2b)

A screenshot of a cell phone

Description automatically generated

These are different ways to write gradient descent, the first here uses vectors and the second uses a summation. These are the same, because vector math essentially allows us to perform the summation in one step, for example subtracting the y vector from our hypothesis subtracts the corresponding y value from each row of our hypothesis, which is the same thing as stepping through each element via an index.

2c)

The first is a better implementation, because while they both provide the same output, there are serious advantages to using vectors. Matrix multiplication is highly optimized, and as a result performs much faster than looping. This means that by converting the loop or summation to a matrix multiplication problem dramatically reduces compute time and lets our gradient descent algorithm perform more iterations in less time.

Learning Rate)

After running gradient descent, I found a learning rate of .01 with a high amount of iterations gave me the value closest to that obtained via the Normal equation. As a result, I think this is a good choice, as it still converged fairly quickly and gave the expected value.