machine learning & knowledge extraction

MDPI

Article

Towards Knowledge Uncertainty Estimation for Open Set Recognition

Catarina Pires [1,†], Marília Barandas [1,2,*,†], Letícia Fernandes [1], Duarte Folgado [1,2] and Hugo Gamboa [1,2]

[1] Associação Fraunhofer Portugal Research, Rua Alfredo Allen 455/461, 4200-135 Porto, Portugal; catarina.pires@fraunhofer.pt (C.P.); leticia.fernandes@fraunhofer.pt (L.F.); duarte.folgado@fraunhofer.pt (D.F.); hugo.gamboa@fraunhofer.pt (H.G.)

[2] Laboratório de Instrumentação, Engenharia Biomédica e Física da Radiação (LIBPhys-UNL), Departamento de Física, Faculdade de Ciências e Tecnologia (FCT), Universidade Nova de Lisboa, 2829-516 Caparica, Portugal

* Correspondence: marilia.barandas@fraunhofer.pt

† These authors contributed equally to this work.

Received: 16 September 2020; Accepted: 26 October 2020; Published: 30 October 2020

Abstract: Uncertainty is ubiquitous and happens in every single prediction of Machine Learning models. The ability to estimate and quantify the uncertainty of individual predictions is arguably relevant, all the more in safety-critical applications. Real-world recognition poses multiple challenges since a model's knowledge about physical phenomenon is not complete, and observations are incomplete by definition. However, Machine Learning algorithms often assume that train and test data distributions are the same and that all testing classes are present during training. A more realistic scenario is the Open Set Recognition, where unknown classes can be submitted to an algorithm during testing. In this paper, we propose a Knowledge Uncertainty Estimation (KUE) method to quantify knowledge uncertainty and reject out-of-distribution inputs. Additionally, we quantify and distinguish aleatoric and epistemic uncertainty with the classical information-theoretical measures of entropy by means of ensemble techniques. We performed experiments on four datasets with different data modalities and compared our results with distance-based classifiers, SVM-based approaches and ensemble techniques using entropy measures. Overall, the effectiveness of KUE in distinguishing in- and out-distribution inputs obtained better results in most cases and was at least comparable in others. Furthermore, a classification with rejection option based on a proposed combination strategy between different measures of uncertainty is an application of uncertainty with proven results.

Keywords: uncertainty; machine learning; open set recognition; entropy; out-of-distribution

1. Introduction

Machine Learning (ML) has continuously attracted the interest of the research community motivated by the promising results obtained in many decision-critical domains. Along with the interest arises some concerns related to the trustworthiness and robustness of the models [1]. The notion of uncertainty is of major importance in ML, and a trustworthy representation of uncertainty should be considered as a key feature of any ML method [2,3]. In application domains such as medicine, information about the reliability of the automated decisions is crucial to improve the system's safety [4,5]. Uncertainty also plays a role in AI at the methodological level, such as in active learning [6,7] and self-training [8,9].

Uncertainty is ubiquitous and happens in every single event we encounter in the real-world arising from different sources in various forms. According to the origin of uncertainty, a distinction

between aleatoric uncertainty and epistemic uncertainty is commonly used. Aleatoric uncertainty refers to the inherent randomness in nature, and epistemic uncertainty refers to uncertainty caused by lack of knowledge of the physical world (knowledge uncertainty), as well as the ability to measure and model the physical world (model uncertainty) [10]. In ML, these two sources of uncertainty are usually not distinguished. However, some studies have been proposed showing the usefulness of quantifying and distinguishing the sources of uncertainties in different applications, such as self-driving cars [11], where the authors emphasize "the importance of epistemic uncertainty or 'uncertainty on uncertainty' in these AI-assisted systems", referring to the first accident of a self-driving car that led to the death of the driver, or in medicine, where the authors focused their evaluation on medical data of chest pain patients and their diagnoses [12].

Until recently, almost all evaluations of ML-based recognition algorithms have taken the form of "close set" recognition, where it is assumed that the train and test data distributions are the same and that all testing classes are known at training time [13]. However, a more realistic scenario for deployed classifiers is to assume that the world is an open set of objects, that our knowledge is always incomplete and, thus, the unknown classes should be submitted to an algorithm during testing [14]. For instance, the diagnosis and treatment of infectious diseases relies on the accurate detection of bacterial infections. However, deploying a ML method to perform bacterial identification is challenging, as real data are highly likely to contain unseen classes not seen in training data [15]. Similarly, verification problems for security-oriented face matching or unplanned scenarios for self-driving cars lead to what is called "open set" recognition, in comparison to systems that use "close set" recognition.

Based on the basic recognition categories of classes asserted by Scheirer et al. [16], there are three categories of classes:

1. known classes: classes with distinctly labeled positive training samples (also serving as negative samples for other known classes);
2. known unknown classes: classes labeled negative samples, not necessary grouped into meaningful categories;
3. unknown unknown classes: classes unseen in training.

Traditional supervised classification methods consider only known classes. Some improved the implementation by starting to include known unknown results in models with an explicit "other class" or a detector trained with unclassified negatives. Open Set Recognition (OSR) algorithms, where new classes unseen in training appear in testing, considers the unknown unknown classes category. In this scenario, the classifier needs not only to accurately classify known classes but also effectively reject unknown classes. Although the classification with a reject option is more than 60 years old [17], the focus on rejection has been the ambiguity between classes (aleatoric uncertainty), not for addressing unknown inputs (epistemic uncertainty). Based on Chow's theory, the inputs are rejected if the posterior probability is not sufficiently high based on a predefined threshold that optimizes the ambiguous regions between classes. Epistemic uncertainty is high near the decision boundary, and most classifiers increase confidence with the distance from the decision boundary. Thus, an unknown far from the boundary is not only incorrectly labeled but will be incorrectly classified with very high confidence [14]. Although classification with rejection is related to OSR, it still works under the close set assumption where a classifier rejects to classify a sample due to its low confidence on an overlapping region between classes, which leads to high aleatoric uncertainty. For OSR problems, One-Class Classification (OCC) is commonly used since it tries to focus on the known class and ignore everything else. A popular approach for OSR scenario using OCC is to adapt the familiar Support Vector Machine (SVM) methodology using a one-vs-one or one-vs-all scenario. OSR problems are usually focused on novel, anomaly or outlier detection, so that they are only interested in the epistemic uncertainty due to the lack of knowledge. Although the OSR method indirectly deals with epistemic uncertainty, a proper uncertainty quantification is rarely done.

In this work, we focus on uncertainty quantification of individual predictions, where an input can be rejected for both aleatoric and epistemic uncertainty. Due to the difficulty of dealing with unknown samples, we propose a new method for knowledge uncertainty quantification and combined it with measures of entropy using ensemble techniques. The experimental results are validated on four datasets with different data modalities:

- a human activity dataset using inertial data from smartphones where uncertainty estimation plays an important role in the recognition of abnormal human activities. Indoor location solutions can also benefit from a proper uncertainty estimation where high confident activity classifications should increase positioning accuracy;
- a handwritten digits dataset using images where uncertainty estimation might be used for unrecognized handwritten digits;
- a bacterial dataset using Raman spectra for the identification of bacteria pathogens where novel pathogens often appear and its identification is critical;
- a cardiotocograms dataset using fetal heart rate and uterine contraction where rarely seen conditions of patient data can be accessed through uncertainty estimation.

Overall, the contributions of this work can be summarized as follows:

- a new uncertainty measure for quantifying knowledge uncertainty and rejecting unknown inputs;
- a combination strategy to incorporate different uncertainty measures, evaluating the increase of classification accuracy versus rejection rate by uncertainty measures;
- an experimental evaluation of in- and out-distribution inputs over four different datasets and eight state-of-the-art methods.

2. Uncertainty in Supervised Learning

The awareness of uncertainty is of major importance in ML and constitutes a key element of ML methodology. Traditionally, uncertainty in ML is modeled using probability theory, which has always been perceived as the reference tool for uncertainty handling [2]. Uncertainty arises from different sources in various forms and is commonly classified into aleatoric uncertainty or epistemic uncertainty. Aleatoric uncertainty is related to data and increases with the increase of noise in the observations, which can cause class overlap. On the other hand, epistemic uncertainty is related to the model and the knowledge that is given to it. This uncertainty increases with test samples in out-of-distribution (OOD) regions, and it captures the lack of knowledge of the model's parameters. Epistemic uncertainty can be reduced with the collection of more samples. However, aleatoric uncertainty is irreducible [18]. Although traditional probabilistic predictors may be a viable approach for representing uncertainty, there is no explicit distinction between different types of uncertainty in ML.

2.1. Uncertainty on Standard Probability Estimation

In the standard probabilistic modeling and Bayesian inference, the representation of uncertainty about a prediction is given by the probability of the predicted class. Considering a distribution $p(x, \omega)$ over input features x and labels ω, where $\omega_k \in \{\omega_1, \ldots, \omega_K\}$ consists of a finite set of K class labels, the predictive uncertainty of a classification model $p(\omega_k|x, D)$ trained on a finite dataset $D = \{(x_i, \omega_i)\}_{i=1}^{N}$, with N samples, is an uncertainty measure that combines aleatoric and epistemic uncertainty. The probability of the predicted class, or maximum probability, is a measure of confidence in the prediction that can be obtained by

$$p(\hat{\omega}|x) = \max_k p(\omega_k|x, D) \tag{1}$$

Another measure of uncertainty is the (Shannon) entropy of the predictive posterior distribution, which behaves similarly to maximum probability, but represents the uncertainty encapsulated in the entire distribution:

$$H[p(\omega|x, D)] = -\sum_{k=1}^{K} p(\omega_k|x) \log_2 p(\omega_k|x) \tag{2}$$

Both maximum probability and entropy of the predictive posterior distribution can be seen as measures of the total uncertainty in predictions [19]. These measures of uncertainty for probability distributions primarily capture the shape of the distribution and, hence, are mostly concerned with the aleatoric part of the overall uncertainty. In this paradigm, the classification with a rejection option introduced by Chow [17] suggests that objects are rejected for which the maximum posterior probability is below a threshold. If the classifier is not sufficiently accurate for the task at hand, then one can take the approach not to classify all examples, but only those whose posterior probability is sufficiently high. Chow's theory is suitable when a sufficiently large training sample is available for all classes and when the training sample is not contaminated by outliers [20]. Fumera et al. [21] show that Chow's rule does not perform well if a significant error in probability estimation is present. In that case, a different rejection threshold per class has to be used. In classifiers with a rejection option, the key parameters are the thresholds that define the reject area, which may be hard to define and may vary significantly in value, especially when classes have a large spread. Additionally, Bayesian inference is more akin to aleatoric uncertainty, and it has been argued that probability distributions are less suitable for representing ignorance in the sense of lack of knowledge [2,22].

2.2. Knowledge Uncertainty

Knowledge uncertainty is often associated with novelty, anomaly or outlier detection where the testing samples come from a different population than the training set. Approaches based on generative models typically use densities, $p(x)$, to decide whether to reject a test input that is located in a region without training inputs. These low-density regions, where no training inputs have been encountered so far, represent a high knowledge uncertainty. Traditional methods, such as Kernel Density Estimation (KDE), can be used to estimate $p(x)$, and often threshold-based methods are applied on top of the density where a classifier can refuse to predict a test input in that region [23]. Related to this topic is the closed-world assumption, which is often violated in experimental evaluations of ML algorithms. Almost all supervised classification methods assume that all train and test data distributions are the same and that all classes in the test set are present in the training set. However, this is unrealistic for many real-world applications where new unseen classes in training appear in testing. This problem has been studied under different names with varying levels of difficulty, including the previously mentioned classification with rejection option, OOD detection and OSR [24]. In the classification with a rejection option, the test distribution has usually the same classes as the training distribution, and the classifier rejects inputs it cannot confidently classify. OOD detection is the ability of a classifier to reject a novel input rather than assigning it an incorrect label. In this setting, OOD inputs are usually considered as outliers that come from entirely different datasets. This topic is particularly important in deep neural networks and has been recognized by several studies showing that deep neural networks usually predict OOD inputs with high confidence [25,26]. The OSR approach is similar to OOD detection and can be viewed as tackling both the classification and novelty detection problem at the same time. Contrary to the OOD detection, the novel classes that are not observed during training are often made up of the remaining classes in the same dataset. This task is probably the hardest one because the statistics of a class are often very similar to the statistics of other classes in the dataset [24]. In each case, the goal is to correctly classify inputs that belong to the same distribution as the training set and to reject inputs that are outside this distribution.

A number of approaches have been proposed in the literature to handle unknown classes in the testing phase [27,28]. A popular and promising approach for open-set scenarios is the OCC since it focuses on the known class and ignores any additional class. OCC problems consist of defining the

limits of all, or most, of the training data, by having a single target class. All the samples outside those limits will be considered outliers [29]. SVM models are commonly used in OCC problems by fitting a hyperplane that separates normal data points from outliers in a high-dimensional space [30]. Typically, SVM model separates the training set containing only samples from the known classes by the widest interval possible. Training samples on the OOD region are penalized, and the prediction is made by assigning the samples to the known or unknown region. Binary classification with the one-vs-all approach can also be applied to the open-set recognition [31]. In this scenario, the most confident binary classifier which classifies as in-distribution is chosen to predict the final class of the multiclass classifier. When there is no in-distribution classification from the binary classifiers, the test sample is classified as unknown. Different adaptions of OCC and variations of the SVM have been applied for OSR aiming at minimizing the risk of the unknown classification [13,16,32]. However, a drawback of these methods is the need for re-training the models from scratch, at a relatively high computational cost, when new classes are discovered and become available. Therefore, they are not well-suited for incremental updates or scalability required for open-world recognition [33]. Distance-based approaches are more suitable to open-world scenarios, since the addition of new classes to existing classes can be made at near-zero cost [34]. Distance-based classifiers with a rejection option are easily applied to OSR because the classifiers can create a bounded known space in the feature space, rejecting test inputs that are far away from training data. For instance, the Nearest Class Mean (NCM) classifier is a distance-based classifier that represents classes by their mean feature vector of its elements [34]. The problem for most of the methods dealing with rejection by thresholding the similarity score is the difficulty to determine such a threshold that defines whether a test input is an outlier or not. In this context, Júnior et al. [35] extended the traditional close-set Nearest Neighbor classifier applying a threshold on the ratio of similarity scores of the two most similar classes and called it Open Set Nearest Neighbors (OSNN).

2.3. Combined Approaches for Uncertainty Quantification

The previously mentioned studies do not explicitly quantify uncertainty nor distinguish the different sources of uncertainty. However, we argue that probabilistic methods are more concerned with handling aleatoric uncertainty to reject low-confident inputs, and OSR algorithms are mainly focused on effectively rejecting unknown inputs, which intrinsically have a high epistemic uncertainty due to the lack of knowledge.

However, in a real-world applications, and advocating a trustworthy representation of uncertainty in ML, both sources are important, and a proper distinction between them is desirable, all the more in safety-critical applications of ML. Motivated by such scenarios, several works have been developed for uncertainty quantification showing the usefulness of distinguishing both types of uncertainty in the context of Artificial Intelligence (AI) safety [11,12]. Some initial proposals for dealing with OSR and properly quantify uncertainty can already be found in the literature, mostly in the area of deep neural networks [19,36].

An approach for the quantification of aleatoric, epistemic, and total uncertainty (given by the sum of the previous two uncertainties) separately is to approximate these measures by means of ensemble techniques [19,37], presenting the posterior distribution $p(h|D)$ by a finite ensemble of M hypotheses, $H = \{h_1, \ldots, h_M\}$, that map instances x to probability distributions on outcomes. This approach was developed in the context of neural networks for regression [38], but the idea is more general and can also be applied to other settings, such as in the work of Shaker et al. [37] where the measures of entropy were applied using the Random Forest (RF) classifier. Using an ensemble approach, $\{p(\omega_k|x, h_i)\}_{i=1}^{M}$, a measure of total uncertainty can be approximate by the entropy of predictive posterior given by

$$H[E_{p(h|D)}p(\omega|x,h)] \approx H\left[\frac{1}{M}\sum_{i=1}^{M} P(\omega|x,h_i)\right] \tag{3}$$

An ensemble estimate of aleatoric uncertainty considers the average entropy of each model in an ensemble. The idea is that by fixing a hypotheses h, the epistemic uncertainty is essentially removed. However, since h is not precisely known, the aleatoric uncertainty is measured in terms of the expectation of entropy with regard to the posterior probability:

$$E_{p(h|D)}H[p(\omega|x,h)] \approx \frac{1}{M}\sum_{i=1}^{M} H[p(\omega|x,h_i)] \tag{4}$$

Epistemic uncertainty is measured in terms of mutual information between hypotheses and outcomes, which is a measure of the spread of an ensemble. Epistemic uncertainty can be expressed as the difference of the total uncertainty, captured by the entropy of expected distribution, and the expected data uncertainty, captured by expected entropy of each member of the ensemble [19].

$$I(\omega,h|x,D) = H[E_{p(h|D)}p(\omega|x,h)] - E_{p(h|D)}H[p(\omega|x,h)] \tag{5}$$

Epistemic uncertainty is high if the distribution $p(\omega|x,h)$ varies a lot for different hypotheses h with high probability but leading to quite different predictions.

Finally, besides the uncertainty quantification into aleatoric, epistemic and total uncertainty, there are also open questions regarding the empirical evaluation of different methods, since data usually do not contain information about ground truth uncertainty. Commonly the evaluation is done indirectly through the increase of successful predictions, such as the Accuracy-Rejection (AR) curve which depicts the accuracy of a predictor as a function of the percentage of rejections [2].

3. Proposed Method

In this paper, we are interested in predictive uncertainty, where the uncertainty related to the prediction $\hat{\omega}$ over input features x is quantified in terms of aleatoric and epistemic (and total) uncertainty. Due to the complexity of novelty, anomaly or outlier detection, a specific uncertainty measure to deal with knowledge uncertainty is also proposed, summarizing the relationship of novelty detection and multiclass recognition as a combination of OSR problems. We formulate the problem as traditional OSR where a model is trained only over in-distribution data, denoted by a distribution Q_{in}, and tested on a mixture distribution with in- and out-distribution inputs, drawn from Q_{in} and Q_{out} where the latter represents the out-distribution data. Given a finite training set $D = \{(x_i, \omega_i)\}_{i=1}^{N}$ drawn from Q_{in} where x_i is the i-th training input and $\omega_k \in \{\omega_1, \ldots, \omega_K\}$ is the finite set of class labels, a classifier is trained to correctly identify the class label from Q_{in} and to reject unknown classes not seen in training from Q_{out} by an uncertainty threshold. In Figure 1, the main steps of the proposed approach are summarized. Besides the traditional classification processes, our method learns the feature density estimation from the training data to feed the Knowledge Uncertainty Estimation measure used for rejecting test inputs with an uncertainty value higher than the learned threshold during training. Finally, each prediction is also quantified using entropy measures in terms of total uncertainty or aleatoric and epistemic uncertainty if ensemble techniques are used.

Uncertainty is modeled through a combination of a normalized density estimation over input feature space for each known class. Assuming an input x_i represented by P-dimensional feature vectors, where $f_j \in \{f_1, \ldots, f_P\}$ is the feature vector in a bounded area of the feature space, an independent density estimation of the P features conditional by the class label is estimated and normalized by its maximum density, in order to set all values in the interval $[0,1]$. Thus, each feature density is transformed on an uncertainty distance, d_{unc}, assuming values in $[0,1]$, where 1 represents the maximum density seen in training, and near-zero values represent low-density regions where no training inputs were observed during training. The combination between each feature distance is computed by the product rule over whole features. Thus, given a test input x_i from class ω_k the

uncertainty is measured using the proposed Knowledge Uncertainty Estimation method, $KUE(x_i|\omega_k)$, calculated by

$$KUE(x_i|\omega_k) = 1 - \left(\prod_{j=1}^{P} d_{unc}(f_j|\omega_k, x_i)\right)^{\frac{1}{P}} \quad (6)$$

An example of the proposed uncertainty measure distribution over a Bacteria dataset (we will present and discuss this dataset in Section 4.1) with a 30-dimensional feature vector, 28 known classes and 2 unknown classes is shown in Figure 2.

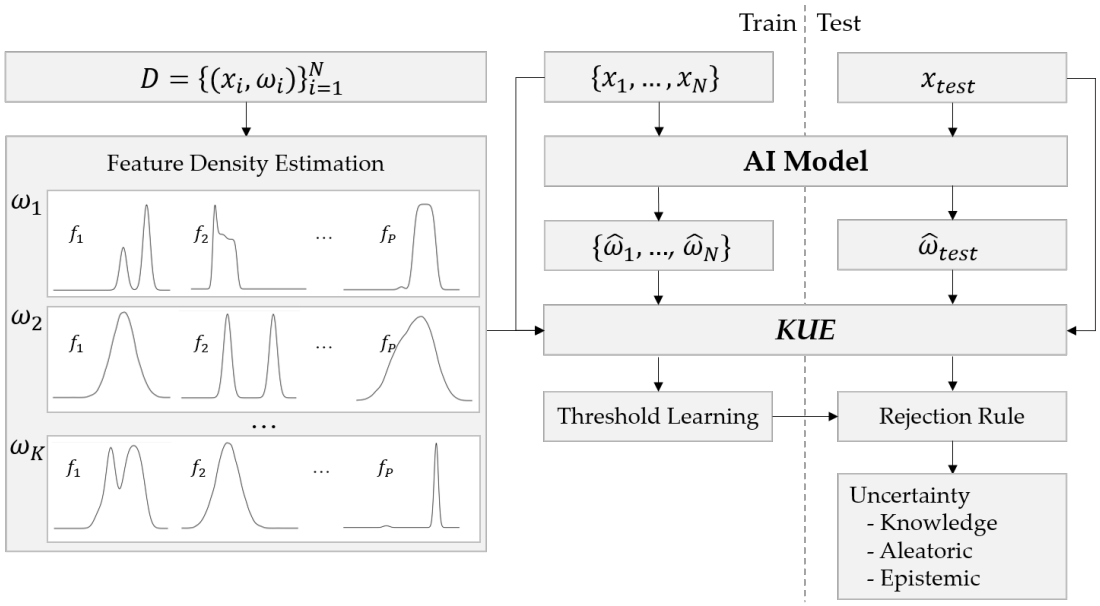

Figure 1. Overview of the main steps of the proposed approach for predictive uncertainty.

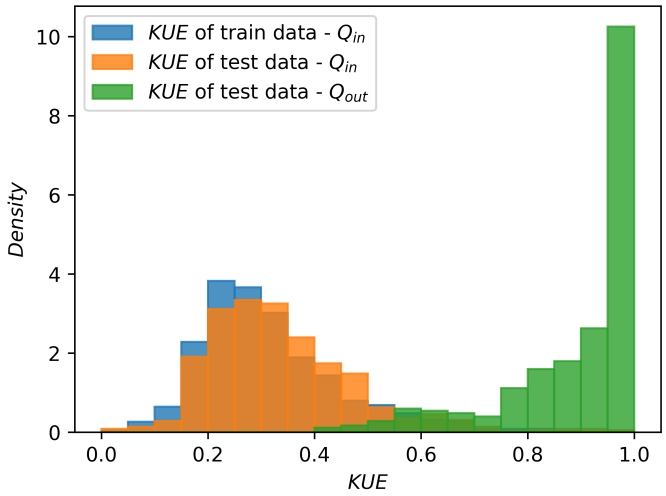

Figure 2. Histogram of the density based on the KUE values, from a train set drawn from in-distribution data, Q_{in}, and test set drawn from both in- and out-distribution data, Q_{in} and Q_{out}, respectively.

A common approach to define a threshold for OOD or even to tune a model's hyperparameters is to use a certain amount of OOD as validation data. However, this approach is unrealistic due to the proper definition of OOD inputs that come from an unknown distribution, leading to compromised performance in real-world applications, as Shafei et al. [39] showed in their recent study. Therefore, we argue that a more realistic approach is to learn a threshold only from in-distribution data. Due to the differences between data from different datasets, learning a global threshold for all

datasets is not a reliable approach. Therefore, our hypothesis is that if we learn the training uncertainty distribution for each class within a dataset, there is a specific threshold for each distribution that will bound our uncertainty space, so input samples that fall outside the upper bound threshold are rejected. The upper bound threshold is defined based on a predefined percentile from the training uncertainty distribution. The percentile choice is defined according to different applications scenarios, whether the end-user is willing reject more or less in-distribution samples. As train and test in-distribution data come from the same distribution it is expected that the percentage of reject samples from test data will represent approximately 10% if the chosen percentile is set to 90%. From this 10% we can also argue that a certain percentage can represent classification errors or, if rejected samples were correctly classified, the classification was done under limited evidence so that a high uncertainty is associated with that decision. Thus, the rejection rule for input sample x_i for in- and out-distribution is given by $g(x_i|\omega_k)$ in Equation (7), where $P_r[U(\omega_k)]$ represents the uncertainty value for the r-th percentile of the train uncertainty data distribution associated with class ω_k. The output values -1 and 1 mean that the input sample x_i is rejected or accepted, respectively:

$$g(x_i|\omega_k) = \begin{cases} -1 & \text{if } KUE(x_i|\omega_k) > P_r[KUE(\omega_k)] \\ 1 & otherwise \end{cases} \tag{7}$$

Since the proposed measure only deals with knowledge uncertainty, besides the in- and out-distribution detection we also combined our proposed approach with the uncertainty measures presented in Equations (3)–(5) to quantify total, aleatoric and epistemic uncertainty, respectively.

4. Experiments

In this section, we describe the datasets used for the experiments and provide a detailed description of two experimental results: (1) classification with a rejection option based on a combination of our proposed KUE method and measures of predictive uncertainty from Section 2.3; (2) effectiveness of KUE in distinguishing in- and out-distribution inputs.

4.1. Datasets

We designed experiments on different data modalities to evaluate our method and to compare it with state-of-the-art methods. The experiments were performed on a real-world bacterial dataset (https://github.com/csho33/bacteria-ID (accessed on June 2020)) from [40] and on a set of standard datasets from the UCI repository [41], namely HAR (https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones (accessed on February 2020)), Digits (https://archive.ics.uci.edu/ml/datasets/Optical+Recognition+of+Handwritten+Digits (accessed on July 2020)) and Cardio (https://archive.ics.uci.edu/ml/datasets/Cardiotocography (accessed on July 2020)). As the datasets do not explicitly contain OOD samples, we adopted a common approach seen in literature to simulate a realistic OSR problem, by re-labeling some of the known classes as unknown [20]. The datasets instances, attributes, classes and OOD combinations are summarized in Table 1. In the following, a brief description of each dataset is given:

- **Bacteria:** This dataset includes bacterial Raman spectra of 30 common bacterial pathogens treated by eight antibiotics. For the feature extraction, we split each Raman spectra in 125 equal-sized windows corresponding to different wavenumber ranges. For each range we extracted minimum, maximum and mean features and applied a feed-forward feature selection algorithm, obtaining a set of 50 features. Due to the high number of possible combinations for known and unknown classes, we grouped the 30 classes by empiric antibiotic treatment, resulting in eight OOD combinations that vary in the number of known and unknown classes. Details of the different combinations are available in Appendix A.
- **HAR:** This dataset contains six different human activities (walking, walking upstairs, walking downstairs, sitting, standing and laying) recorded with accelerometer and gyroscope

smartphone sensors. This dataset has a set of 561 features available for which we applied a feed-forward feature selection algorithm. For the known and unknown classes split, we defined nine OOD combinations, considering each of the six individual classes as unknown and three additional combinations of classes defined as stairs (walking upstairs and walking downstairs), dynamic (walking, walking upstairs and walking downstairs), and static (sitting, standing, and laying).

- **Digits:** This dataset is composed by 10 handwritten digits (from 0 to 9) and 64 attributes. We used each class as unknown resulting in a total of 10 OOD combinations.
- **Cardio:** This dataset contains measurements of fetal heart rate and uterine contraction on cardiotocograms. The dataset has 10 classes and additional labeling as (Normal, Suspicious and Pathologic). Thus, we trained the model using only classes labeled as Normal and consider the unknown classes from the labeling Suspicious and Pathologic.

Table 1. Datasets used and their characteristics.

Dataset	# Instances	# Attributes	# Classes	# OOD Combinations
Bacteria	3000	50 *	30	8
HAR	1800	9 *	6	9
Digits	5620	64	10	10
Cardio	2126	23	10	2

* Feature size was reduced by a feed-forward feature selection algorithm.

4.2. Classification with Rejection Option

The classification with a rejection option based on measures of predictive uncertainty is presented in this section, where we described the evaluation metrics, the uncertainty quantification methods and a detailed description of the experimental results.

4.2.1. Evaluation Metric

The empirical evaluation of methods for quantifying uncertainty is a non-trivial problem, due to the lack of ground truth uncertainty information. A common approach for indirectly evaluating the predicted uncertainty measures is using AR curves [2]. According to Nadeem et al. [42], an accuracy rejection curve is a function representing the accuracy of a classifier as a function of its rejection rate. Therefore, the AR curves plot the rejection rate of the metrics (from 0 to 1) against the accuracy of the classifier. Since the accuracy is always 100% when the rejection rate is 1, all curves converge to the point $(1, 1)$, and they start from the point $(0, a)$, where a is the initial accuracy of the classifier, with 0% of rejected samples.

4.2.2. Uncertainty Quantification Methods

The methods used for the classification with rejection option through uncertainty measures are the following:

1. Knowledge uncertainty measured by our proposed KUE (Equation (6)) using KDE for the probability density function of each feature and Scott's rule [43] for the kernel bandwidth;
2. Total uncertainty approximated by the entropy of the predictive posterior using Equation (3);
3. Aleatoric uncertainty measured with the average entropy of each model in an ensemble using Equation (4);
4. Epistemic uncertainty expressed as the difference between the total uncertainty and aleatoric uncertainty given by Equation (5).

Although KUE can be applied to any ML model with feature level representation, the measures of total, aleatoric and epistemic uncertainty are approximated using an ensemble approach. Therefore,

a RF classifier with 50 trees and a bootstrap approach to create diversity between the trees of the forest was used for this experiment.

4.2.3. Experimental Results

As we explained in Section 3, our classification rule depends on the choice of a predefined percentile of the train data uncertainty values, which can vary depending on the application. As we hypothesized that the percentage of the reject in-distribution data depends on the chosen percentile, we computed the True Positives Rate (TPR) and False Positives Rate (FPR) for a range of train percentiles, as shown in Figure 3, considering the positive samples being the samples classified as out-distribution and the negative samples the ones classified as in-distribution. Additionally, as in- and out-distribution detection does not consider the prediction error, we also computed an adjusted FPR where the classification errors were removed from FPR, i.e., the in-distribution inputs that have an uncertainty value higher than the chosen percentile and were misclassified by the model were removed from the FPR. This adjusted FPR is represented in Figure 3 by FPR*. This metric has an important meaning for our method since our method depends on the classification performance, where the uncertainty of the misclassified inputs is computed using the probability densities of a different class. Therefore, it is expected that the uncertainty value is high for both OOD and for misclassified inputs.

In Figure 3, the variation of TPR, FPR and FPR* according to the train percentile (which defines the uncertainty threshold) for each of the four datasets is presented. Each graph comprises the average and the standard deviation of all OOD combinations for each dataset. As expected, the increase of the train percentile represented almost a linear decrease in FPR, since the distributions of the train data were similar to the distributions of the in-distribution test data. We can see that the FPR* was also linear in all datasets, and both FPR and FPR* converged to 0. This means that, depending on the application and on the risk associated with decisions, we can define the train percentile based on how many in-distribution test samples we are willing to reject. On the other hand, TPR followed a different behavior, where a high percentile could reject most of the OOD samples and a few in-distribution test samples or reject a minor percentage of both in- and out-distribution inputs.

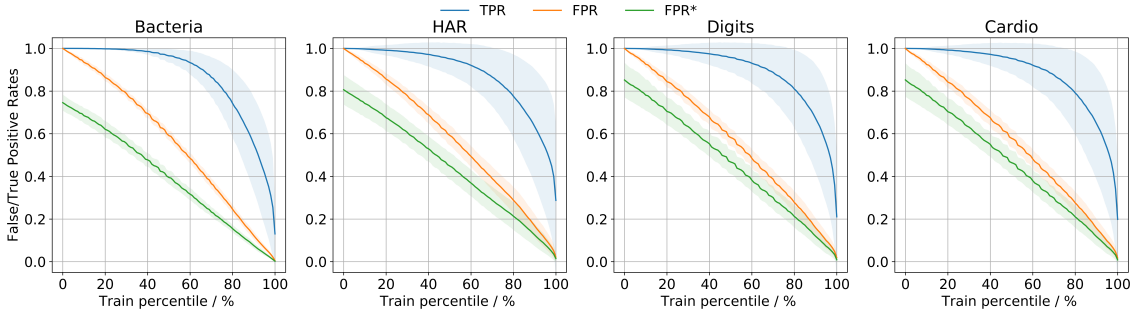

Figure 3. Relation between TPR, FPR, FPR* and train percentiles. FPR* stands for an adjust FPR, where misclassified inputs were removed from the FPR.

Since our proposed approach only deals with knowledge uncertainty, we also quantified the uncertainty in terms of total, aleatoric and epistemic uncertainty by means of ensemble techniques. Although epistemic uncertainty is a combination of model and knowledge uncertainty, its quantification is limited to the use of ensemble approaches. Moreover, specialized OOD detection methods would probably perform better for the knowledge uncertainty quantification. As our approach is only specialized in OOD detection, and total uncertainty encapsulates the uncertainty of the entire distribution, a combination between them should ideally perform better for the overall classification accuracy. Thus, we combined uncertainties by firstly reject input samples based on our method until the chosen percentile and then rejecting samples based on total uncertainty.

For evaluation we used AR curves where the prediction uncertainty can be assessed indirectly by the improved prediction as a function of the percentage of the rejection. If we have a reliable

measure of uncertainty involved in the classification of test inputs, then uncertainty estimation should correlate with the probability of making a correct decision, so that the accuracy should be improved with increasing rejection percentage, and AR curves should be monotone increasing. The comparison between different methods using AR curves should be based on the required accuracy level and/or the appropriate rejection rate [44]. Since we are comparing methods derived from the same classifier, the AR curves always had the same starting accuracy for all methods. Consequently, the relevant variable for the empirical evaluation is the rejection rate. Thus, we moveed vertically over the graph to see which method had a higher accuracy for a certain rejection rate. The AR curves were obtained by varying the rejection threshold, where samples with the highest uncertainty values were rejected first.

In Figure 4, the average rejection rate against the average accuracy for KUE, total, aleatoric and epistemic uncertainty is presented. The proposed combination is also shown in black, and the optimal rejection is represented by the dashed line. The optimal AR curve was computed by rejecting all OOD samples as well as misclassified samples in a row. In order to obtain the AR curves we ran 10 random repetitions using 15% of OOD inputs and using an uncertainty train percentile for the our proposed combination of 95%. As we can see in Figure 4, almost every curve over the different OOD combinations increased the accuracy with the increase of the rejection rate percentage. It is also interesting to note that even with only 15% of OOD inputs, our method always presented the monotone dependency between reject rate and classification accuracy, which means that our method also behaved quite well over the misclassified inputs. Regarding the proposed combination, the AR curve was always better or similar to the total uncertainty. Besides that, we observe that the AR curves tendency for the KUE method did not vary much between different OOD combinations, contrary to the aleatoric and epistemic uncertainty. The AR curves for the other datasets can be found in Appendix B.3.

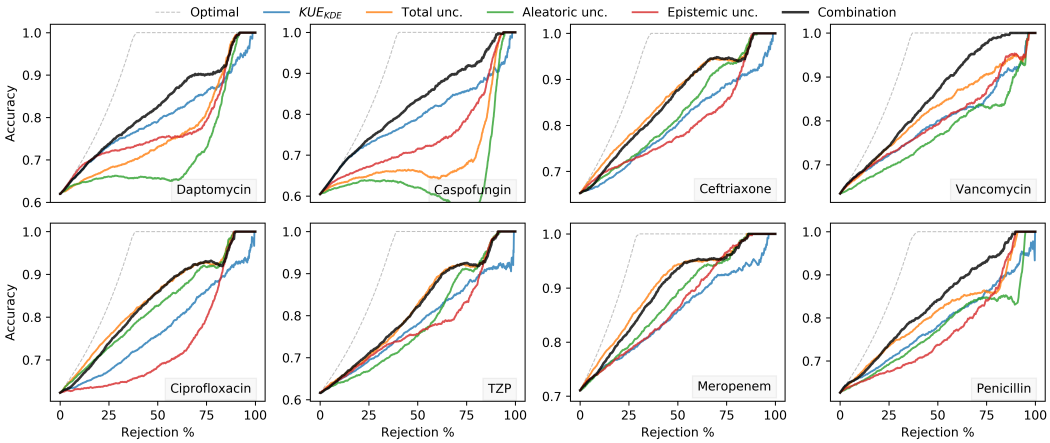

Figure 4. AR curves for aleatoric, epistemic and total uncertainty for the Bacteria dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the antibiotic name used for each OOD inputs combination.

4.3. Out-of-Distribution Detection

The effectiveness of KUE in distinguishing in- and out-distribution inputs is presented in this section, where we described the evaluation metrics, the inference methods for comparison and a detailed description of the experimental results.

4.3.1. Evaluation Metrics

Most of the recent studies employ the Area Under the ROC (AUROC) metric, which is a threshold-independent performance method [45] for evaluating OOD detection methods. The Receiver Operating Characteristic (ROC) curve depicts the relationship between the TPR and FPR. The AUROC can be interpreted as the probability that a positive example is assigned a higher detection score than a negative example [46]. Consequently, a random positive example detector corresponds to

a 50% AUROC, and a perfect detector corresponds to an AUROC score of 100%. Hendrycks and Gimpel [47] stated a more qualitative interpretation of AUROC values as follows: excellent: 90–100%, good: 80–90%, fair: 70–80%, poor: 60–70%, fail: 50–60%. This interpretation was adopted for the overall evaluation over all datasets.

The Area Under the Precision-Recall (AUPR) is another threshold-independent metric frequently applied for OOD detection evaluation [48]. The Precision-Recall (PR) curve is a graph showing the precision and recall against each other. The baseline detector has an AUPR approximately equal to the precision [49], and a perfect detector has an AUPR of 100%. Consequently, the base rate of the positive class greatly influences the AUPR, so the AUPR-In and AUPR-Out are commonly used, where in-distribution and out-distribution inputs are specified as negatives and positives, respectively. The AUPR is sometimes deemed as more informative than AUROC because the AUROC is not ideal when the positive class and negative class have greatly differing base rates.

As for the evaluation of OOD detection we used the same number of in-distribution and out-distribution inputs, and the main metric employed for the evaluation of experiments is the AUROC. Additional details about AUPR-In and AUPR-Out can be found in Appendix B.2.

4.3.2. Inference Methods

For the in- and out-distribution detection we compare the following classification approaches:

1. $\boldsymbol{KUE_{KDE}}$: Our proposed method for knowledge uncertainty estimation using KDE as the probability density function and Scott's rule [43] for the kernel bandwidth;
2. $\boldsymbol{KUE_{Gauss}}$: Our proposed method for knowledge uncertainty estimation a using Gaussian distribution as probability density function;
3. $\boldsymbol{p(\hat{\omega}|x)}$: Maximum class probability. Although standard probability estimation is more akin to the to aleatoric part of the overall uncertainty, OOD data tend to have lower scores than in-distribution data [47].
4. $\boldsymbol{H[p(\omega|x)]}$: Total uncertainty modeled by the (Shannon) entropy of the predictive posterior distribution. High entropy of the predictive posterior distribution, and therefore a high predictive uncertainty, suggests that the test input may be OOD [2].
5. $\boldsymbol{I[\omega, h]}$: Epistemic uncertainty measured in terms of the mutual information between hypotheses and outcomes. High epistemic uncertainty means that $\boldsymbol{p(\omega|x, h)}$ varies a lot for different hypotheses $\boldsymbol{h}$ with high probability. The existence of different hypotheses, all considered probable but leading to quite different predictions, can indeed be seen as a sign of OOD input [2].
6. **OCSVM**: One-Class SVM (OCSVM) introduced by Schölkopf et al. [50] using a radial basis function kernel to allow a non-linear decision boundary. OCSVM learns a decision boundary in feature space to separate in-distribution data from outlier data.
7. $\textbf{SVM}^{\textbf{ovo}}$: Multiclass SVM with one-vs-one approach and calibration across classes using a variation of Platt's extended by [51].
8. $\textbf{SVM}^{\textbf{ova}}$: One-vs-all multiclass strategy fitting one SVM per class.
9. **NCM**: Nearest Class Mean Classifier using a probabilistic model based on multiclass logistic regression to obtain class conditional probabilities [33].
10. **OSNN**: Open Set Nearest Neighbor introduced by Júnior et al. [35] using a distance ratio based on the Euclidean distance of two most similar classes.
11. **IF**: Isolation Forest (IF) introduced by Liu et al. [52] for anomaly detection using an implementation based on an ensemble of an extremely randomized tree regressor.

Note that epistemic uncertainty is approximated by means of ensemble techniques, which is the representation of the posterior distribution by a finite ensemble of hypotheses. For this reason, to make the comparison fair between baseline methods 1–5, we chose a RF classifier for the experiments analysis. Nevertheless, and since different classifiers have different accuracies for classification of the

very same data, a comparison study was carried out on a set of classical algorithms: RF, K-Nearest Neighbors (KNN), Naive Bayes (NB), SVM and Logistic Regression (LR). The detailed results for our method using different algorithms can be seen in Appendix B.1.

4.3.3. Experimental Results

For the problem of detecting OOD inputs we trained the models using only in-distribution inputs, ignoring the OOD inputs during training. For the final evaluation, we randomly selected the same number of in-distribution and out-distribution inputs from the test set. Table 2 compares our method using two variants of the feature modeling (KDE and Gaussian) with the methods mentioned in Section 4.3.2. The OOD names shown in Table 2 indicate the assumed unknown classes for each dataset. Regarding the Bacteria dataset, the names are the antibiotic treatments used to group the unknown classes, which are detailed in Appendix A. The AUROC is the average results over 10 random repetitions for a total of 29 OOD combinations over 4 different datasets.

From a detailed analysis of Table 2 we notice that, in the majority of the OOD combinations, our method obtained better or comparable AUROC with other methods. Moreover, the proposed method performed more consistently for different OOD combinations, unlike the other methods that showed unstable behaviors, where the standard deviation was very large over all combinations considered. For instance, the OCSVM presented the highest performance on the Digits and Cardio datasets. However, in the other datasets its performance varied a lot depending on the assumed unknown classes, with a poor performance on several OOD combinations. As an example in Figure 5, we show the ROC curves for the Caspofungin and Ciprofloxacin OOD combinations of the Bacteria dataset, representing the best and the worst performance of our method in the Bacteria dataset, respectively. It is interesting to note that, after our method, OCSVM presented the highest performance for Caspofungin. However, for Ciprofloxacin the OCSVM performance was lower than random. A similar behavior happened with the maximum class probability, $p(w|x)$, and the total uncertainty, $H[p(w|x)]$, which are the best methods to detect OOD samples on Ciprofloxacin combination and the worse in the case of antibiotic Caspofungin. Both methods had the same behavior over all combinations due to their intrinsic dependency. Maximum class probability can also be seen as a measure of the total uncertainty in predictions. Regarding epistemic uncertainty, although it obtained a few poor performances, it seemed to have more consistent behavior than the other methods. Additionally, it can be seen that all methods obtained high AUROC and comparable performance for all combinations of the Digits dataset. Comparing our two feature modeling strategies (KDE and Gaussian), we observed that results were similar, probably due to the fact that the feature modeling using the KDE in our datasets was approximated to a Gaussian distribution.

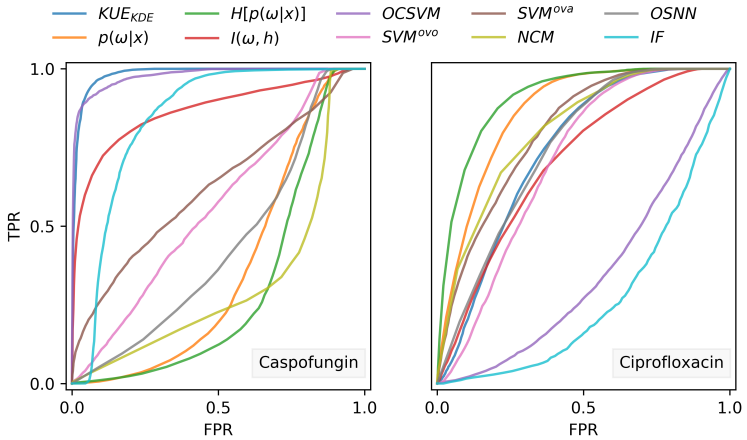

Figure 5. ROC curves for OOD detection using our KUE method and baseline methods on Caspofungin and Ciprofloxacin OOD combinations of the Bacteria dataset. Caspofungin and Ciprogloxacin represent the best and the worst performances of our KUE method, respectively.

Table 2. AUROC for detecting OOD test inputs using two variants of KUE (KDE and Gaussian) and other baseline methods on 4 datasets. The Mean and Standard Deviation (SD) over OOD combinations is presented after each dataset. All values are averages over 10 consecutive repetitions.

		AUROC										
	OOD	KUE_{KDE}	KUE_G	$p(\hat{\omega}\|x)$	$H[p(\omega\|x)]$	$I[\omega, h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Bacteria	Daptomycin	0.91	0.90	0.67	0.68	0.88	0.57	0.89	0.79	0.60	0.59	0.66
	Caspofungin	0.98	0.98	0.37	0.31	0.87	0.98	0.56	0.62	0.32	0.44	0.93
	Ceftriaxone	0.82	0.81	0.82	0.85	0.85	0.50	0.91	0.83	0.82	0.77	0.35
	Vancomycin	0.87	0.87	0.67	0.66	0.80	0.73	0.82	0.73	0.64	0.61	0.74
	Ciprofloxacin	0.74	0.74	0.86	0.91	0.71	0.35	0.88	0.80	0.81	0.75	0.22
	TZP	0.88	0.88	0.76	0.75	0.89	0.65	0.89	0.80	0.84	0.80	0.51
	Meropenem	0.77	0.77	0.87	0.87	0.78	0.48	0.87	0.84	0.83	0.78	0.43
	Penicillin	0.77	0.77	0.73	0.74	0.67	0.60	0.81	0.83	0.70	0.69	0.65
	Mean	0.84	0.84	0.72	0.72	0.81	0.61	0.83	0.78	0.70	0.68	0.56
	SD	0.08	0.08	0.15	0.18	0.08	0.18	0.11	0.07	0.17	0.12	0.21
HAR	Walking	0.69	0.67	0.77	0.78	0.80	0.21	0.73	0.70	0.74	0.73	0.23
	Upstairs	0.86	0.86	0.79	0.82	0.85	0.42	0.71	0.67	0.80	0.72	0.44
	Downstairs	0.84	0.84	0.72	0.70	0.72	0.88	0.55	0.73	0.45	0.70	0.89
	Sitting	0.73	0.75	0.52	0.52	0.66	0.69	0.50	0.43	0.51	0.46	0.66
	Standing	0.54	0.50	0.62	0.67	0.82	0.58	0.54	0.70	0.55	0.44	0.58
	Laying	0.99	0.99	0.25	0.26	0.39	0.99	0.14	0.20	0.79	0.51	1.00
	Stairs	0.90	0.89	0.54	0.58	0.73	0.78	0.25	0.39	0.49	0.43	0.80
	Dynamic	1.00	1.00	0.72	0.76	0.75	0.82	0.57	0.58	0.87	0.92	0.86
	Static	1.00	0.98	0.70	0.69	0.75	0.99	0.29	0.58	0.50	0.81	0.99
	Mean	0.84	0.83	0.63	0.64	0.72	0.71	0.48	0.55	0.63	0.64	0.72
	SD	0.15	0.16	0.16	0.16	0.13	0.25	0.19	0.17	0.15	0.17	0.25

Table 2. *Cont.*

		AUROC										
	OOD	KUE_{KDE}	KUE_{G}	$p(\hat{\omega}\|x)$	$H[p(\omega\|x)]$	$I[\omega,h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Digits	0	0.93	0.95	0.90	0.90	0.97	1.00	0.95	0.90	0.90	0.98	0.80
	1	0.68	0.81	0.88	0.89	0.84	0.95	0.78	0.87	0.85	0.91	0.63
	2	0.90	0.92	0.90	0.89	0.87	0.99	0.90	0.90	0.90	0.95	0.84
	3	0.75	0.81	0.90	0.87	0.82	0.97	0.86	0.82	0.86	0.95	0.64
	4	0.94	0.95	0.88	0.89	0.96	0.99	0.85	0.92	0.84	0.93	0.94
	5	0.87	0.88	0.90	0.89	0.89	0.98	0.84	0.85	0.88	0.96	0.67
	6	0.92	0.93	0.88	0.88	0.97	0.99	0.95	0.85	0.93	0.97	0.81
	7	0.91	0.92	0.94	0.95	0.91	0.99	0.89	0.87	0.93	0.96	0.86
	8	0.90	0.87	0.97	0.98	0.94	0.97	0.96	0.95	0.92	0.96	0.45
	9	0.88	0.89	0.89	0.87	0.84	0.94	0.90	0.87	0.86	0.96	0.61
	Mean	0.87	0.89	0.90	0.90	0.90	0.98	0.89	0.88	0.89	0.95	0.73
	SD	0.08	0.05	0.03	0.03	0.05	0.02	0.05	0.04	0.03	0.02	0.14
Cardio	Suspect	0.67	0.65	0.33	0.31	0.45	0.75	0.48	0.50	0.31	0.67	0.71
	Pathologic	0.83	0.85	0.36	0.31	0.51	0.98	0.23	0.75	0.30	0.66	0.94
	Mean	0.75	0.75	0.34	0.31	0.48	0.86	0.36	0.62	0.30	0.66	0.82
	SD	0.08	0.10	0.01	0.00	0.03	0.12	0.12	0.12	0.01	0.01	0.11
	Mean	0.84	0.85	0.73	0.73	0.79	0.78	0.71	0.73	0.72	0.76	0.68
	SD	0.11	0.11	0.20	0.20	0.14	0.23	0.24	0.17	0.20	0.18	0.21

These results also allow a deeper understanding of the behaviour of our proposed uncertainty combination method from the previous Section 4.2. On the Bacteria dataset, the OOD combinations where our method achieved significantly better AUROC than total uncertainty also had a significantly higher accuracy for the same rejection rate, namely Daptomycin, Caspofungin and Vancomycin. On the other hand, the AR curves between our combination approach and total uncertainty for the other OOD combinations were similar due to the fact that total uncertainty also obtained good AUROC for these combinations.

The conclusions drawn for the AUPR-In and AUPR-Out (see Appendix B.2) are analogous to the AUROC analysis, since we used 50% of in- and out-distributions inputs.

A more qualitative interpretation of the AUROC is presented in Table 3, where the results represent the number of occurrences in each AUROC interval over all datasets. From this table, we can easily conclude that KUE method was at least more robust to changes in OOD combinations/datasets than compared to state-of-the-art methods. Unlike the other methods, our method did not obtain any OOD worse than random. We can also see that OCSVM more occurrences of an excellent qualitative evaluation, but also one of which that had more fail and random classifications.

Table 3. Qualitative AUROC evaluation over all OOD combinations. *Excellent*: 90–100%, *Good*: 80–90%, *Fair*: 70–80%, *Poor*: 60–70%, *Fail*: 50–60%, ↓ *Random*: < 50%

	KUE_{KDE}	KUE_G	$p(\hat{\omega}\|x)$	$H[p(\omega\|x)]$	$I[\omega, h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Excellent	9	10	5	4	5	14	4	3	5	11	5
Good	11	12	8	10	12	2	12	11	11	2	7
Fair	5	4	6	5	7	3	3	7	2	5	2
Poor	3	2	4	4	2	2	0	2	2	5	7
Fail	1	1	2	2	1	4	5	3	4	2	2
↓ Random	0	0	4	4	2	4	5	3	5	4	6

Since our proposed approach for OOD detection is based on a density estimation techniques, and density estimation typically requires a large sample size, we performed an ablation study to evaluate how the AUROC results change with the number of train samples used for modeling. In Figure 6, we present the results of the ablation study for the four datasets used, where we rejected gradually 5% of the original number of train samples in each iteration, making a total of 20 iterations for each OOD combination. We can see that the AUROC values did not change significantly with the number of train samples. This means that the number of training samples caused small changes in feature modeling, resulting in minor variations on the performance of our method.

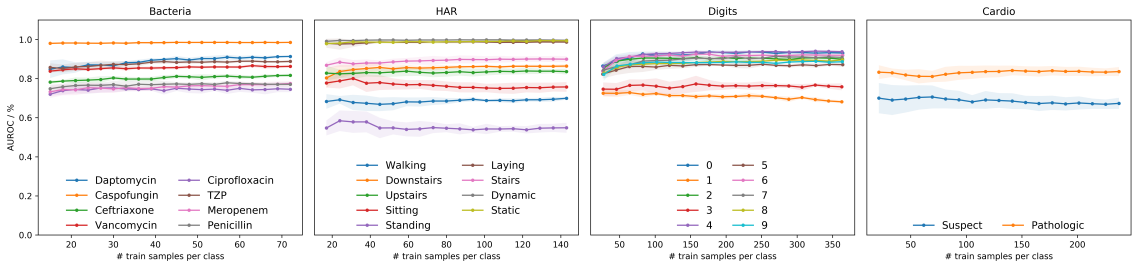

Figure 6. Ablation study of the KUE method, with KDE, for the four datasets. The legend represents each OOD inputs combination, and the title of each plot represents the dataset used.

5. Discussion and Conclusions

The importance of uncertainty quantification in ML has recently gained attention in the research community. However, its proper estimation is still an open research area. In a standard Bayesian setting, uncertainty is reflected by the predicted posterior probability, which is more akin to the aleatoric part of the overall uncertainty. On the other hand, epistemic uncertainty is commonly associated with OOD

detection problems, despite its quantification not being explicitly performed. Although OSR settings are a more realistic scenario for the deployment of ML models, they are mainly focused on effectively rejecting unknown inputs.

With this in mind, we proposed a new method for knowledge uncertainty estimation, KUE, and combined it with the classical information-theoretical measures of entropy proposed in the context of neural networks for distinguishing aleatoric, epistemic and total uncertainty by means of ensemble techniques.

Our proposed KUE method is based on a feature level density estimation of in-distribution train data, and it does not rely on out-distribution inputs for hyperparameters tuning nor for threshold selection. Since different classifiers have different accuracies for the classification of the very same data, we proposed a method that, although dependent on the classification accuracy, can be easily applied to any feature level model without changing the underlying classification methodology. As the nature of the data is often difficult to determine, we proposed a KDE method for feature density estimation. However, due to the computational cost of KDE with the increase of training size, we also compared the proposed method using a Gaussian distribution. For the four different datasets used for evaluation, Gaussian estimation showed similar results with KDE, which can significantly reduce the computational cost on large datasets. Nevertheless, if possible, the train data distribution can be calculated to choose the best kernel to be applied. Regarding the AUROC, our method KUE showed competitive performance results comparable to state-of-the-art methods. Furthermore, we also defined a threshold for OOD input rejection that is chosen based on the percentage of in-distribution test samples that we are willing to reject. We showed its dependency on FPR and also demonstrated that misclassified inputs tend to have high uncertainty values. Although the proposed threshold selection strategy effectively controlled the FPR, the TPR had a high variability between different datasets, and it was not possible to estimate its behavior for unknown inputs. For future research, this limitation should be addressed by combining KUE with different methods adopting a hybrid generative discriminative model perspective.

The aleatoric, epistemic and total uncertainty produced by measures of entropy showed a monotone dependency between reject rate and classification accuracy, which confirmed that these measures of uncertainty are a reliable indicator of the uncertainty involved in a classification decision. Moreover, the proposed uncertainty measures combination between our proposed KUE method and total uncertainty outperformed the individual entropy measures of uncertainty for the classification with a rejection option.

Future research includes the study of different combination strategies of uncertainty measures for classification with a rejection option. Leveraging the uncertainty for the interpretability of the rejected inputs is another interesting research direction. In addition, expanding the testing scenarios with more datasets should provide more indications about the robustness of the measures used. If more specialized OOD detection methods are able to properly quantify their own uncertainty, different combinations between existing methods and other sources of uncertainty should also be explored.

Author Contributions: Conceptualization and methodology, M.B.; software, C.P.; investigation and validation, C.P., M.B. and L.F.; writing—original draft preparation, M.B. and C.P.; writing—review and editing, C.P., M.B., L.F., D.F., H.G.; supervision, M.B., L.F. and H.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was co-funded by Portugal 2020, framed under the COMPETE 2020 (Operational Programme Competitiveness and Internationalization) and European Regional Development Fund (ERDF) from European Union (EU) through the project Geolocation non-Assisted by GPS for Mobile Networks in Indoor and Outdoor Environment (GARMIO), with operation code POCI-01-0247-FEDER-033479.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
AR	Accuracy Rejection
AUPR	Area under the PR
AUROC	Area under the ROC
FPR	False Positives Rate
IF	Isolation Forest
KDE	Kernel Density Estimation
KNN	K-Nearest Neighbor
KUE	Knowledge Uncertainty Estimation
LR	Logistic Regression
ML	Machine Learning
NB	Naive Bayes
NCM	Nearest Class Mean
NNDR	Nearest Neighbor Distance Ratio
OCC	One Class Classification
OCSVM	One Class Support Vector Machine
OOD	Out-of-distribution
OSNN	Open Set Nearest Neighbors
OSR	Open Set Recognition
PR	Precision-Recall
RF	Random Forest
ROC	Receiver Operating Characteristic
SVM	Support Vector Machine
TPR	True Positives Rate

Appendix A. Supplementary Details for Bacteria Dataset Experiments

Bacteria dataset from the work of Ho et al. [40] is publicly available at https://github.com/csho33/bacteria-ID. The detailed combinations used for OOD combinations can be seen in Table A1, where each combination was selected using antibiotic treatment for a specific set of bacteria.

Table A1. Bacterial classes used as OOD for each antibiotic.

Antibiotic	**Bacteria**
Daptomycin	*Enterococcus faecium*
Caspofungin	*Candida albicans* *Candida glabrata*
Ceftriaxone	*Streptococcus pneumoniae* 1 *Streptococcus pneumoniae* 2
Vancomycin	Methicillin-sensitive *Staphylococcus aureus* 1 Methicillin-sensitive *Staphylococcus aureus* 2 Methicillin-sensitive *Staphylococcus aureus* 3 Methicillin-resistant *Staphylococcus aureus* 1 Methicillin-resistant *Staphylococcus aureus* 2 *Staphylococcus epidermidis* *Staphylococcus lugdunensis*
Ciprofloxacin	*Salmonella enterica*
TZP	*Pseudomonas aeruginosa* 1 *Pseudomonas aeruginosa* 2

Table A1. *Cont.*

Antibiotic	Bacteria
Meropenem	*Klebsiella aerogenes* *Escherichia coli* 1 *Escherichia coli* 2 *Enterobacter cloacae* *Klebsiella pneumoniae* 1 *Klebsiella pneumoniae* 2 *Proteus mirabilis* *Serratia marcescens*
Penicillin	*Enterococcus faecalis* 1 *Enterococcus faecalis* 2 *Streptococcus sanguinis* Group A *Streptococcus* Group B *Streptococcus* Group C *Streptococcus* Group G *Streptococcus*

Appendix B. Supplementary Experimental Results

Appendix B.1. Proposed Method Using Different Classifiers

The experimental evaluation of our method for OOD detection was performed using a RF to provide a fair comparison between methods that required the use of ensemble techniques. Nevertheless, we provide detailed results using a set of classical algorithms, namely KNN, NB, SVM and LR, for the uncertainty quantification of our proposed method. In Table A2 we report both AUROC and the respective accuracy of each method on different OOD combinations.

Regarding the results from different classifiers, we notice that AUROC are very similar between the different algorithms. However, algorithms with higher accuracy tend to have also higher AUROC, which makes sense due to the dependency of classification accuracy of our proposed method.

Table A2. AUROC for detecting OOD inputs using our KUE method with KDE applied to 4 different classifiers and the correspondent accuracy (ACC) in % of the classifiers on 4 datasets. All values are averages over 10 consecutive repetitions.

		KUE_{KDE}								
		KNN		NB		SVM		LR		Mean
	OOD	AUROC	ACC	AUROC	ACC	AUROC	ACC	AUROC	ACC	AUROC
Bacteria	Daptomycin	0.92	79.0	0.89	71.5	0.92	84.1	0.91	83.7	0.91 ± 0.01
	Caspofungin	0.98	76.3	0.99	71.2	0.98	84.3	0.97	83.3	0.98 ± 0.01
	Ceftriaxone	0.80	81.7	0.78	73.5	0.79	86.8	0.83	86.7	0.80 ± 0.02
	Vancomycin	0.83	79.0	0.84	74.1	0.83	81.5	0.86	84.0	0.84 ± 0.01
	Ciprofloxacin	0.73	80.4	0.71	72.4	0.72	82.7	0.71	83.7	0.72 ± 0.01
	TZP	0.88	77.9	0.88	71.2	0.88	83.7	0.88	83.7	0.88 ± 0.00
	Meropenem	0.75	85.7	0.76	78.9	0.75	88.3	0.76	88.9	0.75 ± 0.01
	Penicillin	0.76	79.4	0.76	70.9	0.77	83.2	0.77	84.9	0.76 ± 0.01
HAR	Walking	0.69	86.2	0.67	82.1	0.72	89.6	0.71	86.6	0.67 ± 0.02
	Upstairs	0.85	86.9	0.84	85.6	0.87	88.6	0.86	87.1	0.85 ± 0.01
	Downstairs	0.83	84.8	0.82	82.9	0.83	87.9	0.83	85.8	0.83 ± 0.01
	Sitting	0.75	87.3	0.77	88.6	0.76	88.8	0.74	86.6	0.76 ± 0.01
	Standing	0.55	86.6	0.54	89.2	0.54	89.9	0.56	86.0	0.55 ± 0.01
	Laying	0.99	78.0	1.00	76.7	0.99	81.4	0.99	80.1	0.99 ± 0.01
	Stairs	0.89	88.9	0.90	86.1	0.90	91.5	0.90	88.4	0.90 ± 0.00
	Dynamic	1.00	84.5	1.00	81.1	1.00	87.9	0.99	88.4	1.00 ± 0.00
	Static	0.99	79.3	1.00	80.2	0.99	82.0	0.99	83.5	0.99 ± 0.01

Table A2. *Cont.*

		KUE_{KDE}								
		KNN		NB		SVM		LR		Mean
	OOD	AUROC	ACC	AUROC	ACC	AUROC	ACC	AUROC	ACC	AUROC
Digits	0	0.95	97.4	0.93	78.4	0.96	95.8	0.95	94.2	0.95 ± 0.01
	1	0.66	98.3	0.63	82.1	0.65	96.3	0.65	95.2	0.65 ± 0.01
	2	0.90	97.6	0.83	79.7	0.90	96.4	0.91	94.1	0.88 ± 0.03
	3	0.76	98.1	0.74	80.1	0.76	96.9	0.77	94.5	0.76 ± 0.01
	4	0.95	98.0	0.95	81.7	0.94	95.6	0.94	95.4	0.95 ± 0.01
	5	0.86	97.5	0.81	80.0	0.87	96.3	0.89	95.1	0.86 ± 0.03
	6	0.94	98.0	0.89	77.7	0.92	95.8	0.92	94.9	0.91 ± 0.02
	7	0.92	97.8	0.87	78.2	0.92	96.7	0.92	94.9	0.91 ± 0.02
	8	0.90	98.5	0.87	83.7	0.90	97.3	0.90	95.5	0.89 ± 0.01
	9	0.88	98.3	0.84	82.7	0.88	96.9	0.86	94.9	0.87 ± 0.02
Cardio	Suspect	0.65	80.1	0.64	60.8	0.67	87.1	0.67	84.5	0.66 ± 0.01
	Pathologic	0.83	79.3	0.80	62.1	0.83	87.0	0.82	94.9	0.82 ± 0.01

Appendix B.2. AUPR-In and AUPR-Out Results

In Tables A3 and A4 we present the detailed results for AUPR-Out and AUPR-In, where in-distribution and out-distribution inputs are specified as negatives and positives, respectively.

The conclusions drawn for the AUPR are analogous to the AUROC analysis, since we used 50% of in- and out-distribution inputs.

Table A3. AUPR-Out for detecting OOD test inputs using two variants of KUE (KDE and Gaussian) and other baseline methods on 4 datasets. The Mean and Standard Deviation (SD) over OOD combinations is presented after each dataset. All values are averages over 10 consecutive repetitions.

		AUPR-Out										
	OOD	KUE_{KDE}	KUE_{G}	$p(\hat{\omega}\|x)$	$H[p(\omega\|x)]$	$I[\omega,h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Bacteria	Daptomycin	0.86	0.86	0.61	0.66	0.87	0.52	0.87	0.75	0.57	0.59	0.59
	Caspofungin	0.92	0.97	0.40	0.38	0.89	0.98	0.54	0.65	0.40	0.46	0.94
	Ceftriaxone	0.76	0.75	0.76	0.83	0.82	0.53	0.90	0.81	0.83	0.73	0.40
	Vancomycin	0.87	0.87	0.66	0.66	0.82	0.79	0.81	0.72	0.69	0.61	0.81
	Ciprofloxacin	0.66	0.66	0.82	0.89	0.66	0.40	0.85	0.76	0.79	0.71	0.35
	TZP	0.81	0.86	0.68	0.71	0.90	0.73	0.89	0.80	0.85	0.77	0.53
	Meropenem	0.74	0.74	0.81	0.84	0.75	0.48	0.85	0.81	0.80	0.73	0.44
	Penicillin	0.63	0.76	0.70	0.72	0.65	0.67	0.80	0.81	0.70	0.66	0.71
	Mean	0.78	0.81	0.68	0.71	0.80	0.64	0.81	0.76	0.70	0.66	0.60
	SD	0.10	0.09	0.13	0.15	0.09	0.18	0.11	0.05	0.14	0.09	0.20
HAR	Walking	0.59	0.59	0.72	0.71	0.71	0.35	0.69	0.68	0.65	0.68	0.35
	Upstairs	0.80	0.81	0.74	0.79	0.81	0.43	0.64	0.65	0.74	0.66	0.44
	Downstairs	0.78	0.76	0.65	0.63	0.64	0.84	0.54	0.70	0.44	0.65	0.88
	Sitting	0.75	0.72	0.49	0.49	0.70	0.66	0.46	0.43	0.50	0.45	0.64
	Standing	0.50	0.47	0.52	0.58	0.76	0.51	0.47	0.66	0.53	0.45	0.52
	Laying	0.85	0.93	0.36	0.38	0.47	1.00	0.32	0.35	0.71	0.49	1.00
	Stairs	0.83	0.83	0.48	0.52	0.67	0.74	0.33	0.42	0.49	0.42	0.79
	Dynamic	0.02	0.23	0.64	0.69	0.70	0.78	0.52	0.52	0.76	0.88	0.85
	Static	0.65	0.69	0.66	0.66	0.74	0.98	0.37	0.58	0.44	0.78	0.99
	Mean	0.64	0.67	0.58	0.61	0.69	0.70	0.48	0.55	0.58	0.61	0.72
	SD	0.25	0.20	0.12	0.12	0.09	0.22	0.12	0.12	0.12	0.15	0.23

Table A3. *Cont.*

		AUPR-Out										
	OOD	KUE_{KDE}	KUE_{G}	$p(\hat{\omega}\|x)$	$H[p(\omega\|x)]$	$I[\omega, h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Digits	0	0.84	0.89	0.82	0.82	0.96	1.00	0.93	0.46	0.87	0.96	0.72
	1	0.63	0.72	0.86	0.84	0.79	0.94	0.78	0.62	0.81	0.89	0.60
	2	0.70	0.78	0.86	0.82	0.83	0.99	0.88	0.52	0.87	0.94	0.82
	3	0.68	0.74	0.86	0.80	0.75	0.95	0.81	0.61	0.83	0.93	0.60
	4	0.73	0.85	0.86	0.87	0.96	0.99	0.84	0.61	0.82	0.91	0.91
	5	0.73	0.82	0.90	0.88	0.89	0.98	0.81	0.59	0.85	0.94	0.65
	6	0.80	0.86	0.84	0.84	0.96	0.99	0.94	0.52	0.91	0.95	0.75
	7	0.78	0.86	0.92	0.93	0.86	0.99	0.89	0.49	0.91	0.95	0.84
	8	0.79	0.78	0.96	0.98	0.93	0.97	0.95	0.81	0.90	0.95	0.44
	9	0.80	0.85	0.84	0.82	0.80	0.91	0.87	0.73	0.82	0.95	0.57
	Mean	0.75	0.82	0.87	0.86	0.87	0.97	0.87	0.60	0.86	0.94	0.69
	SD	0.06	0.05	0.04	0.05	0.07	0.03	0.06	0.10	0.04	0.02	0.14
Cardio	Suspect	0.60	0.59	0.42	0.42	0.52	0.67	0.47	0.48	0.41	0.59	0.62
	Pathologic	0.79	0.80	0.42	0.42	0.53	0.98	0.36	0.72	0.41	0.61	0.93
	Mean	0.70	0.70	0.42	0.42	0.52	0.82	0.42	0.60	0.41	0.60	0.78
	SD	0.10	0.11	0.00	0.00	0.01	0.15	0.05	0.12	0.00	0.01	0.16
	Mean	0.72	0.76	0.70	0.71	0.77	0.78	0.70	0.63	0.70	0.73	0.68
	SD	0.16	0.15	0.17	0.17	0.13	0.21	0.21	0.13	0.17	0.18	0.19

Table A4. AUPR-In for detecting OOD test inputs using two variants of KUE (KDE and Gaussian) and other baseline methods on 4 datasets. The Mean and Standard Deviation (SD) over OOD combinations is presented after each dataset. All values are averages over 10 consecutive repetitions.

		AUPR-In										
	OOD	KUE_{KDE}	KUE_{G}	$p(\hat{\omega}\|x)$	$H[p(\omega\|x)]$	$I[\omega,h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Bacteria	Daptomycin	0.93	0.93	0.71	0.71	0.88	0.64	0.91	0.82	0.60	0.63	0.72
	Caspofungin	0.99	0.99	0.52	0.48	0.82	0.98	0.53	0.61	0.47	0.55	0.93
	Ceftriaxone	0.85	0.85	0.86	0.87	0.86	0.51	0.93	0.85	0.83	0.82	0.43
	Vancomycin	0.87	0.87	0.65	0.63	0.73	0.65	0.82	0.76	0.58	0.62	0.66
	Ciprofloxacin	0.79	0.80	0.89	0.92	0.73	0.43	0.90	0.83	0.83	0.80	0.36
	TZP	0.89	0.89	0.82	0.81	0.89	0.58	0.90	0.80	0.85	0.84	0.50
	Meropenem	0.79	0.79	0.90	0.90	0.82	0.51	0.89	0.86	0.86	0.83	0.47
	Penicillin	0.78	0.78	0.72	0.73	0.69	0.53	0.80	0.85	0.64	0.73	0.59
	Mean	0.86	0.86	0.76	0.76	0.80	0.60	0.84	0.80	0.71	0.73	0.58
	SD	0.07	0.07	0.12	0.14	0.07	0.16	0.12	0.08	0.14	0.11	0.17
HAR	Walking	0.74	0.72	0.81	0.82	0.84	0.36	0.77	0.73	0.78	0.77	0.39
	Upstairs	0.88	0.88	0.83	0.85	0.88	0.45	0.76	0.71	0.83	0.76	0.49
	Downstairs	0.87	0.87	0.76	0.75	0.77	0.90	0.59	0.70	0.57	0.74	0.90
	Sitting	0.73	0.75	0.59	0.60	0.68	0.66	0.64	0.54	0.60	0.55	0.63
	Standing	0.55	0.56	0.70	0.72	0.82	0.68	0.68	0.73	0.62	0.50	0.68
	Laying	0.99	1.00	0.36	0.36	0.42	1.00	0.37	0.35	0.86	0.59	1.00
	Stairs	0.92	0.91	0.63	0.65	0.73	0.78	0.50	0.48	0.57	0.51	0.80
	Dynamic	1.00	1.00	0.80	0.81	0.77	0.85	0.67	0.68	0.91	0.94	0.87
	Static	1.00	0.99	0.74	0.74	0.76	0.99	0.44	0.55	0.59	0.83	1.00
	Mean	0.85	0.85	0.69	0.70	0.74	0.74	0.60	0.61	0.70	0.69	0.75
	SD	0.14	0.14	0.14	0.14	0.13	0.21	0.13	0.13	0.13	0.15	0.21

Table A4. *Cont.*

		AUPR-In										
	OOD	KUE_{KDE}	KUE_{G}	$p(\hat{\omega}\|x)$	$H[p(\omega\|x)]$	$I[\omega, h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Digits	0	0.94	0.96	0.93	0.94	0.98	1.00	0.96	0.95	0.93	0.98	0.84
	1	0.69	0.85	0.91	0.91	0.86	0.96	0.80	0.92	0.88	0.92	0.66
	2	0.90	0.93	0.92	0.92	0.90	1.00	0.91	0.95	0.92	0.96	0.84
	3	0.79	0.86	0.93	0.91	0.87	0.97	0.88	0.90	0.89	0.97	0.66
	4	0.95	0.96	0.90	0.90	0.96	0.99	0.87	0.96	0.86	0.94	0.95
	5	0.88	0.91	0.91	0.90	0.90	0.98	0.86	0.92	0.90	0.97	0.66
	6	0.93	0.94	0.90	0.90	0.98	0.99	0.96	0.92	0.95	0.97	0.83
	7	0.93	0.94	0.96	0.96	0.93	0.99	0.90	0.93	0.95	0.97	0.88
	8	0.92	0.88	0.97	0.98	0.96	0.98	0.96	0.97	0.93	0.97	0.48
	9	0.90	0.92	0.92	0.90	0.87	0.96	0.92	0.92	0.90	0.97	0.64
	Mean	0.88	0.92	0.92	0.92	0.92	0.98	0.90	0.93	0.91	0.96	0.74
	SD	0.08	0.04	0.02	0.03	0.04	0.01	0.05	0.02	0.03	0.02	0.14
Cardio	Suspect	0.71	0.67	0.39	0.38	0.44	0.80	0.59	0.60	0.39	0.71	0.75
	Pathologic	0.86	0.88	0.40	0.39	0.50	0.98	0.41	0.81	0.41	0.71	0.95
	Mean	0.78	0.78	0.40	0.38	0.47	0.89	0.50	0.70	0.40	0.71	0.85
	SD	0.08	0.10	0.01	0.01	0.03	0.09	0.09	0.11	0.01	0.00	0.10
	Mean	0.86	0.87	0.77	0.77	0.80	0.80	0.76	0.78	0.76	0.79	0.71
	SD	0.10	0.10	0.17	0.18	0.14	0.21	0.18	0.16	0.17	0.16	0.19

Appendix B.3. Accuracy Rejection Curves

In this section we present the AR curves for HAR (Figure A1), Digits (Figure A2) and Cardio (Figure A3) datasets. In the three figures, the average rejection rates against the average accuracy for our method, total, aleatoric and epistemic uncertainty are presented. The proposed combination is also shown in black, and the optimal rejection is represented by the dashed line.

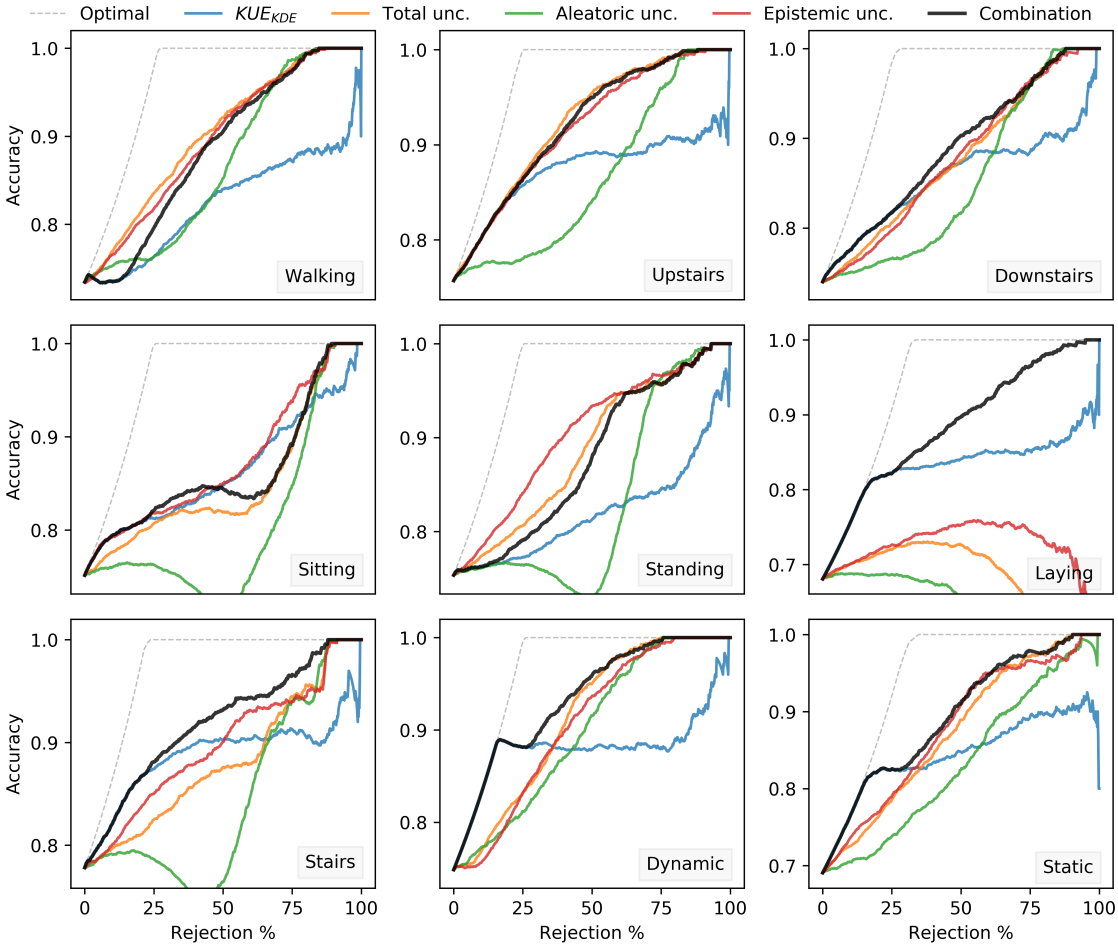

Figure A1. AR curves for aleatoric, epistemic and total uncertainty for HAR dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the activity used for each OOD input combination.

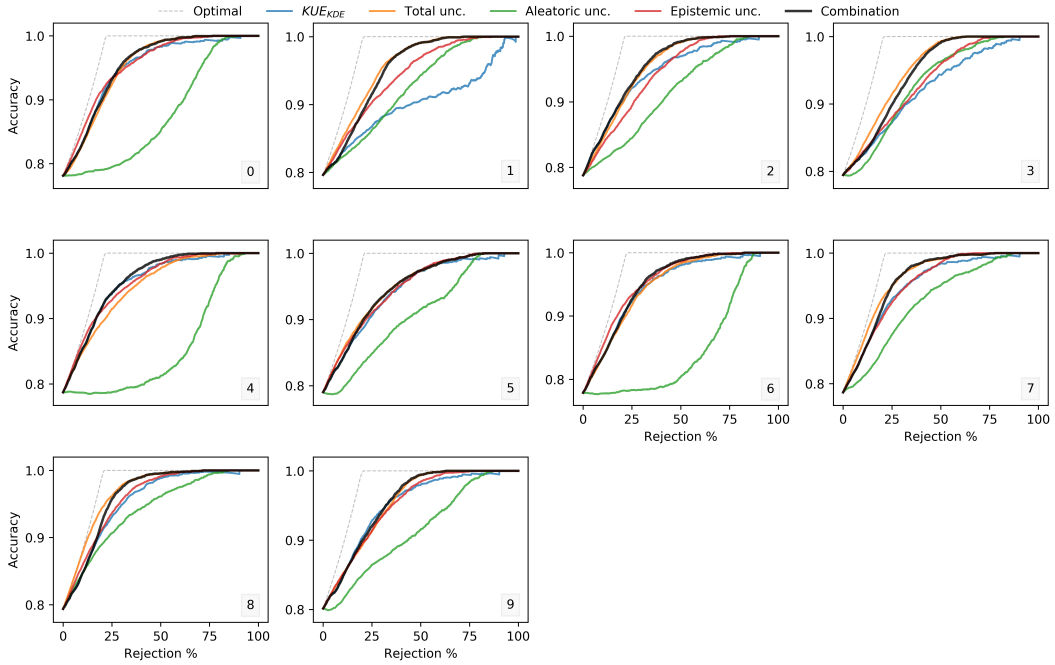

Figure A2. AR curves for aleatoric, epistemic and total uncertainty for the Digits dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the digits used for each OOD input combination.

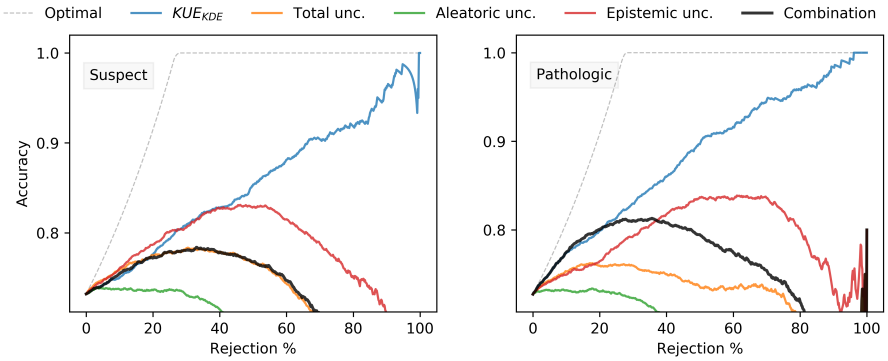

Figure A3. AR curves for aleatoric, epistemic and total uncertainty for the Cardio dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the OOD input combination.

References

1. Begoli, E.; Tanmoy, B.; Dimitri, K. The need for uncertainty quantification in machine-assisted medical decision making. *Nat. Mach. Intell.* **2019**, *1*, 20–23. [CrossRef]
2. Hüllermeier, E.; Waegeman, W. Aleatoric and epistemic uncertainty in machine learning: A tutorial introduction. *arXiv* **2019**, arXiv:1910.09457.
3. Schulam, P.; Saria, S. Can you trust this prediction? Auditing pointwise reliability after learning. *arXiv* **2019**, arXiv:1901.00403.
4. Holzinger, A.; Langs, G.; Denk, H.; Zatloukal, K.; Müller, H. Causability and explainability of artificial intelligence in medicine. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2019**, *9*, e1312. [CrossRef]
5. Campagner, A.; Cabitza, F.; Ciucci, D. Three-Way Decision for Handling Uncertainty in Machine Learning: A Narrative Review. In *International Joint Conference on Rough Sets*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 137–152.
6. Gal, Y. *Uncertainty in Deep Learning*; University of Cambridge: Cambridge, UK, 2016; Volume 1.
7. Bota, P.; Silva, J.; Folgado, D.; Gamboa, H. A semi-automatic annotation approach for human activity recognition. *Sensors* **2019**, *19*, 501. [CrossRef]

8. Mukherjee, S.; Awadallah, A.H. Uncertainty-aware Self-training for Text Classification with Few Labels. *arXiv* **2020**, arXiv:2006.15315.
9. Santos, R.; Barandas, M.; Leonardo, R.; Gamboa, H. Fingerprints and floor plans construction for indoor localisation based on crowdsourcing. *Sensors* **2019**, *19*, 919. [CrossRef]
10. Li, Y.; Chen, J.; Feng, L. Dealing with uncertainty: A survey of theories and practices. *IEEE Trans. Knowl. Data Eng.* **2012**, *25*, 2463–2482. [CrossRef]
11. Varshney, K.R.; Alemzadeh, H. On the safety of machine learning: Cyber-physical systems, decision sciences, and data products. *Big Data* **2017**, *5*, 246–255. [CrossRef]
12. Senge, R.; Bösner, S.; Dembczyński, K.; Haasenritter, J.; Hirsch, O.; Donner-Banzhoff, N.; Hüllermeier, E. Reliable classification: Learning classifiers that distinguish aleatoric and epistemic uncertainty. *Inf. Sci.* **2014**, *255*, 16–29. [CrossRef]
13. Scheirer, W.J.; de Rezende Rocha, A.; Sapkota, A.; Boult, T.E. Toward open set recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 1757–1772. [CrossRef] [PubMed]
14. Boult, T.E.; Cruz, S.; Dhamija, A.R.; Gunther, M.; Henrydoss, J.; Scheirer, W.J. Learning and the unknown: Surveying steps toward open world recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 33, pp. 9801–9807.
15. Ren, J.; Liu, P.J.; Fertig, E.; Snoek, J.; Poplin, R.; Depristo, M.; Dillon, J.; Lakshminarayanan, B. Likelihood ratios for out-of-distribution detection. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2019; pp. 14707–14718.
16. Scheirer, W.J.; Jain, L.P.; Boult, T.E. Probability models for open set recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 2317–2324. [CrossRef] [PubMed]
17. Chow, C. On optimum recognition error and reject tradeoff. *IEEE Trans. Inf. Theory* **1970**, *16*, 41–46. [CrossRef]
18. Zhu, L.; Laptev, N. Deep and confident prediction for time series at uber. In Proceedings of the 2017 IEEE International Conference on Data Mining Workshops (ICDMW), New Orleans, LA, USA, 18–21 November 2017; pp. 103–110.
19. Malinin, A.; Gales, M. Predictive uncertainty estimation via prior networks. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2018; pp. 7047–7058.
20. Tax, D.M.; Duin, R.P. Growing a multi-class classifier with a reject option. *Pattern Recognit. Lett.* **2008**, *29*, 1565–1570. [CrossRef]
21. Fumera, G.; Roli, F.; Giacinto, G. Reject option with multiple thresholds. *Pattern Recognit.* **2000**, *33*, 2099–2101. [CrossRef]
22. Dubois, D.; Prade, H.; Smets, P. Representing partial ignorance. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **1996**, *26*, 361–377. [CrossRef]
23. Perello-Nieto, M.; Telmo De Menezes Filho, E.S.; Kull, M.; Flach, P. Background Check: A general technique to build more reliable and versatile classifiers. In Proceedings of the 2016 IEEE 16th International Conference on Data Mining (ICDM), Barcelona, Spain, 12–15 December 2016; pp. 1143–1148.
24. Roady, R.; Hayes, T.L.; Kemker, R.; Gonzales, A.; Kanan, C. Are Out-of-Distribution Detection Methods Effective on Large-Scale Datasets? *arXiv* **2019**, arXiv:1910.14034.
25. Nguyen, A.; Yosinski, J.; Clune, J. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 427–436.
26. Heaven, D. Why deep-learning AIs are so easy to fool. *Nature* **2019**, *574*, 163–166. [CrossRef]
27. Geng, C.; Sheng-jun, H.; Songcan, C. Recent advances in open set recognition: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**. [CrossRef]
28. Gune, O.; More, A.; Banerjee, B.; Chaudhuri, S. Generalized Zero-shot Learning using Open Set Recognition. In Proceedings of the 30th British Machine Vision Conference, Cardiff, UK, 9–12 September 2019; p. 213.
29. Noumir, Z.; Honeine, P.; Richard, C. On simple one-class classification methods. In Proceedings of the 2012 IEEE International Symposium on Information Theory Proceedings, Cambridge, MA, USA, 1–6 July 2012; pp. 2022–2026.
30. Khan, S.S.; Madden, M.G. One-class classification: Taxonomy of study and review of techniques. *Knowl. Eng. Rev.* **2014**, *29*, 345–374. [CrossRef]
31. Rocha, A.; Siome Klein, G. Multiclass from binary: Expanding one-versus-all, one-versus-one and ecoc-based approaches. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *25*, 289–302. [CrossRef] [PubMed]

32. Júnior, P.R.M.; Boult, T.E.; Wainer, J.; Rocha, A. Specialized support vector machines for open-set recognition. *arXiv* **2020**, arXiv:1606.03802v10.
33. Bendale, A.; Boult, T. Towards open world recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1893–1902.
34. Mensink, T.; Verbeek, J.; Perronnin, F.; Csurka, G. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2624–2637. [CrossRef] [PubMed]
35. Júnior, P.R.M.; De Souza, R.M.; Werneck, R.d.O.; Stein, B.V.; Pazinato, D.V.; de Almeida, W.R.; Penatti, O.A.; Torres, R.d.S.; Rocha, A. Nearest neighbors distance ratio open-set classifier. *Mach. Learn.* **2017**, *106*, 359–386. [CrossRef]
36. Sensoy, M.; Kaplan, L.; Kandemir, M. Evidential deep learning to quantify classification uncertainty. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2018; pp. 3179–3189.
37. Shaker, M.H.; Hüllermeier, E. Aleatoric and Epistemic Uncertainty with Random Forests. In *International Symposium on Intelligent Data Analysis*; Springer: Berlin /Heidelberg, Germany, 2020; pp. 444–456.
38. Depeweg, S.; Jose-Miguel, H.L.; Finale, D.V.; Steffen, U. Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 1184–1193.
39. Shafaei, A.; Schmidt, M.; Little, J.J. A less biased evaluation of out-of-distribution sample detectors. *arXiv* **2018**, arXiv:1809.04729.
40. Ho, C.S.; Neal, J.; Catherine A.H.; Lena, B.; Stefanie S.J.; Mark, H.; Niaz, B.; Amr AE, S.; Stefano, E.; Jennifer, D. Rapid identification of pathogenic bacteria using Raman spectroscopy and deep learning. *Nat. Commun.* **2019**, *10*, 4927. [CrossRef] [PubMed]
41. Dua, D.; Graff, C. *UCI Machine Learning Repository*; University of California: Irvine, CA, USA, 2019.
42. Nadeem, M.S.A.; Zucker, J.D.; Hanczar, B. Accuracy-rejection curves (ARCs) for comparing classification methods with a reject option. *Mach. Learni. Syst. Biol.* **2009**, *8*, 65–81.
43. Scott, D.W. *Multivariate Density Estimation: Theory, Practice, and Visualization*; John Wiley & Sons: Hoboken, NY, USA, 1992.
44. Abbas, M.R.; Nadeem, M.S.A.; Shaheen, A.; Alshdadi, A.A.; Alharbey, R.; Shim, S.O.; Aziz, W. Accuracy Rejection Normalized-Cost Curves (ARNCCs): A Novel 3-Dimensional Framework for Robust Classification. *IEEE Access* **2019**, *7*, 160125–160143. [CrossRef]
45. Davis, J.; Goadrich, M. The relationship between Precision-Recall and ROC curves. In Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, USA, 25–29 June 2006; pp. 233–240.
46. Fawcett, T.; Flach, P.A. A response to Webb and Ting's on the application of ROC analysis to predict classification performance under varying class distributions. *Mach. Learn.* **2005**, *58*, 33–38. [CrossRef]
47. Hendrycks, D.; Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv* **2016**, arXiv:1610.02136.
48. Manning, C.; Schutze, H. *Foundations of Statistical Natural Language Processing*; MIT Press: Cambridge, MA, USA, 1999.
49. Saito, T.; Rehmsmeier, M. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE* **2015**, *10*, e0118432. [CrossRef]
50. Schölkopf, B.; John, C.P.; John, S.T.; Alex, J.S.; Robert, C.W. Estimating the support of a high-dimensional distribution. *Neural Comput.* **2001**, *13*, 1443–1471. [CrossRef]
51. Wu, T.F.; Chih-Jen, L.; Ruby C.W. Probability estimates for multi-class classification by pairwise coupling. *J. Mach. Learn. Res.* **2004**, *5*, 975–1005.
52. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation-based anomaly detection. *ACM Trans. Knowl. Discov. Data* **2012**, *6*, 1–39. [CrossRef]