Homework 2: Linear separability of 4D Boolean functions

Hai Dinh (19960331-4494)

1. Code

```python
import time
import numpy as np

# Global parameters
eta = 0.02
max_number_of_updates = int(1e5)
max_number_of_repetitions = 10

# Loading data
X = np.genfromtxt("input_data_numeric.csv", delimiter=",")[:,1:]
Y = {
    "A": np.array([+1, +1, -1, +1, -1, +1, +1, +1, -1, -1, -1, -1, -1, -1, -1, -1]),
    "B": np.array([+1, -1, +1, +1, +1, -1, +1, +1, +1, +1, +1, +1, -1, +1, +1, +1]),
    "C": np.array([-1, -1, -1, +1, -1, -1, +1, -1, +1, -1, +1, +1, -1, +1, -1, +1]),
    "D": np.array([+1, -1, +1, -1, -1, +1, -1, -1, -1, -1, -1, -1, +1, -1, -1, +1]),
    "E": np.array([+1, +1, +1, +1, -1, -1, -1, -1, -1, +1, -1, -1, +1, -1, -1, -1]),
    "F": np.array([-1, -1, +1, +1, +1, -1, -1, -1, -1, +1, +1, -1, +1, +1, +1, -1]),
}


def check_linear_separability(function_name):
    start_time = time.time()
    y_true = Y[function_name]
    linearly_separable = False

    for i_repetition in range(1, max_number_of_repetitions + 1):
        W = np.random.uniform(low=-0.2, high=0.2, size=(4,))
        theta = np.random.uniform(low=-1.0, high=1.0, size=None)

        for i_update in range(1, max_number_of_updates + 1):
            # Forward propagate for all data points
            local_fields = (X @ W - theta) / 2
            outputs = np.tanh(local_fields)
            y_pred = np.sign(outputs)

            # Check convergence criteria using all data points
            if np.array_equal(y_pred, y_true):
                linearly_separable = True
                break

            # Backward propagate to update weights and threshold
            mu = np.random.choice(X.shape[0])
            error = (y_true[mu] - outputs[mu]) * (1 - np.tanh(local_fields[mu])**2)
```

```python
            W += eta * error * X[mu]
            theta -= eta * error

        if linearly_separable:
            break

    print("Results for Boolean function " + function_name + ":")
    print("    Linearly separable:", "YES" if linearly_separable else "NO")
    print("    Final weights      :", W)
    print("    Final threshold    :", theta)
    print("    Total time taken   :", (time.time() - start_time), "seconds")
    print("    Total #repetitions:", i_repetition)
    print("    Total #updates in last repetition: ", i_update)

if __name__ == "__main__":
    [check_linear_separability(name) for name in Y.keys()]
```

2. Results

```
Results for Boolean function A:
    Linearly separable: YES
    Final weights      : [ 0.38027807 -0.64544005 -0.29010912 -0.64814763]
    Final threshold    : 0.1591475899711617
    Total time taken   : 0.0032126903533935547 seconds
    Total #repetitions: 1
    Total #updates in last repetition:  105
Results for Boolean function B:
    Linearly separable: YES
    Final weights      : [-0.63748586 -0.15120174  0.64939551  0.31684523]
    Final threshold    : -1.1135515460963725
    Total time taken   : 0.0048370361328125 seconds
    Total #repetitions: 1
    Total #updates in last repetition:  170
Results for Boolean function C:
    Linearly separable: YES
    Final weights      : [ 0.42798437 -0.34069052  1.56669423 -0.403037  ]
    Final threshold    : 0.4375861576029526
    Total time taken   : 0.00819253921508789 seconds
    Total #repetitions: 1
    Total #updates in last repetition:  281
Results for Boolean function D:
    Linearly separable: NO
    Final weights      : [ 1.8602917   3.94948184 -2.24257376 -0.04425914]
    Final threshold    : 3.9712242478853836
    Total time taken   : 29.376785039901733 seconds
    Total #repetitions: 10
    Total #updates in last repetition:  100000
Results for Boolean function E:
    Linearly separable: NO
    Final weights      : [-5.21795027 -2.61466707 -5.17015406 -2.6411312 ]
    Final threshold    : 2.632913232394134
    Total time taken   : 29.003007888793945 seconds
    Total #repetitions: 10
```

```
    Total #updates in last repetition:  100000
Results for Boolean function F:
    Linearly separable: NO
    Final weights    : [-2.76792626  0.06457685  0.22788824  2.47077662]
    Final threshold  : -0.1663542164269784
    Total time taken  : 29.53967833518982 seconds
    Total #repetitions: 10
    Total #updates in last repetition:  100000
```