Assignment 4: Module Order Modeling (MOM)

Hector Lopez CAP6673 FAU SPRING 2018

Introduction

Module Order Modeling (MOM) can be used to help focus efforts toward the least reliable modules and increase early detection of these less reliable modules resulting in reliability enhancement of fault-prone modules. In previous assignments, fault-prone (fp) and not fault-prone (nfp) are used as classifications to indicate the modules recommended for reliability enhancement given some threshold of faults to indicate one classification or the other.

A MOM classifier is used with the fault and non-fault prone classifications by ordering the actual faults based on two provided regression models, both training and validation, i.e. test and fit, datasets. MOM, is used as a classifier for classes {fp, nfp} with results indicating a range of possible values to best maximize the early detection of faults without encompassing all possible values.

The actual faults and classes are known thus are used to inform and create the MOM classifiers. A threshold for the number of faults classifying fault-prone modules is known, which is the same threshold used for assignments one through three, this does not necessarily indicate an optimal threshold for determining how many modules to assess for reliability to meet necessary valuation indicators. The threshold used may not even be known. Module Order Modeling can be used to create and quantify these thresholds.

MOM is a quantitative model that predicts rank-order of modules based on the number of faults. MOM can be used to establish thresholds within the dataset for determining what should be considered fp and nfp based on context and resources. So each threshold percentile can be seen as a separate classifier, for instance, thresholds from 95% to 50% of fault ordered datasets were used. The largest inclusion of data is assumed to be 50% because if resources are scarce for reliability enhancements, then it would be unreasonable to assume that 50% of all modules will be enhanced [3].

This rank-ordering only approach assumes a relative view on module quality in that anything, regardless of actual class, above the threshold is considered fault-prone while below the threshold is not fault-prone with MOM. For MOM, in this case, there is no *a priori*, or already known, value to indicate what should be considered the criteria for a fault-prone module so each threshold is the classification. There are two major parts in creating a MOM: the application of a modeling algorithm to predict a dependent variable based on provided independent variables and using these predicted values to order or rank the actual values. A classification can be made based on the same threshold for actual number of faults to indicate a fault-prone or not fault-prone class.

The previous assignments used the number of faults >= 2 to describe fp, otherwise nfp. Table 1 shows the first ten rows of the actual dataset used with corresponding class ordered from greatest to least number of faults. Table 2 is a sample, the first 19 rows, with the actual faults and the predicted faults based on multivariate linear regression and M5 variable selection on the training set. The class is listed as well for both actual and predicted ranking. Notice that in Table 2, the prediction is used to order the actual faults and corresponding class.

Actual class	Actual fault count
Fp	29
Fp	29
Fp	22
Fp	20
Fp	16
Fp	15
Fp	14
Fp	13
Fp	12
Fp	12

Table 1 – actual data

Actual faults	Predicted faults (M5)	Predicted actual faults	Actual ordered faults
29	24.8342	fp	Fp
20	23.0534	fp	Fp
29	17.9212	fp	Fp
8	14.3045	fp	Fp
11	14.1271	fp	Fp
12	13.9218	fp	Fp
22	12.9097	fp	Fp
13	12.4276	fp	Fp
10	11.8135	fp	Fp
2	11.5413	fp	Fp
8	11.2312	fp	Fp
15	11.0025	fp	Fp
7	9.7127	fp	Fp
14	9.3348	fp	Fp
10	8.7199	fp	Fp
2	8.673	fp	Fp
16	8.2117	fp	Fp
8	7.7829	fp	Fp
12	6.9356	fp	Fp

Table 2 – prediction and predictive ordering of actual data (sample from training set)

The predicted values in Table 2 are in red because they are only used for ordering, not for fault classification. Notice also that even if the predicted ordering isn't perfect, i.e. matches the actual ordering, the classification is still correct. For instance, row four indicates an actual fault count of 8, which is clearly not correctly ordered, but it is still classified correctly as fp. Once the predicted order is established, several evaluations can be made to determine MOM accuracy, misclassification rates, and model performance and robustness. The general process for evaluating MOM can be described as follows [2]:

1. Rank the actual faults from greatest to least
2. Rank the predicted faults from greatest to least used to order actual faults
3. Choose cut-off points or thresholds, C, based on the predicted order indicating the selected values for reliability enhancement, with counts above the threshold indicating fp modules
4. Sum the actual faults, $G(c)$, for the actual fault ordering above the threshold described in step 3, where $c \in C$
5. Sum the actual faults, $\hat{G}(c)$, for the predicted fault ordering above the threshold described in step 3, where $c \in C$
6. Calculate the percentage of the total faults based on fp counts in steps 4 and 5, where:

$$^{G(c)}/_{F_{total}} \; where\ F\ is\ the\ total\ count\ of\ fp$$

$$^{\hat{G}(c)}/_{F_{total}} \; where\ F\ is\ the\ total\ count\ of\ fp$$

7. Calculate the ratio of predicted faults to actual faults indicating how closely the predictive ordering count of fp follows that of the actual fp ordering

$$\emptyset(c) = \frac{\hat{G}(c)}{G(c)} \; if\ \emptyset(c) = 1\ then\ the\ prediction\ matches\ the\ actual$$

Once the evaluation of MOM is complete, then the choice rests in those responsible for the resources and results pertaining to module reliability. MOM allows for the flexibility to choose which representation of fault-prone is considered reasonable given common business constraints and resources.

Results and Evaluation

The ordering of the modules is based on the number of faults as discussed using a standard prediction model. The fault predictions are done with both test and fit datasets. MOM is then created per the dataset and prediction model, then evaluated and analyzed. There are two predictive models considered for test and fit datasets to create corresponding MOM for each.

Predictive Models

Module Order Modeling

The evaluation criteria were applied to each instance of MOM over the regression models and datasets. The values of c are selected by partitioning the dataset into roughly even break points as seen in Table 2, which highlights 95% and 90% as examples. These break points are used to assess fault-prone and not fault-prone modules so at or above 95% shows all fp modules and no nfp modules. This process is continued to the break point at a 50% percentile. With the fp counts known, the performance of MOM can be estimated for all thresholds with which the model recommends reliability enhancements on predicted fault-prone modules. Tables 5 and 6 summarize the numbers to evaluate MOM on the training or fit dataset with Figures 4a, 4b, 5a, and 5b the corresponding Alberg Diagrams and performance graphs.

C	break point	# actual fp	# actual nfp	# pred fp	# pred nfp	$G(c)$	$G'(c)$	$\emptyset(c)$
95%	9	9	0	9	0	16.36%	16.36%	100.00%
90%	19	19	0	19	0	34.55%	34.55%	100.00%
85%	28	28	0	26	2	50.91%	47.27%	92.86%
80%	38	38	0	33	5	69.09%	60.00%	86.84%
75%	47	47	0	40	7	85.45%	72.73%	85.11%
70%	56	55	1	43	13	100.00%	78.18%	78.18%
65%	66	55	11	47	19	100.00%	85.45%	85.45%
60%	75	55	20	49	26	100.00%	89.09%	89.09%
55%	85	55	30	50	35	100.00%	90.91%	90.91%
50%	94	55	39	51	43	100.00%	92.73%	92.73%

Table 5 – summary statistics for each MOM classifier using lm M5 for fault prediction with training data

C	break point	# actual fp	# actual nfp	# pred fp	# pred nfp	$G(c)$	$G'(c)$	$\emptyset(c)$
95%	9	9	0	9	0	16.36%	16.36%	100.00 %
90%	19	19	0	19	0	34.55%	34.55%	100.00 %
85%	28	28	0	25	3	50.91%	45.45%	89.29%
80%	38	38	0	34	4	69.09%	61.82%	89.47%
75%	47	47	0	40	7	85.45%	72.73%	85.11%
70%	56	55	1	43	13	100.00%	78.18%	78.18%
65%	66	55	11	46	20	100.00%	83.64%	83.64%
60%	75	55	20	49	26	100.00%	89.09%	89.09%
55%	85	55	30	50	35	100.00%	90.91%	90.91%
50%	94	55	39	51	43	100.00%	92.73%	92.73%

Table 6 – summary statistics for each MOM classifier using lm Greedy for fault prediction with training data

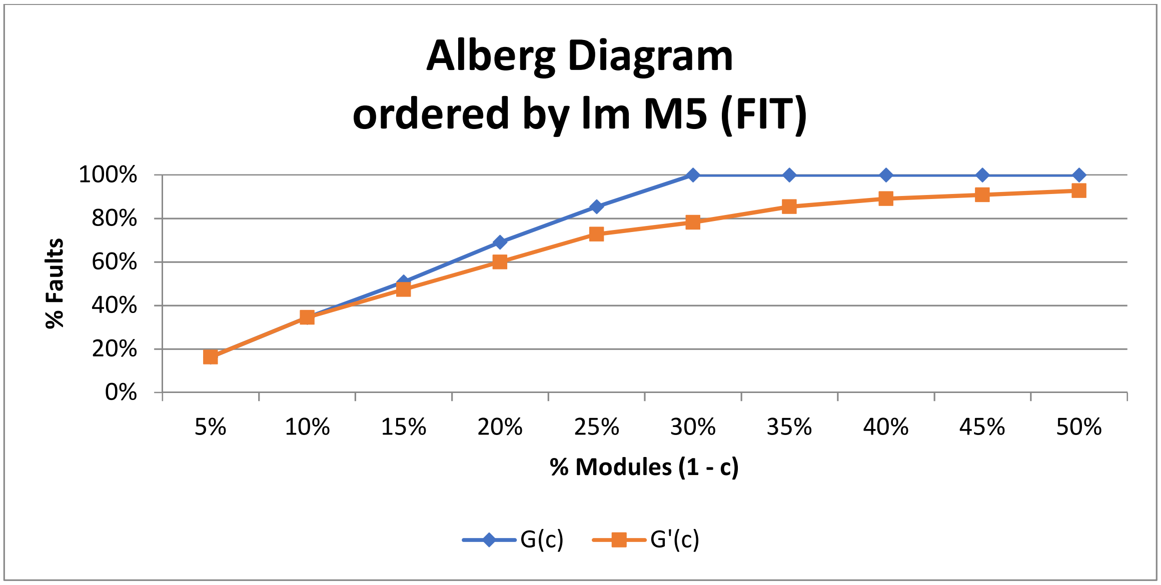

Figure 4a – Alberg Diagram for linear regression model with M5 selection on training set

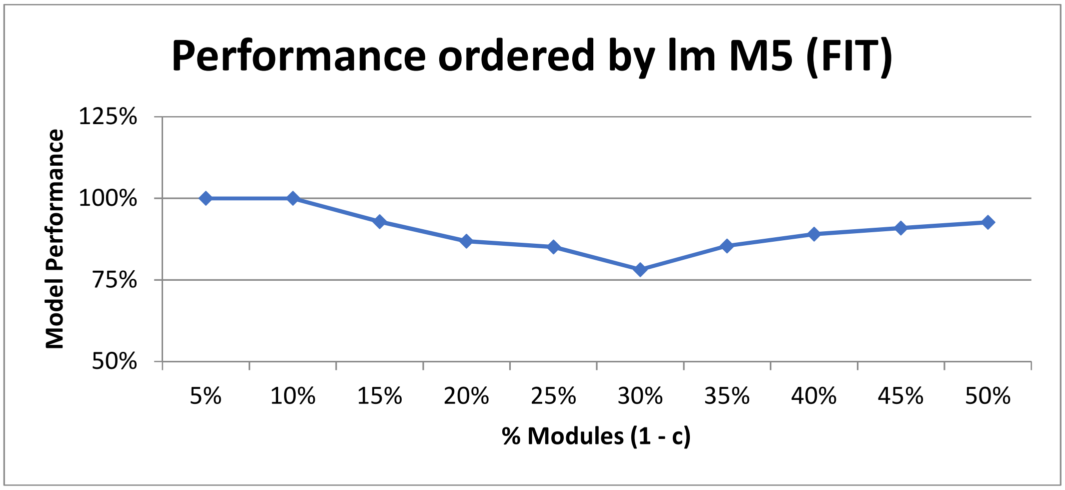

Figure 4b – Performance graph for linear regression model with M5 selection on training set

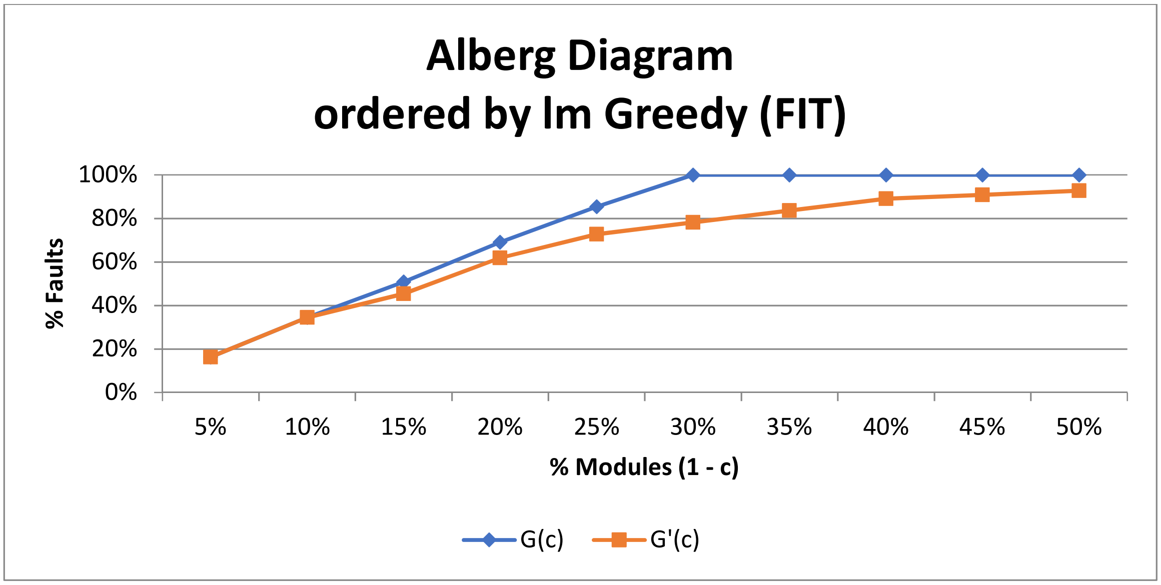

Figure 5a – Alberg Diagram for linear regression model with Greedy selection on training set

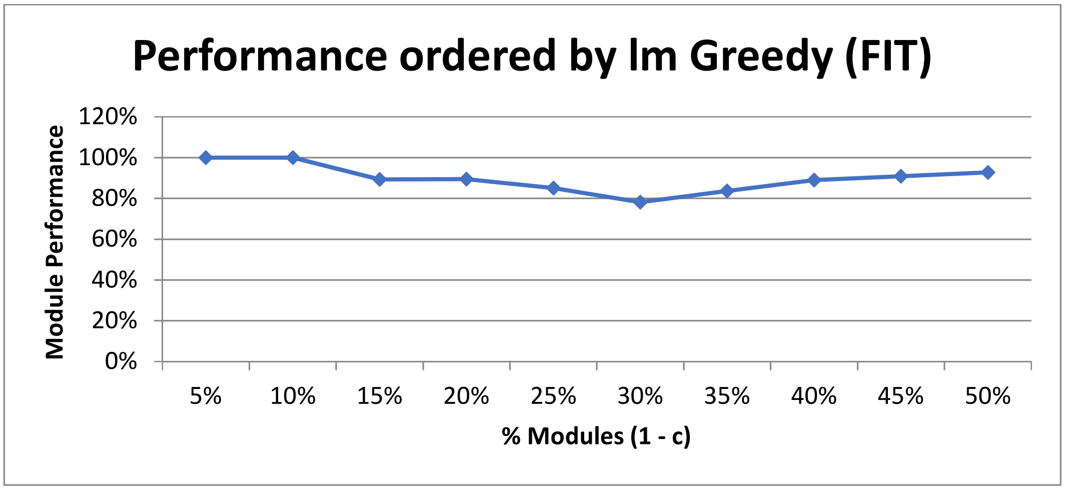

Figure 5b – Performance graph for linear regression model with Greedy selection on training set

The linear regression models with M5 and Greedy variable selection methods, as anticipated, perform very similarly creating similar MOM classifiers. The Alberg Diagrams, Figures 4a and 5a, show the cumulative number of faults for the actual and predicted values.

This shows the threshold percentile and what percentage of faults this threshold is predicted to cover. For instance, from Figure 4a, at the most fault-prone 5% (c=95%) of the modules recommended for reliability improvements, the model accounts for 16.36% of all faults. This means MOM accounts for 100% ($\emptyset(\mathrm{c}) = 100\%$**)** of the faults that perfect (or actual) ordered values would indicate with c=95%.

So, improving 5% of faulty modules would garner up to a 16% early detection of faults, which may be good enough pending resources, but a higher threshold value should be considered. A modest increase to c=90%, ergo 10% of modules, increases the fp reliability enhancements to 34.55% for which MOM accounts for 100% of faults that a perfect recommendation would account for.

Based on these MOM classifiers, there are viable options for useful predictive modeling and early detection of faults without the need to look at all modules. Model performance, ($\emptyset(c)$, is graphed in Figures 4b and 5b indicating the accuracy of the model over the classifier thresholds. The variation over the threshold range, points to the robustness of MOM with less variation being more robust.

Tables 7 and 8 show the same evaluation statistics for the test or validation dataset as with the above training set tables with Figures 6a, 6b, 7a, and 7b the corresponding Alberg Diagrams and performance graphs. The MOM classifiers on the test or validation set are comparable to those created with the training set, but are not as robust or accurate. Both underlying linear models created similar MOM classifiers. At c=90%, MOM accounts for 32.14% of faults rather than 34.55% for the training set. This is a relatively small delta but as seen in Figures 6a and 7a, the delta between the numbers of faults predicted versus actual faults become wider. The performance plots in Figures 6b and 7b show additional variability thus indicating less robust MOM classification models versus using the training set.

C	break point	# actual fp	# actual nfp	# pred fp	# pred nfp	$G(c)$	$G'(c)$	$\emptyset(c)$
95%	5	5	0	5	0	17.86%	17.86%	100.00%
90%	9	9	0	9	0	32.14%	32.14%	100.00%
85%	14	14	0	13	1	50.00%	46.43%	92.86%
80%	19	19	0	17	2	67.86%	60.71%	89.47%
75%	24	24	0	19	5	85.71%	67.86%	79.17%
70%	28	28	0	19	9	100.00%	67.86%	67.86%
65%	33	28	5	21	12	100.00%	75.00%	75.00%
60%	38	28	10	23	15	100.00%	82.14%	82.14%
55%	42	28	14	24	18	100.00%	85.71%	85.71%
50%	47	28	19	25	22	100.00%	89.29%	89.29%

Table 5 – summary statistics for each MOM classifier using lm M5 for fault prediction with test data

c	break point	# actual fp	# actual nfp	# pred fp	# pred nfp	$G(c)$	$G'(c)$	$\emptyset(c)$
95%	5	5	0	5	0	17.86%	17.86%	100.00%
90%	9	9	0	9	0	32.14%	32.14%	100.00%
85%	14	14	0	13	1	50.00%	46.43%	92.86%
80%	19	19	0	17	2	67.86%	60.71%	89.47%
75%	24	24	0	19	5	85.71%	67.86%	79.17%
70%	28	28	0	19	9	100.00%	67.86%	67.86%
65%	33	28	5	20	13	100.00%	71.43%	71.43%
60%	38	28	10	23	15	100.00%	82.14%	82.14%
55%	42	28	14	25	17	100.00%	89.29%	89.29%
50%	47	28	19	25	22	100.00%	89.29%	89.29%

Table 8 – summary statistics for each MOM classifier using lm Greedy for fault prediction with test data

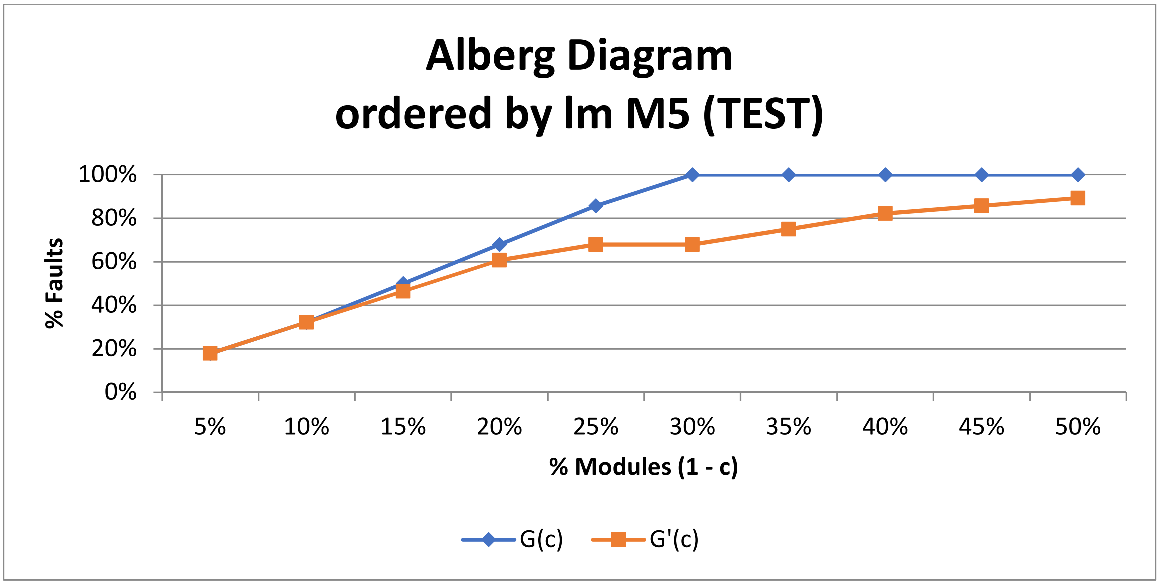

Figure 6a – Alberg Diagram for linear regression model with M5 selection on test/validation set

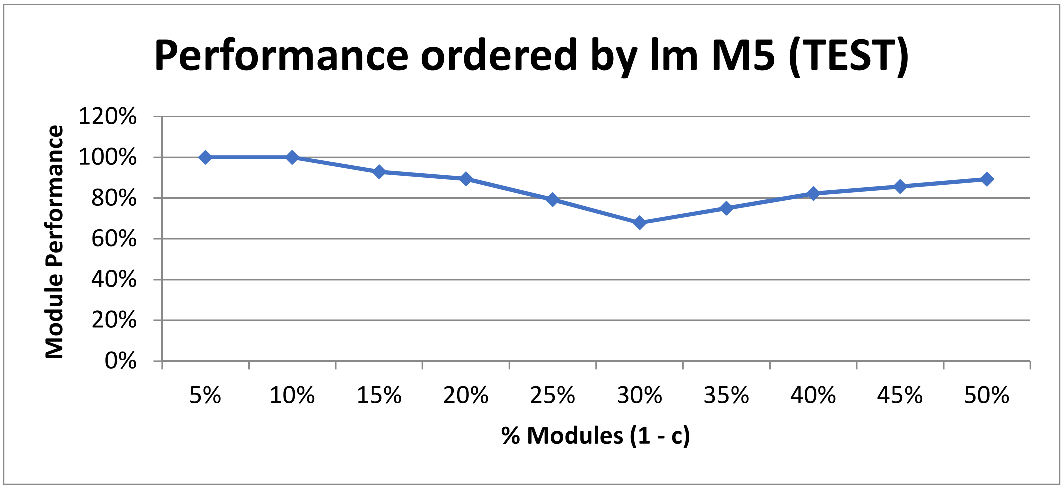

Figure 6b – Performance graph for linear regression model with M5 selection on test/validation set

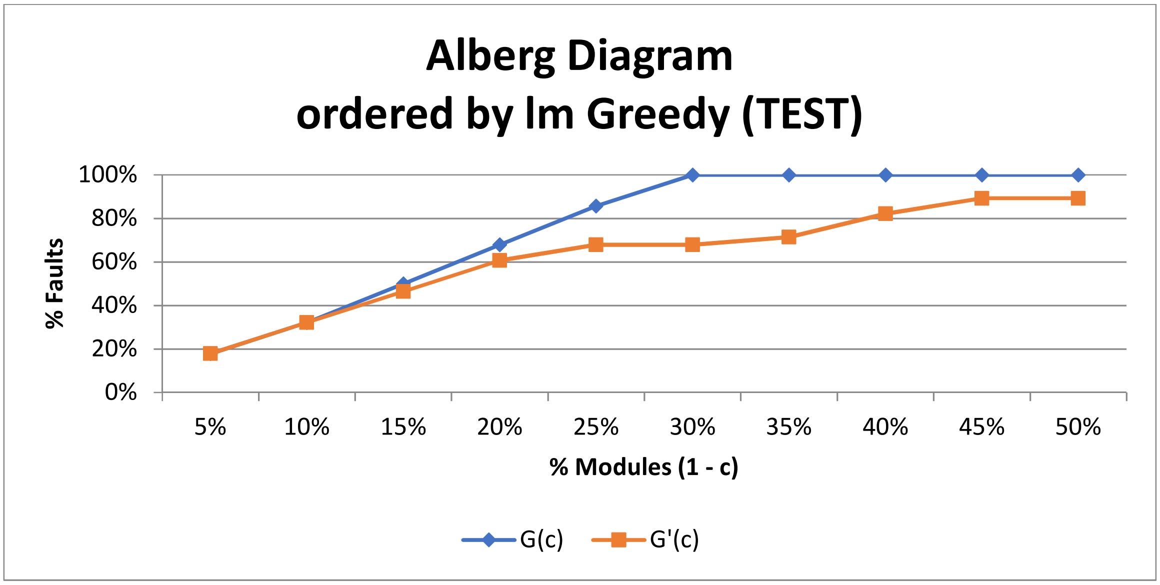

Figure 7a – Alberg Diagram for linear regression model with Greedy selection on test/validation set

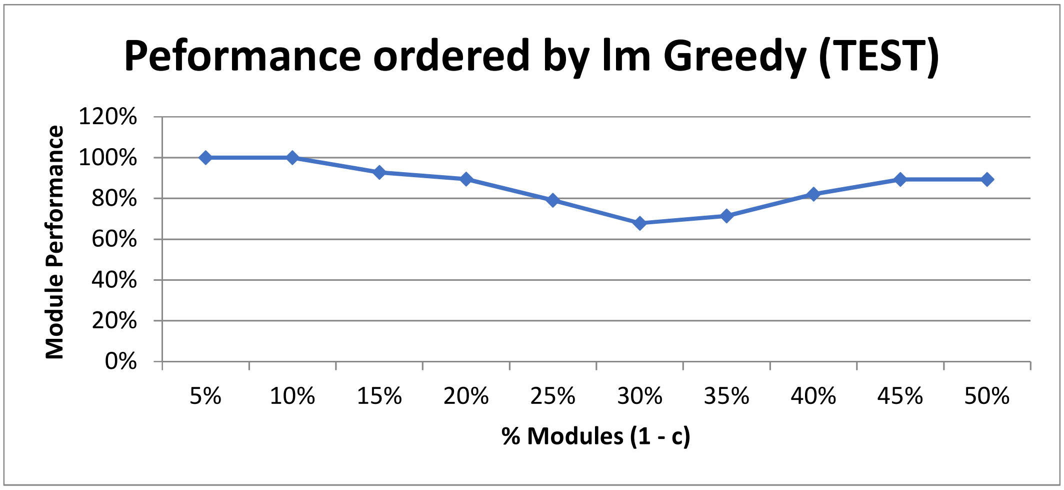

Figure 7b – Performance graph for linear regression model with Greedy selection on test/validation set

Because both fp and nfp are known for each MOM classifier, a confusion matrix can be created to evaluate each classifier for Type I (False Positive Rate) and Type II (False Negative Rate) errors. Assignment #2 created a cost classifier decision tree model based on the same input training and test datasets. Several *c* values, or cost ratios, were chosen to create and assess various decision tree models to come up with the near-optimal model balancing False Positive Rate (FPR) and False Negative Rate (FNR), keeping FNR as low as possible. The same confusion matrix statistics were assessed for the MOM classifiers to compare the FPR and FNR with the decision tree model from the previous assignment. Assignment #2 used a cost sensitive classifier combined with the J48 decision tree model for which an optimal cost ratio was assessed by setting the cost of FPR to one and adjusting the cost of FNR. Table 7 shows a generic confusion matrix with equations used to calculate the FNR and FPR.

Confusion Matrix		Predicted Class			
		Positive	Negative		
Actual Class	Positive	a	b	False Negative Rate (FNR)	*b/(a+b)*
	Negative	c	d	False Positive Rate (FPR)	*c/(c+d)*

Table 7 – confusion matrix template with FPR / FNR equations

Given the choice of the 0.5 Type II error cost, c, running the same error cost values on the validation and training datasets showed that this 0.5 value was the most balanced between FPR and FNR. Table 8 lists all the results for the decision tree model evaluations showing the near-optimal FPR and FNR being 0.128 and 0.109, respectively.

	Model Fit Data			Model Test Data				
Type II Error cost adj (c)	FPR	FNR	RMSE	FPR*	FNR**	RMSE	Accuracy	Error Rate
1	0.09023	0.20000	0.33436	0.07576	0.32143	0.36477	0.85110	0.14890
5	0.03008	0.56364	0.39968	0.00000	0.78571	0.38902	0.76600	0.23400
0.9	0.10526	0.16364	0.32848	0.07576	0.32143	0.36399	0.85110	0.14890
0.8	0.11278	0.16364	0.33362	0.07576	0.28571	0.34739	0.86170	0.13830
0.7	0.12782	0.14545	0.34038	0.09091	0.28571	0.35826	0.85110	0.14890
0.6	0.13534	0.12727	0.33004	0.09091	0.28571	0.35695	0.85110	0.14890
0.5	**0.12782**	**0.10909**	**0.32160**	**0.15152**	**0.28571**	**0.41413**	**0.80850**	**0.19150**
0.4	0.12782	0.14545	0.34911	0.12121	0.28571	0.40139	0.82980	0.17020
0.3	0.13534	0.14545	0.35389	0.12121	0.28571	0.39730	0.82980	0.17020

Table 8 – assignment 2 FPR & FNR cost ratio table

Figure 8 shows the FPR and FNR rates over each MOM classifier for the MOM built based on the linear regression with M5 variable selection; the rates are nearly identical for greed variable selection hence just M5 is shown with the training dataset. The optimal decision tree model FPR and FNR are overlaid to show how this compares to the MOM classifiers misclassification rates. In Figure 8, the red triangle is FNR and the yellow square is FPR, corresponding to the green highlighted values in Table 8. The MOM classifiers always predict a number of nfp modules less than or equal to the actual nfp modules, therefore there are never any nfp modules misclassified as fp, i.e. no FPR. MOM does have false negatives by predicting some fp modules as nfp. A lot of the actual fp modules are predicted by MOM but not all hence the FNR misclassifications. Based on the comparison of misclassification rates, MOM has less misclassification error for the following threshold values: 95, 90, 85, 60, 55, and 50. There is no FPR in MOM so the adage of FPR and FNR being nearly equal does not apply but MOM does have lower instances of FNR with no FPR, thus has less overall misclassifications for both the worse Type II and Type I errors.

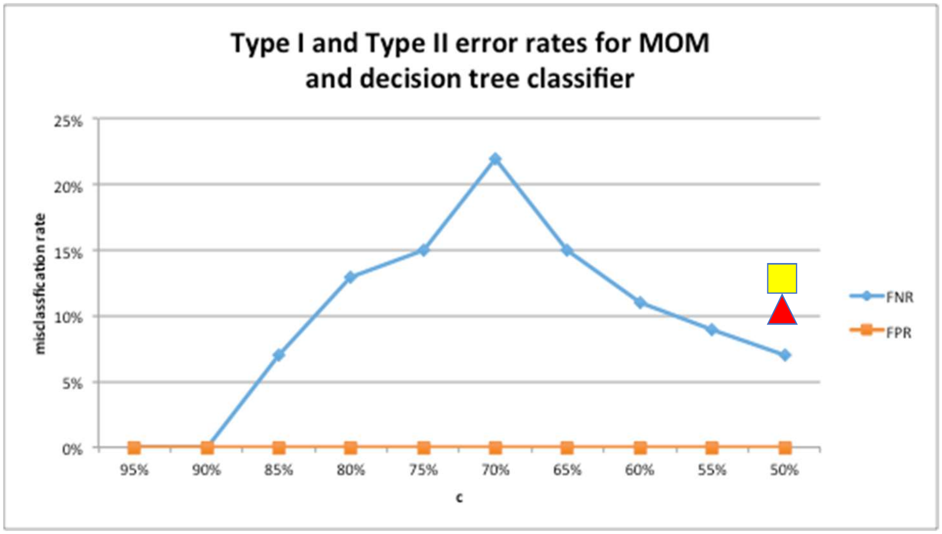

Figure 8 – FPR and FNR MOM rates with decision tree (c=0.5) misclassification rates

Conclusion

In conclusion MOM can be partitioned by percentile so that a solid predictive model can be used by assessing correct possible fault-prone modules without knowing these exact fault-prone thresholds. A MOM with a threshold of 10% would say that any modules with faults above the 10% percentile of the dataset would be considered fault-prone, with the rest not fault-prone. This focuses the reliability enhancement efforts on a portion of the dataset that covers a good portion of all possible fault prone modules. MOM can also be used as a classifier, which is the primary topic of this assignment. It was shown through the number predicted and actual fault-prone modules that MOM is a capable classifier over a reasonable range of thresholds indicating fault-prone or not fault-prone.

A comparison to MOM was also made comparing MOM to the decision tree model from assignment #2 with regards to FPR and FNR with MOM having threshold “options” performing better than the optimal decision tree cost ratio model.

References

[1] Witten, I., Frank, E. (2005), *Data Mining Practical Machine Learning Tools and Techniques*, 2nd edition, Elsevier Inc.

[2] Khoshgoftaar, T.M., Allen, E.B. (2003), Ordering Fault-Prone Software Modules, Software Quality Journal

[3] Khoshgoftaar, T.M., Allen, E.B. (1999), A Comparative Study of Ordering and Classification of Fault-Prone Software Modules, Empirical Software Engineering