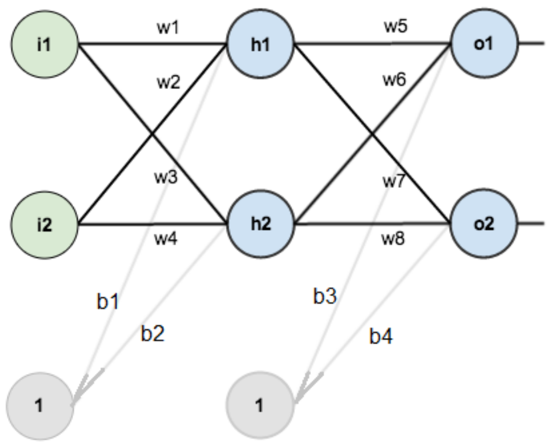
National University of Singapore
College of Design & Engineering - ECE

EE4400 Data Engineering and Deep Learning
Tutorial 1 (MLP/NN)

Q1. The data file `government-expenditure-on-education.csv` depicts a government's educational expenditure over the years 1981 to 2018. Write a Python notebook or program to do the following:
(a) Read the data file and scale the year and expenditure data pairs to the range [0, 1] and let them be the $\mathbf{x}$ and $\mathbf{y}_{\text{target}}$ values.
(b) Construct a linear model $f(\mathbf{x}, \mathbf{w}) = \mathbf{x}^T\mathbf{w}$ with 2 parameters which can be used to predict the educational expenditure in the years 2019 to 2023. The loss function is the sum of squared error and regularization is not required.
(c) Train the model with gradient descent with a small learning rate for some number of iterations.
Hint: Derive the gradient of the cost function with respect to the parameters.
(d) Plot the cost function with respect to the iteration number.
(e) Plot the actual expenditure (for 1981 to 2018) and predicted expenditure (for 1981 to 2023) with respect to the years.

Q2. A multi-layer perceptron (MLP) has two inputs, one hidden layer with 2 neurons and two output neurons.



The inputs are i1=0.10 and i2=0.80. The weight values are:
w1=0.30, w2=0.25, w3=0.11, w4=0.20, w5=0.07, w6=0.15, w7=0.19, w8=0.22
b1=0.13, b2=0.71, b3=0.30, b4=0.14
The learning rate is 0.1.
The sigmoid activation function is used in all the hidden and output neurons.
Compute the outputs o1 and o2.

The target output values are do1=0.95 (i.e. desired o1) and do2=0.05
The back-propagation (BP) algorithm is used to compute the changes to the weights.

The learning rate is 0.1.

What are the weight changes for w5, w7 and w1?

Q3. This question is on the Multi-Layer Perceptron (MLP) and using it to do classification. The aim is to find the best number of hidden nodes in the 3 hidden layers, assuming the same number of hidden nodes in each hidden layer. Cross-validation needs to be done on the training set. The MLP classifier with the best network size is then used for testing.

We shall use the Tensorflow Keras package to implement the MLP classifier. Obtain the data set "`from sklearn.datasets import load_iris`". Import the necessary packages.

(a) Load the data and split the database into two sets: 80% of samples for training, and 20% of samples for testing.
Hint: use `train_test_split()` with `random_state=0`. Since this is a multi-class classification task, converting the target values to categorical values is recommended.

(b) Perform a 5-fold cross-validation using only the training set to determine the best 3-layer MLP classifier with hidden_layer_sizes Nhidd, Nhidd and Nhidd, for Nhidd in range(1,11))^ for prediction.

^ The assumption of hidden_layer_sizes Nhidd, Nhidd and Nhidd is to reduce the search space in this exercise. In actual applications, the search may need to consider different sizes for each hidden layer.

In other words, partition the training set into two sets, 4/5 for training and 1/5 for validation; and repeat this process until each of the 1/5 has been validated.

There are many possible settings for the Keras functions. Start with default values or options which have been covered in lectures, e.g. SGD optimizer, ReLU or sigmoid activation function etc. Start with a limit of 100 epochs of training. *Note*: These are hyperparameters.

Hint: since this is a multi-class classification task, use the "`categorical_crossentropy`" loss function.

(c) Provide a plot of the average 5-fold training and validation accuracies over the different network sizes, i.e. different number of nodes in the hidden layers. Determine the hidden layer size Nhidd that gives the best validation accuracy for the training set.

(d) Using the best hidden layer size Nhidd in the MLP classifier with hidden_layer_sizes Nhidd, Nhidd and Nhidd, evaluate the performance of the MLP by computing the prediction accuracy based on the 20% of samples for testing in part (a).

(e) Determine the number of MLP parameters (weights) in each case of Nhidd and explain the value for each case.

(f) Experiment with slightly different settings for one or two of the hyperparameters and observe any changes in accuracy.