ECE 805 – Machine Learning

Homework #02

Spring 2023

(due on Thursday, 09 February 2023, 16:00, submitted via Teams)

General instructions

This assignment consists of two questions. You are requested to submit both your code and a report; the report should contain the requested visualisation plots and your answers to possible questions.
Please submit one compressed file (zip), named ECE805_HW2_Yourname.

Exercise 1 (50%):

(a) State if the following statement is True or False, and justify your answer.

A Linear SVM (with hard constraints) produces the same decision boundary as Logistic Regression (without regularisation).

(b) You are provided with a dataset (hw2_ex1_dataset.csv), and a Python script which is part of the solution (hw2_ex1_solution.py). It is strongly suggested to use the versions of the Python packages shown in the requirements.txt file. You will be requested to fill in the missing parts in the script (displayed with "TODO"). Do not modify any other part of the code.

i. You will be using scikit-learn's SGDClassifier function for running a Linear SVM with soft constraints, and the SVC function for running an SVM with kernel functions. Import these two functions **(Line 8)**.

ii. Define four classifiers as follows **(Line 16)**.
- A Linear SVM with the hinge loss function and L2 regularisation with value 0.0001.
- A Linear SVM with the hinge loss function and L2 regularisation with value 1.0.
- An SVM with the Radial Basis Function (RBF) kernel.
- An SVM with the Polynomial with degree 2 kernel.

Leave the default values for the rest of the hyper-parameters.

iii. Load the dataset (**Line 24**).

iv. Fit the data to the classifier, and obtain the classifier's score (**Line 81**).

v. Run the code and provide the generated figure with the four plots. The plots show training points in solid colours and testing points in semi-transparent colour. The lower right number in each plot shows the classification accuracy on the test set. Discuss your findings.

Exercise 2 (50%):

It is not necessary to write code, but do submit any code that you might use along with the handwritten assignment. Write down all your reasoning.

You are given the feed forward neural network in Figure 1, with activation functions h_1, h_2, input vector $\boldsymbol{x} = (x_1, x_2, x_3)^T$ weights $k_1, k_2, k_3, k, w_1, w_2$ and w_3, and biases b_1 and b_2.

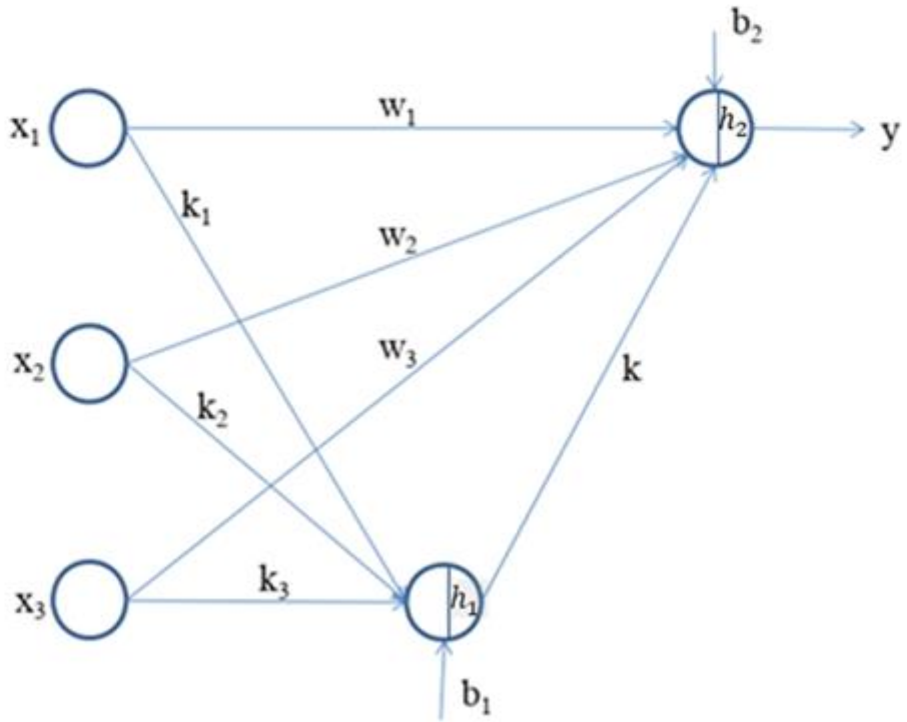

Figure 1

(a) Express the output of the network $y(\boldsymbol{x})$ as a function of the input vector.

(b) Assume that the input vector elements are binary (take values 0 or 1), and that the network parameters are:

$w_1 = w_2 = w_3 = 2$

$k_1 = k_2 = k_3 = 2$

$k = -3$

$b_1 = -3, b_2 = -1{,}5$

And that the activation functions are of the following form:

$$h(a) = \begin{cases} 1, & a > 0 \\ 0, & a \leq 0 \end{cases}$$

Investigate if this architecture can solve the three-bit parity problem, i.e., if the sum of the elements of the binary vector is an even number then the network output should be 0, otherwise it should be 1.

(c) Assume that the activation function h_1 is the identity function. Can you find an equivalent single neuron neural network given the parameter values in (b)? If so, check if this network can solve the three-bit parity problem.

(d) For the neural network in Figure 1, now assume that the activation functions h have the below form, and that is trained using the absolute error loss function $|\hat{y} - y|$ where $\hat{y}$ is the ground truth value, and learning rate $\lambda = 0.01$.

$$h(a) = \begin{cases} a, & a > 0 \\ 0, & a \leq 0 \end{cases}$$

Derive the general update rule for parameter k_2 using standard gradient descent for a single example.