# A Case Study Comparing Binary Classifier Characteristic Curves for Imbalanced Data

Daniel Watson[1], Dr. Karl Reichard[2], and Aaron Isaacson[3]

[1,2,3] *Pennsylvania State University, State College, PA, 16802, United States*
*duw428@psu.edu*
*kmr5@psu.edu*
*aci101@psu.edu*

**ABSTRACT**

Receiver operating characteristic curves are a mainstay in binary classification and have seen widespread use from their inception characterizing radar receivers in 1941. Widely used and accepted, the ROC curve is the default option for many application spaces. Building on prior work the Prognostics and Health Management community naturally adopted ROC curves to visualize classifier performance. While the ROC curve is perhaps the best known visualization of binary classifier performance it is not the only game in town. Authors from across various STEM fields have published works extolling various other metrics and visualizations in binary classifier performance evaluation. These include, but are not limited to, the precision recall characteristic curve, area under the curve metrics, bookmaker informedness and markedness. This paper will review these visualizations and metrics, provide references for more exhaustive treatments on them, and provide a case study of their use on an imbalanced prognostic health management data-set. Prognostic health management binary classification problems are often highly imbalanced with a low prevalence of positive (faulty) cases compared to negative (nominal/healthy) cases. In the presented data-set, time domain accelerometer data for a series of run-to-failure ball-on-disk scuffing tests provide a case where the vast majority of data, $> 94\%$, is from nominally healthy data instances. A condition indicator algorithm targeting the hypothesized physical system response is validated compared to less informed classifiers. Several characteristic curves are then used to showcase the performance improvement of the physics informed condition indicator.

## 1. INTRODUCTION TO BINARY CLASSIFICATION

Binary classification is often encountered in the field of Prognostic Health Management (PHM), machine learning, medical sciences, and information retrieval where there are two mutually exclusive classes, positives (**p**) and negatives (**n**), and the user wishes to accurately predict the class of each instance. During development of the binary classifier a supervised learning approach is utilized where the ground truth class of each instance is known. The binary classifier predicts the class that each instance belongs to by predicting that yes (**Y**), the data is **p**, or no (**N**) the data is not **p**. The true and predicted classifications are then mapped onto the 2x2 confusion matrix shown in Figure 1. The following six (6) variables are then obtained from the confusion matrix and used to derive assorted figures of merit (Takaya & Rehmsmeier, 2015; Fawcett, 2006; Powers, 2008).

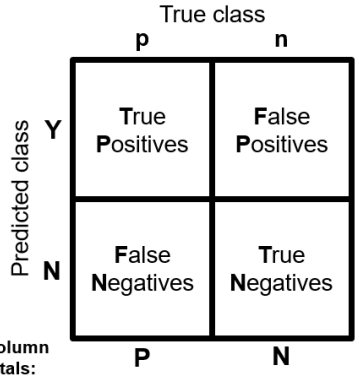


Figure 1. Confusion matrix for binary classifiers.

1. **condition positive (P):** The number of real positive cases in the data
2. **condition negative (N):** The number of real negative cases in the data
3. **true positive (TP):** The number of classification results that correctly predict a positive true class
4. **true negative (TN):** The number of classification results that correctly predict a negative true class

5. **false positive (FP):** The number of classification results that falsely predict **Y** when the true class is negative
6. **false negative (FN):** The number of classification results that falsely predict **N** when the true class is positive

### 1.1. Continuous and Discrete Classifiers

Binary classifiers are generally divided into two types based on the form of their output, continuous and discrete outputs. A continuous classifier outputs a value (e.g. probability) for each evaluated instance, the value output is then compared to a threshold value; values above the threshold are predicted positive (**Y**) while values below the threshold are predicted negative (**N**). Continuous classifiers are also broadly referred to as probabilistic classifiers (Fawcett, 2006), even when the output isn't strictly a probability and may not be bound between $[0, 1]$ . Discrete classifiers operate more as a black box, data on the instance is provided as input and the discrete classifier returns one of two states, predicted positive (**Y**) or predicted negative (**N**). Discrete classification is very common in machine learning applications; however, if the researcher has the ability to see inside the machine learning algorithm and use internally generated values as the input to the binary classifier the discrete machine learning classifier can be treated as a continuous classifier (Fawcett, 2006). It is also possible to use an ensemble of discrete classifiers with weighted scores and majority voting to produce a continuous output. In the case study that follows this introduction continuous classifiers using figures of merit calculated from accelerometer data.

### 1.2. Binary Classifier Metrics

The confusion matrix of Figure 1 is the cornerstone of classifier performance evaluation. For a discrete classifier there is only one state for the confusion matrix, the classifier is applied to the set of instances, predictions are compared to the ground truth classes, and the six variables of the confusion matrix are tallied. Key metrics such as those shown in Table 1 are then calculated for the single confusion matrix state.

For continuous and pseudo-continuous classifiers the prediction of (**Y**) or (**N**) is dependent on the threshold value. At one extreme, all classifier outputs are greater than the threshold value and every instance is predicted (**Y**). At the other extreme of threshold values all classifier outputs are less than the threshold output and every instance is predicted (**N**). Between the minimum and maximum thresholds are confusion matrix states where a subset of instances are predicted (**Y**). Each confusion matrix state represents a singular value of the metrics listed in Table 1 and defined by the performance metric Eqs. (1 through 9) (Takaya & Rehmsmeier, 2015; Fawcett, 2006; Powers, 2008).

Table 1. Binary classifier performance metrics.

| Metric | Common nomenclature |
|---|---|
| **prevalence** | N/A |
| **TPR** | true positive rate, sensitivity, recall, or hit rate |
| **TNR** | true negative rate, specificity, or selectivity |
| **PPV** | positive predictive value or precision |
| **FPR** | false positive rate or fall-out |
| **ACC** | accuracy |
| **NPV** | negative predictive value |
| **BM** | bookmaker informedness or informedness |
| **MK** | markedness or deltaP ($\Delta P$) |

- **prevalence**, is the fraction of total instances that are positive (**p**). A balanced data-set will have a prevalence of $0.5$ while imbalanced data-sets can range widely with values between $[0, 0.5)$ represented in real world data-sets (Fawcett, 2006). While a minimal value of $10^{-6}$ was cited in prior literature, it is feasible for a prevalence of zero $(0)$ to occur when no positive instances have occurred in a data-set.

$$prevalence = \frac{P}{P+N} \tag{1}$$

- **TPR**, true positive rate, hit rate, recall, or sensitivity is the fraction of positive (**p**) instances accurately predicted as positive (**TP**) instances.

$$TPR = \frac{TP}{P} = \frac{TP}{TP+FN} \tag{2}$$

- **TNR**, true negative rate, specificity, or selectivity is the fraction of negative (**n**) instances accurately predicted as negative (**TN**) instances.

$$TNR = \frac{TN}{N} = \frac{TP}{TN+FP} \tag{3}$$

- **PPV**, positive predictive value or precision is the rate at which predicted (**Y**) instances are true positives (**TP**).

$$PPV = \frac{TP}{TP+FP} \tag{4}$$

- **FPR**, false positive rate or fall-out is the fraction of negative (**n**) instances incorrectly predicted as positive (**FP**) instances.

$$FPR = \frac{FP}{N} = \frac{FP}{FP+TN} \tag{5}$$

- **NPV**, negative predictive value is the rate that predicted (**N**) instances are true negatives (**TN**).

$$NPV = \frac{TN}{TN + FN} \tag{6}$$

- **ACC**, accuracy is the rate that an instance of either true class was accurately predicted.

$$ACC = \frac{TP + TN}{P + N} = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

- **BM**, bookmaker informedness, or informedness is a higher level abstraction compared to Eqs. (1 through 7). Bookmaker informedness combines both true positive rate and true negative rate to provide a singular scaler value on the classifier's ability to predict the true class of negative (**n**) and positive (**p**) instances. Powers (Powers, 2008) states that 'Informedness quantifies how informed a predictor is for the specified condition, and specifies the probability that a prediction is informed in relation to the condition (versus chance).' Unlike the previously defined metrics informedness is not a rate bound between $[0, 1]$ and is instead a range between $[-1, 1]$ with a value of $+1$ indicating a perfectly informed classifier and negative values indicating the classifier should be inverted such that any predicted (**Y**) instances should be treated as (**N**) and vice versa.

$$BM = TPR + TNR - 1 = \frac{TP}{P} + \frac{TN}{N} - 1 \tag{8}$$

- **MK**, markedness is a higher level abstraction similar to informedness, this time combining positive predictive value and negative predictive value. A perfect classifier will return zero (0) false positives (**FP**) resulting in a PPV of one (1) and similarly zero (0) false negatives (**FN**) resulting in a NPV of one (1). Markedness is therefore maximized when there are no false predictions. Powers (Powers, 2008) also provides a helpful description of markedness, "Markedness quantifies how marked a condition is for the specified predictor, and specifies the probability that a condition is marked by the predictor (verses chance)." Definitions provided by Powers for informedness and markedness are both derived from bookmaking (the setting of betting odds, e.g. for horse racing).

$$MK = PPV + NPV - 1 = \frac{TP}{TP + FP} + \frac{TN}{TN + FN} - 1 \tag{9}$$

### 1.3. Data Balance

Data balance, or conversely imbalance, is a measure of the binary class distribution. A data-set with near equal proportions of (**n**) and (**p**) instances is considered a balanced data-set and will have a prevalence, as calculated by Eq. (1), of approximately $0.5$. Strictly speaking, an imbalanced data-set could have a value between $[0, 0.5)$ or $(0.5, 1.0]$, for the sake of this discussion we will assume that nominal or (**n**) instances are in the majority for any cases of imbalanced data. In literature, data balance is a major consideration when choosing which metrics and data visualizations to use (Powers, 2008; Takaya & Rehmsmeier, 2015). Of particular importance for comparing ROC and precision recall characteristic (PRC) curves is the effect of data balance on their underlying metrics of TPR, FPR, and PPV (precision) given by Eqs. (2, 4, and 5).

Referring to the confusion matrix of Figure 1, the TPR is only a function of the left hand column, the TPR which is used by both ROC and PRC curves is agnostic to data balance. Likewise the FPR is a function of right hand column variables. For both TPR and FPR used in the ROC curve the relative values of (**P**) and (**N**) do not effect the metric. Precision, or PPV, is different, the (**TP**) instances are divided by the sum of (**TP**) and (**FP**), the metric is dependent on the relative proportion of both columns and is therefore sensitive to class balance. This sensitivity is key when comparing ROC and PRC curve performance for imbalanced data-sets.

## 2. CLASSIFIER OPTIMIZATION

For a discrete classifier, once a performance metric to optimize is identified, comparisons between discrete classifiers is straightforward. Choosing a performance metric for discrete classifiers is by no means trivial and is often domain specific. Medical screening tests may prefer a high TPR at the expense of increased FPR - in the PHM community this is termed a liberal classifier - on the reasoning that more resource intensive and invasive follow-on testing will eliminate the initial false positives (**FP**) while minimizing false negatives (**FN**). A contemporary example is the desire to screen populations for a viral pandemic, better to presumptively identify edge cases as predicted (**Y**) than to let a symptomatic case spread.

A contrary example was provided as an anecdote during the 2022 Analytics for PHM Short Course. One automaker, when developing and fielding PHM products, their engineers were very cautious to avoid false positives (**FP**) based on customer and technician backlash from unnecessary maintenance. The automaker's approach was to focus on a high positive predictive value, defined by Eq. (4). When contacting a customer about a predicted fault, they wanted confidence that the predicted (**Y**) was a (**TP**) (Eklund, 2022).

A typical next step in classifier optimization for a business is to add cost models on top of their classifier metrics. More

detailed treatments on integrating cost data into classifier optimization can be found in (Bradley, 1997) and (Takaya & Rehmsmeier, 2015). While not covered in this work the cost modeling aspect of classifier evaluation is a critical next step.

For continuous classifiers, with each threshold value representing a confusion matrix state, the dimensionality of the comparison is increased and rapidly becomes overwhelming to interpret without some combination of visualization tools and reduction of threshold sweeps to scalar values.

### 2.1. Characteristic Curves and Area Under the Curve

Moving beyond discrete classifiers, the interpretation of continuous classifiers presents a new level of difficulty. Individual discrete classifiers produce a single value output for the performance metrics of Table 1, but continuous classifiers produce outputs for each classifier model and the models range of threshold values. In response to this difficulty several visualizations or curves have become standard approaches to review classifier performance.

#### 2.1.1. Receiver Operating Characteristic Curve

Receiver operating characteristic (ROC) curves are one of many visualization methods for binary classifier performance based on figures of merit derived from the confusion matrix shown in Figure 1.ROC curves are plots showing the rate at which positives are correctly identified, the true positive rate (TPR), on the Y-axis against the rate negatives are incorrectly classified, the false positive rate (FPR) on the X-axis. These rates are calculated using Eqs. (2 and 5). For a given binary classifier, scaler values are calculated for each sample, the scaler values are then compared to a monotonically increasing threshold value, TPR's and FPR's are calculated using Eqs. (2 and 5), and finally the two-dimensional ROC curve is plotted.

ROC curves have found widespread use in disparate areas: from quantitizing radar operator performance in the second world war, to medical decision making, and machine learning applications. No matter the field, the ROC curve has been used to evaluate the performance of binary classifiers - classifiers that predict the division of a data-set into two groups, positives (**p**) and negatives (**n**). Continuous or probabilistic classifiers, which provide a score or probability, can be applied over a range of classification threshold values to provide the data necessary to generate a multi-point ROC curve (Fawcett, 2006; Takaya & Rehmsmeier, 2015; Powers, 2008).

For a binary classifier with an even $50\%$ chance of correctly identifying the correct class in a balanced data-set the expected ROC curve is shown in Figure 2 which was generated using a $1000$ sample data-set where the classifier had equal chance to correctly or incorrectly predict the class. This represents a worst case classifier performance, as typically the classifier predictions would be inverted if it was more prone to the incorrect classification. To interpret Figure 2 imagine a classifier that produces a score between $(0.0, 1.0)$, with higher values nominally correlated with higher probability of a positive (**p**) class, and a vector of monotonically increasing threshold values from $[0.0, 1.0]$. When the threshold value is at a minimum everything is a predicted (**Y**) and therefore FPR is at a maximum. When the threshold value that delineates a predicted (**N**) and predicted (**Y**) is increased, a new point on the ROC curve is generated with updated FPR and TPR. For a random classifier the score is uncorrelated to class, and higher threshold values increase the number of true positives and false positives at an equal rate. This continues until the threshold value exceeds the greatest class score and every result is predicted (**N**) and the value of TPR and FPR converge to $0.0$. The number of threshold values used to generate the ROC curve is application specific, but in general should have sufficient quantity and spacing to accurately visualize the curvature and any knee points in the ROC curve. The relative spacing of threshold values is also application specific, the author has had success using both linear and logarithmically increasing threshold values. In (Fawcett, 2006) an ROC generation algorithm is presented which sorts the classifier outputs by amplitude and for each unique amplitude calculates the requisite performance metrics instead of using a pre-defined threshold set, depending on the application this alternative method may provide computational efficiency.

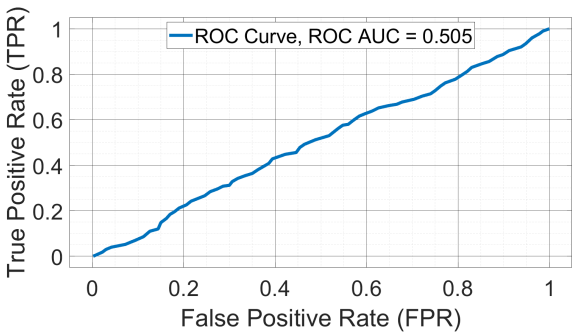


Figure 2. ROC curve for random choice classifier.

A random choice continuous classifier will produce results comparable to Figure 2. On the other extreme of classifier performance is a binary classifier that perfectly classifies every instance resulting in $FPR = 0$, $TPR = 1$, and a point in ROC space at coordinates $[0, 1]$ (Bradley, 1997; Fawcett, 2006; Powers, 2008; Takaya & Rehmsmeier, 2015). A perfect classifier as described will produce an ROC curve with two perpendicular legs as shown in Figure 3.

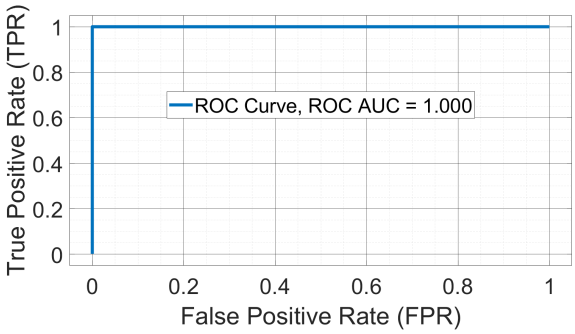


Figure 3. ROC curve for a perfect classifier.

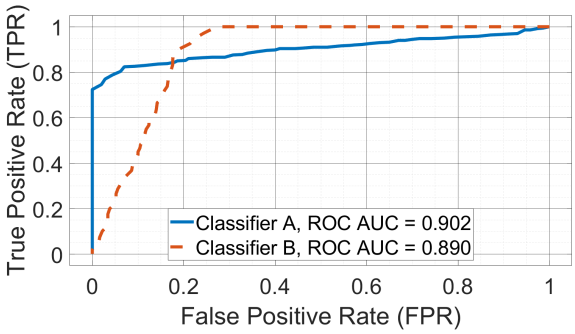


Figure 4. ROC curves for a conservative and liberal classifier.

#### 2.1.2. Evaluation of Classifier Performance via Area Under the Curve

The comparison of Figures 2 and 3 brings us to an important subject - using the ROC curve to objectively evaluate classifier performance - and introduces another figure of merit, the *area under the ROC curve* (AUC) (Bradley, 1997). As the ROC curve is plotted on a unit square, the corresponding AUC is bound between $0.0$ and $1.0$. For the random classifier of Figure 2 the expected AUC is $0.50$ and in the case of this specific $1000$ sample random data-set $AUC = 0.493$. Alternatively the perfect classifier of Figure 3 results in $AUC = 1.0$. These contrasting classifiers illustrate an attractive feature for the ROC AUC, the AUC is equivalent to the probability that a random instance of class (**n**) will produce a classifier output lower than the output of a random instance of class (**p**), known in the field of statistics as the Mann-Whitney U statistic (Hand, 2009).

The prior paragraph paints a rosy picture of using ROC AUC. Figure 4 is included to urge caution before blindly using ROC AUC as the sole metric in choosing a classifier. In Figure 4 classifier A is a simulated data-set where approximately $10\%$ of the data-set is (**FN**); the prediction is (**N**) while the ground truth is (**p**). Classifier B is the opposite and approximately $10\%$ of the data-set is (**FP**) where the prediction is (**Y**) while the ground truth is (**n**). As seen this produces two drastically different ROC Curves that result in ROC AUC values that are equivalent. Recalling the discussion in Section 2, classifier A is conservative and would appeal to the General Motors use case where (**FP**) is to be minimized and a threshold that maximized TPR while $FPR = 0$ is optimal. Classifier B is an example of a liberal classifier, it prioritizes identifying all (**p**) cases at the expense of greater (**FP**), classifier B would be preferred in the medical screening classifier also discussed in Section 2. We will revisit the four classifiers of Figures 2, 3, and 4 during the discussion of an alternative to ROC curves.

Much has been written on alternative measures (Bradley, 1997; Powers, 2008; Hand, 2009; Takaya & Rehmsmeier, 2015; Fawcett, 2006). The AUC is just one measure of classifier performance and can be applied to other permutations of characteristic curves such as the Precision Recall Characteristic (PRC) curve.

#### 2.1.3. Precision Recall Characteristic Curve

Due to an unfortunate lack of coordination the PRC curve's relation to the ROC curve is hidden by nomenclature. The PRC curve is a $2D$ visualization of the performance metrics *Precision* (**PPV**, or positive predictive value) and *Recall* (**TPR**). Yes, a double take is required, *Recall* is simply another name for the true positive rate used by the ROC curve. Eqs. (4) and (2) define these two metrics. As a review; the positive predictive value is the rate at which predicted (**Y**) instances are true positives (**TP**), and the true positive rate is the fraction of positive (**p**) instances accurately predicted as positive (**TP**) instances. Further confounding comparisons, the PRC curve places TPR on the X-axis while the ROC curve place the TPR on the Y-axis.

The PRC curve is commonly used for information retrieval classifiers such as developing search engines (Fawcett, 2006; Takaya & Rehmsmeier, 2015). The combination of TPR and PPV at first seem an unlikely pairing as they are similar metrics. TPR is maximized when all (**p**) instances are predicted (**Y**) and PPV similarly has true positives (**TP**) in the equation numerator. The beauty of the PRC curve is that TPR punishes false negatives (**FN**) while PPV punishes false positives (**FP**), the 'optimization' of the PRC curve therefor leads to a balance between identifying all positive instances while minimizing the number of false positives (**FP**).

Before using the PRC curve to visualize classifier performance in a real world data-set the examples provided in the ROC curve section are revisited using PRC curves, Figures 5, 6, and 7 show the results along with the PRC AUC. Note that the ROC AUC equaling the Mann-Whitney U statistic does not apply to the PRC AUC because $ROCAUC \neq PRCAUC$.

The interpretation of the PRC curve differs from the ROC curve, in Figure 2 the random classifier performance was shown as a line on the diagonal of FPR and TPR. This same

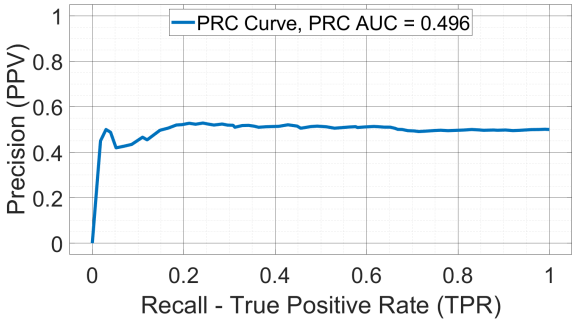


Figure 5. PRC curve for random choice classifier.

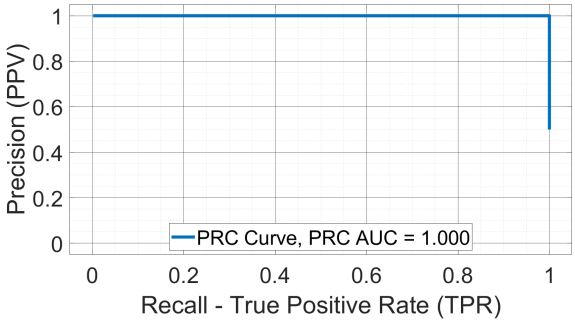


Figure 6. PRC curve for a perfect classifier.

random classifier produces a notably different visualization with the PRC curve. At the far right of the PRC curve a very liberal threshold correctly predicts all positives (**p**) as (**Y**), the value of precision is $0.5$ because there are an equal number of false positives (**FP**) as true positives (**TP**). As the threshold value is increased the [X,Y] pair translates horizontally to the left because true positives (**TP**) decreasing lowers TPR while true positives (**TP**) and false positives (**FP**) are decreasing at the same rate - fixing (within the limits of randomness) the value of precision at $0.5$. As the threshold value of the random classifier continues to increase true positives (**TP**) approach zero $(0)$ and the PRC curve exhibits instability on the far left of the plot. True positives (**TP**) and false positives (**FP**) continue decreasing at the same rate but their sample size decreases until there are not sufficient data points for randomness to average out. In fact depending on how threshold minimum and maximums are established the value of precision at $TPR = 0$ is equally likely $PPV = 0$, as shown in Figure 5, as it is for $PPV = 1$. For reference this paper uses Eqs. (10) and (11) to set the threshold values dynamically for each classifier evaluated. Exact threshold values are then calculated using a linear spacing of $100$ points inclusive of the minimum and maximum values calculated in Eqs. (10) and (11).

$$Threshold_{min} = min(Classifier_{outputs}) \tag{10}$$

$$Threshold_{max} = max(Classifier_{outputs}) \tag{11}$$

The perfect classifier's PRC curve is thankfully more straightforward than the unstable random classifier PRC curve and is shown in Figure 6. With a exclusively liberal threshold value the PRC curve again starts at the coordinate pair $[1, 0.5]$, due to the perfectly balanced class distribution in this synthetic data-set. The PRC Y-axis location when threshold is at a minimum is equal to the prevalence of positive cases in the set. As the threshold value is increased, becoming more conservative, the perfect classifier reaches the PRC space of $[1, 1]$

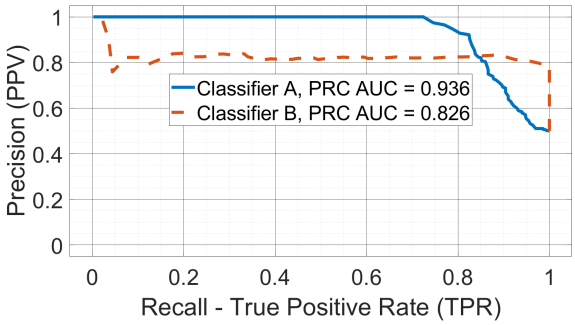


Figure 7. PRC curves for a conservative and liberal classifier.

when all the positive (**p**) and negative (**n**) classes are correctly predicted.

Figure 7 uses the same data-set as Figure 4. To review, classifier A is a simulated data-set where approximately $10\%$ of the data-set is (**FN**); the prediction is (**N**) while the ground truth is (**p**). Classifier B is the opposite and approximately $10\%$ of the data-set is (**FP**) where the prediction is (**Y**) while the ground truth is (**n**). The PRC AUC values for these two classifiers differ by a non-trivial amount, unlike the example with ROC AUC, but ultimately the end user would need to make a decision based on their particular requirements - it is entirely possible that classifier B may be preferred if a high true positive rate is significantly more important than avoiding a few false positives (*FP*).

### 2.2. Bookmaker Informedness and Markedness

In preparing this work, I came across a series of works, (Powers, 2003, 2008; Chicco, Tötsch, & Jurman, 2021), that introduced Bookmaker Informedness and Markedness which at first impression seem well suited to quantifying classifier performance in PHM.

Unlike the well established ROC and PRC curves the sister metrics of informedness (BM) and markedness (MK) were not developed into an easily visualized curve. To demonstrate general trends for (BM) and (MK) the individual metrics are

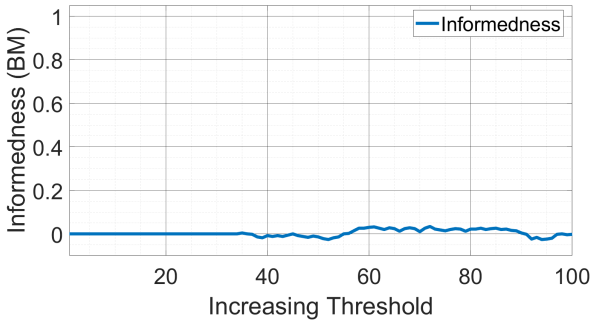


Figure 8. Informedness for random choice classifier.

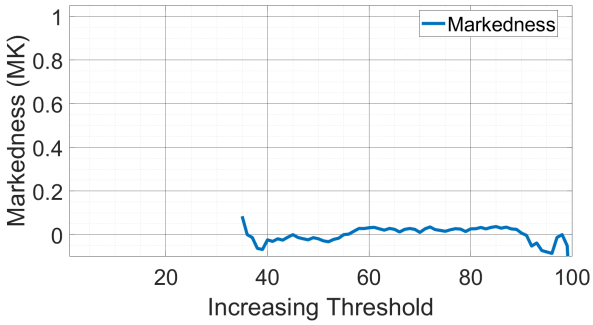


Figure 9. Markedness for random choice classifier.

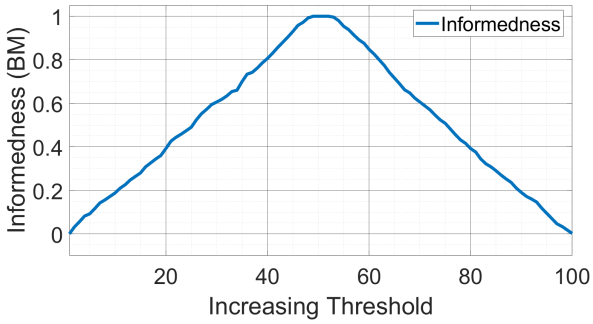


Figure 10. Informedness for a perfect classifier.

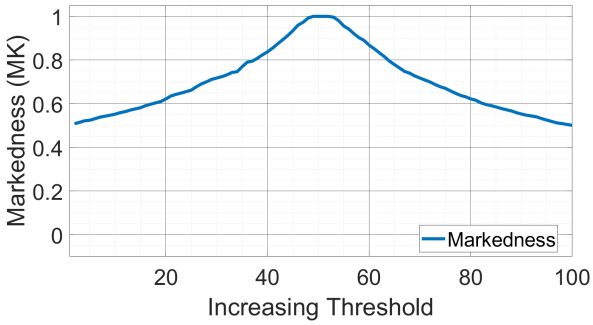


Figure 11. Markedness for a perfect classifier.

plotted verse a linearly increasing threshold for the three synthetic examples previously visited for ROC and PRC curves of Figures 2 through 7. In this work the evaluation threshold maximum is dynamically updated for each classifier. To standardize the X-axis the threshold sample number of the monotonically increasing value is used in place of actual values. First informedness and then markedness are shown for the balanced data-set random classifier in Figures 8 and 9. After reviewing the three synthetic use cases with standalone BM and MK metrics, a combined visualization - the Bookmaker Curve - will be introduced.

Figures 8 and 9 have several noteworthy features. First is the behavior of both figures before the $35^{th}$ threshold value. Prior to threshold value 35 the threshold value was less than the minimum classifier output. The synthetic data-set's balanced nature leads to equal TPR and FPR's and a resulting BM value of $0.0$. The same circumstances lead to no (**FN**) or (**TN**) values prior to the $35^{th}$ threshold value - causing a zero term in the denominator of the NPV calculation of Eq. (6). This $NaN$ is carried into the calculation of MK and is visualized as a missing region in the MK curve of Figure 9. The second noteworthy result of Figures 8 and 9 is their near zero $(0.0)$ amplitude in the threshold region where they are properly defined. This matches the random classifier's uninformed classification scheme. Turning our attention to the case of a perfect classifier, Figures 10 and 11 show the resulting metric curves.

Figures 10 and 11 show similar symmetric behavior as the threshold values are increased from a minimum value of $0.0$ to the classifier's maximum output value. As implied by name, the perfect classifier includes a range of threshold values where each instance is correctly predicted as (**TN**) or (**TP**). Please note 'perfect' does not imply that the classifier performs likewise for any given threshold value, rather the classifier is able to take advantage of some feature of the data-set to separate the instances into accurate discrete classes. For the example data-set used, values between thresholds $[49, 52]$ are in the range of classifier outputs where (**n**) and (**p**) instances are separated..

The third set of figures for the initial visualization of BM and MK stems from two classifiers: classifier A is a simulated data-set where approximately $10\%$ of the dataset is (**FN**); the prediction is (**N**) while the ground truth is (**p**). Classifier B is the opposite and approximately $10\%$ of the data-set is (**FP**) where the prediction is (**Y**) while the ground truth is (**n**). Figure 12 shows the informedness as a function of increasing threshold value for the two classifiers while Figure 13 visualizes the markedness.

An interesting contrast between bookmaker informedness and markedness is their response to a biased classifier. Recall

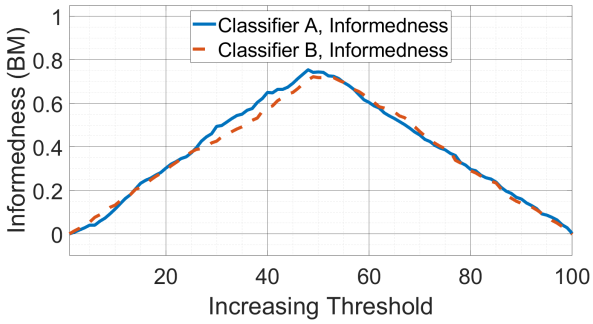


Figure 12. Informedness for a conservative and liberal classifier.

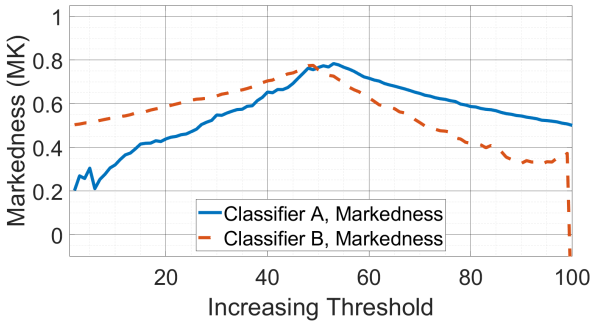


Figure 13. Markedness for a conservative and liberal classifier.

that classifier A is an idealized conservative classifier with $10\%$ of the balanced data-set instances set as (**FN**). The result of this is that $20\%$ of (**p**) instances are misclassified by classifier A. Classifier B, as a liberal classifier, is the inverse - $10\%$ of the balanced data-set instances are (**FP**). Figure 12, which tracks informedness as the threshold is varied is agnostic to classifier permissiveness, showing near identical curves. Conversely Figure 13, which tracks markedness responds to the classifier differences. Tracking classifier A performance, the peak of markedness is reached at a higher threshold than classifier B and their slopes fall off asymmetrically and in opposing skews. A deeper investigation into the markedness metric is reserved for now.

The dual metrics of informedness and markedness have so far been viewed in isolation, making interpretation markedly different than the ROC and PRC curves that are familiar territory in PHM. To thematically match the ROC and PRC curves and hopefully aid in their interpretation the use of a visualization I'll coin the *Bookmaker curve*. In the Bookmaker curve the informedness is plotted on the X-axis while the Y-axis is reserved for markedness. The three established examples are revisited using this new visualization in Figures 14 through 16.

The random classifier Bookmaker curve, shown in Figure 14,

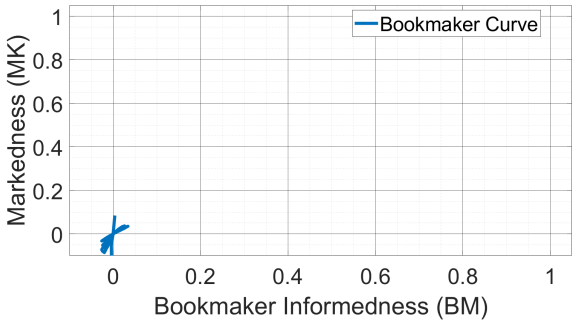


Figure 14. Bookmaker Curve for random choice classifier.

displays a non-linear curve that randomly oscillates around the coordinate pair $[0, 0]$ and indicates that the classifier predictions are not acting on any foresight of the instance class. Skipping to Figure 16 the pair of classifiers trace similar curves with the values approaching coordinate pair $[1, 1]$ at the optimal threshold values. Not shown in the still-frame of Figure 16 is the order in which the respective curves are plotted with increasing threshold values. At the lower bound of threshold values classifier A starts at $[0.0, 0.2]$ and traces counterclockwise until the maximum threshold value at which point the curve is at $[0.0, 0.5]$. The Classifier B trace follows a clockwise path as threshold values are monotonically increased. The non-linear curves are reminiscent of hysteresis loops often encountered in materials science and engineering mechanics and indicate a non-symmetric behavior about a peak.

Returning to Figure 15, the behaviors of the random classifier and imperfect classifiers frame the discussion. Clearly the classifier has some measure of insight into the system to properly predict the true class of instances, unlike the random classifier. The hysteresis loops of Figure 16 are not present in the perfect classifier (for a synthetic data-set and balanced data), but our prior experience allows us to understand the Bookmaker curve is first tracing the curve from $[0.0, 0.5]$ to $[1.0, 1.0]$ before retreating back to the originating coordinate pair as threshold values are increased. Unlike the ROC and PRC curves the Bookmaker does not have a fixed phenotype for a perfect classifier. Depending on the complex interactions of TPR, TNR, PPV, and NPV as threshold is swept the Bookmaker curve may exhibit a wide range of shapes, the only constant is at least one threshold value where the curve is at Bookmaker space of $[1.0, 1.0]$.

The discussion of classifier optimization and visualization sets the stage for analyzing a more complex data-set.

## 3. EXPERIMENT

The remainder of this work will focus on the initial, troubleshooting, run of data collected from a series of run-to-scuffing-failure ball-on-disk (BoD) tests. The initial test run

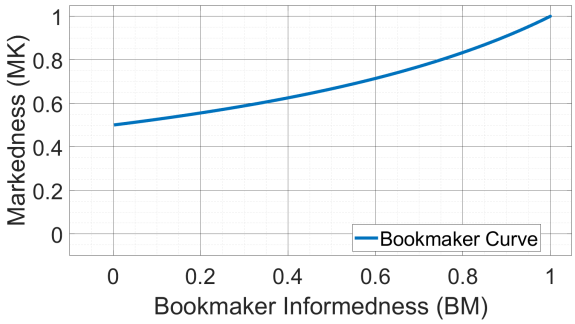


Figure 15. Bookmaker Curve for a perfect classifier.

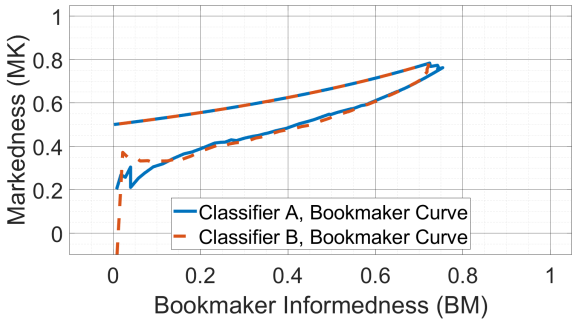


Figure 16. Bookmaker Curve for a conservative and liberal classifier.

used in this work confirmed test procedures and informed planning for the full series of subsequent tests not covered in this paper. Testing was conducted on a Bruker UMT TriboLab at the Penn State University Tribology/Materials Processing Lab. Detection of scuffing, defined in (Ludema, 1984) as "a roughening of surfaces by plastic flow whether or not there is material loss or transfer", has historically been limited to the field of tribology where the conditions at onset of scuffing, as indicated by a rapid increase in coefficient of friction (CoF), are used as a performance metric. Scuffing detection via local CoF is impractical for many applications of prognostic health management (PHM) systems, particularly PHM retroactively installed on legacy systems. The test objectives were to provide bench-top development and validation of scuffing detection algorithms for continuous sliding contact systems such as found on the internals of high pressure diesel fuel pumps. The full problem statement, test description, and analysis are the subject of an in-progress Ph.D. dissertation, but the limited sample used in this work serves the purpose of providing a relevant imbalanced PHM data-set for binary classifier performance metric comparison.

The Bruker UMT TriboLab shown in Figure 17 applies normal force between a rotationally fixed hardened steel ball bearing and the flat face of a rotating hardened steel disk. In this test run the disk was machined from SAE 9310 steel with the wear surface carburized to a Rockwell scale hard-

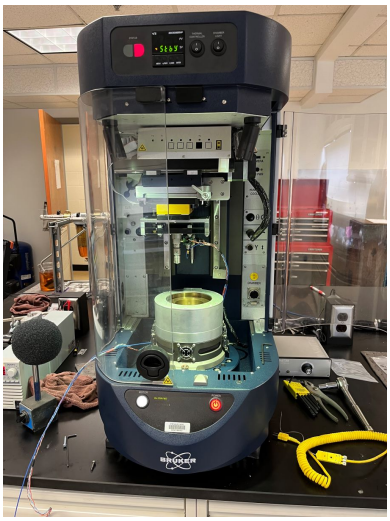

Figure 17. Bruker TriboLab at Penn State University.

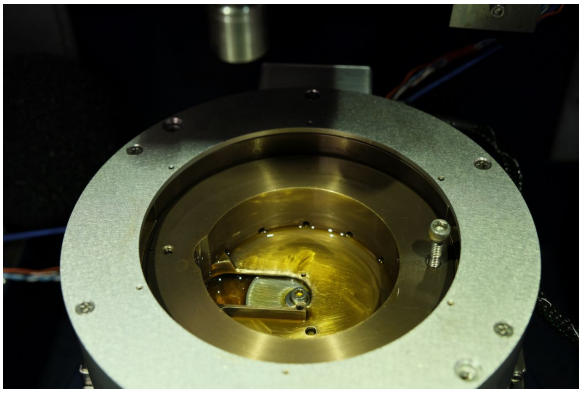

Figure 18. Hardened steel disk spinning inside lubrication bath.

ness of 61 HRC. Each test used a new test disk and new ball bearing. All ball bearings were from the same manufacturing lot of 52100 steel with surface hardness averaging 65 HRC. The disk is held in an oil lubrication bath and spun at a constant $5,000RPM$ as shown in Figure 18. The lubrication was replaced after each run with Mobil Jet Oil II (MIL-PRF-23699 qualified). A stepper motor driven ball screw assembly with PID controller controls normal force between the tribology pair. The load cell, used for data acquisition and PID feedback, has sufficient compliance to allow the ball screw to control force in an otherwise non-compliant assembly. Assembled onto the TriboLab the spindle mechanism is oriented as shown in Figure 19.

The run-to-failure BoD methodology and load profiles used in this testing were previously developed by the Gear Research Institute at Penn State University for analyzing gear steel performance in sliding contact systems such as helical gears. This paper uses data from the initial BoD test. The load profile first brings the test disk up to $5,000RPM$ before

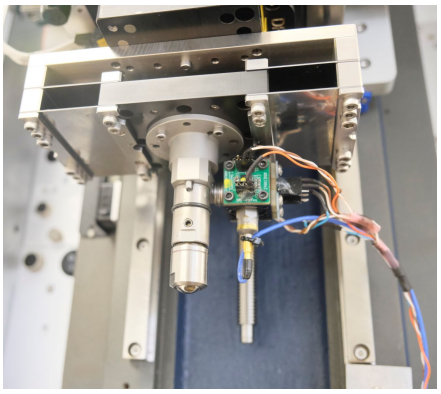

Figure 19. Load cell and spindle mounted on TriboLab.

Table 2. Binary classifier performance metrics.

| Duration (sec) | Description |
|---|---|
| **N/A** | Ramp disk to $5,000RPM$ |
| **2** | Ramp load to $5N$ |
| **600** | Run-in period |
| **1** | Ramp load to $25N$ |
| **600** | First load range |
| **1** | Ramp load to $50N$ |
| **1800** | Second load range |
| **1** | Ramp load to $75N$ |
| **1575** | Third load range |
| **N/A** | Terminate test |

gradually bringing the rotationally fixed ball bearing into contact with the upper face of the test disk. After a short 2 second loading ramp the PID controller held the contact normal force at a nominal $5N$ load for a ten (10) minute run-in period. The full load profile used for the initial BoD test is detailed in Table 2.

### 3.1. Data Collection

Classifiers require an input to predict the class of an instance, in the BoD testing data was collected from two sources. The Bruker TriboLab has built in sensors and feedback that recorded the following relevant data of Table 3 at a $10Hz$ sampling rate.

In addition to the data natively collected by the TriboLab, vibration data collected via accelerometers was identified as key to developing successful scuffing fault classifiers. To enable vibration data collection the upper clamping bolt of the TriboLab spindle was replaced with a through-bolt secured accelerometer mount seen in Figure 19. The accelerometer mount was instrumented with a total of four (4) accelerometers. An IEPE powered PCB 352A60 was installed on the vertical Z-axis while three matching Analog Devices ADXL1005z accelerometers were arranged in a tri-axial configuration. For this work the Z-axis acceleration obtained by the PCB 352A60 is used.

Table 3. Load profile used for initial BoD test.

| Variable | Description |
|---|---|
| **T [sec]** | Elapsed time from test start |
| **Fx [N]** | Reaction force from friction |
| **Fz [N]** | Normal force applied to disk |
| **Z [mm]** | Relative Z-axis position |
| **X [mm]** | Radius of ball contact |
| **V2 [RPM]** | Velocity of disk rotation |
| **Ff [N]** | Derived friction force |
| **COF** | Derived coefficient of friction |

To take advantage of the high bandwidth provided by the PCB 352A60 ($5 - 60kHz$) the maximum sampling rate supported by a NI PCI-4472B card was used - $102,400Hz$. Due to limitations of the legacy data acquisition system used for the initial run data was collected in 25 second records with a 5 second buffer between each record. Subsequent tests used a newer system sampled continuously at $200kHz$ and added a measurement microphone mounted inside the TriboLab test chamber.

### 3.2. Test Observations

The test run for this work's data-set was observed continuously from test initiation through test termination. Lessons learned from this initial test were used to modify future rounds of testing. The test initiated successfully with the disk velocity ramping up to $5,000RPM$ in a counterclockwise direction. The following 2 second load ramp was insufficient time for the TriboLab controller to bring the ball into contact with the spinning disk and ramp the load to $5N$. During the testing load ramps from one load range to the next were originally set to a 1 second duration, this rapid load change led to transient behavior during the testing, subsequent tests would use a uniform 30 second load ramp between loading conditions to allow time for the ball to initially contact the disk and for smoother load ramping dynamics.

Several periods of the test were noteworthy. During the loading ramp from $25N$ to $50N$ there was a brief period of apparent scuffing indicated by increased acoustic noise levels, increased friction force readings from the TriboLab, and increased vibration as measured by the accelerometers. This

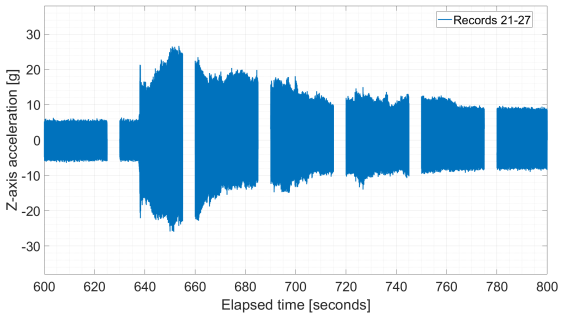


Figure 20. Time domain acceleration during transient.

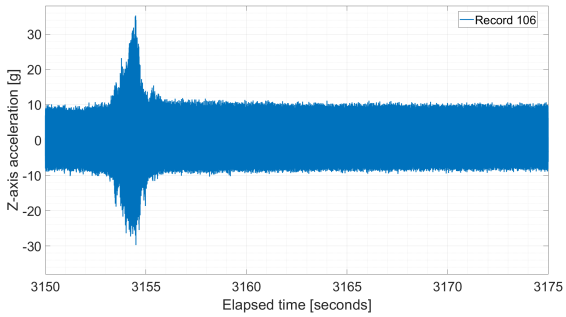


Figure 21. Time domain acceleration spike.

period of increased system response was brief and the system reverted back to a steady state similar to pre-scuffing activity. Figure 20 shows the time domain Z-axis acceleration for this transient event with the initial steady state behavior, scuffing like behavior, and return to a nominal steady state. The time domain acceleration plot covers multiple 25 second records with 5 second gaps due to the limited capabilities of the data acquisition used for this test run. Several future test runs were terminated immediately following similar transient scuffing indicators and support the scuffing classification for this type of temporary elevated outputs.

A challenge encountered during all runs in this experimental setup is establishing a ground truth for active scuffing. The test samples could not be inspected for indications of scuffing until after the test had concluded, as such a ground truth was manually established by comparing contemporaneous notes from auditory cues, TriboLab data, and accelerometer data. While every effort was made to ensure accuracy of the ground truth it remains a significant source of uncertainty in the classifier analysis. For this test a (**n**) or (**p**) class was assigned on a $1Hz$ interval. Due to the transient nature of scuffing events, decreased instance periods could be explored in future work.

After the initial transient scuffing behavior during the $25N$ to $50N$ load ramp there was minimal change in system behavior. 30 minutes into the $50N$ load the $75N$ load ramp was manually triggered. Approximately 2 minutes after the $75N$ load was reached the acoustic, force, and vibration levels spiked for several seconds before once again returning to a steady state baseline. This behavior is shown in the time domain acceleration of record 106 in Figure 21.

The test was allowed to continue running after the brief spike in output variables shown in Figure 21. The test maintained steady state behavior until, suddenly, during record 147 a drastic change in run behavior occurred. Shown in Figure 21, the acceleration rapidly increased to levels greater than seen in the prior transient scuffing. Further, unlike Figure 20, no 'self-healing' of the tribology pair occurred and steady state behavior was never re-established. When it became clear that no return to prior behavior was likely the test was terminated.

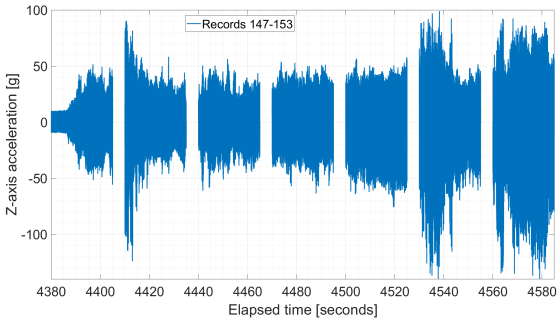


Figure 22. Time domain acceleration at test termination.

Based on later tests results with runs terminated at various stages of the test profile, this last stage of scuffing accounts for the vast majority of surface wear visible on the test disk and ball samples. The worn SAE 9310 carburized steel disk for this run is shown in Figure 23 showing a significant wear scar. In the follow-on testing of 16 test runs the test samples were cleaned of their lubricant film and the wear scars were photographed under a microscope for visual inspection.

At the conclusion of the BoD scuffing test the sample disk and ball were removed from the Bruker TriboLab, cleaned of residual lubricant, and sealed in airtight packaging for preservation. The raw data from the TriboLab instrumentation and accelerometers was then reviewed and for each second of elapsed time a ground truth class was assigned by subject matter experts as either negative (**n**) or positive (**p**) for active scuffing wear during that elapsed second of data. It is important to emphasize that the ground truth designation is subject to error, a reality that is difficult to avoid for all but the least ambiguous data-sets.

### 3.2.1. Data Balance

Compiling the ground truth vector allowed for the calculation of prevalence using Eq. (1). The resulting prevalence of $5.6\%$ indicates an imbalanced data-set, but not an extreme of imbalances as reported by (Fawcett, 2006). The relatively high prevalence of this data-set is primarily attributed to the 168

Figure 23. SAE 9310 carburized steel disk after test.

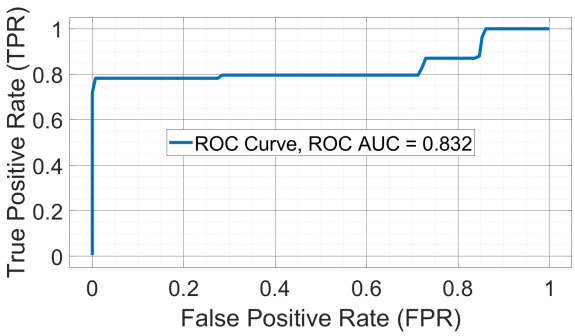


Figure 24. ROC curve for elapsed time classifier.

instances of (**p**) at the test conclusion where the TriboLab was allowed to run to see if a steady state would be re-established. Only including one instance of (**p**) at the test conclusion results in a prevalence of $1.3\%$.

$$prevalence = \frac{P}{P+N} = \frac{216}{216+3609} = 5.6\% \qquad (12)$$

### 3.3. Signal Processing and Classifier Design

Figures 20 through 22 show samples of time domain data for the BoD test. Focusing on the vibration signal from the PCB 352A60 the $153$ records of $25$ seconds were processed into $3825$ one second records and synchronized with the ground truth vector of the same length. To maintain focus on comparing classifier characteristic curves, three simplistic binary classifiers are used.

1. **Elapsed time between test start and end of instance**
2. **Absolute value of first sample for each 1 second instance**
3. **Root mean square (RMS) of full bandwidth power spectral density (PSD)**

The first classifier analyzed was chosen as an example of deceptively good subjective performance and also exemplifies standard interval based maintenance schedules, the second classifier was expected to perform poorly by all objective measures, and the third classifier is expected to perform significantly better as it is informed by the test systems fault physics. The design and optimization of pre-processing algorithms and subsequent scuffing detection algorithms is reserved for future work.

### 3.4. Binary Classifier Characterization Comparison

Using three separate characteristic curves to explore classifier performance is somewhat ironic, when you consider the ROC curve was originally meant to simplify the presentation of classifier performance. By adding two additional characteristic curves, each with their own nuance, returns the user to square one, with too many variables to track simultaneously. With that in mind the following section attempts to add commentary on what each characteristic curve provides an analyst.

The elapsed time classifier, at first inspection, shows respectable ROC curve performance in Figure 24 with a TPR of $0.71$ before FPR becomes non-zero. Observing the broad horizontal shelf of the ROC curve it is clear that the classifier is missing a key component of diagnostic information. The partially informed classifier performs well for a subset of all (**p**) instances but the last $20\%$ of (**p**) instances are seemingly agnostic to the classifier mechanism of elapsed time. With some reflection on ROC curve behavior and understanding the transient scuffing behavior during the test it become obvious why elapsed time does not identify all (**p**) until the FPR has nearly reached a worst case value of $1.0$. Further the elapsed time classifier will be very sensitive to the training data used. A training data-set pulled from a heavily duty-cycled and abused system will produce a very conservative classifier threshold when applied to a system in a milder operating environment and vice versa.

The elapsed time PRC curve of Figure 25 tells a similar story. At first glance, performance seems good, the nosedive in precision as recall approaches a value of $0.8$ indicate there is a non-trivial proportion of (**p**) instances that the classifier is not responsive to. This could be the result of multiple failure modes in the training data with at least one mode veiled to the classifier, or it could simply be a classifier that is predestined to predict (**Y**) at end of a run-to-failure data-set. While the ROC and PRC curves appear drastically different, partially due to Y-axis TPR in the ROC curve and X-axis TPR in the PRC curve, a straightforward interpretation of the visualization provides a similar grasp on classifier interactions. This is not necessarily so for the third visualization, the bookmaker curve.

The Bookmaker curve of Figure 26 is difficult to interpret. The two metrics plotted as classifier threshold increases are both $2^{nd}$ order abstractions of the six variables in the ba-

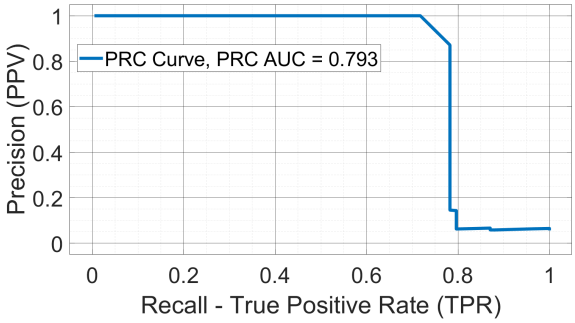


Figure 25. PRC curve for elapsed time classifier.

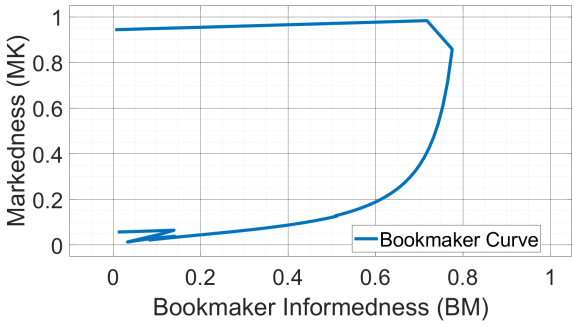


Figure 26. Bookmaker curve for elapsed time classifier.

sic confusion matrix of Figure 1. Knowing that a point at $[1.0, 1.0]$ in Bookmaker curve space represents a perfect classifier certainly helps, but the non-linear changes in MK and BM as threshold changes make AUC calculations meaningless. Likewise the author sees little value in assigning some metric of hysteresis loop area as a quantifying sub-metric of the bookmaker curve. What can be quickly concluded from the bookmaker curve is that there exists a threshold value where the paired metrics of MK and BM have reasonably high amplitude and the smooth curvature of the downward arc suggests relatively stable performance as threshold is varied in this section of Bookmaker curve space. Overall the elapsed time classifier performed subjectively well based on the three visualizations examined, the performance of the classifier was highly dependent on the run to failure nature of the test data-set and is unlikely to transfer well to the same physical system under different duty cycles, the classifier is over-trained to this single test data-set.

Moving to the first value classifier, a clear difference in classifier performance is expected. With a run to failure test data-set there is clear correlation between elapsed time and fault probability, that is not necessarily the case when the absolute value of the first time series acceleration value is used. Observing the time domain acceleration of Figures 20 through 22 the signal amplitude increases significantly during scuffing events - an entire order of magnitude at test ter-

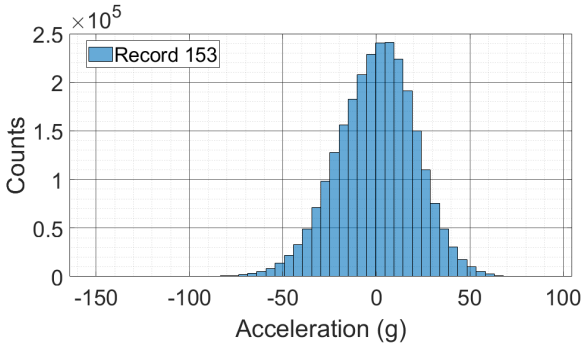


Figure 27. Histogram of record 153 acceleration samples.

mination; however, the signal amplitude is not a DC value. The first sample for a given instance has an amplitude bound by the minimum and maximum acceleration values for that instance but the instantaneous sample could be anywhere within that range. Figure 27 shows the histogram for record 153, the final 25 seconds of data before the test was terminated, the data roughly follows a normal distribution, and statistically any sample is more likely to be near zero than the maximum amplitude. While the probability skews towards lower amplitudes even during high vibration level instances, the distribution of Figure 27 shows that many values will exceed the nominal peak amplitudes for the healthy data ($mean = -0.012$, $STD = 21.05$).

Record 153's data distribution, the most extreme of the faulty data in this BoD test, demonstrates the challenges the first value classifier will experience. The anticipated poor performance of this classifier is confirmed in Figures 28 and 29. The initial sharp rise in the ROC curve's TPR shown in 28 indicates the classifier has some insight into instance class, these are the instances where the first acceleration value are greater than (**n**) instance nominal values and are point in ROC space representing higher threshold values. The low slope of the ROC curve from coordinate $[0.1, 0.6]$ to $[1.0, 1.0]$ is proof that the first value amplitude classifier is a poor performer outside of the most conservative threshold values and at best could be used as an initial screening to identify extreme (**TP**) instances. Frustratingly the ROC AUC for the elapsed time classifier and first value classifier are nearly equivalent despite the classifiers incongruous performance.

Figure 29 shows the PRC curve and PRC AUC for the first value classifier. Unlike the ROC AUC, the PRC AUC is able to differentiate between the elapsed time and first value classifiers. The PRC curve also does a subjectively 'better' job at highlighting the flawed performance of the first value classifier; the classifier can only identify all the (**p**) instances at the expense of a significant number of (**FP**) instances resulting in either high TPR or high precision but never both at the same time.

The first value classifier's Bookmaker curve provides some

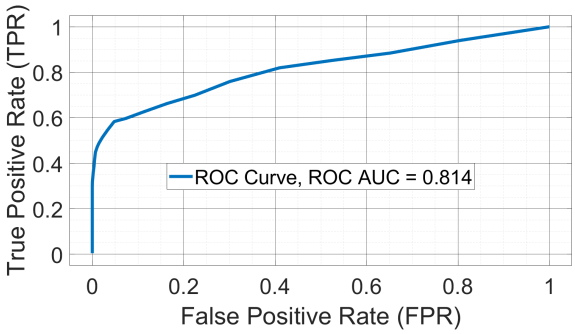


Figure 28. ROC curve for first value classifier.

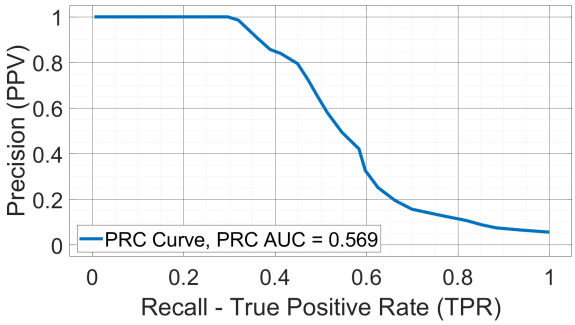


Figure 29. PRC curve for first value classifier.

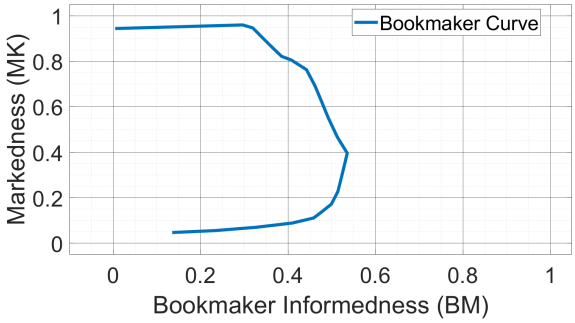


Figure 30. Bookmaker curve for first value classifier.

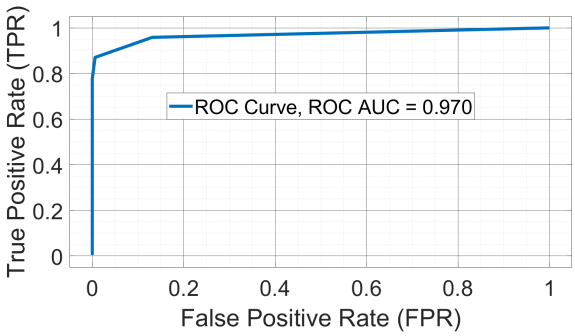


Figure 31. ROC curve full bandwidth RMS classifier.

interesting insight. For some range of threshold values, the values approaching classifier output maximum, the markedness is near its maximum value of one (1). Beyond that region of threshold values the Bookmaker curve generally indicates poor performance. The initial high values of markedness are deceiving and a casualty of data-set class balance. Recall from Eq. (9) that markedness is a higher order abstraction of both precision and NPV. At high threshold values there are few if any (**FP**) instances, maximizing precision. NPV from EQ. (6) is a function of both (**TN**) and (**FN**), while the first value classifier struggles with the normal distribution of acceleration amplitudes the imbalanced data-set minimizes the number of (**FN**) instances and therefore showing a high NPV despite a high FPR. Overall the Bookmaker curve requires the analyst to process a series of scenarios to fully comprehend the visualization and its utility as a quick visual reference is compromised.

The elapsed time classifier showed subjectively good results that must be tempered by the analyst due to the over-trained nature of the classifier and the first value amplitude classifier was plagued by accelerometer output distribution for even the most severe of faulted data. In contrast the curves displayed in Figures 31 through 33 are for a classifier informed by the physical interactions of the test fault mode. In the case of sliding contact scuffing the two hardened steal surfaces are moving against each other in a lubricant bath at a nominal coefficient of friction. The healthy state sliding contact and the TriboLab's drive mechanism generate vibrations detected by the accelerometers. As the test runs, increasing normal force, temperature, and specimen wear lead to conditions where scuffing contact is initiated and vibrations levels drastically increase. Tracking the broadband accelerometer signals should therefore provide an informed classifier to the system class for a given instance, helpfully this classifier is not dependent on a single random sample of time domain making it robust to the identified weakness of the first value classifier. Fine tuned classifiers that focus on specific frequency ranges and frequency domain phenotype are expected to provide improved classifier performance compared to the RMS of vibrations power spectral density but is reserved for future work.

Both the ROC and PRC curve visualizations show better subjective performance than the prior two classifiers with the PRC curve's increased sensitivity to imbalanced data-sets apparent in the AUC values. In many fields the ROC AUC value of $0.970$ would be considered exceptional, this is a trap that PHM analysts must be cognizant of, continuously monitored systems could have thousands, if not millions of (**n**) instances before a single (**p**) instance the visual differences between an ROC curve with $AUC = 0.99$ and $AUC = 0.9999$ will be minuscule yet will drastically effect the real world. For an imbalanced data-set this is where the PRC curve offers ad-

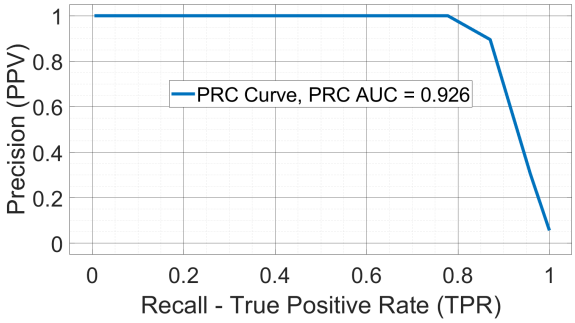


Figure 32. PRC curve full bandwidth RMS classifier.

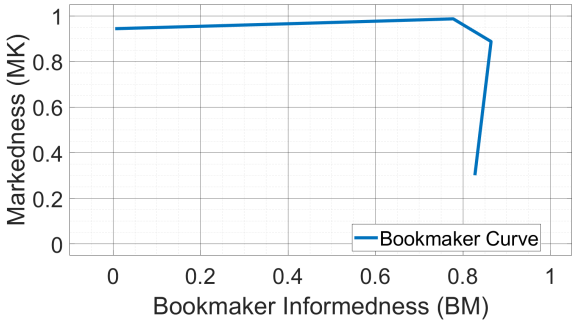


Figure 33. Bookmaker curve full bandwidth RMS classifier.

vantages over the ROC curve, both curves share TPR (a.k.a. recall) as one axis. The secondary axis of each curve provides their differentiation. The second axis of the ROC curve is FPR, defined by Eq. (5) uses (**FP**) and (**TN**), both of which are in the second column of the confusion matrix shown in Figure 1 and are part of the dominate class. As a result the FPR will have a significantly larger denominator than the calculation for precision leading to less sensitivity to false class predictions for the imbalanced data-set. The sensitivity of precision to class balance is beneficial to the analysis. For extreme examples of class imbalance data pre-whitening may also prove helpful to condense duplicated inputs from identical nominal instances, this could be done during the supervised ground truth determination phase in training data-sets.

The Bookmaker curve of Figure 33 shares no resemblance to that of the perfect classifier shown in Figure 15 despite the RMS classifier's good subjective performance. The lack of a common phenotype for well behaved classifiers is a serious impediment to implementing the bookmaker informedness - markedness curve for rapid visual assessment of classifier performance.

## 4. CONCLUSION

In this paper the basics of binary decision classifiers were presented along with key metrics and visualizations to quantify classifier performance. The importance of visualizations to allow analysts to efficiently review classifier performance was discussed and three classifier performance curves were studied in more detail. The ROC curve, PRC curve, and a promising combination of higher order metrics - the Bookmaker curve were detailed by using three synthetic scenarios to build a foundation for readers. Ultimately a PHM run-to-failure experiment was introduced for a real-world comparison of the three binary classifier visualizations.

For poor to moderately performing classifiers the relative differences between ROC and PRC curves were inconsequential to the analysis and either visualization provided a clear depiction of classifier performance. When a vastly superior classifier was introduced the ROC curve is limited because the underlying metrics are not sensitive to data balance and there is little difference between great and truly exceptional classifier performance in the ROC curve and the ROC AUC metric. For the imbalanced data-set studied the relative lack of (**FP**) and (**TP**) instances makes the PRC curve much more sensitive to small changes in classifier performance for high performing classifiers.

For all classifiers studied the Bookmaker curve proved unwieldy and difficult to rapidly interpret because the X- and Y-axis metrics are both 2nd order abstractions from the fundamental variables used in the confusion matrix shown in Figure 1. Perhaps with more experience and examples using the dual metrics of bookmaker informedness and markedness this shortcoming could be overcome. At this time the Bookmaker curve, while a promising concept, provides minimal added value compared to the ROC curve.

For imbalanced data-sets encountered in PHM the PRC curve appears to be the superior visualization if the target audience has similar familiarity with both ROC and PRC curves. The shortcomings of the ROC curve can be worked around and with the proper analytical perspective ignored; however, they may be optimistically misconstrued to a casual audience.

## ACKNOWLEDGMENT

A special thanks to the Penn State University Tribology/Materials Processing Lab for the use of their facilities and assistance for the block-on-ring testing.

## REFERENCES

Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, *30*(7), 1145-1159. doi: https://doi.org/10.1016/S0031-3203(96)00142-2

Chicco, D., Tötsch, N., & Jurman, G. (2021). The matthews correlation coefficient (mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData Mining*, *14*, 1-22. doi: 10.1186/s13040-021-

00244-z

Eklund, N. (2022). Phm 2022 analytics short course..

Fawcett, T. (2006). An introduction to roc analysis. *Pattern Recognition Letters*, *27*, 861-874. (ROC Analysis in Pattern Recognition) doi: https://doi.org/10.1016/j.patrec.2005.10.010

Hand, D. J. (2009, 10). Measuring classifier performance: A coherent alternative to the area under the roc curve. *Machine Learning*, *77*, 103-123. doi: 10.1007/s10994-009-5119-5

Ludema, K. C. (1984). *A review of scuffing and running-in of lubricated surfaces, with asperities and oxides in perspective* (Vol. 100).

Powers, D. M. W. (2003). Recall and precision versus the bookmaker. , 529. doi: 10.13140/RG.2.1.3754.1926

Powers, D. M. W. (2008, 6). Evaluation: From precision, recall and f-factor to roc, informedness, markedness & correlation. *Mach. Learn. Technol.*, *2*.

Takaya, M. S., & Rehmsmeier. (2015, 6). The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, *10*, 1-21. doi: 10.1371/journal.pone.0118432

## BIOGRAPHIES

**Dan Watson** is a researcher with the Applied Research Laboratory (ARL) at Pennsylvania State University and a Ph.D. Candidate in the Graduate Program in Acoustics. His background includes a Master of Science in Engineering in Mechanical Engineering from the University of Michigan-Dearborn (2016) and a Bachelor of Science in Mechanical Engineering from Rose-Hulman Institute of Technology (2012). He spent six years as a systems engineer for U.S. Army Tank Automotive Research, Development and Engineering Center (TARDEC) prior to joining Penn State as a graduate research assistant. Current research includes airborne acoustic and structural vibration analysis within the realm of PHM.

**Dr. Karl M. Reichard** Dr. Karl M. Reichard is an Associate Research Professor with the ARL and the Graduate Program in Acoustics at Penn State University. Dr. Reichard has over 30 years of experience in the design and development of advanced systems for sensing and controls. Dr. Reichard is involved in the development and deployment of systems for prognostic health management in mechanical and electrical systems, acoustic surveillance and detection, active noise and vibration control, and robotics.

Dr. Reichard currently serves as the chief scientist for the ARL's Systems Operations and Automation (SOA) Division. He previously served as the head of the SOA Division's Embedded Monitoring and Control Systems Department. The department conducts research and develops systems for the monitoring, diagnosis and prediction of health and status in mechanical and electrical systems. Dr. Reichard is a Fellow and Member of the Board of Directors of the Prognostics and Health Management Society, and a member of the IEEE and the Acoustical Society of America. He is the author of over 50 papers and articles published in journals and conference proceedings. He teaches courses in digital signal processing, active sound and vibration control, signal analysis, and prognostic health management.

Dr. Reichard received the Ph.D., M.S. and B.S. degrees in Electrical Engineering from the Virginia Polytechnic Institute and State University (Virginia Tech).

**Aaron Isaacson** is managing director of the Gear Research Institute and Head of the Penn State University, Drivetrain Technology Center. Aaron received B.S. (1998) and M.S. (2009) degrees in mechanical engineering from Penn State and is pursuing his Ph.D. in Materials Science and Engineering at Penn State.