**REPORT**

DYNAMIC PERFORMANCE PREDICTION OF A AEROENGINE USING ARTIFICIAL NEURAL NETWORK

SUBMITTED BY

**RAKESH KIRAN**

**204103325**

**INTRODUCTION**

A multilayer feedforward artificial neural network with back propagation algorithm have been used to predict the exhaust gas temperature of an aero engine.

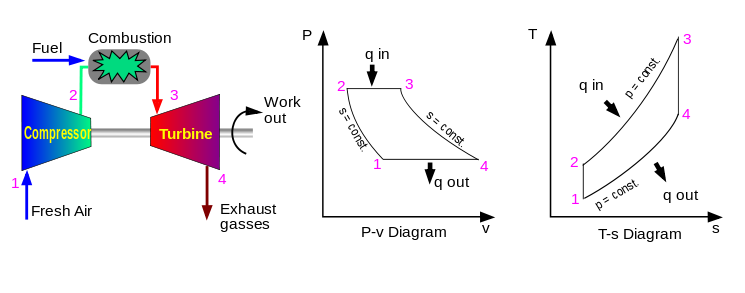
The EGT is the temperature of the gas turbine engine’s exhaust in degrees and indicates the performance status of the gas turbine engine’s design. High EGT values result in further deterioration and wear in the engines. In addition, pilots should pay attention to the EGT value, especially at takeoff, because exceeding the EGT limit may cause damage to the gas turbine engine components or reduce the service life of components. It is therefore important to keep the EGT as low as possible

The dataset was obtained from data article “Data regarding dynamic performance predictions of an aero engine” by Maria Grazia De Giorgi, Marco Quarta , Department of Engineering for Innovation, Via per Monteroni , University of Salento, 73100 Lecce, Italy.

The design of aero engine real-time control systems needs the implementation of machine learning based techniques. The lack of in-flight aero engine performance data is a limit for the researchers interested in the development of these prediction algorithms. Dynamic aero engine models can be used to overcome this .

So a artificial neural network was developed to predict the exhaust gas temperature for which the 10 input each for 1028 pattern is obtained from the dataset obtained from the published journal. And the result obtained from the neural network created was compared with the target values provided in the report itself.

* **MATHEMATICAL MODEL**

Considering the aero engine as a air breathing jet engine and follow the Brayton cycle.

Let the pressure ratio of the cycle be rp = P2/P1 = P3/P4

Exhaust gas temperature T4 = (rp)(1-k)/k  \* T3

k = specific heat ratio = Cp/Cv

T1 = inlet temperature of the gas (K)

T2 = temperature after isentropic compression (K)

T2  = T1 \* rp(k-1)/k

Heat input = Qin = m\*Cp\*(T3 – T2 ) kJ

m = mass flow rate of the gas (kg/s)

**DATA DESCRIPTION**

The data article presents the datasets on the turbojet was studied in the journal to implement the model in the Gas Turbine Simulation Program . The data file contained the data (time, altitude, ambient temperature difference from the one calculated by ISA, Mach number and engine rotational speed) from eight real flight missions used as input to validate the engine GSP model and the output exhaust gas temperature EGT predicted by the GSP software.

The file also reported the data of the 8 flight missions. From which data from one flight mission is taken as a input to the neural network coded. In each sheet there are 10 columns that are in order: Mach number (M), Atmospheric total temperature (Tt1) [K], atmospheric total pressure (pt1) [bar], air mass flow rate (Wa) [kg/s]; compressor pressure ratio [-]; rpm (N) [%]; turbine inlet total temperature (Tt4) [K], fuel mass flow rate (Wf) [kg/s],Thrust[kN], Exhaust gas temperature [K]. The published data article presents data regarding the performance of a turbojet that were predicted by the dynamic engine model that was built using the Gas turbine Simulation Program (GSP) software. The data were also used to implement an Artificial Neural Network (ANN) that predicts the in-flight aero engine performance, such as the Exhaust Gas Temperature (EGT).

**METHODOLOGY**

From the 1028\*10 dataset , for training of the neural network 700 train examples and 300 test set examples were taken each having 10 number of input parameters.

The result obtained is compared with target output provided in the dataset and analysis is done using calculating the mean square error for the training and test set separately. And the mean square error is calculated for the final output provided by the neural network and the target value.

The final output of the neural network is stored to a file and an array of error is created to analyse the performance of the neural network.

A plot is created between the MSE value and the number of iterations.

The optimal number of hidden layers , learning rate , momentum rate was found by trial and error method using the above mentioned error analysis.

A plot was created between number of hidden neurons and MSE of training set , from which it was identified using 10 hidden neurons gives the least MSE.

|  |  |  |  |
| --- | --- | --- | --- |
| **NUMBER OF HIDDEN NEURONS (H)** | **LEARNING RATE** | **MSE**  (Training set) | **MSE**  (Test set) |
| 2 | 0.01 | 0.0183 | 0.0198 |
| 2 | 0.5 | 3.7088\*10^-4 | 0.0018 |
| 2 | 0.85 | 1.3962\*10^-4 | 8.3949\*10^-4 |
| 2 | 0.2 | 3.9003\*10^-4 | 0.0013 |
| 5 | 0.01 | 0.0212 | 0.032 |
| 5 | 0.5 | 1.4406\*10^-4 | 0.0035 |
| 5 | 0.85 | 1.4639\*10^-4 | 0.0027 |
| 5 | 0.2 | 3.3730\*10^-4 | 0.0014 |
| 10 | 0.01 | 0.0171 | 0.0171 |
| 10 | 0.5 | 1.5424\*10^-4 | 0.002 |
| 10 | 0.85 | 1.5357\*10^-4 | 0.0015 |
| 10 | 0.2 | 2.3886\*10^-4 | 7.4985\*10^-4 |
| 15 | 0.01 | 0.0143 | 0.0163 |
| 15 | 0.5 | 1.4515\*10^-4 | 0.0015 |
| 15 | 0.85 | 1.4086\*10^-4 | 0.0024 |
| 15 | 0.2 | 2.7718\*10^-4 | 9.1848\*10^-4 |
| 20 | 0.01 | 0.0141 | 0.0141 |
| 20 | 0.5 | 1.7997\*10^-4 | 0.0018 |
| 20 | 0.85 | 1.2674\*10^-4 | 9.3535\*10^-4 |
| 20 | 0.2 | 1.6427\*10^-4 | 0.0016 |

**TOTAL NUMBER OF ITERATIONS = 10000**

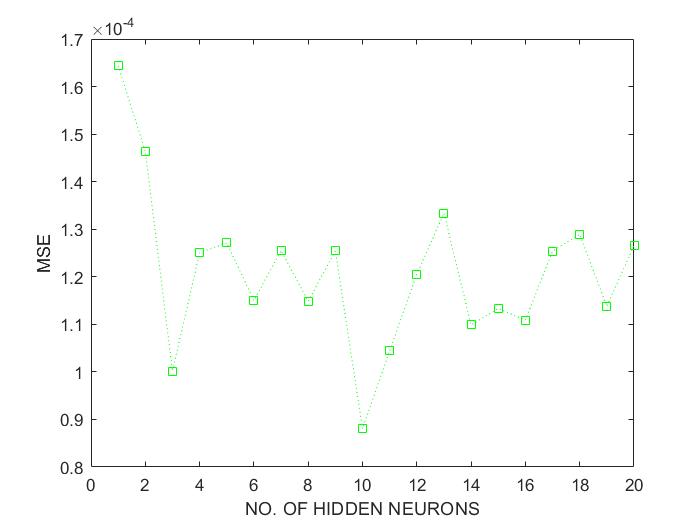
**MOMENTUM COEFFICIENT = 0**

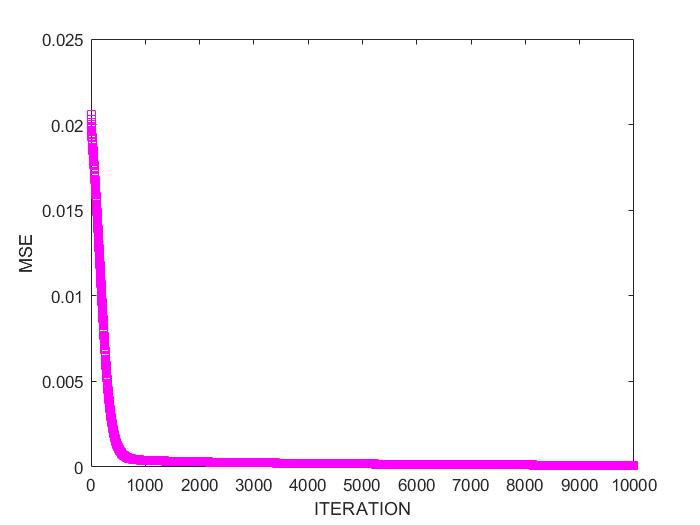
The test set error in the range of 10^-4 is rarely obtained for a combination of parameters. So the best value of parameters is selected based on the MSE of training set after running the code for finite number of times due to change in the output in range of 10^-5 which can be accounted for the randomness in weight matrix. Therefore the parameters obtained are:

* **NUMBER OF HIDDEN NEURONS = 10**
* **LEARNING RATE = 0.85**

|  |  |  |
| --- | --- | --- |
| **MOMENTUM COEFFICEINT** | **MSE**  (Training set) | **MSE**  (Test set) |
| 0.01 | 1.7527\*10^-4 | 0.005 |
| 0.25 | 1.2675\*10^-4 | 0.0021 |
| 0.5 | 1.3561\*10^-4 | 0.0058 |
| 0.75 | 9.5681\*10^-5 | 0.0011 |
| 0.95 | 1.0808\*10^-4 | 0.0097 |

* **MOMENTUM COEFFICIENT = 0.75**

****

****

**RESULT**

* Number of input parameters = 10
* Number of hidden neurons = 5
* Number of output neurons = 1
* Number of training patterns = 700
* Number of testing pattern = 300
* Learning rate = 0.85
* Momentum coefficient = 0.75
* Number of iterations = 10^4
* MSE (Training set ) = 3.7635\*10^-5
* MSE(Test set )

[Error in prediction] = 0.0036

* MSE(Final output) = 6.6057

**CONCLUSIONS**

* The result of the network is highly dependent on the dataset i.e if the target values are of the same order then error obtained was less. This was validated using different dataset like quality of vine prediction dataset, concrete strength prediction dataset.
* High care should be taken during data collection and arrangement of the important features and data points for a particular problem definition.
* Always the test set error is larger than the training set error and both of them was having very near values when learning rate was chosen very low.
* With increase in number of iteration the MSE value increases and the finite value of error in prediction is due to large magnitude of target output.
* The results vary time to time in order of 10^-4 can be attributed the random initialization of connection weights.
* Use of momentum terms has increased the convergence rate.
* Use of more number of training examples improves the accuracy.
* Normalization and randomization of the input data improves the outcome of the neural network.
* The appropriate network parameters like learning rate , number of hidden neurons , momentum coefficient should be selected to get better results.
* Around 1000 iterations MSE reaches its minimum at a faster rate, after that there no appreciable decrease in MSE.
* The obtained MSE of final output is large because the target values were in the order ( 10^2-10^3 ).