EE 659 Neural Networks Assignment

Anupam Khandelwal

160108008

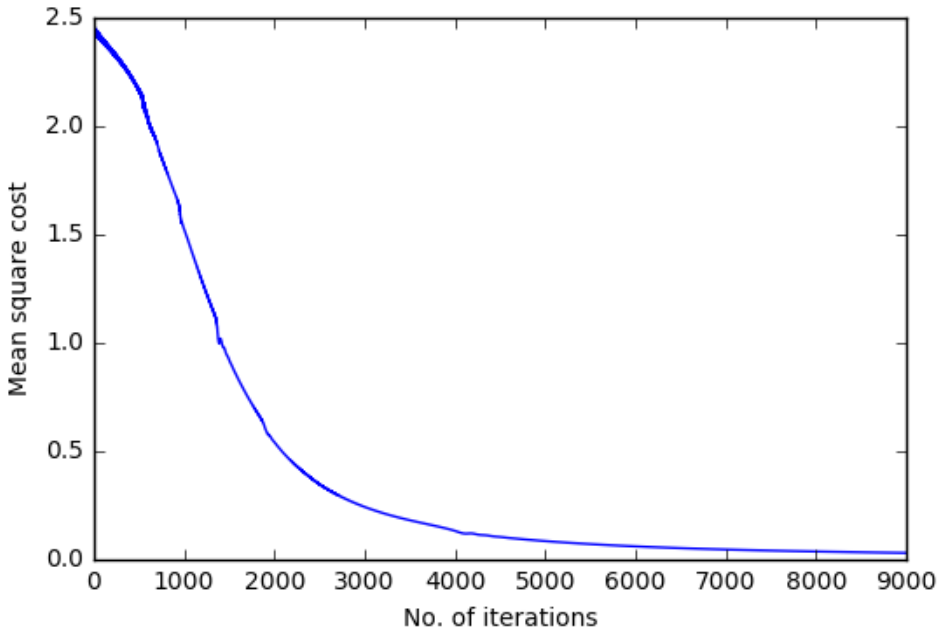
Q1.

- Data Generation:

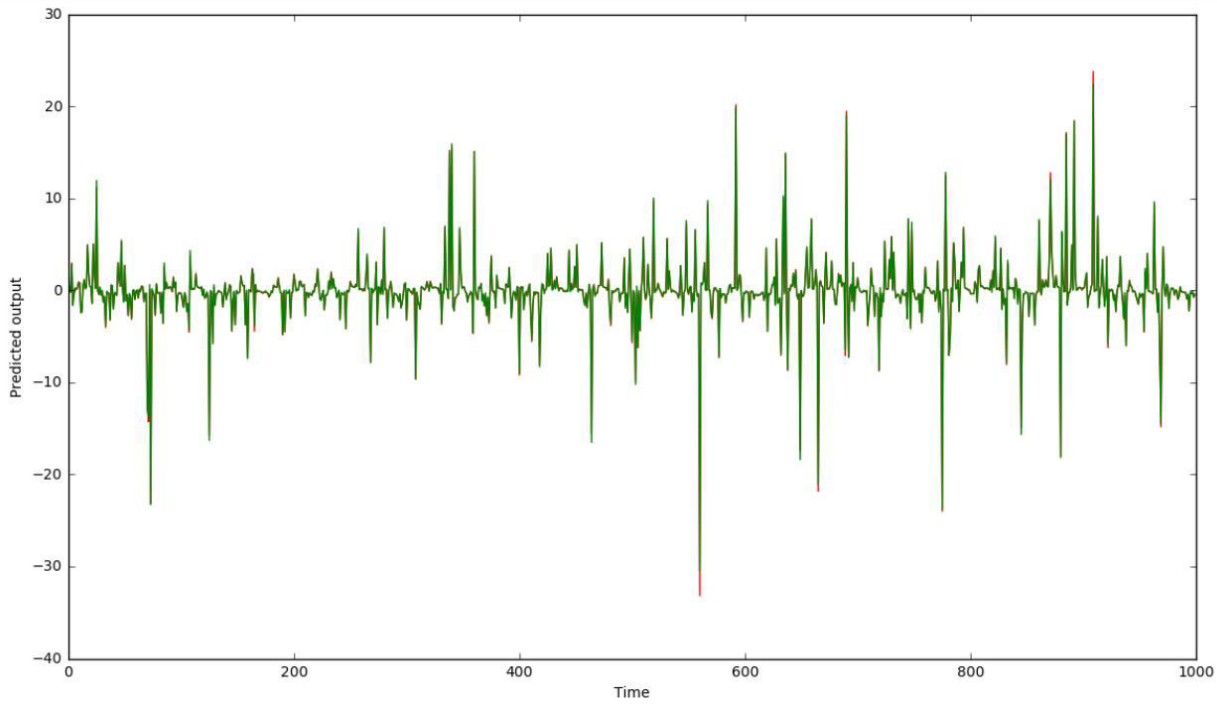
  1000 random samples were randomly generated for control input “u”.

  From the given recurrence relation in the problem output “y” was calculated to generate training data.

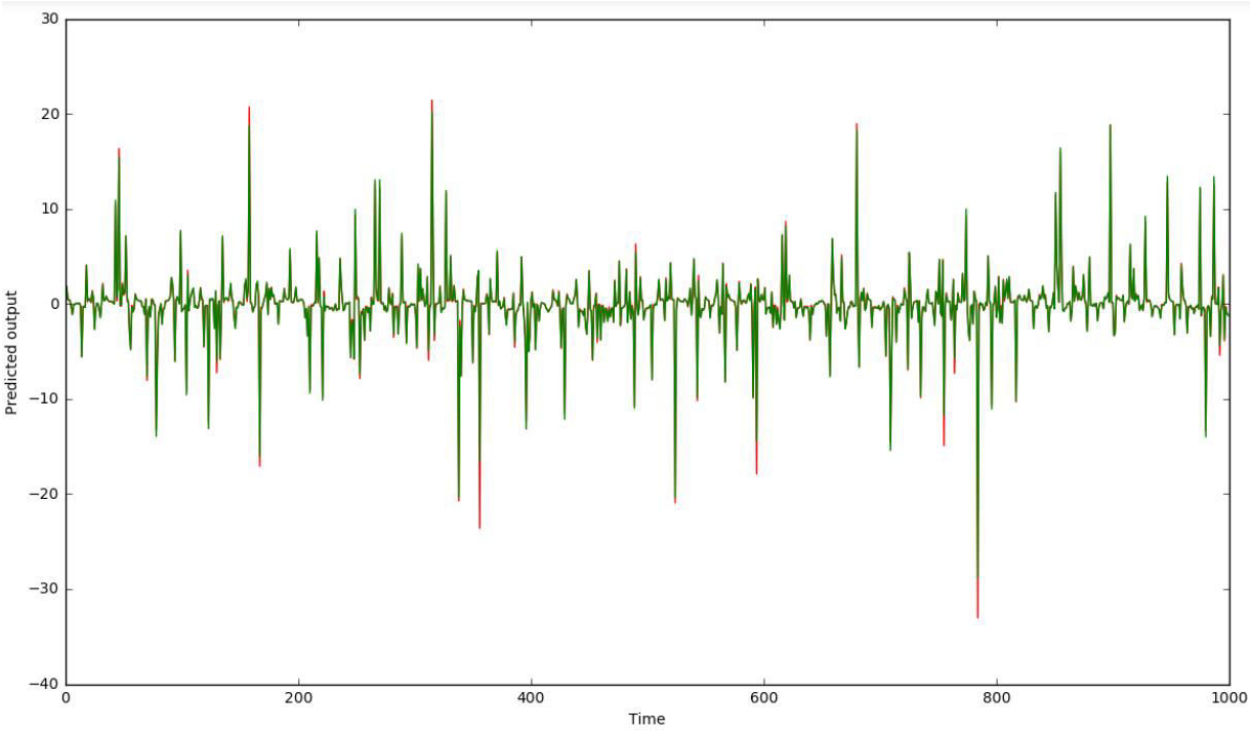
- Hyperparameters of neural network:
  - No. of hidden layers = 1
  - No. of nodes in hidden layer = 100
  - Activation function for hidden layer = tanh() activation
  - Activation function for output layer = linear() activation
  - No. of epochs = 10000
  - Learning rate = 0.03
  - Cost function = Mean squared error cost
- Training Results:
  - Final cost obtained = 0.0310



- Plot of training data:
    - Red curve corresponds to true output.
    - Green curve corresponds to predicted output.

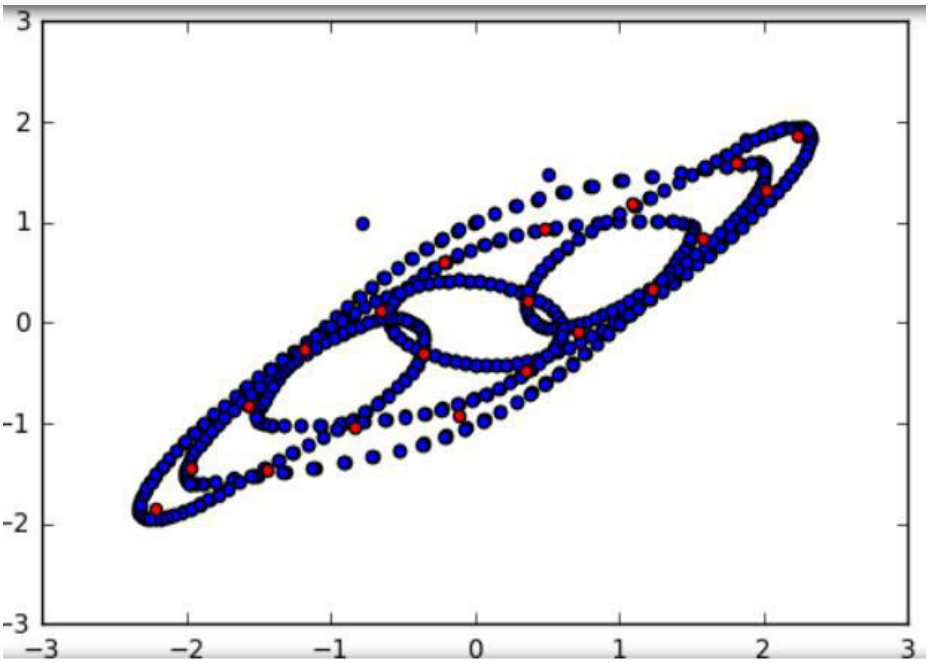


- Test data results:
    - Again 1000 samples were generated for test data.
    - Cost on test data = 0.1281



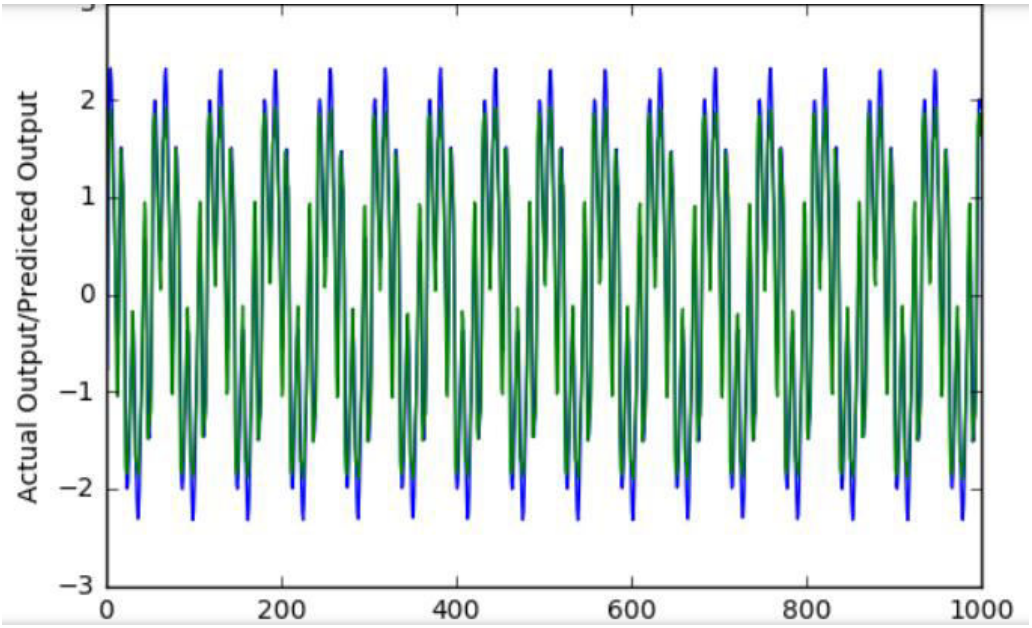
Red curve: Actual output ; Green curve: Predicted output

- Hyperparameters for radial basis function network:
    - No. of hidden layers = 1
    - No. of radial centers = 20
    - Activation for hidden layer = gaussian activation
    - No. of epochs = 15000
    - Learning rate = 0.01
    - For hybrid learning, K-means clustering was used for finding radial centers.



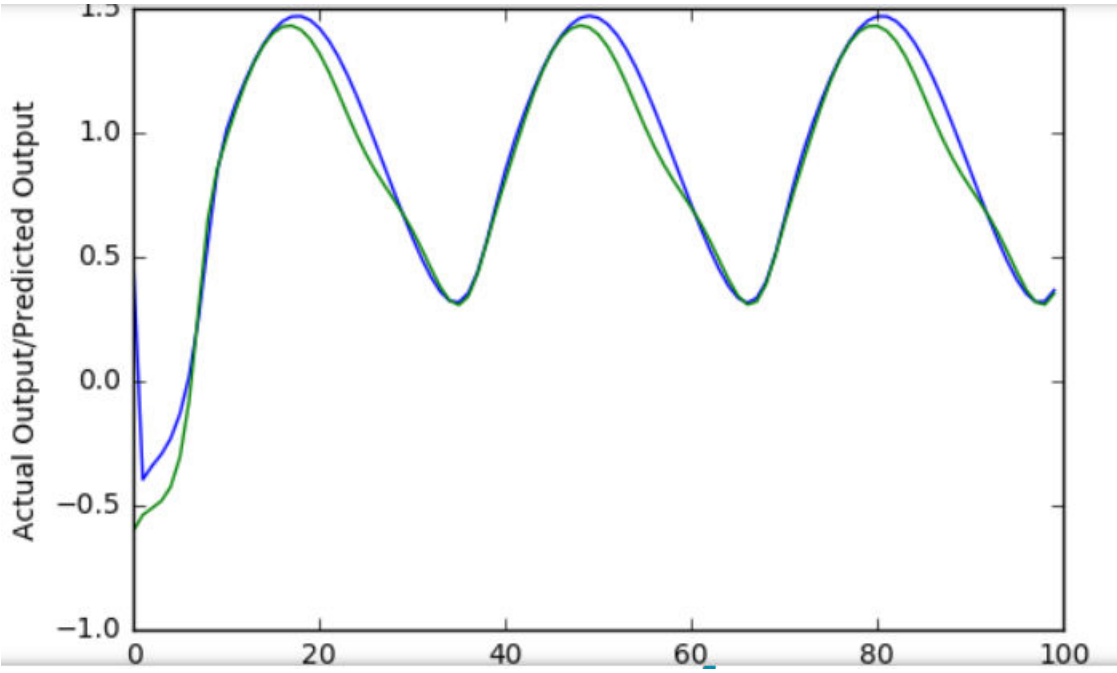
Red points correspond to calculated radial centers.

- Training data results:
  - Final training cost = 0.0454



Blue curve: Actual output ; Green curve: Predicted output

- Test data results:
    - 100 samples were created for test data.
    - Cost on test data = 0.8425



Blue curve: Actual output ; Green curve: Predicted output

Q2.

- Data Generation:

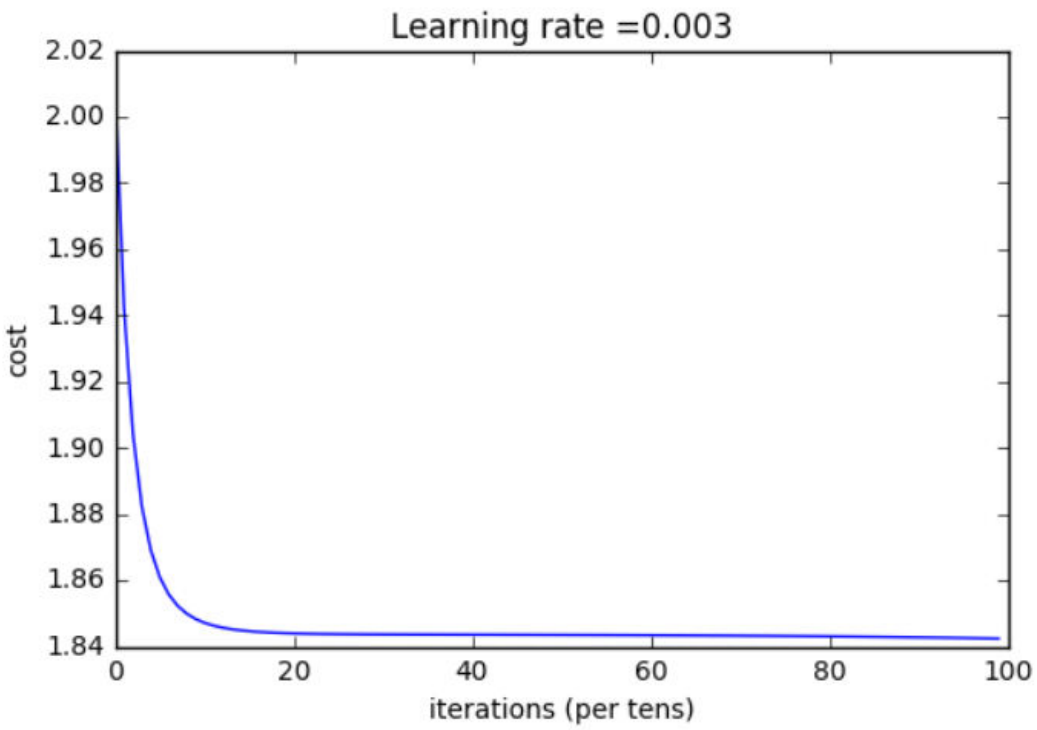
  1000 random samples were randomly generated for control input “u1 & u2”.

  From the given recurrence relation in the problem output “y1 & y2” was calculated to generate training data.

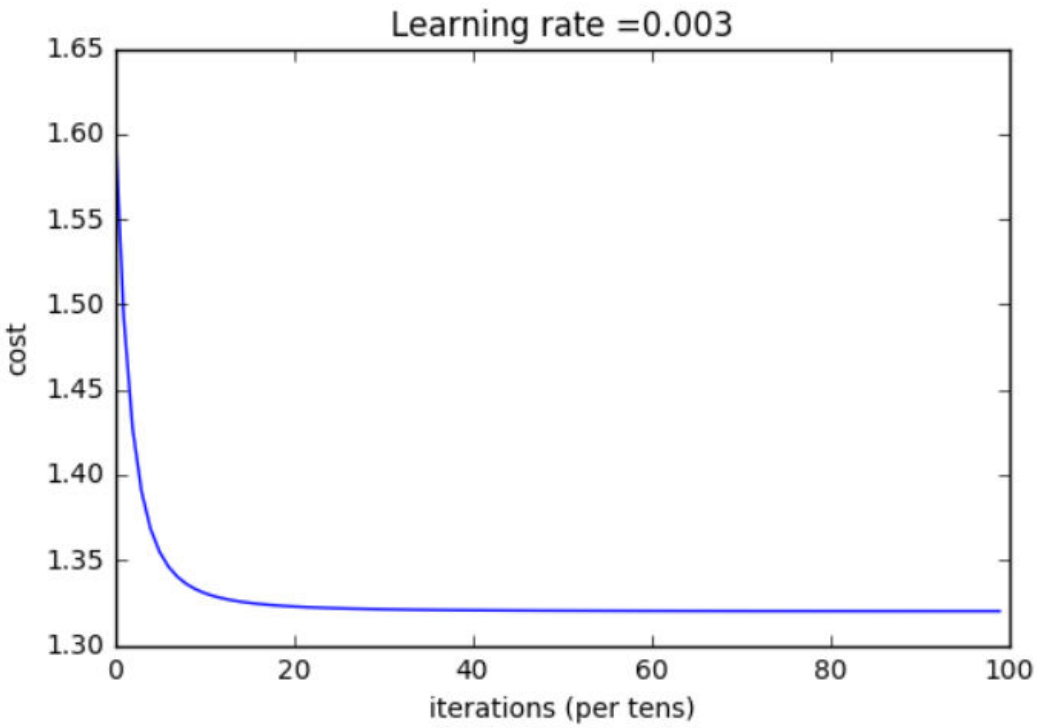
- Hyperparameters of neural network:
    - No. of hidden layers = 2
    - No. of nodes in hidden layer = 10, 10

- Activation function for hidden layer = sigmoid() activation
- Activation function for output layer = linear() activation
- No. of epochs = 10000
- Learning rate = 0.003
- Cost function = Mean squared error cost

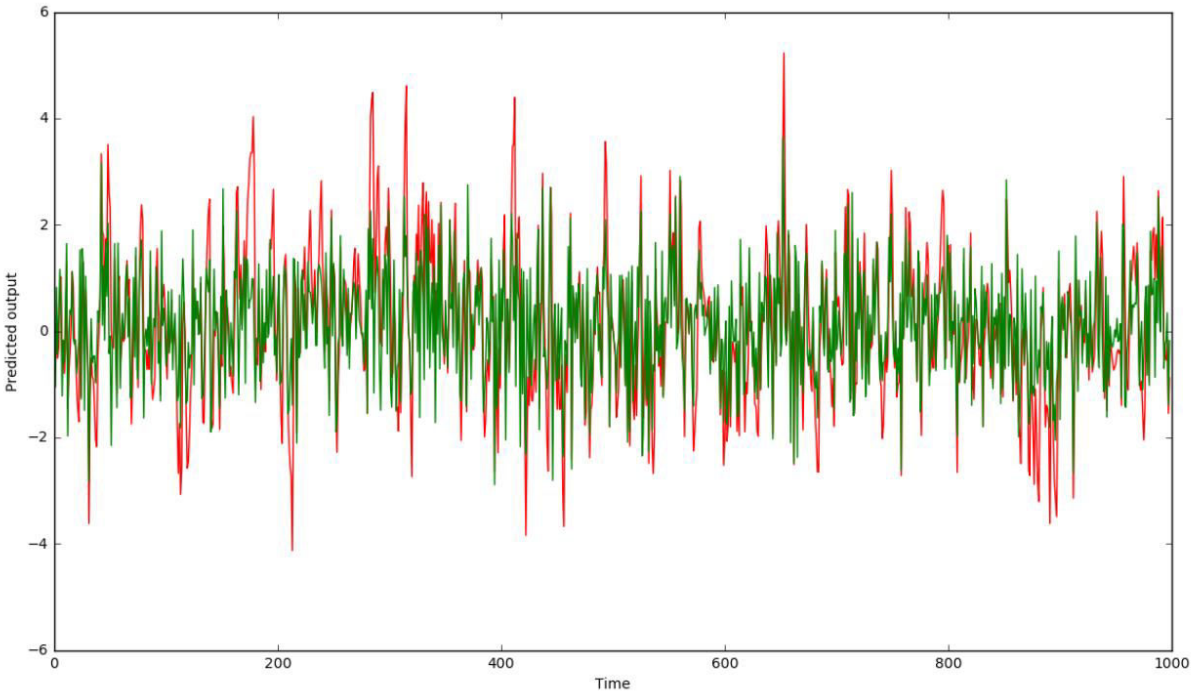
- Training Results:
  - Final cost obtained for first network = 1.8423
  - Final cost obtained for second network = 1.320



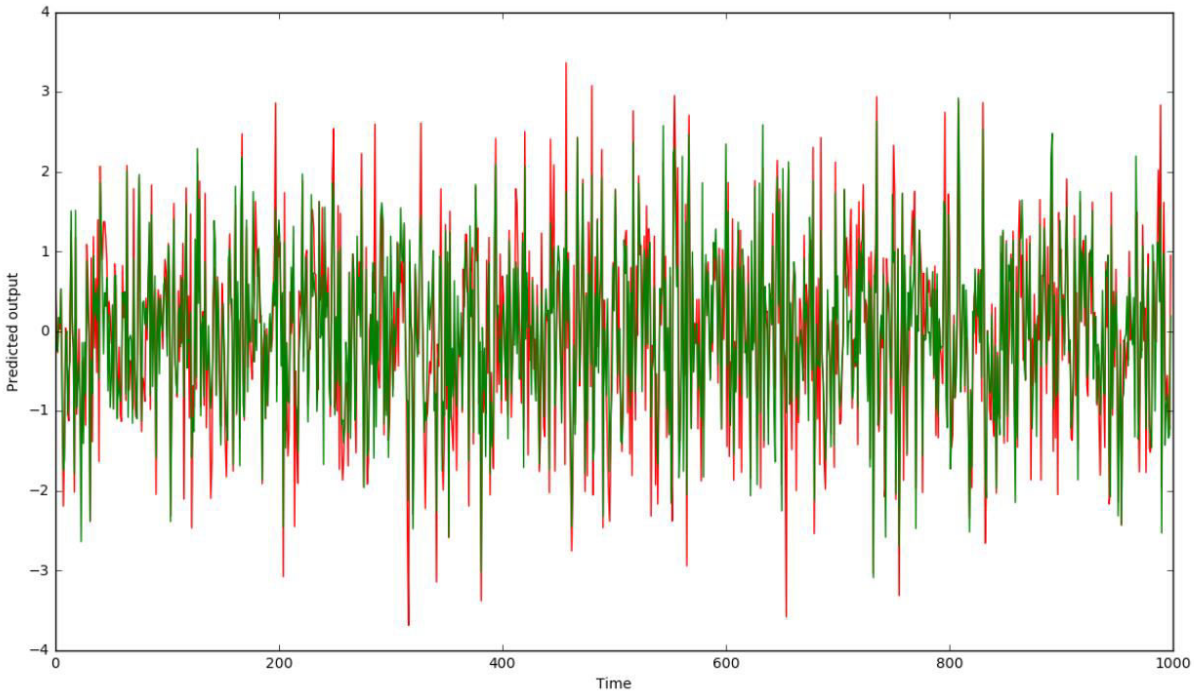
For first neural network corresponding to y1.



For second neural network corresponding to y2.

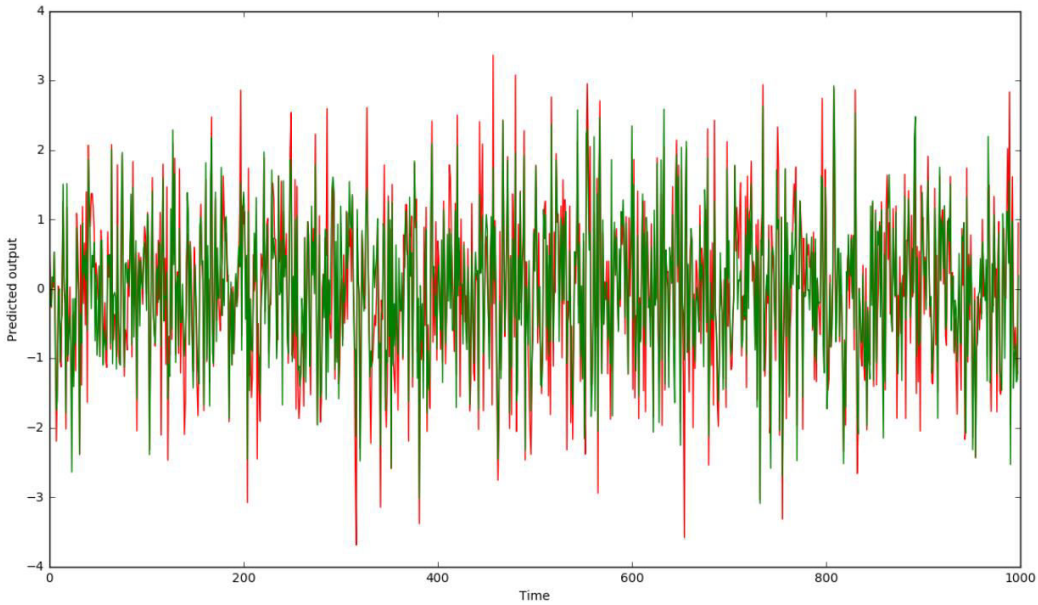


Red curve: Actual output ; Green curve: Predicted output for y1

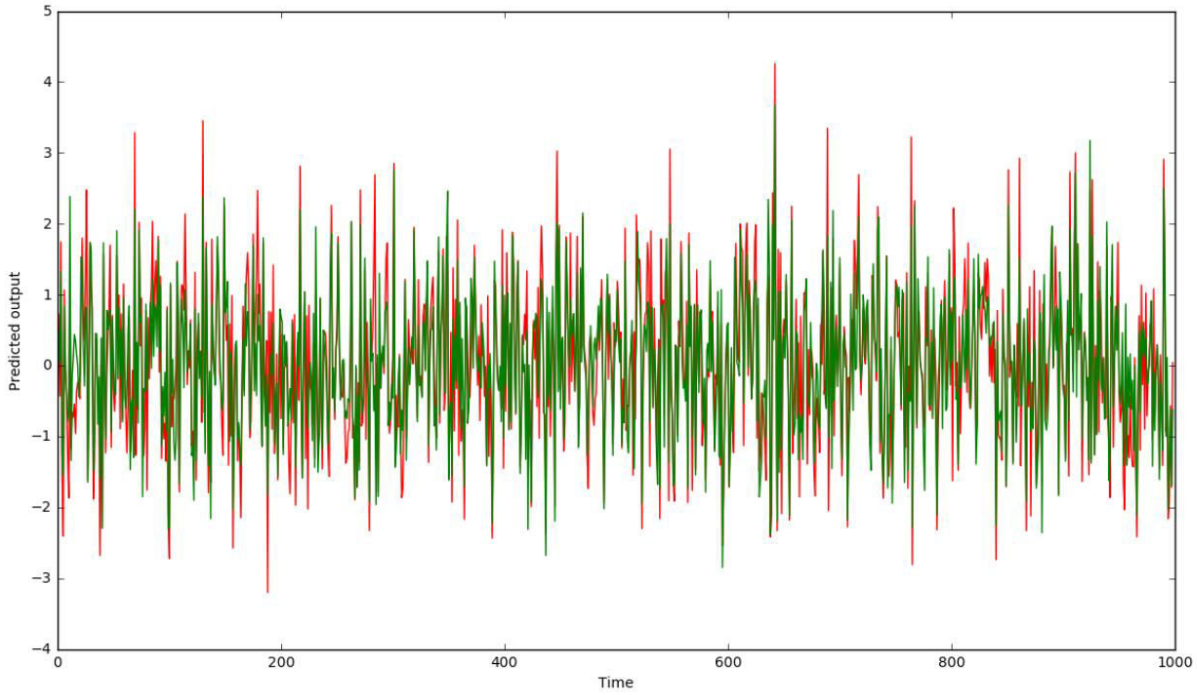


Red curve: Actual output ; Green curve: Predicted output for y2

- Test data results:
  - Computed cost for y1 = 2.144
  - Computed cost for y2 = 1.875



Red curve: Actual output ; Green curve: Predicted output for y1



Red curve: Actual output ; Green curve: Predicted output for y2