類神經網路簡介HW2.2

資訊所P76054088 黃子睿

Dataset 1: Iris

Learning algorithm 1. BFGS Quasi-Newton

1. 網路架構及參數設定



Input layer: 4個neuron

兩層 hidden layer

First hidden layer: 10個neuron (Hyperbolic tangent sigmoid transfer function)

Second hidden layer: 10個neuron (Log-sigmoid transfer function)

Output layer: 1個neuron (Linear transfer function)

Learning Rate: 0.002

Epoch: 3000

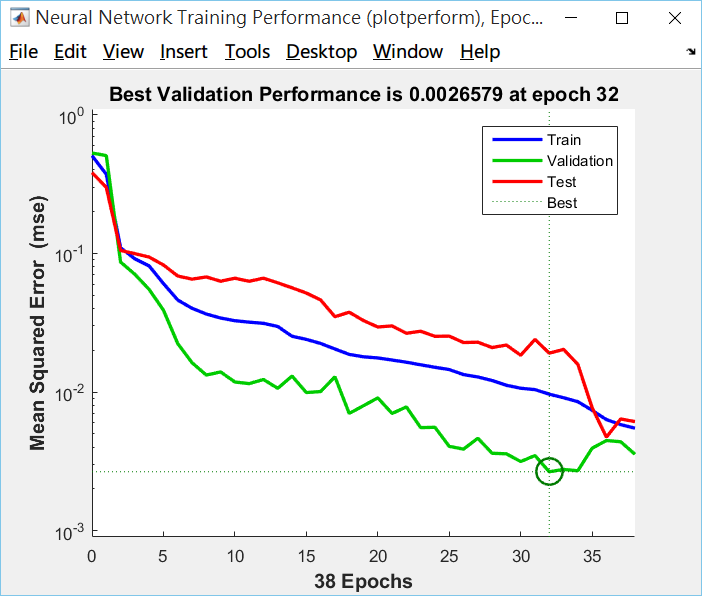
Goal: 0

Show: 25

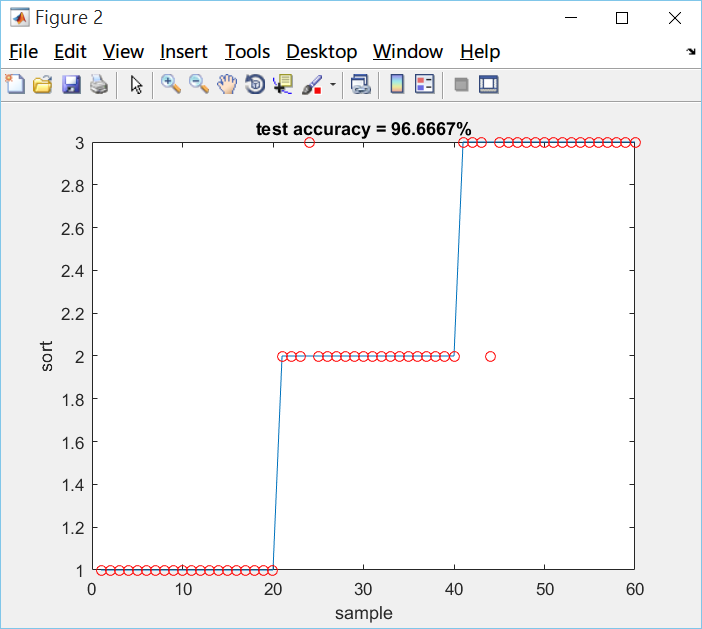
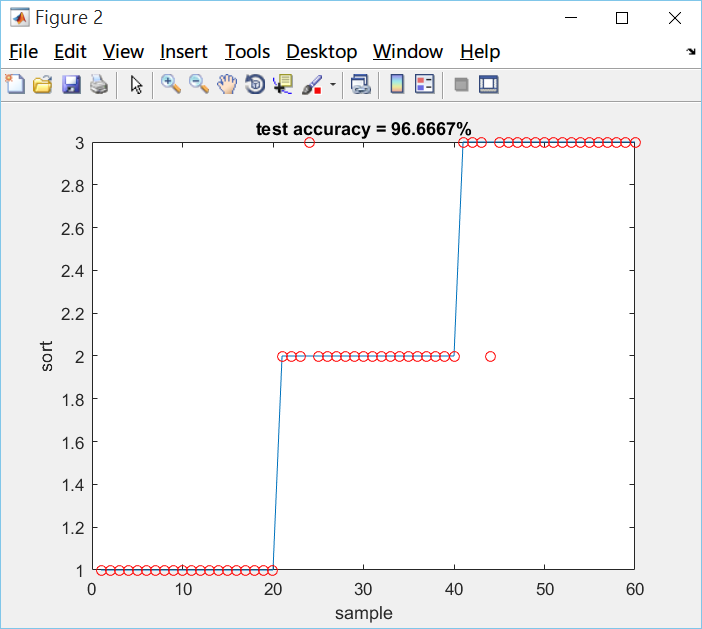
1. Performance(learning curve)
2. 結果圖(需附註辨識率)

第1, 2次結果

Performance

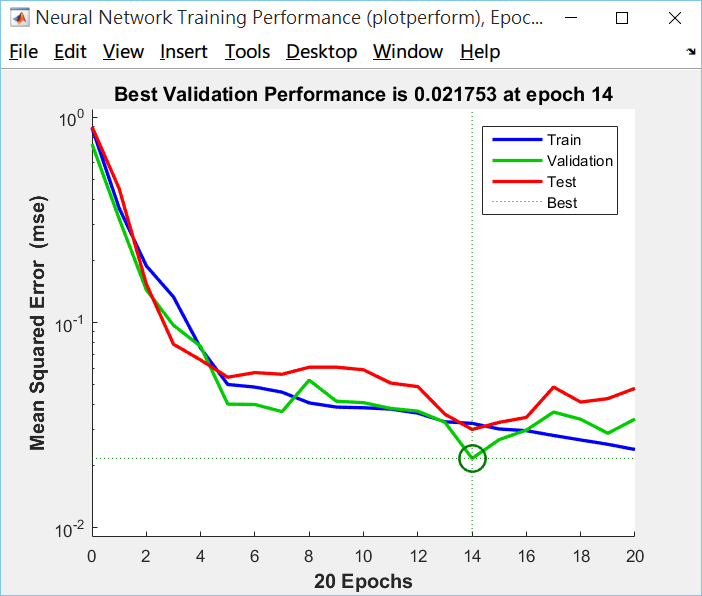
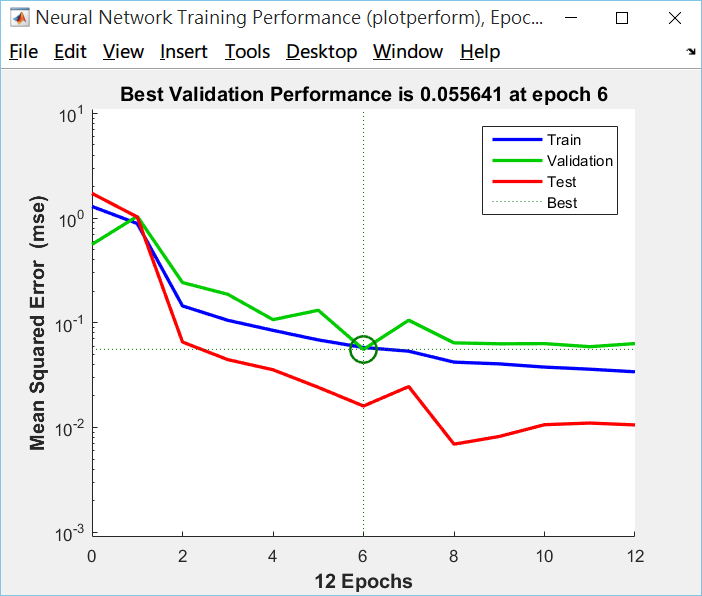


Accuracy(最上方標題部分為辨識成功率)

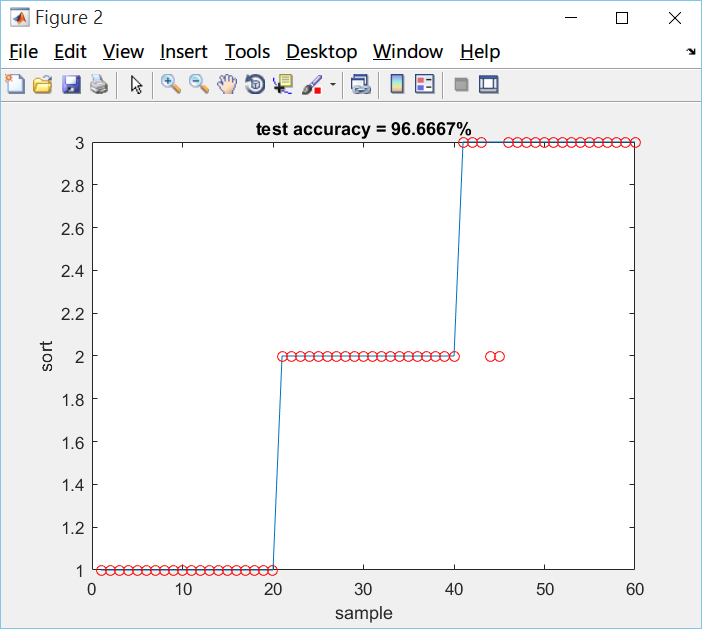
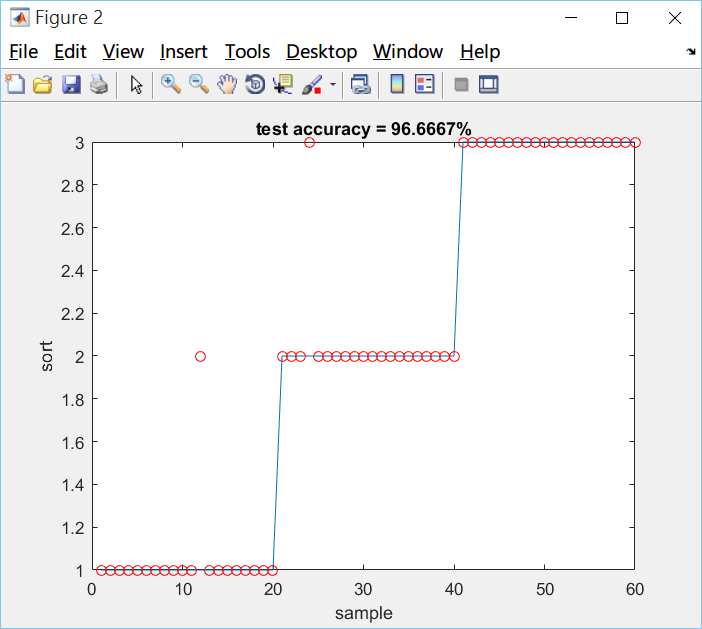


第3, 4次結果

Performance

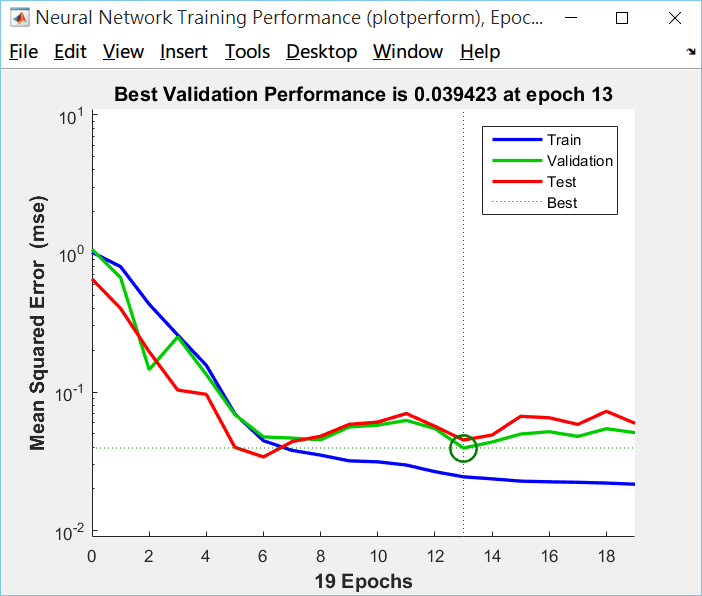


Accuracy(最上方標題部分為辨識成功率)

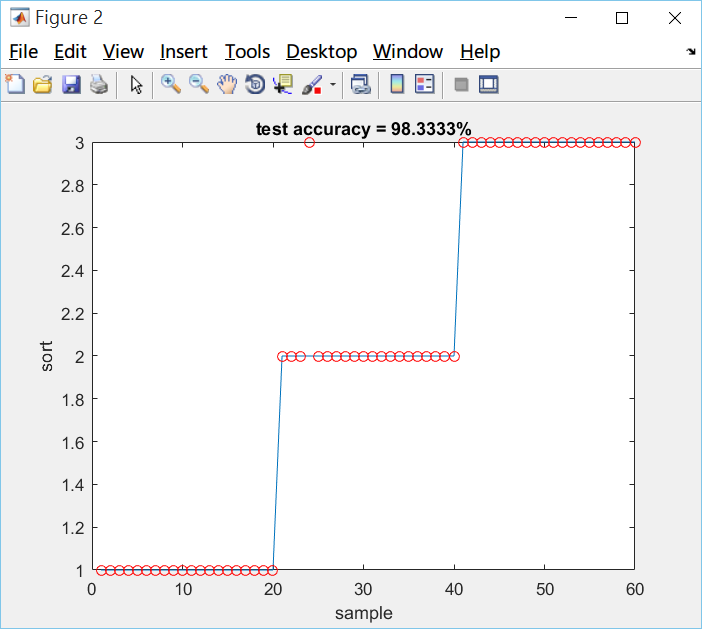


第5次結果

Performance



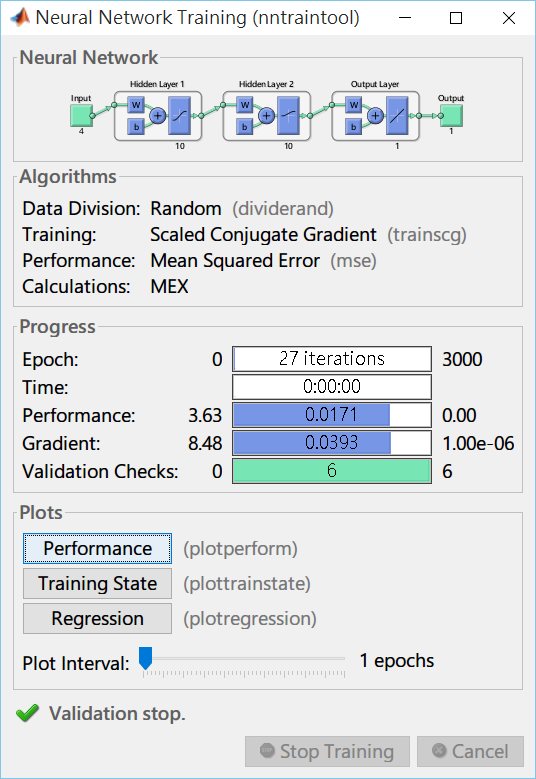
Accuracy(最上方標題部分為辨識成功率)



平均準確率: 97.000% (四捨五入到小數點後三位)

Learning algorithm 2. Scaled Conjugate Gradient

1. 網路架構及參數設定



Input layer: 4個neuron

兩層 hidden layer

First hidden layer: 10個neuron (Hyperbolic tangent sigmoid transfer function)

Second hidden layer: 10個neuron (Log-sigmoid transfer function)

Output layer: 1個neuron (Linear transfer function)

Learning Rate: 0.002

Epoch: 3000

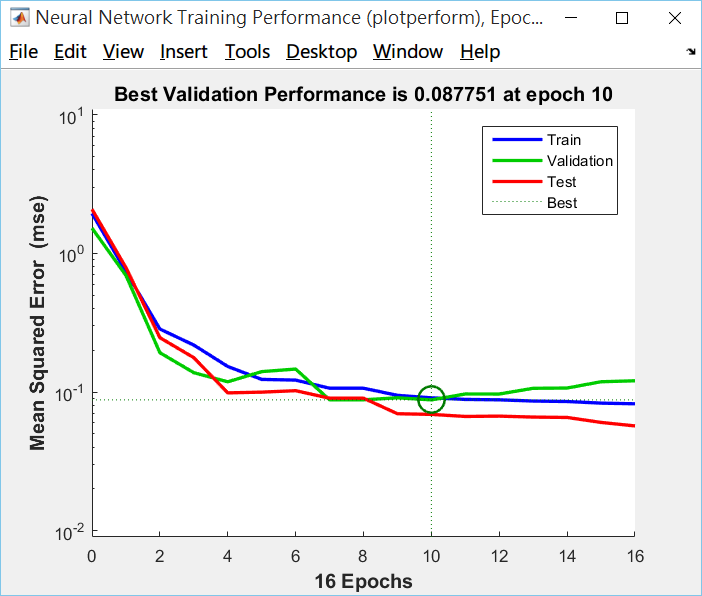
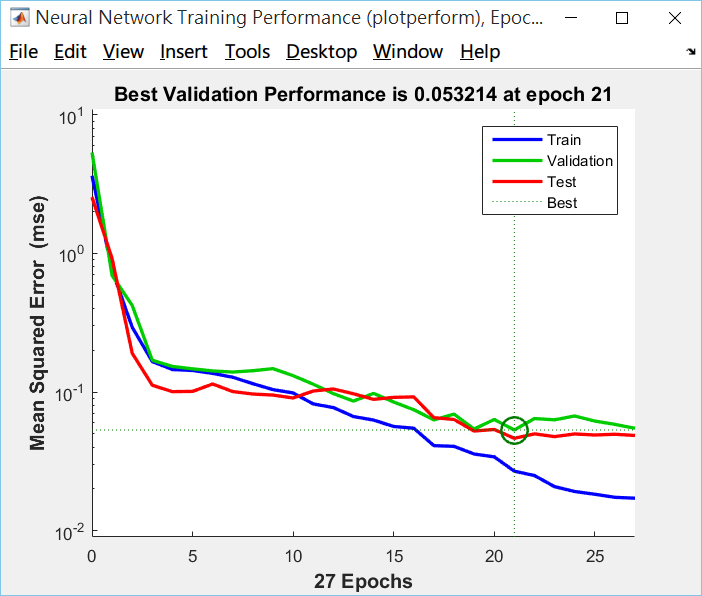
Goal: 0

Show: 25

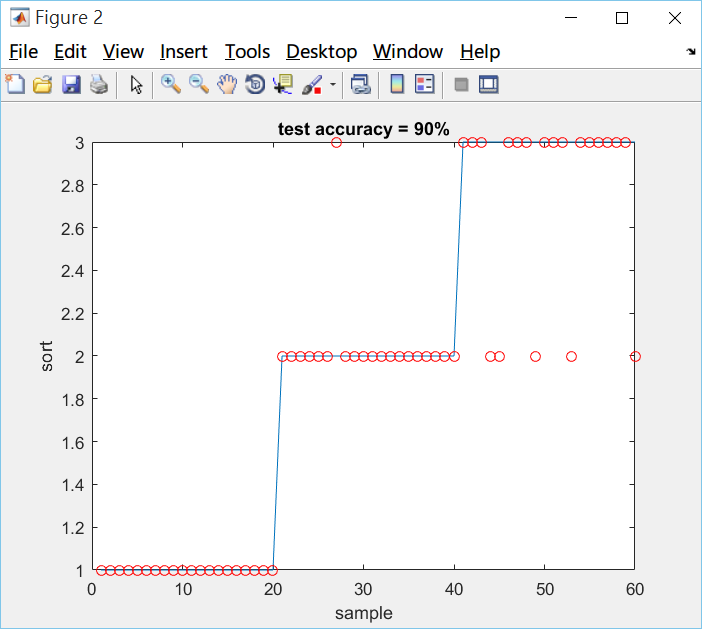
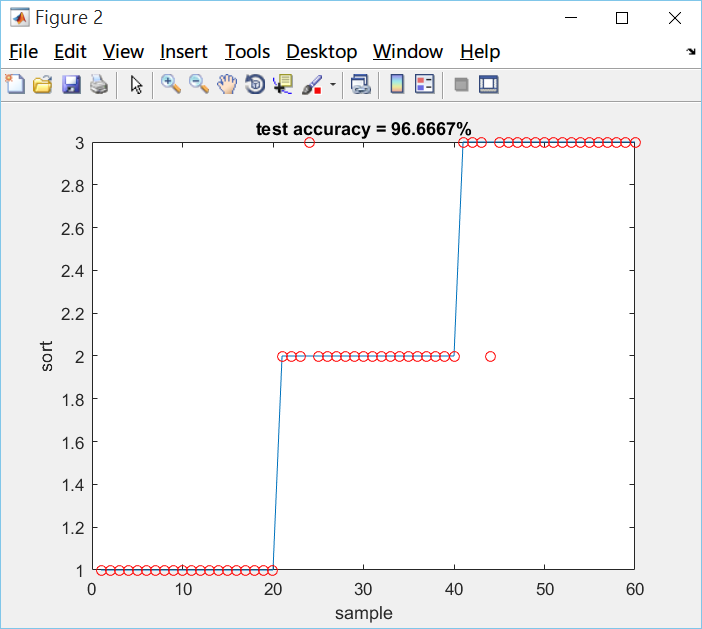
1. Performance(learning curve)
2. 結果圖(需附註辨識率)

第1, 2次結果

Performance

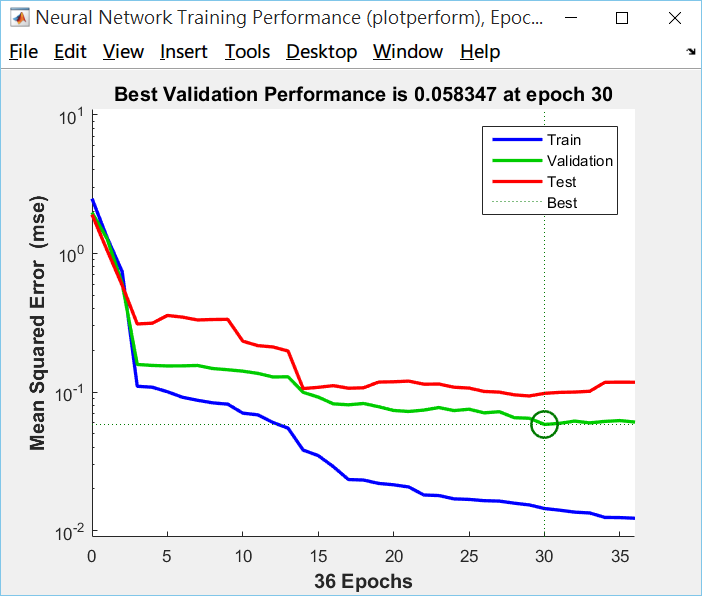
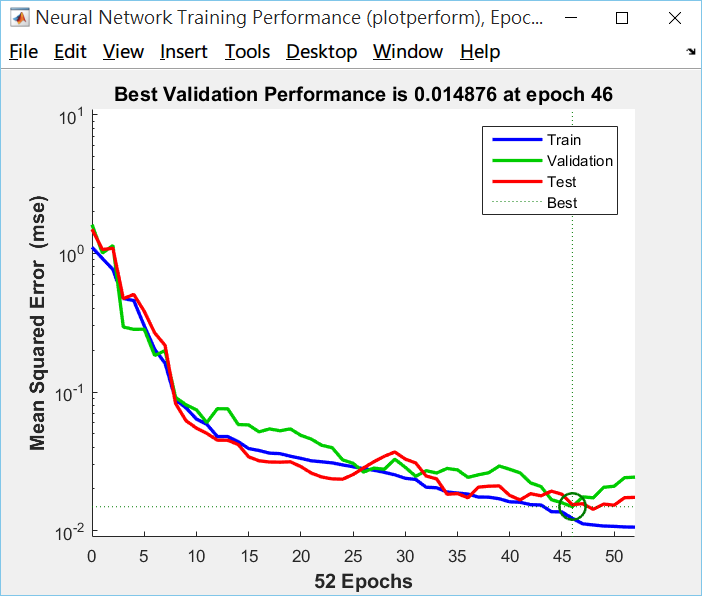


Accuracy(最上方標題部分為辨識成功率)

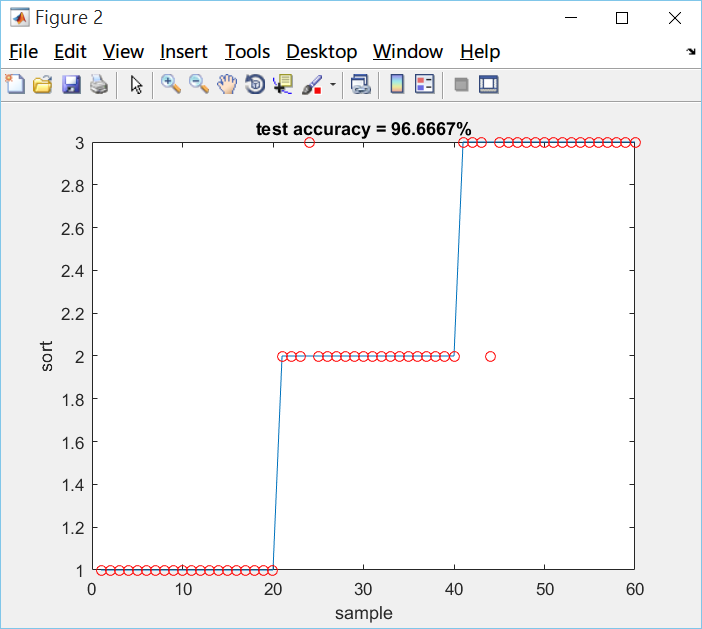
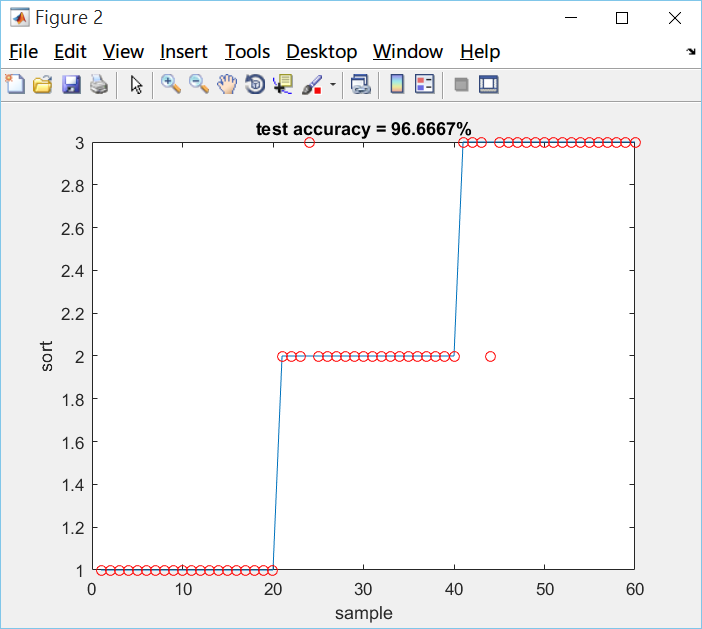


第3, 4次結果

Performance

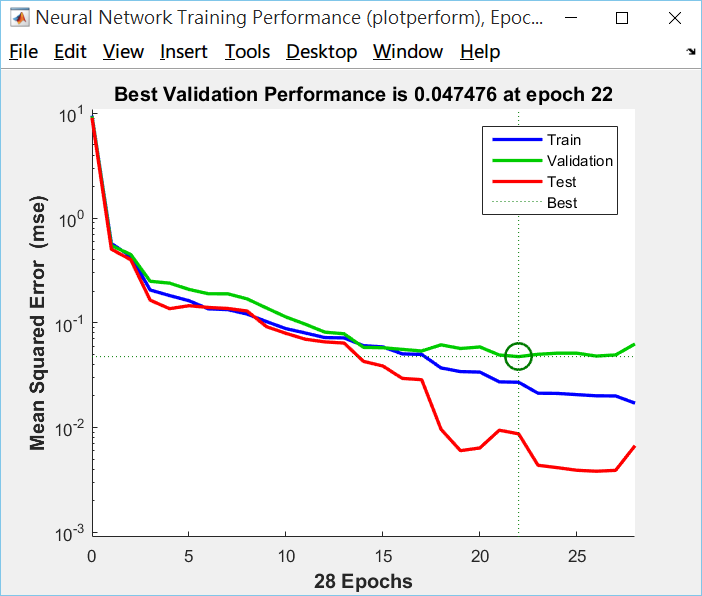


Accuracy(最上方標題部分為辨識成功率)

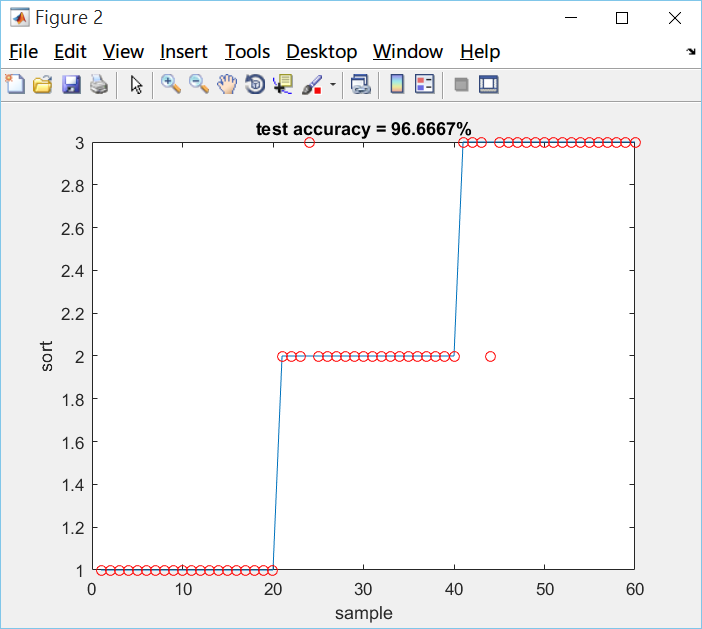


第5次結果

Performance



Accuracy(最上方標題部分為辨識成功率)



平均準確率: 95.333% (四捨五入到小數點後三位)

1. learning algorithms方法差異之討論

從 Quasi-Newton 測出來的結果可以發現，對於第二類的分類總是會有輕微誤判現象，只有第四次的實驗結果沒有被誤判，而對於最準確的分類結果則為第一類，只有第三次的實驗結果誤判一次而已。

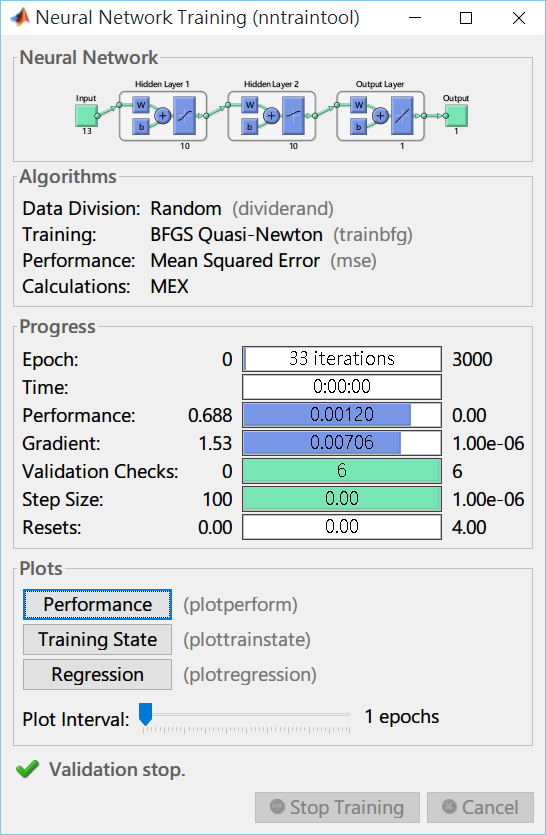
相對於 Quasi-Newton 從 Scaled Conjugate Gradient 測出來的結果可以發現，每次的實驗結果分類都很類似很穩定，唯一例外是第二次的實驗結果在第三類的分類突然變差，在觀察Performance後可以發現Train出來的MSE在比較之後的epoch沒有明顯下降，這或許是造成結果分類錯誤變大的原因。

而跟作業2的實驗結果相比整體分類的辨識率都上升許多，因為自己作業2寫的程式貌似出了點錯誤。

Dataset 2: Wine

Learning algorithm 1. BFGS Quasi-Newton

1. 網路架構及參數設定



Input layer: 13個neuron

兩層 hidden layer

First hidden layer: 10個neuron (Hyperbolic tangent sigmoid transfer function)

Second hidden layer: 10個neuron (Log-sigmoid transfer function)

Output layer: 1個neuron (Linear transfer function)

Learning Rate: 0.002

Epoch: 3000

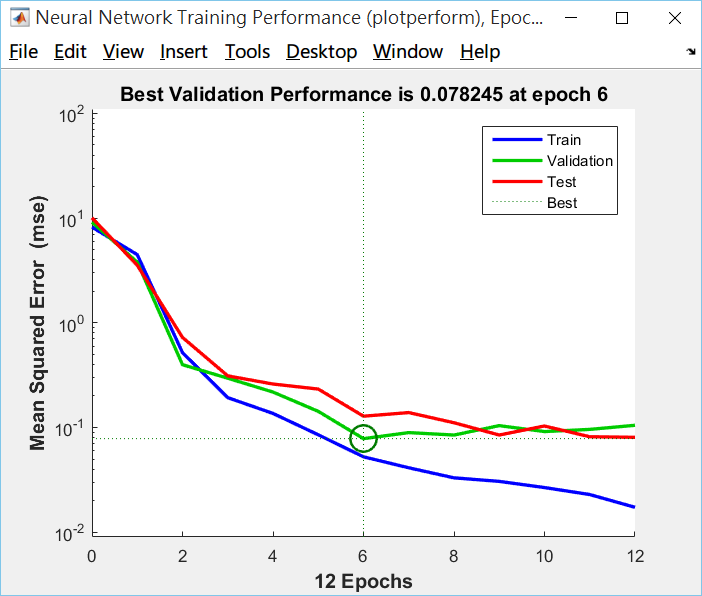
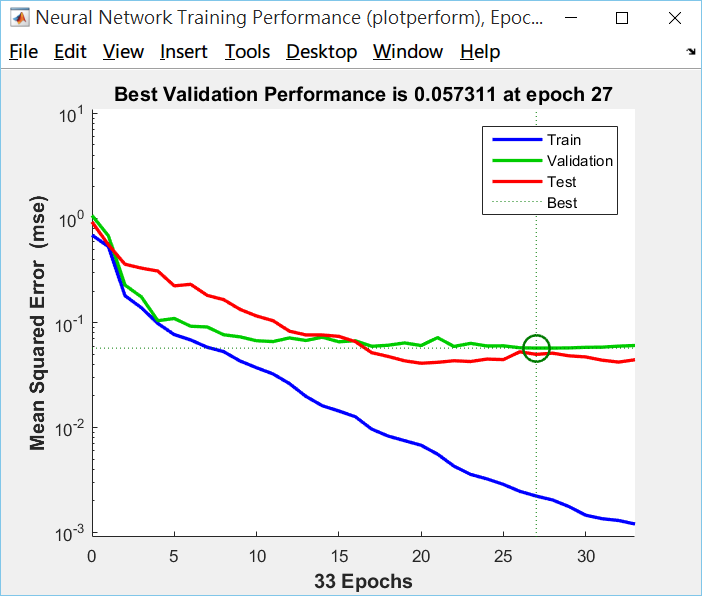
Goal: 0

Show: 25

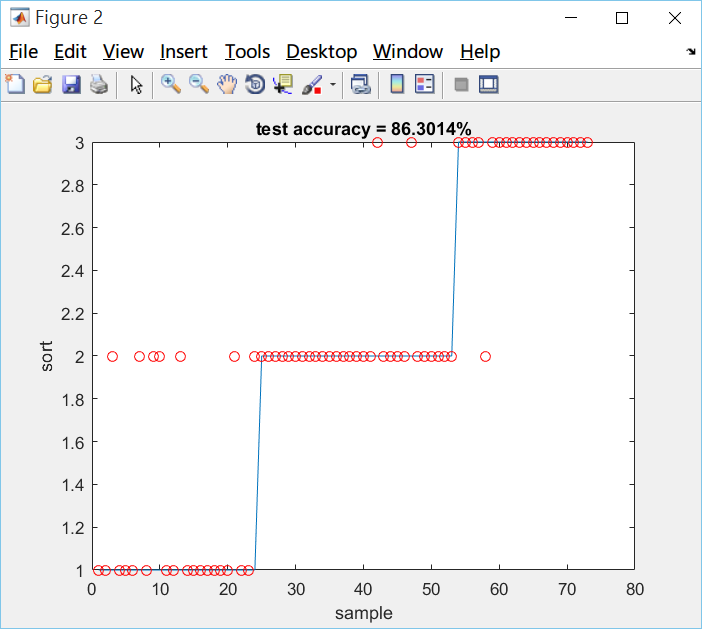
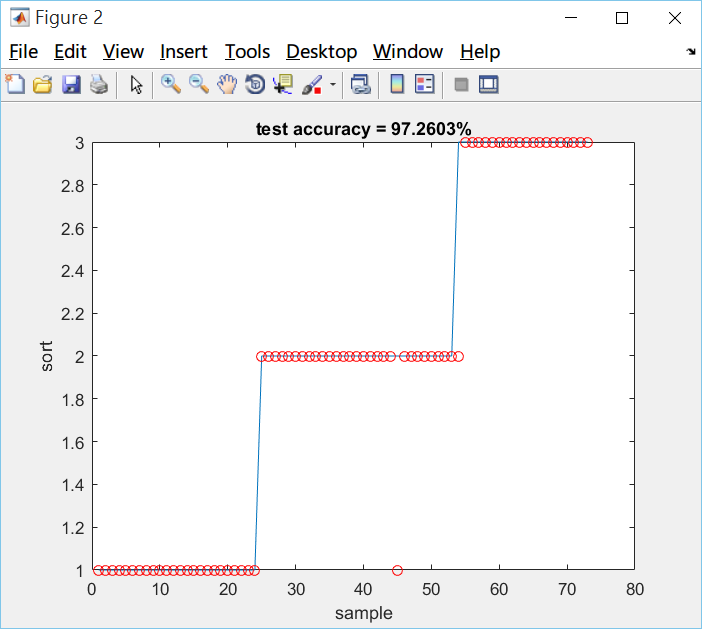
1. Performance(learning curve)
2. 結果圖(需附註辨識率)

第1, 2次結果

Performance

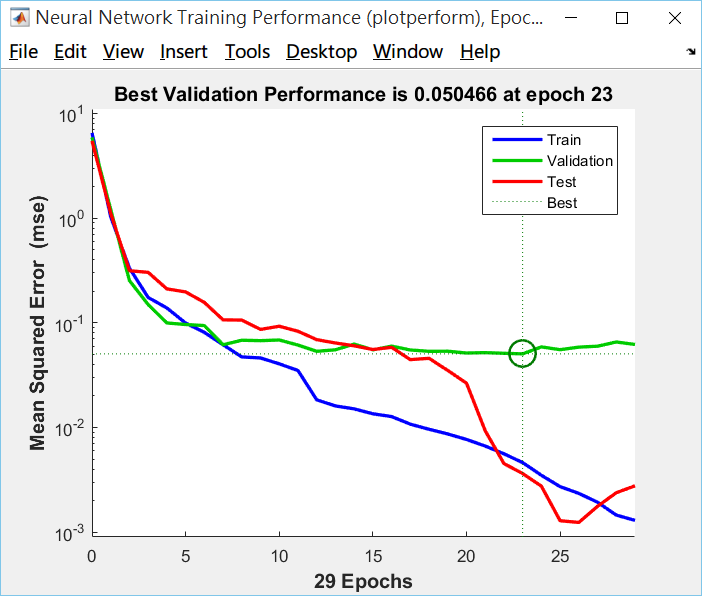
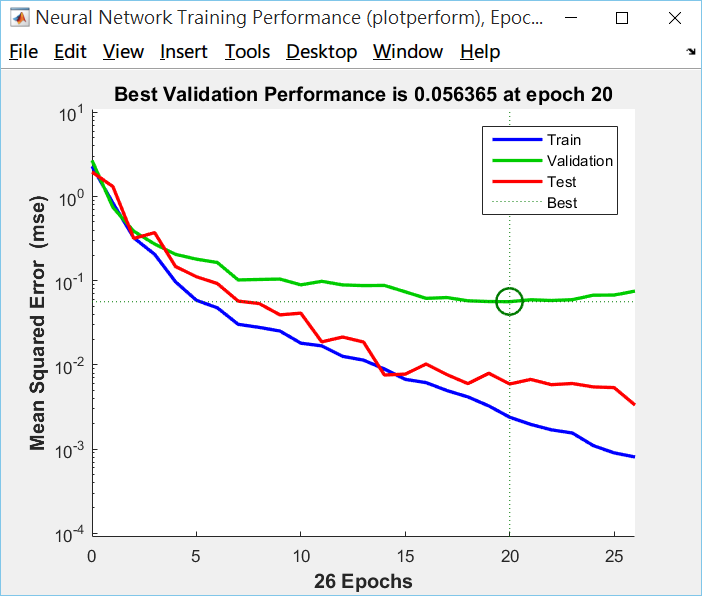


Accuracy(最上方標題部分為辨識成功率)

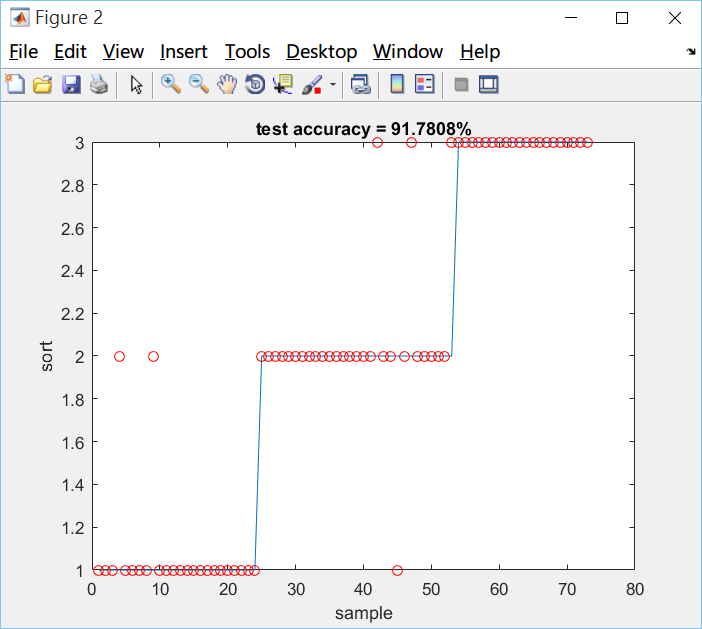
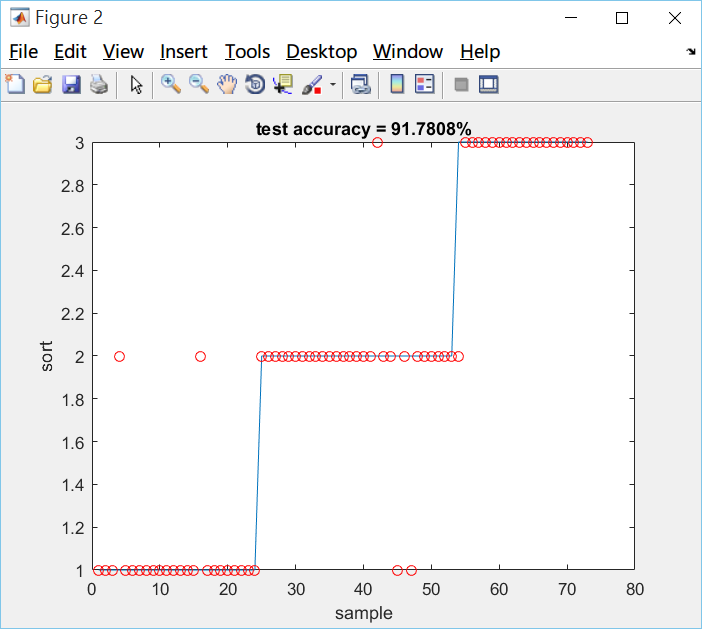


第3, 4次結果

Performance

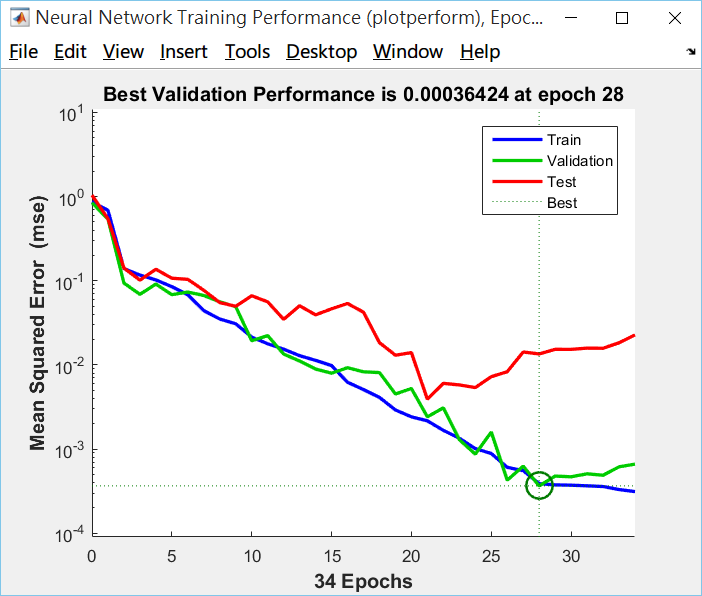


Accuracy(最上方標題部分為辨識成功率)

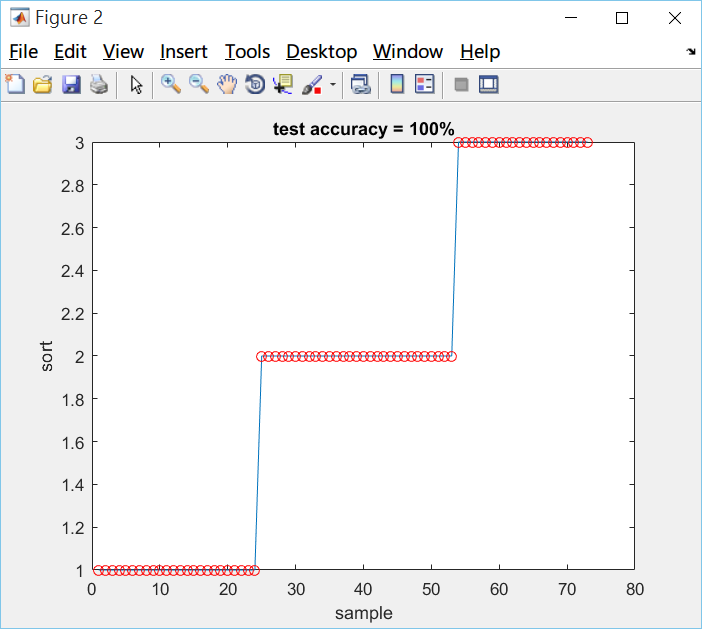


第5次結果

Performance



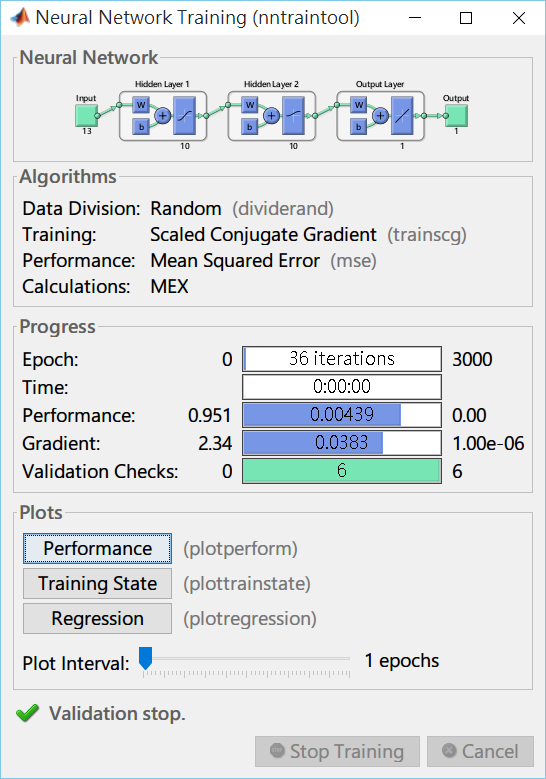
Accuracy(最上方標題部分為辨識成功率)



平均準確率: 93.425%(四捨五入到小數點後三位)

Learning algorithm 2. Scaled Conjugate Gradient

1. 網路架構及參數設定



Input layer: 13個neuron

兩層 hidden layer

First hidden layer: 10個neuron (Hyperbolic tangent sigmoid transfer function)

Second hidden layer: 10個neuron (Log-sigmoid transfer function)

Output layer: 1個neuron (Linear transfer function)

Learning Rate: 0.002

Epoch: 3000

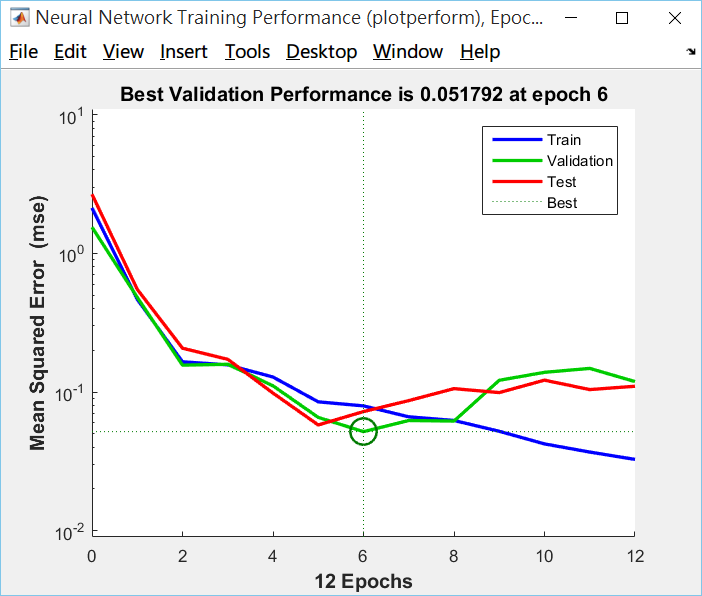
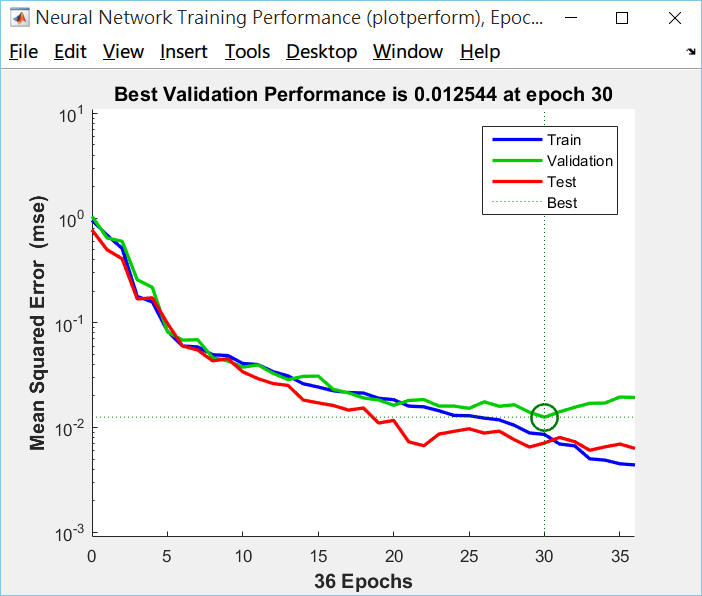
Goal: 0

Show: 25

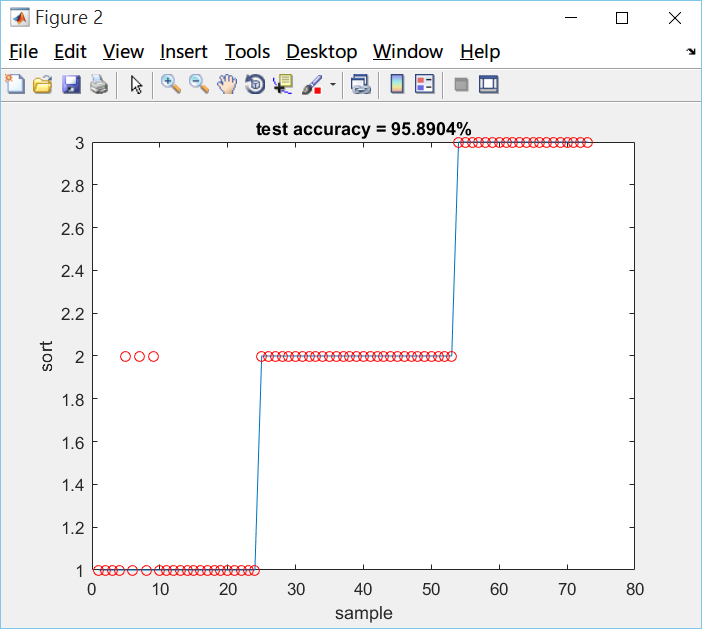
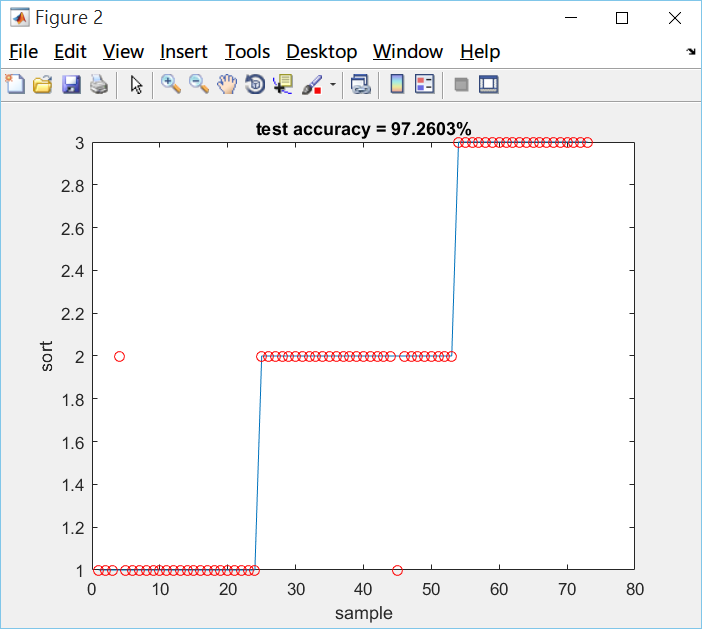
1. Performance(learning curve)
2. 結果圖(需附註辨識率)

第1, 2次結果

Performance

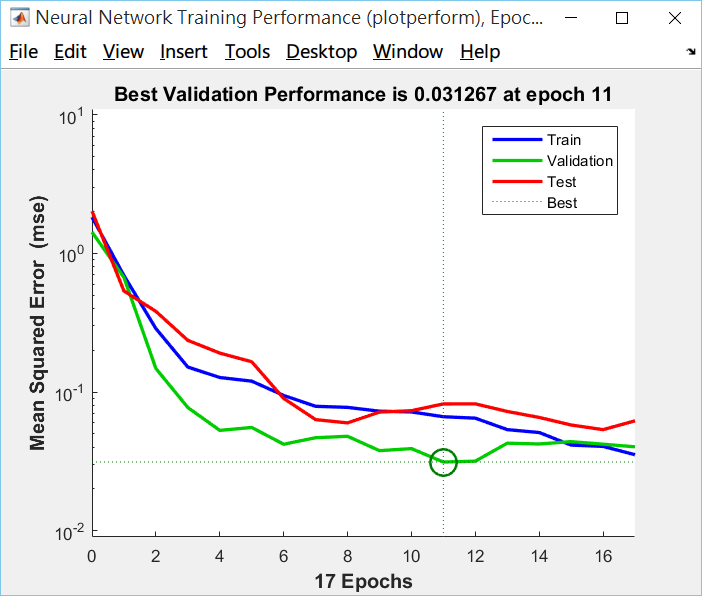


Accuracy(最上方標題部分為辨識成功率)

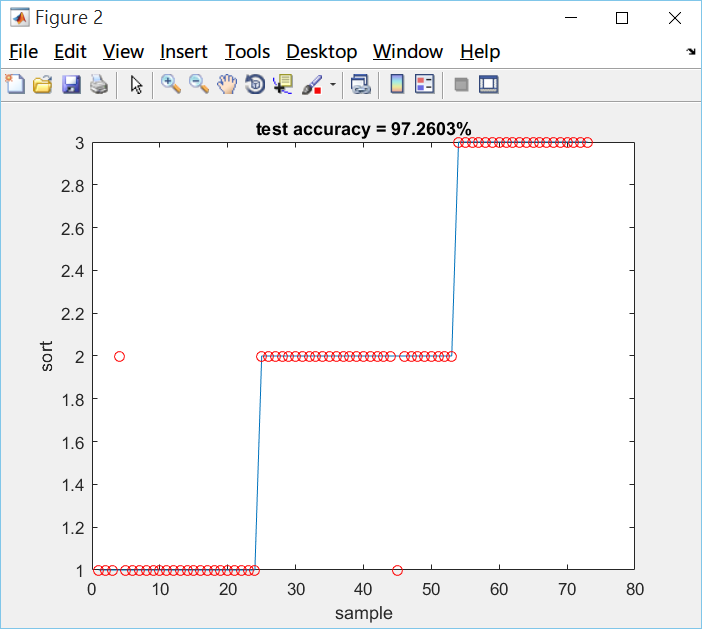
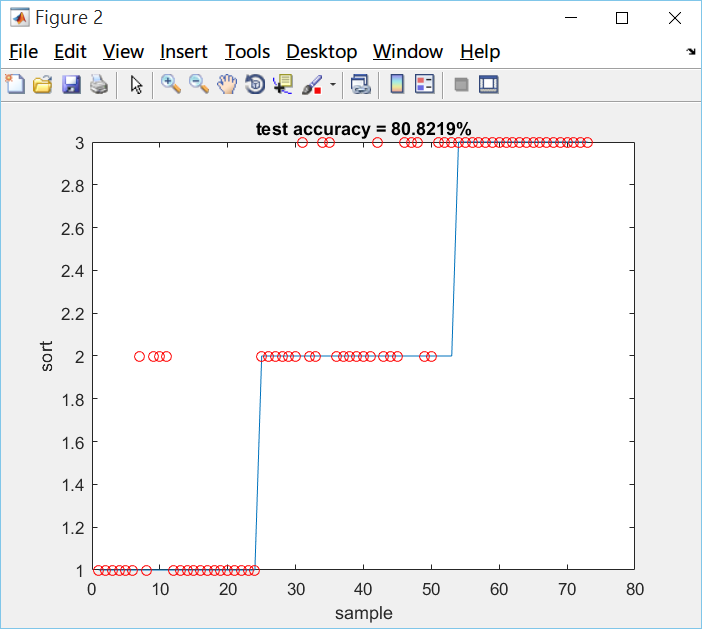


第3, 4次結果

Performance

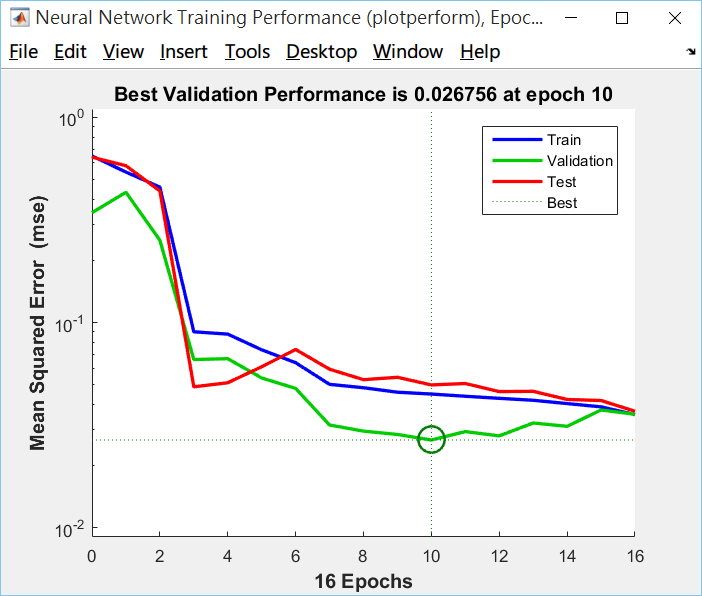


Accuracy(最上方標題部分為辨識成功率)

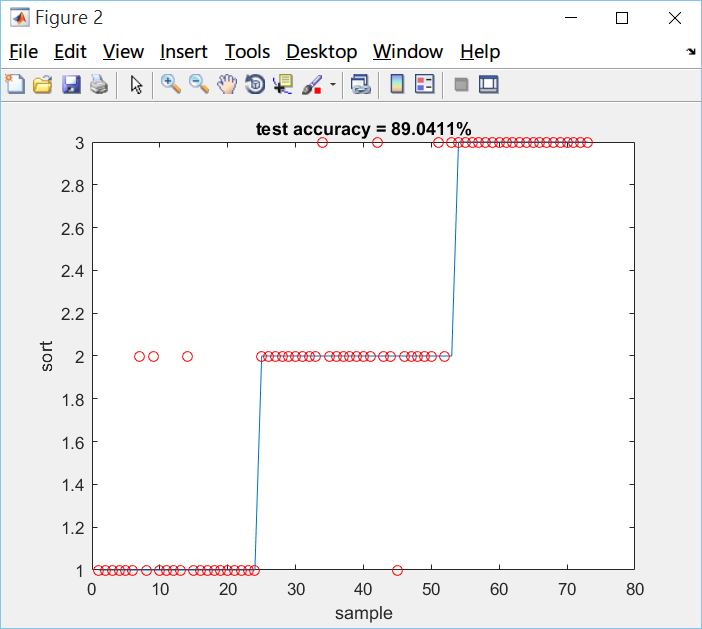


第5次結果

Performance



Accuracy(最上方標題部分為辨識成功率)



平均準確率: 92.055%(四捨五入到小數點後三位)

1. learning algorithms方法差異之討論

從 Quasi-Newton 測出來的結果可以發現，部分實驗結果對於第一類的分類誤判情形相對嚴重，而第二類則有時會有輕微誤判情況，第五次實驗結果則是辨識率100%，可能是運氣非常好的狀況才會這樣。

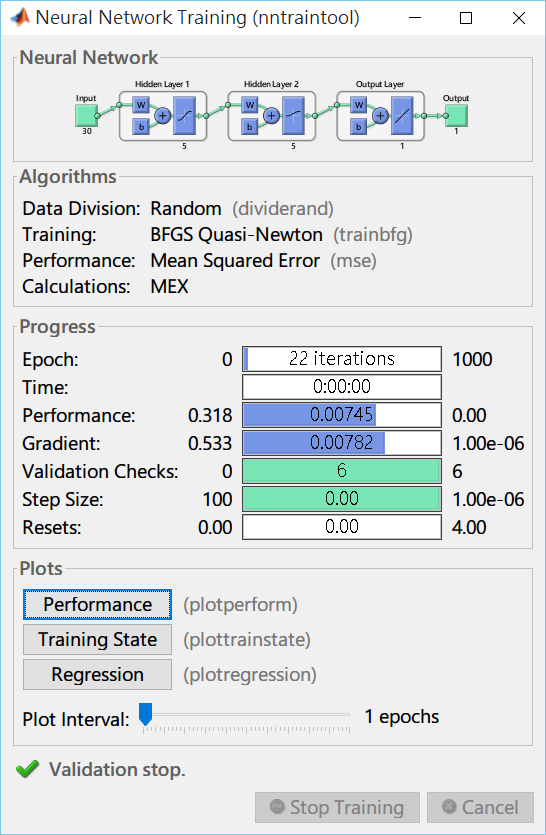
相對於 Quasi-Newton 從 Scaled Conjugate Gradient 測出來的結果可以發現，在一二類分類的情況之中非常不穩定，在第三次實驗結果尤其明顯其辨識率只有80%，但最好辨識率卻有97%。

而最後實驗結果部分無法跟作業2相比，因為作業2並未來得及完成此部份。

Dataset 3: Breast Cancer Wisconsin (Diagnostic)

Learning algorithm 1. BFGS Quasi-Newton

1. 網路架構及參數設定



Input layer: 30個neuron

兩層 hidden layer

First hidden layer: 5個neuron (Hyperbolic tangent sigmoid transfer function)

Second hidden layer: 5個neuron (Log-sigmoid transfer function)

Output layer: 1個neuron (Linear transfer function)

Learning Rate: 0.003

Epoch: 1000

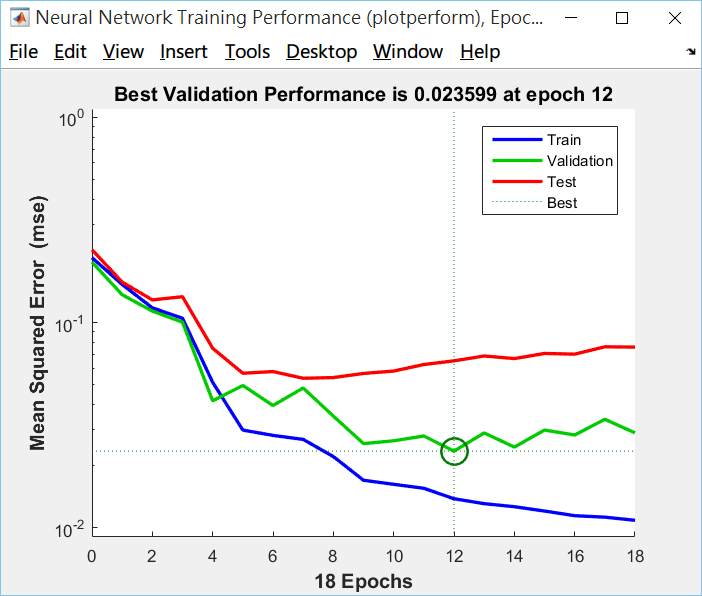
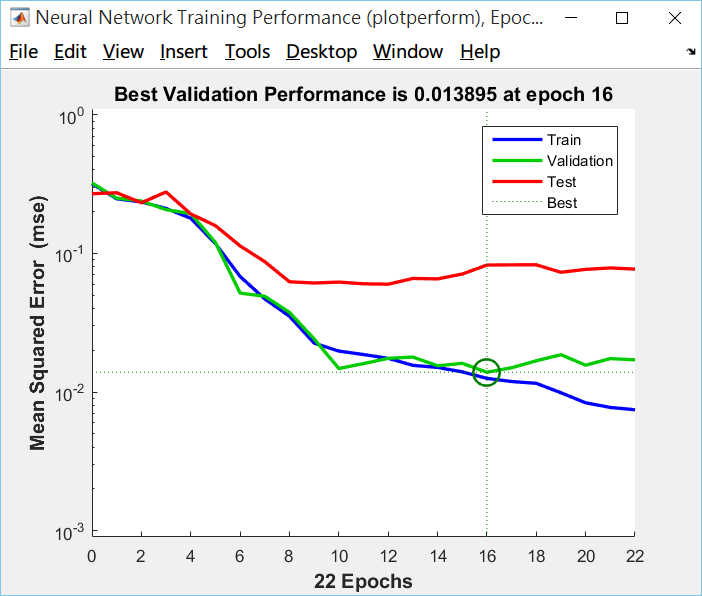
Goal: 0

Show: 25

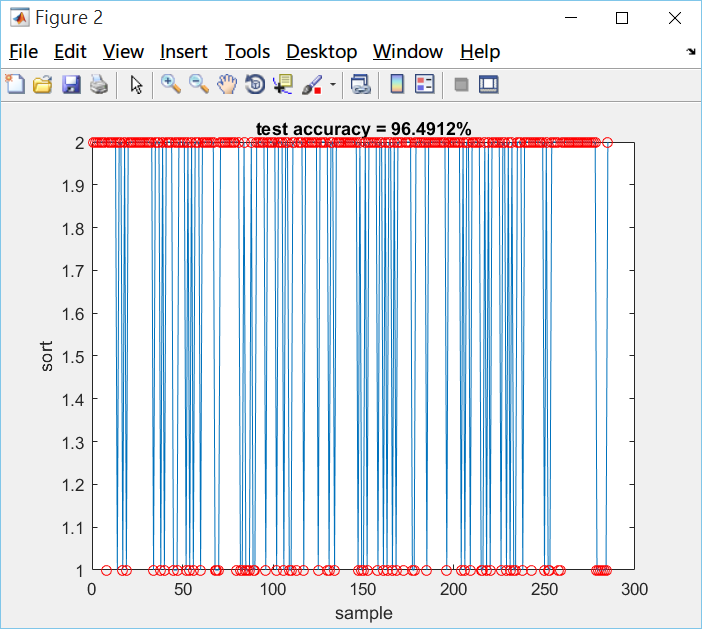
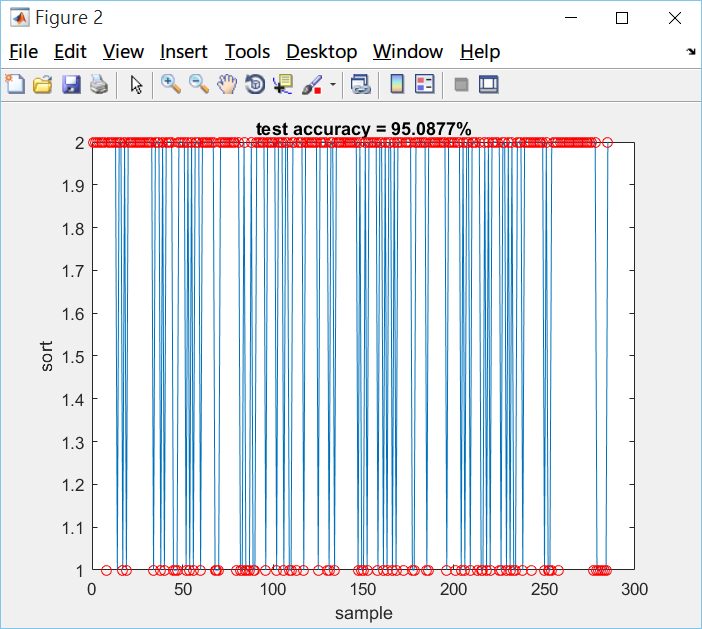
1. Performance(learning curve)
2. 結果圖(需附註辨識率)

第1, 2次結果

Performance

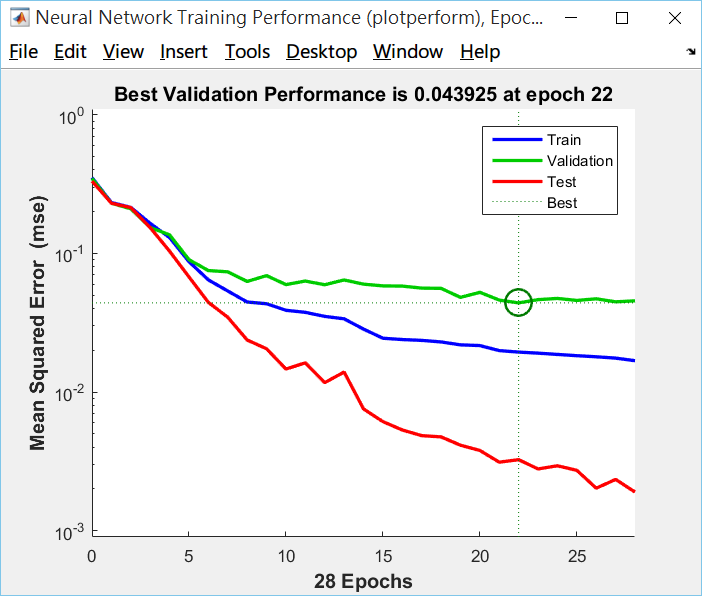
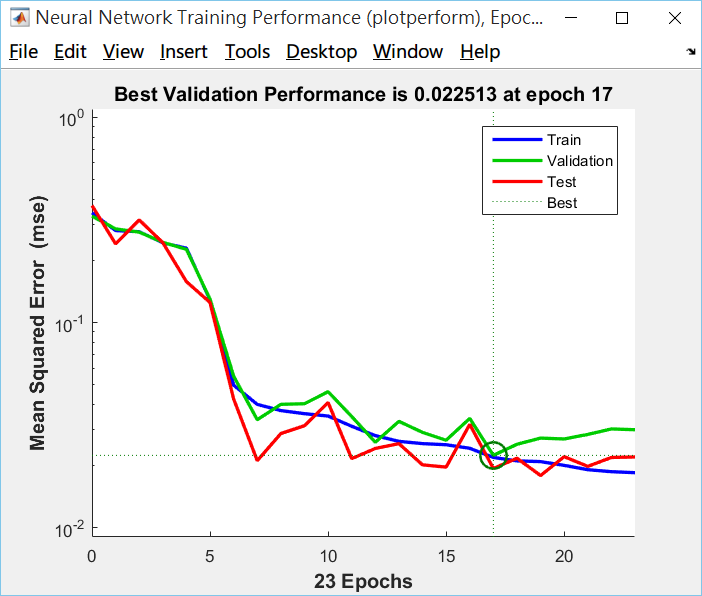


Accuracy(最上方標題部分為辨識成功率)

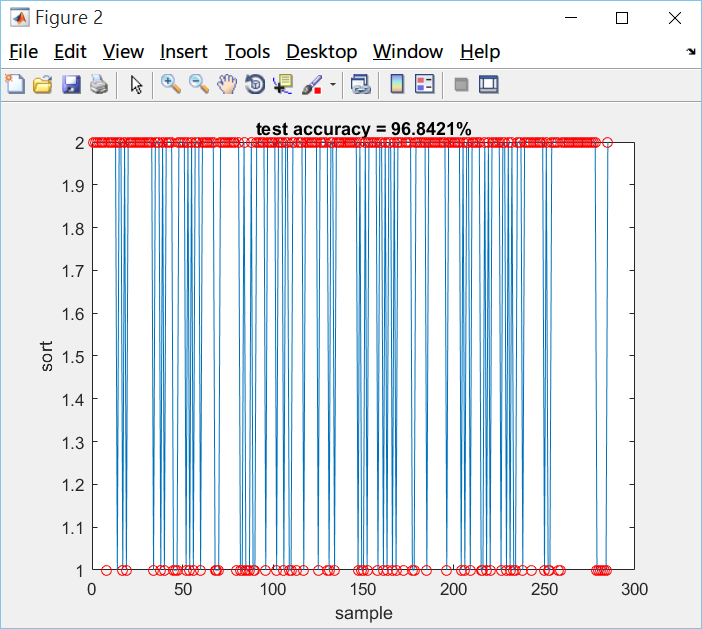
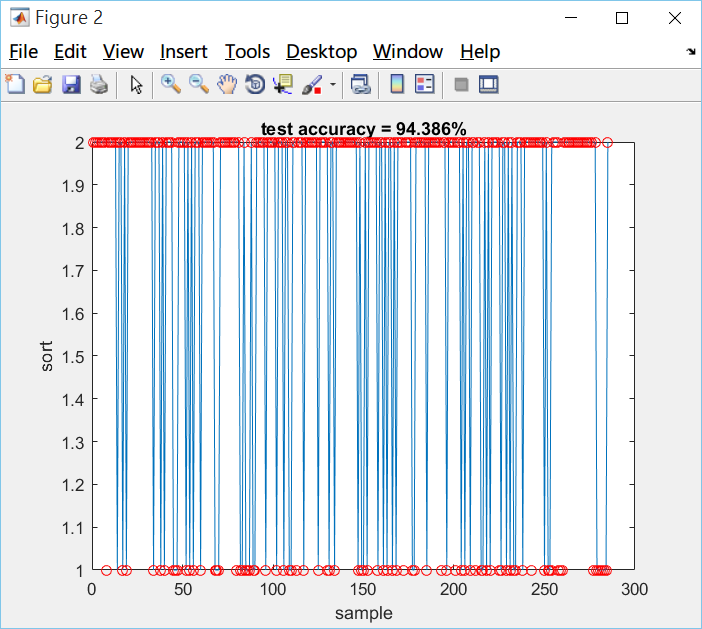


第3, 4次結果

Performance

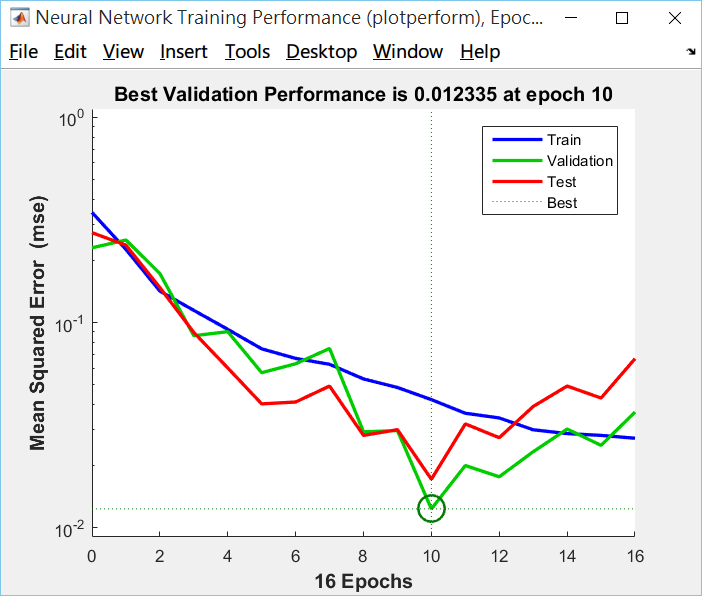


Accuracy(最上方標題部分為辨識成功率)

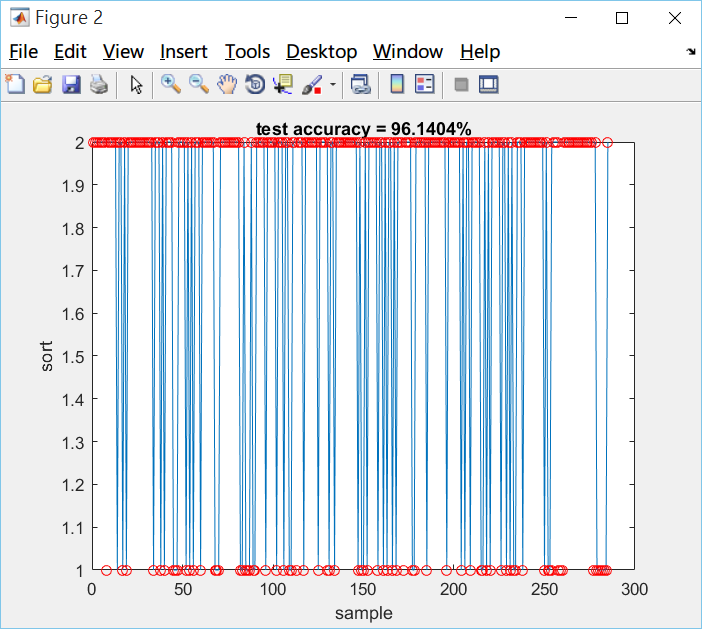


第5次結果

Performance



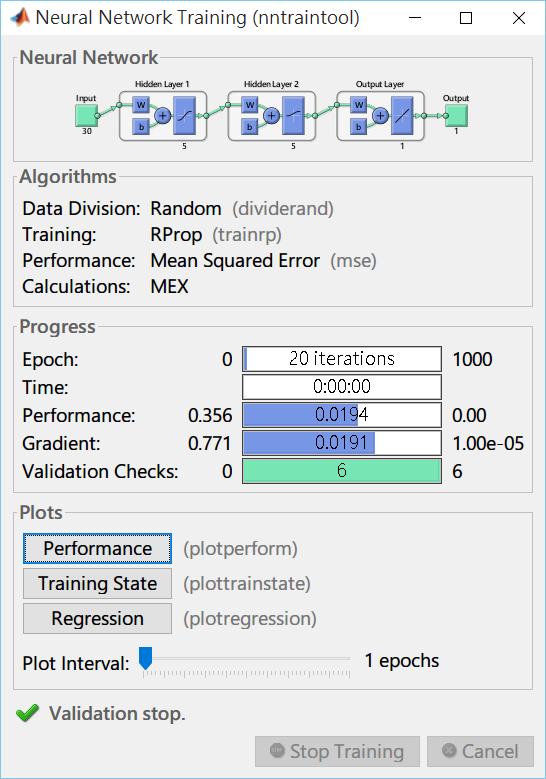
Accuracy(最上方標題部分為辨識成功率)



平均準確率: 95.789%(四捨五入到小數點後三位)

Learning algorithm 2. resillent backpropagation

1. 網路架構及參數設定



Input layer: 30個neuron

兩層 hidden layer

First hidden layer: 5個neuron (Hyperbolic tangent sigmoid transfer function)

Second hidden layer: 5個neuron (Log-sigmoid transfer function)

Output layer: 1個neuron (Linear transfer function)

Learning Rate: 0.003

Epoch: 1000

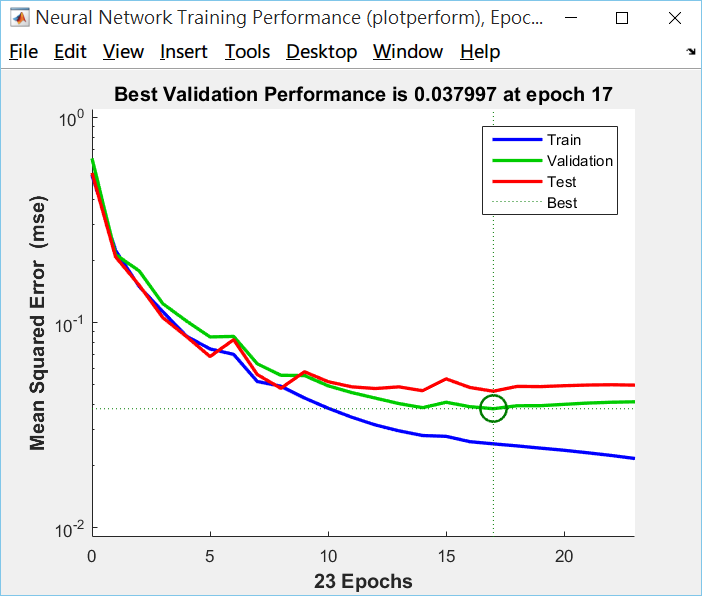
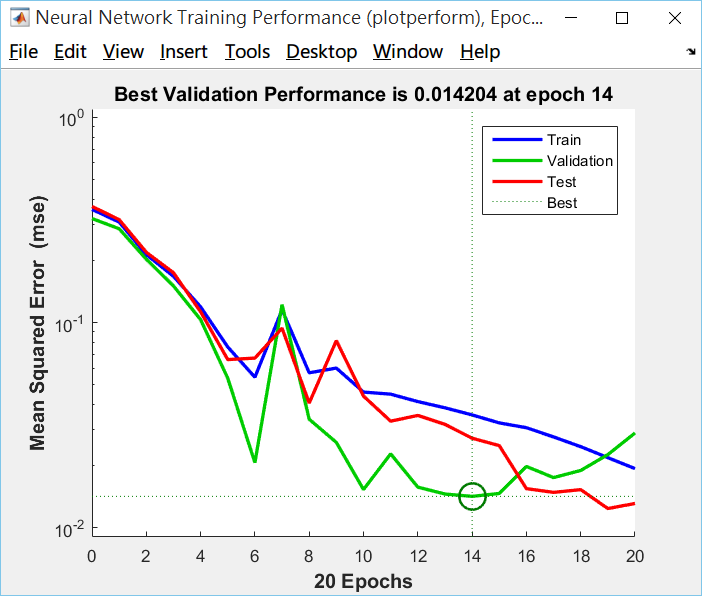
Goal: 0

Show: 25

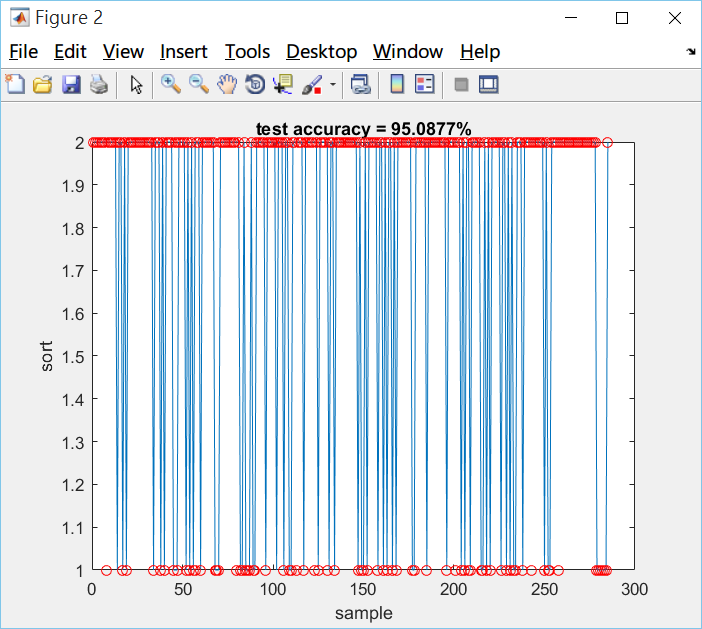
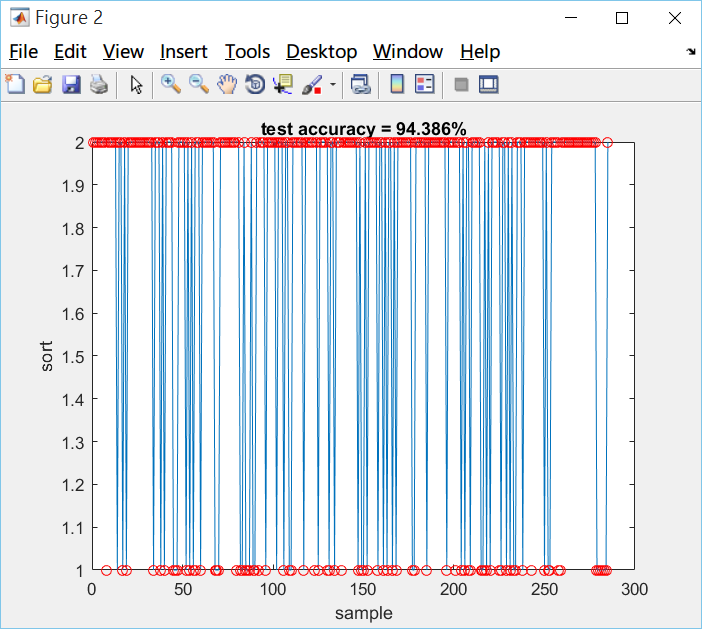
1. Performance(learning curve)
2. 結果圖(需附註辨識率)

第1, 2次結果

Performance

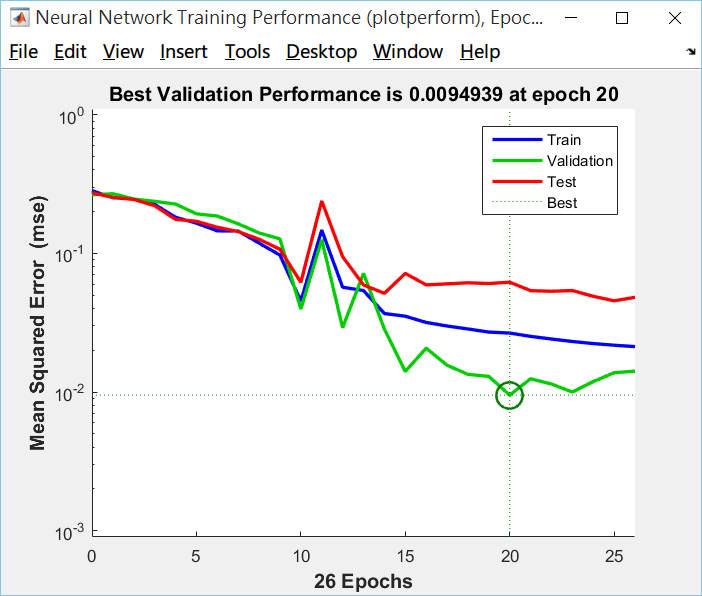
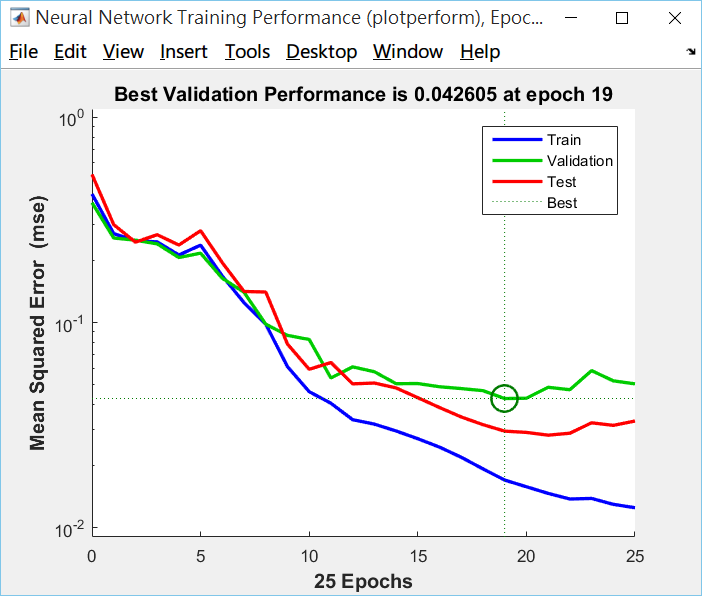


Accuracy(最上方標題部分為辨識成功率)

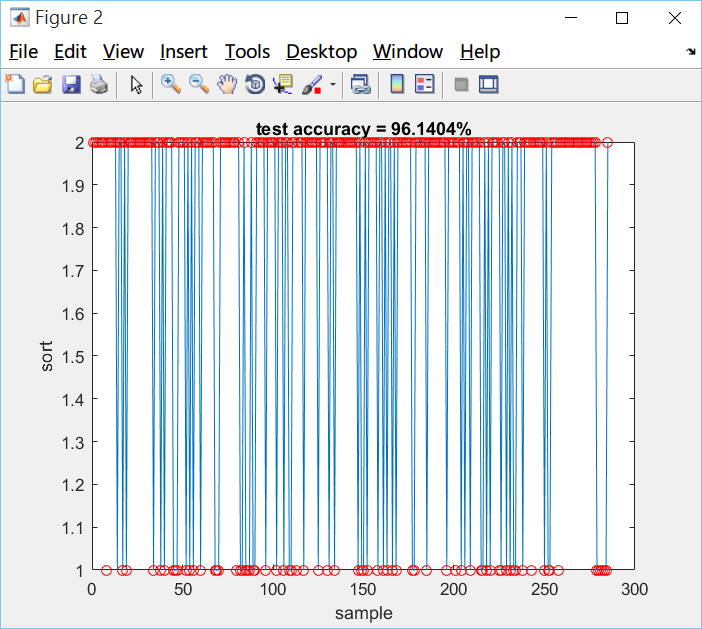
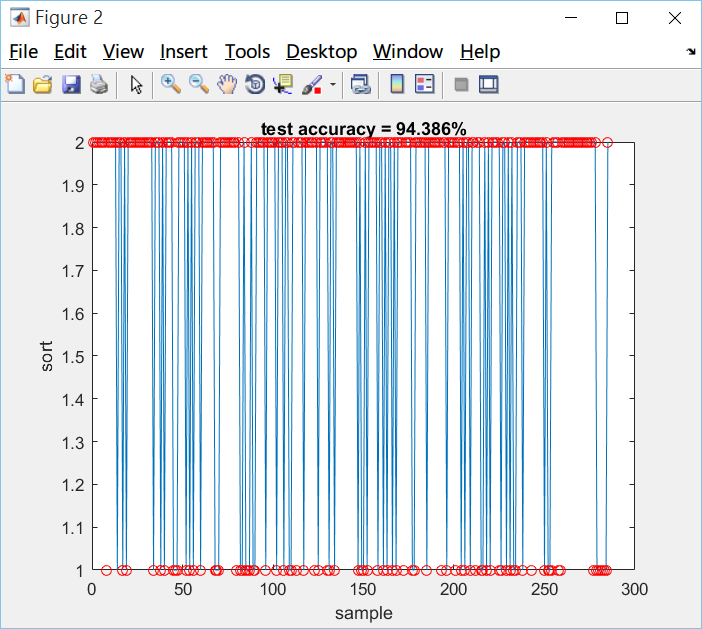


第3, 4次結果

Performance

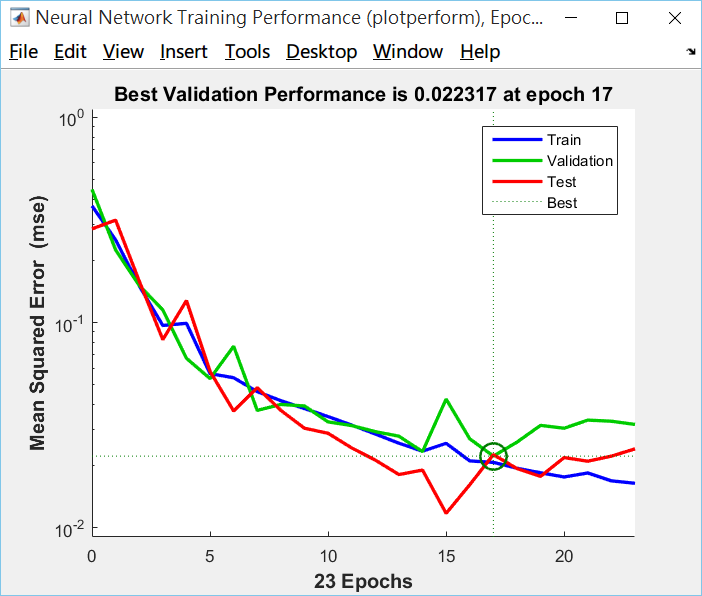


Accuracy(最上方標題部分為辨識成功率)

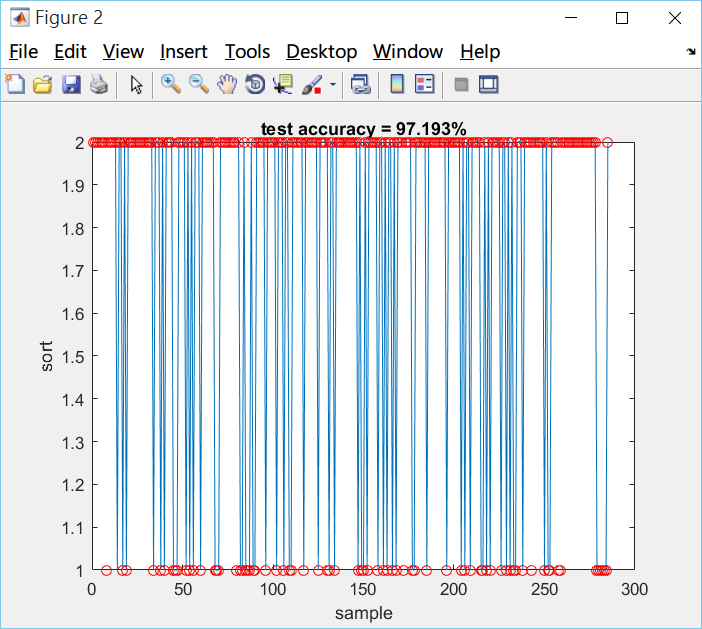


第5次結果

Performance



Accuracy(最上方標題部分為辨識成功率)



平均準確率: 95.439%(四捨五入到小數點後三位)

1. learning algorithms方法差異之討論

從 Quasi-Newton 測出來的結果可以發現，這次實驗結果其實每次都很相近，但由於資料分佈是隨機的，所以從accuracy來分析資料誤判情形會有點困難。

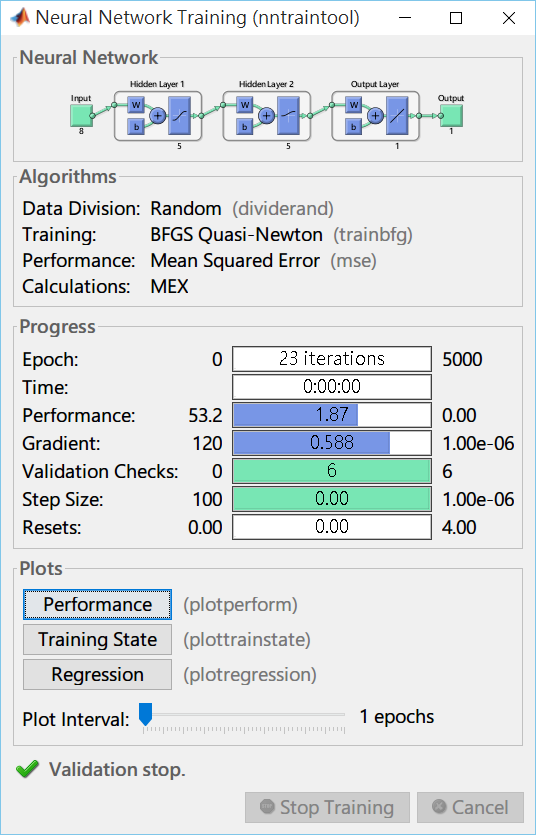
相對於 Quasi-Newton 從 resillent backpropagation 測出來的結果可以發現，其實結果也都很類似。和前兩次data相比，這次資料數量有點多，也可能是造成實驗結果趨於穩定的原因。

而最後實驗結果部分無法跟作業2相比，因為作業2並未來得及完成此部份。

Dataset 4: Yeast

Learning algorithm 1. BFGS Quasi-Newton

1. 網路架構及參數設定



Input layer: 8個neuron

兩層 hidden layer

First hidden layer: 5個neuron (Hyperbolic tangent sigmoid transfer function)

Second hidden layer: 5個neuron (Log-sigmoid transfer function)

Output layer: 1個neuron (Linear transfer function)

Learning Rate: 0.001

Epoch: 5000

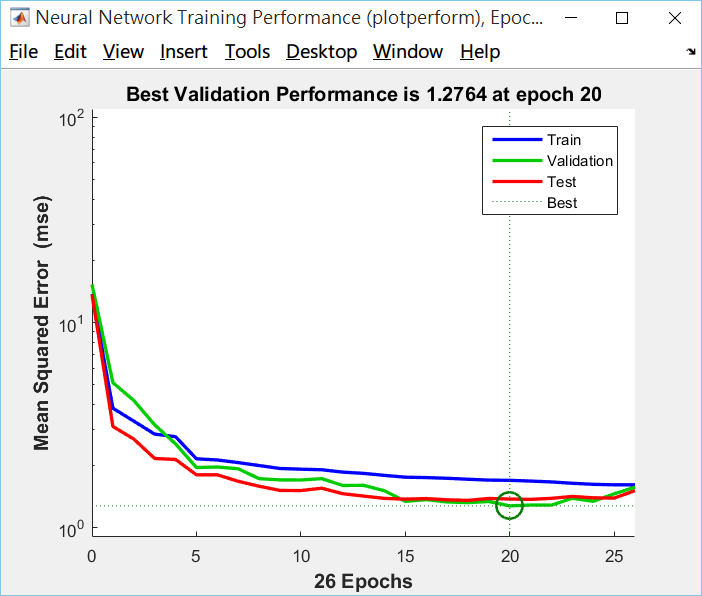
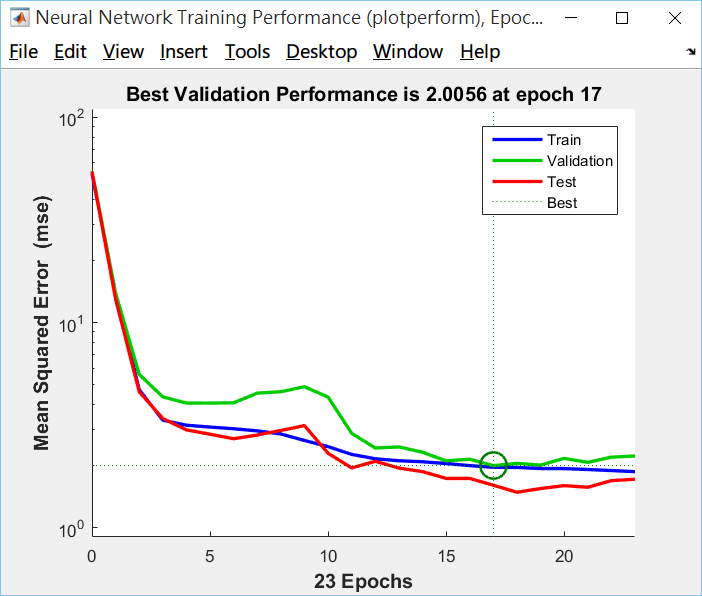
Goal: 0

Show: 25

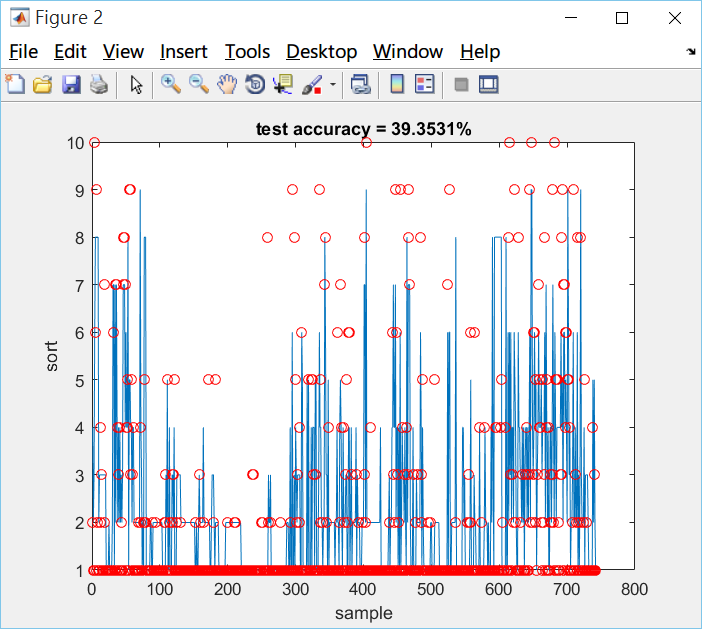
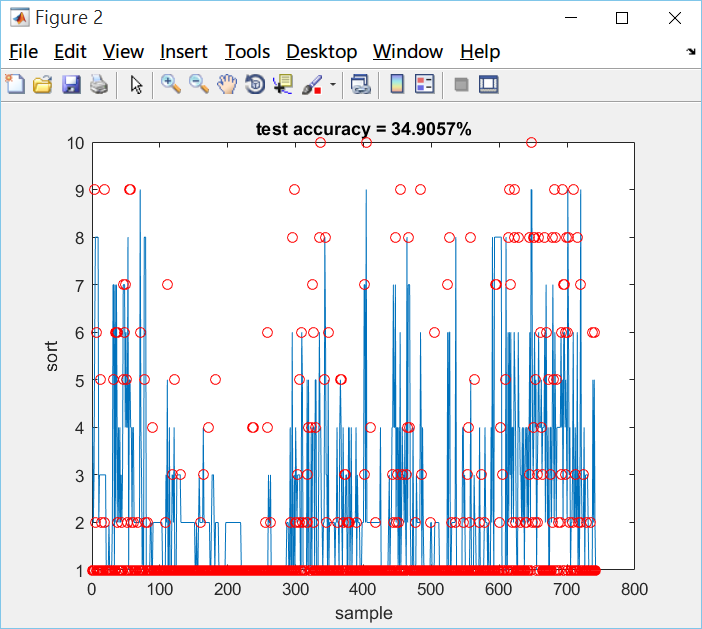
1. Performance(learning curve)
2. 結果圖(需附註辨識率)

第1, 2次結果

Performance

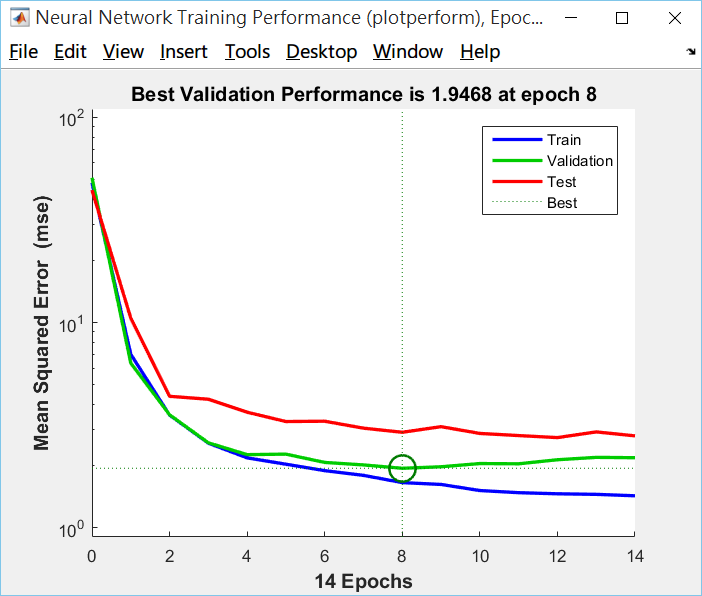
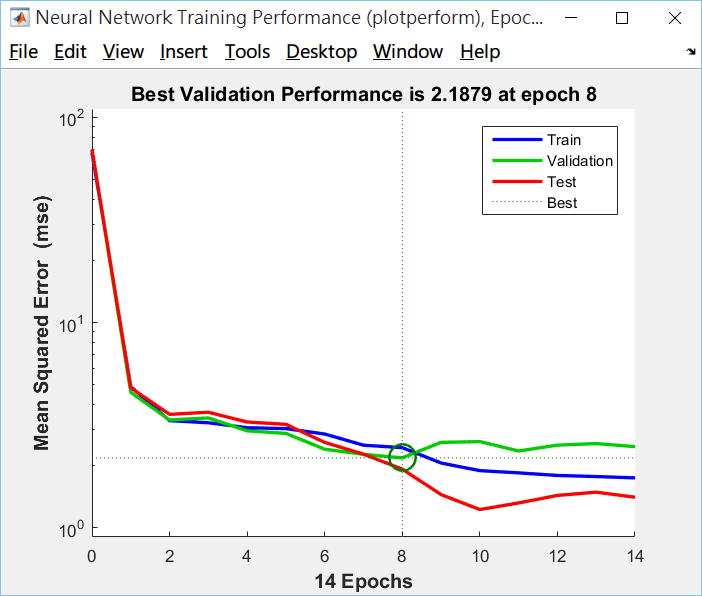


Accuracy(最上方標題部分為辨識成功率)

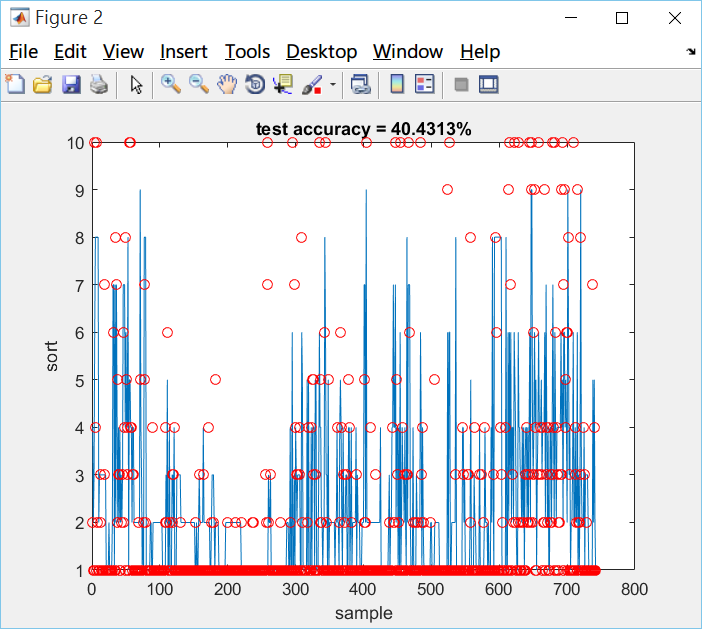
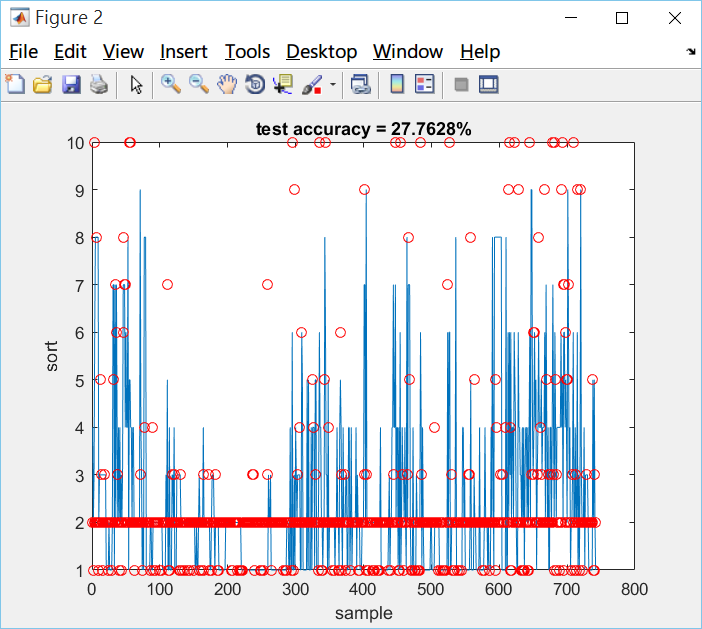


第3, 4次結果

Performance

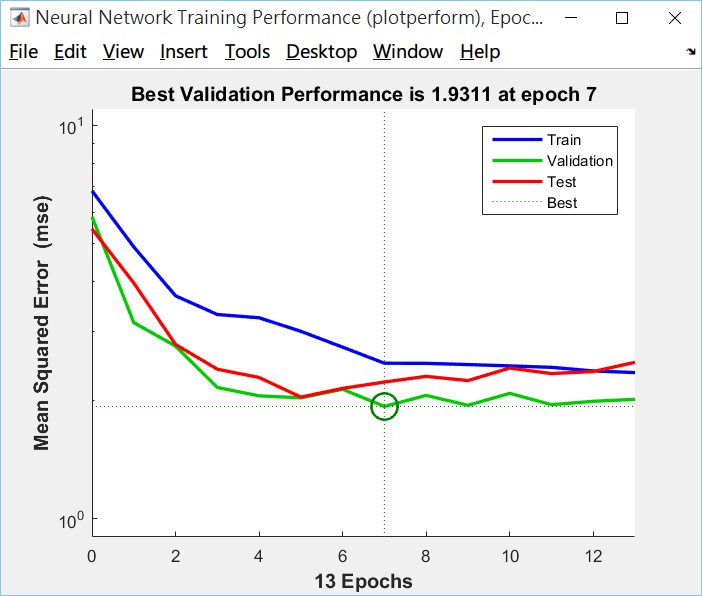


Accuracy(最上方標題部分為辨識成功率)

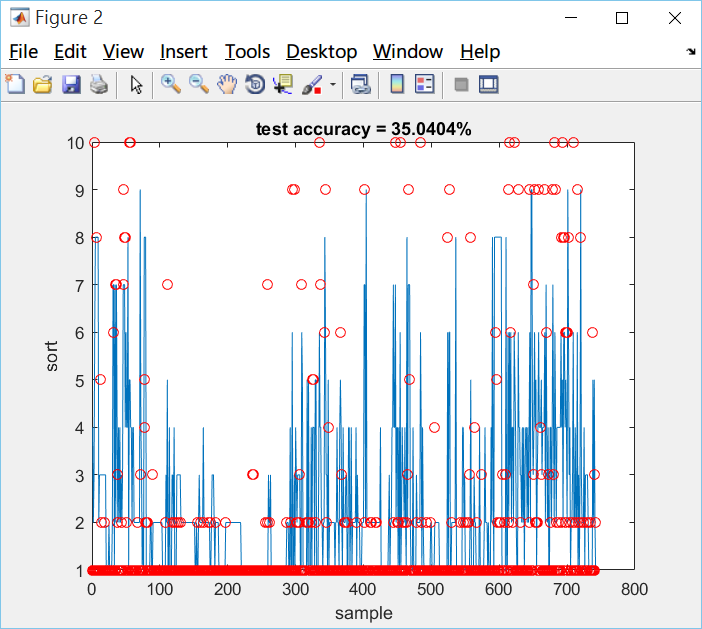


第5次結果

Performance



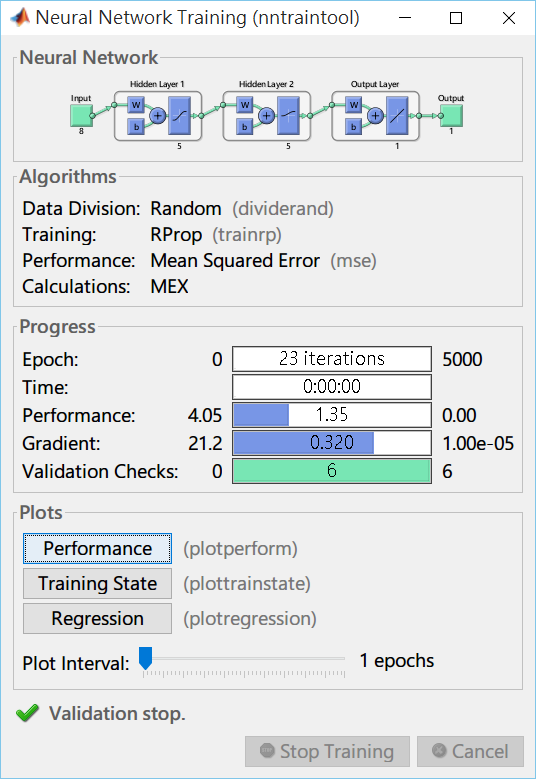
Accuracy(最上方標題部分為辨識成功率)



平均準確率: 35.499%(四捨五入到小數點後三位)

Learning algorithm 2. resillent backpropagation

1. 網路架構及參數設定



Input layer: 30個neuron

兩層 hidden layer

First hidden layer: 5個neuron (Hyperbolic tangent sigmoid transfer function)

Second hidden layer: 5個neuron (Log-sigmoid transfer function)

Output layer: 1個neuron (Linear transfer function)

Learning Rate: 0.003

Epoch: 1000

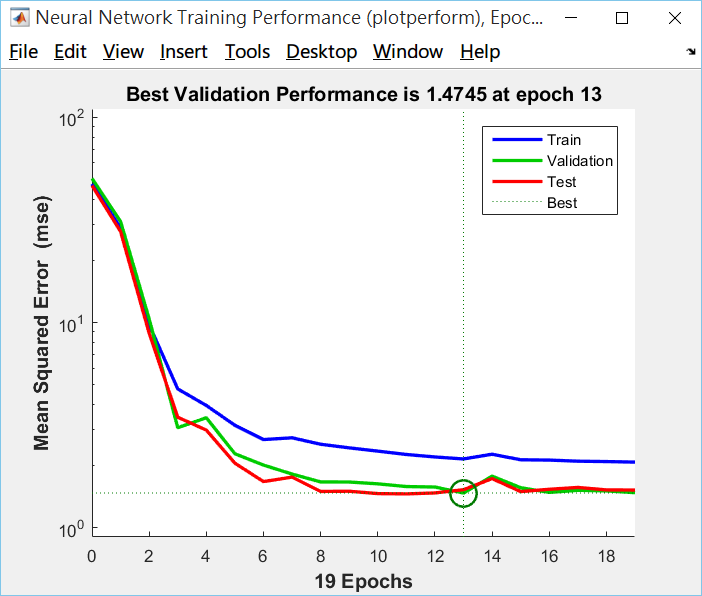
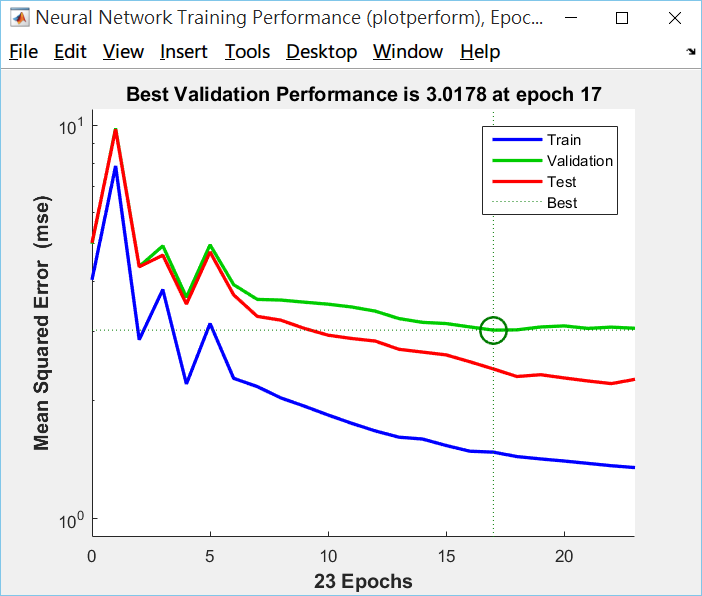
Goal: 0

Show: 25

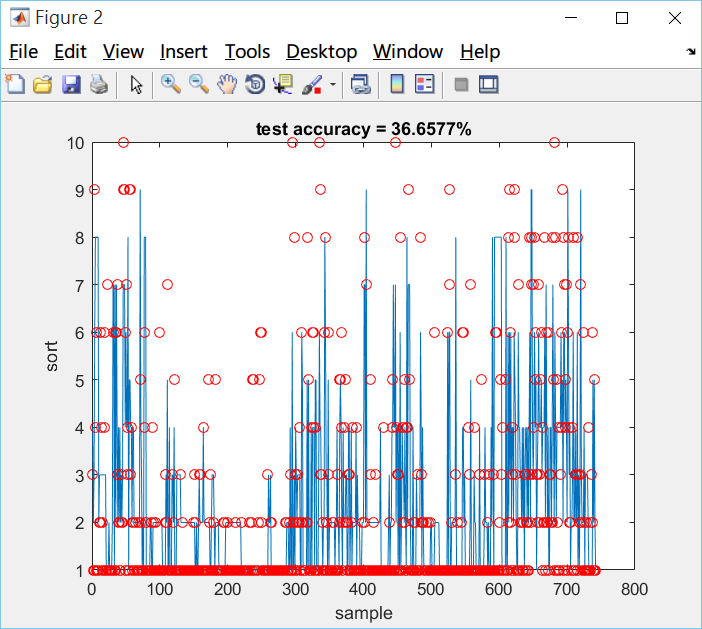
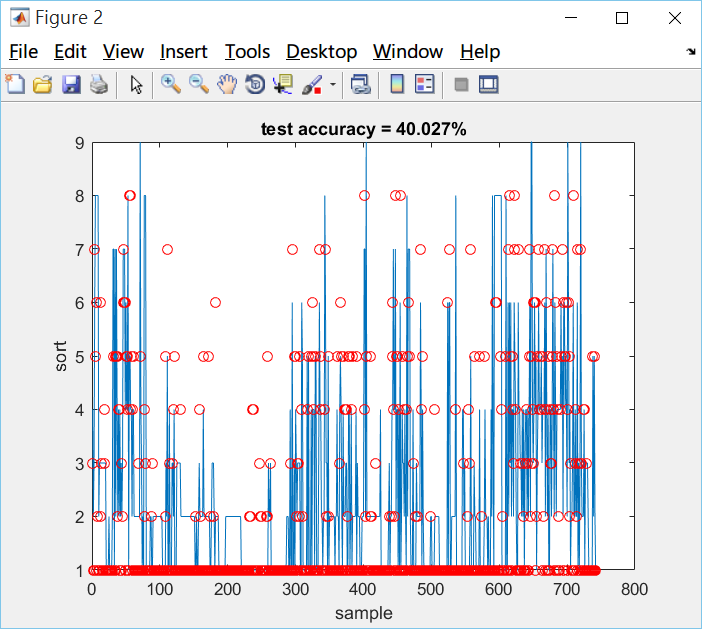
1. Performance(learning curve)
2. 結果圖(需附註辨識率)

第1, 2次結果

Performance

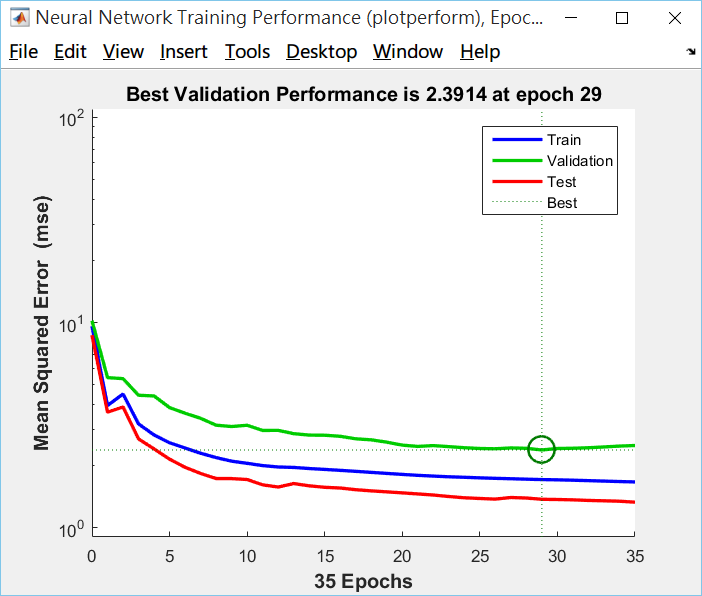
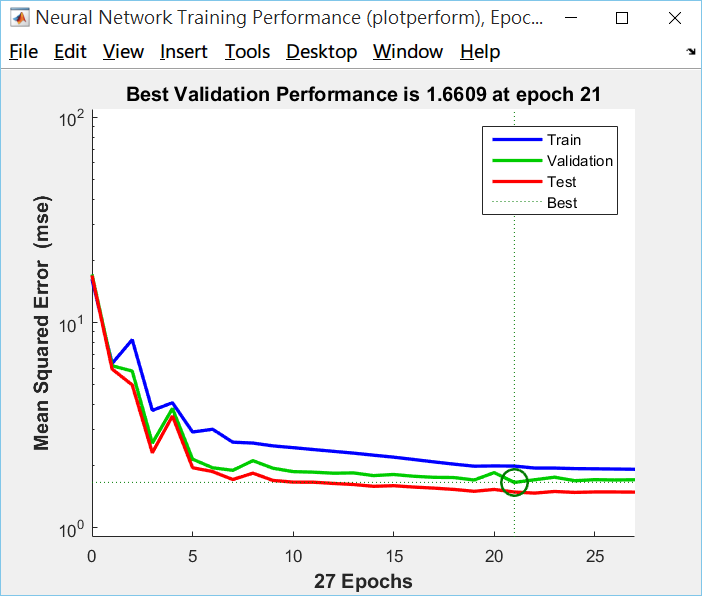


Accuracy(最上方標題部分為辨識成功率)

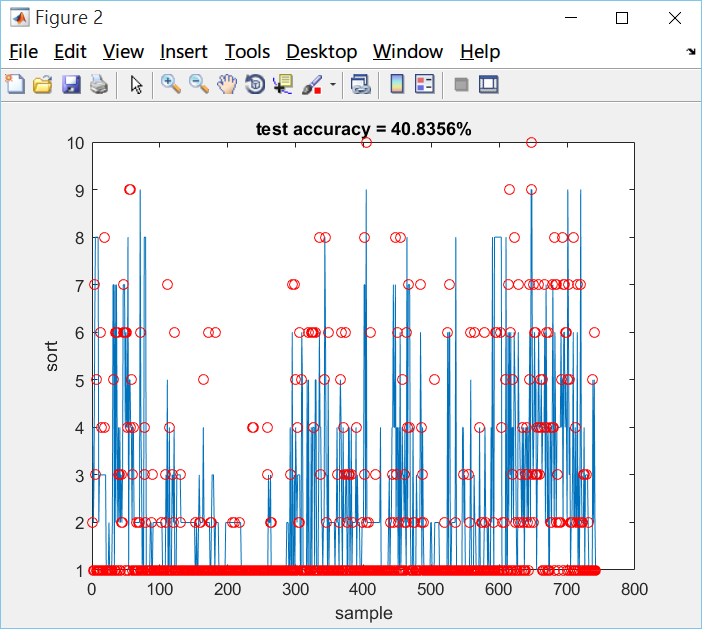
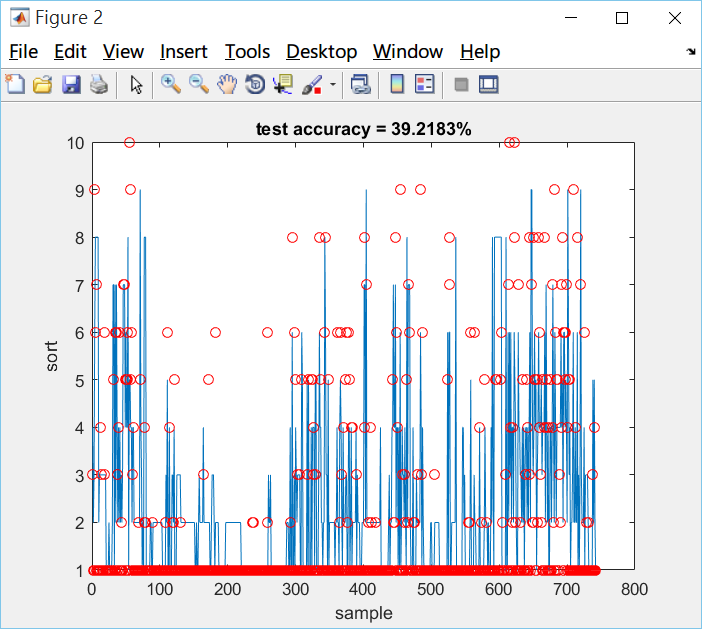


第3, 4次結果

Performance

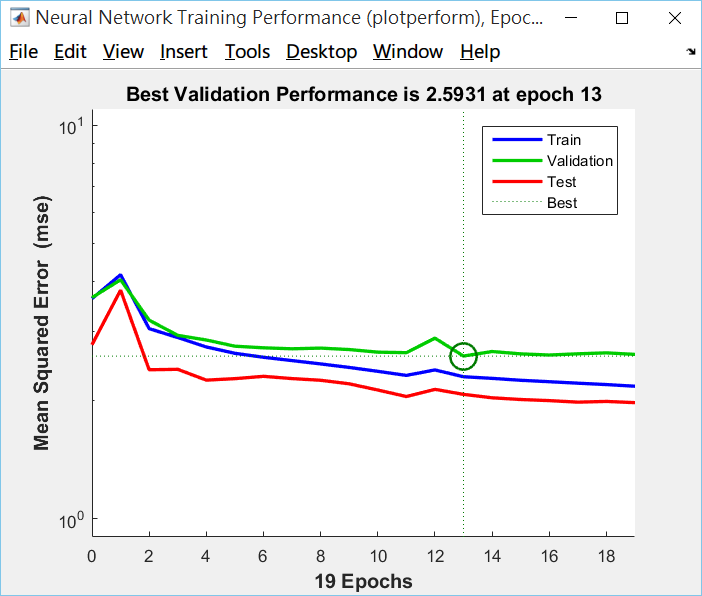


Accuracy(最上方標題部分為辨識成功率)

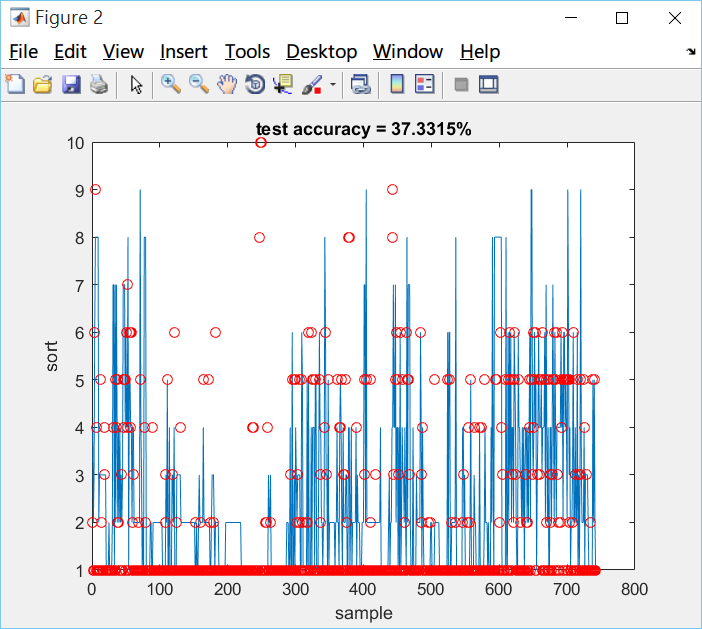


第5次結果

Performance



Accuracy(最上方標題部分為辨識成功率)



平均準確率: 38.814%(四捨五入到小數點後三位)

1. learning algorithms方法差異之討論

從 Quasi-Newton 測出來的結果可以發現，辨識率相對於前面幾組data而言降低了很多，仔細觀察輸入的資料內容能夠發現是因為有些分類的組數資料過少像是ERL在這上千筆資料中只佔了其中5組，要辨識可說是非常難的事情。

相對於 Quasi-Newton 從 resillent backpropagation 測出來的結果可以發現，其實結果也都很類似辨識率極低，但辨識成功率比前一個演算法提高了一點。

而最後實驗結果部分無法跟作業2相比，因為作業2並未來得及完成此部份。