UNIVERSITY OF BOTSWANA

DEPARTMENT OF COMPUTER SCIENCE

TRAFFIC PREDICTION IN CLOUD COMPUTING

USING TIME SERIES MODELS

By:
KEFENTSE MOKGOLODI

THIS DISSERTATION IS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF MASTER OF SCIENCE IN COMPUTER INFORMATION SYSTEMS, UNIVERSITY OF BOTSWANA.

SUPERVISOR:
Dr. G. Anderson

May 2015

ABSTRACT

The concept of Cloud Computing has brought a new dimension in the IT industry by offering online services to the international community on pay-as-you go basis. The architecture allows the clients to pay for what they use, minimizing the operational costs and leaving the hassle of system support to the service providers. Although cloud computing is a good and an attractive concept to the business community; it still has some challenges such as traffic management that need to be addressed. Most network service providers experience traffic congestion when messages arrive and queue at the service point due to limited processing capacity. That is what this research intends to mitigate by proposing a proactive traffic prediction model that would play a role in congestion control and estimation of accurate future resource demand in order to achieve guaranteed Quality of Service (QoS) rather than depending on reactive resource provisioning models.

For this research, four time series models being Single Exponential Smoothing (SES), Double Exponential Smoothing (DES), Triple Exponential Smoothing (TES) and Autoregressive Integrated Moving Average (ARIMA) are compared in order to identify the one that can accurately predict the future performance of an IaaS cloud provider. The research is based on restricted traffic measurements which consist of the number of users and the amount of downloaded data taken from CAIDA database. More emphasis is put on the amount of traffic that goes through cloud network. The measurements used have monthly and yearly data making it a total of four (4) datasets. All the coding, testing of the hypothesis and the statistical work of the project are performed using R- programming language. From the study, the predictions of ARIMA are more accurate as compared to those of others and have smallest RMSE and MAE. It was also observed that all the models perform better on monthly data as compared to yearly data.

ACKNOWLEDGEMENT

First and foremost, my utmost gratitude goes to Dr. George Anderson, whose guidance, support and encouragement has made everything in the research work possible. My thanks go to CAIDA organization, which provided data that was used in this research project.

Next, I would like to thank my colleagues in the department. Their great contribution, support and friendship have been instrumental on both academic and personal level. I would like to sincerely thank my family for the unflinching support they gave me as I hurdle through all the obstacles of completion this dissertation.

Finally I am most grateful to my dear wife, Kebaabetswe Mokgolodi, for proofreading my research work most of the time. Her everlasting support has encouraged me to complete this research work.

Thank you all.

TABLE OF CONTENTS

Content **Page**

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ACRONYMS

ACF	Autocorrelation Function
AIC	Akaike Information Criteria
AR	Auto-regressive
ARIMA	Autoregressive Integrated Moving Average
ARMA	Autoregressive Moving Average
AWS	Amazon Web Services
CAIDA	Center for Applied Internet Data Analysis
CPU	Central Processing Unit
CRAN	Comprehensive R Archive Network
CRM	Customer Relationship Management
CSP	Cloud Service Provider
CSV	Comma Separated Value
DAG	Data Acquisition and Generation
DES	Double Exponential Smoothing
EaLMS	Error-adjusted Least-Mean Square
EC2	Elastic Compute Cloud
ES	Exponential Smoothing
FARIMA	Fractional Autoregressive Integrated Moving Average
GiB	Gibibytes
HW	Holt-winters method
IaaS	Infrastructure-as-a-Service
IDC	International Data Corporation
IMDC	Internet Measurement Data Catalog
ISP	Internet Service Provider
IT	Information Technology

LAN	Local Area Network
LMS	Least-Mean Square
LRD	Long-range dependent
MAE	Mean Absolute Error
NaN	Not a Number error
NIST	National Institute of Standards and Technology
NSFNET	National Science Foundation Network
PaaS	Platform-as-a-Service
PACF	Partial Autocorrelation Function
PC	Personal Computer
QoS	Quality of Service
RMSE	Root Mean Square Error
SaaS	Software-as-a-Service
SES	Single Exponential Smoothing
SIMR	Scalable Internet Measurement Repository
SLA	Service Level Agreement
SMA	Simple Moving Average
SNMP	Simple Network Management Protocol
SRD	Short-range dependent
TES	Triple Exponential Smoothing
VM	Virtual Machine
VoD	Video on Demand

CHAPTER 1 - INTRODUCTION

1.1 Overview

The advent of cloud computing has gained a huge popularity on Information Technology (IT) as more cloud service providers endeavor to provide powerful and reliable cloud platforms. According to National Institute of Standards and Technology (NIST) [1] Cloud Computing is “a model for enabling convenient, on-demand network access to a shared pool of configurable Computing resources”. It is indeed a new technological trend that allows the deployment and delivery of application services to the users via the internet. Cloud computing represent a fundamental change in the way information technology services are developed and deployed to the international community in a pay-as-you-use manner. The approach is cost effective to the business owners who do not have enough capital to setup their own IT infrastructure. Cloud computing implements the idea of utility computing, which was first suggested by John McCarthy [2] in 1961, where cloud computing is viewed as a public utility. It is now a reality as more and more users continue to rent computing as a service, moving the processing power and storage to centralized infrastructures rather than located in client hardware.

Cloud computing is classified depending on the services they offer and these services are Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS). In IaaS, processing power, storage, networks and other fundamental computing resources are provided by the service provider and rented to the cloud users. The customers run their applications and services on the rented infrastructure instead of setting up theirs. In its latest Magic Quadrant report for IaaS providers (Figure 1), Gartner ranked Amazon and Microsoft as top IaaS providers [3]. The PaaS is a category that provides computing platform such as application hosting and deployment environment to the client. The consumer creates their software using tools and libraries from the provider or acquires the application and hosts it in the environment. SaaS provides specific and already-created applications and mostly in the form of web-based applications. The clients rent and use the software that they need in their organization without having to buy and install it in their own servers. An example of this service is Salesforce [4] that offers customer relationship management (CRM) tools.

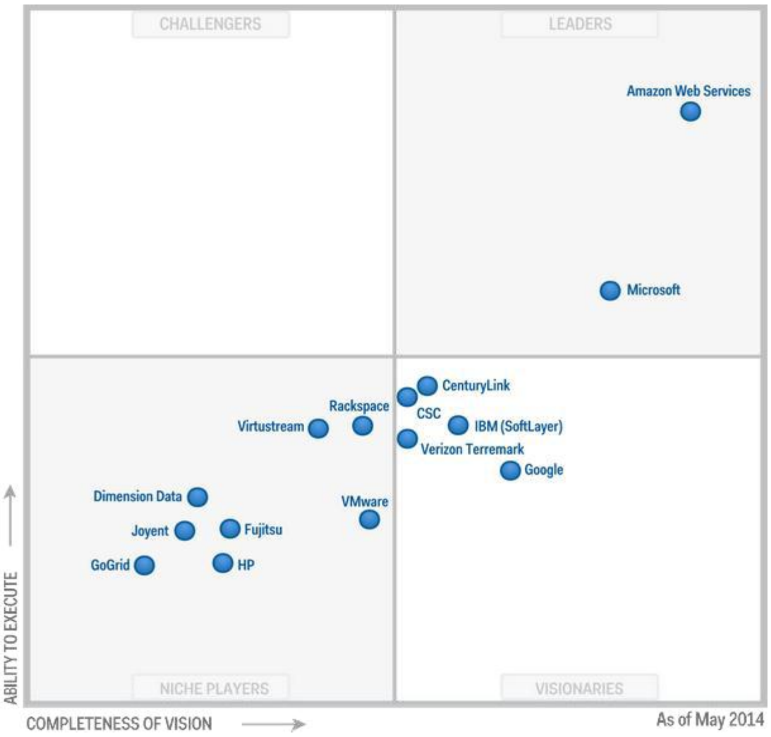

Figure 1: Magic Quadrant for Cloud IaaS (Source: Gartner, May 2014)

The cloud computing services can be deployed in public, community, private or hybrid cloud. Public Cloud is made available to the general public in a pay-as-you go manner. Generally, public cloud service providers own and operate the infrastructure and offer access only via Internet but not direct connectivity. Users are given little or no control over the cloud environment as a way of reducing risk to security. In private cloud, the infrastructure is operated solely for a single organization. It can be managed internally or by a third-party and hosted internally or externally. Organizations using private cloud have great control on the cloud because it runs on the secure self-controlled network boundary [5]. Although private clouds are secure, they are more costly to manage as compared to public clouds. A community cloud involves sharing of computing infrastructure amongst organizations with shared concerns. For example government institutions within one country may share computing resources. A hybrid cloud service is a combination of private and public cloud services from different service providers. An organization may store critical data such as sensitive client data in house on a private cloud application and host applications with relatively less security concerns on the public cloud.

This research is carried out using four time series models for prediction. These models are Single Exponential Smoothing (SES), Double Exponential Smoothing (DES), Triple Exponential Smoothing (TES) and Autoregressive Integrated Moving Average (ARIMA). The research is centered on the traffic datasets taken from CAIDA (Center for Applied Internet Data Analysis) [6], which is hosted on the Equinix IaaS cloud service provider centers. The four CAIDA datasets capture the number of users who access passive cloud measurements. They also capture the amount of downloaded datasets. Each of these datasets has monthly and yearly data, making it four (4) datasets in total. The R-programming suite is used for modelling, plotting the datasets and performing the statistical computations of the project. The IaaS cloud was chosen because it enables the researchers to launch virtual machines (VM) instances and give them access to other resources such as storage, memory and CPU time [7]. The researchers can start or stop the instances at any time for test purposes.

The research is instrumental to any cloud service provider especially the IaaS cloud providers. The providers can use the concept in order to know the performance of their systems as well as traffic that flows through their network. That will enable them to make management and planning decisions. For example if TES model can be used to predict CPU utilization of one of Amazon’s EC2 servers for 2016, and the results show a growth in the usage by double. Depending on the current CPU performance, Amazon has to double the processing power of that server in order to avoid any unforeseen circumstances. Managing variable load is one of the crucial issues in any network including cloud computing environment because it impacts on service delivery. Most cloud providers use reactive approaches in order to plan for future growth, the situation which is not adorable.

1.2 Statement of the problem

Network traffic prediction is one of the important concepts in many areas such as traffic engineering, network management and congestion control. The same applies to cloud computing where service providers are bound to providing quality service. Network traffic can create challenges in cloud computing and hamper service delivery due to limited processing capacity. That is crucial because network congestion is mostly caused by less processing power. At the same time overprovisioning of resources can lead to more power consumption as well as wasting of VM resources.

Currently most cloud computing providers use a hybrid cloud which utilizes both private and public clouds to manage load variations. In that way the resources are allocated based on increased workload which may result in queuing. That model is reactive and is impractical in real time situations. This research intends to mitigate that by proposing a proactive traffic prediction model that will be able to accurately predict the amount of traffic in advance as a way of avoiding network congestion. It specifically addresses the problems that arise related to the performance of network or applications running in clouds. The analysis is based on passive measurements taken from CAIDA and the experiments are performed using R-language. The results of the analysis would assist the cloud service providers who might run in to network performance problems.

1.3 Research Objectives

The research objectives are as follows;

1. The main objective of this work is to compare four time series models and identify the one that can forecast future traffic variations as precisely as possible based on the passive measurements of the traffic history. That includes predicting the number of clients who connect to CAIDA databases as well as the amount of data they download. The accuracy of the prediction models will be achieved by using MAE and RMSE forecast accuracy methods.
2. The researcher also aims to train all the four prediction models in order to get the best model parameters. Each model is exposed to four datasets. The Exponential Smoothing (ES) models use alpha (α), beta (β) and gamma (γ) parameters while ARIMA uses Akaike Information Criteria (AIC). The parameters that give the smallest RMSE and MAE are considered to be the best.
3. The other aim of this research is to compare the performance of the models based on short term and long term data. For this research, the short term data is collected monthly and the long term one is collected yearly.

1.4 Methodology

The research will be carried out using a quantitative approach and it will involve three major phases. The first phase being the training and analysis stage involves tuning model parameters in order to find the best combination that gives the best fit. This stage will utilize the initial 70% of each dataset.

The second phase involves further analysis of the datasets measurements and forecast their future behavior using SES, DES, TES and ARIMA time series models. The predictions will be based on the last 30% of the datasets. The last phase will involve the analysis of the prediction models, using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) analysis methods. These two forecast analysis methods will be used to diagnose the variation of the errors in the forecast models in order to identify the most accurate model. All the statistical calculations, hypothesis testing and plotting of the diagrams will be performed using R-programming suite.

1.5 Scope of the dissertation

This research is based on the IaaS model because it does not provide any restrictions to the programmers as opposed to PaaS and SaaS. It allows for re-testing and repeating the experiments. All the four (4) datasets to be used in this research will be read and modeled using R language. The scope of this research specifically covers:

1. Obtain datasets from CAIDA database.
2. Use R-programming suite to train the models, perform the statistical calculations of the datasets and plot the graphs.
3. Forecast the future performance using the four prediction models.
4. Provide recommendation for the best prediction model.
5. Identify areas of possible future research related to this work

1.6 Research Hypothesis

The following two hypotheses have been proposed with their justifications;

1. The ARIMA model will perform better due to its ability to capture short-range dependence (SRD) and best at modeling stationary data [8]. The performance of the other three ES models will be determined by the number of parameters they use for modelling. TES will be the second best model because it involves three parameters for its modelling followed by DES which uses two parameters. SES will be the last since it is modelled with only one parameter.
2. All the four (4) prediction models will perform better on short term data as opposed to long term data. This is likely to be the case because time series models depend on past values in order to predict the future. Recent data is likely to be accurately predicted as opposed to the one captured long time back.

1.7 Significance of study

This research is vital since it is one of the most desirable trends in computer technology. Its outcome shall not only benefit system administrators dealing with cloud computing but also investors who need to host their services in the cloud. Hence, it is based on real hypothesis and factual events happening across the globe [9]. It will also give suggestions to some of the best models one can use to predict the network traffic of a given cloud computing system.

The research also explores extend at which the models can performance based on long term and short term data. Models perform differently when exposed to different kinds of data. The key to accurate traffic prediction in cloud computing is proper modeling of the relationship between real time historic data and future values and hence the use of time series models. Time series models can best tell the future based on the past trends of data [10], [11].

1.8 Limitations of the research

Under ideal circumstances, a research is expected to be carried out within the required specified time without any hindrances. However constraints are likely to be encountered during the research work which may delay progress. The following have been identified as the main limitations for this research work;

- Most of the datasets measurements procedures are costly. Some big cloud providers such as Amazon can allow researchers to use their computing resources for testing purposes at a cost [12]. The researchers pay for the resources used for performing datasets measurements. For this research, Amazon Web Services (AWS) registration was initially done in order to measure and collect CPU utilization datasets. At first registration used AWS free Tier. During this period CPU utilization of EC2 instances were measured. After the free tier period has elapse Amazon requested for the monthly payments. The researcher could not cope with the costs and ended up changing to the already collected datasets from CAIDA which did not come at a cost.

- Some of the datasets found in the internet need specific tools for decryption. An example is that of Wide Area Performance Measurements which were downloaded from *http://pages.cs.wisc.edu/~keqhe/cloudmeasure_ datasets.html.* The datasets required the use of specialized tool which was not readily available in order to open the *.tar* files. According to the information posted on their website, the dataset contains throughput of EC2 hosted servers, but did not avail the tool that decrypts it. Other tools such as winzip and 7-zip did not display the data correctly. Some researchers end up using wrong tools for decoding such data which may have negative impact on the results. Although such datasets have been used in research, Kandula et al [13] are uncomfortable with their standard remarking on the incompatible of the databases.

1.9 Dissertation Structure

The remainder of this dissertation is organized as follows;

CHAPTER 2 - LITERATURE REVIEW: This chapter reviews critically relevant existing literature pertaining to the research project. The draw backs faced in cloud traffic management case studies were also analyzed.

CHAPTER 3 - RESEARCH METHODOLOGY: The third chapter consists of various methods used in the research to obtain the required results for data analysis.

CHAPTER 4 - RESULTS OF DATA ANALYSIS: The results obtained in this research are analyzed in this chapter. This mainly includes training and prediction results for different prediction models which are presented in tabular and line graphs.

CHAPTER 5 – SUMMARY AND CONCLUSION: This chapter concludes the dissertation, highlighting the research evaluation, challenges encountered, summary of dissertation and findings learnt during the course of the project. This chapter also outlines possible future research work in relation to this project.

CHAPTER 2 - LITERATURE REVIEW

2.1 Introduction

The concept of cloud computing is still evolving and has so far been welcome by the business community as a sharp departure from the traditional method of procuring data center infrastructure which takes several months to implement. More emphasis is now on renting the infrastructure resources or services. It has the potential to make IT organizations more responsive than ever due to flexibility and infinite elasticity. Gartner [14] being the world's leading information technology research and advisory company identified cloud computing amongst the top 10 strategic technology trends for 2014.

The implementation of cloud data centers involves ample bandwidth between the data centers to cater for traffic. The growth in cloud-based traffic is largely due to the economies of scale that virtualization and cloud computing provides. According to Savage [15], by 2017, consumers will generate 81% of cloud traffic compared to 19% by businesses. Their study also states that the vast majority of the data center traffic is caused by the data centers and clouds themselves undertaking activities like backups and replication. Cisco Global cloud index [16] also forecasted that datacenter traffic will grow four times by 2016 to reach a total of 6.6 zettabytes annually. Based on the above studies it is imperative that service providers need to forecast the performance of their datacenter so as to avoid any future traffic congestion.

As people continued to see the needs of cloud computing and its reputable importance, various scientists developed theories and scientific algorithms on how such predictions could be easily found out and be calculated in advance to facilitate the choice of models for computer network development. One of the most sophisticated prediction methods is exponential smoothing. This method has single, double and triple variants which have the ability to forecast data with different characteristics such as trend and seasonality.

2.2 Traffic management in cloud

Characterizing and managing network traffic is becoming a more complex task, especially with cloud computing where traffic changes rapidly [17]. Traffic management is crucial to consumers; hence service providers need to ensure that network resources operate at an acceptable performance. Wolski [18] developed one of the first network measurement systems for predictions. In their research they ran multiple predictors (mean and AR models) with the aim of selecting the one with smallest prediction error. The same was done by Xinyu et al [19] who attempted to improve the least-mean square (LMS) predictor and called it Error-adjusted LMS (EaLMS). The main idea of EaLMS was to decrease the LMS prediction errors. From their work they found that traffic can be best predicted using the prediction method that gives smallest error but no work has been done to compare EaLMS and AR models.

Some researchers have carried out intensive network characteristics of datacenters. For example, Benson [20] observed ten (10) datacenters by characterizing the internal network traffic and studying the applications deployed in the datacenter. The datacenters used belong to three types of organizations being the university, private enterprise and public cloud. They performed the study of traffic at the edges of a datacenter by collecting and observing the SNMP traces from routers and identifying ON-OFF characteristics. Their research was also aimed at examining the applications deployed in datacenters and identifying their impact of network utilization as well as on congestion and packet loss. The above researches are based on traffic management, the area that is crucial for any network administration. They measure and forecast the performance of network applications. This research is more concentrated on the use of traffic datasets downloaded from the cloud environment as a way of envisaging its future behavior. The choice of the cloud environment is compelled by the fact that nowadays more businesses and organizations prefer cloud hosting [15], [16].

2.3 Cloud workload and resource allocation

Most of the time traffic congestion is caused by under provisioning of resources. Bhavani et al [21] did a survey on Static and Dynamic Resource Provisioning Techniques. Their aim was to investigate the advantages and the disadvantages of the different techniques and how they can be deployed in order to meet the Quality of Service (QoS) parameters like availability, throughput and reliability. The twelve (12) techniques used prove that they indeed improve response time and performance of applications with the ultimate goal of maximizing profit from the Cloud Service Provider's Perspective and reducing cost on the Cloud User's Perspective.

In their research Gandhi et al [22] carried a practical hybrid approach of timely availing server resources in data centers. They did it in a way that SLA violations and energy consumption are minimized. Firstly they used a discretization technique on historical workload dataset to identify workload demand patterns. From there, they applied a predictive provisioning technique to handle predicted load at "coarse" time scales (hours) and simultaneously used reactive provisioning to handle any excess workload at "finer" time scales (minutes). From their research it came out that the hybrid provisioning approach performed better as compared to using reactive or predictive provisioning showing a decrease in SLA violation.

Some researchers have proposed predictive and dynamic resource provisioning mostly for VMs and web applications in areas of CPU utilization [23], [24]. Bankole [25] investigated the best way to avoid under provisioning as well as avoid over provisioning of resources. They carried out the research on Amazon Elastic Compute Cloud (EC2) by predicting future CPU and network utilization using three machine learning techniques. The research was done with the aim of coming up with an optimal resource provisioning which would improve performance as well as reduce cost and traffic congestion.

Zaharia et al [26] did their research on how cloud computing could be used to adapt to bandwidth-intensive applications such as MapReduce computations. Their research also covered Video-on-demand (VoD) market that uses network for transferring large amounts of data at high rates. They carried their research as a way of addressing network challenges observed in heterogeneous environments. In 2010 one of the major VoD providers, Netflix moved its data store, video and streaming servers to Amazon AWS in order to enjoy the benefits of the cloud [12]. The idea behind was to enable users to stream Netflix shows and movies from anywhere in the world. Now Netflix streaming-video service is running live across multiple regions of the AWS cloud. All these recent advances enable efficient and quality-assured management of video workload in the cloud. Dickey and Fuller [27] did some experiments for studying the limiting distribution of the OLS estimator of autoregressive models for time series with a simple unit root. Their test experiments proved that when a time series has a unit root, the series is non-stationary.

In their paper Zhang et al [28] did a research on some of the challenges of cloud computing where they identified automated service provision and traffic management amongst them. The service providers have to ensure that resources are allocated and de-allocated from the cloud to satisfy its service level objectives while minimizing its operational cost. That involves analysis of network traffic on datacenters. The area is crucial because network managers need to know the type and size of traffic that flows through their network in order to make planned managerial decisions. This research is also concerned about the workload that passes through the network which impacts on quality of service factors such as reliability and throughput. The research uses Box–Jenkins model (ARIMA), which has the ability of capturing short-range dependence (SRD). The researcher considers it as the best model for cloud environment because it can handle data stationarity well [27].

2.4 Datacenter traffic datasets measurements

Traffic measurement and monitoring is considered as a key research factor in designing improved datacenter networks. That is so because there is a tremendous interest in carrying out research using the existing historical datasets which has its own advantages and disadvantages. Currently there are several intensive studies related to traffic measurements done by many European research projects [29]. Network traffic measurements are imperative in establishing accurate network models and enhancing network performance and quality of service. They can be divided into active measurements and passive measurements. Passive measurements are usually launched through mirroring and monitoring traffic of a certain link as it passes by. On the other hand active measurements involve injecting some test packets into the network in order to measure the traffic performance. Passive measurements as compared to active measurements impose no interference on the operation of the network and can reflect network behavior most accurately [30].

Other databases store information in the form of meta-data, i.e. "data about data". In that case integrated meta-data models are availed for annotating various measurement tools and measurement data. That is accomplished by providing links to tools which can be used to analyze the data. The approach has been acknowledged by several authors as an effective procedure for creating unified access to heterogeneous data. For example, Allman et al [31] proposed a Scalable Internet Measurement Repository (SIMR) system consisting of measurement repositories, clients and a centralized database. The measurement repositories are used for storing actual measurement results and the clients are the researchers who can download the measurement data. The centralized database coordinates the access of the clients to the measurement repositories. The similar approach was implemented by the CAIDA project [32] for correlating heterogeneous measurement data to achieve system-level analysis of internet traffic trends with the goal of creating and populating the Internet Measurement Data Catalog (IMDC) that will facilitate access and long-term storage of Internet data as well as sharing of data amongst the researchers.

2.4.1 Limitations and opportunities of datasets measurements

Even though measurement and archiving of data may be seen as a solution to researchers who do not have time for collecting data for themselves, others argue that it might not give the exact result of what the researcher is looking for. For example, Kandula et al [13] are uncomfortable with the standard of dataset measurement, remarking that the tools and data formats used by different databases are different and often incompatible. The researcher would need a specific tool in order to access appropriate data from many uncoordinated repositories. That may be a challenge due to lack of a standard way of describing the assumed measurement scenario and environment. Their argument is based on the fact that the tools used for data collection may have a negative impact on the analysis of the results which may not be known to the researcher.

In their research Akella et al [33] discussed the challenges of choosing traffic sources or interconnected collection of Autonomous Systems (ASes) when measuring traffic. The process includes measuring traffic flows between a set of sources, and choosing a set of destinations. They highlighted that, stub ASes in the internet vary in size and connectivity to their carrier networks. Large stubs such as large universities and commercial organizations often have high speed links to all of their providers while other stubs such as small businesses usually have a single provider with a much slower connection. That on its own is a challenge because it is difficult to use such measurements when the source network or its connection to the upstream carrier network is itself a bottleneck. They also discussed the challenges of network delay which may be attributed to by choosing a wrong path to measure from their source.

On the other hand, Daniel [34] states that although the use of dataset is unsuitable for professional researchers, it is appropriate for academic research because it is not costly and it saves a lot of time and resources. Academic researchers do not have much time to collect data for themselves when carrying out their research. One thing to note is that the tools for collecting and analyzing data also come at a cost. Other cloud providers like Amazon have the provision for charging the researchers for using AWS resources. It is also not possible for cloud providers to give the researchers access to monitor performance of their systems due to security issues. In that case the only way is to depend on already collected datasets such as the ones used in this research. This research is concentrative on the passive measurements because they are more accurate as they do not obstruct the normal operation of the network as opposed to active measurements. The research measurements are stored in a comma separated values (CSV) format and hence do not require any special tool for decryption.

2.5 Traffic prediction in Cloud computing

The future predictability of cloud computing traffic is challenged by the rapid changes in technology. Internet service providers rely entirely on their experiences for predicting cloud traffic which is quite uncertain. More work still needs to be done in network traffic prediction as a means of monitoring and managing cloud traffic. Sivakumar et al [10] analyzed traffic load of an access point(AP) using Hidden Markov Model and Neural Network Model prediction techniques to predict the number of devices connected to it. From their research they found that Neural Network produces best results. The researchers employed various traffic demand matrices in their WAN traffic engineering, something that is not feasible in cloud datacenters. In cloud datacenters traffic is elastic and is based on real-time where most of the longest-lived flows last only a few seconds or minutes [35]. Cloud computing services need to have the ability to adapt to workload changes by providing the resources when they are needed or ceasing the ones that are not needed at any point in time. By so doing the resources are matched with the current demands.

Using Hadoop as an example, Peng et al [36] developed a passive monitoring agent called HadoopWatch, which monitors Hadoop job meta-data and logs to forecast application traffic. They carried their research by observing rich traffic information in the log and meta-data files of many big data applications in the cloud. Their experiment that was deployed on small-scale testbed with 10 physical machines and 30 virtual machines revealed that HadoopWatch can forecast traffic demand with almost 100% accuracy. Their work was the first step towards comprehensive traffic forecasting through file system monitoring.

In their research, Sang and Li [37] proposed an approach for making predictability analysis of network traffic. The approach assesses the predictability of network traffic by considering how far into the future a traffic rate process can be predicted with bounded error and what is the minimum prediction error over a specified prediction time interval. For their research they used two stationary traffic models: the ARMA model and the Markov-Modulated Poisson Process (MMPP). They did their research to prove that the two models, though both short-range dependent, can capture statistics of self-similar traffic quite accurately for the limited considered time scales.

Schad et al [38] carried their research to find out if Amazon EC2 cloud could be ideal as a scientific research platform. They measured the performance of Amazon EC2 in terms of instance startup time, CPU performance, memory speed and bandwidth of network traffic. For their test they used one small instance and one large standard instance in different locations of the cloud. They then compared the usage of EC2 with their own local 10-computer physical cluster running a 50-node virtual cluster. Each physical computer ran the Linux Open Suse 11.1 operating system on a 2.66 GHz 64-bit Quad Core Xeon and three Gigabit network cards in bonding mode. From their results they concluded that Amazon EC2 is not sufficiently repeatable and reproducible environment which is undesirable for scientific measurements. Their results are supported by the fact that commercial clouds are used for quick and small experiments not for longer term researches.

Dalmazo et al [39] proposed a dynamic window size approach for online traffic prediction that can be incorporated with different traffic predictions mechanisms. The size of the window defining the amount of traffic that is to be considered for traffic prediction. For their research they used cloud computing dataset collected by monitoring the utilization of Dropbox. The evaluation of their proposed prediction mechanisms was performed by Normalized Mean Square Error and Mean Absolute Percent Error of predicted values over observed values. And from the results it shows that Poisson Moving Average approach is more suitable for dynamic cloud environments than Simple Moving Average, Weighted Moving Average and Exponential Moving Average.

Some researchers have evaluated a set of forecast algorithms in order to characterize them based on a specific traffic load. For example Papadopouli et al [40] described the Simple Moving Average (SMA) as the unweighted mean of the previous data points in the time series. In comparison with other complex predictors such as Autoregressive Integrated Moving Average (ARIMA), SMA is less demanding. In their research they emphasize some advantages of SMA, such as its simplicity, low complexity and ease of application.

Likewise the performances of some prediction tools have been put to test in high-speed networks. In [11] the researchers used ARIMA model to capture the detailed estimation of future NSFNET backbone traffic. They performed their study by conducting an in-depth study of modeling FDDI, Ethernet LAN, and NSFNET entry/exit point traffic using ARIMA models. From their research it came out that although ARIMA model can forecast network traffic accurately, it cannot perform automatic online prediction because it needs manual intervention in selecting the required model order. The other researchers, Qiao et al [41] compared the predictability of different types of forecasting methods based on a series of Box–Jenkins models, but concluded that no predictor suitable for performing online traffic prediction of network traffic has been found.

Most of the works in traffic forecasting addresses long period predictions that are important for IP network capacity planning. This research is different from others in that it focuses on the parameters of the training-based models used for both short and long period prediction of the traffic performance. To the best of the researcher's knowledge, none has analyzed traffic patterns of both short and long period prediction using SES, DES, TES and ARIMA in the same experiment. ARIMA was used because it has the ability to capture statistics with short-range dependence (SRD) accurately [8]. The above models have been used in LAN and wireless networks, for example Zhani et al [42] assessed trained based models like ARIMA and identified them to be accurate in a Local Area Network (LAN) environment. On the other hand exponential smoothing methods are relatively simple but widely used for forecasting inventory demand in business [43].

For this research more emphasis is on performing experiments in all of these models so as to test their predictability in a cloud environment. It is important to note that predictability analysis is one of the fundamental aspects of network management which has to be accurately performed.

CHAPTER 3 - RESEARCH METHODOLOGY

3.1 Introduction

This chapter gives an insight into the research methods used to assess and predict the future traffic behavior of CAIDA cloud network services. Experimental design emphasis is based on four (4) time series models using R-language. The initial stage involves the step by step procedure and selection of parameter values used for training the models. A quantitative approach was adopted since the research is based on numerical datasets collected from the cloud databases. The research method employed best satisfies the main research objective which is to compare and choose the best prediction model using forecast accuracy methods.

3.2 Centre for Applied Internet Data Analysis architecture

This research is based on data collected from Center for Applied Internet Data Analysis (CAIDA) website, which is *http://www.caida.org*. CAIDA is an independent analysis and research group based at the University of California's San Diego Supercomputer Center. It investigates both practical and theoretical aspects of the Internet, with particular focus on:

- Collection, analysis and modeling pertinent features and trends of current Internet usage.
- Create state-of-the art infrastructure for internet measurements and management,
- Improving the integrity of operational Internet measurement and management,
- Provide best available datasets and analysis tools to the research community

CAIDA is a collaborative responsibility among organizations in the commercial, government, and research sector which aims at promoting greater cooperation in the engineering of a robust and scalable internet infrastructure. They collect data from several large Cloud Service Providers (CSPs) at geographically diverse locations, and avail it to the research community while preserving the privacy of organizations which donated it. CAIDA’s infrastructure comprises of active and passive data monitors. For active monitors, CAIDA [44] uses a new active measurement platform called Archipelago (Ark) which was deployed in 2007 and the team-probing methodology for collecting accurate measurements. Currently it consists of a central server at CAIDA and about 107 active Archipelago (Ark) monitors deployed in 39 countries on 6 continents. Older monitors use standard PCs while all the monitors deployed since January 2013 are Raspberry Pi-based Network Monitors.

CAIDA’s passive data monitors are hosted in Equinix cloud providers to connect to its customers and partners inside the world’s most networked data centers [6]. The two data collection monitors in the United States are;

- Equinix-Chicago internet data collection monitor located at an Equinix datacenter in Chicago, Illinois. This datacenter is connected to Equinix-Seattle, Washington.
- Equinix-Sanjose internet data collection monitor located at an Equinix datacenter in San Jose, California. This datacenter is connected to Equnix-Los Angeles, California.

The above CAIDA’s monitors are connected to a backbone link of a Tier-1 ISP at a speed of 10GigE as shown in Figure 2. The ISP has multiple links between the cities which are used for load balancing. CAIDA uses Equinix cloud infrastructure which is an organization that provides industry-leading data center services, network connectivity and the interconnected clouds. Equinix [45] also provides hybrid cloud infrastructure services to SaaS providers as well as other enterprise services.

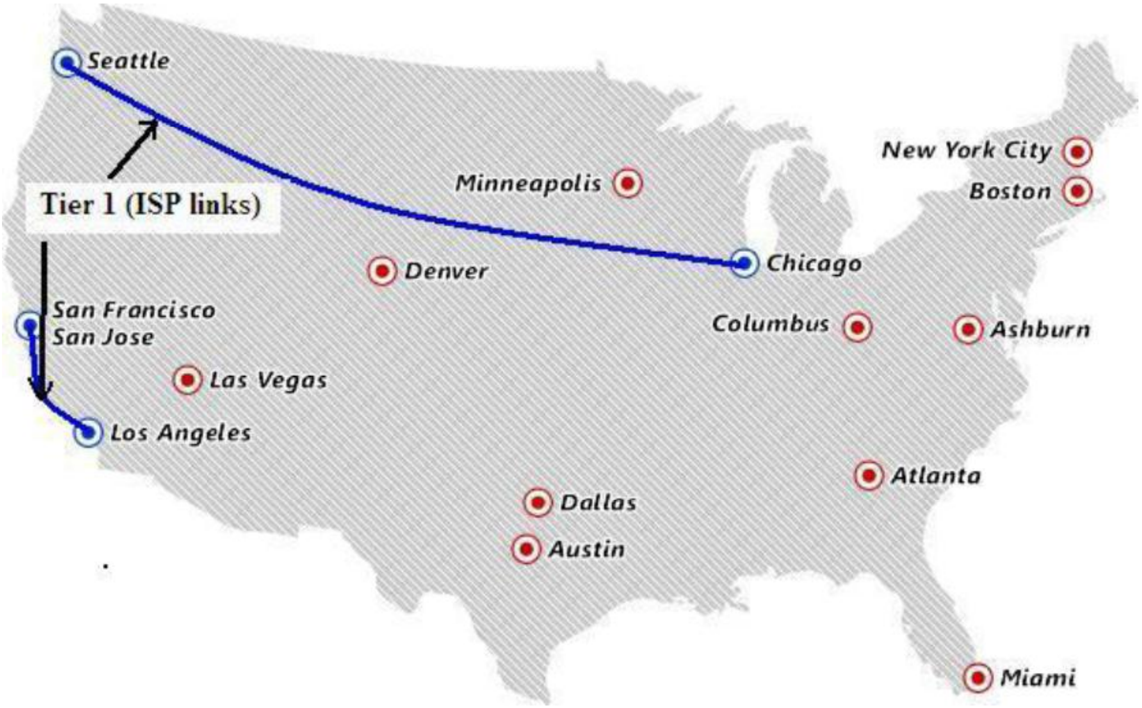

Figure 2: CAIDA data monitors connections in North US(Source: Google.com, Dec. 2014)

The infrastructure of CAIDA consists of two (2) physical machines, each having a single Endace 6.2 DAG network monitoring card [46]. Each DAG card connecting to a single direction of the bi-directional backbone link. The DAG measurement cards have their own internal high-precision clock that allows it to timestamp packets with 15 nanosecond precision. The two CAIDA machines have 2 Intel Dual-Core Xeon 3.00GHz CPUs, with 8 GB of memory and 1.3 TB of RAID5 data disk, running Linux 2.6.15 and DAG software version dag-2.5.7.1.

3.3 Collection of datasets

Datasets are mostly used in research for avoiding laborious process of deploying measurement infrastructure in datacenters which is costly. Nowadays competent organizations perform passive network measurements and publish them on their website for research purposes. It is upon the researcher to verify if the work is representative and appropriate for their particular research work. The data statistics used for this research were derived from the Center for Applied Internet Data Analysis (CAIDA) data server web logs from January 2005 to December 2014. They are available for download at http://www.caida.org/data/about/downloads/tables.xml, [47].

CAIDA gets the internet traces using its data connection monitors located at Equinix-Chicago and Equinix-San Jose datacenters. These anonymized internet traces includes but not limited to packet sizes of IPv4 and IPv6 traffic in bytes, transmission rate in packets per second and duration of traces. CAIDA then stores these traces which will later be accessed by users and other researchers. As users access these traces, CAIDA records the number of users as well as the amount of downloads they make. This research depends on these CAIDA's passive measurements to perform the predictions. The datasets consists of the number of users who accessed CAIDA's passive measurements and the amount of passive measurements they downloaded.

Following are the four primary passive datasets that are used in this research.

(i) Monthly amount of downloaded data

(ii) Yearly amount of downloaded data

(iii) Monthly number of users

(iv) Yearly number of users.

The first two (2) datasets contain the amount of restricted downloaded data captured monthly and yearly respectively and the same applies to the number of users' datasets. The number of users represents the number of unique accounts or IP addresses that accessed CAIDA web servers within a period of one month. Similarly, the calculated size of a download only counts unique files inside one month, i.e., if a file is downloaded multiple times by the same user in the same month it is counted only once. The amount of downloaded data is represented in gibibytes (GiB). The four datasets are as depicted in Figure 3 and Figure 4 .

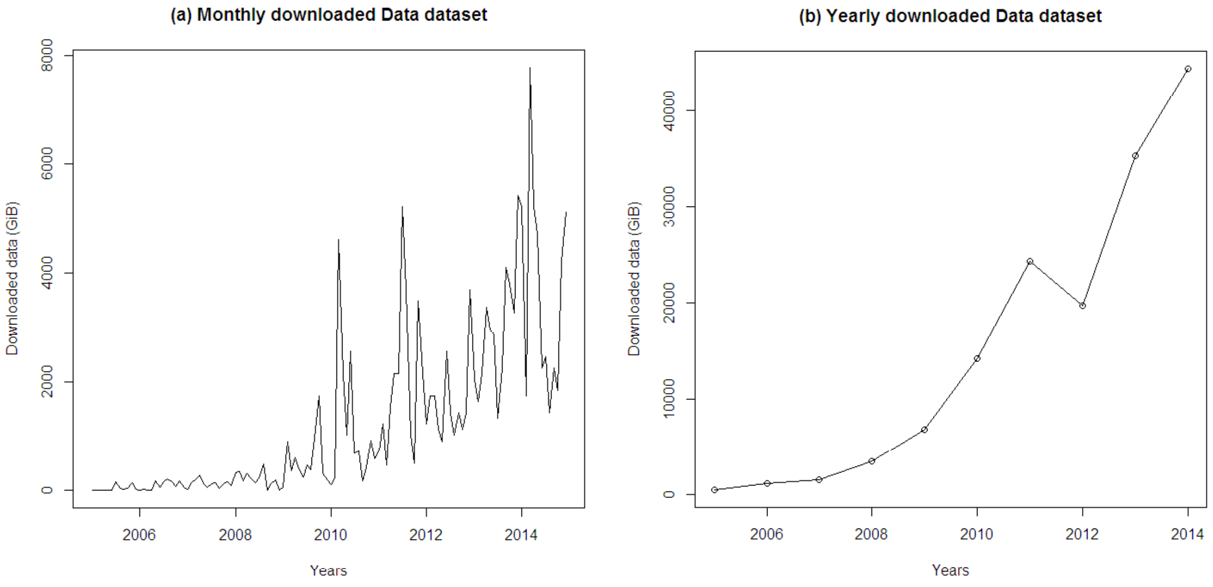

Figure 3: Amount of restricted downloaded data

The amount of restricted downloaded data has a minimal growth between 2005 and 2008. The data starts to experience some rapid growth from 2009 and rapidly growing in March 2010 where it reaches 4608GiB. There after an upwards trend is being experience up to year 2014.

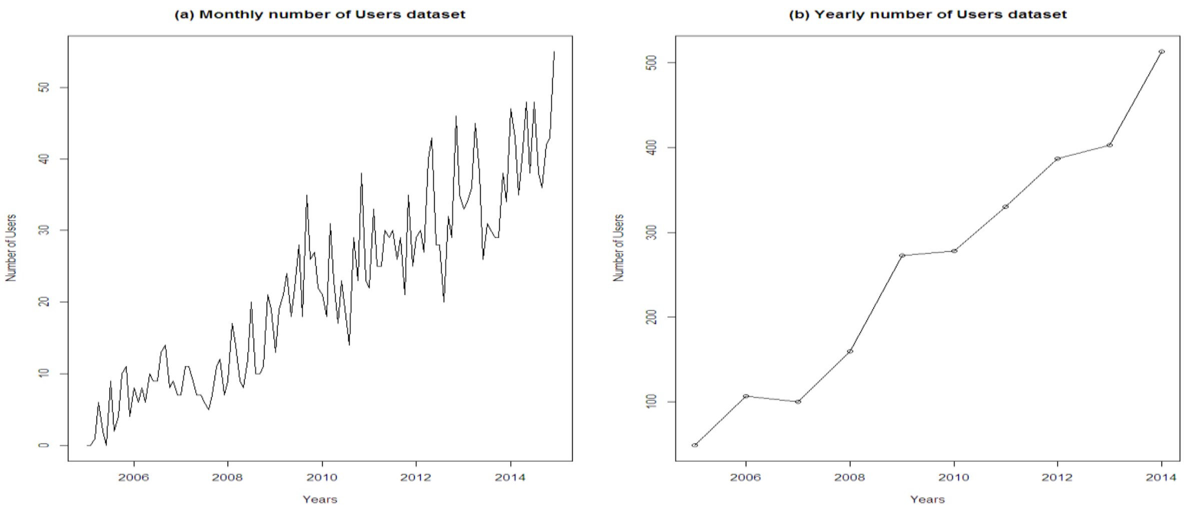

Figure 4: Number of unique users

The number_of_users captured by CAIDA dataset has an upwards trend which increases every year. Both the monthly data and the yearly data do not resemble any particular seasonality or any periodic repeated behavior. The data has a random variation behavior.

3.3.1 Datasets analysis

The process of evaluating quantitative data using mathematical models takes consideration of some assumptions. In this research the daily number_of_users datasets collected from CAIDA is ascertained by the Little's law. Users who access cloud services (webs server) spend some time accessing data and finally logs out. Little's law states that the average number of users (L) in a system is equal to the arrival rate (λ) of the user requests to the system multiplied by the average waiting time (W) each user request spends in the system [48].

$$\begin{aligned} &Average\ \ number\ of\ users(L) = arrival\ rate\ (\lambda)*average \\ &\qquad\qquad\qquad\qquad\qquad time\ spent\ \ in\ the\ system(W) \end{aligned} \qquad ??\ ?$$

3.4 R-programming Suite

R is a freely available language and environment for statistical computing and graphics which provides a wide variety of statistical and graphical techniques. It is an integrated suite of software facilities for data manipulation and for effective data handling. R is based on the "S" language which was developed at AT&T Bell Laboratories by John Chambers and colleagues [49]. The initial version of R was developed by Ross Ihaka and Robert Gentleman, both from the University of Auckland in 1966. Since its introduction R–project has gained many users and contributors, which continuously extend the capabilities of R by releasing add–ons (packages) that offer functions and methods that were not available in the previous versions. Currently the development of R is overseen by a `core team' of people widely drawn from different institutions worldwide to work on improving the functionality of the existing methods [50]. R is downloadable from *http://cran.r-project.org* and is distributed by the “Comprehensive R Archive Network” (CRAN).

The R-programming language used is R-3.1.2, 32-bit version for windows, which was the current available version when the research started in 2014. The environment has the ability to do arithmetic and statistical calculations as well as plotting well-designed quality graphs. It also has the ability to deal with trend and seasonality in time series. The only challenge with R language is that it uses command-line interface, which imposes a slightly steeper learning curve than other software. But once it is understood, R can perform any statistical data analysis. In addition “Metrics” and “forecast” statistical packages were installed as per the code below because they were not available in the downloaded R version.

```
> install.packages("forecast", dependencies=TRUE)
> library(forecast)
```

The forecast package mainly contains methods and tools for displaying and analysing univariate time series forecasts that includes exponential smoothing and ARIMA modelling. The Metrics package covers the accuracy methods such as RMSE and MAE.

3.5 Time Series models

A time series is a sequence of observations, usually collected at regular intervals. Time series can be divided in to two types being continuous and discrete. Discrete meaning that the observations are recorded in discrete times such as quarterly, monthly or weekly, whereas continuous means that observations are recorded continuously. The collected time series data makes much sense if displayed in the order in which they arose, particularly due to the interdependence of successive observations. Time series data is mostly used for forecasting in operation research with the help of decision models. It is worth noting that data which is periodically sampled at fixed intervals could be used to find reiterating patterns in traffic workload or to forecast future values. In that case the result of the time-series X will have the following observation:

$$X = x_{t-1}, x_{t-2}, \ldots, x_{t-w+1}$$

Time series can be decomposed into the following elements as depicted in Figure 5.

1. **Cycles (C_t)** - Cyclical fluctuations that are related to business or economic cycles or follow their own peculiar cycles, **(Figure 5 (a)).**
2. **Trend (T_t)** - Variations that move up or down in a reasonably predictable pattern, **(Figure 5 (b)).**
3. **Seasonal (I_t)** - Fluctuations that repeat over a specific period such as a day, week, month, **(Figure 5 (c)).**
4. **Random Variations (E_t)** - These are any random variations that do not fall under any of the above three classifications, **(Figure 5 (d)).**
5. **Parabola Trend-** It is the kind of trend that tend to accelerate fast. The variations produce a curved line, **Figure 5(e).**

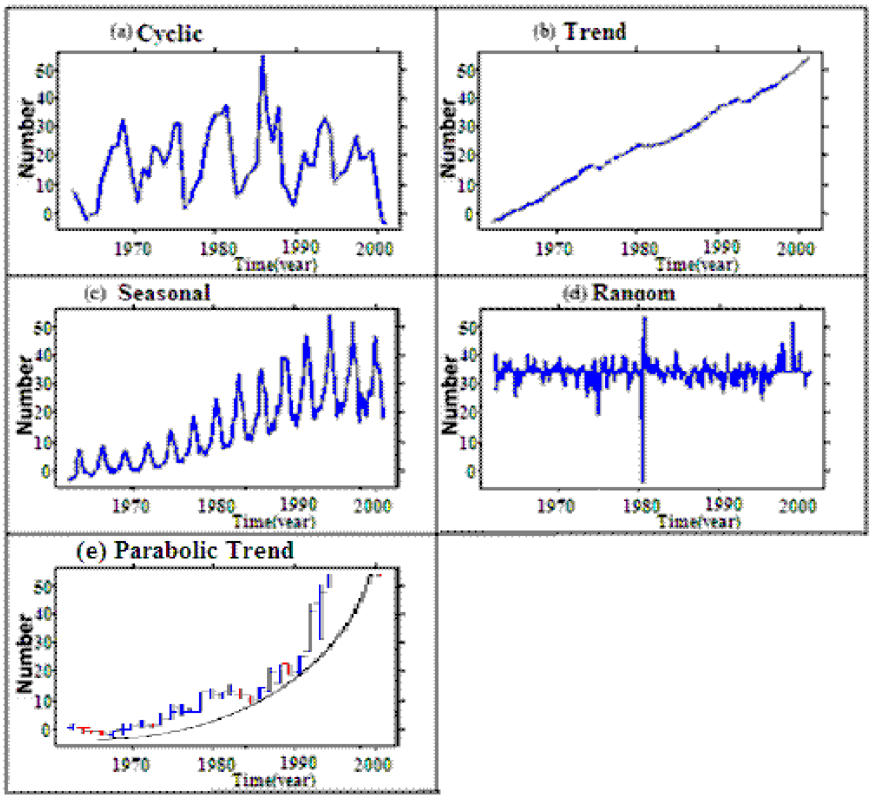

Figure 5: Examples of time series plots

The field of time series is vast and pervades many areas of science and economics particularly statistics. Applying time series models requires a step-by-step approach as depicted in Figure 6. In order to do forecasting effectively, the important step is to ensure that time series data is stationary; that is its statistical properties such as mean, variance, autocorrelation should all be constant over time [51]. If it is not, it will go through the process of removing level, trends and seasonality in order to make it stationary.

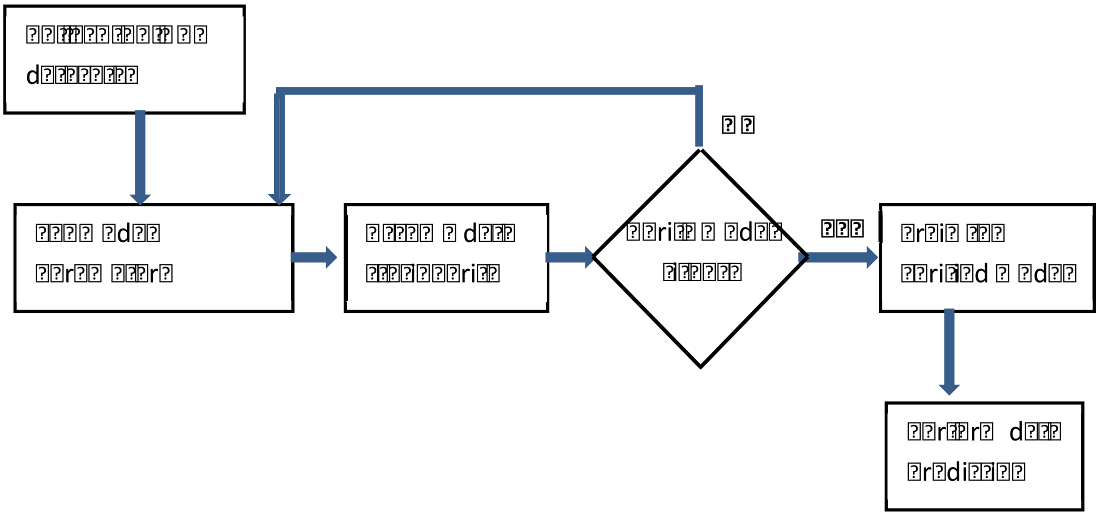

Figure 6: Procedure of using time series models

Most statistical forecasting methods are based on the assumption that data is stationary. The use of a sequence of transformations to stationarize a time series often provides important clues for which forecasting model to use. For example, stationarizing a time series through differencing is an important part of fitting an ARIMA model [52]. In this situation data will be transformed using ACF (Autocorrelation Function) and PACF (Partial Autocorrelation Function).

This research uses four time series models being ARIMA and the three types of Exponential Smoothing (ES) methods which are Single Exponential smoothing, Double Exponential smoothing and Triple Exponential Smoothing models. The selection of the models was necessitated by Hyndman and Khandakar [53] who articulated that the most popular automatic forecasting algorithms are based on either exponential smoothing or ARIMA models. They also stated that those algorithms are applicable to both seasonal and non-seasonal data. These methods have also been used by other researchers in LAN and wireless environments [11],[40],[42] and [43]. This research intends to test the predictability of these models when exposed to the cloud environment. Unlike other methods such as Simple Moving Average (SMA) whose averaging process seems to be capricious in that an observation is given full weight once and not the next time [39]. ES models adjust weights smoothly over time with the latest observation being given greater weights. This is so because all the past observations are important and need to be considered when predicting the future. Due to the above the researcher saw it fit to compare ARIMA with exponential smoothing methods.

3.5.1 Exponential Smoothing

Exponential Smoothing (ES) is a technique that can be applied to time series data, either to produce smoothed data for presentation, or to make forecasts. It uses mathematical recursive functions to predict the future trend. It does that by taking the previous forecast, and adjusts it up or down based on the actual value by how much the previous one was out. That means considering the error. For example, if a mean level of a series changes slowly over time, its first step forecast gives $X_t(1) = X_t$. But if there are many past observations, there is need to assign them weights(ω), giving the most recent observations greater weights(ω) as depicted below;

$$X_t(1) = \frac{1-\omega}{1-\omega^t}(X_t + \omega X_{t-1} + \omega^2 X_{t-2} + \ldots + \omega^{t-1} X_1) \qquad ⍰⍰ ⍰$$

Exponential smoothing is commonly applied to financial market and economic data, but it can be used with any distinct set of recurring measurements. The exponential smoothing methods can be used when the parameters describing the time series are changing slowly with time [54]. Generally, exponential smoothing uses three smoothed statistics that are weighted. These three averages are referred to as single, double, and triple smoothing statistics and are running averages that are weighted in an exponential declining method.

3.5.1.1 Single Exponential Smoothing

The simplest form of exponential smoothing is Single Exponential Smoothing (SES) method. It is mostly effective for short term forecasting purposes such as monthly data [53]. It uses only one weight, alpha (α) during forecasting. The raw observed data is often represented by X_t and the smoothed observation is normally represented by S_t. The subscripts t refer to the time periods, 1,2,…,n. For any time period t, the smoothed value S_t is found by computing;

$$\begin{gathered} S_t = \alpha X_t + (1-\alpha)S_{t-1} \\ F_t = S_t \;\; and \;\; D_t = X_t \end{gathered} \tag{⍰⍰ ⍰}$$

From the above equation, ***D*** $\boldsymbol{D_t}$ is the actual value, $\boldsymbol{F_t}$ is the forecasted value and $_{\mathbf{t}}$ is the current time period. Rearranging the formula and making one-step-ahead forecast gives;

$$\begin{gathered} F_{t+1} = \alpha D_t + (1-\alpha)F_t \\ or \;\; F_{t+1} = F_t + \alpha(D_t - F_t) \end{gathered} \tag{⍰⍰ ⍰}$$

In order for the model to give appropriate answers, α has to be between 0 and 1.

3.5.1.2 Double Exponential Smoothing

The second ES method is Double Exponential Smoothing (DES) method. This method is used to forecast data with trend but without seasonal component. If there is data sequence of observations represented by (x_t), with time beginning at t = 0. Then represent the smoothed value for time t with (s_t) and represent the best estimate of the trend as ($\mathrm{b_t}$). The output of the algorithm which is the estimate of the value x at time $t+m$. will be written as F_{t+m}. In this case m has to be a value greater than 0 and double exponential smoothing will take the form:

$$
\begin{aligned}
s_1 &= x_1 \\
b_1 &= x_1 - x_0
\end{aligned}
$$

For time t>1, it has to take the form;

$$
\begin{aligned}
s_t &= \alpha x_t + (1-\alpha)(s_{t-1} + b_{t-1}) \\
b_t &= \beta(s_t - s_{t-1}) + (1-\beta)b_{t-1}
\end{aligned}
$$

In the above formulas, α is the *data smoothing factor*, $0 < \alpha < 1$, and β is the *trend smoothing factor*, $0 < \beta < 1$. In double exponential smoothing, the *m*-periods-ahead forecast will be represented by;

$$
F_{t+m} = s_t + mb_t
$$

3.5.1.3 Triple Exponential Smoothing

This exponential smoothing method takes into account both trend and seasonal components. It is the best method to address data with repeated behavioral patterns every *L* periods. The seasonality patterns can be divided in to additive and multiplicative. The additive seasonality is the scenario in which the value of a specific period is more than the previous one by a certain amount. With multiplicative seasonality, a constant factor, not an absolute amount is used. For example, a retailer selling 15% more jackets in winter months than in summer months.

Triple exponential smoothing (TES) method was first proposed in 1960 by Holt's student, Peter Winters [55]. In this method the third equation is introduced resulting in a set of equations called “Holt-winters” (HW) method. Suppose there is a sequence of observations (x_t), with a cycle of seasonal change of length *L*. Triple exponential smoothing method calculates a trend line for the data as well as seasonal indices that weight the values in the trend line based on where that time point falls in the cycle of length *L*. For the method to give optimal results a minimum of two full seasons (or 2*L* periods) of historical data is needed to initialize a set of seasonal factors.

The three basic equations of triple exponential smoothing are as follows:

$$\begin{aligned}
s_t &= \alpha \frac{x_t}{c_{t-L}} + (1-\alpha)(s_{t-1} + b_{t-1}) && \mathit{OVERALL\ \ SMOOTHING} \\
b_t &= \beta(s_t - s_{t-1}) + (1-\beta)b_{t-1} && \mathit{TREND\ \ SMOOTHING} \qquad \text{⍰⍰ ⍰} \\
c_t &= \gamma \frac{x_t}{s_t} + (1-\gamma)c_{t-L} && \mathit{SEASONAL\ \ SMOOTHING}
\end{aligned}$$

The final estimate of the value of *x* at time t+*m* is written as F$_{t+m}$.

$$F_{t+m} = (s_t + mb_t)c_{t-L+1+(m-1)\ \bmod L,} \qquad \mathit{FORECAST} \qquad \text{⍰⍰⍰ ⍰}$$

The above equations are explained where;

x_t is the observed value at time t.

S_t is the smoothed series at time *t*

b_t is the trend component at time t

c_t is the seasonal component at time t

F_{t+m} is the forecast at *m* periods ahead

t is an index denoting a time period

and α, β, and γ are constants which must be estimated in order to minimize RMSE and MAE.

In the equations above α is the *data smoothing factor*, 0 < α < 1, β is the *trend smoothing factor*, 0 < β < 1, and γ is the *seasonal change smoothing factor*, 0 < γ < 1. It is crucial to accurately estimate the initial trend and seasonal parameters of the TES method. The following formula is the general one used to estimate the initial trend b_0.

$$b_0 = \frac{1}{L}\left(\frac{x_{L+1} - x_1}{L} + \frac{x_{L+2} - x_2}{L} + \ldots + \frac{x_{L+L} - x_L}{L}\right)$$

The formula below is used for setting up the initial estimates for the seasonal indices. If there are *N* number of complete cycles in the data then:

$$c_i = \frac{1}{N}\sum_{j=1}^{N}\frac{x_{L(j-1)+i}}{A_j} \qquad \forall i = 1,2,\ldots,L$$

where

$$A_j = \frac{\sum_{i=1}^{L} x_{L(j-1)+i}}{L} \qquad \forall j = 1,2,\ldots,N$$

In the formula above A_j is the average value of *x* in the *j*th cycle of data.

3.5.2 Autoregressive Integrated Moving Average

Autoregressive Integrated Moving Average (ARIMA) is a linear time series model that is mostly used to predict network traffic. The ARIMA models are the most general class of models for forecasting a time series which can be made to be "stationary" by differencing if it is not stationary. In this case a time series is stationary if its statistical properties are all constant over time; that is having constant amplitude with no trend. The exponential decay of the autocorrelation function of ARIMA model gives it an ability to capture short-range dependence (SRD) and modeling stationary data traffic. Its shortcoming is that it cannot capture long-range dependent (LRD) characteristics [8].

The ARIMA model is generally referred to as an ARIMA(p,d,q) model where parameters p, d, and q are non-negative integers that refer to the order of the autoregressive, integrated or differencing, and moving average parts of the model respectively. For example, ARIMA(0,1,0) is I(1), ARIMA(1,0,0) is AR(1) and ARIMA(0,0,1) is MA(1). Differencing is performed in ARIMA modelling only if data (Y) is not stationary. This is regarded as the important step of stabilizing the mean of a time series. It removes changes in the level of a time series, and so eliminating trend and seasonality to make data stationary. First order difference (ΔY_t), which is simply the difference between two values is performed as follows,

$$X_t = \Delta Y_t = Y_t - Y_{t-1}$$

or the second order differences

$$X_t = \Delta^2 Y_t = \Delta(\Delta Y)_t = Y_t - 2Y_{t-1} + Y_{t-2}$$

and so on. The second order difference of *Y* is the first-difference-of-the-first difference. The mean of a time series (*y*) can be stabilized and made constant by differencing it *k* times. That is by using operator $(1-L)^k y(t)$, where *L* is the backward shift operator. Differencing it further repeatedly to make it stationary takes form of equation (16). The ARIMA(p, d, q) model is an

ARMA(p, q) model that has been differenced d times. Thus, the ARIMA (p, d, q) can be forecasted by;

$$(1-\sum_{i=1}^{p}\phi_{i}L^{i})(1-L)^{d}y(t)=(1+\sum_{i=1}^{q}\theta_{i}L^{i})\notin(t) \qquad \text{⍰⍰⍰ ⍰}$$

where ϕ_{i} and θ_{i} are the parameters of the model, and $\notin(t)$ are the error terms. The error terms are sampled from a normal distribution with zero mean.

3.6 Data Analysis Methods

The results of the four prediction models will be analyzed using accuracy methods in order to identify the most accurate model based on the given datasets. It is not easy to compare the different models by simply looking at the different values or graphs. The researcher opted to use Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) because they can diagnose error variations and they can also show large but infrequent errors in the forecast [56]. If the difference between RMSE and MAE is larger, it shows that the error size is inconsistent. RMSE gives extra weight to large errors while MAE gives equal weight to all errors. These two forecast accuracy methods are both used to evaluate models by summarizing the differences between the observed and predicted values.

3.6.1 Root Mean Square Error

The Root Mean Square Error (RMSE) is a quadratic scoring rule which measures the average magnitude of the error [57]. The errors are squared before they are averaged and then the square root of the average is taken. The method is a reasonable measure of performance for a forecast (***F***$_t$) and the actual value ($\boldsymbol{A_t}$) and it is mostly used when large errors are experienced. This exercise is performed for all the values in the series and the prediction model with the smallest RMSE is the most accurate one. The number of values in the series is represented by ***n.***

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(A_t - F_t)^2}$$

3.6.2 Mean Absolute Error

Mean Absolute Error (MAE) is the average of absolute differences between forecast value (***f_t***) and actual value (***A_t***). There is no difference between positive and negative error because it is an absolute error. This method is used in scenarios where cost of forecast error is very less and demand is relatively stable. MAE mostly tells how big an error to expect from the forecast based on average. The same applies to MAE, the lower the score the better.

$$MAE = \frac{1}{n}\sum_{t=1}^{n}\left|A_t - F_t\right|$$

3.7 Research Design

This section describes the step by step experimental setup which includes several testing of parameters used for training the four prediction models. The last stage includes validation of the models using accuracy methods. The entire research involves three major phases as depicted in Figure 7. This research design has been applied by other researchers such as Bankole [25] and Zhani et al [42].

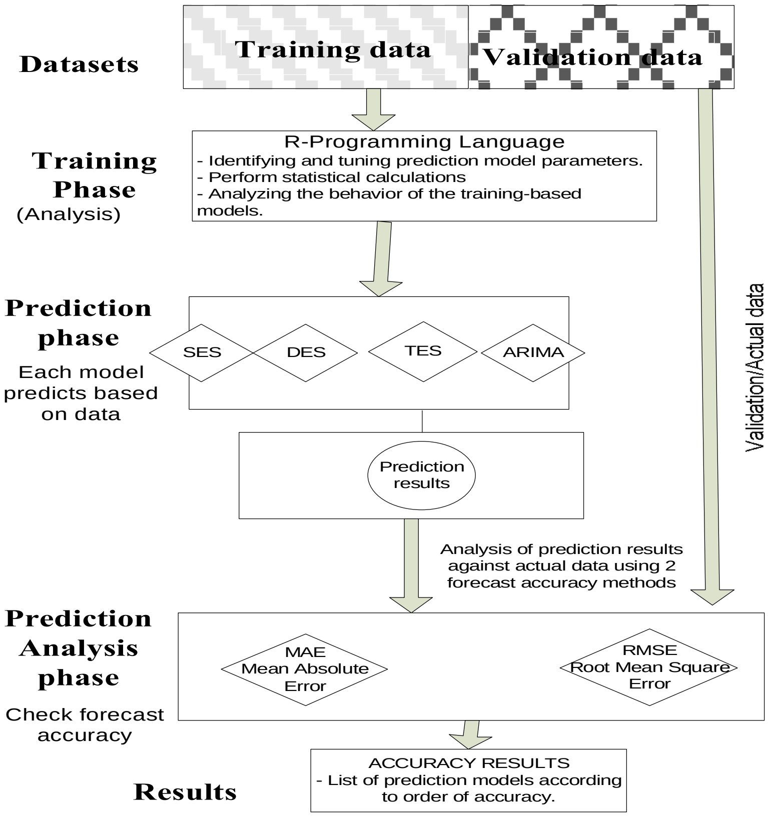
Datasets
Training data
Validation data
Training Phase
(Analysis)
R-Programming Language
- Identifying and tuning prediction model parameters.
- Perform statistical calculations
- Analyzing the behavior of the training-based models.
Prediction phase
Each model predicts based on data
SES
DES
TES
ARIMA
Prediction results
Validation/Actual data
Analysis of prediction results against actual data using 2 forecast accuracy methods
Prediction Analysis phase
Check forecast accuracy
MAE
Mean Absolute Error
RMSE
Root Mean Square Error
Results
ACCURACY RESULTS
- List of prediction models according to order of accuracy.

Figure 7: Prediction Methodology

3.7.1 Training Phase

The training phase is a stage which involves continuous altering of parameters, with the intention of getting the best confidence interval. The parameter values ranges between 0 and 1. Twenty (20) experiments are run per model so as to get accurate and precise results. In this research training phase is used for identifying and tuning model parameters with the aim of finding the best prediction model that can forecast future traffic requirement accurately. The training phase uses 70% of each dataset whilst the rest will be used for validating prediction.

3.7.2 Prediction Phase

This phase involves predicting the future behavior of datasets based on the best fit parameters obtained in the training phase. The monthly and yearly data will be predicted ahead based on the performance of the models. Time series models are mostly used in traffic predictions and simulations because they have the ability to capture prominent traffic characteristics [51].

3.7.3 Prediction Analysis Phase

This is the last phase in which Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) accuracy methods are applied in order to identify the most accurate model based on the given datasets. The accuracy of each model's prediction will be tested against actual data.

3.8 Research Experimentation

The entire research experimentation will be based on four datasets as depicted in Figure 3 and Figure 4. The prediction models have different modeling steps but they will all go through the research phases as described in Figure 7.

3.8.1 Single Exponential Smoothing model

The experimentation starts with SES model as per the following stages using R-language.

3.8.1.1 Training Phase

The training phase of SES involves continuous altering of alpha (α) parameter only, in order to find the best fit. Beta (β) and Gamma (γ) parameters are set to FALSE when using SES. Ten (10) experiments were run using the first 70% of each dataset separately. The experimental results of the monthly downloaded data, yearly downloaded data, monthly number_of_users and yearly number_of_users are as shown in Table 1, Table 2, Table 3 and Table 4 respectively.

The graphical representations of all the four (4) datasets were also executed based on the best alpha (α) parameter and the results are coloured green in Figure 8 and Figure 9.

3.8.1.2 Prediction Phase

The yearly data was predicted two (2) years ahead, that is for year 2015 and year 2016. The monthly data was predicted four (4) months ahead from December 2014. The prediction results are shown in Figure 8 and Figure 9.

3.8.1.3 Prediction Analysis Phase

The predicted data was tested against actual data using RMSE and MAE accuracy methods. The monthly downloaded data gave the smallest RMSE of 862.9 and MAE of 419.853 whereas the yearly data gave RMSE of 5337.003 and MAE of 3969.268. As for the number of users dataset, the monthly data has RMSE of 4.918458 and MAE of 3.788868 while the yearly data graph attained RMSE of 61.44153 and MAE of 49.35875.

3.8.2 Double Exponential Smoothing model

Double Exponential Smoothing (DES) model uses alpha (α) and beta (β) parameters for training.

3.8.2.1 Training Phase

The training phase of DES involves continuous altering of alpha (α) and beta (β) parameters. The best combination of the two parameters is the one with the lowest RMSE and MAE. In order to achieve that, twenty (20) experiments were performed for each dataset using the first 70% of each dataset. The trained results of all the four (4) datasets are as depicted in Table 5, Table 6, Table 7 and Table 8.

The experimental results of the datasets were further represented in their graphical view as shown in Figure 12 and Figure 13. The results are based on the best combination of alpha (α) and beta (β) parameters.

3.8.2.2 Prediction Phase

The monthly data was predicted four (4) months ahead from December 2014. The yearly data was predicted two (2) years ahead, that is for year 2015 and year 2016. The prediction results are shown in Figure 14 and Figure 15.

3.8.2.3 Prediction Analysis Phase

The accuracy of DES predictions was tested against actual data. The monthly downloaded data gave the smallest RMSE of 791.6628 and MAE of 387.4016 while the yearly data has the best fit of RMSE of 2346.092 and MAE of 1977.66. As for the number of users dataset, the monthly data has RMSE of 4.773055 and MAE of 3.77382 while the yearly data graph has RMSE of 38.64853 and MAE of 31.70763.

3.8.3 Triple Exponential Smoothing model

Triple Exponential Smoothing (TES) model calculates dynamic estimates for level, trend and seasonality and it is sometimes referred to as Holt-Winters' method [55].

3.8.3.1 Training Phase

This method uses all the three parameters being; alpha (α), beta (β) and gamma (γ). Twenty (20) experiments were performed for each dataset using R-language. The experimental results of the trained values are shown in Table 9, Table 10, Table 11 and Table 12. The best trained results were used to execute the graphical representation shown in Figure 16 and Figure 17.

3.8.3.2 Prediction Phase

The monthly data was predicted twelve (12) months ahead. As for the yearly data it was predicted two years ahead, that is for year 2015 and year 2016. The prediction results are as shown in Figure 18 and Figure 19.

3.8.3.3 Prediction Analysis Phase

The monthly downloaded data got the smallest RMSE of 843.3648 and MAE of 441.7421 while the yearly data got the best fit at RMSE of 2374.754 and MAE of 1950.505. As for the number of users dataset, the monthly data got RMSE of 4.829408 and MAE of 3.779075 and the yearly data got RMSE of 51.20327 and MAE of 35.3834.

3.8.4 Autoregressive Integrated Moving Average

The ARIMA procedure analyzes and forecasts equally spaced univariate time series data. It predicts a value in a response time series as a linear combination of its own past values, past errors and current and past values of other time series. Generally ARIMA modeling is divided into three stages; Identification, Parameter Estimation and Prediction. The identification and the estimation stages fall under the training phase while the forecasting is the same as the prediction phase. The last phase will be the analysis phase.

3.8.4.1 Identification (Stationary testing)

The first step in this model is to test if data is stationary by using Autocorrelation function (ACF) and Partial Autocorrelation function (PACF). Since there are four different datasets, experiments will be carried out in each separately. If the mean or the variance of the time series changes over time, it indicates non-stationarity. In simple terms a stationary process is the one whose statistical properties do not change over time. Dickey Fuller test is one of the methods used by researchers to test for stationarity. A stationarized series simply predicts that its statistical properties will be the same in the future as they have been in the past. Stationarizing a time series through differencing is an important part of fitting an ARIMA model [52].

The ARIMA experiments start with the running of ACF and PACF in all the four datasets. The ACF of the datasets does not trail off quickly which is an indication of non-stationarity. The detailed discussions are analysed in chapter 4.5.1. Based on that, differencing was performed as depicted in Figure 22 and Figure 23. The differenced data shows large oscillation spikes which signify non-constant variance. Thereafter the ACF and PACF of the differenced data were run as depicted in Figure 24.

3.8.4.2 Parameter Estimation

Estimation is a critical component of time series analysis. Significance tests for parameter estimates indicate whether some terms in the model may be unnecessary. The experiments will be run several times with the aim of testing for the goodness-of-fit for the model which helps in comparing the model to others. To test for the goodness-of-fit we will use Akaike Information Criteria (AIC) which is a widely used measure of statistical models. It will be used together with RMSE and MAE in the analysis of ARIMA models' results. When comparing two models, the one with the lower AIC is generally "better". The AIC is defined as

$$AIC = \log\left[V\left(1+\frac{2n}{N}\right)\right]$$

where *V* denotes the number of parameters and *N* denotes the maximized value of the likelihood function.

The experimentation starts by comparing 25 ARIMA models against the first difference of the four (4) datasets as per the identification section above. That comprises of ARIMA(1,1,1) through to ARIMA (5,1,5). The results of the experiments performed in each model are as depicted in Table 13, Table 14, Table 15 and Table 16.

3.8.4.3 Prediction

The prediction results for both downloaded_data and number_of_users datasets are shown in Figure 26 and Figure 27 respectively. In all the datasets, the monthly data has been predicted 12 months ahead while the yearly data has been predicted two (2) years ahead.

3.8.4.4 Prediction Analysis Phase

The RMSE and MAE of the best ARIMA models were calculated using the accuracy function which comes with the forest package. The function was performed for each best model based on the actual data and recorded in Table 17.

3.9 Ethical Considerations

It is imperative that ethical issues are considered during the project so as to abide by the rules and standards of data or any entity which may be affected. According to Fox et al, [58] integrity of data and anonymity of participants are some of the vital ethical consideration in gaining reliable information. For this research the following ethics were considered;

3.9.1 Safe guarding CAIDA's privacy of information

The CAIDA's datasets are solely used for academic purposes but not for any commercial or profitable purpose.

3.9.2 Honesty

All the arithmetic calculations in this project were carried out using R-language. Data that was taken from CAIDA has been plotted the way it was. The results obtained from the experiments have not been in anyway changed or misrepresented.

3.9.3 Ensuring integrity and correctness of data

There is a possibility of one person downloading the same data many times in the same month, which may affect the dataset count. In that case CAIDA researchers consider that as one download but not many [47]. That process increases the quality and accuracy of data.

CHAPTER 4 - RESULTS OF DATA ANALYSIS

4.1 Introduction

This chapter presents the results of the experiments carried out for determining the prediction accuracy of SES, DES, TES and ARIMA models. The results include training results as well as prediction results for each model. The findings will be discussed in depth and evaluated depending on the performance of the models.

4.2 Single Exponential Smoothing model

The results of SES model are discussed in details in this section. It also includes the graphical representation of the best parameters.

4.2.1 Training results for Downloaded_data using SES

The monthly data has the best fit at α=0.145 with RMSE of 862.9 and MAE of 419.853. The yearly downloaded data gives the best fit at α=0.99 with RMSE of 5337.003 and MAE of 3969.268.

Table 1: Training SES parameter for monthly downloaded_data

Experiment	Alpha (α)	RMSE	MAE
1.	0.1	870.5706	412.7253
2.	0.99	1106.148	552.6612
3.	0.31	890.2204	447.5238
4.	0.23	873.6059	437.3916
5.	0.19	866.17	429.6406
6.	0.15	862.9138	421.0547
7.	0.14	863.0086	418.6345
8.	0.145	862.9	419.853
9.	0.13	863.6534	417.1592
10.	0.12	864.9921	415.6032

Table 2: Training SES parameter for yearly downloaded_data

Experiment	Alpha (α)	RMSE	MAE
1.	0.1	10559.6	7453.258
2.	0.22	9477.631	6748.637
3.	0.54	7309.509	5310.983
4.	0.99	5337.003	3969.268
5.	0.68	6599.608	4832.165
6.	0.33	8629.301	6190.379
7.	0.41	8084.9	5829.287
8.	0.55	7254.823	5274.247
9.	0.67	6646.583	4863.981
10.	0.15	10086.91	7146.407

The graphical representation of downloaded_data results in Figure 8 has satisfactory results when α=0.145 and α=0.99 for monthly and yearly data respectively. It shows that the trained results do not accurately trail on the actual data.

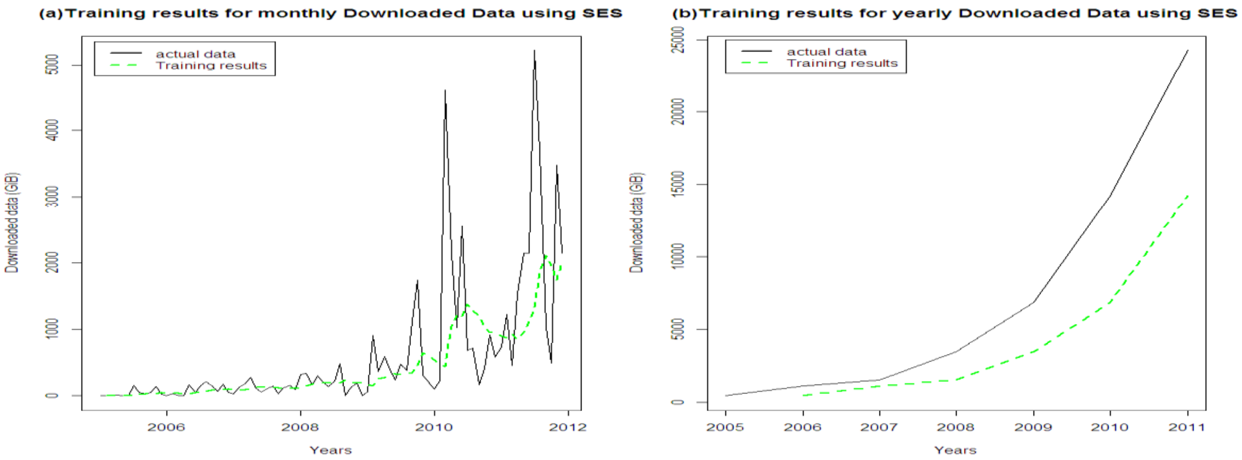

Figure 8: Training results for downloaded_data using SES

4.2.2 Training results for Number_of_Users using SES

The monthly download data has the best fit at α=0.29, with RMSE of 4.918458 and MAE of 3.788868. The yearly data gives the best fit at α=0.99, with RMSE of 61.44153 and MAE of 49.35875.

Table 3: Training SES parameter for monthly Number_of_Users data

Experiment	Alpha (α)	RMSE	MAE
1.	0.23	4.993784	3.809723
2.	0.10	5.855978	4.427224
3.	0.99	6.028541	4.930997
4.	0.29	4.918458	3.788868
5.	0.21	5.040403	3.825266
6.	0.40	4.920245	3.833444
7.	0.62	5.176776	4.195607
8.	0.55	5.07041	4.06516
9.	0.77	5.46439	4.483419
10.	1.0	6.058656	4.95122

Table 4: Training SES parameter for yearly Number_of_Users data

Experiment	Alpha (α)	RMSE	MAE
1.	0.55	82.61206	73.78108
2.	0.21	128.6064	115.6912
3.	0.1	153.5519	136.1948
4.	0.99	61.44153	49.35875
5.	0.69	72.60026	62.91004
6.	0.35	104.9785	95.13448
7.	0.22	175.5549	153.6002
8.	0.55	82.61206	73.78108
9.	0.4	98.30208	89.02261
10.	0.9	63.60997	51.44278

The graphical view in Figure 9 shows the trained results of the monthly number_of_users and yearly number_of_users datasets. The results are based on the best obtained alpha (α) parameter.

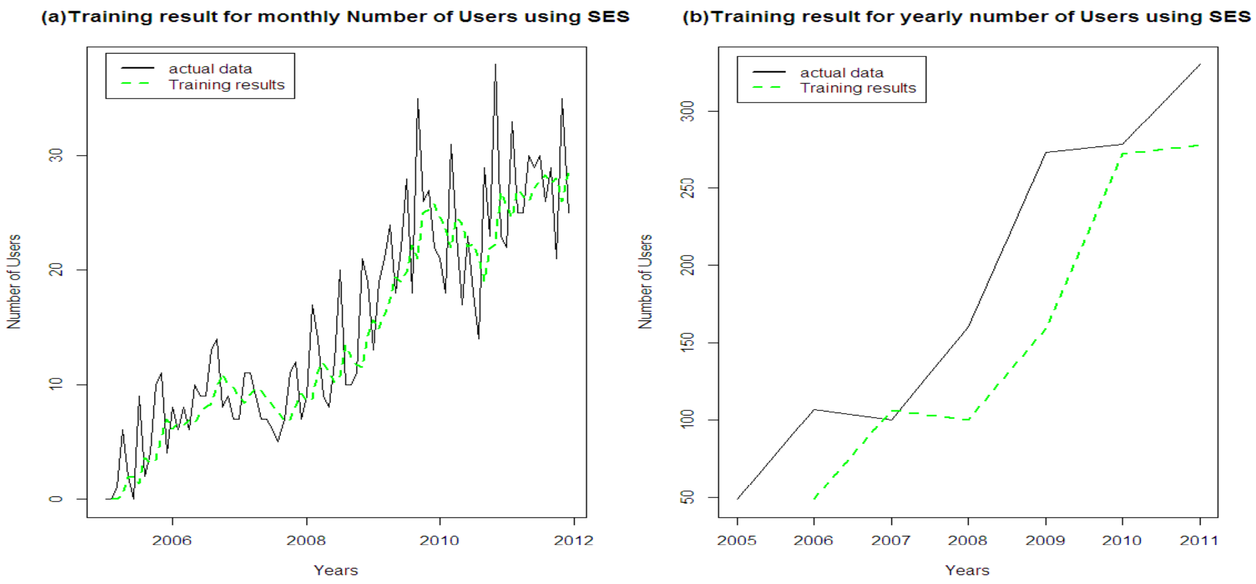

Figure 9: Training results for Number_of_Users using SES

4.2.3 Prediction results for Downloaded_data using SES

In Figure 10, the monthly and yearly prediction results display a constant straight line. The results show that the SES predictions will be the same for the coming months and years. The predictions are shown with a dotted blue line. The upper and lower prediction bounds are shown with a dotted red line.

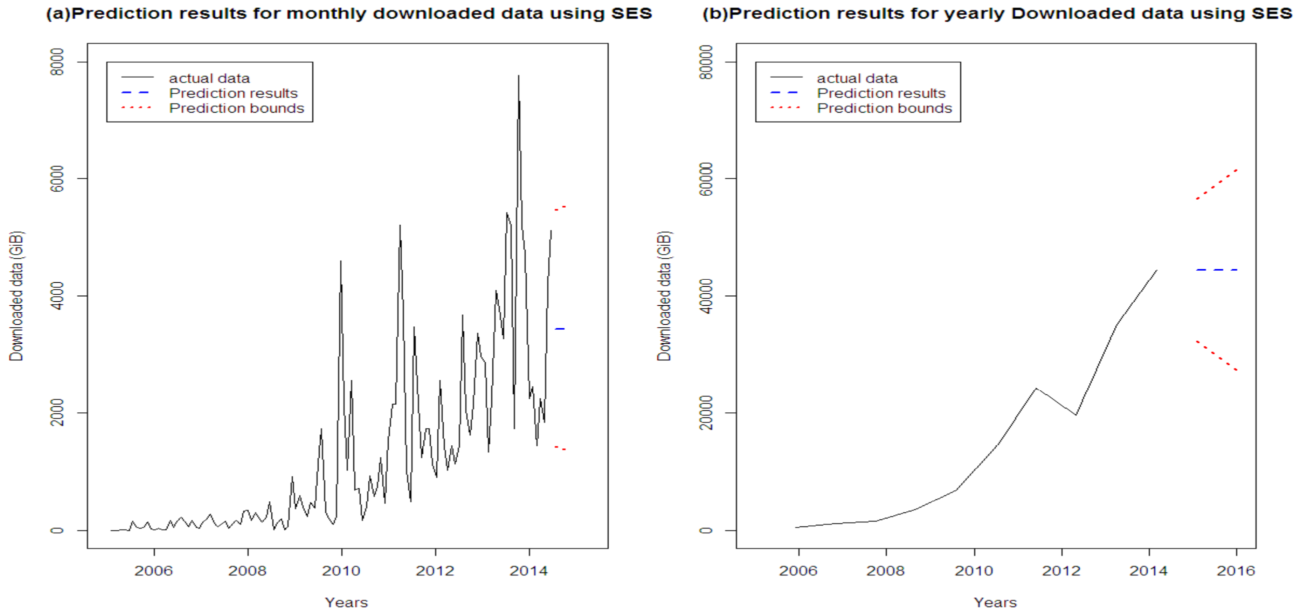

Figure 10: Prediction results for downloaded_data using SES

According to the predictions, the amount of downloaded data for 2015 and 2016 will be 44440.99GiB each. The monthly downloaded data will be 3449.772 GiB from January 2015 until April 2015. The forecasts signify that the amount of downloaded data in the cloud is likely to stay constant in the future.

4.2.4 Prediction results for Number_of_users using SES

From Figure 11, it shows that SES predictions for number_of_users data will be the same in the future. The predictions are shown with a dotted blue line while the upper and lower prediction bounds are shown with a dotted red line.

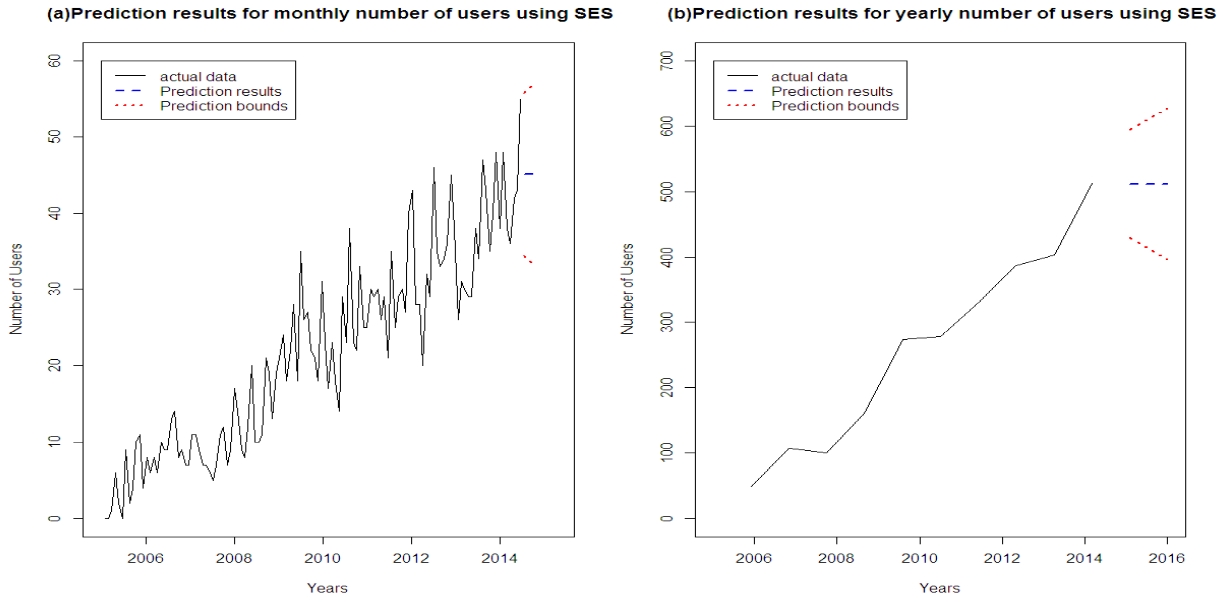

Figure 11: Prediction results for number_of_users using SES

According to the predictions, 511 cloud users will access CAIDA passive datasets each year for 2015 and 2016. The monthly data will experience 45 users monthly from January 2015 until April 2015.

4.3 Double Exponential Smoothing model

In DES, α and β parameters were used for testing the level and the trend. The results for the experiments are as follows.

4.3.1 Training results for Downloaded_data using DES

The rmse and mae results appear to be huge because they deal with large numbers. The monthly data has the best fit at α=0.0106 and β=1.0 with RMSE of 791.6628 and MAE of 387.4016. The yearly downloaded data gives the best fit at α=1.0 and β=1.0 with RMSE of 2346.092 and MAE of 1977.66. The α and β values of yearly data are on their maximum, the situation which is rare.

Table 5: Training DES parameters for monthly downloaded_data

Experiment	Alpha (α)	Beta (β)	RMSE	MAE
1.	0.41	0.79	1044.816	521.5553
2.	0.21	0.79	1027.483	520.3721
3.	1.0	1.0	1467.061	762.6
4.	1.0	0.1	971.6544	489.2971
5.	0.10	0.9	1015.1	539.1016
6.	0.10	0.79	986.8479	522.7369
7.	0.49	0.79	1078.044	526.5066
8.	0.495	0.69	1047.164	513.6936
9.	0.021	0.98	820.8346	428.4539
10.	0.0106	1.0	791.6628	387.4016
11.	0.0106	0.79	993.8737	526.5906
12.	0.0231	1.0	832.0052	437.7124
13.	0.231	0.579	984.9986	498.2097
14.	0.90	0.10	944.3818	476.5591
15.	0.60	0.30	950.4398	474.4698
16.	0.60	0.13	892.8906	443.5886
17.	0.79	0.013	884.4949	442.5584
18.	0.018	0.013	813.11	401.605
19.	0.218	0.013	813.6425	403.9648
20.	0.8721	0.0113	900.6684	452.0223

Table 6: Training DES parameters for yearly downloaded_data

Experiment	Alpha (α)	Beta (β)	RMSE	MAE
1.	0.99	0.0009	5348.734	4079.503
2.	0.21	0.0015	8815.266	6249.073
3.	0.23	0.867	7248.747	5320.199
4.	0.83	0.867	3106.778	2585.037
5.	1.0	0.867	2573.123	2166.052
6.	1.0	0.2	4460.335	3505.089
7.	1.0	0.992	2358.549	1988.416
8.	1.0	1.0	2346.092	1977.66
9.	0.1	0.1992	9325.462	6556.499
10.	0.231	0.1992	8343.41	5970.964
11.	0.231	0.992	7043.608	5196.215
12.	0.231	0.4992	7832.788	5669.373
13.	0.761	0.4992	4352.953	3452.217
14.	0.761	0.8992	3312.292	2737.778
15.	0.81	0.8992	3113.439	2591.311
16.	0.81	0.992	2924.03	2452.289
17.	0.891	0.992	2650.311	2238.301
18.	0.9591	0.92	2583.925	2179.32

Experiment	Alpha (α)	Beta (β)	RMSE	MAE
19.	0.991	0.92	2500.011	2108.069
20.	0.991	0.998	2370.031	1999.688

The trained results of the downloaded data using DES are coloured green in Figure 12. The results are based on the best combination of alpha (α) and beta (β) parameters. Due to the randomness of data, the estimated values shown with green lines do not accurately trail on the actual data.

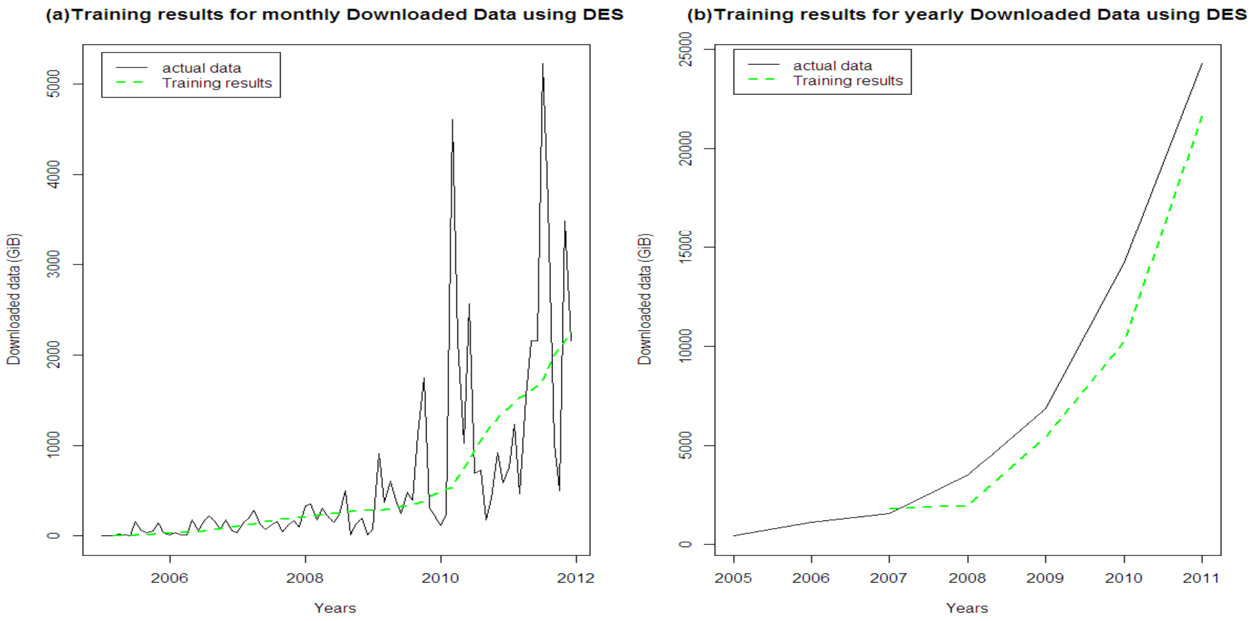

Figure 12: Training results for downloaded_data using DES

4.3.2 Training results for Number_of_Users using DES

The monthly data graph gives the best fit at α=0.232 and β=0.025, with RMSE of 4.773055 and MAE of 3.77382. The yearly data graph has the best fit at α=0.153 and β=1.0, with RMSE of 38.64853 and MAE of 31.70763.

Table 7: Training DES parameters for monthly Number_of_Users data

Experiment	Alpha (α)	Beta (β)	RMSE	MAE
1.	0.131	0.0054	5.101697	3.809575
2.	0.213	0.0054	4.843142	3.742216
3.	0.205	0.867	5.249963	4.149131
4.	0.21	0.34	5.042243	3.945443
5.	0.21	0.43	5.107981	4.033008
6.	0.21	0.64	5.199363	4.121461
7.	0.121	0.94	5.581916	4.409453
8.	0.321	0.54	5.267426	4.173994
9.	0.21	0.14	4.89553	3.866857
10.	0.21	0.867	5.253291	4.156142
11.	0.21	0.0467	4.789713	3.810788
12.	0.232	0.025	4.773055	3.77382
13.	0.20	0.14	4.902745	3.868697
14.	0.03	0.014	7.941142	6.460044
15.	0.23	0.012	4.791619	3.753932
16.	0.23	0.015	4.783148	3.758355
17.	0.22	0.014	4.78919	3.756064
18.	0.24	0.014	4.783739	3.757773
19.	0.22	0.19	4.928924	3.877358
20.	0.2431	0.867	5.316863	4.240905

Table 8: Training DES parameters for yearly Number_of_Users data

Experiment	Alpha (α)	Beta (β)	RMSE	MAE
1.	0.131	0.54	41.1258	37.24802
2.	0.2131	0.54	38.85569	33.63999
3.	0.02	0.9996	51.86988	46.53669
4.	0.19	0.855	38.75615	31.39935
5.	0.20	0.6	38.81404	33.40011
6.	0.21	0.867	39.05931	30.83195
7.	0.153	1.0	38.64853	31.70763
8.	0.21	0.64	38.73155	32.69675

Experiment	Alpha (α)	Beta (β)	RMSE	MAE
9.	0.215	0.71	38.78047	31.92536
10.	0.22	0.867	39.25061	30.91067
11.	0.22	0.91	39.4196	30.99275
12.	0.121	0.94	39.30901	34.14179
13.	0.321	0.54	39.68131	32.18003
14.	0.205	0.17	41.91738	38.89009
15.	0.2131	0.84	39.03235	30.94223
16.	0.2131	0.994	39.60871	31.09226
17.	0.2131	0.74	38.80583	31.71505
18.	0.2531	0.867	40.0388	31.06855
19.	0.2031	0.867	38.94458	30.96034
20.	0.655	0.0012	41.50844	37.66174

The graphical representation of the trained results of the number_of_users dataset is as depicted in Figure 13. The results are based on the best combination of alpha (α) and beta (β) parameters.

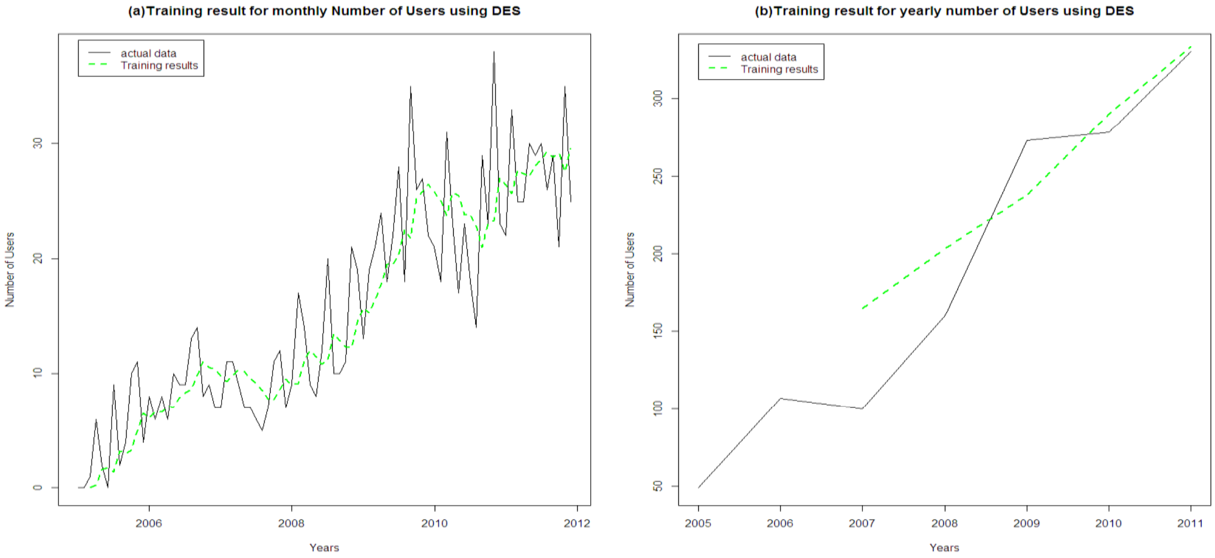

Figure 13: Training results for Number_of_Users using DES

4.3.3 Prediction results for Downloaded_data using DES

The monthly prediction results for the next four months are coloured blue in Figure 14a. The future monthly downloaded data behavior shows an initial decline in January 2015 to 3936.408 GiB. Thereafter an increase from February 2015 reaching 4281.793 GIB in April 2015.

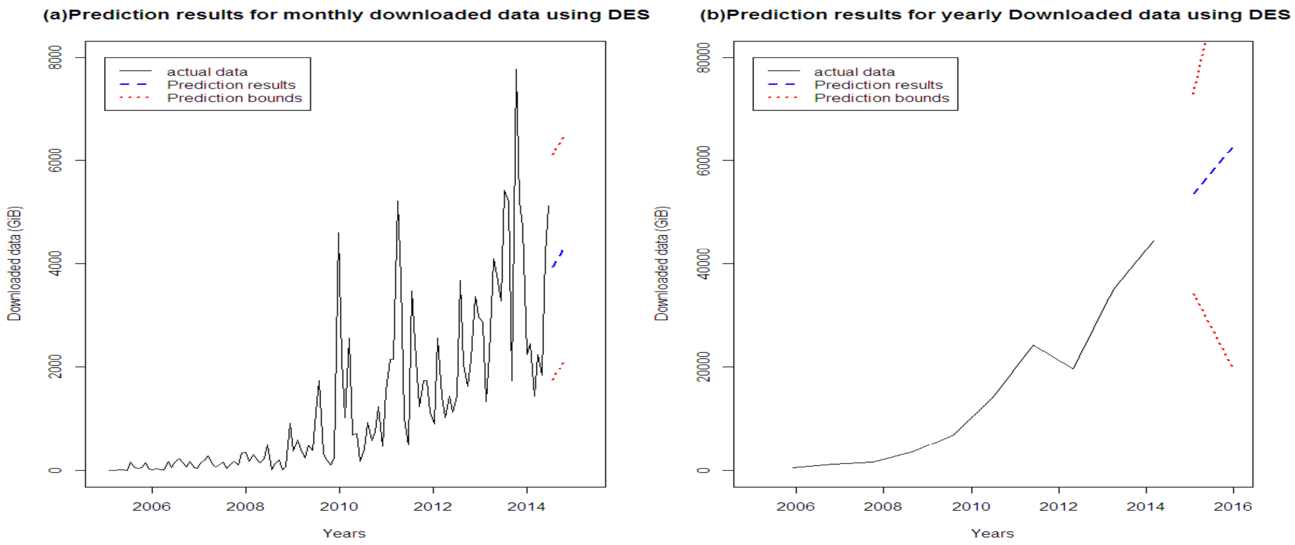

Figure 14: Prediction results for downloaded_data using DES

The prediction results for the years 2015 and 2016 are shown with a blue dotted line in Figure 14b. The amount of downloaded data results shows a growth to 53657.6 GiB and 62873.6 GiB for years 2015 and 2016 respectively. The forecasts signify that the amount of downloaded data in the cloud is likely to increase in the future. For both figures the upper and lower prediction bounds are coloured red.

4.3.4 Prediction results for Number_of_Users using DES

The monthly data changes quite often affecting the prediction accuracy. The predicted results for the next four months are coloured blue in Figure 15a. They show a decline to 45 users in January 2015 and an increase to 46 users in February 2016. The rate at which users log and access cloud services differs every month.

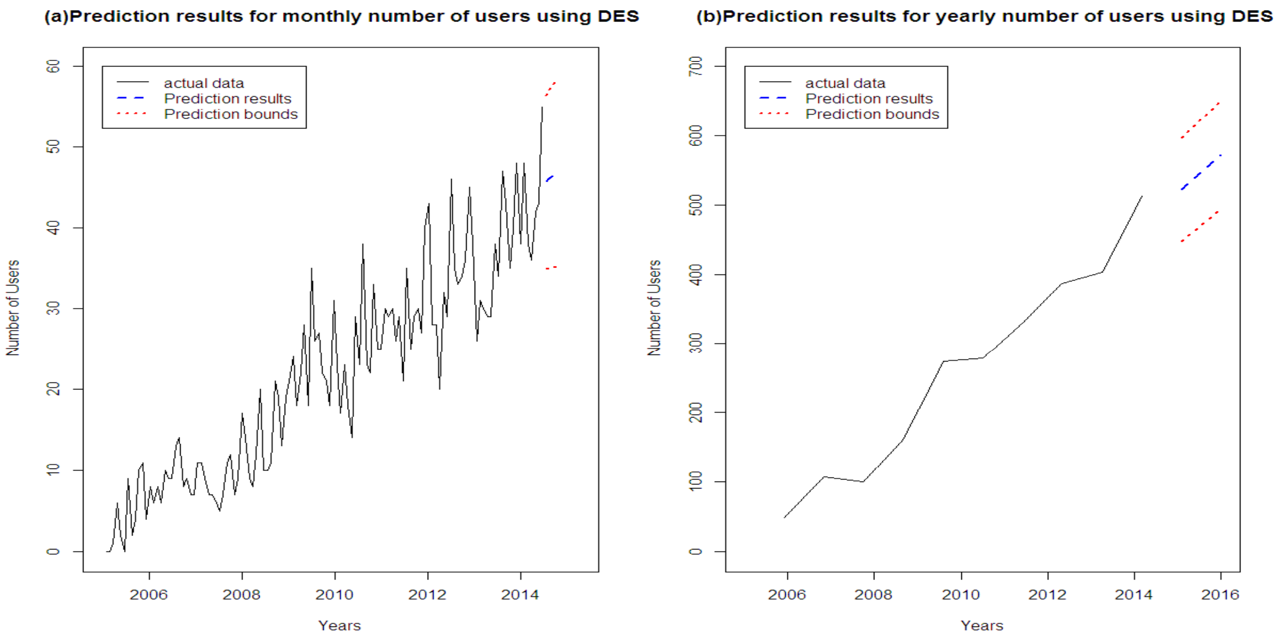

Figure 15: Prediction results for number of users using DES

The prediction results for the next two years are shown with a dotted blue line in Figure 15b. They indicate a slight growth in 2015 to reach 521 cloud users. In 2016 a sharp growth is forecasted at 571 users. The graph also shows the upper and the lower prediction bounds in red.

4.4 Triple Exponential Smoothing model

In Triple Exponential Smoothing method, α, β and γ parameters are all used to find the best fit. They test for the level, trend and seasonality respectively.

4.4.1 Training results for Downloaded_data using TES

The monthly data has the smallest RMSE of 843.3648 and MAE of 441.7421 at α=0.011, β=1.0 and γ=0. The yearly downloaded data gives the best fit at α=1.0, β=1.0 and γ=1.0 and has the smallest RMSE and MAE of 2374.754 and 1950.505 respectively. All the yearly parameters are on their maximum.

Table 9: Training TES parameters for monthly downloaded_data

Experiment	Alpha (α)	Beta (β)	Gamma(γ)	RMSE	MAE
1.	0.03	0.5	0.12	876.1042	474.3635
2.	0.09	0.76	0.43	1125.535	612.3655
3.	0.21	0.15	0.76	1088.242	577.008
4.	0.3	0.65	0.23	1144.109	607.2984
5.	0.011	1.0	0	843.3648	441.7421
6.	0.043	0.45	0.09	888.6496	491.4043
7.	0.13	0.97	0.11	1161.986	659.8682
8.	0.22	0.2	0.42	1015.944	554.2067
9.	0.173	0.015	0.22	903.9053	474.8296
10.	0.011	0.98	0.19	874.7364	456.8169
11.	0.70	0.8	0.20	1255.554	649.3204
12.	0.51	0.015	0.14	914.5303	468.5447
13.	0.002	0.7	0.23	987.4381	473.0615
14.	0.05	0.44	0.17	919.1855	508.1579
15.	0.014	0.1	0.013	956.0855	450.1112
16.	0.03	0.01	0.90	1091.058	525.2545
17.	0.12	0.34	0.61	1090.906	564.8692
18.	0.73	0.02	0.72	1013.132	526.8944
19.	0.08	0.05	0.34	925.5737	465.1907
20.	0.1	0	0.01	880.9783	450.0824

Table 10: Training TES parameters for yearly downloaded_data

Experiment	Alpha (α)	Beta (β)	Gamma(γ)	RMSE	MAE
1.	0.63	0.5	0.23	4747.061	3660.402
2.	0.79	0.76	0.13	3444.62	2735.008
3.	0.99	0.89	0.66	2572.904	2081.453
4.	0.41	0.65	0.43	5470.592	4170.419
5.	0.73	0.45	0.09	4494.365	3479.082
6.	0.23	0.97	0.11	6637.594	4965.508
7.	0.22	0.2	0.42	7730.733	5690.615
8.	1.0	1.0	1.0	2374.754	1950.505
9.	0.73	0.615	0.23	4003.802	3137.05
10.	0.61	0.98	0.19	3730.091	2946.604
11.	0.70	0.8	0.14	3708.763	2927.865
12.	0.51	0.015	0.18	6935.866	5141.184
13.	0.92	0.7	0.24	3140.944	2510.129
14.	0.5	0.44	0.87	5245.872	4013.076
15.	0.54	0.1	0.93	6054.965	4559.451
16.	0.43	0.11	0.87	6458.25	4836.061
17.	0.12	0.24	0.91	7657.49	5653.042
18.	0.73	0.07	0.72	5649.594	4273.607

Experiment	Alpha (α)	Beta (β)	Gamma(γ)	RMSE	MAE
19.	0.08	0.81	0.44	8213.148	6018.895
20.	0.1	0	0.01	9502.638	6838.481

The graphical view of the trained results for the monthly and yearly downloaded_data dataset is presented in Figure 16. The yearly trained data trails slightly closer to the actual data in years 2007 and 2008.

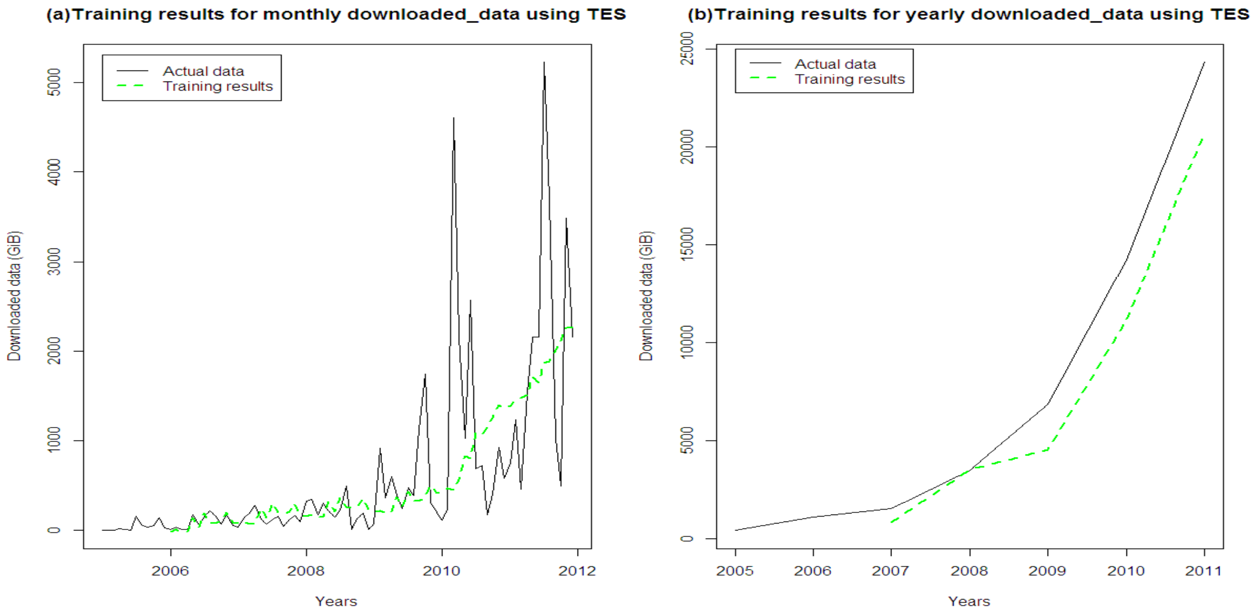

Figure 16: Training results for Downloaded_data using TES

4.4.2 Training results for Number_of_Users using TES

The monthly data obtained the best fit at α=0.239, β=0 and γ=0.303 with RMSE of 4.829408 and MAE of 3.779075. The yearly data graph has the best fit at α=0.529, β=0.217 and γ=1.0. Its RMSE and MAE results are 51.20327 and 35.3834 respectively.

Table 11: Training TES parameters for monthly Number_of_Users data

Experiment	Alpha (α)	Beta (β)	Gamma (γ)	RMSE	MAE
1.	0.23	0.015	0.33	4.855702	3.77591
2.	0.29	0.76	0.33	5.642038	4.670212
3.	0.30	0.15	0.67	5.320009	4.151531
4.	0.13	0.35	0.83	6.994933	5.33947
5.	0.03	0.45	0.99	7.266329	5.805905
6.	0.23	0	0.31	4.830191	3.776374
7.	0.22	0	0.32	4.833159	3.773857
8.	0.193	0.015	0.34	4.874445	3.772721
9.	0.21	0.1	0.39	5.050894	3.901727
10.	0.20	0.2	0.40	5.235487	4.075403
11.	0.21	0.015	0.30	4.857086	3.780515
12.	0.239	0	0.303	4.829408	3.779075
13.	0.22	0	0.33	4.834947	3.771744
14.	0.25	0	0.67	5.090623	3.933112
15.	0.24	0.01	0.83	5.353555	4.097007
16.	0.03	0.01	0.99	6.274008	5.055275
17.	0.22	0	0.31	4.83192	3.777415
18.	0.18	0.05	0.34	4.960309	3.812049
19.	0.1	0	0.01	5.226879	4.049058
20.	0.87	0.89	0.9	8.266038	6.621713

Table 12: Training TES parameters for yearly Number_of_Users data

Experiment	Alpha (α)	Beta (β)	Gamma (γ)	RMSE	MAE
1.	0.43	0.25	0.43	54.0278	33.30806
2.	0.29	0.76	0.33	53.13702	32.87693
3.	0.3	0.15	0.67	56.10707	39.29061
4.	0.13	0.35	0.83	58.58459	42.20405
5.	0.03	0.45	0.99	61.88195	46.30685
6.	0.23	0.6	0.31	57.09664	40.02572
7.	0.22	0.2	0.32	64.72164	51.7497
8.	0.193	0.15	0.64	60.51431	46.54531
9.	0.529	0.217	1.0	51.20327	35.3834
10.	0.21	0.11	0.39	65.69555	53.21947
11.	0.20	0.32	0.40	61.80879	48.12958
12.	0.21	0.33	0.70	56.52163	39.26642
13.	0.22	0.4	0.33	60.45256	45.89396
14.	0.25	0.1	0.67	58.56586	43.66623
15.	0.24	0.01	0.83	58.24572	43.00909
16.	0.03	0.01	0.99	62.89297	47.78153
17.	0.19	0	0.31	71.91391	60.07061
18.	0.23	0.67	0.32	56.11586	38.00093

Experiment	Alpha (α)	Beta (β)	Gamma (γ)	RMSE	MAE
19.	0.45	0.5	0.37	52.6191	35.34596
20.	0.89	0	0.01	62.90748	49.01425

The trained results of the number_of_users datasets using TES are coloured green in Figure 17. The results are based on the best combination of alpha (α), beta (β) and gamma (γ) parameters.

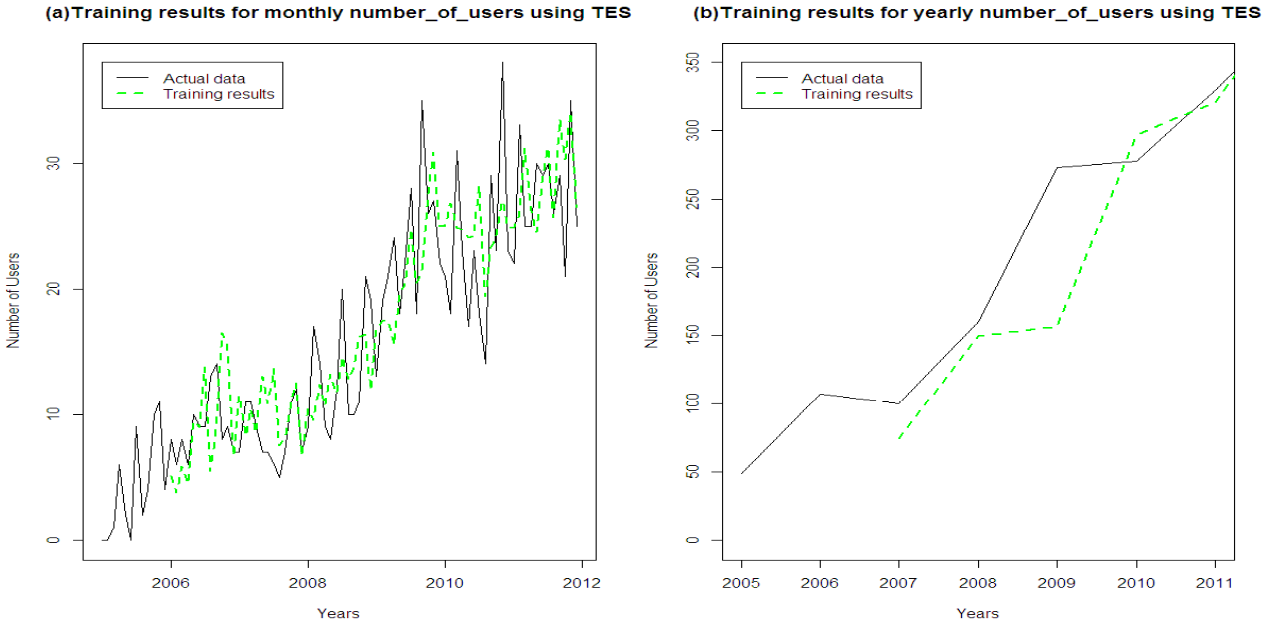

Figure 17: Training results for Number_of_Users using TES

4.4.3 Prediction results for Downloaded_data using TES

The monthly predictions shows some minor changes but mimicking the past data. The yearly prediction results shows an upwards growth. Figure 18 shows that the amount of data download will increase in 2015 and 2016. The predictions are shown with a dotted blue line while the upper and lower prediction bounds are shown with a dotted red line.

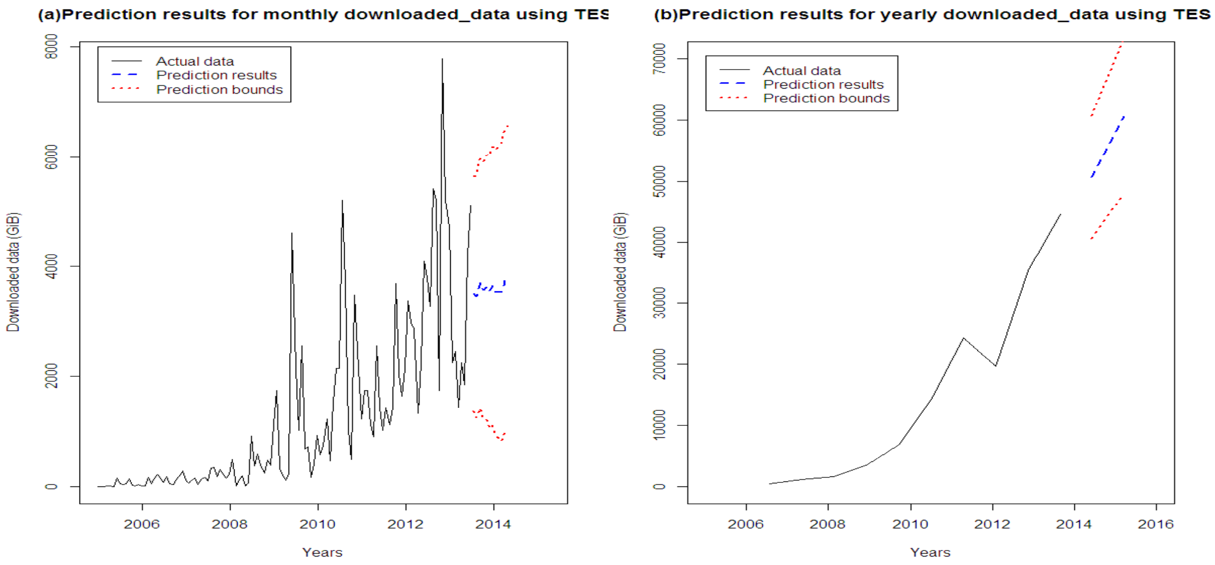

Figure 18: Prediction results for downloaded_data using TES

From TES predictions, the amount of downloaded data for 2015 and 2016 will be 50624.22 GiB and 60534.24 GiB respectively. The monthly downloaded data will be 3503.229 GiB in January 2015 reaching 3727.991 GiB in December 2015. The forecasts signify that the amount of downloaded data in the cloud is likely to experience an increase in the future.

4.4.4 Prediction results for Number_of_Users using TES

Both monthly and yearly prediction results in Figure 19 show an upwards growth in number of cloud users in the future. The predictions are shown with a dotted blue line. The upper and lower prediction bounds are shown with a dotted red line.

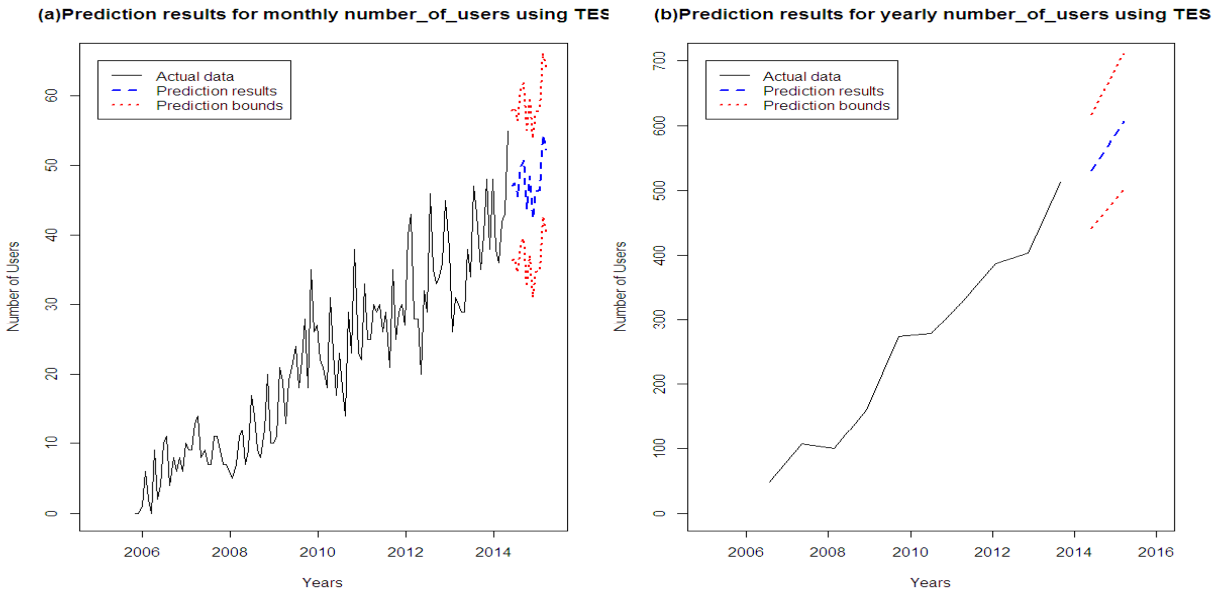

Figure 19: Prediction results for Number_of_users using TES

According to the TES predictions, 530 cloud users will access CAIDA passive datasets by year 2015. Thereafter the services will experience an increase to 607 users by 2016. The monthly data will experience 47 users in January 2015 and an increase to 52 users by December 2015. That shows that CAIDA cloud services will experience more cloud users in the future.

4.5 Auto regression Integrated Moving Average model

The results for downloaded_data and number_of_users datasets for both monthly and yearly period are presented and discussed as follows.

4.5.1 ACF and PACF results

Examining Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) is the first step in analyzing time series. Observing the plots can best tell whether the series is stationary or non-stationary. ACF and PACF help in modeling a series as a function of its past values and past random errors. The autocorrelation plot shows how values of the series are correlated with past values of the series. Their analysis assists in identifying the order of “pure” AR and MA processes. PACF plays an important role in data analysis of identifying the extent of the lag in an autoregressive model. It shows the partial correlation coefficients between the series and its lags. In general, the "partial" correlation between two variables is the amount of correlation between them which is not explained by their mutual correlations with a specified set of other variables.

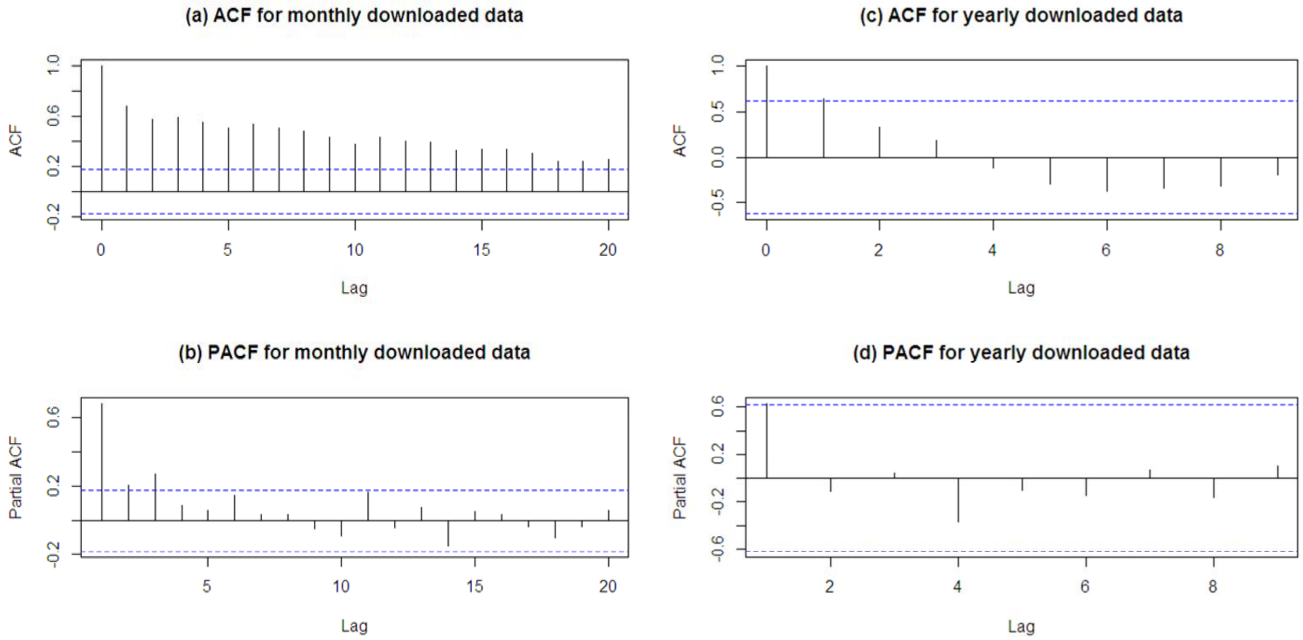

Figure 20: ACF and PACF results for downloaded data

The autocorrelation plots shows how values of the series are correlated with past values of the same series. In Figure 20(a), the ACF does not tail off quickly hence it is not stationary. Based on that, there is a need to do differencing to make it become stationary. The PACF in Figure 20(b), has a strong first lag, with the second and third lags becoming smaller.

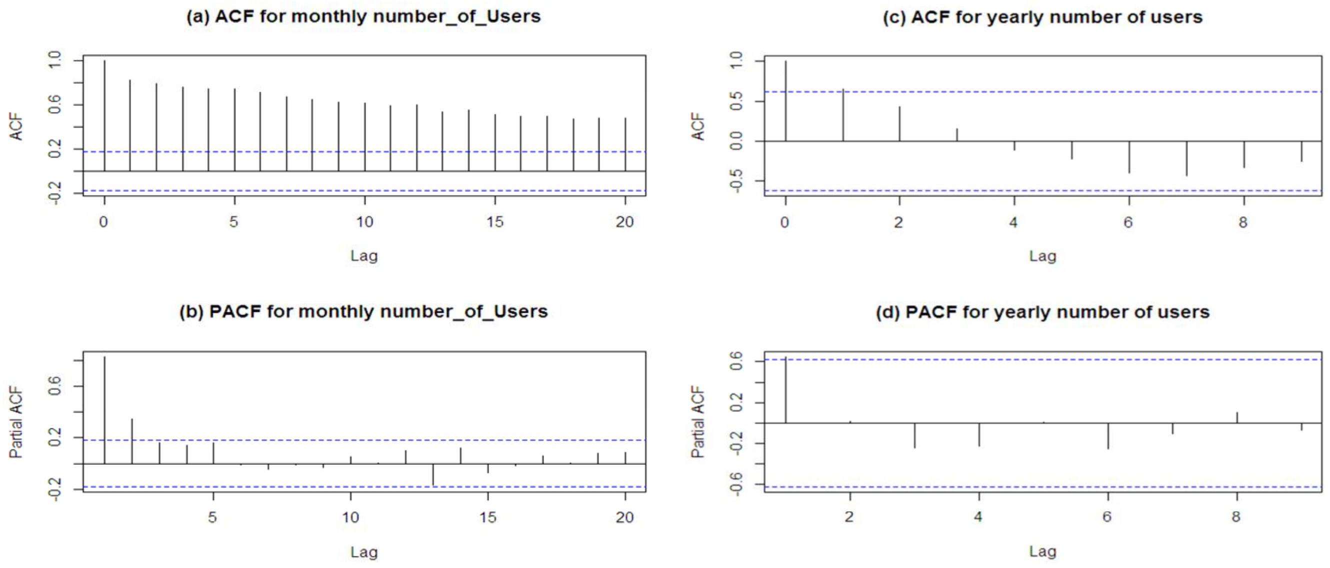

Figure 21: ACF and PACF results for number_of_users data

In Figure 21(a), the ACF shows a very slow decaying function which is an indication of non-stationarity. In the PACF there is a very strong first lag with the second lag becoming smaller. Thereafter the correlation of the other lags is insignificant showing no sign of autoregressive (AR) in the model.

4.5.2 Results of differenced datasets

The differenced series is just a white noise process. If plotted it looks like the figure of stationary series. In general, the order of integration can be thought of as the number of differencings a series requires to be made stationary. The main objective of this stage is to find the length of the differencing. For example, a non-stationary I(1) series, after it is differenced once, it becomes stationary. Similarly, an I(d) series is one which, when differenced (d) times, it becomes stationary. R language uses *diff()* function which takes each observation and differences it from the one previous to it.

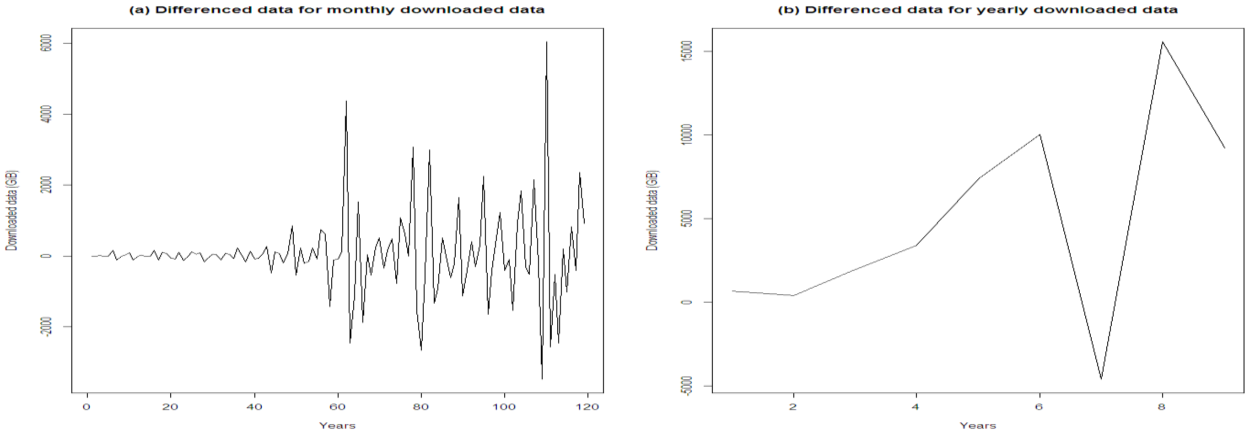

Figure 22: Results of differenced downloaded data

The plot no longer has the trend to it, only minor oscillation spikes are visible at the start of Figure 22(a). The oscillation spikes keeps growing large and larger which signifies that there is non-constant variance and shows random scattering of points. It also shows that the mean is little bit constant. The non-constant variance is not an issue at this stage but it can be corrected by taking the log.

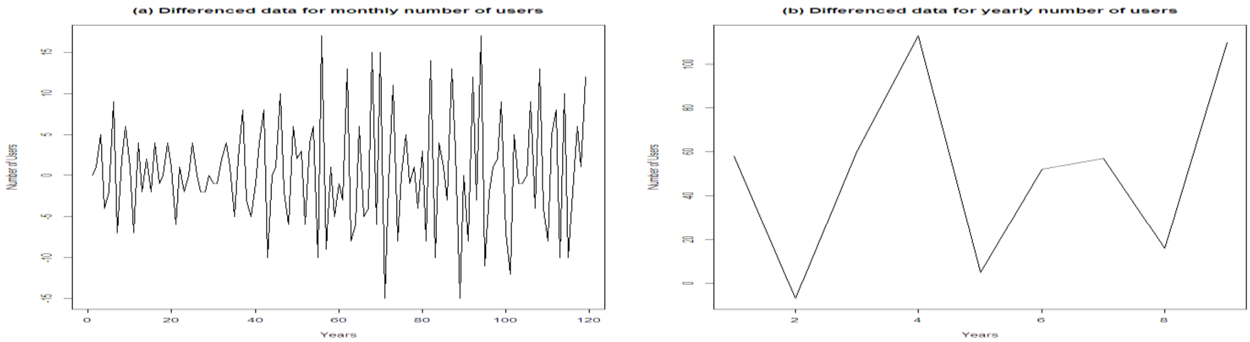

Figure 23: Results of differenced number of users

Figure 23 shows oscillation spikes which keeps growing large and larger indicating non-constant variance. The graphs do not have a trend which is an indication of a constant mean.

4.5.3 ACF and PACF of differenced data

The ACF and PACF plots of the differenced data can assist in determining the values of p or q. That can be helpful in determining the appropriate ARIMA model for prediction in the estimation stage.

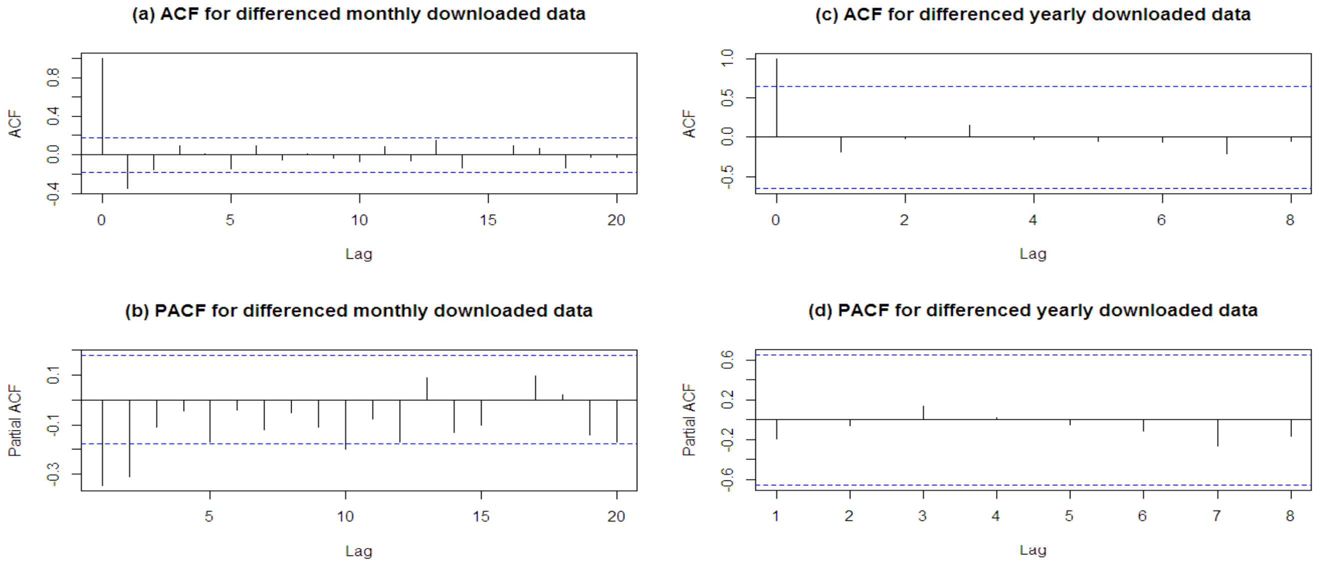

Figure 24: ACF and PACF results of differenced downloaded data

The ACF in Figure 24 tails off rapidly which is good. There are no regular spikes which correlates to that there is no seasonality component. In the PACF there is a strong first and second lags and then the other lags are all insignificant. They are outside of the confidence interval indicated by the blue lines.

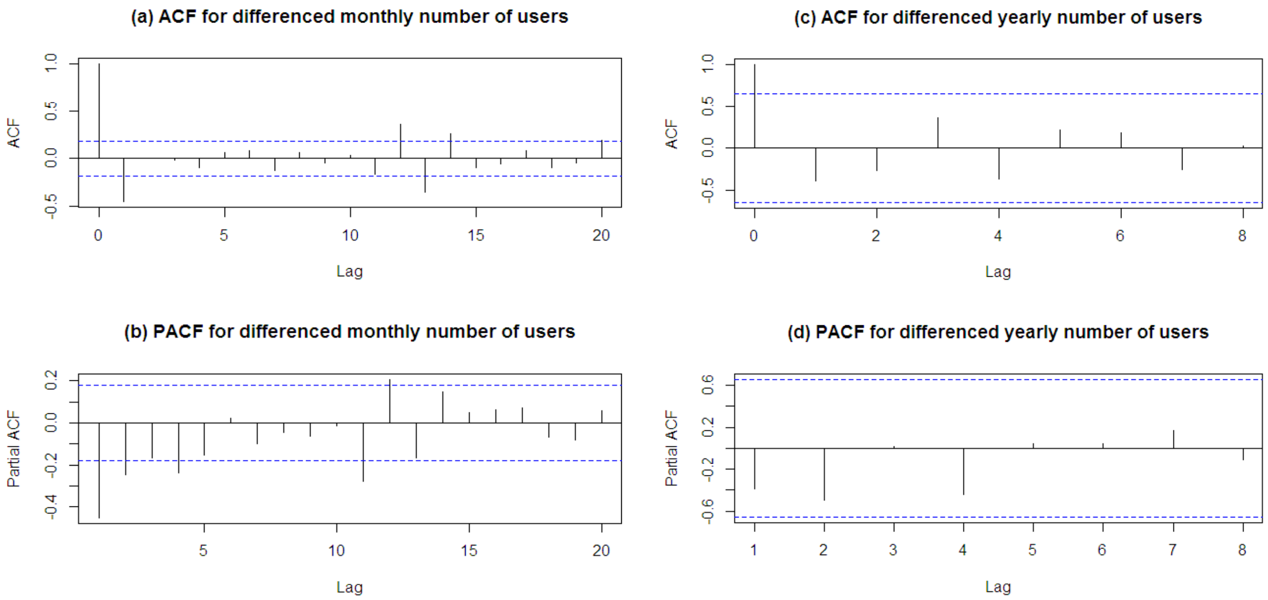

Figure 25: ACF and PACF results of differenced number of users' data

The ACF for number of users also tails off rapidly with regular spikes on the 12th period which corresponds to this to be monthly data. Since ACF has many regular spikes after the first one, it indicates AR(1) process. In the PACF, there are two spikes decreasing with the lag and then no significant spikes thereafter. This is an indication of having slow tailing off on the ACF of the original data which is also a signal of an AR(1) process.

4.5.4 Akaike Information Criteria results

The best ARIMA models are the ones with the lowest AICs. Their low AIC values suggest that the models meet the requirements of goodness-of-fit and parsimony. For example, since the best ARMA model for the first differenced monthly downloaded data is ARMA (1, 2). The monthly downloaded data will use ARIMA (1, 1, 2). It is worth noting that AIC has some limitations and that is the reason why some results show some non-stationarity and Not a Number (NaN) errors in Table 14 and Table 16. That indicates that the AIC for that combination is not possible with that kind of data. The best ARIMA models are highlighted in green and the worst ones are highlighted red in Table 13, Table 14, Table 15 and Table 16.

Table 13: AIC results of the 1st differenced monthly downloaded data

dMonthly_download data		**AR (p)**					
		0	**1**	**2**	**3**	**4**	**5**
MA(q)	**0**	1372.34	1367.96	1355.35	1357.06	1351.93	1353.29
	1	1352.09	1347.56	1348.29	1348.05	1348.69	1349.3
	2	1344.71	1340.96	1342.61	1344.43	1346.13	1348.12
	3	1344.63	1342.66	1344.24	1346.2	1348.13	1350.39
	4	1343.33	1344.46	1346.19	1342.14	1345.88	1352.2
	5	1344.88	1346.27	1348.13	1345.87	1352.17	1354.16

Table 14: AIC results of the 1st differenced yearly downloaded data

dYearly_download data		**AR (p)**					
		0	**1**	**2**	**3**	**4**	**5**
MA(q)	**0**	119.18	117.03	NaNs	NaNs	NaNs	NaNs
	1	116.85	124.33	115.44	NaNs	NaNs	NaNs
	2	116.41	117.06	114.82	120.04	NaNs	NaNs
	3	117.55	Error	117.68	117.85	NaNs	NaNs
	4	119.91	121.21	115.71	113.83	NaNs	NaNs
	5	121.45	122.06	NaNs	119.58	NaNs	NaNs

Table 15: AIC results of the 1st differenced monthly number of users

dMonthly_Users		AR (p)					
		0	1	2	3	4	5
MA(q)	0	541.5	517.43	507.24	504.02	502.65	504.07
	1	496.7	498.2	499.89	501.86	503.78	502.11
	2	498.12	500.02	501.87	503.86	505.76	502.83
	3	499.91	501.89	502.76	505.87	502.64	501.74
	4	501.87	500.03	498.38	502.02	503.31	502.88
	5	502.91	501.46	500.71	501.17	503.39	504.88

Table 16: AIC results of the 1st differenced yearly number of users

dYearly_Users		AR (p)					
		0	1	2	3	4	5
MA(q)	0	65.14	66.38	63.16	64.49	NaNs	NaNs
	1	63.96	65.95	64.02	65.57	NaNs	NaNs
	2	65.4	67.12	65.4	Error	69.33	72.05
	3	65.65	67.38	66.94	Error	71.24	73.33
	4	66.79	68.74	68.98	Error	72.83	74.87
	5	68.67	70.33	70.66	Error	74.58	76.54

4.5.5 Prediction results for ARIMA

Figure 26 shows the monthly and yearly prediction results for the downloaded data. The prediction results are coloured blue with upper and lower prediction bounds coloured red. The monthly data is predicted 12 months ahead and it imitates the previous patterns while the yearly data is predicted 2 years ahead. The predictions show some seasonal pattern just like the actual data with an upward trend. Figure 26(b) shows an anticipated increase of downloaded data to 49495.37 GiB in 2015 and thereafter an increase to 57431 GiB in year 2016.

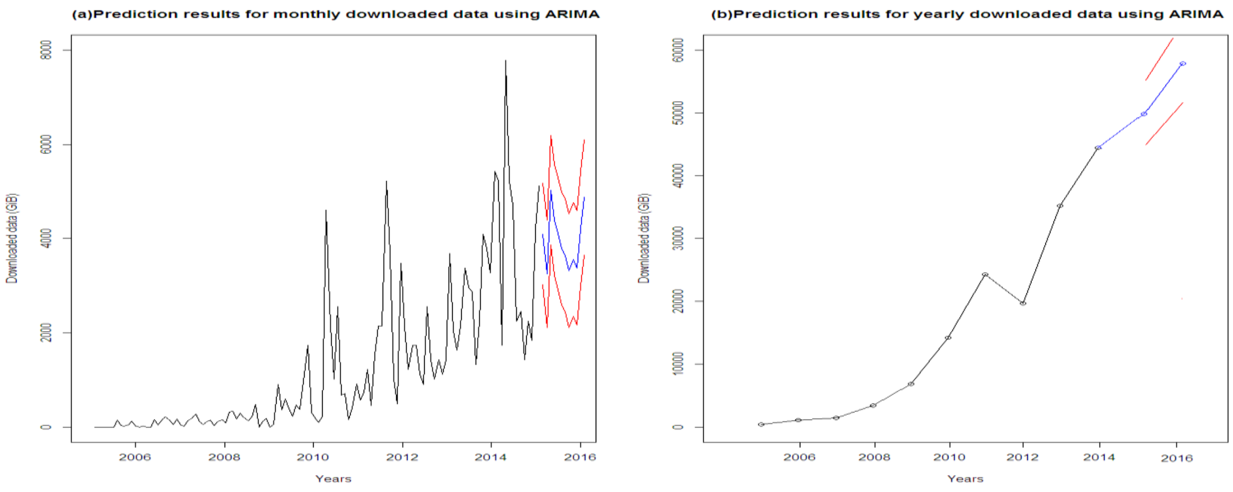

Figure 26: Prediction results for downloaded_data using ARIMA.

The monthly data is predicted 12 months ahead and it keeps the same upwards trend, mimicking the previous months. The yearly data is predicted 2 years ahead and also shows an upwards trend. The prediction results are coloured blue with upper and lower prediction bounds coloured red. The projections show an increase to 545 users in 2015 and 600 users in year 2016.

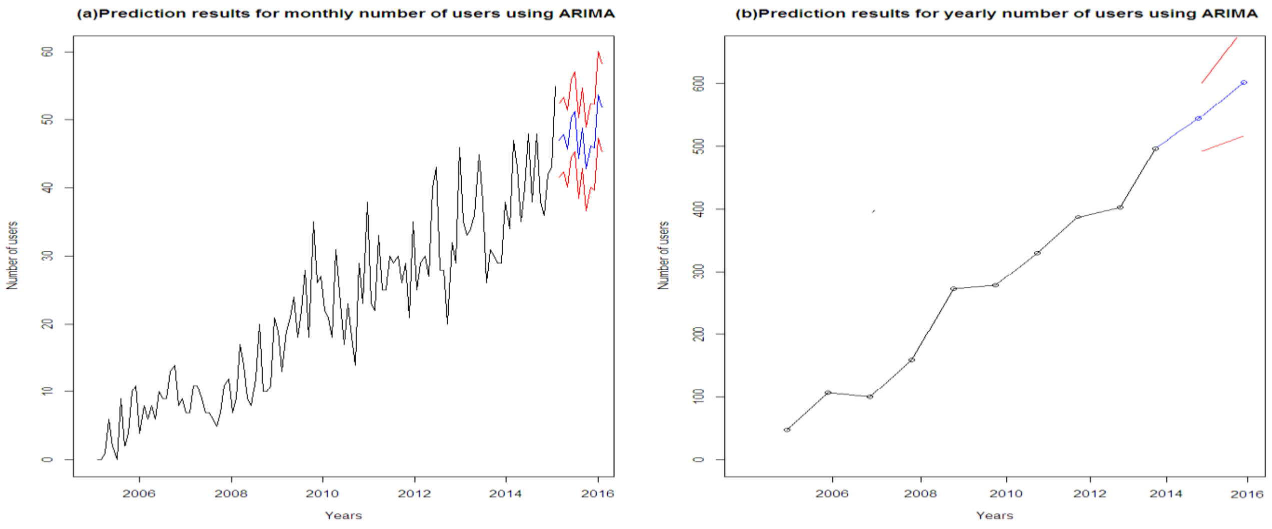

Figure 27: Prediction results for number_of_users data using ARIMA

4.5.6 Analysis results of best ARIMA models

The consolidated analysis results for the best ARIMA models are shown in Table 17. These models are the ones with the smallest AIC's as per section 4.5.4

Table 17: RMSE and MAE of best ARIMA models

	Downloaded_data		Number_of_Users	
	Monthly **ARIMA (1,1,2)**	Yearly **ARIMA (2,1,2)**	Monthly **ARIMA (0,1,1)**	Yearly **ARIMA (2,1,0)**
AIC	**1340.96**	**114.82**	**496.7**	**63.16**
RMSE	714.6	716.01	4.618	18.89995
MAE	419.3	564.468	3.685	16.20872

The best ARIMA models for monthly downloaded_data and yearly downloaded_data are ARIMA(1,1,2) and ARIMA(2,1,2) respectively. Similarly, for the number_of_users the best models for monthly and yearly data are ARIMA(0,1,1) and ARIMA(2,1,0) respectively.

4.6 Comparison of the prediction models.

In this sub-section the results of the four prediction models are discussed in details inorder to identify the best one. The comparative analysis is based on the prediction results obtained by each model. All the datasets were examined using SES, DES, TES and ARIMA models in order to do the comparative analysis. Exponential Smoothing methods started with testing for the best combination of alpha, beta and gamma parameters. Thereafter RMSE and MAE were calculated. As for ARIMA model, the process started by testing for the smallest AIC before calculating RMSE and MAE. The results obtained in this research are accurate and precise because 20 experiments done for each dataset are more than enough to test for the best solution.

4.6.1 Downloaded_data dataset

Table 18 shows RMSE and MAE results of the models based on the downloaded_data dataset.

Table 18: Analysis results of models based on downloaded_data

MODELS	Monthly downloaded_data		Yearly downloaded_data	
	RMSE	MAE	RSME	MAE
SES	862.9	419.853	5337.003	3969.268
DES	791.6628	387.4016	2346.092	1977.66
TES	843.3648	441.7421	2374.754	1950.505
ARIMA	714.6	419.3	716.01	564.468

From the results ARIMA model gave the smallest RMSE and MAE as compared to other models. For monthly downloaded_data ARIMA(1,1,2) was the best with RMSE of 714.6 and MAE of 419.3. On the other hand ARIMA (2,1,2) was the best for yearly downloaded data getting RMSE and MAE of 716.01 and 564.468 respectively. Double Exponential Smoothing(DES) method came second with TES taking the third spot. On the yearly data DES and TES results are almost the same with minor differences. SES became the last one with the highest RMSE and MAE results. Figure 28 shows a diagrammatic representation of the prediction results for the four (4) models.

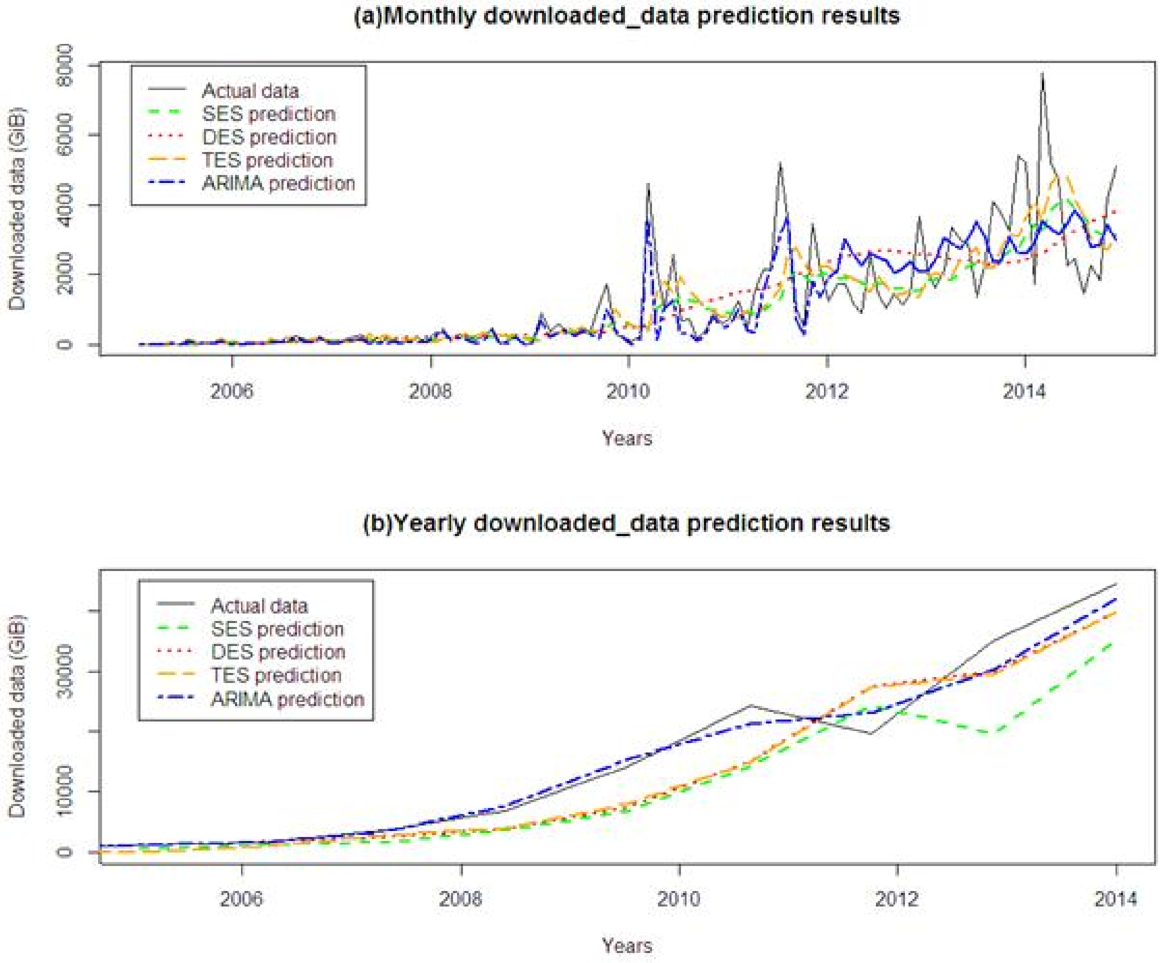

Figure 28: Models' prediction results based on downloaded_data

The predictions of the models do not accurately trail on the actual data in the graphs; however they all have an interesting trend especially on Figure 28(a) which goes up and down mimicking the actual data. The DES and TES results in Figure 28(b) are almost the same with ARIMA being much closer to the actual data.

4.6.2 Number_of_users dataset

Table 19 shows RMSE and MAE results of the models based on the number_of_users dataset.

Table 19: Analysis results of models based on number_of_users

MODELS	Monthly number_of_users		Yearly number_of_users	
	RMSE	MAE	RSME	MAE
SES	4.918458	3.788868	61.44153	49.35875
DES	4.773055	3.77382	38.64853	31.70763
TES	4.829408	3.779075	51.20327	35.3834
ARIMA	4.618	3.685	18.89995	16.20872

ARIMA model outclassed other models by getting the smallest RMSE and MAE results. The best ARIMA for monthly number_of_users is ARIMA (0,1,1) with RMSE of 4.618 and MAE of 3.77382. For the yearly number_of_users ARIMA (2,1,0) obtained RMSE of 18.90 and MAE of 16.21. DES obtained the second position reaching the lowest RMSE of 4.773055 and MAE of 3.77382 for monthly data. The difference between DES and TES results is insignificant. SES came last after obtaining higher results than any model. The prediction results of the four models are further compared in Figure 29.

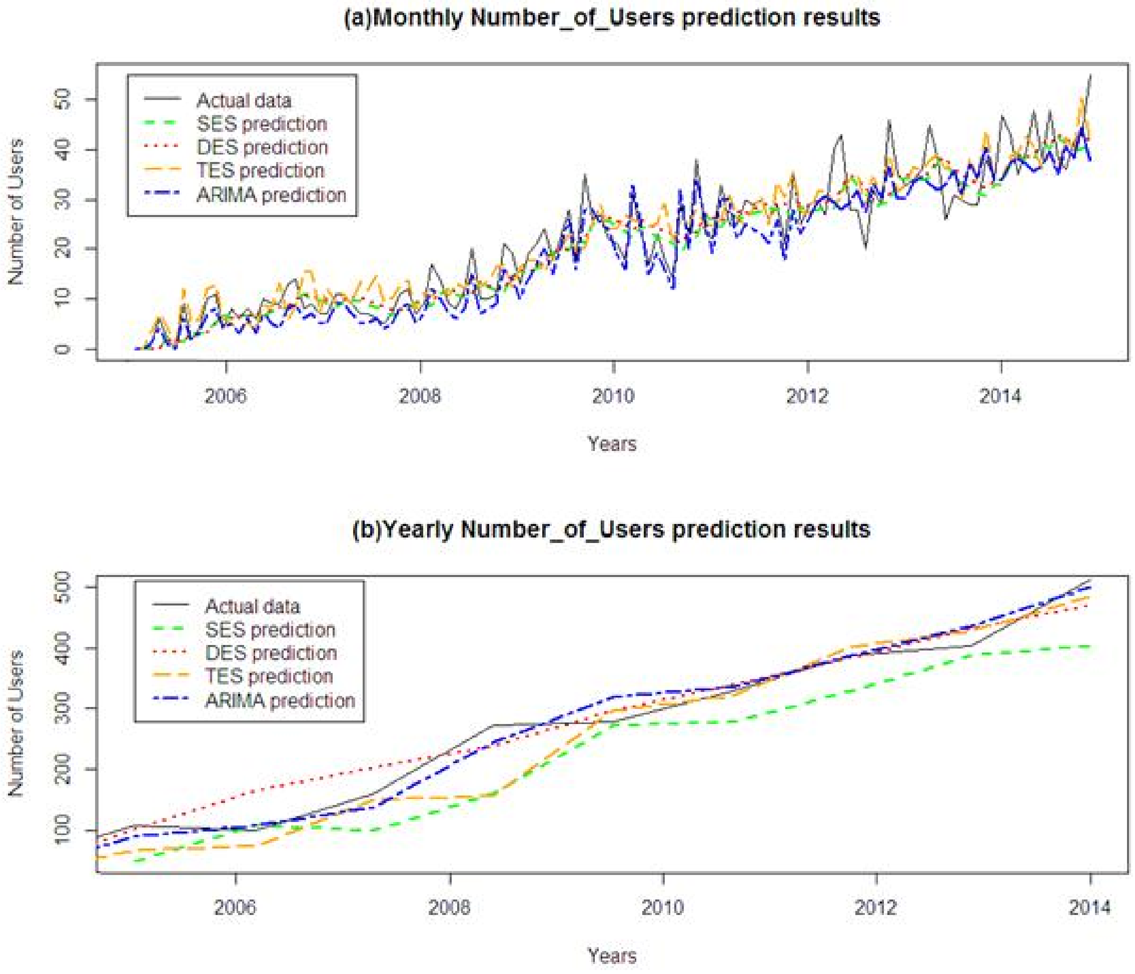

Figure 29: Models' prediction results based on number_of_users

The prediction techniques follow the actual data with ARIMA being too close. Most importantly, their forecast predictions follow an upwards trend. In Figure 29(b) SES predictions trail behind the actual data mimicking its behavior. That is so because its α parameter is 1.0. In that case the next forecasted value will always be the previous actual value.

4.6.3 Summary

The results in Section 4.6.1 and Section 4.6.2 show how the models responded to the proposed hypothesis. The results of all the four (4) datasets go in favor of ARIMA model because it has the lowest RMSE and MAE. The best results being that of monthly number_of_users, reaching RMSE of 4.62 and MAE of 3.69. The results support the first hypothesis. There after DES came second in all the experiments overcoming TES and SES models. This is the only part which did not go in favour of the hypothesis. According to the research hypothesis, TES was supposed to perform better than DES. In monthly number_of_users, TES got the results which are close to those of DES model. DES is normally best for data with trends while TES is best for handling parabolic trends and seasonality [59]. Parabolic trends tend to accelerate fast as shown in Figure 5(e). The results obtained for DES and TES indicate that there is no seasonality in the datasets. Even from the datasets (Figure 3 and Figure 4) there is no sign of any particular seasonality or any parabolic trend. The datasets used do not have any seasonality behavior and that is one of the reasons why DES performed better than TES.

SES came last in all the experiments obtaining higher results than any other model. Under the yearly number of users and yearly downloaded data, SES model got the best fit with alpha(α)=0.99. That parameter is almost equals to 1.0, something that is not usual. Based on formula (5), if alpha(α)=1.0, the next estimate will simply be the previous actual value. In that case the latest previous value is the one that matters. In contrary, if alpha(α)=0, it means that the actual values will be ignored and the forecasted value will be the same throughout. The results obtained show that SES is not the most accurate model even though it requires less computation.

The other goal of this dissertation was to study how the four prediction models perform when exposed to monthly and yearly data. Based on the results of Table 18 and Table 19, it is evident that the monthly RMSE and MAE results are less as compared to those of the yearly data. The results prove that the second proposed hypothesis is true. The models perform better on short-term data as opposed to long-term data.

CHAPTER 5 – SUMMARY AND CONCLUSION

5.1 Introduction

This chapter summarizes all the research work that has been carried out in this study. It includes the evaluation of the research objective and the overall challenges faced during the research. The chapter further recommends some of the possible future research work based on what has been learnt in the subject area.

5.2 Lessons learnt during the research

This research work concentrated much on cloud computing infrastructure and how its network congestion could be mitigated. During the research it surfaced that Cloud Computing is a hot topic which many enterprises are interested in it [12], [15-16]. Some organizations have little idea about it because it is still in its infant stages and not much has been published about it.

During this research, it emerged that cloud computing is categorized in to IaaS, PaaS and SaaS. Those justifying the boundaries of Cloud Computing and the services they offer. Some of the big companies at times offer all of them at the same time on pay-as-you go manner which is considered to be cheap. It also appeared that some services such as hotmail and gmail are offered in the cloud, something that was not known. Enterprises save their capital by renting cloud services instead of building their own data centers which is expensive. This relocation of enterprises to cloud services forces datacenter providers to manage their traffic load at all times. In a nutshell, cloud computing is emerging as a big and beneficial technology which is mostly ideal for medium and small sized enterprises in terms of cost.

5.3 Evaluation of Research Objectives

The evaluation of all the research objectives is as follows;

Table 20: Evaluation of research objectives

Research objective	**Evaluation of the objectives**
1. Analyze and compare prediction accuracy of SES, DES, TES and ARIMA models	The objective has been achieved by using four passive measurements (datasets). The models were used to predict the future behavior of each dataset. Thereafter the accuracy of the models was tested using MAE and RMSE accuracy functions as evidenced in section 3.8. The research results show that ARIMA is the best model for traffic predictions as per the research hypothesis. For one of the experiments, it forecasted the amount of CAIDA's downloaded data for 2015 and 2016 to be 49495.37GiB and 57431GiB respectively. That being a growth from 44441.6GiB of 2014. The results indication that CAIDA has to expect a growth in the number of downloads in 2015 and 2016 hence need to increase their server's processing power as well as memory. ARIMA model can be utilized by any cloud service provider for planning purposes. The future researchers can also base their research on it because it has proven to be the best model.
2. Training of the four prediction models in order to get the best parameters.	This objective was fully achieved. Each model was continuously trained in all the four (4) datasets in order to get the best combination of

Research objective	**Evaluation of the objectives**
	parameter. Twenty (20) experiments were run for each model. SE models used different alpha (α), beta (β) and gamma (γ) parameters. ARIMA model use the smallest AIC as the best. This experimental process is challenging because it is done continuously for each dataset.
3. Analyze the performance of SES, DES, TES and ARIMA models based on short term and long term data	This objective was achieved. Two datasets were collected monthly and the other two were collected yearly. Monthly was considered short time while yearly was considered long time between data points. During the experiments the behaviors of these models were analyzed to see how they perform based on these two time variations. All the models perform better on short term data than on long term data.

5.4 Challenges faced during the research

Finding the traffic datasets that are available for the public use was challenging. Most of the datasets were availed at a cost. It proved to be costly to continue measuring and using Amazon EC2 datasets. Some of the initial downloaded datasets needed some specialized tools to decrypt them. That process took much of the time before the free datasets supplied by CAIDA research organization were found.

Testing for the best fit using R- language was hectic because it required manual testing of all possible alternatives. For example ARIMA needed to be tested 25 times per each dataset in order to get the best AIC. As for Exponential Smoothing methods, twenty (20) experiments with different combinations of alpha and beta parameters had to be run per each dataset in order to get the smallest RMSE and MAE.

5.5 Overall Research Summary

This research provides an effective and proactive traffic prediction model to be used in IaaS cloud computing environment. The model that will arrest network congestion and facilitate effective resource management. Throughout the study, the performance of SES, DES, TES and ARIMA models were evaluated based on the traffic datasets obtained from CAIDA research organization. The data obtained gave a clear picture of how cloud datacenters operate. With the help of R-statistical package, the four models were compared and ARIMA turned out to be the best. All the predictions of ARIMA were in support of the research hypothesis. They were much precise as compared to those of other models. The comparison was based on RMSE and MAE accuracy methods which were also part of R-language.

The performance of the models was also evaluated when exposed to data with short term variation and long term variation. The monthly data was used as short term data while the yearly data was used as long term data. The results proved that all the models perform better on short term variation as compared to long term. In general, all the set out objectives were achieved with valid evidence.

5.6 Future Research Recommendations

This research investigated the best traffic prediction model in cloud computing environment based on monthly and yearly data. From this research it became clear that there is room for further research in the same subject area and two of the recommended research areas are as follows:

1. Testing for data seasonality in a cloud simulated environment using time series models. One of the challenges involving prediction research work is to test for data seasonality period. This project will involve creating a cloud simulated environment using CloudSIM simulator. Cloudlets or tasks will be initiated to virtual machines (VMs). Workload such as CPU Utilisation and throughput of the VMs will then be measured and analyzed to test for seasonality. If the data is seasonal, then seasonal models should be used to predict its future behavior. This type of research is scalable because it allows the researcher to perform any kind of experiment at no cost.

2. Comparing the prediction of ARIMA and FARIMA (Fractional Autoregressive Integrated Moving Average) models. FARIMA is a self-similar linear time series model with the capability of modeling processes with both the short-range dependent (SRD) and long-range dependent (LRD) characteristics. On the other hand ARIMA can only model process with SRD not LRD characteristics. The research will best determine how ARIMA performs when matched with FARIMA model in a cloud environment. The prediction models should be exposed to a wide variety of workloads; such as seconds, minutes and days, so as to test their performance.

REFERENCES

[1] P. Mell and T. Grance. (2009, July 10). *The NIST Definition of Cloud Computing v1.* [Online]. Available: http://www.nist.gov/itl/cloud/upload/cloud-def-v15.pdf

[2] F. Dupre. (2008, September 25). *Utility (Cloud) Computing* [Online]. Available: http://computinginthecloud.wordpress.com/2008/09/25/utility-cloud-computingflashback-to-1961-prof-john-mccarthy/

[3] J. Schwartz. (2014, June 03). *IaaS Magic Quadrant Report* [Online]. Available: http://rcpmag.com/articles/2014/06/03/azure-ranked-with-amazon-iaas.aspx

[4] Salesforce.com. (2014). *The all-in-one #1 CRM Solution* [Online]. Available: http://www.salesforce.com

[5] I. Astrova, S.G. Grivas and M. Schaaf, "Security of a Public Cloud," In *6th International Conf. on Innovative Mobile and Internet Services in Ubiquitous Computing (ICIMISUC).*, Palmero, Italy., July 2012, pp. 564–569.

[6] Center for Applied Internet Data Analysis. (2015, May 14). Passive Data Monitors [Online]. Available: https://www.caida.org/data/monitors/

[7] L. Wang, A. Nappa, J. Caballero, T. Ristenpart and A. Akella, "WhoWas: A Platform for Measuring Web Deployments on IaaS Clouds," In *Proc. of the 2014 Internet Measurement Conf. (IMC '14),* New York, NY, USA., 2014, pp. 101-114.

[8] B. Zhou, D. He, Z. Sun and W.H. Ng, "Network Traffic Modeling and Prediction with ARIMA/GARCH"., University of Surrey., Guildford, Surrey., United Kingdom

[9] X. Xiaobing, B. Chao and C. Feng, "An Insight into Traffic Safety Management System Platform based on Cloud Computing," *Journal of Procedia - Social and Behavioral Sciences (CICTP2013).,* vol. 96, pp. 2643-2646, Nov 2013.

[10] R. Sivakumar, E.A Kumar and G. Sivaradje, ‘Prediction of Traffic Load in Wireless Network Using Time Series Model,” In *Proc. of 2011 International Conf. on Process Automation, Control and Computing (PACC 2011).*, Coimbatore, India., 2011, pp. 20-22.

[11] S. Basu, S. Klivansky and A. Mukherjee, “Time series models for internet traffic,” In *Proc. of INFOCOM '96. 15th Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation (Volume:2).*, San Francisco, CA., 1996, pp. 611–620.

[12] J. Ciancutti. (2010, December 14). *Four Reasons We Choose Amazon's Cloud as Our Computing Platform* [Online]. Available: http://techblog.netflix.com/2010/12/four-reasons-we-choose-amazons-cloud-as.html

[13] S. Kandula, S. Sengupta, A. Greenberg, P. Patel and R. Chaiken, “The Nature of Data Center Traffic: Measurements and Analysis,” In *Proc. of the 9th ACM SIGCOMM Conf. on Internet measurement.*, New York, NY, USA., 2009, pp. 202–208.

[14] J. Rivera. (2013, October 8). *Gartner Identifies the Top 10 Strategic Technology Trends for 2014* [Online]. Available: http://www.gartner.com/newsroom/id/2603623

[15] M. Savage. (2013, October 15). *Cloud Traffic Will Rule The Data Center, Cisco Says* [Online]. Available: http://www.networkcomputing.com/networking/cloud-traffic-will-rule-the-data-center-cisco-says/d/d-id/1234482

[16] A. Venkatraman. (2012, October 24). *Datacentre traffic will grow four times by 2016, predicts Cisco* [Online]. Available: http://www.computerweekly.com/news/2240169111/Datacentre-traffic-will-grow-four-times-by-2016-predicts-Cisco

[17] H. Ballani P. Costa, T. Karagiannis and A. Rowstron, “Towards predictable datacenter networks,” *ACM SIGCOMM Computer Communication Review.,* vol. 11, no. 4, pp. 242–253, August 2011

[18] R. Wolski, “Forecasting network performance to support dynamic scheduling using the network weather service,” In *Proc. of the 6th High-Performance Distributed Computing Conf. (HPDC)*, Portland., OR, August 1997, pp. 316 – 325.

[19] Y. Xinyu, Z. Ming, Z. Rui and S. Yi, “A novel LMS method for real-time network traffic prediction,” In *Proc. of Computational Science and Its Applications (ICCSA).*, Assisi, Italy., May 2004, pp. 127-136.

[20] T. Benson, A. Akella and D.A. Maltz, “Network Traffic Characteristics of Data Centers in the Wild,” In *Proc. 10th ACM SIGCOMM Conf. on Internet measurement.*, New York, NY, USA., 2010, pp. 267–280.

[21] B.H Bhavani and H.S. Guruprasad, “Resource Provisioning Techniques in Cloud Computing Environment: A Survey,” *International Journal of Research in Computer and Communication Technology.*, vol. 3, no. 3, March 2014.

[22] A. Gandhi, Y. Chen, D. Gmach, M. Arlitt and M. Marwah, “Hybrid resource provisioning for minimizing data center SLA violations and power consumption,” *Journal of Sustainable Computing: Informatics and Systems.,* vol. 2, no. 2, pp. 91–104, June 2012.

[23] D. Gmach, J. Rolia, L. Cherkasova and A. Kemper, “Workload Analysis and Demand Prediction of Enterprise Data Center Applications,” In *Proc. 2007 IISWC of IEEE 10th Symp. Workload Characterization.*, Washington DC, USA., 2007, pp. 171-180.

[24] Z. Gong, X. Gu and J. Wilkes, “PRESS: PRedictive Elastic ReSource Scaling for Cloud Systems,” In *Proc. of IEEE International Conf. on Network and Services Management (CNSM)*, Niagara Falls, ON., 2010, pp. 9-16.

[25] A.A. Bankole, “Cloud Client Prediction Models for Cloud Resource Provisioning in a Multitier Web Application Environment,” M.S. thesis, Dept. Sys. and Comp. Eng., Carleton Univ., Canada., 2013.

[26] M. Zaharia, A. Konwinski, A.D. Joseph, R. Katz and I. Stoica, “Improving MapReduce Performance in Heterogeneous Environments,” In *Proc. of OSDI’08 of 8th USENIX Conf. on Operating systems design and implementation.*, Berkeley, CA, USA., 2008, pp. 29-42.

[27] D.A Dickey, and W.A Fuller, "Distribution of the Estimators for Autoregressive Time Series With a Unit Root," *Journal of the American Statistical Association*, Vol. 74, No. 366, pp. 427-431, 1979.

[28] Q. Zhang, L. Cheng and R. Boutaba, “Cloud computing: state-of-the-art and research challenges,” *Journal of Internet Services and Applications.*, vol. 1, pp. 7-18, 2010.

[29] P.A.A. Gutiérrez, A. Bulanza, M. Dabrowski, B. Kaskina, J. Quittek, C. Schmoll, F. Strohmeier, A. Vidacs and K. S. Zsolt, “An advanced measurement meta-repository,” In *Proc. of 3rd International Workshop on Internet Performance, Simulation, Monitoring and Measurement (IPS-MoMe).,* Warsaw, Poland., 2005.

[30] H. Wang, W. Ding and Z. Xia, “A Cloud-Pattern based Network Traffic Analysis Platform for Passive Measurement,” In *Proc. 2012 Int. Conf. on Cloud and Service Computing (CSC’12).*, Washington, DC, USA., 2012, pp. 1-7.

[31] M. Allman and E. Blanton, "A Scalable System for Sharing Internet Measurements," In *Proc. of Passive and Active Measurements, PAM 2002*, Fort Collins, 2002.

[32] Center for Applied Internet Data Analysis. (2012, June 12). *Correlating Heterogeneous Measurement Data to Achieve System-Level Analysis of Internet Traffic Trends* [Online]. Available: http://www.caida.org/projects/trends/

[33] A. Akella, S. Seshan and A. Shaikh, “An Empirical Evaluation of Wide-area Internet Bottlenecks,” In *Proc. of the (ed) 3rd ACM SIGCOMM Conf. on Internet measurement.*, New York, NY, USA., 2003, pp. 101–114.

[34] J. A. Daniel, “Data Management in the Cloud: Limitations and Opportunities,” *IEEE Data Eng. Bull.*, vol: 32, pp. 3-12, January 2009.

[35] A. R. Curtis, W. Kim and P. Yalagandula, “Mahout: Low-overhead datacenter traffic management using end-host-based elephant detection,” In *Proc. of IEEE INFOCOM ’11.*, Shanghai, China., 2011, pp. 1629 – 1637.

[36] Y. Peng, K. Chen, G. Wang, W. Bai, Z. Ma and L. Gu, “HadoopWatch: A First Step Towards Comprehensive Traffic Forecasting in Cloud Computing,” In *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications.*, 2014, doi: 10.1109/INFOCOM.2014.6847920

[37] A. Sang and S. Li, “A predictability analysis of network traffic,” In *Proc. of INFOCOM 2000 19th Annual Joint Conf. of the IEEE Computer and Communications Societies (Volume:1).*, Tel Aviv, Israel., 2000, pp. 342 - 351.

[38] J. Schad, J. Dittrich and J.-A. Quiané-Ruiz, “Runtime measurements in the cloud: observing, analyzing, and reducing variance,” *Proc. of the VLDB Endowment*, vol. 3, no. 1–2, pp. 460–471, September 2010.

[39] B.L. Dalmazo, J.P. Vilela and M. Curado, “Online Traffic Prediction in the Cloud: A Dynamic Window Approach,” *Int. Conf. on Future Internet of Things and Cloud (FiCloud).,* pp. 9-14, August 2014, doi:10.1109/FiCloud.2014.12.

[40] P. Papadopouli, E. Raftopoulos and H. Shen, “Evaluation of short term traffic forecasting algorithms in wireless networks,” In *IEEE Conf. on Next Generation Internet Design and Engineering (NGI’06).*, Valencia., 2006, pp. 102–109. doi: 10.1109/NGI.2006.1678229.

[41] Y. Qiao, J. Skicewicz and P. Dinda, “An Empirical Study of the Multiscale predictability of network traffic,” In *Proc. of the 13th IEEE International Symposium on High Performance Distributed Computing (HPDC 2004).*, Honolulu, Hawaii, June 2004, pp. 66-76.

[42] M. F. Zhani and H. Elbiaze, “Analysis and Prediction of Real Network Traffic,” *Journal of Networks*, vol. 4, no. 9, pp. 855-865, Nov, 2009.

[43] E.S. Gardner., “Exponential smoothing: The state of the art,” *Journal of Forecasting.,* vol. 4, no. 1, pp. 1-28, 1985.

[44] Center for Applied Internet Data Analysis. (2014, November 4). *Archipelago Measurement Infrastructure* [Online]. Available: http://www.caida.org/projects/ark/

[45] Equinix, (2015). Cloud Infrastructure Solutions [Online]. Available: http://www.equinix.com/solutions/cloud-infrastructure/

[46] Center for Applied Internet Data Analysis. (2015, May 1). Passive Monitor: (Hardware) [Online]. Available: https://www.caida.org/data/monitors/passive-equinix-chicago.xml

[47] Center for Applied Internet Data Analysis. (2015, January 2). *Downloads of CAIDA Online Datasets* [Online]. Available: http://www.caida.org/data/about/downloads/tables.xml

[48] D. Chhajed and T.J. Lowe (eds.), “Little’s Law,” in *Building Intuition: Insights From Basic Operations Management Models and Principles*, NewYork, NY, Springer Science + Business Media, 2008, ch. 5, sec. 1, pp. 81-100, doi: 10.1007/978-0-387 -73699-0.

[49] R Project. *Introduction to R* [Online]. Available: http://www.r-project.org/about.html

[50] D. Lillis, “Use R for data analysis and research,” *New Zealand Science Review.,* vol. 68, no. 2, pp. 73-79, 2011.

[51] P.J. Brockwell and R.A. Davis, “Forecasting Techniques,” in Introduction to Time Series and Forecasting, New York, Springer, 1996, ch. 9, pp. 317.

[52] *Stationarity and differencing* [Online]. Available: http://people.duke.edu/~rnau/411diff.htm

[53] R.J. Hyndman and Y. Khandakar, “Automatic Time Series Forecasting: The forecast Package for R,” *Journal of Statistical Software,* vol. 27, no. 3, pp. 1-22, July 2008.

[54] S. Makridakis, S.C. Wheelwright and V.E. McGee, *Forecasting methods for management.,* New York – Chichester – Brisbane – Toronto – Singapore., John Wiley & Sons, 1983.

[55] P.R. Winters, "Forecasting Sales by Exponentially Weighted Moving Averages," *Management Science.*, vol. 6, no. 3, pp. 324–342, April 1960.

[56] T. Wood. (2012, January 23). *Using Mean Absolute Error for Forecast Accuracy* [Online]. Available: http://canworksmart.com/using-mean-absolute-error-forecast-accuracy/

[57] *Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE)* [Online]. Available: http://www.eumetcal.org/resources/ukmeteocal/verification/www/english/msg/ver_cont_var//uos3/uos3_ko1.htm

[58] J. Fox, C. Murray and A. Warm, "Conducting research using wed-based questionnaires: Practical, methodological, and ethical considerations," *Int. Journal of Social Research Methodology*, vol. 6, pp. 167-180, 2003.

[59] H. Arsham, (2015). *Forecasting by Smoothing Techniques* [Online]. Available: https://home.ubalt.edu/ntsbarsh/business-stat/otherapplets/ForecaSmo.htm