.py` Please create a ANN model to classify pictures of clothing. Please use `from tensorflow.keras.datasets import fashion_mnist` to read the training dataset. The data contains 10 classes of labels, the classes is shown in the figure below:

```
from tensorflow.keras.datasets import fashion_mnist
(x_train, y_train), (x_test, y_test) = fashion_mnist.load_data()
```

Label	0	1	2	3	4
	T-shirt/top 	Trouser 	Pullover 	Dress 	Coat 
Label	5	6	7	8	9
	Sandal 	Shirt 	Sneaker 	Bag 	Ankle boot 

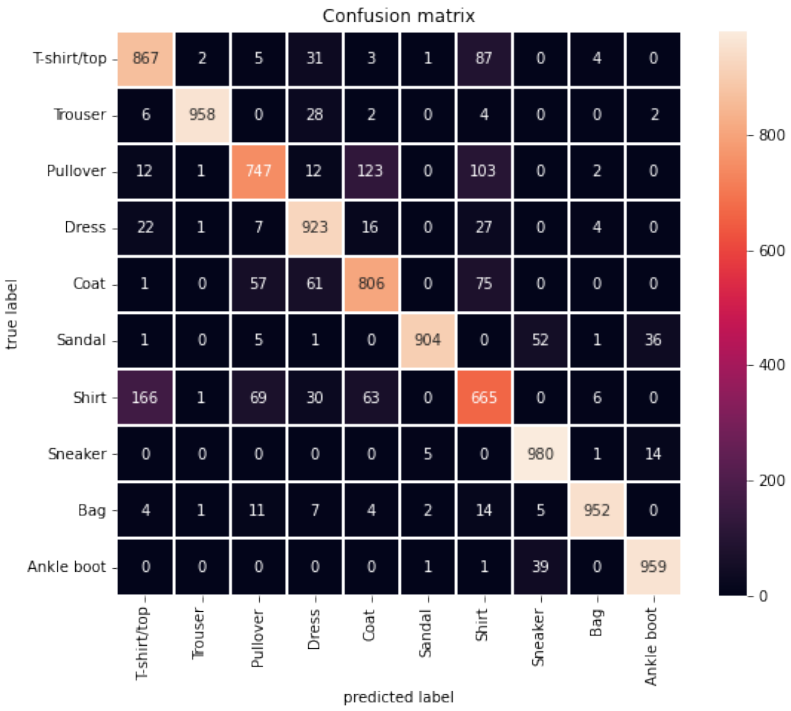
(a) (**20%**)Please use the training data `(x_train, y_train)` to train the model and plot the training history. Below is the sample plot:





(b) (**10%**)Please use the trained model to predict the test data `(x_test, y_test)` and plot the score and confusion matrix.

```
MLP score: 87.61%
```



(c) (**20%**)From the confusion matrix, it can be seen that some categories have more prediction errors, which may be the characteristics of the categories are more similar. Please plot the image of the clothes which prediction result is T-shirt but the label is Shirt. (Please plot 5 wrongly predicted clothes images)

`Predict shirt as T-shirt: 166`