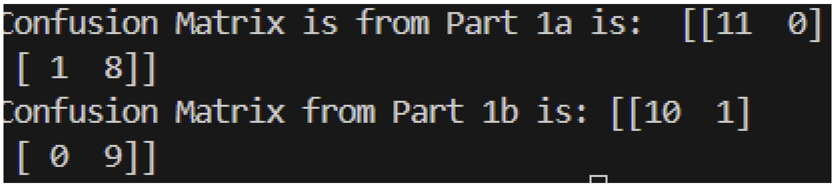
Part 1:

1. tol is a stopping criterion that looks at how much the training loss improves between iterations. If tol is not None, the iterations will stop when (loss > previous_loss - tol).

2. No it might not pass the 5000 times. The Perceptron can stop early once when (loss > previous_loss - tol).

3. We can use an np array to change the Perceptron class's coef_ attribute for changing weights.

4. They are very close to each other as they are using the same algorithm.



```
Confusion Matrix is from Part 1a is:  [[11  0]
 [ 1  8]]
Confusion Matrix from Part 1b is: [[10  1]
 [ 0  9]]
```

Part2:

1. Yes, X is a singular matrix as the columns are linearly dependent. So X^T X is also singular and has no inverse.

2. ERROR

3. weights: [-1.59872116e-14 2.00000000e-01 4.00000000e-01]

np.linalg.inv computes the exact inverse of a full rank matrix whereas np.linalg.pinv computes the Moore-Penrose Pseudoinverse, which always exists.