Time-Series Models for Cloud Workload Prediction: A Comparison

Abiola Adegboyega
Electrical & Computer Engineering
The University of Calgary
Calgary, Alberta
aadegboy@ucalgary.ca

***Abstract*—dynamic cloud workloads necessitate forecasting methodologies for accurate resource provisioning affecting both cloud providers and clients. This paper focuses on forecasting in the cloud in order to understand its underlying workload dynamics. It analyzes recent workload traces and discovers characteristics that are not adequately captured by traditional linear & nonlinear models employed for forecasting in the cloud. This paper completes a comprehensive statistical analysis of 8 workloads realized from production cloud environments. Through characterization, time-series elicitation and model fitting, it isolates a limited but important set of statistical distributions that capture cloud traffic dynamics. Furthermore, it adopts a recent econometric modeling technique called the Autoregressive Conditional Score (ACS) model that improves forecasting accuracy over existing methods. To exploit our findings from the workload characterization of the traces, we also extend the ACS model to realize a variant called ACS-*l* that models errors using the lognormal distribution. Compared with existing models, the ACS-*l* offers a 10%-25% improvement in forecasting accuracy when right-tailed distributions are observed in workloads. Furthermore, the score-based characteristics observed in time-series and their diversity has inspired a novel classification of cloud workloads into three distinct groups according to the most appropriate model: linear, nonlinear and hybrid models. A methodology that employs statistical measures to guide this selection has also been developed.**

Keywords—forecasting; errors; prediction; workloads

I. Introduction

The characterization of aggregate cloud workloads and its application in prediction makes for *accurate* provisioning whereby resources can be allocated over appreciable forecast windows into the future. Forecasting is however challenging due to the attendant fluctuation in cloud workloads given the diversity of applications and the cloud pay-as-you-go deployment model. Current practices to mitigate load fluctuation include resource over-provisioning & scaling [1],[2]. These however lead to inefficient resource usage while impacting both customer and provider Quality-of-Service (QoS) and profit margins.

The ability to accurately forecast future workloads given cloud application diversity is of primary importance in the achievement & maintenance of customer QoS objectives. This paper focuses on the characterization of workloads that represent the diversity of cloud applications as well as their usage areas. Eight unique datasets from production cloud environments used in current research were selected. They include storage, video, web & analytics workloads. We elicit each individual workload's time-series and employ statistical methods to capture salient features. The methods can be generalized for the variety of existing cloud workloads provided their accumulated history is available for realization as a time-series. The work here discovers a limited set of statistical distributions that define the studied time-series and corroborates the same findings in current research. It also examines and models volatility exhibited with the development of methodologies that effectively tracks such workload dynamics as observed in production cloud environments. The methods here discussed improve forecasting accuracy by 10% – 25% when compared with existing methods.

Existing methods for time-series prediction in the cloud are based primarily on linear models captured in the Auto-Regressive Integrated Moving Average (ARIMA) model of Box and Jenkins [3]. Their use in online prediction is understood for arrival processes that are well understood and linear models are adequate. Beyond linear models, cloud traffic volatility captured by the statistical property of variance has inspired the adoption of nonlinear econometric models. The Generalized Auto-Regressive Conditional Heteroskedastic (GARCH) model of Engle [4] has found application in the modeling and forecasting of cloud traffic variability [5]. Recent studies however stress the need for augmenting both linear and nonlinear methods discussed in order to efficiently track workload dynamics given modeling drawbacks. Furthermore, recent studies [6],[7] indicate the need for the realization of new statistical models to effectively capture cloud traffic dynamics.

In this paper traffic characterization is employed in the realization of a novel time-series model. The salient feature is the modeling of time-series errors, the difference between its original and forecasted value, by capturing volatility differently from the variance as done in classical nonlinear models. Here, it is captured with the score function that provides a more accurate measure of volatility based on the conditional probability distribution of observed errors. The integration of this component into the realized model has demonstrated improvement in forecasting accuracy. The new model affords a tradeoff between the complexity of nonlinear models and the simpler features employed in linear models. The summary of contributions is:

- A novel workload selection methodology with a *global* view that determines when linear models are suited to time-series under study and when there is statistical justification to pursue nonlinear models. The introduction of the score function enables the realization of models that

978-3-901882-89-0 @2017 IFIP

bridge the gap from simple to complex model selection. Current practices are limited to either linear or nonlinear models often without a statistical decision making methodology in place to determine the model selection.

- A novel time-series model that captures the dynamics of cloud workloads specifically in the area of storage traffic.
- A forecasting algorithm realized as two variants which integrate model-based estimators for future time-series prediction over time-windows that are useful for resource provisioning. Their forecasting advantages and drawbacks are explained.

The rest of the paper is organized as follows. Section II details the datasets selected for study from current research, the statistical basis that provided a new perspective on error modeling and subsequently the novel workload characterization methodology. Section III presents the novel time-series model developed. Section IV presents the performance evaluation of the forecasting algorithm and prediction comparisons with existing methods. Section V presents related work while Section VI provides conclusion and future work.

II. DATASETS STUDIED

The datasets selected for study are listed in table 1 below. The diversity of datasets explored is similar to work by Di, Kondo and Cirne [8] where 8 workloads were also studied. Series I is from a comprehensive study of workloads obtained from 10 datacenters [7] & is composed of multicast video traffic in a multi-layer networked datacenter environment. Series II comes from the dataset of the well-researched Google compute cluster of 12,500 nodes spanning one month of collection. Series IIIA and IIIB are from a private production IaaS cloud cluster running business critical workloads [9]. The dataset is aggregated from the communication of 1750 VMs spanning 4 months for CPU, Disk, Memory and Network I/O. Series IVA and IVB were released from current research in characterizing video traffic [10]. The environment is a video-server cluster providing streaming services. Series V and VI come from an extensive characterization of traffic from the popular personal storage platforms of Dropbox, Box and SugarSync [11].

The analysis of all the time-series realized from the datasets employs bandwidth as the metric of observation. An initial comparison is done in terms of the standard deviation and the Coefficient of Variation (CoV). This metric serves as a first measure of variability. It is however of limited use given that it becomes an inefficient metric of variability if the mean value under observation is of magnitude close to zero. It however serves its purpose as a starting point in the realization of metrics that are better able to track variability applicable to the time-series under study. The method of analysis follows.

A. Analysis Methodology

Upon the realization of time-series for each dataset listed, the standard methodology employed in analysis was used [12], a process that involves initial visual analysis. We adopt the signal + error modeling approach given that its basis for the linear classical models of Box and Jenkins [3]. With reference to Figure 1, the plot of each time-series is subjected to an initial visual inspection to discover observable properties such as trends and seasonality. Real data traffic may contain outliers and gaps which should be removed to benefit more accurate modeling. The identification of these properties is evidence of non-stationarity, a property whereby its statistical measures of the mean and variance are non-constant with time. Doing a logarithm transformation and/or differencing of the initial time-series is done to achieve stationarity. Subsequently, the Auto-Correlation Function, ACF, is examined. This is a measure of any relationships that may exist between the observations of the time-series over lags. The ACF of a series Y_t is given by:

$$ACF(t,h) = \frac{E[(Y_t - \mu_t)(Y_h - \mu_h)]}{\sigma_t \sigma_h} \quad (1)$$

Where μ is its mean value, σ is the standard deviation and t,h is the lag, the separation over time at which the values of the time-series are observed. After the determination of the model order, observable as the number of lags after which the ACF graph decays exponentially, the errors are examined to determine their statistical properties. The standard assumption is that they are Gaussian white noise for classical linear regressive models. Through the analysis of the time series

Table 1: Basic Time-Series Statistics

Series	Type	Metric	Mean	CoV	S.Dev.
I	IaaS	Packets/s	104904	52.45	55031
II	Compute	Jobs/min	132399	10.98	14532
IIIA	IaaS	Megabits/s	485	47.42	230
IIIB	IaaS	Megabits/s	204	54.9	112
IVA	VoD	Megabits/s	158	45.36	71.67
IVB	VoD	Megabits/s	181	34.86	63.10
V	Storage	Kbytes/s	821	16.32	134
VI	Storage	Kbytes/s	843	79.12	667

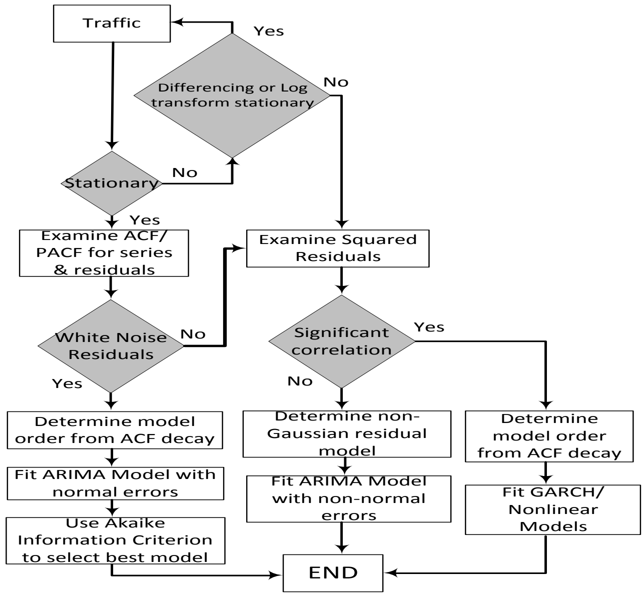

Figure 1: Time-Series Model Fitting Methodology

studied, three types of errors have been identified according to their distributions: (1) Gaussian errors (2) Right-tailed errors and (3) Heavy tailed errors.

With Gaussian observations in the errors, the ARIMA modeling process follows. With reference to Figure 1, when the errors are non-Gaussian and the log transformation and differencing does not yield Gaussian errors, the methodology examines either the squared errors or fits the observed error distribution. The examination of the ACF of squared errors enables the determination of when the nonlinear GARCH model can be adopted. Furthermore, we explore the modeling of non-Gaussian errors as an alternative to GARCH models. To do so, we avail ourselves of a recent modeling method which enables the realization of hybrid models that measures traffic variability different from the standard nonlinear measure of variance while still being able to retain the autoregressive components of linear models. We proceed with an analysis of the arrival process of all time-series studied.

B. Arrival Process

In Figure 2, the empirical Cumulative Distribution Function (CDF) for the arrival process of each time-series is illustrated. The disparity in the bandwidth measures have been normalized in order to bring all series into one graph for easier visual exploration. With the exception of series V, it can be observed that a large percentage of the arrival process for all series is dominated by small values which suggests fitting with heavy-tailed distributions. This is evident if we consider the sections of the CDF graphs that account for the arrival process at 60% and 80% for all time-series studied. To corroborate this initial visual conclusion, the histogram for each series was observed after which statistical testing was completed to determine the model with the best fit. Figure 3 illustrates representative distributions for 4 of the studied time-series. It will be observed that a right-tailed distribution is common to series IIIB & IVB. Series II fits a (skewed) student-t distribution while the normal distribution is observed for series I. Observations from fitting the empirical histograms discovered three types of distributions: normal, skewed and right-tailed distributions. This determines the workload model for the original time-series while playing an important role in the modeling of its errors which will be discussed subsequently. Subsequent fitting was done according to the observed distributions and Akaike's Information Criterion (AIC) was employed to determine the model with the best fit.

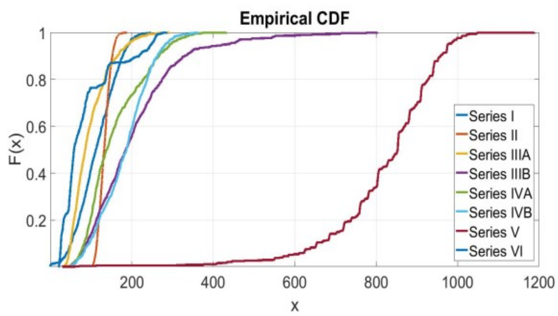

Figure 2: Empirical CDF for all Time-Series

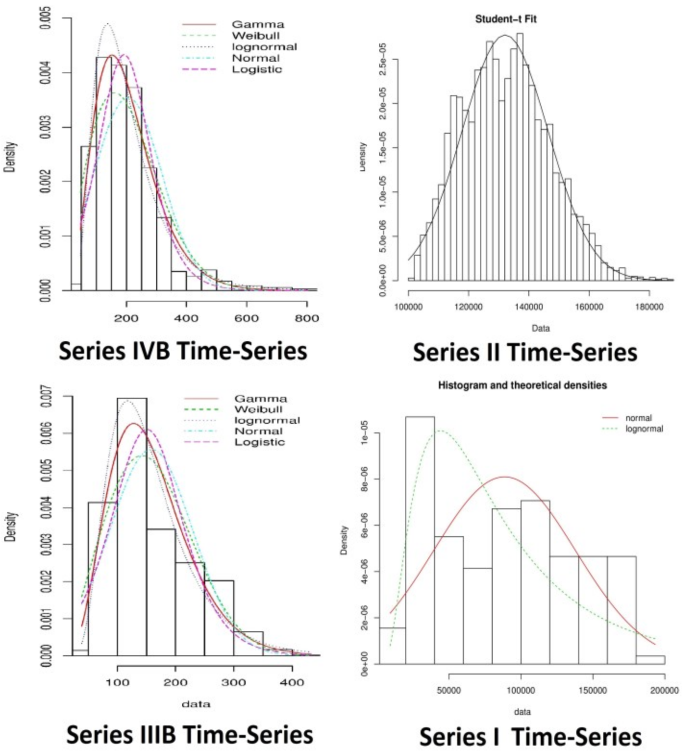

Figure 3: Empirical Histogram for Selected Time-Series

For series IIIB & IVB, the lognormal distribution returned the lowest value for the AIC. This observation was made for both the original time-series and in the probability distribution of the error term. This also corroborates observation of the lognormal distribution among others in the right-tailed family, for arrival processes as done in current research [13-15]. The same procedure was carried out for all time-series studied.

C. Model Fitting

The focus of workload characterization enables the discovery of features that enable isolating the best model. It also enables the classification methodology. This proceeds with Figure 1. Using linear models as a starting point, initial time-series differencing and log transformation yields a stationary form of the series by which to determine the autoregressive component done by an examination of the ACF graph. The examination of errors follows and this guides the selection of models as linear, nonlinear and hybrid. Using one representative plot from each group, Figure 4 provides the ACF & empirical histograms, one each, for the classification of models as linear, nonlinear and hybrid. Series I's ACF decays rapidly after the first four lags and can be described as white noise thereafter. The histogram also shows the regular bell-curve that describes the Gaussian distribution. Series II didn't yield normal errors with a log transformation and differencing. The squared errors show evidence of correlation as shown, it suggests evidence of time-variation in the variance of the time-series otherwise described as Heteroscedasticity in the econometrics literature [4]. ARIMA models are not suited to volatility. Series II shown in

Figure 5a is differenced in Figure 5b. Here, it displays non-constant magnitude the phenomenon of time-variation in variance as described in the econometrics literature. Series IIIB presents an interesting departure from the white noise errors observed in series I as well as the squared errors of series II. A log-transform and differencing did not result in stationary errors. Furthermore, squaring the errors did not show correlation over appreciable lags. The observation of skewed distributions in the original time-series for series IIIB suggests the realization of models better able to capture traffic dynamics as observed in cloud environments.

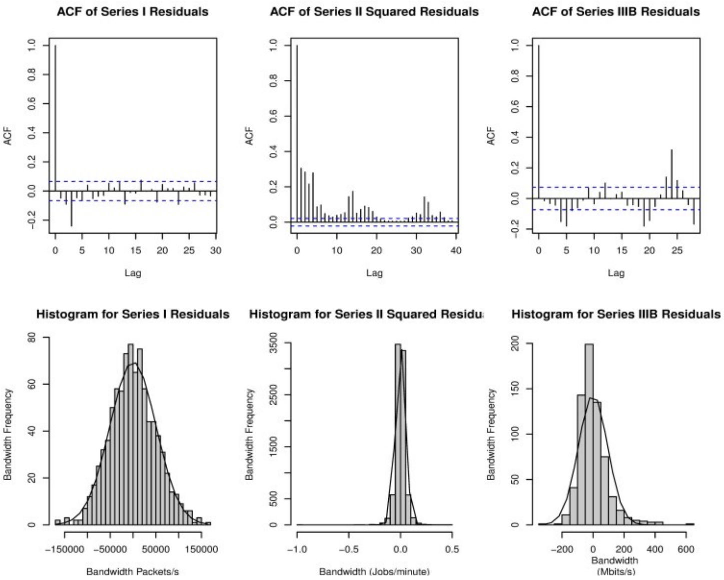

Figure 4: Error ACF & Histograms for Series I, II and IIIB

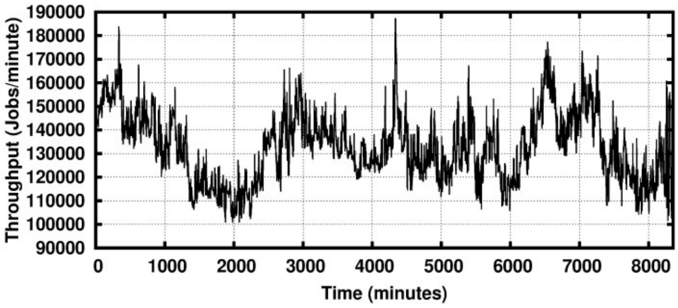

Figure 5a: Series II from Google's Compute Cluster

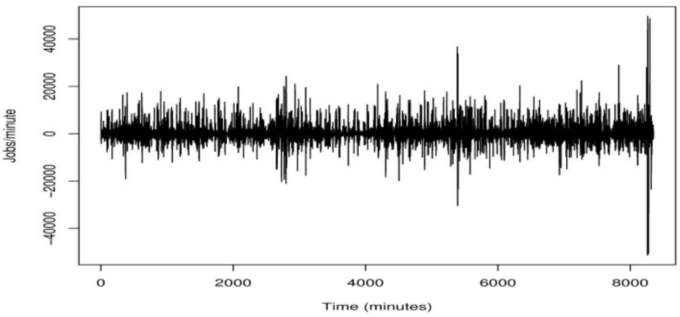

Figure 5b: Series II after taking a first difference.

With reference to Figure 4, the empirical histogram of the errors for series IIIB shows a right-tailed distribution. Furthermore, conclusions from current research regarding the arrival process and inter-arrival processes for compute and storage clusters are of right-skewed distributions [13-15]. The Ljung-Box test for error autocorrelation was conducted for all time-series studied. Furthermore, based on the observations and an analysis of their errors as illustrated, the series are classified into linear (I, IVA), nonlinear (II,V,VI) and hybrid (IIIA, IIIB, IVB). For the linear models as illustrated, the error observations that follow the Gaussian distribution and ARIMA models were deemed fit. For the nonlinear models, the squared errors displayed significant correlation as determined by employing the Ljung-Box test. For the hybrid models, these were determined for those series where both the original and errors show skewed empirical distributions. For these, right-skewed distributions were observed for those in the study and the lognormal distribution returned the lowest AIC value. The realization of models for fitting is done after discussing related work.

III. MODELING

A. Linear ARIMA Models: Mean As Estimator

In the standard linear ARIMA model, we denote the independent variable (input application traffic say) by X_t with the error denoted as z_t an additive component: $Y_t = X_t + z_t$, with X_t regressed on itself to order p & coefficients $b_1,...,b_n$ likewise the error term regressed to order q & coefficients $\psi_1,...,\psi_n$, the series differenced for stationarity Y_t: $\nabla Y_t = Y_t - Y_{t-d}$, then $\nabla^d Y_t = (B)Y_t$ where $B = \nabla^d$ is a backshift operator that shows the differencing order, the ARIMA model is given by:

$$b(B)Y_t = \psi(B)e_t \tag{2}$$

The error is Gaussian with zero mean and finite variance σ^2 denoted by $\mathcal{W}(0, \sigma^2)$.

B. Nonlinear GARCH Model: Variance As Estimator

The GARCH model retains the form of the ARIMA model. The focus however shifts to errors which are squared. Equation (2) becomes $z_t = Y_t - X_t$ with $z_t = \sigma_t e_t$ where e_t is the same as the white noise earlier discussed and σ_t is the standard deviation with $\sigma_t^2 = a_0 + b_1 z_{t-1}^2 + \cdots + b_p z_{t-p}^2$, the generalization of the GARCH model is:

$$\sigma_t^2 = a_0 + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2 \tag{3}$$

The standard GARCH model also models white noise.

C. Conditional Score Models: Score As Estimator

The models discussed thus far are able to capture the dynamics of some types of cloud traffic observed in the analysis of the time-series selected for study. Recent research has made the observations of *extreme value distributions* in cloud traffic [6]. This same observation we have made in 5 of 8 time-series studied. Two of them (IIIB & IVB) are illustrated in Figure 3. Furthermore, recent research in econometrics presents a new modeling approach that provides adaptations to the extreme value distributions observed in cloud traffic more accurately than the linear and nonlinear models discussed. This is done by modeling the error component in terms of its score, the derivative of the log-likelihood of the observed distribution. This is elaborated more in the ensuing paragraphs.

Recent econometrics research [16],[17] applicable for modeling non-normal errors discovered that: (1) GARCH recursions conditional on past observations are impacted by outliers affecting forecast estimates developed. (2) Once Gaussian assumptions are dropped, the expression of volatility by employing variance may not be the best modeling choice. This is because of the observations of non-Gaussian distributions observed are expressed in terms of volatility specific to the identified distributions (e.g. lognormal, Weibull) which each have their specific expression of variance [16]. Thus the sample variance will not apply for the identified distributions.

Given this, dynamic models for time-varying parameters have been realized given the independent works of Harvey [16] and Creal et al [17]. In the new approach, modeling time-varying properties, both in the mean and the variance, of time series is described as Autoregressive Conditional Score (ACS) models. The score refers to the derivative of the maximum likelihood estimate of the probability density function describing the errors.

For the purpose of Maximum Likelihood Estimation (MLE), let Y_t as previously parameterized be conditioned according to a time-varying parameter f_t (for instance in the GARCH model $f_t = \sigma^2$). Furthermore, let θ_t be a parameter vector. For modeling, these are often restricted to the first and second order parameters of the mean and variance μ and σ^2 respectively. Thus we have $Y_t \sim p(Y_t|f_t,\theta_t)$ and writing Y_t as an autoregressive function of f_t: $f_t = \beta f_{t-1} + \alpha s_t$, the score s_t is the derivative of the log-likelihood given as:

$$s_t = \frac{\partial^2 \ln g(y_t|f_t;\theta)}{\partial^2 f_t} \tag{4}$$

Here, f_t is autoregressive on past terms while the noise is now replaced by the score, α & β are parameters to be estimated. Given that estimation is by MLE, if we assume a Gaussian distribution for Y_t, then $Y_t \sim p(Y_t|\mu,\sigma^2)$ where it is parameterized by the mean and the variance and log-likelihood of *Yt* is the log-likelihood of the normal distribution. Given the diverse collection studied, the lognormal distribution provided a fit for 4 of the 8 time-series studied. This has motivated the development of the ACS model with lognormal errors described as ACS-*l*. The likelihood of *Yt* is the log-likelihood of the *lognormal* distribution:

$$\text{log like}\left[\frac{1}{\sqrt{(2\pi\sigma^2)}}\exp\left[-\frac{(\ln Y-\mu)^2}{2\sigma^2}\right]\right] \tag{5}$$

The score of the lognormal distribution $s_t = \ln y^2 - \sigma^2$ and given that the time-varying parameter is the variance, $f_t = \sigma^2$, and substituting s_t and f_t : $f_t = \beta f_{t-1} + \alpha s_t$, it yields:

$$\sigma_t^2 = \beta\sigma_{t-1}^2 + \alpha(\ln y_{t-1}^2 - \sigma_{t-1}^2) \tag{6}$$

Equation (6) is the specification of the ACS-*l* model. It will be noticed that if the errors are assumed to be normal, then the standard GARCH model is realized from log-likelihood. In order to test the forecasting accuracy of the realized model, a forecasting algorithm was developed in MATLAB. The next section details the performance evaluation and comparison with existing methods.

IV. PERFORMACE EVALUATION

The model comes in two realizations. The first is an algorithm in MATLAB and the second is its integration into the R statistical computing package. Datasets that yield long time-series (1 month or greater) can leverage the parallel computation toolbox in MATLAB for model training by using multiple CPUs for processing. Several MATLAB APIs also exist to integrate the application realization of the forecasting algorithm to provide online data for prediction. The R implementation benefits from an extensive repository of existing forecasting applications. These can be combined for parallel or ensemble forecasting for better predictive results. Furthermore, testing is done according to: (1) In-sample forecasting and (2) Out-of-sample forecasting. In-sample forecast evaluation makes use of available time-series data in order to make current and future predictions. In this method, the model parameters are estimated using the time-series observations in order to make predictions. The prediction procedure employs a rolling-forecast. In this case, all available observations up to time T is employed to predict $T + 1$. This is conducted continuously for $T + 2$, $T + 3$, as required. The motivation for this is the minimization of the error in prediction as it continually makes use of the available observations both in modeling as well as in forecasting. In-sample forecasting is variously described as *point*, *one-step-ahead* and *rolling* forecasts.

Out-of-Sample forecast evaluation employs a subset of all available time-series observations for model fitting and forecasts over the withheld observations in order to validate the model while testing forecast accuracy. This method is employed in forecasting horizons $T + n$. Here, n is the number of unused observations over which the forecasting accuracy is tested. In order to compare the ACS-*l* model with existing methods, the measure selected is the Mean Absolute Percentage Error is calculated. This is given by:

$$MAPE = \frac{100}{n}\sum_{t=1}^{n}\left|\frac{y_t - \bar{y}_t}{y_t}\right| \tag{7}$$

Where y_t and $\bar{y}_t$ are the real and predicted observations of the time-series. The evaluation proceeds with in-sample forecast performance. Given space constraints, the performance evaluation is done with a few representative workloads for visual analysis while the MAPE is used for evaluation.

To examine the diversity of cloud application areas according to the methodology developed the series evaluated are categorized as linear (I,IVA), GARCH (II), Hybrid (IIIA, IIIB and IVB). Their evaluation follows.

A. In-Sample Forecast Evaluation

For in-sample forecasting, the algorithm was realized in MATLAB was employed. The forecasting methodology employed is captured in Figure 6 below. The workload serves as input to the model by which it is used for training. It is the observations of the time-series that determine the model parameters subsequent to forecasting. Once these are determined, prediction commences followed by the determination of predictive accuracy. This is one by an examination of the errors post prediction. The comparison begins by a performance evaluation of ARIMA representing existing methods with the ACS-*l* representing nonlinear/hybrid methods. To illustrate when ARIMA models are adequate, series IVA is evaluated with both the ARIMA and ACS-*l* models. Figure 7 illustrates the in-sample forecast for the ARIMA model while Figure 8 illustrates the same for ACS-*l*. The MAPE returned for ARIMA is 19% and 18% for ACS-*l* respectively. Given that error examination backed by statistical inference determined Gaussian noise, in this instance the linear ARIMA model is accurate enough for forecasting.

For the nonlinear GARCH, series II is evaluated and the comparison is done given two representative models: (1) ARIMA-GARCH which represents current methods employed for linear and nonlinear prediction and (2) the ACS-*l* for hybrid models. Each series underwent training as illustrated in Figure 6 before the subsequent prediction phase. Comparison is done with their returned MAPE values. Figure 9 illustrates the rolling 5-minute ahead forecast for the ARIMA-GARCH model while Figure 10 displays the same for the ACS-*l* model. To draw the distinction between employing variance to track volatility as occurs in most of current methods and the new proposed model, Figure 11a shows the score function over time for the same time-series compared with variance of the ARIMA-GARCH model. While the conditional variance is persistent throughout the time-series as illustrated in Figure 11b by the variation in amplitude of the measured variance observed over the x-axis, the method of the score is better able to track the fluctuations of the time-series as observed in Figure 11a. Another property of the score is that its forecast remains within the range of the original time-series compared to the ARIMA-GARCH forecast. To compare forecast accuracy, the calculated MAPE for ACS-*l* is 4.1 compared to 5.5 for ARIMA-GARCH which is a 25% improvement in accuracy.

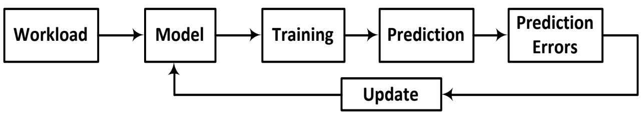

Figure 6: Forecasting Algorithm

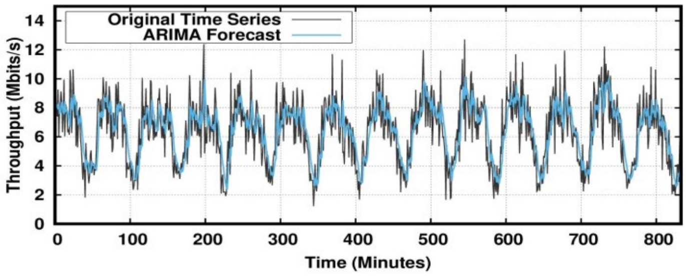

Figure 7: ARIMA In-Sample Forecast for Series IVA.

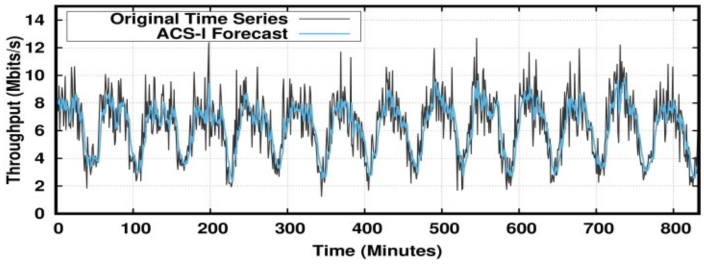

Figure 8: ACS-*l* In-Sample Forecast for Series IVA

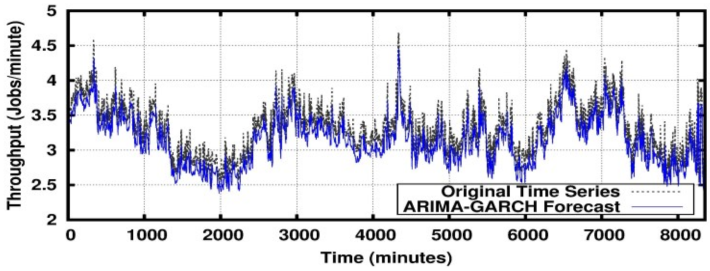

Figure 9: Forecast of Series II with ARIMA-GARCH

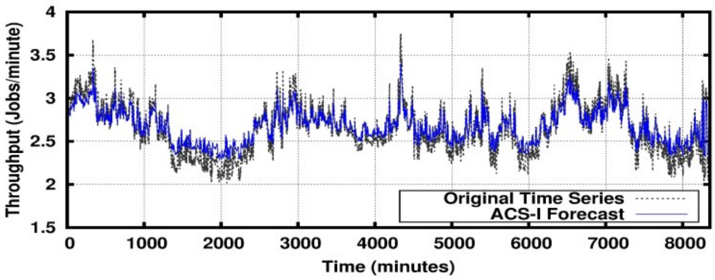

Figure 10: Forecast of Series II with ACS-*l*

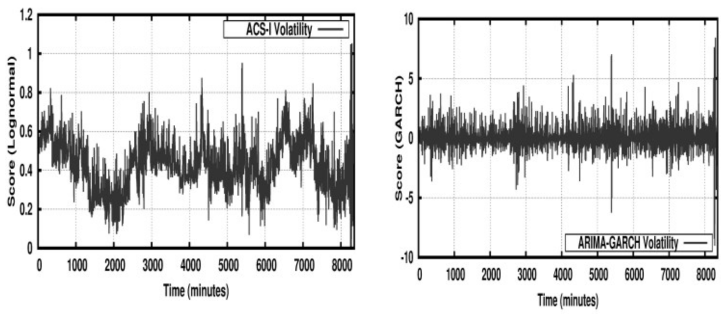

Figure 11: (a) ACS-*l*, Score & (b) ARIMA-GARCH, Variance

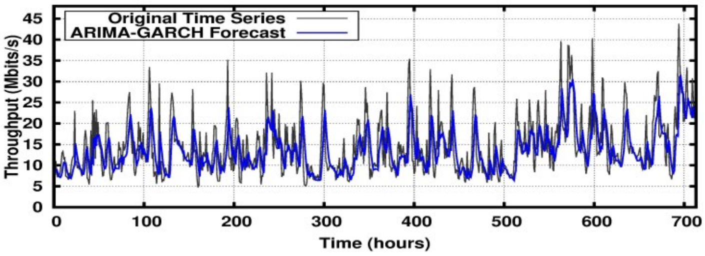

Figure 12: Series IIIA Forecast with ARIMA-GARCH

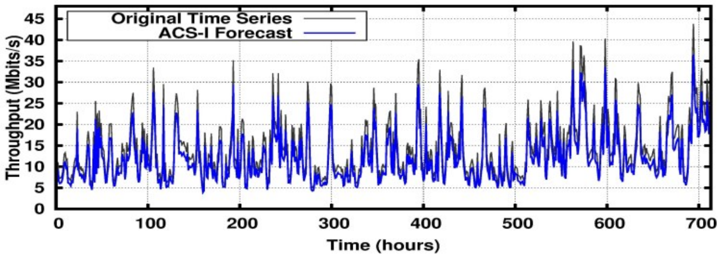

Figure 13: Series IIIA Forecast with ACS-*l*

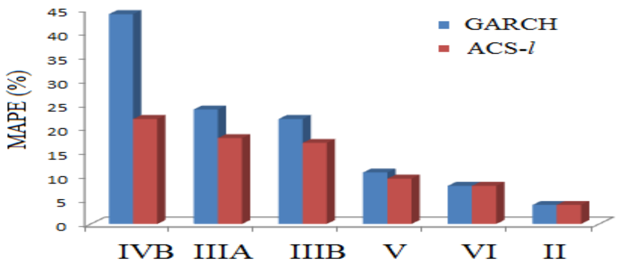

Figure 14: In-Sample Forecasting Accuracy for all Series

Series IIIA is characterized by a high-degree of burstiness. The forecast window for this series is much longer than for the Google compute cluster (1 hour). For this time-series, the Figures 12 and 13 illustrate the series IIIA forecast evaluated with the ARIMA-GARCH and the ACS-*l* models respectively. The upward trend noticeable from the 500th time-interval is accounted for by ARIMA given the moving average component that tracks mean value evolution while the GARCH component tracks the volatility. The ACS-*l* model demonstrates improvement on the GARCH recursion. In GARCH the calculation is conditional on the square of past errors where the ACS-*l* model employs the log of the dependent variable. The improvement in performance of the ACS-*l* model over ARIMA-GARCH is by 20%. Series VI was also compared but the forecast graphs are not illustrated due to space constraints.

Table 2 provides a comparison of all models for the time-series evaluated. The key contribution of the ACS-*l* model is the reduction in forecast errors when *right-tailed distributions* are statistically evident as the distribution for the time-series under study. This is when it provides a 10% - 25% reduction in forecast errors. The modeling accuracy for all three categories evaluated is given in Figure 14.

B. *Out-of-Sample Forecast Evaluation*

In the previous section, forecasts were made whereby all time-series observations up to and including time *T* were used

Table 2: MAPE Comparison for All Models

Series	MAPE (%)	
	ACS-l	ARIMA-GARCH
Linear Models: (ARIMA)		
I	28	30
IVA	18	19
Hybrid Models: (ACS-l)		
IIIA	18	25
IIIB	15	22
IVB	25	42
Nonlinear Models (GARCH)		
II	5	5
VI	8	11

to make predictions at time $T + 1$. The practice is to ensure low forecasting error by using as much information as possible continuously about the time-series to forecast its future. However, in order to conclusively validate realized models, only part of all the observations of a time-series are used in training the model and then forecasts are subsequently made. To this end, a subset of observations from each time-series selected for evaluation was withheld from the model fitting process with forecasting over the withheld observations used for validation. Furthermore, for evaluation, the model was realized as original C++ code which was integrated into the R statistical computing package. For training, the entire series except the last 60 observations were used. This makes for variable forecast windows according to the time-series under evaluation. For instance for series II (Google), this gives a 5-hour forecast horizon and for series IIIA, a two-day forecast horizon. This means resource provisioning can be planned over these forecast horizons as required. To show that the ARIMA forecast becomes inadequate especially for volatility prediction for series II (Google), a comparative forecast is illustrated in Figure 15. Given constant variance linear models are unable to accurately predict the series as shown. This is the out-of-sample forecast which begins 27 days into the time-series. For illustration, the two and half day out-of-sample forecast for series IIIA is given in Figure 16 and 17.

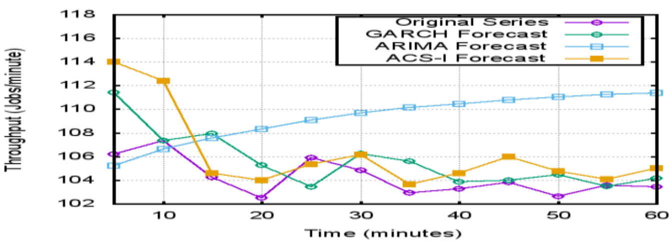

Figure 15: Out-of-Sample Forecast Comparison for Series II

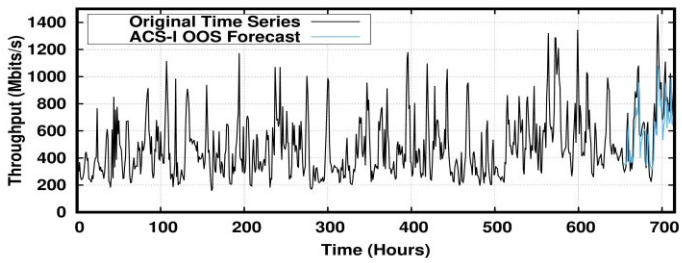

Figure 16: ACS-*l* Out-of-Sample Forecast for Series IIIA

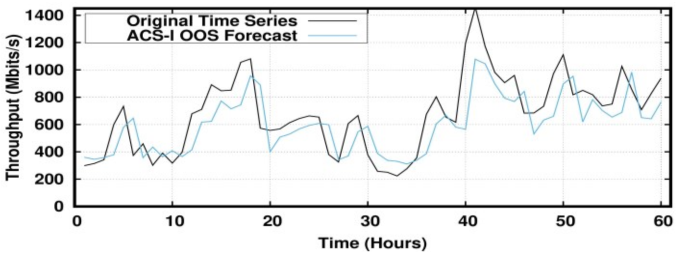

Figure 17: ACS-l Series III Out-of-Sample Forecast (Only)

C. Discussion of ACS-l Model.

The ACS-*l* model is able to capture the dynamics of time-series that exhibit right-skewed tails in their empirical distributions. It is easy to understand given that it is parameterized by the mean and the variance while tracking long-tailed distributions compared to others in the family like the Gamma and Weibull distributions. As illustrated, it is able to improve forecasting accuracy when compared with existing methods. However the realized model is not without its drawbacks. It has a tendency to over-fit time-series as observed for some of the time-series studied. In addition, given that it is right-tailed, all observations in a time-series must be positive or adjusted accordingly for the realization of a valid forecast. Given these drawbacks the forecasting algorithm is presented under two variants. The first which is prone to over-fitting employs the standard deviation of the random variable in its calculation of volatility as obtains in traditional methods. The variant that affords stability employs the practice of volatility adjustment as done for lognormal forecasting [18]. This realization employs the exponent of the random variable and it is able to minimize the observed over-fitting.

V. RELATED WORK

Prediction in the cloud has been largely reliant on the general class of linear ARIMA models. Cloud workload prediction based on ARIMA according to the research conducted by Calheiros, Masoumi, Ranjan & Buyya [19] was employed in Software-as-a-Service (SaaS) provider scenarios. Similar work by Han, Chan & Leckie [20] researched arrival and departure processes on the Amazon EC2 & Windows Azure platforms with the development of methods evaluated in different application scenarios. Similarly, the research conducted in [21] employed ARIMA models for resource usage predictions to minimize SLA violations & resource usage patterns and dynamic traffic in the cloud [22].

Beyond the linear methods, in [23], ARIMA was employed to track the mean value of cloud workloads while the GARCH model was employed to forecast trend and volatility. In [24], the GARCH model was combined with Artificial Neural Networks to predict future requests used in the attendant resource requirements. The GARCH model was combined with ARIMA in [25] for the optimization of cloud-assisted video distribution in content delivery networks. The utility of the GARCH model in predicting volatility in cloud video systems was realized as forecasting solutions in [26].

The forecasting methods discussed thus far belong to the category of classical linear and nonlinear methods. Nonlinear models that do not employ statistical parameters of time-series but are inspired by nature and Artificial Intelligence (AI) belong in this category. The class of Artificial Neural Network (ANN) time-series models has enjoyed adoption in forecasting cloud traffic. Xue et al [27] employed an ANN model for the realization of predictive solutions for CPU, memory and network bandwidth in IBM's cloud computing environments. Comparison was made with significant improvements over ARIMA methods. In [28], the predictive accuracy of cloud auto-scaling was investigated with an ANN solution realized with improvements in forecast error performance when compared with existing methods. In [29], an evolutionary neural network solution was realized to forecast and mitigate energy consumption in the cloud. Resource scheduling for increased optimization was the focus of research in [30] where average web response time was improved with an ANN solution.

The methods and models here proposed are based on a careful statistical analysis of diverse workloads. There are distinct properties by which to determine the model appropriate for forecasting. The models introduced combine linear and nonlinear components of existing methods in a novel manner and belong to a hybrid class of models applicable to specific traffic patterns in the cloud.

VI. CONCLUSION & FUTURE WORK

In this paper, we present a methodology that guides the selection of models for time-series realized from cloud metrics. This is based on statistical analysis of empirical distributions from the original cloud datasets. Furthermore, it develops a novel model also based on the same statistical observations for predicting various cloud metrics that can be employed in resource planning solutions. We embarked on a performance evaluation of the model and compared it with existing methods. The realized forecasting algorithm offers a 10%-25% improvement over existing methods. The drawbacks of the model have been identified also with efforts to mitigate its adverse impact on forecasting. Future work will explore failure prediction as occurs in environments like Google's compute cluster, VM consolidation planning in IaaS cloud environments and the QoS predictive solutions.

REFERENCES

[1] M. L. D. Vedova, D. Tessera, and M. C. Calzarossa, "Probabilistic provisioning and scheduling in uncertain Cloud environments," in *2016 IEEE Symposium on Computers and Communication (ISCC)*, 2016, pp. 797-803.

[2] J. Wen, L. Lu, G. Casale, and E. Smirni, "Less Can Be More: Micro-managing VMs in Amazon EC2," in *2015 IEEE 8th International Conference on Cloud Computing*, 2015, pp. 317-324.

[3] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*: Wiley, 2013.

[4] R. F. Engle, "Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation," *Econometrica,* vol. 50, pp. 987-1007, 1982.

[5] A. Zinnen and T. Engel, "Deadline constrained scheduling in hybrid clouds with Gaussian processes," in *High Performance Computing and Simulation (HPCS), 2011 International Conference on*, 2011, pp. 294-300.

[6] C. Z. Loboz, "Cloud resource usage: extreme distributions invalidating traditional capacity planning models," presented at the Proceedings of the 2nd international workshop on Scientific cloud computing, San Jose, California, USA, 2011.

[7] T. Benson, A. Akella, and D. A. Maltz, "Network traffic characteristics of data centers in the wild," presented at the Proceedings of the 10th annual conference on Internet measurement, Melbourne, Australia, 2010.

[8] S. Di, D. Kondo, and W. Cirne, "Characterization and Comparison of Cloud versus Grid Workloads," presented at the Proceedings of the 2012 IEEE International Conference on Cluster Computing, 2012.

[9] S. Siqi, V. van Beek, and A. Iosup, "Statistical Characterization of Business-Critical Workloads Hosted in Cloud Datacenters," in *Cluster, Cloud and Grid Computing (CCGrid), 2015 15th IEEE/ACM International Symposium on*, 2015, pp. 465-474.

[10] R. Yanggratoke, J. Ahmed, J. Ardelius, C. Flinta, A. Johnsson, D. Gillblad*, et al.*, "Predicting service metrics for cluster-based services using real-time analytics," in *Network and Service Management (CNSM), 2015 11th International Conference on*, 2015, pp. 135-143.

[11] R. Gracia-Tinedo, M. S. Artigas, A. Moreno-Mart, x00Ed, nez, C. Cotes*, et al.*, "Actively Measuring Personal Cloud Storage," in *2013 IEEE Sixth International Conference on Cloud Computing*, 2013, pp. 301-308.

[12] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*: OTexts, 2014.

[13] N. Tankovi, x, N. Bogunovi, x, T. G. Grbac, M*, et al.*, "Analyzing incoming workload in Cloud business services," in *Software, Telecommunications and Computer Networks (SoftCOM), 2015 23rd International Conference on*, 2015, pp. 300-304.

[14] X. Wang, Z. Lu, J. Wu, T. Zhao, and P. Hung, "In STechAH: An Autoscaling Scheme for Hadoop in the Private Cloud," in *Services Computing (SCC), 2015 IEEE International Conference on*, 2015, pp. 395-402.

[15] A. Ali-Eldin, M. Kihl, J. Tordsson, and E. Elmroth, "Analysis and characterization of a video-on-demand service workload," presented at the Proceedings of the 6th ACM Multimedia Systems Conference, Portland, Oregon, 2015.

[16] A. C. Harvey, *Dynamic Models for Volatility and Heavy Tails: With Applications to Financial and Economic Time Series*: Cambridge University Press, 2013.

[17] D. Creal, S. J. Koopman, and A. Lucas, "GENERALIZED AUTOREGRESSIVE SCORE MODELS WITH APPLICATIONS," *Journal of Applied Econometrics,* vol. 28, pp. 777-795, 2013.

[18] W. T. Shaw, *Modelling Financial Derivatives with MATHEMATICA* ®: Cambridge University Press, 1998.

[19] R. N. Calheiros, E. Masoumi, R. Ranjan, and R. Buyya, "Workload Prediction Using ARIMA Model and Its Impact on Cloud Applications’ QoS," *IEEE Transactions on Cloud Computing,* vol. 3, pp. 449-458, 2015.

[20] Y. Han, J. Chan, and C. Leckie, "Analysing Virtual Machine Usage in Cloud Computing," in *2013 IEEE Ninth World Congress on Services*, 2013, pp. 370-377.

[21] A. Nadjar, S. Abrishami, and H. Deldari, "Hierarchical VM scheduling to improve energy and performance efficiency in IaaS Cloud data centers," in *Computer and Knowledge Engineering (ICCKE), 2015 5th International Conference on*, 2015, pp. 131-136.

[22] S. Mistry, A. Bouguettaya, H. Dong, and A. K. Qin, "Predicting Dynamic Requests Behavior in Long-Term IaaS Service Composition," in *Web Services (ICWS), 2015 IEEE International Conference on*, 2015, pp. 49-56.

[23] W. Wei, W. Xuanzhong, C. Tao, G. Xiaofeng, and C. Guihai, "Dynamic correlative VM placement for quality-assured cloud service," in *Communications (ICC), 2013 IEEE International Conference on*, 2013, pp. 2573-2577.

[24] M. Barati and S. Sharifian, "A new hybrid model for request rate prediction in mobile cloud computing," in *2015 23rd Iranian Conference on Electrical Engineering*, 2015, pp. 775-780.

[25] J. He, D. Wu, Y. Zeng, X. Hei, and Y. Wen, "Toward Optimal Deployment of Cloud-Assisted Video Distribution Services," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 23, pp. 1717-1728, 2013.

[26] N. Di, X. Hong, L. Baochun, and Z. Shuqiao, "Quality-assured cloud bandwidth auto-scaling for video-on-demand applications," in *INFOCOM, 2012 Proceedings IEEE*, 2012, pp. 460-468.

[27] J. Xue, F. Yan, R. Birke, L. Y. Chen, T. Scherer, and E. Smirni, "PRACTISE: Robust prediction of data center time series," in *Network and Service Management (CNSM), 2015 11th International Conference on*, 2015, pp. 126-134.

[28] A. Y. Nikravesh, S. A. Ajila, and C. H. Lung, "Towards an Autonomic Auto-scaling Prediction System for Cloud Resource Provisioning," in *2015 IEEE/ACM 10th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, 2015, pp. 35-45.

[29] Y. W. Foo, C. Goh, H. C. Lim, Z. H. Zhan, and Y. Li, "Evolutionary Neural Network Based Energy Consumption Forecast for Cloud Computing," in *2015 International Conference on Cloud Computing Research and Innovation (ICCCRI)*, 2015, pp. 53-64.

[30] F. F. d. Almeida, A. d. A. Neto, and M. M. Teixeira, "Resource Scheduling in Web Servers in Cloud Computing Using Multiple Artificial Neural Networks," in *2015 Fourteenth Mexican International Conference on Artificial Intelligence (MICAI)*, 2015, pp. 188-193.