Feature Selection to address High-Dimensionality in Industry 4.0 Multi-emitter Laser Modules Assembly Lines

Nikolaos G. Markatos*, *Civil and Environmental Engineering Department, Water Engineering, Imperial College London, London, SW7 2BX, UK.*

Alireza Mousavi, *College of Engineering, Design and Physical Sciences, Brunel University, Uxbridge, Greater London, UB8 3PH, UK.*

Evina Katsou, *Civil and Environmental Engineering Department, Water Engineering, Imperial College London, London, SW7 2BX, UK.*

Giulia Pippione, *Convergent Photonics Italia S.r.l, Via Schiaparelli 12, 10148 Torino, Italy.*

Roberto Paoletti, *Convergent Photonics Italia S.r.l, Via Schiaparelli 12, 10148 Torino, Italy.*

***Corresponding author:** *n.markatos@imperial.ac.uk*

***Abstract*— Industry 4.0 has increased data depth and breadth in high-tech manufacturing, but high-dimensionality and sparsity persist. High-dimensional space's sparsity makes classical learning and knowledge extraction algorithms ineffective and error-prone. Dimension reduction methods like feature selection seem to address this problem. This study addresses these challenges by conducting a comparative analysis on a real laser assembly industrial case of high dimensions. We explore five standalone methods—NCFS, RReliefF, MRMR, RFE, and Lasso—applied to datasets from two laser modules (d-serie and s-serie). Additionally, two hybrid methods—RReliefF-RFE and MRMR-RFE—are evaluated, broadening the scope of feature selection strategies. Time efficiency prioritizes RReliefF, NCFS and Lasso, while RReliefF-RFE, NCFS and Lasso excel in interpretability, achieving significant predictor reduction without compromising accuracy. The study thus provides insights into the selection of FS methods in a challenging industrial laser assembly setting.**

Automation in the industry has led to the accumulation of high-dimensional datasets, demanding increased computational power and storage capacity. Handling high-dimensional data introduces a challenge known as the "curse of dimensionality". This problem refers to the sparsity of the data in high-dimensional space and the negative effect it has on algorithms that are designed for low-dimensional space. Dimension reduction techniques like feature selection (FS) prove effective in mitigating these challenge [1].

In multi-step laser assemblies, the datasets that are available are usually high-dimensional. This makes it difficult for predictive algorithms to be applied in real-time, as the irrelevant or inconsistent data may pollute the state space and reduce the accuracy of the predictive models. Thus, efficient feature selection is essential for rationalizing and controlling knowledge extraction from the assembly process.

This paper focuses on studying various feature selection methods, to decide which of those perform better in multi-step laser assembly processes. Two filter methods (Minimum Redundancy – Maximum Relevance (MRMR) and RReliefF), one wrapper method (Recursive Feature Elimination (RFE)) and two embedded methods (Neighbourhood Component Feature Selection (NCFS) and Least Absolute Shrinkage and Selection Operator (Lasso)) are tested for feature selection. Furthermore, combinations of filters and wrappers (MRMR-RFE and RReliefF-RFE) are tested. Each FS method creates a subset of predictors. These subsets are then fed to two regression methods (Artificial Neural Networks (ANNs) and XGBoost) with the aim to predict the output power of the module. Based on the performance of the regression methods, we can decide which FS methods created optimal subsets of predictors, thus performed better.

These regression methods were chosen as they are considered among the most suitable algorithms in the field of machine learning, lending themselves to the problem statement.

The FS methods are applied on two laser modules' assembly datasets (d-serie and s-serie) from a laser manufacturing. The datasets (datasets of two different laser models) contain predictors from all the stages of the assembly processes of the laser modules. In both datasets, the "output" variable that needs to be predicted is the final power of the laser module.

The results reveal that, RReliefF, RFE, NCFS, RReliefF-RFE and Lasso are the five methods that create subsets of predictors that perform well in both datasets and their performance is not dependent on the choice of the regression model that will be used for predicting the output power of the module. The researcher and practitioner can choose the most appropriate method based on the industrial setting. If the computational time vis-à-vis the product life cycle is important, then RReliefF, NCFS and Lasso are the best options. If interpretability (as discussed in the results section) is the main focus, then RReliefF-RFE, NCFS and Lasso are the best options.

RELEVANT WORK

Feature selection methods are being used in many areas as dimension reduction techniques. Although the work around FS is vast, we present in this section some representative works where FS has been successfully applied to enhance interpretability, improve the accuracy of later-applied prediction models and, improve the operation speed. In [2], Recursive Feature Elimination (RFE) with a Random Forest (RF) classifier is used to analyze machining audio collected during down-milling operations using a single microphone. The incorporation of RFE facilitates straightforward feature elimination, producing a concise and easily interpretable set of analyzed dimensions (interpretability). In [3], the authors introduce a model-based approach for predicting the Air Quality Index (AQI). To optimize the model's efficacy, the RF–mRMR technique is employed for the selection of influential variables influencing AQI (accuracy improvement).

In [4], the study assessed six feature selection methods to estimate wind speed quantiles in Canada. LASSO and MRMR emerged as the most efficient algorithms, requiring fewer parameters with good generalization performance, and identified specific predictors as more important for distinct exceedance probabilities (accuracy improvement).

In [5], the authors introduce a hybrid fault diagnosis method aimed at addressing challenges regarding the large amount of data generated during the operation of wind turbine and, inaccurate and untimely fault diagnosis for wind turbines. The proposed approach leverages ReliefF to efficiently extract fault-sensitive features accuracy improvement/ operation speed).

In [6], NCFS was utilized to reduce the dimensionality of the feature space in microscopic images by selecting discriminative feature sets from individual as well as combinations of features (accuracy improvement).

In [7], the author sought to establish a novel machine learning-assisted hybrid-input model for forecasting automobile demand, addressing research gaps related to input data, methodology, and the scope of demand forecast. The identification and verification of significant input features were conducted using the RReliefF algorithm (interpretability).

As it can be concluded, feature selection methods play a vital role in various domains where dimension reduction is needed.

FEATURE SELECTION METHODS

There are three categories of feature selection methods: filter, wrapper, and embedded methods.

Filter methods employ diverse ranking techniques, assigning scores to variables based on specific criteria, and subsequently applying a threshold to exclude less relevant variables [8]. This approach offers notable advantages: it demands minimal computational resources, has a negligible impact on predictive error rates [9], and helps prevent overfitting [8]. However, a key limitation lies in the potential exclusion of important variables that, may not individually be informative, but contribute to interrelationships and enhancement of general knowledge (state space) of the system.

Wrapper methods necessitate a classifier (or regressor), helping to specify feature subsets. While these methods often surpass filter methods, their drawbacks include heightened computational demands, as the classifier (or regressor) undergoes training for each feature subset. Applying wrapper methods in high-dimensionality problems can be challenging and time-consuming [10].

Embedded methods streamline the quest for an optimal feature subset as part of the learning process [11]. Distinct from wrapper methods, these techniques seamlessly integrate feature selection and modeling concurrently. This unique approach translates to lower computational requirements compared to wrapper methods, presenting a more efficient alternative [12].

RReliefF

RReliefF [13] is a non-parametric method that selects the relevant features in regression based on weight vectors. RReliefF firstly iterates over each training instance. For each instance, it randomly selects another instance as a reference point and identifies its k nearest neighbors based on a specified distance metric. During the iteration over nearest neighbors, the algorithm updates the weights based on differences between instances. After accumulating the weights, RReliefF computes the final estimation for each attribute. The algorithmic form of the method can be seen below (as presented in [13]).

Input: for each training instance a vector of attribute values **x** and the predicted value $\tau(\mathbf{x})$

Output: the vector W of estimations of the qualities of attributes

```
1.  set all $N_{dC}$ ,$N_{dA}[A], N_{dC\&dA}[A], W[A]$ to 0 ;
where $N_{dC}$: weights for different prediction (class), $N_{dA}[A]$: weights for different attribute, $N_{dC\&dA}[A]$: weights for different prediction & different attribute and $W[A]$: final estimation of each attribute
2.  for i: = 1 to m do begin
3.      randomly select instance $R_i$;
4.      select k instances $I_j$ nearest to $R_i$;
5.      for j: = 1 to k do begin
6.          $N_{dC} := N_{dC} + |f(R_i) - f(I_j)| \cdot d(i,j)$;
7.          for A: = 1 to #all_attributes do begin
8.              $N_{dA}[A] := N_{dA}[A] + \mathrm{diff}(A, R_i, I_j) \cdot d(i,j)$;
9.              $N_{dC\&dA}[A] := N_{dC\&dA}[A] + |f(R_i) - f(I_j)| \cdot \mathrm{diff}(A, R_i, I_j) \cdot d(i,j)$;
10.         end;
11.     end;
12. end;
13. for A: = 1 to #all_attributes do
$W[A] := \frac{N_{dC\&dA}[A]}{N_{dC}} - \frac{(N_{dA}[A] - N_{dC\&dA}[A])}{(m - N_{dC})}$;
```

Minimum Redundancy – Maximum Relevance (MRMR)

MRMR was introduced in [14] and is an algorithm that finds the optimal set of features that is mutually and maximally dissimilar by minimizing the redundancy and maximizing the relevance of a feature set to the response variable. MRMR iteratively selects features based on their relevance and redundancy. It starts by selecting the feature with the highest relevance and adds it to the set of selected features. Then, it iteratively adds features that have high relevance and low redundancy until redundancy becomes nonzero for all features. Finally, it incorporates features with zero relevance into the selected set. This process aims to create a feature set with features being ordered by feature importance. For more information on the method please refer to [14]. The algorithmic form of the method can be seen below (as presented in [15]).

1. Select the feature with the largest relevance, $\max_{x\in\Omega} V_x$, where $V_x = I(x,y)$ is the mutual information. Add the selected feature to an empty set S.
2. Find the features with nonzero relevance and zero redundancy in the complement of S, S^c.
   a. If S^c does not include a feature with nonzero relevance and zero redundancy, go to step 4 .
   b. Otherwise, select the feature with the largest relevance, $max_{x\in S^c, W_x=0} V_x$. Add the selected feature to the set S.
3. Repeat Step 2 until the redundancy is not zero for all features in S^c.
4. Select the feature that has the largest MIQ value with nonzero relevance and nonzero redundancy in S^c, and add the selected feature to the set S.
$$\max_{x\in S^c} \mathrm{MIQ}_x = \max_{x\in S^c} \frac{V_x}{W_x} = \max_{x\in S^c} \frac{I(x,y)}{\frac{1}{|S|}\sum_{z\in S} I(x,z)}$$
5. Repeat Step 4 until the relevance is zero for all features in S^c.
6. Add the features with zero relevance to S in random order.

Neighbourhood Component Feature Selection (NCFS)

In [10], the authors introduced the Neighbourhood Component Feature Selection (NCFS) algorithm, a nearest neighbor-based feature selection method. NCFS was inspired by the Neighborhood Components Analysis (NCA) algorithm, proposed by [16]. With a regularisation term, NCFS maximises the expected leave-one-out classification accuracy using the gradient ascent technique. NCFS aims to iteratively update a weight vector to minimize an error function. It does so by computing probabilities based on the current weights, updating the weights based on these probabilities, and adjusting the step length dynamically to optimize convergence. The algorithm continues iterating until convergence, at which point it returns the final weight vector. The algorithmic form of the method can be seen below (as presented in [10], please refer to it for more detailed information about NCFS).

```
$T$ : training set, $\alpha$ : initial step length, $\sigma$ : kernel width, $\lambda$ : regularization parameter, $\eta$ : small positive constant;
Initialization: $\mathbf{w}^{(0)} = (1,1,\dots,1), \epsilon^{(0)} = -\infty, t = 0$
repeat
  for $i = 1, \cdots, N$ do
    Compute $p_{ij}$ and $p_i$ (probabilities) using $\mathbf{w}^{(t)}$ according to the following equations
```

$$D_{\mathbf{w}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{l=1}^{d} w_l^2 |x_{il} - x_{jl}|$$

$$p_{ij} = \begin{cases} \frac{\kappa(D_{\mathbf{w}}(\mathbf{x}_i,\mathbf{x}_j))}{\sum_{k\neq i} \kappa(D_{\mathbf{w}}(\mathbf{x}_i,\mathbf{x}_k))}, & \text{if } i \neq j \\ 0, & \text{if } i = j \end{cases}$$

$$p_i = \sum_j y_{ij} p_{ij}$$

where $\kappa(z) = \exp(-z/\sigma)$ is a kernel function and the kernel width σ is an input parameter that influences the probability of each point being selected as the reference point.

```
  for $l = 1, \cdots, d$ do
    $\Delta_l = 2\left(\frac{1}{\sigma}\sum_i \left(p_i \sum_{j\neq i} p_{ij} |x_{il} - x_{jl}| - \sum_j y_{ij} p_{ij} |x_{il} - x_{jl}|\right) - \lambda\right) w_l^{(t)}$
  $t = t + 1$
  $\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} + \alpha\mathbf{\Delta}$
  $\epsilon^{(t)} = \xi(\mathbf{w}^{(t-1)})$
  if $\epsilon^{(t)} > \epsilon^{(t-1)}$ then
    $\alpha = \text{constant1} * \alpha$ (constant1 selected by the user)
  else
    $\alpha = \text{constant2} * \alpha$ (constant2 selected by the user)
until $|\epsilon^{(t)} - \epsilon^{(t-1)}| < \eta$
$\mathbf{w} = \mathbf{w}^{(t)}$
return $\mathbf{w}$
```

Least Absolute Shrinkage and Selection Operator (LASSO)

In [17], the author introduced the least absolute shrinkage and selection operator (LASSO) as a method for both shrinking and selecting variables in regression and generalised regression problems. The lasso algorithm does not prioritise subsets, but instead implements a continuous shrinking process that can generate coefficients that are equal to zero. For a given value of λ (positive), the aim of the algorithm is to minimise the following:

$$\left(\frac{1}{2N}\sum_{i=1}^{N} (y_i - \beta_0 - x_i^T\beta)^2 + \lambda \sum_{j=1}^{p} |\beta_j|\right)$$

where:
N is the number of observations.
y_i is the response at x_i
x_i is a vector of length p
λ is a regularization parameter
The parameters β_0 and β are a scalar and a vector of length p, respectively.

Recursive Feature Elimination (RFE)

Recursive Feature Elimination is a greedy algorithm that searches for the optimal subset. The primary concept is to repeatedly create a model and then choose the worst variable. The selected variable is then eliminated, and the procedure is repeated until all of the remaining variables have been explored. To develop the regressor and assess the importance of predictors, the RFE algorithm requires a machine learning algorithm (e.g., support vector machine, random forest, logistic regression, or naive Bayes) [18]. Random Forest Regressor (RFR) technique was used as the RFE model's regressor in this investigation. The algorithmic form is presented below [19].

```
Inputs: Training set Tr, Set of features $\mathbf{F} = \{\boldsymbol{f_1} \dots\dots . \boldsymbol{f_\alpha}\}$, Ranking method RFR
Outputs: Ranked list of features
 for $i$ =1:a do:
     Rank set F based on RFR
     $\boldsymbol{f}^* \leftarrow$ last ranked feature in F
     $\mathbf{F} \leftarrow \mathbf{F} - \boldsymbol{f}^*$
end
Return ranked list of features
```

The stopping criteria is either a number of required features (specified by the user) or performance metrics (e.g. when the specified criterion value does not decrease anymore).

USE CASE, DATASETS AND PREPROCESSING

Use Case

The objective function in the currently studied laser assemblies is to utilize the gathered information throughout multiple step assembly operation and to predict whether a product will be defective or otherwise.

Information was gathered through sensors, actuators and Human Machine Interface (HMI), for the assembly of two laser modules (d-serie and s-serie) which consist of emitters, mirrors and lenses that combine the laser beams of the emitters in one unified beam (output beam). For the d-serie laser module, there are 62 assembly steps (Fast-Axis Collimating (FAC) placements, Slow-Axis Collimating (SAC) placements, Mirrors placements, Turning Mirror placement, Polarization and Fiber Lens placement) in the assembly process.

For the s-serie laser module, there are 31 assembly steps (FAC, SAC, Mirrors and Fiber Lens placement). The assembly processes follow the same pathway with a number of differences. The process is automated using a special assembly robot.

Noteworthy, from this section onwards the details of the assembly process and associated components are proprietary and commercially sensitive. The case is explained in wider context avoiding the revelation of sensitive information from the manufacture.

Datasets and preprocessing

Prior to the assembly process, the emitters' power is measured to ensure that they are working. Measuring the geometry of the beams, their power, and their convergence to the centroid is done at each stage of the assembly process. Each assembly stage (FAC, SAC etc.) introduces new variables as inputs. The width of the Fast Axis beam, the width of the Slow Axis beam, the centroid of the beam, the beam pointing and the power of the beam are important inputs that are measured. The response variable that needs to be predicted is the output power of the laser module.

The data accumulated for the d-serie module reached 1411 observations (completed assembled laser modules), and the input features (predictors) for the d-serie module are 1181. The data accumulated for the s-serie module reached 1628 observations, with 530 input features. The datasets were shuffled in order to reduce bias.

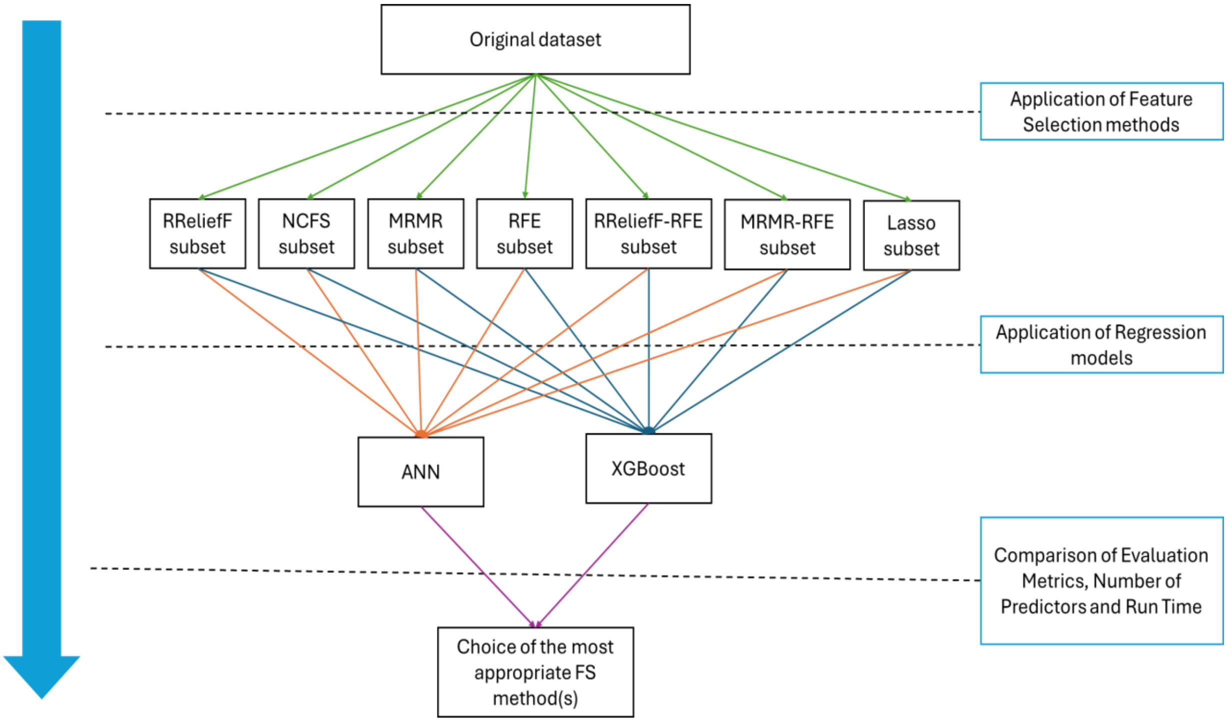
Original dataset
Application of Feature Selection methods
RReliefF subset
NCFS subset
MRMR subset
RFE subset
RReliefF-RFE subset
MRMR-RFE subset
Lasso subset
Application of Regression models
ANN
XGBoost
Comparison of Evaluation Metrics, Number of Predictors and Run Time
Choice of the most appropriate FS method(s)

FIGURE 1. Methodology for the current study

IMPLEMENTATION

The feature selection methods applied on the provided datasets are RReliefF, MRMR, RFE, RReliefF-RFE, MRMR-RFE, NCFS and Lasso. The methodology steps can be seen in Figure 1.

To create the combined methods, a filter method is applied to the original dataset and then a wrapper method is applied to the subset that was created by the filter. The filter is applied first, as it does not need high computational power, it is not as time consuming as the wrapper in high-dimensional datasets and it avoids overfitting. The wrapper is applied second, on the subset that was created by the filter, to reduce the size of the subset even more but at the same time maintain the predictive power.

To create the hybrid methods, RFE was applied on the subsets generated by RReliefF and MRMR. However, for the d-serie dataset, RFE was not applied to the subset formed by MRMR due to MRMR identifying only five features as significant, which is considered a limited number.

For NCFS, variables that have weight less than 0.1 are considered as irrelevant and are not included in the subset.

To execute RReliefF, k neighbours were set to 10. Predictors that scored a positive importance weight were included in the RReliefF subset.

For MRMR, features that had importance weight higher than 0.1 were considered important.

To deploy the RFE method, RFR was chosen as the regressor. Based on RFR's performance, the importance of the predictors was assessed. Due to the high-dimensionality of both datasets, the step of the RFE was set to 10. That way, in each iteration, instead of 1 predictor being removed, 10 were removed. This measure was taken in order to minimise the computational time, because of the high complexity of the algorithm. 5-fold cross-validation was also applied to avoid misleading results.

Regarding Lasso, all the features that had importance that was not 0 were included in the subset.

The size of the created subsets along with the computational time needed to create them, can be seen in Table 1 and Table 2 in Appendix I.

After all the subsets were created, ANNs and XGBoost were applied to all of them in order to predict the output power of the laser module.The metrics that were used to evaluate the performance of ANN and XGBoost are , root mean square error (RMSE), mean absolute error (MAE) and mean absolute percentage error (MAPE).

RESULTS

The evaluation metrics for the training and testing sets can be seen in Figure 2, Figure 3 and in Tables 1 and 2 (Appendix I).

It should be noted that overfitting has been observed in XGBoost for the s-serie module. Probably tuning of the applied models would solve this problem, but it should be emphasized that enhancing the predictive power of the models lies outside the scope of this research.

<u>d-serie module dataset</u>

Predictors

All methods, except RReliefF, reduced the original number of predictors (1181 predictors in the original dataset) by more than 80% (see Table 1). RReliefF achieved a reduction of 17.4%.

Evaluation metrics

In the following results, small differences between the metrics of different models will not be taken into consideration (e.g. $MAPE$ difference of 0.1%-0.2% will be considered insignificant).

<u>ANN</u>

RReliefF, and RFE performed similar to the original dataset on both training and testing sets. NCFS and RReliefF-RFE demonstrated slightly inferior performance to the original dataset on both training and testing sets (but the differences are insignificant). MRMR performed poorly on both sets. Lasso demonstrated similar performance to the original dataset on the training set, and slightly better on the testing set.

<u>XGBoost</u>

RReliefF, RFE and Lasso demonstrated similar or slightly better performance to the original dataset on both training and testing sets. NCFS and RReliefF-RFE exhibited slightly worse performance than the original dataset on the training set but similar performance on the testing set. MRMR performed poorly on both sets.

It can be concluded that the feature selection methods that performed well, regardless of the applied regression model, are: RReliefF, RFE, RReliefF-RFE, NCFS and Lasso. MRMR performed poorly with both prediction methods and for that reason it will not be considered anymore as a possible candidate for feature selection for the d-serie dataset.

Computational time

The model can only be useful if the prediction is achieved before the end of assembly cycle time. Verification required us to to compare the computational time of the feature selection methods by setting one method's computation time as benchmark for our calculations. This allows for a more comprehensive representation of the time factor irrespective of hardware settings. The computation time of RFE will be considered as the benchmark computational time for our calculations. The saving percentage in computational time is given by:

$$Saving = 100x\frac{t_{benchmark} - t_{solution}}{t_{benchmark}}$$

All methods, except RReliefF-RFE, achieved saving percentages higher than 96%. RRelief-RFE achieved a saving percentage of 30%. It should be noted here that the importance of computational time is defined by the nature of the problem. In cases like the one studied in this paper, feature selection is not a repetitive process; it takes place at the beginning of the research to exclude redundant information, and acts as a guide for researchers/experts on the field to understand which variables and processes have a higher impact on the system and the final outcome. Thus, the computation time in the specific case is not important. In cases that require real-time update of the models and repetitive feature selection, computational time is important and is a factor that the researchers should take into consideration to decide which feature selection method is the most appropriate to use.

Conclusions for the feature selection on the d-serie module dataset

RReliefF, RFE, RReliefF-RFE, NCFS and Lasso are the methods that perform well regardless of the applied regression model.

If computational time is the most important factor, the ranking (from best to worst) is as follows: RReliefF, Lasso, NCFS, RReliefF-RFE, RFE.

If achieving the highest reduction in the number of predictors whilst maintaining predictive accuracy is the main goal, the ranking (from best to worst) is as follows: RReliefF-RFE, NCFS, Lasso, RFE, RReliefF. Researchers may choose the method that produces the smallest subset for various reasons. A smaller subset of predictors is often easier to comprehend and communicate, making it more accessible to a broader audience, including stakeholders and decision-makers. This enhanced interpretability not only facilitates a clearer understanding of the model's behaviour but can also lead to more confident and informed decision-making. Additionally, a concise subset may highlight the most influential predictors, shedding light on the key factors driving the predictions models.

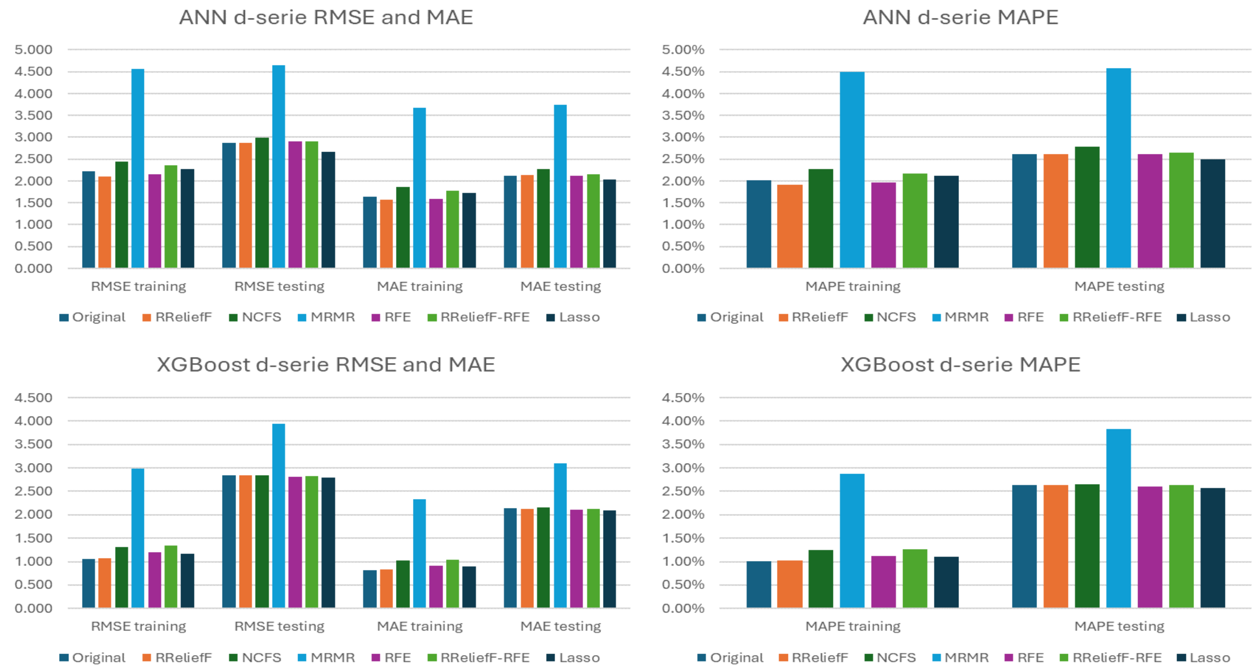

FIGURE 2. Train and Test metrics for all methods for the d-serie module. (Top ANN) (Bottom XGBoost)

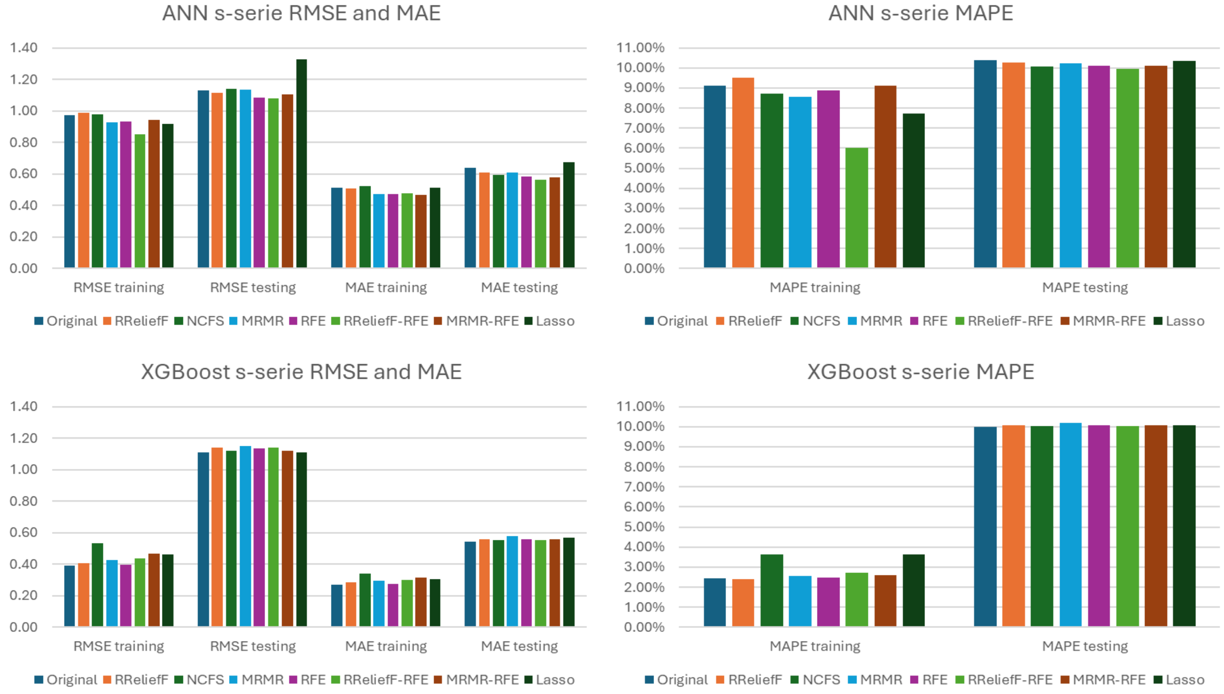

FIGURE 3. Train and Test metrics for all methods for the s-serie module. (Top ANN) (Bottom XGBoost)

s-serie module dataset

Predictors

NCFS, RRelieF-RFE, MRMR-RFE and Lasso reduced the original number of predictors (530 predictors in the original dataset) by more than 70% (see Table 2). RReliefF, MRMR and RFE did not reduce the original predictors by a high percentage, compared to the other methods.

Evaluation metrics

ANN

RReliefF and NCFS performed similar or slightly worse than the original dataset on the training set and performed slightly better on the testing set. MRMR, RFE, RReliefF-RFE and MRMR-RFE performed better than the original dataset on both training and testing sets. Lasso performed slightly better than the original dataset on the training set and slightly worse on the testing set, except the testing MAPE that was better (10.34% compare to the 10.40% of the original dataset).

The best performance was demonstrated from RReliefF-RFE and NCFS which achieved a testing MAPE of 9.96% and 10.08%, respectively, compared to the original dataset that had 10.40%.

XGBoost

RReliefF, MRMR and RFE performed similar to the original dataset on both training and test sets. NCFS performed worse on the training set and similar on the testing set compared to the original dataset. RReliefF-RFE and MRMR-RFE performed slightly worse on the training set and similar on the testing set compared to the original dataset. Lasso performed worse on the training set and similar on the testing set.

The best performance was demonstrated from NCFS and RReliefF-RFE which achieved a testing MAPE of 10.01% and 10.04%, respectively, compared to the original dataset that had 10.00%.

Computational time

All methods, except RReliefF-RFE and MRMR-RFE, achieved saving percentages higher than 97%. RRelief-

RFE and MRMR-RFE achieved a saving percentage of 74.26% and 38.23%, respectively.

Conclusions for the feature selection on the s-serie module dataset

All methods performed well regardless of the applied regression model.

If computational time is the most important factor, the ranking (from best to worst) is as follows: MRMR, RReliefF, NCFS, Lasso, RReliefF-RFE,MRMR-RFE and RFE.

If achieving the highest reduction in the number of predictors whilst maintaining predictive accuracy is the main goal, the ranking (from best to worst) is as follows: NCFS, Lasso, RReliefF-RFE, MRMR-RFE, RReliefF, MRMR and RFE.

CONCLUSION AND FUTURE WORK

The primary objective of this study was to provide actionable advice on selecting the most suitable feature selection methods for intricate industrial scenarios, particularly within manufacturing and assembly sectors, where the advent of Industry 4.0 has led to a proliferation of available data.

Overall, all feature selection methods, with the exception of MRMR on the d-serie dataset, exhibited either similar or slightly better accuracy compared to the original datasets. This parity in performance metrics underscores the importance of tailoring the choice of feature selection method to the unique requirements of each individual case.

For instance, if the main concern is the time efficiency in the feature selection process, methods such as RReliefF, NCFS, and Lasso emerge as particularly attractive options due to their ability to efficiently identify relevant features without sacrificing predictive accuracy. Conversely, if the primary goal is to enhance interpretability, thereby facilitating a deeper understanding of the manufacturing process and pinpointing critical areas for improvement or optimization, then methods like RReliefF-RFE, NCFS, and Lasso offer greater value. By reducing the number of predictors while maintaining predictive accuracy, these methods not only enhance interpretability but also provide actionable insights that can guide decision-making in manufacturing processes.

It's worth noting that while MRMR demonstrated poor generalization, especially on the d-serie dataset, its performance on the s-serie dataset suggests potential utility in specific contexts. However, its inconsistent performance and the limited subset size on the d-serie dataset render it less favorable compared to other methods.

Throughout the study, evaluating the generalization of MRMR-RFE was rendered unfeasible due to the MRMR subset for the d-serie dataset containing only 5 predictors, rendering RFE inapplicable.

Despite the study's focus on laser module assembly lines, the findings are likely applicable to a broader range of complex, real-time assembly processes within manufacturing. This suggests that the insights obtained from this research have the potential to inform and improve manufacturing processes beyond the specific datasets examined, indicating the broader relevance and applicability of the selected feature selection methods.

The conclusions from this study also provide a validation for RReliefF-RFE's (which was briefly introduced in our prior work [20]) performance on manufacturing settings, proven in a laser assembly process. As future work, testing the proposed RReliefF-RFE on wider applications and datasets (e.g. water treatment, supply chains and environmental impact) can further evaluate the generalisation of the method in wider industrial applications.

DECLARATION OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

ACKNOWLEDGMENT

The authors would like to express their sincere thanks to Kyriakos Kandris for his insight on the importance of feature selection in real-time scenarios and the suggestions on how to present the computational time.

FUNDING

This work has been carried out in the framework of the IQONIC project, which received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 820677.

REFERENCES

[1] J. Li and H. Liu, “Challenges of Feature Selection for Big Data Analytics,” *IEEE Intell Syst*, vol. 32, no. 2, pp. 9–15, Mar. 2017, doi: 10.1109/MIS.2017.38.

[2] S. St. John *et al.*, “Predicting chatter using machine learning and acoustic signals from low-cost microphones,” *International Journal of Advanced Manufacturing Technology*, vol. 125, no. 11–12, pp. 5503–5518, Apr. 2023, doi: 10.1007/s00170-023-10918-z.

[3] H. Wu, T. Yang, H. Li, and Z. Zhou, "Air quality prediction model based on mRMR–RF feature selection and ISSA–LSTM," *Sci Rep*, vol. 13, no. 1, Dec. 2023, doi: 10.1038/s41598-023-39838-4.

[4] F. Houndekindo and T. B. M. J. Ouarda, "Comparative study of feature selection methods for wind speed estimation at ungauged locations," *Energy Convers Manag*, vol. 291, Sep. 2023, doi: 10.1016/j.enconman.2023.117324.

[5] X. Wen and Z. Xu, "Wind turbine fault diagnosis based on ReliefF-PCA and DNN," *Expert Syst Appl*, vol. 178, Sep. 2021, doi: 10.1016/j.eswa.2021.115016.

[6] A. Nawandhar, N. Kumar, V. R, and L. Yamujala, "Stratified squamous epithelial biopsy image classifier using machine learning and neighborhood feature selection," *Biomed Signal Process Control*, vol. 55, 2020, doi: 10.1016/j.bspc.2019.101671.

[7] S. Kim, "Innovating knowledge and information for a firm-level automobile demand forecast system: A machine learning perspective," *Journal of Innovation and Knowledge*, vol. 8, no. 2, Apr. 2023, doi: 10.1016/j.jik.2023.100355.

[8] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers and Electrical Engineering*, vol. 40, no. 1, pp. 16–28, Jan. 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[9] L. Y. Chuang, H. W. Chang, C. J. Tu, and C. H. Yang, "Improved binary PSO for feature selection using gene expression data," *Comput Biol Chem*, vol. 32, no. 1, pp. 29–38, 2008, doi: 10.1016/j.compbiolchem.2007.09.005.

[10] W. Yang, K. Wang, and W. Zuo, "Neighborhood component feature selection for high-dimensional data," *J Comput (Taipei)*, vol. 7, no. 1, pp. 162–168, 2012, doi: 10.4304/jcp.7.1.161-168.

[11] M. Fernandes, A. Canito, V. Bolón-Canedo, L. Conceição, I. Praça, and G. Marreiros, "Data analysis and feature selection for predictive maintenance: A case-study in the metallurgic industry," *Int J Inf Manage*, vol. 46, pp. 252–262, Jun. 2019, doi: 10.1016/j.ijinfomgt.2018.10.006.

[12] R. J. Urbanowicz, M. Meeker, W. la Cava, R. S. Olson, and J. H. Moore, "Relief-based feature selection: Introduction and review," *Journal of Biomedical Informatics*, vol. 85. Academic Press Inc., pp. 189–203, Sep. 01, 2018. doi: 10.1016/j.jbi.2018.07.014.

[13] M. Robnik-Sikonja and I. Kononenko, "An adaptation of Relief for attribute estimation in regression," 1997.

[14] C. Ding and H. Peng, "Minimum Redundancy Feature Selection from Microarray Gene Expression Data," 2003. [Online]. Available: http://www.nersc.gov/~cding/MRMR/.

[15] "Rank features for regression using minimum redundancy maximum relevance (MRMR) algorithm." Accessed: Apr. 22, 2024. [Online]. Available: https://www.mathworks.com/help/stats/fsrmrmr.html

[16] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighbourhood Components Analysis," 2005.

[17] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," 1996.

[18] C. Wang, Y. Pan, J. Chen, Y. Ouyang, J. Rao, and Q. Jiang, "Indicator element selection and geochemical anomaly mapping using recursive feature elimination and random forest methods in the Jingdezhen region of Jiangxi Province, South China," *Applied Geochemistry*, vol. 122, Nov. 2020, doi: 10.1016/j.apgeochem.2020.104760.

[19] J. L. F. Chacón, T. F. de Barrena, A. García, M. S. de Buruaga, X. Badiola, and J. Vicente, "A novel machine learning-based methodology for tool wear prediction using acoustic emission signals," *Sensors*, vol. 21, no. 17, Sep. 2021, doi: 10.3390/s21175984.

[20] N. G. Markatos, A. Mousavi, G. Pippione, and R. Paoletti, "Industry 4.0-Based Framework for Real-Time Prediction of Output Power of Multi-Emitter Laser Modules during the Assembly Process," *Electronics (Switzerland)*, vol. 12, no. 3, Feb. 2023, doi: 10.3390/electronics12030766.

BIOGRAPHIES

Nikolaos Grigorios Markatos received the B.Sc. degree and integrated M.Sc. in applied mathematics and physics from National Technical University of Athens (NTUA) in 2018 and the Ph.D. degree in artificial intelligence and process optimization from Brunel University London, London, U.K in 2023. He is a Research Fellow with the College of Engineering, Design and Physical Sciences, Brunel University. His research interests include Industry 4.0 automation, real-time predictive models and minimization of defects in manufacturing processes.

Alireza Mousavi received the B.Sc. degree in industrial engineering and the Ph.D. degree in system engineering from Brunel University London, London, U.K., in 1994 and 2000, respectively. He is Professor with the College of Engineering, Design and Physical Sciences, Brunel University. His research interests include digital transformation and smartification of industrial systems, especially within the Industry 4.0 context covering sensors-actuation, signal processing and feature extraction, machine learning, modeling, control and optimization.

Evina Katsou received the MEng in Chemical Engineering, Department of Chemical Engineering, National Technical University of Athens (NTUA), Greece, in 2005. In 2008, she received the MSc in Water Resources Science & Technology from the department of Civil Engineering, NTUA, Greece. In 2011, she received the PhD in 'Wastewater Treatment with the Use of Membranes', Department of Chemical Engineering, NTUA, Greece. She is Professor in Water Engineering in Imperial College London. Her research is leading to measurement, control and mitigation of greenhouse gas emissions of treatment processes, and is being adopted to set strategies and plans for developing the AI-Based Water Engineering Solutions. Her research also focuses on the development and implementation of frameworks for sustainability and circularity measurement and assessment.

Giulia Pippione received the M.Sc. in advanced technologies in Physics from University of Turin in 2016. Currently she is responsible of the Optical Packaging Department in laser diode multi-emitter manufacturing in Convergent Photonics - Prima Electro since 2017. Her research interests include design for optical and mechanical packaging of semiconductor devices, optimization of manufacturing processes in a production line and automation of micro-optics alignments for laser beams collimation. Currently she is involved in R&D projects for the development of single and multi-emitter laser sources in visible range.

Roberto Paoletti graduated in Electronic Engineering, summa cum laude at University of Genoa in 1991 and received the Ph.D. at Politecnico of Turin, in 1994, dealing with design, simulation and characterization of semiconductor lasers for advanced telecommunication systems. He is responsible for the Laser Diode Fab of Convergent Photonics in the Prima Electro Organization, development and manufacturing site of high power diode laser for industrial applications. His research interests include design for manufacturing, device engineering and long term reliability.

This article has been accepted for publication in IEEE Intelligent Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MIS.2024.3416678

APPENDIX I

TABLE 1. d-serie module predictors, reduction percentage, run time and evaluation metrics

d-serie module ANN										
Method	***Predictors***	***Reduction percentage***	***Run Time (s)***	***Saving percentage***	***MAPE training***	***MAPE testing***	***RMSE training***	***RMSE testing***	***MAE training***	***MAE testing***
Original	1181				2.02%	2.61%	2.23	2.87	1.65	2.12
RReliefF	975	-17.40%	13.46	99.80%	1.92%	2.62%	2.10	2.86	1.57	2.13
NCFS	130	-89.00%	137.73	97.93%	2.28%	2.79%	2.44	2.99	1.86	2.27
MRMR	5	-99.60%	52.1	99.22%	4.50%	4.58%	4.56	4.64	3.67	3.74
RFE	201	-83.00%	6646.69	0.00%	1.96%	2.62%	2.16	2.90	1.59	2.12
RReliefF-RFE	95	-92.00%	4650.93	30.03%	2.11%	2.74%	2.30	2.97	1.72	2.23
Lasso	148	-87.50%	43.54	99.34%	2.12%	2.50%	2.27	2.67	1.73	2.03
d-serie module XGBoost										
Original	1181				1.00%	2.64%	1.05	2.84	0.82	2.14
RReliefF	975	-17.40%	13.46	99.80%	1.02%	2.63%	1.07	2.84	0.83	2.13
NCFS	130	-89.00%	137.73	97.93%	1.24%	2.66%	1.32	2.85	1.02	2.15
MRMR	5	-99.60%	52.1	99.22%	2.88%	3.82%	2.99	3.95	2.34	3.10
RFE	201	-83.00%	6646.69	0.00%	1.12%	2.61%	1.19	2.82	0.92	2.11
RReliefF-RFE	95	-92.00%	4650.93	30.03%	1.27%	2.63%	1.35	2.83	1.04	2.13
Lasso	148	-87.50%	43.54	99.34%	1.10%	2.58%	1.17	2.79	0.90	2.08

TABLE 2. s-serie module predictors, reduction percentage, run time and evaluation metrics

s-serie module ANN										
Method	***Predictors***	***Reduction percentage***	***Run Time (s)***	***Saving percentage***	***MAPE training***	***MAPE testing***	***RMSE training***	***RMSE testing***	***MAE training***	***MAE testing***
Original	530				9.11%	10.40%	0.97	1.13	0.51	0.64
RReliefF	234	-55.85%	7.78	99.35%	9.49%	10.25%	0.99	1.12	0.51	0.61
NCFS	25	-95.28%	14.6	98.77%	8.72%	10.08%	0.98	1.14	0.52	0.59
MRMR	369	-30.38%	6.21	99.48%	8.57%	10.22%	0.93	1.14	0.47	0.61
RFE	390	-26.42%	1191.1	0.00%	8.88%	10.13%	0.93	1.08	0.47	0.58
RReliefF-RFE	124	-76.60%	306.64	74.26%	6.03%	9.96%	0.85	1.08	0.48	0.56
MRMR-RFE	139	-73.77%	735.79	38.23%	9.10%	10.12%	0.94	1.10	0.47	0.58
Lasso	78	-85.28%	15.38	98.71%	7.72%	10.34%	0.92	1.33	0.51	0.68
s-serie module XGBoost										
Original	530				2.44%	10.00%	0.39	1.11	0.27	0.54
RReliefF	234	-55.85%	7.78	99.35%	2.41%	10.06%	0.41	1.14	0.29	0.56
NCFS	25	-95.28%	14.6	98.77%	3.61%	10.01%	0.53	1.12	0.34	0.55
MRMR	369	-30.38%	6.21	99.48%	2.55%	10.17%	0.43	1.15	0.29	0.58
RFE	390	-26.42%	1191.1	0.00%	2.46%	10.05%	0.40	1.13	0.28	0.56
RReliefF-RFE	124	-76.60%	306.64	74.26%	2.73%	10.04%	0.43	1.14	0.30	0.55
MRMR-RFE	139	-73.77%	735.79	38.23%	2.61%	10.05%	0.46	1.12	0.31	0.56
Lasso	78	-85.28%	15.38	98.71%	3.64%	10.07%	0.46	1.11	0.30	0.57