Impact of Telemetry Data Preprocessing on the Accuracy of Fuel Consumption Predictive Models in Heavy-Duty Truck Transport of Sugarcane Stalks

Lucas E Zonfrilli[1]; Almir R Camolesi[2]; Tatiana F Canata[1]

[1]São Paulo State University (UNESP), School of Agricultural and Veterinary Sciences, Jaboticabal, São Paulo, Brazil.
[2]Educational Foundation of the Municipality of Assis (FEMA), School of Computer Science, Assis, São Paulo, Brazil.

A paper from the Proceedings of the
17th International Conference on Precision Agriculture and 11th Brazilian Congress on Precision and Digital Agriculture
13-15 July 2026
Porto Alegre, Brazil

Abstract.

Fuel-consumption efficiency in biomass transport is decisive for the sustainability of agribusiness, yet telemetry data from heavy-duty agricultural vehicles carry intrinsic problems such as embedded-sensor noise and logical inconsistencies. This study aimed to demonstrate that rigorous data preprocessing matters more than algorithmic complexity for predicting fuel consumption, and to locate where that effect arises. The raw dataset comprised 43,112 trips of trucks hauling sugarcane stalks from field to mill, queried from Google BigQuery. A business rule required that the trip distance, divided by two, not deviate more than 30% above or below the mean distance of the harvest zone; combined with removal of null and negative values, this step eliminated 25.9% of the records (11,176 trips). Of the trips with negative readings (sensor errors), 99.8% had already been flagged by this rule, validating the filter. Quartile analysis motivated excluding variables such as moving time, stopped time and mean speed, since 76.5% of their values clustered in the first quartile and were predominantly null. After a final interquartile-range (IQR) outlier removal, the consolidated dataset retained 26,429 records (61.3% of the original). The dependent variable was fuel consumption (litres) and the predictors were the operational telemetry and weighing variables retained after screening. Three algorithms (Random Forest, XGBoost and CatBoost) were evaluated on the treated and raw datasets under an identical feature set and validation protocol. On the treated dataset the gradient-boosting models led: CatBoost reached $R^2 = 0.986$ (MAE 1.16 L; RMSE 1.60 L) and XGBoost $R^2 = 0.986$ (RMSE 1.61 L), while Random Forest reached $R^2 = 0.980$ (RMSE 1.91 L). On the raw dataset CatBoost and XGBoost degraded to $R^2 = 0.589$ and 0.534 (RMSE > 36 L), whereas Random Forest retained a deceptively high $R^2 = 0.936$ (RMSE 14.58 L) whose 44.1% normalised error revealed poor calibration. A two-way ANOVA on cross-validated R^2 confirmed that the preprocessing factor dominated the algorithm factor (η^2 0.39 vs. 0.07; $p = 0.0001$ vs. 0.17). Crucially, a controlled experiment scoring both training origins on a common clean test set isolated this effect: with the evaluation set fixed, training origin changed R^2 by at most 0.6%, and a model trained only on clean data but evaluated on raw data gave the worst result of the entire experiment. The tree-based ensembles are therefore intrinsically robust to training-set contamination, and the decisive contribution of preprocessing lies in the quality of the data presented to the model at evaluation and inference time, corroborated by a SHAP analysis confirming reliance on genuine, physically interpretable predictors. We conclude that data preprocessing is not optional but the determining factor that makes machine learning viable in digital agriculture: a fleet may be trained on minimally processed telemetry provided that every record reaching the model at inference is rigorously

Zonfrilli, L. E., Camolesi, A. R., & Canata, T. F. (2026). Impact of telemetry data preprocessing on the accuracy of fuel consumption predictive models in heavy-duty truck transport of sugarcane stalks. In Proceedings of the 17th International Conference on Precision Agriculture and 11th Brazilian Congress on Precision and Digital Agriculture (unpaginated, online). Monticello, IL: International Society of Precision Agriculture and Brazilian Congress on Precision and Digital Agriculture.

filtered.

Keywords. *Telemetry. Precision Logistics. Data ETL. Machine Learning. Truck.*

Introduction

The sugarcane harvesting system (CTTA—cutting, transshipment, transportation, and support) has an estimated cost of US$ 9.54 per ton (PECEGE Consultoria e Projetos, 2026; values originally reported in Brazilian reais and converted at the 2025 average reference exchange rate of US$ 1.00 = R$ 5.64, Receita Federal do Brasil, 2025). Of this total, the transportation stage from field to mill—widely recognized as one of the most costly and logistically complex operations in the sugarcane supply chain—accounts for approximately 31.3% of the expenses. For delivery radii between 25 and 30 km, the specific cost of this operation reaches US$ 2.98 per ton, covering operators, diesel fuel, maintenance, and equipment leasing. Given the substantial share of diesel in both operational costs and greenhouse gas emissions, accurately predicting its consumption is a strategic lever for the economic competitiveness and sustainability of the sector. In this context, telemetry systems embedded in heavy-duty trucks provide continuous operational variables (such as distance traveled, engine load, idle time, and braking data), ideal inputs for predictive models and decision-support tools.

In practice, however, the value of this information is constrained by the quality of the data rather than by the sophistication of the analytical method applied to it. Telemetry streams produced by embedded sensors are affected by transmission failures, calibration drift, mechanical resets and integration errors between on-board units and the corporate database. These problems manifest as null fields, physically impossible negative readings, duplicated trips and extrapolated distances. A recurrent and largely unquestioned assumption in applied machine learning is that more powerful algorithms compensate for such imperfections, shifting the analyst's effort towards model selection and hyper-parameter tuning while treating data cleaning as a routine preliminary task.

The literature on data mining has long argued the opposite. Preprocessing typically consumes the largest share of an analytical project and is frequently the decisive factor in the final quality of a model (García et al., 2015; Kotsiantis et al., 2006). In agriculture, where machine learning has expanded rapidly across yield prediction, disease detection and machinery management (Kamilaris & Prenafeta-Boldú, 2018; Liakos et al., 2018; Benos et al., 2021; Wolfert et al., 2017), the empirical demonstration of this principle on real, noisy operational data remains scarce, and the interaction between the chosen algorithm and the level of data treatment is rarely quantified. This gap is especially clear in fuel-consumption modelling for vehicle fleets, a subfield that tends to chase accuracy by adding ever more sophisticated learners and input variables: recent reviews report that neural-network and hybrid models increasingly dominate the field and favour deep architectures (Zhao et al., 2023), and state-of-the-art studies on heavy-duty trucks integrate fine-grained signals such as vehicle weight into long short-term memory networks to push accuracy further (Fan et al., 2024). Evidence that the learner is not the binding constraint is nonetheless accumulating: gradient-boosting models such as XGBoost have been shown to match or exceed deep-learning approaches on agricultural prediction tasks while remaining more interpretable and far less data-hungry (Huber et al., 2022). These observations motivate the present study, which asks not which algorithm is best but how much of the achievable accuracy is determined by data treatment rather than by model complexity.

This study addresses that gap using a real-world telemetry dataset of sugarcane-transport trucks. The objective is to demonstrate, with a controlled and reproducible factorial comparison, that rigorous data preprocessing has a greater impact on predictive accuracy than the choice of algorithm. Three widely used tree-based regressors (Random Forest, XGBoost and CatBoost) are evaluated on two versions of the same data: a treated dataset built through an explicit cleaning pipeline and a raw dataset subjected only to minimal type conversion. By decomposing the total

gain into a preprocessing component and an algorithm component, and by analysing the residual behaviour that explains the differing resilience of bagging and boosting, the paper provides quantitative evidence that data treatment is a determining factor, not an optional one, for the viability of machine learning in digital agriculture.

Materials and Methods

Study Context and Dataset

The data originated from the operational telemetry of a fleet of heavy-duty trucks transporting harvested sugarcane stalks from the agricultural fields to a partner mill in Brazil. Telemetry was captured by on-board embedded systems during the 2025 harvest season: operational data were collected between April and November 2025, and the experimental evaluation reported here was carried out in April 2026. The trips were stored and consolidated in Google BigQuery, a fully managed, serverless cloud data warehouse for large-scale SQL analytics (Google LLC, Mountain View, CA, USA), and all analyses were performed in Python (pandas, scikit-learn, XGBoost, CatBoost and statsmodels) in the Google Colab environment with an NVIDIA Tesla T4 GPU runtime (Python 3), querying the consolidated analytical view directly.

The operational cycle comprises three stages: loading of the harvested stalks onto the truck in the field, road transport over unpaved farm roads to the mill, and unloading at the plant. Each trip is materialised at the unloading stage through an automated weighbridge that registers the gross weight of the loaded vehicle and the tare weight of the empty vehicle, the difference yielding the net transported mass; this weighing station is integrated with the fleet telemetry system, which transmits the operational variables of each trip.

The resulting raw extraction contained 43,112 trips described by 35 fields. Each record corresponds to one transport trip and aggregates qualitative attributes (date, work shift, harvest block, truck model, driver, farm, data-management zone and field plot) together with the quantitative telemetry and weighing variables used in this study: trip distance (in kilometres), fuel consumed (in litres), the actual-versus-target fuel efficiency (in kilometres per litre) defined by the sugarcane operation, engine load, idle time, coasting, the percentage of time within and above the economic engine-speed band, braking events, stop events, over-revving, movement and stop durations, and the weighing fields (tare, net and gross weight). The dependent variable of interest was the fuel consumed per trip, measured in litres.

Primary Cleaning (Business Rules)

The first cleaning stage applied domain-derived business rules implemented directly in the BigQuery analytical view and materialised as an outlier flag. Two distance metrics are distinguished throughout this paper: the trip distance—the total distance recorded by the on-board system for a given haul (in kilometres)—and the mean zone distance—the historical average one-way field-to-mill distance of the harvest zone to which the trip belongs. The central rule encoded a physical-consistency constraint linking the two: the trip distance, divided by two to approximate the one-way leg, could not deviate by more than 30% above or below the mean zone distance. This 30% tolerance was calibrated empirically with the operation's logistics team to bracket routine field-to-mill distance variability within a zone, rather than derived from a distributional criterion; a formal sensitivity analysis of the threshold was beyond the scope of this study and is left as future work, although its adequacy is supported a posteriori by the flagged-versus-negative overlap analysis below. Trips violating this constraint were treated as extrapolated or implausible and flagged accordingly. Removing the flagged trips, together with records containing null conversions and physically impossible negative readings, eliminated 25.9% of the base (11,176 trips) and yielded an intermediate set of 31,936 trips.

To justify the rule rather than apply it mechanically, we quantified the overlap between business-flagged trips and trips with negative numeric readings. Of the 3,666 trips with at least one negative value, 3,660 (99.8%) had already been flagged by the distance rule. This near-complete overlap confirmed that negative readings were not independent random errors but a symptom of the same extrapolated or excessively short trips already captured by the physical-consistency rule, supporting the validity of the filter.

Statistical Outlier Analysis and Feature Screening

The intermediate set was then characterised with a univariate statistical summary that, for each numeric variable, computed the mean, standard deviation, the proportion of records in each quartile and the number of outliers detected by the interquartile-range (IQR) criterion (Tukey, 1977), with lower and upper fences placed at Q1 minus 1.5 IQR and Q3 plus 1.5 IQR. This screening revealed two problems. First, several variables presented a strongly degenerate distribution: the moving-time, stopped-time and total-time fields concentrated 76.5% of their mass in the first quartile, where the values were predominantly null, while mean speed and over-revving were almost entirely zero. Carrying little usable information and liable to inject artificial structure into the models, these variables were excluded from the modelling set. Second, the screening quantified, for the retained variables, the number of trips flagged as statistical outliers, providing the basis for the final IQR-based removal.

Construction of the Treated and Raw Datasets

Two datasets were constructed from the same source to isolate the effect of preprocessing. The treated dataset was obtained from the intermediate set by removing the degenerate variables, then removing any trip flagged as an outlier on any retained numeric variable by the IQR criterion, which discarded a further 5,507 trips (17.24%), and finally applying one-hot encoding to the categorical context. This pipeline retained 26,429 trips (82.76% of the intermediate set, or 61.3% of the original extraction). The raw dataset, by contrast, was built from the original 43,112 trips with only minimal handling: numeric type coercion, removal of records with missing distance or fuel, zero-filling of remaining nulls and one-hot encoding; none of the business or statistical cleaning rules were applied. Both datasets shared the same target and the same family of predictors, so that any difference in model performance is attributable to the level of data treatment rather than to differences in variables.

Predictor Variables and Target

The target variable was fuel consumption in litres. The predictor set comprised the operational and weighing variables retained after screening, namely trip distance, mean zone distance, tare weight, net weight, idle time, coasting, percentage of time within the economic band, percentage of time above the economic band, braking events, stop events and engine load, complemented by one-hot dummies for harvest zone and truck model. Gross weight was excluded because it is the deterministic sum of tare and net weight; retaining it produced an effectively infinite variance-inflation factor and added no information to the tree-based models, whereas its two independent components were preserved.

Predictive Modelling and Evaluation

Three tree-based regression algorithms were compared: Random Forest (Breiman, 2001), a bagging ensemble of 100 trees; XGBoost (Chen & Guestrin, 2016), a gradient-boosting ensemble of 1,500 trees with a learning rate of 0.05, maximum depth 6, and 0.8 subsampling of rows and columns; and CatBoost (Prokhorenkova et al., 2018), a gradient-boosting ensemble of 1,500 iterations with a learning rate of 0.05 and depth 8. These hyper-parameters were fixed at values commonly recommended in the literature and not tuned for either dataset: since the aim is to isolate the effect of preprocessing, holding the model configuration constant prevents tuning

differences from confounding the comparison, so any change in performance can be attributed to the data rather than to the search for optimal parameters.

Each of the six combinations of three algorithms and two datasets was first evaluated with a single 80/20 train-test split, with the test partition never used for early stopping to avoid optimistic bias, and performance was reported on the held-out test set as the coefficient of determination (R^2), the mean absolute error (MAE), the root-mean-square error (RMSE) and the normalised RMSE (nRMSE), defined as the RMSE divided by the mean of the observed fuel consumption and expressed as a percentage (Equation 1). The nRMSE is reported because R^2 alone can be misleadingly high when the test set contains extreme values that inflate the total variance.

$$nRMSE = (RMSE / \bar{y}) \times 100\% \qquad (1)$$

To assess stability and test the effect formally, the same models were then re-evaluated with five-fold cross-validation, and a two-way ANOVA was fitted to the per-fold R^2 with dataset and algorithm as factors and their interaction. Finally, three robustness analyses were performed on the treated dataset: a variance-inflation-factor (VIF) analysis of the numeric predictors, a group feature-ablation test that removed both collinear distance variables together, and a SHAP (SHapley Additive exPlanations) analysis (Lundberg & Lee, 2017) of the champion model to quantify the contribution of each predictor and to verify the substitution effect between the two distance variables.

Results and Discussion

Data Quality Before and After Treatment

The statistical-screening stage is illustrated in Figure 1, which ranks the variables by the number of trips flagged as outliers under the IQR criterion. The time-based variables (stopped time, moving time and total time) dominate the ranking with roughly 7,500 flagged trips each, a direct consequence of their degenerate distribution and the reason for their exclusion from the modelling set. Behavioural and load variables present far fewer outliers, on the order of a few hundred to about 1,700 trips, which the final IQR removal could discard without compromising sample size.

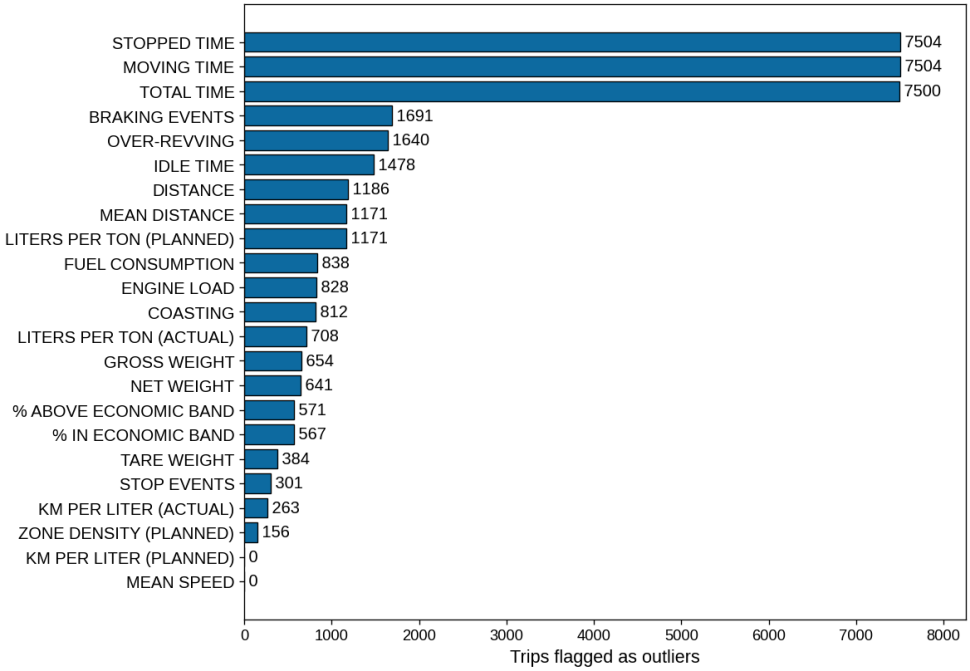

Fig 1. Number of trips flagged as statistical outliers per variable using the interquartile-range (IQR) criterion. The dominance of the time-based variables motivated their exclusion from the modelling set.

The effect of the cleaning pipeline on data integrity is summarised in Figure 2, which reports, for each telemetry variable, the percentage of records carrying a valid (non-zero) reading in the raw and in the treated datasets. In the raw data the behavioural variables that depend on continuous on-board sensing were the most affected: engine load, the percentage of time above the economic band, and the braking- and stop-event counts carried a valid reading in only about 71% of records, indicating that roughly three in ten trips lacked a usable measurement for these fields. The cleaning pipeline raised data density unevenly, in line with the physical origin of each defect: the structurally complete variables (distance, tare weight and net weight) reached full coverage, while idle time, coasting and the percentage within the economic band rose to about 96%. Engine load, the percentage above the economic band and the braking- and stop-event counts remained close to 72% even after treatment, indicating that for these fields the missing values are not noise removable by row-level cleaning but genuine operational zeros and persistent sensor gaps, a distinction that matters when interpreting their predictive role.

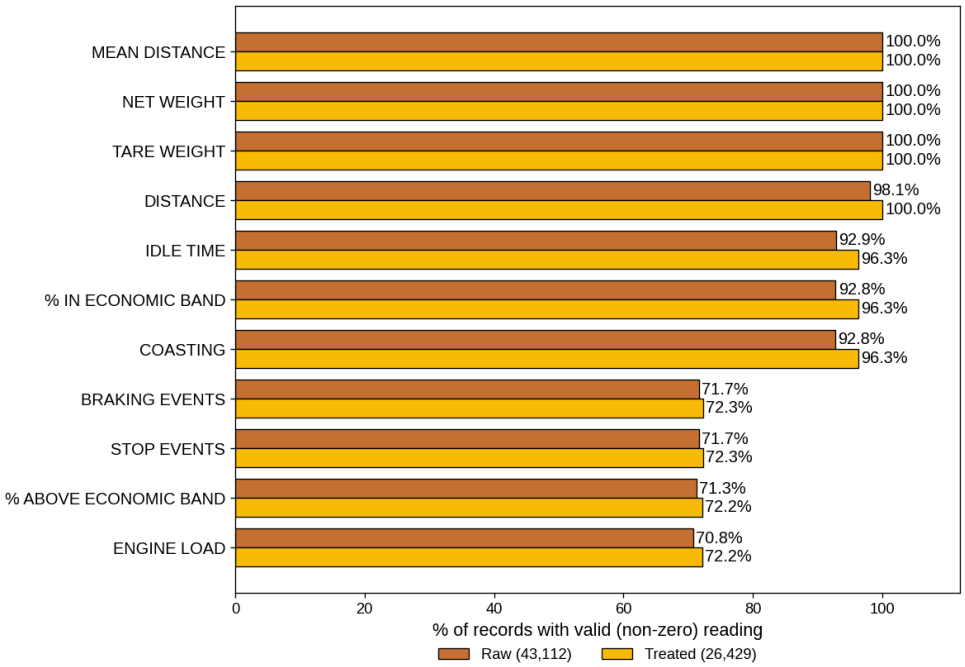

Fig 2. Percentage of records with a valid (non-zero) reading per telemetry variable, in the raw dataset (43,112 trips) and in the treated dataset (26,429 trips).

The nature of these defects is best understood from the data-acquisition pipeline described in the Materials and Methods. Because the trip record is anchored on the weighbridge event, the structurally complete fields, namely gross weight, tare weight, net weight and distance, are captured reliably (close to 100% valid readings), whereas the behavioural variables that depend on continuous on-board sensing during transport over unpaved farm roads, such as engine load, the economic-band shares, braking and stop events, are the ones most affected by transmission gaps and resets, which is consistent with their roughly 71% valid-reading rate in the raw data. Likewise, the negative and extrapolated distance readings that the business rule removed are an expected artefact of odometer resets and of round-trip records being attributed to a single leg, which explains why nearly all of them coincided with the trips already flagged by the distance rule. The cleaning pipeline is therefore not an arbitrary statistical filter but a procedure aligned with the physical mechanism that generates the data.

Relationships Among Predictors

Figure 3 presents the Pearson correlation matrix among the modelling variables. Fuel consumption is most strongly associated with trip distance (r = 0.96) and mean zone distance (r = 0.93), followed by the percentage of time within the economic band (r = 0.79) and engine load (r = 0.73), confirming that the dominant drivers of consumption are the length of the route and the operating regime of the engine. The very high correlation between trip distance and mean zone distance (r = 0.98) anticipates a multicollinearity concern that is examined quantitatively in the robustness analysis. Tare weight shows only a weak negative correlation with consumption (r = −0.37); as a near-static vehicle attribute rather than a per-trip operating variable, it is not expected to be a direct physical driver of fuel use, and it is therefore interpreted with caution and retained only as a low-importance predictor, consistent with its minor SHAP contribution.

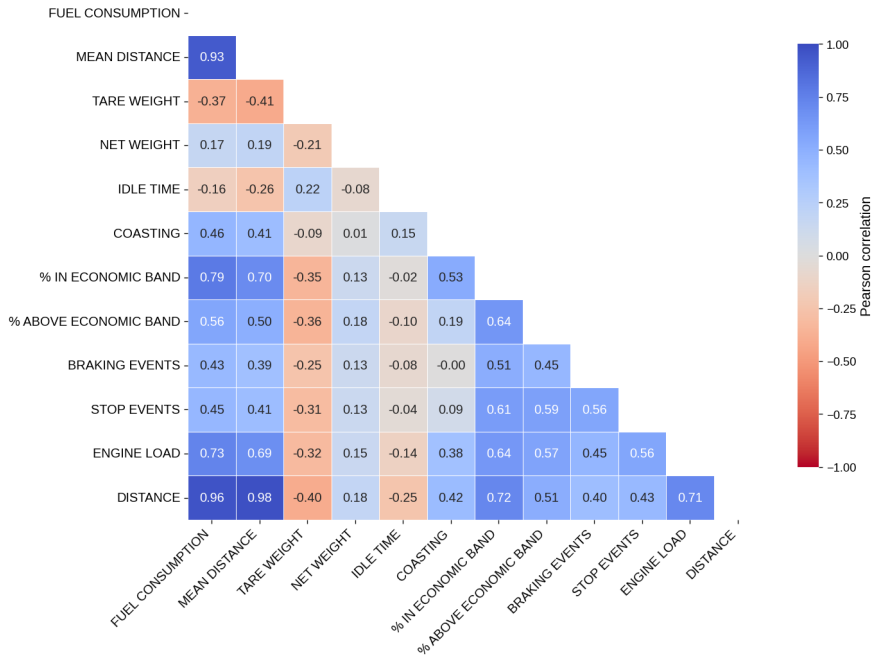

Fig 3. Pearson correlation matrix among the modelling variables on the treated dataset. Fuel consumption is driven mainly by distance, mean distance, the economic-band share and engine load.

Predictive Performance: Treated versus Raw

Table 1 reports the single-split performance of the six model-dataset combinations. On the treated dataset all three algorithms reached excellent and almost indistinguishable accuracy: CatBoost led with R^2 = 0.986 and RMSE 1.60 L, XGBoost matched it at R^2 = 0.986 and RMSE 1.61 L, and Random Forest followed closely at R^2 = 0.980 and RMSE 1.91 L. On the raw dataset, however, the two boosting algorithms collapsed: XGBoost fell to R^2 = 0.534 with an RMSE of 39.34 L and CatBoost to R^2 = 0.589 with an RMSE of 36.95 L, errors more than twenty times larger than on the treated data. Random Forest was the notable exception, retaining a high R^2 of 0.936 even on the raw data, although its RMSE still rose to 14.58 L, almost eight times its treated value.

Table 1. Predictive performance of the three algorithms on the raw and treated datasets (single 80/20 split). All metrics were computed on the held-out test set; nRMSE is the RMSE normalised by the mean observed fuel consumption.

Dataset	Model	R^2	MAE (L)	RMSE (L)	nRMSE (%)	N (test)
Raw	Random Forest	0.936	1.81	14.58	44.1	8,623
Raw	XGBoost	0.534	2.77	39.34	119.1	8,623
Raw	CatBoost	0.589	2.66	36.95	111.8	8,623
Treated	Random Forest	0.980	1.38	1.91	6.2	5,286
Treated	XGBoost	0.986	1.16	1.61	5.3	5,286
Treated	CatBoost	0.986	1.16	1.60	5.2	5,286

Decomposition of the Gain: Preprocessing versus Algorithm

The central question of the study is whether the observed accuracy is produced mainly by the choice of algorithm or by the data treatment. Decomposing the improvement makes the answer unambiguous. Moving from the raw to the treated dataset raised R^2 by 84.7% for XGBoost and by 67.5% for CatBoost, and even the resilient Random Forest gained 4.7% (Figure 4). In sharp contrast, once the data were properly treated, switching between the best and the worst of the three algorithms changed R^2 by only 0.6%. In other words, the preprocessing effect was roughly two orders of magnitude larger than the algorithm effect.

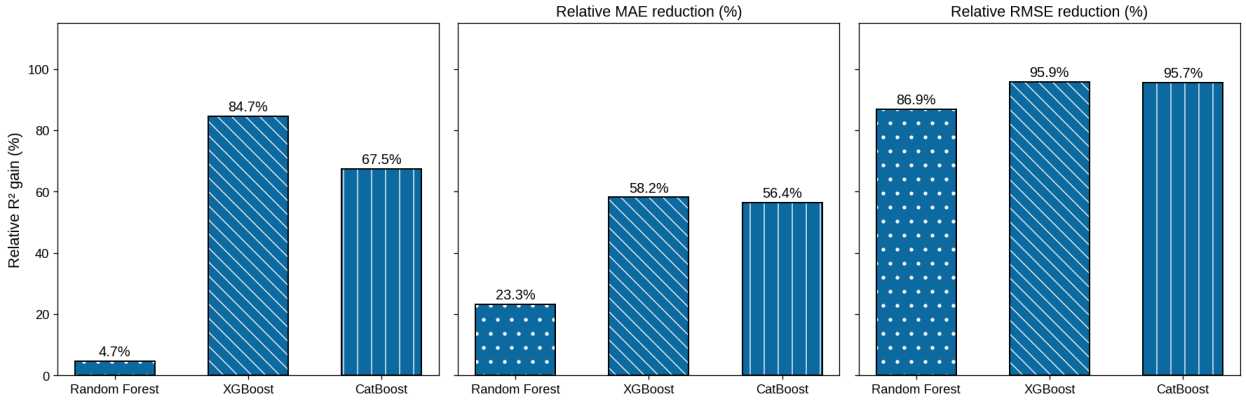

Fig 4. Relative gain in R^2 obtained by moving from the raw to the treated dataset, per algorithm. Preprocessing yields up to an 84.7% improvement.

This conclusion was confirmed formally by a two-way ANOVA on the per-fold cross-validated R^2. The dataset (preprocessing) factor was highly significant and explained 39% of the variance ($F = 20.72$; $p = 0.0001$; $\eta^2 = 0.39$), whereas the algorithm factor was not significant and explained only 7% ($F = 1.92$; $p = 0.17$; $\eta^2 = 0.07$), and the interaction was likewise non-significant ($\eta^2 = 0.08$). The statistical model thus attributes the dominant explained variation to how the data were prepared, not to which learner was used.

Cross-Validation and Stability

Table 2 reports the five-fold cross-validated means and standard deviations. The cross-validation reinforces the single-split picture and adds a crucial stability dimension. On the treated dataset the three algorithms were not only accurate (mean R^2 between 0.981 and 0.986) but extraordinarily stable, with R^2 standard deviations of 0.0003 to 0.0007 and RMSE standard deviations below 0.03 L. On the raw dataset the same models were both worse and far more erratic: the R^2 standard deviation rose to roughly 0.15 to 0.19 and the RMSE standard deviation to between 9.9 and 14.2 L, meaning that the raw-data performance depended heavily on which trips happened to fall in each fold. Preprocessing therefore improves not only the expected accuracy but its reproducibility, a property that is essential for an operational decision-support tool. Because cross-validation was applied within each dataset separately, the folds are equally sized within a dataset but differ in size between datasets (about 8,622 trips per fold for the raw

dataset and about 5,286 for the treated one); the controlled experiment reported below neutralises this difference by scoring every model on a single common test set.

Table 2. Five-fold cross-validation results (mean ± standard deviation) by algorithm and dataset.

Dataset	Model	R^2 (mean ± sd)	RMSE (mean ± sd)	MAE (mean ± sd)
Raw	Random Forest	0.884 ± 0.152	15.94 ± 9.86	1.86 ± 0.15
Raw	XGBoost	0.660 ± 0.191	30.72 ± 12.28	2.50 ± 0.33
Raw	CatBoost	0.788 ± 0.184	23.85 ± 14.21	2.35 ± 0.35
Treated	Random Forest	0.981 ± 0.0007	1.88 ± 0.03	1.37 ± 0.02
Treated	XGBoost	0.986 ± 0.0004	1.60 ± 0.03	1.15 ± 0.02
Treated	CatBoost	0.986 ± 0.0003	1.60 ± 0.02	1.16 ± 0.01

The contrast between MAE and RMSE is itself informative. From raw to treated, the MAE of the boosting models fell by about 56 to 58% (CatBoost 2.66 to 1.16 L; XGBoost 2.77 to 1.16 L), but their RMSE fell by about 96% (CatBoost 36.95 to 1.60 L; XGBoost 39.34 to 1.61 L). Because RMSE penalises large errors far more heavily than MAE, the much larger relative drop in RMSE shows that the degradation on raw data was driven by a small number of very large errors, that is by the extreme outliers, rather than by a uniform loss of accuracy across all trips.

Why Boosting Collapses on Raw Data: Residual Diagnostics

The residual diagnostics in Figures 5 and 6 explain the differing resilience of the two ensemble paradigms. On the raw dataset (Figure 5) the predicted-versus-observed cloud departs strongly from the identity line, the residual distribution is dominated by a heavy tail, and the residuals-versus-fitted plot shows extreme points reaching several hundred litres—the signature of a model pulled away from the bulk of the data by a handful of extrapolated trips. On the treated dataset (Figure 6) the same XGBoost model produces a tight, symmetric residual distribution centred near zero (standard deviation of about 1.6 L) and a homoscedastic residuals-versus-fitted pattern.

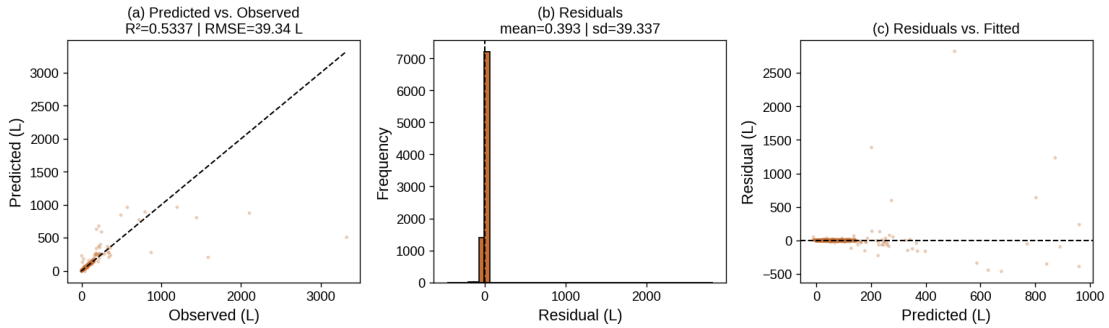

Fig 5. Residual diagnostics of XGBoost on the raw dataset (worst case): (a) predicted versus observed, (b) residual distribution and (c) residuals versus fitted values. The heavy tail reveals the influence of extreme outliers.

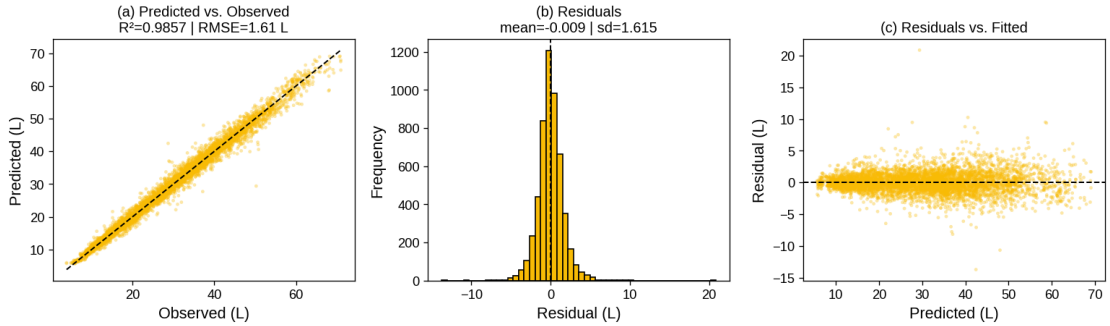

Fig 6. Residual diagnostics of XGBoost on the treated dataset: (a) predicted versus observed, (b) residual distribution and (c) residuals versus fitted values. Residuals are tight, symmetric and homoscedastic.

The residual pattern admits two non-exclusive explanations. The first is intrinsic to how each ensemble is built: Random Forest aggregates many trees trained on independent bootstrap samples, so the error of the few trees exposed to outliers is diluted by the majority that are not, whereas boosting fits each new tree to the residuals of the previous ones and can therefore chase extreme, physically impossible values during training. The second explanation, easily overlooked, lies in the evaluation itself: because the raw-data metrics in Table 1 and Figure 5 were obtained by testing on a partition that still contained those extreme trips, the heavy residual tail and the inflated RMSE partly reflect outliers in the test set rather than a genuine failure of the model on ordinary trips. These two contributions—contamination of the training data and contamination of the evaluation data—are confounded in every comparison so far, because each dataset was scored on its own test partition: a low score on the raw data could mean either that the model genuinely learned less well or simply that it was judged on a harder, outlier-laden test set, and the two readings cannot be separated as long as the test set changes together with the training set. Disentangling them requires holding the evaluation set fixed while varying only the training data, which is the purpose of the controlled experiment reported next.

Isolating the Locus of the Effect: Training versus Evaluation Contamination

To separate the effect of cleaning the training data from the effect of cleaning the evaluation data, a controlled experiment was performed in which the test set was held identical across all conditions. A single clean test set of 5,286 trips was drawn from the treated dataset and, to prevent any leakage, those exact trips were removed by index from the raw training pool. Each algorithm was then trained twice, once on the raw training pool (37,826 trips) and once on the treated training pool (21,143 trips), and both versions were scored on that same clean test set. Because the evaluation inputs are now identical, any difference in performance is attributable solely to the contamination of the training data.

Table 3. Performance on a single common clean test set (5,286 trips) for models trained on the raw and on the treated data. With the evaluation set held fixed, the training origin barely affects accuracy.

Model	R^2 (raw-trained)	R^2 (treated-trained)	RMSE raw (L)	RMSE treated (L)
Random Forest	0.981	0.980	1.88	1.91
XGBoost	0.984	0.985	1.69	1.63
CatBoost	0.980	0.986	1.91	1.60

With the evaluation set held fixed, the origin of the training data becomes almost irrelevant: CatBoost trained on the raw pool reached $R^2 = 0.980$ against 0.986 when trained on the treated pool, a difference of only 0.6%, while XGBoost and Random Forest differed by about 0.1%, the raw-trained forest being marginally better. Even when fed a contaminated training pool, then, the tree-based ensembles lose less than one percent of accuracy on clean inputs, confirming that they are intrinsically robust to contamination of the training set.

To map the full design, the champion CatBoost model was additionally evaluated in a two-by-two factorial that crosses the origin of the training data (raw or treated pool) with the nature of the evaluation set (raw or clean). To keep the comparison on identical trips, the factorial is anchored on a single raw 20% test partition: the raw-evaluation column scores the models on that partition with its extreme trips retained, while the clean-evaluation column scores them on the subset of those very same trips that survive cleaning. Because the two columns differ only in the presence of contaminated trips — not in which trips are sampled — the four R^2 values (Figure 7) isolate the influence of training origin (rows) from that of evaluation quality (columns). The test trips are disjoint by index from each training pool to prevent leakage.

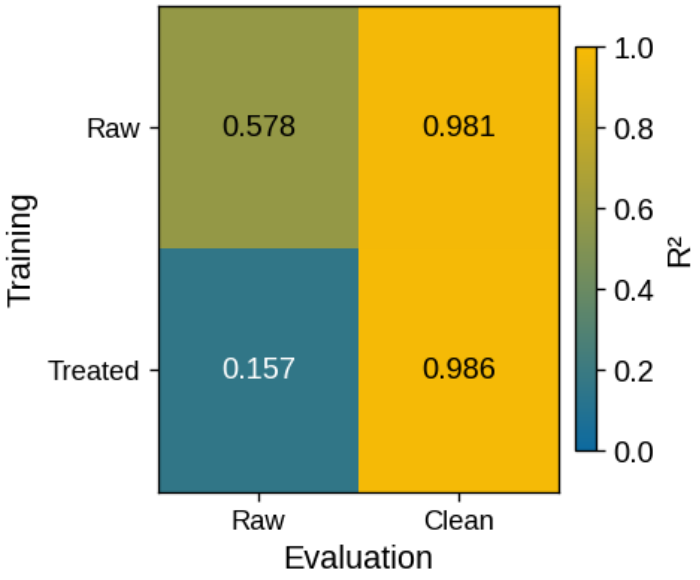

Fig 7. Coefficient of determination (R^2) of the champion CatBoost model in a two-by-two factorial that crosses the origin of the training data (rows: raw or treated) with the nature of the evaluation set (columns: raw or clean).

The factorial makes the locus of the effect unmistakable. Reading across the columns, the evaluation axis dominates: R^2 ranges from 0.157–0.578 when the champion model is judged on contaminated data to 0.981–0.986 when it is judged on clean data, a spread of about 0.83, whereas reading down the rows the training axis is comparatively minor, with a spread of about 0.42. The CatBoost model that, in Table 1, appeared to collapse to $R^2 = 0.589$ on raw data in fact recovered $R^2 = 0.981$ once evaluated on a clean test set, even when trained on the contaminated pool; the apparent collapse was not a failure of learning but an artefact of evaluation.

A counter-intuitive cell of Figure 7 deserves emphasis because it answers a natural objection, namely whether a model trained exclusively on clean data might come to treat the messy operational reality as noise. The model trained on the treated pool and evaluated on raw data attained the lowest score of the entire experiment ($R^2 = 0.157$), below even the model both trained and evaluated on raw data ($R^2 = 0.578$). A learner that never sees the extrapolated and reset-induced regimes during training has no representation for them and fails hardest when they appear at inference, whereas a learner exposed to the full raw distribution during training is partially inoculated against them. This asymmetry confirms that the binding constraint is the quality of the data presented to the model at evaluation and inference time, not the purity of the training corpus.

This finding refines, rather than contradicts, the central thesis. Preprocessing remains the determining factor, but its decisive contribution is concentrated on the data that enter the model at evaluation and inference time. In operational terms, the tree-based ensembles studied here are robust enough to be trained on large volumes of minimally processed telemetry, trading additional machine processing for a substantial saving in manual cleaning effort, provided that

the records on which predictions are produced and judged are rigorously filtered. This points to a practical architecture in which the model is trained on minimally processed telemetry but a rigorous validation filter is applied at inference time, so that only physically consistent records reach the model when predictions are generated and judged.

Robustness: Multicollinearity and Feature Ablation

Two robustness checks confirm that the excellent performance on the treated dataset is not an artefact of redundant predictors. The variance-inflation-factor analysis (Table 4) identified only two variables with elevated multicollinearity, the trip distance (VIF 24.0) and the mean zone distance (VIF 22.0), consistent with their 0.98 correlation; every other predictor fell below 5. Adopting the conventional interpretation in which a VIF above 10 indicates severe multicollinearity, these two values are formally high, yet they are acceptable in the present context for two reasons. First, tree-based ensembles split on one variable at a time and are intrinsically tolerant of correlated inputs, so the redundancy does not bias their predictions. Second, as the ablation below shows, the redundancy is informative rather than spurious, since the two distance variables encode the same physical quantity from two sources. The only methodological consequence is that a naive single-variable ablation would underestimate the importance of route length, which is why the group ablation removes both distance variables together.

Table 4. Variance-inflation factors (VIF) of the numeric predictors on the treated dataset; VIF > 10 indicates severe multicollinearity.

Predictor	VIF	Note
Trip distance (km)	24.04	high (corr. with mean distance)
Mean zone distance	22.00	high (corr. with distance)
% within economic band	4.29	acceptable
Engine load	2.67	acceptable
Stop events	2.28	acceptable
% above economic band	2.11	acceptable
Coasting	1.96	acceptable
Braking events	1.75	acceptable
Tare weight	1.31	acceptable
Idle time	1.22	acceptable
Net weight	1.07	acceptable

This substitution effect is quantified in Table 5. Removing only the dominant trip-distance variable lowered the CatBoost R^2 marginally, from 0.986 to 0.975 (a relative loss of just 1.1%, with RMSE rising to 2.14 L), because the model simply fell back on the collinear mean-zone-distance feature as a proxy and the true importance of route length remained masked. When both distance variables were removed simultaneously, in a group ablation that eliminates the distance information rather than a single column, the R^2 fell to 0.958 and the RMSE rose from 1.60 to 2.78 L, a relative R^2 loss of 2.9% and an RMSE increase of 73%. The naive single-variable ablation therefore underestimated the role of distance by roughly a factor of three. Crucially, even with all explicit distance information removed the model retained an R^2 of 0.958: the behavioural and load variables carry substantial, non-redundant predictive information, and the model degrades gracefully rather than collapsing when deprived of its strongest predictors.

Table 5. Group feature ablation on the treated dataset (CatBoost, fixed 80/20 split). Removing a single distance variable underestimates its importance because the collinear partner acts as a proxy; only the group ablation, which removes both, reveals the true effect.

Scenario	R^2	RMSE (L)	ΔR^2 (%)	ΔRMSE (%)
Baseline (all features)	0.986	1.60	—	—
− Distance only	0.975	2.14	−1.1	+33.4
− Both distance features	0.958	2.78	−2.9	+73.4

Model Interpretability (SHAP)

The SHAP analysis of the champion CatBoost model on the treated dataset, summarised in Figure 8, makes the contribution of each predictor explicit and visually corroborates the ablation result. The mean absolute SHAP value ranks trip distance first by a wide margin (5.51 L), followed by the percentage of time within the economic band (3.15 L), engine load (2.38 L) and the mean zone distance (1.16 L); the remaining predictors each contribute less than 0.5 L. Crucially, SHAP reveals that the masking between the two distance variables is asymmetric rather than an even split: trip distance is the dominant signal (5.51 L) and the mean zone distance acts as a redundant reserve (1.16 L) on which the model falls back. This explains why removing trip distance alone barely changes R^2, since the mean zone distance reassumes the signal, while removing both distance variables together lowers R^2 by 2.9% and raises RMSE by 73%, reconciling the SHAP ranking with the group-ablation result and confirming that the model relies on genuine, physically interpretable information.

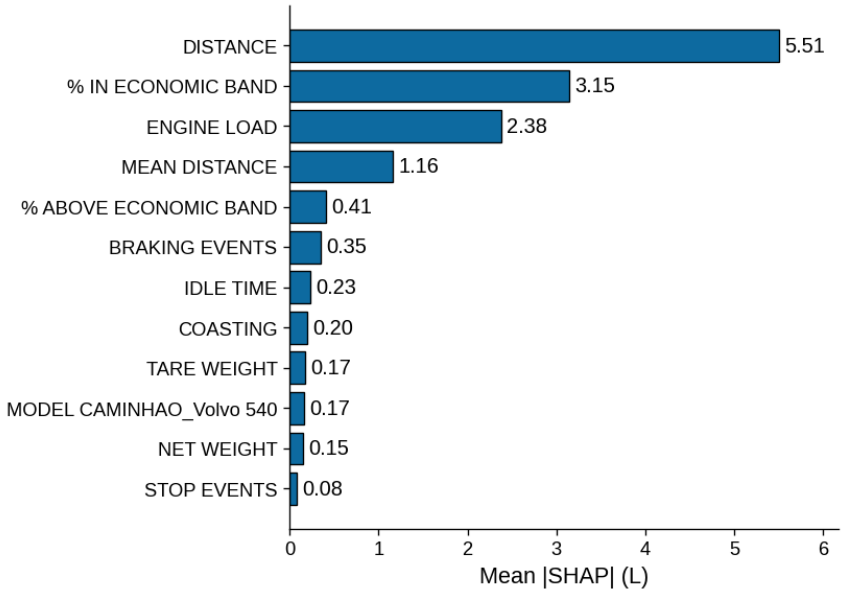

Fig 8. SHAP analysis of the champion CatBoost model on the treated dataset: Mean absolute SHAP value (impact on predicted fuel consumption, L) per predictor.

Limitations

Several limitations bound these conclusions. The analysis draws on a single fleet serving one mill during the 2025 harvest season, so the absolute error figures and cleaning thresholds should be re-estimated before transfer to other fleets, crops or regions. The hyper-parameters of the three algorithms were fixed at commonly recommended values rather than tuned; this isolates the effect of preprocessing but leaves open whether targeted tuning would narrow or widen the small gap observed between learners. The robustness to training-set contamination was established only for tree-based ensembles, and model families that interpolate more aggressively—such as neural networks or unregularised linear models—may not share this property; testing them is left for future work. Finally, the recommendation to train on minimally processed telemetry while filtering

rigorously at inference was demonstrated in a controlled offline experiment; its operational value still requires validation in a deployed decision-support setting, where the inference-time filter must run in real time on streaming records.

Conclusions

This study tested, on real sugarcane-transport telemetry, whether rigorous data preprocessing matters more than the choice of algorithm for predicting fuel consumption. The evidence is unambiguous. A cleaning pipeline combining a physically grounded business rule, screening of degenerate variables and IQR-based outlier removal raised the gradient-boosting R^2 from below 0.59 to 0.986 and cut RMSE by about 96% (from more than 36 L to roughly 1.6 L), whereas switching between algorithms on treated data changed R^2 by only 0.6%; a two-way ANOVA confirmed that preprocessing dominated the algorithm factor (η^2 0.39 vs. 0.07).

A controlled experiment with a fixed test set refined this finding: the decisive effect of preprocessing lies in the data evaluated and served to the model, not in the purity of the training corpus. The tree-based ensembles proved robust to contaminated training data: CatBoost recovered R^2 above 0.98 on a clean test set even when trained on the raw pool, yet failed when forced to predict on contaminated data. The practical implication is direct: in digital agriculture, investment in data quality returns more than investment in algorithmic sophistication, and a fleet operator may train on minimally processed telemetry provided that the records used for inference are rigorously filtered. Future work will extend the pipeline to additional fleets and crops and evaluate algorithm families beyond tree-based ensembles (such as neural networks and regularised linear models) to test whether the robustness to training-set contamination observed here is specific to tree ensembles or generalises more broadly.

Acknowledgments

The authors thank the São Paulo State University (UNESP), School of Agricultural and Veterinary Sciences, Jaboticabal, and the partner mill for providing the operational telemetry data used in this study. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

References

Benos, L., Tagarakis, A. C., Dolias, G., Berruto, R., Kateris, D., & Bochtis, D. (2021). Machine learning in agriculture: A comprehensive updated review. Sensors, 21(11), 3758. https://doi.org/10.3390/s21113758

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). ACM. https://doi.org/10.1145/2939672.2939785

Fan, P., Song, G., Zhai, Z., Wu, Y., & Yu, L. (2024). Fuel consumption estimation in heavy-duty trucks: Integrating vehicle weight into deep-learning frameworks. Transportation Research Part D: Transport and Environment, 131, 104157. https://doi.org/10.1016/j.trd.2024.104157

García, S., Luengo, J., & Herrera, F. (2015). Data preprocessing in data mining. Springer. https://doi.org/10.1007/978-3-319-10247-4

Huber, F., Yushchenko, A., Stratmann, B., & Steinhage, V. (2022). Extreme gradient boosting for yield estimation compared with deep learning approaches. Computers and Electronics in Agriculture, 202, 107346. https://doi.org/10.1016/j.compag.2022.107346

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. Computers and Electronics in Agriculture, 147, 70–90. https://doi.org/10.1016/j.compag.2018.02.016

Kotsiantis, S. B., Kanellopoulos, D., & Pintelas, P. E. (2006). Data preprocessing for supervised learning. International Journal of Computer Science, 1(2), 111–117.

Liakos, K. G., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine learning in agriculture: A review. Sensors, 18(8), 2674. https://doi.org/10.3390/s18082674

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In Advances in Neural Information Processing Systems (Vol. 30, pp. 4765–4774).

Pecege Consultoria e Projetos. (2026). Custos de produção da cana-de-açúcar, açúcar e etanol — safra 2025/26 [Sugarcane production cost survey]. Expedição Custos Cana 2026, Piracicaba, SP, Brazil. https://www.custoscana.com.br/

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased boosting with categorical features. In Advances in Neural Information Processing Systems (Vol. 31, pp. 6638–6648).

Receita Federal do Brasil. (2025). Tabelas para conversão de dólar em 2025 [USD-to-BRL conversion tables for income-tax purposes]. Ministério da Fazenda, Brasília, DF, Brazil. https://www.gov.br/receitafederal/pt-br/assuntos/meu-imposto-de-renda/tabelas/conversao/2025

Tukey, J. W. (1977). Exploratory data analysis. Addison-Wesley.

Wolfert, S., Ge, L., Verdouw, C., & Bogaardt, M.-J. (2017). Big data in smart farming – A review. Agricultural Systems, 153, 69–80. https://doi.org/10.1016/j.agsy.2017.01.023

Zhao, D., Li, H., Hou, J., Gong, P., Zhong, Y., He, W., & Fu, Z. (2023). A review of the data-driven prediction method of vehicle fuel consumption. Energies, 16(14), 5258. https://doi.org/10.3390/en16145258