BENIN JOURNAL OF STATISTICS
ISSN 2682-5767
Vol. 8, pp. 72– 83 (2025)

Use of K-Means Clustering and Artificial Neutral Network for Predicting Waiting time of Queue system in different Distributions of Arrival and Service times

O. R. Ajewole[1, ∗], A .A. Osagiede [2], S. O. Nwanya [3], C. O. Mmaduakor [4],

and B. A. Ngwu [5]

[1,4,5]*Department of Mathematics, Federal University Oye-Ekiti, Ekiti, Nigeria*
[2]*Department of Mathematics, University of Benin, Benin City, Nigeria*
[3]*Department of Data Science, University of East London, London*

(*Received: 24 June 2024; Accepted: 15 August 2025*)

Abstract. The combination of Machine learning techniques, K-means clustering and Artificial Neural Network(ANN) for analyzing data has not been fully explored especially for queue systems hence the keen interest in this area of research. Machine learning process selected for analysis plays a major role in the effectiveness of the analysis. This study proposes the use of the K-Means clustering and Artificial Neural Network for developing a structure when making predictions of waiting time of a queue system with different distributions of arrival and service times. This is done to understand data in unique ways. In this study, data that consists of the customers' arrival and service duration (for exponential and erlang distributions) are simulated using Google Colab platform and thereafter used for the analysis. Data are grouped by distribution with the use of K-Means Clustering Method. It consists of 6 features altogether, with 3 features used for the prediction of waiting time. For the learning of the data, the optimization techniques; Stochastic Gradient Descent (SGD), Adaptive Moment Estimate (ADAM) and an optimization technique (Ajewole et al 2024a) were used for training the data with the Artificial Neural Network (ANN). The model was trained for 400 epochs with 80data used for training and 20(combination of K-means clustering and ANN) adapted for the analysis. The results obtained showed the performance of the new structure as effective for queue system analysis with different distributions. With this structure, making predictions for new arrivals into the queue system will be less time consuming.

Keywords: K-means clustering; Artificial Neural Network; Waiting time; Performance measure; Prediction.

Published by: Department of Statistics, University of Benin, Nigeria

∗Corresponding author. Email:oghenekevwe.ajewole@fuoye.edu.ng

1. Introduction

According to Donkuan and Yingjie (2015), Clustering is an unsupervised learning method that deals with the partitioning of data structure in an area that is unknown and is further used for learning. The properties of clustering methods include; objects that are similar and are to be grouped in same cluster, dissimilar objects are to be in different clusters, the measurement of similarity of objects in the same cluster or dissimilarity of objects in different clusters must be clear and meaningful. The process of clustering involves extraction of features from original data set, designing of clustering algorithm and then result evaluation and explanation. Clustering can be done based on hierarchy, partition, distribution, grid and model.

A cluster is used for organizing similar objects into clusters based on some criteria. Partitioning the data set into the clusters has to do with finding the minimum squared error between the data sets and the mean of a cluster (centroid) and then assign the data sets of the cluster center nearest to it. Thus, the given data set are partitioned into subsets based on the similarity or closeness among the data. Clustering algorithms belong to either partitioning class or hierarchical clustering class. K-means clustering and Artificial Neural Network have been used for prediction of performance measures in a paper titled "An Exploration Analysis on Half-Hourly Electricity Load Patterns leading to Higher Performances in Neural Network Predictions" authored by Deshani et al. (2014).

In the study, clustering method was used for grouping similar time types. Kmeans cluster was used to identify three intra-day clusters. The first cluster represents the sleeping time of people (11:00pm-5:00am) and the time when most people may be in transit from one place to another (7:30am-8:00am). The second cluster represents the time when most families will be at home (6:30pm-9:00pm). It is regarded as the peak period. While the 3rd cluster represents the rest of the hours in a day. Based on the month or day type for the grouping, the three clusters were further grouped. Then the most appropriate number of clusters was used for the prediction. The training data set collection was done based on the days and months selected for compilation and the inputs used for predicting performances of Neural Network. This analysis was based on daily total Electricity demand by customers and they had a target of making predictions of these demands on a daily basis. Due to the changes in Electricity demand during the day, the predicting half hour demand was introduced. A data set that consists of half-hourly demands for electricity was considered for a period of 1st of January 2008 to 31st December 2012. With this input selection process, the neural network prediction for performance was improved. The author noted that if the duration of examination was expanded, the results obtained could give better outputs.

Gabriel et al. (2018) investigated on the redistribution of tasks in computing systems that work with clusters using various algorithms. The characteristics of the probabilistic-time of transitioning, between queues and connection with shortness of queues is calculated. The node failures and models with different performance were also described. This description was necessary because it was observed that as the number of nodes increased with the queues accordingly, there is more complication in the selection of queue strategy and the description of the model. In this study, results of simulation and analytical modeling for

the system considered were compared. ANN application to Queue Theory Systems for making predictions has been researched on by numerous authors. Some of which include: A Performance Evaluation of Queueing systems by Machine Learning (Suguru Nii et al., 2020), Waiting-Time Estimation in Bank Customer Queues using RPROP Neural Networks (Satya H.R.P.,2018), A Review on the Application of Machine Learning Techniques for predictions of performance measures of queue systems (Ajewole et al., 2024b), Prediction of queueing behavior through the use of artificial neural networks (Josefin and Fredrik, 2017), Predicting patient waiting time in the queue system using Deep Learning Algorithms in the Emergency Room (Hassan and Richard, 2021) and Optimizing Artificial Neural Network with a New Optimizer for Waiting Time Prediction in Queue System (Ajewole et al., 2024a). The papers reviewed paid attention to the use of Artificial Neural Network for predictions. When we have a queue system which has different distributions, an algorithm that combines the K-means and Artificial Neural Network becomes necessary because it improves effectiveness and efficiency.
The data set employed in this study is a randomly generated synthetic data. The exponential distribution was used to generate data that follow exponential distribution of 3,128,363 rows of customers waiting time, service duration and inter arrival time. The gamma distribution was used to generate data that follow Erlang distribution with 4,910,479 rows of customers waiting time, service duration and inter arrival time.

2. Materials and Method

2.1 *K-Means clustering*

Describing the K-means cluster algorithm, we use the explanations given in Jiming and Yu (2005):
Assuming there exists a set "N" of n points in a d-dimensional Euclidean space which is denoted by

$$N = \{(x_i, ...x_{ik})^T \in \Re^d i = 1, ..., n\} \tag{1}$$

The goal is to assign n points into k clusters which are disjointed. The clusters are centered at centers $(c_j, j = 1, ...k)$.

Steps for K-Means clustering:

(1) Specify number of clusters
(2) Create similarity metric
(3) Run clustering algorithm;

2.2 *Artificial Neural Network*

According to Gabriel et al. (2018), when discussing a multilayer perceptron, the learning and prediction process has to be considered. For efficient and reliable results to be obtained, the number of layers and activation functions used have to

be the same. A neuron (node) for the system considered in this study is defined below:
For inputs $x \in \Re^d = \{T_\lambda, T_\mu\}$, bias $b \in \Re$ and weights $w \in \Re^d$ given, a neuron is the function

$$\hat{T}_w = f(wx + b) \tag{2}$$

Where $f : \Re \rightarrow \Re$ is the activation function, T_λ denotes the arrival rate, T_μ denotes the service rate, $\hat{T}_w$ the predicted waiting time and x represents the inputs.
The structure of a simple neural network consists of three layers. They are the input layer, the hidden layer and the output layer as shown in figure 2. They are given below as defined in Eloy (2017).

(1) Input Layer (x_i): This is the layer that stores the data inputs of the network. Information is received from external sources into the input layer and this is passed on to the network to process. The inputs are values which can be viewed as features that we need to predict the output value.
(2) Weight(w): This enables the Artificial Neural Network to adjust the strength of connections between neurons
(3) Bias (b): It is a constant value added to the product of input and weights. It is utilized to offset the result i.e. shift the result of activation function towards positive or negative.
(4) Hidden layer: The hidden layer is added to be able to find complex structures in the data. The hidden layer processes information received from the input layer. A multi-layer neural network is a neural network that has multiple hidden layers. Here, the inputs are modified by the weights.
(5) Output layer (a): This layer of neurons sends output and processed information out of the system. This is the layer in which the prediction of the network is done.

2.3 *ReLu activation function*

The ReLu activation function is given by

$$f(z) = max(0, z) \tag{3}$$

Thus,

$$\hat{T}_w = f(z) = max(0, z) \tag{4}$$

where

$$z = wx + b \tag{5}$$

for w and x defined in 2.2

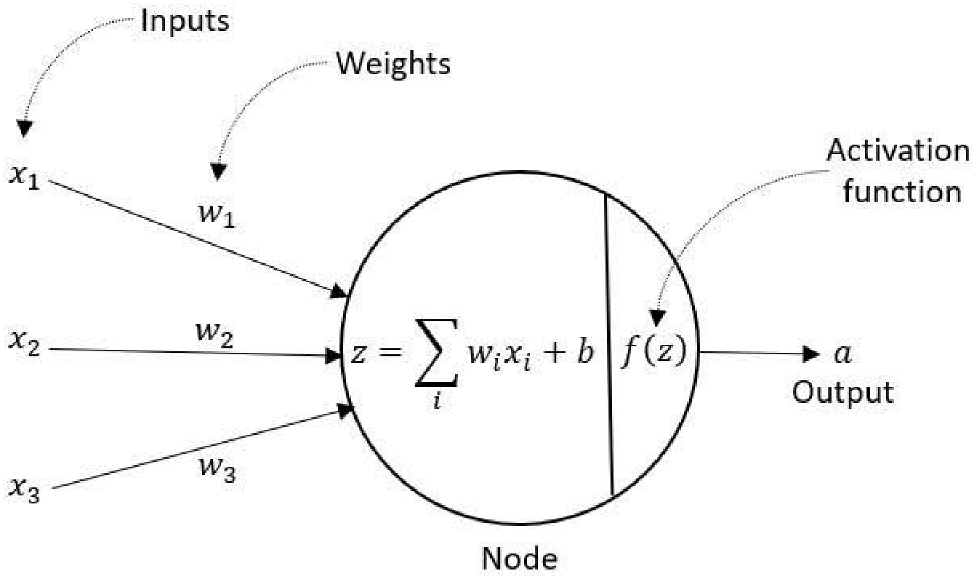

Figure 1: A simple Neural Network, (Devraj,2020)

2.4 *Cost function*

The cost function is given by the equation below:

$$C = [\frac{1}{n}\sum_{i=1}^{N}(\hat{T}_w - T_w)^2] + \psi J_w \tag{6}$$

The first part of (6) is the sum of the squared difference between actual (desired) waiting duration Tw and predicted waiting duration (target) $\hat{T}_w$. The second part represents the l_2 regularized loss with the regularization parameter ψ and $J_w = \sum_{j=1}^{p} w_j^2$ for $w_j \in \Re^d, j = 1, 2, ..., p$

2.5 *Cost of function Optimization*

To find the optimal value or weights for cost reduction, the function $C(w_1, w_2, ..., wp$ has to be minimized for C in (6). This is done by solving the multivariate loss function $C(w_1, w_2, ..., wp$. First, the partial derivatives $\frac{\partial C}{\partial w_j}$, for $forj = 1, 2, ...p$ are computed and then the gradient is given by

$$\nabla C \tag{7}$$

is solved which represents the gradient of the cost function. In Machine Learning, numerical methods of minimization have been developed to approximate the global minimum of the function C. Algorithms are developed to represent the Machine Learning methods.
In this study, Ajewole et al. (2024a), stochastic gradient descent and Adam methods are used. The methods are explained below:

(1) Update rule for Stochastic gradient descent (Dokkyun et al., 2020); The gradient is the information of the direction in which a function has the rate of change that is steepest. Most objective functions are stochastic. The Gradient Descent update rule for the weights is given by:

$$w_{t+1} = w_t - \alpha \nabla c(w) \tag{8}$$

with w_{t+1} representing the updated weights at time $t + 1$, w_t denoting the weights for the previous iteration at time t, α, the learning rate and $\nabla c(w)$ the gradient of the objective function of the problem. The process of taking the gradient of the loss by random sampling is referred to as the Stochastic Gradient Descent (SGD).

(2) Update rule for Adam(Diederick and Jimmy, 2015):
The formula for updating the weights is given by

$$w_{t+1} \longleftarrow w_t \leftarrow - \frac{\alpha_t \hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \tag{9}$$

where α_t is the learning rate at step t and ϵ is included for numerical stability. First and second moments are given by

$$m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1 \cdot \nabla c(w), \tag{10}$$

$$v_t \leftarrow \beta_1 \cdot v_{t-1} + (1 - \beta_1 \cdot (\nabla c(w))^2, \tag{11}$$

with m_t representing the mean of the gradient at step t and v_t, the variance of the gradient at step t with β_1 and β_2, the decay rates. m_{t-1} and v_{t-1} represent the mean and variance of the previous steps respectively. The bias correction for the mean of $\triangledown c(w)$ at step t are introduced due to the fact that the first and second moments are relatively small and this leads to initiation of large step size. They are:

$$\hat{m}_t = \frac{m_t}{(1 - \beta_1^t)} \tag{12}$$

and

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{13}$$

β_1^t represents β_1 with power t and β_2^t with power t.

(3) Update rule for Ajewole et al (2024a) optimization algorithm

$$w_j^{l+1} = w_j^l - \frac{\alpha \hat{m}_t \sqrt{r_t}}{\sqrt{\hat{v}_t} + \epsilon} \tag{14}$$

where w_j^{l+1} is the jth weight at layer $l + 1$, w_j^l is the jth weight at layer

l, $\hat{m}_t$ is the bias correction of the gradient mean of the objective function, $\hat{v}_t$ is the bias correction variance of the objective function, at time t. ϵ is included in the numerator for numerical stability. We have

$$r_t = \beta_2[(\gamma_{t,1} + \gamma_{t,2})/2] \tag{15}$$

where

$$\gamma_{t,1} = |C_t(w_t) - C_t(w_{t-1})| \tag{16}$$

and

$$\gamma_{t,2} = |C_t(w_{t-1}) - C_t(w_{t-2})| \tag{17}$$

$C_t(w_t)$,$C_t(w_{t-1}$ and $C_t(w_{t-2})$ are used to represent values of objective function at times t,$t-1$ and $t-2$ respectively. β_2 controls the decay rate as defined in Adam. The decay rate is to stabilize r_t.

2.6 *Performance measures*

The equations for the performance measures for the queue system computations used in this study are given below:

$$\text{Average Service duration}(\bar{\mu}) = \frac{\text{Total Service duration(TTS)}}{\text{Number of Arrivals}(T_n)} \tag{18}$$

$$\text{Service rate}(T_\mu) = \frac{1}{\text{Average Service duration}(\bar{\mu})} \tag{19}$$

$$\text{Average inter arrival time}(\bar{\lambda}) = \frac{\text{Number of Arrivals}(T_n)}{\text{Total inter arrival time}(TIT)} \tag{20}$$

$$\text{Arrival rate}(T_\lambda) = \frac{1}{\text{Average inter arrival time}(\bar{\lambda})} \tag{21}$$

2.7 *Description of Queue System*

We have a Queue system of N training large samples with input features $\{T_\lambda T_\mu\}$. The data sets have the following features: number of arrivals (T_n), duration of service (T_s) and waiting time duration (T_w) for a given period of time. The arrival and service times follow a general distribution and hence can be represented using the Kendall's notation as $G/G/s/K$. The capacity of the system is finite, denoted K with s servers.
The K-means clustering and Artificial Neural Network are the Machine learning techniques used for the processing of the queue data for predictions. Before

the samples are trained, they are grouped into clusters based on their various distributions using the K-means clustering method. Assignment to each centroid is based on the cluster that represents each of the distributions the data sets are grouped into. This is given by

$$N_k = \sum_{i=1}^{n} x_i \in c_k. \tag{22}$$

The data (arrival and service rate) are exponentially distributed and k-Erlang distributed. Thus, the dataset N is partitioned into $k = 2$ disjoint clusters for this study.
Given the data set N_k, x_i belongs to the kth cluster, where $x_i, ...x_n$ is the d-dimension data point of size N partitioned into k clusters named c_1 & c_2 which represent the number of centroids. c_k is predefined centroid. For a point x_i to belong to the cluster k, it must satisfy

$$d = \sum_{k=1}^{k} \sum_{i=1}^{N} ||(x_i - c_k|| \tag{23}$$

Where N is the number of points denoting the data sets, x_i is the i^{th} point in x_i. It is when the function d achieves its minimum for known cluster centers then we have each point assigned to the cluster center closest to them. (23) is called cluster assignment in Machine Learning. i.e. the lowest value in (23) is assigned to the centroid. After the clustering is done, the forward process Multi-layer Neural Network is initiated by introducing the inputs and passing them through two fully connected hidden layers with each layer performing a specific transformation. The Neural Network has two hidden layers and one output $T_{\hat{w}}$, the predicted waiting time. The desired output is the label (target) T_w. The ReLu activation is used for this experiment. For the training, the arrival rate T_λ, and the service rate T_μ are the input features.

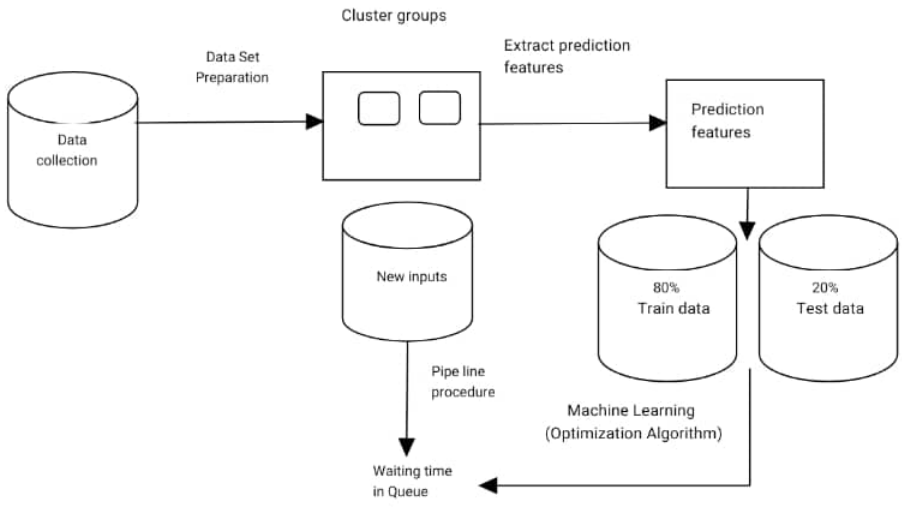

Figure 2: FLOW CHAT

Figure 2 represents the structure of the prediction process of the Queue data collected for clustering and prediction.

2.8 *Waiting time and Service time of Queue System*

Waiting time in a queue is the time a task has to wait in the queue before service begins. In other words, it is the difference between the service start time and the arrival time [Medhi (1975)]. According to queue theory experts, customers often over estimate their wait times by percentage up to 36. The service time is the length of time a task spends in the process. This time begins when service starts for a task and ends at the completion of service (Muhammad, et al. (2019)).

3. Results and Discussions

Table (1) is a sample Data Set which includes information of a queue system with customers waiting time T_w, service duration T_s and inter arrival times λ. The data set is a randomly generated data with the use of python code on the Google Colab platform. We have $3,128,363$ rows of customers service duration and inter arrival time that follow exponential distribution and $4,910,479$ rows of customers' service duration and inter arrival time that follow Erlang distribution. The exponential distribution was used to generate data that follow exponential distribution and the gamma distribution was used to generate data that follow Erlang distribution. The Customer waiting time T_w, Service duration T_s and inter arrival times λ are further used to compute the performance measures.

Table 1: Sample Data set 1

T_w (sec)	T_s (sec)	λ (sec)
687.19	112.52	20.47
47.3	72.35	80.71
528.29	41.81	26.53
17.37	20.12	29.11
94.44	25.75	0.25
915.25	17.24	28.86
18.96	23.7	17.46
17.27	20.24	15.91
164.42	64.69	27.26
233.72	49.42	50.32

Table 2 is the computed performance measures $(18) - (21)$ for generated data represented in groups of arrivals. Where $\hat{\mu}$ - Average Service, T_w - Average waiting time, λ - Average inter arrival time, T_μ - Service rate and T_λ - Arrival rate. The Service rate T_μ and the Arrival rate T_μ on computed data set are the input label used for training the Machine Learning model. After distribution-type clustering, the data is entered in the Machine Learning model for learning. The model is trained for 4000 epochs and there was improvement in the results for prediction based on the different distributions. 80% of the

data was used for training and 20% for testing. For the first epoch, $\alpha = 0.01$ was used as the learning rate and $\alpha_t = \alpha/2 \forall t$. The data is used to show the performance of the learning and prediction process using the structure of our study that involves the Machine Learning methods, K-means clustering and Artificial Neural Network. Model by Ajewole et al. (2024), Stochastic Gradient Descent (SGD) and Adam optimization methods are adopted for training with the cost function given in (6)
The cost plotted against the number of epochs is represented in figure 2.

Table 2: Sample Data set 2

Arrival	$\bar{\mu}$	T_w	λ	T_μ	T_λ	Distribution
164	19.65	24.1	0.05	0.05	19.7	Erlang
166	96.5	636.76	0.01	0.01	104.54	Erlang
179	22.07	25.11	0.05	0.05	21.84	Exponential
162	101.51	645.94	0.01	0.01	95.99	Erlang
194	50.01	278.0	0.02	0.02	50.08	Exponential
145	20.88	19.21	0.05	0.05	20.68	Erlang
139	20.93	26.55	0.05	0.05	20.65	Erlang
186	105.27	733.98	0.01	0.01	100.69	Erlang
145	49.97	243.0	0.02	0.02	50.06	Exponential
148	19.75	12.14	0.05	0.05	19.99	Erlang

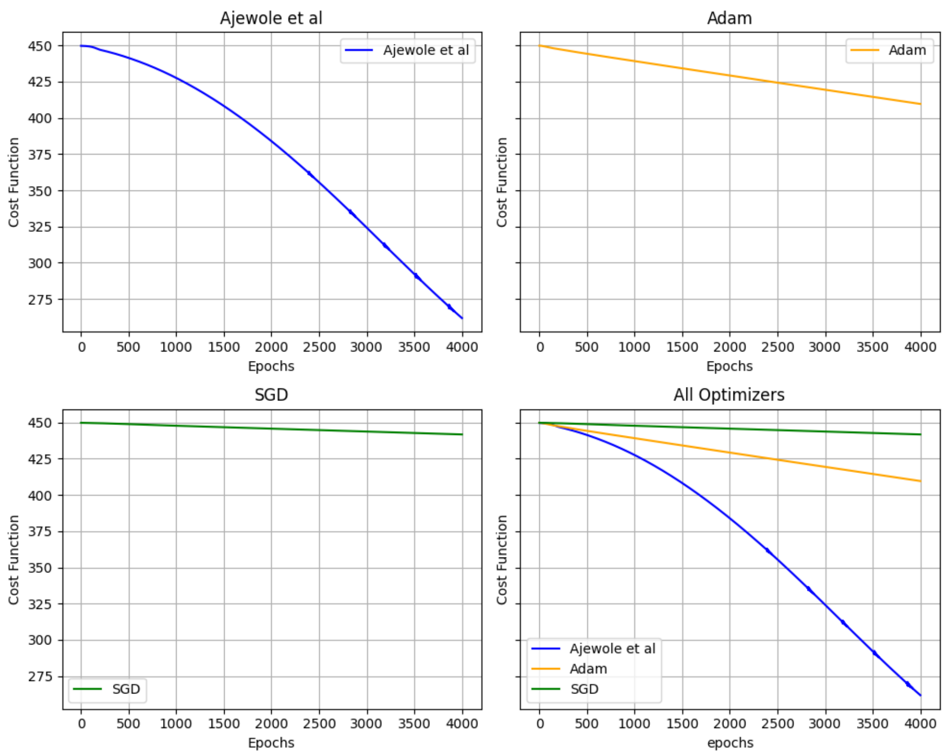

Figure 3: Graph of Cost Function versus Epoch for three optimizers for Computed data

Figure 3 is the graph representing the cost function plotted against the number of epochs for the trained big data set for the three optimizers considered. The number of epochs is the x-axis and the optimizers are plotted on the y-axis. The cost function of the problem for the Ajewole et al.(2024), Adam and SGD converge consistently for 4000 epochs but not at the same rate as observed in the graph. Ajewole et al. (2024a) converged to 275 seconds, Adam converged to 410 seconds and SGD converged to 440 seconds at the $4000th$ epoch. With the convergence of the optimizers, the results obtained show that clustering data sets using distribution type gives accurate results.

4. Conclusion

In this study, a structure for making predictions of performance measures for queue systems is developed with Machine Learning techniques. The Machine Learning techniques used include the K-means clustering method and the Artificial Neural Network. The performance measure predicted in the implementation for a queue system is the waiting time using the Erlang and exponential distributions. This Artificial Neural Network was trained with three optimizers which include Ajewole et al. (2004a), Adaptive Moment Estimate (ADAM) and Stochastic Gradient Descent. With the results obtained, it is obvious that the structured process is efficient for the analysis of data for prediction purpose especially for scenarios where data involves different distributions.

References

Ajewole, O.R., Osagiede, A.A, Nwanya S.O. , Efebeh, A.F. and Ubaka, O.N. (2024). Optimizing Artificial Neural Network with a New Optimizer for Waiting Time Prediction in Queue System, Faculty Journal of Pure and Applied Sciences (FJPAS),9(2)

Ajewole, O.R., Osagiede A.A and J.O. Okoro (2024). A Review on the Application of Machine Learning for predictions of performance measures of queue systems, International Conference on Science, Engineering & Business for driving Sustainable Development goals (SEB4SDG)

Deshani, M.D.T., Attygalle, L.L.H., Hansen, L.L. and Karunaratne, A.(2014). An Exploratory Analysis on Half-Hourly Electricity Load Patterns Leading to Higher Performances in Neural Network Predictions. International journal of Artificial Intelligence and Applications (IJAIA), 5(3), May 2014

Diederick, P.K. and Jimmy, L.B. (2015). Adam, a Method for Stochastic Optimization, published as a Conference paper at ICLR

Donkuan, X. and Yingjie, T. (2015). A Comprehensive Survey of Clustering Algorithms, Ann. Data Sci. (2015) (2): 165-193

Dokkyun, Y., Jaehyun, A. and Sanguine, J. (2020). An Effective Optimization Method for Machine Learning Based on Adam, Applied Sciences, 10,1073

Gabriel, V., Juan, F. Pablo, C. and Fernando D.L.P. (2018). Artificial neural networks used in optimization problems, Neurocomputing 272(2018)10-16

Hassan, H. and Richard, O. (2021). Predicting Patient Waiting Time in the Queue system using Deep Learning Algorithms in the Emergency Room, International Journal of Industrial Engineering, 3(1), October 2021, 33-45.

Jiming, P. and Yu, X (2005). A new theoretical framework for K-means-type clustering, Studies in Fuzziness and Soft Computing, January 2005, 79-96

Josefin, S. and Fredrik, N. (2017). Prediction of Queueing behavior through the use of Artificial Neural Networks, K^{th} Royal Institute of Technology school of computer science and communication.

Medhi, J. (1975). Waiting time distribution in a Poisson queue with a general bulk service rule, Management Science, 21(7), 777-782

Mohammed, D., Armann, L., Bora, K. and Kenneth, S. (2019). Load effect on Service times, European Journal of Operational Research.

Satya, H.R.P., Suharjito, S. Diana and D. Ariadi Nugroho (2018). Waiting-Time Estimation in Bank Customer Queues using RPROP Neural Networks, Procedia Computer Science 135:35-42