Evaluating vehicle structural crash robustness using a Machine Learning approach

Saket Kumar, Pravinkumar Shedolkar, Kalyani Avhale

Abstract This paper presents a machine learning approach for evaluating crash robustness of a vehicle structure. Uncertainties associated with design parameters due to manufacturing and other test variations pose performance risk with regard to vehicle crashworthiness. The available methodology of robustness evaluation is time-consuming and associated with high computational cost in the CAE environment. Understanding the relationship between design parameter variables and vehicle structural response, such as steering column intrusion, seat mounts intrusion and firewall intrusion, requires extensive datasets. A methodology that combines datasets from a previous robustness evaluation carried out for a similar vehicle platform offers a viable solution. A machine learning model is proposed in this work, one that is trained by using a fraction of combined datasets selected randomly from the data space. The predictions from the trained machine learning model show a good correlation with the actual CAE results, as the mean absolute error was found to be less than 5 mm for all of the response parameters.

Keywords Design robustness evaluation, machine learning model, CAE simulations, randomForest, safety critical parameters.

I. INTRODUCTION

CAE simulations are used extensively in the development of automotive crashworthiness. For CAE simulations to be more realistic, uncertainties associated with design parameters due to manufacturing and other test variations must be incorporated into the simulations. Design robustness studies are performed in the CAE environment so that the vehicle delivers the desired performance even with likely uncertainties. Safety critical parameters, such as panel thickness, joinery strengths and test setup tolerances, are considered as noise variables for the robustness evaluation. A simulation matrix is defined in a design optimization software environment. The simulation matrix contains random combinations of the noise variables, which are randomly sampled using the Latin Hypercube algorithm [1]. Each simulation defined in the matrix is processed in the CAE software environment. The results for each simulation are compared with the performance targets and design actions are taken if necessary, based on the performance gap. The drawback of this method of robustness evaluation is that it requires multiple simulation servers and it takes a long time to complete the evaluation. To make the robustness evaluation more time- and cost-efficient, it is necessary to understand the relationship between the noise variables and the performance responses. Establishing this relationship is a complex task and requires sufficient datasets. These complexities can be addressed by using the machine learning approach.

The purpose of this work was to develop, train and validate a machine learning model to predict the effect of changing values of noise variables on the vehicle performance. This was accomplished by creating datasets from existing robustness evaluations carried out for vehicle variants on a similar platform. Supervised machine learning regression algorithms were studied comprehensively. The machine learning model was developed, trained and validated by selecting the most suitable and sophisticated algorithm. The developed machine learning model could be used to quickly evaluate the

S. Kumar (tel: 7219401837; e-mail: saketme0072@gmail.com) and P. Shedolkar are CAE engineers at Tata Motors Passenger Vehicle Limited. K. Avhale is a data scientist at Tata Technologies Limited.

vehicle crash robustness without using the CAE computational servers.

II. METHODS

The process of creating and validating a machine learning model comprises multiple steps, including datasets preparation, selection of an appropriate machine learning algorithm, and training and validation of the machine learning model.

Step 1: Datasets preparation

To prepare the datasets for model training and validation, existing data from previous robustness evaluations are utilised. The robustness datasets, including their parametrised variables, and the output responses from each vehicle variant of a platform are combined. The non-common variables, which are least sensitive towards an output response, are treated as constants. Figure 1 and Fig. 2 show an example of preparing the combined datasets.

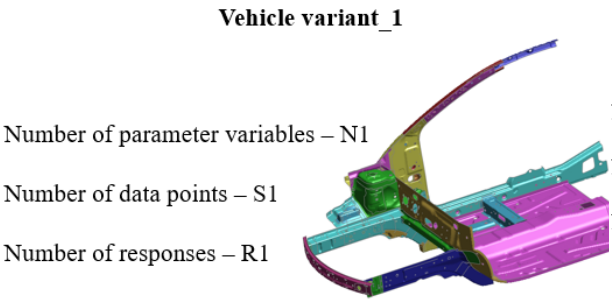


Fig. 1. Input parameters of vehicle variant_1.

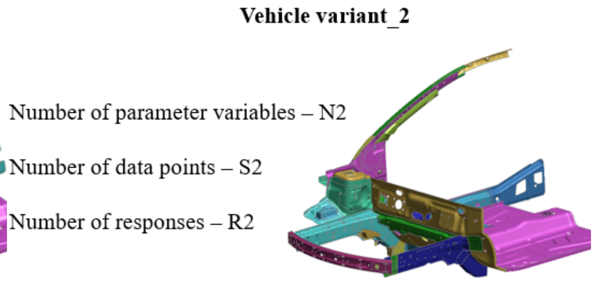


Fig. 2. Input parameters of vehicle variant_2.

After dropping the least sensitive non-common variables from the overall variable list, the common input data-set after combining both of the vehicle variants will have a total “N” number of parameter variables and “S” number of data points (sum of data points available for each vehicle variants). Hence, a dataset comprising of “S” data points is available for machine learning model development and validation. Table I shows the common Input table combining the vehicle variants and the parameters they contain. Table II shows the common response table for both of the vehicle variants.

TABLE I
COMMON INPUT VARIABLE TABLE

Serial No.	Part	Variables
1	Load path member	Panel thickness
2	Joineries	Strength
		Failure time
3	Test setup	Barrier position in lateral direction
		Barrier position in vertical direction

TABLE II
COMMON RESPONSE TABLE

Serial No.	Performance parameter
1	Steering displacement in X-direction
2	Steering displacement in Y-direction
3	Steering displacement in Z-direction
4	A-Pillar 100 mm below waist displacement
5	A-Pillar 100 mm above sill displacement
6	Driver seat rear inboard mount displacement
7	Driver seat front outboard mount displacement
8	CCB displacement at steering mount in X-direction
9	CCB displacement at steering mount in Y-direction
10	Brake residual in X-direction
11	Brake residual in Z-direction
12	LH foot node displacement for driver
13	RH foot node displacement for driver

Step 2: Selection of an appropriate machine learning algorithm
A comprehensive study was carried out on the below-mentioned supervised machine learning regression algorithms.

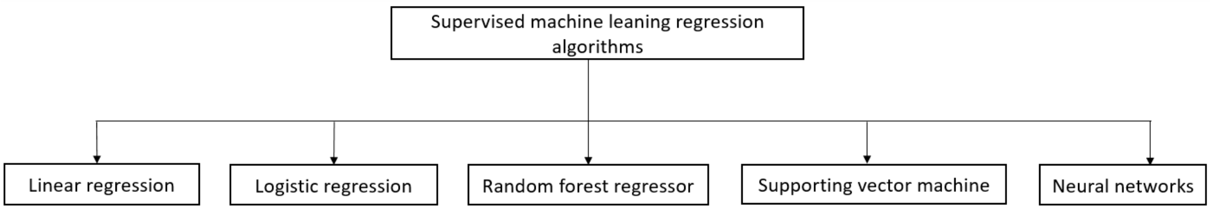


Fig. 3. Supervised machine learning regression algorithms.

randomForest regressor algorithm was selected for machine learning model development because it is easy to operate and requires less calculation time [2].

Step 3: Training and validation of machine learning model
To train the machine learning model in its first iteration, 70% of the available dataset is utilised. The selection for the data points is done by random sampling of the data from the available data space. Predictions for the remaining part of the dataset are generated from the trained model. The predicted outcome is then compared with the actual simulation results.

In the second iteration, only 15% of the fine-tuned data is selected randomly from the available data space. This fraction of the dataset is used for the training purpose. Predictions are generated by the model for the rest of the data and then compared with the actual simulation results.

III. RESULTS

The predictions were generated for the remaining data points using the models trained in both the first and second iterations. Initially, the predicted outcomes from the model trained in first iteration are compared with the actual CAE results. Scatter for the predicted data points are also checked.

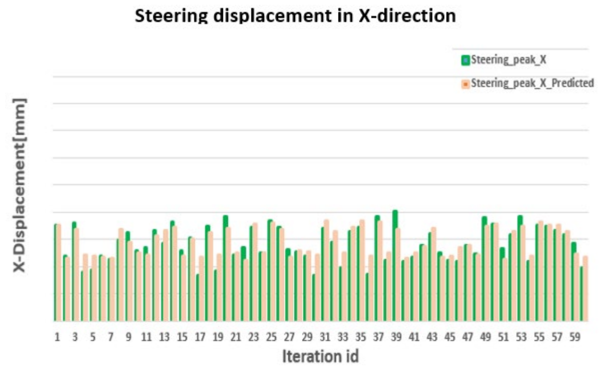


Fig. 4. First iteration: Simulation vs Prediction For steering displacement.

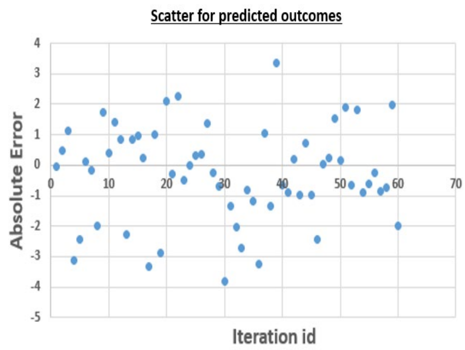


Fig. 5. First iteration: scatter for predicted data points for steering displacement.

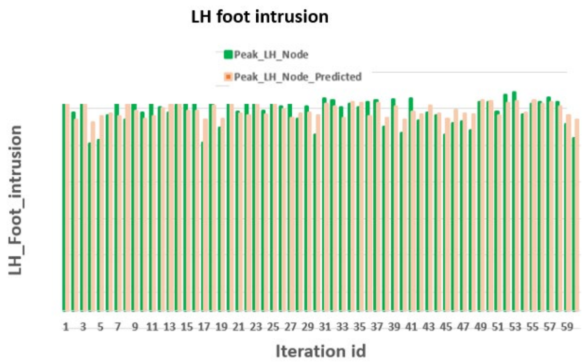


Fig. 6. First iteration: Simulation vs Prediction for LH foot intrusion.

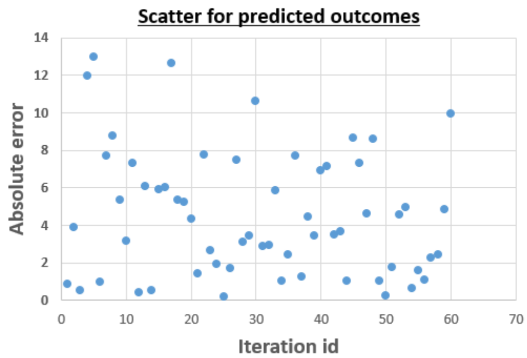
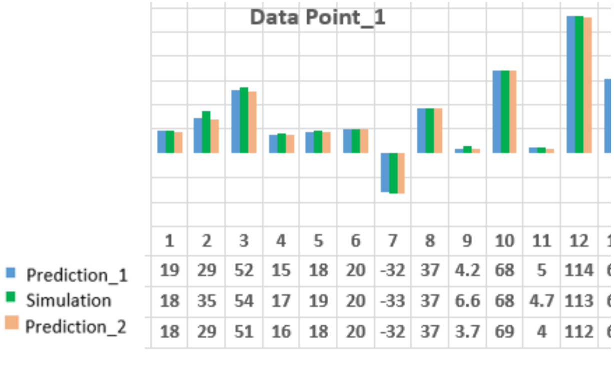


Fig. 7. First iteration: scatter for predicted data points for LH foot intrusion.

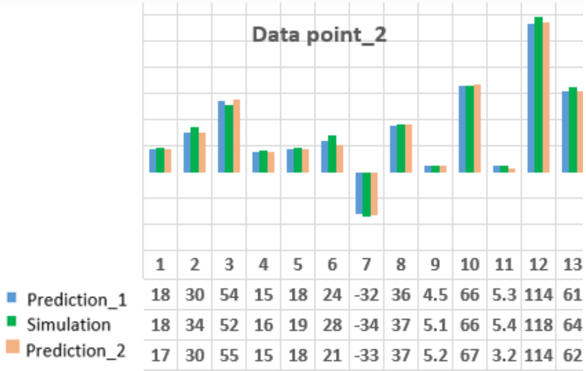
A comparison between the prediction for the remaining data points from iteration 1 and the simulation results is shown in Fig. 4 and Fig. 6 for two of the performance parameters listed in Table I. The scatter for prediction is also shown in Fig. 5 and Fig. 7. It was observed that prediction by the machine learning model showed good correlation with the simulation results. The mean absolute difference between the prediction and the simulation was found to be not more than 5 mm, which is acceptable.

The predictions from the model trained in the second iteration, with reduced number of data points, were also compared with the actual simulation and with the predicted outcome from the trained model in the first iteration. Predicted outcomes for two of the input data points compared with the actual simulation results are shown in Fig. 8 and Fig. 9. The numbers on x-axis in these figures represent the output performance parameters mentioned in Table I.



	1	2	3	4	5	6	7	8	9	10	11	12
Prediction_1	19	29	52	15	18	20	-32	37	4.2	68	5	114
Simulation	18	35	54	17	19	20	-33	37	6.6	68	4.7	113
Prediction_2	18	29	51	16	18	20	-32	37	3.7	69	4	112

Fig. 8. Simulation vs Predictions from ML model from both of the iterations.



	1	2	3	4	5	6	7	8	9	10	11	12	13
Prediction_1	18	30	54	15	18	24	-32	36	4.5	66	5.3	114	61
Simulation	18	34	52	16	19	28	-34	37	5.1	66	5.4	118	64
Prediction_2	17	30	55	15	18	21	-33	37	5.2	67	3.2	114	62

Fig. 9. Simulation vs Predictions from ML model from both of the iterations.

A good correlation was observed among ML model predictions and the actual simulation outcomes. The mean absolute deviation of the predicted outcomes from the actual simulation outcomes was again found to be less than 5 mm for all of the performance parameters, hence the machine learning model can be trained using a lesser fraction of the fine-tuned data, as was done in the second iteration of model training.

IV. DISCUSSION

The machine learning model developed in this work was validated against the actual simulation outcomes. A good correlation was observed between the simulation and the predicted outcomes. This machine learning model is useful in quick analysis of vehicle crash robustness. It could also be used in analysing performance response of a vehicle when the vehicle is upgraded by changing some of the identified parameter variables. Though the machine learning model is well correlated with the actual simulation outcome, in order to enhance the usefulness of this model it will be necessary to combine the parameter variables of multiple vehicle platforms in future.

V. CONCLUSION

Vehicle structural crash robustness evaluation in CAE environment is time- and resource-consuming work. To address these challenges, a machine learning methodology was proposed. The proposed machine learning model was able to predict the outcomes with good precision. This approach significantly reduced the time and resources required for robustness evaluation. However, the challenge remains to combine parameter variables and responses from multiple vehicle platforms. Future work will focus on developing a robust machine learning model that can be used for evaluation across all the vehicle platforms.

VI. REFERENCES

[1] ESTECO (2019) s.p.a, modeFRONTIER®2019R3 Tutorial.

[2] Liaw, A. and Wiener, M. (2002) Classification and Regression by randomForest. *R News*, ISSN 1609-363, vol. **2/3**, December 2002, pp.18–22.