Methods for Cloud Workload Analysis and Cloud Cost Forecasting

Saurabh Desale[1], Varad Gujar[2], Atharva Raut[3], Satej Patil[4], Vitthal Gutte[5]
[1,2,3,4]*B.tech Students, School of Computer Engineering and Technology, Dr. Vishwanath Karad MIT World Peace University Pune, India*
[5]*Assitant Professor, School of Computer Engineering and Technology, Dr. Vishwanath Karad MIT World Peace University Pune, India*

Abstract **- Cloud Computing has become the information technology (IT) backbone for all types and size of businesses. Organizations, as well as individual users, are extensively using cloud services and resources for their everyday business transactions to individual IT needs. Most typical cloud computing costing models are based on pay peruse. While such costing models are suitable for many businesses, the challenge it imposes to end consumers is the difficulty to plan the budget as the cost is known only post the cloud resource usage. This paper address this very challenge with cloud storage resource for manifestation, in this paper, we have analyzed workload costing of the ubiquitously consumed cloud storage resource based on the block type of storage known as a standard persistent disk. Based on cloud storage as a resource we simulated storage workload (derived from the publicly available stats of amazon's e-commerce website monthly traffic) using file input-output (fio) simulation tool on a Google Cloud Platform's E2 compute machine. The simulation data is recorded in a time series format that contains cloud storage consumption and cost for each month. This paper then analyses and presents ARIMA time series model which is trained on the simulated data to forecast the cloud storage cost for the next upcoming months.**

Index Terms **- Cloud Computing, Forecasting, Workload Analysis, Cloud Cost, Time Series Model.**

I.INTRODUCTION

The domain of cloud computing [8] technology is expanding expeditiously as cloud computing offers wide numbers of computing services such as compute services, database services, storage services, networking services, IOT services etc. And variety of tools for technologies like machine learning and big data. There are many cloud service providers such as Google cloud, Amazon web services, azure by Microsoft who deliver all these computing services on a single infrastructure hosted on web. Many companies and individuals make use the cloud services for their daily computational tasks as it cuts down on investing for a high-end machine. There consist of mainly three types of services of the domain cloud computing [9] which are 1) Infrastructure as a service (IaaS) which provides only a basic work environment where the cloud services and resources has to be fully customized by an user. 2)Software as a service (SaaS) which includes all computational resources and features in one environment where an used can directly consume some example of (SaaS) are Gmail, Drive, Dropbox. 3)Platform as a service (PaaS) is a development-based platform which provides services like development tools, testing tools and provides hoisting and deployment for applications.

A workload is characterized based on the usage of resources mainly hardware resources which are storage, GPU, CPU, and memory. Analyzing the workloads given pattern of usage is what is called a workload analysis. In all e-commerce premised websites an online transaction processing (OLTP) is seen which is CPU and I/O intensive. ERP workloads are memory intensive workload were SAP HANA in-memory database runs. Cloud workload is where the interaction is with the internet which consists of virtual machines, online containers, web servers, databases which are handled by remote servers or any instance at a given time are said as cloud workload. Cloud storage workload such as Meta-Data-Test is a types of storage workload which can be achieved by using simulation tools like Flexible Input Output (FIO) on compute engine, examples of cloud storage can be google drive and drop box as these applications are based on block storage type. The parameters for

costing of cloud workload are based on hardware and software, maintenance of an application and provision of workload charges.

The area of machine learning and its forecasting models [5] are being used in almost all fields of work. To forecast/predict the future result outcomes using forecasting modelling it requires historical data available in a dataset format. We have made use of forecasting modelling to forecast/predict the future cloud cost. There are various forecasting models like K-Means, Naïve Bayes, Time Series, Random Forest etc. We have made use of time series model for the forecasting as are workload simulation data is serially correlated data which provide well founded forecasting. Time series forecasting model have few methods that can be used forecasting a model, the methods are 1) Autoregression (AR), 2) Moving Average (MA), 3) Autoregressive Integrated Moving Average (ARIMA), 4) Seasonal Autoregressive Integrated Moving Average (SARIMA), 5) Vector Autoregression (VA). We are used ARIMA model for cloud cost forecasting as it is more flexible compared to another statistical model.

II.REVIEW OF LITERATURE

A.Review Paper on Cloud Computing [1]

In this study the authors have well explained paper which covers almost all-important topics related to cloud computing domain. It lists out the important three main layers of cloud computing such as software, platform, infrastructure-based service models and their individual functionalities and features. Gap here that can be identified in this review paper is that the paper has not introduced the cloud cost pricing model based on the services the cloud services providers provide. This paper can come handy for those who are new and want a quick and detailed guide towards cloud computing technology.

B. A Research paper on Smart Metering of Cloud Services [2]

In this paper the authors have used ARIMA statistical model to predict the cost of cloud, they have also proposed a well price and bill model for the smart metering for cloud service provides such as Amazon and Rack space. They have listed monitoring tools for resource monitoring such as Nagios or Hyper Sigar, but they have opted for Hyper Sigar API framework. An Application Program interface framework which can be used for utilization of resources monitoring. A flow of algorithm for monitoring of resources has been given along with price forecasting and calculation of bill. To conclude they have obtained a well billing model for the services of cloud for the intelligent metering in power girds. Gaps that can be identified in this paper is that the during the implementation for price prediction the use of auto ARIMA is not implemented that would have given more reliable parameters for the ARIMA (p, d,q). The advantage of reading this paper is that it gives a detail explanation of the time series model ARIMA for resource monitoring and price prediction.

C. Role of predictive modeling in cloud services pricing: A Survey[4]

In this survey the authors present a well detailed model for pricing for the well-known cloud service providers like Azure, Amazon. The paper lists out some of the popular predictive methods such as binary classifier, logistic regression, and linear regression. It also lists out factors that can affect the costing of cloud. Gaps identified here are that less in detail explanation has been given in the section of the factors that affected in cloud computing. This paper can be handy for those who are planning to implement forecasting modelling.

III.PROPOSED ARHITECTURE

The scope of the proposed architecture is given in Fig 1 and is divided into 5 interrelated phases.

A. Workload provisioning

Here the required cloud resource (Storage) has been allocated on a compute engine. A storage workload has been simulated by using file input output tool on a VM instance where valued parameters are assigned such as numjobs, time duration, size of file in MB.

B.Metering the resources

The workload is simulated with different variations in a snap of 1 to 2 weeks. The daily utilization of storage as a resource and cost of utilized storage will be recorded in a time-series sheet having months, amazon traffic data, storage used in GB and MB and Cost. After the simulation recording is completed an csv file including cells for month and cost in usd is created which will be used for the heuristic/ prediction phase.

C.Curation of price and storage as a resource usage data

Normalize the price and storage as a resource usage data into common denominator and assuming 1MB equal to 1GB and the cloud price in USD and getting recorded data ready to proceed heuristic/prediction phase.

D.Forecasting of future Bill

Study the existing literature and prior art to understand and listing out potential time series model such as ARIMA algorithm and approaches available for prediction or forecasting using such time series model to forecast the cloud cost. After the forecasting model is trained and tested on a python IDE. So, the results can be visualized.

E.Insights via visualization

The output of the prediction model can be visualized in a single pane of glass on the web where our model will be deployed, where one can upload their historical storage workload analysis costing, select the upcoming months for which they want a forecasted results in an input. In the backend the user's dataset file will be posted and the Arima model will be trained on their dataset and will generate forecasted results in the form of graph and table.

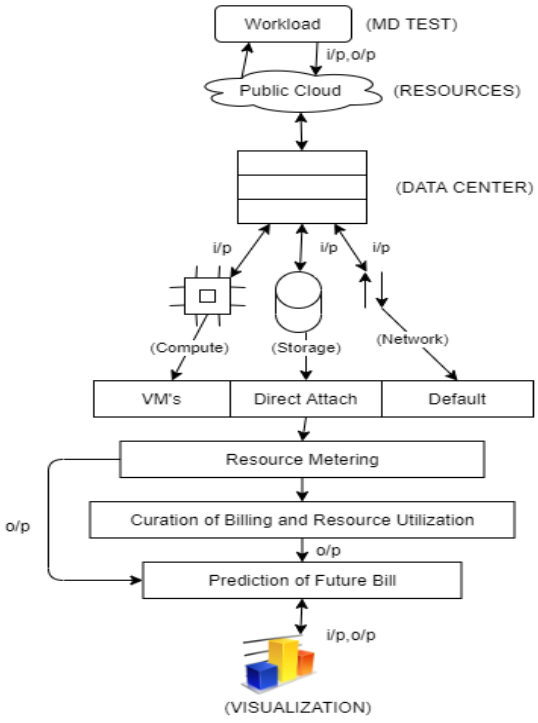

Fig-1 Cloud cost forecasting architecture

IV.IMPLEMENTATION

The implementation is divided into two parts, first part is the simulation of the storage workload on a Vm instance using FIO, and second part is the time series forecasting using ARIMA forecasting model to forecast the cloud cost and storage resource utilization.

A. Simulation of a Storage Workload.

The Simulation of a workload by using simulation tools life file input output on a cloud infrastructure and its execution is described in 4 steps.

1) Creating an VM instance on a cloud infrastructure: A compute engine (VM instance) is created on a google cloud service infrastructure. VM having hardware/software specification such as E2 series, Debian OS as an operating system, a, machine type of e2micro, GPU of a2highgpu-1g(12vCPU) and a 20GB of standard persistent disk attached.

2) Installation of required packages like simulation tool FIO and creating multiple directories for the simulations. Explaining FIO (flexible input output) is basically an open-source Input Output tool that can be used for hardware verification and also can be used for benchmarking of a system. In our case we have used it for randomly writhing files and a given parameters for the storage workload simulation. The FIO command contains parameters like name, its ioengine, iodepth, random read, random write, block size, size of the file (Input) which a user has to input based on their analysis, numjob's which are basically threads that create number on random read write files that one assigns, and last parameter is the runtime that defines the time for which one simulation will run.

3) Creating a Time series recording sheet which contains cell like i) Amazon Traffic in numbers of users (in millions) ii) Number of database processes required to handle the load(numjobs), iii) Workload Storage Consumptions (in MB), iv) Workload Storage Consumptions with simulation parameters with hypothesis of (1MB=1GB) in GB, v) Price in USD (Google cloud standard persistent disk price per GB=0.04USD) in the given table: below. Before the simulation to find out the numjob what we will be needing for the simulation. A formula has been declared to find out the numjob's value for a simulation it is given as.

$$Numjob's = \frac{CUT - MUT}{35} + MN \quad (1)$$

Here in our amazon online traffic analysis the minimum user is 2059 million, current user is the targeted value. In the above equation CUT is current user traffic, MUT is minimum user traffic and MN is the minimum numjobs, 35 is the difference for minimum number of users and current number of users and adding 1 numjob or every 35 number until the count reaches to desired current users, and the minimum numjob's that we have considered in 5 Numjobs which is assigned to the minimum users traffic.

4) Simulating and recording: After finding out the value of numjob's, next step is the start of simulation using FIO command given as: fio --name=simulate1--ioengine=libaio --iodepth=1 --rw=randwrite --bs=4k --direct=0 --size=158M --numbs=5 --runtime=300--group_reporting. Now recording the the simulated data in a sheet and calculating the cost in USD. In our recording there are total 24 cells for recording we have added 5 to show.

TimeLine	M1	M2	M3	M4	M5
Amazon Traffic in numbers of users (in millions	2059	2189	2224	2324	2300
Number of database process required to handle the load (numjobs)	5	9	10	13	12
Workload Storage Consumptions (in MB)	790 MB	1422 MB	1580 Mb	2054 MB	1896 MB
Workload Storage Consumptions with simulation parameters with hypothesis of (1MB=1GB) in GB	790 GB	1422 GB	1580 GB	2054 GB	1896 GB
Price in USD(Google cloud standard persistent disk price per GB=0.04USD)	32	57	63	82	75

Table 1: Simulation Data Recording

B. Forecasting of cloud cost and storage resource utilization.

Making use of ARIMA time series methods the forecasting of cloud cost and resource utilization is possible. ARIMA is one the most industry claimed model for forecasting, some popular areas where ARIMA has been actively used are stock market prediction, whether forecasting, product sales prediction, etc. ARIMA model makes use of historical data and trains a model based on it and gives the forecasting result for the coming time duration. In Arima there are three terms p,d,q where p stands for auto regression , d stand for difference/integrate and q stands for moving average. Where these terms are used train the ARIMA model and are responsible for the accuracy of the model based on their values that range from 0 to 3 respectively. There are two was to determine pdq value that are i) Plotting correlation graphs to determine the values, ii) Making use of auto Arima library to automatically determine to pdq values. The equation of Arima model is:

$$Y_t = c + \phi_1 Y_{t-1} + \cdots + \phi_p Y_{t-p} + \theta_1 e_{t-1} + \cdots \theta_q e_{t-q} + e_t \quad (2)$$

Here the Y_t is the variable, c is he given constant also called as intercept, ϕ is p coefficient, θ is q coefficient and e_t is the error time.

1) Determining Auto Regression.

Auto Regression model forecasts result based on previously generated lagging values (t-1), based on past behavior it predicts the future behavior. In Arima AR is given by the p term. The formula for auto regression is.

$$Y(t) = \sum_{i=1}^{p} h(i) \cdot y(t-1) + \varepsilon(t) \quad (3)$$

$$y_t = c + \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \phi_3 Y_{t-3} + \cdots \phi_p Y_{t-p} + \varepsilon_t \quad (4)$$

In the given mathematical equation c is determined as a constant, y_t is the time for which the variable is dependent. Y_{t-1} ,Y_{t-2} and Y_{t-3} are the previous time period variables, and p is the order which is given in Arima and ε is the noise, h are the coefficients. P value can be found out by plotting partial auto correlation plot.

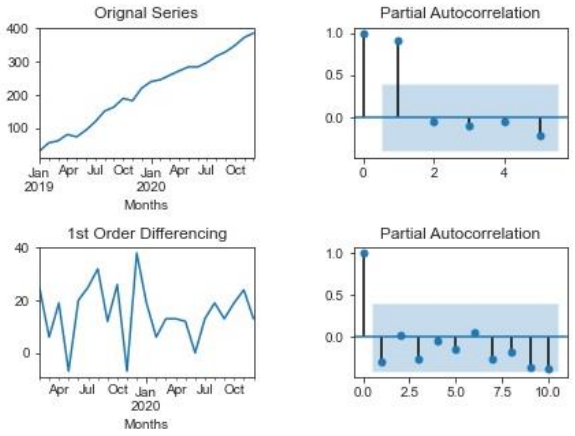

Fig:2 A Graph of Partial Autocorrelation

As seen in our original series of graphs with a differencing of 0, as we add a differencing of diff=1 we see the data is now stationary and we have got our p value as 1, where we see a sharp curve in the confidence interval that value is considered as a p value and can be added to the Arima's p value.

2) Determining Moving Average.

Moving average is an error-based prediction model which is given by q in Arima model, in moving average previous errors are considered in action to forecast the coming result. The formula for moving average is.

$$y_t = \mu_0 + \varepsilon_t - \omega_1\varepsilon_{t-1} - \omega_2\varepsilon_{t-2} - \omega_3\varepsilon_{t-3} - \cdots \omega_q\varepsilon_{t-q} \quad (5)$$

In the given equation y_t is the time for which the variable is dependent, $\omega_1\varepsilon_{t-1}$ are the previous time period errors occurring, where time y_t is dependent upon the q which are the previous error values.

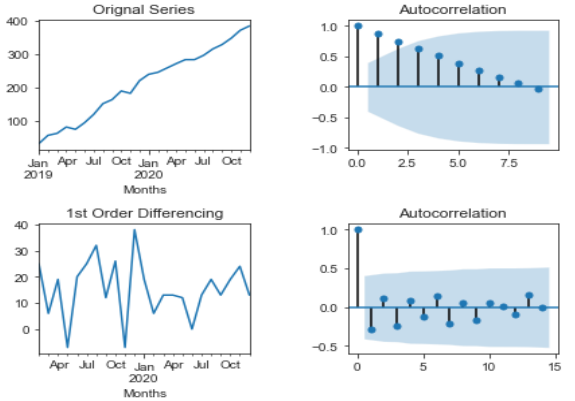

Fig:3 A Graph of Autocorrelation

As seen in the graph we see the differencing in the first order, and as observed a steep curve in the confidence interval of the 1st order of differencing. We get the value for difference 1 that is 1.

3) Determining Difference(d).

In Arima difference is used to convert a non-stationary data into a stationary data by determining the d team called as difference or integrate, In the below Fig: 4 you can observe that are data is in a upward trend which is a non-stationary.

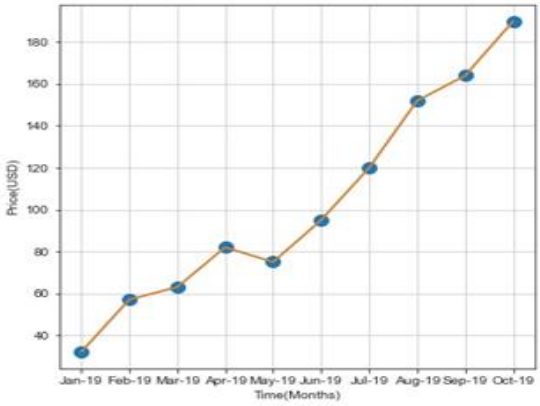

Fig:4 A non -stationary graph

This non-stationary data can be converted to stationary by giving a differencing, difference value can be achieved by running a Augmented Dickey Fuller Test which is used to test null hypothesis of unit root which is been present in a time series. The adf test result returns a p-value, if any p -value after giving the difference value is greater than 0.05 its considered as a not valuable value and cannot be taken as a differencing value, but if the p-value is less than 0.05 then the difference value that was assigned to the adf test can be considered in the 3 Arima parameters. In our case we found out the p-value less than 0.05 at the difference of 1 so we considered it in our Arima parameters.

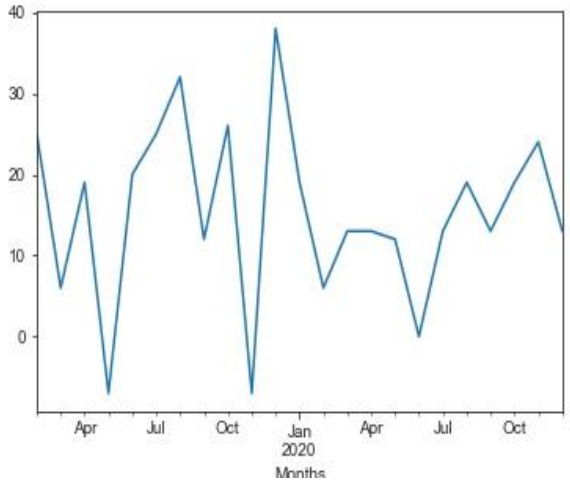

Fig:5 A stationary graph

As seen in Fig:5 after adding a difference of 1 the data is now turned into stationary data.

C. Algorithm Pseudo Code

1) Using the Classical way to forecast using Arima for Cost Forecasting:

In the Classical way we must find out Arima p, d, q terms in order to train the model and forecast it likewise.

i) Plot (Cloud cost dataset)
ii) train_split_percentage = training percentage value (%)// Divide the data into train and remaining into test
iii) while (graph is non-stationary = True)
{adding a difference(k_diff) of 1 or more to make graph stationary}
iv) Plot (PACF graph)
// To find out the p value (AR)
v)Plot(ACF graph)
// To find out the q value (MA)
vi)forecast_model_fit ← ARIMA (order=p, d, q)
return: forecast_model_fit.
vii)forecast_model_fit. summary ()
viii)Plot (Forecasting graphs)
// Result in the form of visualization's

2) Making use of Auto Arima

In the above B) section discusses regarding the classical way to determine the pdq values by plotting correlation graphs and testing values. So Auto Arima is used to automatically find the best p, d ,q values by doing a grid search and the model with the lowest AIC(Akaike information criterion) score is best model sequence of p,d,q terms. Making use of Auto Arima function once does not need to plot graphs or test values in order to get the p, d, q terms. In the below Fig 6, we get the best model in sequence (0,1,0) seeing the lowest AIC score of 151.355.

```
Performing stepwise search to minimize aic
 ARIMA(2,1,2)(0,0,0)[0] intercept   : AIC=inf, Time=0.15 sec
 ARIMA(0,1,0)(0,0,0)[0] intercept   : AIC=151.355, Time=0.01 sec
 ARIMA(1,1,0)(0,0,0)[0] intercept   : AIC=151.532, Time=0.02 sec
 ARIMA(0,1,1)(0,0,0)[0] intercept   : AIC=151.552, Time=0.02 sec
 ARIMA(0,1,0)(0,0,0)[0]             : AIC=167.761, Time=0.01 sec
 ARIMA(1,1,1)(0,0,0)[0] intercept   : AIC=152.923, Time=0.04 sec

Best model:  ARIMA(0,1,0)(0,0,0)[0] intercept
Total fit time: 0.242 seconds
```

Fig:6 Auto Arima results

In the above Fig 6, we get the best model in sequence (0,1,0) seeing the lowest AIC score of 151.355. The Algorithm Pseudo Code is:

i) Plot (Cloud Cost dataset)
ii) train_split_percentage = training percentage value (%)
// Divide the data into train and remaining into test
iii) auto_arima (train_data ['Cost data'])
// Running auto Arima on the training data
iv) return: best model ← ARIMA (order)]
//returns best model based on AIC Score.
v) forecast_model_fit ← ARIMA (order=p, d, q)
return: forecast_model_fit.
vi) forecast_model_fit. summary ()
vii) Plot (Forecasting graphs)
// Result in the form of visualization's

V.RESULTS

In this section we have displayed the results of our implementation, we have taken the graphs and tables form for the visualization.

A. Splitting of data into train and test data

In the below graph Fig 7 we have we have distributed the weightage for training and testing where 85% is given for training and remaining 15 % is to test with the forecasted and actual results. As seen the y-axis shows the price in USD and x-axis shows the months. So out of 24 total months, 20 months are for training as seen Jan 2019 to Aug 2020. And Sep, Oct, Nov and Dec are for the testing. The blue line indicates the trained price and orange line indicated the testing price.

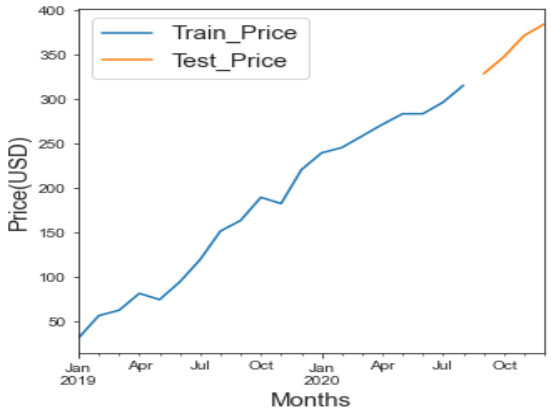

Fig 7: Train and Test graph

B.Forecasted Results.

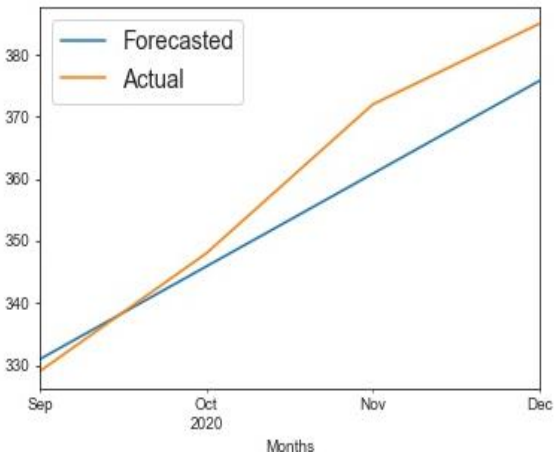

Fig 8: Actual vs Forecasted graph.

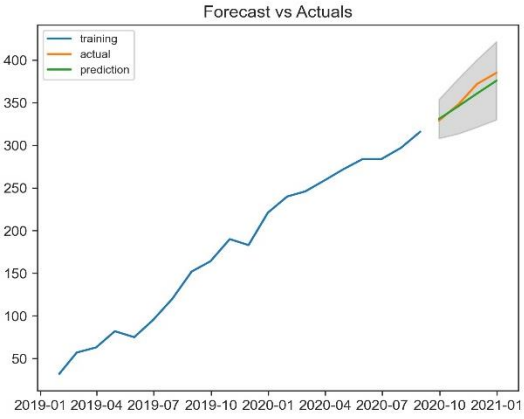

Fig 9: Train and Actual vs Forecasted graph

In the given Fig 9 includes the trained data in blue, the forecasted data green and actual test data in orange. During the implementation we had given the months that was to be forecasted same as the testing data months that are in total 4 months for testing. So as seen we get next forecasted results for the next 4 months applying Arima algorithm. Observing the prediction and actual results we see that are results are quite similar to the actual results.

Months	Cost in USD
2020-09-30	329 USD
2020-10-31	348 USD
2020-11-30	372 USD
2020-12-31	385 USD

Table 2: Actual Test Data

Months	Cost in USD
2020-09-30	330 USD
2020-10-31	345 USD
2020-11-30	360 USD
2020-12-31	375 USD

Table 3: Forecasted Data

Here in above tables Table 2 is the actual test output and Table 3 is the forecasted/predicted result for the given 4 months and the results can be compared.

C. Testing the accuracy using mean square error.

Using Mean squared error method we can find out the number of error occurrence in our model, lesser the mean squatted error better the accuracy of the model. In our model the mean squared error value is 7.3 which is comparatively very less and makes our model accurate. Sqrt function must be declared containing the forecasted series and test data. The formula for mean squared is:

$$\sqrt{\frac{\sum(predicted\_value - real\_value)^2}{n}} \qquad (6)$$

Here the summation of predicted/forecasted value has been subtracted with the actual input data values squaring it and dividing it with the number of data in the database, then the whole square root is evaluated and the mean squared error is found.

VI.CONCLUSION

Cloud computing is amongst the most growing technology and is widely studied and been used. Our approach towards this project and study can come handy to provide accurate forecasting of the cost using time series Arima model and can benefit companies as well as for an individual who will make use the cloud services. The Future scope can extend to apply to pricing prediction of other cloud-based services or resource utilization forecasting. The enhancement can include a detailed comparative study of various other time series non time series models like k-means, Garth, Dynamic Linear Models (DLM) for model casual effects or Singular Spectrum Analysis.

REFERENCES

[1] P. Srivastava and R. Khan, "A Review Paper on Cloud Computing", International Journal of Advanced Research in Computer Science and Software Engineering, vol. 8, no. 6, p. 17, 2018. Available: 10.23956/ijarcsse.v8i6.711

[2] A.Narayan, S. Rao, G. Ranjan, and K. Dheenadayalan,, “Smart metering of cloud services,” in *2012 IEEE International Systems Conference SysCon 2012*, 2012.

[3] VK. Prasad, A. Nair and S.Tanwar ,"Resource Allocation in Cloud Computing", Research gate: Instant Guide to Cloud Computing (pp.343-376).
[4] M. Kandpal, M. Gahlawat, and K. Patel, "Role of predictive modeling in cloud services pricing: A survey," in 2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence, 2017.
[5] M. Borkowski, S. Schulte, and C. Hochreiner, "Predicting cloud resource utilization," in Proceedings of the 9th International Conference on Utility and Cloud Computing, 2016.
[6] S. Namasudra, P. Roy, and B. Balusamy, "Cloud computing: Fundamentals and research issues," in 2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM), 2017.
[7] S. Singh and I. Chana, "Metrics based Workload Analysis Technique for IaaS Cloud," arXiv [cs.DC], 2014.
[8] V.S. Gutte and K. Iyer, "Cost and Communication Efficient Framework for Privacy Assured Data Management Cloud" International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8 Issue-4, April 2019.
[9] V.S. Gutte and D. Sita ""Achieving Cloud Security Using a Third Party Auditor and Preserving Privacy for Shared Data Over a Public Cloud" International Journal of Knowledge and Systems Science (IJKSS), Volume 11, Issue 1 , January-March 2020, DOI: 10.4018/ IJKSS.2020010104.