Analysis of migration time of live migration of VM's running benchmarks from a subset of SPEC CPU2006 benchmark suite

Subramanya Joshi
Supervisor: M.Sc. Kateryna Rybina
TU Dresden

30.04.2015

Abstract

Live migration [1] is a process of transferring the execution of virtual machines (VM's) from one host to another without stoping the execution of the VM or any workload running inside the VM. Migration of the VM can be used to consolidate the workload on a small set of host servers, allowing for rapid elasticity without compromising the availability. In the process of live migration the user workload running inside the VM's will not be interrupted, hence making the process transparent to the end-user. This process of VM migration incurs an overhead cost on the source and destination host servers. Such costs are migration time, increases time of execution of user tasks, power consumption etc [2]. In this Internship, I have carried out a study on the impact of various parameters like page dirty rate, last level cache misses, RAM utilisation, network bandwidth and Memory access rate in ten SPEC2006 benchmarks on migration time [3].

Contents

1 Assigned Task **1**

2 Introduction **2**

3 Concept suggested by the supervisor **2**

4 Workload and Benchmark **2**

5 Experiment Setup **4**

6 Results **4**
6.1 Multiple Linear Regression Model for Migration Time 5

7 Evaluation **6**

8 Conclusion and Future Work **7**

1 Assigned Task

1. Get familiar with the state-of-the-art papers concerning the live migration of the VM's and modelling the VM migration time.

2. Select 10 sub-benchmarks from SPEC CPU2006 benchmark suite (5 CPU intensive and 5 memory intensive), run them as the workload on the VM's during live VM migration.

3. Realise the VM migration varying the available for migration bandwidth from 70 MBps to 100 MBps in steps of 10 MBps.

4. Estimate the execution time of benchmarks with and without the live migration of the VMs. Quantify the service degradation.

5. As suggested by the supervisor test the concept, namely how the migration time of the VM's depends on the system parameters such as: 1) memory page dirty rate, 2) last level cache miss rate 3) cpu utilisation, 4) RAM utilisation, 5) available for migration network bandwidth etc.

6. Model these dependencies using multiple linear regression technique.

7. Include only significant parameters in the model which can be used to predict migration time.

2 Introduction

Server virtulization technology has recently emerged as essence of data centres and cloud computing systems, mainly due to its capabilities of isolating, consolidating and migrating workload. Altogether, these features allow a data centre to serve multiple users in a secure, flexible and efficient way. Consequently, these virtualized infrastructures are considered as a key component to drive the emerging Cloud Computing paradigm [4].
Migration of the VM's can be used to consolidate the workload on a small set of host servers, allowing for rapid elasticity without compromising the availability. The ability to migrate an entire operating system overcomes most difficulties that traditionally have made process-level migration a complex operation. The applications themselves and their corresponding processes do not need to be aware that a migration is occurring. Hypervisors, such KVM, allow migrating an OS as it continues to run. Such procedure is termed as live or hot migration, as opposed to pure stop-and-copy or cold migration, which involves halting the VM, copying all its memory pages to the destination host and then restarting the new VM. The main advantage of live migration is the possibility to migrate an OS with near-zero downtime, an important feature when live services are being served.[5]

3 Concept suggested by the supervisor

Process of live VM migration increases the resource consumption on both source and destination machines. This process of live migration of VM has significant influence on CPU utilisation, CPU cache hit-miss, memory utilisation , network bandwidth utilisation and power consumption of the hosts. As suggested by my supervisor, I carried out the experiment by creating the VM with SPEC2006 sub benchmarks executing within the VM (List of subset of benchmark considered at listed in Table-1) and recording the various parameters (List of various parameters recorded are listed in Table-2) of resource consumption during the process of live migration of the VM. In the following section I will describe the details of the Workload , Experiment setup, dataset acquired from the experiments and linear regression models which could be used to explain the dataset.

4 Workload and Benchmark

In this experiment I illustrate the influencing of various parameter like - page dirty rate, last level cache misses, RAM utilisation, network bandwidth and Memory access rate on determining the VM migration time which is running the workloads from a subset of SPECCPU2006 benchmark. Based on the study by [6], which determine the memory requirements of workloads from the SPEC CPU2000 and SPEC CPU2006 benchmark suites. Based on this study [6], choosing ten sub-benchmarks from SPECCPU2006 benchmark suit based on the benchmarks memory and CPU utilisation. These ten sub-benchmarks are divided into CPU Intensive and Memory Intensive benchmarks. A benchmark is CPU Intensive if it has highest Clockticks per Instructions Retired (CPI) [1] in SPECCPU2006 benchmark suit. A benchmark is Memory Intensive if it has highest CPI as well as higher memory read's-write's. Table-1 shows the list of Benchmarks and its characterisation.

Table 1: Subset of SPEC2006 benchmark considered for this expirement

Benchmark Name	Characterisation	Summary (take from [7])	
zeusmp	CPU Intensive	A computational fluid dynamics code developed at the Laboratory for Computational Astrophysics (NCSA, University of Illinois at Urbana- Champaign) for the simulation of astrophysical phenomena.	Floating Point Benchmark.
gromacs	CPU Intensive	A versatile package that performs molecular dynamics.	Floating Point Benchmark
namd	CPU Intensive	A parallel program for the simulation of large biomolecular systems.	Floating Point Benchmark
sphinx3	CPU Intensive	A speech recognition system from Carnegie Mellon University.	Floating Point Benchmark
soplex	CPU Intensive	SoPlex solves a linear program using the Simplex algorithm.	Floating Point Benchmark
mcf	Memory Intensive	A program used for single-depot vehicle scheduling in public mass transportation.	Integer Benchmark
GemsFDTD	Memory Intensive	solves the Maxwell equations in 3D in the time domain using the finite-difference time-domain (FDTD) method.	Floating Point Benchmark
omnetpp	Memory Intensive	Simulation of a large Ethernet network, based on the OMNeT++ discrete event simulation system, using an ethernet model which is publicly available .	Integer Benchmark
astar	Memory Intensive	2D path-finding library that is used in game's artificial intelligence.	Integer Benchmark
milc	Memory Intensive	Simulations of four dimensional lattice gauge theory on multiple instruction - multiple data parallel systems.	Floating Point Benchmark

[1] https://software.intel.com/sites/products/documentation/doclib/iss/2013/amplifier/lin/ug_docs/GUID-5C38FB45-A3ED-41F2-B57C-2B513A666205.htm

5 Experiment Setup

The experiment is based on the live migration of virtual machine, which is running a subset of SPECCPU2006 benchmark, from the source physical machine to the destination physical machine using network attached storage (NAS), and the client machine to trigger migration as shown in Figure 1.Two physical machines under test are homogeneous with system configuration of: Intel 15- 680 Dual Core 3.6 GHz processors, 4 GB DDR3- 1333 SDRAM, and with 1 Gbit/s Ethernet NIC. They are interconnected via a 1 Gbit/s switch. The NAS on which the VM images are located has the following characteristics: Intel Xeon E5620 Quad-Core 2.4 GHz processor, 10 GB DDR3-1333 SDRAM memory, and 1 Gbit/s Ethernet NIC. NAS is always accessed by the source and the destination hosts.
Ten sub benchmarks from SPECCPU2006 benchmark suit are executed one after another on the virtual machines. While each benchmark is being executed, VM is migrated from the source to destination server at different network bandwidth like 70 MBps, 80 MBps, 90 MBps and 100 MBps. When the benchmarks are being executing, based on their execution time, a total of ten migrations were performed between the source and destination hosts.
During the process of migration various tools are used to measure the system wide resources utilisation. dstat [2] program on the source and destination servers, as well as VMs was used to observe and record resource utilisation (CPU and memory), Intel Performance counter monitor [8] program was used to monitor and record the CPU data like - L1,L2 ,L3 hit-miss, CPU temperature, Instructions retired etc. And Page dirty statistics were recorded from Linux Kernal for Source , Destination and VM machines. Before the experiments took place all the servers were time synchronised in order to accurately determine the beginning and the end of a VM migration. Besides the resource utilisation and migration times the power consumption of the two servers was also recorded.

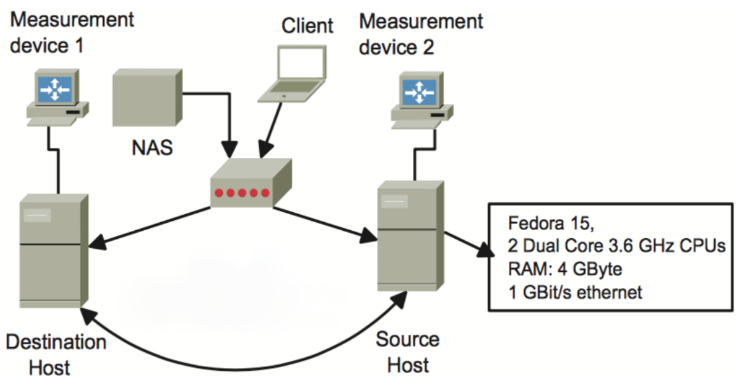

Figure 1: Experimental setup for live migration of VM [9]

6 Results

Overall, our experimental results show that overhead due to live migration is acceptable but cannot be disregarded. Figure 2 shows the total execution time taken for each benchmark. The figures in the Figure 1 show, time taken for the execution with and without VM migration during the process of execution of the benchmark. Figure 3 shows the time taken for each migration under different bandwidth capacity.

[2]http://linux.die.net/man/1/dstat

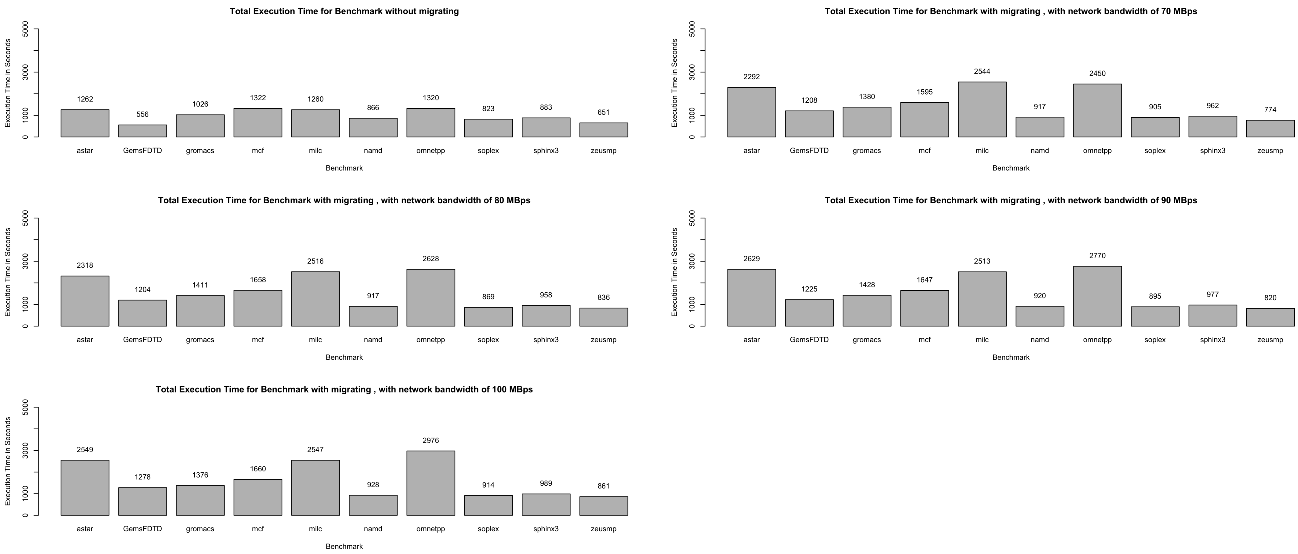

Figure 2: Time taken for execution of benchmarks with and without migration

6.1 Multiple Linear Regression Model for Migration Time

Many variables observed in this experiments are related. The type of their relation can often be expressed in a mathematical form called Regression. Establishing and testing such a relation enables us :

1. To understand interactions, causes and effects among variables.
2. To predict unobserved variables on the basis of observed ones.
3. To determine which variables significantly affect the variable of interest.

Linear Regression model assumes that the conditional expectation

$$Y = G(x) = \beta_0 + \beta_1 x \tag{1}$$

is a linear function of x. As any linear function , it has an intercept β_0 and a slope of β_1. A Multiple Linear Regression model assumes that the conditional expectation of a response: [10]

$$Y = G(x) = \beta_0 + \beta_1 x^{(1)} + \beta_2 x^{(2)} + ... + \beta_k x^{(k)} [10] \tag{2}$$

is a linear function of predictors $x^{(1)}, x^{(2)}, \ldots, x^{(k)}$. The intercept β_0 is the expected response when all the predictors equal to zero. The regression slope β_j is the expected change of the response Y when the corresponding predictor $X^{(j)}$ changes by 1 while all the other predictor remain constant. Essentially in this experiment we collect a sample of n units and measured all k predictors on each unit such as Total L3 Miss, Mean CPU utilisation, Total Page Dirty, Memory Access Rate, Total Instructions etc. on each unit. Also, the measured response variable Migration Time for each number of migration was recorded. The estimation of $\beta_1, \beta_2, \ldots, \beta_k$ was realised using the method of least squares. [10]

Table-2 list all the variables considered for the multiple linear regression model. For this experiment, we have considered four models with different parameters from host machine and the VM. In this experiment, two sets of experiments were conducted, one for CPU intensive benchmarks and one for Memory intensive benchmarks. The dataset collected in the experiment was divided into two types, one for training and one for testing. Training set include all the migration time for the bandwidths 70, 80 and 90 Mbps. With the exclusion of the benchmarks zeusmp , GemsFDTD and milc in training set, due to the fact that the first migration took the entire execution time and for the rest of the nine migrations the VM was not running any load. Test set include all the migration time for the bandwidth 100 Mbps and for all benchmarks. Based on the Multiple

Linear Regression Models, we predict the migration time for the Test dataset. Figure 4 (Model1) shows the regression model for the CPU intensive benchmark dataset without any parameters from the VM. Figure 5(Model2) shows the regression model for the Memory intensive benchmark dataset without any parameters from the VM. Figure 6(Mode3) and Figure 7(Model4) shows the regression model for the CPU intensive benchmark and Memory intensive benchmark datasets with total page dirtied from the VM. Considering Model1 for instance, the regression coefficient for TotalL3Miss and TotalINST is 1.05×10^{-7} and 1.6×10^{-4} respectively, suggesting that an increase of 1 percent in TotalL3Miss and TotalINST is associated with a 1.05×10^{-7} and 1.6×10^{-4} percent increase in the Migration time, controlling for TotalINST, MAR and other parameters. Note that, for the CPU bound benchmarks the TotalPageDirty is also significant which is showed by $p = 0.00504$. On the other hand, since this is model relates to CPU bound benchmark, the MAR is less significant and its not linearly related to Migration. Taken together the predictor variables account for around 99 percent of the variance in migration time for all the CPU intensive benchmarks. Figure 5 shows Model2, here the regression coefficient for TotalPageDirty is 7.3×10^{-1}, suggesting that an increase of 1 percent in TotalPageDirty is associated with a 7.3×10^{-1} percent increase in the Migration time, controlling for other parameters. Also, from the $Pr(> |t|)$ column the interaction between MemoryBandwidthUsed and TotalPageDirty is significant. These results shows that, the parameter we considered can be used to explain the behaviour of our experiment. Taken together the predictor variables account for around 99 percent of the variance in Migration Time for all the Memory intensive benchmarks. Figure 6 and Figure 7 shows the Model3 and Model4, which include parameters like VM_TotalPageDirty, interaction between MemoryBandwidthUsed with TotalPageDirty and interaction between MemoryBandwidthUsed and MAR. From Model3, the regression coefficient for VM_TotalPageDirty is less significant. But for Model4 the regression coefficient for VM_TotalPageDirty is very significant and also the interaction between MemoryBandwidthUsed and MAR. All together, the models explain the data with predictor variables account for around 99 percent of the variance in migration time.
Using the Test dataset, we intended to predict the migration time for CPU and Memory intensive benchmarks migrated under 100 Mbps bandwidth. The results for the predictions are shown in Figure 12-15. These figures shows the Benchmark name, actual time measure as TotalTime, Number as the migration, Predicted as values predicted from the model, and Error which is a difference between actual value and predicted value. The SquaredError column is the square of Error column, this is need to calculate the standard error as required by ordinary least squares(OLS).

7 Evaluation

To evaluate the statistical assumptions underlying the models which is built using regression analysis. Four graphs that are useful for evaluating the model fit. Figure 8-11 [12] shows the Diagnostic plots for the four models considered in this study. With the assumptions from OLS regression we can understand these graphs. Normal Q-Q plot, which is in the upper left hand side, is a probability plot of the standardised residual against the values that would be expected under normality. All the four models the dependent variable is normally distributed for a fixed set of predictor values and the residual values are normally distributed with a mean close to 0. The Residual vs Fitted graphs captures all the systematic variance present in the data, leaving nothing but random noise. Apart from Model2, all models show clearly that Residual vs Fitted is curved relationship. From the Figure-9 Residual vs Fitted graph suggest that we can add a quadratic term to the regression.
To evaluate the prediction accuracy, we need to calculate the standard error of the estimate. The regression model estimates that minimises the sum of squared deviations of prediction. The standard error of the estimate is closely related to this quantity and is defined [3] as:

$$\sigma_{est} = \sqrt{\Sigma(actualvalue - predictedvalue)/totalnumberofsamples} \quad (6)$$

[3]http://onlinestatbook.com/2/regression/accuracy.html

The standard error of the estimate from all four models is given in Table-3. From the table, there is a tight fit for Model1 and Model3 suggesting that the predication valued are as close to actual values. Since the Model1 and Model3 are used for CPU intensive benchmark dataset, its evident that we can consider these models for prediction of migration time. In Model2 and Model4, which is the models for Memory Intensive benchmark dataset, has high sum of the squared errors when compared with other two models. As we can see from predicted data from Figure 13 and Figure 15, the predicted values are not too large compared to actual data. The high values for sum of the squared errors for models Model2 and Model4 can also be explained from the nature of the benchmark itself. As explained by [6] these benchmarks generate load in different time of the its execution time.

8 Conclusion and Future Work

In this internship, I investigate how various system resource (modelled as random variables) consumption can be used to estimate the VM migration time. Various other modelling technique have already suggested by [13]. But, under the guidance of my supervisor, I showed that the migration time can estimated and analysed by using Multiple Linear Regression model with various predictors. Based on the models produced, these models can be used to predict the migration time. The concepts suggested by the supervisor was proves that the migration time can be estimated and predicted using the various parameters like page dirty rate, last level cache misses, RAM utilisation, network bandwidth and Memory access rate. SPECCPU2006 benchmark has 19 different benchmark tests, and more benchmarks for investigation is kept for future work. To conclude, we see a high potential the linear regression models presented and can be use to estimate and predict the migration time.

References

[1] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," in *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2*, pp. 273–286, USENIX Association, 2005. (document)

[2] A. Strunk and W. Dargie, "Does live migration of virtual machines cost energy?," in *Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on*, pp. 514–521, IEEE, 2013. (document)

[3] K. Rybina, A. Patni, and A. Schill, "Analysing the migration time of live migration of multiple virtual machines," in *4th International Conference on Cloud Computing and Services Science (CLOSER 2014)*, 2014. (document)

[4] W. Voorsluys, J. Broberg, S. Venugopal, and R. Buyya, "Cost of virtual machine live migration in clouds: A performance evaluation," in *Cloud Computing*, pp. 254–265, Springer, 2009. 2

[5] I. Habib, "Virtualization with kvm," *Linux Journal*, vol. 2008, no. 166, p. 8, 2008. 2

[6] A. Jaleel, "Memory characterization of workloads using instrumentation-driven simulation," *Web Copy: http://www. glue. umd. edu/ajaleel/workload*, 2010. 4, 7

[7] J. L. Henning, "Spec cpu2006 benchmark descriptions," *ACM SIGARCH Computer Architecture News*, vol. 34, no. 4, pp. 1–17, 2006. 1

[8] R. P. Knight, "Method and apparatus for configuring and collecting performance counter data," Sept. 14 2004. US Patent 6,792,392. 5

[9] K. Rybina, W. Dargie, A. Strunk, and A. Schill, "Investigation into the energy cost of live migration of virtual machines," in *Sustainable Internet and ICT for Sustainability (SustainIT), 2013*, pp. 1–8, IEEE, 2013. 1

[10] M. Baron, *Probability and statistics for computer scientists.* CRC Press, 2013. 6.1, 2, 6.1

[11] C. Poellabauer, L. Singleton, and K. Schwan, "Feedback-based dynamic voltage and frequency scaling for memory-bound real-time applications," in *Real Time and Embedded Technology and Applications Symposium, 2005. RTAS 2005. 11th IEEE*, pp. 234–243, IEEE, 2005. 2

[12] R. Kabacoff, *R in Action.* Manning Publications Co., 2011. 7

[13] H. Liu, H. Jin, C.-Z. Xu, and X. Liao, "Performance and energy modeling for live migration of virtual machines," *Cluster computing*, vol. 16, no. 2, pp. 249–264, 2013. 8

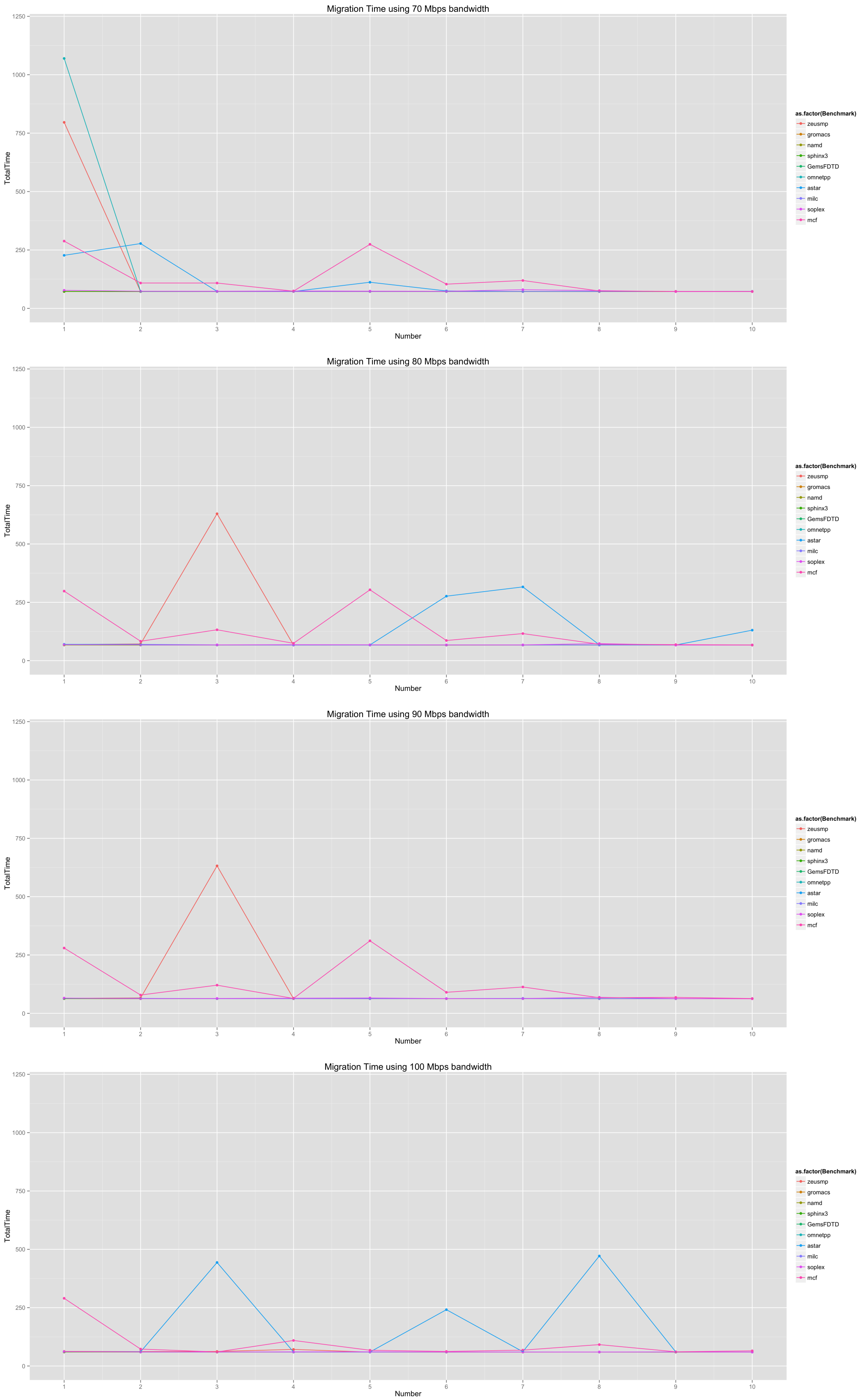

Figure 3: Time taken for Migration of VM

Table 2: Various parameters considered for multiple linear regression model

Name of the variable	Description	Units
TotalTime	Migration time for the benchmark and for a particular migration.	Seconds (Time).
Bandwidth	Bandwidth used for the migration.	MBps.
TotalL3Miss	Total number of L3 Miss for the benchmark and for a particular migration.	L3 cache line misses in millions.
TotalINST	Total number of instructions successfully retired by the CPU for the benchmark and for a particular migration.	Number of instructions retired.
MAR [11]	$$TotalL3Miss/TotalINST \quad (3)$$ for the benchmark and for a particular migration.	L3Miss in millions per Number of instructions retired.
MeanCpuUtilisation	Mean of CPU utilisation for the benchmark and for a particular migration.	Percent of CPU utilized.
MemoryBandwidthUsed	$$MemoryUsed/Bandwidth \quad (4)$$ for the benchmark and for a particular migration.	MBps.
TotalPageDirty, VM_TotalPageDirty	Total number of Page dirtied for the benchmark and for a particular migration. Collected on host and VM.	Number of Pages.
AvgROCL3M	Average Rate of change of L2 Misses.	L3 cache line misses in millions per Second.
AvgROCPageDirty	Average Rate of change of page dirty.	Page dirtied per Second.
PageDirtyPerSecondPerMigration	$$TotalPageDirty/TotalTime \quad (5)$$ for the benchmark and for a particular migration.	Page dirtied per Total Migration Time in Seconds.

```
Call:
lm(formula = TotalTime ~ TotalL3Miss + TotalINST + CpuUtil +
    MemoryBandwidthUsed + TotalPageDirty + MAR + TotalPageDirty:MemoryBandwidthUsed +
    MAR:MemoryBandwidthUsed, data = cpu_data_set)

Residuals:
    Min      1Q  Median      3Q     Max
-6.7328 -1.5583 -0.0308  1.8229  8.0590

Coefficients:
                                    Estimate Std. Error t value Pr(>|t|)
(Intercept)                        5.425e+01  8.508e+00   6.376 2.86e-09 ***
TotalL3Miss                        1.052e-07  9.137e-09  11.508  < 2e-16 ***
TotalINST                          1.666e-04  8.091e-06  20.586  < 2e-16 ***
CpuUtil                           -1.221e+00  4.130e-02 -29.552  < 2e-16 ***
MemoryBandwidthUsed               -5.407e-07  2.447e-07  -2.210  0.02886 *
TotalPageDirty                    -4.750e-01  1.665e-01  -2.852  0.00504 **
MAR                                2.075e-04  2.423e-03   0.086  0.93189
MemoryBandwidthUsed:TotalPageDirty  1.366e-08  4.783e-09   2.855  0.00501 **
MemoryBandwidthUsed:MAR           -1.217e-10  7.283e-11  -1.671  0.09709 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.044 on 131 degrees of freedom
Multiple R-squared:  0.9981,    Adjusted R-squared:  0.9979
F-statistic:  8449 on 8 and 131 DF,  p-value: < 2.2e-16
```

Figure 4: Mode1:Multiple Linear Regression Model for CPU intensive benchmark without parameters from VM

```
Call:
lm(formula = TotalTime ~ TotalL3Miss + TotalINST + CpuUtil +
    MemoryBandwidthUsed + TotalPageDirty + MAR:MemoryBandwidthUsed +
    TotalPageDirty:MemoryBandwidthUsed, data = memory_data_set)

Residuals:
    Min      1Q  Median      3Q     Max
-80.801  -6.149   1.330   6.491  85.820

Coefficients:
                                    Estimate Std. Error t value Pr(>|t|)
(Intercept)                        2.109e+00  7.718e+00   0.273   0.7850
TotalL3Miss                        4.902e-08  3.219e-09  15.226  < 2e-16 ***
TotalINST                          1.001e-04  8.986e-06  11.137  < 2e-16 ***
CpuUtil                           -3.015e-01  2.958e-01  -1.020   0.3094
MemoryBandwidthUsed                6.950e-07  3.677e-07   1.890   0.0604 .
TotalPageDirty                     7.378e-01  8.607e-02   8.573 5.65e-15 ***
MemoryBandwidthUsed:MAR           -2.663e-10  5.823e-11  -4.573 9.16e-06 ***
MemoryBandwidthUsed:TotalPageDirty -3.330e-09  1.864e-09  -1.787   0.0758 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 17.93 on 172 degrees of freedom
Multiple R-squared:  0.9962,    Adjusted R-squared:  0.996
F-statistic:  6385 on 7 and 172 DF,  p-value: < 2.2e-16
```

Figure 5: Model2:Multiple Linear Regression Model for Memory intensive benchmark without parameters from VM

```
Call:
lm(formula = TotalTime ~ CpuUtil + MemoryBandwidthUsed + TotalL3Miss +
    TotalINST + TotalPageDirty + VM_TotalPageDirty + MemoryBandwidthUsed:TotalPageDirty,
    data = combined_cpu_data_set)

Residuals:
    Min      1Q  Median      3Q     Max
-7.5683 -1.9832 -0.1963  1.7854  8.3622

Coefficients:
                                    Estimate Std. Error t value Pr(>|t|)
(Intercept)                        5.658e+01  8.494e+00   6.661 6.69e-10 ***
CpuUtil                           -1.305e+00  3.282e-02 -39.749  < 2e-16 ***
MemoryBandwidthUsed               -6.641e-07  2.428e-07  -2.735  0.00710 **
TotalL3Miss                        7.613e-08  2.511e-09  30.316  < 2e-16 ***
TotalINST                          1.921e-04  2.402e-06  79.949  < 2e-16 ***
TotalPageDirty                    -4.866e-01  1.719e-01  -2.830  0.00538 **
VM_TotalPageDirty                  5.330e-04  4.978e-04   1.071  0.28630
MemoryBandwidthUsed:TotalPageDirty  1.399e-08  4.938e-09   2.834  0.00532 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.143 on 132 degrees of freedom
Multiple R-squared:  0.9979,    Adjusted R-squared:  0.9978
F-statistic:  9053 on 7 and 132 DF,  p-value: < 2.2e-16
```

Figure 6: Model3:Multiple Linear Regression Model for CPU intensive benchmark with parameters from VM

```
Call:
lm(formula = TotalTime ~ CpuUtil + MemoryBandwidthUsed + TotalL3Miss +
    TotalINST + MAR:MemoryBandwidthUsed + TotalPageDirty + VM_TotalPageDirty,
    data = combined_memory_data_set)

Residuals:
    Min      1Q  Median      3Q     Max
-50.033  -2.617   1.557   3.887  49.771

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)              2.274e+01  5.279e+00   4.309 3.35e-05 ***
CpuUtil                 -4.904e-01  2.372e-01  -2.068  0.04078 *
MemoryBandwidthUsed      5.892e-07  2.835e-07   2.078  0.03980 *
TotalL3Miss              4.310e-08  2.784e-09  15.481  < 2e-16 ***
TotalINST                1.455e-04  7.950e-06  18.296  < 2e-16 ***
TotalPageDirty           2.562e-01  5.605e-02   4.570 1.18e-05 ***
VM_TotalPageDirty        1.488e-01  1.701e-02   8.745 1.47e-14 ***
MemoryBandwidthUsed:MAR -1.386e-10  4.537e-11  -3.055  0.00276 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.58 on 122 degrees of freedom
Multiple R-squared:  0.9979,    Adjusted R-squared:  0.9978
F-statistic:  8385 on 7 and 122 DF,  p-value: < 2.2e-16
```

Figure 7: Model4:Multiple Linear Regression Model for Memory intensive benchmark with parameters from VM

Table 3: The standard error of the estimate from all four models

Model	standard error of the estimate
Model1	2.98426
Model2	22.86872
Model3	3.172431
Model4	29.60733

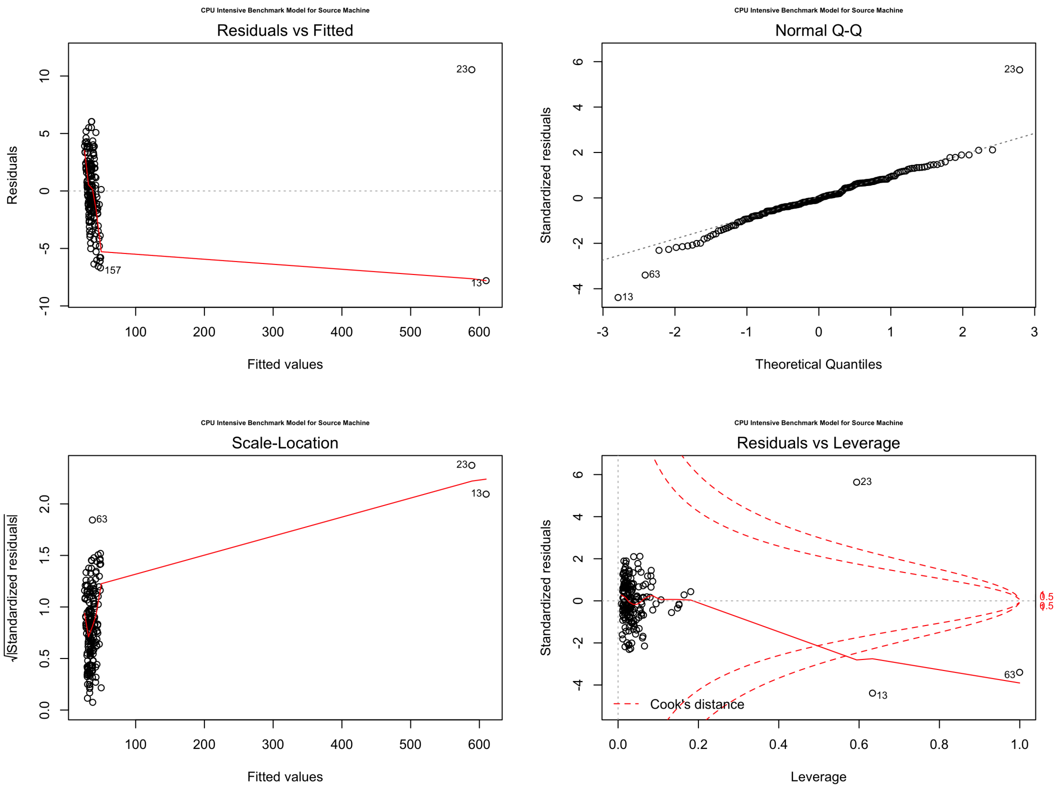

Figure 8: Model1:Diagnostic plots for Multiple Linear Regression Model for CPU intensive benchmark without parameters from VM

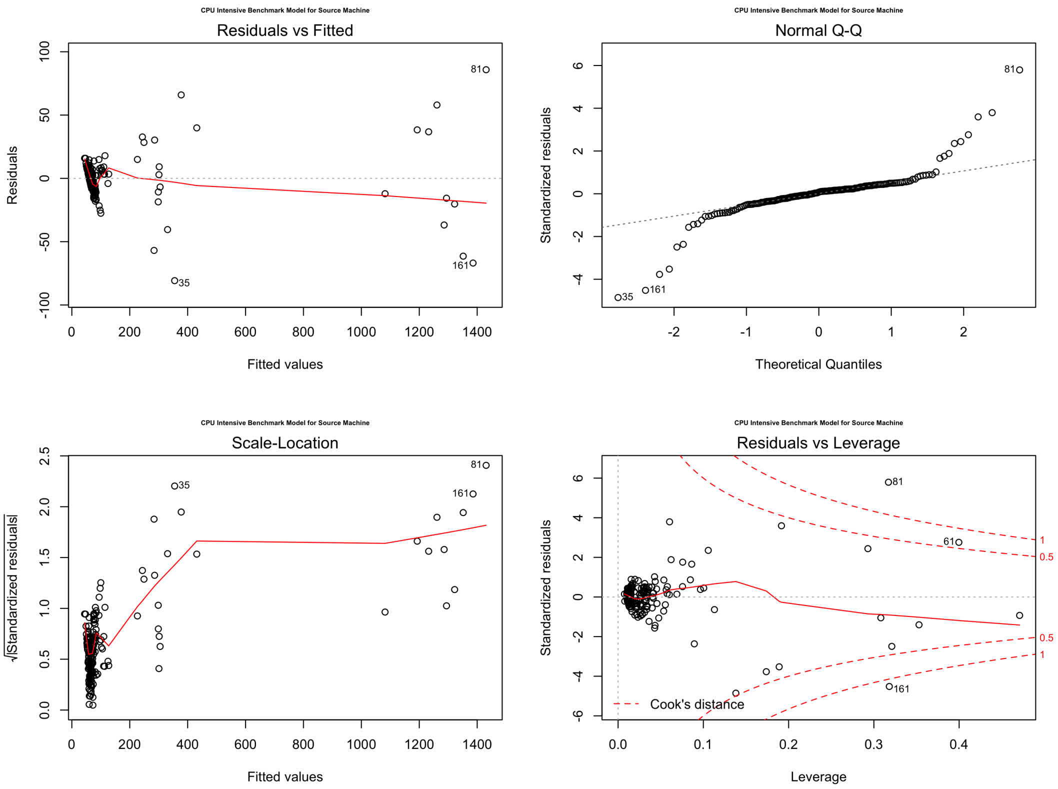

Figure 9: Model2:Diagnostic plots for Multiple Linear Regression Model for Memory intensive benchmark without parameters from VM

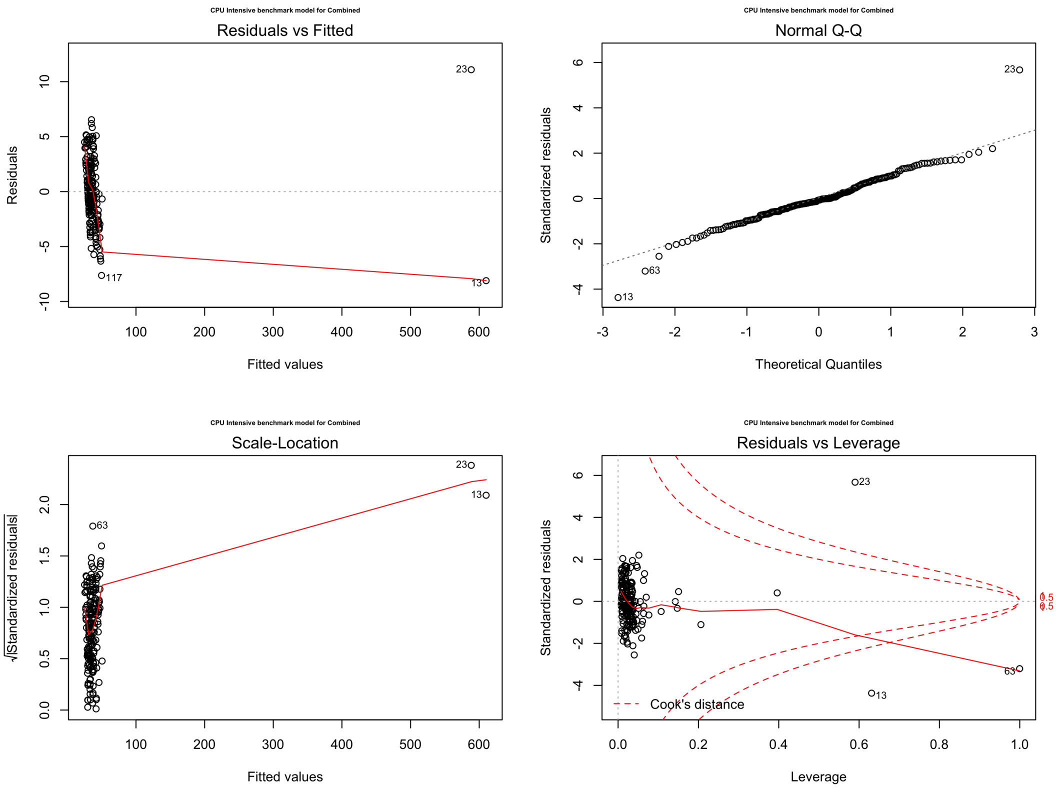

Figure 10: Model3:Diagnostic plots for Multiple Linear Regression Model for CPU intensive benchmark with parameters from VM

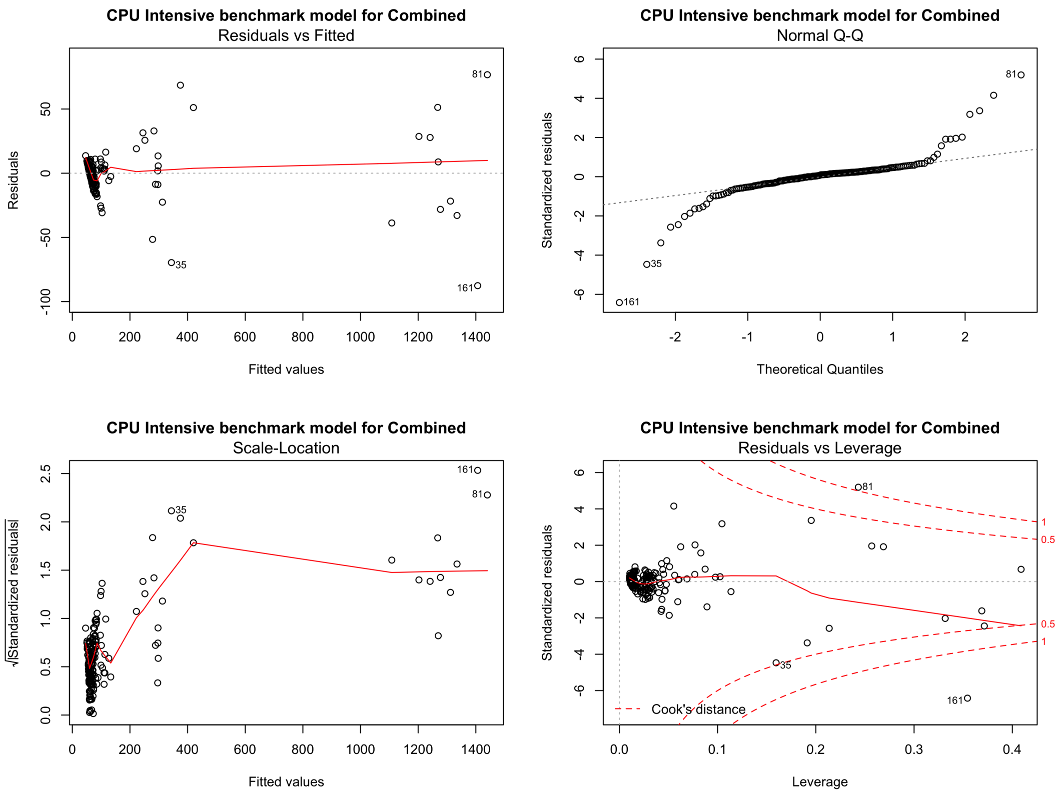

Figure 11: Model4:Diagnostic plots for Multiple Linear Regression Model for Memory intensive benchmark with parameters from VM

	Benchmark	TotalTime	Number	Predicted	Error	SquaredError
1	zeusmp	32.71996	1	30.35430	2.365651758	5.596308e+00
2	zeusmp	31.91802	2	29.03517	2.882845717	8.310799e+00
3	zeusmp	32.45324	3	27.55340	4.899837295	2.400841e+01
4	zeusmp	41.17010	4	35.64373	5.526366776	3.054073e+01
5	zeusmp	29.81432	5	33.39714	-3.582822279	1.283662e+01
6	zeusmp	29.51124	6	34.05884	-4.547605594	2.068072e+01
7	zeusmp	29.53576	7	34.15380	-4.618048315	2.132637e+01
8	zeusmp	29.59988	8	36.85466	-7.254773300	5.263174e+01
9	zeusmp	29.49310	9	35.30563	-5.812530638	3.378551e+01
10	zeusmp	29.51257	10	35.27801	-5.765445653	3.324036e+01
41	gromacs	30.89828	1	26.59862	4.299652696	1.848701e+01
42	gromacs	29.73636	2	30.35760	-0.621231425	3.859285e-01
43	gromacs	29.60169	3	26.22053	3.381162908	1.143226e+01
44	gromacs	29.62395	4	27.63933	1.984621641	3.938723e+00
45	gromacs	29.61509	5	27.89725	1.717844215	2.950989e+00
46	gromacs	29.60538	6	25.51842	4.086962603	1.670326e+01
47	gromacs	29.90053	7	32.53260	-2.632074563	6.927817e+00
48	gromacs	29.70384	8	27.81493	1.888913195	3.567993e+00
49	gromacs	29.55723	9	27.52133	2.035898263	4.144882e+00
50	gromacs	30.17714	10	30.98887	-0.811724267	6.588963e-01
81	namd	29.85728	1	30.06058	-0.203298761	4.133039e-02
82	namd	30.12475	2	28.29531	1.829438858	3.346847e+00
83	namd	29.80604	3	28.87283	0.933211837	8.708843e-01
84	namd	29.82809	4	27.22093	2.607164765	6.797308e+00
85	namd	29.80597	5	27.00677	2.799197374	7.835506e+00
86	namd	29.72035	6	32.55261	-2.832260955	8.021702e+00
87	namd	29.77985	7	25.00898	4.770877233	2.276127e+01
88	namd	30.03116	8	30.95162	-0.920455347	8.472380e-01
89	namd	29.91602	9	31.60944	-1.693424555	2.867687e+00
90	namd	30.28315	10	30.11774	0.165416915	2.736276e-02
121	sphinx3	31.16866	1	33.38135	-2.212691051	4.896002e+00
122	sphinx3	29.90979	2	34.29607	-4.386281198	1.923946e+01
123	sphinx3	29.66862	3	31.76883	-2.100209573	4.410880e+00
124	sphinx3	29.84055	4	30.53086	-0.690306362	4.765229e-01
125	sphinx3	29.71857	5	32.76171	-3.043136109	9.260677e+00
126	sphinx3	29.67307	6	31.56674	-1.893670458	3.585988e+00
127	sphinx3	29.68300	7	30.19632	-0.513320106	2.634975e-01
128	sphinx3	29.81768	8	31.05511	-1.237425698	1.531222e+00
129	sphinx3	29.69407	9	31.02154	-1.327472681	1.762184e+00
130	sphinx3	29.69465	10	31.07493	-1.380281681	1.905178e+00
161	soplex	31.75701	1	34.43626	-2.679251884	7.178391e+00
162	soplex	30.34135	2	30.07384	0.267511213	7.156225e-02
163	soplex	30.03616	3	32.28709	-2.250920849	5.066645e+00
164	soplex	30.86704	4	31.48302	-0.615981416	3.794331e-01
165	soplex	31.10397	5	32.20822	-1.104246924	1.219361e+00
166	soplex	30.08275	6	30.09072	-0.007969836	6.351829e-05
167	soplex	29.81538	7	30.38919	-0.573812812	3.292611e-01
168	soplex	30.18335	8	30.78536	-0.602002946	3.624075e-01
169	soplex	30.20875	9	29.97650	0.232258078	5.394381e-02
170	soplex	29.56235	10	33.77249	-4.210140531	1.772528e+01

Figure 12: Predictions from Model1

	Benchmark	TotalTime	Number	Predicted	Error	SquaredError
1	mcf	290.15782	1	321.37590	-31.218086	9.745689e+02
2	mcf	72.60985	2	81.37648	-8.766631	7.685381e+01
3	mcf	60.29172	3	55.94714	4.344581	1.887539e+01
4	mcf	109.53842	4	101.46288	8.075540	6.521435e+01
5	mcf	67.58828	5	77.98450	-10.396217	1.080813e+02
6	mcf	62.38882	6	56.17038	6.218444	3.866905e+01
7	mcf	68.08207	7	65.74675	2.335319	5.453717e+00
8	mcf	91.68822	8	90.16625	1.521964	2.316374e+00
9	mcf	60.95438	9	51.59308	9.361305	8.763403e+01
10	mcf	65.08196	10	66.18148	-1.099521	1.208947e+00
41	GemsFDTD	1301.84155	1	1309.85117	-8.009619	6.415400e+01
42	GemsFDTD	59.94999	2	58.07760	1.872396	3.505866e+00
43	GemsFDTD	59.93689	3	55.63745	4.299447	1.848524e+01
44	GemsFDTD	59.86342	4	54.61318	5.250245	2.756507e+01
45	GemsFDTD	59.94692	5	58.82878	1.118138	1.250233e+00
46	GemsFDTD	59.81896	6	62.28309	-2.464134	6.071959e+00
47	GemsFDTD	59.83038	7	48.09914	11.731238	1.376220e+02
48	GemsFDTD	59.95761	8	66.39643	-6.438819	4.145839e+01
49	GemsFDTD	59.82963	9	61.91566	-2.086034	4.351538e+00
50	GemsFDTD	59.82702	10	60.03874	-0.211723	4.482663e-02
81	omnetpp	1516.68822	1	1414.52640	102.161814	1.043704e+04
82	omnetpp	60.33062	2	57.90983	2.420793	5.860239e+00
83	omnetpp	59.96213	3	65.22568	-5.263552	2.770498e+01
84	omnetpp	60.37479	4	64.89877	-4.523980	2.046640e+01
85	omnetpp	59.89789	5	62.48660	-2.588713	6.701435e+00
86	omnetpp	59.89287	6	63.39464	-3.501766	1.226236e+01
87	omnetpp	60.32499	7	74.15781	-13.832819	1.913469e+02
88	omnetpp	59.86715	8	57.98932	1.877822	3.526214e+00
89	omnetpp	59.89939	9	52.90082	6.998572	4.898001e+01
90	omnetpp	60.01507	10	55.48116	4.533918	2.055641e+01
121	astar	61.33934	1	57.65513	3.684206	1.357337e+01
122	astar	61.17416	2	68.85045	-7.676287	5.892539e+01
123	astar	443.65335	3	370.21979	73.433561	5.392488e+03
124	astar	60.24101	4	59.23727	1.003739	1.007492e+00
125	astar	60.03311	5	68.26422	-8.231108	6.775115e+01
126	astar	241.43662	6	217.76715	23.669469	5.602438e+02
127	astar	61.37644	7	73.96426	-12.587814	1.584531e+02
128	astar	471.36638	8	417.46150	53.904886	2.905737e+03
129	astar	60.44100	9	65.69637	-5.255369	2.761891e+01
130	astar	59.91512	10	74.11997	-14.204849	2.017777e+02
161	milc	1318.84168	1	1382.05208	-63.210397	3.995554e+03
162	milc	60.07653	2	61.20490	-1.128378	1.273238e+00
163	milc	60.13990	3	73.48135	-13.341443	1.779941e+02
164	milc	59.89423	4	62.60783	-2.713605	7.363650e+00
165	milc	59.89632	5	64.57868	-4.682362	2.192451e+01
166	milc	60.12155	6	66.21653	-6.094981	3.714880e+01
167	milc	59.88766	7	60.95748	-1.069821	1.144517e+00
168	milc	59.92021	8	59.60287	0.317340	1.007047e-01
169	milc	60.49937	9	54.57225	5.927123	3.513079e+01
170	milc	60.01663	10	54.92951	5.087122	2.587881e+01

Figure 13: Predictions from Mode2

	Benchmark	TotalTime	Number	Predicted	Error	SquaredError
1	zeusmp	32.71996	1	29.82718	2.89277142	8.368127e+00
2	zeusmp	31.91802	2	28.49440	3.42361923	1.172117e+01
3	zeusmp	32.45324	3	27.01967	5.43356427	2.952362e+01
4	zeusmp	41.17010	4	35.02050	6.14960018	3.781758e+01
5	zeusmp	29.81432	5	33.82954	-4.01522128	1.612200e+01
6	zeusmp	29.51124	6	34.37188	-4.86064444	2.362586e+01
7	zeusmp	29.53576	7	34.43806	-4.90230837	2.403263e+01
8	zeusmp	29.59988	8	37.21761	-7.61772522	5.802974e+01
9	zeusmp	29.49310	9	35.62243	-6.12932948	3.756868e+01
10	zeusmp	29.51257	10	35.65007	-6.13750228	3.766893e+01
41	gromacs	30.89828	1	26.97510	3.92317636	1.539131e+01
42	gromacs	29.73636	2	29.97809	-0.24172267	5.842985e-02
43	gromacs	29.60169	3	25.68983	3.91186594	1.530270e+01
44	gromacs	29.62395	4	27.19683	2.42711472	5.890886e+00
45	gromacs	29.61509	5	27.40527	2.20982062	4.883307e+00
46	gromacs	29.60538	6	24.96567	4.63971787	2.152698e+01
47	gromacs	29.90053	7	32.08295	-2.18242267	4.762969e+00
48	gromacs	29.70384	8	27.37862	2.32521562	5.406628e+00
49	gromacs	29.55723	9	27.09716	2.46006691	6.051929e+00
50	gromacs	30.17714	10	30.54193	-0.36478819	1.330704e-01
81	namd	29.85728	1	30.89491	-1.03762441	1.076664e+00
82	namd	30.12475	2	28.88876	1.23599122	1.527674e+00
83	namd	29.80604	3	29.64276	0.16328500	2.666199e-02
84	namd	29.82809	4	27.69699	2.13110062	4.541590e+00
85	namd	29.80597	5	27.51855	2.28742352	5.232306e+00
86	namd	29.72035	6	33.42270	-3.70235072	1.370740e+01
87	namd	29.77985	7	25.39611	4.38374056	1.921718e+01
88	namd	30.03116	8	31.54089	-1.50972320	2.279264e+00
89	namd	29.91602	9	32.30348	-2.38746233	5.699976e+00
90	namd	30.28315	10	30.27031	0.01284539	1.650041e-04
121	sphinx3	31.16866	1	35.13860	-3.96993630	1.576039e+01
122	sphinx3	29.90979	2	33.36637	-3.45657663	1.194792e+01
123	sphinx3	29.66862	3	31.17687	-1.50824875	2.274814e+00
124	sphinx3	29.84055	4	30.03324	-0.19268536	3.712765e-02
125	sphinx3	29.71857	5	32.26624	-2.54767168	6.490631e+00
126	sphinx3	29.67307	6	31.18593	-1.51286528	2.288761e+00
127	sphinx3	29.68300	7	29.92095	-0.23795216	5.662123e-02
128	sphinx3	29.81768	8	30.71795	-0.90027134	8.104885e-01
129	sphinx3	29.69407	9	30.74266	-1.04859090	1.099543e+00
130	sphinx3	29.69465	10	30.85831	-1.16365651	1.354096e+00
161	soplex	31.75701	1	34.98607	-3.22906090	1.042683e+01
162	soplex	30.34135	2	29.97781	0.36353387	1.321569e-01
163	soplex	30.03616	3	33.14062	-3.10445231	9.637624e+00
164	soplex	30.86704	4	31.10427	-0.23722935	5.627776e-02
165	soplex	31.10397	5	31.53189	-0.42792091	1.831163e-01
166	soplex	30.08275	6	30.32295	-0.24019637	5.769430e-02
167	soplex	29.81538	7	31.27151	-1.45613215	2.120321e+00
168	soplex	30.18335	8	30.95024	-0.76688301	5.881096e-01
169	soplex	30.20875	9	29.48153	0.72722125	5.288507e-01
170	soplex	29.56235	10	34.05337	-4.49101731	2.016924e+01

Figure 14: Predictions from Model3

	Benchmark	TotalTime	Number	Predicted	Error	SquaredError
1	mcf	290.15782	1	290.51272	-0.3549023	1.259556e-01
2	mcf	72.60985	2	76.85862	-4.2487759	1.805210e+01
3	mcf	60.29172	3	56.80769	3.4840287	1.213846e+01
4	mcf	109.53842	4	101.63663	7.9017886	6.243826e+01
5	mcf	67.58828	5	88.67611	-21.0878211	4.446962e+02
6	mcf	62.38882	6	56.78928	5.5995431	3.135488e+01
7	mcf	68.08207	7	65.02721	3.0548649	9.332199e+00
8	mcf	91.68822	8	89.79828	1.8899354	3.571856e+00
9	mcf	60.95438	9	53.74897	7.2054129	5.191798e+01
10	mcf	65.08196	10	95.47856	-30.3966040	9.239535e+02
41	GemsFDTD	1301.84155	1	1373.49290	-71.6513550	5.133917e+03
42	GemsFDTD	59.94999	2	61.15021	-1.2002181	1.440524e+00
43	GemsFDTD	59.93689	3	60.83459	-0.8976973	8.058604e-01
44	GemsFDTD	59.86342	4	53.28784	6.5755808	4.323826e+01
45	GemsFDTD	59.94692	5	57.27813	2.6687921	7.122451e+00
46	GemsFDTD	59.81896	6	58.66861	1.1503454	1.323295e+00
47	GemsFDTD	59.83038	7	51.39523	8.4351456	7.115168e+01
48	GemsFDTD	59.95761	8	60.21432	-0.2567088	6.589940e-02
49	GemsFDTD	59.82963	9	58.11691	1.7127248	2.933426e+00
50	GemsFDTD	59.82702	10	56.68235	3.1446630	9.888905e+00
81	omnetpp	1516.68822	1	1470.77813	45.9100883	2.107736e+03
82	omnetpp	60.33062	2	79.27438	-18.9437599	3.588660e+02
83	omnetpp	59.96213	3	58.40084	1.5612915	2.437631e+00
84	omnetpp	60.37479	4	59.76267	0.6121174	3.746877e-01
85	omnetpp	59.89789	5	57.81951	2.0783776	4.319654e+00
86	omnetpp	59.89287	6	57.79092	2.1019498	4.418193e+00
87	omnetpp	60.32499	7	63.81490	-3.4899051	1.217944e+01
88	omnetpp	59.86715	8	56.20408	3.6630652	1.341805e+01
89	omnetpp	59.89939	9	52.84590	7.0534927	4.975176e+01
90	omnetpp	60.01507	10	56.35145	3.6636233	1.342214e+01
121	astar	61.33934	1	63.90989	-2.5705540	6.607748e+00
122	astar	61.17416	2	66.81429	-5.6401261	3.181102e+01
123	astar	443.65335	3	373.33703	70.3163238	4.944385e+03
124	astar	60.24101	4	57.75496	2.4860479	6.180434e+00
125	astar	60.03311	5	66.66781	-6.6346964	4.401920e+01
126	astar	241.43662	6	212.16447	29.2721510	8.568588e+02
127	astar	61.37644	7	68.64241	-7.2659694	5.279431e+01
128	astar	471.36638	8	400.99777	70.3686129	4.951742e+03
129	astar	60.44100	9	62.22725	-1.7862472	3.190679e+00
130	astar	59.91512	10	69.38675	-9.4716258	8.971170e+01
161	milc	1318.84168	1	1471.60678	-152.7650987	2.333718e+04
162	milc	60.07653	2	59.19388	0.8826487	7.790687e-01
163	milc	60.13990	3	63.33065	-3.1907466	1.018086e+01
164	milc	59.89423	4	60.18613	-0.2919043	8.520813e-02
165	milc	59.89632	5	58.79094	1.1053745	1.221853e+00
166	milc	60.12155	6	59.86916	0.2523894	6.370039e-02
167	milc	59.88766	7	56.90447	2.9831933	8.899442e+00
168	milc	59.92021	8	56.20485	3.7153648	1.380394e+01
169	milc	60.49937	9	54.41392	6.0854529	3.703274e+01
170	milc	60.01663	10	53.95287	6.0637628	3.676922e+01

Figure 15: Predictions from Model4