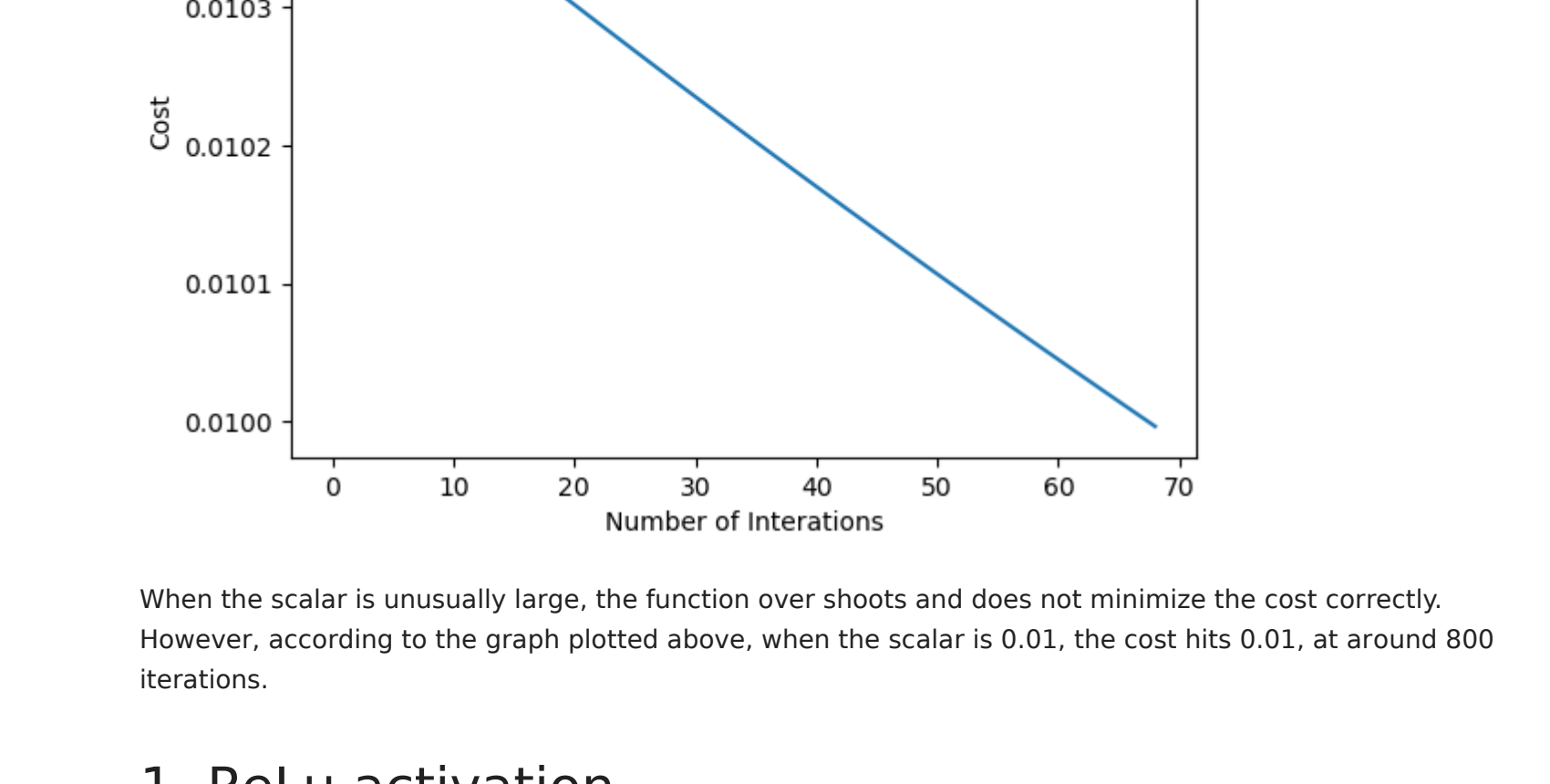
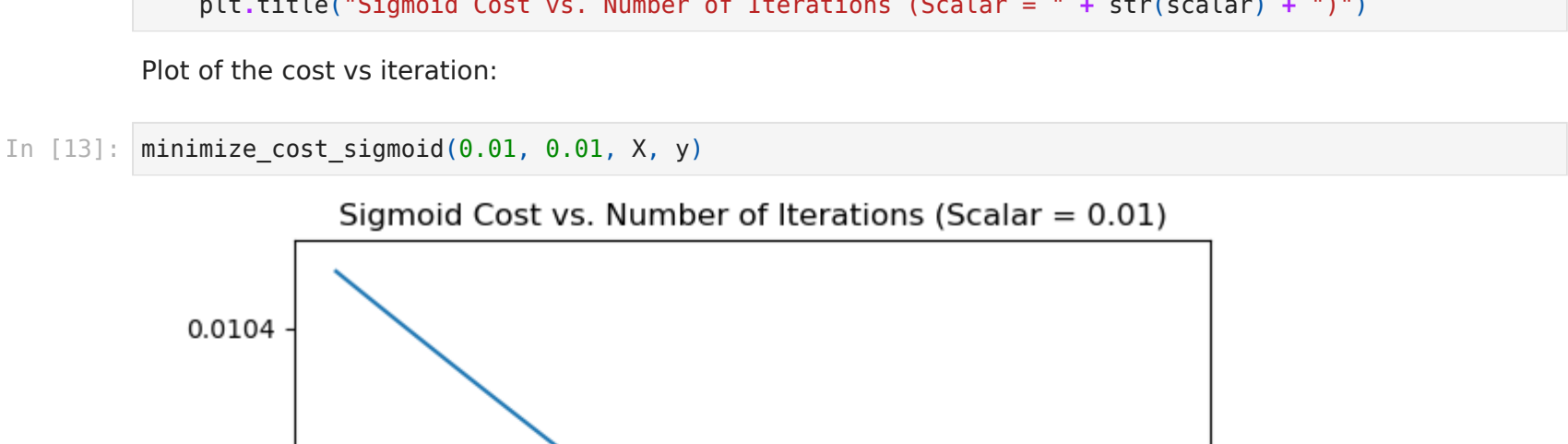


1.2 Structure of a neural network

A program that minimizes the cost function to a given accuracy set in advance:

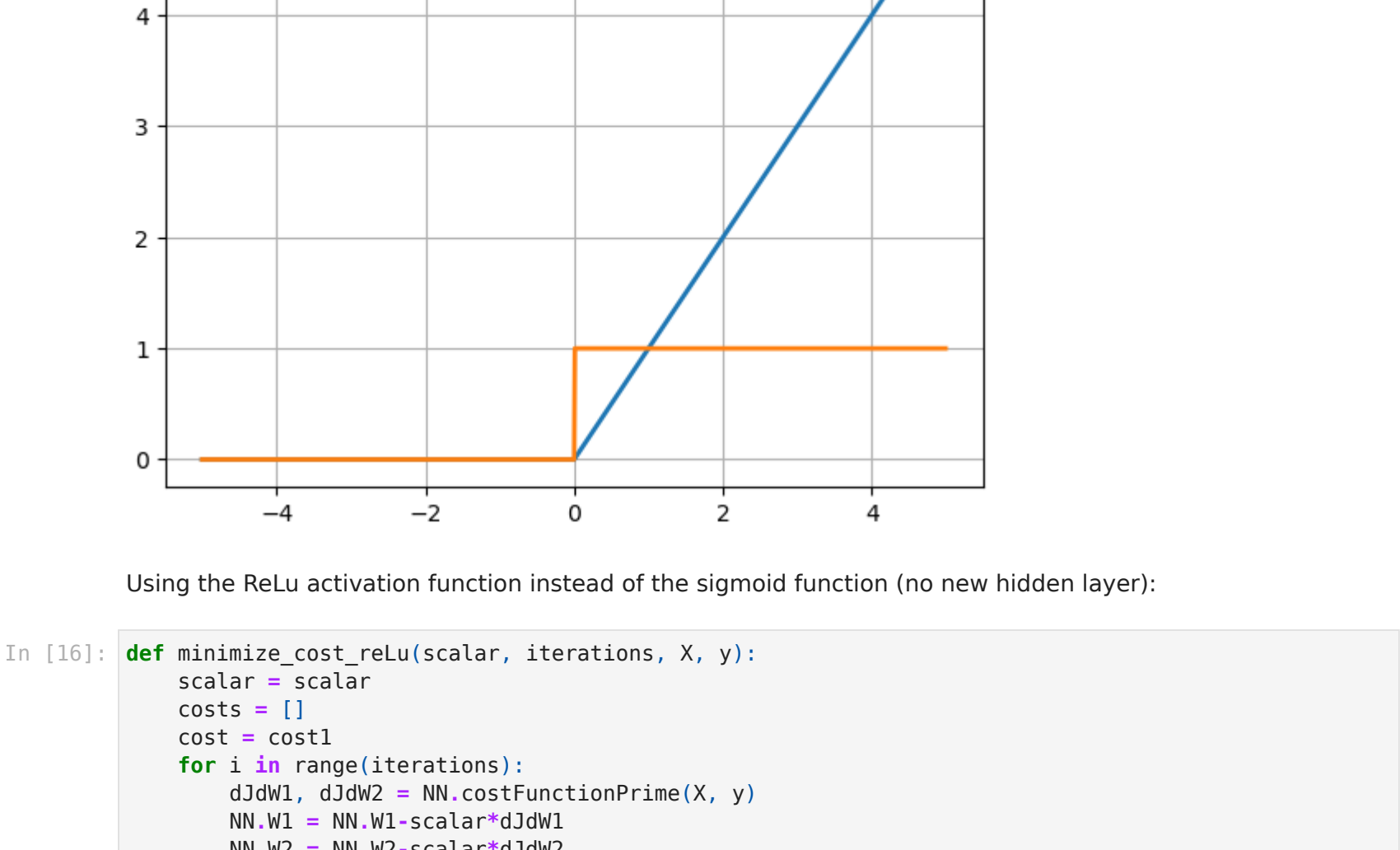


When the scalar is unusually large, the function over shoots and does not minimize the cost correctly. However, according to the graph plotted above, when the scalar is 0.01, the cost hits 0.01, at around 800 iterations.

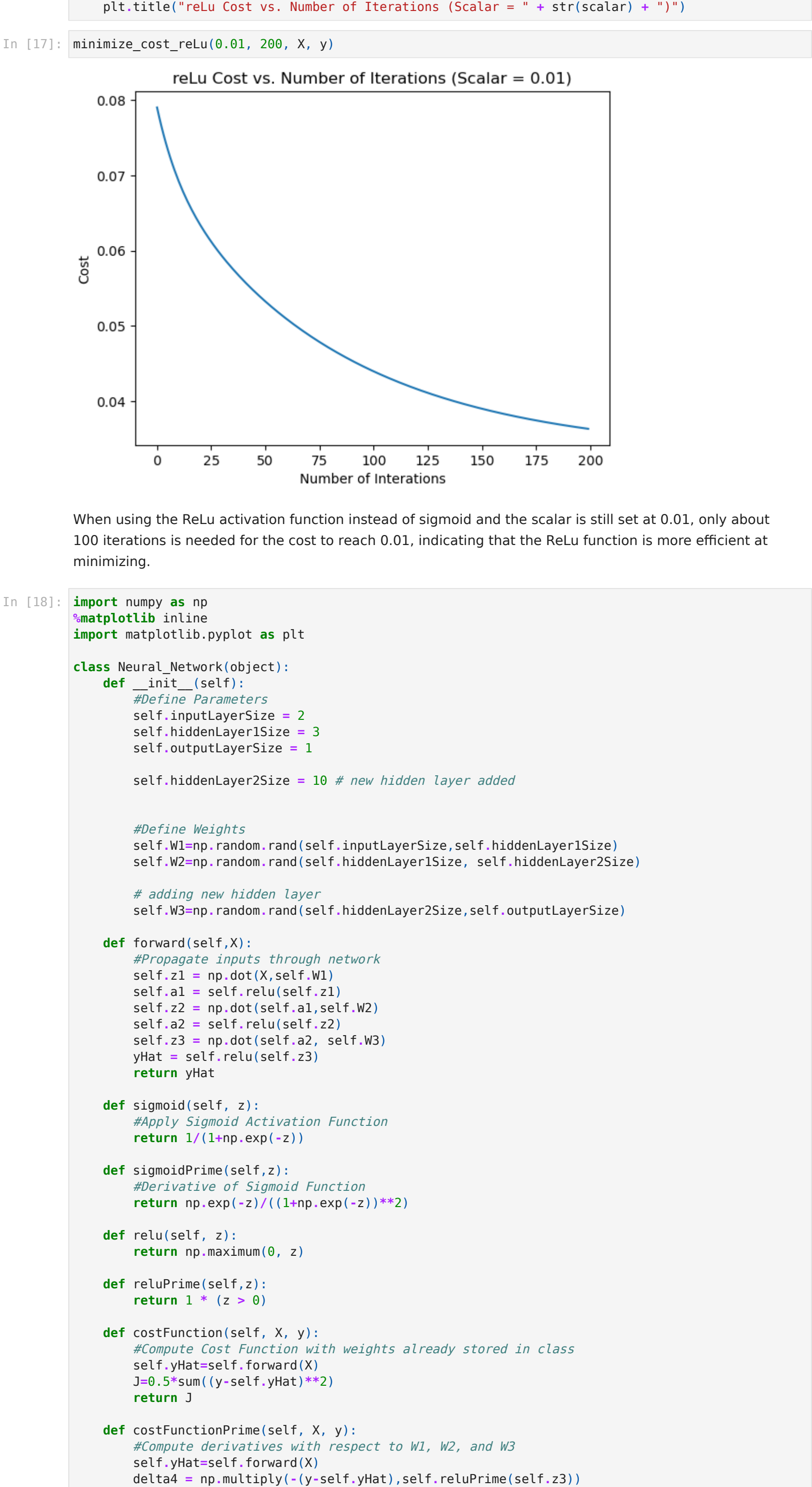
1. ReLu activation

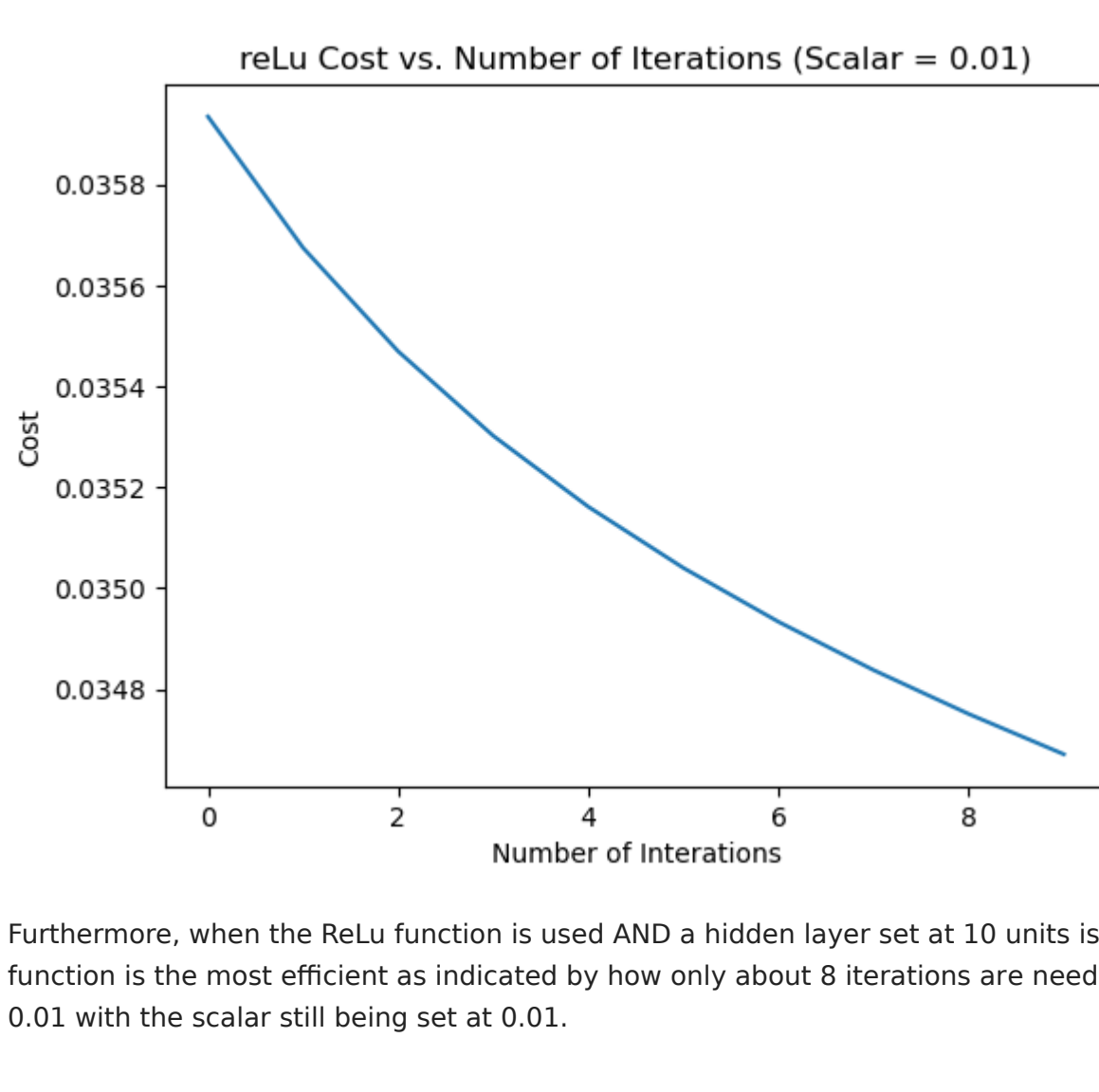


Using the ReLu activation function instead of the sigmoid function (no new hidden layer):



When using the ReLu activation function instead of sigmoid and the scalar is still set at 0.01, only about 100 iterations is needed for the cost to reach 0.01, indicating that the ReLu function is more efficient at minimizing.





Furthermore, when the ReLu function is used AND a hidden layer set at 10 units is added, the minimization function is the most efficient as indicated by how only about 8 iterations are needed for the cost to hit 0.01 with the scalar still being set at 0.01.