Reduced Order Modeling of a Heat Exchanger with a Stacking Ensemble to reduce Computational Inefficiencies

Vinayak Vijaya Chandran
Applied Research and Technology (ART)
Collins Aerospace
Bengaluru, India
VinayakVijaya.Chandran@collins.com

Roopa Adepu
Applied Research and Technology (ART)
Collins Aerospace
Bengaluru, India
Roopa.Adepu@collins.com

2022 IEEE International Symposium on Systems Engineering (ISSE) | 978-1-6654-8182-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/ISSE54508.2022.10005464

***Abstract*—Reduced Order Modeling is a technique for reducing the computational complexity of a model while preserving the expected fidelity within a controlled error. One of the techniques used to create a Reduced Order Model (ROM) is Artificial Neural Networks (ANN). A successful approach to reducing the variance of ANN model prediction is to train multiple models instead of a single model and to combine the predictions from these models, which is commonly called Ensemble learning. When the predictions from the multiple models are combined using another regression model, it is called Stacking ensemble. This paper studies the effectiveness of using Genetic programming algorithm in taking the outputs of each model as input and attempting to learn how to best combine the input predictions to make a better output prediction.**

The above-mentioned approach is used to create a ROM for a crossflow heat exchanger steady-state component. There are 6 inputs parameters namely Cold & Hot inlet temperature, Cold & Hot outlet pressure and Cold & Hot inlet flow. There are four outputs namely Hot & Cold outlet temperature and Hot & Cold inlet pressure. A multi-input single output (MISO) ROM is created for each of the outputs. There are 3 different configurations of ANNs used to cover a good range of the Hyperparameter values. The output from each of the ANNs is then combined using Genetic Programming Algorithm. The Overall model has an R2 value of above 95% for each of the outputs. The ROM thus created can run simulations at a much faster rate. The ROM of the HX component is a black box and can be shared with third party without any concerns over propriety information loss.

***Keywords*—Artificial Neural Networks, Genetic Programming, Ensemble learning, Stacking Ensemble, Reduced Order Modeling, Random Sampling technique.**

I. Introduction

Ensemble learning techniques aim to create a meta-regressor by combining several regressors, typically by voting, created on the same data and improve their performance [1] [2]. Ensembles are usually used to overcome three types of problems associated with base learning algorithms: the statistical problem; the computational problem; and the representational problem [3]. When the training data is too small, the algorithm picks a classifier or regressor which performs very well on the training data, but the accuracy comes down significantly on new data that regressor has not seen. This constitutes the statistical problem as in that the problem here is to identify a training set that completely represents the sample space. The computational problem occurs when the algorithm gets trapped in a local minimum instead of global minimum. And finally, the representational problem occurs when the regressor within the hypothesis space is not a good approximation of the true function [4].

There are many approaches and algorithms that are proposed regarding the design of an ensemble model. There are 3 major elements of the ensemble learning method that can be varied to create a new model. Training Data is the first element and varying how the training data is chosen or split for each ensemble model create a new variation. Ensembles created by manipulating the training data, the input features, or the output labels of the training data, or by injecting randomness into the learning algorithm [3]. For example, Bagging learning ensembles, or bootstrap aggregating, introduced by Breiman [5], generates multiple training datasets with the same sample size as the original dataset using random sampling with replacement. A regression model is then trained on each of the bootstrap sample and the prediction from each of the model is averaged to get the final prediction. While Bagging can significantly improve the performance of unstable learning algorithms such as neural networks, it can be ineffective or even slightly deteriorate the performance of the stable ones such as k-nearest neighbor methods [5].

The second element is the choice of ensemble models. Different algorithms or techniques can be used for each of the regression models. And different configuration of the same model or technique can also be used for each of the member models. The advantage here is that some models would do better on a subset of the data than the other and we combine all such models over the entire sample range resulting in an overall better prediction. The third element is the choice of combination. A simple approach is to average the prediction of each of the models. An alternative approach is to create a generalized additive model which chooses the weighted sum of the component models that best fit the training data. For example, boosting methods can be used to improve the

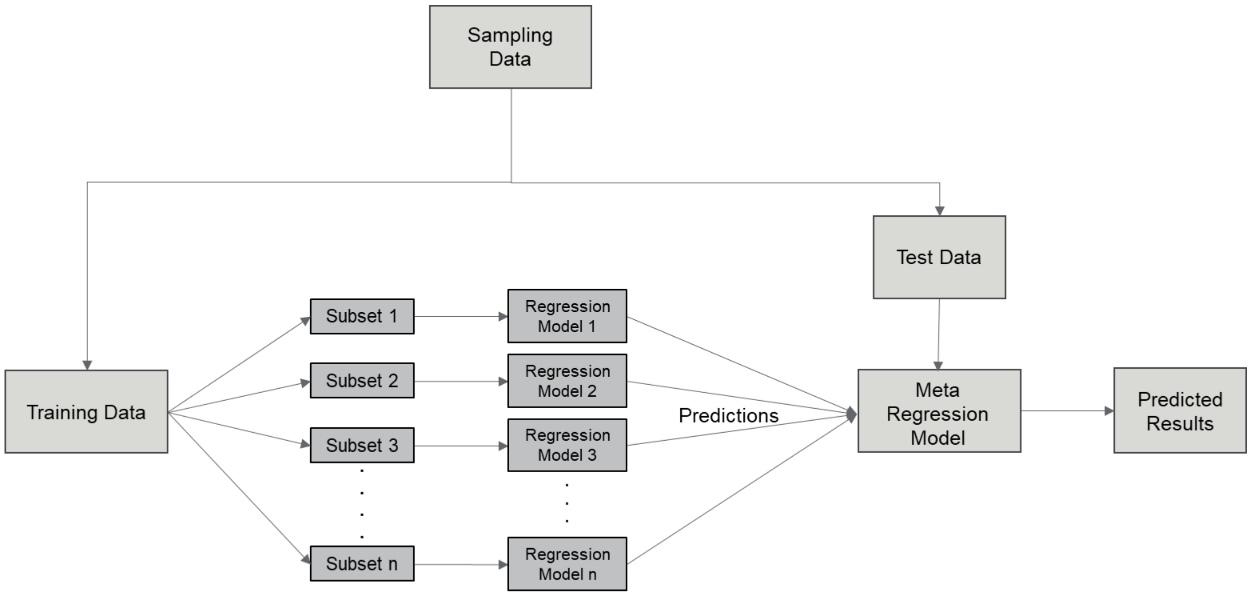

Figure 1: The Standard Stacked Ensemble Learning

accuracy of any "weak" learning algorithm by assigning higher weights for the misclassified instances [4]. The same algorithm is then reapplied several times and weighted voting is used to combine the predictions of the resulting series of classifiers [6]. Examples of Boosting methods include AdaBoost, AdaBoost.M1 and AdaBoost.M2 which were proposed by Freund & Schapire [7].

One of the more sophisticated approaches is to train an entirely new meta model to combine the prediction from all the ensemble models. This technique is called stacked ensemble learning. There are several algorithms that can be used for stacking. Sikora & Al-laymoun [4] proposed using Genetic programming as the algorithm for stacking and noted the improvement in performance over the individual learning algorithms as well as over the standard stacking algorithm.

Abbreviations and Acronyms

AdaBoost	Adaptive Boosting
ANN	Artificial Neural Network
Bagging	Bootstrap Aggregation
GP	Genetic Programming
HX	Heat Exchanger
MISO	Multi Input Single Output
R2	R Squared
ROM	Reduced Order Model

II. STACKING ENSEMBLE LEARNING

Since the last decade, Ensemble Learning has become more prominent because of its good performance and good accuracies resulted from various regression or classification problems [8]. Ensemble Learning includes combining different learning models with another learning model to improve the results by each model [9]. A successful approach to reduce the variance of ANN model prediction is to train multiple models instead of a single model and to combine the predictions from these models [10]. The reason why this method works better than any of the constituent models is that each model has its own range where it does better than the others and combining the results of such models gives an overall better prediction. There are usually two levels in this ensemble learning. In the first level, a set of models are created from training the data, and in the second level, the models that are created in the first level are combined to create one prediction model which gives good accuracies. The Standard Stacked Ensemble Learning Method is shown in figure 1. Here, the training data is divided into n subsets. For each subset, a regression model is created which gives its predictions as outcome. A meta regression model is then trained to best combine the predictions from the n individual models. Once the meta-regressor is trained, the performance of this regressor is tested using a separate set of data labelled test data. The regressor is now able to predict outputs much better than the individual models.

Usually, in a standard stacked ensemble learning, the meta regression model is created by taking an average of all the regression models that are created using the n subsets. The more effective way, which gives better predictions is the weighted average ensemble learning. In this technique, the weight distribution of the meta regression model is created by assigning a weight to a regression model that is proportional to its performance. These two linear approaches are primitive and yield only satisfactory results. A more advanced method is to train another non-linear regressor to best combine the results from these models. It is however worthwhile to note that the choice of meta regressor depends on the problem at hand. Since this paper studies the application of this technique on a Heat exchanger component, nonlinear meta regressors are the best choice.

III. STACKING ENSEMBLE LEARNING USING GENETIC PROGRAMMING

Genetic Programming is a machine learning method which is biologically inspired that evolves computer programs to perform a task. In principle Genetic Programming (GP), given enough time, should be capable of simultaneously identifying and combining useful program subexpressions to yield an overall program that maximizes fitness [11]. Programs are 'bred' through continuous improvement of an initially random population of programs. Improvements are made possible by stochastic variation of programs and selection according to prespecified criteria for judging the quality of a solution [12].

Genetic Programming first randomly creates mathematical equation (which are usually represented by parse trees) that try to approximate the input output relationship. It then uses techniques called mutation and cross-over to generate children or derivate solutions that try to better approximate the relationship. This process is continued or iterated until the mathematical equations give the best predictions for the inputs. Figure 2 outlines the whole method pictorially.

In this case, the Genetic programming implements the weights assigned to each of the ensemble members (the regression model of each subset) as an individual member of the population. Each population member is an equation consisting of weights and relations between the ensemble members.

Since the number of runs cannot be infinitely big, early stopping criterion has been used wherein the run stops if there is no improvement in the training after 50 generations. Another technique employed in the Genetic programming run is the Random Sampling technique [13].

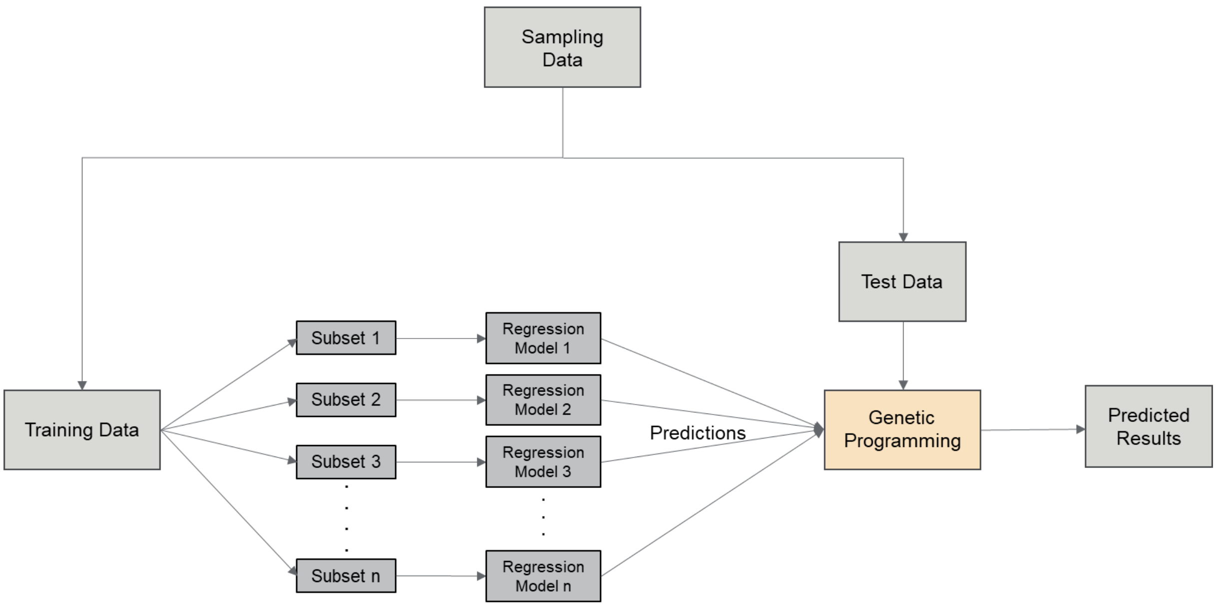

Figure 2: The Stacked Ensemble Learning using Genetic Programming as the meta regression model

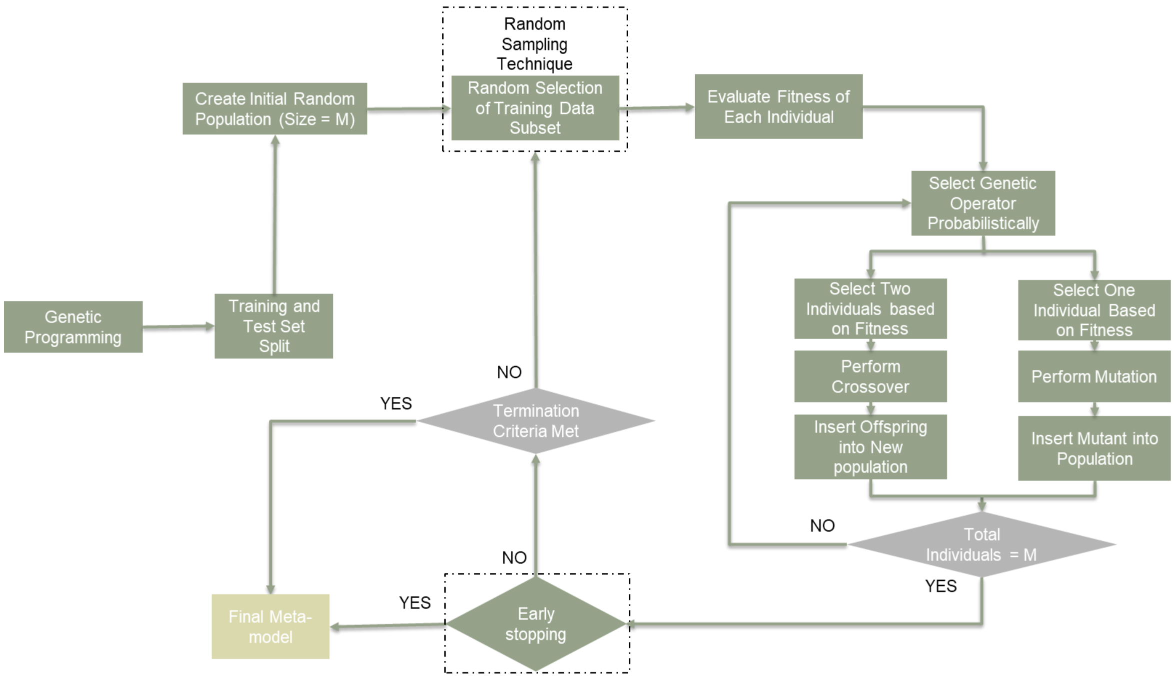

Figure 3: Genetic programming with Early stopping and Random Sampling Technique to avoid overfit

Here the training sample is changed in each generation so that the population that is propagated through the generations are individuals with low generalization error. In other words, Random sampling technique will ensure that overfit individuals are not used for further generation of newer samples in the generations that follow [13][14]. The whole technique is clearly shown in figure 3.

IV. Observations

In this study, this technique is used for Reduced Order Modeling of a cross flow heat exchanger component, which has 6 input parameters, and 4 output parameters as shown in table 1.

TABLE 1 – Inputs and Outputs of Heat Exchanger Component

Inputs	Outputs
Cold Inlet Temperature	Hot Inlet Pressure
Hot Inlet Temperature	Cold Inlet Pressure
Cold Outlet Pressure	Hot Outlet Temperature
Hot Outlet Pressure	Cold Outlet Temperature
Cold Inlet Flow	
Hot Inlet Flow	

The Stacking Algorithm technique for both the versions, one with average stacked ensemble learning and other with genetic programming, is implemented in Python using Tensor Flow library. For all the experiments, the data sets are split as 80% for training and 20% for testing. The training data is again split to 3 equal subsets of samples, for which individual Artificial Neural Network (ANN) models are created. For each experiment, the predictions' performance is observed based on the R Squared values. One thing to note here is, the training data can be split into n number of subsets, in this case after various experiments, it was decided to split the data into 3 subsets.

Table 2 shows the structure of ANNs of each subset for each output. Both versions of Stacked Ensemble Learning are experimented on all the outputs, and their predictions are observed. Table 3 shows the performance of predictions.

It can be observed from the table that there is a significant improvement in the performance of the stacked ensemble using Genetic Programming as meta-regression model over the standard average stacked ensemble model. When Standard Stacked Ensemble technique is implemented, the R-Squared values of all the outputs is in the range of 85-90%. But when Genetic Programming is used as meta-regression model, the R-Squared values is increased to over 95% for all the outputs.

One thing to note here is that the training time for the modified stacked ensemble learning is higher compared to the standard stacked ensemble learning, as modified stacked ensemble learning involves performing the Genetic Programming.

TABLE 2 – Structure of ANNs of the Outputs

Output Parameter		Layers	Nodes	Learning Rate
Hot Inlet Pressure	Subset 1	2	5	0.5
	Subset 2	2	10	0.38
	Subset 3	2	6	0.351
Cold Inlet Pressure	Subset 1	1	7	0.675
	Subset 2	2	6	0.34
	Subset 3	2	5	0.35
Hot Outlet Temperature	Subset 1	2	12	0.28
	Subset 2	2	12	0.3
	Subset 3	3	7	0.25
Cold Outlet Temperature	Subset 1	1	12	0.625
	Subset 2	2	12	0.3
	Subset 3	5	8	0.15

V. Conclusion

In this paper, reduced order modeling of cross flow heat exchanger component is discussed. For the Reduced Order Model, a modified version of stacking ensemble learning is implemented, which uses Genetic Programming to create an ensemble. Both standard stacked ensemble and modified stacked ensemble were studied on the component. Standard Stacked Ensemble has given R-Squared values of range 85 to 90 for all the four outputs, whereas the Modified Stacked Ensemble has given R-Squared values greater than 95, which shows that Genetic Programming as a meta regression model has significantly improved the outputs predictions.

TABLE 3 – R Squared values of the Outputs

Output Parameter	R Squared	
	Average Stacked Ensemble	Stacked Ensemble using Genetic Programming
Hot Inlet Pressure	85.91557721	95.21288194
Cold Inlet Pressure	90.53757974	97.56807844
Hot Outlet Temperature	89.60701413	96.57353094
Cold Outlet Temperature	87.51268948	95.92970784

Creation of Reduced Order Model of this Heat Exchanger component helped us in 2 ways.

1. Running the Heat Exchanger Simulink model consumes a lot of time. Reduced Order Model will bring down the computational efforts and gives the results in few seconds. For this study, 100 different inputs of Heat Exchanger in Simulink had to be run, which took a minimum of 10 minutes, whereas a Reduced Order Model of this Heat Exchanger was run by giving the same number of inputs, it took less than 30 seconds.
2. The Reduced Order Model of this Heat Exchanger can be shared to third party as a black box. Because the ROM in the end has only the relation between inputs and outputs and its weights. User can still run the cases with the ROM and get the outputs. This ensures protection of the company's proprietary information.

REFERENCES

[1] Dzeroski, S., & Zenko, B. (2004). Is combining classifiers with stacking better than selecting the best one? Machine Learning, 255–273.

[2] Opitz ,D., & Maclin, R. (1999). Popular ensemble methods: an empirical study, Journal of Artificial Intelligence Research, 11, 169-198.

[3] Dietterich, T. (2002). Ensemble learning, in The Handbook of Brain Theory and Neural Networks, 2nd ed., M. Arbib, Ed., Cambridge MA: MIT Press.

[4] Sikora, Riyaz and Al-laymoun, O'la Hmoud (2014) "A Modified Stacking Ensemble Machine Learning Algorithm Using Genetic Algorithms," Journal of International Technology and Information Management: Vol. 23 : Iss. 1 , Article 1.

[5] Breiman, L. (1996). Heuristics of instability and stabilization in model selection. *The annals of statistics*, *24*(6), 2350-2383.

[6] Pal, M. (2007). Ensemble learning with decision tree for remote sensing classification. *World Academy of Science, Engineering and Technology*, *36*, 258-260.

[7] Freund, Y., & Schapire, R. E. (1996, July). Experiments with a new boosting algorithm. In *icml* (Vol. 96, pp. 148-156).

[8] Ren, Y., Zhang, L., & Suganthan, P. N. (2016). Ensemble classification and regression-recent developments, applications and future directions. *IEEE Computational intelligence magazine*, *11*(1), 41-53.

[9] Mendes-Moreira, J., Soares, C., Jorge, A. M., & Sousa, J. F. D. (2012). Ensemble approaches for regression: A survey. *Acm computing surveys (csur)*, *45*(1), 1-40.

[10] Hansen, L. K., & Salamon, P. (1990). Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, *12*(10), 993-1001.

[11] Arnaldo, I., Krawiec, K., & O'Reilly, U. M. (2014, July). Multiple regression genetic programming. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation* (pp. 879-886).

[12] Brameier, M., & Banzhaf, W. (2001). Evolving teams of predictors with linear genetic programming. *Genetic Programming and Evolvable Machines*, *2*(4), 381-407.

[13] Gonçalves, I., Silva, S., Melo, J. B., & Carreiras, J. (2012, April). Random sampling technique for overfitting control in genetic programming. In *European Conference on Genetic Programming* (pp. 218-229). Springer, Berlin, Heidelberg.

[14] Gonçalves, I., & Silva, S. (2011, October). Experiments on controlling overfitting in genetic programming. In *15th Portuguese conference on artificial intelligence (EPIA 2011)* (pp. 10-13).