electronics

MDPI

Article

Uncertainty Quantification of Machine Learning Model Performance via Anomaly-Based Dataset Dissimilarity Measures †

Gabriele Incorvaia *, Darryl Hond * and Hamid Asgari *

Thales UK - Research, Technology & Innovation, Reading RG2 6GF, UK

* Correspondence: gabriele.incorvaia@uk.thalesgroup.com (G.I.); darryl.hond@uk.thalesgroup.com (D.H.); hamid.asgari@uk.thalesgroup.com (H.A.)

† This paper is an extended version of our conference proceedings paper: G. Incorvaia, D. Hond and H. Asgari, "Uncertainty quantification for machine learning output assurance using anomaly-based dataset dissimilarity measures", International Conference on Artificial Intelligence Testing, IEEE, Athens, Greece, 17–20 July 2023.

Abstract: The use of Machine Learning (ML) models as predictive tools has increased dramatically in recent years. However, data-driven systems (such as ML models) exhibit a degree of uncertainty in their predictions. In other words, they could produce unexpectedly erroneous predictions if the uncertainty stemming from the data, choice of model and model parameters is not taken into account. In this paper, we introduce a novel method for quantifying the uncertainty of the performance levels attained by ML classifiers. In particular, we investigate and characterize the uncertainty of model accuracy when classifying out-of-distribution data that are statistically dissimilar from the data employed during training. A main element of this novel Uncertainty Quantification (UQ) method is a measure of the dissimilarity between two datasets. We introduce an innovative family of data dissimilarity measures based on anomaly detection algorithms, namely the Anomaly-based Dataset Dissimilarity (ADD) measures. These dissimilarity measures process feature representations that are derived from the activation values of neural networks when supplied with dataset items. The proposed UQ method for classification performance employs these dissimilarity measures to estimate the classifier accuracy for unseen, out-of-distribution datasets, and to give an uncertainty band for those estimates. A numerical analysis of the efficacy of the UQ method is conducted using standard Artificial Neural Network (ANN) classifiers and public domain datasets. The results obtained generally demonstrate that the amplitude of the uncertainty band associated with the estimated accuracy values tends to increase as the data dissimilarity measure increases. Overall, this research contributes to the verification and run-time performance prediction of systems composed of ML-based elements.

Keywords: uncertainty quantification; artificial neural networks; machine learning; image classification; performance prediction; anomaly detection; data dissimilarity measures

Citation: Incorvaia, G.; Hond, D.; Asgari, H. Uncertainty Quantification of Machine Learning Model Performance via Anomaly-Based Dataset Dissimilarity Measures. *Electronics* **2024**, *13*, 939. https://doi.org/10.3390/electronics13050939

Academic Editors: Valentina E. Balas, Hong Zhu, Junhua Ding and Aktouf Oum-El-Kheir

Received: 29 January 2024
Revised: 21 February 2024
Accepted: 23 February 2024
Published: 29 February 2024

1. Introduction

Autonomous systems (AS) make use of a suite of algorithms in order to understand the environment in which they are deployed and make independent decisions. In recent years, the ability of AS to gain situational awareness has been boosted by significant improvements in the performance of Machine Learning (ML) algorithms. Artificial Neural Networks (ANNs) are a class of ML algorithms that are typically employed to solve one or more classic problems, such as classification or regression. The decision boundaries generated by such networks are highly non-linear, and the choice of data used to prepare and test a network can have a dramatic impact on performance. These factors could influence automated decisions in a way that might compromise the safety of people interacting with the system. Therefore, it is vital to establish that ANN algorithms operating within an AS

are trustworthy. There are a number of requirements which must be met by an ANN for it to be deemed trustworthy, and those requirements must be subject to verification. One such requirement is that the uncertainty of ANN outputs should be quantified. There is also a need to quantify the uncertainty of information generated when monitoring ANN operation at run-time, such as when making predictions of algorithm behavioral performance.

The popularity of Machine Learning (ML) models has been steadily increasing in recent years due to their effectiveness, flexibility, applicability, and the machine-speed acceleration they provide for decision-making processes [1,2]. However, according to [3], one of the challenges posed by these models is that their outputs will exhibit uncertainty. In other words, even a well-trained ML-based model can produce unexpectedly erroneous predictions if the uncertainty associated with both data and model parameters is not taken into account. Incorrect predictions might hinder users' confidence in a model, thus affecting its applicability in practice [4]. Due to the stochastic nature of environmental systems, uncertainty can be minimized but not completely eliminated in real-world applications [3]. The goal of Uncertainty Quantification (UQ) is to capture this uncertainty in terms of probability distributions [5,6].

The accuracy of ML models tends to fall when used on data that are statistically different from their training data [4,7]. The term in-distribution is used to describe data which are drawn from the training data-generating distribution (i.e., the probability distribution from which training samples are drawn); out-of-distribution data are not drawn from the training data-generating distribution. Mathematically, when an ML-based model is successfully trained on a given dataset, the model is expected to produce accurate predictions for unseen, in-distribution test data. Conversely, the accuracy of the model is expected to fall when processing data instances drawn from out-of-distribution test data. This is of practical importance since it will often be the case that when ML models are deployed within real-world AS, they will receive out-of-distribution data during operation.

In this paper, we assume that a shift between training and test data-generating distributions is a source of ML model uncertainty. Thus, our aim is to introduce a UQ method based on the idea of first quantifying the statistical divergence between a training and a test distribution (or dataset), and secondly using this information to estimate the expected level of performance (e.g., output accuracy) of an ML-based model. In addition, we estimate the uncertainty interval associated with this point estimation of performance.

Here, we propose and elaborate on the following novel methods and measures:

- An Uncertainty Quantification (UQ) method aimed at predicting the level at which an ML model will perform, and the uncertainty associated with such a performance prediction, using data dissimilarity measures;
- A novel family of data dissimilarity measures based on anomaly detection algorithms, which are computed from the features represented by ANN activation values.

An experimental analysis of the UQ method introduced above, and the related family of dissimilarity measures, is conducted using public domain datasets and established ANN architectures.

This paper is organized as follows. Section 2 introduces the materials and methods employed in this work. More specifically, Section 2.1 describes how UQ methods have been used within ML applications, and provides a brief review of prominent performance and data dissimilarity measures that have been proposed in the literature. Section 2.2 introduces a UQ method for predicting the level of performance and the corresponding uncertainty of an ML-based model using measures of data dissimilarity. Section 2.3 describes a novel family of data dissimilarity measures, the Anomaly-based Dataset Dissimilarity (ADD) measures, where anomaly scores are employed to quantify the degree of discrepancy between two datasets of interest. Section 2.4 describes the experimental setting employed to perform a numerical analysis of our proposed UQ method and ADD measures operating on public domain data. The results of this numerical analysis are provided in Section 3. Finally, some observations are made and conclusions drawn in Section 4.

For completeness, we conclude Section 1 by specifying that this work is an extension of our conference paper [8].

2. Materials and Methods

In this section, we provide information about the datasets, ML models and algorithms used in this work. In Section 2.1, we provide a review of the relevant literature, which includes Uncertainty Quantification (UQ) methods for ML models, measures of ML performance, and measures of dataset dissimilarity. In Section 2.2, we introduce a UQ method for ML performance prediction based on dataset dissimilarity measures. In Section 2.3, we describe a novel family of dataset dissimilarity measures based on anomaly detection algorithms, which we refer to as Anomaly-based Dataset Dissimilarity (ADD) measures. In Section 2.4, details about the datasets and ML models employed in this work are given.

2.1. Related Work

The interest of the ML community in UQ methods is confirmed by the fact that, according to [6], more than 2500 papers addressing the use of UQ in ML were produced between 2010 and 2021. The literature refers to a broad variety of applications including computer vision and image processing [9–13]; medical studies [14,15]; weather forecasting [3,16,17]; and natural language processing and text mining [18,19]. UQ methods have been developed for supervised learning [20], unsupervised learning [21,22] and reinforcement learning [23,24]. Prominent subjects in the UQ literature include: (i) Bayesian approaches, which aim to learn the relationship between ML model inputs and ML model outputs in terms of a conditional probability distribution, but require assumptions on the prior distribution [6,9,10]; and (ii) ML calibration, which aims to provide confidence scores for ML model predictions that correspond to the actual probability of correctness of these predictions, but often involves further, potentially computationally expensive, transformations of the ML model outputs [25,26]. In this paper, we propose a novel UQ method based on data dissimilarity measures.

2.1.1. Measures of ML Performance

ML algorithm performance measures have been extensively investigated in the literature, and their definition depends on the particular type of problem addressed. In other words, different measures might be needed for different tasks. For instance, examples of performance metrics for classification problems include accuracy, balanced accuracy, precision, recall, F1 score and confusion matrices [27,28]. The Mean Absolute Error (MAE), Mean Squared Error (MSE) and Root-Mean-Square Error (RMSE) [27,29] are examples of performance measures which are widely used for regression problems.

2.1.2. Measures of Dataset Dissimilarity

A brief overview of established data dissimilarity measures that have been proposed in the literature is provided below.

A number of statistical measures that were originally developed to quantify the similarity between probability distributions can be used to quantify the divergence between data-generating distributions. Examples of such statistical measures include the Hellinger distance, Bhattacharyya distance, Mahalanobis distance, and the Kullback–Leibler divergence [30,31]. However, they present challenges when characterizing distributions in high-dimensional spaces [30]. Another popular example of a statistical measure is the Maximum Mean Discrepancy (MMD) [32], which quantifies the distance between two distributions by representing them as elements of a Reproducing Kernel Hilbert Space.

A confidence score based on generative models that is interpretable as a measure of data dissimilarity has been introduced in [33]. The authors argue that the pre-trained features of a discriminative softmax classifier can be fitted well by a class-conditional generative Gaussian distribution, which can then be employed to calculate confidence

scores for test data instances. In addition, as shown in [33], these confidence scores can be used to detect abnormal test data instances, such as out-of-distribution ones.

A surprise-based data distance is proposed in [34] and used to develop an adequacy criterion for testing ML-based models. The general approach is to quantify how surprising a test input is in relation to a trained ML-based model. In this context, a greater degree of surprise is associated with a test data item that is more dissimilar from the training samples, and consequently more likely to produce prediction errors.

Finally, we note that further data dissimilarity measures computed on suitably selected feature spaces have been proposed, where the choice of these features depends on the particular application addressed [20,35,36]. For example, the fractional Neuron Region Distance (fNRD) proposed in [35] is computed in terms of activation values of an Artificial Neural Network (ANN) into which the data of interest have been fed as input. Additional dissimilarity distances based on features extracted from imagery and represented within Convolutional Neural Networks (CNNs) and Siamese Networks [37,38] are introduced in [39,40].

2.1.3. Relationship between Dataset Dissimilarity Measures and Anomaly Detection Algorithms

The task of computing data dissimilarity measures is related to that of recognizing sample outliers or anomalies. An anomaly can be defined as a data instance that does not conform to a well-defined notion of normal behavior [41] or, more intuitively, that deviates so much from other available observations as to arouse suspicion that it has been generated by a different mechanism [42]. Several anomaly scores which quantify the degree of outlierness of an instance with respect to a reference set have been proposed in the literature [43–46]. These scores can be employed to define novel data dissimilarity measures, and we will explore this possibility more extensively in Section 2.3.

Examples of algorithms which output anomaly scores include the Empirical Cumulative distribution functions for Outlier Detection (ECOD) [44], the Simple univariate Probabilistic Anomaly Detector (SPAD) algorithm [45], and the Histogram-Based Outlier Score (HBOS) algorithm [46]. A brief description of the latter now follows.

HBOS is a histogram-based anomaly detection algorithm with high computational efficiency whose goal is to assign a score to a given test input x (e.g., an image). This score provides information as to the probability that this input is an outlier with respect to a reference set. Suppose we have a reference dataset (e.g., a training dataset) whose samples are represented in terms of d features. For each data feature, a normalized, univariate histogram HG_i, $i = 1, \ldots d$, is constructed for all instances in the training dataset. Next, given a test input x, we extract the corresponding d feature values. Then, for each histogram HG_i, we find the bin b_i in which the i-th feature value falls. The notation $hist_i(x)$ is used to indicate the height of the bin b_i within HG_i. Finally, a score $s(x)$ is assigned to the test point x according to Equation (1) as follows:

$$s(x) = \sum_{i=1}^{d} \log\left(\frac{1}{hist_i(x)}\right) \tag{1}$$

2.2. A UQ Method for Model Performance Prediction Based on Data Dissimilarity Measures

In this section, we outline a UQ method aimed at predicting the performance of an ML-based model using data dissimilarity measures. The UQ method also returns an estimate of the uncertainty interval associated with a performance prediction. The proposed UQ method is an extension of the technique discussed in [7,47], where the authors introduce the idea of studying the relationship between ANN classifier performance and data dissimilarity in the context of ML verification. The degree of shift is to be gauged by data dissimilarity measures. Verification might be undertaken in practice by recording classifier accuracy at design time for a series of test datasets, which are designed or selected to return progressively increasing dissimilarity values when compared to the training

dataset. This enables a performance–dissimilarity relationship to be established and then verified against robustness requirements. Moreover, they suggest using the following process for run-time performance prediction and verification. Upon receipt of an unseen operational dataset, calculate its dissimilarity from the training dataset. Then, employ this measurement in combination with the performance–dissimilarity relationship observed at design time in order to make a prediction of the expected level of accuracy of the classifier for the operational run-time dataset.

Here, we complement the technique introduced in [7] by adding performance estimation uncertainty. Specifically, we assume that the relationship between model performance and input data dissimilarity is statistical in nature and, as observed in [7], decreasing. We can therefore characterize this relationship in terms of an expected trend and an uncertainty band, shown as a solid-lined orange curve and by dashed orange curves in Figure 1, respectively. In line with [7], this plot would be empirically determined using (data dissimilarity, performance) measurement pairs generated from a series of available test datasets. In Figure 1, these empirical pairs are illustrated as blue circles.

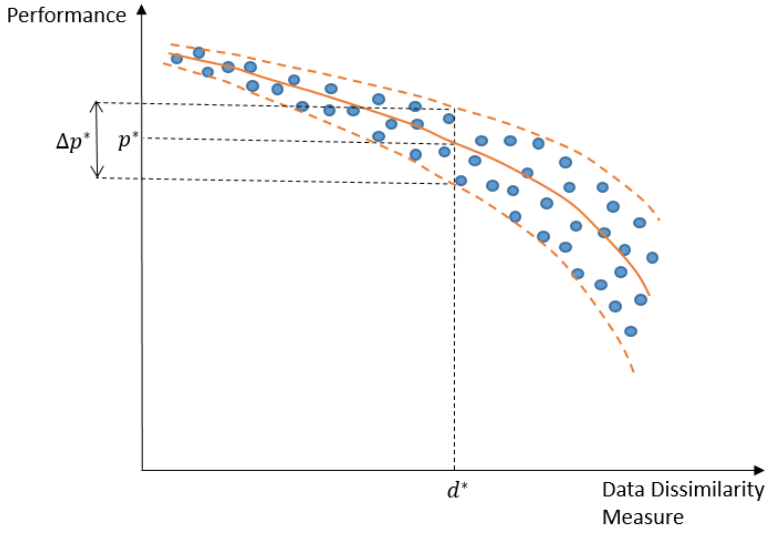

Figure 1. An illustrative, statistical relationship between model performance and input data dissimilarity.

Figure 1 can also be used to show how performance can be predicted having empirically established a relationship between data dissimilarity and performance. Suppose we are given an unseen operational dataset whose data dissimilarity from the model training dataset is measured as d^* (see Figure 1). Then, reference to the pre-established performance–dissimilarity relationship enables not only a determination of the performance point estimate p^* for the operational dataset, but also of its uncertainty interval Δp^*.

To summarize, we first establish a statistical relationship between measures of ML performance and measures of dataset dissimilarity to characterize the behavior of an ANN model of interest in the presence of data shift. Then, we use this relationship to predict the expected performance level, and the corresponding uncertainty, that would be achieved by the model when operating on unseen data. Therefore, the method provides information about the performance (e.g., accuracy) of the model when processing out-of-distribution data, and quantifies the confidence with which it can be applied to the problem under consideration.

It should be noted that our proposed UQ method is not only limited to ANN classifiers but can also be applied to ML-based models more broadly. This is because the proposed UQ method requires only two key elements: (i) a measure of model performance, and (ii) a measure of dataset dissimilarity. In other words, any application that allows for the definition of a performance metric and a data dissimilarity measure could potentially benefit from this UQ method. However, we recognize that the selection of these measures

is problem specific, and would need to take into account the particular requirements of the application considered.

2.3. Anomaly-Based Dataset Dissimilarity (ADD) Measures

2.3.1. Description of ADD Measures

Here, we introduce a novel family of data dissimilarity measures based on anomaly detection algorithms which we refer to as Anomaly-based Dataset Dissimilarity (ADD) measures. Given two datasets of interest, D_{test1} and D_{test2}, our goal is to quantify their statistical dissimilarity via a suitably defined measure $d(D_{test1}, D_{test2})$. As anticipated in Section 1, this measure will be formulated so that it returns an indication of the divergence between the data-generating distributions associated with the datasets being compared. The steps that comprise an ADD computation are given below.

First, we determine a secondary representation of the two datasets by projecting them from their primary, raw form onto a suitably chosen feature space. Then, in the latter feature space, we characterize each dataset with respect to a third, reference dataset, which has also been projected onto the feature space. This operation employs anomaly detection algorithms to compute the degree to which the data instances of D_{test1} and D_{test2} are anomalous with respect to the reference dataset. In other words, the reference dataset is used to define a notion of data normality for the purpose of calculating anomaly scores. Finally, the anomaly scores computed for all dataset instances are used to calculate a measure of the data dissimilarity between D_{test1} and D_{test2}, i.e., $d(D_{test1}, D_{test2})$. The data dissimilarity measure therefore reflects the differences between the anomaly scores computed for D_{test1} and those computed for D_{test2}.

For the purpose of generality, in the steps outlined above, we have deliberately not specified how to derive secondary data representations and which anomaly detection algorithms to use. This highlights the flexibility of our proposed measures as it allows for the definition of ADD variants by selecting different feature extraction methods and diverse anomaly detection algorithms. In this respect, our proposed measures constitute a family of data dissimilarity measures, such that specific ADD variants could be defined taking into account the characteristics and requirements of a given task.

In the sections below, we describe a feature extraction process based on ANN activation values, and a computation of anomaly scores based on the HBOS algorithm. The HBOS algorithm is selected because it is numerically efficient and applicable to high-dimensional feature spaces.

2.3.2. Feature Extraction Process

The selection of a suitable secondary data representation should take into account multiple factors and is generally dependent on the properties of relevant datasets. In other words, different types of data (e.g., images, text, radar measurements) could benefit from different feature descriptions. Furthermore, computational constraints might impose limitations on the employability of expensive representations, and the nature of the problem addressed might require additional considerations such as data anonymization due to privacy-related requirements.

A specific two-step feature extraction process is selected for experimentation in this study:

(i) A raw dataset is fed to a CNN, i.e., a chosen ANN. The activation values output by a fixed subset of CNN neurons are recorded for each data instance. Each data instance is thus effectively mapped to a feature vector, each of whose components corresponds to the activation value output by a given neuron. This is an intermediate representation of the raw dataset;

(ii) A transformation is applied to the aforementioned feature vectors to determine a further feature vector representation. This is the secondary representation of the raw dataset.

More precisely, we use a reference set D_{train} to train a selected CNN to which the instances of D_{test1} and D_{test2} are then fed as inputs. A subset of N_{neu} neurons of this CNN is selected, whose output activation values are used to compute an intermediate feature representation of D_{test1} and D_{test2}. Looking ahead to our experiments, we define the subset to be the penultimate layer of the chosen CNN, and set N_{neu} to be equal to the number of neurons in this penultimate layer. This means that for each data instance x of the datasets being compared, we determine its corresponding activation trace $A(x) \in \mathbb{R}^{N_{neu}}$ as a vector whose components are the activation values of the N_{neu} selected neurons. Next, we employ a projection mechanism (i.e., a vector-to-vector transformation) that maps the activation traces to a final feature vector representation.

In this work, we define and compare the projection mechanisms specified below.

Projection Mechanism 0 (PM0): represents an identity mapping. Mathematically, $\text{PM0}: \mathbb{R}^{N_{neu}} \rightarrow \mathbb{R}^{N_{neu}}$, where a generic dataset instance x is transformed to a final representation given by the feature vector $A_{pm0}(x) = A(x)$. In other words, data dissimilarity measures are directly computed in a feature space defined by neuron activation values.

Projection Mechanism 1 (PM1): final features are defined by multiplying a fixed number of CNN activation values randomly chosen from the neuron subset. Mathematically, $\text{PM1}: \mathbb{R}^{N_{neu}} \rightarrow \mathbb{R}^{N_{pm1}}$, $N_{pm1} < N_{neu}$, such that a generic input x is assigned a final representation given by the feature vector $A_{pm1}(x) \in \mathbb{R}^{N_{pm1}}$, where each vector component $A^{(j)}_{pm1}(x)$, $j = 1, \ldots, N_{pm1}$, is the product of three randomly selected components of the activation trace $A(x)$.

Projection Mechanism 2 (PM2): defines a final representation by applying a Principal Component Analysis to the set of activation traces generated by the data instances in D_{train}. That is, the principal directions (i.e., components) of the activation traces associated with D_{train} are first calculated and subsequently used to define a feature space onto which the instances of D_{test1} and D_{test2} are projected. Mathematically, $\text{PM2}: \mathbb{R}^{N_{neu}} \rightarrow \mathbb{R}^{N_{pm2}}$, $N_{pm2} < N_{neu}$. The final representation is thus a feature vector comprising the projections of $A(x)$ onto the first N_{pm2} principal components. The selection of the dimensionality of the new feature space, comprising N_{pm2} principal components, is made on the basis that at least 90% of the variance of the training activation traces is retained.

Projection Mechanism 3 (PM3): represents a variation of PM2. Mathematically, $\text{PM3}: \mathbb{R}^{N_{neu}} \rightarrow \mathbb{R}^{N_{pm3}}$, $N_{pm3} < N_{neu}$. Here, test dataset instances are projected onto a feature space defined by lower order principal components of the training activation traces, i.e., those principal components associated with lower variance. Specifically, for a given dataset instance x, only the last N_{pm3} principal components of the N_{pm2} computed as described for PM2 are employed to define the final representation of x, with $N_{pm3} < N_{pm2}$.

We have defined the above mappings in order to overcome some of the known limitations of the anomaly detection algorithm that has been adopted for the ADD computation. PM0 is formulated for benchmarking purposes. PM1 is introduced in response to the observations discussed in [45], according to which the hypothesis of feature independence used to justify several anomaly detection algorithms will not always hold; consequently, more effective outlier detection would require feature interdependence to be taken into account. PM2 is investigated for the following reasons:

- Principal component analysis accounts for linear interdependencies between the variables corresponding to CNN neuron activation values;
- PCA analysis leads to a dimensionality reduction of the $A(x)$ feature vector (i.e., data compression) which, in turn, lowers the computational costs of the subsequently applied HBOS algorithm.

PM3 is a variation of PM2 motivated by the following observation. If the variance of the values generated by projecting training activation traces onto a principal component is low, it might be more likely that anomalies will be detected in that direction when projecting test activation traces.

In summary, the above projection mechanisms define four distinct feature spaces in which anomaly scores can be computed. This leads to four distinct ADD measures, which will be evaluated and compared in Section 3.

2.3.3. Anomaly Scores

We employ the HBOS algorithm [46] to assign an anomaly score to each data instance in D_{test1} and D_{test2} using D_{train} as a reference set. As described in Section 2.3.1, this operation is performed within a secondary feature space defined through the projection mechanisms described in Section 2.3.2. We denote by $\left\{s^{test1}(x_i)\right\}_{i=1}^{N_{test1}}$ the anomaly scores associated with the N_{test1} data instances of D_{test1}, and by $\left\{s^{test2}(x_i)\right\}_{i=1}^{N_{test2}}$ those corresponding to the N_{test2} data instances of D_{test2}.

2.3.4. Anomaly-Based Dataset Dissimilarity—Formula

A formula is now given which converts the anomaly scores calculated for the data instances in D_{test1} and D_{test2} into a final dissimilarity value $d(D_{test1}, D_{test2})$. A histogram $\mathcal{H}_{test1}$ is constructed to represent the distribution of the anomaly scores $\left\{s^{test1}(x_i)\right\}_{i=1}^{N_{test1}}$ associated with D_{test1}; similarly, the histogram $\mathcal{H}_{test2}$ represents the anomaly scores $\left\{s^{test2}(x_i)\right\}_{i=1}^{N_{test2}}$ associated with D_{test2}. Histogram bins b are specified so that they cover the full range of HBOS scores generated for the datasets, with the bins being common to both histograms. Dynamic histogram bins are employed following [46]. Thereafter, the degree of overlap between these two histograms is calculated, where a greater degree of overlap indicates a lower dissimilarity. The overlap computation is used to quantify the dissimilarity between the datasets being compared by means of the following formula:

$$d(D_{test1}, D_{test2}) = 1 - \frac{\sum_b \min(\mathcal{H}_{test1}(b), \mathcal{H}_{test2}(b))}{\sum_b \mathcal{H}_{test1}(b)} \tag{2}$$

where the sums are computed over all histogram bins, and $\mathcal{H}_{test}(b)$ returns the normalized frequency that a histogram $\mathcal{H}_{test}$ records for a particular bin b. The absence of any overlap between $\mathcal{H}_{test1}$ and $\mathcal{H}_{test2}$ results in the right, fractional term of Equation (2) returning 0; if $\mathcal{H}_{test1}$ and $\mathcal{H}_{test2}$ are identical, the right term returns 1. Thus, it follows that $d(D_{test1}, D_{test2})$ is defined within the range $[0, 1]$. Furthermore, the greater the statistical dissimilarity between D_{test1} and D_{test2}, the greater the value of the measure.

2.4. Materials and Experimental Setting

2.4.1. Datasets

To assess the performance of our proposed data dissimilarity measures, we employ two public domain datasets: MNIST [48] and CIFAR-10 [49]. The MNIST dataset is randomly split into three subsets, as follows:

- A training set D_{train}^{MNIST} comprising 51,000 images;
- A validation set D_{val}^{MNIST} comprising 9000 images;
- A test set D_{test}^{MNIST} comprising 10,000 images.

The training set D_{train}^{MNIST} is used to train CNNs for feature extraction, as described in Section 2.3. The validation set D_{val}^{MNIST} is used to monitor the training process. The test set D_{test}^{MNIST} is used to evaluate the performance of trained networks on unseen data.

Similarly, we randomly split the CIFAR-10 dataset into three subsets, as follows:

- A training set D_{train}^{CIFAR} comprising 50,000 images;
- A validation set D_{val}^{CIFAR} comprising 9000 images;
- A test set D_{test}^{CIFAR} comprising 1000 images.

In addition, following [50], the training dataset D_{train}^{CIFAR} is subsequently extended via random image cropping data augmentation.

We recall that the objective of the proposed ADD measures is to quantify the dissimilarity between two datasets, namely D_{test1} and D_{test2}. In this work, we generate these datasets as follows. With respect to MNIST, a baseline dataset D_{test1} is prepared by randomly selecting a subset of 1000 images from D_{test}^{MNIST}. With respect to CIFAR-10, we adopt D_{test}^{CIFAR} as D_{test1}. D_{test2}, on the other hand, is generated by applying a synthetic transformation to the images in D_{test1}. For instance, a test dataset D_{test2} can be created by rotating each image in D_{test1}. For our experimentation and numerical analysis, the following image transforms are employed: image rotation, blur (performed via a Gaussian filter with standard deviation σ), the addition of pixel-wise random Gaussian noise (with zero mean and standard deviation σ), and brightness variation (performed by adding a constant value μ to each pixel).

The choice of the procedure above for generating synthetic data is motivated by the fact that by making step-wise changes to the magnitude of the applied data transforms, the degree of dissimilarity between D_{test1} and D_{test2} can be systematically controlled. For example, with reference to image rotation, a series of T datasets $\left\{D_{test2}^{(t)}\right\}_{t=1}^{T}$ can be generated from D_{test1} by applying rotations of progressively greater magnitude. A dataset $D_{test2}^{(t)}$ in the series might be produced by rotating all the images in D_{test1} by $t\theta$, where θ is a constant. For instance, if $\theta = 5°$, the first datasets in the series would be generated by rotating images in D_{test1} by $5°$, $10°$, $15°$, ... Such a series of datasets will be progressively more dissimilar to D_{test1} in terms of orientation. The ability of the ADD measures to respond to progressively more perturbed datasets can then be studied by quantifying the data dissimilarity between D_{test1} and each dataset in the series $\left\{D_{test2}^{(t)}\right\}_{t=1}^{T}$.

2.4.2. Neural Networks

In order to examine the relationship between dataset dissimilarity and ML model performance, we train models to classify MNIST and CIFAR-10 images. We apply several convolutional neural network architectures to the image classification tasks that we undertake. Specifically, for MNIST data, we employ two variants of the LeNet-5 CNN [51] that differ in terms of the activation functions operating within their hidden layers. We refer to [51] for a description of the technical details of the LeNet-5 architecture. It comprises seven layers: two convolutional layers, each followed by an average pooling layer, and three dense layers. The first variant uses the Rectified Linear Unit (ReLU), while the second variant uses the hyperbolic tangent (tanh) activation function [29]. We select the $N_{neu} = 84$ neurons which constitute the penultimate layer of these CNNs to populate the activation traces (as defined in the feature extraction process described in Section 2.3). Furthermore, for CIFAR-10 data, we employ a ResNet-18 CNN [50], namely an 18-layer network with shortcut connections, and use the $N_{neu} = 512$ neurons in its penultimate layer for feature extraction. However, we highlight that other ANN models could play the same role. The implementations of these CNNs access the Keras (version 2.10.0) and TensorFlow (version 2.10.0) Python libraries [52,53]. CNN training and evaluation is conducted on a NVIDIA GeForce RTX 3080 Ti GPU and an Intel(R) Xeon(R) CPU. The latter CPU is also used to compute ADD measures based on CNN activation values.

3. Results

We perform a numerical analysis aimed at evaluating the following:

- The ability of the proposed ADD measures to indicate progressively greater dissimilarity when applied to a series of datasets which have been systematically transformed to a progressively greater extent;
- The applicability of the ADD measures to the UQ method designed to predict ML model performance and quantify the associated uncertainty, as outlined in Section 2.2.

Each step of the investigation employs both the LeNet-5 CNNs applied to the MNIST dataset and the ResNet-18 CNN applied to the CIFAR-10 dataset.

3.1. Numerical Results and Evaluation: The Relationship between the Magnitude of Image Transform Parameters and ADD Measures

In this section, we perform a numerical analysis aimed at evaluating the responsiveness of the ADD measures to specific image transformations, which are applied to induce data dissimilarity. To this end, we employ ADD measures to quantify the discrepancy between pairs of datasets $\left\{ \left(D_{test1}, D_{test2}^{(t)} \right) \right\}_{t=1}^{T}$, $T = 6$. Since the datasets $\left\{ D_{test2}^{(t)} \right\}_{t=1}^{T}$ are generated from D_{test1}, by applying an image transform whose parameter is progressively increased, they are also expected to be progressively more dissimilar to D_{test1} in terms of ADD measures. In other words, when characterizing the relationship between the magnitude of the image transform parameter and the ADD measure value, a monotonically increasing trend is expected.

We begin by examining the relationship between ADD measure values and the magnitude of the applied image transformations. Four ADD measures are evaluated and these are differentiated by, and will be denoted by, the form of their projection mechanism. The mechanisms are termed PM0, PM1, PM2 and PM3, as described in Section 2.3.2. In addition, we adopt the following parameter values for the projection mechanisms: $N_{pm1} = 60$; $N_{pm2} = 40$ and $N_{pm3} = 10$ when processing MNIST images; $N_{pm2} = 80$ and $N_{pm3} = 25$ when processing CIFAR-10 images.

The results of this analysis are summarized in Figures 2 and 3, for the MNIST dataset, and in Figure 4, for the CIFAR-10 dataset. These figures present results for the following input data transformations: image rotation, image blur, additive random Gaussian noise, and change in image brightness. The results associated with different image transformations are illustrated in different charts within these figures.

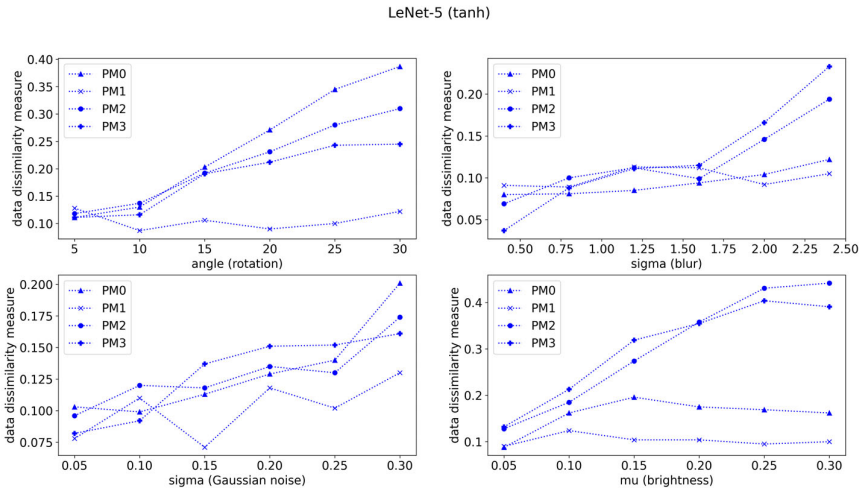

Figure 2. Data dissimilarity measure values against amplitude for four image transformations. The trends correspond to ADD measures that use PM0, PM1, PM2, and PM3, respectively. The measures are applied to a LeNet-5 CNN with tanh activation functions.

We first examine the application of the data dissimilarity measures to two LeNet-5 CNNs, one with tanh activation functions (Figure 2), and one with ReLU activation functions (Figure 3). Figures 2 and 3 show that, as anticipated above, the general trend is for data dissimilarity measure values to increase with the amplitude of the applied transformations. However, some deviations from that trend can be observed. Overall, these trends suggest that our proposed dissimilarity measures are sensitive to the progressively greater perturbations used to generate the series of datasets (at least for this set of image transformations, and with respect to features extracted via the specified projection mechanisms).

The results for the individual projection mechanisms and image transforms will now be examined in more detail. Computing data dissimilarity measures directly from network activation values, i.e., subsequent to PM0, leads to curves that increase with transform

magnitude. However, some irregularities occur; that is to say, trends which are not perfectly monotonically increasing. These can be observed for the image rotation transform, where the network applies ReLU activation functions, and for the image brightness transform, in the case of network tanh activation functions. The curves corresponding to PM1 generally do not display an increasing trend. This is particularly evident with respect to the tanh activation function (see Figure 2). The ADD measures that use PM2 and PM3, which process features defined by principal components derived from CNN activation values, produce more markedly upward trends than those associated with PM1. Moreover, the measures employing PCA-based projection mechanisms outperform in terms of monotonicity the benchmark PM0 measure for image blur with ReLU activations, and change in image brightness with tanh activations. In addition, we note that the ADD variants (i.e., ADD measures based on different projection mechanisms) respond differently to the image transformations examined. For instance, with reference to ReLU activations, the measure based on PM0 produces monotonically increasing trends in all cases except image rotation.

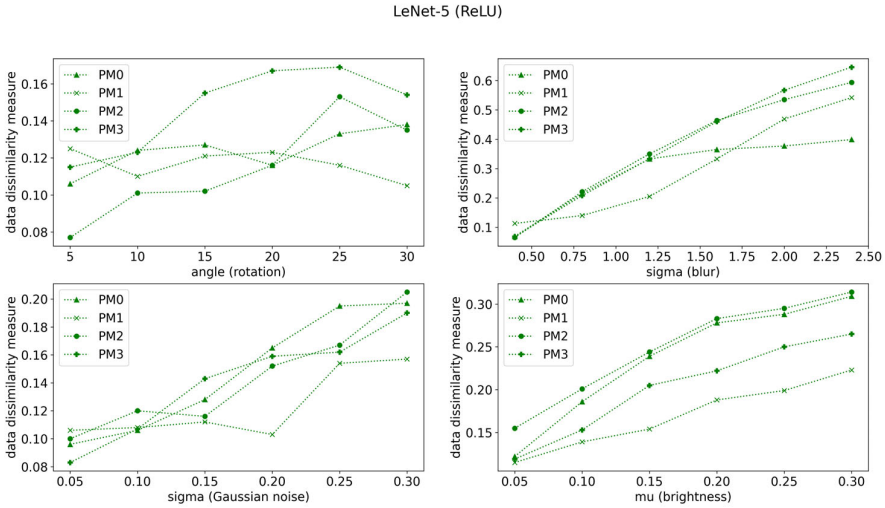

Figure 3. Data dissimilarity measure values against amplitude for four image transformations. The trends correspond to ADD measures that use PM0, PM1, PM2, and PM3, respectively. The measures are applied to a LeNet-5 CNN with ReLU activation functions.

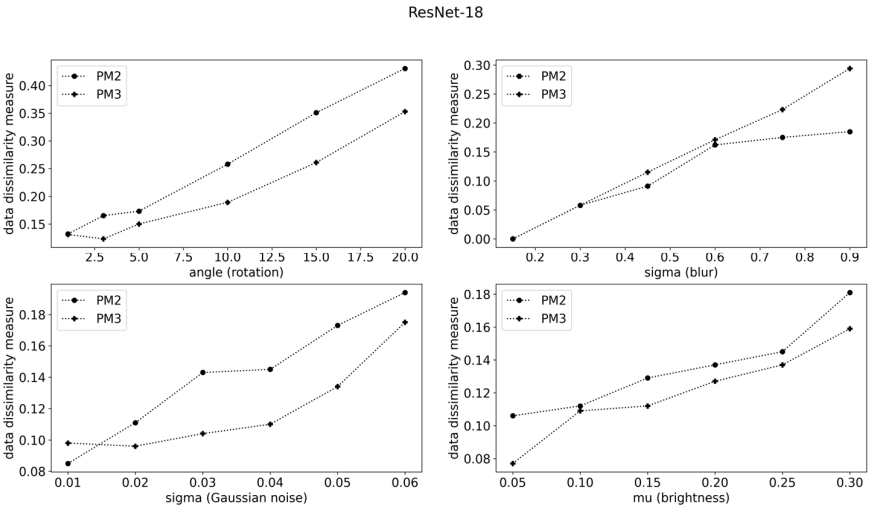

Figure 4. Data dissimilarity measure values against amplitude for four image transformations. The trends correspond to ADD measures that use PM2 and PM3, respectively. The measures are applied to a ResNet-18 CNN.

A quantitative analysis of the monotonicity of the trends displayed in Figures 2 and 3 based on the computation of the Spearman's correlation coefficient is provided in Tables 1 and 2, respectively. A coefficient value of 1 indicates a perfectly monotonic increasing trend and a value of −1 indicates a perfectly monotonic decreasing trend. Table 1 (tanh activation functions) shows that PM2 and PM3 lead to trends which are closest to being increasing monotonic for all transforms. Table 2 (ReLU activation functions) records that PM0 and PM2 yield the most consistently monotonic increasing trends. Both tables reveal that PM1 generates relationships which are furthest from the expected monotonicity.

Table 1. Spearman's correlation coefficients associated with the trends displayed in Figure 2.

Spearman's Coefficient	Rotation	Blur	Gaussian Noise	Brightness
PM0-based ADD	1.00	1.00	0.94	0.32
PM1-based ADD	−0.03	0.43	0.60	−0.06
PM2-based ADD	1.00	0.83	0.89	1.00
PM3-based ADD	1.00	1.00	1.00	0.94

Table 2. Spearman's correlation coefficients associated with the trends displayed in Figure 3.

Spearman's Coefficient	Rotation	Blur	Gaussian Noise	Brightness
PM0-based ADD	0.83	1.00	1.00	1.00
PM1-based ADD	−0.60	1.00	0.66	1.00
PM2-based ADD	0.94	1.00	0.94	1.00
PM3-based ADD	0.66	1.00	1.00	1.00

Figure 4 shows the ADD measure trends with respect to image transformation magnitude which are obtained for a further neural network and dataset pair, a ResNet-18 CNN trained on CIFAR-10. Here, CIFAR-10 images are subject to image transforms, and the ADD measures are computed from ResNet-18 CNN activation values. In order to limit the computational costs of experimentation, in this analysis, we only focus on the PM2- and PM3-based ADD variants. These ADD variants produce similar results. However, the project mechanism PM2 yields perfectly monotonic trends for all four image transforms, whilst the selection of PM3 leads to small irregularities for image rotation and the addition of Gaussian noise transforms. The Spearman's correlation coefficients associated with the curves displayed in Figure 4 are provided in Table 3.

Table 3. Spearman's correlation coefficients associated with the trends displayed in Figure 4.

Spearman's Coefficient	Rotation	Blur	Gaussian Noise	Brightness
PM2-based ADD	1.00	1.00	1.00	1.00
PM3-based ADD	0.94	1.00	0.94	1.00

3.2. Numerical Results and Evaluation: The Relationship between ADD Measures and CNN Classification Performance

In this section, we investigate the applicability of the ADD measures to the UQ method outlined in Section 2.2. This UQ method requires the establishment of a relationship between model classification performance and dataset dissimilarity. To be more specific, a relationship must be found between the performance that an ML-based model attains for each of a set of test datasets, and the dissimilarity of each of those test datasets to the training dataset. In general, model performance is measured in terms of a metric such as accuracy, and dissimilarity in terms of a data dissimilarity measure such as an ADD measure.

Section 3.2 extends the analysis of the experiments reported in Section 3.1, where dataset dissimilarities are synthetically induced by means of image transformations. We

process the MNIST and CIFAR-10 datasets, and employ LeNet-5 and ResNet-18 CNNs to extract data features for ADD computations.

Figures 5 and 6 summarize the results obtained when the two LeNet-5 CNNs are applied to the MNIST dataset. Overall, they show that the general trend is for accuracy to decrease as the data dissimilarity measure values increase. This general trend is in line with the curve shown in Figure 1, and confirms that the neural networks examined are less capable of correctly classifying progressively more perturbed images. A more detailed visual inspection of Figures 5 and 6 leads to the following observations. The trends associated with PM0 are overall decreasing, but some deviations appear for the image brightness transform, seen for the CNN with tanh activations, and for image rotation, seen for the CNN with ReLU activations. The ADD variant based on PM1 produces subpar results, especially for tanh activation functions. Decreasing trends are recorded for the ADD measures corresponding to the projection mechanisms PM2 and PM3. However, some small irregularities occur for PM2 and PM3, as can be observed for image rotation with ReLU activations.

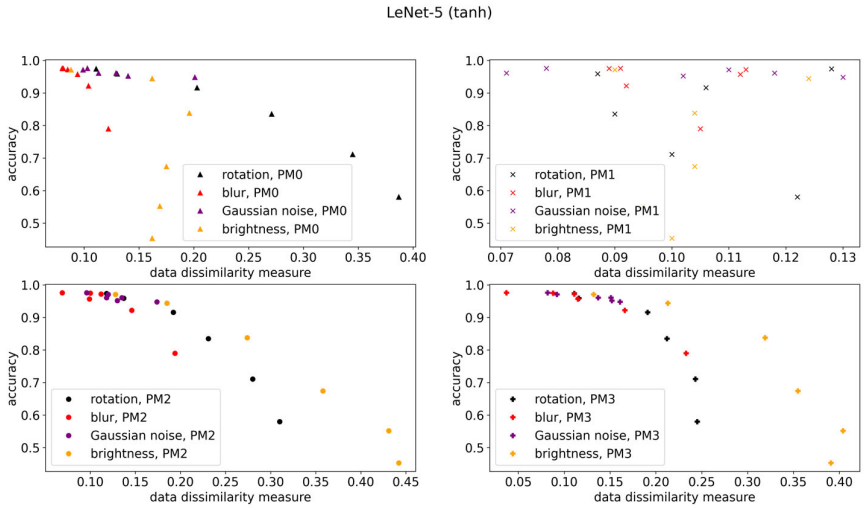

Figure 5. Accuracy against ADD measure values for a LeNet-5 CNN with tanh activation functions. The trends observed for different ADD variants are displayed in separate charts.

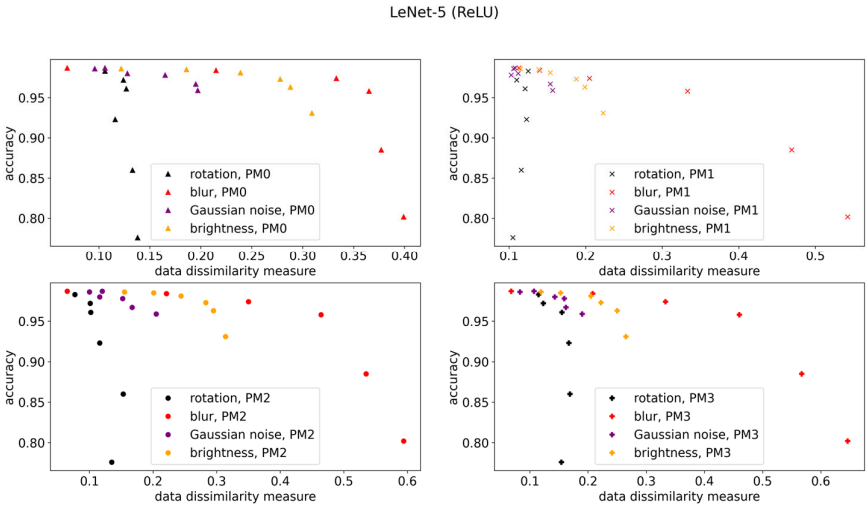

Figure 6. Accuracy against ADD measure values for a LeNet-5 CNN with ReLU activation functions. The trends observed for different ADD variants are displayed in separate charts.

Figure 7 shows the results obtained when the ResNet-18 CNN is applied to the CIFAR-10 dataset. For computational efficiency, only PM2- and PM3-based ADD variants are

considered. Analogously to what was observed in Figures 5 and 6, the general trend is for accuracy to decrease as the data dissimilarity measure values rise. A visual comparison of the projection mechanisms being investigated indicates that PM3 leads to slightly more irregular trends, especially for lower magnitude random Gaussian noise perturbations.

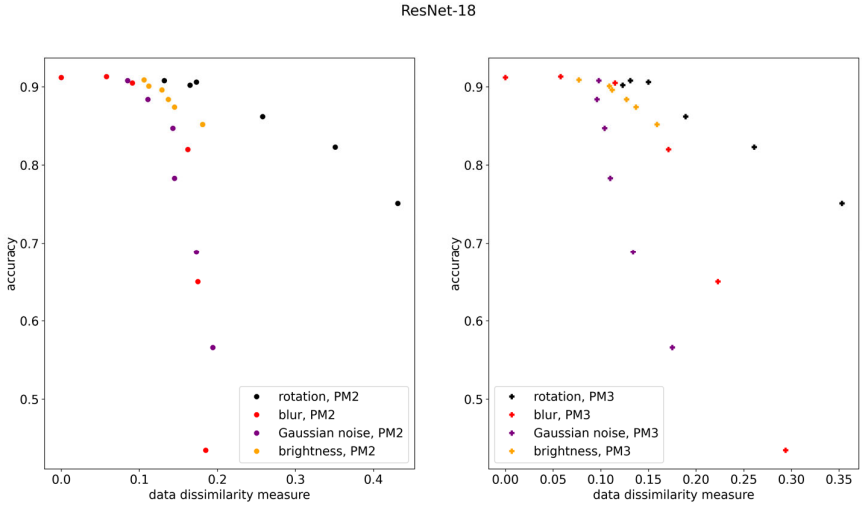

Figure 7. Accuracy against ADD measure values for a ResNet-18 CNN. The trends observed for different ADD variants are displayed in separate charts.

3.3. Numerical Results and Evaluation: UQ Method for Predicting CNN Performance Using ADD Measures

The relationships presented in Section 3.2 can be used for predicting the performance of CNN models at run-time, as well as estimating the uncertainty intervals associated with those predictions, using the UQ method outlined in Section 2.2. This necessitates the determination of a curve that gives the expected CNN performance level as a function of the input data dissimilarity, and delineating the uncertainty band associated with this curve (see Figure 1).

We assume that the relationship between CNN performance p and data dissimilarity measure d can be approximated as follows:

$$p = a * d^2 + p_0 \tag{3}$$

where p_0 is the CNN classification performance recorded for the unperturbed dataset D_{test1} (corresponding to an ADD measure value equal to zero), and $a \in \mathbb{R}$ is a free parameter that we determine through a least squares curve-fitting process.

Following [54,55], a measure of error for the fitted curve is calculated as a confidence band $p \pm k \cdot u_{conf}$, with $k = 1$, which encompasses all the possible curves that might fit the data with approximately 70% confidence. The u_{conf} component of the confidence band is derived from Equation (3): $u_{conf}(d) = 2ad \cdot \sigma_d$, where σ_d is the standard deviation of the ADD measure values over all image transforms. The confidence band provides information about the uncertainty of the predicted CNN performance with respect to the data dissimilarity measure. Although we have adopted this particular form of uncertainty band for the fitted curve (see Figure 1), there are a number of alternative ways in which the band could have been defined.

Curves and confidence bands are fitted to the scatter graph relationships shown in Figures 5–7 using the formulas given above. We first turn our attention to the scatter graphs which are generated for the two LeNet-5 CNNs and the MNIST dataset, namely Figures 5 and 6. Since the PM1-based measure yields a subpar relationship between accuracy and dissimilarity, it is neglected in this analysis. An inspection and a visual comparison of the fitted curves and associated uncertainty bands reveal that they are broadly similar for PM0, PM2, and PM3. Figure 8 shows that the width of the uncertainty

band tends to increase as the data dissimilarity measure values rise. However, some differences can be noted. First, the choice of the activation function markedly affects the range of the accuracy values returned by the models for the examined data transformations. More specifically, using tanh activation functions leads to a larger range of accuracies than using ReLU activation functions over the same set of perturbed datasets. Second, the width of the uncertainty band associated with ReLU activation functions is generally lower than that associated with tanh activation functions. These observations are significant since, in practice, decision-making processes might rely on worst-case performance predictions, and would therefore benefit from narrow uncertainty intervals.

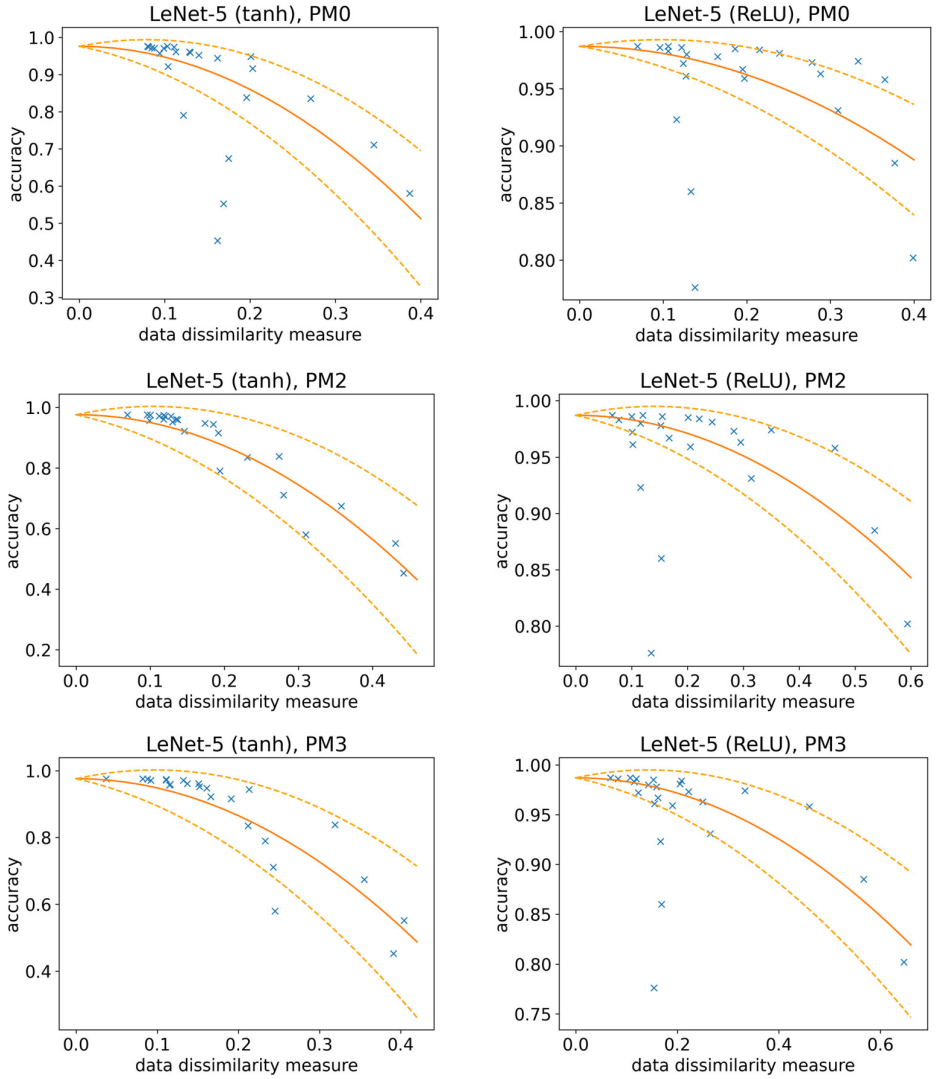

Figure 8. Network accuracy against PM0-, PM2-, and PM3-based ADD measures. Left column: LeNet-5 CNN with tanh activation functions; right column: LeNet-5 CNN with ReLU activation functions. Solid orange lines chart the fitted functions; dotted orange lines indicate the associated uncertainty bands.

With reference to Figure 8, the results of a quantitative analysis based on the calculation of the Root-Mean-Square Error (RMSE) between empirical points and fitted curves are summarized in Table 4. The average RMSE over the ADD measure variants is lower for the LeNet-5 CNN which uses ReLU activation functions (0.054) than for the LeNet-5 CNN which uses tanh activation functions (0.084).

Table 4. Root-Mean-Square Error (RMSE) between the empirical points and the fitted curves shown in Figure 8.

RMSE	LeNet-5 (tanh)	LeNet-5 (ReLU)
PM0-based ADD	0.132	0.056
PM2-based ADD	0.049	0.053
PM3-based ADD	0.071	0.052

Figure 9 shows the parameterized relationships between accuracy and ADD measures for the ResNet-18 network and for CIFAR-10 data. In a similar manner to Section 3.1, we only consider PM2- and PM3-based ADD variants for computational efficiency. A visual inspection of the results obtained shows that although similar decreasing trends are observed, PM2 produces a slightly narrower uncertainty band than PM3.

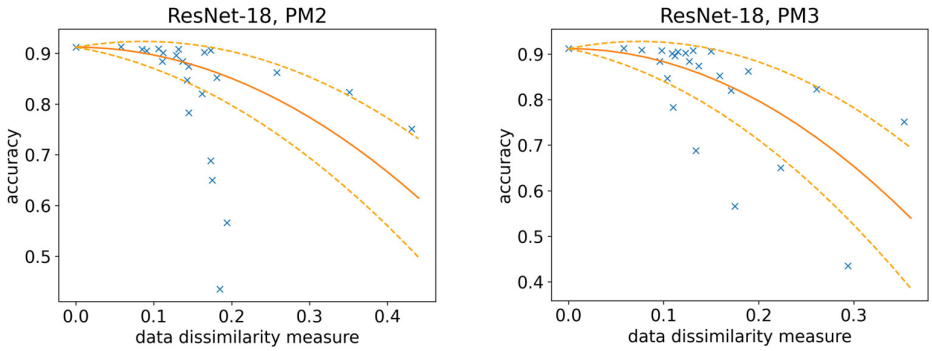

Figure 9. ResNet-18 CNN accuracy against PM2- and PM3-based ADD measures. Solid orange lines chart the fitted functions; dotted orange lines indicate the associated uncertainty bands.

With reference to Figure 9, the results of a quantitative analysis based on the calculation of the Root-Mean-Square Error (RMSE) between empirical points and fitted curves are summarized in Table 5.

Table 5. Root-Mean-Square Error (RMSE) between the empirical points and the fitted curves shown in Figure 9.

RMSE	ResNet-18
PM2-based ADD	0.127
PM3-based ADD	0.099

Once the relationship between CNN classification performance and a dataset dissimilarity measure has been characterized, it can be used to make predictions of CNN performance at run-time on unseen data. More details are provided in the following example, with reference to the ResNet-18 CNN trained on CIFAR-10, and the PM2-based ADD measure. We begin by synthetically generating new (i.e., unseen) test datasets by applying image transforms to the instances of D_{test}^{CIFAR}. Specifically, we employ the following transforms: (1) pixel-wise addition of random noise drawn from a uniform probability distribution, which yields a test dataset D_{unif}^{CIFAR}; and (2) image magnification, which yields a test dataset D_{mag}^{CIFAR}. Note that these transforms differ from the four which were employed to generate the scatter graphs used for curve fitting. Then, we compute the ADD measure values d_{unif} and d_{mag} corresponding to D_{unif}^{CIFAR} and D_{mag}^{CIFAR}, respectively. Next, we use these measure values in conjunction with the fitted curve (and associated uncertainty band) shown in Figure 9 to infer the CNN accuracy on D_{unif}^{CIFAR} and D_{mag}^{CIFAR}, respectively (and the associated uncertainty intervals). This leads to predicted accuracy intervals that are displayed as black intervals in Figure 10. In order to assess the goodness of these predictions, we compare them with the ground truth accuracy values returned by the CNN under

investigation when fed with D_{unif}^{CIFAR} and D_{mag}^{CIFAR}, respectively. These ground truth values are illustrated as red points in Figure 10. A visual analysis of Figure 10 shows that, for both test datasets investigated, the ground truth value falls within the predicted accuracy range. This confirms the applicability of the UQ method proposed in this paper to CNN performance prediction.

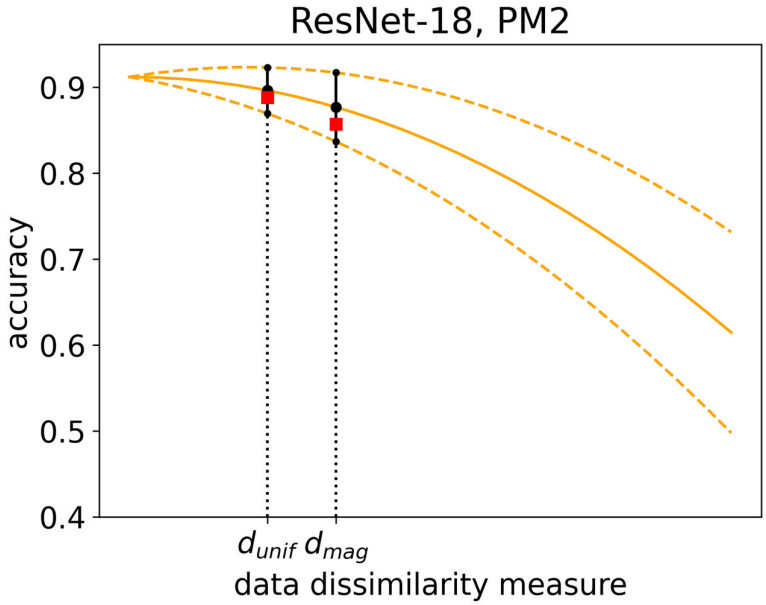

Figure 10. The orange lines chart the parameterized relationship between ResNet-18 CNN accuracy and a PM2-based ADD measure. The solid orange line charts the fitted function; dotted orange lines indicate the associated uncertainty band. The predicted accuracy intervals for the test datasets D_{unif}^{CIFAR} and D_{mag}^{CIFAR} are illustrated in black, whereas the red dots show the corresponding ground truth accuracy values.

4. Discussion

In this paper, we have introduced an Uncertainty Quantification (UQ) method that predicts the performance level of an ML-based model for a given dataset and provides an uncertainty interval for the estimate. The method is directed in particular at predicting performance for data that are statistically dissimilar from model training data. Furthermore, in developing our UQ method, we have addressed the problem of quantifying dataset discrepancies by defining an innovative family of data dissimilarity measures, i.e., the Anomaly-based Dataset Dissimilarity (ADD) measures. These measures combine anomaly detection algorithms and feature representations based on Artificial Neural Network activation values.

We have performed a numerical analysis of the results produced when our UQ method and measures are applied to established CNN models for image classification. We have demonstrated the ability of the ADD measures to respond to progressively more perturbed datasets, especially when PCA-based projection mechanisms are employed. We have recorded statistical relationships between CNN model accuracy and dataset dissimilarity when models are supplied with perturbed versions of prominent public domain datasets. We have fitted parameterized curves with associated uncertainty bands to these relationships. We have observed the following: (i) the general trend is for model accuracy to decrease as ADD values increase; and (ii) the amplitude of the uncertainty band associated with a curve tends to increase with ADD value. We have demonstrated the use of the curves and bands for predicting the accuracy attained by CNN models when supplied with unseen datasets, and for quantifying the uncertainty of those predictions. Our proposed UQ method was shown to be effective, as demonstrated for LeNet-5 and ResNet-18 CNNs.

Further investigations to evaluate our UQ method and ADD measures with respect to other datasets and ML-based models are planned as future work. This UQ method could be further developed for the verification and run-time performance prediction of operational systems composed of multiple ML-based elements.

Author Contributions: Conceptualization, G.I., D.H. and H.A.; methodology, G.I., D.H. and H.A.; software, G.I.; validation, G.I. and D.H.; formal analysis, G.I. and D.H.; investigation, G.I., D.H. and H.A.; data curation, G.I.; writing—original draft preparation, G.I.; writing—review and editing, G.I., D.H. and H.A.; visualization, G.I. and D.H.; supervision, H.A.; project administration, H.A.; funding acquisition, G.I., D.H. and H.A. All authors have read and agreed to the published version of the manuscript.

Funding: The work leading to the publication of our paper [8] was funded by UK MoD DSTL via "Serapis Lot 6 U89 AI Verification, Validation—task6". The extension of that work presented in this paper was funded by Thales UK.

Data Availability Statement: Publicly available datasets were analyzed in this study. Descriptions of the datasets used are provided in [48,49].

Acknowledgments: This work was fully supported by Thales UK - Research, Technology & Innovation. This paper is partially reprinted from a paper published in the AITest 2023 conference proceedings with permission from IEEE.

Conflicts of Interest: Authors Gabriele Incorvaia, Darryl Hond and Hamid Asgari were employed by the company Thales UK. The authors declare that the research was conducted in the absence of any other commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Kumar, Y.; Komalpree, K.; Gurpreet, S. Machine learning aspects and its applications towards different research areas. In Proceedings of the International Conference on Computation, Automation and Knowledge Management, Dubai, United Arab Emirates, 9–10 January 2020.
2. Pugliese, R.; Regondi, S.; Marini, R. Machine learning-based approach: Global trends; research directions, and regulatory standpoints. *Data Sci. Manag.* **2021**, *4*, 19–29. [CrossRef]
3. Siddique, T.; Mahmud, M.S.; Keesee, A.M.; Ngwira, C.M.; Connor, H. A survey of uncertainty quantification in machine learning for space weather prediction. *Geosciences* **2022**, *12*, 27. [CrossRef]
4. Amodei, D.; Olah, C.; Steinhardt, J.; Christiano, P.; Schulman, J.; Mané, D. Concrete problems in AI safety. *arXiv* **2016**, arXiv:1606.06565.
5. Cobb, A.D.; Jalaian, B.; Bastian, N.D.; Russell, S. Toward safe decision-making via uncertainty quantification in machine learning. In *Systems Engineering and Artificial Intelligence*; Springer: Cham, Switzerland, 2021.
6. Abdar, M.; Pourpanah, F.; Hussain, S.; Rezazadegan, D.; Liu, L.; Ghavamzadeh, M.; Fieguth, P.; Cao, X.; Khosravi, A.; Acharya, U.R.; et al. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Inf. Fusion* **2021**, *76*, 243–297. [CrossRef]
7. Hond, D.; Asgari, H.; Jeffery, D.; Newman, M. An integrated process for verifying deep learning classifiers using dataset dissimilarity measures. *Int. J. Artif. Intell. Mach. Learn.* **2021**, *11*, 1–21. [CrossRef]
8. Incorvaia, G.; Hond, D.; Asgari, H. Uncertainty quantification for machine learning output assurance using anomaly-based dataset dissimilarity measures. In Proceedings of the International Conference on Artificial Intelligence Testing, Athens, Greece, 17–20 July 2023.
9. Kendall, A.; Yarin, G. What uncertainties do we need in bayesian deep learning for computer vision? In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; Volume 30.
10. Harakeh, A.; Smart, M.; Waslander, S. Bayesod: A bayesian approach for uncertainty estimation in deep object detectors. In Proceedings of the International Conference on Robotics and Automation, Paris, France, 31 May–31 August 2020.
11. Le, M.T.; Diehl, F.; Brunner, T.; Knoll, A. Uncertainty estimation for deep neural object detectors in safety-critical applications. In Proceedings of the International Conference on Intelligent Transportation Systems, Maui, HI, USA, 4–7 November 2018.
12. Martinez, C.; Potter, K.M.; Smith, M.D.; Donahue, E.A.; Collins, L.; Korbin, J.P.; Roberts, S.A. Segmentation certainty through uncertainty: Uncertainty-refined binary volumetric segmentation under multifactor domain shift. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 16–17 June 2019.
13. Xu-Darme, R.; Girard-Satabin, J.; Hond, D.; Incorvaia, G.; Chihani, Z. Interpretable out-of-distribution detection using pattern identification. *arXiv* **2023**, arXiv:2302.10303.

14. Combalia, M.; Hueto, F.; Puig, S.; Malvehy, J.; Vilaplana, V. Uncertainty estimation in deep neural networks for dermoscopic image classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020.
15. Dusenberry, M.W.; Tran, D.; Choi, E.; Kemp, J.; Nixon, J.; Jerfel, G.; Heller, K.; Dai, A.M. Analyzing the role of model uncertainty for electronic health records. In Proceedings of the ACM Conference on Health, Inference, and Learning, Toronto, ON, Canada, 2–4 April 2020.
16. Licata, R.; Mehta, P. Uncertainty quantification techniques for data-driven space weather modeling: Thermospheric density application. *Sci. Rep.* **2022**, *12*, 7256. [CrossRef] [PubMed]
17. Moosavi, A.; Rao, V.; Sandu, A. Machine learning based algorithms for uncertainty quantification in numerical weather prediction models. *J. Comput. Sci.* **2021**, *50*, 101295. [CrossRef]
18. Ott, M.; Auli, M.; Grangier, D.; Ranzato, M.A. Analyzing uncertainty in neural machine translation. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018.
19. Xiao, Y.; Wang, W. Quantifying uncertainties in natural language processing tasks. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019.
20. Dong, X.; Guo, J.; Li, A.; Ting, W.T.; Liu, C.; Kung, H.T. Neural mean discrepancy for efficient out-of-distribution detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
21. Mahjour, S.K.; da Silva, L.O.M.; Meira, L.A.A.; Coelho, G.P.; Santos, A.A.d.S.d.; Schiozer, D.J. Evaluation of unsupervised machine learning frameworks to select representative geological realizations for uncertainty quantification. *J. Pet. Sci. Eng.* **2021**, *209*, 109822. [CrossRef]
22. Angermann, C.; Haltmeier, M.; Siyal, A. Unsupervised joint image transfer and uncertainty quantification using patch invariant networks. In Proceedings of the Computer Vision—ECCV 2022 Workshops, Paris, France, 2–6 October 2023.
23. Kahn, G.; Villaflor, A.; Pong, V.; Abbeel, P.; Levine, S. Uncertainty-aware reinforcement learning for collision avoidance. *arXiv* **2017**, arXiv:1702.01182.
24. Metelli, A.; Likmeta, A.; Restelli, M. Propagating uncertainty in reinforcement learning via wasserstein barycenters. In Proceedings of the 32nd Annual Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019; Volume 32.
25. Guo, C.; Pleiss, G.; Sun, Y.; Weinberger, K. On calibration of modern neural networks. In Proceedings of the International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017; pp. 1321–1330.
26. Zadrozny, B.; Elkan, C. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In Proceedings of the International Conference on Machine Learning ICML, Williamstown, MA, USA, 28 June–1 July 2001.
27. Kubat, M.; Kubat, J. *An Introduction to Machine Learning*; Springer International Publishing: Cham, Switzerland, 2017.
28. Fawcett, T. An introduction to ROC analysis. *Pattern Recogn. Lett.* **2006**, *27*, 861–874. [CrossRef]
29. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2006.
30. Venturini, G.; Garcıa, A. Statistical Distances and Probability Metrics for Multivariate Data, Ensembles and Probability Distributions. Ph.D. Thesis, Universidad Carlos III de Madrid, Getafe, Spain, 2015.
31. Markatou, M.; Chen, Y.; Afendras, G.; Lindsay, B.G. Statistical distances and their role in robustness. In *New Advances in Statistics and Data Science*; Springer International Publishing: Cham, Switzerland, 2017.
32. Gretton, A.; Borgwardt, K.; Rasch, M.; Schölkopf, B.; Smola, A. A kernel method for the two-sample-problem. In Proceedings of the 20th Annual Conference on Neural Information Processing Systems (NIPS 2006), Vancouver, BC, Canada, 4–7 December 2006; Volume 19.
33. Lee, K.; Lee, K.; Lee, H.; Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, QC, Canada, 3–8 December 2018; Volume 31.
34. Kim, J.; Feldt, R.; Yoo, S. Guiding deep learning system testing using surprise adequacy. In Proceedings of the International Conference on Software Engineering, Montreal, QC, Canada, 25–31 May 2019.
35. Hond, D.; Asgari, H.; Jeffery, D. Verifying artificial neural network classifier performance using dataset dissimilarity measures. In Proceedings of the International Conference on Machine Learning and Applications, Virtual, 14–17 December 2020.
36. Mandelbaum, A.; Weinshall, D. Distance-based confidence score for neural network classifiers. *arXiv* **2017**, arXiv:1709.09844.
37. Melekhov, I.; Juho, K.; Esa, R. Siamese network features for image matching. In Proceedings of the International Conference on Pattern Recognition, Cancun, Mexico, 4–8 December 2016.
38. Motiian, S.; Piccirilli, M.; Adjeroh, D.A.; Doretto, G. Unified deep supervised domain adaptation and generalization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
39. Garcia, N.; Vogiatzis, G. Learning non-metric visual similarity for image retrieval. *Image Vis. Comput.* **2019**, *82*, 18–25. [CrossRef]
40. Chen, H.; Wu, C.; Du, B.; Zhang, L. DSDANet: Deep Siamese domain adaptation convolutional neural network for cross-domain change detection. *arXiv* **2020**, arXiv:2006.09225.
41. Chandola, V.; Arindam, B.; Vipin, K. *Anomaly Detection: A Survey*; ACM Computing Surveys: New York, NY, USA, 2009.
42. Hawkins, D. *Identification of Outliers*; Chapman Hall: London, UK, 1980.
43. Breunig, M.M.; Kriegel, H.P.; Ng, R.T.; Sander, J. LOF: Identifying density-based local outliers. In Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, Dallas, TX, USA, 16–18 May 2000.

44. Li, Z.; Zhao, Y.; Hu, X.; Botta, N.; Ionescu, C.; Chen, G. Ecod: Unsupervised outlier detection using empirical cumulative distribution functions. *IEEE Trans. Knowl. Data Eng.* **2022**, *35*, 12181–12193. [CrossRef]
45. Aryal, S.; Kai, T.; Gholamreza, H. Revisiting attribute independence assumption in probabilistic unsupervised anomaly detection. In *Intelligence and Security Informatics: 11th Pacific Asia Workshop*; Springer International Publishing: Cham, Switzerland, 2016.
46. Goldstein, M.; Dengel, A. Histogram-based outlier score (HBOS): A fast unsupervised anomaly detection algorithm. In Proceedings of the 35th German Conference on Artificial Intelligence (KI-2012), Saarbrücken, Germany, 24–27 September 2012; Volume 1, pp. 59–63; Poster and Demo Track.
47. Hond, D.; Asgari, H.; Symonds, L.; Newman, M. Layer-wise analysis of neuron activation values for performance verification of artificial neural network classifiers. In Proceedings of the International Conference on Assured Autonomy, Virtual, 22–24 March 2022.
48. LeCun, Y.; Cortes, C.; Burges, C. *MNIST Handwritten Digit Database*; ATT Labs (Online): Atlanta, GA, USA, 2010.
49. Krizhevsky, A.; Hinton, G. Learning Multiple Layers of Features from Tiny Images. 2009. Available online: https://www.cs.utoronto.ca/~kriz/learning-features-2009-TR.pdf (accessed on 22 February 2024).
50. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
51. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
52. Github. "Keras". 2015. Available online: https://github.com/keras-team/keras (accessed on 22 February 2024).
53. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Jozefow, R. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Software. 2015. Available online: https://www.tensorflow.org/ (accessed on 22 February 2024).
54. Crowder, S.; Delker, C.; Forrest, E.; Martin, N. *Introduction to Statistics in Metrology*; Springer: Berlin/Heidelberg, Germany, 2020.
55. Delker, C.; Auden, E.; Solomon, O. Calculating interval uncertainties for calibration standards that drift with time. *NCSLI Meas.* **2018**, *12*, 9–20. [CrossRef]