Portion of time in System mode

Jagrati Sharma (sharmj@rpi.edu)

[1]Rensselaer Polytechnic Institute, Tetherless World Constellation, Troy, NY, United States,

Abstract

For an operating system to work correctly it has two modes; system/kernel mode and user mode. When the CPU is in user mode it is running or executing a user application and the mode bit is set to 1 for identifying the user mode. In the kernel-mode, this bit is set to 0. The CPU performs the critical operation in the kernel-mode like booting, loading the operating system. The system can not execute the function of one mode in another, otherwise, a trap may be generated which would crash the whole system. The kernel-mode is very important for the system and is reserved for the lowest-level and most trusted function of the operating system.

The poster will mainly focus on the time spent in the system mode. Beginning with the motivation of choosing this dataset, followed by the details about the dataset. First, the data is cleaned and explored. Different models are built on the cleaned dataset to predict the portion of time spent in system mode or kernel mode. The last part of the presentations shows the prediction results and accuracy of Multi-variate linear regression and decision tree.

Motivation

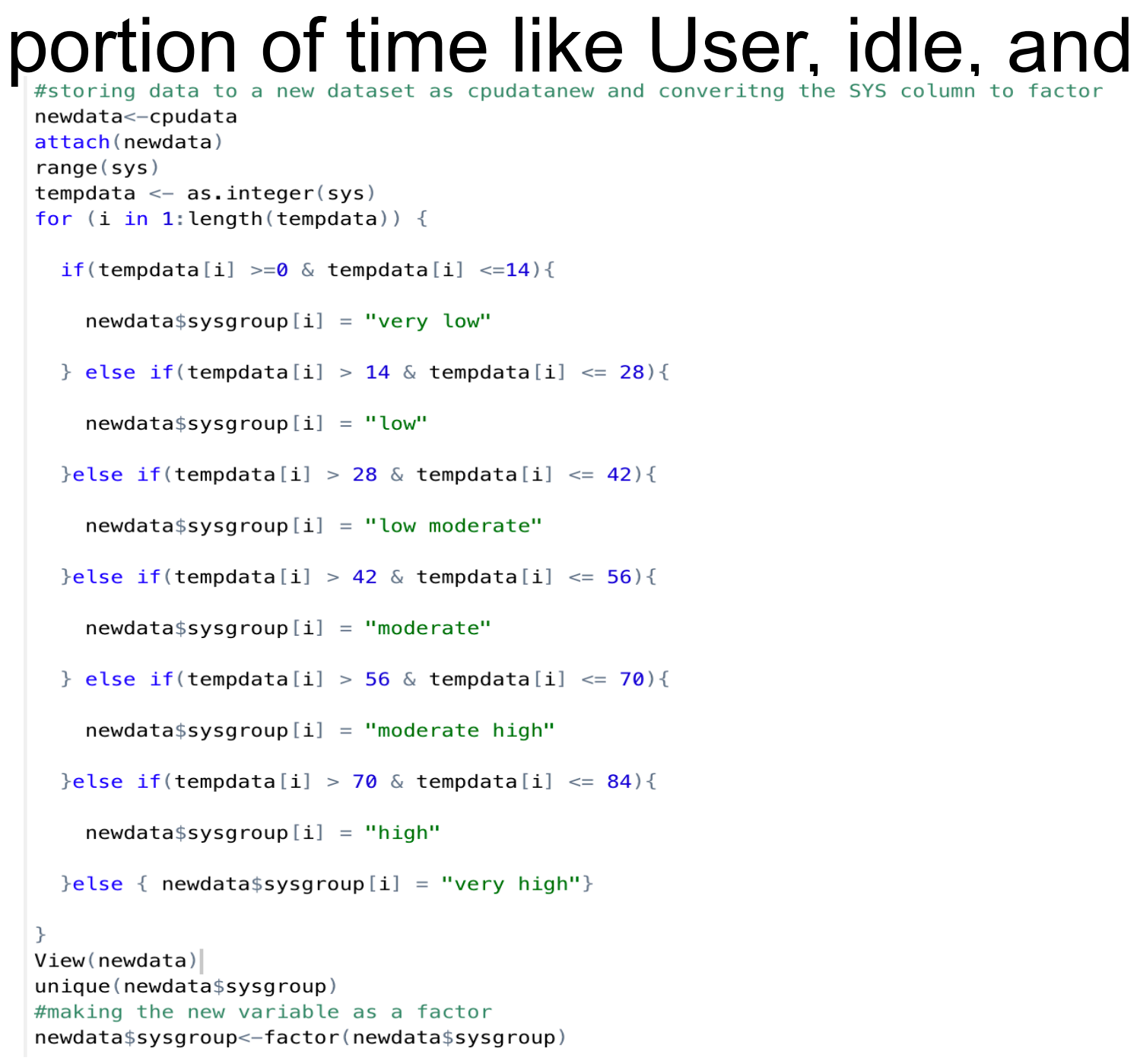
System mode is an important part of the CPU as all the critical operations are performed here. By determining the potion of time in CPU mode one can determine the system load and know how much time CPU invests in performing the critical operations. The division of modes is important for the CPU. As it might be possible that the user program accidentally affects the OS by overwriting it with the user data which may crash the whole system.

About Dataset

The dataset is the recorded performance measures from a Sun SPARCstation 20/712 model with 2 CPUs and 128 Mbytes of Main memory. The attributes in the dataset are a number of reads and write between system and user memory, system calls for different methods like fork, exec, read, write, information about pages, and a portion of the time of CPU. There is a total of 8192 observations with 27 variables when the data was loaded first.

Data Pre-processing

The data is either integer or numeric type but I converted sys variable to factors. I applied a check for NAs and even tried to normalize the data which can be used by different models. I removed the other portion of time like User, idle, and waiting.



```
#storing data to a new dataset as cpudatanew and converitng the SYS column to factor
newdata<-cpudata
attach(newdata)
range(sys)
tempdata <- as.integer(sys)
for (i in 1:length(tempdata)) {

  if(tempdata[i] >=0 & tempdata[i] <=14){

    newdata$sysgroup[i] = "very low"

  } else if(tempdata[i] > 14 & tempdata[i] <= 28){

    newdata$sysgroup[i] = "low"

  }else if(tempdata[i] > 28 & tempdata[i] <= 42){

    newdata$sysgroup[i] = "low moderate"

  }else if(tempdata[i] > 42 & tempdata[i] <= 56){

    newdata$sysgroup[i] = "moderate"

  } else if(tempdata[i] > 56 & tempdata[i] <= 70){

    newdata$sysgroup[i] = "moderate high"

  }else if(tempdata[i] > 70 & tempdata[i] <= 84){

    newdata$sysgroup[i] = "high"

  }else { newdata$sysgroup[i] = "very high"}

}
View(newdata)
unique(newdata$sysgroup)
#making the new variable as a factor
newdata$sysgroup<-factor(newdata$sysgroup)
```

FIG 1: CONVERTING SYS TO FACTORS

Data Exploration

The **Correlation Plot** is useful in highlighting the most correlated variables in a data table.

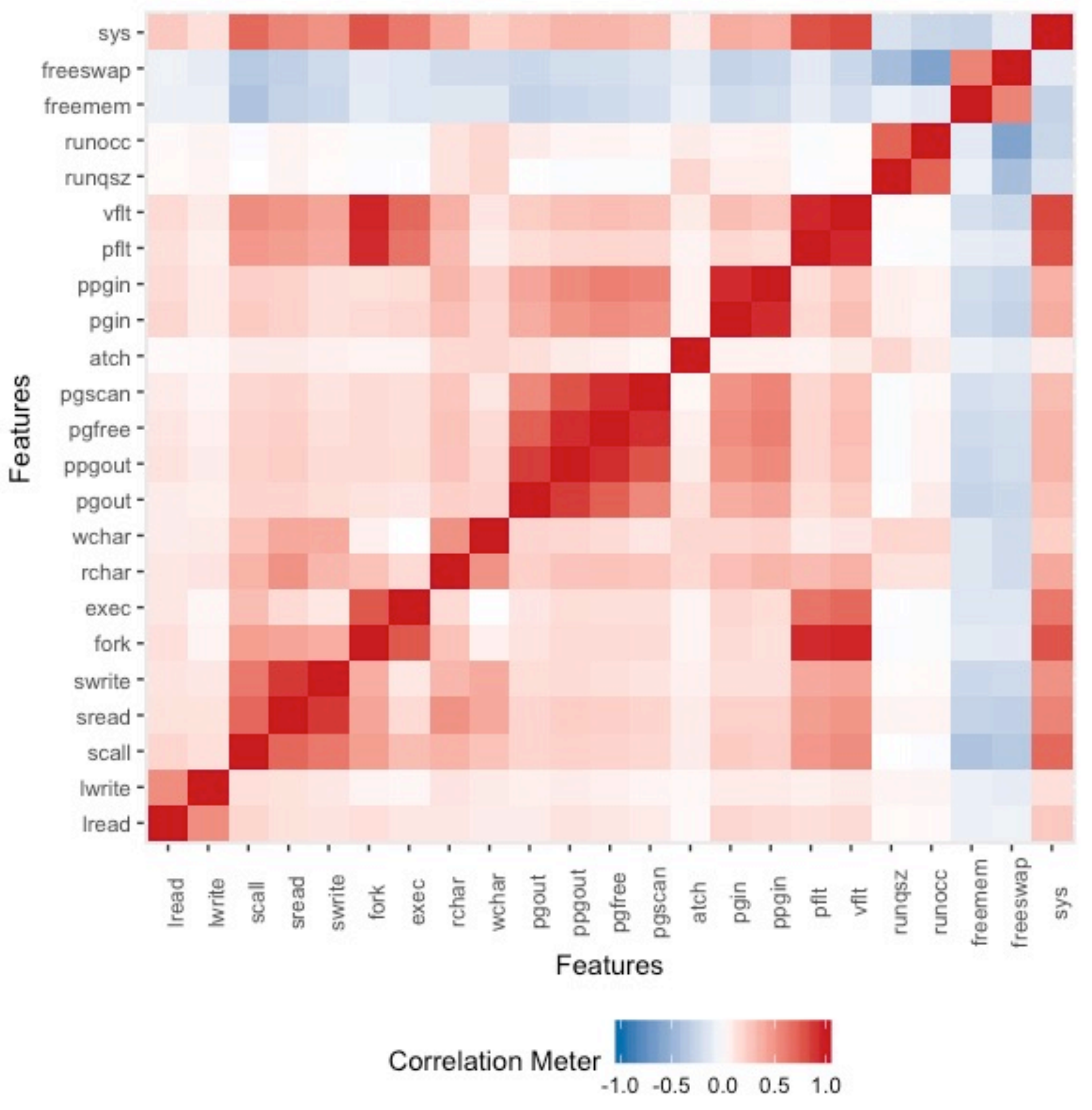


FIG 2: CORRELATION AMONG VARIABLES

The red color is identifying the positive correlation and blue is for the negative correlation.

Principle component Analysis helps in visualizing the variation present in the dataset.

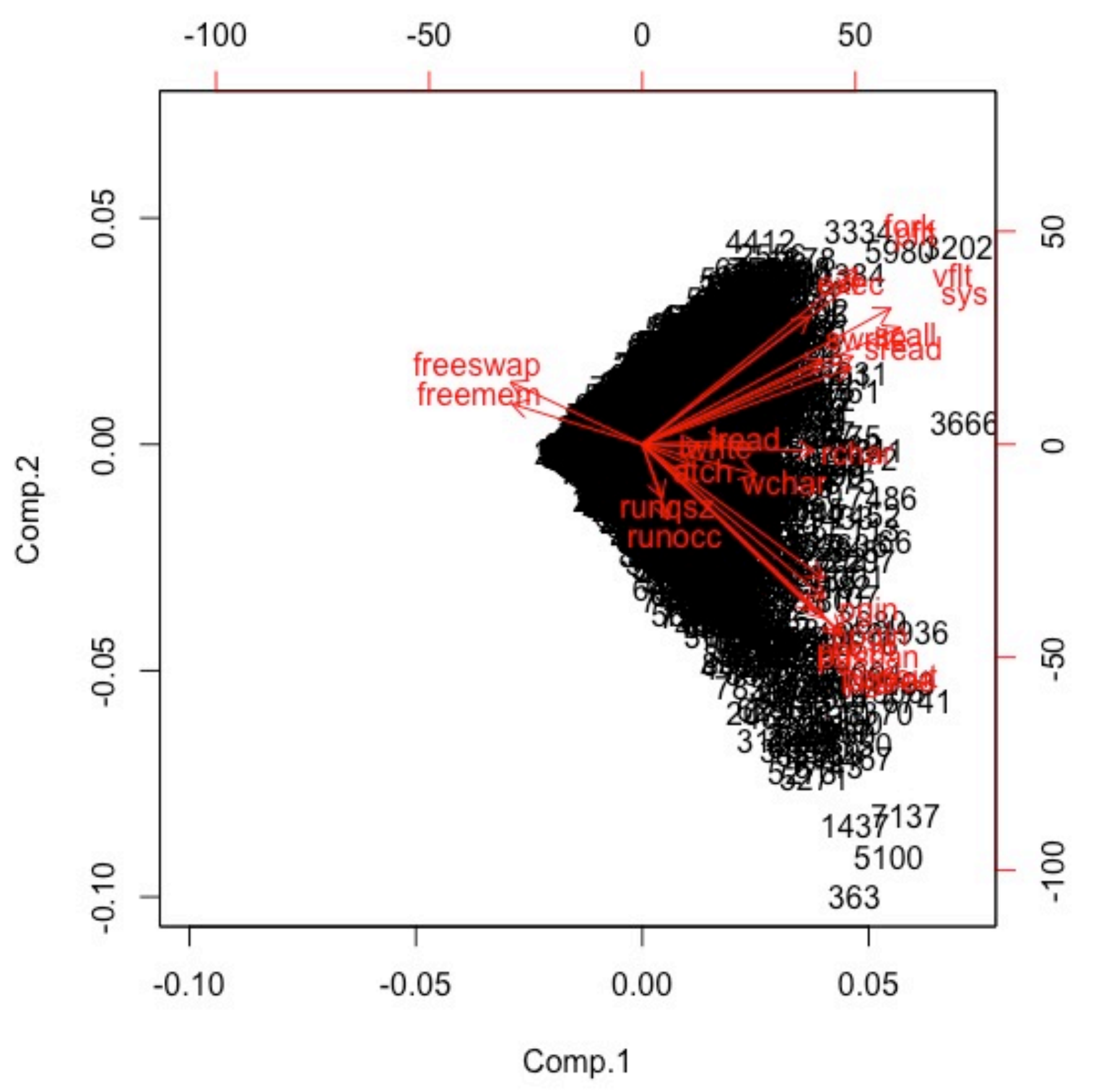


FIG 3: PRINCIPLE COMPONENT ANALYSIS

Here the vectors which are close to each other are strongly positive correlated and the vectors like freeswap and freemem are negatively correlated.

Data Analysis

Model 1: Multi-variate linear Regression

Here System mode is predicted using different other variables and a graph is plotted between predicted results and the actual system variable in the training set.

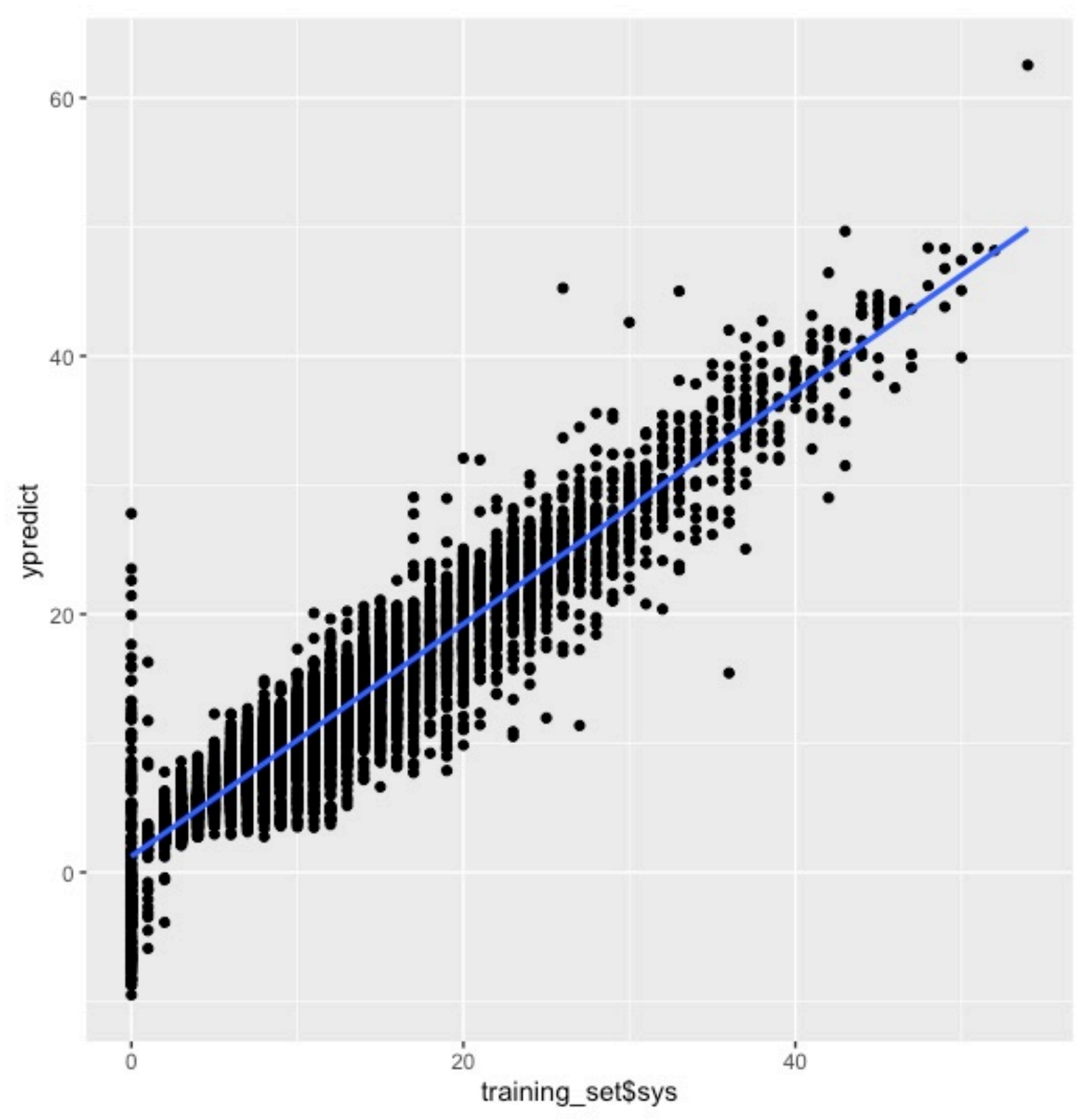


FIG 4: MULTI-VARIATE LINEAR REGRESSION MODEL

Model 2: Decision tree:

This approach will provide us with alternate solutions that are available to solve the problem.

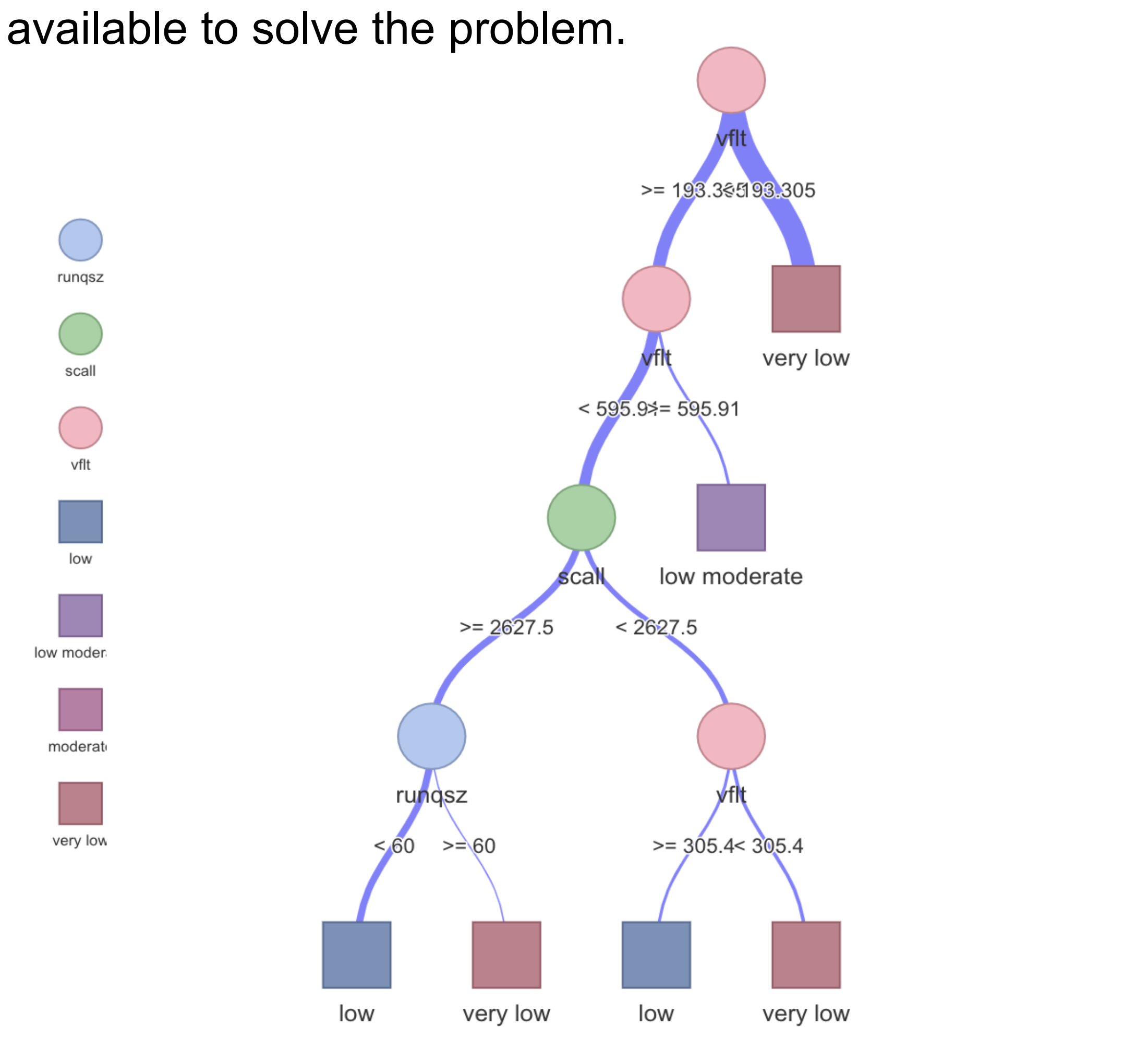


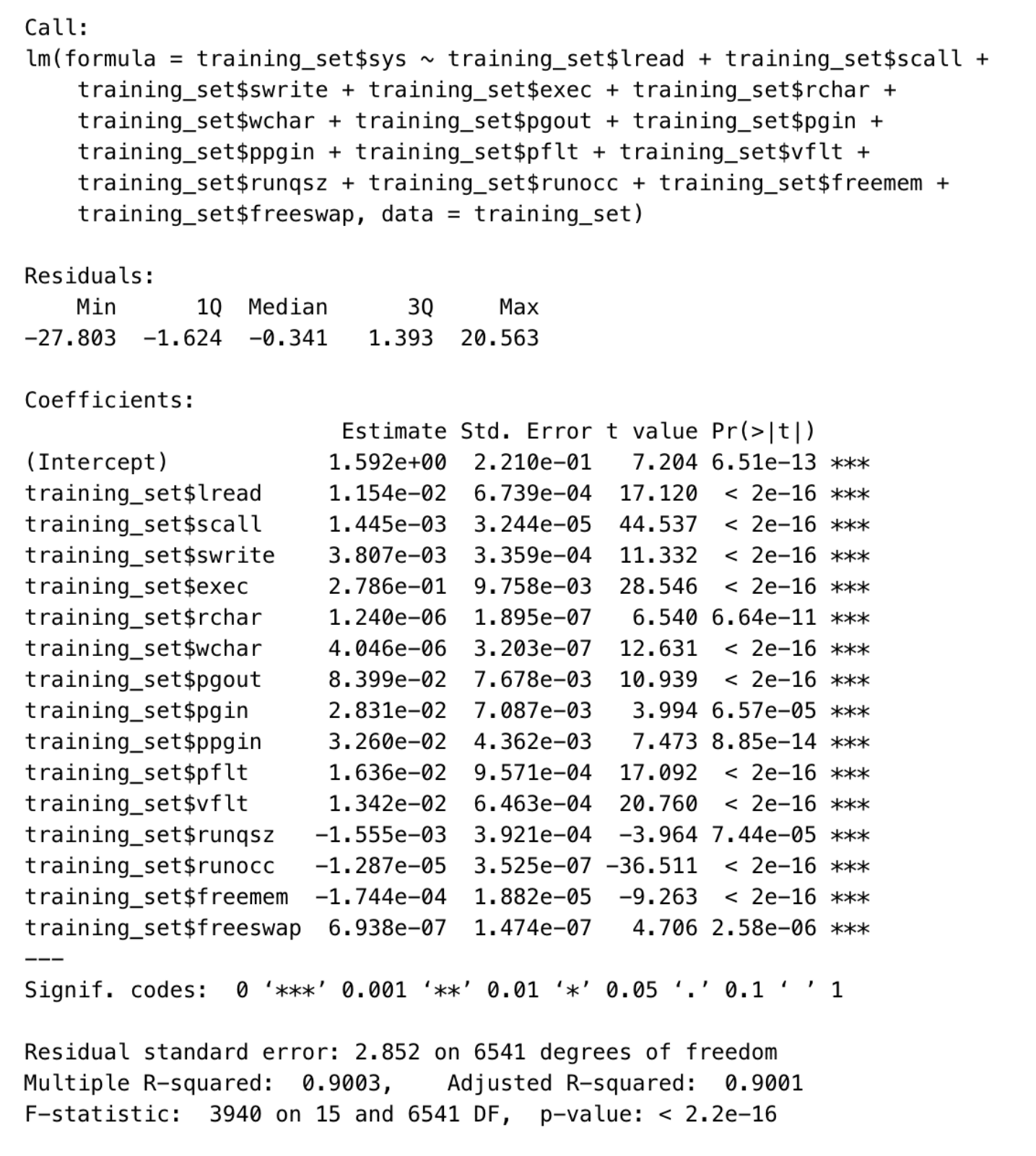
FIG 4: DECISION TREE

This tree is helping us to make a decision on the System mode by using different alternate branches with different attributes like scall, vflt, and runqsz.

Results

For Multi-Variate Linear Regression Model:

This model use is predicting time spent in system mode using lread, scall, wchar, pgout, rchar, ppgin, pflt, vflt, runocc, freemem, and freeswap. The system has a strong significance code. The multiple R-squared value is 0.9.

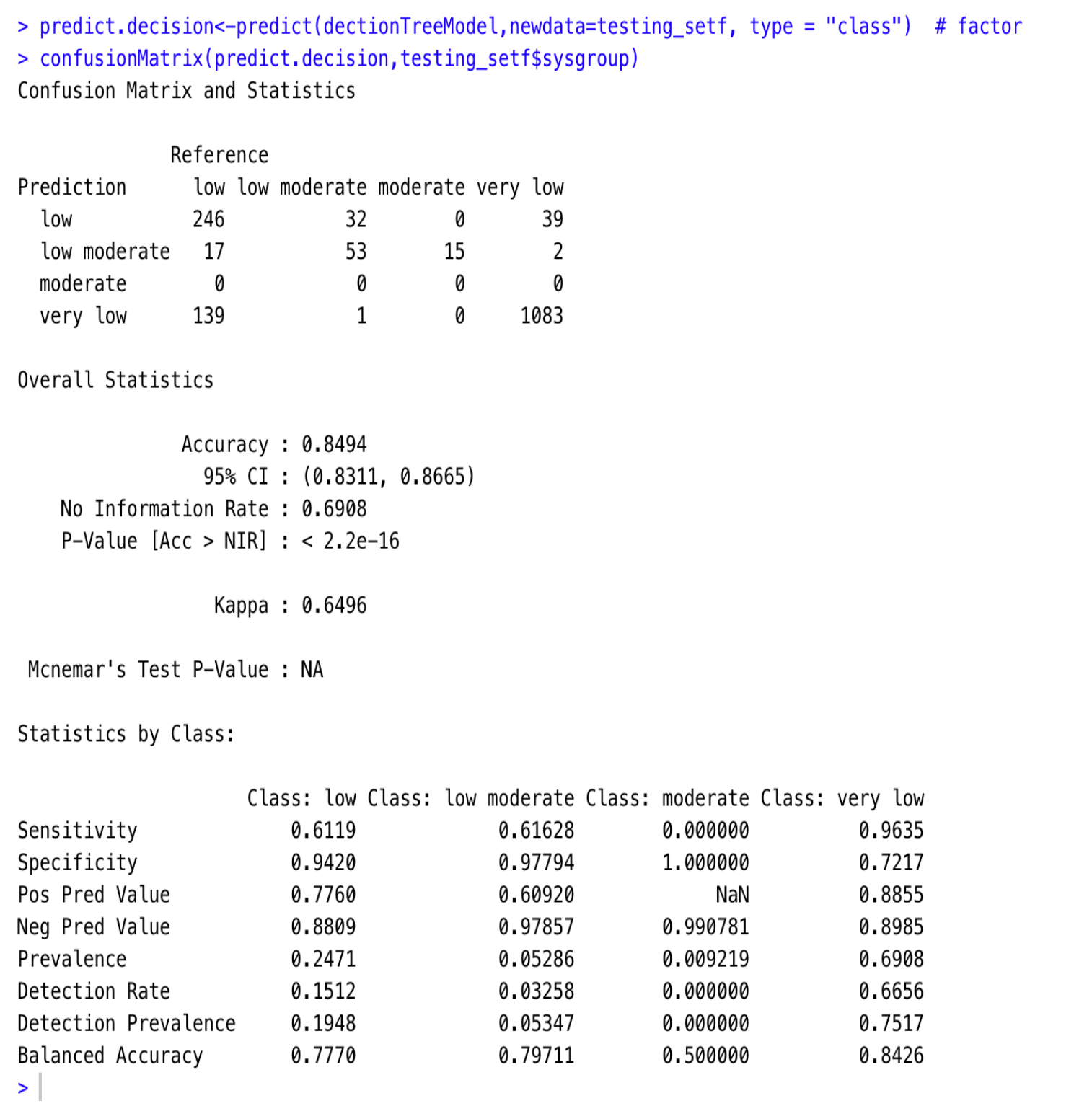


```
Call:
lm(formula = training_set$sys ~ training_set$lread + training_set$scall +
    training_set$swrite + training_set$exec + training_set$rchar +
    training_set$wchar + training_set$pgout + training_set$pgin +
    training_set$ppgin + training_set$pflt + training_set$vflt +
    training_set$runqsz + training_set$runocc + training_set$freemem +
    training_set$freeswap, data = training_set)

Residuals:
    Min      1Q  Median      3Q     Max
-27.803  -1.624  -0.341   1.393  20.563

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)            1.592e+00  2.210e-01   7.204 6.51e-13 ***
training_set$lread     1.154e-02  6.739e-04  17.120  < 2e-16 ***
training_set$scall     1.445e-03  3.244e-05  44.537  < 2e-16 ***
training_set$swrite    3.807e-03  3.359e-04  11.332  < 2e-16 ***
training_set$exec      2.786e-01  9.758e-03  28.546  < 2e-16 ***
training_set$rchar     1.240e-06  1.895e-07   6.540 6.64e-11 ***
training_set$wchar     4.046e-06  3.203e-07  12.631  < 2e-16 ***
training_set$pgout     8.399e-02  7.678e-03  10.939  < 2e-16 ***
training_set$pgin      2.831e-02  7.087e-03   3.994 6.57e-05 ***
training_set$ppgin     3.260e-02  4.362e-03   7.473 8.85e-14 ***
training_set$pflt      1.636e-02  9.571e-04  17.092  < 2e-16 ***
training_set$vflt      1.342e-02  6.463e-04  20.760  < 2e-16 ***
training_set$runqsz   -1.555e-03  3.921e-04  -3.964 7.44e-05 ***
training_set$runocc   -1.287e-05  3.525e-07 -36.511  < 2e-16 ***
training_set$freemem  -1.744e-04  1.882e-05  -9.263  < 2e-16 ***
training_set$freeswap  6.938e-07  1.474e-07   4.706 2.58e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.852 on 6541 degrees of freedom
Multiple R-squared:  0.9003,	Adjusted R-squared:  0.9001
F-statistic:  3940 on 15 and 6541 DF,  p-value: < 2.2e-16
```

FIG 5 : SUMMARY OF MULTI-VARIATE REGRESSOR FUNCTION

For Decision Tree:

This model makes use of confusion matrix to know the accuracy which is 0.84.



```
> predict.decision<-predict(dectionTreeModel,newdata=testing_setf, type = "class")  # factor
> confusionMatrix(predict.decision,testing_setf$sysgroup)
Confusion Matrix and Statistics

              Reference
Prediction      low low moderate moderate very low
  low           246           32        0       39
  low moderate   17           53       15        2
  moderate        0            0        0        0
  very low      139            1        0     1083

Overall Statistics

               Accuracy : 0.8494
                 95% CI : (0.8311, 0.8665)
    No Information Rate : 0.6908
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6496

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: low Class: low moderate Class: moderate Class: very low
Sensitivity              0.6119             0.61628        0.000000          0.9635
Specificity              0.9420             0.97794        1.000000          0.7217
Pos Pred Value           0.7760             0.60920             NaN          0.8855
Neg Pred Value           0.8809             0.97857        0.990781          0.8985
Prevalence               0.2471             0.05286        0.009219          0.6908
Detection Rate           0.1512             0.03258        0.000000          0.6656
Detection Prevalence     0.1948             0.05347        0.000000          0.7517
Balanced Accuracy        0.7770             0.79711        0.500000          0.8426
>
```

FIG 6 : SUMMARY OF DISEASE TREE

Conclusion

After applying models and seeing the accuracy of the models, time spent in system mode can be predicted by different variables using lread, scall, wchar, pgout, rchar, ppgin, pflt, vflt, runocc, freemem, and freeswap.

Sponsors:

Glossary:

- R - A program to process data and perform statistical analysis
- Package (P) or Library (R) - software package to be loaded to perform extra tasks
- Principle Component Analysis – It is an unsupervised learning technique for exploratory data analysis
- Multi-variate linear Regression model – Multiple independent variables contributing to the dependent variable
- Decision tree – Flowchart-like structure used to make decision for regression and classification problems

Resources:

- For dataset - https://www.cs.toronto.edu/~delve/data/comp-activ/desc.html
- For system mode - https://blog.codinghorror.com/understanding-user-and-kernel-mode/
- For abstract - https://stackoverflow.com/questions/1311402/what-is-the-difference-between-user-and-kernel-modes-in-operating-systems
- For R- https://www.rdocumentation.org/