Defect Prediction on Production Line

Souhaiel KHALFAOUI, Eric MANOUVRIER, Alexandre BRIOT, Stephane BUTEL,
Aitor GONZALEZ, Miroslav ZIMA, Romain DELENTE, Aurele PILLOUD-PASSIN,
Benoit VERGER, Fabrice BLASENHAUER Jean-Jacques LOPEZ, Remi LARONDE,
Sebastien ODOUARD, Stephane DE-CLERCQ, Stephane WYSOCKI,
Jesutofunmi IBRAHIM Tatenda KANYERE Bola ORIMOGUNJE
Amr ABDULLATIF Daniel NEAGU David DELAUX

June 2021

Abstract

Quality control has long been one of the most challenging fields of manufacturing. The development of advanced sensors and the ease of collecting a high amount of data designate the machine learning techniques as a timely natural step forward to leverage quality decision support and manufacturing challenges. This paper introduces an original dataset provided by VALEO, coming from a production line, and hosted by the ENS Data Challenge to predict defects using non anonymized features, but without having access to the final test results to validate the part status (defective or not). We propose in this paper a complete workflow from the data exploration to the modelling phase while addressing at each stage challenges and techniques to solve them.

Keywords: *manufacturing; quality control; machine learning; supervised learning.*

1 Introduction

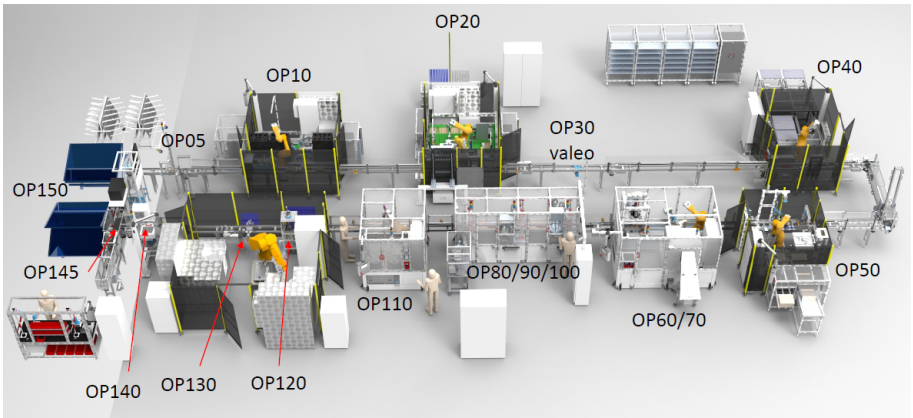


Figure 1: Production Line

The Fourth Industrial Revolution also referred to as Industry 4.0, is defined by the continuous automation of conventional manufacturing practices through modern innovative technologies. For instance, the application of Machine-to-Machine communication (M2M) and the integration of the Internet of Things (IoT) in the manufacturing processes offer new opportunities and become crucial to optimize the production procedure. This objective can be met by implementing efficient self-monitoring techniques within the vision to produce autonomous machines that can diagnose, adapt to and face issues with minimal human intervention. They generate a considerable amount of information regarding a particular production sub-process. The collection of the overall produced data generated from different sensors provides the manufacturer with numerous opportunities to

improve the product quality sustainably through data-driven approaches [1].

Current advances in applied statistics [2] and computer science [3], and the availability of a wide range of data analysis tools, offer a great potential the digital transformation of the manufacturing domain. The Machine Learning (ML) field, including Data Mining (DM) [4], Big Data (BD) [5], Knowledge Discovery (KD) from databases [6], is considered one of the most promising breakthroughs when it comes to data analysis, pattern identification and model extraction tasks. However, these ML approaches are very diverse depending on the objectives of the prediction task and the nature of the available data, offering challenges along with the named opportunities.

Looking at the manufacturing sectors that are quite likely to be optimized nowadays, such as monitoring and control, the increasing amount of available data represents a major step through and a challenge that needs to be efficiently handled. Furthermore, the high dimensionality and the heterogeneity of the data resources harden the task of finding complex and (non-linear) patterns in the data that comes from different types and sources of deployed devices. With such challenges in mind, numerous manufacturers start reaching out to the academic community and the data science community by providing access to necessary data resources in order to opt for advanced and more adequate learning techniques that would boost manufacturing performances. For instance, in 2016 the German engineering and technology company Bosch has offered a large-scale dataset of a production line and organized a challenge on the Kaggle platform aiming at predicting the manufacturing failures through anonymized features [7][8]. In this context, the manufacturing engineering company VALEO hosted the Data Challenge ENS [9] to predict the failed products in a production line.

This paper produces a review of the VALEO Data Challenge data, modelling approaches and results, and consequently offers an insight and vision of how engineering, ML and data science communities join forces within the new Industry 4.0 era. The rest of the paper is organized as follows: Section 2 details the production line context of information gathering. Section 3 explores the VALEO dataset. Section 4 details the dataset pre-processing phase followed by the modeling phase results and discussion in Section 5. Conclusions and future work are discussed in Section 6.

2 Dataset Description

2.1 Production Workflow

The production line subject of this study is an assembly line for electric starter motors (see figure 1). A starter is made of several components to assemble all together, to screw and to insert pieces based on a-priori designed processes. During the assembling stage, different values (torques, angles, etc.) are measured on different mounting stations. At the end of the line, additional measures are performed on two test benches in order to identify and isolate defects. As a result, samples are tagged either 'OK' or 'KO'. The unitary production time for a starter on one full automatic line is $12s$. Before starting the assembly process, each starter is given a unique ID, PROC_TRACEINFO, to ensure traceability. Table 3, details the the different steps of the starter assembly.

2.2 Dataset Content

The VALEO Data Chalenge dataset contains data retrieved from stations OP70, OP90, OP110, OP120 and OP130 (see figure 1). The neglected working stations at this stage are those for subparts preparation. The process variables have 14 independent variable and 1 target variable (binary classed). The variables have 14 numerical (floats and integers) and 1 categorical value. Table 2 details all the dataset features.
The unique ID code, PROC_TRACEINFO, e.g I-B-XA1207672-190701-00494 is read as follows:

- XA1207672 is the reference.
- 190701 is the assembly date (July 1^{st}, 2019).
- 00494 is the unique code given to the product, whatever it happens, the product will have this id number frozen forever.

Working Station	Description
OP 05	Re-introduction if needed (reworked parts)
OP10	Bracket loading
OP20	CED loading
OP 30	Armature placement
OP 40	Yoke placement
OP 60	M8 placement
OP 70	Screwing
OP 80	Ring placement
OP 90	Ring insertion
OP 100	Cap assembly
OP 110	Screw M8
OP 120	Lapping
OP 130	Performance test
OP 140	Printer
OP 150	Noise subjective hearing and final downloading

Table 1: Production Line Workstations.

This latter number is increased by 1 each time we process a new product, every 12s. Thus, the next product will have I-B-XA1207672-190701-00495 as a unique code.

The product status is the result value of OP130 (test bench). Value 1 is assigned to OK samples (passed) and value 2 is assigned to KO samples (failed). This is the combined result of multiple electrical, acoustic and vibro-acoustic tests.

3 Dataset Exploration

We propose the following workflow adapted to the VALEO Data Challenge that allows flexibility of task re-definition.

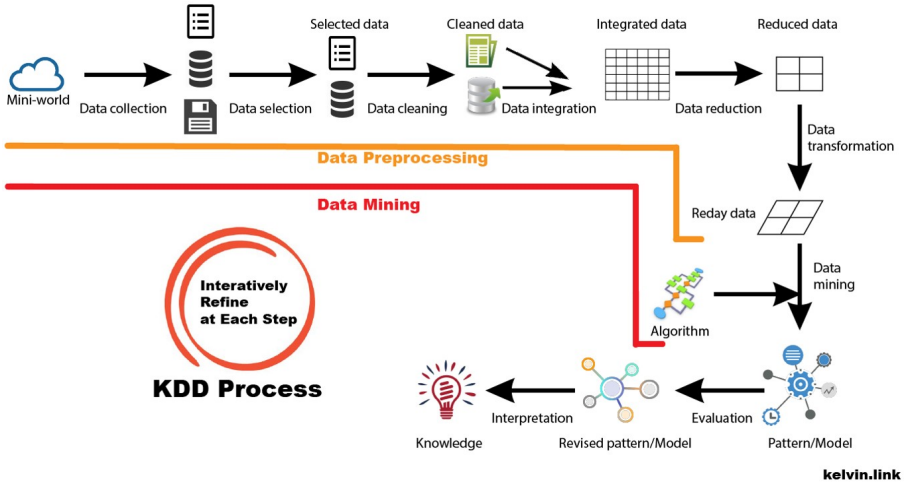


Figure 2: VALEO Data Challenge Processing Workflow.

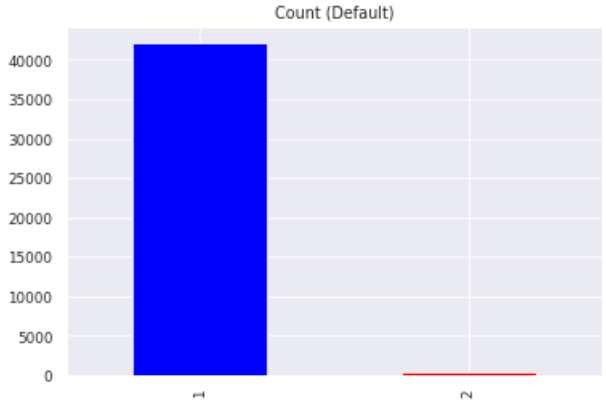
One of the main challenges of this dataset is its high imbalance since the amount of OK and KO products are highly imbalanced because only a small proportion of the products are failures. Train and Test datasets are therefore highly imbalanced. Thus, a subsequent challenge for the modeling techniques is the need to be able to handle this highly imbalanced data characteristic with impact

Variable	Source	Description
PROC_TRACEINFO	MES	Unique identification number given. Product reference including production date
OP070_V_1_angle_value	OP070	Measured angle of the first screwer of the station.
OP070_V_2_angle_value	OP070	Measured angle of the second screwer of the station.
OP070_V_1_torque_value	OP070	Real value measured of the setting (we set the torque, and it gives the angle). Screwer 1 applied torque on the product.
OP070_V_2_torque_value	OP070	Real value measured of the setting (we set the torque, and it gives the angle). Screwer 2 applied torque on the product.
OP090_StartLinePeakForce_value	OP090	Measured force (KISTLER sensor). Start force of one curve applied to insert one ring in the product (point 1).
OP090_SnapRingMidPointForce_value	OP090	Measured force (KISTLER sensor). Mid force of one curve applied to insert one ring in the product (Point 2).
OP090_SnapRingPeakForce_value	OP090	Measured force (KISTLER sensor). Peak force of one curve applied to insert one ring in the product (Point 3).
OP090_SnapRingFinalStroke_value	OP090	Measured force (KISTLER sensor). Final force of one curve applied to insert one ring in the product (Point 4).
OP100_Capuchon_insertion_mesure	OP100	Measured displacement (HEIDENHAIN LINEAR RULE). Displacement measure of one cap inserted in the product.
OP110_Vissage_M8_torque_value	OP110	Real value measured of the setting (we set the torque, and it gives the angle) . Torque applied on the long screw M8.
OP110_Vissage_M8_angle_value	OP110	Angle measured on the product of the screw M8.
OP120_Rodage_I_mesure_value	OP120	Measured intensity consequent of the tension applied as setting.
OP120_Rodage_U_mesure_value	OP120	Measure of the setting value. We set the tension and we read the intensity to facilitate the lapping.
OP130_Resultat_Global_v	OP0130	Result of the testing benches. Good or Bad according final End Of Line (EOL) testing of power and acoustic characteristics.

Table 2: Dataset description.

to the entire workflow. Additional goals of this stage are to analyse the different process variables and: 1) characterise the data quality; 2) identify factors that lead to detecting the non-ok parts; 3) Understand the interaction (correlation) between the features, e.g. how to map defects with different process variables.

The dataset contains complementary features that are coming from the same workstations. This may introduce noise in the collected data. Exploring outliers is a crucial step to minimize



a.

Training		Test	
OK	KO	OK	KO
34210	305	7935	66

b.

Table 3: Data imbalance: a. The global dataset contains 42145 good parts and 371 defects b, which falls under less than one percent of the entire dataset. It is analysed as a supervised binary classification task as it is a prediction challenge with 2 outputs/classes. The task is to predict the defect on starter motor production line as either ok or non-ok parts.

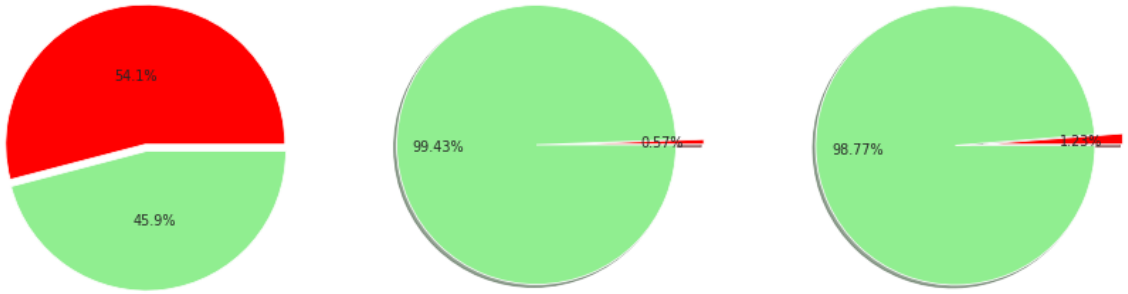


Figure 3: Missing data: OP100_Capuchon_insertion_mesure has 22987 missing values (54%). The ratio of defects corresponding to the missing values (in the middle) is equivalent to the one corresponding to the existing OP100 values.

noisy data effect. As shown in figure 3, large portion of the feature extracted from OP100 workstation has missing values (54%). A positive statement is that the missing and existing values for this feature have the same distribution with respect to the targeted outputs. It can be interpreted as low correlation between the considered feature and the output.

The analysis of the daily production does not show a significant increase of the defected parts production except for periods when the production rate was high. The production seems to be quite cyclic. This is a valuable information for targeting new informative features.
As it was expected, the frequency distribution analysis confirms the presence of outliers in features coming from the same working stations such as OP070, OP090, OP110 and OP120. The feature extraction phase become a challenging step for the success of our binary classification problem.

4 Feature Engineering

4.1 Tagging Features by Binomial Distribution

As shown in table 4, 5 of the above features have 2 frequency peaks:

- OP070_V_1_torque_value
- OP070_V_2_torque_value
- OP090_StartLinePeakForce_value

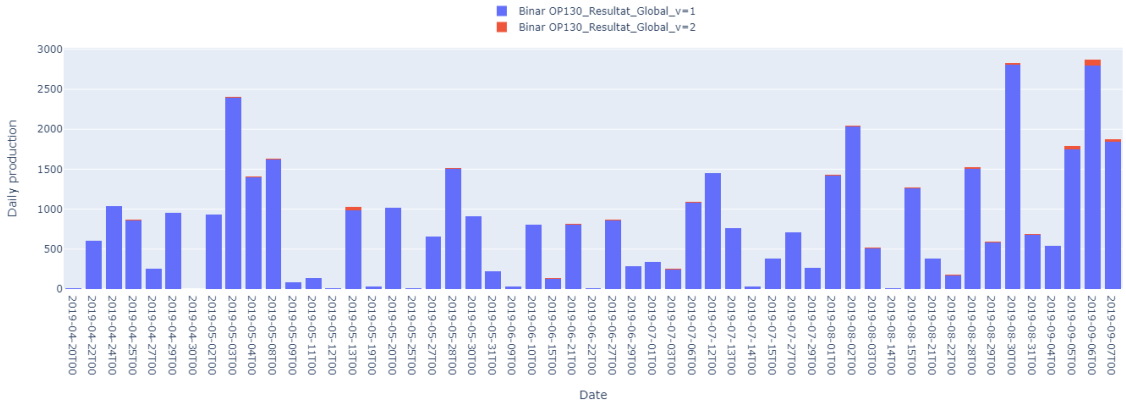


Figure 4: Daily production: good and bad parts.

- OP090_SnapRingMidPointForce_val
- OP090_SnapRingFinalStroke_value

In order to emphasize this particularity, OP070 and OP090 have been tagged with binomial distribution to enrich the list of features.

4.2 Time-based Features

PROC_TRACEINFO feature contains valuable details about the production time and the produced reference count. The time information has been extracted to create new features such as *day, month, day of the week and weekend*. Time is also a rich source to identify cyclical patterns in data. Therefore, additional features have been created to encode cyclical day, week and month using two features each using *sin* and *cos* transformations.

4.3 Feature Aggregation

Aggregating time related features and production references allows us to extract derivative information about *daily, weekly and monthly production* and also the corresponding *batches*. The original dataset features have been also aggregated with time based features to complete the time sequenced production to better guide the modeling technique when detecting gaps in data.

4.4 Binning

Binning is the procedure of splitting the interval with all observed values into smaller sub-intervals, called bins or groups, and assigning the central value characterizing this interval. All observations that lay on an articular sub-interval form an associated bin. Binning is also a form of discretization as it tunes continuous variables into categorical values. Binning is a widely used technique for decreasing overall complexity and reducing the impact of statistical noise.
There are several unsupervised and supervised binning techniques. Commonly used unsupervised techniques are equal-width and equal-size or equal-frequency interval binning. Supervised binning techniques can be divided into two main categories: merging and decision tree-based approaches. Commonly used merging based binning techniques are Monotone Adjacent Pooling Algorithm (MAPA), also known as Maximum Likelihood Monotone Coarse Classifier (MLMCC) [10] and ChiMerge [11]. On the other hand, common decision tree-based techniques are CART [12], Minimum Description Length Principle (MDLP) [13] and more recently, condition inference trees (CTREE) [14]. To take advantage of this technique, the binning process must follow some specific guidelines regarding (1) the missing values, (2) the minimum number of observations per bin and (3) the accounts of good or bad.

For our study, we made the choice to combine several binning techniques to cover all the particularities of our dataset. As illustrated in table 4, our dataset is composed of variables with various distributions making the choice of the binning technique more efficient for some of them and less

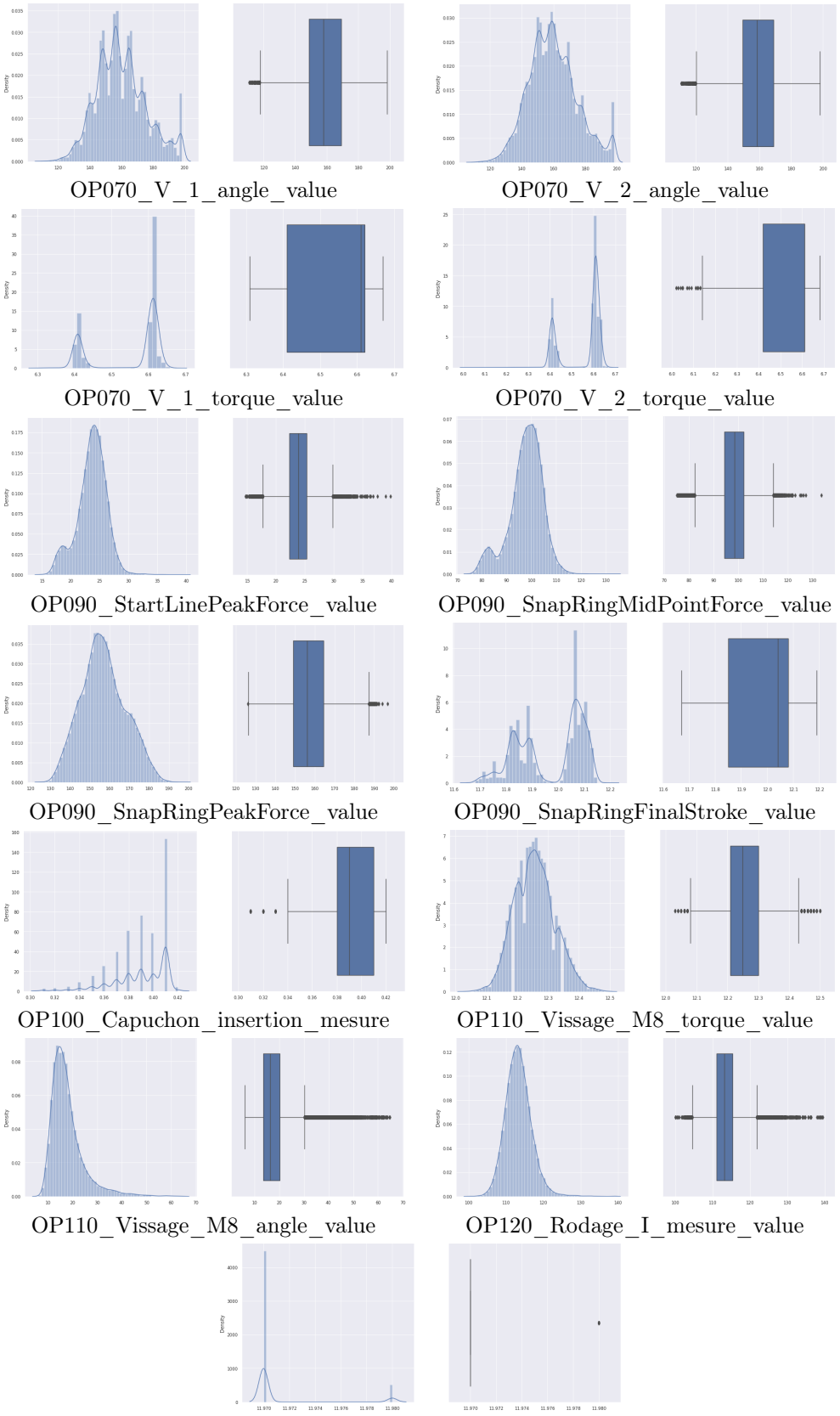


Table 4: Dataset exploration: frequency and distribution analysis.

for others. Because of the large gaps for the range of some numerical feature, using fixed-width binning may not be effective since there are empty or low-density bins. This problem can be solved by positioning the bins based on. This can be done using the quantiles of the distribution. Quantiles are values that divide the data into equal portions. 13 additional quantiles binning based features have been created.

Monotone optimal binning [15] comes naturally as one of the the most efficient techniques. Indeed, it satisfies monotonicity constraints, particularly, due to the use of *Weight-of-Evidence* (WoE) for separating good accounts from bad accounts and the *Information Value* (IV) which expresses the amount of information of the predictor in separating Goods from Bads in the target variable.

4.5 Features Interaction

Datasets generally contain features that appear to be irrelevant with the class individually, but when combined with other features, they may highly correlate to the class. Individually, the features do not carry any details about the class, however, the combination of the two features completely control the class. Most commonly feature interaction techniques are boosting based such as Gradient Tree Boosting (GTB) [16], LambdaMART [17] and XGBoost which is the most used in challenges [18]. The XGBoost system relies on the use of ensembles of decision trees to tackle this issue. This choice is motivated by the natural behaviour of this particular data structure that includes both interacting variables when they are in the same tree. However, the variables, whose effects do not interact, are located in different trees.

4.6 Feature Selection

As detailed in previous sections, using binning techniques, features interaction analysis and time-based extraction the total number of generated features is 287. Instead of using all the created features, one may select the most relevant ones for our application. Reducing the number of features allow to both reduce the computational cost of modelling and, in some cases, to improve the performance of the model.

Feature selection methods are divided into three categories: supervised, semi-supervised and unsupervised based on the availability of the target labels [19]. Statistical-based feature selection methods involve evaluating the relationship between each input variable and the target variable using statistics and selecting those input variables that have the strongest relationship with the target variable. These methods can be fast and effective, although the choice of statistical measures depends on the data type of both the input and output variables.

For our problem of binary classification, we adopt the IV and WoE as the appropriate statistical measures to perform feature selection. Highly correlated features have also been filtered (with a threshold of 0.95). The final number of features considered is 145.

5 Modeling Results and Discussion

5.1 Performance Metric

The Area Under the Receiver Operating Characteristic Curve (AUROC) [20] is for the robustness of the algorithm. The classification results are provided as the probability of whether or not the data belongs to one class. The ROC curve is created by plotting the true positive (TP) rate against the false positive (FP) rate at various threshold settings. The AUROC usually ranges from 0.5 to 1 (perfect classification.).

5.2 Models Comparison

Since the missing values are caused by different manufacturing processes and are not missing at random, they are replaced by 0 or a dummy value "-1". Experiments showed that both choices provide equivalent results. For this paper, we made the choice to use zero fillings of missing values.

Used technique	Test AUROC
Logistic Regression	0.63
Gaussian Naive Bayes	0.63
Decision Tree Classifier	0.56
Linear Discriminant Analysis	0.57
Random Forest Classifier	0.64
DNN based approach	0.74

Table 5: Evaluation of the selected models.

The combination of multiple binning techniques and mainly the use of optimal binning (with WoE and IV) have mitigated the impact of OP100 missing values.

As illustrated in table 5, we compared several widely used supervised techniques. The AUROC scores obtained with most of the techniques indicates the complexity of such a failure detection problem. The use of ensemble methods like Random forest improves slightly the result compared to the Decision Tree model which can be explained by its ability to overcome overfitting problem when voting several Decision Trees.

Deep Neural Network (DNN) based approach provided the best score. This can be explained by the diversity within the enriched data easily guiding the Neural Network to extract hidden structures completing the already pre-computed knowledge within the Feature Engineering stage.

6 Conclusions and Future Work

The emergence of new technologies in the industry offers new possibilities to handle classical challenges such as quality control and failure prediction while combining the power of machine learning techniques and the rich content of the collected data in manufacturing. In this article, we detailed our approach for failure detection on the assembly line using the VALEO data challenge. The data exploration phase allows us to better understand the content and the challenges of the dataset. It was a crucial step to better target the feature engineering phase objectives. Despite the low scores, except for the DNN based approach, the modelling phase showed the importance of exploring new techniques that may isolate the easiest sample from hard samples in order to improve the prediction scores and also the model confidence.

In future work, the DNN based approach will be used for missing data handling as well as modeling. Focal loss will be considered for confidence increasing.

Acknowledgment

The authors thank VALEO for providing access to the Data Challenge resources, SAFI and the University of Bradford consortium for the opportunity to collaborate and integrate studies of the industry and academic experts. VALEO// SAFI// BRADFORD University

References

[1] Thorsten Wuest, Daniel Weimer, Christopher Irgens, and Klaus-Dieter Thoben. Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research*, 4(1):23–45, 2016.

[2] Ivanna Baturynska. Statistical analysis of dimensional accuracy in additive manufacturing considering stl model properties. *The International Journal of Advanced Manufacturing Technology*, 97(5):2835–2849, 2018.

[3] Duc T Pham and Ashraf A Afify. Machine-learning techniques and their applications in manufacturing. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 219(5):395–412, 2005.

[4] Andrew Kusiak. Data mining: manufacturing and service applications. *International Journal of Production Research*, 44(18-19):4175–4191, 2006.

[5] Eric Auschitzky, Markus Hammer, and Agesan Rajagopaul. How big data can improve manufacturing. *McKinsey & Company*, 822, 2014.

[6] Niclas Feldkamp, Soeren Bergmann, and Steffen Strassburger. Knowledge discovery in manufacturing simulations. In *Proceedings of the 3rd ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*, pages 3–12, 2015.

[7] Darui Zhang, Bin Xu, and Jasmine Wood. Predict failures in production lines: A two-stage approach with clustering and supervised learning. In *2016 IEEE international conference on big data (big data)*, pages 2070–2074. IEEE, 2016.

[8] Bosch. Production line performance, 2016.

[9] ENS and College de France. Challenge data, 2019-2020.

[10] Lyn C. Thomas, Jonathan Crook, and David Edelman. *Credit Scoring and Its Applications*. Society for Industrial and Applied Mathematics, USA, 2002.

[11] Randy Kerber. Chimerge: Discretization of numeric attributes. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, AAAI'92, page 123–128. AAAI Press, 1992.

[12] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[13] Usama M. Fayyad and Keki B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *IJCAI*, pages 1022–1029, 1993.

[14] Torsten Hothorn, Kurt Hornik, and Achim Zeileis. Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical Statistics*, 15(3):651–674, 2006.

[15] Guillermo Navas-Palencia. Optimal binning: mathematical programming formulation. *CoRR*, abs/2001.08025, 2020.

[16] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[17] Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81, 2010.

[18] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[19] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, and Huan Liu. Feature selection. *ACM Computing Surveys*, 50(6):1–45, Jan 2018.

[20] Christopher D Brown and Herbert T Davis. Receiver operating characteristics curves and related decision measures: A tutorial. *Chemometrics and Intelligent Laboratory Systems*, 80(1):24–38, 2006.