International Journal of Material Forming (2023) 16:70
https://doi.org/10.1007/s12289-023-01791-y

DEVELOPMENTS IN MODELLING AND SIMULATION..JAPAN, SOUTH KOREA AND CHINA

Formability classifier for a TV back panel part with machine learning

Piemaan Fazily[1] · Donghyuk Cho[1] · Hyunsung Choi[2] · Joon Ho Cho[3] · Jongshin Lee[3] · Jeong Whan Yoon[1,4]

Received: 17 June 2023 / Accepted: 30 September 2023 / Published online: 2 November 2023

Abstract
This study proposes a machine learning-based methodology for evaluating the formability of sheet metals. An XGBoost (eXtreme Gradient Boosting) machine learning classifier is developed to classify the formability of the TV back panel based on the forming limit curve (FLC). The input to the XGBoost model is the blank thickness and cross-sectional dimensions of the screw holes, AC (Alternating Current), and AV (Audio Visual) terminals on the TV back panel. The training dataset is generated using finite element simulations and verified through experimental strain measurements. The trained classification model maps the panel geometry to one of three formability classes: safe, marginal, and cracked. Strain values below the FLC are classified as safe, those within 5% margin of the FLC are classified as marginal, and those above are classified as cracked. The statistical accuracy and performance of the classifier are quantified using the confusion matrix and multiclass Receiver Operating Characteristic (ROC) curve, respectively. Furthermore, in order to demonstrate the practical viability of the proposed methodology, the punch radius of the screw holes is optimized using Brent's method in a Java environment. Remarkably, the optimization process is completed swiftly, taking only 3.11 s. Hence, the results demonstrate that formability for a new design can be improved based on the predictions of the machine learning model.

Keywords Artificial Intelligence · Forming Limit Diagram · XGBoost algorithm · Brent's method · Finite Element Analysis

Introduction

Supervised machine learning (ML) involves mapping input features to their associated output labels. In classification problems, the output label belongs to a category, which may be binary or multi-class [1]. "Boosting" [2], among classification algorithms, has been identified as excelling in standard classification benchmarks [3]. Boosting involves an ensemble of multiple base learners combined to form a strong prediction model. Therefore, boosting methods have proven to be accurate in various applications, ranging from open competitions such as the Netflix prize [4] and ad clicks at Facebook [5] to more recent applications in sheet metal forming [6].

The literature survey also indicates the application of machine learning approaches [7, 8], including neural networks [9, 10], to predict forming limit. The forming limit diagram (FLD), developed by Keeler [11] and improved by Goodwin [12], is widely used as an evaluation index for predicting the formability of sheet metals. The regions below the forming limit curve are classified as safe, while the above indicates necking or potential deformation instability. Therefore, predicting forming limits in sheet metal is well-suited to the application of supervised classification models.

Concerning the characterization of the forming limit curve on the FLD, experimental [12–17] and theoretical methodologies, including those based on bifurcation theory [18–20], geometrical imperfection theory [13, 21], and continuum damage mechanics [22], have been developed. Zhang et al. [23] provide

✉ Jeong Whan Yoon
j.yoon@kaist.ac.kr

1 Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 34141, Republic of Korea

2 Korea Institute of Materials Science, 797 Changwon-daero, Changwon 051508, Republic of Korea

3 Metal Forming Technology Team, LG Electronics Inc, LG-ro, Jinwi-myeon, Pyeongtaek-si, Gyeonggi-do, Republic of Korea

4 School of Engineering, Deakin University, 75 Pigdons Rd., Wauran Ponds, Burwood, VIC 3216, Australia

a comprehensive review of the theoretical models used for the formability prediction of sheet metals. The FLD offers accurate assessments of formability in single-stage forming processes, considering proportional loading and linear strain path. Otherwise, forming-limit stress diagram (FLSD) may be utilized [24, 25].

Assuming accurate prediction of the formability limit, the next logical question is how to improve the formability of sheet metals. To address this question, Song et al. [26] proposed a feasible preform design method based on deformation history to enhance the formability of tube hydroforming. Similarly, Ko et al. [27] suggested a feasible formability diagram for selecting process variables, such as the initial blank shape, the blank holding force, and the shape of the draw bead, for stamping the turret suspension. On the other end of the spectrum, Attanasio et al. [28] reported on the feasibility and formability of parts in incremental forming. Additionally, Kim et al. [29] demonstrated the effectiveness of asymmetric rolling of strip-cast sheet for producing high-formability Al alloy sheets.

Concerning the application of machine learning for formability improvement, Zimmerling et al. [30] presented an image-based recognition technique to assess and improve the formability of continuous fiber-reinforced plastics (CoFRP). Bae et al. [31] developed a machine-learning model that uses the XGBoost algorithm to predict processing parameters that ensure high formability for Ti alloys. Lu et al. [32] proposed a gradient-boosting regression to optimize the multiple variable blank holder force, which improves the formability in aluminum hot stamping. Marques et al. [33] compared the accuracy of various machine learning algorithms, such as multilayer perceptron, Gaussian processes, support vector regression, and random forest, in predicting fracture strain in hole expansion tests. The study concluded that Gaussian processes and support vector regression outperformed others, enabling robust predictive models. Similarly, Singh et al. [34] developed a machine learning framework using HDR imaging to automate defect detection in sheet metal 'Nakajima' samples that were pressed with an industrial stamping press.

The primary objective of this work is to develop a machine learning-based classifier and corresponding optimization program to evaluate the formability of the TV back panel. The proposed methodology introduces a novel approach that predicts whether a change in the geometry of the TV back panel is safe, marginal, or likely to cause cracking. Moreover, the approach provides the user with optimized cross-sectional dimensions of the panel to ensure that the new product is formable.

Section "Materials and methods" briefly overviews the TV back panel and its corresponding cross-sectional geometries in the screw holes, AC (Alternating Current), and AV (Audio Visual) terminal regions. The finite element setup and corresponding experimental verification are explained in Sections "Finite element analysis and Strain measurement system", respectively, while Section "Dataset generation" discusses the data generation process for the machine learning model. Section "Accuracy and performance metrics" establishes the accuracy and performance metrics for the multi-class classification, while Section "Baseline classifier" evaluates a baseline classifier and investigates the improvement in the tree model by boosting. In Section "Training the XGBoost", the XGBoost machine learning model [35] is trained on the screw hole, AC, and AV terminal datasets. After training, the XGBoost classifier is exported and implemented in the JAVA environment. Section "Optimization process" discusses the optimization process based on the proposed XGBoost classifier. Additionally, the punch radius of the screw holes is optimized using Brent's method to demonstrate the improvement in the formability of the TV back panel. Figure 1 illustrates the XGBoost architecture for the proposed framework. $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ denotes the training dataset, obtained through finite element analysis, while $\hat{y}_i$ is the forming probability estimate by the XGBoost classifier. XGBoost employs Newton tree boosting to fit additive tree models. A brief overview of the additive tree models and XGBoost method is presented in Appendix 1.

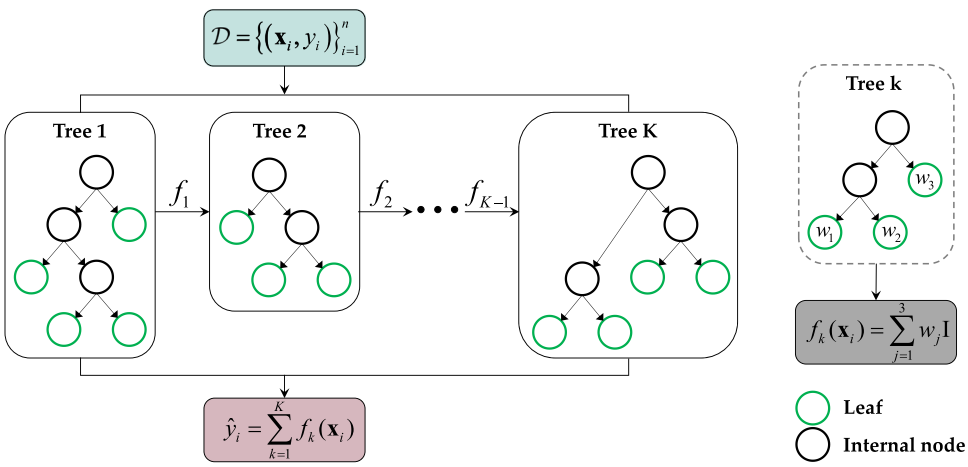


Fig. 1 Illustration of XGBoost architecture for the proposed framework

Materials and methods

The proposed XGBoost classifier is a supervised machine learning approach that predicts the formability of the TV back panel based on the forming limit curve (FLC). The trained classification model maps the panel geometry to one of three formability classes: safe, marginal, and cracked. Strain values below the FLC are classified as safe, within 5% margin of the FLC as marginal, and above as crack. Figure 2 illustrates the TV back panel and the respective screw holes, AC (Alternating Current), and AV (Audio Visual) terminals.

The input to the classification model is the cross-sectional dimensions of the screw holes, AC, and AV terminals on the TV back panel. The orthographic projection and cross-section of the screw holes, AC, and AV terminals are shown in Fig. 3. It can be noted that AC and AV share the same cross-sectional geometry. The output to the classification model is the formability classes based on the strain values on the forming limit diagram obtained through the finite element simulations.

Subsequently, finite element analysis was carried out in PAM-STAMP and Anaconda (python distribution) is used to construct the XGBoost classifier.

FEA of panel forming

Generally, finite element analysis is conducted to evaluate the material's formability when subjected to changes in material dimensions. This provides the limits of the two principal strains, ε_1, and ε_2, represented on a forming limit diagram. However, the procedure follows an error-and-trial approach that is inefficient and time-consuming. An

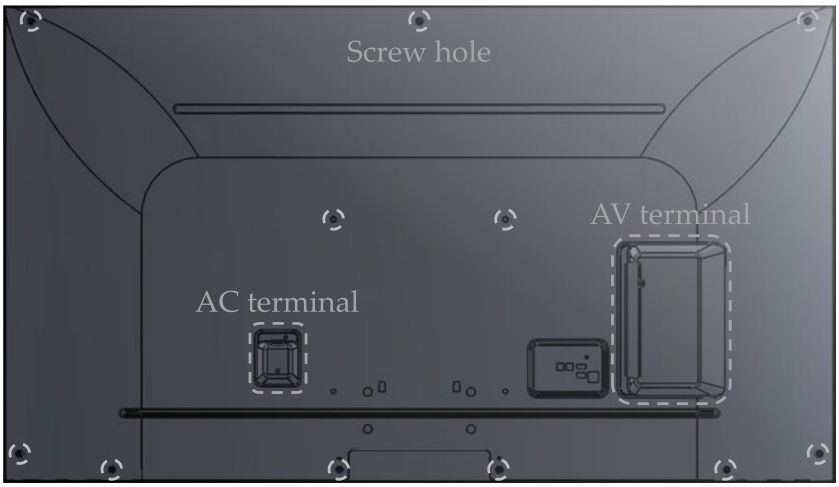


Fig. 2 Illustration of TV back panel and screw holes, AC, and AV terminals

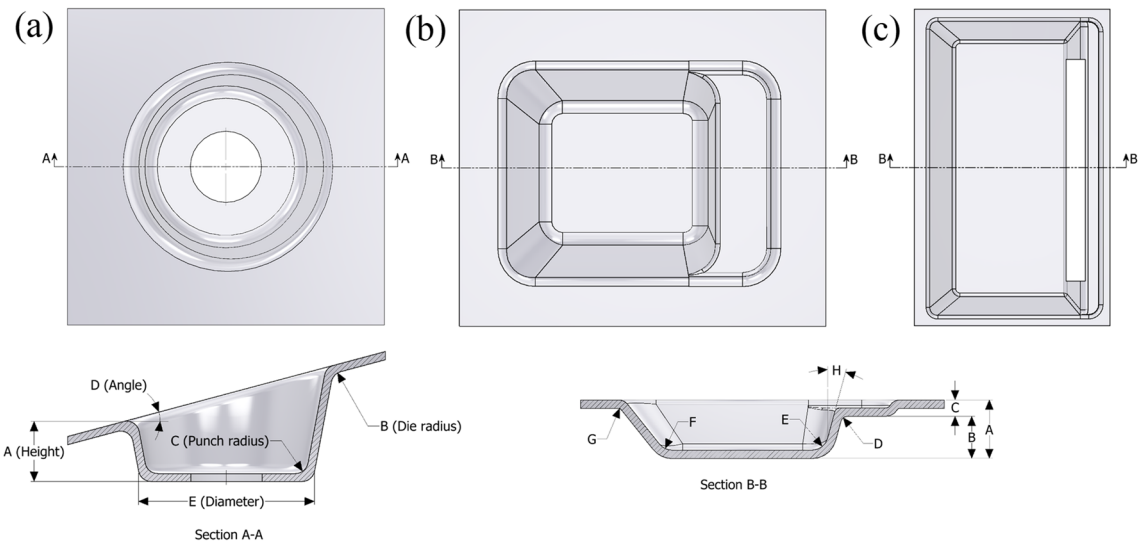


Fig. 3 Orthographic projections and sectional view (**a**) Screw hole (**b**) AC terminal (**c**) AV terminal

alternative solution is to use a machine learning (ML) model to assess formability in terms of probability estimates. Comparatively, the machine learning approach provides faster solutions corresponding to the changing inputs.

Therefore, a machine learning classifier is employed in this study to classify the formability of the TV back panel, and the training data is generated numerically from finite element simulations. Section "Finite element analysis" details the FE setup and analysis, while Section "Strain measurement system" discusses the validation of FE simulations through experimental strain measurements. Finally, Section "Dataset generation" concludes the feature selection and data generation phase.

Finite element modeling

The commercial FEM package, PAM-STAMP, with the explicit solver, was used for forming the TV back panel. The tool, i.e., punch, die, and blank holder, was modeled as discrete rigid, while the blank was assumed to be deformable elastic–plastic material. The friction coefficient between the tool and blank was set to 0.16. The stamping velocity was set to 5 mm/ms, while blank holding force was 40 tons. Schematic of the finite element model used for the simulations is shown in Fig. 4.

The blank workpiece was meshed using Belytschko-Tsay membrane elements, resulting in a final mesh with 309,465 elements, 323,480 nodes, and 5 through-thickness integration points. The material data is Ultra-low carbon steel, obtained experimentally through uniaxial tensile tests performed on ASTM E8/E8M [36] specimens cut at 0°, 15°, 30°, 45°, 60°, 75°, and 90° with respect to the rolling direction. The material data along the rolling direction is presented in Table 1. The hardening curve used during finite element simulation was obtained by calibrating the Swift law using the flow stress curve along the rolling direction such that,

$$\overline{\sigma} = 513.73\left(0.00853 + \overline{\varepsilon}_p\right)^{0.241}[MPa] \quad (1)$$

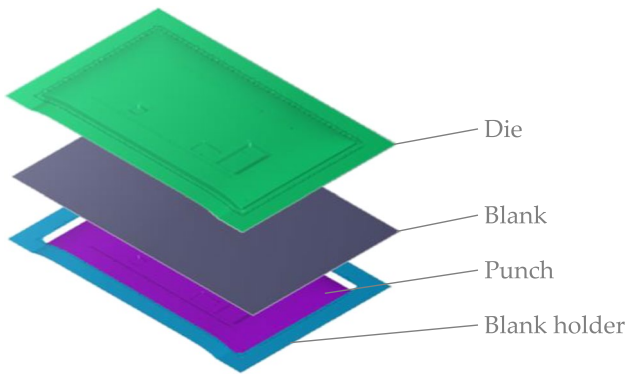


Fig. 4 Schematic of the finite element model for forming the TV back panel

Table 1 Material data for Ultra-low carbon steel along the rolling direction

Yield strength (MPa)	Ultimate strength (MPa)	Elongation (%)	Uniform Elongation (%)	Elastic modulus (GPa)
162.6	290.3	44.44	25	160.06

Furthermore, biaxial tension test was conducted on a cruciform specimen, proposed by Kuwabara et al. [37]. A comprehensive overview of the testing system was provided by Wu et al. [38]. Table 2 presents the corresponding normalized yield stress and r-values. Therefore, given the eight material parameters, the anisotropic coefficients for the yld2000-2d criterion [39] were identified using the PAM-STAMP add-on, ESI Mat-Wizard. Subsequently, finite element analysis was carried out using the with different instances of sheet thickness while varying the cross-sectional dimensions of the screw holes, AC, and AV terminals. Figure 5 shows the major and minor plastic strain contours of one of the instances.

Strain measurement system

To assess the surface strains on the stamped TV back panel, grid marking and a 3D automated strain measurement system are employed [40]. Figure 6 shows the stamped TV back panel and identifies the regions of interest for strain measurements.

Initially, a square grid pattern is produced on the undeformed blank using electrochemical etching. The blank is then stamped to form the back panel. Following that, the strain measurement system calculates the surface strains using digital image correlation based on the 3D coordinates of the deformed grids. Figure 7 provides a schematic representation of the strain measurement procedure.

To obtain the 3D coordinates of the deformed grids, two or more views of a region on the back panel are photographed. Meanwhile, a photogrammetric dice-shaped target of known dimensions is placed next to the region being measured. This arrangement enables the strain measurement system to automatically compute the camera position, lens focal length, and the angle between offset views. Finally, a common grid square is identified in the captured images and combined to generate the 3D map of the surface area. The surface geometry of the AV terminal with the reference target obtained using the strain measurement system is shown in Fig. 8.

Furthermore, the captured images are post-processed to compute strain values along a section. The major and minor strain values for the AV terminal, based on the experimental measurements and finite element simulations, are shown in Fig. 9. While the FEM analysis was carried out using the Yld2000-2d criterion, with the anisotropic coefficients listed in Table 3. It is observed from the strain contours that the

Table 2 Normalized yield stress and r-values for Ultra-low carbon steel

σ_0/σ_0	σ_{45}/σ_0	σ_{90}/σ_0	σ_b/σ_0	r_0	r_{45}	r_{90}	r_b
1.0000	1.0305	1.0309	1.0534	1.753	1.811	2.180	0.771

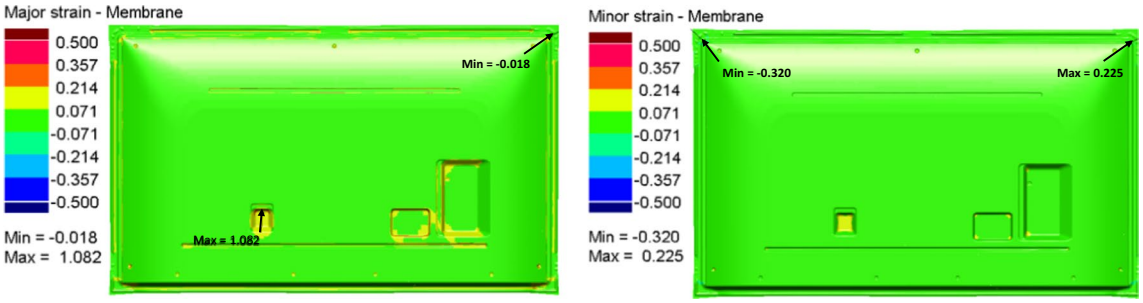


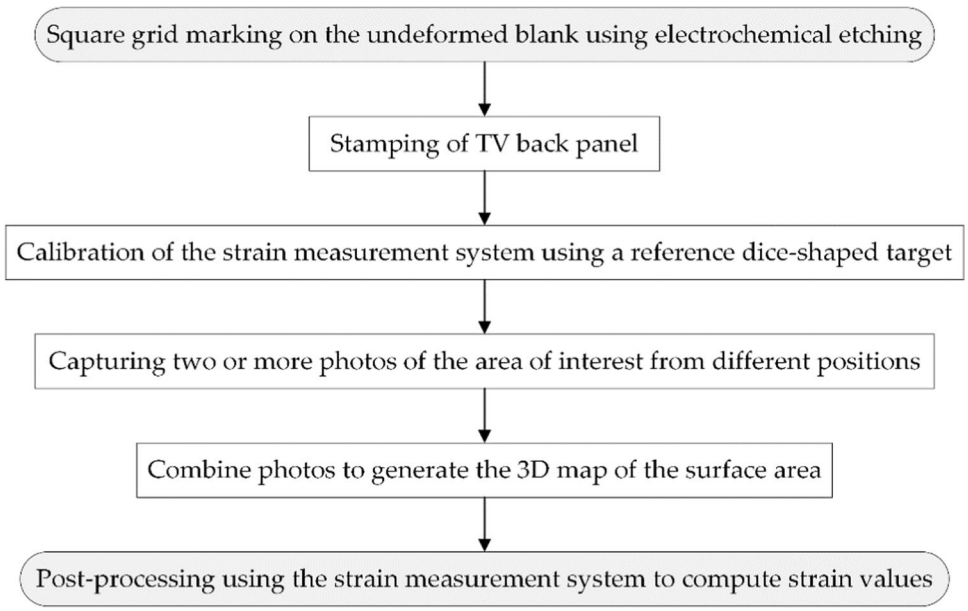
Fig. 5 Major and minor strain contours of the TV back panel

Fig. 6 Stamped TV back panel obtained experimentally. Regions (AV1 ~ AV4) of the AV terminal are highlighted to demonstrate the strain measurement system (as in Figs. 8 and 9)

FEM analysis agrees with the experimental measurements both in tendency and amplitude.

In addition, to establish the accuracy of the finite element simulations, the draw-in profile was measured experimentally and compared with that obtained from FEM analysis. Figure 10 shows the draw-in profile of the TV back panel, quantified by the dimensions at the top, bottom, left, and right sides. The comparison of experimental and simulation results is shown in Figs. 11 and 12. In the drawing simulation, the bead gap (clearance) was uniformly set at 1.1 times the sheet thickness for all sides. However, in practice, due to the limitations in manufacturing precision, the bead gap in the experimental die had non-uniform clearance in some sections, resulting in minor draw-in deviations, as indicated in Fig. 11. However, it can be observed that the predicted draw-in profile shows good

Fig. 7 Schematic representation of the grid marking and strain measurement procedure



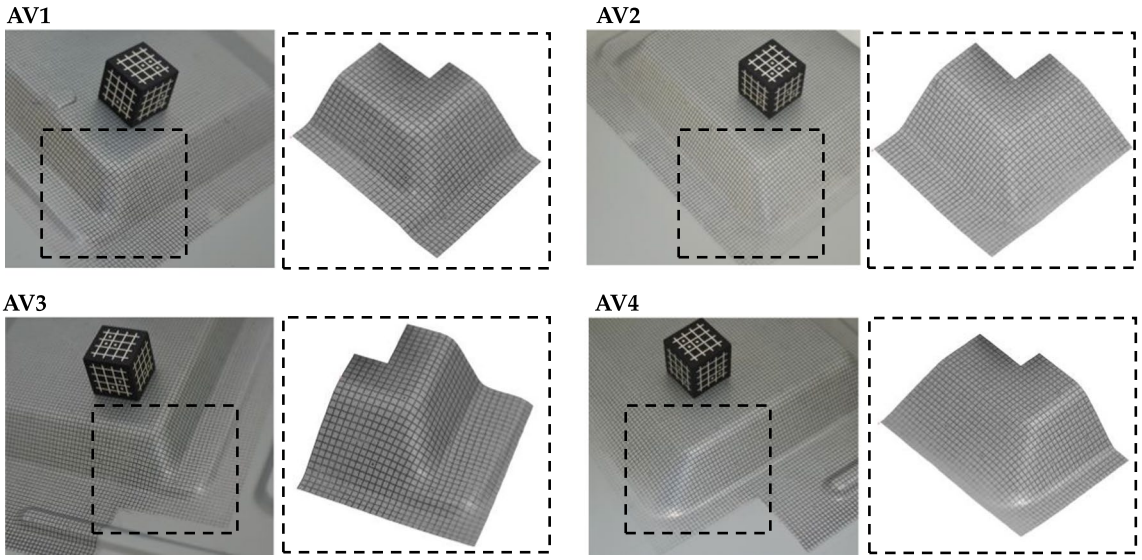


Fig. 8 Surface geometry of AV terminal obtained using the 3D strain measurement system

agreement with the experiment in overall trend, as further seen in Fig. 12. Therefore, it is appropriate to prepare the dataset of the classification model based on finite element simulation.

Dataset generation

The data generation phase aims to create a comprehensive training dataset enabling the classification model to accurately predict the formability of the TV back panel. During the data generation phase, 630 finite element simulations were carried out with combinations of classification inputs, each belonging to a specific design domain. The design domain is user-defined and corresponds to the geometry of the TV back panel.

In this particular application, the input to the classification model is the blank thickness and cross-sectional dimensions of the screw holes, AC, and AV terminals on the TV back panel. For the thickness variation of the Ultra-low carbon steel blank, four cases were considered, ranging from 0.4, 0.5, 0.6, and 0.8 [mm]. Additionally, the variation in cross-sectional dimensions of the screw holes, AC, and AV terminals corresponding to the Fig. 3 are presented in Table 4.

The output of the classification model is the prediction of formability classes based on the strain values obtained from the forming limit diagram through finite element simulations. These classes are categorized into five zones, and for this study, three classes have been selected as the output for the machine learning classifier. The classification model outputs probability estimates for each class and makes classifications based on the highest probability value.

The training output labels of one of the simulation instances., i.e., safe, marginal, and crack classes, are shown in Fig. 13. It can be noted that strain values below the FLC are classified as safe, within 5% margin of the FLC as marginal, and above as crack.

ML model for panel forming

In this study, the XGBoost classifier is employed as the machine learning model to evaluate whether a new change in the geometry of the TV back panel is safe, marginal, or likely to cause cracking. The XGBoost classifier outputs probability estimates for each class and makes classifications based on the highest probability value.

Section "Accuracy and performance metrics" establishes the accuracy and performance metrics for the multi-class classification. Section "Baseline classifier" investigates a baseline classifier and assesses the improvement in the tree model by boosting. The performance of vanilla XGBoost is evaluated, and hyperparameter tuning is carried out based on the grid search strategy. Finally, Section "Training the XGBoost" discusses the training of the XGBoost on the screw hole, AC, and AV terminal datasets.

Accuracy and performance metrics

The XGBoost follows a supervised learning approach, sequentially adding trees to minimize the residual errors of the previous trees during training. As the training proceeds, the XGBoost assigns weights to new leaves of a tree and adjusts the tree structure by pruning out the internal nodes according to a boosting rule and design variables. The design variables include the learning rate, number of rounds (trees), loss function, evaluation metric, regularization, and stopping criteria. Henceforth, the XGBoost training is terminated after explicit stopping criteria are met.

For quantifying the statistical accuracy of classification, the confusion matrix C_{ij} and Matthews correlation coefficient

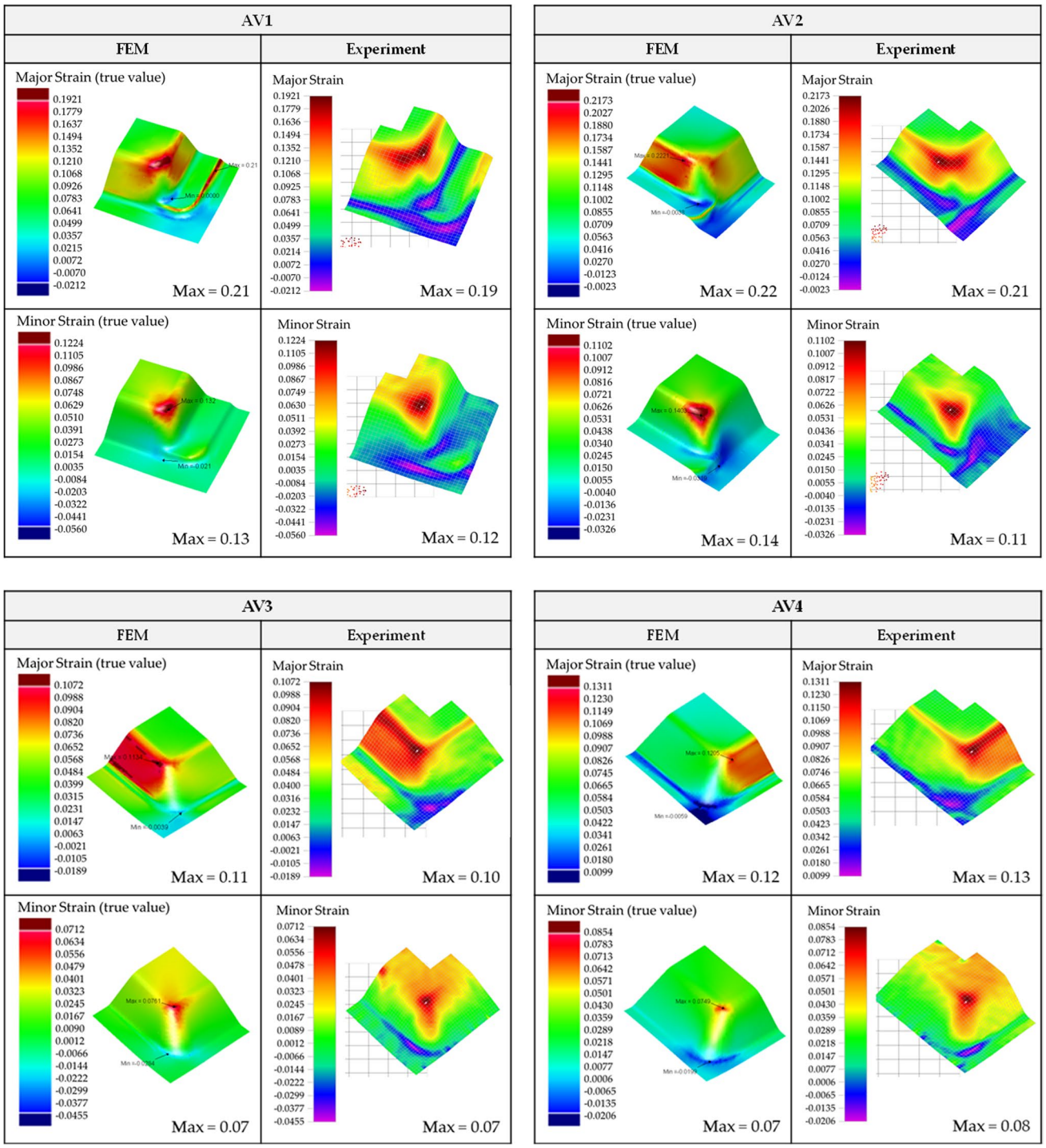


Fig. 9 Comparison of major and minor strains for the AV terminal

Table 3 Anisotropic coefficients of Yld2000-2d for Ultra-low carbon steel

α_1	α_2	α_3	α_4	α_5	α_6	α_7	α_8	a
1.0821	0.9870	0.9383	0.9131	0.9529	1.0147	1.0159	0.8728	6

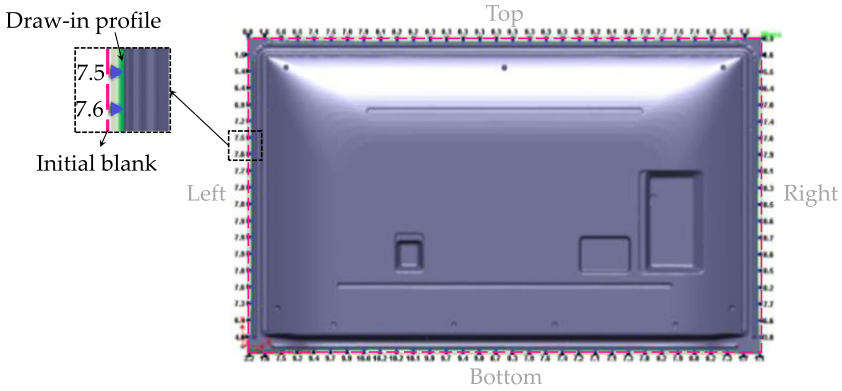


Fig. 10 Measurement of draw-in profile during forming of the TV back panel

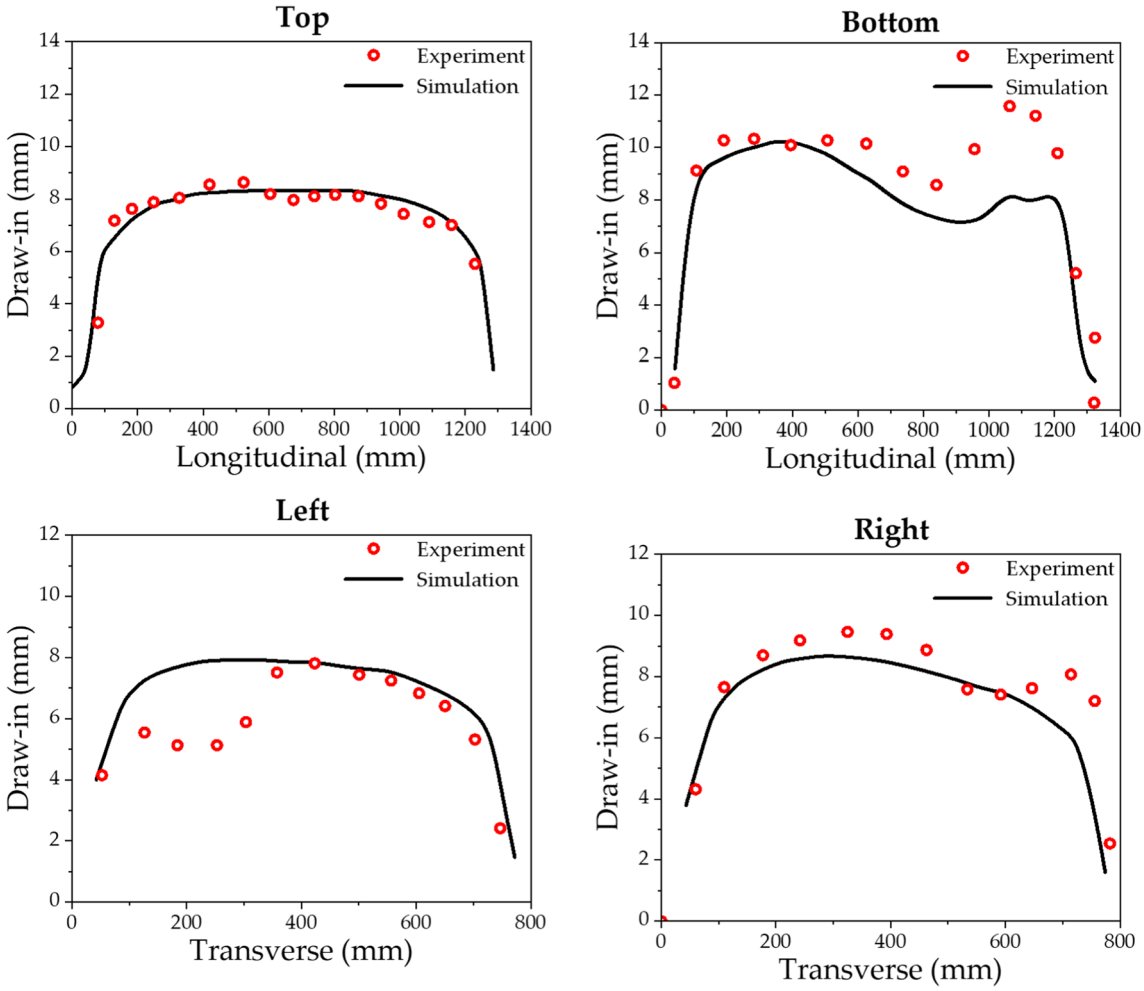


Fig. 11 Comparison of the draw-in profiles for the locations in Fig. 10

(MCC) [41] were chosen as accuracy metrics. Jurman et al. [42] reported MCC as an optimal off-the-shelf accuracy metric of a classifier in binary and multiclass problems. Hence, for multi-class classification with K total classes, as such in the present case being safe, crack, and margin classes, MCC is given as follows,

$$MCC = \frac{c \times s - \sum_{k}^{K} p_k \times t_k}{\sqrt{\left(s^2 - \sum_{k}^{K} p_k^2\right)\left(s^2 - \sum_{k}^{K} t_k^2\right)}} \tag{2}$$

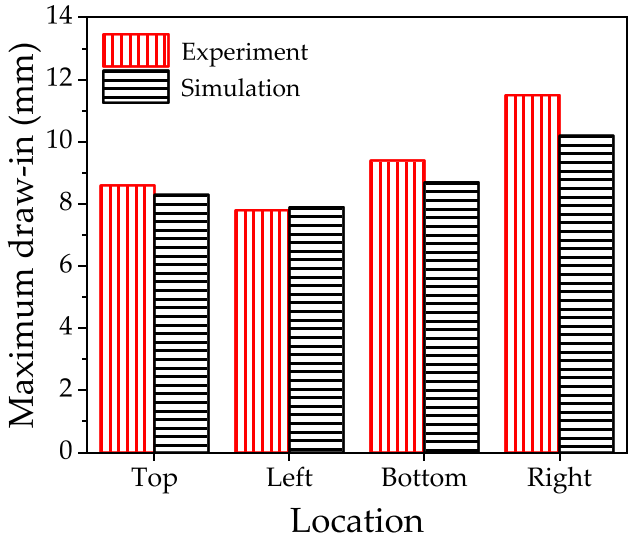


Fig. 12 Comparison of maximum draw-in during forming of the TV back panel

$$c = \sum_k^K C_{kk};\ s = \sum_i^K \sum_j^K C_{ij};\ t_k = \sum_i^K C_{ik};\ p_k = \sum_i^K C_{ki}$$

For quantifying the performance of the classifier to distinguish between the safe, crack, and margin classes, a multiclass (Receiver Operating Characteristic) ROC curve is chosen. The area under the ROC curve, generally termed the ROC-AUC score, measures the ability of a multiclass classifier to distinguish a given class compared to the remaining classes (One-vs-Rest). Hence, for multiclass classification, the ROC curve is plotted between the true positive rate (TPR) against the false positive rate (FPR) for each class against all the others; as such,

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}};\text{TPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{3}$$

where

TP : true positive;FP : false positive;
TN : true negative;FN : false negative.

Table 4 Training input labels for the classification model

Screw hole	*A [mm]*	*B [mm]*	*C [mm]*	*D [mm]*	*E [deg]*	
	[0–2.99]	[1–3.2]	[0.4–2]	[0–37]	[9-10]	
AC terminal	*A [mm]*	*C [mm]*	*D [mm]*	*E [mm]*	*G [mm]*	*H [deg]*
	[16–18]	[1.7–3]	[3–8.49]	[8.001–12]	[5-7]	[18.16–26]
AV terminal	*A [mm]*	*C [mm]*	*D [mm]*	*E [mm]*	*G [mm]*	*H [deg]*
	[15.3–28.1]	[0–12.5]	[3.5–8]	[3–10]	[3.22–9]	[55.2–65]

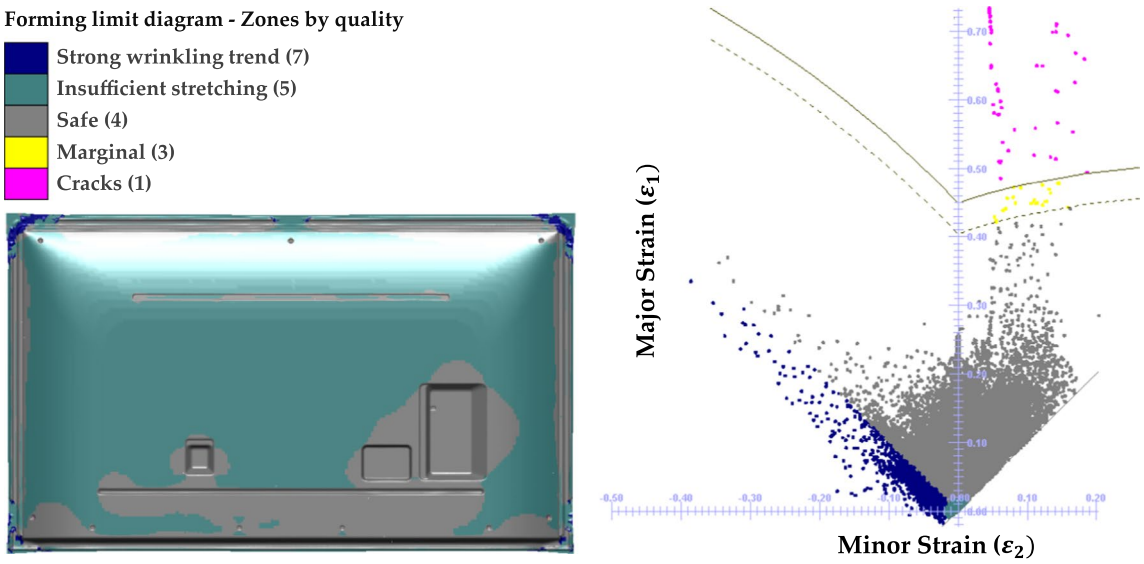


Fig. 13 Training output labels, safe, marginal, and crack classes for the classification model

where,

Consider the example of Safe-vs-Non-Safe formability, where Non-safe includes both the margin and crack classes. In this case, TP represents instances where the classifier correctly predicts safe formability, while FN indicates the instances where the classifier underestimates the formability, predicting non-safe instead of safe. Similarly, TN represents the instances where the classifier correctly predicts non-safe formability, while FP indicates instances where the classifier overestimates the formability, predicting safe instead of non-safe.

Baseline classifier

The decision tree (DT) classifier is utilized as the baseline classifier. The specified classifier is tested with a dataset generated using the finite element method for the screw holes of the TV back panel. The training dataset is divided into two sets: training and test sets. The training set comprises 80% of the total samples, while the test set contain 20%. Therefore, the training set size is 504, and the test set is 126. The training is carried out with the default parameters provided by the scikit-learn library [43]. A constant seed value is used to ensure reproducibility, providing the same confusion matrix during the re-run. Figure 14 demonstrates the normalized confusion matrix for the training and test sets predicted by the DT classifier. The entries C_{ii} on the main diagonal of the confusion matrix establish the accuracy of the predictions, while the off-diagonal entries represent misclassifications.

It can be noted that the DT classifier is a weak learner, as it poorly predicts the margin class in both the training and testing sets. Therefore, to assess the improvement in the tree model by boosting, the performance of vanilla XGBoost is evaluated for prediction accuracy and compared with the DT classifier. Figure 15 shows the normalized confusion matrix for the training and test sets obtained using the vanilla XGBoost classifier.

Although the vanilla XGBoost outperformed the DT classifier, a low prediction of 0.75 for the margin class in the test set indicates poor generalization and overfitting. Therefore, hyperparameter tuning for XGBoost is carried out using grid search. Grid search exhaustively considers all provided parameter combinations for tuning the hyperparameters of the XGBoost. The search range and the optimal hyperparameters for the XGBoost classifier based on the screw hole dataset is reported in Table 5.

In addition, an early stopping criterion is adopted for the training phase to avoid overfitting. The test set is utilized for validation, and the loss metric is set to the cross-entropy loss function [44]. Hence, the training is stopped if the value of validation cross-entropy loss stagnates or increases over 10 consecutive epochs.

For multi-class classification with K total classes, let Y be a matrix with true labels encoding 1-of-K binary indicator and P be a matrix of probability estimates; the cross-entropy loss is given as follows,

$$\mathcal{L}(Y,P) = -\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{K-1} y_{ik} ln\, p_{ik} \tag{4}$$

where N is the total number of instances, y_{ik} is the binary indicator with the expected labels, and p_{ik} is the predicted probability of instance i belonging to class k.

Training the XGBoost

The XGBoost classifier with tuned parameters is trained individually on all distinct datasets, and the cross-entropy loss values are logged during the training process. For the

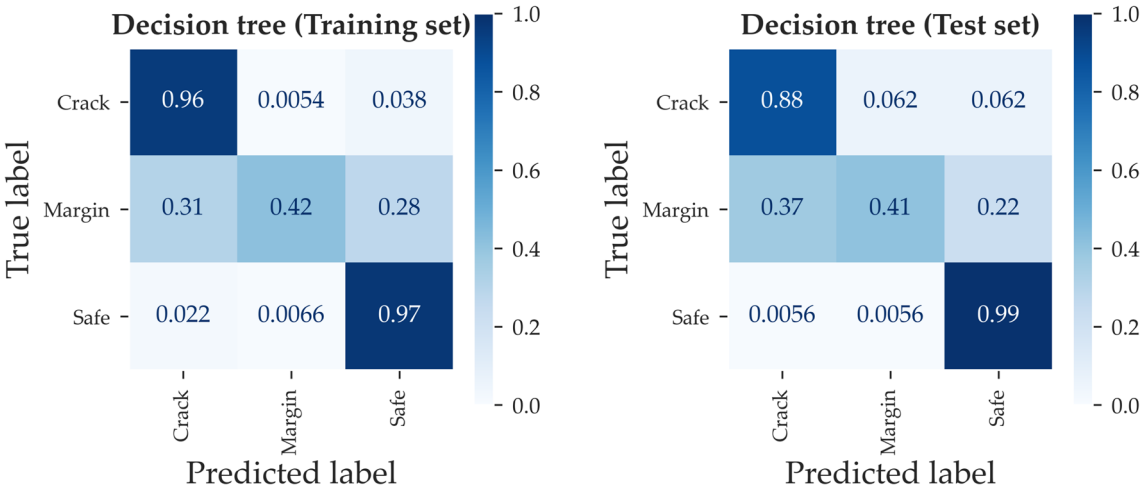


Fig. 14 Normalized confusion matrix predicted by the decision tree for the screw holes

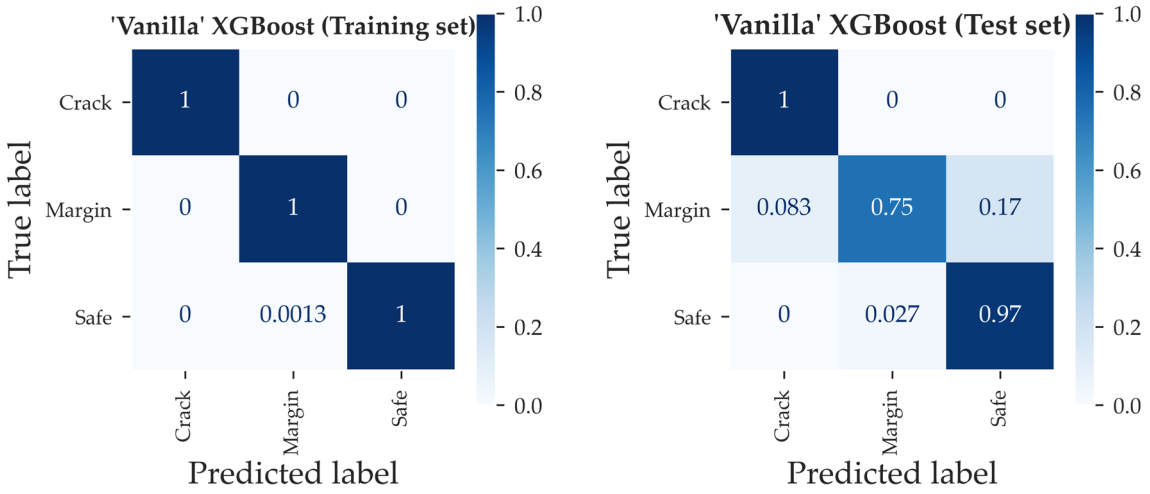


Fig. 15 Normalized confusion matrix predicted by the vanilla XGBoost for the screw holes

Table 5 Grid search reporting the hyperparameters for the XGBoost classifier

	Learning rate	*gamma*	*Maximum tree depth*	*Subsample ratio of training instances*	*Subsample ratio of columns by tree*
Search range	[0.01–0.1]	[0–0.5]	[2–6]	[0.7–1.0]	[0.3–1.0]
Hyperparameters	0.106	0.047	5	0.756	0.835

screw hole dataset, the XGBoost classifier is trained using the training set of 504 and validated using the test set of 126. The training process is stopped if the validation loss remains stagnant or increases over 10 consecutive epochs. The evolution of training and validation losses for datasets based on screw holes is shown in Fig. 16.

Figure 17 demonstrates the normalized confusion matrix for the training and test sets predicted by the XGBoost classifier based on the grid search and early stopping criterion. A value of 1.0 on the main diagonal of the confusion matrix indicates that all instances of a particular class have been correctly classified. Thereafter, the Matthews correlation coefficient (MCC) for the test set are calculated and are reported in Table 6. MCC outputs values in the interval [-1,1], with a coefficient of 1 representing perfect prediction and 0 and -1 representing average random and inverse prediction, respectively.

In addition to the screw hole dataset, the XGBoost classifier is trained on the AC and AV terminal datasets. Similarly, both datasets are split into training and test sets with an 0.8 ratio, and the cross-entropy loss is monitored during training. Hence, utilizing the early-stopping criteria, the training stopped at 447 and 197 epochs for the AV and AC, respectively. The evolution of training and validation losses for both datasets is shown in Fig. 18.

To quantity, the robustness of the XGBoost classifier to distinguish between the safe, crack, and margin classes, a

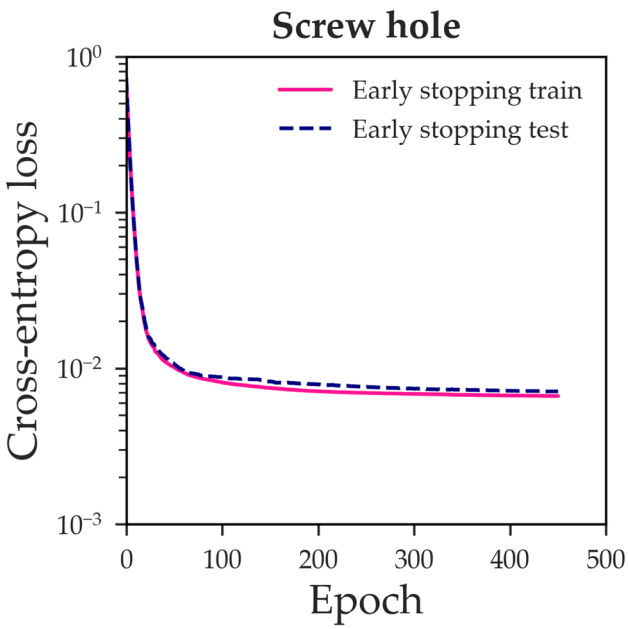


Fig. 16 Evolution of training and validation losses for datasets based on screw holes

receiver operating characteristic (ROC) curve is plotted for AC and AV terminal datasets. As shown in Fig. 19, the AUC score of 1.0 indicates that the XGBoost classifier correctly

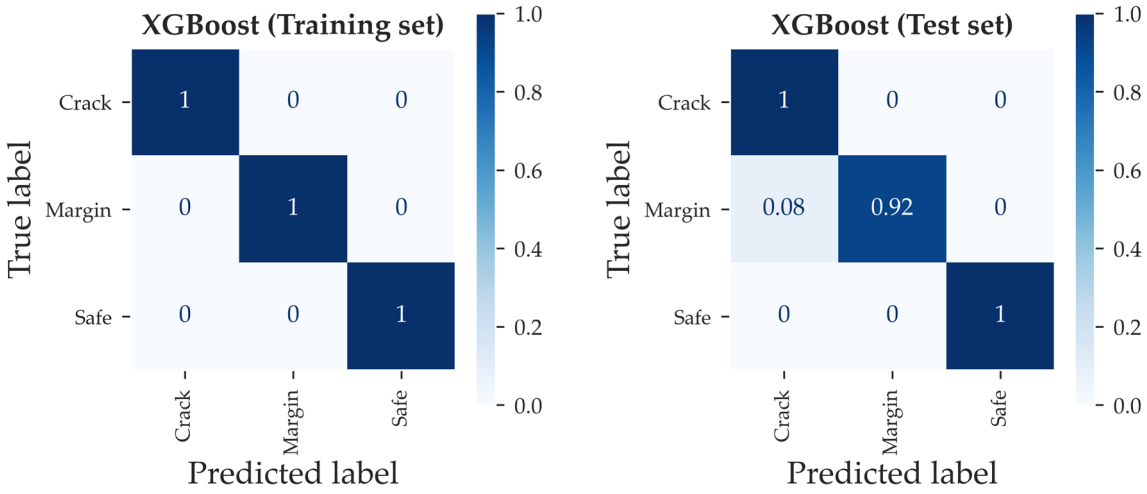


Fig. 17 Normalized confusion matrix predicted by the tuned XGBoost for the screw holes

Table 6 Matthews correlation coefficient (MCC) of the test set based on screw holes

Decision Tree	XGBoost (Vanilla)	XGBoost
0.7877	0.9004	0.9823

distinguishes a given class compared to the remaining classes (One-vs-Rest) over all possible threshold values. Hence, with the accuracy and performance metrics established for all distinct datasets of the TV back panel, the training of the XGBoost classifier is concluded. After that, the trained classifier is extracted and implemented in a multi-language pipeline. In this particular case, the XGBoost classifier was exported to Predictive Model Markup Language (PMML) format. The PMML model was executed in Java by adding the PMML dependency using the Apache Maven build tool. Thus, providing real-time classification output for formability.

Optimization process for formability improvement

The optimization process aims to obtain an optimized geometry of the TV back panel to improve its formability. To demonstrate the applicability of the methodology, the punch radius of the screw holes was optimized using Brent's method in the Java environment. Section "Optimization process" discusses the optimization process based on the XGBoost classifier.

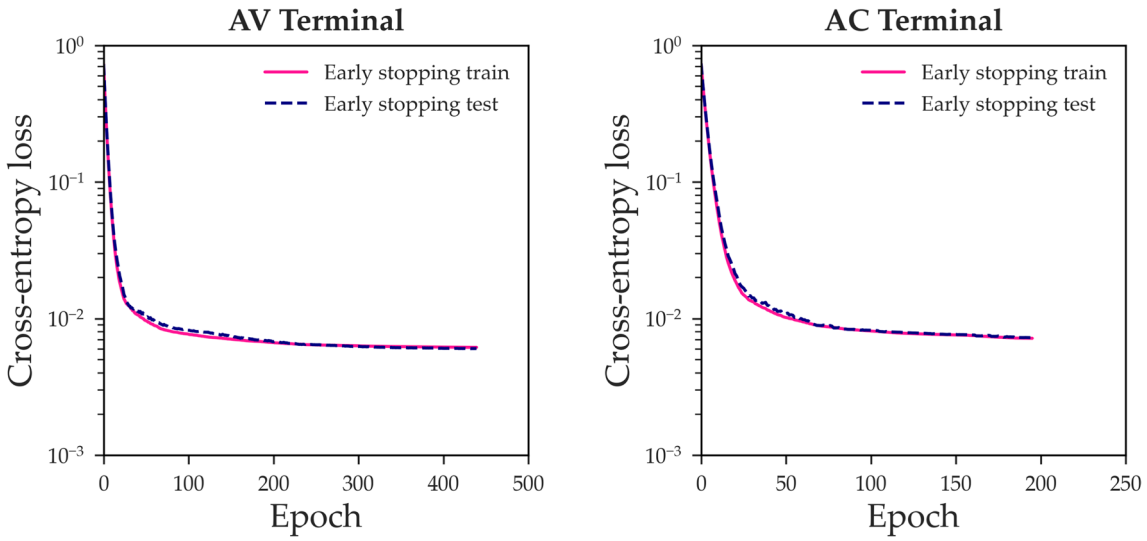


Fig. 18 Evolution of training and validation losses for datasets based on AV and AC terminals

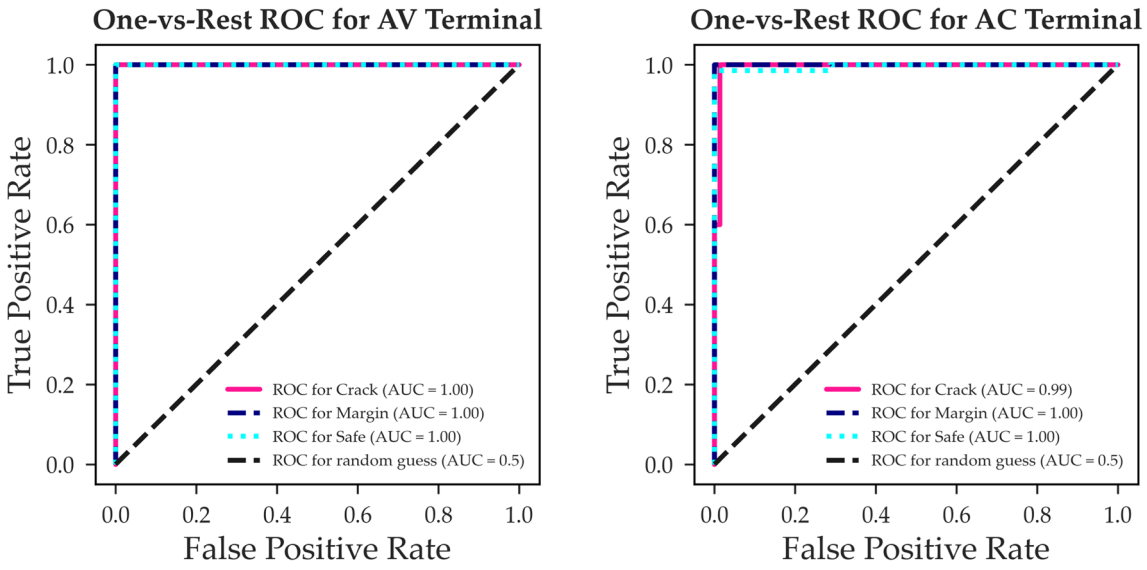


Fig. 19 Receiver operating characteristic (ROC) curve for the test sets obtained using AV and AC datasets

Optimization process

The XBoost classifier predicts probability estimates for the safe, crack, and marginal classes based on the geometrical dimensions of the screw holes. The predicted probabilities are used to minimize an error function. In this particular case, the punch radius of the screw holes is chosen as the design variable for the optimization process. Figure 20 shows the schematic representation of the optimization process for the formability improvement of the TV back panel.

Initially, the XGBoost classifier assesses the forming probability $y(x_i)$ of the screw hole based on the blank thickness and its cross-sectional dimensions (see Fig. 3a), denoted as x_i. The term x_i represents the initial input vector to the optimization program, with i ranging from 1 to 6, indicating a total of six inputs. The optimization program terminates if the classification output indicates safe class. Otherwise, the XBoost classifier re-evaluates the forming probability $\hat{y}(\hat{x}_i)$ using the given screw hole punch radius $\hat{x}_6$ and predefined inputs $(\hat{x}_{i-1})$. The objective function is defined as the root mean square error between the predicted and target probability estimates, as follows,

$$obj = \left[y^{\text{Target}} - \left(\hat{y}(\hat{x}_i) - y(x_i)\right)\right]^2 \tag{5}$$

where $\hat{y}(\hat{x}_i)$ is the forming probability estimate by the XGBoost classifier, and $y^{\text{Target}} = 1.0$ is the user-defined safe probability that satisfies design tolerance.

Thereafter, Brent's method minimizes the objective function by searching the optimized value of the punch radius of the screw holes in the range $0.4 \leq C \leq 2.0$. The number of iterations is set to 200. The optimization process is stopped if the value of the objective function stagnates or reaches a predefined tolerance. The optimized parameter $\hat{x}_6^{optimized}$ corresponds to the punch radius the XGBoost identifies as the optimal forming geometry for the screw holes.

For instance, during the FEM simulation, the forming of the screw hole with the geometric dimensions listed in Table 7 resulted in strain values above the forming limit curve. Consequently, this particular data point was classified as a crack in the training dataset. The initial XGBoost prediction correctly classified the forming of the screw hole as likely to result in a crack.

Therefore, the optimization process was carried out to determine the optimal punch radius for the screw hole. Figure 21 shows the evolution of the objective function during the optimization process. The optimized value of the punch radius C was found to be 1.5404 mm. For the computation time, the optimization process took 3.11 s using an Intel Core™ i9–12900F, 64 GB of DDR4 RAM, and an NVIDIA GeForce RTX™ 3080 Ti GPU. The objective function value at the end of the optimization is reported to be 7.13e-5. Table 8 provides the initial and final probability estimates by the XGBoost classifier.

To evaluate the effectiveness of the optimization process, forming simulation was conducted using the optimized value of the punch radius. Thereafter, the strain values obtained are plotted on the forming limit diagram for both the initial and optimized geometric dimensions of the screw hole, as shown in Fig. 22. It is observed that the optimization process effectively improves the formability of the screw hole.

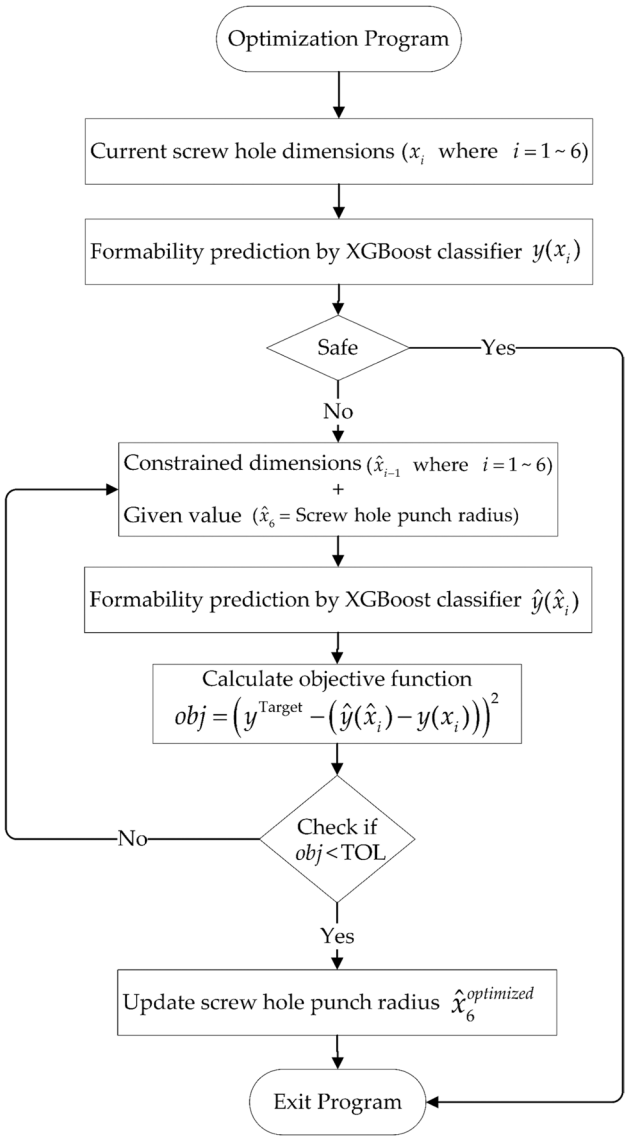


Fig. 20 Schematic representation of optimization process for formability improvement of back panel

Conclusions

This study presents a machine learning-based methodology for formability analysis in sheet metals. The machine learning (ML) model uses an XGBoost classifier to classify the formability of the TV back panel based on the forming limit curve (FLC). The trained classification model maps the panel geometry, including screw holes, AC (Alternating Current), and AV (Audio Visual) terminal, to one of three formability classes: safe, marginal, and cracked. Strain values below the FLC are classified as safe, those within 5% margin of the FLC are classified as marginal, and those above are classified as cracked. Further specific conclusions are summarized as follows:

1. The training data was generated numerically using finite element simulations and verified through experimental strain measurements. The commercial FEM package, PAM-STAMP, with the explicit solver, was used to form the TV back panel. During the data generation phase, 630 finite element simulations were carried out with combinations of classification inputs, each belonging to a specific design domain. The design domain is user-defined and corresponds to the geometry of the TV back panel.
2. The XGBoost classifier was developed to evaluate whether a new change in the geometry of the TV back panel is safe, marginal, or likely to cause cracking. The XGBoost classifier outputs probability estimates for each class and makes classifications based on the highest probability value. Accuracy and performance metrics indicate that XGBoost makes accurate classifications for all datasets of screw hole, AC, and AV terminal.
3. The optimization process was developed using the trained XGBoost classifier and aims to obtain an optimized geometry of the TV back panel to improve its formability. To demonstrate the applicability of the methodology, the punch radius of the screw holes was optimized. Regarding computation time, the optimization process took 3.11 s.

Appendix 1

A. Tree Boosting

(I) *Decision Tree*

Decision tree learning is a supervised learning approach that utilizes if-else or true–false feature questions to predict a category in a classification problem or continuous numeric value in a regression problem. For a given dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, a tree model is given by,

$$f(\mathbf{x}_i) = \sum_{j=1}^{T} w_j \mathrm{I}(\mathbf{x}_i \in R_j) \tag{6}$$

Table 7 Input dimensions of the screw hole for the optimization process

Input $\boldsymbol{x_i}$	*A (Height)*	*B (Die radius)*	*C (Punch radius)*	*D (θ)*	*E (Diameter)*	*Thickness*
Given value	2.22 [mm]	1.0 [mm]	0.5 [mm]	16.82 [deg]	9.0 [mm]	0.5 [mm]

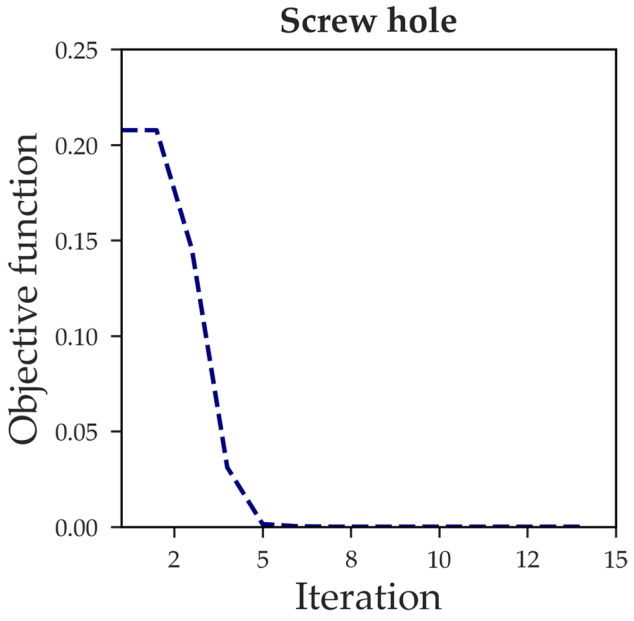


Fig. 21 Evolution of the objective function during the optimization process

Table 8 Initial and final probability estimates by the XGBoost classifier

XGBoost Output	Probability estimates		
	Safe	*Margin*	*Crack*
$y(x_i)$	0.0043	8.7707e-4	0.9948
$\hat{y}(\hat{x}_i)$	0.9958	0.0037	3.9304e-4

where w_j is the score (prediction) of the j-th leaf, referred to as weight in the region R_j, I is the set of indices of data points assigned to the j-th leaf, and T is the total number of leaves in the tree (see Fig. 1).

(II) *Boosting*

Boosting is a class of machine learning algorithms that iteratively combines multiple base learners to form a prediction model. The base learners are generally weak but provide accurate predictions when combined in an ensemble hence the term 'boosting.' Given the base learner to be decision trees with K trees, the predicted output $\hat{y}_i$ corresponding to the input vector $\mathbf{x}_i$ of the i-th instance is given by,

$$\hat{y}_i = \sum_{k=1}^{K} f_k(\mathbf{x}_i), f_k \in \mathcal{F} \tag{7}$$

where f_k is the output of the k-th tree, and $\mathcal{F}$ is a set of all possible classification and regression tree (CART) functions. Tree boosting learns by iteratively adding $f_t(x_i)$ to base learners, such that it minimizes the following objective function,

$$\begin{aligned} \mathcal{L}^{(t)} &= \sum_{\mathcal{D}} l\left(y_i, \hat{y}_i^{(t)}\right) \\ &= \sum_{\mathcal{D}} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)\right) \end{aligned} \tag{8}$$

where,

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{9}$$

D is the size of the training set, $\hat{y}_i^{(t)}$ and y_i are the predicted and target values at the t-th iteration, respectively, and the l term is a differentiable convex loss function that measures the difference between $\hat{y}_i^{(t)}$ and y_i. In general settings, boosting utilizes a second-order taylor approximation of the loss function, and the objective function is re-written as follows,

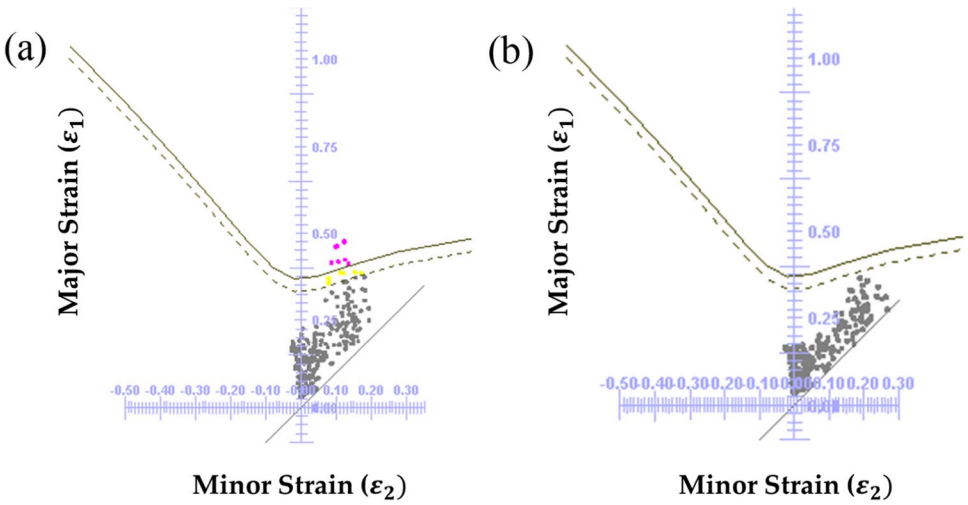


Fig. 22 Strain distribution in the screw hole forming simulation with geometric dimensions (**a**) Initial (Table 7) (**b**) Optimized C = 1.5404 mm (Punch radius)

$$\widetilde{\mathcal{L}}^{(t)} = \sum_{\mathcal{D}} \left[g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i) \right] \tag{10}$$

where g_i and h_i are the gradient and hessian, respectively, defined as,

$$g_i = \partial_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right); h_i = \partial^2_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right) \tag{11}$$

(III) *XGBoost method*

XGBoost employs Newton tree boosting to fit additive tree models. In Newton boosting, the base learners are the tree models and for a given dataset with n examples $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, a tree model from Equation 6 is given by,

$$f_t(\mathbf{x}_i) = \sum_{j=1}^{T} w_j \mathrm{I}(\mathbf{x}_i \in R_j) \tag{12}$$

XGBoost learns by iteratively adding $f_t(\mathbf{x}_i)$ to base learners, such that it minimizes the following objective function (as described in Equation 10),

$$\begin{aligned} \widetilde{\mathcal{L}}^{(t)} &= \sum_{\mathcal{D}} \sum_{j=1}^{T} \left[g_i w_j + \tfrac{1}{2} h_i w_j^2 \right] \\ &= \sum_{j=1}^{T} \left[G_j w_j + \tfrac{1}{2} H_j w_j^2 \right] \end{aligned} \tag{13}$$

where

$$G_i = \sum_{\mathcal{D}} g_i; H_i = \sum_{\mathcal{D}} h_i$$

At each iteration, XGBoost learns the leaf weights and tree structure through the following steps:

1. For each leaf j, the optimization problem in Equation 13 is quadratic in w_j. Hence, the optimal leaf weight or prediction w_j^* for a proposed (fixed) tree structure is obtained by setting $d\widetilde{\mathcal{L}}^{(t)}/dw_j = 0$ as follows,

$$w_j^* = -\frac{G_j}{H_j}, j = 1, \ldots, T. \tag{14}$$

2. Learning the tree structure involves searching for splits of internal nodes. To compute the optimal split, the objective reduction following Equations 13 and 14 is,

$$obj^* = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j} \tag{15}$$

Equivalently, the splits are determined that maximizes the gain given by,

$$\text{Gain} = \frac{1}{2} \left[\frac{G_L^2}{H_L} + \frac{G_R^2}{H_R} - \frac{G^2}{H} \right] \tag{16}$$

Here, the terms with subscripts L and R correspond to the left and right leaf scores, respectively, and the last term is the score of the original node. The nodes with negative gain are pruned out in bottom-up order.

3. The final leaf weights $\hat{w}_j$ are computed following Equation 14 for the learned tree structure.

Acknowledgements This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (2023R1A2C2005661). The authors are grateful for the supports. Also, this work is partially supported from the Technology development Program (S3288770) funded by the Ministry of SMEs and Startups (MSS, Korea).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Har-Peled, S, Roth, D, Zimak, D, (2002) Constraint Classification for Multiclass Classification and Ranking. Adv Neural Inf Process Syst 15: Proceedings of the 2002 Conference, MIT Press
2. Breiman, L, (1996) Bias, Variance, And Arcing Classifiers. Statistics Department: University of California. Technical report 460
3. Li, P, (2010) Robust LogitBoost and Adaptive Base Class (ABC) LogitBoost. In Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence (UAI'10), 897–904
4. Bennett J, Lanning S (2007) The netflix prize. In Proceedings of the KDD Cup Workshop 2007:3–6
5. He, X, Pan, J, Jin, O, Xu, , Liu, B, Xu, T, Shi, Y, Atallah, A, Herbrich, R, Bowers, S, Candela JQ N, (2014) Practical lessons from predicting clicks on ads at facebook. In Proceedings of the Eighth International Workshop on Data Mining for Online Advertising, ADKDD'14
6. Choi DK (2019) Data-Driven Materials Modeling with XGBoost Algorithm and Statistical Inference Analysis for Prediction of Fatigue Strength of Steels. Int J Precis Eng Manuf 20(1):129–138
7. Chheda AM, Nazro L, Sen FG, Hegadekatte V (2019) Prediction of forming limit diagrams using machine learning. IOP Conf Ser Mater Sci Eng 651:012107
8. Finamor FP, Wolff MA, Lage VS (2021) Prediction of forming limit diagrams from tensile tests of automotive grade steels by a machine learning approach. IOP Conf Ser Mater Sci Eng 1157:012080
9. Elangovan K, Narayanan CS, Narayanasamy R (2011) Modelling of forming limit diagram of perforated commercial pure aluminium sheets using artificial neural network. Comput Mater Sci 47(4):1072–1078
10. Bonatti C, Mohr D (2021) Neural network model predicting forming limits for Bi-linear strain paths. Int J Plast 137:102886
11. Keeler, SP, (1961) Plastic instability and fracture in sheets stretched over rigid punches. MIT. Dept. of Metallurgy
12. Goodwin GM (1968) Application of Strain Analysis to Sheet Metal Forming Problems in the Press Shop. SAE Trans 77:380–387

13. Marciniak Z, Kuczyński K (1967) Limit strains in the processes of stretch-forming sheet metal. Int J Mech Sci 9:609–620
14. Nakazima K, Kikuma T, Hasuka K (1968) Study on the formability of steel sheets. Yawata Tech Report 264:141–154
15. Ragab AR, Baudelet B (1982) Forming limit curves: Out-of-plane and in-plane stretching. J Mech Work Technol 6(4):267–276
16. Raghavan KS (1995) A simple technique to generate in-plane forming limit curves and selected applications. Metall Mater Trans A 26:2075–2084
17. ASTM E2218–15, (2016) Standard Test Method for Determining Forming Limit Curves
18. Hill R (1952) On discontinuous plastic states, with special reference to localized necking in thin sheets. J Mech Phys Solid 1(1):19–30
19. Swift HW (1952) Plastic instability under plane stress. J Mech Phys Solid 1(1):1–18
20. Stören S, Rice JR (1975) Localized necking in thin sheets. J Mech Phys Solid 23(6):421–441
21. Hutchinson, JW, Neale, KW, (1978) Sheet Necking-II. Time-Independent Behavior. In: Mechanics of Sheet Metal Forming. 127–153
22. Brunet M, Morestin F (2001) Experimental and analytical necking studies of anisotropic sheet metals. J Mater Process Technol 112(2–3):214–226
23. Zhang R, Shao Z, Lin J (2018) A review on modelling techniques for formability prediction of sheet metal forming. Int J Lightweight Mater Manuf 1(3):115–125
24. Stoughton TB, Zhu X (2004) Review of theoretical models of the strain-based FLD and their relevance to the stress-based FLD. Int J Plast 20(8–9):1463–1486
25. Stoughton TB, Yoon JW (2012) Path independent forming limits in strain and stress spaces. Int J Plast 49(25):3616–3625
26. Song WJ, Heo SC, Kim J, Kang BS (2006) Investigation on preformed shape design to improve formability in tube hydroforming process using FEM. J Mater Process Technol 177(1–3):658–662
27. Ko D-C, Cha S-H, Lee S-K, Lee C-J, Kim B-M (2010) Application of a feasible formability diagram for the effective design in stamping processes of automotive panels. Mater Des 31(3):1262–1275
28. Attanasio A, Ceretti E, Fiorentino A, Mazzoni L, Giardini C (2009) Experimental Tests to Study Feasibility and Formability in Incremental Forming Process. In Key Engineering Materials 410–411:391–400
29. Kim H-K, Kim H-W, Cho J-H, Lee J-C (2013) High-formability Al alloy sheet produced by asymmetric rolling of strip-cast sheet. Mater Sci Eng A 574:31–36
30. Zimmerling C, Dörr D, Henning F, Kärger L (2019) A machine learning assisted approach for textile formability assessment and design improvement of composite components. Compos - A: Appl 124:105459
31. Bae MH, Kim M, Yu J, Lee MS, Lee SW, Lee T (2022) Enhanced processing map of Ti–6Al–2Sn–2Zr–2Mo–2Cr–0.15Si aided by extreme gradient boosting. Heliyon 8(10):10991
32. Lu W, Xiao W, Li Y, Zheng K, Wu Y (2023) Machine learning in the prediction of formability in aluminum hot stamping process with multiple variable blank holder force. Int J Comput Integr Manuf 36(5):702–771
33. Marques AE, Dib MA, Khalfallah A, Soares MS, Oliveira MC, Fernandes JV, Ribeiro BM, Prates PA (2022) Machine Learning for Predicting Fracture Strain in Sheet Metal Forming. Metals 12(11):1799
34. Singh AR, Bashford-Rogers T, Marnerides D, Debattista K, Hazra S (2023) HDR image-based deep learning approach for automatic detection of split defects on sheet metal stamping parts. Int J Adv Manuf Technol 125:2393–2408
35. Chen, T, Guestrin C, (2016) XGBoost: A Scalable Tree Boosting System. arXiv:1603.02754
36. ASTM E8/E8M-16a, (2016) Standard Test Methods for Tension Testing of Metallic Material. ASTM. E8/E8MQuery
37. Kuwabara T, Ikeda S, Kuroda K (1998) Measurement and analysis of differential work hardening in cold-rolled steel sheet under biaxial tension. J Mater Process Technol 80:517–523
38. Wu X-D, Wan M, Zhou X-B (2005) Biaxial tensile testing of cruciform specimen under complex loading. J Mater Process Technol 168–1:181–183
39. Barlat F, Brem JC, Yoon JW, Chung K, Dick RE, Lege DJ, Pourboghrat F, Choi S-H, Chu E (2003) Plane stress yield function for aluminum alloy sheets—part 1: theory. Int J Plast 19:1297–1319
40. Ozturk, F, Dilmeç, M, Turkoz, M, Ece, RE, Halkaci, HS (2009) Grid Marking and Measurement Methods for Sheet Metal Formability. In: 5th International Conference and Exhibition on Design and Production of Machines and Dies, Molds
41. Gorodkin J (2004) Comparing two K-category assignments by a K-category correlation coefficient. Comput Biol Chem 28(5–6):367–374
42. Jurman G, Riccadonna S, Furlanello C (2012) A Comparison of MCC and CEN Error Measures in Multi-Class Prediction. PLoS ONE 7(8):e41882
43. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay É (2011) Scikit-learn: Machine Learning in Python. J Mach Learn Res 12(85):2825–2830
44. Bishop CM (2006) Pattern Recognition and Machine Learning. Springer