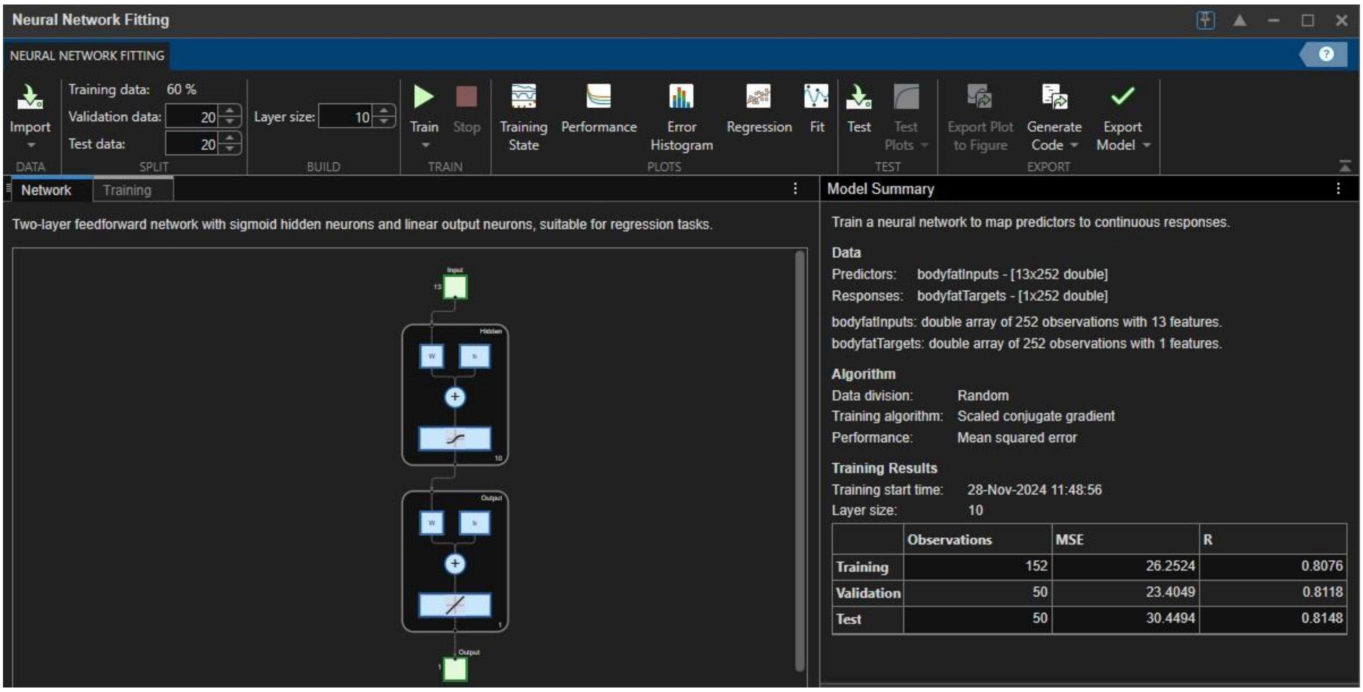
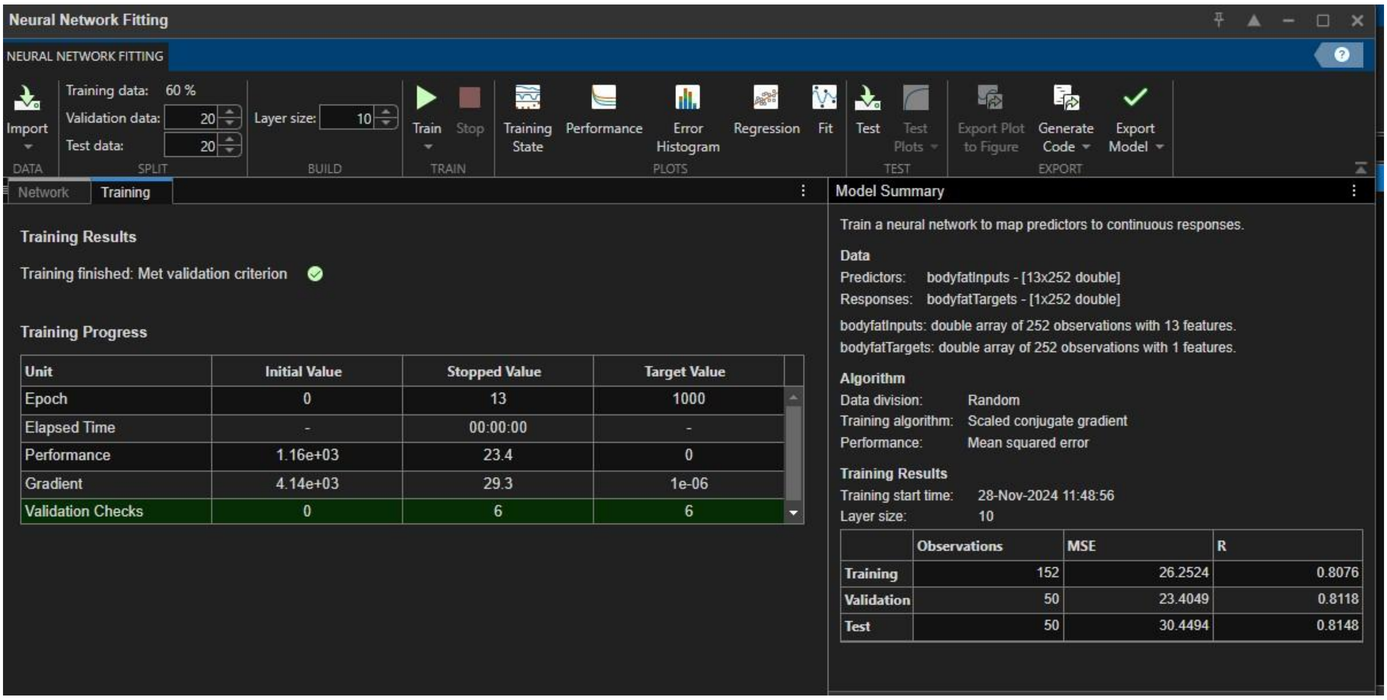
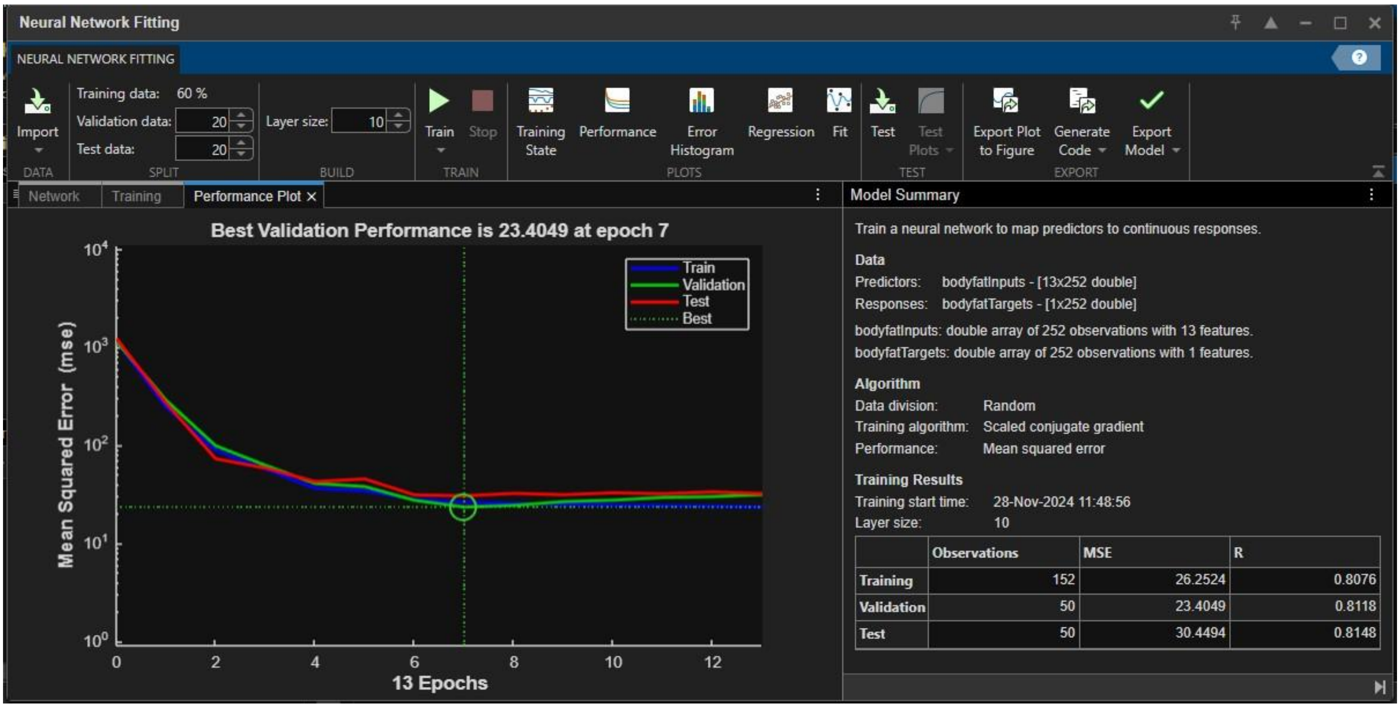
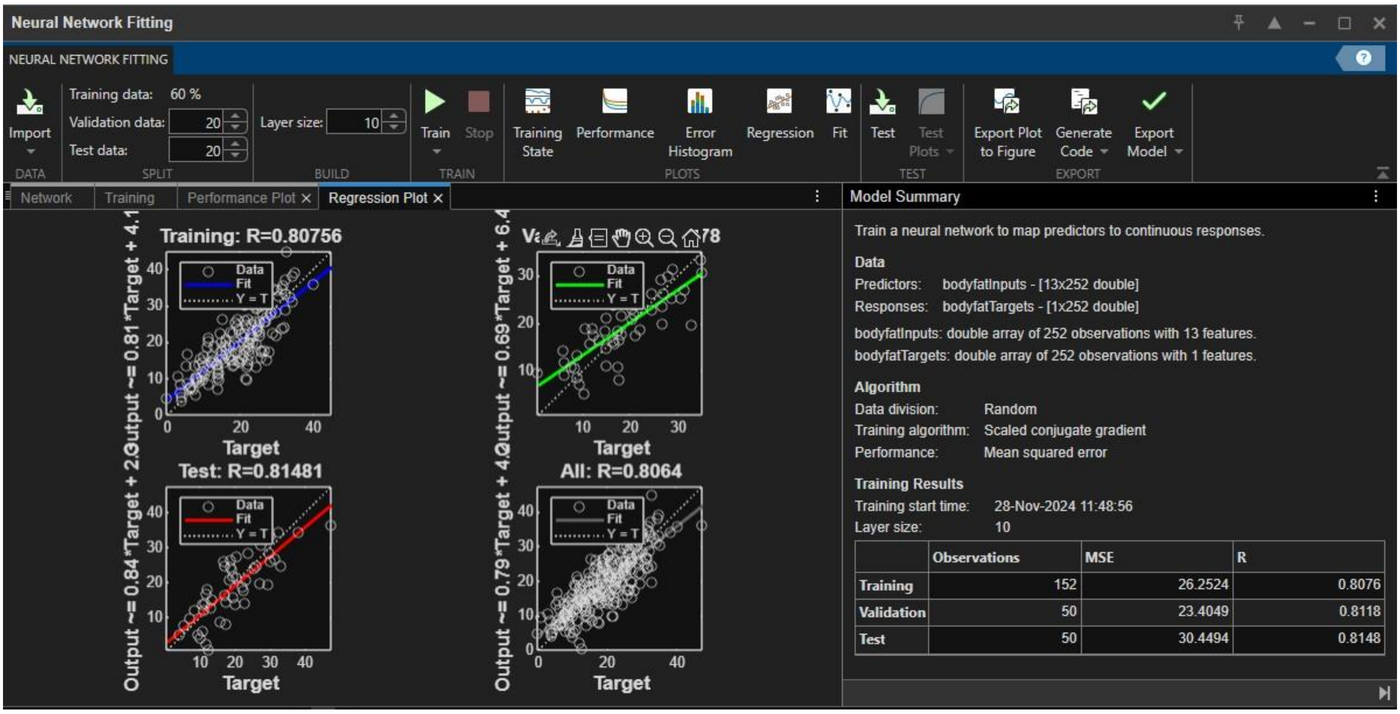
Project : Mini Project using Body fat dataset

Part 1: Neural Net5work Fitting
Experiment: Import Body Fat Data Set
Configuration no.1:
Training data = 60%, Validation data = 20%, Testing data = 20% and Layer Size = 15
Results:



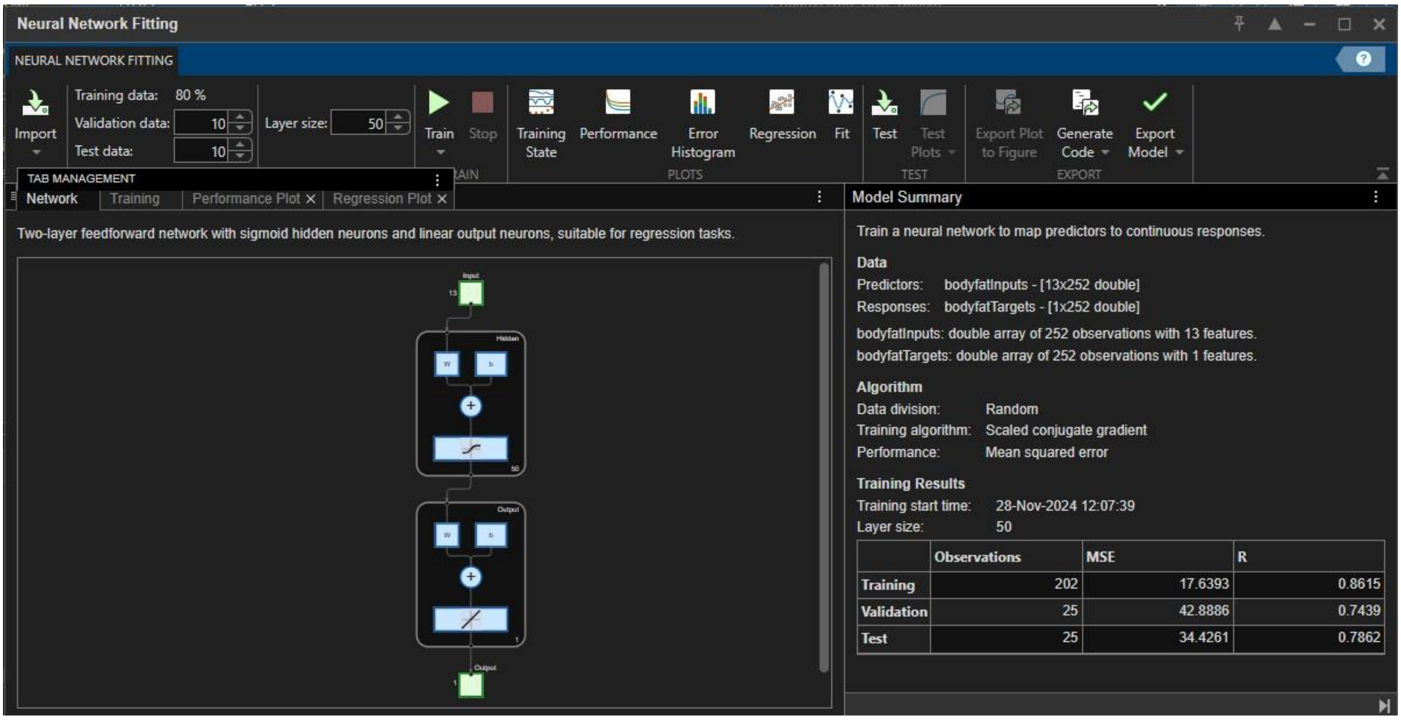


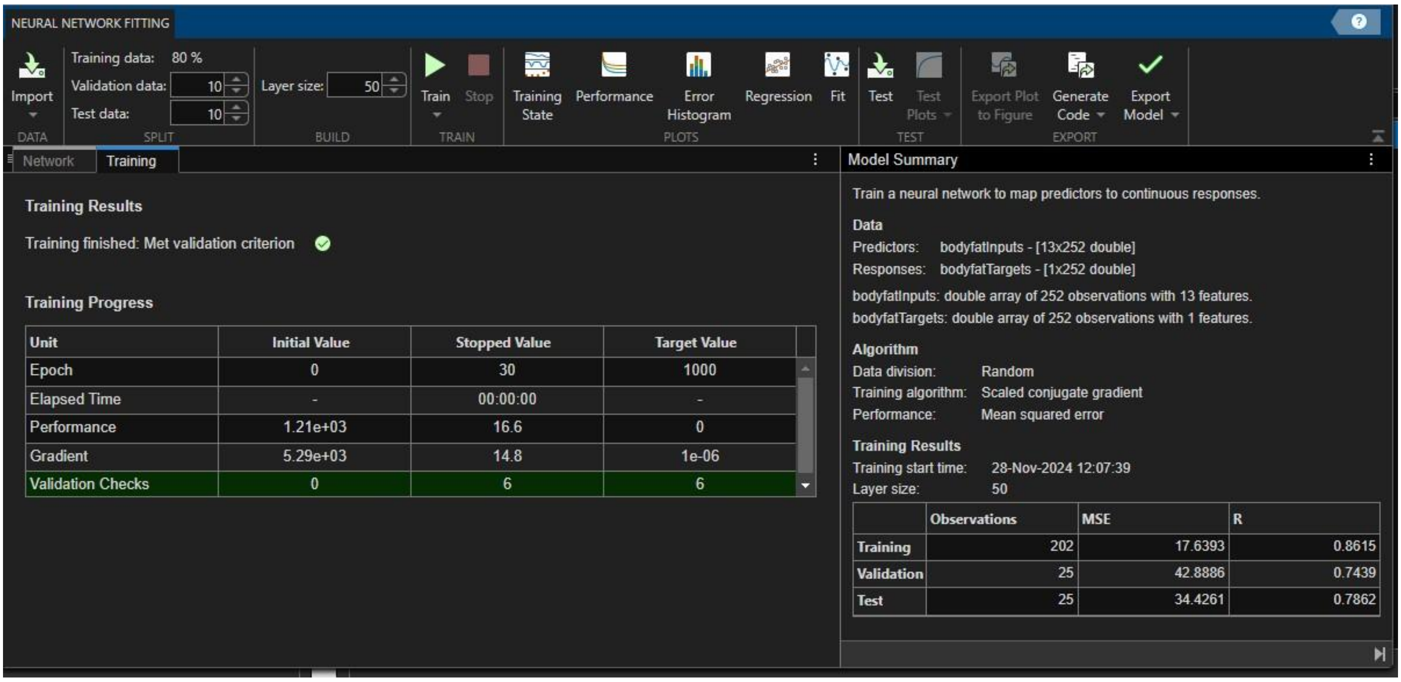


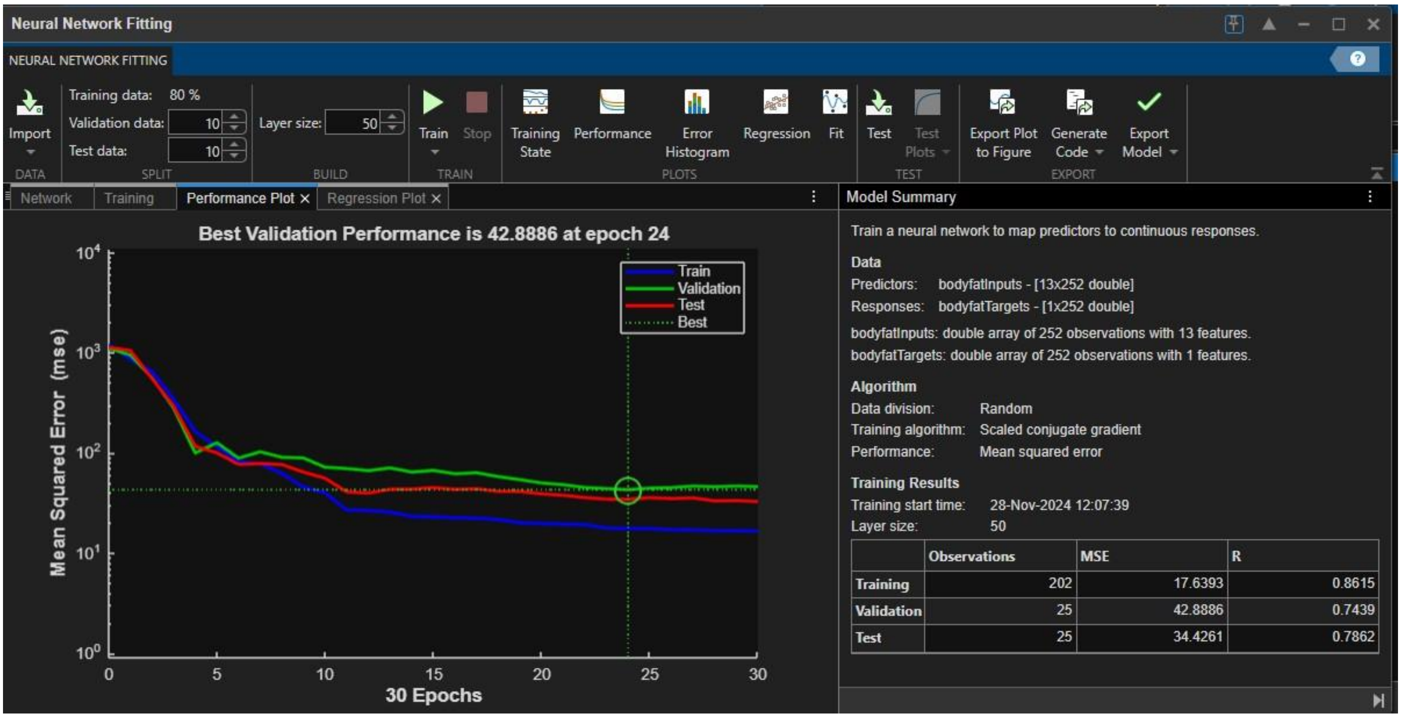


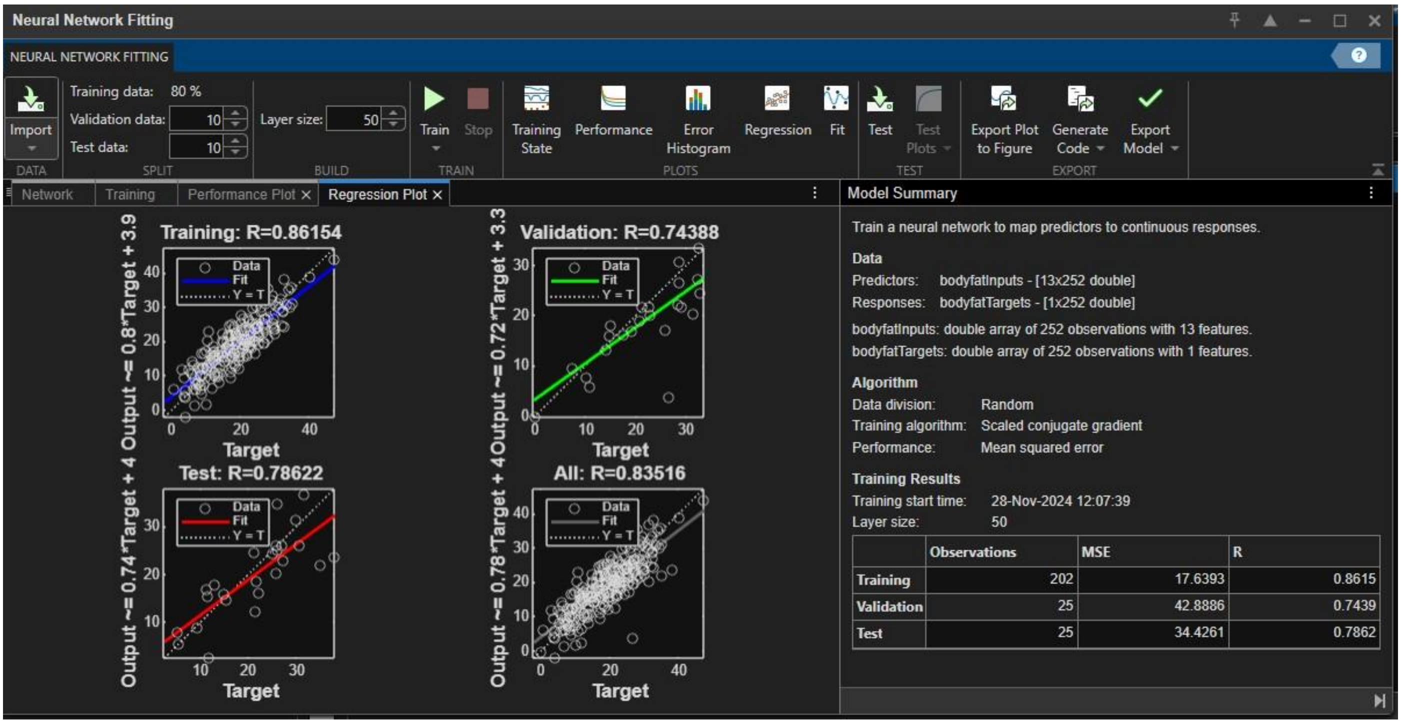
Configuration no.2:

Training data = 80%, Validation data = 10%, Testing data = 10% and Layer Size = 50


Neural Network Fitting
NEURAL NETWORK FITTING
Import
Training data: 80 %
Validation data: 10
Test data: 10
Layer size: 50
Train
Stop
Training State
Performance
Error Histogram
Regression
Fit
Test
Test Plots
Export Plot to Figure
Generate Code
Export Model
TAB MANAGEMENT
PLOTS
TEST
EXPORT
Network
Training
Performance Plot
Regression Plot
Two-layer feedforward network with sigmoid hidden neurons and linear output neurons, suitable for regression tasks.
Model Summary
Train a neural network to map predictors to continuous responses.
Data
Predictors: bodyfatInputs - [13x252 double]
Responses: bodyfatTargets - [1x252 double]
bodyfatInputs: double array of 252 observations with 13 features.
bodyfatTargets: double array of 252 observations with 1 features.
Algorithm
Data division: Random
Training algorithm: Scaled conjugate gradient
Performance: Mean squared error
Training Results
Training start time: 28-Nov-2024 12:07:39
Layer size: 50
Observations
MSE
R
Training 202 17.6393 0.8615
Validation 25 42.8886 0.7439
Test 25 34.4261 0.7862


NEURAL NETWORK FITTING
Import
Training data: 80 %
Validation data: 10
Test data: 10
Layer size: 50
Train
Stop
Training State
Performance
Error Histogram
Regression
Fit
Test
Test Plots
Export Plot to Figure
Generate Code
Export Model
DATA
SPLIT
BUILD
TRAIN
PLOTS
TEST
EXPORT
Network
Training
Training Results
Training finished: Met validation criterion
Training Progress
Unit
Initial Value
Stopped Value
Target Value
Epoch 0 30 1000
Elapsed Time - 00:00:00 -
Performance 1.21e+03 16.6 0
Gradient 5.29e+03 14.8 1e-06
Validation Checks 0 6 6
Model Summary
Train a neural network to map predictors to continuous responses.
Data
Predictors: bodyfatInputs - [13x252 double]
Responses: bodyfatTargets - [1x252 double]
bodyfatInputs: double array of 252 observations with 13 features.
bodyfatTargets: double array of 252 observations with 1 features.
Algorithm
Data division: Random
Training algorithm: Scaled conjugate gradient
Performance: Mean squared error
Training Results
Training start time: 28-Nov-2024 12:07:39
Layer size: 50
Observations
MSE
R
Training 202 17.6393 0.8615
Validation 25 42.8886 0.7439
Test 25 34.4261 0.7862





Conclusion:

The neural network with a layer size of 15 and the given data split (60-20-20) performed well. The model achieved good regression values (R > 0.8) across training, validation, and testing datasets, demonstrating a reliable predictive capability. The validation and test errors were close to the training error, showing that the model generalizes well without significant overfitting.

Part 2:

Tasks Performed

1. **Experiment 1: 15-node MLP (training ratio = 70%, validation ratio = 15% and test data ratio = 15%)**
    - A single hidden layer neural network with 15 nodes was trained.
    - Data was split into 70% for training, 15% for validation, and 15% for testing.
    - The network was retrained 10 times with random initializations and mean and variance of the MSEs were calculated.

2. **Experiment 2: 2-node MLP (training ratio = 30%, validation ratio = 20% and test data ratio = 50%)**

    - A single hidden layer neural network with 2 nodes was trained to test the performance of a reducedcapacity model.
    - The dataset was split into 30% for training, 20% for validation, and 50% for testing.
    - The model was retrained 10 times, each time with random initializations to evaluate consistency and stability.
    - The mean and variance of the MSEs were calculated for training, validation, and testing datasets.

3. **Experiment 3: 80-node MLP (training ratio = 30%, validation ratio = 20% and test data ratio = 50%)**
    - A single hidden layer neural network with 80 nodes was trained to explore the performance of an increased-capacity model.
    - The dataset was partitioned into 30% for training, 20% for validation, and 50% for testing.
    - The model was retrained 10 times using random weight initializations to account for variability in training.
    - The mean and variance of the MSEs were computed for training, validation, and testing datasets.

4. **Experiment 4: 80-node with Regularization Effects** The 80-node model was tested with:
    - Case – 1: Regularization parameter = 0.1.
    - Case – 2: Regularization parameter = 0.5.
    - Each case involved retraining 10 times, and mean and variance of MSEs for training, validation, and testing were calculated.

Results Summary:

The table below summarizes the mean and variance of MSEs for all configurations:

Configuration	Train MSE Mean	Train MSE Variance	Val MSE Mean	Val MSE Variance
15 Nodes	0.007912	7.2E-06	0.017531	2.38E-05
2 Nodes	0.030587	0.000376	0.022511	0.000134
80 Nodes	0.003331	2.3E-06	0.011092	6.9E-06

80 Nodes (Reg=0.1)	0.055139	0.00039	0.046592	0.000413
80 Nodes (Reg=0.5)	0.093975	0.002116	0.080364	0.001691

Observations:

1. Effect of Hidden Layer Size

a. As the number of hidden nodes increases from 2 to 80, the training and validation MSEs significantly improve.
b. The variance of the MSEs decreases with more hidden nodes, indicating improved stability and consistency in model training.

2. Effect of Regularization

a. With Regularization parameters = 0.1 and 0.5, training MSE increases compared to the unregularized 80-node model.
b. Validation and test MSEs also increase, suggesting that regularization might slightly underfit the model, especially for Regularization parameter = 0.5.
c. Regularization reduces the variance in MSEs, which implies better generalization across different training runs.

3. Comparison Across Configurations

a. The 80-node model with no regularization has the lowest training and validation MSEs but could benefit from a slight regularization to mitigate overfitting on unseen data.
b. Increasing regularization further (Regularization parameter = 0.5) reduces overfitting but sacrifices predictive accuracy.

Conclusion

The experiments show that increasing the hidden layer size improves model accuracy and stability, with the 80-node MLP achieving the lowest MSEs. Regularization reduces overfitting but can lead to underfitting at higher levels. The unregularized 80-node model performs best in terms of accuracy, but slight regularization with 0.1 offers a good balance between accuracy and generalization. The 15-node model is a simpler, efficient alternative, while the 2-node model is insufficient for capturing data complexity.