Abstract—With the advent of Industry 4.0, Data Science and Explainable Artificial Intelligence (XAI) has received considerable interest in recent literature. However, the entry threshold into XAI in terms of computer coding and the requisite mathematical apparatus, is really high. For fault diagnosis of steel plates, this work reports on a methodology of incorporating XAI based insights into the Data Science process of development of high precision classifier. Using Synthetic Minority Oversampling Technique (SMOTE) and notion of medoids, insights from XAI tools viz. Ceteris Peribus profiles, Partial Dependence and Breakdown profiles have been harvested. Additionally, insights in the form of IF-THEN rules have also been extracted from an optimized Random Forest and Association Rule Mining. Incorporating all the insights into a single ensemble classifier, a 10 fold cross validated performance of 94% has been achieved. In sum total, this work makes three main contributions viz.: methodogy based upon utilization of medoids and SMOTE, of gleaning insights and incorporating into model development process. Secondly the insights themselves are contribution, as they benefit the human experts of steel manufacturing industry, and thirdly a high precision fault diagnosis classifier has been developed.

Index Terms—Automated Manufacturing; Predictive Maintenance; Explainable Artificial Intelligence; Insight Harvesting; Fault Diagnosis

I. INTRODUCTION

Data Science carries two senses of meaning namely ‘Science of Data’ and ‘Science by Data’. In the sense of Science of Data, new methods, techniques and insights for handling, manipulating and predictive modeling of data are developed and extended. In the second sense i.e. ‘Science by Data’, data of a specific scientific domain is examined and knowledge is extracted from it for the betterment and enhancement of the original scientific domain. It is this second sense namely doing ‘Science by Data’, that this paper owes the spirit and motivation.

Insight, the structural understanding of phenomena, is the essence of human intelligence. Usually, insight is distilled from the fusion of information from various sources e.g. past experiences, analysis techniques and methods (Roscher et al. 2020). There is no reason to not to expect insights from Artificial Intelligence (AI) as well, which more appropriately may (in our view, must) be called Machine Intelligence (MI). AI has shown great strides in recent times. Passing through the stages of Symbolic AI, Computational AI, it has now reached to another milestone namely, Explainable AI (XAI) or more appropriately Explainable Machine Learning (XML).

Explainability has remained a much sought after feature of AI systems right from the era of Expert Systems hype of AI, mainly because even the best human experts needed assistance/explanation from the machine to reach the right decisions. Non-availability of explainability has also remained one of the major barriers in a wide-scale practical adoption of AI. Improved understanding of the inner working of a phenomenon and its machine representation i.e. ML model leads to the correction of its weaknesses and flaws. Understanding of one subsystem, obtained from XAI, may also be transferred to another system thus ‘democratizing’ the invaluable expertise in a cumulative manner. Black box models, though considerably successful in prediction tasks, but at the same time they present a totally opaque face when it comes to explanation and insight. In recent time the confluence of Data Science (as Science of Data) with AI techniques has achieved great deal of success to peep into even a high precision predictive models of black-box genre. Clearly, it is an area located at the crossroads of data-fusion and information fusion.

The understanding, structural relations and the relational insight achieved by data is just as good as the data itself. It cannot guarantee the causal scientific relationship of the external world. Nevertheless, an explainable AI model offers a viable hypothesis to begin with (cf. Section 3 of (Roscher et al. 2020)). The journey of scientific enquiry, at least may be begun in most of the cases in the right direction. Furthermore, explainability also ensures that only meaningful variables are allowed to take part in final modelling tasks thus approaching the ideal of a truly parsimonious model as well. Christina et. al. (Christina B. Azodi, Jiliang Tang, and Shin-Han Shiu 2020) have highlighted the same point more aptly: “ML interpretation strategies mostly do not identify causal relationships between input features and labels. Instead, interpretations should be used to generate hypotheses of cause-effect relations that can be tested experimentally.”

Industrial plant monitoring focuses upon optimization of resources and thus attempts to achieve minimal error costs,
improved quality of production and safety of workplace. A timely, precise and indicative of the root-cause detections of defects or faults is fundamental to such monitoring. Compliance management also necissitates the explainablity of the machine learning models. That is why due to heavy cost of production, in general all industry, and in particular steel production industry has shown great intrest and effort to reduce the hugely expensive faults of production. Fault diagnostics in conjuction with fault isolation and root-cause analysis also contributes to precautionary maintenance or the prognostics of the production system (Soylemezoglu, Jagannathan, and Saygin 2011). Due to cost of faults and the ushering of Industry 4.0, Internet of Things (IoT) and Cyber Physical System (CPS), interest in data analytics based solutions has increased to a remarkable degree.

Fault diagnosis within an industrial production setup is further important on two more counts. Obviously, one for maintaining the quality of production. This requires fast and precise, ether human experts (Gabriel Schwartz and Ko Nishino 2020) or preferably intelligent digital solutions like machine and statistical learning based predictive models. As the human expertise becomes more and more scarce and costly, the automated digital solutions become even more relevant and in-demand. Second and more important utility of fault diagnosis is its use to identify the exact causes, variables and factors of production which are responsible for a particular fault. This usage of fault diagnosis, in fact, demands developing the insight into the production system through its various variables. It is exactly here that the present work is located. It aims to benefit the quality production of steel plates at industrial scale by enhancing the human insight of the fault producing factors and causes.

Steel plates fault diagnosis has been studied by various researchers. In (Nkonyana et al. 2019), authors employed various models viz. ANN (Kharal and Saleem 2012), SVM and Random Forests. Random Forest achieved the highest accuracy of 77.80% while SVM followed by 73.60%. So far the highest accuracy of prediction has been reported by Tian in (Tian, Fu, and Wu 2015). This SVM implementation reported a performance of 80.7443%. However, authors also noted that other sample balancing techniques may bring better result than the one used. In the settings of smart cities, the fault reporting and diagnostics using linked data has also been studied in (Consoli et al. 2017). Authors of (E.L. Russell, L.H.Chiang, and R.D.Braatz 2000) have applied the Canonical Variate Statistic, residual space PCA, DPCA, and CVA to the Tennessee Eastman process simulator of a wide variety of faults occurring in a chemical plant. Kazemi et.al. (Kazemi, Hajian, and Kiani 2018) calculate the positions (ranking) of each classifier compared to the five datasets and calculate the average ranking, and showed the quality of the proposed meta-classifier to be the best.

This paper is organized as follows: It comprises two main parts. Section III describes various details of machine learning experiments e.g. Class balancing, optimization and learner comparisons. Section IV focuses upon extraction of insights from the trained machine learning model. It extracts insights from XAI tools, namely Feature Importance (III-A1), Ceteris Peribus Profiles (III-A2), Partial Dependence Profiles (III-A3) and Breakdown Profiles (III-A4). Section III-B and III-C present the insights extracted from optimized random forest and association rules mining, respectively. Section IV describes how all the obtained insights may be incorporated into an ensemble learner. Finally Section V presents the discussion and conclusions. A graphical summary is presented in the following flow diagram:

Graphical Summary: Balancing of data has been followed by experimenting with 9 different classifier algorithms thus resulting into 36 different machine learning experiments. Optimized best performer, XAI and Association Rule Mining has been employed to develop an ensemble with high accuracy.

II. MACHINE LEARNING DETAILS

To correctly classify the type of surface defects of steel plates during industrial production, Semeion of Italy collected a data of 1941 faulty steel plates and recorded 27 observable characteristics/features of each of the faulty plate. This data is publically available through UCI (Semeion, n.d.). Data has 34 columns of which 27 are the observable variables/features and 7 binary columns each showing presence or absence of each of the seven types of faults. The last seven columns are one hot encoded classes, i.e. if the plate fault is classified as (say) “Stains” there will be a 1 in that column and 0’s in the other columns. However for the machine learning experiment requirements, following two changes have been made in present work:

1. Merged two factor variables viz. ‘TypeOfSteel_A300’ and ‘TypeOfSteel_A400’ as a single variable named ‘TypeOfSteel’.
2. Merged the 7 fault indicator variables into one variable showing all the 7 faults. This new target variable has been named as ‘Fault’.

Variable names of the dataset after the abovementioned two changes are given in Table II.
### TABLE I
**Variables of the Dataset**

| Variable                        | Description                                      |
|---------------------------------|--------------------------------------------------|
| Edges_Index                     | Outside_X_Index                                  |
| Edges_X_Index                   | Pixels_Areas                                     |
| Edges_Y_Index                   | SigmoidOfAreas                                   |
| Empty_Index                     | Square_Index                                     |
| Fault                           | Steel_Plate_Thickness                            |
| Length_of_Conveyor              | Sum_of_Luminosity                                |
| Log_X_Index                     | TypeOfSteel                                      |
| Log_Y_Index                     | X_Maximum                                        |
| LogOfAreas                      | X_Minimum                                        |
| Luminosity_Index                | X_Perimeter                                      |
| Maximum_of_Luminosity           | Y_Maximum                                        |
| Minimum_of_Luminosity           | Y_Minimum                                        |
| Orientation_Index               | Y_Perimeter                                      |
| Outside_Global_Index            |                                                  |

All the variables in dataset are numeric except ‘TypeOfSteel’ and ‘Fault’ which are categorical. ‘TypeOfSteel’ indicates type to be either A300 or A400. ‘Fault’ is the target variable containing 7 labels of fault names (cf. Figure 2). ‘Type Of Steel’ has important implications for the possible Faults. Interestingly Fault distribution for both kinds of steels i.e. A300 and A400 is highly imbalanced e.g. only one each instance of K_Scratch and Stains is found in A300 type of steel. Also Dirtiness appears to be rare for this type of steel. On the other side Z_Scratch is relatively a rare event for A400 type. Details may be further seen in Table II Medoids are important notion used in this work. Details shall be given latter, for the time being it is roughly defined as a prototypical observation (row of data) for a given category (say) ‘A300’. Medoids of A300 and A400 types of steel have been given in Figure 1.

### TABLE II
**TypeOfSteel vs. Faults**

| Fault      | A300 | A400 |
|------------|------|------|
| Bumps      | 279  | 123  |
| Common_Other | 266  | 407  |
| Z_Scratch  | 172  | 18   |
| Pastry     | 49   | 109  |
| Dirtiness  | 9    | 46   |
| K_Scratch  | 1    | 390  |
| Stains     | 1    | 71   |
| Total      | 777  | 1164 |

A. **Class Balancing**

As this is an imbalanced multiclass problem class balancing has rightly been pointed by (Tian, Fu, and Wu 2015) to be a promising direction of improvement. Counts and percentages for each class are shown in the bar plot of Figure 2.

As most of the machine learning algorithms carry a tacit assumption of labels of a classification dataset being equally distributed, therefore for an imbalanced problem, like the present one incorporate bias towards the majority class (Pan et al. 2020). Even if the size of dataset is large enough to provide for considerable cases of each class, the problem of validation of the generalization capability of the trained model becomes difficult. The situation even worsens for an extreme rarity condition (Kang 2020) alongwith the inherent randomness of data.

Various solutions and techniques have been developed to address this imbalance problem (Santiago Egea Gomez et al. 2019). Out of these three main ones have been chosen namely, undersampling, oversampling and SMOTE. Whereas the names of undersampling and oversampling render their internal working clear, technique of SMOTE (Synthetic Minority
Oversampling Technique) needs some explanation. SMOTE was originally introduced in (Chawla et al. 2002) and it is basically an oversampling method but before oversampling it creates, or technically saying 'synthesizes', a new instance of the minority class which is mathematically on the straight line between a pair of minority instance and one of its K nearest neighbors. This procedure is replicated for a predefined number of times.

By its very design SMOTE is implemented for a binary classification settings. However this work extended its computation for a multiclass settings using turn by turn section of max-class and each one class from the rest of smaller classes and then applying SMOTE upon each of the binary-class splits of the original data. Using all the 3 strategies upon the original dataset in total, yielded four datasets. Characteristics of each of these datsets are given in Table III.

Various machine learning algorithms have been applied to the problem of fault diagnostics of steel plates as noted in Introduction of this paper. Almost all of these methods were though aimed at achieving higher accuracy rather than gaining insight of the fault producing mechanism, as is the case here. To make an exhaustive and definitive search, present work employed 9 machine learning algorithms on each of the four datasets of Table III, thus resulting to 36 machine learning experiments in total. However such a large number of comparisons requires a level-field for all algorithms. Therefore a common-to-all 10 fold cross validation scheme was provided to each of the 36 ML experiments. Hardware is an HP Laptop with Intel® Core i7-7500U CPU @ 2.70Ghz. Final results of the comparison of 9 learners over the 4 datasets are given in Table IV, with 'True' in rows and ‘Predicted’ in Columns:

It is notable here for later use that Bumps and Common_Other seem to have something common in them. An ‘overlap’ while labeling the faults may not be ruled out here. Error percentage for each type of fault as calculated from the Confusion Matrix is also shown in Table VI.

### III. Insight Harvesting

This work primarily intends to harvest insight using which it achieves higher predictive precision. Insight is the accurate and deep (in the sense of structural) understanding of a system which is rooted into external real world. This in itself remains a subject of philosophical analysis whether explanation equates to an insight. This question becomes even more pronounced in view of the consideration that explanation is about the internal working whereas insight is somewhat externalized and related to real world causal relations. By its very nature insight is obtained by infusion of various pieces of information obtained from different sources and attempts. In the same vein this work gleans various pieces of information namely XAI, Optimized Random Forest based extracted rules and Association Rule Mining (henceforth ARM). Following point made in (Christina B. Azodi, Jiliang Tang, and Shin-Han Shiu 2020) is very much

### TABLE III
**DATASETS FOR (IM)BALANCING**

| Dataset       | Rows | Features | Class Size |
|---------------|------|----------|------------|
| Original      | 1941 | 27       | NA         |
| Undersample   | 385  | 27       | 55         |
| Oversample    | 4711 | 27       | 673        |
| SMOTE         | 4711 | 27       | 673        |

### TABLE IV
**CLASSIFICATION ACCURACY OF LEARNERS VS. DATASETS**

| Learner                  | Original | Oversample | SMOTE | Undersample |
|--------------------------|----------|------------|-------|-------------|
| Random Forest            | 0.7831   | 0.9418     | 0.9263| 0.7425      |
| K Nearest Neighbours     | 0.7285   | 0.9091     | 0.9125| 0.7040      |
| Extreme Gradient Boosting| 0.7274   | 0.8790     | 0.8588| 0.7016      |
| Support Vector Machine   | 0.7537   | 0.8664     | 0.8792| 0.7167      |
| Gen. Linear Model (elastic net) | 0.7100 | 0.8400     | 0.8700| 0.7300      |
| Linear Discriminant Analysis | 0.6847 | 0.7442     | 0.7739| 0.6963      |
| Naive Bayes              | 0.5806   | 0.6948     | 0.7334| 0.6651      |
| Decision Tree            | 0.6811   | 0.6911     | 0.7056| 0.6651      |
| Logistic Regression      | 0.3936   | 0.1874     | 0.1900| 0.1506      |

**TABLE V
CONFUSION MATRIX**

| Fault       | Bumps | Common_Other | Dirtiness | K_Scratch | Pastry | Stains | Z_Scratch |
|-------------|-------|--------------|-----------|-----------|--------|--------|-----------|
| Bumps       | 579   | 63           | 1         | 1         | 27     | 1      | 1         |
| Common_Other| 124   | 467          | 4         | 6         | 51     | 4      | 17        |
| Dirtiness   | 1     | 0            | 672       | 0         | 0      | 0      | 0         |
| K_Scratch   | 3     | 12           | 0         | 656       | 1      | 1      | 0         |
| Pastry      | 9     | 11           | 0         | 0         | 653    | 0      | 0         |
| Stains      | 0     | 1            | 0         | 0         | 672    | 0      | 0         |
| Z_Scratch   | 1     | 3            | 0         | 0         | 0      | 0      | 669       |

**TABLE VI
CONFUSION MATRIX : PROPORTIONS**

| Fault        | %age |
|--------------|------|
| Bumps        | 15.2 |
| Common_Other | 11.8 |
| Dirtiness    | 14.4 |
| K_Scratch    | 14.1 |
| Pastry       | 15.5 |
| Stains       | 14.4 |
| Z_Scratch    | 14.6 |

B. Optimization

Random Forest performed best on both the simple oversampling and SMOTE datasets. Considering that Oversampling...
in point here: “Just like there is no one universally best ML algorithm, there will not likely be one ML interpretation strategy that works best on all data or for all questions. Rather, the interpretation strategy should be tailored to what one wants to learn from the ML model and confidence in the interpretation will come when multiple approaches tell the same story.”

A. Explainable Artificial Intelligence

XAI is an umbrella term for a host of tools and techniques ranging from Mean Loss based variable importance and Model Specific tools to Model Agnostic techniques. The taxonomies introduced in (Barredo Arrieta et al. 2020) and (Baniecki and Biecek 2020) are very beneficial to the interested reader. XAI provides for domain expert learning from ML, human requirement of automatic decisions, compliance of regulations and passing of security audits of the trained predictive models. Industrial Informatics has also flourished in recent times by using XAI (De Silva et al. 2020). Specifically present work resorts to Mean Loss based feature importance, Ceteris Peribus Profiles, Partial Dependence Profiles and Breakdown Profiles for harvesting insights into the trained model.

1) Feature Importance: Feature importance is to be kept in view throughout as a number of techniques in XAI are available for almost all variables and thus produce unmanageable heap of information. Knowing the most significant as well as the non-collinear variables helps to reduce the cognitive load for ML interpretations. Non-collinearity is also a strong requirement in XAI tools like PDP. Hence computation of feature importance is the natural first step. Besides the feature importance obtained from the optimized Random Forest, a number of other measures may be computed and combined to benchmark the features for subsequent use (Figure 3). Short description of each computed measure is given as:

- CMIM: Minimal Conditional Mutual Information
- DT: Decision Tree
- Performance: Predictive Performance
- Ranger: RF Optimized using permutation strategy
- RF_simple: Random Forest simple
- Mean Loss: model agnostic XAI technique

![Importance of Variables (Scaled and Centered)](image)

Fig. 3. Variable Importance: Various criteria are available for importance measurement. This work mainly depends upon the Mean Loss (thick red line). All other measures have been scaled appropriately for plotting purpose.

Table VII shows the importance ranks of each feature (with 1 having largest mean loss). This importance rank shall be used subsequently to figure out their relative places within the Breakdown plots of each fault-medoid.

| Feature Rank | Feature Name        | Feature Rank |
|--------------|---------------------|--------------|
| 1            | Length_of_Conveyor  | 14           |
| 2            | Steel_Plate_Thickness| 15           |
| 3            | Orientation_Index   | 16           |
| 4            | TypeOfSteel         | 17           |
| 5            | X_Minimum           | 18           |
| 6            | Minimum_of_Luminosity| 19          |
| 7            | X_Maximum           | 20           |
| 8            | Square_Index        | 21           |
| 9            | LogOfAreas          | 22           |
| 10           | Pixels_Areas        | 23           |
| 11           | Edges_Y_Index       | 24           |
| 12           | Log_X_Index         | 25           |
| 13           | Outside_X_Index     | 26           |

Most of the XAI tools work best only for the least correlated features with no or minimum collinearity. Therefore correlation coefficient of all features has been calculated for filtering of features and five features namely, Sum_of_Luminosity, Pixels_Areas, X_Perimeter, X_Minimum, Y_Minimum are having correlation more than 0.90. It is noted here for subsequent usage in XAI tools where needed e.g. in Ceteris Peribus profiles and Breakdown plots.

2) Ceteris Paribus for Fault Medoids: A model which is otherwise opaque, may easily be peeped into if effect of each explanatory variable is examined separately. A further improvement is to examine one variable while keeping all others at some constant value, so to say, average value. Ceteris Peribus (CP) profile is the tool for this kind of analysis.
CP may be rigorously defined as follows: Let $D$ stand for a dataset with $n$ rows and $p$ columns. Here $p$ stands for the number of variables while $n$ stands for the number of observations. As local methods operate on a single observation, let $x^* \in \mathbb{R}^p$ stand for an observation of interest. Let $f : X \rightarrow \mathbb{R}$ denote for the model of interest, where $X = \mathbb{R}^p$ is the $p$-dimensional input space. Then Ceteris Paribus is the profile $g(z)$ for variable $x_i$ and observation $x$ is defined as:

$$g_{x^*}(z) = f(x^* | x_i = z)$$

As CP is a local technique i.e. based upon a single instance (row) of data. Thus any row in data may be chosen, however it is more suitable for the current scenario to choose to represent each Fault type by its most representative instance. In cluster analysis an instance with minimum dissimilarity (Yu et al. 2018) within a cluster is said to be its medoid. This notion is akin to that of ‘centroid’ in computation of clusters. While various methods of medoid calculation are available we choose the fastest one i.e. medians of all numeric variables and mode (most frequent level) of a categorical variable. It is important to note that medoid here have been calculated from the original imbalanced dataset so that they remain ‘typical’ and no randomness creeps in. So to say, for this work a medoid refers to an object within a Fault-set for which average dissimilarity between it and all the other members of the cluster is minimal. Other such works demonstrating the usefulness of typical prototypes for XAI may be found in (Gurumoorthy et al. 2019) and the references therein.

Typical fault profiles (medoids) used herein are given as Table VIII ordered by Variable Importance ranks as given in Table VIII.

Once each numeric variable is scaled, these medoids may also be visualized as radar plots (Figure 4). A closer examination of the radar patterns reveals following insights:

**Insights**

1. Bumps and Common_Other are having striking similarity of shape with each other.
2. Z_Scratch & K_Scratch are very distinct and their mutual differentiation is easier.
3. A typical Dirtiness case (i.e. medoid) is spread almost all over the 26 features.
4. Except K_Scratch all other typical fault cases have presence on the ‘Minimum_Of_Luminosity’ feature. Visually it may be see as a tail in the radar plot. This renders this variable important for classification.

![Figure 4. Radar Plot(s) of Medoids: Smaller radars present the profile of each medoid individually and at the same time they are legend for the combined radar plot at left.](image-url)
and 6 variables these turns out to be 42 CP-plots. The real utility of CP profile shines once the insights, general observing of the similarities, regularities, peculiarities and/or differences, are noted and separated for later use by human experts or by the machine algorithms. Therefore for the variables not shown here, just the gleaned insights are listed below:

**Insights**

1. Bumps and Common_Other almost always mirror each other in CP profile of all variables except X_Maximum. This indicates a kind of ‘overlap’ while labeling has been originally done.

2. For thinner plates (with thickness less than 100) maximum probability is for Bumps, which is understandable.

3. As the density rugs provide indication of support (by number of available observations), therefore e.g. Steel_Plate_Thickness is ‘interesting’ for values less than 100. Beyond that point curves are flat and hence uninteresting. Moreover beyond 100 supporting observations are scarce thus weakening the confidence of prediction.

![Ceteris Peribus Profiles](image)

**Fig. 5. Ceteris Peribus Profiles:** Although CP is essentially a local (i.e. limited to the single observation of interest) technique, however here the localness syndrome has been balanced by the use of Fault Medoids. Density rugs on the horizontal axis provide an indication of strength of available observations equivalently strength of evidence for the CP patterns.

3) **Partial Dependence Plots:** Partial Dependence (PD) Plot describes how certain set of variables affect an average prediction. It is important to differentiate between Ceteris Peribus (CP) and PD. CP is localized to a certain instance, and in present work this happens to be medoid of a fault. PD is non-local in its character as it talks about a general ‘average’ prediction, albeit the set of variables being chosen by the user. PD shows how the prediction ‘partially’ depends upon the few variables of interest. PD also describes the kind of relationship e.g. linear, cuvilinear and step etc. More elaborately, a PDP examines a variable of interest at a specific range. At each value within the range, the predictive model predicts for all cases (rows), and the prediction is then averaged out. Therefore a relation depicted by PDP is useful only once the features are non-colinear.

Technically, a PD may be written (Friedman 2001) as:

\[
\hat{f}_x(x_S) = E_{x_C} \left[ \hat{f}_{x} (x_S, x_C) \right] = \int \hat{f}_x(x_S, x_C) dP(x_C)
\]

where \(x_S\) are the features for which the partial dependence function should be plotted and \(x_C\) are the other features used in the machine learning model \(f\).

The partial function \(\hat{f}_x\) is estimated by calculating averages in the training data, also known as Monte Carlo method:

\[
\hat{f}_x(x_S) = \frac{1}{n} \sum_{i=1}^{n} \hat{f}_x(x_S, x_C^{(i)})
\]

This work, in the manner of (R. Berk and J. Bleich 2013), demonstrates the benefit of using optimized Random Forest and its PDPs to precisely indicate the predictor-response relationships under imbalanced classification settings which are quite common in industrial problems.

As humans can perceive only a few number of variables at any given time therefore in Figure 5[7] and 8[8] target features for PD plots are chosen to be small in number and those also from the most important variables/features.

![PDP profiles](image)

**Fig. 6. Partial Dependence Plots - 1:** Dependence of an average prediction upon different variables. Contribution of Steel Plate Thickness to CommonOther type of fault is noteworthy.

4) **Breakdown Profiles:** Medoids of each fault provides an important opportunity to figure out a hypothesis of how different variables affect the model-prediction of the specific medoid. Thus indicating the significance of different variables for different types of predictions. This is a kind of variable attribution exercise which attributes contribution of each individual variable to the present result. Technically known as Breakdown (BD) Profiles, there full plots showing each of the 27 variables is too space consuming. Therefore only one BD plot is shown in Figure 9[9]. One easily finds the most important contributors from a visual analysis of this plot. The variables in red color drag the prediction ‘backward’ and positive contributors push it forward and hence the net prediction result (in deep blue) is obtained.
For rest of the Fault-medoids only top-10 bigger contributors are listed alongwith their positive or negative contributions (using the +/- sign). For saving space we present the top ten features from the Breakdown plot of each fault-medoid in the following:

**Bumps**

| Feature                  | Contribution |
|--------------------------|--------------|
| Empty_Index              | +0.095       |
| Square_Index             | +0.068       |
| Luminosity_Index         | +0.056       |
| TypeOfSteel              | +0.047       |
| Orientation_Index        | +0.046       |
| Length_of_Conveyor       | +0.045       |
| Edges_X_Index            | +0.042       |
| X_Perimeter              | +0.038       |
| Minimum_of_Luminosity    | +0.038       |
| Maximum_of_Luminosity    | +0.033       |

**Common_Other**

| Feature                  | Contribution |
|--------------------------|--------------|
| Length_of_Conveyor       | +0.085       |
| Steel_Plate_Thickness    | +0.052       |
| X_Minimum                | +0.033       |
| Luminosity_Index         | -0.027       |
| Minimum_of_Luminosity    | -0.027       |
| Orientation_Index        | +0.027       |

**Dirtiness**

| Feature                  | Contribution |
|--------------------------|--------------|
| Y_Maximum                | +0.132       |
| Y_Minimum                | +0.121       |
| Edges_X_Index            | +0.043       |
| Edges_Y_Index            | +0.037       |
| X_Maximum                | +0.027       |
| Minimum_of_Luminosity    | +0.024       |
| TypeOfSteel              | -0.023       |
| Minimum_of_Luminosity    | +0.020       |
| Luminosity_Index         | +0.016       |
| Outside_X_Index          | +0.016       |

**K_Scratch**

| Feature                  | Contribution |
|--------------------------|--------------|
| Log_Y_Index              | +0.109       |
| X_Minimum                | +0.070       |
| Sum_of_Luminosity        | +0.058       |
| X_Perimeter              | +0.05        |
| TypeOfSteel              | -0.05        |
| Pixels_Areas             | +0.045       |
| X_Maximum                | +0.034       |
| LogOfAreas               | +0.033       |
| SigmoidOfAreas           | +0.032       |

**Pastry**

| Feature                  | Contribution |
|--------------------------|--------------|
| Edges_Y_Index            | +0.091       |
| Empty_Index              | +0.059       |
| Orientation_Index        | +0.054       |
| Outside_X_Index          | +0.051       |
| Y_Perimeter              | +0.046       |
| Length_of_Conveyor       | +0.044       |
| X_Perimeter              | +0.044       |
| Luminosity_Index         | +0.039       |
| Minimum_of_Luminosity    | +0.036       |
| Edges_X_Index            | +0.033       |

**Stains**

| Feature                  | Contribution |
|--------------------------|--------------|
| Length_of_Conveyor       | +0.1         |
| Steel_Plate_Thickness    | +0.089       |
| LogOfAreas               | +0.075       |
| Pixels_Areas             | +0.074       |
| Sum_of_Luminosity        | +0.065       |
| SigmoidOfAreas           | +0.049       |
| TypeOfSteel              | -0.048       |
| Y_Perimeter              | +0.045       |
| Log_Y_Index              | +0.029       |
| X_Maximum                | +0.026       |

**Z_Scratch**

| Feature                  | Contribution |
|--------------------------|--------------|
| Length_of_Conveyor       | -0.101       |
| TypeOfSteel              | +0.014       |
| Empty_Index              | -0.010       |
| X_Minimum                | -0.008       |
| X_Minimum                | -0.007       |
| Outside_X_Index          | -0.007       |
| Steel_Plate_Thickness    | +0.006       |
| Luminosity_Index         | +0.006       |
| Log_Y_Index              | -0.005       |
| Log_X_Index              | -0.004       |
B. Random Forest Rules

In search of greater explanability and interpretability a natural target is to search under the hood of a relatively high precision predictive model as is the case with present dataset and Random Forest (RF) model developed and optimized earlier. Use of variable selection for a minimal and relevant set of conditions from the association rules has been enunciated by (Deng et al. 2014). Therefore exposing of the internal structure of RF model has been carried out in line with the works (Deng et al. 2014) and (Deng 2019).

Rule based models are very good at interpretability across a wide area of applications. Such models base upon a simplified understanding of their internal workings. However one of the major problems with rule generation is the large number of rules besides their coverage and specificity. In simpler terms coverage is the amount and specificity is the length of a rule. Usage of these measures of rule quality is fundamentally dictated by the intention of the user. For the sake of explainability a smaller number of rules with as much coverage as possible is a pre-requisite. A rule having very lengthy antecedent or consequent becomes very difficult to understand and interpret. Therefore transparency and explainability of a model is directly proportional to the coverage and specificity of the rule-base.

In literature a large number of research works may be found regarding production of rules but considerably fewer works are seen for filtering and pruning strategies of the rulebase. This work proposes a novel approach for model building using rules mining: discovering a large number of rules and then focusing in the rulebase using apriori expertise e.g. semantic analysis and the quantitative filtering on the basis of various quality measures of rules.

### TABLE IX
**EXTRACTED RULES FROM OPTIMIZED RANDOM FOREST**

| ID | Rules | Confidence | Count | Lift | Support |
|----|-------|------------|-------|------|---------|
| RF-1 | A IF B THEN C | 0.9999999 | 1999999 | 9.999999 | 0.999999 |
| RF-2 | A IF B THEN C | 0.9999999 | 1999999 | 9.999999 | 0.999999 |
| RF-3 | A IF B THEN C | 0.9999999 | 1999999 | 9.999999 | 0.999999 |
| RF-4 | A IF B THEN C | 0.9999999 | 1999999 | 9.999999 | 0.999999 |
| RF-5 | A IF B THEN C | 0.9999999 | 1999999 | 9.999999 | 0.999999 |

The rules produced by machine are of course mechanistic and may be ‘humanized’ by re-writing them in near human-language.

### C. Association Rules

Association rule mining is now a well established sub-area of machine learning with the additional advantage of being fully transparent from the explainability view point (Grabot 2020).

### Comments on Rules

1. Length of RF-3 is too prohibitive.
2. Rules for K_Scratch and Stains have highest counts and confidence, however at the same time they have smallest lengths. Thus making these rules very attractive insights for human use.

### IV. FINAL MODEL BUILDING

Insights alongwith the extracted rules from RF and associations mining may be incorporated into a data frame for computing implementation. An example may be put here for explaining the rule-classifier ensemble as used in this work: Suppose following 5 rules have been decided to be included in the ensemble:

1. IF V1 is A4v1, V2 is Cv3 THEN R1
2. IF V2 is A4v3, V3 is Cv3 THEN R2
3. IF V1 is Bv1, V1 is A1v1, V3 is A3v3 THEN R3
4. IF V2 is Bv3, V3 is Cv3 THEN R4
5. IF V1 is Cv1, V2 is Bv2, V3 is Bv3 THEN R5

In present case A1v1 etc are the semi-open intervals appearing in rules listings of Table X. Next, these may be re-written as the following table (a data frame):

| V1   | V2   | V3   | Result |
|------|------|------|--------|
| A1v1 | C3v3 | R1   |        |
| A2v3 | C5v2 | R2   |        |
| B1v5 | A4v3 | R3   |        |
| B2v4 | C5v4 | R4   |        |
| C3v1 | B3v2 | R5   |        |

The Directed Acyclic Graph for development of the customized model based upon XAI insights combined with
rules from RF and Association Rules Mining (ARM) is shown in Figure (10):

![Schematic Diagram of High Precision Ensemble Classifier](image)

Fig. 10. Schematic Diagram of High Precision Ensemble Classifier

Performance of the developed model is given as:

| Target        | Prediction |
|---------------|------------|
| Z_Scratch     | 14.2%      |
| Stains        | 14.3%      |
| Pastry        | 13.9%      |
| Stains        | 14.3%      |
| K_Scratch     | 13.9%      |
| Bumps         | 13.9%      |
| Bumps         | 13.9%      |

Same Confusion matrix for our developed classifier may be visualized as follows:

![Confusion Matrix of the Developed Classifier](image)

Fig. 11. Confusion Matrix of the Developed Classifier

V. DISCUSSION AND CONCLUSIONS

This work makes three contributions viz: insights, methodology and a high precision classifier. First of all it obtains valuable-for-humans insights using XAI, secondly it introduces a methodology to incorporate insights from three areas to build a high precision fault classifier. This high precision classifier is the third contribution.

Use of Synthetic Minority Oversampling Technique to balance the multi-class imbalance and the use of Medoids for taking full advantage of the local model agnostic XAI tools like Ceteris Peribus profiles and Breakdown plots are also important methodological innovations of this work.

The considerably improved performance of the classifier, in our view, is the result of few basic factors of this research. Idea of using various class balancing techniques for the problem of steel plate faults dataset was originally proposed by (Tian, Fu, and Wu 2015) in their conclusion of the paper as a future direction. However, this they proposed while ignoring all the 673 cases of Common_Other type of faults.

In literature one finds a considerable number of synthetic data generation and class balancing techniques (Laopez et al. 2013).

A suitable choice of prototypical cases (Gurumoorthy et al. 2019) have been demonstrated to be useful for the purpose of explainability. Our present work makes a suitable choice of the representative, equivalently central and prototypical, cases as medoids. This choice of medoids allows for peaking into the global behaviour of the trained model while using a localized technique, for example, Ceteris Peribus.

Representing collection of rules as a data frame has been in practice for various practical use cases e.g. illustrative example in (Deng 2019). Present work makes some changes for its own usage to cobble various If-Then rules obtained from XAI Insights, Random Forest and Association Mining.

Insights help to do Science (and Engineering, of course) ‘by’ Data. The insights obtained in this work, at the very first place, are themselves directly useful for the production engineers. These insights provide for sharpening of the skill of humans to detect a fault and its relevant factors/variables.

Usually XAI is used to develop white-box models for human consumption. However in this work XAI has been used in a less trodden manner i.e. getting insights from XAI and incorporating them back into rule-based ensemble development. These insights have been re-entered into the loop
of model building which has clearly benefited the new model by improving its performance by almost 7%.

This approach has also benefited from the wholistic view developed by Baniecki et. al. (Baniecki and Biecek 2020). Techniques labeled as 1-4 were applied from Figure 2 (reproduced below) of Baniecki paper.

Certain areas remained unaddressed as well. There are other than SMOTE techniques available for oversampling and balancing of data (Laopez et al. 2013). These techniques should also be checked for even higher accuracy of the developed classifier. Another approach may be to develop a surrogate model for obtaining even more transparency. Although in view of an Optimized Random Forest and its extracted rules this appears not too useful, but the quality measures of RF rules indicates a possible direction of improvement.

REFERENCES

Baniecki, Hubert, and Przemyslaw Biecek. 2020. “The Grammar of Interactive Explanatory Model Analysis.” arXiv:2005.00497 [Cs, Stat], May. http://arxiv.org/abs/2005.00497.

Barredo Arrieta, Alejandro, Natalia Daz-Rodriguez, Javier Del Ser, Adrien Bennetot, Sitham Tabik, Alberto Barbado, Salvador Garcia, et al. 2020. “Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges Toward Responsible AI.” Information Fusion 58 (June): 82–115. https://doi.org/10.1016/j.inffus.2019.12.012.

Chawla, N. V., K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. 2002. “SMOTE: Synthetic Minority over-Sampling Technique.” Journal of Artificial Intelligence Research 16 (June): 321–57. https://doi.org/10.1613/jair.953.

Christina B. Azodi, Jiliang Tang, and Shin-Han Shiu. 2020. “Opening the Black Box: Interpretable Machine Learning for Geneticists.” https://doi.org/10.1016/j.tig.2020.03.005.

Consoli, Sergio, Valentina Presuti, Diego Reforgiato Recupero, Andrea G. Nuzzolese, Silvio Peroni, Misuel Mongiovì, and Aldo Gangemi. 2017. “Producing Linked Data for Smart Cities: The Case of Catania.” Big Data Research 7 (March): 1–15. https://doi.org/10.1016/j.bdr.2016.10.001.

Deng, Houtao. 2019. “Interpreting Tree Ensembles with inTrees.” International Journal of Data Science and Analytics 7 (4): 277–87. https://doi.org/10.1007/s41060-018-0144-8

Deng, Houtao, George Runger, Eugene Tuv, and Wade Bannister. 2014. “CBC: An Associative Classifier with a Small Number of Rules.” Decision Support Systems 59 (March): 163–70. https://doi.org/10.1016/j.dss.2013.11.004

De Silva, Daswin, Seppo Sierla, Damminda Alahakoon, Evgeny Osipov, Xinghao Yu, and Valeriy Vyatkin. 2020. “Toward Intelligent Industrial Informatics: A Review of Current Developments and Future Directions of Artificial Intelligence in Industrial Applications.” IEEE Industrial Electronics Magazine 14 (2): 57–72. https://doi.org/10.1109/MIE.2019.2952165

E.L. Russell, L.H.Chiang, and R.D.Braatz. 2000. “Fault Detection in Industrial Processes Using Canonical Variate Analysis and Dynamic Principal Component Analysis” 51 (1): 81–93.

Friedman, Jerome H. 2001. “Greedy Function Approximation: A Gradient Boosting Machine.” Annals of Statistics 29 (5): 1189–1232. https://doi.org/10.1214/aos/1013204511.

Gabriel Schwartz, and Ko Nishino. 2020. “Recognizing Material Properties from Images,” August. https://doi.org/10.1109/TPAMI.2019.2907850.

Goldstein, Alex, Adam Kaplan, Justin Bleich, and Emil Pitkin. 2015. “Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation.” Journal of Computational and Graphical Statistics 24 (1): 44–65. https://doi.org/10.1080/10618600.2014.907095.

Grabot, Bernard. 2020. “Rule Mining in Maintenance: Analysing Large Knowledge Bases.” Computers & Industrial Engineering 139 (January): 105501. https://doi.org/10.1016/j.cie.2018.11.011.

Gurumoorthy, Karthik S., Amit Dhurandhar, Guillermo Cecchi, and Charu Aggarwal. 2019. “Efficient Data Representation by Selecting Prototypes with Importance Weights.” arXiv:1707.01212 [Cs, Stat], August. http://arxiv.org/abs/1707.01212.

Kang, Seokho. 2020. “Model Validation Failure in Class Imbalance Problems.” Expert Systems with Applications 146 (May): 113190. https://doi.org/10.1016/j.eswa.2020.113190.

Kazemi, Mohammad Ali Afshar, Sima Hajian, and Neda Kiani. 2018. “Quality Control and Classification of Steel Plates Faults Using Data Mining.” Applied Mathematics & Information Sciences Letters 6 (2): 59–67. https://doi.org/10.18576/amisl/060202.

Kharal, Athar, and Ayman Saleem. 2012. “Neural Networks Based Airfoil Generation for a Given Cp Using BezierPARSEC Parameterization.” Aerospace Science and Technology 23 (1): 330–44. https://doi.org/10.1016/j.ast.2011.08.010.

Laopez, Victoria, Alberto Fernandez, Salvador Garcia, Vasilje Palade, and Francisco Herrera. 2013. “An Insight into Classification with Imbalanced Data: Empirical Results and Current Trends on Using Data Intrinsic Characteristics.” Information Sciences 250 (November): 113–41. https://doi.org/10.1016/j.ins.2013.07.007.
