iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 6 **Issue:** III **Month of publication:** March 2018

DOI: http://doi.org/10.22214/ijraset.2018.3480

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue III, March 2018- Available at www.ijraset.com

Adaptive Neuro-Fuzzy Inference System Approach on Predicting Hard Disk Failures towards Reliable Data Center

Jenniea A. Olalia [1], Dr. Ariel M. Sison[2], Dr. Cristina Aragon [3], Dr. Ruji P. Medina [4]
[1, 2, 3, 4,] *Graduate Studies, Technological Institute of the Philippines*

Abstract: Despite the high accuracy showcased by some studies in predicting hard disk failure using decision tree in the classification process, the accuracy of the decision tree is in question due to its vast difference among other algorithms ranging from 17.7% to 60.92%. This paper confirms the claim of some studies about the overfitting of decision tree when used with a large amount of data, real-valued and numeric attributes like SMART at-tributes. Utilizing ANFIS algorithm in predicting imminent hard disk failure surpasses other algorithms (CHAID, C&R Tree, Neural Network, MLR, and SVM) by 4.2% while keeping its distance from the very high percentage (99.58) over-fitted decision tree at 86.08%. The ANFIS also predicted the failure 5 days before it actually happens.

Keywords: ANFIS, Predicting Hard Disk Failure, SMART Attributes, Data Center, Imminent Disk Failure

I. INTRODUCTION

Having files rested and backed up on a data center or a cloud storage platform ensures continuous operation, data protection and recovery [1]. It is the responsibility of the data center to do all their necessary plans and procedures to be so. Critical factors such as experience, financial stability, security, support and physical infrastructure need to be maintained to have a reliable data center [2].

In the Philippines, 80% of business enterprises experienced data loss costing them around $8 billion worth of data loss [3]. On a larger scale, companies around the world experienced an average data loss of 400% in just two years accumulating a total amount loss of $1.7 trillion, 30% of which came from cloud storage. Admittedly, 51% of surveyed companies have no disaster recovery plan [4]. Study shows that the leading cause of data loss is the hardware failure, human error, software corruption, computer viruses and natural disaster among which hardware failure rank first at 57% [5]. Recent studies conducted by Data Barracks shows that hardware failure is still one of the topmost cause of data loss at 25%, being human error as the leader at 29% [6].

With the help of machine learning, failure can be preempted thereby avoiding data loss before it happens. There are some researches delving into this problem [7][8][9][10][11][12]. Each research presented varying results. Among these different researches and algorithms used, decision tree came out as the most accurate.

Similarly, Suchatpong and Bhumkittipich's study com-pared decision tree with the neural network, SVM, CHAID and C&R Tree. Decision tree ranks first at 99.58% while neural network is at 56.09% accurate, SVM is at 38.66%, CHAID at 50.42% and C&R tree at 56.93%, which are way lower that the decision tree. While the result of the decision tree is promising, there is an observable irregularity in the result. The decision tree is too accurate compare to other algorithm in predicting hard drive failure. IBM states that if an algorithm has 98% accuracy while other techniques tried has 60% accuracy, it is most probably overfitting, which is exactly the case [13]. Initial investigation shows that if a decision tree is fed with large amount of data, it tends to over-fit [14]. A slight variations in the training data will also make decision tree unstable [15][16]. Also, when a decision tree is supplied with real-valued attributes like the values in SMART Attributes, it will over-fit and it will give each numeric idea a branch thus the tree will become big [17]. Decision tree has also problems regarding robustness, adaptability, scalability and height optimization [18].

With the accuracy of the decision tree in questions and with low accuracy result of other algorithm, the researcher argues that the problem of predicting hard disk failure is still open and needs to be innovated. With the careful selection of data sets, systematic selection of predictors and use of suitable machine learning algorithm, the researcher believes that it will generate interesting result and produce new knowledge that will benefit not only the data centers but also the owner of the data.

This study aims to verify the truthfulness of the claim of other researchers that decision tree is inappropriate in this area and that this topic is still an open problem. This study introduced the use of ANFIS algorithm and compares its result to decision tree and other stated algorithms used by other researchers.

.

II. RELATED LITERATURE

Cloud storage is a service where data is remotely man-aged and maintained [19]. This can be accessed through the internet from any device. From 2015 to 2016, IDC's CloudView survey found 137% increase in utilization of data storage [20] and 95% of the respondents are using cloud of which 89% are public clouds, 72% are private clouds and 67% are combination or hybrid [21]. As the needs for data storage grow, it is expected that need for managing storage devices also increase. One of the top identified data growth strategies in a data center is the re-placement of existing hardware such as hard disk which will result to the top most challenges in data center at 31% [22].

Hard disk is still the most commonly used storage system in a data center at 75%, followed by hybrid storage (55%), all-flash (21%), software-defined storage (21%) and hy-per-converged infrastructures (16%) [23]. Hard disk, when deployed in a data center last for 1.38 years on the average with an annualized failure rate of 2.12% [24] or 1,000,000 – 1,500,00 hours at a nominal annual failure rate of at most 0.88% [25]. Other researches see the annual failure rate at 0.7% [26]. Therefore, constant monitoring of thousands of hard drive should be done not only for replacement of failed hard drive but also the prediction of pending future failure.

Being said, failure to determine that a hard drive will crash will cause catastrophic amount of data lost. A study conducted by Emerson and Ponemon Institute shows that the average total cost of per minute of unplanned outages is $8,851 or $740, 357 for year 2016 which indicates a 38% increase in downtime since their study on 2010 [27]. Since hard disk is a physical hardware, it logical to use physical characteristics or behaviour of this disk to predict imminent failure. The interrelationship of temperature, workload, and hard disk drive failures were studied and shows that temperature exhibits stronger correlation to failure than the correlation of disk utilization. However, monitoring drives using its physical characteristics will require special or cus-tom-made monitoring tools just to classify failure. On the other hand, interesting data are available that may be an alternative to physical predictors. These data include SMART attributes, daily logs, and complain logs.

Earlier studies conducted by Google suggest that some SMART attributes are correlated with high failure probabilities [28]. SMART (Self-Monitoring, Analysis, and Reporting Tool) is a monitoring system for computer hard disk that shows indicators of reliability [29] that are available to hard disk drives (HDD) and Solid State Drives (SSD).

BackBlaze's analysis of nearly 40,000 drives shows that among 253 defined SMART attributes, five (5) metrics correlate strongly with impending disk failure. These are 5, 187, 188, 197 and 198 [30]. Using time series analysis of SMART attributes from 30,000 disks from two major manufacturers, Botezatu and Giurgui identified combination of SMART 197, 188, 10, 201 and 5 to be the cause of failure [31]. Although the two researches uses SMART at-tributes as predictors, the difference between their data might cause the distinction between their selected attributes. These can also be ascribed to the difference in the combination of hard disk brand, model, capacity, working environment and selection algorithm used.

In the classification process, rule-based decision support algorithms were utilized. Algorithms employed includes Maximum Likelihood Rules, Classification and Regression Trees, Bayesian Networks, Decision Tree, Support Vector Machine + Time Series & Survival Analysis and multi-instance learning framework using Naïve Bayes that yields varying and opposing results and comparison. Some researches employed novel or hybrid algorithms such as Gaussian mixture based fault detection [32], Two-Step Parametric Method [33], feature selection-based Mahalanobis distance [34], Gradient Boosted Regression Trees [35]. With all the different algorithms used, Adaptive Neuro-Fuzzy Inference System (ANFIS) was not used.

It can be observed that among different algorithms, decision tree exhibited better performance for classification having an accuracy percentage from 98% to 99.58% while other algorithms' performances are way too low. Though the performance of the decision tree is promising, it can be somehow puzzling or confusing. Decision tree is too accurate and the result is too good to be true. IBM, in its article states that if an algorithm has 98% accuracy while other techniques used has 60% accuracy, it is most probably overfitting [36]. Overfitting happens when a given algorithm models the training data set too well [37]. Overfitting can also be caused by inappropriate data. Study shows that if a decision tree is fed with large amount of data, it also tends to over-fit [38]. This will influence the performance of the decision tree learning. A slight variation in the training data will also make decision tree unstable [39][40].

Moreover, decision tree has issues with real-valued attributes. Since SMART attributes are real numbers when these are subjected to decision tree, the decision tree will over-fit and it will give each numeric idea a branch thus the tree will become big [41]. It can create over-complex trees that do not represent the data well [42]. Additionally, decision tree is good when the values are discrete and not continuous which contradicts the nature of hard disk data set [43]. Decision tree has also problems regarding robustness, adaptability, scalability and height optimization [44].

Being said, with the decision tree's performance in question and the poor performances of other algorithms, the researcher believes that the problems in this area is still un-clear and still open for innovation. Appropriate algorithm can be identified and used to

predict pending hard drive failure while addressing the issues and inappropriateness of decision tree. An algorithm whose strength is on continuous values can handle large amount of data sets, can handle overfitting easily, adaptable and scalable. An example of this algorithm is ANFIS (Adaptive Neuro-Fuzzy Inference System).

ANFIS is a combination of neural network and fuzzy inference system [45]. It has been implemented in wide variety of applications that perform forecasting such as in wind power [46], electric load [47], social living [48], pricing ([49], communication [50], agriculture [51] and even tourism [52]. ANFIS also leads comparative analysis with other algorithms for classification [53][54]. Additionally, ANFIS is a reliable system with relatively high degree of accuracy [55], has very low mean absolute percentage error [56], has the ability to train quickly with just a few epochs [57] and efficient with an extreme standard of accuracy and using minimum need of data set samples.

ANFIS is very useful when solving complex problems especially for technical diagnostics and measurement assignments [58]. It can handle large amount of data set without any problem [59]. ANFIS may also over-fit but can be easily addressed. During training, it can be stop to a number of epochs after reaching the desired accuracy. Overfitting can also be avoided if the number of hidden layers be regulated [60]. This can be done by minimizing the number of predictors, thus predictor selection is needed. It can also handle and can be used with categorical and continuous data [61]. Similar to decision tree, ANFIS is capable of providing significant answer even if the data is incomplete or contains error [62]. Unseen data or relationship can also be infered after learning from the initial inputs and relationships [63].

ANFIS was used with hard disk failure. however, it was used to detect hard drive failure distribution to determine the specific hard disk part that has a defect in a manufacturing environment [64]. Although it was used for hard drive, it was not used for detecting pending failure on the hard drive during actual operation. SMART attributes were not also used in the prediction or classification process.

III.DESIGN ARCHITECTURE

This study classified or predict imminent disk drive failure by employing adaptive neuro-fuzzy inference system (ANFIS) classifier from an actual data center. The data set should also be comprehensive enough to get the best result when conducting classification.

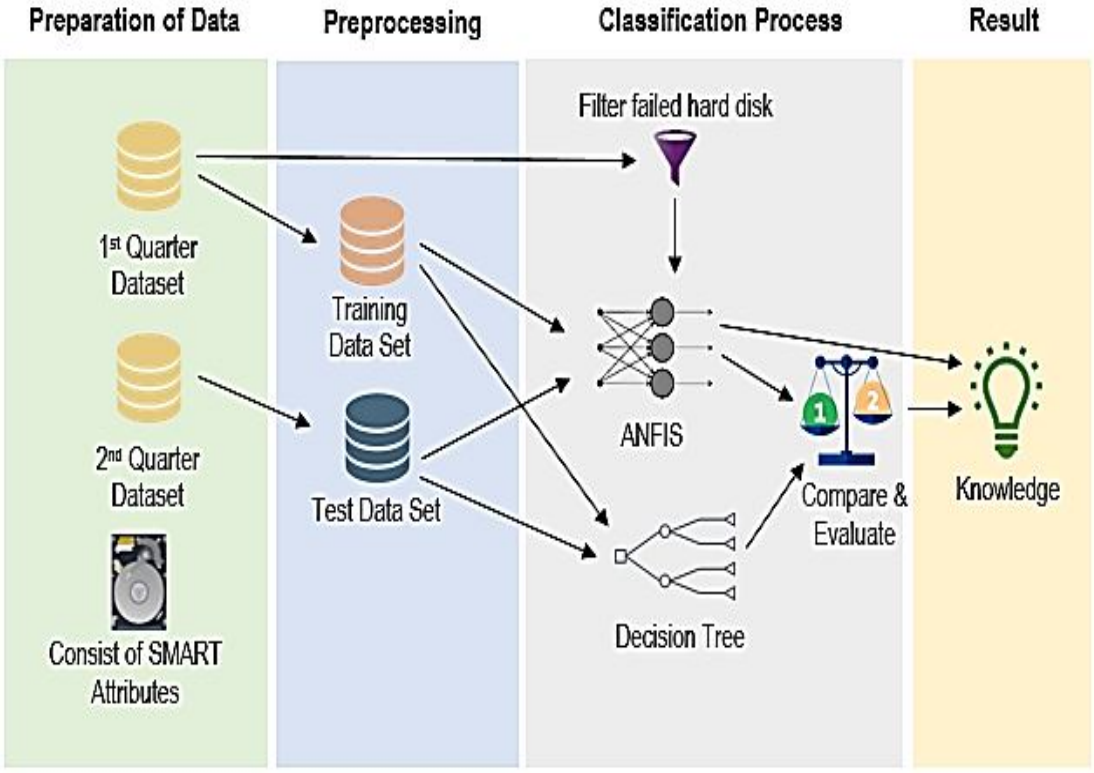


Fig. 1 A sample line graph using colors which contrast well both on screen and on a black-and-white hardcopy

The study went through four phases: preparation of data, generation and training of models, testing of models, and the interpretation of result.

A. Data Preparation

Data sets were downloaded from the official website of Backblaze (backblaze.com). Data sets for the year 2017 were grouped by quarters: Quarter 1 (January to March logs) and Quarter 2 (April to June logs). Quarter 1 was used as the training data set while Quarter 2 was used as the test data set. Q1 data set comprises of 6,632,223 records while Q2 data set composes of 7,568,630 records. To determine the needed columns or attributes for the machine learning, some unnecessary columns were re-moved such as the date, serial_number, model, and capacity_bytes columns. SMART attributes related to disk errors were determined since not all of these attributes have something to do with disk errors. The basis for the predeter-mination of these columns / attributes are the researches conducted by several researchers. All SMART attributes backed by researches were taken into consideration and was used for the training of both Decision Tree and ANFIS. These includes SMART 5, 10, 184, 187, 188, 196, 197, 198, 201 and 230. However, since SMART 230 is not present in the dataset from BackBlaze, it is removed from the final set of columns as shown in Table I.

TABLE I
FINAL SET OF ATTRIBUTES USED

SMART Attributes	Description
Failure	Status of the drive. 0-working, 1-failed
SMART 5	Reallocated Sector Count. Primarily used as a metric of the life expectancy of the drive.
SMART 10	Spin Retry Count. Stores a total count of spin start attempts to reach the full operation speed.
SMART 184	End-to-end error. Contains parity error count that occurs in the data path to the media via the drive's cache RAM.
SMART 187	Reported Uncorrectable Error. The count of errors that cannot be recovered.
SMART 188	Command Timeout. The count of aborted operations due to HDD timeout.
SMART 196	Reallocation Event Count. Total count of attempts to transfer data from reallocated sectors to spare area.
SMART 197	Current Pending Sector Count. Count of an unstable sector.
SMART 198	Uncorrectable Sector Error. The total count of uncorrectable errors when reading/writing a sector.
SMART 201	Soft Read Error Rate. The number of un correctable software read errors.

As observed in the data sets, some columns have null values. Null values were replaced with blanks so that it can be imported to MySQL. However, to make it work with decision tree, these values were reverted back to 0 using MySQL queries.

Finally, to make the data set ready for machine learning, duplicate records were detected. Distinct set of row values were queried and saved into another table for safe-keeping. Table II shows the initial number of records per month and per quarter and the distinct number of row sets.

Quarter 2 sub-total initially summed up to 2,421 records. However, after subjecting the same consolidated number of records to identification of duplicate records it further trims down to 1,512 only.

The first quarter data set was used as the training data set while the second quarter data set will be used as the test data set. The first quarter data set contains 185 records of failed drives and 2,219 healthy drives while second-quarter data set contains 166 record of failed drives and 1,346 healthy drives.

TABLE IIi
DISTINCT NUMBER OF ROW SET ON DATA SET

Quarter / Month	Original Number of Records		Distinct Number of Row Set		Percent decrease from initial values
	No. of Records	Sub-Total	No. of Records	Sub-Total	
Quarter 1		6,632,223		2,404	99.96%
- January	1,989,581		756		99.96%
- February	2,132,361		808		99.96%
- March	2,510,281		840		99.97%
Quarter 2		7,568,630		1,512	99.98%
- April	2,509,044		853		99.97%
- May	2,554,182		786		99.97%
- June	2,505,404		782		99.97%
Total Records		14,200,853		3,916	99.97%

Interesting results can be seen in Table II. While there is a colossal amount of records that was initially gathered from Backblaze, most of these records have duplicate which means when fed to machine learning algorithm will cause the algorithm to just prolonging its operation in training and testing on duplicate records. The percentage decrease (mostly 99.97%) on the number of data sets will directly influence the performance of algorithms to be used.

B. The ANFIS Structure

Title must be in 24 pt Regular font. Author name must be in 11 pt Regular font.

ANFIS is represented normally by a 6-layer feed-forward neural network as shown in Figure 2 and implements four rules:

Rule 1: IF x1 is A1 AND x2 is B1 THEN y = f1 = k10 + k11x1 + k12x2
Rule 2: IF x1 is A2 AND x2 is B2 THEN y = f2 = k20 + k21x1 + k22x2
Rule 3: IF x1 is A2 AND x2 is B1 THEN y = f3 = k30 + k31x1 + k32x2
Rule 4: IF x1 is A1 AND x2 is B2 THEN y = f4 = k40 + k41x1 + k42x2

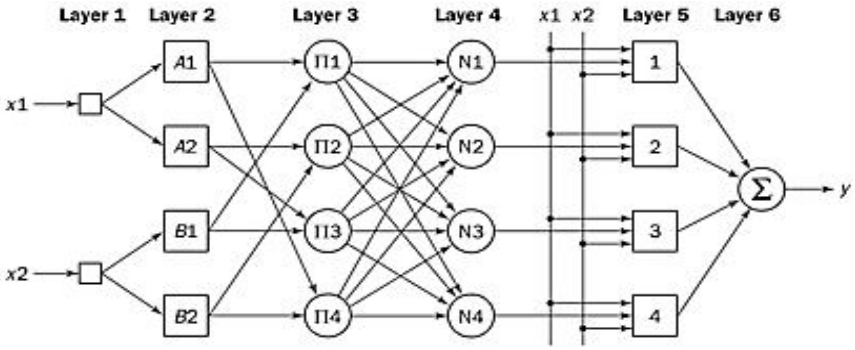


Fig. 2 ANFIS Structure

These rules were used in layers of the ANFIS structure. These layers functions as follows:

Layer 1 (Input Layer). Neuron on this layer passes crisp values to Layer 2. Specifically, $y_i^{(1)}=x_i^{(1)}$, where $x_i^{(1)}$ is the input and $y_i^{(1)}$ is the output.

Layer 2 (Fuzzification Layer). In this layer, neuron performs fuzzification using bell activation function (1):

$$y_i^{(2)} = \frac{1}{1+\left(\frac{x_i^{(2)}-a_i}{c_i}\right)^{2b_i'}}, \quad (1)$$

wherein $x_i^{(2)}$ is the input and $y_i^{(2)}$ is the output in neuron *i* in layer 2 and a_i, b_i and c_i are the center, width and slope of the bell activation function of neuron *i*.

Layer 3 (Rule Layer). Each neuron on this layers corresponds to a Sugeno-type fuzzy rule. The neuron receives inputs from fuzzification neurons from layer 2 and calculates the firing strength of the rule it represents. Output in the neuron i in Layer 3 is attained using (2):

$$y_i^{(3)} = \sum_{j=1}^{k} x_{ji}^{(3)} \quad (2)$$

where $x_{ji}^{(3)}$ are the inputs and $y_i^{(3)}$ is the output.

Layer 4 (Normalization Layer). Neurons in this layer receives input from the Rule Layer and calculates the normalized firing strength which is the ratio of the firing strength is to the sum of entire firing strengths of rules. It uses the following formula (3):

$$y_i^{(4)} = \frac{x_{ji}^{(4)}}{\sum_{j=1}^{n} x_{ii}^{(4)}} = \frac{\mu_i}{\sum_{j=1}^{n} \mu_i} = \bar{\mu}_i \quad (3)$$

where $x_{ji}^{(4)}$ is the input from neuron *j* in Layer 3 to neuron *i* in Layer 4 and n is the total number of rule neuron.

Layer 5 (Defuzzification Layer). Neuron in this layer receive inputs from Layer 4 and initial inputs, x_1 and x_2. This neuron calculate the weighted consequent value of a given rule using the formula (4):

$$y_i^{(5)} = x_i^{(5)}\left[k_{i0} + k_{i1}x1 + k_{i2}x2\right] = \bar{\mu}_i\left[k_{i0} + k_{i1}x1 + k_{i2}x2\right] \quad (4)$$

where $x_i^{(5)}$ is the inpu and $y_i^{(5)}$ is the output and k_{i0}, k_{i1}, and k_{i2} is the set of consequent parameters in rule *i*.

Layer 6 (Summation Layer). This layer comprises of only a single neuron that calculates the sum of all defuzzification neurons using (5):

$$y = \sum_{i=1}^{n} x_i^{(6)} = \sum_{i=1}^{n} \bar{\mu}_i\left[k_{i0} + k_{i1}x1 + k_{i2}x2\right] \quad (5)$$

C. Generation and Training of Models

1) *Decision Tree:* Having prepared the data sets, models were generated using Orange 3.8 data mining software for the decision tree and MatLab R2016 for ANFIS. Same training data sets (q1final.csv) was used to train each model.

A workflow was prepared using Orange. Figure 3 shows the workflow in training the model for decision tree. A data file was inserted and loaded with the training data sets. All columns corresponding to SMART attributes were set as “feature” and the type is numeric since these attributes are real numbers. The “failure” column which is the last feature in the data set was set to “target” and the type is categorical.

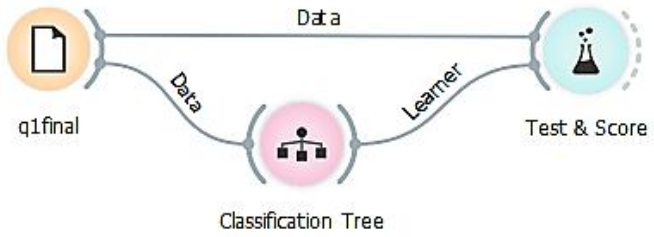


Fig. 3 Decision tree workflow in Orange

A tree algorithm model with forward pruning was placed in the work flow. Default parameters have been left as is. This tree algorithm model was trained using the test data set. To view the result of the training and the cross-validation accuracy estimation, a Test & Score widget was used. The data set provides input to the classification tree model while the Test & Score widget gets its input from both the data set file and the classification tree model as shown in Fig. 4.

The model training was tested by different evaluation results: classification accuracy, precision, recall, and F1 score. The classification accuracy (CA) computes the subset accuracy wherein the set of labels predicted for a sample must exactly match the corresponding set of labels in y_true. Precision is the ability of the classifier not to label as positive a sample that is negative. It is computed as (6):

Precision = (true positive) / (true positive + false positive) (6)

Recall is the ability of the classifier to find all the positive samples. It is computed as (7):

Recall = (true positive)/(true positive + false negative) (7)

F1 score is the weighted average of the precision and recall. F1 score is computed as (8):

F1 = (2 x (precision x recall)) / (precision + recall) (8)

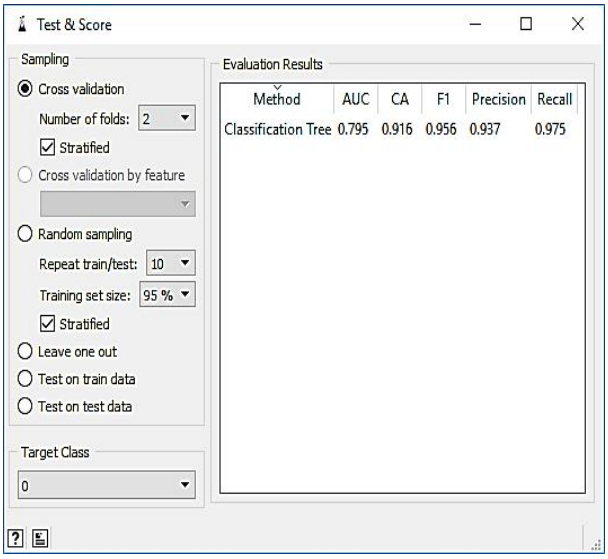


Fig. 4 Training and evaluation results of decision tree model

The training data set were sampled using cross-validation with multiple numbers of folds, random sampling with multiple repetition and leaving one out. Cross-validation is the average of values in a given loop or fold. Random sampling gets random samples from the training data set percentage of which was set an was repeated several times. Leave one out train on all data except for 1 record. Fig. 4 shows the actual training of the model and its corresponding evaluation results.

Table III shows the results of the entire test on different sampling used in the training data set targeting both 0 (healthy drive) and 1 (failed drive). The table shows that the decision tree lowest classification accuracy when targeting 0 (healthy drives) is 91.5% while it reaches up to a maximum of 93.4%. The precision of the classification tree on not mistaking false positive to true positive ranges from 93.3% to 94.1%. On the other hand, training accuracy on recall, that is its ability to determine all the positive samples sky-rocketed between 97.5% to 99.1%. F1 Score falls on the weighted average of precision and recall.

TABLE IIIIII
TRAINING ACCURACY RESULTS OF DECISION TREE ON VARYING PARAMETERS (TARGET CLASS: 0)

Sampling	Parameters	Target Class: 0 (in percentage)			
		CA	F1	Precision	Recall
Cross Validation	2	91.6	95.6	93.7	97.5
	3	91.7	95.6	93.7	97.6
	5	91.7	95.6	93.6	97.6
	10	91.5	95.5	93.5	97.5
	20	91.6	95.5	93.3	97.9
Random Sampling (95%)	2	93.4	96.5	94.1	99.1
	3	92.6	96.1	93.5	98.8
	5	93.1	96.3	94.0	98.8
	10	92.6	96.1	93.8	98.5
	20	92.5	96.0	93.9	98.3
Leave out one		91.8	95.6	93.4	98.0

Moreover, when targeting 1 (failed drive) as the result, its precision falls from 38.2% to 66.7% and the recall between 14.8% to 22.2% accuracy (Table IV). Same classification accuracy was recorded as when targeting 0 (healthy drive) as the result.

TABLE IVV
TRAINING ACCURACY RESULTS OF DECISION TREE ON VARYING PARAMETERS (TARGET CLASS: 1)

Sampling	Parameters	Target Class: 1 (in percentage)			
		CA	F1	Precision	Recall
Cross Validation	2	91.6	28.5	41.7	21.6
	3	91.7	28.1	41.9	21.1
	5	91.7	27.5	41.8	20.5
	10	91.5	25.5	38.9	18.9
	20	91.6	22.2	38.2	15.7
Random Sampling (95%)	2	93.4	33.3	66.7	22.2
	3	92.6	22.9	50.0	14.8
	5	93.1	32.3	58.8	22.2
	10	92.6	27.4	50.0	18.9
	20	92.5	28.3	48.6	20.0
Leave out one		91.8	23.8	41.3	16.8

D. ANFIS (Adaptive Neuro-Fuzzy Inference System)

In generating the Fuzzy Inference System (FIS) model, MatLab R2016a was used using its Neuro-Fuzzy Designer toolbox. The training data set (q1final.csv) was loaded as training data to the ANFIS. Values were drawn in the plot that shows the status of the hard drive as 0s and 1s (Figure 5).

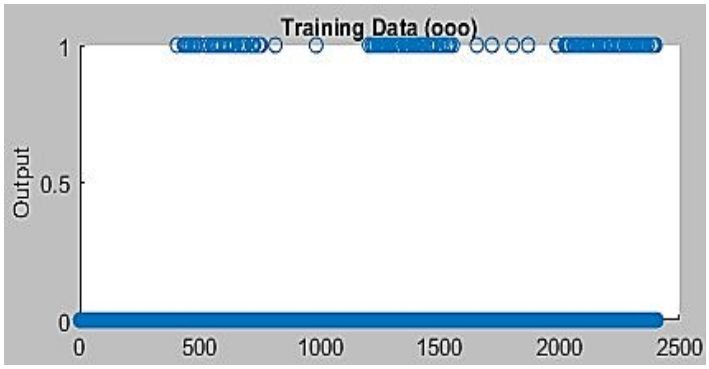


Fig. 5 Plotted hard drive status

After loading the data set in the Neuro-Fuzzy Designer, FIS (Fuzzy Inference System) model was generated. In clustering method, Grid Partitioning was used. Grid partitioning creates membership functions automatically by uniformly partitioning input variable ranges while creating a solitary output. The number of membership function used is 2 for each input which represents 1 for failed drive and 0 for healthy ones. Psigmf function or the Product of Two Sigmoidal membership function was used as the membership function type and the output as constant in type.

After setting all the parameters, the FIS model was generated (Figure 6). From 9 input attributes, each value will go through the membership functions, 2 for each input. Values will then be tested on generated rules to produce an output membership functions before finally generating the output. With this model, there are 512 rules where each record in the training data set will pass through.

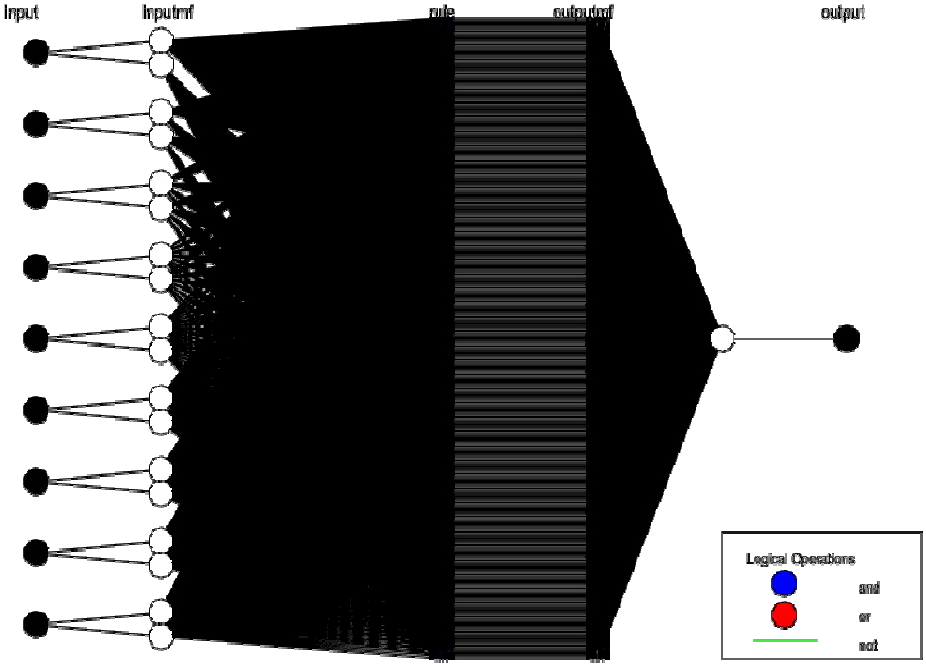


Fig. 6 FIS Model

In training the model, hybrid optimization was used. Hybrid optimization was used to tune the parameters of the FIS. Looping through 73 epochs, the FIS model achieves a final error rate of 0.227914 (22.79%) from initial 0.26652 (26.65%). A steady downfall of error rate was observed until reaches 73 epochs where error stabilized at 0.227914 (22.79%). Fig. 7 shows the error rate of FIS for each epoch.

To check for the actual accuracy of the training and to compare the exact value FIS generated against the drive status (1, 0), the FIS model was exported to the Matlab Workspace. The training data set, with the drive status re-moved, was also imported to the Matlab Workspace. It was then evaluated using evalfis() function of the Matlab which generates the actual output value based on the

training data set. The output was pasted parallel to the drive status column of the training data set to compare how close the values are to 0 and 1.

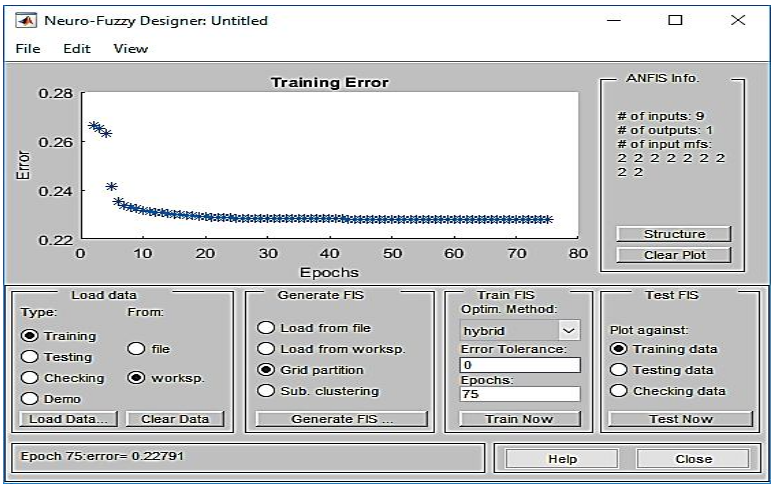


Fig. 7 Error rates of FIS model

In getting the accuracy percentage of each record, if the drive status is 0 and the actual value generated by FIS is 0.161 it means that it is 83.90% accurate. It was computed using the formula:

$$\text{Accuracy} = 100 - (\text{Actual Value} * 100) \quad (9)$$

However, if the drive status is 1, and the actual value is 0.8137, it just multiplied by 100, thus;

$$\text{Accuracy} = \text{Actual Value} * 100 \quad (10)$$

Going through all the records, accuracy values were averaged. ANFIS garnered a training accuracy of 89.29%. Comparing the training accuracy of Decision Tree with FIS, the training accuracy of the FIS model is 9.81% lower. Although FIS model achieves lower result than Decision Tree, it absolutely confirmed and proved the hypothesis of the research and validated the claim of the related literature that decision tree tends to overfit when used with this types of data sets. It also affirms the ANFIS algorithm performs way better than previous studies on hard disk failure prediction.

E. Testing of Models and Evaluation of Results

1) *Decision Tree:* With the decision tree model trained using training data set, test data set will be subjected to see how decision tree will predict the hard disk failure. Using the earlier workflow, a test data file loaded with the test data set was placed and connected to a Prediction widget together with the learner and model output of the Decision Tree algorithm widget (Fig. 8). The Prediction widget displays the evaluation results and prediction of the algorithm attached to it.

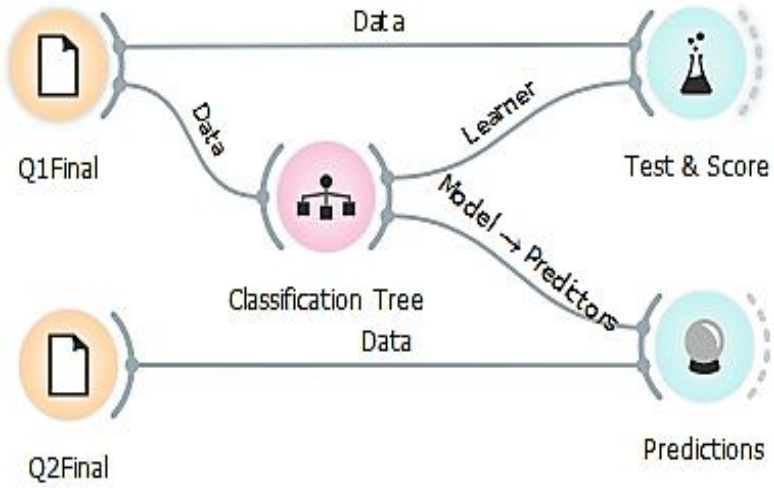


Fig. 8 Decision tree testing workflow

Looking at the result of the prediction, Decision Tree algorithm classified / predicted healthy drives (0) by an impressive 92%, however, it never classified a failed drive (1) correctly, not even once. It is also disturbing that all classification renders the same value as shown in Fig. 9.

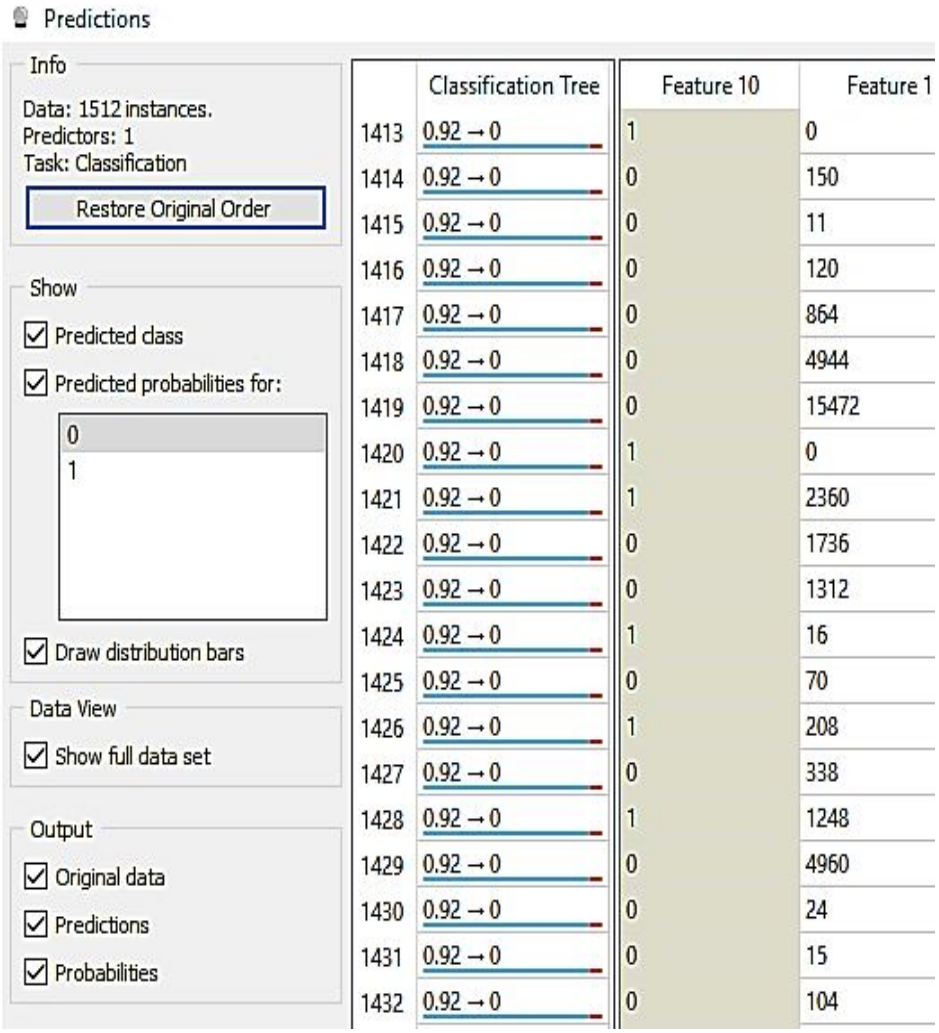


Fig. 9 Cut out from the decision tree evaluation result

If overall computation will be on the percentage decision tree predicted that the drive is healthy, it predicted 1,346 correctly out of 1512 records. That is 89.02% accuracy. Specifically, the decision can detect healthy drives by 92% and failed hard drive by 8%. However, the percentage value is misleading because 1,346 or 89.02% is actually the number of hard disk drive with status 0 out of 1512 records. It can, therefore, be ruled out that the prediction of decision tree using this kind of data set is unpredictable as well as unreliable and misleading.

F. ANFIS (Adaptive Neuro-Fuzzy Inference System

Having the same process of finding the actual accuracy of the FIS model, test data set (hard disk status column re-moved) was loaded in the workspace of Matlab. Using the *evalfis()* function, values were generated and also copied paralleled to the original test data set (Q2Final.csv) file. Us-ing the same formula during training, prediction or classification accuracy reaches 86.08% which is higher by 4.2% compared to the work of Queiroz that is 81.88%; the highest accuracy percentage on previous researches on hard drive failure. Specifically, ANFIS has 91.53% accuracy on detecting healthy drives and 41.88% accuracy on detecting failed hard drive. The lower accuracy on the detection of the failed hard drive can be easily ruled out due to the fact the ratio healthy hard drive to failed hard drive is 89.02:10.98. Also, the accuracy of ANFIS in detecting failed hard drive is way much higher than the decision tree by 41.08%.

To further evaluate the effectiveness of the ANFIS mod-el, records of a particular hard drive that failed and was removed due to its failure state was filtered. Its record com-prises of smart attributes, date and its hard drive from its healthy stage to failed stage. Table V shows the values of each column for the month of June. Its record from January to May was not retrieved because the values in these previous months are also the same with the records on June 1-3, 2017, except for its initial value in March 8, 2017 wherein all columns are all zero and March 9, 2017, which starts the record same as June 1-3, 2017. The last record was the record when the hard disk failed and was removed from the storage pod on June 11, 2017.

TABLE V
FILTERED RECORD FOR ANFIS EVALUATION

Date	SMART 5	SMART 10	SMART 184	SMART 187	SMART 188	SMART 196	SMART 197	SMART 198	SMART 201	Failure	ANFIS Prediction
3/8/2017	0	0	0	0	0	0	0	0	0	0	0
3/9/2017	0	0	0	3	0	0	0	0	0	0	0
6/1/2017	0	0	0	3	0	0	0	0	0	0	16.75
6/2/2017	0	0	0	3	0	0	0	0	0	0	16.75
6/3/2017	0	0	0	3	0	0	0	0	0	0	16.75
6/4/2017	0	0	0	3	0	0	32	32	0	0	16.75
6/5/2017	0	0	0	3	0	0	32	32	0	0	32.64
6/6/2017	0	0	0	3	0	0	32	32	0	0	32.64
6/7/2017	0	0	0	3	0	0	72	72	0	0	60.13
6/8/2017	0	0	0	3	0	0	88	88	0	0	66.26
6/9/2017	0	0	0	3	0	0	120	120	0	0	74.22
6/10/2017	0	0	0	3	0	0	144	144	0	0	77.95
6/11/2017	8	0	0	4	0	0	232	232	0	1	90.51

In this result, the ANFIS predicted that the drive will soon fail starting March 9, 2017 and the values were increasing as the days passes by. By June 7, 2017, the ANFIS model was sure that it will definitely fail anytime sooner because the percentage reaches more than 50%. The date it was predicted that it is more likely to fail is 5 days prior to its actual failure. Figure 10 shows the steady climb of the percentage of the possibility of the failure.

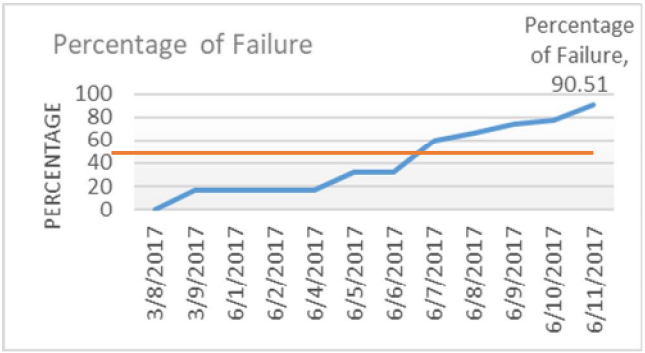


Fig. 10 Slope on the prediction of the before an actual failure happens.

IV. CONCLUSIONS AND RECOMMENDATIONS

Utilizing the training data set, the decision tree achieves an average classification accuracy of 92.45%, precision ranges from 93.3% to 94.1% and its recall at 97.5% to 99.1%. This findings confirm and prove the hypothesis of the research and validated the claim of related literature that decision tree really overfit and is unreliable when it comes to real-type and continuous data such as SMART attributes. ANFIS model, on the other hand, after going through all the records reaches achieves 89.29% accuracy. Comparing with the training accuracy of decision tree, FIS mode is 9.81% lower but did not overfit.

Using the test data set, decision managed to classify healthy drives by an impressive 89.02% (1,346 predicted correctly out of 1,512 records), however, it never classified a failed drive correctly, not even once. This disturbing result was found out by carefully examining the test result. All hard drive records were classified as healthy by 89.02% which is actually exact ratio of the percentage or number of healthy is to the failed hard drive. It can therefore be ruled out that the prediction of decision tree using this kind of data set is unpredictable as well as unreliable and misleading. Oppositely, during testing, ANFIS was able to reach 86.08% classification accuracy. Although ANFIS model achieves lower result than decision tree, it surpasses accuracy results of known researches on data drive failure using other algorithms by 4.2% (compare to the work of Queiroz, et. Al. at 81.88%, being the highest) proving that by far, ANFIS is the best among them.

In the evaluation of the effectiveness of the ANFIS, selected record has been subjected. ANFIS was able to predict the drive failure 5 days before the actual failure occurred.

Although this research displays promising result, it is recommended that newer data set be used to further con-firm the validity of this study. It is also suggested that data set from other sources and a larger number of records be used as training and test data sets. It is also highly recommended that the number of failed hard drive be at least equal to the number of healthy hard drive to have a more accuracy classification on both outcomes. Further study in this area is suggested and other tools can be utilized to strengthen the facts and background information for future researches. Other SMART attributes can also be considered and tested to create new knowledge in the area.

V. ACKNOWLEDGMENT

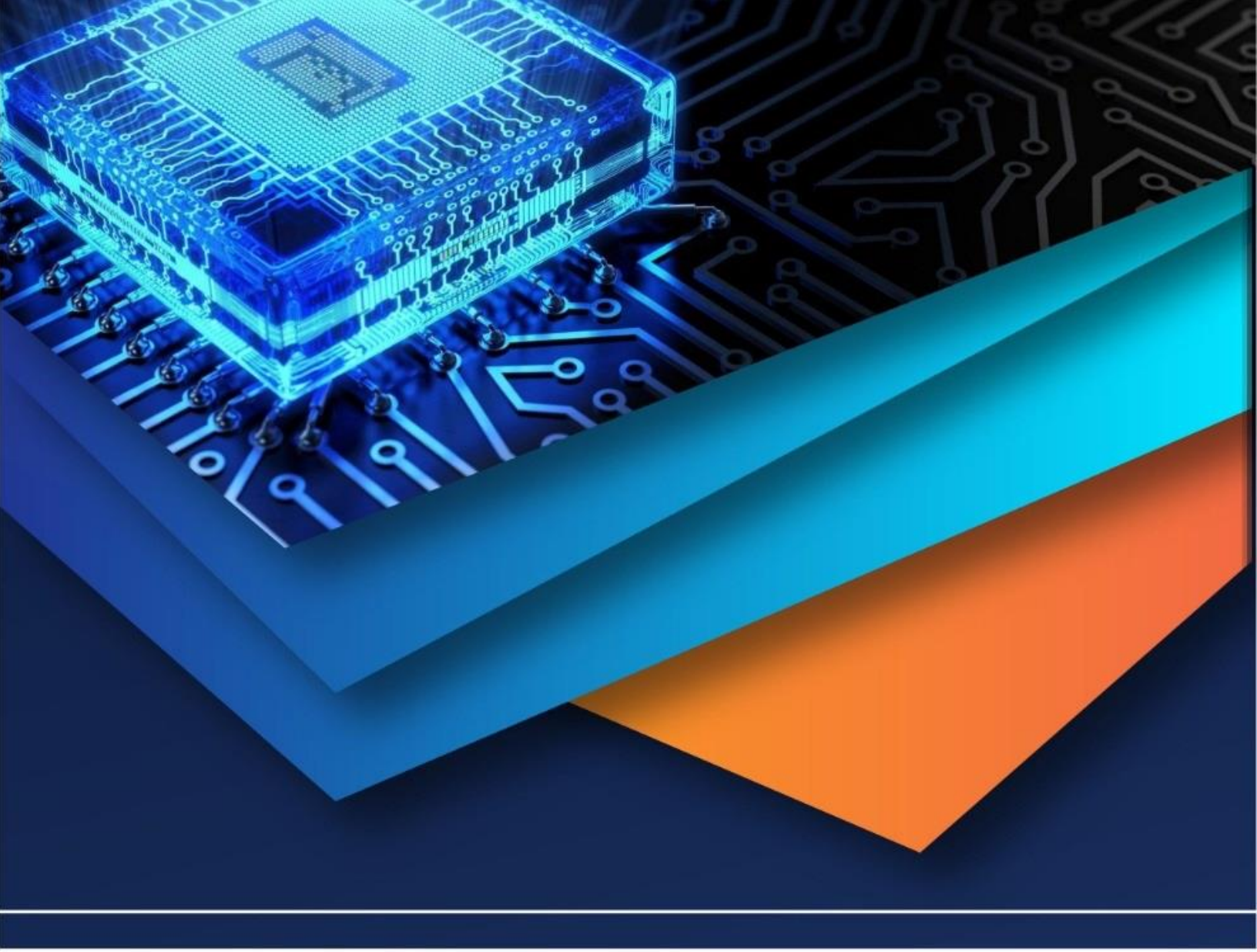
J. A. Olalia would like to acknowledge the financial support provided by the Commission on Higher Education, Kto12 Project Management Unit, Philippines.

REFERENCES

[1] General Electric, “Five Essential Components for Highly Reliable Data Centers,” p. 6, 2011.
[2] M. Schafer, “Executive Report Seven Critical Success Factors for Choosing a Reliable Data Center Provider,” 2016.
[3] Manila Times, “IT study: Data loss, downtime cost PH firms $8B yearly,” 2014.
[4] Dell, “Over $1.7 Trillion Lost Per Year from Data Loss and Downtime According to Global IT Study,” 2014. [Online]. Available: https://www.emc.com/about/news/press/2014/20141202-01.htm.
[5] D. EMC, “Data Loss : Understanding the Causes and Costs,” 2014.
[6] DataBarracks, “Data Health Check 2017,” 2017.
[7] T. Suchatpong and K. Bhumkittipich, “Hard Disk Drive failure mode prediction based on industrial standard using decision tree learning,” Electr. Eng. Comput. Telecommun. Inf. Technol. (ECTI-CON), 2014 11th Int. Conf., pp. 1–4, 2014.
[8] M. M. Botezatu, I. Giurgiu, J. Bogojeska, and D. Wiesmann, “Predicting Disk Replacement towards Reliable Data Centers,” Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. - KDD ’16, no. 1, pp. 39–48, 2016.
[9] T. Tsai, N. Theera-Ampornpunt, and S. Bagchi, “A study of soft error consequences in hard disk drives,” IEEE/IFIP Int. Conf. Dependable Syst. Networks (DSN 2012), pp. 1–8, 2012.
[10] V. Agrawal, C. Bhattacharyya, T. Niranjan, and S. Susarla, “Discovering rules from disk events for predicting hard drive failures,” 8th Int. Conf. Mach. Learn. Appl. ICMLA 2009, pp. 782–786, 2009.
[11] I. C. Chaves, M. R. P. De Paula, L. G. M. Leite, L. P. Queiroz, J. P. P. Gomes, and J. C. Machado, “BaNHFaP : A Bayesian Network based Failure Prediction Approach for Hard Disk Drives,” 2016 5th Brazilian Conf. Intell. Syst., 2016.
[12] L. P. Queiroz et al., “A fault detection method for hard disk drives based on mixture of gaussians and nonparametric statistics,” IEEE Trans. Ind. Informatics, vol. 13, no. 2, pp. 542–550, 2017.
[13] J. F. Puget, “Overfitting In Machine Learning,” IBM Developer Blogs, 2016. [Online]. Available: https://www.ibm.com/developerworks/community/blogs/jfp/entry/Overfitting_In_Machine_Learning?lang=en.
[14] H. Yang, A. Xu, H. Chen, and C. Yuan, “A review: The effects of imperfect data on incremental decision tree,” Proc. - 2014 9th Int. Conf. P2P, Parallel, Grid, Cloud Internet Comput. 3PGCIC 2014, pp. 34–41, 2014.
[15] M. Somvanshi and P. Chavan, “A review of machine learning techniques using decision tree and support vector machine,” 2016 Int. Conf. Comput. Commun. Control Autom., pp. 1–7, 2016.
[16] Y. Zhao and Y. Zhang, “Comparison of decision tree methods for finding active objects,” Adv. Sp. Res., vol. 41, no. 12, pp. 1955–1959, 2008.
[17] D. Sontag, “Decision Trees Lecture.” New York University, 2016.
[18] Anuradha and G. Gupta, “A self explanatory review of decision tree classifiers,” Int. Conf. Recent Adv. Innov. Eng. ICRAIE 2014, no. 3, 2014.
[19] NTC, “Cloud Storage,” Nonprofit Techonology Collaboration. pp. 3–6, 2013.
[20] B. Casemore, “Enabling Digital Transformation in Datacenters and Hybrid Cloud : Cisco Analyze , Simplify , Automate , and Protect (ASAP),” no. November 2016, 2016.
[21] RightScale, “State of the Cloud Report: Public Cloud Adoption Grows as Private Cloud Wanes,” 2017.
[22] Seagate, “Data Centre Management: Trends and Challenges,” 2013. [Online]. Available: http://www.seagate.com/as/en/tech-insights/data-center-management-master-ti/.
[23] L. Scott, J. Green, and D. Davis, “Building a Modern Data Center: Principles and Strategies of Design,” 2016.

[24] A. Klein, "One Billion Drive Hours and Counting: Q1 2016 Hard Drive Stats," 2016. [Online]. Available: https://www.backblaze.com/blog/hard-drive-reliability-stats-q1-2016.
[25] B. Schroeder and G. a. Gibson, "Disk failures in the real world: What does an MTTF of 1,000,000 hours mean to you," Conf. File Storage Technol., pp. 1–16, 2007.
[26] Z. S. Ye, M. Xie, and L. C. Tang, "Reliability evaluation of hard disk drive failures based on counting processes," Reliab. Eng. Syst. Saf., vol. 109, pp. 110–118, 2013.
[27] Emerson, "Emerson Network Power Study Says Unplanned Data Center Outages Cost Companies Nearly $9,000 Per Minute," 2016. [Online]. Available: http://www.emerson.com/en-us/news/corporate/network-power-study.
[28] E. Pinheiro, W. Weber, and L. Barroso, "Failure trends in a large disk drive population," Proc. 5th USENIX Conf. File Storage Technol. (FAST 2007), no. February, pp. 17–29, 2007.
[29] Calomel, "Using S.M.A.R.T. to Monitor Hard Drive Health," 2017. [Online]. Available: https://calomel.org/smart_hd_status.html.
[30] L. Mearian, "The 5 SMART stats that actually predict hard drive failure," 2014. [Online]. Available: http://www.computerworld.com/article/2846009/the-5-smart-stats-that-actually-predict-hard-drive-failure.html.
[31] Botezatu, et. al., 2016
[32] L. P. Queiroz et al., "A fault detection method for hard disk drives based on mixture of gaussians and nonparametric statistics," IEEE Trans. Ind. Informatics, vol. 13, no. 2, pp. 542–550, 2017.
[33] Y. Wang, E. W. M. Ma, T. W. S. Chow, and K. L. Tsui, "A two-step parametric method for failure prediction in hard disk drives," IEEE Trans. Ind. Informatics, vol. 10, no. 1, pp. 419–430, 2014.
[34] Y. Wang, Q. Miao, E. W. M. Ma, K. L. Tsui, and M. G. Pecht, "Online anomaly detection for hard disk drives based on mahalanobis distance," IEEE Trans. Reliab., vol. 62, no. 1, pp. 136–145, 2013.
[35] B. Li, S. C. H. Hoi, P. Zhao, and V. Gopalkrishnan, "Datacenter Scale Evaluation of the Impact of Temperature on Hard Disk Drive Failures," Proc. 14th Int. Conf. Artif. Intell. Stat., vol. 15, no. 212, pp. 434–442, 2013.
[36] J. F. Puget, "Overfitting In Machine Learning," IBM Developer Blogs, 2016. [Online]. Available: https://www.ibm.com/developerworks/community/blogs/jfp/entry/Overfitting_In_Machine_Learning?lang=en.
[37] J. Brownlee, "Overfitting and Underfitting With Machine Learning Algorithms," Machine Learning Mastery, 2016. [Online]. Available: https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/.
[38] Yang, et. al., 2014
[39] Somyansi & Chavan, 2016
[40] Y. Zhao and Y. Zhang, 2016
[41] Zontag, 2016
[42] Scikit, "Decision Trees," Scikit Learn, 2016. [Online]. Available: http://scikit-learn.org/stable/modules/tree.html.
[43] E. Alpaydin, "Decision Trees," Introd. to Mach. Learn., no. Cluj Napoca, pp. 185–208, 2010.
[44] Anuradha and G. Gupta, 2014
[45] J. S. R. Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System," IEEE Trans. Syst. Man Cybern., vol. 23, no. 3, pp. 665–685, 1993.
[46] A. Eseye, J. Zhang, D. Zheng, and G. Jingfu, "Short-Term Wind Power Forecasting Using a Double-Stage Hierarchical Hybrid GA-ANFIS Approach," 2017 IEEE 2nd Int. Conf. Cloud Comput. Big Data Anal., pp. 499–503, 2017.
[47] J. Peng, S. Gao, and A. Ding, "Study of the Short-Term Electric Load Forecast Based on ANFIS," 2017 32nd Youth Acad. Annu. Conf. Chinese Assoc. Autom., vol. 2015, pp. 832–836, 2017.
[48] A. N. Averkin, "Hybrid Approach for Time Series Forecasting Based on ANFIS and Fuzzy Cognitive Maps," 2017 IEEE Int. Conf. Soft Comput. Meas., no. 17, pp. 379–381, 2017.
[49] Z. Amekraz, "Prediction of Amazon spot price based on chaos theory using ANFIS model," 2016 IEEE/ACS 13th Int. Conf. Comput. Syst. Appl., pp. 1–6, 2016.
[50] G. Georgiev, I. Balabanova, S. Kostadinova, and R. Dimova, "Structure synthesis of ANFIS classifier for teletraffic system resources identification," 2016 IEEE Int. Black Sea Conf. Commun. Networking, BlackSeaCom 2016, 2017.
[51] F. Nhita, D. Saepudin, D. Triantoro, Adiwijaya, and U. N. Wisesty, "Implementation of Moving Average and Soft Computing algorithm to support planting season calendar forecasting system on mobile device," Proceeding - 2016 2nd Int. Conf. Sci. Inf. Technol. ICSITech 2016 Inf. Sci. Green Soc. Environ., pp. 114–118, 2017.
[52] D. Kofinas, E. Papageorgiou, C. Laspidou, N. Mellios, and K. Kokkinos, "Daily multivariate forecasting of water demand in a touristic island with the use of artificial neural network and adaptive neuro-fuzzy inference system," 2016 Int. Work. Cyber-physical Syst. Smart Water Networks, CySWater 2016, pp. 37–42, 2016.
[53] S. Akkoc, "An empirical comparison of conventional techniques, neural networks and the three stage hybrid Adaptive Neuro Fuzzy Inference System (ANFIS) model for credit scoring analysis: The case of Turkish credit card data," Eur. J. Oper. Res., vol. 222, no. 1, pp. 168–178, 2012.
[54] I. Alam, M. A. Ansari, and N. S. Pal, "A Comparative Study between Wavelet Primarily Based ANN and ANFIS Algorithm Technique to Locate Fault in a Transmission Line," 1st IEEE Int. Conf. Power Electron. Intell. Control Energy Syst., pp. 1–6, 2016.
[55] S. A. Khan, M. D. Equbal, and T. Islam, "A comprehensive comparative study of DGA based transformer fault diagnosis using fuzzy logic and ANFIS models," IEEE Trans. Dielectr. Electr. Insul., vol. 22, no. 1, pp. 590–596, 2015.
[56] H. H. Çevik and M. Çunkas, "A Comparative Study of Artificial Neural Network and ANFIS for Short Term Load Forecasting," ECAI 2014 - Int. Conf. – 6th Ed. Electron. Comput. Artif. Intell., pp. 0–5, 2014.
[57] A. Kaboli, M. H. Savoji, and E. Square, "Non-Linear Prediction Of Speech Using ANFIS : Comparison With Neural Nets .," Comput. Eng., 2004.
[58] Z. J. Viharos and K. B. Kis, "Survey on Neuro-Fuzzy systems and their applications in technical diagnostics and measurement," Meas. J. Int. Meas. Confed., vol. 67, pp. 126–136, 2015.

[59] MathWorks, "Improve Neural Network Generalization and Avoid Overfitting," MathWorks Documentation, 2017. [Online]. Available: https://www.mathworks.com/help/nnet/ug/improve-neural-network-generalization-and-avoid-overfitting.html.
[60] A. Singh, "Neural Networks," Mach. Learn., 2010.
[61] A. Nowak-Brzezinska, "Artificial neural network," Uniwersytet Slaski, 2006. [Online]. Available: zsi.ii.us.edu.pl/~nowak/bien/w7.pdf.
[62] S. J. S. Hakim and H. Abdul Razak, "Adaptive neuro fuzzy inference system (ANFIS) and artificial neural networks (ANNs) for structural damage identification," Struct. Eng. Mech., vol. 45, no. 6, pp. 779–802, 2013.
[63] J. Mahanta, "Introduction to Neural Networks, Advantages and Applications," Medium Corporation, 2017. [Online]. Available: https://medium.com/towards-data-science/introduction-to-neural-networks-advantages-and-applications-96851bd1a207.
[64] U. C. Moon, H. Jang, and M. V. Dos Santos, "An application of a neuro-fuzzy inference system for pattern classification of HDD defect distribution," Annu. Conf. North Am. Fuzzy Inf. Process. Soc. - NAFIPS, 2010.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY