

USING MACHINE LEARNING TO PREDICT PROCESSING TIMES IN AN ENGINEER TO ORDER ENVIRONMENT: A CASE STUDY

Pedro Amaral Pereira

Instituto Nacional de Tecnologia (INT)
pedro.amaral@int.gov.br

Andréa Regina Nunes de Carvalho

Instituto Nacional de Tecnologia (INT)
andrea.carvalho@int.gov.br

Maria Luiza Cantanhede Guillaume

Instituto Nacional de Tecnologia (INT)
marialuizacw@poli.ufrj.br

Iuri Martins Santos

Petrobras
iurims@petrobras.com.br

Raquel da Conceição Santos

Instituto Nacional de Tecnologia (INT)
raquelsantoos1950@gmail.com

Manoel Carlos Pego Saisse

Instituto Nacional de Tecnologia (INT)
manoel.saisse@int.gov.br

ABSTRACT

The Engineer-to-Order (ETO) context is characterized by high demand variability and uncertainty, making it particularly challenging to predict processing times accurately at the proposal stage. This study developed predictive models to estimate processing times using real historical data from an ETO company specialized in the production of highly complex equipment. A complete data science pipeline was implemented, encompassing data preprocessing, exploratory analysis, and predictive algorithm modeling. Sixty models were generated by combining data encoding strategies with machine learning algorithms, supported by statistical analyses of data distribution and goodness-of-fit tests. Although predictive performance was limited, the study yielded valuable insights into challenges specific to the ETO context—such as the distinction between lead time and processing time, limitations of product groupings, data variability, and the difficulty of developing generalized models—thereby contributing to the advancement of ML applications in project-driven production environments, which are still underexplored.

KEYWORDS. Machine Learning. Engineer to Order. Processing Times.

Paper topics: Operations Research in Production Management and Administration. Operational Research Analytics in Statistics and Machine Learning.

1. Introduction

In an Engineer-to-Order (ETO) environment, while accurate prediction of lead time is crucial for setting realistic delivery commitments to customers, the estimation of processing time plays a central role in cost estimation and price definition [Chehade et al., 2024; Rapaccini et al., 2023]. The high degree of customization in ETO environments adds considerable uncertainty to production planning, making it difficult to reliably estimate processing times. [Fortes et al., 2023]. At the bidding stage, critical product information—such as the final design, bill of materials, and routing—is often not fully defined, despite this, companies must still commit to promised delivery dates and pricing. This challenge is even more complex in ETO contexts dedicated to the production of large-scale industrial equipment, where product structures are highly intricate and deeply hierarchical. The uniqueness of ETO products—often built as one-of-a-kind solutions—limits the direct applicability of historical data. While historical inputs may support the development of predictive models, their generalizability is constrained by the variability inherent in ETO production [Rapaccini et al., 2023]. Nevertheless, patterns may emerge within the company's core business areas, where similar processes and technical expertise are repeatedly employed, offering a foundation upon which machine learning (ML) models can be applied.

Current scientific literature on ML-based models in ETO contexts is often constrained to specific product families or processes characterized by higher degrees of standardization. This enables them to deal with typical ETO challenges in a manner closely aligned with Make-to-Order (MTO) [Chehade et al., 2024] or Make-to-Stock (MTS) production contexts [Sherwin et al., 2025]. Such an approach is commonly adopted to address the inherent complexity and variability of data generated in ETO systems, where decision-making still heavily relies on the expertise and tacit knowledge of managers and people related to the production system [Sylla et al., 2021]. Narrowing the scope of the problem to specific product families, segments or operations within a wider context can lead to more homogeneous and less complex datasets, improving the performance of predictive models in localized areas. However, this approach can result in loss of valuable information narrowing the model's degree of generality and subsequently limiting the ability to define action plans for broader operational contexts [Chehade et al., 2024]. This complexity is reflected in the limited number of studies involving real-world applications of the ML-ETO binomial, particularly those addressing prediction of processing times and lead times, which are essential for estimating costs, setting prices and due dates.

To address this research gap, this work aims to develop a predictive model, focused on estimating processing times, using real historical data from an ETO company specializing in the production of unique and highly complex industrial equipment. In this case study, accurate estimation of processing times can lead to better control over the production resources and delivery times, reducing the uncertainties related to production costs. Although the predictive performance of the model was limited, the study yielded valuable insights into the characteristics of the ETO context, the quality and structure of the available data, the design of the ML pipeline, and the selection of performance evaluation metrics. These aspects are critically examined and discussed throughout the paper. Given the scarcity of research on the application of ML techniques in ETO environments, this work contributes to the advancement of knowledge in this emerging field.

The remainder of this paper is organized as follows. Section 2 briefly presents the theoretical background on the application of ML in the ETO context. Section 3 describes the production environment, and the problem investigated in this study. Section 4 presents the methodology used to develop the predictive models. Section 5 reveals the results obtained by the models. Section 6 discusses the problem and the results. Section 7 presents the conclusion and consequences of this study.

2. Theoretical background

In the literature, few studies explore the use of ML models in ETO environments, and those that do typically focus on predictive models built from historical production data. The objectives of these studies include: predicting production costs based on product characteristics [Rapaccini et al., 2023; Ziegler et al., 2022]; estimating manufacturing times to set prices [Chehade et al., 2024]; reducing production delays by anticipating variations in component quantities [Burggraf et al., 2021]; predicting component maintenance needs through equipment performance monitoring [Cao et al., 2024]; predicting the behavior of external variables to support decision-making, such as the performance of suppliers [Sherwin et al., 2025].

Among studies exploring the ML-ETO relationship and its impact on prediction of production costs, Rapaccini et al. (2023) implemented a predictive model with managerial implications for cost estimation in a large ETO oil and gas company. The authors employed a Random Forest algorithm to develop an ML model capable of handling different combinations of variables, predominantly numerical/quantitative. The model's outcomes were decisive in defining cyclical action plans. Ziegler et al. (2022), in an attempt to improve the accuracy of cost quotations by selecting the most suitable machine tools, applied artificial neural networks (ANN) to a database containing historical operational and financial data. The models produced an acceptable mean absolute error (MAE), and the results contributed to the automation of cost quotations.

In the ETO context, accurate estimations of processing and lead times is essential for both cost calculation and price setting. Considering alternatives for lead time prediction in ETO-based environments using ML techniques, Burggraf et al. (2020) highlighted a total of 23 publications looking at different databases, where models based on ANN were the most common (42%) followed by simple linear regression models (30%) and decision trees (26%). Chehade et al. (2024) developed predictive models based on manufacturing times and characteristics of product families, processing unstructured data to capture more information about the production process. Among the implemented models, Catboost returned the best results, improving price estimates. However, in cases where a given product family had few historical records, the accuracy of the models was negatively impacted.

ML models are also applied in ETO contexts to anticipate production delays and define predictive maintenance. Cao et al. (2024) used ML-based models to detect opportune moments to perform preventive maintenance on machinery, using data collected from sensors attached to each piece of equipment. Models based on Random Forest performed best, and the results showed that predictive models combined with blockchain technologies can anticipate failures and malfunctions. Burggraf et al. (2021) addressed operational delays in assembly tasks by developing data-driven models to predict missing components. The study compared various ML techniques, including tuned classification and regression models, across heterogeneous datasets. Despite the depth of the analysis and the number of alternatives tested, the models did not obtain satisfactory results for establishing action plans, which reinforces the complex nature of ETO production.

The ML-ETO literature reveals different contexts for the application of predictive models, with the primary challenge being the high variability or lack of data provided to train such models. These challenges are particularly evident in project-driven manufacturing environments—such as the ETO environment explored in this paper where the products are designed from scratch, tailored to specific client or project requirements, rather than merely customized. The frequent design changes inherent to this production environment directly impacts the data uniformity and consequently, the quality of the predictive models.

3. Problem description

This study extends previous work conducted at an ETO industrial site, as presented in Carvalho et al. (2016), which introduced a robust optimization model to support tactical production planning. In that earlier work, processing time was treated as an uncertain parameter, handled

through user-defined upper and lower bounds. In contrast, the present study aims to leverage historical production data to build a predictive model for processing time estimation. By uncovering patterns and relationships from past projects, this data-driven approach seeks to replace subjective ranges with more accurate, evidence-based forecasts—ultimately enhancing the reliability of both production planning and cost estimation.

The production environment examined is a medium-sized, project-driven company that manufactures high-value, customized equipment—such as high-pressure boilers and complex process reactors—following an ETO policy. Its production system consists of specialized work centers (cutting, stamping, machining, assembly, and welding), all requiring a skilled and costly workforce. The manufactured products are structurally complex, involve numerous components and production stages, and have long lead times, ranging from 5 to 18 months. The company primarily serves large oil & gas enterprises, which hold significant bargaining power. Failing to meet delivery deadlines may result in disqualification as a supplier, making reliable planning essential.

During the order acceptance stage, the planning team seeks to align demand with available capacity while contributing to the development of competitive bids. At this point, detailed product designs are not yet available, and product structures are defined only at aggregate levels, typically in terms of macrocomponents. Processing time estimates are derived from a reference model that resembles a normal distribution curve, segmented by key milestones along the project timeline. Each milestone represents a cumulative estimate of labor hours expected to be consumed at each work center up to that point, for each macrocomponent. The temporal distribution of effort tends to follow a consistent pattern across projects where the normal-like curve serves as a practical conceptual model—particularly useful for estimating how work progresses from early stages, such as cutting and stamping, to later stages like final assembly, where multiple pre-processed macrocomponents are integrated.

The reference model serves as a tool to support due date commitments and cost estimation during the early stages of planning. However, the method for estimating the processing times remains relatively coarse, relying heavily on averages or expert judgment. Therefore, the analysis proposed, within this research project, is based on a historical database which includes projects not only for boilers but also for other equipments, such as heat exchangers, process reactors and pressure vessels. Although the company manufactures one-of-a-kind products, efforts have been made to classify boilers into families according to their functional purpose and the type of energy source used in the steam generation process (e.g., chemical recovery boilers, oil and gas-fired boilers, solid fuel-fired boilers). Product structures vary across family groups and even within the same family. An important consideration is the significant variability in processing times, even among boilers belonging to the same family.

4. Methodology

To develop a model capable of predicting processing times, this study implemented a comprehensive data science pipeline, starting with data preprocessing, followed by exploratory analysis, and culminating in the modeling of prediction algorithms.

The raw data was supplied by the company from various sources, undergoing different groupings and modifications to make it more homogeneous and suitable for analysis. After an initial phase of organization and pre-processing, the data was consolidated into a single file with general information on historical operations. The resulting file contained approximately 2428 operation records, which were analyzed from the perspective of 19 attributes, all related to the nature of the tasks and the products. To validate the models, two different approaches were considered. The first one involves the entire dataset, while the second includes only products of the “boiler” type, which corresponds to 1752 records (approximately 72.4% of the original dataset).

The idea behind using this second approach is to take advantage of unique features of this type of product to train the model.

Among the preprocessing stages, data handling consisted of manipulating unstructured data to extract more information from the dataset. The data cleaning stage consisted of removing columns that contained predominantly missing values, low-frequency entries, or information considered irrelevant data for the purpose of this work. Table 1 reveals details of the features in the dataset, including those added and modified during the data handling stage, as well as those removed in the data cleaning stage for each approach. Each record in the dataset corresponds to a task, containing general information about the order and the corresponding product, the item manufactured, and the processing time.

Table 1: Attributes considered

Fn	F.Desc	Type	NClass	NAN (%)	NAN-B (%)	Stats
1	Source of data	C	6	-	-	-
2	Order code	C	141	-	-	-
3	Order description	S	-	-	-	R (1; 2)
4	Quantity	N	-	-	-	R (1; 2)
5	Product type	C	7	-	-	-
6	Product code	C	15	-	-	-
7	Product name	C	15	-	-	-
8	Processing time (target)	N	-	-	-	M (1; 2)
9	Work Center	C	5	-	-	-
10	Item code	C	211	-	-	-
11	Item description	S	-	0,2%	0,2%	R (1; 2)
12	Linear Power (B)	N	-	48,0%	28,7%	R (1; 2)
13	Z Power (B)	N	-	48,0%	28,7%	R (1); M (2)
14	Capacity (B)	N	-	78,7%	70,9%	R (1; 2)
15	Linear Capacity (B)	N	-	78,7%	70,9%	R (1; 2)
16	Z Capacity (B)	N	-	78,7%	70,9%	R (1; 2)
17	Evaporation (B)	N	-	69,3%	57,8%	R (1; 2)
18	Linear Evaporation (B)	N	-	69,3%	57,8%	R (1; 2)
19	Z Evaporation (B)	N	-	69,3%	57,8%	R (1; 2)
20	Activity type	C	2	-	-	A (1; 2)
21	Order class	C	28	-	-	A (1; 2)
22	Item class	C	20	0,5%	0,3%	A (1; 2)

Fn: Feature number; **F.Desc:** Feature Description (“B” represents exclusive features for Boiler type products); **Type:** Type of variable (C = Categorical; N = Numerical; S = String/Text); **NClass:** Number of classes (only for categorical features); **NAN (%):** Percentage of missing data; **NAN-B (%):** Percentage of missing data considering only boiler-type products; **Stats:** Represents whether the feature has remained original (-), has been removed (R), has been modified (M), or has been added in the preprocessing stage (A) (The first approach (1) used the entire dataset, while the second approach (2) only considered boiler-type products).

Feature 1 represents the origin of the data, considering the source from which the information was extracted. Feature 2-7 represents general information about the order, including product characteristics (5-7) and the quantity to be manufactured (4). Feature 8 is the target attribute and represents the processing time of a task. Feature 9 represents the workstation on which the task was carried out. Features 10-11 represent information about the item or components produced in a given task. The features in the range 12-19 represent technical characteristics of the “Boiler” product type, related to burning potential (12-13), capacity (14-16) and evaporation potential (17-19). As these features are highly correlated, it was decided to use only one of them (13) in the second approach, removing missing data. The features added during the pre-processing stage (20, 21, and 22) were derived from unstructured string-type features (3 and 11), in combination with information from categorical variables.

The exploratory analysis consisted of an in-depth investigation of the dataset using data visualization tools and statistical techniques based on goodness-of-fit tests to characterize the distribution of the data. Among the distributions considered were the Exponential, Gamma, Lognormal, Normal and Weibull. The fit of the distributions to the dataset was measured according to their performance in the Kolmogorov-Smirnov (KS) and Cramér-von-Mises (CMV) tests.

The stages relating to the implementation of the predictive model include various ML algorithms and their respective training methodologies. Considering the categorical nature of the analyzed dataset, it was noted that both the predictive model and the selection of suitable encoding mechanisms were crucial to the model's performance. To this end, different encoding techniques were tested in combination with each ML algorithm. Among the encoders used were the Label, Target and variations of the OneHot techniques, applied either purely, combined with Label (OneHot + Label) or using Principal Component Analysis (PCA) to reduce dimensionality and prevent excessive model complexity.

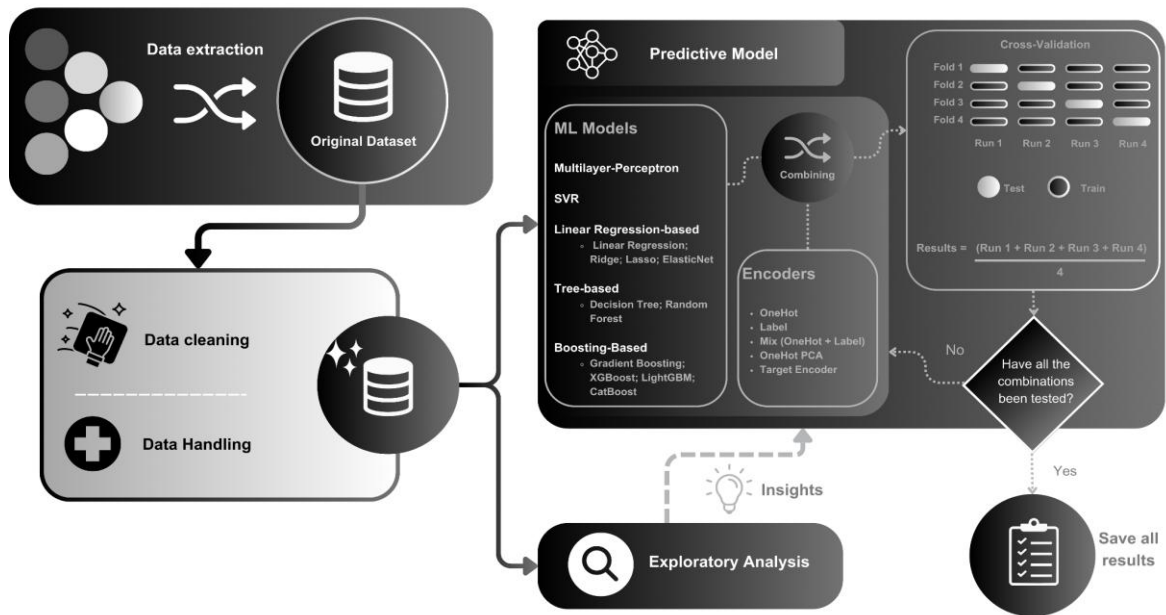
The ML algorithms employed include simple regression models (Linear Regression, Ridge, Lasso, Elastic Net), support vector regression (SVR), artificial neural networks (Multi-Layer Perceptron) and models based on decision trees, either simple (Decision Tree), or Ensemble models, based on Bagging (Random Forest) or Boosting (Gradient Boosting, XGBoost, LightGBM, Catboost). Linear regression-based models were selected to serve as baseline, providing a point of comparison for evaluating the performance of more complex algorithms.

Given the high asymmetry of the data (observed in the exploratory analysis) and the low number of records, stratified K-fold cross-validation was used to generate multiple sets of training and test data. This approach considered the specificities of each algorithm and encoding method to avoid possible overfitting. This cross-validation structure separated the dataset into 4 folds, where each model (Encoder + ML algorithm) was executed 4 times, so that at each run, a different fold corresponds to the test set. Each model (Encoder + ML Model) was evaluated considering the average performance obtained by each fold. It is important to note that, while most encoders were applied to the entire dataset (prior to the splitting into folds), some encoders (such as the Target Encoder) were applied internally within the cross-validation loop, since they carry information from the target attribute, and could bias the model by bringing privileged information from the target attribute into the training stage, leading to overfitting.

The result of each encoder-algorithm combination obtained at each iteration was stored in a list, and performance was evaluated from the perspective of different metrics, including the coefficient of determination (R^2), the Mean Absolute Error (MAE), the Root Mean Square Error (RMSE), and the Root Mean Square and Logarithmic Error (RMSLE). Figure 1 summarizes the workflow of the pipeline development process.

The pipeline was developed entirely in Python programming language. Different functionalities from the Pandas and NumPy libraries were also used. For the statistical analysis related to the distributions, including goodness-of-fit tests, the Scikit-Learn library was used. Matplotlib was used to support visual data analysis. The predictive models (ML and encoding tool) were developed by combining dedicated libraries, Scikit-Learn packages and custom-built techniques developed in-house by the authors.

Figure 1: Workflow of the pipeline development process



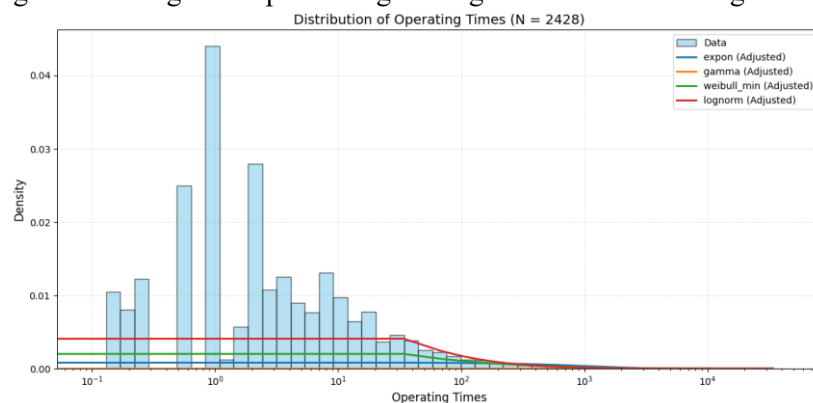
5. Results

This section presents the findings from the exploratory analysis, which was crucial to understand data structure and to adjust the dataset, and the results obtained by the predictive models.

5.1. RESULTS OF EXPLORATORY ANALYSIS

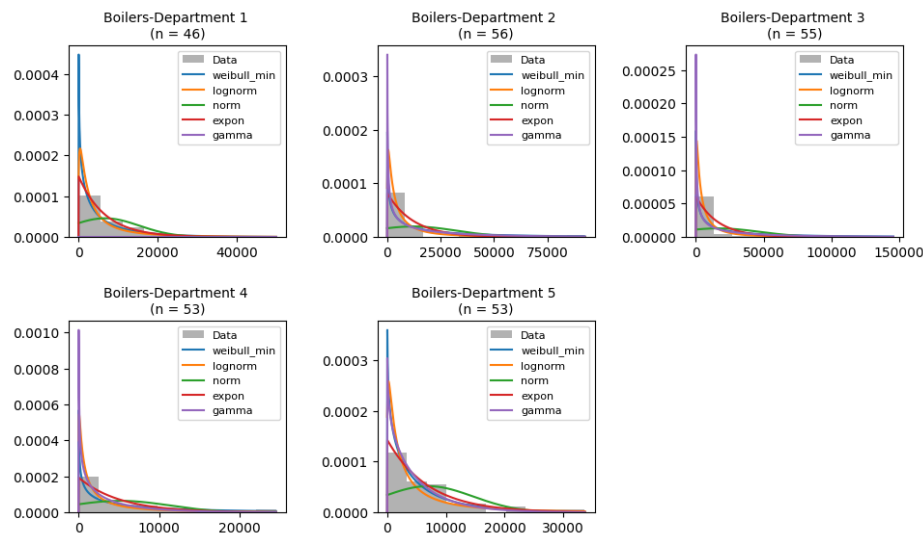
The exploratory analysis focused on identifying the behavior of the data, guided the choice of ML models and the entire pipeline developed in the subsequent stage. Using statistical techniques and tools for data visualization, the exploratory analysis revealed the asymmetrical nature of the data in relation to processing times, where the frequency of records was predominantly concentrated at values close to 0, although it could reach values as high as 40,000. Figure 2 shows the histogram of processing time records, along with the distribution functions fitted to this behavior (note that logarithmic scale was used to ease visualization). It can be seen, from the graphical display, that the lognormal distribution was able to better represent the variability considering low values of processing times, which corresponds to the main area of concentration of records.

Figure 2: Histogram of processing times generated from the original dataset



To allow for a more consistent and homogeneous analysis of the data, a new dataset was generated within this stage, considering only “Boiler” type products. Subsequently, the records in the new dataset were grouped by work center and order code, so that operations from a particular work center relating to the same order have their processing times added together. From this approach, it was possible to create histograms separated by work center, allowing visualization of the behavior of the distribution functions in each of them. At the same time, goodness-of-fit tests were carried out to check the accuracy of the fit of each distribution. Figure 3 shows the distributions of processing times grouped by order code, considering different work centers or departments.

Figure 3: Distribution of processing times grouped by order code in each work center



For each of the datasets generated (for each work center), 5 distributions were tested: Normal, Lognormal, Exponential, Gamma, Weibull. The Kolmogorov-Smirnov (KS) and Cramér-von Mises (CVM) goodness-of-fit tests were applied, and the distributions were evaluated according to the p-value. The higher the p-value, the better the distribution fit. Table 2 shows the performance of each of the distributions in each work center from the perspective of the aforementioned goodness-of-fit tests.

Table 2: Performance of the distributions considering different goodness of fit tests

	Work Center 1		Work Center 2		Work Center 3		Work Center 4		Work Center 5	
	Dist.	p-value	Dist.	p-value	Dist.	p-value	Dist.	p-value	Dist.	p-value
KS	Weibull	0,970260	Logn.	0,849288	Logn.	0,079465	Logn.	0,383739	Expo.	0,105336
	Logn.	0,695284	Gamma	0,002396	Gamma	2,84E-05	Gamma	0,211853	Normal	0,052003
	Expo.	0,223530	Expo.	0,002261	Normal	3,98E-08	Normal	0,023746	Weibull	0,047882
	Normal	0,017259	Normal	1,13E-05	Exp.	3,89E-08	Weibull	0,021934	Gamma	0,026278
	Gamma	6,57E-77	Weibull	8,01E-06	Weibull	2,69E-09	Expo.	0,005409	Logn.	0,025136
CVM	Weibull	0,943985	Logn.	0,725618	Logn.	0,311220	Logn.	0,275636	Expo.	0,187917
	Logn.	0,804775	Gamma	0,007391	Gamma	0,001415	Gamma	0,149896	Weibull	0,181366
	Expo.	0,279974	Expo.	0,001671	Normal	3,21E-06	Expo.	0,041057	Logn.	0,16471
	Normal	0,019789	Weibull	2,47E-05	Expo.	2,46E-06	Normal	0,023286	Gamma	0,080336
	Gamma	2,53E-09	Normal	2,02E-05	Weibull	9,44E-07	Weibull	0,004524	Normal	0,044077

Considering the p-values, the distribution that yielded the best fits for the majority of cases ($p > 0.05$) was Lognormal. That result confirms the asymmetrical nature of the data, proving that extreme processing times are not statistical outliers but rather intrinsic characteristics of the process behavior.

5.2. RESULTS OF PREDICTIVE MODEL

The main challenge faced in the predictive model was to predict a numerical attribute (processing time) using predominantly categorical features (numerical features had a significant amount of missing data) in an unbalanced dataset (verified in the exploratory analysis). Considering the combination of different encoders with ML models, Table 3 shows the best 5 models and the best regression-based model (outlined in blue) from the perspective of 4 different metrics, considering the complete dataset (Approach 1) and only Boiler-type products (Approach 2).

In general, the implemented approaches yielded similar results. Approach 2 was unable to identify consistent patterns within the data, despite employing a refined dataset ($n = 1,237$) containing a distinctive feature similar to Approach 1. In both approaches, although the Boosting-based models performed better in terms of R^2 and absolute error metrics (especially XGBoost and Catboost), almost all the models yielded relatively high error values. Concerning the RMSLE, Random Forest usually performed better, although it still presented a considerable error scale. Models based on linear regression (*i.e.* Linear Regressor, Ridge, Lasso, Elastic Net) yielded the worst performance results across all metrics.

Table 3: Top five models vs Best Regression-based model considering each metric

	R² (higher is better)			MAE (lower is better)			RMSE (lower is better)			RMSLE (lower is better)		
	Alg	Enc	value	Alg	Enc	value	Alg	Enc	value	Alg	Enc	value
Approach 1 n = 2428	XGB	target	0,594	XGB	target	797,086	XGB	target	2064,678	RF	onehot	1,535
	CAT	target	0,576	CAT	target	820,063	CAT	target	2114,630	DT	label	1,615
	CAT	label	0,559	RF	target	870,905	CAT	label	2155,143	DT	onehot	1,669
	CAT	mix	0,554	LGBM	target	886,481	CAT	mix	2168,667	RF	label	1,719
	RF	label	0,550	RF	label	911,228	RF	label	2178,950	DT	mix	1,731
	RDG	onehot	0,333	LAS	onehot	1222,89	RDG	onehot	2651,969	EN	onehot	2,705
Approach 2 n = 1237	RF	mix	0,642	CAT	target	794,892	RF	mix	1992,077	RF	onehot	1,621
	CAT	mix	0,638	CAT	onehot	806,159	CAT	mix	2000,609	DT	label	1,645
	CAT	onehot	0,628	XGB	onehot	826,030	CAT	onehot	2023,110	DT	onehot	1,684
	RF	label	0,627	CAT	mix	846,053	RF	label	2026,290	DT	mix	1,752
	CAT	label	0,626	RF	mix	846,167	CAT	label	2031,608	RF	target	1,787
	LAS	onehot	0,308	LAS	onehot	1347,46	LAS	onehot	2777,274	EN	onehot	2,837

Alg: ML Algorithm (CAT = Catboost; DT = Decision Tree; EN = Elastic Net; LAS = Lasso; LGBM = LightGBM; RDG = Ridge; RF = Random Forest; XGB = XGBoost)

Enc: Encoder (mix = Mix Encoder (OneHot + Label); onehot = OneHot Encoder; label = Label Encoder; target = Target Encoder)

6. Discussion

Several insights emerged from this study, spanning different dimensions including problem formulation, contextual characteristics, data structure, and the machine learning (ML) pipeline implementation. These insights are presented and discussed in this section and are

intended to inform both practitioners and researchers working with ML applications in Engineer-to-Order (ETO) environments.

The first insight concerns the problem formulation. ETO systems pose challenges in both due date prediction and cost estimation accuracy. It is therefore crucial to distinguish between lead time and processing time, as they may serve distinct purposes: lead time is essential for delivery scheduling, whereas processing time is a key factor in determining cost estimates.

The second insight relates to the specific characteristics of the problem context. The case study was based on a company that manufactures one-of-a-kind, highly complex products with deep and detailed product structures, often comprising thousands of components. Although the company classifies products into families, there are considerable variations even within the same family. These families are grouped based on certain high-level characteristics, which are not necessarily aligned with similarities in processing time or production workload. This makes clustering or grouping for ML purposes particularly challenging.

The analyzed dataset included 7 product categories, where the “boiler” product type corresponds to approximately 72.4% of the records. It is important to note that each product category is associated with a distinct set of attributes, making it difficult to develop a unified model that generalizes well across all types. This work used two approaches: In the first one, all records were used to train a single model, while in the second approach considered only operations related to boilers. However, an alternative approach would involve building separate models for the other product types (and not just boilers). Although this alternative approach could improve prediction accuracy, it would require a greater number of records and information about other product types. A cost-benefit analysis should be carried out to determine the feasibility of such an approach.

Another relevant observation concerns the high variability of the output variable—processing time—which, in some cases, ranged from 0,17 to 44.232 hours. This wide dispersion further complicates the task of learning accurate patterns from the data. The variability of the dataset was verified in the exploratory analysis, which confirmed the asymmetrical nature of the distribution of processing times, with a concentration of records at the lowest ranges. In both approaches, the lognormal distribution was the best fit, as it was able to better capture the differences in processing times considering the lower values (which correspond to the main data concentration zone). However, considering an analysis of the distributions by work center using goodness-of-fit tests, there are few records to measure the performance of each distribution for each department using only the p-value. For future work, new criteria could be used, such as the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC).

A total of 60 models were evaluated, generated by the combination of five data encoding strategies with twelve different ML algorithms. Among the models tested, ensemble models such as Random Forest, XGBoost and Catboost showed more consistent results for the two approaches, across different encoding techniques. This performance is justified by the robustness of these models when dealing with categorical variables (especially Catboost, which is designed for this purpose) and capturing complex and non-linear interactions in the data (typical of tree-based models). In turn, models based on linear regression yielded the worst results due to the high degree of asymmetry in the data. However, even the best-performing models showed unsatisfactory results in terms of error metrics.

The high value of the RMSE in relation to the MAE reveals the difficulties faced by the models when predicting considerably higher processing time values, which correspond to a small fraction of the dataset. The moderate R^2 values indicate that the models manage to explain a good part of the variance, however the high RMSE value may reveal that the R^2 is failing or ignoring the variance in extreme regions of the distribution (with high processing times for example). The best models in relation to RMSLE (the use of this metric was justified by the nature of the data distribution) were simple decision trees and Random Forest, however the high error may indicate problems related to the quality of the data provided, which may be insufficient to predict the target

attribute. Removing records with high processing time values can drastically reduce the error, but they remain significant, which can reinforce the idea that the features considered are insufficient to map the target attribute.

7. Conclusion

This study developed predictive models to estimate processing times using real historical data from an ETO company specialized in the production of unique and highly complex industrial equipment. A complete data science pipeline was implemented, encompassing data preprocessing, exploratory analysis, and the modeling of predictive algorithms. In total, 60 models were generated by combining five data encoding strategies with twelve different machine learning algorithms. The modeling process was supported by statistical analyses of data distribution, employing visualization tools and goodness-of-fit tests. Two distinct modeling approaches were explored: one using a dataset encompassing all product types, and another focused specifically on a single product category (boilers). Model performance was assessed using various metrics, including the coefficient of determination (R^2) and multiple error-based indicators.

The results yielded by the models were not satisfactory in terms of the error metrics. The models that used more sophisticated Ensemble-based techniques—such as Bagging-based models (Random Forest) or Boosting-based models (Catboost, XGBoost, LightGBM)—, yielded the best results. Despite this poor performance, robust tree-based techniques in general showed more consistent results compared to simpler linear regression-based models, which struggled to deal with the data's variability. This suggests the need for more informative features or additional domain-specific variables to improve model performance in future research. In an attempt to obtain better performance, parameter optimization techniques could be considered in future work, using simple parameter “tuning” techniques—such as GridSearch or RandomSearch—, or sophisticated parameter search mechanisms based on metaheuristics or reinforcement learning.

Beyond the predictive results themselves, the study also yielded several important insights that contribute to the broader understanding of ML applications in the ETO context. These include: (i) the distinction between lead time and processing time, which is critical when estimating production costs; (ii) the observation that product groupings based on high-level characteristics do not necessarily reflect similarities in processing time, making it challenging to apply ML models effectively in this context; (iii) the difficulty of developing a unified predictive model, highlighting the need for a larger and detailed dataset about different product types; and (iv) the high variability of the target variable which, although expected in an ETO environment, presents significant challenges for learning consistent patterns from the data.

This study was based on a real-world dataset composed primarily of categorical variables, aimed at predicting a numerical target using aggregated data. It is important to emphasize that the data used reflects an actual industrial context and a concrete operational challenge. Sharing the insights and lessons learned through this work is valuable for both practitioners and researchers interested in developing solutions for similar problems in complex production environments.

8. Data Availability

Data will be made available on request.

9. Acknowledgements

The authors are grateful to the National Institute of Technology (INT), to National Council for Scientific and Technological Development (CNPq) and to Rio de Janeiro State Foundation to Support Research (FAPERJ - E26/0.10.001920/2019).

References

- Burggraf, P., Wagner, J., Koke, B., and Steinberg, F. (2020). Approaches for the prediction of lead times in an engineer to order environment - a systematic review. *IEEE Access*, 8:142434–142445.
- Burggraf, P., Wagner, J., Heinbach, B., and Steinberg, F. (2021). Machine learning-based prediction of missing components for assembly—a case study at an engineer-to-order manufacturer. *IEEE Access*, 9:105926–105938.
- Cao, X., Jing, Z., Zhao, X., and Xu, X. (2024). A security-enhanced equipment predictive maintenance solution for the eto manufacturing. *International Journal of Network Management*, 34(6):e2263.
- Carvalho, A. N., Oliveira, F., and Scavarda, L. F. (2016). Tactical capacity planning in a real- world eto industry case: A robust optimization approach. *International Journal of Production Economics*, 180:158–171.
- Chehade, M. H., Sylla, A., Diallo, A. R., and Doremus, Y. (2024). Manufacturing time estimation for offer pricing: A machine learning application in a french metallurgy industry. *Engineering Applications of Artificial Intelligence*, 137:109089.
- Fortes, C. S., Tenera, A. B., and Cunha, P. F. (2023). Engineer-to-order challenges and issues: A systematic literature review of the manufacturing industry. *Procedia Computer Science*, 219: 1727–1734.
- Rapaccini, M., Cadonna, V. L., Leoni, L., and De Carlo, F. (2023). Application of machine learning techniques for cost estimation of engineer to order products. *International Journal of Production Research*, 61(20):6978–7000.
- Sherwin, M., Medal, H., MacKenzie, C., and Hradecny, M. (2025). A machine learning approach for engineer-to-order firms to predict supplier performance in critical supply chains. *International Journal of Management Science and Engineering Management*, p. 1–19.
- Sylla, A., Coudert, T., and Geneste, L. (2021). A case-based reasoning (cbr) approach for engineer-to-order systems performance evaluation. *IFAC-PapersOnLine*, 54(1):717–722.
- Ziegler, C. C., Dobhan, A., and Heusinger, M. (2022). Applications of neural networks in engineer-to-order environment. *Procedia CIRP*, 112:140–145.