Question 1 a & b:

a) **Experiment 1: Base line (Nothing changed)**

Parameters:

Classes: 0,1
N: none
Hidden_layer_sizes: 8
Max_iter: 200

Results:

2-class| Us: 0.998; Sklearn: 0.998 Logreg: 0.999

Comments:

a) **Experiment 1: Base line (Nothing changed)**

Parameters:

Sklearn_kwargs:
Hidden_layer_sizes: 64
Max_iter: 2000

Results:

10-class| Sklearn: 0.946; Logreg: 0.897

Comments:

a) **Experiment 2: Halved max iterations**

Parameters:

Classes: 1
N: none
Hidden_layer_sizes: 8
Max_iter: 100

Results:

2-class| Us: 0.997; Sklearn: 0.998 Logreg: 0.999

Comments:

The accuracy is very close and the same as in the baseline and thus the lower number of iterations still fitted the model very well.

b) **Experiment 2: Halved max iterations**

Parameters:

Sklearn_kwargs:
Hidden_layer_sizes: 64
Max_iter: 1000

Results:

10-class| Sklearn: 0.946; Logreg: 0.900

Comments:

The accuracy is very close and the same as in the baseline and thus the lower number of iterations still fitted the model very well.

a) **Experiment 3: Doubled max iterations**

Parameters:

Classes: 1
N: none
Hidden_layer_sizes: 8
Max_iter: 400

Results:

2-class| Us: 0.998; Sklearn: 0.998 Logreg: 0.999

Comments:

The accuracy is the same as in the baseline and thus the extra number of iterations had no influence as the model was already well fitted at a lower amount of iterations.

b) **Experiment 3: Doubled max iterations**

Parameters:

Sklearn_kwargs:
Hidden_layer_sizes: 64
Max_iter: 4000

Results:

10-class| Sklearn: 0.946; Logreg: 0.897

Comments:

The accuracy is very close and the same as in the baseline and thus the extra number of iterations had no influence as the model was already well fitted at a lower amount of iterations.

a) **Experiment 4: Decreased hidden layers**

Parameters:

Classes: 1
N: none
Hidden_layer_sizes: 2
Max_iter: 200

Results:

2-class| Us: 0.997; Sklearn: 0.998 Logreg: 0.999

Comments:

A decrease in the number of hidden layers had no significant effect on the accuracy of the trained model.

b) **Experiment 4: Decreased hidden layers**

Parameters:

Sklearn_kwargs:
Hidden_layer_sizes: 16
Max_iter: 2000

Results:

10-class| Sklearn: 0.905; Logreg: 0.897

Comments:

A decrease in the hidden layers decreased Sklearn's accuracy because there were less layers to learn to end at an accurate prediction.

a) **Experiment 5: Increased hidden layers**

Parameters:

Classes: 1
N: none
Hidden_layer_sizes: 32
Max_iter: 200

Results:

2-class| Us: 0.998; Sklearn: 0.998 Logreg: 0.999

Comments:

An increase in the number of hidden layers had no effect on the accuracy of the trained model.

b) **Experiment 5: Increased hidden layers**

Parameters:

Sklearn_kwargs:
Hidden_layer_sizes: 120
Max_iter: 2000

Results:

10-class| Sklearn: 0.956; Logreg: 0.897

Comments:

By almost doubling the hidden layers and thus increasing the running time we have no significant increase in accuracy and thus the base case of 64 layers was a good starting point.

a) **Experiment 6: Change n -> 10**

Parameters:

Classes: 1
N: 10
Hidden_layer_sizes: 8
Max_iter: 200

Results:

2-class| Us: 1.000; Sklearn: 0.500 Logreg: 1.000

Comments:

With such a small example batch the function learns a very specific model and the accuracy output data is unreliable. Sklearn has a very bad accuracy which I think is a more trustworthy number than Logreg and Us. I think the Sklearn is a better algorithm and takes into account the amount of examples and gives value to that when computing the accuracy.

a) **Experiment 7: Change n -> 200**

Parameters:

Classes: 1
N: 200
Hidden_layer_sizes: 8
Max_iter: 200

Results:

2-class| Us: 0.958; Sklearn: 0.917 Logreg: 1.000

Comments:

200 examples is getting to a better batch size as we can see that the accuracy starts to approximate the baseline case where all the examples where used. The accuracy is still lower because we are not using all the available data examples.

a) **Experiment 8: Change class -> 6,9**

Parameters:

Classes: 6,9
N: None
Hidden_layer_sizes: 8
Max_iter: 200

Results:
2-class| Us: 0.995; Sklearn: 0.995 Logreg: 0.998

Comments:
For very different shaped digits the accuracy increases.

a) **Experiment 9: Change class -> 0,8**

Parameters:
Classes: 0,8
N: None
Hidden_layer_sizes: 8
Max_iter: 200

Results:
2-class| Us: 0.988; Sklearn: 0.990 Logreg: 0.987 - 0,8

Comments:
For similar shaped digits accuracy decreases.

Discussion:

Yes the result matches up with what was discussed in class.

For the two class what surprised me was that when we decreased the number of examples (n) then both the Us (Our model) model and the Logreg model displayed an accuracy of 1.00 whereas the Sklearn displayed an accuracy of 0.5. This was in experiment 6. With such a small set of examples our model will overfit and thus it will not generalize well. My thoughts is that because Sklearn is so widely used it is more developed and thus takes into account the small set of examples and thus does not give a false sense of 1.0 accuracy as with our model.

For the ten class what surprised me was that by doubling the max_iterations Sklearn did not increase its accuracy, but by increasing the hidden layers and keeping the max iterations as the base case the accuracy increased. Thus more hidden layers helped to learn new connections and increase the accuracy.

c) We would need to change the architecture of our code to handle 10 input elements (0-9) rather than just handling two input numbers as it currently does.