World Academy of Science, Engineering and Technology
International Journal of Transport and Vehicle Engineering
Vol:15, No:2, 2021

Comparing Machine Learning Estimation of Fuel Consumption of Heavy-Duty Vehicles

Victor Bodell, Lukas Ekstrom, Somayeh Aghanavesi

Abstract—Fuel consumption (FC) is one of the key factors in determining expenses of operating a heavy-duty vehicle. A customer may therefore request an estimate of the FC of a desired vehicle. The modular design of heavy-duty vehicles allows their construction by specifying the building blocks, such as gear box, engine and chassis type. If the combination of building blocks is unprecedented, it is unfeasible to measure the FC, since this would first require the construction of the vehicle. This paper proposes a machine learning approach to predict FC. This study uses around 40,000 vehicles specific and operational environmental conditions information, such as road slopes and driver profiles. All vehicles have diesel engines and a mileage of more than 20,000 km. The data is used to investigate the accuracy of machine learning algorithms Linear regression (LR), K-nearest neighbor (KNN) and Artificial neural networks (ANN) in predicting fuel consumption for heavy-duty vehicles. Performance of the algorithms is evaluated by reporting the prediction error on both simulated data and operational measurements. The performance of the algorithms is compared using nested cross-validation and statistical hypothesis testing. The statistical evaluation procedure finds that ANNs have the lowest prediction error compared to LR and KNN in estimating fuel consumption on both simulated and operational data. The models have a mean relative prediction error of 0.3% on simulated data, and 4.2% on operational data.

Keywords—Artificial neural networks, fuel consumption, machine learning, regression, statistical tests.

I. Introduction

FUEL consumption (FC) is arguably one of the most important aspects of heavy-duty vehicles. Customers tend to weigh in the FC of a vehicle when deciding which vehicle to buy, since this stands in direct correlation to their fuel expenses.

Current practices of providing FC estimations of a vehicle uses simulation tools. One of such primary tools is the Vehicle Energy Consumption Calculation Tool (VECTO) [1]. It is developed and distributed by the European Union. The simulation takes around one minute to complete, and is typically not as exact as the measured FC once the vehicle is in operation.

Execution time and accuracy provide reasons for defining an alternative estimation scenario that may be used when estimating FC. One such approach is to apply machine learning (ML). ML models are currently recognized for their potential to more efficiently handle problems in several different domains. There are recent studies investigating the use of machine learning in estimating FC of heavy-duty vehicles [2]–[4], aircraft [5] and buses [6]. Results of previous studies see a mean relative prediction error ranging from 4% to 14% on operational vehicles. No study offers a conclusive best-performing algorithm.

This study investigates the applicability of ML in estimating FC of heavy-duty vehicles. The research question is: Within what error margin can a ML model estimate FC generated from VECTO simulations and real world measurements of heavy-duty vehicles? Two separate scenarios are considered: 1) The reproducibility of simulated (VECTO) FC using vehicle features, and 2) reproducibility of measured FC using vehicle and operational features.

Victor Bodell and Somayeh Aghanavesi are affiliated with KTH Royal Institute of Technology, Stockholm, Sweden (e-mail: vbodell@kth.se, somagh@kth.se).

Lukas Ekström is with the Scania Group, Södertälje, Sweden (e-mail: lukas.ekstrom@scania.com).

II. Data

The dataset contains vehicle features, such as vehicle mass and engine specification, and operational data such as measured average fuel consumption and cruise control usage. The full set of considered vehicle features, their types, and their sources (operational vs. vehicle) are listed in Table I.

TABLE I
Input Features for the Models in Alphabetical Order

Variable	Discrete (O) /Continuous (X)	Operational data (O) /Vehicle data (X)
Airdrag	X	X
Average gross train weight	X	X
Average speed	X	O
Axle Model	O	X
Brake frequency	X	O
Chassis adaptation	O	X
Corrected Curb Mass	X	X
Country of purchase	O	X
Cruise Control usage	X	O
Engine displacement	O	X
Engine model	O	X
Horse power	X	X
Idling with Power take-off	X	O
Idling without Power take-off	X	O
Crusing powertrain ratio	X	X
Retarder usage	X	O
Stop frequency per 100km	X	O
Simulation payload weight	X	X
Total rolling resistance coefficient	X	X
Virtual velocity variance	X	O
Virtual slope average	X	O
Wheel configuration	O	X

The dataset for simulated fuel consumption contains 40,789 vehicles. The operational dataset contains almost 80,000

vehicles, but less than 20,000 have values for all desired features. The final merged dataset contains 39,001 unique vehicles.

A. Prediction

For both simulated and real world fuel consumption cases, the regression value was predicted as the fuel consumption flow in liters per 100 km. For the estimated data, the target variable is a floating point value. For the operational data, the values are recorded by control units in the vehicles. The control units are sometimes reset due to software upgrades. The target value is aggregated over the life cycle since the last occurrence of such a reset, and rounded to the nearest integer.

III. Method

The experiment consists of three phases and is described by Fig. 1.

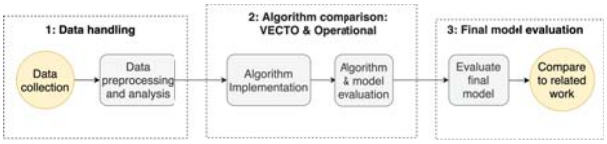

Fig. 1 The project process. Gray shapes are confined to a software-environment, whereas yellow shapes describe consulting external references

The first phase is about data collection and preprocessing. The second phase includes comparison of different machine learning algorithms where two baseline algorithms: Linear Regression (LR) [7, ch. 3] and K-nearest neighbor regression (KNN) [7, ch. 2]; and two advanced regression models: Artificial Neural Networks (ANN) [8, ch. 1] using Stochastic Gradient Descent (SGD) [9, ch 5.9], and ADAM-optimizer [10] were considered. Both ANN algorithms were considered as fully connected multi-layer perceptrons (MLP). Algorithm and model comparison was done using a variant of cross-validation known as Nested Cross-validation [11] on the dataset. If a best-performing algorithm is determined in the evaluation, a final model is trained and evaluated using the full dataset.

A. Data Preprocessing

The data was preprocessed such that all categorical features were one-hot encoded, and all continuous features were normalized according to the normal score of the feature ($\frac{X-\mu}{\sigma}$). Up to 1 percentile of upper and lower outliers were clipped for all operational features. Only vehicles produced in 2018, with diesel engines and a mileage of more than 20,000 km were used. All vehicles with missing values for any of the considered input features were removed. This resulted in 34,712 vehicles in the VECTO scenario, and 15,976 vehicles in the operational scenario.

B. Evaluating and Comparing Models

For evaluation of the ML models nested cross-validation (CV) was used in preference over normal 5- or 10-fold CV because it accounts for more reliable estimates of the true model error, as shown by Varma and Simon [11].

The nested CV approach consists of two nested cross-validation loops, where the inner loop is used to define model hyperparameters. The outer loop then evaluates the model with the best performing hyperparameters from the inner loop. The 5x2 nested CV configuration was used, meaning 5-fold CV in the outer loop and 2-fold CV in the inner loop.

The full dataset was split into 80% training data and 20% test data. The 5x2 nested CV was applied to the training set and resulted in five error measurements for each algorithm.

When error distributions had been determined, the Friedman-statistic [12] was calculated, in order to determine whether the performance difference between the algorithms is statistically significant. The threshold used for rejecting the null hypothesis on the Friedman statistic was $\alpha = 0.05$. If the null-hypothesis was rejected, a post-hoc analysis was performed to determine which performance differences are significant. The post-hoc analysis applied Li's two-step rejection procedure [13] with the Kolmogorv-Smirnov 2-sample test [14]. Due to the small sample size of the error distributions, in relation to the evaluation procedure of the Kolmogorov-smirnov method the post-hoc analysis gave more slack in rejecting H_0 by using a threshold of $\alpha = 0.10$.

After post-hoc analysis, the algorithms were ranked according to their performance. The ranking procedure ranks all error measurements compared to each other. In the case of four algorithms and 5-fold outer CV this yielded a total of 20 measurements. The ranking was then determined by ranking the mean ranks of each model. This is equivalent to the Average Rank procedure presented by Brazdil and Soares [15]. If post-hoc analysis found a statistically significant best-ranking model, this model was then trained on the full training set and evaluated on the original test set. If an ANN was found to be the best model, it was trained for 1000 epochs as the final model.

C. Algorithms and Hyperparameters

The considered hyperparameters for the algorithms are presented in Table II.

In the algorithm comparison phase a constant number of 200 epochs was used when training the ANNs. An early stopping technique was also used, to ensure that the error decreased by at least 10^{-4} error units during 10 consecutive epochs, otherwise training was stopped. The considered activation functions were $Relu$ $(max(0, x))$ and $sigmoid$ $(\frac{1}{1+e^{-x}})$.

IV. Result

The following sections describe the results for the simulation and operational data scenarios (IV-A and IV-C) as well as the final model results of both scenarios (IV-B and IV-D).

TABLE II
CONSIDERED HYPERPARAMETERS FOR DIFFERENT ALGORITHMS

Algorithm	Hyperparameter
KNN	k (neighbors)
	u (inverted distance weight [16])
MLP w/ SGD	hidden nodes
	hidden layers
	activation function
	learning rate
	momentum
	l2-regularization (weight-decay)
MLP w/ ADAM	hidden nodes
	hidden layers
	activation function
	learning rate
	l2-regularization (weight-decay)
	γ (first adaptive moment)
	β (second adaptive moment)

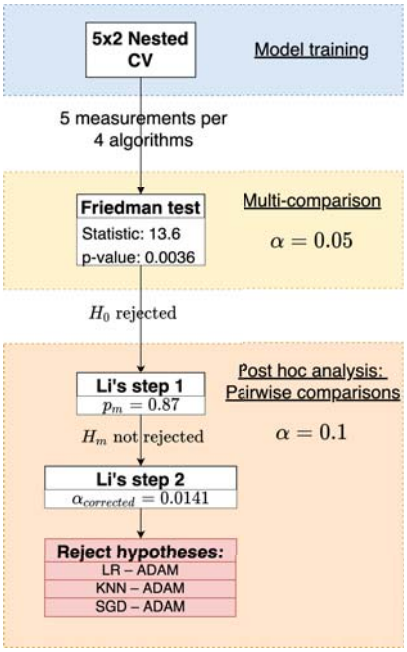

Fig. 2 Statistical evaluation procedure for the four algorithms in the VECTO scenario

A. Vecto Estimation

The error distributions resulting from the nested CV procedure had a median error of 0.0593, 0.0585, 0.0501 and 0.0312 for LR, KNN, SGD and ADAM, respectively. The ADAM-algorithm had a non-overlapping lowest error population compared to the other algorithms.

Results from the statistical analysis procedure is shown in Fig. 2. Using Li's 2-step procedure, the null hypotheses are not rejected in the first step, since $P_m = 0.87 > 0.10$ in contrast to the second step where p-value ≤ 0.0141.

From the results presented in Fig. 2 it is clear that all null hypotheses associated with the ADAM-models can be rejected, yielding a conclusive best model.

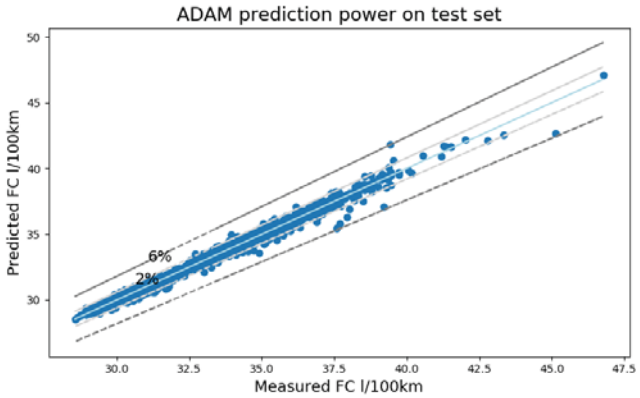

Fig. 3 Prediction accuracy of final model for the VECTO estimation scenario, using the full training set for training. The predicted fuel consumption is shown as a function of the true fuel consumption value

B. Final VECTO Model

The best ADAM model found after iterative hyperparameter searches is defined according to Table III. The performance of the ADAM model is presented in Table IV. Results have been rounded to a 3-digit precision. The root mean squared error (RMSE) is 0.160, the mean relative error (MRE) is 0.3% and the root mean squared relative error (RMSE$_{rel}$) is 0.5%. The prediction error on the 95th percentile (abbreviated 95%) of data is 0.9%.

TABLE III
FINAL SELECTED ADAM-CONFIGURATION FOR VECTO ESTIMATION

Hidden layers	Activation	l2 regularization	γ	β	Learning rate
(120, 120)	sigmoid	0	0.8	0.99	0.0075

TABLE IV
PERFORMANCE FOR THE FINAL ADAM MODEL

MSE	RMSE	MRE	RMSE$_{rel}$	95%	99%	99.9%
0.026	0.160	0.003	0.005	0.009	0.017	0.030

Fig. 3 presents the prediction power of the ADAM-algorithm on the test set. The plot shows the predicted value as a function of the true value, a completely accurate prediction set would thus lie on the light blue line with no deviation. The majority of points lie in the two percent error margin, and no point lies outside the six percent error margin.

C. Operational Estimation

Application of the 5x2 nested CV on the train-set yielded error distributions with median values of 4.66, 5.67, 4.03, and 4.06 for LR, KNN, SGD and ADAM, respectively.

The statistical significance evaluation procedure is presented in the left plot in Fig. 4. Step two in Li's rejection procedure yields to no null hypotheses to be rejected, since P_m in this scenario is 1.0 for the ADAM-SGD pair.

The two algorithms SGD and ADAM are both neural networks. Focusing on SGD as the preferred ANN-algorithm,

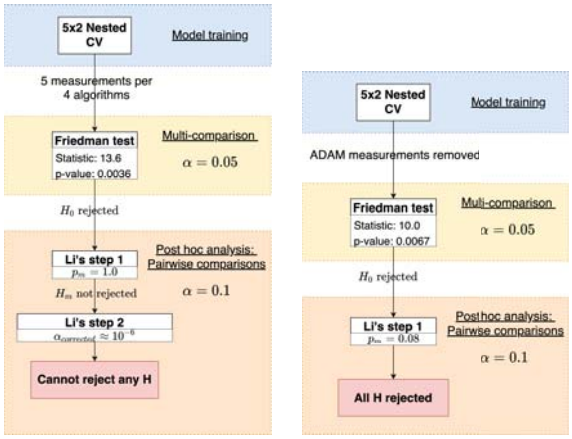

Fig. 4 Statistical evaluation procedure, including (left) and excluding (right) ADAM measurements

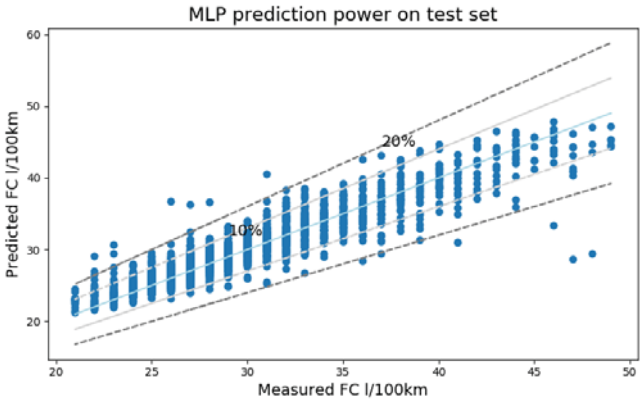

Fig. 5 Prediction accuracy for the final SGD model in the operational estimation scenario

the statistical analysis is performed excluding measurements for ADAM. This yielded a statistically significant difference at the 10% level, described by the right side of Fig. 4. Since all null hypotheses are rejected in the first step of Li's procedure, a statistically significant ranking between SGD, KNN and LR is given.

D. Final Operational Model

The final selected best model trained on the full training set and evaluated on the test set is described by Table V. The resulting performance of the model is presented in Table VI. The model has a RMSE value of approximately 1.714, a mean relative error of 4.2%, and a relative error of 11.2% on the 95th percentile.

TABLE V
FINAL SELECTED SGD-CONFIGURATION FOR OPERATIONAL ESTIMATION

Hidden layers	Activation	l2 regularization	momentum	Learning rate
(70)	sigmoid	0.1	0.775	0.02

TABLE VI
PERFORMANCE FOR THE SGD ALGORITHM TRAINED ON THE FULL TRAIN SET, AND EVALUATED ON THE TEST SET

MSE	RMSE	MRE	$RMSE_{rel}$	95%	99%	99.9%
2.939	1.714	0.042	0.057	0.112	0.167	0.272

Fig. 5 depicts the prediction power of the final model on the test. The distribution of the points in the top-most figures show that the true values have been rounded to the nearest integer. Only a few points lie outside the 20% margin, in accordance with the results presented in Table VI.

V. DISCUSSION AND CONCLUSION

In this study we used vehicle and environmental data in three different machine learning approaches to predict the fuel consumption of the vehicles. From the results we found that ANNs perform better than KNN and LR on the given regression problem. In the VECTO scenario the ADAM algorithm performed best, while in the operational scenario the two ANN algorithms performed equally.

For the final VECTO estimation model, we saw a test MSE of 0.026. Comparing this with the median nested CV error of the ADAM-algorithm (section IV-A), it seems that training for 1000 epochs instead of 200 does not improve the performance by more than a few percent. The final model had a relative error of less than 1% on the 95th percentile, and an absolute mean relative error of roughly 0.3% on average, which should be considered a good estimation of the simulated fuel consumption.

Comparing Figs. 3 and 5 we find that the predictions on the operational scenario are worse by a factor of approximately 10 compared to the VECTO estimations. It is likely that there is not enough information in the input features of the operational scenario available to predict the aggregated fuel consumption value.

This study offers an evaluation framework for finding a best performing algorithm using nested cross-validation alongside statistical hypothesis testing. The testing procedure used the Friedman test as a multi-comparison test and Li's two-step rejection procedure with the Kolmogorov-smirnov 2-sample test to distinguish a best performing algorithm. Once a best-performing algorithm is chosen, it is tuned for hyperparameters and evaluated on the full training set. Using the nested CV and statistical hypothesis testing provides a decision as to which model (if there is any) is best suited for the problem at hand. The step-wise procedure can also be used effectively in an industry context when deciding which model to put into production for a given problem.

The algorithm evaluation procedure in this report made use of statistical analysis tests, such as the Friedman test and Li's two-step rejection procedure. The Friedman test is typically used when assessing m models on k datasets, where the datasets are assumed to be independent. In this study the k datasets are drawn from the five outer folds of the nested CV procedure. While each validation fold is distinct, since no datapoint is present in more than a single fold, the trained models have overlapping train sets. This results in the error

measurement on a fold not being completely independent from another fold, since the same $100 * \frac{k-2}{k}\%$ of the data has been used when training both models.

The statistical analysis was done using the Friedman test, it might be worth consulting other statistical evaluation approaches further to provide a more complete picture of the case. [7], [11], [12], [17], [18].

Concerning the post-hoc analysis, Li's two-step rejection procedure was used. Consulting Trawinski, Smetek, Telec, et al., it may however be preferable to use a different post-hoc strategy since the evaluation in their study concerned NxN hypothesis testing. I.e. all pairs of algorithms are compared. Trawinski, Smetek, Telec, et al. only apply Li's procedure when comparing $Nx1$ tests. In this study, this would identify one algorithm as a "control" algorithm, and remaining algorithms as "test" algorithms to compare with.

A. Conclusions

From our experiments, we state that 1) Artificial Neural networks perform better compared to Linear Regression and K-nearest neighbor. 2) The VECTO estimation scenario can predict fuel consumption error with good accuracy. 3) The operational estimation scenario can predict fuel consumption error accurately, with an error roughly 10 times worse than for the VECTO scenario. We conclude that ANNs can be useful for providing fast and reliable estimates of fuel consumption.

B. Future Work

Further research in this area could be to compare different statistical testing methods to determine other possible null hypothesis rejection procedures within the current case setting. As discussed by other authors [11], [18], the paired t-test is generally considered to perform poorly with a high type 2 error. It would therefor be interesting to compare it with other non-parametrical statistical tests. Other possibilities on evaluating the true model error include bootstrap [7, ch. 5.2] sampling of the sample errors.

The architectures of ANNs and statistical evaluation of machine learning models would be relevant to investigate further. One could consider e.g. establishing deep learning architectures and apply them to fuel consumption. It is however likely that this requires more high resolution data to be useful.

References

[1] E. Union, *Simulation tool for heavy duty vehicles (hdvs)*, https://ec.europa.eu/clima/policies/transport/vehicles/vecto_en, 2020. (visited on 04/27/2020).

[2] L. Ekström, "Estimating fuel consumption using regression and machine learning," Master's thesis, KTH, School of Engineering Sciences, 2018.

[3] H. Almér, "Machine learning and statistical analysis in fuel consumption prediction for heavy vehicles," Master's thesis, KTH, School of Computer Science and Communication (CSC), 2015.

[4] F. Perrotta, T. Parry, and L. C. Neves, "Application of machine learning for fuel consumption modelling of trucks," in *2017 IEEE International Conference on Big Data (Big Data)*, IEEE, 2017, pp. 3810–3815.

[5] N. Hong and L. Li, "A data-driven fuel consumption estimation model for airspace redesign analysis," in *2018 IEEE/AIAA 37th Digital Avionics Systems Conference*. DOI: 10.1109/DASC.2018.8569564.

[6] S. Wickramanayake and H. M. N. Dilum Bandara, "Fuel consumption prediction of fleet vehicles using machine learning: A comparative study," in *2016 Moratuwa Engineering Research Conference (MERCon)*, pp. 90–95. DOI: 10.1109/MERCon.2016.7480121.

[7] G. James, T. Hastie, R. Tibshirani, and D. Witten, *An Introduction to Statistical Learning: With Applications in R*. Springer, 2017, (Online). Available: https://faculty.marshall.usc.edu/gareth-james/ISL/.

[8] R. Rojas, *Neural Networks A Systematic Introduction*. Springer, 1996, (Online). Available: http://page.mi.fu-berlin.de/rojas/neural/index.html.html.

[9] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[10] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, 2014. arXiv: 1412.6980.

[11] S. Varma and R. Simon, "Bias in error estimation when using cross-validation for model selection," *BMC bioinformatics*, vol. 7, no. 1, p. 91, 2006.

[12] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Dec. 2006, ISSN: 1532-4435.

[13] J. D. Li, "A two-step rejection procedure for testing multiple hypotheses," *Journal of Statistical Planning and Inference*, vol. 138, no. 6, pp. 1521–1527, 2008.

[14] H. W. Lilliefors, "On the kolmogorov-smirnov test for normality with mean and variance unknown," *Journal of the American Statistical Association*, vol. 62, no. 318, pp. 399–402, 1967. DOI: 10.1080/01621459.1967.10482916.

[15] P. B. Brazdil and C. Soares, "A comparison of ranking methods for classification algorithm selection," in *European conference on machine learning*, Springer, 2000, pp. 63–75.

[16] D. Shepard, "A two-dimensional interpolation function for irregularly-spaced data," in *Proceedings of the 1968 23rd ACM National Conference*, ser. ACM '68, New York, NY, USA: Association for Computing Machinery, 1968, pp. 517–524, ISBN: 9781450374866.

[17] T. Mitchell, *Machine Learning*, ser. McGraw-Hill International Editions. McGraw-Hill, 1997, ISBN: 9780071154673.

[18] B. Trawinski, M. Smetek, Z. Telec, and T. Lasota, "Nonparametric statistical analysis for multiple comparison of machine learning regression algorithms," *International Journal of Applied Mathematics and Computer Science*, vol. 22, Jan. 2012. DOI: 10.2478/v10006-012-0064-z.