



- 3.6: implement the BP algorithm. 30 points
- 3.7: update all weights and bias. 5 points
- 3.8: calculate the accuracy. 5 points
- 4. Evaluation. 15 points.

```
In [5]: import numpy as np
from matplotlib import pyplot as plt
import keras
from sklearn.model_selection import train_test_split
```

```
In [6]: ## Load the digits dataset
def load_digits(show_sample = True):
    (x_train, y_train), (x_test, y_test) = keras.datasets.mnist.load_data()

    #show first 100 images
    if show_sample == True:
        nImg = 4
        for i in range(nImg*nImg):
            plt.subplot(nImg, nImg, i+1)
            plt.imshow(x_train[i], cmap = 'Greys_r')
        plt.show()

    x_train_1 = np.reshape(x_train, [x_train.shape[0], x_train.shape[1] * x_train.shape[2]])
    x_test_1 = np.reshape(x_test, [x_test.shape[0], x_test.shape[1] * x_test.shape[2]])

    return x_train_1, y_train, x_test_1, y_test

# x_train, y_train, x_test, y_test = load_wine()
```

```
In [3]: ## Task 1: Activation functions: implement the softmax function. 5 points
def sigm(z):
    return 1/(1 + np.exp(-z))

def dsigm(z):
    return sigm(z)*(1 - sigm(z))
```