Received May 9, 2022, accepted June 4, 2022, date of publication June 8, 2022, date of current version June 16, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3181524

Evaluating Machine Learning-Based Classification Approaches: A New Method for Comparing Classifiers Applied to Human Driver Prediction Intentions

DANIEL ADOFO AMEYAW, QI DENG, AND DIRK SÖFFKER, (Member, IEEE)
Chair of Dynamics and Control, University of Duisburg-Essen, 47057 Duisburg, Germany
Corresponding author: Daniel Adofo Ameyaw (daniel.adofo-ameyaw@uni-due.de)

ABSTRACT In this research, a new performance assessment based on the Probability of Detection (POD) reliability measure is developed integrating and discussing the effect of further parameters on classification results and therefore establishing a new connection between relevant process parameters and the related classifier evaluation. To illustrate the approach, machine learning-based recognition of complex driving situations for human drivers is interpreted. Using sensor signals and a complex driving scenario, related dynamical changes are classified and compared using the POD approach. Based on the POD-related evaluation, different machine learning approaches can be clearly distinguished with respect to their ability to predict the correct driver behavior as a function of time prior to the event itself. The introduced approach allows a very detailed comparison of classifiers relative to the effects of parameters affecting the processes to be classified. In addition to recently published results on this novel approach, an extension of the POD approach by considering false positives and varying decision threshold in the comparison process is proposed. Generalization of the introduced approach for binary and continuous data is presented.

INDEX TERMS Classification, machine learning, performance evaluation, probability of detection.

I. INTRODUCTION

Machine learning (ML) approaches are essential processes or sets of procedures that helps a model adapt to given data. These ML approaches use algorithms and statistical models providing suitable processor hardware the ability to perform tasks without specific instructions. The significance of ML has been accepted and implemented in medical diagnosis, cybersecurity, spam filtering, fraud detection, and significantly in the field of computer vision [1], [2] [3]. The receiver operating characteristic (ROC) is among the commonly used evaluation tools for ML algorithms. The ROC has been used to evaluate edge detector performance [4] and multi-class classification problems [5]. Other authors [6] have improved image classification using an intermediate representation. The final evaluation process provides a measure to select possibly optimal models discarding properties related to process parameters [7]. This implies, the ROC cannot quantitatively relate detectability to a process parameter [8], [9]. This limitation in verifying the effect of process parameters on the classifier performance is addressed in this research by extending the Probability of detection (POD) metric.

Probability of detection has been implemented in the field of nondestructive testing (NDT), radar systems, and lately structural health monitoring (SHM) systems. The POD is a probabilistic method that allows to compare the performance of different monitoring techniques by estimating the sensitivity and reliability of the inspection process. This allows the quantification of the reliability of a procedure taking into account statistical variability of sensor and measurements properties [9], [10] [8]. Probability of detection is employed in many industries nowadays. The aircraft industry, and particularly in a military context, the POD information is used for damage tolerance analysis of components and scheduling of inspection intervals [11]. The pipeline industry has relied on POD analysis to develop fitness for purpose acceptance criteria for the construction of pipelines. Nuclear industries

The associate editor coordinating the review of this manuscript and approving it for publication was Baoping Cai.

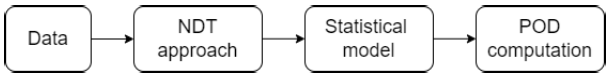

FIGURE 1. Classical POD approach.

are adopting this form of analysis to assess the reliability of NDT to detect flaws in components during in-service inspections [11]. The NDT/SHM fields utilize the POD curve. The POD curve is constructed by plotting the accrual of flaws detected against the varying parameter or producing a response over a specified threshold [10], [11]. The statistical and probabilistic assessments of a measurement procedure are time-consuming and costly since several samples have to be verified and compared using destructive methods. This has given rise to Model-assisted POD (MAPOD) to improve the effectiveness of POD models with little or no specimen testing by utilizing model generated data [12]. Two MAPOD methods are currently utilized. The first approach utilizes physics-based models to propagate directly the uncertainty of a given set of assessment parameters to another with same parameters [13], [14]. The transfer function approach is a physics-based method that transfers the computed POD curve for a specific process to another with different parameters. However numerical efforts and computational time difficulties have to be solved for convenient application in practice. A detailed explanation of best practices for the use of simulation data in estimation POD curves is presented in [15]. Classical POD approaches in the NDT field utilize the approach indicated in Fig. 1.

In this research, the POD approach is adapted and extended to evaluate the ability of classifiers to predict drivers' intention as a function of remaining time to the predicted event. Previously solved problems relating to evaluating the performance of classifier using the POD will be introduced for understanding. This serves as a prelude to the newly developed procedure. The direct approach of comparing ML algorithms based on the selection of a common decision threshold [16] will be introduced and its merit over the ROC demonstrated. The limitation of the common decision threshold approach will be explained and an advanced approach examining concurrently the detection probability, false positive, decision threshold, and process parameter in the comparison process will be presented. This new extended proposal is relevant because it provides an approach to additionally evaluate changes in the false positive rate when varying a threshold value, which is not possible with the conventional common threshold approach.

The article is organized as follows: in section II the ML approaches used are briefly introduced, followed by the developed POD reliability measure. The application to driving maneuver prediction is detailed in section III. An experimental validation of the proposed approach is presented in section IV. Results, discussions, and comparisons between different classifiers are given in section V. False positive analysis procedure and a trade-off between probability of false positive and POD together with the generalization of the introduced approach are presented in section VI and VII respectively. Conclusion finalizes the contribution.

II. MACHINE LEARNING ALGORITHMS

To illustrate existing challenges in evaluating ML approaches, in this contribution eight ML algorithms are evaluated. The comparison of the results will be additionally realized by the POD approach which is the core illustration of the new approach. In [17] an approach to improve training of conventional algorithms is proposed. The authors showed that usually a set of unknown classifier tuning parameters are needed to be set manually before training when a conventional algorithm is used. To improve the prediction performance of the model, a prefilter is proposed to quantize the signals into observed sequences with specific features. Here optimality is defined as the optimal segments describing a quantized prefilter mapping the vehicle's environment to quantized states. With the proposed training procedure, the most suitable values of these unknown parameters can be determined automatically to optimize the performance of conventional algorithms. The ML approaches analyzed as examples are conventional/improved ANN [17], conventional/improved HMM [17], [18], conventional/improved RF [19], and conventional/improved SVM. These classifiers are selected to illustrate the proposed approach, and they are not exhaustive and are meant only as examples. The proposed approach can be extended to other classifiers.

A. CONVENTIONAL EVALUATION

To illustrate the conventional evaluation of ML approaches, classical measures are first presented. These conventional evaluation is applied to the ML approaches in their conventional form (default implementation) and their improved form (as published in [17]). The development of the improved version is not part of this contribution, however the results are evaluated to verify the correspondence with the POD. The conventional and improved models are calculated in the training phase. Based on these models, the driving behaviors in the upcoming driving processes could be determined. The measured and estimated driving behaviors are compared to check the correspondence.

To evaluate the performance of classifiers, the detection rate (DR) and false alarm rate (FAR) values are calculated as shown in [20] as

$$DR = \frac{TP}{TP + FN} \quad and \tag{1}$$

$$FAR = \frac{FP}{TN + FP}. \tag{2}$$

III. POD ASSESSMENT OF ML ALGORITHMS

Probability of Detection is a certification tool [10]. Data used in producing POD curves are categorized by the main POD controlling factors/variables. These factors/variables

Multiclass Confusion Matrix		Predicted		
		S_1	S_2	S_3
Actual	S_1	TP	FN	FN
	S_2	FP	TN	TN
	S_3	FP	TN	TN

FIGURE 2. S_1 **multiclass confusion matrix, TP: true positive FP: false positive FN: false negative TN: true negative.**

are either discrete or continuous and can be classified as shown in [9], [10] as

1) Hit/miss: produce binary statement or qualitative information about the existence of targets.
2) Target-response: systems that provide quantitative measure of targets.

The target-response approach is adapted and implemented here in evaluating lane changing prediction capabilities of different ML algorithms. A brief introduction to the POD measurement is presented in preparation for the development of an extended version of the approach.

A. TARGET-RESPONSE APPROACH TO POD

The target-response approach is used when there exists a relationship between a dependent function and an independent variable [9]. The characteristic parameter of the target is usually used (size, length, etc). The response denotes the measurement output of the target. In the derivation of the POD curve, a predictive modeling technique is required. One such method is regression analysis of the data gathered [8], [21]. Ordinary Least Squares is a popular and often used linear regression technique however it is ineffective in the presence of censored data. In such situations, alternative techniques like the Maximum Likelihood Estimation must be implemented.

The data distribution could be linear or not. A strategy to linearize the data distribution is by plotting four models: *X vs Y, log X vs Y, log Y vs X, and log X vs log Y*. The model with the best linearity and variance is used in the construction of the POD curve [22]. The regression equation for a line of best fit to a given data set is given by

$$y = b + mx + \epsilon, \tag{3}$$

where the regression coefficients m and b represent the slope and intercept respectively and $\epsilon \sim N(0, \tau)$ is the corresponding error term, with normal distribution and having zero mean and a standard deviation τ. The confidence bounds are constructed to define a confidence interval that contains 95 % of the observed data [22]. Here the 95 % confidence bounds on y is constructed by

$$y_{a=0.95} = y + 1.645\tau_y, \tag{4}$$

where 1.645 is the z-score of 0.95 for a one-tailed standard normal distribution and τ_y the standard deviation of the regression line. The Delta method is a statistical technique used to transition from regression line to POD curve [8], [9]. The confidence bounds are computed using the covariance matrix for the mean and standard deviation POD parameters μ and σ respectively. To estimate the entries, the covariance matrix for parameters and distribution around the regression line needs to be determined. This is done using the Fisher's information matrix I. The information matrix is derived by computing the maximum likelihood function f of the standardized deviation z of the regression line values. The entries of the information matrix are calculated by the partial differential of the logarithm of the function f using the parameters of $\Theta(m, b, \tau)$ of the regression line.

From

$$z_i = \frac{(y_i - (b + mx_i))}{\tau} \tag{5}$$

and

$$f_i = \prod_{i=1}^{n} \frac{1}{2\pi} e^{-\frac{1}{2}(z_i)^2} \tag{6}$$

the information matrix I can be computed as

$$I_{ij} = -E\left(\frac{\partial}{\partial\Theta_i \partial\Theta_j} log(f)\right). \tag{7}$$

The inverse of the information matrix yields ϕ as

$$\phi = I^{-1} = \begin{bmatrix} \sigma_b^2 & \sigma_b\sigma_m & \sigma_b\sigma_\tau \\ \sigma_m\sigma_b & \sigma_m^2 & \sigma_m\sigma_\tau \\ \sigma_\tau\sigma_b & \sigma_\tau\sigma_m & \sigma_\tau^2 \end{bmatrix}. \tag{8}$$

The mean μ and standard deviation σ of the POD curve are calculated by $\mu = \frac{y_{th}-b}{m}$, where y_{th} is the decision threshold and $\sigma = \frac{\tau}{m}$. The decision threshold determines whether the data are censored or observed and it is very useful in the POD computation. The cumulative distribution Φ is calculated as

$$\Phi(\mu, \sigma) = \frac{1}{2}\left[1 + erf\,\frac{x-\mu}{\sqrt{2}\sigma}\right]. \tag{9}$$

The POD as function of target a is derived as

$$POD(a) = \Phi\left[\frac{a-\mu}{\sigma}\right]. \tag{10}$$

Using equation 10, the POD-curve can be set up for varying parameters. In this article, the varying parameter is the time t. Using the Maximum Likelihood Estimation approach the prediction parameters for the intercept $\hat{\beta}_0$ and gradient $\hat{\beta}_1$ will be estimated. Both parameters are statistically estimated from the observations. [10].

IV. EXPERIMENTAL DESIGN

The experimental set-up and data acquisition process are detailed in this section. The reliability of the outputs of classifiers will be examined using the POD. Data resulting from a professional driving simulator SCANeR^{TM} studio are used in this work. The simulator uses virtual sensors such as cameras, radar, and lasers to collect data. These sensors

FIGURE 3. SCANeR^{TM} studio, Chair Dynamics and Control, UDuE, Germany.

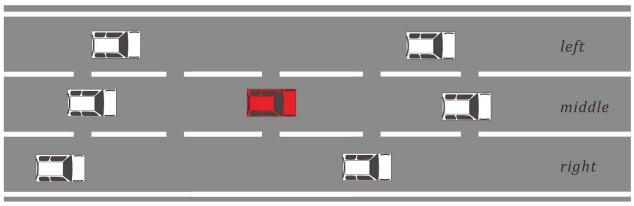

FIGURE 4. Driving environment.

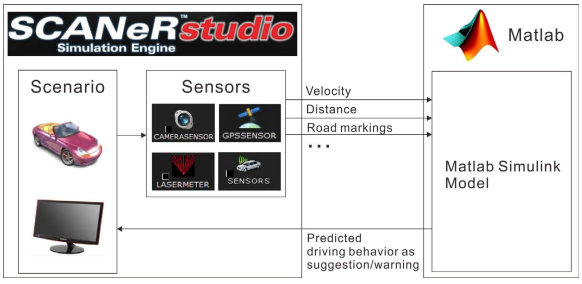

FIGURE 5. SCANeR^{TM} studio simulation engine.

provide a comprehensive understanding of the vehicle's environment. The obtained results and the analysis are discussed in detail.

A. EXPERIMENTAL SET-UP

A driving simulator *SCANeR*^{TM} studio (Fig. 3) is used to perform the driving simulation.

A typical driving scenario is illustrated in Fig. 4 and requires the ego driver (red vehicle) to make a decision.

The driving simulator simulation engine and corresponding input sensors (Fig. 5) aid in decision making.

In total three participants with ages ranging from 25 to 38 years were recruited. They all held valid driving licenses. The training dataset is related to each participant performing 40 minutes drive. Data from another 10 minutes drive are used for the test. To evaluate the predicted performance, a method [7] is used. Here, each lane change behavior is defined as a separate event. From 7 seconds before to the time of actual lane change (see Fig. 6) a DR value will be calculated for performance evaluation. The time interval is divided into 140 time points, i.e. every 0.05 *s*. These time points are defined as ''recognition time point'', and for each time point, a DR value will be calculated for performance evaluation.

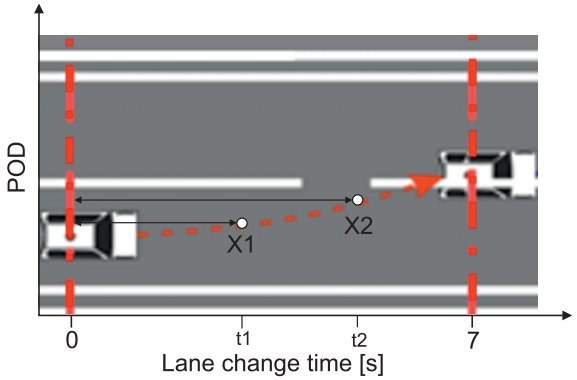

FIGURE 6. Time coordinate for lane change.

The earlier (with respect to the distance in time to the event itself) the algorithm predicts the event, the better this is. From Fig. 6, the ML approach that is able to predict at time $t1$ corresponding to position $X1$ is better compared to an approach that predicts at time $t2$ corresponding to position $X2$. This is because it is able to predict the event occurring at 7 *s* faster.

B. DATA ANALYSIS

In this contribution, four algorithms are selected to train the driving behaviors prediction models. Each algorithm is used to train a conventional and an improved model. A conventional model can be determined using raw data and default design parameters. To obtain improved models [17] the unknown parameters are designed as design parameters of a model, which has to be defined before training. With the proposed training procedure [17], the optimal design parameters can be determined automatically, then the improved models can be trained. All mentioned models use the same observation variables (26 total inputs). In the training phase, all models for each test data set are calculated and saved. Based on these models, the driving behaviors in the test phase could be determined. Finally, the measured (actual labels) and estimated driving behaviors are compared to evaluate the model performance.

The driving behaviors prediction model based on the classifiers is shown in Fig. 7. It consists of two important processes namely driving behavior prediction and parameter definition.

To demonstrate the new approach first classical evaluation will be presented. Classical evaluation employs the use of accuracy (ACC), detection rate (DR), false alarm rate (FAR) among others to compute the suitability of an approach. To evaluate the performance of driving behavior prediction model, a common method [17] is used, in which the values of ACC, DR, and FAR are calculated for the complete driving sequence. The measured driving behaviors and the estimated driving behaviors calculated by the model are compared to check correspondence, then the performance measures of each driving behavior are calculated. For illustration, the evaluation values of the driver data using conventional and

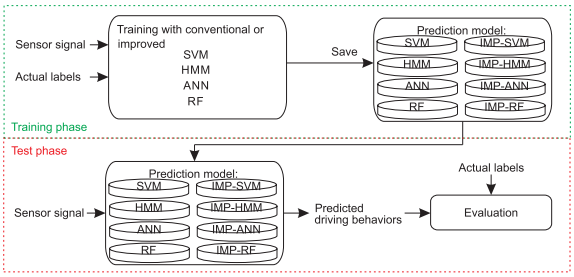

FIGURE 7. Test/training model.

TABLE 1. Conventional evaluation.

	Right lane chane [%]			Left lane change [%]		
	ACC	DR	1-FAR	ACC	DR	1-FAR
ANN	98.27	68.55	99.82	97.57	60.64	99.78
ANN Imp	97.98	87.70	98.52	97.47	69.15	99.16
Difference	-0.29*	19.15	-1.30*	-0.10*	8.51	-0.61*
HMM	94.04	86.29	94.45	93.67	69.50	95.12
HMM Imp	96.47	87.70	98.52	96.28	70.39	97.83
Difference	2.43	4.23	2.34	2.61	0.89	2.71
RF	97.67	71.98	99.01	96.44	56.21	98.85
RF Imp	98.71	88.91	99.22	97.49	65.96	99.37
Difference	1.04	16.94	0.21	1.05	9.75	0.53
SVM	85.94	33.67	88.67	96.06	60.64	98.18
SVM Imp	97.29	70.97	98.66	96.38	60.99	98.50
Difference	11.35	37.30	9.99	0.32	0.35	0.32

Legend- Imp: improved, red: worse results, blue: improved results

improved algorithms are shown in Table 1. The difference between the improved and conventional algorithms are also calculated.

The results using improved algorithms are better relative to conventional algorithms with the exception of ANN. However, the presented results in Table 1 do not provide a means to evaluate the effect of lane change time on the classification results. This is confirmed by the ROC graph (see Fig. 8) of the models. Figure 8 shows the best DR and FAR results for the ML algorithms. However the ROC does not provide a means to evaluate the classification results at each time point. To integrate the effect of time on the classification results, the POD approach is implemented on the same data.

C. EVALUATION OF ML APPROACHES BY THE COMMON THRESHOLD METHOD

The POD is introduced to overcome limitations of the ROC and other evaluation metrics like accuracy. The common threshold approach for comparison of classifiers is illustrated in Fig. 9.

The common threshold method compares the POD values of the individual ML approaches from the built statistical model. To illustrate the common threshold approach, graphical representation of the target-response method is presented. Four models comprising combinations of logarithmic and Cartesian scales (Fig. 10) are established for each classification data to ascertain model with

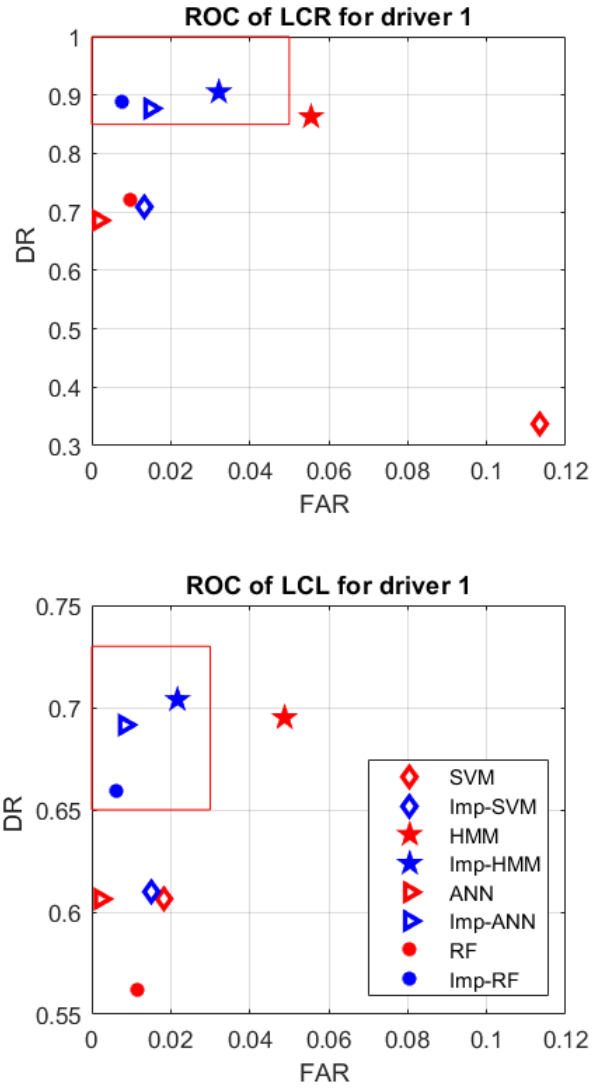

FIGURE 8. ROC results for all models; LCR: lane change to right and LCL: lane change to left.

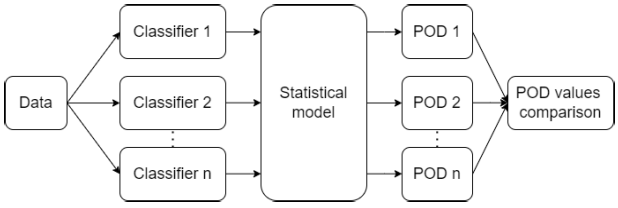

FIGURE 9. Common threshold approach.

1 Linearity of the parameters: $E(y_i|X) = x_i\beta$, where x_i is the $i-th$ row of X,
2 Uniform variance: $var(y_i|X) = \sigma^2, i = 1, 2, 3, \ldots, n$
3 Uncorrelated observations: $cov(y_i, y_j|X) = 0, (i \neq j)$.

The model that fits the aforementioned criteria best is selected. From Fig. 10, model b satisfies all three conditions and is therefore selected. Regression analysis is implemented on the selected model and the decision threshold (red marked

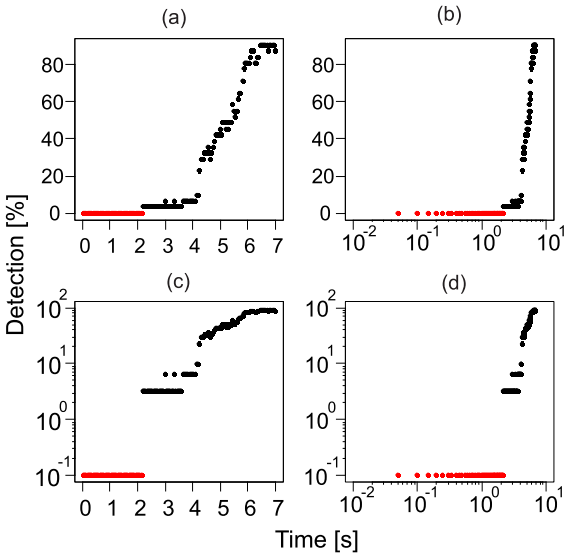

FIGURE 10. Four possible models: a) X vs Y b) log X vs Y c) X vs log Y d) log X vs log Y.

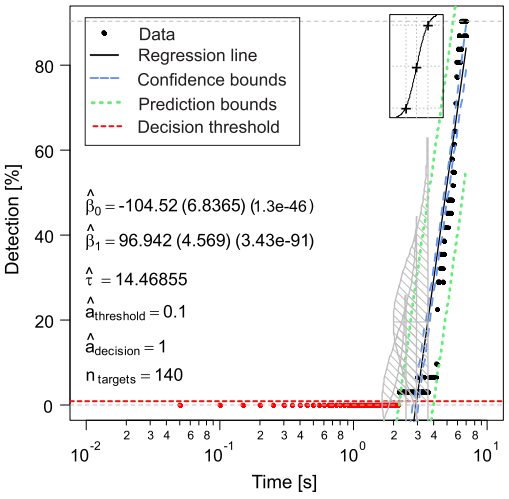

FIGURE 11. POD generation procedure.

line), the response value below which data is considered as noise, is constructed as illustrated in Fig. 11.

The POD curve is generated using the area of the cumulative density function above the decision threshold. The generated curve is relative to a single threshold value. Increasing the decision threshold results in the increase of the POD value, thus the POD curve shifts to the right. Eight PODs are constructed for left and eight PODs for right lane changes. Three of the generated POD curves are illustrated in Figures 12, 13, and 14.

The comparison of Fig. 12 - 14 is given in Fig. 15 using the common threshold approach [16]. The common threshold approach ensures a fair comparison by using the same decision threshold value for all classifiers. The process is repeated for two more drivers resulting in $48[(8 + 8) \times 3]$ POD curves calculated as: 8 classifiers for left lane change and 8 classifiers for right lane change. The procedure is for 3 different drivers.

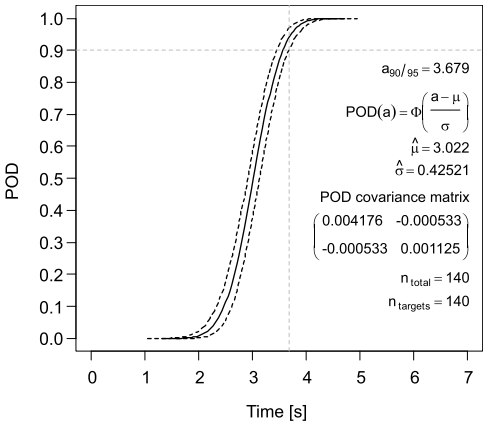

FIGURE 12. Improved ANN POD for Right lane change of driver 1.

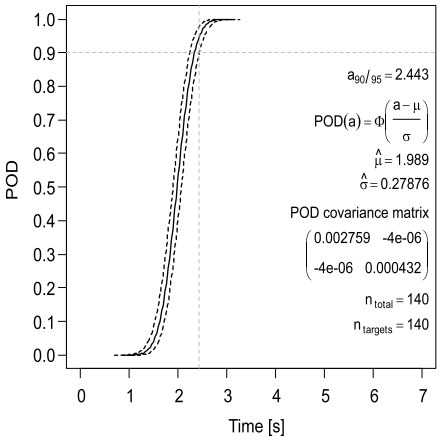

FIGURE 13. SVM POD for right lane change of driver 1.

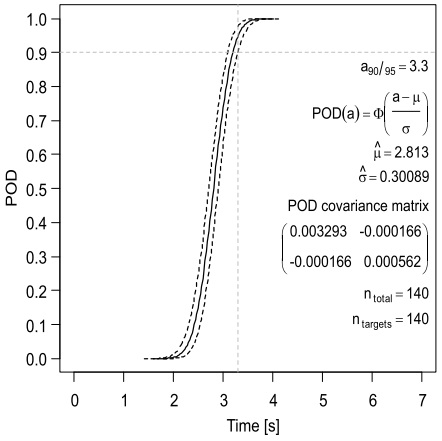

FIGURE 14. Improved SVM POD for right lane change of driver 1.

V. RESULTS AND DISCUSSION

The detailed analysis of the results is discussed in this section. Based on the proposed approach, the POD curves of all

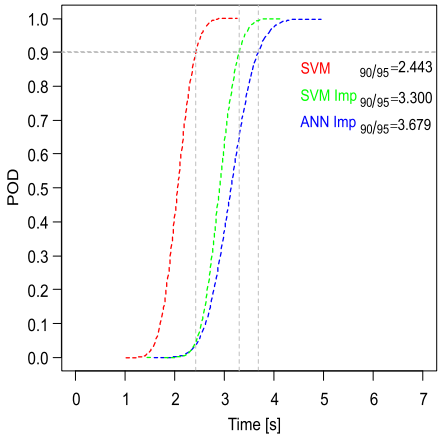

FIGURE 15. Comparison of the $a_{90/95}$ POD values for SVM, improved SVM, and improved ANN.

TABLE 2. Right lane change 90/95 POD.

Algorithm	Driver 1 POD [s]	Driver 2 POD [s]	Driver 3 POD [s]
ANN	3.067	6.383**	4.400
ANN Imp	1.143	5.611	3.128
HMM	1.439	3.518	3.052
HMM Imp	0.6355*	2.801	2.853
RF	4.219**	4.680	5.840**
RF Imp	2.883	5.185	5.223
SVM	2.443	2.548*	1.182*
SVM Imp	3.300	4.084	5.412

Legend- Imp: improved *: best result **: worst results

classifiers are generated. The 90/95 certification standard is used in this analysis. The 90/95 certification in this context expresses the time required to predict complete lane change with 90 % probability at 95 % confidence level. To explain the results; with the lane change occurring at 7 *s*, the improved ANN classifier is able to predict the impending lane change at 3.679 *s* (see Fig. 12) with 90 % probability at 95 % reliability level. Improved SVM (Fig. 13) and conventional SVM (Fig. 14) does the prediction at 3.3 *s* and 2.443 *s* respectively. This provides a means to compare the three classifiers based on their 90/95 POD values. The earlier the prediction time, the better the results. The introduced approach incorporates a process parameter (here: time) and provides a means to directly compare the prediction times (as illustrated in Fig. 15). The 90/95 values of all classifiers for right and left lane change estimations are illustrated in Table 2 and 3 respectively.

From the results in Table 2, in estimating right lane prediction capabilities of different classifiers for driver 1, ANN predicts in 3.067 *s*, improved ANN in 1.143 *s*, HMM in 1.439 *s*, improved HMM in 0.6355 *s*, RF in 4.219 *s*, improved RF in 2.883 *s*, SVM in 2.443 *s*, and improved SVM in 3.300 *s*. The least time values represent the best results. This is because the algorithm is able to predict the complete lane change within the shortest possible time. This implies

TABLE 3. Left lane change 90/95 POD.

Algorithm	Driver 1 POD [s]	Driver 2 POD [s]	Driver 3 POD [s]
ANN	3.325	5.605	4.568
ANN Imp	3.679**	4.143	4.302
HMM	1.204	3.352	3.593
HMM Imp	0.5748*	2.431*	1.625*
RF	3.354	3.974	10.90**
RF Imp	3.181	6.005**	5.631
SVM	2.951	4.037	5.525
SVM Imp	3.485	5.688	5.616

Legend- Imp: improved *: best result **: worst results

improved HMM has best results and RF worst results for driver 1 right lane change estimation. Similar analysis can be made for left lane change estimation for driver 1. The results are extended to data from two other drivers and a summary of the results is as follows.

In estimating right lane change:

1 For driver 1: HMM Imp produces the best results (0.6355 *s*) and RF producing the worst results (4.219 *s*).
2 For driver 2: SVM produces the best results (2.548 *s*) and ANN producing the worst results (6.383 *s*).
3 For driver 3: SVM produces the best results (1.182 *s*) and RF producing the worst results (5.840 *s*).

In estimating left lane change:

1 For driver 1: HMM Imp produces the best results (0.5748 *s*) and ANN Imp producing the worst results (3.679 *s*).
2 For driver 2: HMM Imp produces the best results (2.431 *s*) and RF Imp producing the worst results (6.005 *s*).
3 For driver 3: HMM Imp produces the best results (1.625 *s*) and RF producing the worst results (10.90 *s*).

Accordingly the following statements can be deduced for the experimental results:

1 For this example task, the most suitable classifier is improved HMM producing 4/6 best results.
2 The worst classifier for this example task is RF/ RF Imp producing 4/6 worst results.
3 The application of prefilter to define features and influence prediction performance generally results in an improved POD except for SVM.

From the discussion it can be seen that the introduced approach approach permits a new POD-based certification and comparison method for binary classifiers based on a common threshold value. The results from Table 1 show the reliability as a percentage of correct prediction. Multiple classifiers can be compared using the common threshold approach provided that their false positive probabilities are the same. However the common threshold method is limited in evaluating the corresponding false positive vis-a-vis the POD, decision threshold, and process parameter. To overcome this difficulty, a noise analysis is undertaken in section VI. This new insight extends the comparison of ML approaches to include corresponding false positive values.

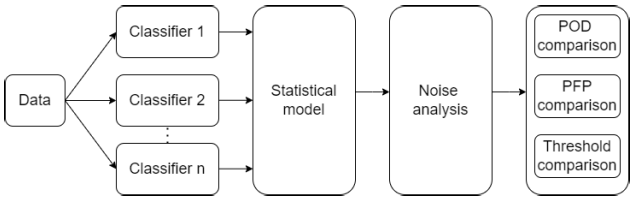

FIGURE 16. Proposed evaluation approach.

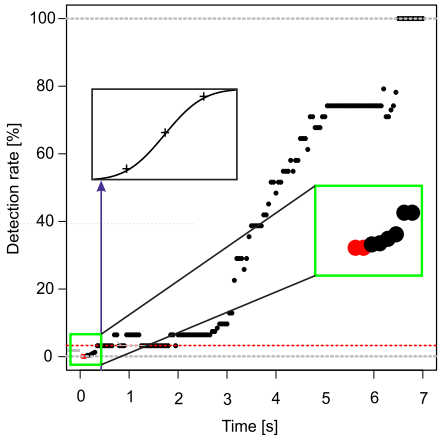

FIGURE 17. SVM data right lane change for driver 1.

The visual representation of how the aforementioned metrics changes as the threshold is varied is presented.

VI. FALSE POSITIVE ANALYSIS

The POD depends on the selected decision threshold y_{th}. Decreasing y_{th} will improve POD but at the cost of increasing the probability of false positive (PFP). It is therefore useful to evaluate the relationship between the POD and PFP using the method described in Fig. 16. Here, noise analysis is incorporated in the evaluation process.

The observed data aggregate the characteristics of the target's signature corrupted by aberrant signals generally referred to as noise. Classical POD methods usually measure noise components as part of experimental measurements, however that is absent in the current work. Noise will be inferred from the observed data. Noise in this context refers to data with no useful target characterization information. Using data from SVM right lane estimation of driver 1 as shown in Fig. 17, the observed data up to the 0.45 s mark has a zero POD value (as can be seen from the POD inset) and therefore has no useful characterizing information. The data up to 0.45 s time is extracted as noise as shown in Fig. 18 and analyzed. The data beneath the decision threshold is censored while those above is observed. The PFP for the extracted noise data can be calculated as

$$PFP = P(y_{noise} > y_{th}) \tag{11}$$

Statistical χ^2 (Chi-squared) hypothesis test is undertaken to identify the nature of noise distribution. Various distributions (Gaussian, Weibull, and Lognormal) are tested.

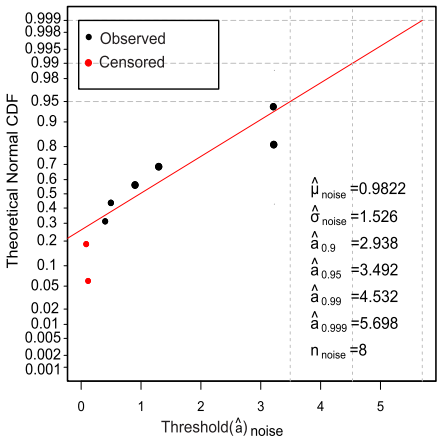

FIGURE 18. Inferred noise data parameters.

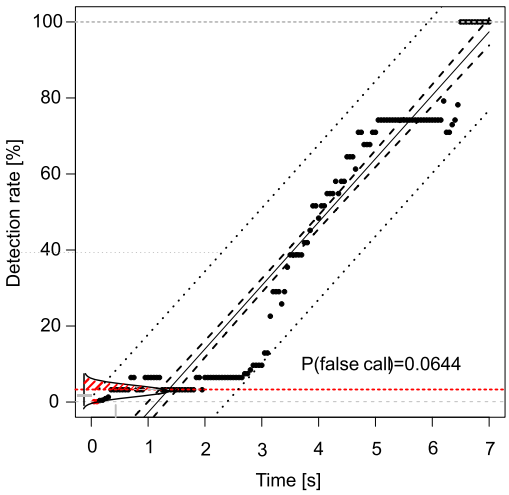

FIGURE 19. Probability of false positive.

The Gaussian distribution emerging most plausible with a p-value of 0.98. Analysis is carried out on the noisy data by analyzing probability densities. By plotting the cumulative density function, the mean μ_{noise} and standard deviation σ_{noise} are computed including the critical target sizes (90% and 90/95 POD values) are evaluated as shown in Fig. 18. For a Gaussian distribution, the probability of false positive is computed as

$$PFP = \int_{y_{th}}^{\infty} \frac{1}{\sqrt{2\pi}\hat{\sigma}_{noise}} e^{-\frac{(y-\hat{\mu}_{noise})^2}{2\hat{\sigma}^2_{noise}}} dy. \tag{12}$$

The distribution with regards to false positive is illustrated in Fig. 19 (shaded red area relative to the selected decision threshold). From Fig. 17 and 19, it can be concluded that for a selected decision threshold, a corresponding unique FAR value exists however the detection probability varies relative to a parameter (here: time). This implies, the premise for the construction of the ROC/PR curve for applications requiring the incorporation of process parameters is deficient. For a selected cut-off point there is not one false positive to one

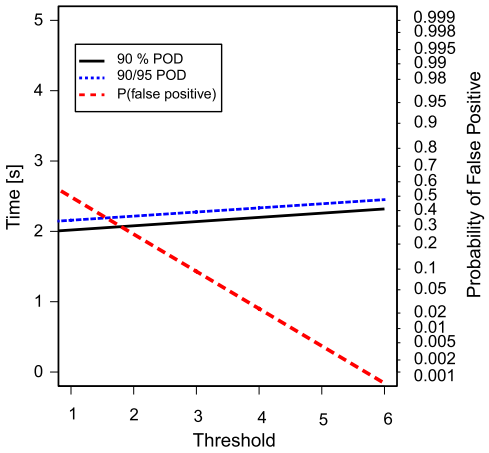

FIGURE 20. Trade-off between POD and probability of false positive.

detection rate value but one false positive to many detection rate values. This consideration was not considered initially in the ROC construction [9]. However modern applications are concerned with how the characteristics of the target change the probability of detecting it.

A. TRADE-OFF BETWEEN PROBABILITY OF FALSE POSITIVE AND POD

From the noise analysis, it becomes possible to analyze the trade-off between false positive and POD. To introduce the novel approach, a single probability point is analyzed.

Here the 0.9 probability is used and drawn to intercept POD and confidence curves. At the point of intersection: the POD, false positive, threshold, and the process parameter (here: time) are known. Changing the decision threshold changes the probability of false positive and the target sizes. A graph depicting the relationship between the POD, false positive, threshold, and time is shown in Fig. 20. Using the approach illustrated in Fig. 20 it becomes possible to visualize and therefore demonstrate the relationship between the POD, false positive, threshold, and process parameter. This is unique due to the newly introduced extension which is not possible using the common threshold approach or the ROC curve. To analyze other probabilities requires evaluating the specific points as is done for the 90 % probability. The introduced method presents a novel and significant approach to concurrently examine all properties affecting the classification results.

B. COMPARISON OF CLASSIFICATION APPROACHES INCORPORATING POD-BASED NOISE ANALYSIS

Applying the introduced POD-based noise analysis, ML approaches can now be analyzed in more detail. For a comprehensive comparison between the POD of two classification approaches, their corresponding PFP needs to be computed. A comparison is made between HMM (Fig. 21) and HMM Imp (Fig. 22) left lane change to illustrate the new approach.

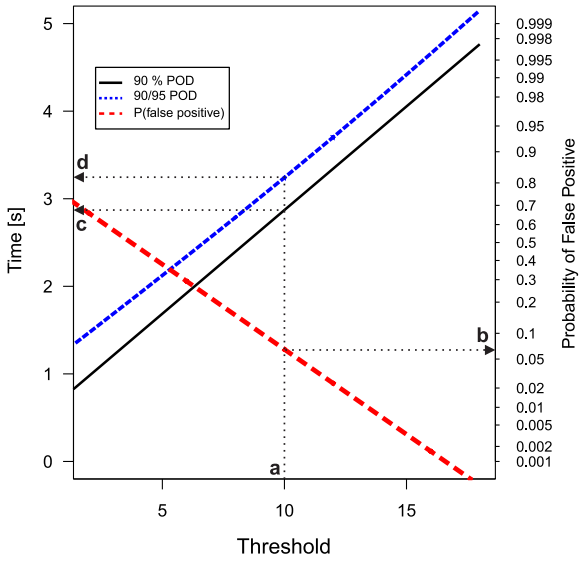

FIGURE 21. Trade-off for HMM left lane change.

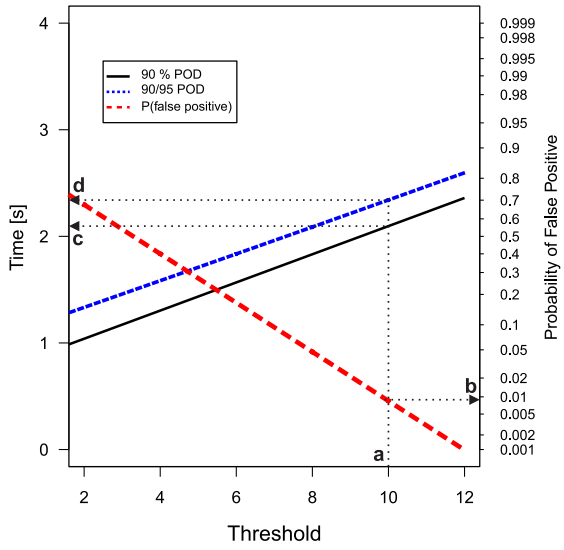

FIGURE 22. Trade-off for HMM-Imp left lane change.

For a selected decision threshold of 10 corresponding to point *a*, the HMM classifier (see Fig. 21) produces a false positive of 0.066 while the improved HMM classifier (see Fig. 22) produces a false positive value of 0.009 corresponding to point *b*. The 90 % POD for HMM and improved HMM are 2.8707 *s* (Fig. 21) and 2.096 *s* (Fig. 22) respectively, corresponding to point *c*. The 90/95 POD for HMM and improved HMM are 3.24 *s* (Fig. 21) and 2.341 *s* (Fig. 22) respectively, corresponding to point *d*. Table 4 summarizes the 90 % and 90/95 POD and PFP values for varying decision thresholds.

For a selected decision threshold of 2 (see Table 4) it is seen that HMM Imp has a better 90/95 POD (1.333 *s*) compared to HMM (1.472 *s*). However HMM Imp has worse PFP value (0.677 *s*) compared to HMM (0.660 *s*) for the same

TABLE 4. POD and PFP values for different thresholds.

DT	HMM 90% POD	HMM 90/95 POD	HMM PFP	HMM Imp 90% POD	HMM Imp 90/95 POD	HMM Imp PFP
0	0.504	1.043	0.814	0.775	1.084	0.879
2	0.977	1.472	0.660	1.039	1.333	0.677
4	1.451	1.905	0.473	1.304	1.584	0.400
6	1.924	2.344	0.292	1.568	1.835	0.168
8	2.397	2.788	0.152	1.832	2.088	0.047
10	2.871	3.241	0.066	2.096	2.341	0.009
12	3.344	3.704	0.023	2.360	2.596	0.001

Legend- DT: decision threshold, PFP: probability of false positive

threshold value of 2. Therefore comparing classifiers using solely the 90/95 POD values (common threshold approach) is not conclusive. A fair comparison will require comparing the corresponding PFP values. For a threshold value of 4 and above, HMM Imp has a better POD and PFP value compared to HMM. Also from this discussion it can be seen that the new approach provides a more comprehensive comparison detailing the associated false positive for any selected decision threshold. The earlier comparison using solely the 90/95 evaluation metric and a common threshold does not allow this detailed analysis.

VII. GENERALIZATION OF THE POD APPROACH

The introduced approach is based on a linear regression (eg: 11). However, not all data will assume a linear structure. The data distribution can be binary or nonlinear. In such situations different implementation strategies may apply but the core idea remains. For binary data, the log-odds (logit) distribution is found to be of good fit [10] by linking the binary response to the explanatory variables through the probability of either outcome, which varies continuously from 0 to 1. The POD function for binary data can be expressed as shown in [9] as

$$POD = \frac{e^{\frac{\pi}{\sqrt{3}}(\frac{\ln a-\mu}{\sigma})}}{1+e^{\frac{\pi}{\sqrt{3}}(\frac{\ln a-\mu}{\sigma})}}. \tag{13}$$

Equation 13 can be conveniently written in the form

$$POD = \frac{e^{(\alpha+\gamma \ln a)}}{1+e^{(\alpha+\gamma \ln a)}}. \tag{14}$$

Here the parameters α and γ are related to μ and σ by

$$\mu = -\frac{\alpha}{\gamma} \tag{15}$$

$$\sigma = \frac{\pi}{\gamma\sqrt{3}} \tag{16}$$

The POD function for discrete target response data can be expressed as shown in [10] as

$$POD(a) = Probability(\ln(\hat{y}) > \ln(y_{th})). \tag{17}$$

Equation 17 represents the area contained between the probability density function of $\ln(\hat{y})$ and above the decision threshold $\ln(y_{th})$. The POD can be expressed as

$$POD(a) = 1 - \Phi\left[\frac{\ln(y_{th})-(\alpha+\beta \ln(a))}{\sigma_\tau}\right]. \tag{18}$$

For continuous data, a continuous random variable X can assume any value in a particular interval rather than any value from a set of discrete values. Therefore, for non-discrete data it is necessary to define a continuous function to describe the probability distribution of X. This function is the continuous probability density function $f(X)$ and it is defined within the interval $-\infty < x < \infty$. The random variable X is defined in terms of $f(x)$ as shown in [22] as

$$P(-\infty < X < \infty) = \int_{-\infty}^{\infty} f(x)dx = 1. \tag{19}$$

The cumulative distribution function (CDF) $\Phi(x)$ for continuous data is calculated as

$$\Phi(x) = \int_{-\infty}^{x} f(t)dt = P(X \leq x) \, for \, (-\infty < x < \infty). \tag{20}$$

The CDF function $\Phi(x)$ can be substituted in equation 18 in the case of a continues variable.

The outlined formulae are important to generate a POD that accurately defines the data. The aim is to provide precise information on POD curves, which will be useful for the Machine learning community, particularly in safety-critical applications.

VIII. CONCLUSION

In this research, our recently proposed contribution for certifying and comparing machine learning methods using the common threshold approach is extended to assess the effects of varying decision thresholds and false positives on the likelihood of detection. The new approach is based on the extension of the POD approach and is demonstrated on experimental data from a driving simulator. The simulator has sensor components providing aids for vehicles to measure dynamical and complex driving environments. The data from the multi-sensor system are fused by classification. Applying ML algorithms decisions/statements about the lane change intentions are made and subsequently evaluated using the modified POD. The introduced approach permits the comparison of different ML algorithms thereby aiding in the selection of desirable approach for a specific task by considering the detection probabilities and false positive rates. To achieve this, a noise analysis procedure is developed that simultaneously considers the trade-off between the probability of false positive, POD, decision threshold, and process parameter. Based on the new approach, improved HMM and conventional HMM are compared, and a visual representation of how the POD and PFP change with varying thresholds is presented. The introduced approach is an alternative to the ROC with the extra advantage of additionally evaluating the effect of process parameter on the classification results and therefore useful for sequence problems requiring the evaluation of temporal dynamic behavior.

REFERENCES

[1] J. Bekker and J. Davis, "Learning from positive and unlabeled data: A survey," *Mach. Learn.*, vol. 109, no. 4, pp. 719–760, Apr. 2020.

[2] F. Carcillo, Y.-A. Le Borgne, O. Caelen, Y. Kessaci, F. Oblé, and G. Bontempi, "Combining unsupervised and supervised learning in credit card fraud detection," *Inf. Sci.*, vol. 557, pp. 317–331, May 2021.

[3] I. González-Carrasco, J. L. Jiménez-Márquez, J. L. López-Cuadrado, and B. Ruiz-Mezcua, "Automatic detection of relationships between banking operations using machine learning," *Inf. Sci.*, vol. 485, pp. 319–346, Jun. 2019.

[4] K. Bowyer, C. Kranenburg, and S. Dougherty, "Edge detector evaluation using empirical ROC curves," *Comput. Vis. Image Understand.*, vol. 84, no. 1, pp. 77–103, Jan. 2001.

[5] D. Lorente, N. Aleixos, J. Gómez-Sanchis, S. Cubero, and J. Blasco, "Selection of optimal wavelength features for decay detection in citrus fruit using the ROC curve and neural networks," *Food Bioprocess Technol.*, vol. 6, no. 2, pp. 530–541, Feb. 2013.

[6] Y. Su and F. Jurie, "Improving image classification using semantic attributes," *Int. J. Comput. Vis.*, vol. 100, no. 1, pp. 59–77, 2012.

[7] D. A. Ameyaw, Q. Deng, and D. Söffker, "Probability of detection (POD)-based metric for evaluation of classifiers used in driving behavior prediction," in *Proc. Annu. Conf. PHM Soc.*, vol. 11, 2019, pp. 1–7.

[8] C. Annis, "Statistical best-practices for building probability of detection (POD) models," R Package mh1823, Version, 4(6.0.4), Stat. Eng., Tallahassee, FL, USA, Tech. Rep. 4, 2020.

[9] *Department of Defense Handbook: Nondestructive Evaluation System Reliability Assessment*, MIL-HDBK-1823A, Dept. defense USA, Wright-Patterson AFB, Virginia, USA,2009.

[10] G. A. Georgiou, "PoD curves, their derivation, applications and limitations," *Insight-Non-Destructive Test. Condition Monitor.*, vol. 49, no. 7, pp. 409–414, Jul. 2007.

[11] E. Ginzel, "Introduction to the statistics of NDT," *NDE. Net-E-J. Nondestruct. Test.*, vol. 11, no. 5, pp. 4–7, 2006.

[12] J. S. Knopp, J. C. Aldrin, E. Lindgren, and C. Annis, "Investigation of a model-assisted approach to probability of detection evaluation," in *Proc. AIP Conf.*, 2007, pp. 1775–1782.

[13] R. B. Thompson, L. H. Brasche, D. Forsyth, E. Lindgren, P. Swindell, and W. Winfree, "Recent advances in model-assisted probability of detection," *Proc. 4th Eur.-Amer. Workshop Rel. NDE*, Berlin, Germany, Jun. 2009.

[14] C. A. Harding, G. R. Hugo, S. J. Bowles, D. O. Thompson, and D. E. Chimenti, "Application of model-assisted pod using a transfer function approach," in *Proc. AIP Conf.*, 2009, pp. 1792–1799.

[15] B. Chapuis, P. Calmon, and F. Jenson, *Best Practices for the Use of Simulation in POD Curves Estimation*. Berlin, Germany: Springer, 2018.

[16] D. A. Ameyaw, Q. Deng, and D. Söffker, "How to evaluate classifier performance in the presence of additional effects: A new POD-based approach allowing certification of machine learning approaches," *Mach. Learn. Appl.*, vol. 7, Mar. 2022, Art. no. 100220.

[17] Q. Deng and D. Soffker, "Classifying human behaviors: Improving training of conventional algorithms," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 1060–1065.

[18] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[19] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[20] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, and C. A. C. Coello, "A survey of multiobjective evolutionary algorithms for data mining: Part I," *IEEE Trans. Evol. Comput.*, vol. 18, no. 1, pp. 4–19, Feb. 2014.

[21] L. Gandossi and C. Annis, "Probability of detection curves: Statistical best-practices," Eur. Commission, Joint Res. Centre, Inst. Energy Petten, The Netherlands, Tech. Rep. ENIQ 41, 2010.

[22] M. H. KutnerC, J. Nachtsheim, J. Neter, W. Li, *Applied Linear Statistical Models*, vol. 5. New York, NY, USA: McGraw-Hill, 2005.

DANIEL ADOFO AMEYAW received the Dr.-Ing. degree in mechanical engineering from the University of Duisburg-Essen, Campus Duisburg, Germany, in 2020. He is currently a Postdoctoral Researcher with the Chair of Dynamics and Control, University of Duisburg-Essen. His research interests include structural analysis, vibration analysis of elastic structures, and developing reliability evaluation methods for machine learning approaches.

QI DENG received the Dr.-Ing. degree in mechanical engineering from the University of Duisburg-Essen, Campus Duisburg, Germany, in January 2021. Until recently, she was a Researcher and a Scientific Co-Worker with the Chair of Dynamics and Control, University of Duisburg-Essen. Her current research interests include modeling and prediction of driving behaviors and applications of machine learning methods.

DIRK SÖFFKER (Member, IEEE) received the Dr.-Ing. degree in mechanical engineering and the Habilitation degree in automatic control/safety engineering from the University of Wuppertal, Germany, in 1995 and 2001, respectively. Since 2001, he has been the Chair of Dynamics and Control, University of Duisburg-Essen, Germany. His current research interests include diagnostics and prognostics, modern methods of control theory, safe human interaction with technical systems, safety and reliability control engineering of technical systems, and cognitive technical systems.