Australian Government
Department of Defence
Science and Technology

TECHNICAL REPORT

Uniform Calibration of Anomaly Detectors with Multiple Sub-Classes for Robust Performance

T. L. Keevers

Joint and Operations Analysis Division
Defence Science and Technology Group

DST-Group-TR-3765

Produced by

Joint and Operations Analysis Division
Defence Science and Technology Group

DST Headquarters
Department of Defence F2-2-03
PO Box 7931
Canberra BC ACT 2610

www.dst.defence.gov.au
Telephone: 1300 333 362

EXECUTIVE SUMMARY

Anomaly detection is the task of sorting data points into normal and anomalous classes. In a semi-supervised setting, there is only access to normal class samples during the training phase. Through generating and ranking an appropriate scalar quantity, these training samples can be used to calibrate the anomaly detector so that it produces a specified false alarm rate. However, this procedure only controls the global false alarm rate, which can lead to excess misclassification of certain (normal) sub-classes, or the detector may become miscalibrated if the proportion of sub-classes drift.

In this technical report, we propose a method for constructing an anomaly detector with a uniform false alarm rate for each sub-class: an anomaly detector is trained for each sub-class independently and tuned for a specified false alarm rate. These can then be combined into a single anomaly detector (that flags an anomaly if and only if each sub-detector identifies an anomaly) that has a maximum specified false alarm rate, regardless of any drift in the sub-class distribution.

This approach brings a number of benefits:

- It protects against imbalance in the sub-classes. The traditional practice of using a single threshold can cause sub-classes with few samples to become consistently misclassified as anomalies. This will not happen for separate thresholds, although there can be degraded performance because of small sample sizes.

- It can produce more robust results when there are concerns about algorithmic bias or discrimination for particular sub-classes. For example, when sub-classes represent populations with socially-protected attributes.

- It protects against drift in the sampling distribution. The false alarm rate will remain nearly constant, even if the proportion of each sub-class changes dramatically.

- It allows different algorithms or metrics to be used for each sub-class. For instance, one sub-class could use a Gaussian Naïve Bayes algorithm to classify anomalies with the probability density used as the classification metric, while another sub-class may use a encoder-decoder neural network with the reconstruction error instead.

- The ensemble anomaly detector can be easily customized. Sub-class anomaly detectors can be added or removed without having to retrain any other components. This is not true when a single global threshold is used.

The false positive and negative rates can be estimated by validating each sub-class detector

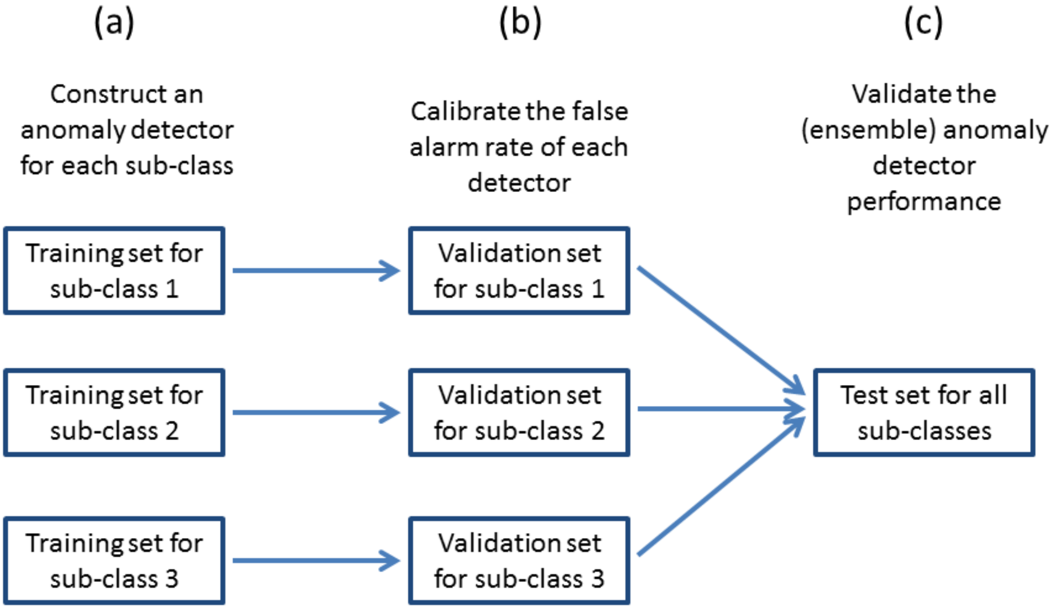
(a)
Construct an anomaly detector for each sub-class
(b)
Calibrate the false alarm rate of each detector
(c)
Validate the (ensemble) anomaly detector performance
Training set for sub-class 1
Training set for sub-class 2
Training set for sub-class 3
Validation set for sub-class 1
Validation set for sub-class 2
Validation set for sub-class 3
Test set for all sub-classes

Figure 1 Cross-validation can be modified for semi-supervised anomaly detection. The data is separated into sub-classes, and training, validation, and test sets are constructed. Sub-class anomaly detectors are constructed (a) and calibrated (b) separately during the training phases. Anomaly detector false alarm rates and sensitivity to outliers are estimated using data from all of the sub-classes (c).

against all of the data points, as shown in Figure 1. This allows cross-validation to be applied to a semi-supervised learning technique (where usually supervised learning is required). While the actual false negative rate will depend on the specific anomalies encountered, this provides a general method for assessing the properties of an anomaly detector.

Cross-validation can also provide a level of granular insight not normally available. This is demonstrated by training a convolutional autoencoder on the MNIST data set: we find the sub-detector for ‘1’s can effectively reject other digits, while the ‘7’ and ‘9’ detectors are unable to recognize ‘1’s as anomalies. These kinds of insights can be used to guide further improvement of the anomaly detector. While the sub-class detectors we constructed seem to somewhat reduce the ability to detect anomalies (Type-II errors), this nevertheless provides a promising direction for building more robust and flexible anomaly detectors.

CONTENTS

FIGURES

TABLES

NOTATION

c	Normal sub-class
n	Number of normal sub-classes
p_{FA}	Probability of false alarm
p_{FP}	Probability of false positive
x	Set of features
y	Classes
δ	Anomaly detector
δ_i	Anomaly detector for i-th class

1. INTRODUCTION

In many situations data is generated from a combination of normal and anomalous processes [1]. Discriminating between these classes is called anomaly detection and is pertinent to radar detection [2, 3], cyber-intruder detection [4], image processing [5], fraud detection [6], and other domains [7]. The properties that make a process 'normal' or 'anomalous' typically follow from the context of the problem or user-defined requirements. For example, sensors readings may be 'anomalous' when the sensor has been contaminated with an undesired chemical.

In semi-supervised learning there is access to training data generated from the normal process, but no data for the anomalous process. It is common to create anomaly detectors that have a constant false alarm rate [8–10]. However, this false alarm rate is a global property and in some situations this may have unintended side effects. Normal processes may be composed of several subpopulations, or sub-classes. Instances arising from one or more sub-classes may be consistently misidentified as anomalies if they are rarely observed [11]. This could be a problem if the relative proportion of the two or more normal sub-classes drifts over time, leading to a higher than expected false alarm rate, or the two components could represent different social groups and disparate false alarm rates could contribute to algorithmic bias or discrimination.

The concept of sub-classes and their detection is shown in Figure 1. The normal process is generated from two normal distributions with standard deviations of 1, centred at 0 and 8, with relative weightings of 0.97 and 0.03, respectively. An anomaly detector with a 5% false alarm rate may flag nearly every sample arising from the smaller normal component because of its relatively low density (the uncoloured region in Figure 1a). An alternative approach is to train anomaly detectors on each sub-class separately and then combine them into an aggregated anomaly detector for all the sub-classes. An example of this is shown in Figure 1b with normal regions defined for each of the Gaussian components.

To the best of our knowledge the approach of aggregating anomaly detectors in this fashion has not been previously explored in the literature. It is an approach that has a number of intriguing properties, some of these are beneficial, and some are detrimental. In this technical report, we evaluate the use of anomaly detectors with uniform false alarm rates across sub-classes. A brief overview of Bayesian anomaly detection in the supervised anomaly detection case is given in Section 2. It provides a description of the process we want to emulate for the semi-supervised case. In Section 3, we describe our uniform threshold approach in more detail and examine its potential benefits and limitations. In Section 4, we demonstrate the benefits of this approach for training data with unbalanced sub-classes using a synthetic data set. In Section 5, a simple anomaly detector for written digits is constructed by applying the Gaussian Naïve Bayes algorithm to the MNIST data set [12] which highlights some possible limitations of the approach. An improved an-

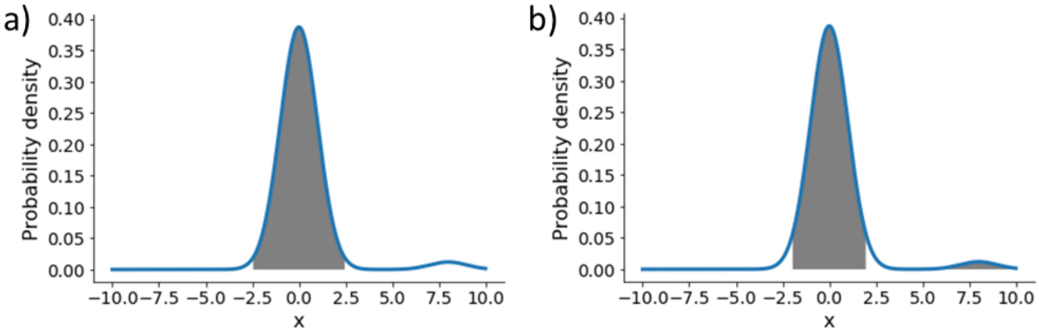

Figure 1 Two potential anomaly detector schemes for a normal class with a bi-Gaussian distribution. Both have a 5% false alarm rate. The dark regions indicate points that are identified as coming from the normal class and the blue line shows the probability density function. a) This detector uses the regions of high probability density to classify anomalies. b) This anomaly detector is the combination of high probability density from each of the Gaussians. Note that the central region in (a) is slightly wider than that found in (b).

omaly detector constructed from autoencoders is presented in Section 6, which demonstrates the ability to iteratively develop an effective anomaly detector using our approach. We discuss other approaches to calibration and model scoring in Section 7. Finally, concluding remarks are made in Section 8.

2. BAYESIAN SOLUTION TO ANOMALY DETECTION

Before considering semi-supervised learning, it is useful to revisit the Bayesian solution to supervised anomaly detection, where we assume we have access to the distributions for the normal and anomalous classes. The odds of a data point x belonging to the normal class are

$$\frac{p(y=0|x)}{p(y=1|x)} = \frac{p(x|y=0)p(y=0)}{p(x|y=1)p(y=1)}, \tag{1}$$

where $y = 0$ is the normal class, and $y = 1$ is the anomalous class. We can control the false positive/negative rates by changing the threshold at which we flag $\frac{p(y=0|x)}{p(y=1|x)}$ as an anomaly. For example, we could choose to control the overall false positive rate, the accuracy, or some other quantity.

In semi-supervised learning, we no longer have access to $p(x|y = 1)$ or $p(y = 1)$. Access to $p(y = 1)$ is arguably not too important - we can still control the false discovery rate using the Bayes factor, $\frac{p(y=0|x)}{p(y=1|x)}$. However, without $p(y = 1|x)$, we are likely to have either too many false positives or false negatives.

A potential solution is to replace the real anomalous distribution with an uninformative prior distribution, which represents anomalies being drawn from a high-entropy distribution. While this allows Equation 1 to be used, other issues remain. First, prior distributions are usually defined for the parameters of a model, and not the observations x directly, and for semi-supervised learning, we do not have a model for the anomaly generating process; second, uninformative prior distributions can be problematic in high dimensions. The uniform prior distribution is not invariant under coordinate transformations. Invariance can be enforced by using Jeffrey's prior, but this prior can have other undesirable properties [13, 14]; finally, uninformative distributions are too easy to discriminate from the normal class, which inflates the apparent utility of the anomaly detector. For the MNIST dataset, we applied the anomaly detectors trained in Sections 5 and 6 to images drawn from a uniform distribution and found nearly perfect discrimination, despite the anomaly detectors having limited capability for identifying realistic anomalies.

In spite of these problems with using uninformative prior distributions, we often still employ them implicitly for semi-supervised anomaly detection. A common approach is to flag anomalies whenever the likelihood of the normal class, $p(y = 0|x)$, drops below a threshold. This is equivalent to using the Bayes factor with a uniform probability for the anomalous class ($p(y = 1|x) \propto 1$).

Apart from access to the anomalous class, we are also interested in being robust against changes in the sub-class distribution. When the probability of each sub-class is known, Equation 1 can be expanded to

$$\frac{p(y=0|x)}{p(y=1|x)} = \frac{\sum_{i=1}^{n} p(x|y=0, c=i)p(c=i|y=0)p(y=0)}{p(x|y=1)p(y=1)}, \tag{2}$$

where c indicates the sub-class and there are n sub-classes.

When the probability of each sub-class is unknown, we can still bound the false discovery rate,

$$\operatorname*{argmin}_i \frac{p(x|c=i, y=0)p(y=0)}{p(x|y=1)p(y=1)} \leq \frac{p(y=0|x)}{p(y=1|x)} \leq \operatorname*{argmax}_i \frac{p(x|c=i, y=0)p(y=0)}{p(x|y=1)p(y=1)}. \tag{3}$$

If we again treat $p(x|y = 1)$ as constant, this is equivalent to using a threshold for the maximum likelihood of an anomaly that is independent of the sub-class. When treating semi-supervised learning, we bound the false detection rate, not the false discovery rate, but it is easy to change between these two approaches, as shown above.

3. UNIFORM CALIBRATION

As mentioned, we do not have access to the anomalous probability distribution (and usually have limited knowledge about the normal probability distribution) in semi-supervised learning, so we cannot apply Bayes' theorem without further information or by making assumptions about the anomalous process. Many of the common approaches to semi-supervised learning develop implicit or explicit representations of how much a data point deviates from normal behaviour. These include measuring the distance from nearby data points [15], one-class support vector machines [16], measuring reconstruction error after passing the data through a neural network autoencoder-autodecoder pair [17], or by approximating the probability density function using parametric [18] or non-parametric models [19].

These approaches transform a multi-dimensional data point x into a scalar quantity [9]. Points are classified as anomalies depending whether this scalar quantity is greater or less than the threshold. Rather than choosing a single threshold, we can create an anomaly detector for each sub-class and calibrate their false alarm rates separately. The threshold values can be estimated non-parametrically [20], and can be done reliably for even modest sample sizes.

Cross-validation is usually used to estimate the performance of a supervised learning algorithm. We adapt this procedure to assess the performance of an ensemble anomaly detector. An outline of the procedure is shown in Figure 2. The data set is partitioned by sub-class, and then further split into training, validation, and test sets. Depending on the type of anomaly detector being used, not all of these sets may be required. The training set is used to construct a representation of each normal sub-class. This could, for instance, involve representing each sub-class as a separate Gaussian distribution and tuning their means and standard deviations. The validation set is used to tune the thresholds to achieve a desired false positive rate. Finally, the test set can be used to confirm the false positive rate and to estimate the false negative rate by validating the classification against other sub-classes. The false positive rate for the validation and test set are usually similar. This is because the thresholds are one-dimensional scalar quantities, which restricts our ability to unintentionally overfit their values, and the sub-classes are partitioned during the training phase, so data leakage between sub-class detectors is minimized. There may be scope to tune the anomaly detectors to improve their power for a given false alarm rate. However, the ultimate aim is to create an anomaly detector that discriminates data points that are generated from the anomalous class, which are not included in the test set. Nevertheless, the cross-validation false negative rates can serve as a useful guide.

Once each sub-class detector has been calibrated and assessed, they can be combined into an *ensemble anomaly detector*. This ensemble anomaly detector will only identify a data point as an anomaly if and only if each sub-detector identifies it as an anomaly,

$$\delta(x) = \prod_{i=1}^{n} \delta_i(x), \tag{4}$$

where δ is the aggregate anomaly detector and outputs 1 to indicate an anomaly, and δ_i is the anomaly detector for the i-th normal sub-class.

By design the ensemble anomaly detector will never produce a false positive rate (P_{FP}) greater than that specified for each of the sub-classes, irrespective of changes in the mixture,

$$P_{FP} = \int \delta(x)p(x|y = 0, c = i)dx = \int \prod_{j=1}^{n} \delta_j(x)p(x|y = 0, c = i)dx \tag{5}$$

$$\leq \int \delta_i(x)p(x|y = 0, c = i)dx = \alpha, \tag{6}$$

where the c is used to identify the sub-class for normal samples, α is the maximum false alarm rate for each sub-class, and the inequality applies for samples from any sub-class ($i \in 1, \cdots, n$). When detectors for different sub-classes disagree about the status of a data point, we call this interference. Interference leads to a reduction in the false positive rate, which in isolation is beneficial. However, this is counter-balanced by a higher false negative rate (P_{FN}),

$$P_{FN} = \int (1 - \delta(x))p(x|y = 0, c = i)\ dx = \int \left(1 - \prod_{j=1}^{n} \delta_j(x)\right)p(x|y = 0, c = i)\ dx \tag{7}$$

$$\geq \int (1 - \delta_i(x))p(x|y = 0, c = i)\ dx. \tag{8}$$

Again, this holds true for any sub-class ($i \in 1, \cdots, n$). While we can set uniform false positive rates for each of the sub-class detectors, there is likely to be some variation in the sub-class false positives rates during test time, and achieving true uniform calibration across all sub-classes may be difficult or impossible. As a counter-example, consider two one-dimensional Gaussian distributions with identical means, but different standard deviations. If we use the high probability regions to perform anomaly detection, then the false alarm rates can never be the same, except in the trivial cases of complete or vanishing coverage.

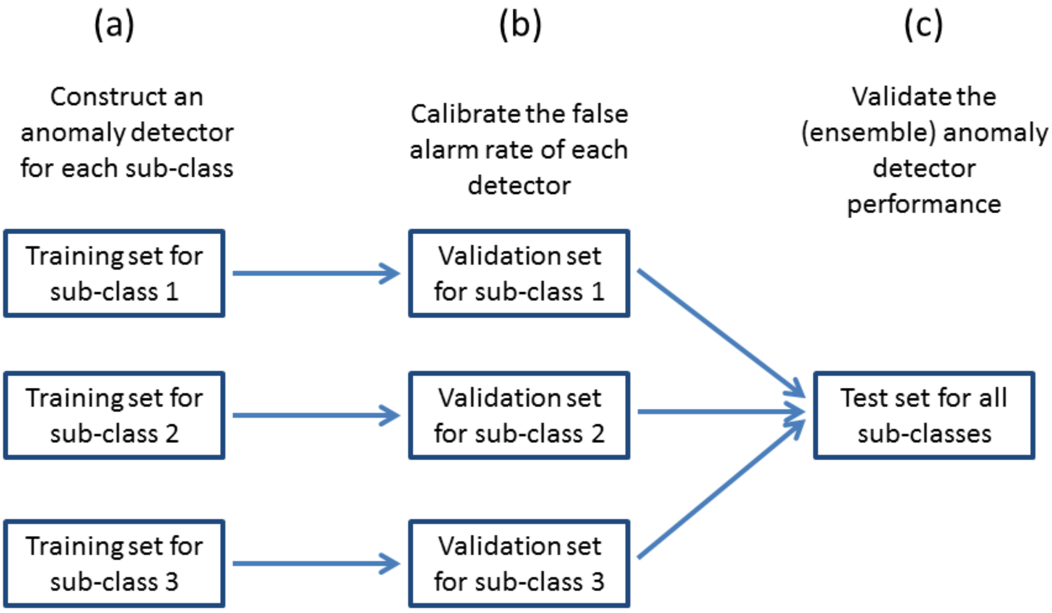
(a)
Construct an anomaly detector for each sub-class
(b)
Calibrate the false alarm rate of each detector
(c)
Validate the (ensemble) anomaly detector performance
Training set for sub-class 1
Training set for sub-class 2
Training set for sub-class 3
Validation set for sub-class 1
Validation set for sub-class 2
Validation set for sub-class 3
Test set for all sub-classes

Figure 2 Cross-validation can be modified for semi-supervised anomaly detection. The data is separated into sub-classes, and training, validation, and test sets are constructed. Sub-class anomaly detectors are constructed (a) and calibrated (b) separately during the training phases. Anomaly detector false alarm rates and sensitivity to outliers are estimated using data from all of the sub-classes (c).

4. EFFECT OF COMPOSITION ON GAUSSIAN MIXTURES

To demonstrate the benefits of a uniform false detection rate when there is sub-class imbalance, we consider a normal class consisting of two Gaussians centred at the points $(-1, 0)$ and $(1, 0)$, and the anomalous class is a single Gaussian centred at the point (0,2). Each Gaussian has a standard deviation of 1 for each component with no covariance. The robustness of the global and uniform false alarm rates are verified by varying the mixture of the two normal sub-classes at either test or training time. The training and test sets are generated using random sampling, and consist of the normal, and the normal and anomalous classes, respectively. Owing to the large sample sizes, the Monte Carlo error is negligible.

The first row of Figure 3 shows the change in the false positive and negative rates of an anomaly detector constructed using a single threshold (blue) and one generated from two sub-class detectors with uniform thresholds (red). The detection rates are shown as the mixture varies between 5% and 95% of the first sub-class. The uniform threshold detector has consistent performance for different training sets, showing the expected increase in robustness. The false alarm rate of the uniform threshold detector is lower than its nominal rate (shown by the dashed black line in Figure 3c) because of the interference between the sub-class detectors. While the reduced false alarm rate is beneficial, it has the undesired side effect of reduced statistical power, too (the difference between the red and blue lines in Figure 3d). The global anomaly detector (blue) attains its nominal false alarm rate when the training and test sets have the same sampling distributions, which can be seen by the minimum at 0.6 in Figure 3a. The false alarm rate increases for highly polarized mixtures ($<$0.2 and $>$0.8), with some improvement in the statistical power of the detector. The inflated false alarm rate is consistent with our expectations, and demonstrates the limitations of a global threshold when there is drift in the data.

Plots (c) and (d) show the changes in detector effectiveness when a fixed training set is used with a variable test set. Again, the uniform detector has a lower false alarm rate and statistical power than the global detector and stable performance. Since the training set is constant the statistical power of the global detector does not vary either (Figure 3d). The false alarm rate is over twice its nominal rate for very polarized mixtures (less than 10% of the first sub-class), but actually has a better-than-expected false alarm rate when the test mixture is strongly biased in the other direction. This is because samples are predominately generated in the high probability region of the Gaussian mixture.

The trade-off between the false alarm rate and statistical power is visualized through the receiver operating characteristic curve. In (e) the training and test sets come from the same distribution. The performance of detector trained separately on each sub-class is virtually the same. When there is some discrepancy between the training and test data (f), the local threshold detector maintains its original performance, while the global threshold detector is degraded.

As a final comparison, we repeat the analysis in which one of the Gaussian components in the normal mixture is replaced by a bi-Cauchy distribution with no correlation between the features and scale parameters of 0.3. The introduction of a heavy-tailed distribution is used to simulate cases in which the normal component has significant variation, so the notion of anomalous behaviour is harder to define. The results are shown in Figure 4, with the same patterns of behaviour as in the Gaussian case: the uniform threshold detector has both a lower false alarm rate and lower statistical power. The main benefit is that the performance is more consistent when there are changes in the relative sub-class proportions. The inconsistency of the global threshold model is, again, most acute for highly polarized mixtures. The receiver operator characteristic curves shown in Figure 4a and b show the uniform and global threshold detectors have similar performance. The global detector outperforms the uniform detector somewhat when the training and test sets are drawn from the same distribution (e), but the trend is reversed for small false alarm rates when this is not the case (f). While these two datasets lack realistic complexity, they provide a demonstration that anomaly detectors can, under some circumstances, be made more robust without a loss in performance.

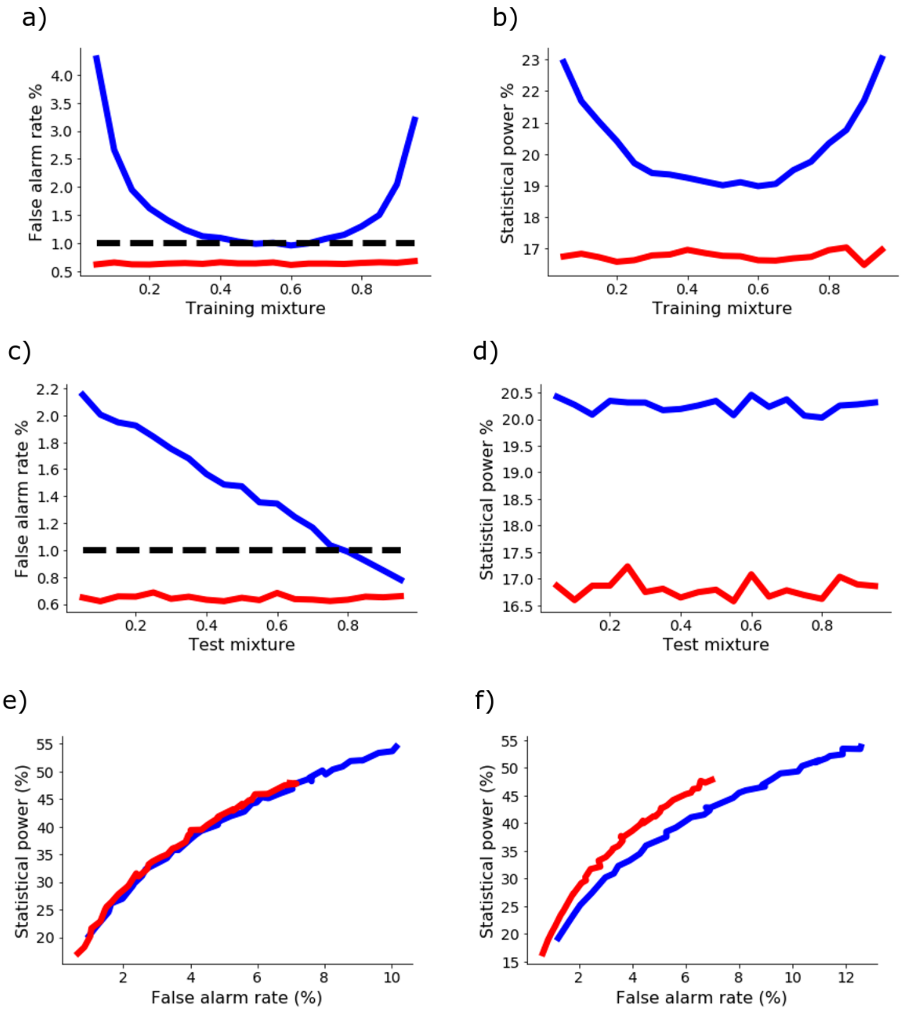

Figure 3 Semi-supervised anomaly detector trained on a bi-Gaussian distribution. The false alarm rate and statistical power of an anomaly detector generated from the high probability regions (blue) or by composing anomaly detectors from each sub-class (red). a) The change in the true false alarm rate as the training set is varied for a fixed nominal false alarm rate (indicated by the black dashed line). b) The change in the statistical power as the training set is varied. c and d are similar to a and b, but with the training set held fixed and the test set varied. Receiver operator characteristic curves are shown for the cases of the training and test set being drawn from the same (e) and different (f) distributions.

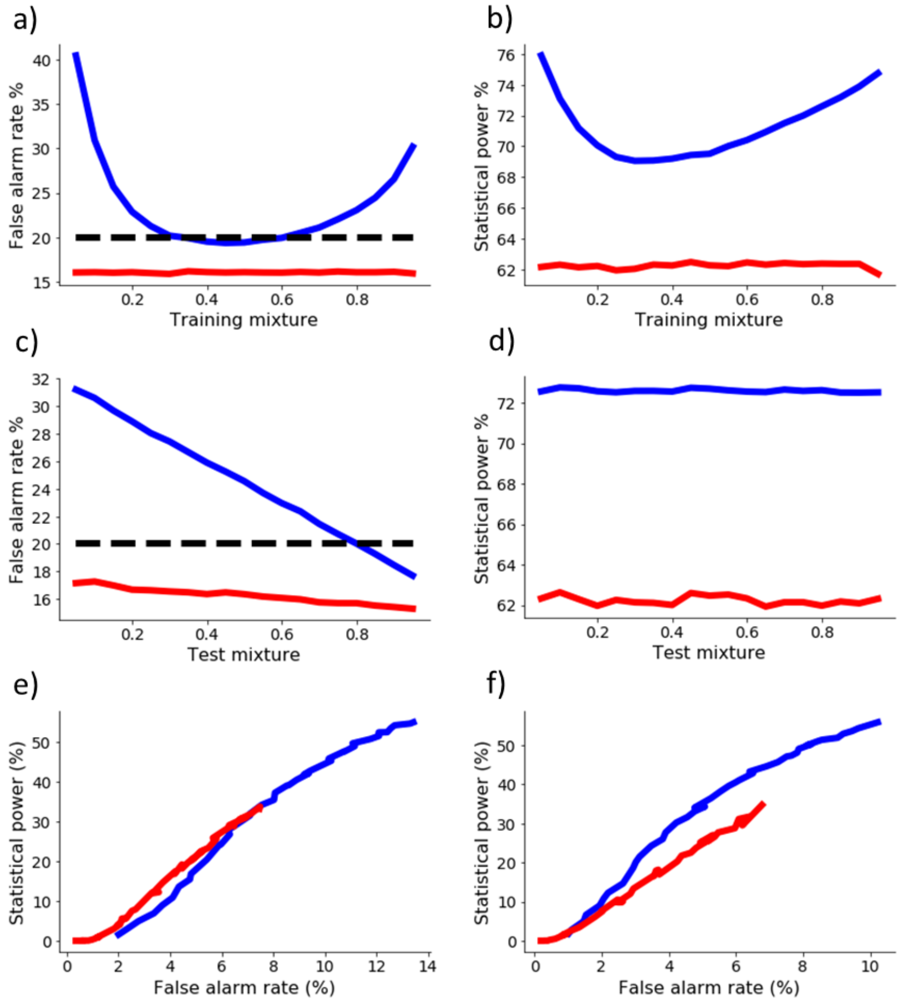

Figure 4 Semi-supervised anomaly detector trained on a Gaussian and Cauchy mixture distribution.The false alarm rate and statistical power of an anomaly detector generated from the high probability regions (blue) or by composing anomaly detectors from each sub-class (red). a) The change in the true false alarm rate as the training set is varied for a fixed nominal false alarm rate (indicated by the black dashed line). b) The change in the statistical power as the training set is varied. c and d are similar to a and b, but with the training set held fixed and the test set varied. Receiver operator characteristic curves are shown for the cases of the training and test set being drawn from the same (e) and different (f) distributions.

5. COMBINING GAUSSIAN MIXTURES TO DETECT MNIST ANOMALIES

An anomaly detector was developed for handwritten digits by fitting a Gaussian Naïve Bayes model for each digit of the MNIST data set using Scikit-learn [21]. The pixels were scaled to have a maximum intensity of 255. Thresholds were chosen to produce a false alarm rate of 20% using the χ^2 statistic, which is equivalent to thresholding the marginal likelihood of each digit, with any pixel with an average intensity of less than 1 ignored as they are likely to be part of the background. The MNIST pixel intensities display strong spatial correlations and the variation is clearly not Gaussian, so we expect the Gaussian Naïve Bayes detectors to perform poorly, and create a loss in statistical power when the sub-class detectors are combined.

The statistical power and false alarm rates for each sub-class detector is given in Table 1, along with the effectiveness of the ensemble detector, where we define statistical power as the percentage of data points that are flagged as anomalies when the sub-class detector for the true class is removed. The individual detector performance is separated into the performance for each digit to provide a finer-grained view of their performance. The empirical false alarm rate for each detector channel is given by the diagonal entries and also along one of the bottom rows. The values all hover around 20%, as expected from the calibration step. The minor discrepancy possibly comes from different people contributing to the digits in the training and test sets (a deliberate feature of the MNIST data set). The detector channels for digits '0' and '1' seem particularly effective at discriminating other digits, with the ability to correctly identify anomalies more than 90% of the time. Surprisingly, the other detectors seem to have difficultly correctly rejecting '1's as false alarms. The '2' and '5' digit anomaly detectors only reject 3% of the '1's as anomalies, and the '8' channel only identifies it as an anomaly 2% of the time. The low anomaly detection rate violates the common assumption that the false negative rate will exceed the false positive rate, and exemplifies the need to check anomaly detection performance. The high false negative rate could arise from filtering out the pixels with low activation for each anomaly detector, and suggests this information should be retained.

The overall performance of the ensemble anomaly detector is along the bottom of Table 1 with the false alarm rate and power against each class. The false alarm rate is how many times the true class was flagged as an anomaly by the corresponding detector channel, and is extracted from the diagonal entries. The 'false alarm rate (all)' is the probability that all the detector channels identify the digit as an anomaly and therefore the ensemble anomaly detector would flag it as an anomaly. Since all of these data points are from the normal sub-classes, any anomalies are false alarms. When the sub-class detectors are combined the false alarm rate is reduced from the 20% for each individual channel because of interference between channels. The false alarm rate of the '4' and

Table 1 Information about the anomaly detector constructed from a series of Gaussian mixtures. The top section shows the anomaly detection rate for different classes and detector channels. The bottom rows provide a summary of the overall performance. The first of these rows provides the false alarm rate of each individual detector (which are identical to the diagonal entries in the top section of the table), the second provides the false alarm rate when all of the detectors are used, and the final row provides the power of the anomaly detector when the corresponding class detector is removed.

		True class									
		0	1	2	3	4	5	6	7	8	9
Detector channel	0	0.16	1.00	0.98	0.96	1.00	0.94	0.98	0.91	0.99	0.99
	1	1.00	0.19	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	2	0.83	0.03	0.19	0.53	0.67	0.75	0.57	0.63	0.65	0.76
	3	0.81	0.05	0.71	0.16	0.61	0.59	0.79	0.50	0.41	0.49
	4	0.98	0.44	0.98	0.93	0.19	0.88	0.86	0.45	0.76	0.19
	5	0.80	0.03	0.85	0.43	0.40	0.19	0.83	0.51	0.31	0.34
	6	0.96	0.42	0.93	0.95	0.84	0.95	0.22	0.90	0.96	0.77
	7	1.00	0.75	0.99	0.97	0.77	0.97	0.99	0.20	0.96	0.43
	8	0.82	0.02	0.75	0.52	0.49	0.45	0.55	0.51	0.18	0.34
	9	0.99	0.41	0.99	0.94	0.55	0.95	0.90	0.50	0.89	0.19
	False alarm	0.16	0.19	0.19	0.16	0.19	0.19	0.22	0.20	0.18	0.19
	False alarm (all)	0.14	0.01	0.16	0.12	0.17	0.14	0.17	0.13	0.14	0.12
	Power	0.73	0.01	0.61	0.27	0.35	0.32	0.47	0.29	0.24	0.14

‘6’ digits remain high at around 17%, while the false alarm rate for the ‘1’ digits is only 1%. Interference between channels also reduces the power of the ensemble, which is the proportion of data points identified as an anomaly by every detector channel, excluding the detector channel of the true class. This ranges from just 1% for the ‘1’ digits, up to 73% for the ‘0’ digits. The strong interference between the channels probably arises from the simplistic nature of the detectors we used.

Table 1 suggests our ability to identify new types of anomalies will strongly depend on their structure. Anomalies that have a shape similar to that of ‘1’s will usually be identified as normal, while anomalies that have loops in them, like ‘0’s, ‘6’s and some of the ‘2’s, are much more likely to be flagged as anomalies.

6. COMBINING CONVOLUTIONAL AUTOENCODERS TO DETECT MNIST ANOMALIES

Handwritten digits tend to be quite distinctive to humans and can be classified with more than 99% accuracy using state-of-the-art classifiers [22]. This suggests that more sophisticated detectors should allow additional detector channels to be added without such a large drop in statistical power. To test this hypothesis, we trained a series of convolutional autoencoders [23]. The encoder had six layers, alternating between convolutional layers and max-pooling layers. The convolutional layers each had a three by three kernel, and six, four and two filters, respectively. The max-pooling used a two-by-two window. The decoder layer was like a mirror image, with six layers alternating between convolutional layers and upsampling layers. The convolutional layers had two, four, and six filters, respectively, and a three by three kernel. The upsampling layers were two-by-two. We also used the mean-square reconstruction error to detect anomalies. The mean-square error was used because it is commonly employed in the literature (for example [17]) and conceptually simple. Other metrics like absolute-mean reconstruction error can also be implemented. Convolutional neural networks are effective for classifying MNIST digits and convolutional autoencoders have been found to provide excellent anomaly detection for perturbed MNIST digits and other data sets [24, 25], so we expected convolutional-based methods to work well when the anomaly detector for each digit is trained separately.

Overall, the results were encouraging (Table 2). The power for most digits was over 70%. However, there was substantial variation at the more granular level. Like the Gaussian mixtures, the anomaly detectors tended to identify ‘1’s as belonging to their own class. They only recognized ‘1’ as an anomaly 2-20% of the time, in many cases less than the false alarm rate of the ‘1’ subdetector. More encouragingly, the ‘1’ detector channel was able to effectively reject the other digits. We are unsure why this asymmetry with ‘1’ digits occurs. It could be due to the relatively low spatial complexity of ‘1’s as compared with the other digits. The digits ‘2’, ‘4’, ‘5’, ‘7’, and ‘9’ sometimes have lines that look similar to ‘1’s in their structure. This may mean their autoencoders automatically learnt to reproduce line-like structures, allowing the ‘1’s to be effectively reproduced. Conversely, the autoencoder for the ‘1’ digit probably has little incentive to learn loop and curve structures that occur in the other digits, causing it to have a large reconstruction error for other shapes.

It is likely that the results could be improved using more sophisticated autoencoder architectures. Nevertheless, the low false alarm rate and high power for most of the digits (the exceptions being 1, 7 and 9) show combining anomaly detectors can be an effective strategy.

Table 2 Information about the anomaly detector constructed from a series of auto-encoders. The top section shows the anomaly detection rate for different classes and detector channels. The bottom rows provide a summary of the overall performance. The first of these rows provides the false alarm rate of each individual detector (which are identical to the diagonal entries in the top section of the table), the second provides the false alarm rate when all of the detectors are used, and the final row provides the power of the anomaly detector when the corresponding class detector is removed.

		True class									
		0	1	2	3	4	5	6	7	8	9
Detector channel	0	0.20	0.20	1.00	1.00	1.00	1.00	0.99	0.98	1.00	1.00
	1	1.00	0.19	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	2	0.97	0.06	0.18	0.97	0.95	0.99	0.98	0.28	0.95	0.69
	3	1.00	0.12	0.94	0.16	0.99	0.92	0.99	0.61	0.97	0.84
	4	1.00	0.10	1.00	1.00	0.17	1.00	0.97	0.76	1.00	0.53
	5	0.89	0.06	0.99	0.86	0.98	0.19	0.88	0.85	0.99	0.86
	6	1.00	0.03	1.00	1.00	0.99	0.99	0.21	0.98	1.00	0.99
	7	1.00	0.12	1.00	1.00	0.99	1.00	1.00	0.19	1.00	0.94
	8	0.97	0.02	0.97	0.91	0.83	0.93	0.83	0.80	0.19	0.45
	9	1.00	0.13	1.00	1.00	0.90	1.00	1.00	0.62	1.00	0.16
	False alarm	0.20	0.19	0.18	0.16	0.17	0.19	0.21	0.19	0.19	0.16
	False alarm (all)	0.19	0.01	0.18	0.16	0.16	0.18	0.20	0.13	0.19	0.12
	Power	0.85	0.01	0.91	0.79	0.78	0.87	0.77	0.24	0.91	0.29

7. DISCUSSION

The fundamental idea of statistical calibration can be traced back to the work of Ronald Fisher and Jezzy Neyman. Fisher advocated for using p-values, while Neyman favoured testing the null and alternate hypotheses [26]. These distinct approaches to statistical tests somewhat mirror the differences in unsupervised, semi-supervised, and supervised anomaly detection. The concept of calibration has evolved and been applied to contemporary machine learning models, like one-class support vector machines [16] and neural networks [27]. These approaches tend to rely on fixing the false positive rate while associating the normal class with the high density regions [28, 29].

The high density regions are often implicitly or explicitly identified by assuming some kind of reference distribution for the anomalous class, usually a uniform distribution [30]. This approach effectively bridges the gap between unsupervised and supervised learning [8], which provides quantitative criteria for assessing the performance of anomaly detectors and increases the breadth of techniques that can be applied. However, the reference distribution will not be invariant under coordinate transformations [31], so the 'best' anomaly detector will be sensitive to (for example) the units the features are expressed in. Another problem is that the anomalous class is unlikely to resemble the reference distribution. For example, if we are looking for anomalous images (as in sections 5 and 6), then these images are likely to have some kind of structure. The reference of a uniform density would only be useful if we were trying to identify random noise. In fact, for some anomaly detectors that use the high density region for classification [27], it was found they were more confident in their (incorrect) classification of anomalies as normal samples than they were for true normal samples! This parallels our results for MNIST, where we found the '7' detector was more likely to identify '7's as anomalies than '1's.

There is ongoing research into making anomaly detectors more robust against adversarial examples [32], noise [33], for different anomalous classes [34, 35], and concept drift [36, 37]. It is the final area which has the greatest similarity to our work, which focuses on achieving robustness against drift in the composition of the normal class. Although we do not demonstrate it here, there is no reason to think robustness against (say) adversarial examples could not be incorporated alongside sub-class composition robustness.

We have used a single model for each sub-class channel, although it is easy to build an ensemble for each sub-channel, so that the overall architecture consists of an ensemble of ensembles. There are a range of available techniques, including bagging [8], boosting [8], stacking [8], and voting procedures [38]. Similarly, Bayesian techniques can be used to select an appropriate generative model, or an elicitation procedure [39, 40] can be used to rank models, or to perform model checking [14]. This could be useful when we have contextual information about the kinds of anomalies we are interested in detecting.

Unlike supervised learning, where cross-validation is used, there is no de facto standard for evaluating the performance of semi-supervised anomaly detection, especially with multiple sub-classes. This makes it difficult to compare the performance of two or more anomaly detectors. Faria *et al* [37] suggest using a confusion matrix to evaluate multiple sub-class performance of the anomaly detector. The matrix specifies the classification of each class, with an additional column for 'unknown'. Their approach is closer to conventional cross-validation than ours, and uses unsupervised learning to identify clusters that represent different sub-classes, while our sub-classes were generated from pre-existing labels. Mass-volume and excess-mass curves [28, 29] provide metrics that are similar to receiver operating characteristic curves, which makes them easy to interpret. They tend to assume a reference distribution and this can introduce some other problems as discussed above. Identification of anomalies from clusters can be evaluated from standard unsupervised learning criteria, like the sum-of-squares within clusters or the elbow method [8]. These criteria are generally seen as more subjective than other evaluation methods and perhaps less trustworthy. Finally, there can also be a role for human-based evaluation, for example, identifying that certain handwritten digits have an atypical structure. This allows domain knowledge to be incorporated into the evaluation process, but it is hard to automate and what constitutes the normal class may not be obvious to a human either. It is difficult to directly compare these different approaches for evaluating anomaly detector performance because they rely on such diverse principles. There is also little reason to expect their results to be strongly correlated. While this places some burden on the end user to justify their choice of evaluation method, subjective criteria can also appear in supervised learning problems, like the choice of the loss function.

In this technical report, we decided on a threshold for each detector channel, and then estimated the overall false alarm rate. This procedure could be modified so that an overall false alarm rate is selected, then the false alarm rate for the sub-class detectors is adjusted until that overall false alarm rate is obtained. There is also scope for implementing a false discovery rate [41] for each sub-class detector or the ensemble detector. This flexibility allows the general idea to be tailored for a specific application.

8. CONCLUSION

We have developed a procedure for calibrating an anomaly detector such that there is a uniform false alarm rate for each sub-class. The primary benefits of using a uniform false alarm rate across sub-classes are:

- It protects against imbalance in the sub-classes. The traditional practice of using a single threshold can cause sub-classes with few samples to become consistently misclassified as anomalies. This will not occur for separate thresholds, although there can be degraded performance because of small sample sizes.

- It can produce more robust results when there are concerns about algorithmic bias or discrimination for particular sub-classes. For example, when sub-classes represent populations with socially-protected attributes.

- It protects against drift in the sampling distribution. The false alarm rate will remain nearly constant, even if the proportion of each sub-class changes dramatically.

- It allows different algorithms or metrics to be used for each sub-class. For instance, one sub-class could use a Gaussian Naïve Bayes algorithm to classify anomalies with the probability density used as the classification metric, while another sub-class may use a encoder-decoder neural network with the reconstruction error instead.

- The ensemble anomaly detector can be easily customized. Sub-class anomaly detectors can be added or removed without having to retrain any other components. This is not true when a single global threshold is used.

The downsides of using separate thresholds for each sub-group are:

- It can reduce the power of the ensemble anomaly detector for a given false detection rate.

- It can introduce elements of subjectivity when the criteria for creating sub-classes is ambiguous.

On the balance there can be legitimate concerns about a reduction in statistical power. However, there are a number of compelling benefits that may override these concerns in some circumstances.

9. REFERENCES

1. Agrawal, S. and Agrawal, J. (2015) ‘Survey on anomaly detection using data mining techniques’. In: *Procedia Computer Science* **60**, 708–713.
2. Rohling, H. (1983) ‘Radar CFAR thresholding in clutter and multiple target situations’. In: *IEEE transactions on aerospace and electronic systems* (4), 608–621.
3. Lehtomaki, J. J. et al. (2007) ‘CFAR outlier detection with forward methods’. In: *IEEE Transactions on Signal Processing* **55** (9), 4702–4706.
4. Reddy, R. R., Kavya, B and Ramadevi, Y (2014) ‘A survey on svm classifiers for intrusion detection’. In: *International Journal of Computer Applications* **98** (19).
5. Matteoli, S., Diani, M. and Corsini, G. (2010) ‘A tutorial overview of anomaly detection in hyperspectral images’. In: *IEEE Aerospace and Electronic Systems Magazine* **25** (7), 5–28.
6. Ahmed, M., Mahmood, A. N. and Islam, M. R. (2016) ‘A survey of anomaly detection techniques in financial domain’. In: *Future Generation Computer Systems* **55**, 278–288.
7. Henrion, M. et al. (2013) ‘CASOS: a subspace method for anomaly detection in high dimensional astronomical databases’. In: *Statistical Analysis and Data Mining: The ASA Data Science Journal* **6** (1), 53–72.
8. Friedman, J., Hastie, T. and Tibshirani, R. (2001) *The elements of statistical learning*. Vol. 1. 10. Springer series in statistics New York.
9. Gao, J. and Tan, P.-N. (2006) ‘Converting output scores from outlier detection algorithms into probability estimates’. In: *Sixth International Conference on Data Mining (ICDM’06)*. IEEE, 212–221.
10. Menon, A. K. and Williamson, R. C. (2018) ‘A loss framework for calibrated anomaly detection’. In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. Curran Associates Inc., 1494–1504.
11. Galar, M. et al. (2011) ‘A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches’. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* **42** (4), 463–484.
12. LeCun, Y. and Cortes, C. (2010) ‘MNIST handwritten digit database’. In: URL: http://yann.lecun.com/exdb/mnist/.
13. Gelman, A. et al. (2006) ‘Prior distributions for variance parameters in hierarchical models (comment on article by Browne and Draper)’. In: *Bayesian analysis* **1** (3), 515–534.
14. Gelman, A. et al. (2013) *Bayesian data analysis*. CRC press.

15. Breunig, M. M. et al. (2000) ‘LOF: identifying density-based local outliers’. In: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 93–104.

16. Schölkopf, B. et al. (2001) ‘Estimating the support of a high-dimensional distribution’. In: *Neural computation* **13** (7), 1443–1471.

17. Sakurada, M. and Yairi, T. (2014) ‘Anomaly detection using autoencoders with nonlinear dimensionality reduction’. In: *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*, 4–11.

18. Thatte, G., Mitra, U. and Heidemann, J. (2010) ‘Parametric methods for anomaly detection in aggregate traffic’. In: *IEEE/ACM Transactions On Networking* **19** (2), 512–525.

19. Escalante, H. J. and Fuentes, O. (2006) ‘Kernel methods for anomaly detection and noise elimination’. In: *Proceedings of the International Conference on Computing (CORE 2006)*, 69–80.

20. Sheskin, D. J. (2003) *Handbook of parametric and nonparametric statistical procedures*. crc Press.

21. Pedregosa, F. et al. (2011) ‘Scikit-learn: Machine learning in Python’. In: *Journal of machine learning research* **12** (Oct), 2825–2830.

22. Assiri, Y. (2020) ‘Stochastic Optimization of Plain Convolutional Neural Networks with Simple methods’. In: *arXiv preprint arXiv:2001.08856*.

23. Goodfellow, I., Bengio, Y. and Courville, A. (2016) *Deep learning*. MIT press.

24. Sabokrou, M. et al. (2018) ‘Deep-anomaly: Fully convolutional neural network for fast anomaly detection in crowded scenes’. In: *Computer Vision and Image Understanding* **172**, 88–97.

25. Chalapathy, R., Menon, A. K. and Chawla, S. (2018) ‘Anomaly detection using one-class neural networks’. In: *arXiv preprint arXiv:1802.06360*.

26. Berger, J. O. et al. (2003) ‘Could Fisher, Jeffreys and Neyman have agreed on testing?’ In: *Statistical Science* **18** (1), 1–32.

27. Nalisnick, E. et al. (2018) ‘Do deep generative models know what they don’t know?’ In: *arXiv preprint arXiv:1810.09136*.

28. Goix, N. (2016) ‘How to evaluate the quality of unsupervised anomaly detection algorithms?’ In: *arXiv preprint arXiv:1607.01152*.

29. Clémençon, S., Thomas, A. et al. (2018) ‘Mass volume curves and anomaly ranking’. In: *Electronic Journal of Statistics* **12** (2), 2806–2872.

30. Goix, N., Sabourin, A. and Clémençon, S. (2015) ‘On anomaly ranking and excess-mass curves’. In: *Artificial Intelligence and Statistics*, 287–295.

31. Berger, J. (2013) *Statistical decision theory: foundations, concepts, and methods*. Springer Science & Business Media.

32. Carrara, F. et al. (2017) 'Detecting adversarial example attacks to deep neural networks'. In: *Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing*, 1–7.

33. Hu, W., Liao, Y. and Vemuri, V. R. (2003) 'Robust anomaly detection using support vector machines'. In: *Proceedings of the international conference on machine learning*. Citeseer, 282–289.

34. Choi, H., Jang, E. and Alemi, A. A. (2018) 'Waic, but why? generative ensembles for robust anomaly detection'. In: *arXiv preprint arXiv:1810.01392*.

35. Hendrycks, D., Mazeika, M. and Dietterich, T. (2018) 'Deep anomaly detection with outlier exposure'. In: *arXiv preprint arXiv:1812.04606*.

36. Masud, M. et al. (2010) 'Classification and novel class detection in concept-drifting data streams under time constraints'. In: *IEEE Transactions on Knowledge and Data Engineering* **23** (6), 859–874.

37. Faria, E. R., Gama, J. and Carvalho, A. C. (2013) 'Novelty detection algorithm for data streams multi-class problems'. In: *Proceedings of the 28th annual ACM symposium on applied computing*, 795–800.

38. Liu, F. T., Ting, K. M. and Zhou, Z.-H. (2008) 'Isolation forest'. In: *2008 Eighth IEEE International Conference on Data Mining*. IEEE, 413–422.

39. Burgman, M. et al. (2006) 'Eliciting expert judgments: literature review'. In:

40. Hemming, V. et al. (2020) 'Improving expert forecasts in reliability: application and evidence for structured elicitation protocols'. In: *Quality and Reliability Engineering International* **36** (2), 623–641.

41. Benjamini, Y. and Hochberg, Y. (1995) 'Controlling the false discovery rate: a practical and powerful approach to multiple testing'. In: *Journal of the Royal statistical society: series B (Methodological)* **57** (1), 289–300.

This page is intentionally blank

DEFENCE SCIENCE AND TECHNOLOGY GROUP **DOCUMENT CONTROL DATA**		DLM/CAVEAT (OF DOCUMENT) Official
TITLE Uniform Calibration of Anomaly Detectors with Multiple Sub-Classes for Robust Performance		SECURITY CLASSIFICATION (FOR UNCLASSIFIED LIMITED RELEASE USE (U/L) NEXT TO DOCUMENT CLASSIFICATION) Document (O) Title (O)
AUTHOR(S) T. L. Keevers		PRODUCED BY Defence Science and Technology Group DST Headquarters Department of Defence F2-2-03 PO Box 7931 Canberra BC ACT 2610
DST GROUP NUMBER DST-Group-TR-3765	TYPE OF REPORT Technical Report	DOCUMENT DATE September, 2020
TASK NUMBER 17-557	TASK SPONSOR RLMCA	RESEARCH DIVISION Joint and Operations Analysis Division
MAJOR SCIENCE AND TECHNOLOGY CAPABILITY Maritime Capability Analysis		SCIENCE AND TECHNOLOGY CAPABILITY Maritime Mathematical Sciences
SECONDARY RELEASE STATEMENT OF THIS DOCUMENT Approved for public release.		
ANNOUNCABLE No limitations		
CITABLE IN OTHER DOCUMENTS Yes		
RESEARCH LIBRARY THESAURUS Machine learning, Interpretability, Statistics		