Q1.

1. 1 Hidden Layer and 3 Nodes with the TanH activation function and squared penalty:

It took 20 seconds to run the model.

Weights: 15\*3 + 3\*1 = 45+3 = 48

Biases: 3 + 1 = 4

The total parameters are 48 + 4 = 52

Validation AUC: 0.6930

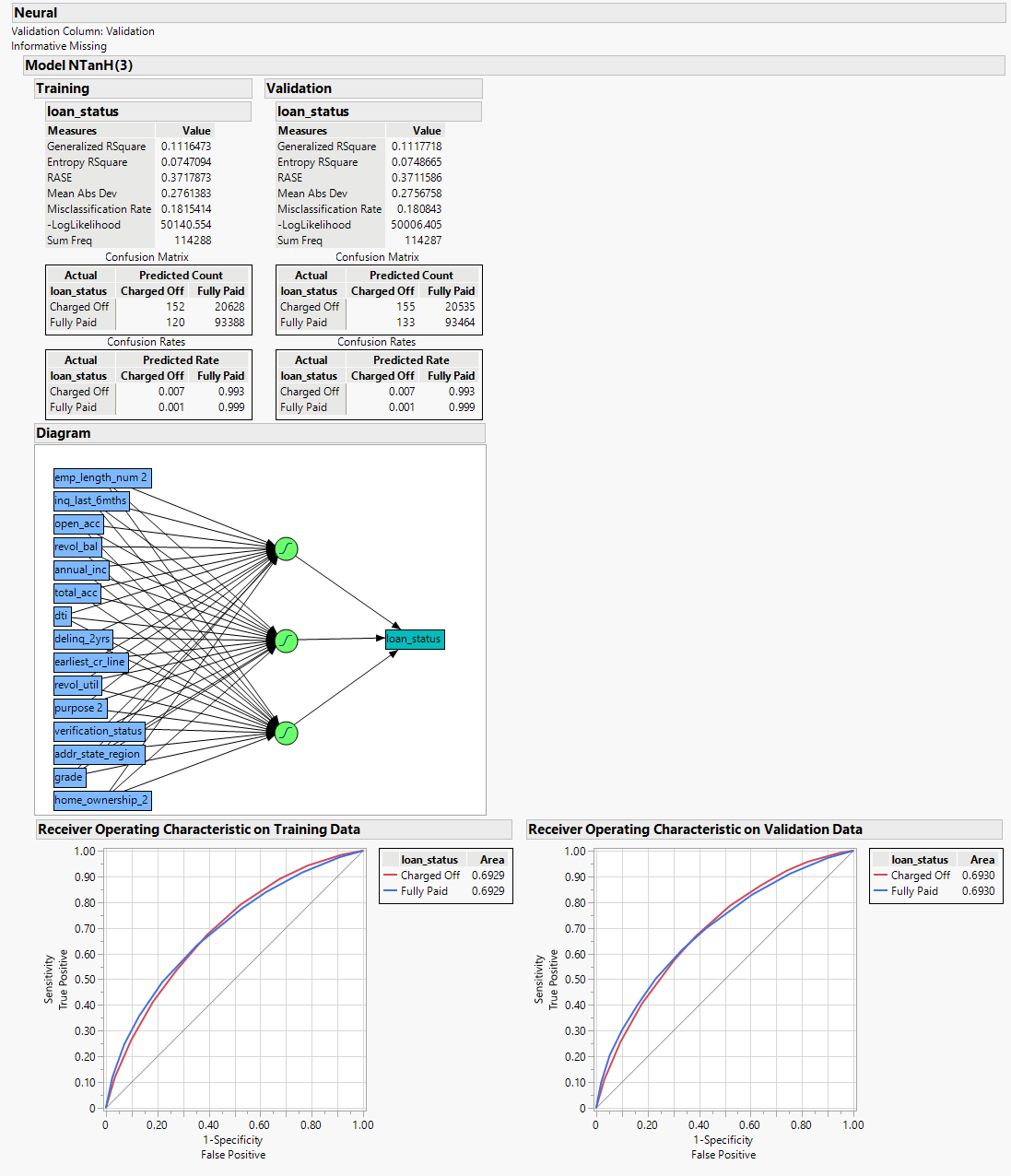


Fig 1. Neural Model with 1 Hidden Layer and 3 Nodes

1. 1 Hidden Layer and 6 Nodes with the TanH activation function and squared penalty:

It took 32 seconds to run the model.

Weights: 15\*6 + 6\*1 = 90+6 = 96

Biases: 6 + 1 = 7

The total parameters are 96 + 7 = 103

Validation AUC: 0.6911

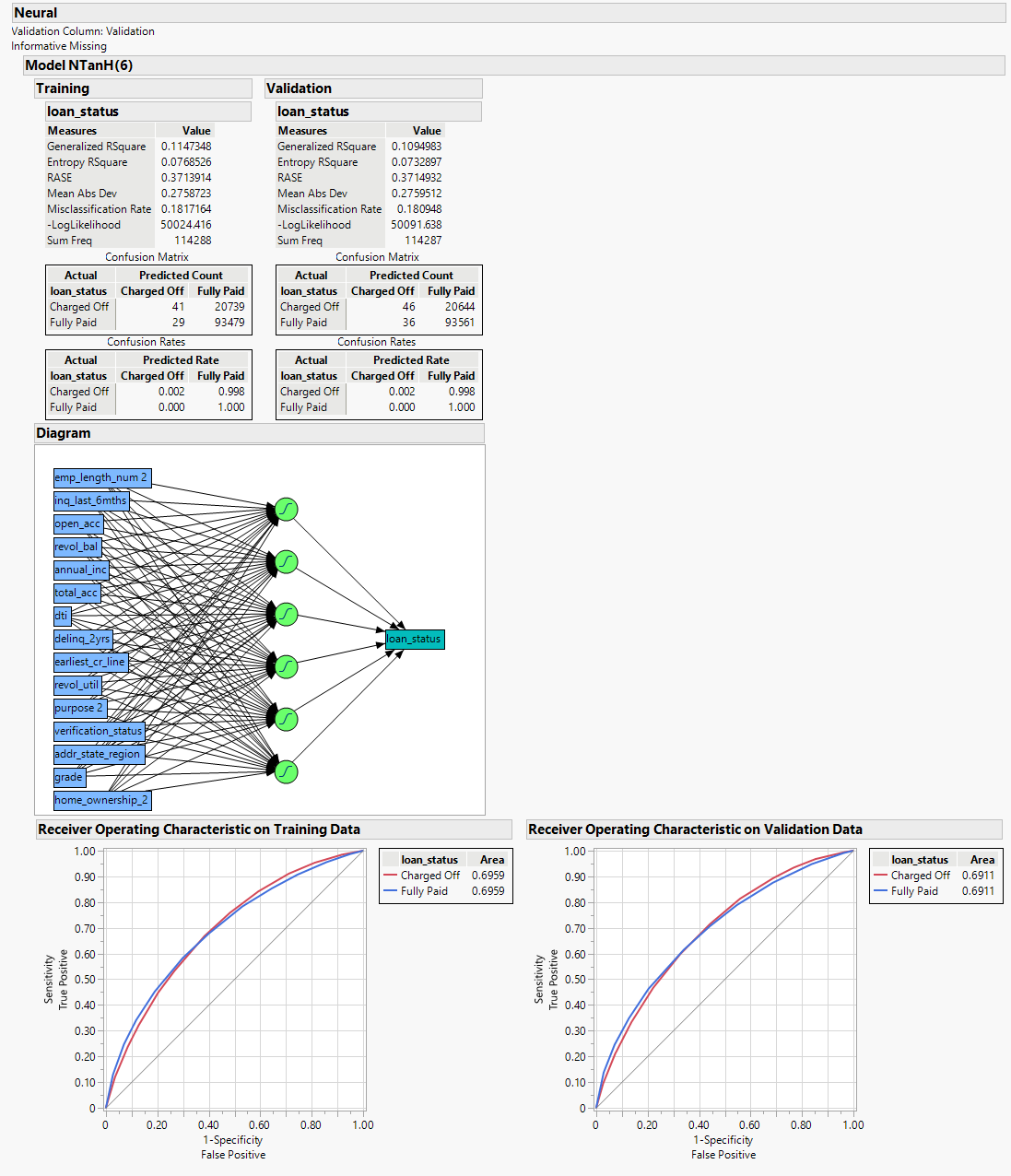


Fig 2. Neural Model with 1 Hidden Layer and 6 Nodes

1. 2 Hidden Layer and 3 Nodes each with the TanH activation function and squared penalty:

It took 40 seconds to run the model.

Weights: 15\*3 + 3\*3 + 3\*1 = 45+9+3 = 57

Biases: 3 + 3 + 1 = 7

The total parameters are 57 + 7 = 64

Validation AUC: 0.6929

A screenshot of a computer

Description automatically generated

Fig 3. Neural Model with 2 Hidden Layers and 3 Nodes each

1. 2 Hidden Layer and 9 Nodes each with the TanH, Linear, and Gaussian activation function, and squared penalty:

It took 1 minute 42 seconds to run the model.

Weights: 15\*27 + 27\*27 + 27\*1 = 405+729+27 = 1161

Biases: 27 + 27 + 1 = 55

The total parameters are 1161 + 55 = 1216

Validation AUC: 0.6943

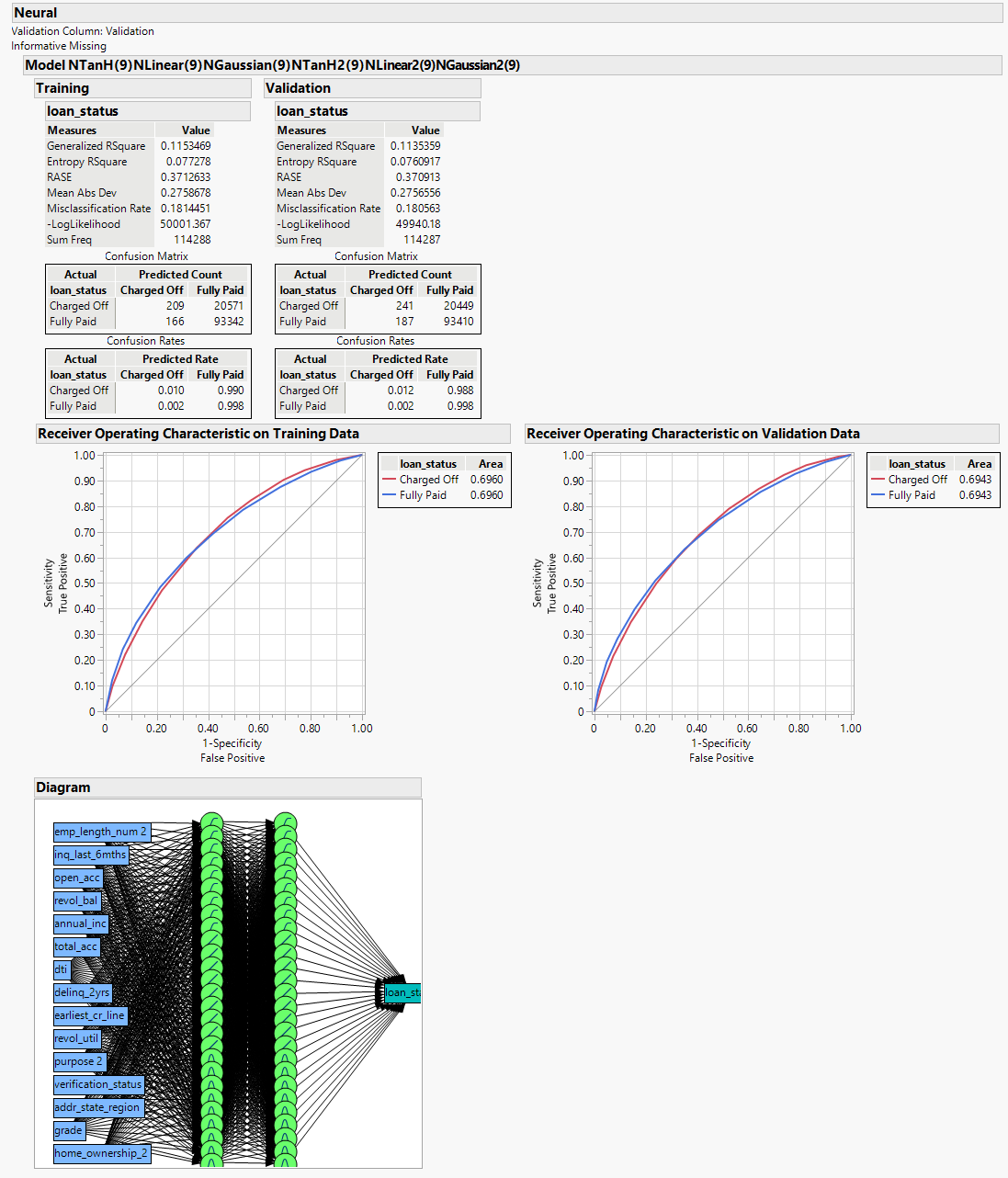


Fig 4. Neural Model with 2 Hidden Layers and 9 Nodes each with 3 activation functions

Model Comparison:

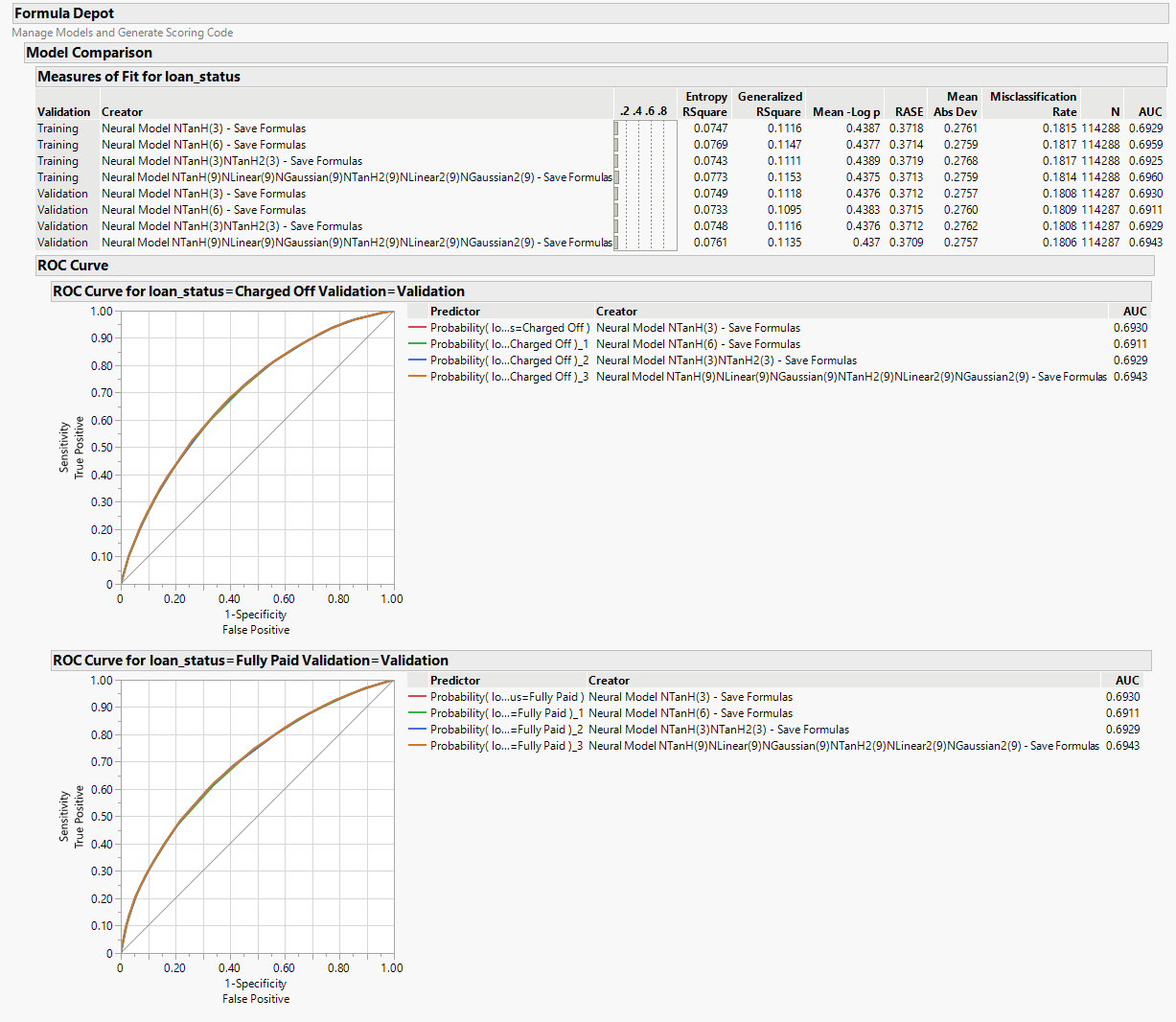


Fig 5. Model Comparison

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Layers and Nodes** | **AUC** | **Entropy R2** | **Generalized R2** | **RMSE** | **Time Taken** |
| Final Logistic Linear Regression Model | 0.6926 | 0.0745 | 0.1113 | 0.3712 |  |
| 1 Layer 3 Nodes Each | 0.6930 | 0.0749 | 0.1118 | 0.3712 | 20 Seconds |
| 1 Layer 6 Nodes Each | 0.6911 | 0.0733 | 0.1095 | 0.3715 | 32 Seconds |
| 2 Layers 3 Nodes Each | 0.6929 | 0.0748 | 0.1116 | 0.3712 | 40 Seconds |
| 2 Layers 9 Nodes Each for TanH, Linear, Gaussian Activation Function | 0.6943 | 0.0761 | 0.1135 | 0.3709 | 1 Min 42 Seconds |

The above table expresses all the major differences we look at in the models. Overall, the model with 2 layers and 9 nodes each for TanH, Linear, and Gaussian Activation Functions works best with AUC at 0.6943 and RMSE at the lowest at 0.3709. But at the same time, it took the longest to run (1 Min 42 Seconds) with the complexity of 1216 parameters, 1161 Weights, and 55 Biases. The 1st model with 1 layer and 3 nodes worked 2nd best after the model with 2 layers and 9 nodes for each activation function, but at the same time, it took the least time (20 seconds) to run the model with the least complexity of 48 Weights, 4 Biases, and 52 Parameters. Overall, as the complexity increases, the accuracy of the model also increases, but with a huge toll at times. So, it depends on the case, whether time is important or accuracy, for instance, in the health sector, accuracy would be more important as compared to time.

The model with 2 layers and 9 nodes for each activation function has an AUC of 0.6942 with an RMSE of 0.3709 and the final logistic regression model from homework 5 has an AUC of 0.6926 with an RMSE of 0.3712. As the logistic regression model is supervised data, I could control the parameters but in Neural network, I cannot control the parameters. I could see all the changes in realtime as I was making them, but in neural model as it is a black box, I’m not able to see the realtime changes which are going on in the background of the model.