

Comparative Analysis of Neural Network Models for Petroleum Products Pipeline Monitoring

Udoinyang G. Inyang¹, Samuel S. Udoh¹, Oluwole C. Akinyokun²

¹Department of Computer Science, University of Uyo, Uyo, Nigeria

²Department of Computer Science, School of Sciences, Federal University of Technology, Akure, Nigeria

Correspondence: Oluwole C. Akinyokun, Department of Computer Science, School of Sciences, Federal University of Technology, Akure, Nigeria.

Received: September 28, 2016 Accepted: November 17, 2016 Online Published: April 6, 2017

doi:10.11114/set.v4i1.2340

URL: <https://doi.org/10.11114/set.v4i1.2340>

Abstract

In recent years, Neural Network (NN) has gained popularity in proffering solution to complex nonlinear problems. Monitoring of variations in Petroleum Products Pipeline (PPP) attributes (flow rate, pressure, temperature, viscosity, density, inlet and outlet volume) which changes with time is complex due to existence of non linear interaction amongst the attributes. The existing works on PPP monitoring are limited by lack of capabilities for pattern recognition and learning from previous data. In this paper, NN models with pattern recognition and learning capabilities are compared with a view of selecting the best model for monitoring PPP. Data was collected from Pipelines and Products Marketing Company (PPMC), Port Harcourt, Nigeria. The data was used for NN training, validation and testing with different NN models such as Multilayer Perceptron (MLP), Radial Basis Function (RBF), Generalized Feed Forward (GFF), Support Vector Machine (SVM), Time Delay Network (TDN) and Recurrent Neural Network (RNN). Neuro Solutions 6.0 was used as the front-end-engine for NN training, validation and testing while My Structured Query Language (MySQL) database served as the back-end-engine. Performance of NN models was measured using Mean Squared Error (MSE), Mean Absolute Error (MAE), Correlation Coefficient (r), Akaike Information Criteria (AIC) and Minimum Descriptive Length (MDL). MLP with one hidden layer and three processing elements performed better than other NN models in terms of MSE, MAE, AIC, MDL and r values between the computed and the desired output.

Keywords: pipeline, monitoring, neural network, multi-layer, perceptron

1. Introduction

Petroleum Products Pipelines (PPP) provide a major transportation infrastructure for conveying Premium Motor Spirit (PMS), Dual Purpose Kerosene (DPK), Automotive Gas Oil (AGO) and other petroleum products from refineries to the marketing and distribution depots. Abnormalities in flow rates of these products are a major sign of products leakages. The aftermath of petroleum products leakages include environmental pollution, degradation of soil fertility, poor agricultural production, fire outbreak, loss of life and property as well as serious threats to national economy and security. Researches in PPP monitoring and response to anomalies have been carried out in (Udoh, 2009; Akinyokun and Inyang, 2013; Inyang and Akinyokun, 2014). The existing systems for monitoring PPP are limited by lack of a systematic way of learning from previous data and generalizing into unseen patterns. NN offers a plausible solution to this learning and pattern recognition problem. NN is presented in (Akinyokun and Inyang, 2013), as a mathematical model of biological nervous systems, having the capabilities of fault tolerance, parallelism, learning from training data, recalling memorized information and generalizing to the unseen patterns. NN are applied to an increasing number of complex real life problems. They offer ideal solutions to signal recognition problems as well as prediction and system modeling where the physical processes are not understood or are highly complex.

In this paper, review of some related works on PPP monitoring are presented in Section 2, Section 3 addresses the methodology and design of NN models for monitoring PPP. Results and research findings are presented in Section 4 while discussion and recommendations for further research are presented in Section 5 and Section 6 respectively.

2. Related Works

Petroleum products pipeline monitoring and leak detection technique based on pipeline state equations was proposed in

(Chis, 2007). State equations depend on the type of fluid being modeled. No one equation fully describes the variety of products that are shipped in pipeline. Therefore, different equations are needed to modeled different products, thereby making the task of monitoring of PPP very tedious and difficult. In (Nikles *et al.*, 2009), PPP monitoring system was presented using fibre optics. The occurrence and location of leakages were determined by analysis of the temperature profiles and the achievable detection limits were within 0.01% of the total throughput for oil leaks. The work suffers from false detection and localization of oil spills due to inefficient tools for analysis of pressure and temperature profiles. (Cafaro *et al.*, 2010) presented oil-derivatives pipeline logistics system with the aim of evolving a computational tool to manage oil products pipeline in a reliable and cost effective manner. The work was limited by lack of capability for efficient priority scheduling for pipes and products delivery. The system could not learn patterns from previous petroleum products data. (Jiangwen *et al.*, 2012) used a hierarchical leak detection equations based on time difference of signal arrival to monitor and localize defects on PPP. Acoustic sensors were placed at fixed points along the pipeline to measure and calculate the relative time difference of signal waves transmitted in the pipeline. It was difficult to distinguish the leak patterns from the signal waves, this led to high probability of false detection of pipeline defects. Pipeline leak detection and control using NN technique was presented in (Adewuji and Okelola, 2013). The work was motivated by the need to minimize the increasing oil spillage losses through efficient monitoring and timely detection of oil spillage events on PPP. A four-layer NN was designed comprising the input layer, two hidden layers and an output layer. The input layer accepted PPP data such as pressure, flow rates and temperature to estimate hydraulic behaviour of the products being transported. The second and third layers were used for processing of the input while the output was used for estimation of the result. Hyperbolic transfer function was used in the hidden layers and sigmoid transfer function was used in the output layer. Back propagation algorithm was used in the training of the network. The work contributed to knowledge an intelligent approach for monitoring PPP, but was limited by the use of few PPP attributes in the monitoring and training processes, thereby hindering the exploitative and learning power of NN in complex scenarios. In (Akinyokun and Inyang, 2013), a framework for oil spillage risk management using neuro-fuzzy-genetic techniques was presented. The aim was to predict the occurrence of oil spillage and manage the resultant risk. A neuro-fuzzy-genetic model for oil spillage management was designed. Adaptive neuro-fuzzy inference engine was designed for the neuro-fuzzy-genetic model using Mamdani inference mechanism. A software system driven by neuro-fuzzy-genetic model was designed using 11 Ants Analytics, Matrix Laboratory (MatLab) and Microsoft Access database. The functionality of the system was assessed using oil spillage data. The system contributed intelligent model for assessment and management of oil spillage but was limited by its lack of capability to monitor the attributes that led to oil spillage. This paper attempts to overcome the above limitations of the existing works by comparing NN models with a view of selecting the best model for knowledge acquisition and pattern recognition in complex data of PPP oil spillage induced parameters.

3. Methodology

The methodological workflow adopted in this work spans data collection, description and preprocessing, design of NN model, training, validation and testing of designed NN model and performance evaluation.

3.1 Data Description and Preprocessing

A set of 2290 data was collected from Pipeline and Products Marketing Company (PPMC) in Port Harcourt Area office, Nigeria. The data include attributes from Port Harcourt and Aba (PHAB) pipeline segment, Port Harcourt and Enugu (PHEN) pipeline segment as well as Enugu and Markurdi (ENMK) segment. The data attributes collected comprised Inlet Pressure (IPS), Inlet Valve Size (IVS), Inlet Volume (IVM), Inlet Temperature (ITP), Viscosity (VSC), Density (DTY), Pipe Diameter (PDM), Pipe Length (PLT), Corrosion (CRN), Transient Pressure (TPS), Outlet Valve Size (OVS), Outlet Temperature (OTP), Outlet Pressure (OPS), Outlet Volume (OVM) and Flow Rate (FLR) which was used as output of the neural network. The 2290 dataset was divided into three parts for training, validation and testing of the system in the ratio of 8:1:1 which translates into 1832 data set for training and 229 dataset each for validation and system testing. The input and output data were subjected to different NN models such as MLP, RBF, GFF, SVM, TDN and RNN invoked from Neuro Solutions 6.0 to facilitate the assessment and selection of best NN model for PPP monitoring based on MSE, MAE and r values.

3.2 Design of Neural Network for PPP Monitoring.

The design of NN for monitoring PPP was carried out using a three layered feed forward MLP architecture. The input layer consists of fourteen (14) data attributes of PPP namely: IPS, IVS, IVM, ITP, VSC, DTY, PDM, PLT, CRN, TPS, OVS, OTP, OPS and OVM. The hidden layer is made of six (6) nodes which receive and process the signal from the input layer using the connection weights and hyperbolic transfer function. The output layer is made of one (1) node. It computes the Flow rate (FLR) of PPP using the connection weights of the hidden layer and sigmoid transfer function. The computed output is compared with the desired output and the difference is the error term which is used to adjust the connection weights by means of backpropagation algorithm. The process is repeated until the error term is within the

acceptable threshold. The NN architecture for PPP monitoring is depicted in Figure 1.

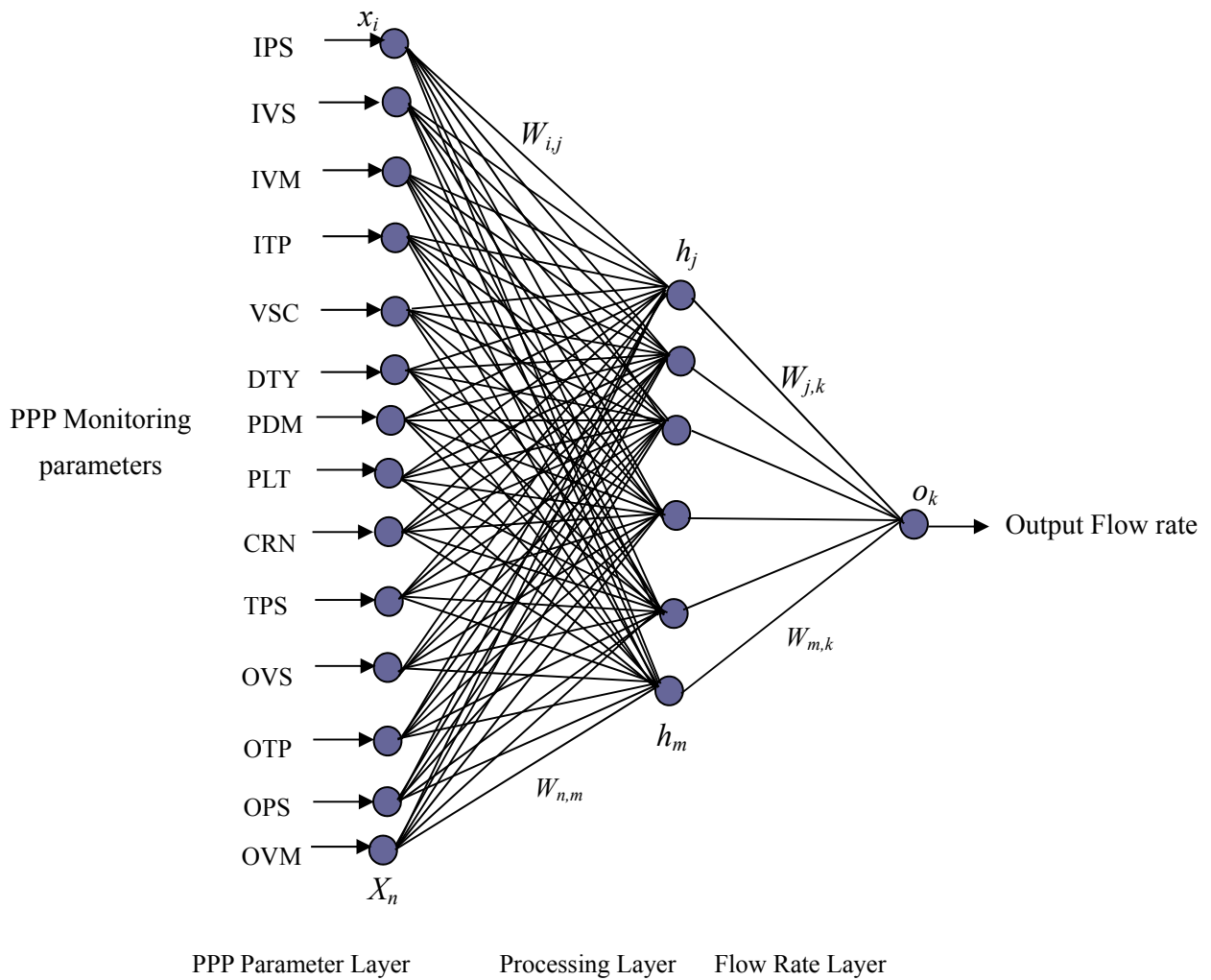


Figure 1. Neural Network of PPP Monitoring

From the multilayer feed-forward schema of Figure 1, the network layers are represented as follows:

- a. PPP parameter layer X_i : $i = 1, 2, \dots, n$
- b. Processing layer h_j : $j = 1, 2, \dots, m$
- c. Flow rate layer O_k : $k = 1, 2, \dots, p$

Let h_j^* and o_k^* be the pre-output (output to be processed by transfer function) at the hidden and output layers. Let $W_{i,j}$ be the matrix of weights on the connection from the i th node in the input layer to the j th node in the hidden layer.

A system of equations is formulated for the input and hidden layers as follows:

$$\begin{aligned}
 w_{1,1}x_1 + w_{1,2}x_2 + \dots + w_{1,n}x_n &= h_1^* \\
 w_{2,1}x_1 + w_{2,2}x_2 + \dots + w_{2,n}x_n &= h_2^* \\
 &\vdots \\
 w_{m,1}x_1 + w_{m,2}x_2 + \dots + w_{m,n}x_n &= h_m^*
 \end{aligned} \tag{1}$$

In terms of matrices, the system of equations is represented as: $Wx = h^*$, where W is the matrix of weights, x is the activity detection input vector and h^* is the hidden layer pre-output as shown in Equation 1 which is compressed in Equation 2 as follows:

$$\begin{matrix} & W & & x & & h^* \\ & & & & & \\ \left[\begin{matrix} w_{1,1} & w_{1,2} & \cdots & w_{1,n} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,n} \\ & & \vdots & \\ w_{m,1} & w_{m,2} & \cdots & w_{m,n} \end{matrix} \right] & \left[\begin{matrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{matrix} \right] & = & \left[\begin{matrix} h_1^* \\ h_2^* \\ \vdots \\ h_m^* \end{matrix} \right] & & (2)
 \end{matrix}$$

$$\sum_{j=1}^m \sum_{i=1}^n W_{i,j} x_i = h_j^* \tag{3}$$

The actual output of hidden layer node h_j is obtained by subjecting the pre-output in Equation 3 to the hyperbolic transfer function as shown in Equation 4

$$h_j = \frac{e^{h_j^*} - e^{-(h_j^*)}}{e^{h_j^*} + e^{-(h_j^*)}} \tag{4}$$

Similarly, the computation in the output layer node is performed. Let $W_{j,k}$ be a matrix of weights that connects j th node in the hidden layer to the k th-output layer, The output layer equation is formulated as shown in Equation 5, composed in vector form as given in Equation 6 and compressed as shown in Equation 7

$$w_{1,1}h_1 + w_{1,2}h_2 + \cdots + w_{1,m}h_m = o_k^* \tag{5}$$

$$\left[\begin{matrix} w_{1,1} & w_{1,2} & \cdots & w_{1,m} \end{matrix} \right] \left[\begin{matrix} h_1 \\ h_2 \\ \vdots \\ h_m \end{matrix} \right] = o_k^* \tag{6}$$

$$\sum_{k=1}^p \sum_{j=1}^m w_{j,k} h_j = o_k^* \tag{7}$$

The actual node value of the output layer node o_k is obtained by subjecting the pre-output in Equation 7 to the sigmoid transfer function as shown in Equation 8

$$o_k = \frac{1}{1 + e^{-(o_k^*)}} \tag{8}$$

The backpropagation learning procedure employed in (Udoh, *et al.*, 2012; Udoh, 2016) was used to train the network to facilitate knowledge acquisition and pattern recognition.

4. Results

The results of NN models (MLP, RBF, GFF, SVM, TDN, RNN) based on Mean Squared Error (MSE), Mean Absolute Error (MAE) and correlation coefficient (r) performance indices are depicted in Table 1. These models were built and tested for PPP monitoring using Network Building Window of Neuro Solutions 6.0 as depicted in Figure 2. Inputs to the system were IPS, IVS, IVM, ITP, VSC, DTY, PDM, PLT, CRN, TPS, OVS, OTP, OPS and OVM while Flow rate (FLR) served as the output as shown in Figure 3. Further investigation into NN parameters using MLP is depicted in Figure 4 and summarized in Table 2.

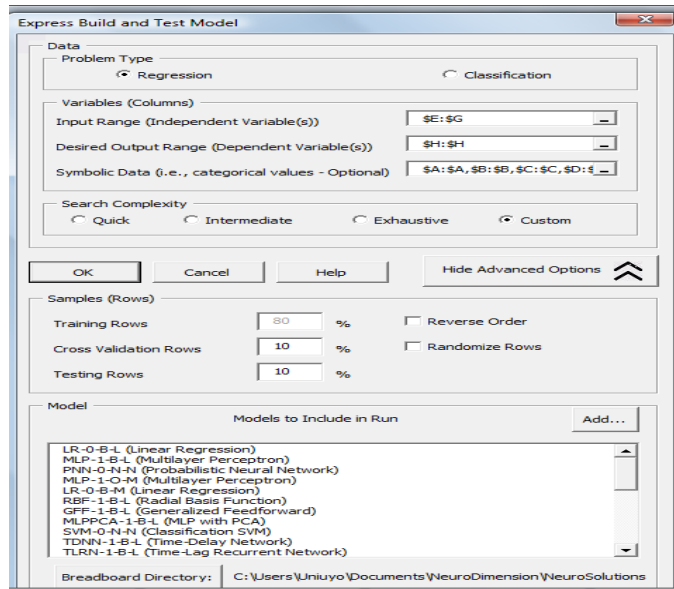


Figure 2. Network Building Window

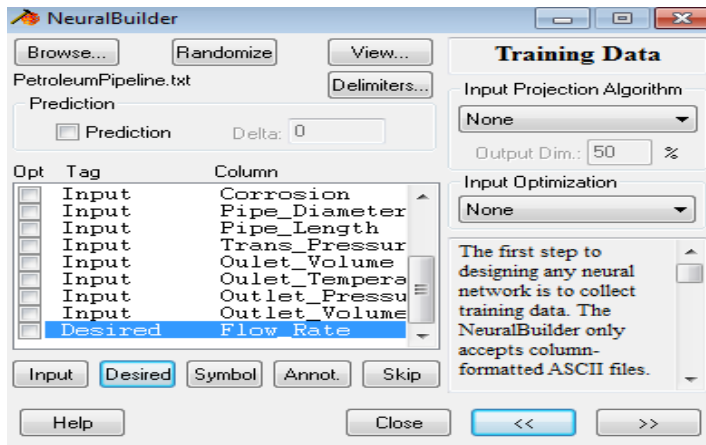


Figure 3. Input and Desired Output Panel

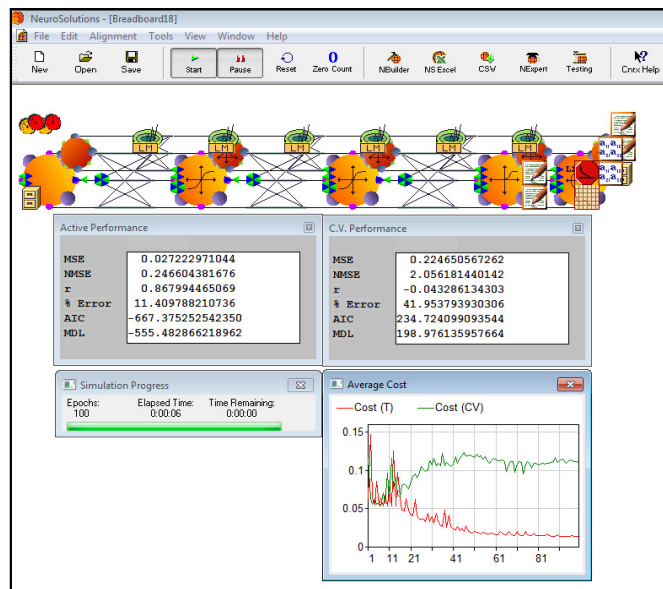


Figure 4. Performance Measures Window

Table 1. Performance of NN Models on Dataset.

MODEL NAME	Training			Cross Validation			Testing			Ave.	Ave.	Ave. r
	MSE	MAE	r	MSE	MAE	r	MSE	MAE	r	MSE	MAE	
Multilayer Perceptron	0.0002	0.0083	0.998	0.0002	0.0096	0.998	0.0003	0.0102	0.997	0.0002	0.009	0.997
Radial Basis Function	0.0022	0.0233	0.978	0.0004	0.017	0.996	0.0005	0.018	0.995	0.0010	0.019	0.990
Generalized Feed	0.0004	0.0096	0.997	0.0001	0.0087	0.994	0.0003	0.0108	0.997	0.0003	0.010	0.996
Classification SVM	0.1068	0.2427	0.656	0.0792	0.194	0.664	0.1145	0.2502	0.567	0.1002	0.229	0.629
Time Delay Network	0.0005	0.0046	0.998	0.0006	0.0124	0.993	0.0007	0.0128	0.994	0.0006	0.010	0.995
Recurrent Network	0.0141	0.0823	0.888	0.0048	0.0544	0.964	0.0077	0.0697	0.950	0.0089	0.069	0.934

MLP performed better than other NN models. It has the least average MSE of 0.0002, average MAE of 0.009 and highest average correlation coefficient of 0.997. Subsequent investigation of NN parameters for monitoring PPP was carried out using MLP. Sessions of training with varying number of MLP components were carried out using hyperbolic and sigmoid transfer functions in the hidden and output layers respectively. Performance measures were MSE, Normalized Mean Squared Error (NMSE), Correlation Coefficient (r), Percentage Error (%), Akaike Information Criteria (AIC) and Minimum Descriptive Length (MDL). The summary of the model performance metrics are presented in Table 2

Table 2. Summary of NN Performance Metrics

Hidden Layer	PEs of Hidden layer	MSE		NMSE		r		% Error		AIC		MDL	
		Train	CV	Train	CV	Train	CV	Train	CV	Train	CV	Train	CV
1	2	0.0038	0.0024	0.0864	0.0593	0.9614	0.9725	26.45	4.81	-1447.7	-176.2	-1439.1	-178.9
	3	0.0036	0.0061	0.0546	0.0367	0.9723	0.9818	14.32	4.06	-1193.3	-136.2	-1180.6	-140.3
	4	0.0043	0.0027	0.0973	0.0640	0.9533	0.9679	26.39	5.05	-1397.1	-153.7	-1380.5	-158.9
	5	0.0039	0.0025	0.0899	0.0609	0.9557	0.9691	24.45	5.31	-1407.8	-145.3	-1387.3	-151.8
	6	0.0041	0.0026	0.0936	0.0622	0.9551	0.9689	26.29	5.10	-1387.2	-134.6	-1362.7	-142.4
Mean of layer 1		0.0051	0.0033	0.0844	0.0566	0.9596	0.9720	23.58	4.87	-1366.6	-149.2	-1350.0	-154.5
2	2	0.0089	0.0084	0.2038	0.2032	0.9510	0.9497	42.21	6.58	-1209.8	-123.5	-1196.4	-127.9
	3	0.0084	0.0073	0.1908	0.1775	0.9440	0.9453	43.92	6.34	-1205.3	-106.1	-1183.2	-113.1
	4	0.0071	0.0061	0.1607	0.1458	0.9450	0.9471	39.09	6.32	-1224.3	-86.54	-1192.2	-96.9
	5	0.0063	0.0052	0.1435	0.1269	0.9480	0.9528	36.46	6.19	-1224.5	-61.12	-1180.1	-25.2
	6	0.0054	0.0043	0.1242	0.1055	0.9543	0.9595	33.83	5.73	-1228.4	-33.21	-1170.9	-51.6
Mean of layer 2		0.0072	0.0063	0.1646	0.1518	0.9485	0.9509	39.10	6.23	-1218.5	-82.1	-1184.6	-82.9
3	2	0.0389	0.0377	0.8822	0.9089	0.9490	0.9580	85.56	14.86	-811.03	-62.16	-792.40	-67.9
	3	0.0312	0.0306	0.7094	0.7388	0.9378	0.9460	61.82	16.88	-834.57	-35.04	-803.05	-45.1
	4	0.0178	0.0180	0.4040	0.4328	0.9369	0.9386	38.25	12.36	-940.90	-10.65	-892.83	-26.0
	5	0.0099	0.0108	0.2550	0.2425	0.9474	0.9471	29.26	9.18	-1045.5	20.24	-977.78	-1.41
	6	0.0115	0.0110	0.2607	0.2653	0.9513	0.9522	44.27	8.27	-948.8	81.20	-858.23	52.25
Mean of layer 3		0.0219	0.0216	0.5023	0.5177	0.9445	0.9484	51.83	12.31	-916.2	-1.3	-864.9	-17.6
4	2	0.0549	0.0539	1.2444	1.3006	0.9080	0.9657	98.72	18.90	-708.22	-38.34	-685.37	-45.64
	3	0.0611	0.0605	1.3872	1.4604	0.9193	0.9595	95.73	22.75	-633.54	11.48	-592.56	-1.61
	4	0.0988	0.1001	2.2407	2.4124	0.6321	0.8598	98.23	34.03	-448.96	86.04	-385.13	65.65
	5	0.1109	0.1117	2.2490	2.6920	0.4790	0.7610	97.91	36.79	-358.56	159.67	-259.09	130.47
	6	0.0313	0.0305	0.7102	0.7370	0.9450	0.9492	72.79	13.92	-600.27	198.92	-476.56	159.39
Mean of layer 4		0.0714	0.0713	1.5663	1.7205	0.7767	0.8990	92.68	25.28	-549.9	83.6	-479.7	61.7
5	2	0.1107	0.1124	2.5098	3.7096	0.8100	0.9179	97.98	36.94	-511.01	-2.12	-483.43	10.93
	3	0.1191	0.1210	2.6990	2.9170	0.8184	0.9394	97.72	38.85	-433.77	58.31	-383.34	42.19
	4	0.1196	0.1215	2.7124	3.9306	0.8850	0.9590	96.99	39.08	-358.43	-132.47	-278.84	107.03
	5	0.0664	0.0663	1.5060	1.5980	0.8380	0.9392	99.42	23.82	-423.35	202.46	-308.81	165.71
	6	0.1479	0.1507	3.3540	3.6322	0.6410	0.8120	97.026	44.71	-106.43	335.54	50.367	285.45
Mean of layer 5		0.1127	0.1144	2.5562	3.1575	0.7985	0.9135	97.83	36.68	-366.6	92.3	-280.8	122.3
6	2	0.1953	0.1896	1.1011	1.1420	0.3560	0.4130	100.72	14.08	-349.12	27.14	-316.79	16.81
	3	0.1763	0.1665	0.9996	1.0035	0.6787	0.7044	104.89	9.82	-306.06	92.84	-246.18	73.71
	4	0.1176	0.1195	2.6660	2.8810	-0.211	-0.106	97.84	38.52	-341.08	153.90	-252.82	125.71
	5	0.0733	0.0735	1.6610	1.7720	0.7670	0.9224	99.07	26.43	-337.86	265.87	-199.17	221.56
	6	0.0390	0.0386	0.8850	0.9320	0.9393	0.9430	75.67	17.62	-374.18	374.67	-184.28	313.99
Mean of layer 6		0.1203	0.1175	1.4625	1.5461	0.5060	0.5754	95.64	21.29	-341.7	182.9	-239.8	150.4

The MLP model with 1 hidden layer having 3 processing elements performed better than other models. It has the least MSE of 0.0036, a positive correlation value of 0.9723, lowest NMSE of 0.0546 as well as values of -1193.3 and -1180.6 for AIC and MDL respectively.

5. Discussion

The duration of training for 100 cycles (epochs) as depicted in Figure 4 was 6 seconds. The training MSE values of 0.0272 and 0.2246 as well as correlation values of 0.867 and -0.043 were observed for the training and validation dataset respectively. The observed MSE and negative correlation value indicate that the NN was not well trained. A lot of training sessions with varying number of components was conducted for the NN and presented in Table 2. A trained NN helps in proffering solution to pattern recognition problem as well as aid in the determination of input sensitivity to output. The sensitivity of PPP attributes to the determination of flow rate obtained from a trained NN via Neuro Solutions 6.0 is presented in Figure 5. IPS has the highest sensitivity value of 0.653 in the determination of Flow rate of petroleum products in pipeline. IVM and OVM have a score of 0.551 and 0.502 respectively in the sensitivity analysis. The variables ITP, VSC, TPS, OTP, DTY and OPS scored above 0.109 each in their contribution to output. PLT and PDM scored 0.042 and 0.030 respectively while IVS and OVS contributed 0.001 each to the determination of flow rate. The best four contributors to flow rate are IPS, IVM, OVM and ITP.

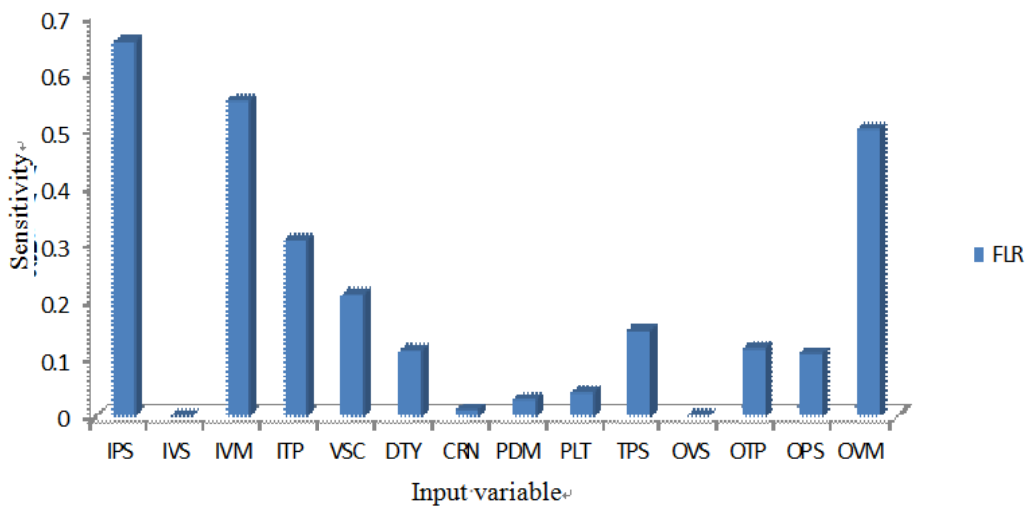


Figure 5. Sensitivity of Pipeline Variables to Flow rate

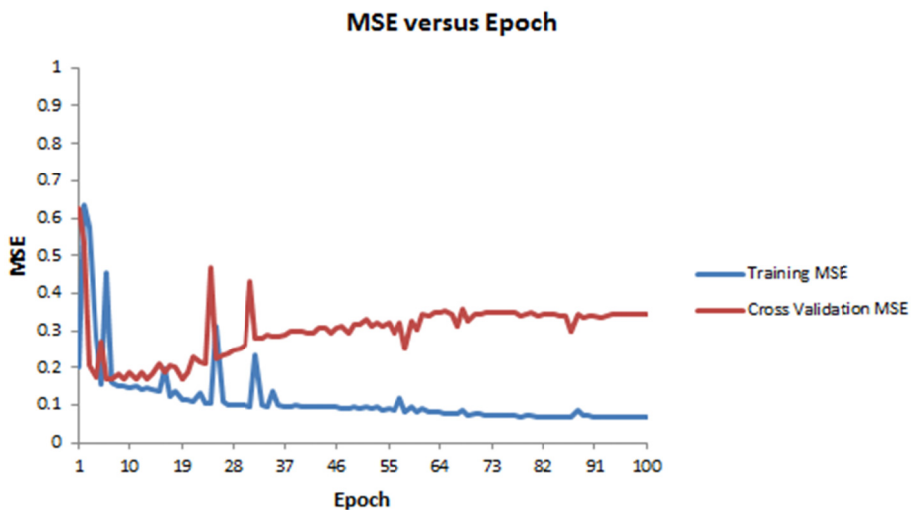


Figure 6. Training and Validation MSE

Demonstration of the training and validation MSE for 100 epochs is shown in Figure 6. At epoch 1, training and validation errors begin at 0.63 and 0.62 respectively. The errors decrease in the same direction till epoch 5. At epoch 6 the validation MSE increases from 0.17 to 0.322 while the training MSE decreases from 0.16 to 0.065, this shows that NN is over trained

at epoch 6. The best NN connection weights for updating model parameters are obtained at epoch 5. These weights are used to evaluate the flow rate of PPP.

6. Conclusion and Recommendations

In this paper, NN models with pattern recognition and learning capabilities for PPP monitoring has been presented. The research findings show that MLP has MSE, MAE and r values of 0.0002, 0.009 and 0.997 respectively between the desired and the computed output. It performed better than RBF, GFF, SVM, TDN and RNN models in the task of monitoring PPP. Further investigation into PPP monitoring using MLP revealed that the model with 1 hidden layer and 3 processing elements performed better than other MLP models. Determination of PPP attributes importance was also carried out using a trained NN. The best four sensitive attributes to flow rate namely: IPS, IVM, OVM and ITP would be used in future work for PPP monitoring to reduce NN size and complexity. Alternate use of sigmoid and hyperbolic transfer functions in different NN layers would be carried out for optimum result. This work recommends further research into self-organising map such as Kohonen neural networks and Adaptive Resonance Theory (ART) with a view of establishing an unsupervised learning for visualizing PPP attributes for PPP activities monitoring.

References

- Adewuji, P. A., & Okelola, M. O. (2013). An Investigation into the Use of Artificial Neural Network to Monitor and Detect Leakage Point(s) along a select Pipeline. *Pacific Journal of Science and Technology*, 14(1), 96-101.
- Akinyokun, O. C., & Inyang U. G. (2013). Experimental Study of Neuro-Fuzzy- Genetic Framework for Oil Spillage Risk Management. *Journal of Artificial Intelligence Research*, 2(4), 14-36. <https://doi.org/10.5430/air.v2n4p13>
- Awosiyan, K. (2011). <http://news2.onlinenigeria.Com/government/68765-pipeline-vandalism-lagos-task-force-uncovers-100-illegal-oil-wells.html>. Visited May 26, 2011
- Cafaro, V. G., Cafaro, D. C., Méndez, C. A., & Cerdá, J. (2010). Oil-Derivatives Pipeline Logistics Using Discrete-Event Simulation. *Proceedings of the Winter Simulation Conference*. Johansson eds IEEE, 2101-2113. <https://doi.org/10.1109/wsc.2010.5678864>
- Chis, T. (2007). Pipeline Leak Detection Techniques. *Annals Computer Science Series*, Andrei Saguna, University, Constanta, Romania, 5(1), 25-34.
- Inyang, U. G., & Akinyokun, O. C. (2014). A Hybrid Knowledge Discovery System for Oil Spillage Risks Pattern Classification. *Journal of Artificial Intelligence Research*, 3(4), 73-86. <https://doi.org/10.5430/air.v3n4p77>
- Jiang, W. W., Yang, Y., Yin, F. W., Ren, J. F., & Ning, Y. (2012). Hierarchical Leak Detection and Localization Method in Natural Gas Pipeline Monitoring Sensor Networks. *Sensors (Basel)*, 12(1), 189–214.
- Niklès, M., Ravet, F., & Briffod, F. (2009). A Comprehensive Long Distance And Real- Time Pipeline Monitoring System Based on Fiber Optic Sensing. *Brazilian Petroleum, Gas and Biofuels Institute, Technical Paper in the Rio Pipeline Conference Proceedings*, 56-60.
- Udoh, S. S. (2009). Artificial Neural Network for the Prediction of the Demand of Petrol at the Depot. An M.Tech Thesis, Department of Computer Science, Federal University of Technology, Akure. Nigeria.
- Udoh, S. S. (2016). Adaptive Neuro-Fuzzy Discrete Event System Specification for Monitoring Petroleum Products Pipeline. *PhD Dissertation, Department of Computer Science, School of Sciences, Federal University of Technology, Akure, Nigeria*, 178-212.
- Udoh, S. S., Obot O. U., George, U. D., & Isong, E. B. (2012). Modified S-S Inventory Model Using Artificial Neural Network. *International Journal of Scientific and Engineering Research (IJSER)* ISSN 2229-5518. Paper ID- I014544, America, 3(5), 1-4.

Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the [Creative Commons Attribution license](#) which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.