

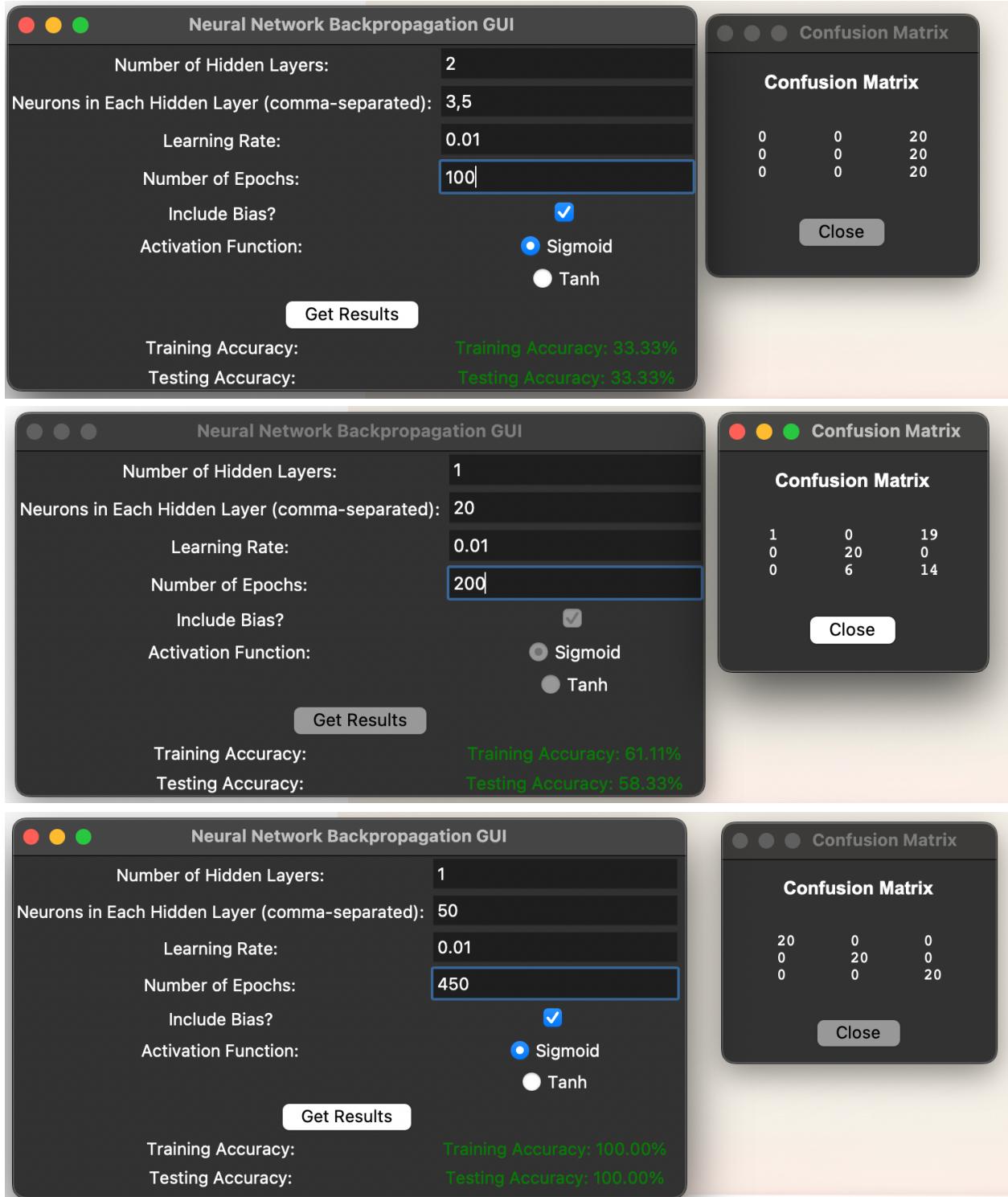
Neural Network Task 2

SC_3

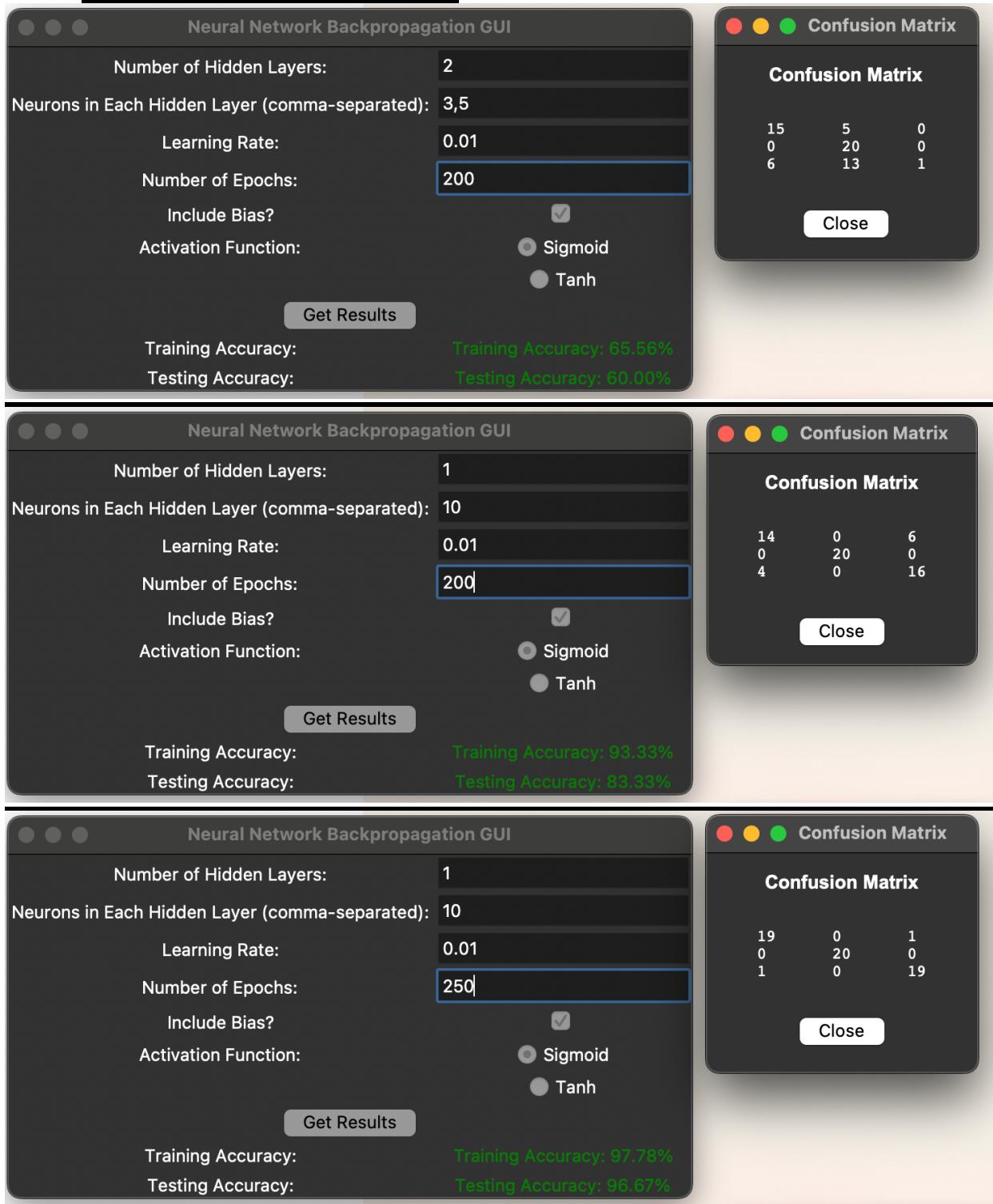
No	Name	ID
1	حبيبة علاء الدين عادل الصناديدي	2021170161
2	مارك جمال انور فهمي	2021170423
3	ملك رأفت فؤاد فاضل	2021170538
4	رنا وحيد تمام عثمان	2021170195
5	مايكيل نادر عماد الدين	2021170442
6	ماجد سيد محمود	2021170421

Sigmoid Function

- Init Random weights:



- Xavier initialization:



Conclusion:

- Using random initialization of weights with the sigmoid activation function can lead to the vanishing gradient problem, particularly with multiple layers. While good results can still be achieved by increasing the number of epochs, this approach is not ideal due to the extended training time.
- On the other hand, Xavier initialization performs well regardless of the number of layers, neurons, or epochs. It effectively mitigates the vanishing gradient problem, enabling better results without the need for excessively long training periods.

Tanh

- Init Random weights:

The image displays three separate instances of the Neural Network Backpropagation GUI, each accompanied by its own Confusion Matrix window.

Top Instance (3 Hidden Layers):

- Number of Hidden Layers: 3
- Neurons in Each Hidden Layer (comma-separated): 10,20,30
- Learning Rate: 0.01
- Number of Epochs: 200
- Include Bias?
- Activation Function: Sigmoid Tanh

Training Accuracy: 98.89% Testing Accuracy: 98.33%

Middle Instance (1 Hidden Layer):

- Number of Hidden Layers: 1
- Neurons in Each Hidden Layer (comma-separated): 10
- Learning Rate: 0.01
- Number of Epochs: 150
- Include Bias?
- Activation Function: Sigmoid Tanh

Training Accuracy: 97.78% Testing Accuracy: 100.00%

Bottom Instance (1 Hidden Layer):

- Number of Hidden Layers: 1
- Neurons in Each Hidden Layer (comma-separated): 20
- Learning Rate: 0.01
- Number of Epochs: 200
- Include Bias?
- Activation Function: Sigmoid Tanh

Training Accuracy: 98.89% Testing Accuracy: 96.67%

Confusion Matrix Results (All Instances):

- Top Instance Confusion Matrix:**

	0	1
0	19	0
1	0	20
Total	19	20
- Middle Instance Confusion Matrix:**

	0	1
0	20	0
1	0	20
Total	20	20
- Bottom Instance Confusion Matrix:**

	0	1
0	18	0
1	0	20
Total	18	20

- Xavier initialization:

The figure displays three separate runs of the Neural Network Backpropagation GUI, each followed by its corresponding Confusion Matrix window.

Run 1 (Top):

- Number of Hidden Layers: 3
- Neurons in Each Hidden Layer (comma-separated): 10,20,30
- Learning Rate: 0.01
- Number of Epochs: 100
- Include Bias?
- Activation Function: Sigmoid (radio button)

Training Accuracy: 97.78%
Testing Accuracy: 95.00%

Confusion Matrix		
17	0	3
0	20	0
0	0	20

Run 2 (Middle):

- Number of Hidden Layers: 1
- Neurons in Each Hidden Layer (comma-separated): 20
- Learning Rate: 0.01
- Number of Epochs: 200
- Include Bias?
- Activation Function: Tanh (radio button)

Training Accuracy: 98.89%
Testing Accuracy: 100.00%

Confusion Matrix		
20	0	0
0	20	0
0	0	20

Run 3 (Bottom):

- Number of Hidden Layers: 1
- Neurons in Each Hidden Layer (comma-separated): 20
- Learning Rate: 0.01
- Number of Epochs: 50
- Include Bias?
- Activation Function: Tanh (radio button)

Training Accuracy: 98.89%
Testing Accuracy: 96.67%

Confusion Matrix		
20	0	0
0	19	1
1	0	19

Conclusion:

For the tanh activation function, the accuracy does not significantly differ with different weight initialization methods. However, using Xavier initialization can still lead to higher accuracy with fewer epochs compared to random initialization, making the training process more efficient.