JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Host Overload Detection in Cloud Using Multiple Regression

S. Jyothsna, PhD Scholar, Osmania University , Hyderabad,

Dr.K.Radhika, Professor CBIT, Hyderabad

Abstract: **Cloud computing provides many advantages to the digitally connected world, so it has become a crucial component of enterprises today. cloud computing has helped enterprises in more than one way from getting rid of the heavy hardware to up-scaling and down-scaling whenever required, resulting in huge savings of the organisations. Artificial intelligence (AI) has the ability to further automate the vast possibilities of cloud computing. AI enables machines to study and learn from previous data, find patterns, and make decisions in real-time. Due to the fact that the majority of cloud services are real-time and load changes occur often, this can be used for automatic, dynamic overload detection in cloud servers.**

The majority of the effort to determine how busy cloud servers are is based only on CPU usage. For cloud servers, the dynamic load overload condition must be detected by taking into account memory usage, network usage, and CPU usage. The issue of SLA violation can be solved by improving response time. CloudSim uses a variety of statistical techniques for scheduling depending on static demand. supervised learning method, multiple regression is used to determine the load on cloud servers by looking at how CPU, memory, and the network are being used. Analyzed the data, patterns were identified and monitored the load of cloud servers for overload or underload.

Index Terms: **Cloud Computing, Host overload, AI, supervised learning..**

1. INTRODUCTION

Virtual Machine (VM) consolidation is a critical procedure in enhancing the usage of the resource in cloud computing services. VM consolidation involves host overload/under-load detection, VM selection and VM placement operations. The majority of host overload/under-load detection methods for VM consolidation currently in use only consider CPU utilisation when calculating host load. When relying on a single aim, there are issues with effective load condition prediction. In this study, host overload detection is carried out using three resources: CPU, memory, and bandwidth use.

Three general categories can be used to classify earlier methods of energy-efficient host overload detection: (1) static threshold based heuristics, (2) adaptive utilisation based heuristics, and (3) regression based heuristics.

Threshold-based approaches work by establishing a static CPU utilisation threshold that separates the node's non-overflow and overload modes. These methods compare the host's present CPU use to a predetermined threshold. In contexts where several types of applications may share a physical resource and workloads are dynamic and unpredictable, fixed utilization criteria are inappropriate.

Adaptive utilization-based techniques allow for the automatic adjustment of the utilization thresholds based on a statistical analysis of historical data computed over the course of the VMs' lifetimes.

These are effective in dynamic contexts but offer subpar host overload prediction. Therefore, estimating future CPU consumption may also be useful for host overload detection.

Because they rely on estimates of future CPU consumption, regression-based approaches offer a better forecast of host overloading. Despite being difficult, they might be worthwhile in the long run.

The structure of the essay is as follows. Regression models and supervised and unsupervised learning methods are covered in Section 2. Section 3 evaluates the relevant literature. Host Overload Detection based on Logistic Regression is covered in Section 4. (LRHOD). The technique of the evaluation is covered in Section 5. Section 6 provides the conclusion in the end.

2. Supervised and Unsupervised Learning

Supervised Learning

The learning of a model with input variables (x) and output variables (y) as well as a technique to map the input to the output is known as supervised learning.

$$y=f(x)$$

The primary goal is to sufficiently approximate the mapping function such that it is possible to anticipate the related output variable from new input data (x).

Because the process of learning from the prior load on the servers (training dataset) can be viewed as a supervisor who oversees the entire process, it can be referred to as supervised learning. The learning process therefore ends after the system reaches an acceptable level of accuracy, and the learning algorithm generates predictions on the usage data iteratively and is corrected by the supervisor. Server overload is successfully recognized in terms of performance. The past usage data used here is referred to as training data, and it is from this data that the algorithm learns new things. It has a response variable called "y" that acts as a decision variable to determine cloud server overload.

Unsupervised Learning

When there is only the input data (x) and no corresponding output variable, this is known as unsupervised learning. Unsupervised learning's primary goal is to model the data's distribution in order to understand it better. It is so named because there is neither a right or a wrong supervisor. Algorithms are given free rein to find and display any intriguing data structures or patterns.

As more classes are known beforehand, supervised learning has a higher computational complexity than unsupervised learning and produces more accurate and consistent outcomes. Support vector machines, decision trees, logistic and linear regression, and multi-class classification are examples of supervised machine learning techniques.

Regression

Regression is the process of using the relationship between variables to identify the regression equation or best fit line that can be used to generate predictions. The issue arises when the output variable has a real or continuous value, like weight or salary. The output variable in this case is server load. which examines the interaction between an independent variable (or variables) and a dependent variable (target) (predictor). Forecasting, time series modelling, and determining the causal relationship between the variables are all done using this method.

Types of regression

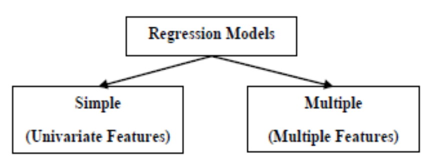


Figure 1. Regression Models

Regression models are of following two types −

Simple regression model − This is the most basic regression model in which predictions are formed from a single, uni variate feature of the data.

Multiple regression model − As name implies, in this regression model the predictions are formed from multiple features of the data.

Linear Regresssion

A relationship between a dependent variable (Y) and one or more independent variables (X) using a best-fit straight line is called a linear regression. The equation Y=a+b*X + e, where an is the intercept, b is the line's slope, and e is the error term, is used to express it. Based on the provided predictor variable, this equation can be used to predict the value of the target variable (s). When there are several independent variables, we can choose the most important independent variables by using a forward selection, backward elimination, or step-by-step technique.

A fundamental and often employed form of predictive analysis called linear regression typically uses continuous data.

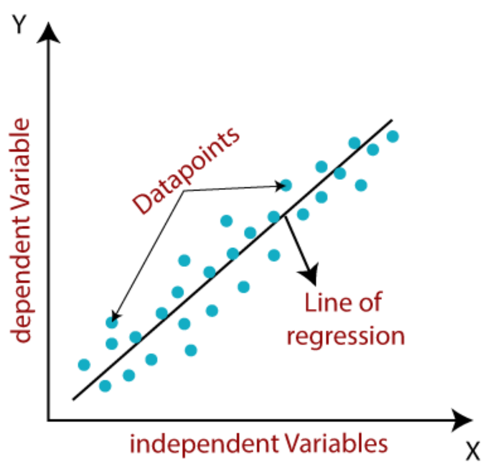


Fig 1. Linear Regression

Multiple Regression

Multiple regression is a technique used when the dependent variable has more than one class. The multiple regression model can be expressed as follows if we define p as the probability that the result is 1.

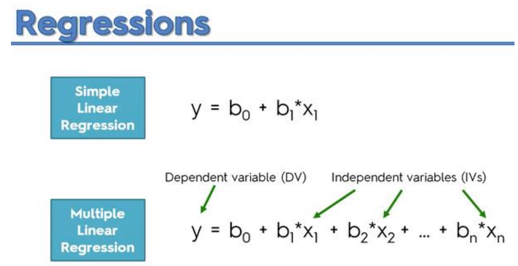
$$p = \frac{\exp(y)}{1+\exp(y)} \text{----------1}$$

where $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$

p is the expected probability that the outcome is present; x_1 through x_k are distinct independent variables and β_0 through β_k are the regression coefficients.

In the current work, y is dependent variable i.e. cloud server is overloaded or not by using independent variables x_0, x_1, x_2 as CPU utilization, RAM utilization and Network utilization respectively.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3$$



Y= b0+b1*utilization$_{CPU}$+b2* utilization $_{RAM}$ +b3* utilization $_{BW}$

4. Related Work

Multiple Regression Multi-Objective Seven-Spot Ladybird Optimization (MR-MOSLO) is an adaptive regression-based methodology suggested by the authors [1]. The host overload detection is carried out using the suggested MR-MOSLO method in two steps.

1. Multiple regressions are used in the initial step to estimate the host's anticipated use.

2. The MOSLO algorithm best identifies the upper and lower criteria for expected usage in the second stage.

The threshold selection problem can be resolved using MOSLO, which is based on the foraging strategies of seven-spot ladybirds. By calculating the utilization in 'n' runs and operating the cloud host iteratively, the MOSLO algorithm sets the threshold levels as efficiently as possible.

The algorithm compares the RAM and BW numbers to arrange these utilization values, with CPU receiving higher precedence. Therefore, the host utilization values are arranged in descending order using the CPU utilization as the main sorting criterion. The highest ranking utilization value, except 1 or 100%, is designated as Thupper, and the lowest ranking utilization value, excluding 0, is designated as Throw. By contrasting these thresholds with the anticipated consumption from MR, they are used in the host overload detection process.

$$predictedutilization = b0 + (b1 \times CPUutilization) + (b2 \times RAMutilization) + (b3 \times BWutilization)$$

The effectiveness of this method is evaluated in comparison to the host overload detection algorithms already in use, including THR, IQR, MAD, LR, LRR, MMSD, MRHOD, and HLRHOD. Total energy consumption, SLA, SLATAH, PDM, SLAV, and ESV characteristics are the comparison criteria. The findings demonstrated that MR-MOSLO offers comparably higher performance in host overload detection with high accuracy, nearly comparable energy and SLA values, and comparably lower SLATAH, PDM, SLA violations, and ESV values. It also offers superior QoS performance.

In [2], authors hypothesized that the number of virtual machines, resource requests for each virtual machine, virtual machine migration, the memory size of the migrated virtual machine, network flow, and processing power of the present server were all related to the load status of servers. The server's load condition was non-linear, abrupt, and somewhat periodic. Therefore, the Gauss kernel was investigated as a kernel function in the process of solving support vector regression (SVR GA).

For the purpose of detecting host overload, the authors of [3] suggested the Multi-Dimensional Regression Host Utilization method (MDRHU). Multiple regression is used to forecast future host utilization values; the regressor needs two key elements. The profiled data for the independent elements (dimensions) of the running VMs that go into the evaluation of host utilization makes up the first part. Secondly, and most The function to put the independent components together to generate a dependent metric indicating the total host utilization is a crucial component required to execute the multiple regression method.

Host utilization= $\frac{CPU+RAM+BW}{\text{normCost}}$

The utilization of the CPU, RAM, and BW are measured using various scales. Therefore, merely adding these variables together will not yield the host usage. The aim is divided by a normalization constant to get around this problem (normConst)

Two alternative models considering the space distance (the multi-dimensionality of the problem) for the host utilization are proposed as discussed, namely, Euclidean Distance (ED) and Absolute Summation (AS).

MDRHU-ED uses the Euclidean distance between the current and previous host utilizations as the normalization constant. In particular, the normalization constant used by MDRHU-ED

normConstED
$=\sqrt{d(CPU)^2 + d(RAM)^2 + d(BW)^2}$

In particular, the normalization constant used by MDRHU-AS is

normConstAS= |d(CPU)|+|d(RAM)|+|d(BW)|

where d(CPU), d(RAM), and d(BW) stand for the respective relative differences between the present and past CPU, memory, and BW utilizations.

The results shows that space distance based multiple regression algorithms (MDRHU- D and MDRHU-AS) provide superior improvement in terms of energy usage than Geometric Relation (GR) based multiple regression algorithm. where d(CPU), d(RAM), and d(BW) stand for the respective relative differences between the present and historical CPU, memory, and BW utilizations (MRHOD). MDRHU-ED performs best when there are 898, 1463, and 1516 VMs, while MDRHU-AS performs best when there are 1033 and 1233 VMs.

These policies offer auto-adjustment of the utilization thresholds based on a statistical analysis of historical data obtained during the lifetime of the VMs.

In [4], writers put forth a Markov chain model that takes into account both the use of resources now and in the future. The first-order Markov chain model is used to construct a Markov host prediction model that forecasts future use.

The Median Absolute Deviation Markov Chain Host Detection technique (MadMCHD), which is based on statistical analysis of historical data gathered during the lifespan of VMs, is used to set upper and lower CPU utilisation criteria initially.

The definition of the upper CPU usage threshold (Tu) is

$$T_u=1-S*MAD$$

The MAD represents the median of the absolute values of residuals (differences) from the median of the data. The method's safety can be adjusted using the parameter S. Higher tolerance for variance in

CPU use is achieved by setting the value of S to a lower value. They contrasted the state-of-the-art five host detection algorithms—IQR (inter quartile range), MAD (mean absolute deviation), LRR (local robust regression), and LR—with the proposed technique.

Type	**Static utilization threshold based algorithms**	**Adaptive utilization based algorithms**	**Regression based algorithms**
Explanation	Based on fixed CPU utilization threshold	Based on statistical analysis of historical data of VM	Based on estimation of future CPU utilization
Pros	Simple	Suitable for dynamic environment (robust)	Better predictions of host overloading
Cons	Unsuitable for dynamic environment	Poor prediction of host overloading	Complex
Examples	THR (Averaging threshold-based algorithm)	MAD (Median Absolute Deviation) , IQR (Inter Quartile Range)	LR (Local Regression) , LRR (Robust local Regression)

The PDM (Performance degradation due to migration) metric is decreased by 71.02%, 72.11%, 58.52%, and 79.05% for the actual workload by the suggested host load detection algorithm.

A heuristic for dynamic consolidation of VMs was presented by Beloglazov and Buyya [7] and is based on an examination of previous data from VM resource utilisation. Statistical approaches are used to determine the highest CPU utilisation level (Median absolute deviation and Inter quartile range) are used. Also Regression based algorithms LR and LRR that are based on estimation of future CPU utilization are used. These approaches do not consider hybrid parameters for host utilization calculation on contrary they depend on CPU only.

The following table gives an analysis of different types of overload detection

5. Multiple Regression Host Overload Detection (MRHOD) Algorithm

The goal of multiple regression is to create a model that relates a dependent variable (y) to numerous independent variables.

Multiple regression analysis enables explicit control for a wide range of additional factors that simultaneously affect the dependent variable. Modeling the relationship between a dependent variable and one or more independent variables is the goal of regression analysis. Let k be the number of variables, represented by the symbols x1, x2, x3,..., xk. When the values of x are known, such an equation can be used to forecast the value of y.

To determine whether the host is overloaded or not, and hence the possibility of the event occurring.

x_1=CPU utilization

x_2=RAM utilization

x_3=BW utilization

x_1, x_2, x_3 are calculated as the sum of utilizations of CPU,RAM and Bandwidth of all VMs of Host respectively i.e. to compute the total utilization , compute the individual utilization of each VM using the procedure below.

$$y = b_0 + b_1x_1 + b_2x_2 + b_3x_3$$

As the number of predictor variables increases, the b constants also increase correspondingly. The coefficient of determination (R-squared) is a statistical metric that is used to measure how much of the variation in outcome can be explained by the variation in the independent variables. R^2 always increases as more predictors are added to the MLR model, even though the predictors may not be related to the outcome variable.

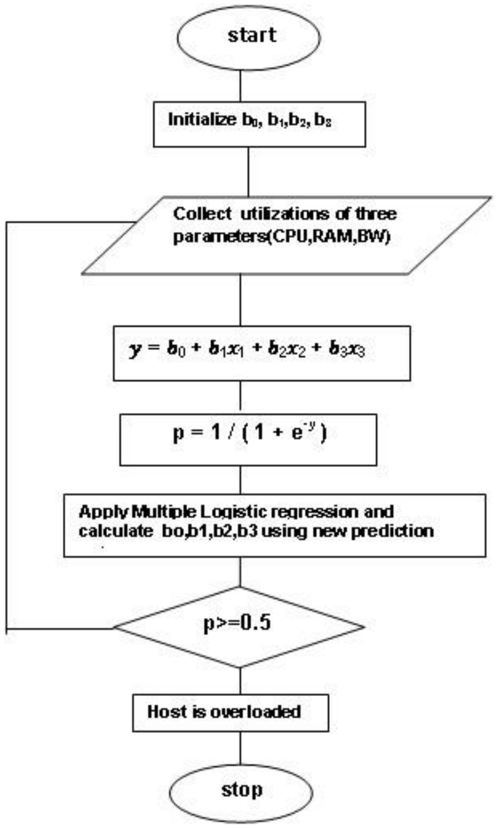


Figure 2. Host overload detection flowchart

Here x_1 is CPU utilization, x_2 is RAM utilization and x_3 is bandwidth utilization.

y=host overloaded(1) or not(0)

The use of the CPU,RAM and BW of each PM is calculated as the average use of all VMs in the PM by the maximum use of the PM.

$\ln[Y/(1-Y)]=b_0+b_1*\text{utilization}_{CPU}+b_2*\text{utilization}_{RAM}+b_3*\text{utilization}_{BW}$

we find the slopes (b_1, b_2,b3 etc.) and intercept (b_0) of the best-fitting equation in a multiple logistic regression using the maximum-likelihood method, rather than the least-squares method used for multiple linear regression.

Algorithm 1: MLRHOD

1. **INPUT:** Host in cloud
2. **OUTPUT:** Host Status
3. Initialize b1=0, b2=0, b3=0,x=1.0 and alpha=0.
4. n=Number of VM's of Host
5. repeat
6. $x1 = \sum_{k=0}^{n} \binom{n}{k}$ CPU utilization
7. x2 = $\sum_{k=0}^{n} \binom{n}{k}$ RAM utilization
8. $x3 = \sum_{k=0}^{n} \binom{n}{k}$ BW utilization
9. $y = b_0 + b_1x_1 + b_2x_2 + b_3x_3$
10. p= 1 / (1 + e^{-y})
11. $b_0 = b_0$ + alpha * (y – p) * p * (1 – p) * x
12. b_1 = b1 + alpha * (y – p) * p * (1 – p) * x_1
13. $b_2 = b_2$ + alpha * (y – p) * p * (1 – p) * x_2
14. $b_3 = b_3$ + alpha * (y – p) * p * (1 – p) * x_3
15. go to 10
16. IF (p >= 0.5) THEN
17. host Status <-Overloaded
18. ELSE
19. host Status<- not Overloaded
20. ENDIF
21. RETURN host Status
22. END

Where b_1,b_2,b_3 are the coefficients we are updating and p is the output of making a prediction using the algorithm. Alpha is parameter specified at the beginning as 0.3. This is the learning rate and controls how much the coefficients (and therefore the model) changes or learns each time it is updated. This process is repeated for regular intervals and update the model for each training instance in the dataset.

6. Experimental Results and Discussion

The pseudocode for the Multiple Regression Host Overload Detection (MRHOD) algorithm is presented in Algorithm 1. First, the host CPU utilization, RAM utilization and BW utilization for each host are calculated as the total utilization of the VMs in a host. that captures the combined CPU-network-memory load of virtual and physical servers. The multiple regression model is applied on collected data to find the slopes and intercept.

The performance of the proposed multiple regression host overload detection algorithm (MRHOD) for host load detection is evaluated on the utilization data collected from Amazon web services (AWS) EC2 instances. In particular, The proposed algorithm MRHOD is tested with three factors CPU, Memory and Network bandwidth using multiple regression model. The results shows overload detection is predicted in a better way compared to single factor like only CPU utilization.

Multiple Regression test output

Multiple regression significance test is performed using Anova ,data analysis tool in Excel . R-square is a statistical measure of how close the data are to the fitted to regression line. In our experiment adjusted R square is 0.91 which means the model greatly fits the data.It is clear that the p value of CPU and Memory is smaller than 0.05 so CPU and Memory values are significant. While the p value for Network is larger than 0.05 so it is insignificant.

Since the results obtained from hybrid factors is better than the results from single factor so the VM behavior is rarely a function in one variable, it should be a function of multiple factors. Regression based algorithms outperform the threshold-based and adaptive-threshold based algorithms using hybrid factors as well as using single factors.

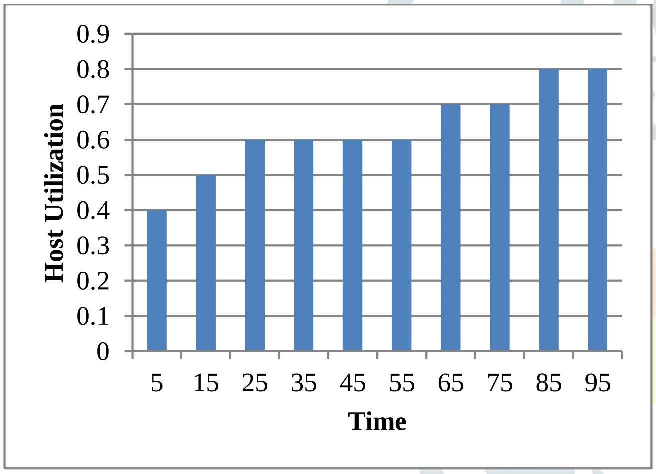


Figure 4. Overload Prediction

Figure 4 depicts that if the host utilization is greater than or equal to 50% then the host is determined as overloaded.

5. Conclusion

The proposed multiple regression host overload detection(MRHOD) algorithm is detecting overload condition based on CPU, Memory, Bandwidth so it gives a real indication of host utilization. The significance of the independent factors (CPU utilization, RAM utilization, BW utilization) are tested on the dependent factor (host utilization) using multiple regression because the performance does not rely only on CPU utilization. For applications that require communication among services, the communication cost can also influence the overall performance. Furthermore, there are applications require a huge amount of memory hence memory utilization can also influence the overall performance. So the CPU, RAM and BW are very important parameters for detecting server load.

References

1. Akram Saeed Aqlan Alhammadi1* Vasanthi Varadharajan2,"MR-MOSLO:VM Consolidation Using Multiple Regression Multi-Objective Seven-Spot Ladybird Optimization for Host Overload Detection" International Journal of Intelligent Engineering and Systems, Vol.13, No.2, 2020 DOI: 10.22266/ijies2020.0430.03

2. Xin Sui1,2, Dan Liu1 , Li Li1*, Huan Wang1 and Hongwei Yang1,"Virtual machine scheduling strategy based on machine learning algorithms for load balancing" Sui et al. EURASIP Journal on Wireless Communications and Networking (2019) 2019:160 https://doi.org/10.1186/s13638-019-1454-9

3. Ali A. El-Moursy , Amany Abdelsamea , Rukshanda Kamran and Mohamed Saad "Multi-Dimensional Regression Host Utilization algorithm (MDRHU) for Host Overload Detection in Cloud Computing" El-Moursy et al. Journal of Cloud Computing: Advances, Systems and Applications (2019) 8:8 https://doi.org/10.1186/s13677-019-0130-2

4. SUHIB BANI MELHEM 1 , (Member, IEEE), ANJALI AGARWAL1 , (Senior Member, IEEE), NISHITH GOEL2 , AND MARZIA ZAMAN2 "Markov Prediction Model for Host Load Detection and VM Placement in Live Migration" Digital Object Identifier 10.1109/ACCESS.2017.2785280

5. Ali El-Moursy, "virtual machine consolidation using Hybrid Regression Algorithms" ,Article in Egyptian Informatics Journal · January 2017 DOI: 10.1016/j.eij.2016.12.002

6. Sayed M. A. H. Monil and R. M. Rahman, "VM consolidation approach based on heuristics, fuzzy logic, and migration control," Journal of Cloud Computing, Vol.5, No.1, p.8, 2016.
7. A. Beloglazov and R. Buyya, ''Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers,'' Concurrency Comput., Pract. Exper., vol. 24, no. 13, pp. 1397–1420, Sep. 2012.
8. "Linear and Logistic Regression Based Monitoring for Resource Management in Cloud "

9. Ali El-Moursy,Elsayed Hemayed,Hesham Eldeeb,"Virtual machine consolidation enhancement using hybrid regression algorithms"

10. Akram Saeed Aqlan Alhammadi, Dr. V. Vasanthi "Multiple Regression Particle Swarm Optimization for Host Overload and Under-Load Detection", The Mattingley Publishing Co., Inc.,Volume 82, 10253 – 10261,January-February 2020

11. http://www.real-statistics.com/multiple-regression/multiple-regression-analysis/multiple-regression-analysis-excel/

12. https://docs.aws.amazon.com/cli/latest/reference/cloudwatch/get-metric-statistics.html