National College of Ireland

Dynamic Load Balancing and Resource Management Using Machine Learning

MSc Research Project
MSc in Cloud Computing

Amit Goswami
x21152594

School of Computing
National College of Ireland

Supervisor: Vikas Sahni

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Amit Goswami

Student ID: x21152594

Programme: Cloud Computing **Year:** 2022-23

Module: MSc Research Project

Lecturer: Vikas Sahni

Submission Due Date: 01/Feb/2023

Project Title: Dynamic Load Balancing and Resource Management Using Machine Learning

Page Count: 18

Word Count: 4750

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: .....

Date: 31/Jan/2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□

You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Dynamic Load Balancing And Resource Management Using Machine Learning

Amit Goswami
21152594

Abstract

The key objective of this research paper is to explain the processes used in the implementation of Load balancing and Resource management methods. Load balancing is considered to be one of the key aspects in networking as it maintains the incoming network traffic and subsequently in a long duration, it saves data, time and money. In this project, AWS Cloud Service is utilised in order to perform load balancing tasks. The data presented here shows various features like server id, time stamp, and also load on servers. Moreover, Machine Learning algorithms have been employed in the procedure and three datasets are utilised in the entire process. The dataset is gathered from an open source website. In order to achieve better results, proper analysis of data is conducted by eliminating null values and noise. With the help of ML techniques, the servers with high load are selected and the load is then transferred to servers with low load so as to achieve load balancing.

The entire research is conducted with the implementation of Python Programming language and programs run in the Jupyter Notebook environment. Artificial Neural Network (ANN) and Linear Regression are the two chief Machine Learning algorithms employed in this research as they provide highly promising results. Furthermore, the Linear Regression approach is considered more effective than ANN algorithm as its R2 value is closest to 1 hence produces much more accurate results. Therefore, it can be concluded that highly accurate results are achieved and can be utilised practically.

1. Introduction

Cloud load balancing is a process in which reduces network traffic by effectively distributing the workload on servers. In order to effectively employ the machine learning approach, a dynamic resource balancing management system has been used. Moreover, the data is collected from online sources and transferred in the program to execute the algorithm.

Importantly, linear regression approach has been conducted in this program. Also ANN algorithm is performed that provides similar results. In this research, Python language has been used to conduct load balancing programs. By means of big data analysis, the load balancing algorithm assists the division of data elements.

1.1. Background

The background of the research delivers the research analysis in accordance with themachine learning model. The machine learning approach is capable of loading the data and also examining the nature of the data elements. This ability of the model facilitates the loading of data elements and simultaneously analyses them (Greener et al. 2022). Moreover,machine learning model can effectively examine the data sources and can import data in a certain order. Now, the task is required to describe the data in 2 parts. One part is conducted on the sender initiated and the other one on the receiver initiated.

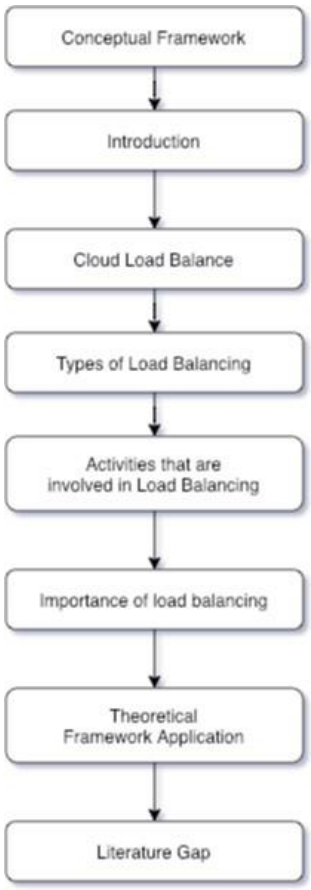


Figure 1: Document Structure.

1.2. Research Question (RQ)

Q1. How the data did gather from the website and arranged the data in the cloud system?
Q2. What is the exact process to manage all the data together and build the balance model?
Q3. What is the technique employed in the process of identification in order to define the "Workload Distribution in Cloud Computing"?
Q4. What are the ways in which the order task scheduling and distribution organize in an appropriate manner within the cloud?

2. Related Work

Significance of load balancing in cloud

The key aspect of load balancing is the strengthening of response time. load balancing can be beneficial in boosting various elements like migration time, fault tolerance, resource scalability, execution cost etc (Mishra et al. 2020).

Furthermore, factors such as energy and power consumption, resource wastage, and migration cost can be effectively diminished. According to the views of Neelima et al. (2020), the service quality is determined by the factor quality, like, if the quality of the factors mentioned above is declined then the service quality will automatically decline. So, there arises a challenge to offer the QoS established on the requirements of CSC and the economy as well. All these obstacles can be easily resolved by the load balancing feature in the cloud in which the network could be routed on the basis of the server's existing resources.

2.1. Cloud Load Balancing

The procedure of redistribution of workload among the servers in the system is termed as load balancing. The load distributing system that has been specified in this paper is a cloud computing system Afzal et al.(2019). Importantly, various aspects such as execution time, response time, system stability and more can be efficiently enhanced with the help of this cloud computing service. According to Ullah (2019), Load balancing system can be managed and governed by various techniques such as artificial bee colony. Load balancing system can be seen and used for a lot of specific purposes such as resource allocation, resource scheduling, resource management, task allocation, and task scheduling.

Specifically, load balancing is implemented in the situations where a single server is unable to manage the complete workload of an application. It also helps in minimizing workload handled by a single server and consequently reduces the chances of drop requests, crashes, and slow down of systems. Load balancing process is responsible for evenly distributing network traffic and eventually helps in mitigating overloading difficulties Mishra et al.(2020). This particular event takes place between the servers and the clients. Moreover, it acts like a hidden facilitator for clients and always ensures that “connection requests” are not missed or lost. In addition to that, if demand or requests on servers escalate in the absence of cloud load balancing, the database might collapse or fail to work in an appropriate manner. The cloud load balancing approach plays a crucial role in effectively reducing downtime of servers by distribution process.

2.2. Load balancing types

Basically, the two types of load balancing techniques are: software load balancers and hardware load balancers.

- **Software load balancers-** These load balancers are flexible and economical in nature and can be deployed easily. According to Ebadifard et al.(2020) usage of the software load balancers are in association with the environment of software development. The certain needs of users can be fulfilled with the assistance of software approach as it effectively inspects the system environment. This type of load balancer is a form of virtual machine which enables the users to make alterations and can upgrade as well. Moreover, these are accessible in installable solutions form wherein users have to configure and govern whichis regarded a load balancer as a service (LBaaS).

- **Hardware load balancer-** These load balancers are comparatively harder to deploy in systems. In the case of this particular load balancer, network traffic is distributed on the basis of the existing connections on utilization of process, performance of servers, and the server.

2.3. Some activities that take place in the process of load balancing

The cloud workload is generally assigned and scheduled tasks to virtual machines on the basis of the necessities of a specific task. Listed below are the activities linked with the load balancing:

Identifying resources of a VM

Current state of the resource integrated with the VM is inspected continuously in this specific activity. It provides information regarding the utilization of the latest resources and also informs about the unallocated resources. Moreover, it shows the balanced state of virtual machines on the basis of overloaded or underloaded status of available resources which is analyzed in accordance with the threshold.

Resource allocation

Assigning resources to the scheduled tasks in order to execute them. Scheduling procedures accelerate the execution process. Allocation policies are essential as they regulate and govern the strength of load balancing and also manage the resources.

The user task requirement identification

In this particular stage, it is responsible for providing the conditions to execute the scheduled tasks.

Migration

It is an important phase of load balancing. The 2 kinds of migration such as VM migration and task migration. VM migration is regarded as the movement of VM between 2 hosts (Mohanty et al.2019). Also, it can minimize the obstacles such as overloading. VM migration can be classified into live migration and non-live migration.

3. Research Methodology

This particular segment deals with the methods used in the establishment of linear regression. The foremost step in the implementation of linear regression is for the task to load the dataset. Now, it is important to find the target column within the model as it helps indeterminant to construct classification or regression model.

This research has selected the regression model for implementation. Linear regression has three phases that are examined on the basis of direction and correlation of data. The model is evaluated and it declares the evaluating validity, fitting of the line, and the significance of the model. It shows association between independent and dependent variables and also ascertain interactions with predictor variables.

3.1 Block Diagram

This section of this paper discusses the applications in order to construct the research system. A programming language, Python, has been employed in order to develop few essential phases and also to determine the importance of the outcomes. The chief requirements in the regression process are all the imported and essential library functions (Abiodun et al. 2019). Imported libraries assist in leading the implementation process. During dataset observations, identification of the data takes place.

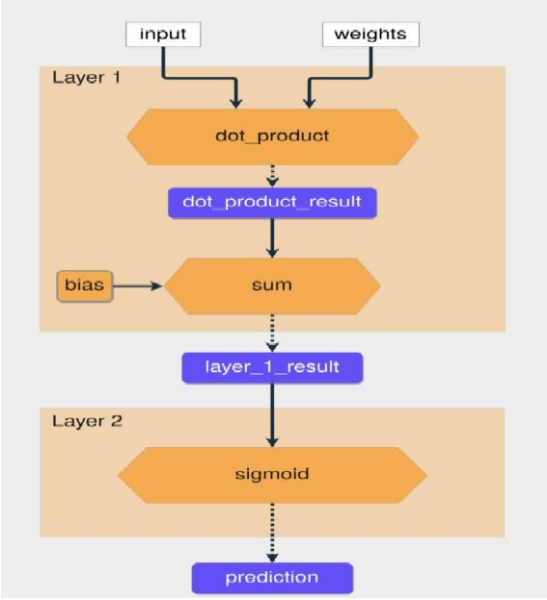


Figure 2: Block Diagram

3.2. Workflow

Research philosophy (Source: Learner developed) Cloud extraction method has been employed in order to complete the task as it is capable of accessing, extracting, analysing, and retaining data saved in the cloud. This process is employed in various technology companies in order to assess the data stored remotely. Moreover, data can be recovered from the devices present in the third party companies. The cloud storagehas been boosted and made available for cloud extraction, social media, and all internet- accessible devices as well. Whole process can be easily divided into three parts such as extraction,

transformation, and loading. The process of extraction states that the data can be effectively extracted from one or more systems. The process of transformation states that the extracted data is ready for filter. Following the loading step, transformation is conducted with high-quality data (Tavares et al. 2019). The data is transferred to a unified target for analysis and storage purposes. Now, the data is run with the help of python programming language for ANN and linear regression as well. It is important to carry out linear regression as the data present in the dataset are associated with every single entity. During analysis, few events that are related to observation and interest are examined.

3.3. Research Approach

The research approach of linear regression makes use of certain relationships among the "data points" for drawing a linear line throughout all of them (Kadam et al. 2019). The line is used in order to predict possible future values. Machine learning is entirely about predicting future possibilities which is based on the algorithm over supervised learning. The next step is to evaluate the error prediction and then the partial derivative is done. Simultaneously, ANN is applied which is the typical random function of approximation. These cost-effective technologies are utilized by the users in order to arrive at the solution which specifies certain distribution. Accordingly, ANN offers the output result which is based on the sample data rather than the complete dataset. The research is based on the paradigm which is inspired by the human brain. Unlike humans, ANN is learned from past experiences or examples. Configuring ANN within a certain application like that in pattern recognition and data classification.

3.4 Proposed Algorithms

This section of this research provides the appetite rages that describe the approaches in the model to build the ML model. The regression ML model has been invented in this research to describe the load simulator. The research provides the model specifications so as to examine the system of the data application. Therefore, the data has been loaded into the system to describe the library's function. The appropriate libraries can easily import the model. The research has gathered data from online sources. Gathering the data in the model the process needs to carry all the liberty and implement the task. Therefore, he talks about the need to merge the datasets that are used in the load management process. The research provides a significant way to connect with the cloud server and build the "dynamic load balancing and resource management" model in python.

```
from sklearn import metrics
from sklearn.model_selection import cross_val_score
def cross_val(model):
    pred = cross_val_score(model, X, y, cv=10)
    return pred.mean()
def print_evaluate(true, predicted):
    mae = metrics.mean_absolute_error(true, predicted)
    mse = metrics.mean_squared_error(true, predicted)
    rmse = np.sqrt(metrics.mean_squared_error(true, predicted))
    r2_square = metrics.r2_score(true, predicted)
    print('MAE:', mae)
    print('MSE:', mse)
    print('RMSE:', rmse)
    print('R2_Square', r2_square)
def evaluate(true, predicted):
    mae = metrics.mean_absolute_error(true, predicted)
    mse = metrics.mean_squared_error(true, predicted)
    rmse = np.sqrt(metrics.mean_squared_error(true, predicted))
    r2_square = metrics.r2_score(true, predicted)
    return mae, mse, rmse, r2_square
```

Figure 3: Development of a Machine Learning Model

3.4. Data collection method

The method of data collection is very essential as it involves the process of measuring as well as gathering information from countless various sources. The data are utilized by collecting and developing the AI along with the solution machine learning. In order to build this model, data collection is an important and necessary option in the research. The research clearly describes the sources from the dataset. The datasets are gathered from online sources. It has been shown that the dataset that has been collected online holds the data time stamp, server id, and load specification.

Dataset description (Source: Developed by the learner) The data specified the values in the machine learning model so as to analyse the system protocol to define all the outcomes. Moreover, the task implements the ANN and regression model on the base of the data elements. Therefore, the data download the dataset from the online sources the taskneeds to store the data in the machine. The task has loaded three datasets in the process. Now,it is necessary to put the data together and merge all the data. To complete the mergedata the task needs to arrange a new dataset from the implementation process.

4. Design Specification

While implementing the process of machine learning, it essentially needs a huge amount of “processing power” and data storage as well. There are several servers which also work in the background simultaneously for working along with the algorithm. There is an essential role within cloud computing which assists the user in providing various servers along with predefined data as well as changing the resources all over the internet or cloud. The cloud generally makes the capabilities accessible in terms of advanced skills within artificial intelligence and data science. There are several learning options which help in figuring out the learning theory. The learning algorithm of the model has established the load balancing model. The load balancing model has generated the proper task distribution model by means of the regression and ANN model. The process has followed these stages to describe the system in the proper way.

Research design (Source: Developed by the learner) The research has designed the essential stages to make the process very clear. Moreover, the task needs to follow this section to make the research model in an exact way. The job specification has started with the need to collect information. The information collection process uses the decision maker stage that helps to

reload the data in the model. It generates the values in the model which are required to be connected with the code analysis. Now, the code analysis depends on the target-specific generation and execution of the code in a proper manner. To finally complete all the arguments in the code management process, the operation needs to work with the run compiler. The code compiler helps to derive the data in the load balancer. According to the process, the task has made the decision to get the regression model with the help of the continuous target values. Set those values as a target variable and make this regression model. Now, for data delivery, the task has invented the linkage with the module manager. The module manager provides all the steps, makes all the rectification in the data elements and executes the outputs from the model.

5. Implementation

This research project utilises the processes to construct load balancing by employing Python language within the Jupyter notebook.In order to carry out the whole procedure, it is essential to select server time datasets with server loads and the timestamps as well.This facilitates the effective implementation of some machine learning procedures like Artificial Neural Network and Linear Regression. Appropriate library system has been employed so as to perform the process and produce results. Moreover, the process efficiently sets the algorithm with the assistance of the library system.Three different data frames have been shown here and named as df, df 1, df 2 and this requires three different datasets to be imported in this project itself.It can be concluded from the steps that different data frames i.e. df, df 1, df 2 have been unified. For implementing ANN and linear regression, It is necessary for the task to merge the existing datasets and also generate a new dataset within the model. Now, conversion of data type has been done so as to avoid any possible errors that might happen due to the data of the variables. The process of obtaining null values in the dataset is analysed significantly. This step of the implementation discusses the Exploratory Data Analysis in order to efficiently examine the details about the dataset. The details such as shape, size, datatype, mean, column name, count, standard deviation, minimum and maximum values for the dataset can be easily obtained. The process of machine learning is initiated with the help of selection of the target column and storing in variable ‘y’. Now the entire data frame is split into 4 subsets named X-train, X-test, y-train, and y-test with the 20% test size. Now in order to develop the entire model, the values of Mean Square Error, Mean Absolute Error, R Square error, and Root Mean Square Error are calculated for each regression model.The process of executing pipeline procedure with the help of standard scaler attributes of sklearn library and for this procedure X_test and X_train datasets are selected. The value of the target variable is predicted and a detailed comparison is performed so as to obtain the errors.Initially, this procedure was converted into a NumPy array and 4 layers are developed with an activation function.

```
from sklearn.linear_model import LinearRegression
```

```
lin_reg = LinearRegression(normalize=True)
lin_reg.fit(X_train,y_train)
```

```
LinearRegression(normalize=True)
```

```
print(lin_reg.intercept_)
```

```
0.9092708333333334
```

```
pred = lin_reg.predict(X_test)
```

```
pred
```

```
array([0.44798769, 0.22995793, 1.2259464 , 1.40140884, 1.35689409,
       1.06402877, 1.53859075, 1.12833293, 0.2731744 , 0.28326406,
       0.67849629, 0.48445147, 1.44592359, 0.96750617, 1.44527445,
       1.39006032, 1.35424569, 0.31833829, 0.5708844 , 1.58740268,
       1.58870096, 0.36220391, 1.53988904, 0.50368299, 1.16055136,
       0.4756601 , 0.27382354, 0.33462275, 1.68777033, 0.57153355,
       1.26916287, 1.38876203, 0.44863684, 1.25837224, 0.36155476,
       0.95222134, 1.54664173, 0.76752579, 1.04125085, 1.0659762 ,
       1.23099849, 1.44462531, 1.08052063, 0.36025647, 0.667407  ,
       0.90900488, 0.87163272, 0.59141421])
```

Figure 4: Linear Regression Procedure

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Input, Dense, Activation, Dropout
from tensorflow.keras.optimizers import Adam
X_train = np.array(X_train)
X_test = np.array(X_test)
y_train = np.array(y_train)
y_test = np.array(y_test)
model = Sequential()
model.add(Dense(X_train.shape[1], activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(64, activation='relu'))
model.add(Dense(128, activation='relu'))
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.1))
model.add(Dense(1))
model.compile(optimizer=Adam(0.00001), loss='mse')
r = model.fit(X_train, y_train,
              validation_data=(X_test,y_test),
              batch_size=1,
              epochs=100)
```

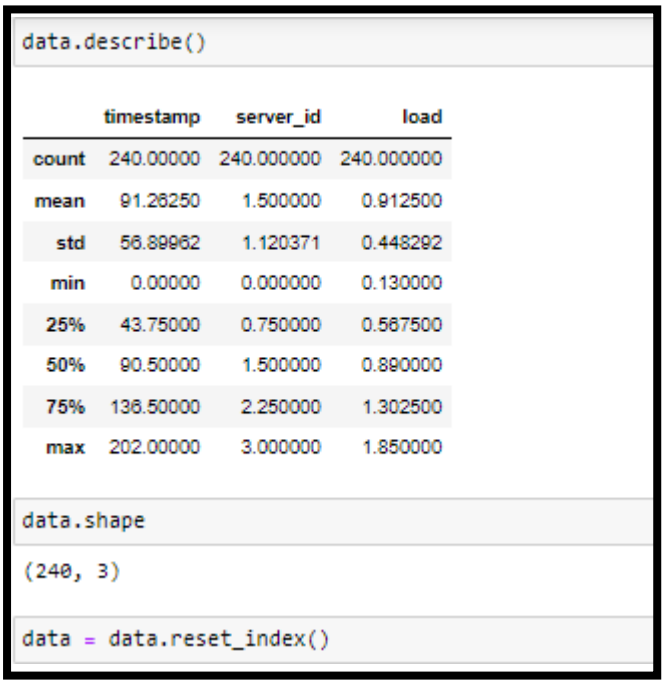
Figure 5: Development of Artificial Neural Network Model

6. Evaluation

This section of the paper talks about an entire analytical aspect emphasising the processes required in exhibiting the dynamic load balancing. Furthemore, these procedures display the processes employed in the implementation of machine learning in order to manage resources during pandemic conditions.

The ongoing procedure makes use of the procedures required in developing load balancing with the use of Python language programs run in Jupyter Notebook. In order to conduct the entire process effectively, server time datasets are selected comprising serverloads and their timestamps are also chosen. This helps in the employment of machine learning processes like Artificial Neural Network (ANN) and Linear Regression.

6.1. Experiment / Case Study 1



```
data.describe()
```

	timestamp	server_id	load
count	240.00000	240.000000	240.000000
mean	91.26250	1.500000	0.912500
std	56.89962	1.120371	0.448292
min	0.00000	0.000000	0.130000
25%	43.75000	0.750000	0.567500
50%	90.50000	1.500000	0.890000
75%	136.50000	2.250000	1.302500
max	202.00000	3.000000	1.850000

```
data.shape
```

```
(240, 3)
```

```
data = data.reset_index()
```

Figure 7: Dataset Analysis
(Source: Produced in Jupyter notebook)

The figure in the above section is considered to obtain the descriptive statistics about the dataset finding out the maximum, minimum, mean, standard deviation, and count value for all the columns from the Dataset.

```
sns.jointplot(data=data, x="load", y="timestamp")
```

```
<seaborn.axisgrid.JointGrid at 0x1da7615ee80>
```

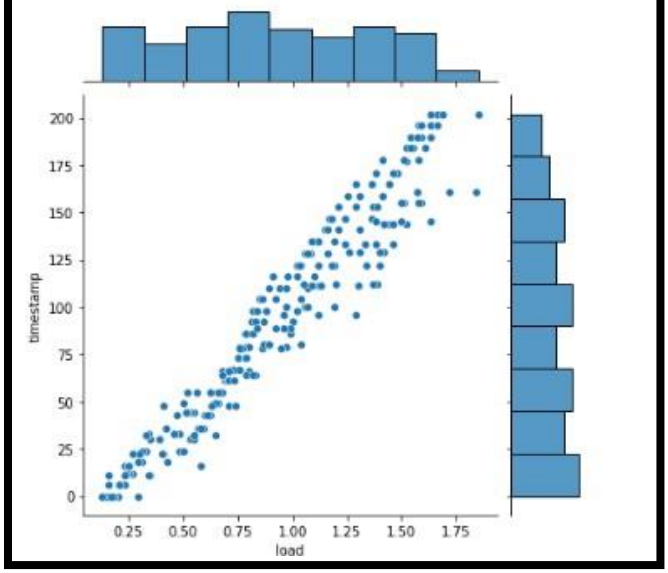


Figure 8: Joint Plot Visualization procedure
(Source: Produced in Jupyter notebook)

6.2. Experiment / Case Study 2

```
test_pred = model.predict(X_test)
train_pred = model.predict(X_train)
print('Test set evaluation:\n_____________________________________')
print_evaluate(y_test, test_pred)
print('Train set evaluation:\n_____________________________________')
print_evaluate(y_train, train_pred)
results_df_2 = pd.DataFrame(data=[["Artficial Neural Network", *evaluate(y_test, test_pred), 0]],
                            columns=['Model', 'MAE', 'MSE', 'RMSE', 'R2 Square', 'Cross Validation'])
results_df = results_df.append(results_df_2, ignore_index=True)
```

```
Test set evaluation:
_____________________________________
MAE: 0.13055536442746715
MSE: 0.030779134611534933
RMSE: 0.17543983188413895
R2_Square 0.8403570418707144
Train set evaluation:
_____________________________________
MAE: 0.14878610362298786
MSE: 0.04175059734145348
RMSE: 0.2043296291325697
R2_Square 0.7932194942423758
```

Figure 9: Accuracy of ANN
(Source: Produced in Jupyter notebook)

The procedure of obtaining results from both test and train data sets on implementing the ANN algorithm to obtain the required results. In addition to this, it can also be conveyed that the entire procedure has comprehensive results that are significantly accurate.

6.3. Discussion

This segment summarises the processes utilised in the deployment of two machine learning algorithms and simultaneously displaying their results. Further analysis concluded that the entire procedure is highly accurate with high quality outcomes. Furthermore, it states that the Linear Regression algorithm has demonstrated better output as the r square value is nearest to 1 and also its RMSE value is lowest among both the algorithms. The ongoing procedure makes use of the ways required in developing load balancing with the use of Python language programs run in Jupyter Notebook. In order to conduct the entire process effectively, server time datasets are selected comprising serverloads and their timestamps are also chosen. This helps in the employment of machine learning processes like Artificial Neural Network (ANN) and Linear Regression.

7. Conclusion and Future Work

7.1. Conclusion

In this research paper, the load balancing approaches and some applications of machine learning have been discussed. The process of load distribution explains the applications and importance of Jupyter notebook. The application of logistic regression is available which explains the binary of the server's state. The methodologies that are necessary in task completion have been efficiently maintained and managed. The objectives of the research have been written and elaborated appropriately and following these the literature review parthas been concluded thoroughly with detailed information about load balancing.

In order to complete the research study, the data collected from the sources have been produced in the Jupyter notebook. After further analysis, the status of servers being overloaded or underloaded can be determined and hence, network traffic can be routed easily based on the analysis. Activities such as task scheduling, resource allocation, usertask requirement identification, and migration are involved in the load balancing process.

Python programming language has been used in order to implement the load balancing model. The accuracy can be determined with the help of analysis. In this case, ANN model exhibits proper accuracy and it can be concluded that this model can help in building load balancers. Moreover, it can be assured that no particular person or ethnicity was harmed during this research.

7.2. Future Work

In this particular research study, appropriate data analysis techniques have been applied and the topics regarding load balancing have been explained clearly (Devaraj et al. 2020). In order to maintain stability, applications regarding distributed algorithms are required to be applied. Load balancing is a highly systematic process and aims to distribute workloads,

while network traffic leads to network hazards and system failure. Cloud load balancing simply detects the traffic and systematically adds new servers in the traffic distribution list and evenly distributes workload among the group of servers. Data plotting is conducted which assists in the analyses and verification of direction as well as data correlation.

Dependent variable is a variable utilized in the process of linear regression. The dependent variable facilitates the predictions for the other variable called the independent variable. Conducting ANN algorithm after linear regression produces advantages such as the ability to detect complex relationships, and a less formal statistical form of training (Abiodun et al. 2018). In regression models, the target prediction values are largely independent variables. The value conveys the relationship between variables and forecasting. Moreover, values can be easily predicted regarding the dependent variable with the assistance of independent variables by initializing the parameters.

References

Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A., & Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, *4*(11), e00938.

Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Umar, A. M., Linus, O. U., ... & Kiru, M. U. (2019). Comprehensive review of artificial neural network applications to pattern recognition. *IEEE Access*, *7*, 158820-158846.

Afzal, S., & Kavitha, G. (2019). Load balancing in cloud computing–A hierarchical taxonomical classification. *Journal of Cloud Computing*, *8*(1), 1-24.

Devaraj, A. F. S., Elhoseny, M., Dhanasekaran, S., Lydia, E. L., & Shankar, K. (2020). Hybridization of firefly and improved multi-objective particle swarm optimization algorithm for energy efficient load balancing in cloud computing environments. *Journal of Parallel and Distributed Computing*, *142*, 36-45.

Ebadifard, F., Babamir, S. M., & Barani, S. (2020, April). A dynamic task scheduling algorithm improved by load balancing in cloud computing. In *2020 6th International Conference on Web Research (ICWR)* (pp. 177-183). IEEE.

Greener, J. G., Kandathil, S. M., Moffat, L., & Jones, D. T. (2022). A guide to machine learning for biologists. *Nature Reviews Molecular Cell Biology*, *23*(1), 40-55.

Kadam, A. K., Wagh, V. M., Muley, A. A., Umrikar, B. N., & Sankhua, R. N. (2019). Prediction of water quality index using artificial neural network and multiple linear regression modelling approach in Shivganga River basin, India. *Modeling Earth Systems and Environment*, *5*(3), 951-962.

Mishra, K., & Majhi, S. (2020). A state-of-art on cloud load balancing algorithms. *International Journal of computing and digital systems*, *9*(2), 201-220.

Mishra, S. K., Sahoo, B., & Parida, P. P. (2020). Load balancing in cloud computing: a big picture. *Journal of King Saud University-Computer and Information Sciences*, *32*(2), 149-158.

Mohanty, S., Patra, P. K., Sahoo, S. S., & Mohanty, A. (2017). Forecasting of solar energy with application for a growing economy like India: Survey and implication. *Renewable and Sustainable Energy Reviews*, *78*, 539-553.

Neelima, P., & Reddy, A. (2020). An efficient load balancing system using adaptive dragonfly algorithm in cloud computing. *Cluster Computing*, *23*(4), 2891-2899.

Tavares Júnior, I. D. S., Rocha, J. E. C. D., Ebling, Â. A., Chaves, A. D. S., Zanuncio, J. C., Farias, A. A., & Leite, H. G. (2019). Artificial neural networks and linear regression reduce sample intensity to predict the commercial volume of Eucalyptus clones. *Forests*, *10*(3), 268.

Ullah, A. (2019). Artificial bee colony algorithm used for load balancing in cloud computing. *IAES International Journal of Artificial Intelligence*, *8*(2), 156.