Laboratory 2 Instructions
The Perceptron and Its Applications

Hasan Fleyeh, Ross May, Arend Hintze

September 16, 2020

The objectives of this lab are:

1. To implement a single perceptron and understand its ability for linear seperability.
2. To apply linear seperability using a single perceptron on the logical functions, *AND* ($\wedge$) and *OR* ($\vee$).
3. To use the perceptron on a real dataset, namely, the *Iris* dataset.

1 Task 1: Implement a perceptron learning algorithm

- Implement the following learning algorithm for the perceptron model of an artificial neuron (where the first column $\mathbf{x}_1 \in \mathbf{X}$ is the column of artificial ones; see slide 20 of Lecture 3: the Perceptron):

Algorithm 1 Perceptron learning algorithm for linearly seperable datasets

Input: Data features ($\mathbf{X}$), labels ($\mathbf{y}$), randomly intialised weights ($\mathbf{w}$), learning rate (η)
Output: Updated weights

while $\hat{\mathbf{y}} \neq \mathbf{y}$ **do** ▷ Predicted vector $\hat{\mathbf{y}} \equiv \mathbf{X}\mathbf{w}$ and $\mathbf{y}$ true class vector
  Select a pair $(\mathbf{x}^{(i)}, y^{(i)}) \in \mathbf{X}$ ▷ True class label, $y^{(i)}$, corresponding to input, $\mathbf{x}^{(i)}$
  if $\mathbf{w}^T\mathbf{x}^{(i)} \geq 0$ **then**
    $\hat{y}^{(i)} \leftarrow 1$ ▷ Predicted output, $\hat{y}^{(i)}$
  else
    $\hat{y}^{(i)} \leftarrow 0$
  end if
  if $\hat{y}^{(i)} \neq y^{(i)}$ **then**
    $\Delta\mathbf{w} \leftarrow \eta(y^{(i)} - \hat{y}^{(i)})\mathbf{x}^{(i)}$
    $\mathbf{w} \leftarrow \mathbf{w} + \Delta\mathbf{w}$
  end if
end while
return: $\mathbf{w}_{new}$ ▷ Learned weights

2 Task 2: Apply the perceptron learning rule to $\wedge$ and $\vee$

Extend Task 1 by applying the perceptron learning algorithm to the logical functions

1. $\wedge$ (AND gate)
2. $\vee$ (OR gate)

- Use the truth tables of these gates as patterns (i.e. inputs) and the output of the gates as the true class for each given input.
- Plot the positions of these four points on a graph using a suitable scale in the (x, y)-plane.
- Plot the decision lines after each epoch.

3 Task 3: Apply the perceptron learning rule to a real dataset

For this task the perceptron learning algorithm will be used to classify the *IRIS dataset.*. Notice that the dataset contains 3 classes of 50 instances (samples) each where the class attribute refers to the type of Iris plant. Furthermore, notice that one class is linearly seperable from the other two and the latter are not linearly separable from each other - this can be seen by creating an appropriate plot of the dataset.

- Write a function that retrieves the IRIS dataset (url: *https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data*) and converts it into a suitable format that can be dealt with by the perceptron. This can be achieved by slicing the appropriate samples (remember that two are not linearly separable from each other) and converting the class labels from text to numbers.
- Train the perceptron in Task 1 to classify the two classes.
- Plot the line of separability between the two classes.

4 Task 4: Discussion

Report and document the steps you followed to achieve the tasks above. Discuss your results and suggest a way to improve them.