Artificial Neural Networks

Exercise Session 1 - Supervised learning and Generalisation,

report by,

Marimuthu Ananthavelu (r0652832), KU Leuven.

Comparison of various algorithms:

The following Backpropagation algorithms are compared for performance with the given data.

1. ***trainlm*** is a network training function that updates weight and bias values per Levenberg-Marquardt optimisation.
2. ***traingd*** is a network training function that updates weight and bias values per gradient descent.
3. ***trainbfg*** is a network training function that updates weight and bias values per the BFGS quasi-Newton method.

The data are trained and tested using different feedforward network architecture with varying combinations of optimization techniques and the findings are as below;

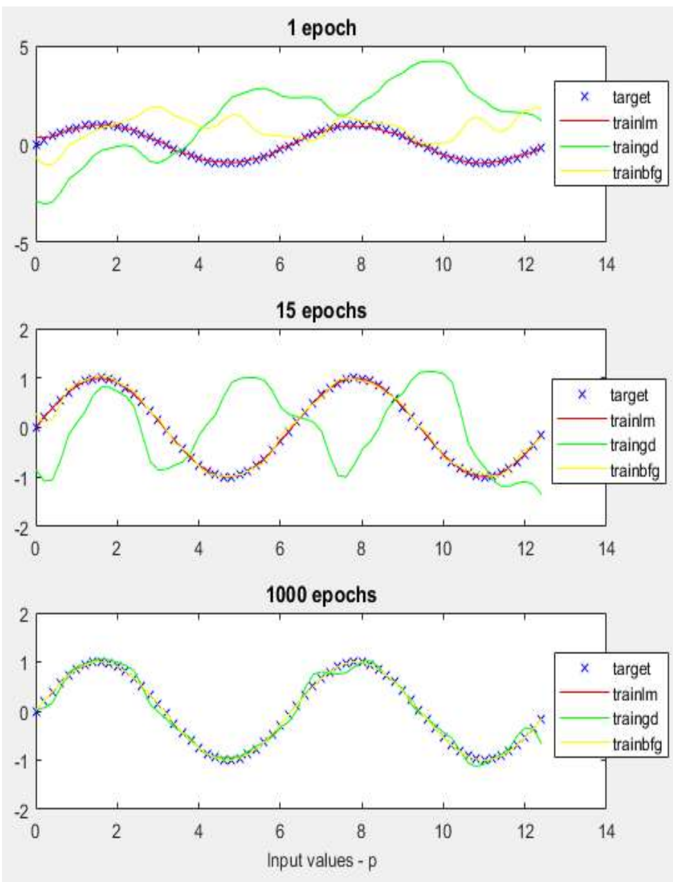

Fig.1.Curve fitting for various algorithms

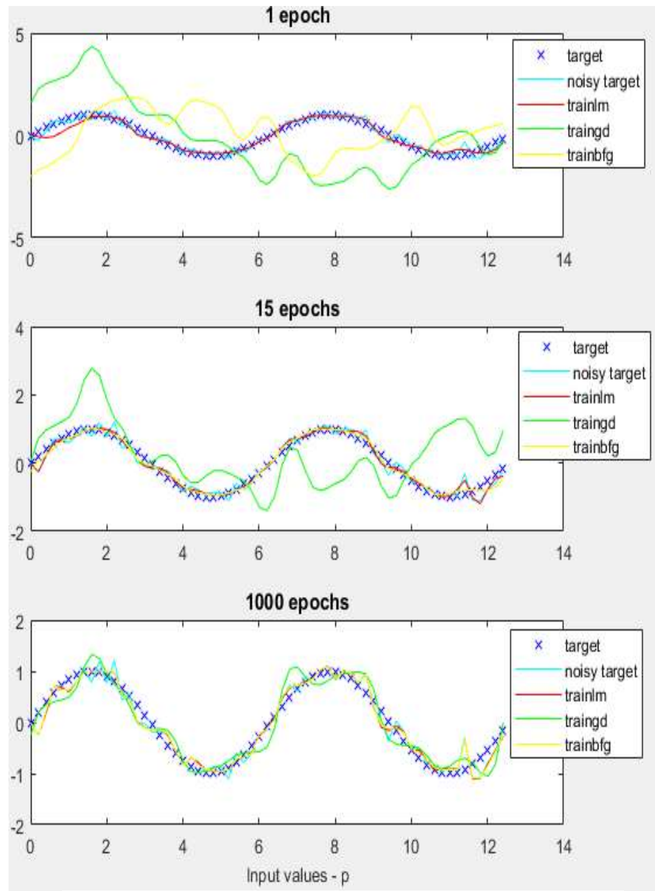

Fig.2.Curve fitting for various algorithms with added noise

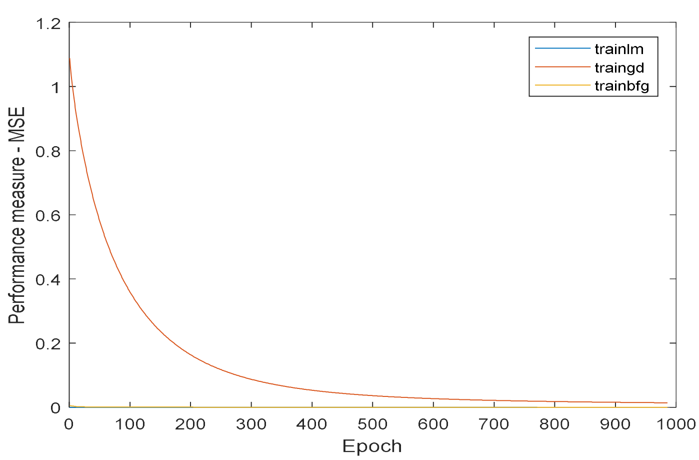

Fig.3.Performance measures comparison on training

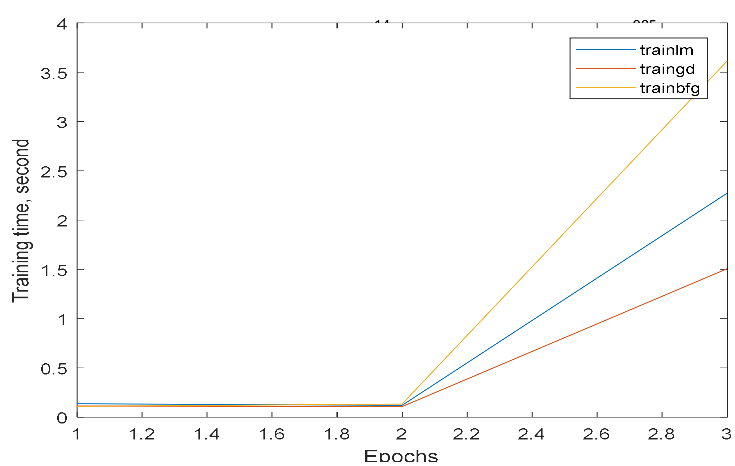

Fig.4.Training time comparison

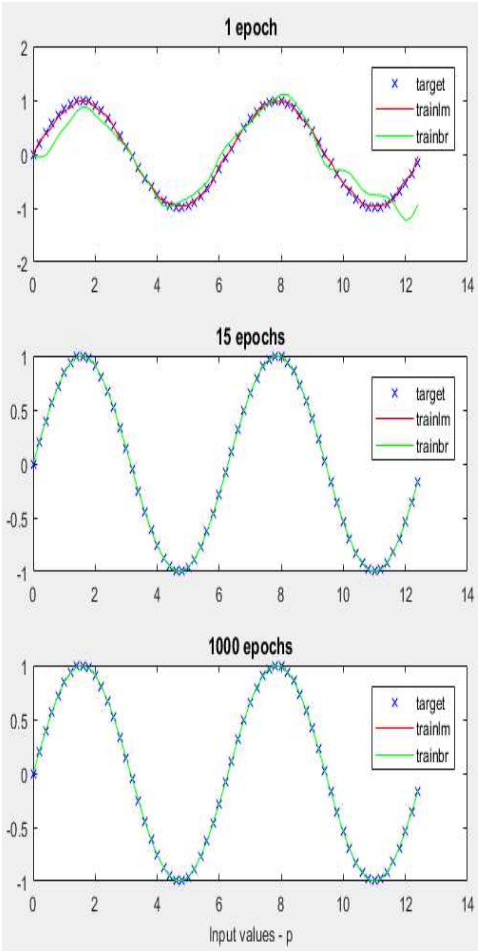

Fig.5..Curve fitting for trainlm vs trainbr (Neurons=20)

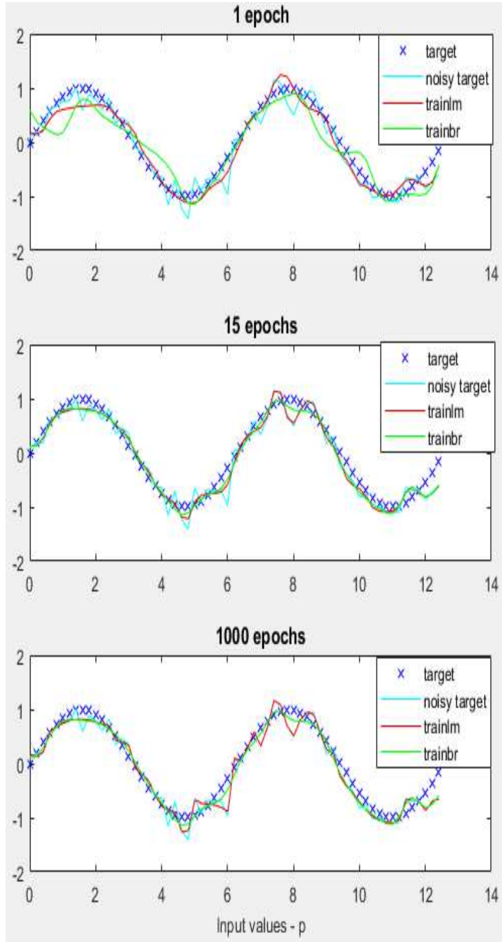

Fig.6.Curve fitting for trainlm vs trainbr with noise (Neurons =20)

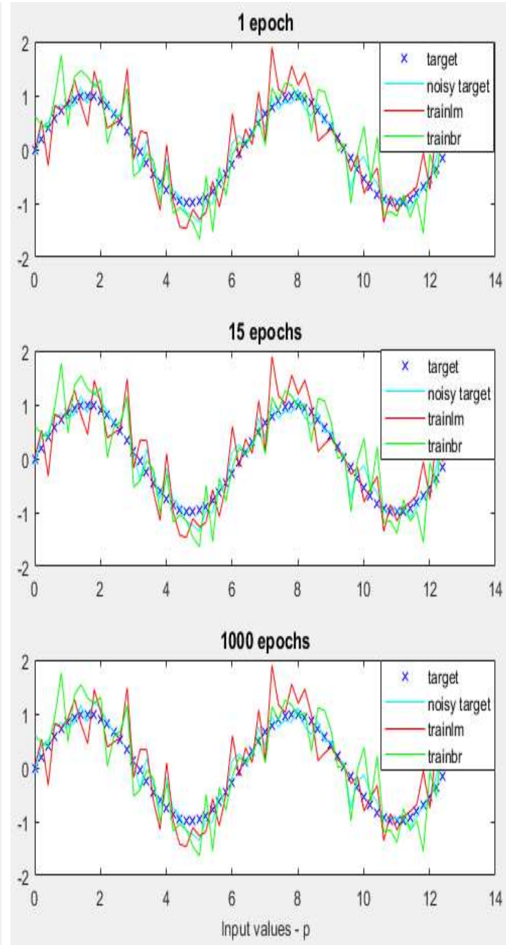

Fig.6.Curve fitting for trainlm vs trainbr with noise and overparameterization (Neurons=400)

The above figures summarise the differences regard to the performance of the chosen algorithm, it´s training time, the changes in Mean Squared Error value with respect to different Epoch values.

Discussion:

Trainlm fits close with a minimum mean squared error with the target attributes for a given set of data. This is a case for all the three different Epoch values.

- With increased Epoch, all of the algorithms are performing better with a sharp decrease in the mean squared error except the gradient descent algorithm ('traingd').
- Training the network takes more time for 'trainbfg' algorithm in comparison to other algorithms with increased values of epoch.
- The performance measure Mean squared Error value for 'trainlm' algorithm is achieved minimum quickly with high accuracy for the whole range of given epoch values. Gradient descent algorithm ' perform less better than the other algorithms.
- After having added the noise data, the performance
- The performance of the algorithms are very similar to the one with no noise.
- Bayesian learning fits the target well where as the other learning algoritms fits the noise, though the convergence properties of the Bayesian looks very similar to the Levenberg-Marquardt optimisation technique as Bayesian uses the Levenberg-Marquardt as base algorithm.
- **Overparameterization**: Increasing the number of neurons does not have impact on the performance over the increase in the epochs converging to less number of epochs. LM overfits to the noisy training data which makes it generalizes poorly on test set. Though Bayesian approach did not improve with increase in the number of neurons, it performs better than LM because of its regularization leaving space for improvement.