DETECTING MISCLASSIFICATION ERRORS IN NEURAL NETWORKS WITH A GAUSSIAN PROCESS MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

As neural network classifiers are deployed in real-world applications, it is crucial that their predictions are not just accurate, but trustworthy as well. One practical solution is to assign confidence scores to each prediction, then filter out low-confidence predictions. However, existing confidence metrics are not yet sufficiently reliable for this role. This paper presents a new framework that produces more reliable confidence scores for detecting misclassification errors. This framework, RED, calibrates the classifier's inherent confidence indicators and estimates uncertainty of the calibrated confidence scores using Gaussian Processes. Empirical comparisons with other confidence estimation methods on 125 UCI datasets demonstrate that this approach is effective. An experiment on a vision task with a large deep learning architecture further confirms that the method can scale up, and a case study involving out-of-distribution and adversarial samples shows potential of the proposed method to improve robustness of neural network classifiers more broadly in the future.

1 INTRODUCTION

Classifiers based on Neural Networks (NNs) are widely deployed in many real-world applications (LeCun et al., 2015; Anjos et al., 2015; Alghoul et al., 2018; Shahid et al., 2019). Although good prediction accuracies are achieved, lack of safety guarantees becomes a severe issue when NNs are applied to safety-critical domains, e.g., healthcare (Selişteanu et al., 2018; Gupta et al., 2007; Shahid et al., 2019), finance (Dixon et al., 2017), self-driving (Janai et al., 2017; Hecker et al., 2018), etc.

One way to estimate trustworthiness of a classifier prediction is to use its inherent confidence-related score, e.g., the maximum class probability (Hendrycks & Gimpel, 2017), entropy of the softmax outputs (Williams & Renals, 1997), or difference between the highest and second highest activation outputs (Monteith & Martinez, 2010). However, these scores are unreliable and may even be misleading: high-confidence but erroneous predictions are frequently observed (Provost et al., 1998; Guo et al., 2017; Nguyen et al., 2015; Goodfellow et al., 2014; Amodei et al., 2016). In a practical setting, it is beneficial to have a detector that can raise a red flag whenever the predictions are suspicious. A human observer can then evaluate such predictions, making the classification system safer.

In order to construct such a detector, quantitative metrics for measuring predictive reliability under different circumstances are first developed, and a warning threshold then be set based on users' preferred precision-recall tradeoff. Existing such methods can be categorized into three types based on their focus: error detection, which aims to detect the natural misclassifications made by the classifier (Hendrycks & Gimpel, 2017; Jiang et al., 2018; Corbière et al., 2019); out-of-distribution (OOD) detection, which reports samples that are from different distributions compared to training data (Liang et al., 2018; Lee et al., 2018a; Devries & Taylor, 2018); and adversarial sample detection, which filters out samples from adversarial attacks (Lee et al., 2018b; Wang et al., 2019; Aigrain & Detyniecki, 2019).

Among these categories, error detection, also called misclassification detection (Jiang et al., 2018) or failure prediction (Corbière et al., 2019), is the most challenging (Aigrain & Detyniecki, 2019) and underexplored. For instance, Hendrycks & Gimpel (2017) defined a baseline based on maximum class probability after softmax layer. Although the baseline performs reasonably well in most testing cases, reduced efficacy in some scenaria indicates room for improvement (Hendrycks & Gimpel,

2017). Jiang et al. (2018) proposed Trust Score, which measures the similarity between the original classifier and a modified nearest-neighbor classifier. The main limitation of this method is scalability: the Trust Score may provide no or negative improvement over the baseline for high-dimensional data. ConfidNet (Corbière et al., 2019) builds a separate NN model to learn the true class probablity, i.e. softmax probability for the ground-truth class. However, ConfidNet itself is a standard NN, so its confidence scores may be unreliable or misleading: A random input may generate a random confidence score, and ConfidNet do not provide any information regarding uncertainty of these confidence scores. Moreover, none of these methods can differentiate natural classifier errors from risks caused by OOD or adversarial samples; if a detector could do that, it would be easier for practitioners to fix the problem, e.g., by retraining the original classifier or applying better preprocessing techniques to filter out OOD or adversarial data.

To meet these challenges, a new framework is developed in this paper for error detection in NN classifiers. The main idea is to utilize RIO (Residual Input/Output) (Qiu et al., 2020), a regression model based on Gaussian Processes, on top of the original NN classifier. Although the original RIO is limited to regression problems, the proposed approach manages to extend RIO for misclassification detection by modifying its components. The new framework not only produces a calibrated confidence score based on the original maximum class probability, but also provides a quantitative uncertainty estimation of that score. The reliability of error detection is therefore enhanced. It is notable that the proposed method does not change the prediction accuracy of the original classification model. Instead, it is a supporting tool that can provide a quantitative metric for detecting misclassification errors of the original classification model. The framework, referred to as RED (RIO for Error Detection), is compared empirically to existing approaches on 125 UCI datasets and on a large-scale deep learning architecture. The results demonstrate that the approach is effective and robust. A further case study with OOD and adversarial samples shows the potential of using RED to diagnose the sources of mistakes as well, thereby leading to a possible comprehensive approach for improving trustworthiness of neural network classifiers in the future.

2 Related Work

In the past two decades, a large volume of work was devoted to calibrating the confidence scores returned by classifiers. Early works include Platt Scaling (Platt, 1999; Niculescu-Mizil & Caruana, 2005), histogram binning (Zadrozny & Elkan, 2001), isotonic regression (Zadrozny & Elkan, 2002), with recent extensions like Temperature Scaling (Guo et al., 2017), Dirichlet calibration (Kull et al., 2019), and distance-based learning from errors (Xing et al., 2020). These methods focus on reducing the difference between reported class probability and true accuracy, and generally the rankings of samples are preserved after calibration. As a result, the separability between correct and incorrect predictions is not improved. In contrast, RED aims at deriving a score that can differentiate incorrect predictions from correct ones better.

A related direction of work is the development of classifiers with rejection/abstention option. These approaches either introduce new training pipelines/loss functions (Bartlett & Wegkamp, 2008; Yuan & Wegkamp, 2010; Cortes et al., 2016), or define mechanisms for learning rejection thresholds under certain risk levels (Dubuisson & Masson, 1993; Santos-Pereira & Pires, 2005; Chow, 2006; Geifman & El-Yaniv, 2017). In contrast to these methods, RED assumes an existing pretrained NN classifier, and provides an additional metric for detecting potential errors made by this classifier, without specifying a rejection threshold.

Designing metrics for detecting potential risks in NN classifiers has also become popular recently. While most approaches focus on detecting OOD (Liang et al., 2018; Lee et al., 2018a; Devries & Taylor, 2018) or adversarial examples (Lee et al., 2018b; Wang et al., 2019; Aigrain & Detyniecki, 2019), work on detecting natural errors, i.e., regular misclassifications not caused by external sources, is more limited. Ortega (1995) and Koppel & Engelson (1996) conducted early work in predicting whether a classifier is going to make mistakes, and Seewald & Fürnkranz (2001) built a meta-grading classifier based on similar ideas. However, these early works did not consider NN classifiers. More recently, Hendrycks & Gimpel (2017) and Haldimann et al. (2019) demonstrated that raw maximum class probability is an effective baseline in error detection, although its performance was reduced in some scenaria.

More elaborate techniques for error detection have also been developed recently. Mandelbaum & Weinshall (2017) proposed a confidence score based on the data embedding derived from the penultimate layer of a NN. However, their approach requires modifying the training procedure in order to achieve effective embeddings. Jiang et al. (2018) introduced Trust Score to measure the similarity between a base classifier and a modified nearest-neighbor classifier. Trust Score outperforms the maximum class probability baseline in many cases, but negative improvement over baseline can be observed in high-dimensional problems, implying poor scalability of local distance computations. ConfidNet (Corbière et al., 2019) learns to predict the class probability of true class with another NN, while Introspection-Net (Aigrain & Detyniecki, 2019) utilizes the logit activations of the original NN classifier to predict its correctness. Since both models themselves are standard NNs, the confidence scores returned by them may be arbitrarily high without any uncertainty information. Moreover, existing approaches for error detection cannot differentiate natural misclassification error from OOD or adversarial samples, making it difficult to diagnose the sources of risks. In contrast, RED explicitly reports its uncertainty about the estimated confidence score, providing more reliable error detection. The uncertainty information returned by RED may also be helpful in clarifying the cause of classifier mistakes, as will be demonstrated in this paper.

3 METHODOLOGY

This section gives the general problem statement, introduces the basic idea of original RIO, on which RED is built, and describes the technical details of RED.

3.1 PROBLEM STATEMENT

Consider a training dataset $\mathcal{D} = (\mathcal{X}, \mathbf{y}) = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, and a pretrained NN classifier that outputs a predicted label $\hat{y}_i$ and class probabilities for each class $\sigma_i = [\hat{p}_{i,1}, \hat{p}_{i,2}, \ldots, \hat{p}_{i,K}]$ given $\mathbf{x}_i$, where N is the total number of training points and K is the total number of classes. The problem is to develop a metric that can serve as a quantitative indicator for detecting natural misclassification errors made by the pretrained NN classifier.

3.2 RIO

The original RIO (Qiu et al., 2020) was developed to quantify point-prediction uncertainty in regression models. More specifically, RIO fits a Gaussian Process (GP) to predict the residuals, i.e. the differences between ground-truth and original model predictions. It utilizes an I/O kernel, i.e. a composite of an input kernel and an output kernel, thus taking into account both inputs and outputs of the original regression model. As a result, it measures the covariances between data points in both the original feature space and the original model output space. For each new data point, a trained RIO model takes the original input and output of the base regression model, and predicts a distribution of the residual, which can be added back to the original model prediction to obtain both a calibrated prediction and the corresponding predictive uncertainty.

In the original RIO work, SVGP (Hensman et al., 2013; 2015) was used as an approximate GP to improve the scalability of the approach. Both empirical results and theoretical analysis showed that RIO is able to consistently improve the prediction accuracy of base model as well as provide reliable uncertainty estimation. Moreover, RIO can be directly applied on top of any pre-trained models without retraining or modification. It therefore forms a promising foundation for improving reliability of error detection metrics as well.

3.3 RIO FOR ERROR DETECTION (RED)

Although RIO performs robustly in a wide variety of regression problems, it cannot be directly applied to classification models. A new framework, namely RED, is proposed to utilize RIO for error detection in classification domains.

Building on the fact that the original maximum class probability is a strong baseline for error detection (Hendrycks & Gimpel, 2017; Haldimann et al., 2019), the main idea of RED is to derive a more reliable confidence score by stacking RIO on top of the original maximum class probability. Since RIO was designed for single-output regression problems, it contains an output kernel only for scalar

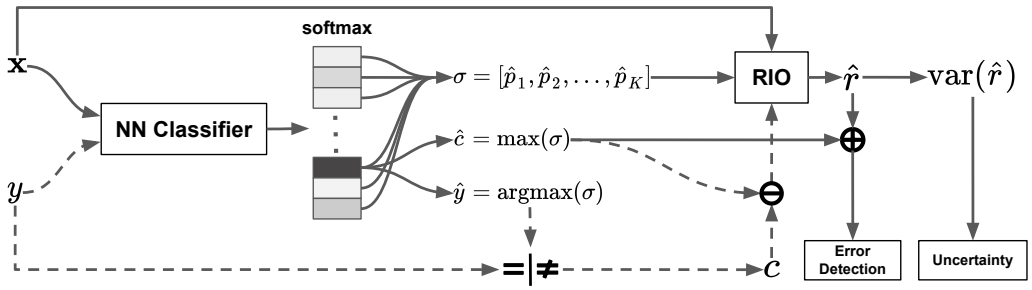

Figure 1: **The RED Training and Deployment Processes.** The solid pathways are active in both training and deployment phase, while the dashed pathways are active only in the training phase. During the training phase, a target confidence score c is assigned to each training sample according to whether it is correctly predicted by the original NN classifier or not. A RIO model is then trained to predict the residual between the target confidence score c and the original maximum class probability $\hat{c}$. The I/O kernel in RIO utilizes both the raw feature $\mathbf{x}$ and softmax outputs σ to predict the residuals. In the deployment phase, given a new data point, the trained RIO model provides a Gaussian distribution of estimated residual $\hat{r}$ defined by the mean $\bar{\hat{r}}$ and variance $\mathrm{var}(\hat{r})$. Addition of $\hat{r}$ and $\hat{c}$ forms a calibrated confidence score for error detection, and $\mathrm{var}(\hat{r})$ indicates the corresponding uncertainty.

Algorithm 1 RED Training and Deployment Procedures

Require:

$(\mathcal{X}, \mathbf{y}) = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$: training data

$\hat{\mathbf{y}} = \{\hat{y}_i\}_{i=1}^N$: labels predicted by original NN classifier on training data

$\boldsymbol{\sigma} = \{\sigma_i = [\hat{p}_{i,1}, \hat{p}_{i,2}, \ldots, \hat{p}_{i,K}]\}_{i=1}^N$: softmax outputs of original NN classifier on training data

$\hat{\mathbf{c}} = \{\hat{c}_i = \max(\sigma_i)\}_{i=1}^N$: maximum class probability returned by original NN classifier on training data

$\mathbf{x}_*$: data to be predicted

σ_*: softmax outputs of original NN classifier on $\mathbf{x}_*$

$\hat{c}_*$: maximum class probability returned by original NN classifier on $\mathbf{x}_*$

Ensure:

$\hat{c}'_* \sim \mathcal{N}(\hat{c}_* + \bar{\hat{r}}_*, \mathrm{var}(\hat{r}_*))$: $\hat{c}_* + \bar{\hat{r}}_*$ can be used as confidence score for error detection, and $\mathrm{var}(\hat{r}_*)$ represents the uncertainty of returned confidence score

Training Phase:

1: obtain target confidence score $\mathbf{c} = \{c_i = \delta_{y_i,\hat{y}_i}\}_{i=1}^N$, where $\delta_{y_i,\hat{y}_i}$ is the Kronecker delta ($\delta_{y_i,\hat{y}_i} = 1$ if $y_i = \hat{y}_i$, otherwise $\delta_{y_i,\hat{y}_i} = 0$)

2: calculate residuals $\mathbf{r} = \{r_i = c_i - \hat{c}_i\}_{i=1}^N$

3: **for** each optimizer step **do**

4: calculate covariance matrix $\mathbf{K}_c((\mathcal{X}, \boldsymbol{\sigma}), (\mathcal{X}, \boldsymbol{\sigma}))$, where each entry is given by $k_c((\mathbf{x}_i, \sigma_i), (\mathbf{x}_j, \sigma_j)) = k_{\mathrm{in}}(\mathbf{x}_i, \mathbf{x}_j) + k_{\mathrm{out}}(\sigma_i, \sigma_j)$, for $i, j = 1, 2, \ldots, N$

5: optimize GP hyperparameters by maximizing log marginal likelihood $\log p(\mathbf{r}|\mathcal{X}, \boldsymbol{\sigma}) = -\frac{1}{2}\mathbf{r}^\top(\mathbf{K}_c((\mathcal{X}, \boldsymbol{\sigma}), (\mathcal{X}, \boldsymbol{\sigma})) + \sigma_n^2\mathbf{I})^{-1}\mathbf{r} - \frac{1}{2}\log|\mathbf{K}_c((\mathcal{X}, \boldsymbol{\sigma}), (\mathcal{X}, \boldsymbol{\sigma})) + \sigma_n^2\mathbf{I}| - \frac{n}{2}\log 2\pi$

Deployment Phase:

6: calculate residual mean $\bar{\hat{r}}_* = \mathbf{k}_*^\top(\mathbf{K}_c((\mathcal{X}, \boldsymbol{\sigma}), (\mathcal{X}, \boldsymbol{\sigma})) + \sigma_n^2\mathbf{I})^{-1}\mathbf{r}$ and residual variance $\mathrm{var}(\hat{r}_*) = k_c((\mathbf{x}_*, \sigma_*), (\mathbf{x}_*, \sigma_*)) - \mathbf{k}_*^\top(\mathbf{K}_c((\mathcal{X}, \boldsymbol{\sigma}), (\mathcal{X}, \boldsymbol{\sigma})) + \sigma_n^2\mathbf{I})^{-1}\mathbf{k}_*$, where $\mathbf{k}_*$ denotes the vector of kernel-based covariances (i.e., $k_c(\mathbf{x}_*, \mathbf{x}_i)$) between $\mathbf{x}_*$ and all the training points

7: return distribution of calibrated confidence score $\hat{c}'_* \sim \mathcal{N}(\hat{c}_* + \bar{\hat{r}}_*, \mathrm{var}(\hat{r}_*))$

outputs. In RED, this original output kernel is extended to multiple outputs, i.e. to vector outputs such as those of the final softmax layer of a NN classifer, representing estimated class probabilities for each class. This modification allows RIO to access more information from the classifier outputs.

The targets for RIO training need to be redesigned as well. The raw targets of a classification problem are the ground-truth labels; they are in categorical space, while RIO works in continuous space. To solve this issue, RED constructs a different problem: Instead of predicting the labels

directly, it predicts whether the original prediction is correct or not. A target confidence score is assigned to each training data point accordingly. The residual between this target confidence score and the originally returned maximum class probability is calculated, and a RIO model is trained to predict these residuals. Given a new data point, the trained RIO model combined with the original NN classifier thus provides a calibrated confidence score for detecting misclassification errors.

Figure 1 illustrates the RED training and deployment processes conceptually, and Algorithm 1 specifies them in detail. In the training phase, the first step is to define a target confidence score c_i for each training sample $(\mathbf{x}_i, y_i)$. For simplicity, all training samples that are correctly predicted by the original NN classifier receive 1 as the target confidence score, and those that are incorrectly predicted receive 0. The validation dataset during the original NN training is included in the training dataset for RED. After the target confidence scores are assigned, a regression problem is formulated for the RIO model: Given the original raw features $\{\mathbf{x}_i\}_{i=1}^N$ and the corresponding softmax outputs of the original NN classifier $\{\sigma_i = [\hat{p}_{i,1}, \hat{p}_{i,2}, \ldots, \hat{p}_{i,K}]\}_{i=1}^N$, predict the residuals $\mathbf{r} = \{r_i = c_i - \hat{c}_i\}_{i=1}^n$ between target confidence scores $\mathbf{c} = \{c_i\}_{i=1}^N$ and the original maximum class probabilities $\hat{\mathbf{c}} = \{\hat{c}_i = \max(\sigma_i)\}_{i=1}^N$.

The RIO model relies on an I/O kernel consisting of two components: the input kernel $k_{\text{in}}(\mathbf{x}_i, \mathbf{x}_j)$, which measures covariances in the raw feature space, and the modified multi-output kernel $k_{\text{out}}(\sigma_i, \sigma_j)$, which calculates covariances in the softmax output space. The hyperparameters of the I/O kernel are optimized to maximize the log marginal likelihood $\log p(\mathbf{r}|\mathcal{X}, \boldsymbol{\sigma})$. In the deployment phase, given a new data point $\mathbf{x}_*$, the trained RIO model provides a Gaussian distribution for the estimated residual $\hat{r}_* \sim \mathcal{N}(\bar{\hat{r}}_*, \text{var}(\hat{r}_*))$. By adding the estimated residual back to the original maximum class probability $\hat{c}_*$, a distribution of calibrated confidence score is obtained as $\hat{c}'_* \sim \mathcal{N}(\hat{c}_* + \bar{\hat{r}}_*, \text{var}(\hat{r}_*))$. The mean $\hat{c}_* + \bar{\hat{r}}_*$ can be directly used as a quantitative metric for error detection, and the variance $\text{var}(\hat{r}_*)$ represents the corresponding uncertainty of the confidence score.

4 Empirical Evaluation

In this section, the error detection performance of RED is evaluated comprehensively on 125 UCI datasets, comparing it to other related methods. Its scale-up properties are then evaluated in a large-scale deep learning architecture, and its potential to improve robustness more broadly in a case study involving OOD and adversarial samples.

4.1 Experiments on UCI datasets

As a comprehensive evaluation of RED, an empirical comparison with seven related approaches on 125 UCI datasets (Dua & Graff, 2017) was performed. The reference approaches were maximum class probability (MCP) baseline (Hendrycks & Gimpel, 2017), Trust Score (Jiang et al., 2018), ConfidNet (Corbière et al., 2019), Introspection-Net (Aigrain & Detyniecki, 2019), entropy of the original softmax outputs (Steinhardt & Liang, 2016), DNGO (Snoek et al., 2015), and the original SVGP (Hensman et al., 2013; 2015). The 125 UCI datasets include 121 datasets used by Klambauer et al. (2017) and four more recent ones. Full details about the datasets and parametric setups of all tested algorithms, and a downloadable link for source codes are provided in Appendix A.1.

Following the experimental setup of Hendrycks & Gimpel (2017); Corbière et al. (2019); Aigrain & Detyniecki (2019), the task for each algorithm is to provide a confidence score for each testing point. An error detector can then use a predefined fixed threshold on this score to decide which points are probably misclassified by the original NN classifier. For RED, the mean of calibrated confidence score $\hat{c}_* + \bar{\hat{r}}_*$ was used as the reported confidence score.

Five threshold-independent performance metrics were used to compare the methods: AUPR-Error, which computes the area under the Precision-Recall (AUPR) Curve when treating incorrect predictions as positive class during the detection; AUPR-Success, which is similar to AUPR-Error but uses correct predictions as positive class; AUROC, which computes the area under receiver operating characteristic (ROC) curve for the error detection task; AP-Error, which computes the average precision (AP) under different thresholds treating incorrect predictions as positive class; and AP-Success, which is similar to AP-Error but uses correct predictions as positive class. AUPR and AUROC are commonly used in prior work (Hendrycks & Gimpel, 2017; Corbière et al., 2019; Aigrain & Detyniecki, 2019), but as discussed by Davis & Goadrich (2006) and Flach & Kull (2015), AUPR may

Table 1: Mean Rank on UCI datasets

Method	AP-Error mean±std	AUPR-Error mean±std	AP-Success mean±std	AUPR-Success mean±std	AUROC mean±std
RED	**1.39±0.61***	**1.49±0.78***	**1.74±0.97***	**1.80±1.03***	**1.65±0.82***
MCP Baseline	2.93± 0.89	3.06±0.92	2.77±1.07	2.75±1.11	2.80±1.08
Trust Score	3.92±2.45	3.86±2.50	3.64±2.25	3.61±2.25	3.76±2.31
ConfidNet	6.13±1.37	6.33±1.38	6.07±1.51	6.07±1.41	5.97±1.45
Introspection-Net	5.34±1.65	5.38±1.65	5.83±1.46	5.89±1.51	5.71±1.50
Entropy	3.47±1.08	3.59±1.19	3.19±1.26	3.23±1.32	3.26±1.28
DNGO	6.19±1.51	5.46±1.82	6.84±1.33	6.80±1.44	6.57±1.47
SVGP	6.59±1.60	6.80±1.49	5.89±1.54	5.83±1.49	6.24±1.61

The symbols * indicates that the differences between the marked entry and all other counterparts are statistically significant at the 5% significance level for both paired t-test and Wilcoxon test. The best entries that are significantly better than all the others under both statistical test are marked in boldface.

Table 2: A Pairwise Comparison between RED and Other Methods on UCI datasets

Method	AP-Error + / = / -	AUPR-Error + / = / -	AP-Success + / = / -	AUPR-Success + / = / -	AUROC + / = / -
MCP Baseline	87 / 35 / 0	90 / 32 / 0	58 / 63 / 1	56 / 65 / 1	61 / 60 / 1
Trust Score	53 / 44 / 16	49 / 47 / 17	50 / 47 / 16	48 / 49 / 16	59 / 37 / 17
ConfidNet	100 / 22 / 0	100 / 22 / 0	106 / 16 / 0	106 / 16 / 0	109 / 13 / 0
Introspection-Net	93 / 29 / 0	90 / 32 / 0	98 / 24 / 0	98 / 24 / 0	101 / 21 / 0
Entropy	74 / 47 / 1	75 / 46 / 1	53 / 68 / 1	53 / 68 / 1	52 / 69 / 1
DNGO	92 / 17 / 0	73 / 31 / 5	99 / 10 / 0	97 / 12 / 0	98 / 11 / 0
SVGP	98 / 23 / 1	98 / 23 / 1	97 / 25 / 0	97 / 25 / 0	102 / 19 / 1

The columns labeled + show the number of datasets on which RED performs significantly better at the 5% significance level in a paired t-test, Wilcoxon test, or both; those labeled - represent the contrary case; those labeled = represent no statistical significance.

provide overly-optimistic measurement of performance. To compensate for this issue, AP-Error and AP-Success are included as additional metrics. Since the target for the confidence metrics is to detect misclassification errors, the following discussion will focus more on AP-Error and AUPR-Error.

Ten independent runs were conducted for each dataset. During each run, the dataset was randomly split into training dataset and testing dataset, and a standard NN classifier trained and evaluated on them. The same dataset split and trained NN classifier was used to evaluate all methods. Full details of the experimental setup are provided in Appendix A.1.

Table 1 shows the ranks of each algorithm averaged over all 125 UCI datasets. The rank of each algorithm on each dataset is based on the average performance over 10 independent runs. RED performs best on all metrics; the performance differences between RED and all other methods are statistically significant under paired t-test and Wilcoxon test. Trust Score has the highest standard deviation, suggesting that its performance varies significantly across different datasets.

As a more detailed comparison, Table 2 shows how often RED performs statistically significantly better, how often the performance is not significantly different, and how often it performs significantly worse than the other methods. RED is most often significantly better, and very rarely worse. In a handful of datasets Trust Score is better, but most often it is not.

To further illustrate the performance of RED and four state-of-the-art approaches, Figure 2 shows the distribution of the relative rank of RED, MCP baseline, Trust Score, ConfidNet and Introspection-Net as a function of the number of samples and the number of features in the dataset. These plots are based on the AP-Error metric; other metrics provide similar results. RED performs consistently well over different dataset sizes and feature dimensionalities. Trust Score performs best in several datasets, but occasionally also worst in both small and large datasets, making it a rather unreliable choice. ConfidNet generally exhibits worse performance on datasets with large dataset sizes and high feature dimensionalities, i.e. it does not scale well to larger problems.

To investigate the robustness of RED when applying to other types of base models. A NN classifier using MC-dropout technique (Gal & Ghahramani, 2016) and a Bayesian Neural Network (BNN) classifier (Wen et al., 2018) are trained as the base classifiers, RED is then applied to each of them. Same experiments as in Table 2 are performed on 125 UCI datasets for both MC-dropout and BNN classifiers. More implementation details are provided in Appendix A.1. Table 3 shows the pairwise

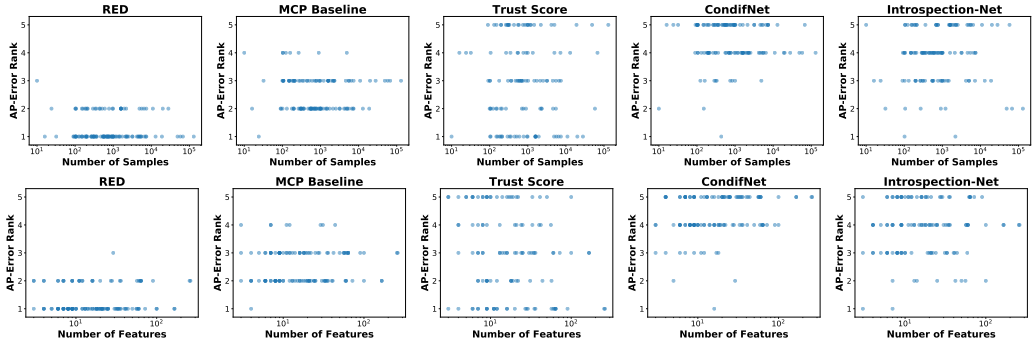

Figure 2: **Performance Ranks Across Dataset Sizes and Dimensionalities on UCI Datasets.** Each plot represents the distribution of relative ranks for one method (each column) as a function of the dataset size (top row) and the feature dimensionality (bottom row). Each dot in each plot represents the relative rank in one dataset. RED performs consistently well over datasets of different sizes and dimensionalities. Trust Score performs inconsistently, and ConfidNet performs poorly on larger datasets.

Table 3: A Pairwise Comparison between RED and Other Methods on UCI datasets

Method	AP-Error + / = / -	AUPR-Error + / = / -	AP-Success + / = / -	AUPR-Success + / = / -	AUROC + / = / -
BNN MCP	102 / 20 / 0	104 / 18 / 0	95 / 26 / 1	88 / 33 / 1	95 / 26 / 1
BNN Entropy	67 / 53 / 2	68 / 52 / 2	48 / 66 / 8	48 / 66 / 8	53 / 64 / 5
MC-Dropout MCP	87 / 35 / 0	88 / 34 / 0	70 / 52 / 0	67 / 55 / 0	71 / 51 / 0
MC-Dropout Entropy	54 / 68 / 0	55 / 67 / 0	38 / 77 / 7	38 / 76 / 8	42 / 74 / 6

The columns labeled + show the number of datasets on which RED performs significantly better at the 5% significance level in a paired t-test, Wilcoxon test, or both; those labeled - represent the contrary case; those labeled = represent no statistical significance.

comparisons between RED and the internal confidence scores returned by the base models. MCP and Entropy represent the maximum class probability and entropy of softmax outputs, respectively, after averaging over 100 test-time samplings. According to the results, RED significantly improves over both MC-dropout and BNN classifier in most datasets.

4.2 Scaling up to Larger Architectures

To confirm that the RED approach scales up to large deep learning architectures, a VGG16 model (Simonyan & Zisserman, 2015) was trained on the CIFAR-10 dataset using a state-of-the-art training pipeline (see Appendix A.2 for details). In order to remove the influence of feature extraction in image preprocessing and to make the comparison fair, all approaches used the same logit outputs of the trained VGG16 model as their input features. 10 independent runs are performed. During each run, a VGG16 model is trained, and all the methods are evaluated based on this VGG16 model.

Table 4 shows the results on the two main error detection performance metrics. It is notable that the numbers in Table 4 are the absolute values of the corresponding metrics. Trust Score performs much better than in previous literatures (Corbière et al., 2019). This difference may be due to the fact that logit outputs are used as input features here, whereas Corbière et al. (2019) utilized a higher dimensional feature space for Trust Score. Based on the results, RED significantly outperforms all the counterparts in both metrics. This result demonstrates the advantages of RED in scaling up to larger architectures.

4.3 A Case Study with OOD and Adversarial Samples

In all experiments so far, the mean of calibrated confidence score $\hat{c}_* + \bar{\hat{r}}_*$ was used as RED's confidence score. Although good performance is observed in error detection by only using the mean, the variance of calibrated confidence score $\mathrm{var}(\hat{r}_*)$ may be helpful if the scenario is more complex, e.g., the dataset includes some OOD data, or even adversarial data.

Table 4: A Comparison based on the VGG16 Network Architecture on the CIFAR-10 Task

Metric	RED mean±std	MCP Baseline mean±std	Trust Score mean±std	ConfidNet mean±std	Introspection-Net mean±std	Entropy mean±std	DNGO mean±std	SVGP mean±std
AP-Error(%)	**49.88±1.99***	47.09±2.19	48.76±2.28	45.80±2.85	42.11±1.98	47.91±2.17	33.91±2.94	40.71±2.33
AUPR-Error(%)	**49.79±2.00***	46.99±2.21	48.68±2.29	45.70±2.86	42.01±1.98	47.81±2.19	34.43±2.92	40.60±2.34

The symbols * indicates that the differences between the marked entry and all other counterparts are statistically significant at the 5% significance level for both paired t-test and Wilcoxon test. The best entries that are significantly better than all the others under both statistical test are marked in boldface.

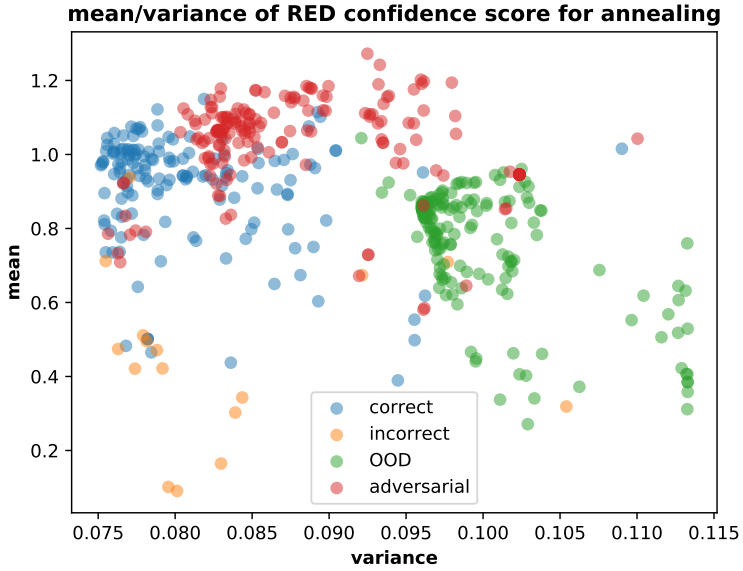

Figure 3: **Identifying OOD and Adversarial Samples Based on Mean and Variance of Confidence Scores.** Each dot represents one sample in the testing set in the UCI Annealing task. The horizontal axis denotes the variance of RED-calibrated confidence score, and the vertical axis denotes the mean. If an in-distribution sample is correctly classified by original NN classifier, it is marked as "correct", otherwise it is marked "incorrect". Mean is a good separator of correct and incorrect classifications. High variance, on the other hand, indicates that RED is uncertain about its confidence score, which can be used to identify OOD and adversarial samples. In this manner, RED can serve as a foundation for improving robustness of classifiers more broadly in the future.

A preliminary investigation of RED in such scenaria was performed by manually adding OOD and adversarial data into the test set of the UCI "annealing" dataset, on which RED detected errors well. The synthetic OOD and adversarial samples were created to be highly deceptive, aiming to evaluate the performance of RED under difficult circumstances. The OOD data were sampled from a Gaussian distribution with mean 0 and variance 1, and the number of added OOD data was the same as the number of samples in the original test set. Note that all data points from original dataset are normalized to have mean 0 and variance 1 for each feature dimension during preprocessing, so the OOD data and in-distribution data have similar scales. The adversarial data was created by adding negligible modifications to training samples that the original NN classifier predicted incorrectly with highest confidence. This process mimics an adversary that can arbitrarily alter the output of the NN classifier with minuscule changes to the input (Goodfellow et al., 2014).

Figure 3 shows the distribution of mean and variance of calibrated confidence scores for testing samples, including correctly and incorrectly labeled actual samples, as well as the synthetic OOD and adversarial samples. The mean is a good separator for correctly classified and incorrectly classified samples, which tend to cluster on the top and bottom half of the image, respectively. On the other hand, variance is a promising indicator of OOD and adversarial samples. RED's confidence scores of in-distribution samples have low variance because they covary with the training samples. The variance thus represents RED's confidence in its confidence score. Samples with large variance indicate that RED is uncertain about its confidence score, which can be used as a basis for detecting OOD and adversarial samples.

Although the main usage of RED is on misclassification detection, the preliminary results in this subsection show the potential of extending RED for detecting broader error types.

5 Discussion and Future Work

One interesting observation from the experiments is that RED almost never performs worse than the MCP baseline. This result suggests that there is almost no risk in applying RED on top of an existing NN classifier. Since RED is based on a GP model, the estimated residual $\bar{\hat{r}}_*$ is close to zero if the predicted sample is far from the distribution of the original training samples, resulting in no change to the original MCP. In other words, RED does not make random changes to original MCP if it is very uncertain about the predicted sample, and this uncertainty is explicitly represented in the variance of the estimated confidence score. This property makes RED a particularly reliable technique for error detection.

Another interesting observation is that the variance is also helpful in detecting OOD and adversarial samples. This result follows from the design of the RIO uncertainty model. Since RIO in RED has an input kernel and an output kernel, lower estimated variance requires that the predicted sample is close to training samples in both the input feature space and the classifier output space. This requirement is difficult for OOD and adversarial attacks to achieve, providing a basis for detecting them.

In real-world deployment, a threshold is required to be pre-defined for triggering error warnings based on confidence scores returned by RED. A practical way is to use a validation dataset for investigating how the tradeoffs between precision and recall rate in terms of error detection change over different thresholds. The users can then decide the threshold based on their preference on the precision-recall tradeoff.

The most compelling direction of future work is to extend this capability of RED further. Instead of using a single dimensional confidence score for error detection, it is possible to use mean and variance simultaneously, leading to a two dimensional detection space. Further separation between OOD and adversarial samples may be possible by adding one more dimension, such as the ratio between input kernel output and output kernel output. Also, instead of using a hard target confidence score (i.e. either 0 or 1), it may be possible to define a soft target confidence that may be more informative. Further, RED may be used to calibrate other existing confidence metrics, such as the Trust Score, which may lead to a further improvement in detection performance.

6 Conclusion

This paper introduced a new framework, RED, for error detection in neural network classifiers that can produce a more reliable confidence score than previous methods. RED is able to not only provide a calibrated confidence score, but also report the uncertainty of the estimated confidence score. Experimental results show that RED's scores consistently outperform state-of-the-art methods in separating the misclassified samples from correctly classified samples. Preliminary experiments also demonstrate that the approach scales up to large deep learning architectures, and can form a basis for detecting OOD and adversarial samples as well. It is therefore a promising foundation for improving robustness of neural network classifiers.

References

Jonathan Aigrain and Marcin Detyniecki. Detecting adversarial examples and other misclassifications in neural networks by introspection. *CoRR*, abs/1905.09186, 2019. URL `http://arxiv.org/abs/1905.09186`.

Ahmed Alghoul, Sara Al Ajrami, Ghada Al Jarousha, Ghayda Harb, and Samy S. Abu-Naser. Email classification using artificial neural network. *International Journal of Academic Engineering Research (IJAER)*, 2(11):8–14, 2018.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Man. Concrete problems in ai safety. 06 2016.

Ofélia Anjos, Carla Iglesias, Fátima Peres, Javier Martínez, Ángela García, and Javier Taboada. Neural networks applied to discriminate botanical origin of honeys. *Food Chemistry*, 175:128 – 136, 2015. ISSN 0308-8146. doi: https://doi.org/10.1016/j.foodchem.2014.11.121. URL http://www.sciencedirect.com/science/article/pii/S0308814614018597.

Peter L. Bartlett and Marten H. Wegkamp. Classification with a reject option using a hinge loss. *J. Mach. Learn. Res.*, 9:18231840, June 2008. ISSN 1532-4435.

C. Chow. On optimum recognition error and reject tradeoff. *IEEE Trans. Inf. Theor.*, 16(1):4146, September 2006. ISSN 0018-9448. doi: 10.1109/TIT.1970.1054406. URL https://doi.org/10.1109/TIT.1970.1054406.

Charles Corbière, Nicolas THOME, Avner Bar-Hen, Matthieu Cord, and Patrick Pérez. Addressing failure prediction by learning model confidence. In *Advances in Neural Information Processing Systems 32*, pp. 2902–2913. Curran Associates, Inc., 2019.

Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In *Algorithmic Learning Theory*, pp. 67–82, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46379-7.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. volume 06, 06 2006. doi: 10.1145/1143844.1143874.

Terrance Devries and Graham W. Taylor. Learning confidence for out-of-distribution detection in neural networks. *ArXiv*, abs/1802.04865, 2018.

Matthew Dixon, Diego Klabjan, and Jin Bang. Classification-based financial markets prediction using deep neural networks. *Algorithmic Finance*, 12 2017. doi: 10.2139/ssrn.2756331.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Bernard Dubuisson and Mylne Masson. A statistical decision rule with incomplete knowledge about classes. *Pattern Recognition*, 26(1):155 – 165, 1993. ISSN 0031-3203. doi: https://doi.org/10.1016/0031-3203(93)90097-G. URL http://www.sciencedirect.com/science/article/pii/003132039390097G.

Peter Flach and Meelis Kull. Precision-recall-gain curves: Pr analysis done right. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 28*, pp. 838–846. Curran Associates, Inc., 2015.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pp. 1050–1059. JMLR.org, 2016. URL http://dl.acm.org/citation.cfm?id=3045390.3045502.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 48854894, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv 1412.6572*, 12 2014.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 13211330. JMLR.org, 2017.

Cota Navin Gupta, Ramaswamy Palaniappan, Sundaram Swaminathan, and Shankar M. Krishnan. Neural network classification of homomorphic segmented heart sounds. *Applied Soft Computing*, 7(1):286 – 297, 2007. ISSN 1568-4946. doi: https://doi.org/10.1016/j.asoc.2005.06.006. URL http://www.sciencedirect.com/science/article/pii/S1568494605000694.

David Haldimann, Hermann Blum, R. Siegwart, and C. Cadena. This is not what I imagined: Error detection for semantic segmentation through visual dissimilarity. *ArXiv*, abs/1909.00676, 2019.

Simon Hecker, Dengxin Dai, and Luc Van Gool. Failure prediction for autonomous driving, 05 2018.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*, 2017.

James Hensman, Nicolò Fusi, and Neil D. Lawrence. Gaussian processes for big data. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, UAI'13, pp. 282–290, Arlington, Virginia, United States, 2013. AUAI Press. URL http://dl.acm.org/citation.cfm?id=3023638.3023667.

James Hensman, Alexander Matthews, and Zoubin Ghahramani. Scalable Variational Gaussian Process Classification. In Guy Lebanon and S. V. N. Vishwanathan (eds.), *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pp. 351–360, San Diego, California, USA, 09–12 May 2015. PMLR. URL http://proceedings.mlr.press/v38/hensman15.html.

Joel Janai, Fatma Güney, A. Behl, and Andreas Geiger. Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art. *ArXiv*, abs/1704.05519, 2017.

Heinrich Jiang, Been Kim, and Maya Gupta. To trust or not to trust a classifier, 05 2018.

Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 972981, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Moshe Koppel and Sean P. Engelson. Integrating multiple classifiers by finding their areas of expertise. In *In: AAAI-96 Workshop On Integrating Multiple Learning Models*, pp. 53–58, 1996.

Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. In *Advances in Neural Information Processing Systems 32*, pp. 12316–12326. Curran Associates, Inc., 2019.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521:436 EP –, 05 2015. URL https://doi.org/10.1038/nature14539.

Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*, 2018a. URL https://openreview.net/forum?id=ryiAv2xAZ.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 7167–7177. Curran Associates, Inc., 2018b.

Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1VGkIxRZ.

Amit Mandelbaum and D. Weinshall. Distance-based confidence score for neural network classifiers. *ArXiv*, abs/1709.09844, 2017.

Kristine Monteith and Tony R. Martinez. Using multiple measures to predict confidence in instance classification. *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2010.

Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. pp. 427–436, 06 2015. doi: 10.1109/CVPR.2015.7298640.

Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, pp. 625632, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595931805. doi: 10.1145/1102351.1102430. URL https://doi.org/10.1145/1102351.1102430.

Julio Ortega. Exploiting multiple existing models and learning algorithms. In *In AAAI 96 - Workshop in Induction of Multiple Learning Models*, pp. 239–266, 1995.

John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pp. 61–74. MIT Press, 1999.

Foster J. Provost, Tom Fawcett, and Ron Kohavi. The case against accuracy estimation for comparing induction algorithms. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, pp. 445453, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc. ISBN 1558605568.

Xin Qiu, Elliot Meyerson, and Risto Miikkulainen. Quantifying point-prediction uncertainty in neural networks via residual estimation with an i/o kernel. In *International Conference on Learning Representations*, 2020.

Carla M. Santos-Pereira and Ana M. Pires. On optimal reject rules and roc curves. *Pattern Recogn. Lett.*, 26(7):943952, May 2005. ISSN 0167-8655.

Alexander K. Seewald and Johannes Fürnkranz. An evaluation of grading classifiers. In *Proceedings of the 4th International Conference on Advances in Intelligent Data Analysis*, IDA '01, pp. 115124, Berlin, Heidelberg, 2001. Springer-Verlag. ISBN 3540425810.

Dan Selişteanu, Vladimir A. Maksimenko, Semen A. Kurkin, Elena N. Pitsik, Vyacheslav Yu. Musatov, Anastasia E. Runnova, Tatyana Yu. Efremova, Alexander E. Hramov, and Alexander N. Pisarchik. Artificial neural network classification of motor-related eeg: An increase in classification accuracy by reducing signal complexity. *Complexity*, 2018:9385947, 2018. doi: 10.1155/2018/9385947. URL https://doi.org/10.1155/2018/9385947.

Nida Shahid, Tim Rappon, and Whitney Berta. Applications of artificial neural networks in health care organizational decision-making: A scoping review. *PLOS ONE*, 14(2):1–22, 02 2019. doi: 10.1371/journal.pone.0212356. URL https://doi.org/10.1371/journal.pone.0212356.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Jasper Snoek, Oren Rippel, Kevin Swersky, Ryan Kiros, Nadathur Satish, Narayanan Sundaram, Md. Mostofa Ali Patwary, Prabhat Prabhat, and Ryan P. Adams. Scalable bayesian optimization using deep neural networks. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pp. 21712180. JMLR.org, 2015.

Jacob Steinhardt and Percy Liang. Unsupervised risk estimation using only conditional independence structure. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pp. 36643672, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

Nalika Ulapane, Karthick Thiyagarajan, and Sarath Kodagoda. Hyper-parameter initialization for squared exponential kernel-based gaussian process regression, 03 2020.

J. Wang, G. Dong, J. Sun, X. Wang, and P. Zhang. Adversarial sample detection for deep neural network through model mutation testing. In *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, pp. 1245–1256, 2019.

Yeming Wen, Paul Vicol, Jimmy Ba, Dustin Tran, and Roger Grosse. Flipout: Efficient pseudo-independent weight perturbations on mini-batches. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJNpifWAb.

Gethin Williams and Steve Renals. Confidence measures for hybrid hmm/ann speech recognition. In *In Proceedings of EuroSpeech*, pp. 1955–1958, 1997.

Chen Xing, Sercan Arik, Zizhao Zhang, and Tomas Pfister. Distance-based learning from errors for confidence calibration. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BJeB5hVtvB.

Ming Yuan and Marten Wegkamp. Classification methods with reject option based on convex risk minimization. *Journal of Machine Learning Research*, 11(5):111–130, 2010. URL http://jmlr.org/papers/v11/yuan10a.html.

Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pp. 609616, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1558607781.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pp. 694699, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 158113567X. doi: 10.1145/775047.775151. URL https://doi.org/10.1145/775047.775151.

A Appendix

A.1 Experimental Setups for Section 4.1

General Information 10 indenpendent runs are performed for each dataset. During each run, the dataset is randomly split into a training set (80%) and a testing set (20%), then a fully connected feed-forward NN classifier with 2 hidden layers, each with 64 hidden neurons, are trained on the training set. The activation function is ReLU for all the hidden layers. The maximum number of epochs for training is 1000. 20% of the training set is used as validation set, and the split is random at each independent run. An early stop is triggered if the loss on validation set has not be improved for 10 epochs. The optimizer is Adam with learning rate 0.001, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. The loss function is cross entropy loss. During each independent run, the same random dataset split and trained base NN classifier is used for evaluating all algorithms. All source codes for reproducing the experimental results can be downloaded from (https://drive.google.com/drive/folders/1X5R6sEkjmucR7B4MvF-rY6QnCvos6a7n?usp=sharing).

Dataset Description In total, 125 UCI datasets are used in the experiments, among which 121 are from Klambauer et al. (2017), and 4 are recent datasets released in Dua & Graff (2017). All features in all datasets are normalized to have mean 0 and standard deviation 1. Full details regarding the number of samples N, number of features M, and number of classes K for each dataset are shown in Table 5.

Parametric Setup for Algorithms

- RED: SVGP (Hensman et al., 2013; 2015) is used as an approximator to original GP. The number of inducing points are 50. RBF kernel is used for both input and multi-output kernel. Automatic Relevance Determination (ARD) feature is turned on. The signal variances and length scales of all the kernels plus the noise variance are the trainable hyperparameters. The optimizer is L-BFGS-B with default parameters as in Scipy.optimize documentation (https://docs.scipy.org/doc/scipy/reference/optimize.minimize-lbfgsb.html), and the maximum number of iterations is set as 1000. The optimization process runs until the L-BFGS-B optimizer decides to stop. To overcome the sensitivity of GP optimization to initialization of the hyperparameters (Ulapane et al., 2020), 20 random initialization of the

hyperparameters are tried for each independent run. For each random initialization, the signal variances are generated from a uniform distribution within interval $[0, 1]$, and the lengthscales are generated from a uniform distribution within interval $[0, 10]$. For 10 initializations, the hyperparameters of input kernel are first optimized while the multi-output kernel is temporarily turned off, then after the optimizer stops, the multi-output kernel is turned on, and both the two kernels are optimized simultaneously. For the other 10 initializations, both kernels are optimized simultaneouly from the start. The average performance of the 3 best optimized model in terms of corresponding metrics are used as the final performance of RED on each independent run. During our preliminary investigation, several statistic metrics on training set is effective in picking the true best-performing model out of these 20 trials, e.g., the gap between average estimated confidence scores of correctly classified training samples and incorrectly classified training samples, the scale of optimized noise variance of SVGP model, the ratio between sum of signal variances and noise variance after optimization, etc. Since improving initialization and optimization of GP hyperparameters is out of the scope of this work, we simply use average performance of the best 3 models (top 15%) in comparison.

- MCP baseline: the maximum class probability of softmax outputs of the base NN classifier is used as the confidence score of MCP baseline. Setups of the base NN classifier is provided above.
- Trust Score: k=10, $\alpha = 0$, without filtering. This is the same as the default setup in https://github.com/google/TrustScore.
- ConfidNet: During training, the input to ConfidNet is the raw feature, and the target is the class probability of the ground-truth class returned by base NN classifier. The architecture of ConfidNet is a fully connected feed-forward NN regressor with 2 hidden layers, each with 64 hidden neurons. The activation function is ReLU for all the hidden layers. The maximum number of epochs for training is 1000. An early stop is triggered if the loss on validation data has not be improved for 10 epochs. The optimizer is RMSprop with learning rate 0.001, and the loss function is mean squared error (MSE).
- Introspection-Net: During training, the input to Introspection-Net is the logit outputs of base NN classifier, and the target is 1 for correctly classified sample, and 0 for incorrectly classified sample. The architecture of ConfidNet is a fully connected feed-forward NN regressor with 2 hidden layers, each with 64 hidden neurons. The activation function is ReLU for all the hidden layers. The maximum number of epochs for training is 1000. An early stop is triggered if the loss on validation data has not be improved for 10 epochs. The optimizer is RMSprop with learning rate 0.001, and the loss function is mean squared error (MSE).
- Entropy: the entropy of softmax outputs of the base NN classifier is used as the confidence score of Entropy. Setups of the base NN classifier is provided above.
- DNGO: a Bayesian linear regression layer same as in (Snoek et al., 2015) is added after the logits layer of the original NN classifier to predict whether an original prediction is correct or not (1 for correct and 0 for incorrect). Default parametric setups as in https://github.com/automl/pybnn/blob/master/pybnn/dngo.py is used.
- SVGP: the original SVGP without output kernel is used to directly predict whether a prediction made by the base NN classifier is correct or not (1 for correct and 0 for incorrect). All other parametric setups are identical with those in RED.
- BNN MCP: the standard dense layers in the base NN classifier described in RED setups above is replaced with Flipout layers (Wen et al., 2018). All the other setups are identical with those in RED setups. The maximum class probability averaging over 100 test-time samplings is used as the confidence score for error detection.
- BNN Entropy: same setup as BNN MCP, except that now the entropy of softmax outputs averaging over 100 test-time samplings is used as confidence score for error detection.
- MC-Dropout MCP: a dropout layer with dropout rate 0.5 is added after each dense layer of the base NN classifier described in RED setups. All the other setups are identical with those in RED setups. The maximum class probability averaging over 100 test-time Monte-Carlo samplings is used as the confidence score for error detection.

- MC-Dropout Entropy: same setup as MC-Dropout MCP, except that now the entropy of softmax outputs averaging over 100 test-time Monte-Carlo samplings is used as confidence score for error detection.

A.2 Experimental Setups for Section 4.2

Setup of VGG16 Training The standard VGG16 architecture (Simonyan & Zisserman, 2015) is used. The training pipeline is using the same default setups as in https://github.com/geifmany/cifar-vgg/blob/master/cifar10vgg.py. For CIFAR-10 dataset, 40000 samples are used as training set, 10000 as validation set, and 10000 as testing set.

Parametric Setup for Algorithms All algorithms use the logit outputs of the trained VGG16 model as input features. The maximum class probability of softmax outputs of the trained VGG16 model is used as the confidence score of MCP baseline. The parametric setups for RED, Trust Score, Entropy, DNGO and SVGP are identical with the UCI experiments. For ConfidNet and Introspection-Net, all parametric setups are the same as in UCI experiments, except for that the number of hidden neurons for all hidden layers is incresed to 128.

A.3 Detailed Results for Section 4.1

This subsection shows all the detailed results for experiments performed in Section 4.1. The results averaging over 10 independent runs in terms of AP-Error, AP-Success, AUPR-Error, AUPR-Success, and AUROC are shown in Table 5, Table 6, Table 7, Table 8, and Table9, respectively. The column "N", "M", and "K" denotes the number of samples, number of features, and number of classes for corresponding datasets. For datasets that the original NN classifier achieves 100% accuracy, the entries are marked as "NA". For dataset splits that the number of samples in one particular class is too small for Trust Score to calculate neighborhood distance, the entries are marked as "NA".

Table 5: Comparison between RED and Counterparts Using AP-Error

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
abalone	4177	8	3	0.543	0.539	0.480	0.518	0.528
acute-inflammation	120	6	2	NA	NA	NA	NA	NA
acute-nephritis	120	6	2	NA	NA	NA	NA	NA
adult	48842	14	2	0.420	0.411	0.328	0.391	0.412
annealing	898	31	5	0.553	0.392	0.453	0.350	0.399
arrhythmia	452	262	13	0.569	0.580	0.618	0.409	0.413
audiology-std	196	59	18	0.776	0.721	NA	0.428	0.518
balance-scale	625	4	3	0.879	0.816	0.540	0.046	0.155
balloons	16	4	2	0.824	0.806	0.704	0.694	0.806
bank	4521	16	2	0.393	0.393	0.390	0.372	0.339
bioconcentration	779	9	3	0.494	0.470	0.510	0.426	0.461
blood	748	4	2	0.389	0.389	0.272	0.382	0.372
breast-cancer	286	9	2	0.516	0.503	0.381	0.467	0.431
breast-cancer-wisc	699	9	2	0.418	0.393	0.405	0.313	0.245
breast-cancer-wisc-diag	569	30	2	0.619	0.599	0.500	0.089	0.374
breast-cancer-wisc-prog	198	33	2	0.415	0.403	0.296	0.343	0.373
breast-tissue	106	9	6	0.816	0.743	0.800	0.696	0.601
car	1728	6	4	0.561	0.462	0.256	0.140	0.053
cardiotocography-10clases	2126	21	10	0.508	0.492	0.400	0.310	0.302
cardiotocography-3clases	2126	21	3	0.436	0.386	0.393	0.273	0.282
chess-krvk	28056	6	18	0.551	0.523	0.715	0.486	0.449
chess-krvkp	3196	36	2	0.350	0.339	0.163	0.037	0.087
climate	540	20	2	0.455	0.428	0.276	0.093	0.394
congressional-voting	435	16	2	0.467	0.465	0.429	0.474	0.459
conn-bench-sonar-mines-rocks	208	60	2	0.531	0.531	0.697	0.296	0.343
conn-bench-vowel-deterding	990	11	11	0.653	0.523	0.875	0.148	0.170
connect-4	67557	42	2	0.406	0.390	0.326	0.259	0.395
contrac	1473	9	3	0.594	0.590	0.529	0.552	0.584
credit-approval	690	15	2	0.364	0.359	0.336	0.264	0.331
cylinder-bands	512	35	2	0.502	0.482	0.506	0.345	0.422
dermatology	366	34	6	0.702	0.648	0.605	0.072	0.253
echocardiogram	131	10	2	0.474	0.413	0.325	0.353	0.407
ecoli	336	7	8	0.437	0.421	0.472	0.283	0.284
energy-y1	768	8	3	0.438	0.380	0.289	0.195	0.187
energy-y2	768	8	3	0.488	0.434	0.498	0.322	0.278
fertility	100	9	2	0.259	0.238	0.239	0.193	0.217

Continued on next page.

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
flags	194	28	8	0.773	0.717	0.784	0.666	0.664
glass	214	9	6	0.583	0.525	0.657	0.507	0.412
haberman-survival	306	3	2	0.454	0.446	0.386	0.412	0.425
hayes-roth	160	3	3	0.688	0.576	0.728	0.603	0.572
heart-cleveland	303	13	5	0.761	0.748	0.707	0.689	0.705
heart-hungarian	294	12	2	0.440	0.418	0.442	0.261	0.397
heart-switzerland	123	12	5	0.741	0.692	0.726	0.697	0.662
heart-va	200	12	5	0.755	0.732	0.676	0.678	0.732
hepatitis	155	19	2	0.532	0.441	0.495	0.423	0.330
hill-valley	1212	100	2	0.643	0.546	0.478	0.508	0.594
horse-colic	368	25	2	0.445	0.431	0.426	0.263	0.394
ilpd-indian-liver	583	9	2	0.469	0.459	0.427	0.434	0.443
image-segmentation	2310	18	7	0.534	0.455	0.471	0.251	0.214
ionosphere	351	33	2	0.548	0.493	0.335	0.158	0.405
iris	150	4	3	0.655	0.633	0.747	0.154	0.202
led-display	1000	7	10	0.572	0.571	0.301	0.553	0.512
lenses	24	4	3	0.896	0.900	0.786	0.665	0.790
letter	20000	16	26	0.490	0.461	0.785	0.245	0.081
libras	360	90	15	0.593	0.512	0.657	0.307	0.301
low-res-spect	531	100	9	0.555	0.515	0.454	0.304	0.305
lung-cancer	32	56	3	0.873	0.819	0.695	0.690	0.841
lymphography	148	18	4	0.664	0.546	0.619	0.278	0.386
magic	19020	10	2	0.366	0.355	0.294	0.309	0.350
mammographic	961	5	2	0.470	0.468	0.288	0.414	0.454
messidor	1151	19	2	0.463	0.449	0.359	0.364	0.426
miniboone	130064	50	2	0.379	0.361	0.190	0.299	0.371
molec-biol-promoter	106	57	2	0.556	0.541	0.786	0.330	0.505
molec-biol-splice	3190	60	3	0.463	0.455	0.375	0.193	0.208
monks-1	556	6	2	0.602	0.444	0.167	0.242	0.343
monks-2	601	6	2	0.519	0.466	0.611	0.417	0.426
monks-3	554	6	2	0.640	0.561	0.369	0.319	0.453
mushroom	8124	21	2	NA	NA	NA	NA	NA
musk-1	476	166	2	0.524	0.508	0.501	0.101	0.354
musk-2	6598	166	2	0.414	0.372	0.083	0.009	0.077
nursery	12960	8	5	0.600	0.448	0.074	0.035	0.127
oocytes_merluccius_nucleus_4d	1022	41	2	0.442	0.401	0.277	0.296	0.383
oocytes_merluccius_states_2f	1022	25	3	0.507	0.488	0.413	0.256	0.358
oocytes_trisopterus_nucleus_2f	912	25	2	0.480	0.451	0.342	0.275	0.460
oocytes_trisopterus_states_5b	912	32	3	0.436	0.423	0.346	0.155	0.291
optical	5620	62	10	0.451	0.429	0.582	0.048	0.036
ozone	2536	72	2	0.300	0.293	0.270	0.207	0.146
page-blocks	5473	10	5	0.448	0.386	0.399	0.330	0.275
parkinsons	195	22	2	0.561	0.516	0.541	0.155	0.239
pendigits	10992	16	10	0.493	0.362	0.684	0.178	0.132
phishing	1353	9	3	0.451	0.433	0.369	0.347	0.371
pima	768	8	2	0.498	0.492	0.422	0.412	0.486
pittsburg-bridges-MATERIAL	106	7	3	0.575	0.501	0.485	0.360	0.511
pittsburg-bridges-REL-L	103	7	3	0.590	0.536	0.654	0.522	0.549
pittsburg-bridges-SPAN	92	7	3	0.570	0.550	0.543	0.367	0.507
pittsburg-bridges-T-OR-D	102	7	2	0.479	0.397	0.456	0.391	0.506
pittsburg-bridges-TYPE	105	7	6	0.807	0.736	0.758	0.684	0.662
planning	182	12	2	0.367	0.364	0.387	0.348	0.331
plant-margin	1600	64	100	0.581	0.572	0.696	0.331	0.266
plant-shape	1600	64	100	0.688	0.666	0.759	0.524	0.477
plant-texture	1599	64	100	0.604	0.596	0.708	0.308	0.291
post-operative	90	8	3	0.416	0.416	0.289	0.377	0.413
primary-tumor	330	17	15	0.792	0.783	0.693	0.725	0.721
ringnorm	7400	20	2	0.349	0.344	0.275	0.042	0.116
seeds	210	7	3	0.595	0.514	0.526	0.344	0.333
semeion	1593	256	10	0.537	0.527	0.704	0.092	0.237
soybean	683	35	18	0.569	0.516	0.448	0.317	0.191
spambase	4601	57	2	0.315	0.292	0.299	0.179	0.238
spect	265	22	2	0.521	0.506	0.384	0.443	0.471
spectf	267	44	2	0.479	0.427	0.429	0.251	0.428
statlog-australian-credit	690	14	2	0.474	0.455	0.418	0.417	0.416
statlog-german-credit	1000	24	2	0.425	0.422	0.383	0.329	0.389
statlog-heart	270	13	2	0.462	0.438	0.388	0.322	0.462
statlog-image	2310	18	7	0.552	0.464	0.484	0.240	0.114
statlog-landsat	6435	36	6	0.432	0.398	0.554	0.322	0.312
statlog-shuttle	58000	9	7	0.560	0.318	0.547	0.233	0.157
statlog-vehicle	846	18	4	0.541	0.526	0.319	0.392	0.361
steel-plates	1941	27	7	0.525	0.517	0.533	0.452	0.437
synthetic-control	600	60	6	0.657	0.541	0.462	0.052	0.117
teaching	151	5	3	0.635	0.604	0.540	0.632	0.593
thyroid	7200	21	3	0.445	0.413	0.160	0.126	0.190
tic-tac-toe	958	9	2	0.664	0.538	1.000	0.160	0.513
titanic	2201	3	2	0.314	0.313	0.202	0.310	0.315

Continued on next page.

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
trains	10	29	2	0.750	0.750	0.917	0.833	0.750
twonorm	7400	20	2	0.369	0.357	0.404	0.160	0.290
vertebral-column-2clases	310	6	2	0.496	0.490	0.346	0.378	0.432
vertebral-column-3clases	310	6	3	0.544	0.518	0.308	0.489	0.396
wall-following	5456	24	4	0.420	0.378	0.304	0.208	0.248
waveform	5000	21	3	0.466	0.423	0.331	0.357	0.394
waveform-noise	5000	40	3	0.427	0.398	0.322	0.230	0.326
wine	178	13	3	0.867	0.742	0.661	0.101	0.458
wine-quality-red	1599	11	6	0.534	0.522	0.540	0.494	0.491
wine-quality-white	4898	11	7	0.540	0.529	0.626	0.534	0.523
yeast	1484	8	10	0.571	0.563	0.510	0.506	0.498
zoo	101	16	7	0.910	0.804	0.793	0.297	0.546

Continued from previous page.

Table 6: Comparison between RED and Counterparts Using AP-Success

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
abalone	4177	8	3	0.852	0.850	0.821	0.845	0.844
acute-inflammation	120	6	2	NA	NA	NA	NA	NA
acute-nephritis	120	6	2	NA	NA	NA	NA	NA
adult	48842	14	2	0.968	0.967	0.944	0.964	0.961
annealing	898	31	5	0.976	0.899	0.943	0.909	0.916
arrhythmia	452	262	13	0.816	0.810	0.809	0.654	0.668
audiology-std	196	59	18	0.947	0.926	NA	0.625	0.740
balance-scale	625	4	3	0.995	0.999	0.997	0.972	0.983
balloons	16	4	2	0.818	0.806	0.806	0.769	0.806
bank	4521	16	2	0.975	0.980	0.976	0.968	0.937
bioconcentration	779	9	3	0.747	0.728	0.782	0.717	0.726
blood	748	4	2	0.892	0.892	0.873	0.902	0.883
breast-cancer	286	9	2	0.815	0.820	0.765	0.796	0.775
breast-cancer-wisc	699	9	2	0.998	0.997	0.998	0.994	0.985
breast-cancer-wisc-diag	569	30	2	0.995	0.996	0.997	0.970	0.981
breast-cancer-wisc-prog	198	33	2	0.859	0.860	0.797	0.824	0.835
breast-tissue	106	9	6	0.909	0.878	0.898	0.863	0.767
car	1728	6	4	0.999	0.999	0.999	0.992	0.988
cardiotocography-10clases	2126	21	10	0.956	0.959	0.927	0.890	0.872
cardiotocography-3clases	2126	21	3	0.986	0.991	0.988	0.972	0.964
chess-krvk	28056	6	18	0.900	0.890	0.934	0.867	0.838
chess-krvkp	3196	36	2	1.000	1.000	0.995	0.990	0.994
climate	540	20	2	0.992	0.989	0.976	0.938	0.976
congressional-voting	435	16	2	0.692	0.690	0.672	0.692	0.684
conn-bench-sonar-mines-rocks	208	60	2	0.962	0.963	0.980	0.815	0.901
conn-bench-vowel-deterding	990	11	11	1.000	1.000	1.000	0.980	0.984
connect-4	67557	42	2	0.969	0.969	0.938	0.942	0.958
contrac	1473	9	3	0.716	0.714	0.581	0.682	0.675
credit-approval	690	15	2	0.941	0.941	0.916	0.909	0.936
cylinder-bands	512	35	2	0.867	0.862	0.872	0.789	0.809
dermatology	366	34	6	0.999	0.999	0.997	0.975	0.981
echocardiogram	131	10	2	0.939	0.936	0.925	0.916	0.924
ecoli	336	7	8	0.959	0.956	0.969	0.911	0.879
energy-y1	768	8	3	0.989	0.994	0.992	0.975	0.974
energy-y2	768	8	3	0.985	0.986	0.985	0.962	0.946
fertility	100	9	2	0.949	0.928	0.948	0.920	0.929
flags	194	28	8	0.735	0.716	0.817	0.573	0.617
glass	214	9	6	0.853	0.811	0.915	0.779	0.738
haberman-survival	306	3	2	0.850	0.849	0.854	0.821	0.835
hayes-roth	160	3	3	0.932	0.881	0.959	0.887	0.864
heart-cleveland	303	13	5	0.857	0.859	0.836	0.800	0.822
heart-hungarian	294	12	2	0.926	0.908	0.927	0.847	0.915
heart-switzerland	123	12	5	0.499	0.480	0.478	0.451	0.452
heart-va	200	12	5	0.510	0.499	0.462	0.439	0.504
hepatitis	155	19	2	0.969	0.965	0.975	0.930	0.896
hill-valley	1212	100	2	0.758	0.704	0.559	0.653	0.741
horse-colic	368	25	2	0.925	0.918	0.920	0.833	0.913
ilpd-indian-liver	583	9	2	0.878	0.874	0.814	0.860	0.863
image-segmentation	2310	18	7	0.997	0.997	0.998	0.983	0.984
ionosphere	351	33	2	0.990	0.988	0.976	0.900	0.973
iris	150	4	3	0.987	0.991	0.997	0.938	0.917
led-display	1000	7	10	0.883	0.883	0.759	0.874	0.850
lenses	24	4	3	0.984	0.960	0.944	0.871	0.883
letter	20000	16	26	0.994	0.996	0.999	0.980	0.947
libras	360	90	15	0.955	0.943	0.956	0.808	0.807
low-res-spect	531	100	9	0.977	0.968	0.953	0.908	0.892
lung-cancer	32	56	3	0.773	0.734	0.650	0.547	0.743

Continued on next page.

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
lymphography	148	18	4	0.938	0.925	0.958	0.809	0.861
magic	19020	10	2	0.966	0.966	0.954	0.961	0.959
mammographic	961	5	2	0.922	0.919	0.882	0.910	0.911
messidor	1151	19	2	0.859	0.861	0.764	0.806	0.830
miniboone	130064	50	2	0.992	0.992	0.983	0.988	0.986
molec-biol-promoter	106	57	2	0.895	0.888	0.953	0.724	0.880
molec-biol-splice	3190	60	3	0.953	0.953	0.926	0.833	0.846
monks-1	556	6	2	0.997	0.997	0.984	0.988	0.989
monks-2	601	6	2	0.880	0.857	0.918	0.809	0.810
monks-3	554	6	2	0.999	0.999	0.993	0.981	0.993
mushroom	8124	21	2	NA	NA	NA	NA	NA
musk-1	476	166	2	0.983	0.983	0.986	0.912	0.957
musk-2	6598	166	2	1.000	1.000	0.998	0.995	0.995
nursery	12960	8	5	1.000	1.000	1.000	0.999	0.999
oocytes_merluccius_nucleus_4d	1022	41	2	0.939	0.930	0.881	0.892	0.904
oocytes_merluccius_states_2f	1022	25	3	0.991	0.991	0.991	0.967	0.970
oocytes_trisopterus_nucleus_2f	912	25	2	0.939	0.931	0.908	0.872	0.923
oocytes_trisopterus_states_5b	912	32	3	0.991	0.990	0.985	0.952	0.967
optical	5620	62	10	0.999	0.999	0.999	0.981	0.981
ozone	2536	72	2	0.994	0.996	0.993	0.988	0.966
page-blocks	5473	10	5	0.998	0.997	0.993	0.993	0.989
parkinsons	195	22	2	0.994	0.992	0.995	0.939	0.944
pendigits	10992	16	10	0.999	0.999	1.000	0.992	0.992
phishing	1353	9	3	0.973	0.973	0.958	0.959	0.946
pima	768	8	2	0.899	0.894	0.875	0.848	0.873
pittsburg-bridges-MATERIAL	106	7	3	0.972	0.969	0.953	0.921	0.933
pittsburg-bridges-REL-L	103	7	3	0.788	0.780	0.845	0.766	0.744
pittsburg-bridges-SPAN	92	7	3	0.841	0.808	0.785	0.678	0.756
pittsburg-bridges-T-OR-D	102	7	2	0.950	0.943	0.924	0.926	0.946
pittsburg-bridges-TYPE	105	7	6	0.816	0.692	0.747	0.651	0.635
planning	182	12	2	0.729	0.722	0.737	0.730	0.728
plant-margin	1600	64	100	0.950	0.950	0.966	0.828	0.817
plant-shape	1600	64	100	0.861	0.849	0.880	0.718	0.671
plant-texture	1599	64	100	0.953	0.950	0.958	0.809	0.804
post-operative	90	8	3	0.679	0.673	0.739	0.684	0.685
primary-tumor	330	17	15	0.757	0.752	0.654	0.640	0.633
ringnorm	7400	20	2	0.998	0.999	0.989	0.982	0.977
seeds	210	7	3	0.988	0.986	0.992	0.964	0.961
semeion	1593	256	10	0.994	0.994	0.996	0.927	0.950
soybean	683	35	18	0.996	0.995	0.993	0.970	0.942
spambase	4601	57	2	0.989	0.988	0.984	0.965	0.966
spect	265	22	2	0.809	0.802	0.767	0.731	0.788
spectf	267	44	2	0.955	0.951	0.942	0.913	0.919
statlog-australian-credit	690	14	2	0.702	0.692	0.656	0.668	0.678
statlog-german-credit	1000	24	2	0.889	0.891	0.873	0.837	0.850
statlog-heart	270	13	2	0.939	0.935	0.906	0.882	0.917
statlog-image	2310	18	7	0.997	0.998	0.998	0.986	0.980
statlog-landsat	6435	36	6	0.985	0.987	0.991	0.964	0.959
statlog-shuttle	58000	9	7	1.000	1.000	1.000	0.998	0.998
statlog-vehicle	846	18	4	0.965	0.967	0.927	0.929	0.913
steel-plates	1941	27	7	0.919	0.917	0.928	0.882	0.850
synthetic-control	600	60	6	1.000	1.000	0.999	0.986	0.988
teaching	151	5	3	0.676	0.648	0.598	0.657	0.640
thyroid	7200	21	3	0.998	0.999	0.996	0.992	0.984
tic-tac-toe	958	9	2	1.000	0.999	1.000	0.999	0.998
titanic	2201	3	2	0.877	0.876	0.805	0.874	0.873
trains	10	29	2	0.750	0.750	0.917	0.833	0.750
twonorm	7400	20	2	0.998	0.999	0.998	0.991	0.991
vertebral-column-2clases	310	6	2	0.977	0.977	0.953	0.952	0.942
vertebral-column-3clases	310	6	3	0.978	0.977	0.943	0.959	0.933
wall-following	5456	24	4	0.980	0.980	0.972	0.928	0.949
waveform	5000	21	3	0.968	0.966	0.947	0.947	0.941
waveform-noise	5000	40	3	0.962	0.962	0.935	0.881	0.900
wine	178	13	3	0.999	0.999	0.998	0.948	0.990
wine-quality-red	1599	11	6	0.734	0.727	0.751	0.705	0.696
wine-quality-white	4898	11	7	0.673	0.663	0.745	0.650	0.655
yeast	1484	8	10	0.757	0.751	0.706	0.718	0.692
zoo	101	16	7	0.999	0.998	0.997	0.928	0.963

Continued from previous page.

Table 7: Comparison between RED and Counterparts Using AUPR-Error

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
abalone	4177	8	3	0.541	0.537	0.478	0.516	0.526

Continued on next page.

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
acute-inflammation	120	6	2	NA	NA	NA	NA	NA
acute-nephritis	120	6	2	NA	NA	NA	NA	NA
adult	48842	14	2	0.419	0.410	0.328	0.390	0.412
annealing	898	31	5	0.542	0.375	0.440	0.331	0.385
arrhythmia	452	262	13	0.558	0.570	0.606	0.396	0.399
audiology-std	196	59	18	0.763	0.704	NA	0.401	0.494
balance-scale	625	4	3	0.869	0.794	0.501	0.030	0.124
balloons	16	4	2	0.764	0.736	0.796	0.569	0.736
bank	4521	16	2	0.388	0.387	0.385	0.367	0.333
bioconcentration	779	9	3	0.486	0.461	0.501	0.418	0.453
blood	748	4	2	0.376	0.375	0.312	0.367	0.357
breast-cancer	286	9	2	0.500	0.483	0.354	0.447	0.405
breast-cancer-wisc	699	9	2	0.372	0.341	0.347	0.253	0.184
breast-cancer-wisc-diag	569	30	2	0.586	0.561	0.438	0.070	0.348
breast-cancer-wisc-prog	198	33	2	0.381	0.369	0.254	0.310	0.344
breast-tissue	106	9	6	0.795	0.717	0.771	0.660	0.556
car	1728	6	4	0.526	0.419	0.236	0.111	0.041
cardiotocography-10clases	2126	21	10	0.502	0.486	0.394	0.301	0.295
cardiotocography-3clases	2126	21	3	0.425	0.372	0.377	0.259	0.268
chess-krvk	28056	6	18	0.550	0.523	0.715	0.485	0.448
chess-krvkp	3196	36	2	0.312	0.299	0.141	0.024	0.067
climate	540	20	2	0.419	0.395	0.239	0.081	0.362
congressional-voting	435	16	2	0.486	0.503	0.659	0.505	0.543
conn-bench-sonar-mines-rocks	208	60	2	0.492	0.487	0.666	0.260	0.305
conn-bench-vowel-deterding	990	11	11	0.596	0.455	0.865	0.123	0.144
connect-4	67557	42	2	0.405	0.389	0.326	0.258	0.394
contrac	1473	9	3	0.590	0.585	0.525	0.546	0.579
credit-approval	690	15	2	0.343	0.337	0.316	0.241	0.311
cylinder-bands	512	35	2	0.490	0.464	0.493	0.331	0.406
dermatology	366	34	6	0.612	0.572	0.573	0.046	0.194
echocardiogram	131	10	2	0.421	0.350	0.274	0.303	0.357
ecoli	336	7	8	0.413	0.396	0.443	0.244	0.259
energy-y1	768	8	3	0.401	0.333	0.258	0.163	0.165
energy-y2	768	8	3	0.467	0.409	0.479	0.298	0.260
fertility	100	9	2	0.212	0.200	0.180	0.153	0.165
flags	194	28	8	0.766	0.703	0.774	0.651	0.649
glass	214	9	6	0.565	0.501	0.637	0.480	0.385
haberman-survival	306	3	2	0.429	0.420	0.363	0.389	0.400
hayes-roth	160	3	3	0.671	0.560	0.796	0.569	0.537
heart-cleveland	303	13	5	0.755	0.739	0.694	0.675	0.694
heart-hungarian	294	12	2	0.409	0.388	0.416	0.234	0.360
heart-switzerland	123	12	5	0.726	0.672	0.711	0.677	0.639
heart-va	200	12	5	0.745	0.720	0.658	0.664	0.721
hepatitis	155	19	2	0.488	0.383	0.435	0.375	0.273
hill-valley	1212	100	2	0.640	0.540	0.472	0.502	0.591
horse-colic	368	25	2	0.425	0.407	0.399	0.245	0.369
ilpd-indian-liver	583	9	2	0.457	0.445	0.410	0.419	0.428
image-segmentation	2310	18	7	0.521	0.434	0.449	0.229	0.200
ionosphere	351	33	2	0.518	0.452	0.303	0.128	0.364
iris	150	4	3	0.588	0.553	0.684	0.105	0.173
led-display	1000	7	10	0.574	0.573	0.566	0.557	0.512
lenses	24	4	3	0.865	0.887	0.798	0.572	0.759
letter	20000	16	26	0.488	0.459	0.784	0.242	0.125
libras	360	90	15	0.574	0.485	0.639	0.287	0.282
low-res-spect	531	100	9	0.540	0.493	0.437	0.281	0.288
lung-cancer	32	56	3	0.850	0.784	0.609	0.633	0.812
lymphography	148	18	4	0.645	0.512	0.582	0.228	0.339
magic	19020	10	2	0.364	0.353	0.293	0.307	0.348
mammographic	961	5	2	0.458	0.456	0.277	0.399	0.441
messidor	1151	19	2	0.455	0.442	0.350	0.356	0.417
miniboone	130064	50	2	0.379	0.361	0.190	0.299	0.370
molec-biol-promoter	106	57	2	0.512	0.497	0.765	0.292	0.436
molec-biol-splice	3190	60	3	0.459	0.450	0.372	0.189	0.203
monks-1	556	6	2	0.550	0.359	0.126	0.199	0.305
monks-2	601	6	2	0.505	0.452	0.622	0.405	0.412
monks-3	554	6	2	0.591	0.509	0.290	0.269	0.403
mushroom	8124	21	2	NA	NA	NA	NA	NA
musk-1	476	166	2	0.489	0.470	0.476	0.081	0.319
musk-2	6598	166	2	0.380	0.339	0.064	0.007	0.061
nursery	12960	8	5	0.562	0.388	0.070	0.027	0.129
oocytes_merluccius_nucleus_4d	1022	41	2	0.432	0.389	0.264	0.284	0.368
oocytes_merluccius_states_2f	1022	25	3	0.488	0.466	0.387	0.234	0.338
oocytes_trisopterus_nucleus_2f	912	25	2	0.469	0.439	0.329	0.262	0.447
oocytes_trisopterus_states_5b	912	32	3	0.407	0.394	0.321	0.137	0.266
optical	5620	62	10	0.436	0.411	0.572	0.043	0.037
ozone	2536	72	2	0.278	0.269	0.253	0.189	0.127
page-blocks	5473	10	5	0.437	0.374	0.388	0.317	0.263

Continued on next page.

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
parkinsons	195	22	2	0.491	0.456	0.477	0.108	0.197
pendigits	10992	16	10	0.482	0.344	0.677	0.168	0.124
phishing	1353	9	3	0.437	0.416	0.357	0.332	0.355
pima	768	8	2	0.487	0.480	0.408	0.399	0.477
pittsburg-bridges-MATERIAL	106	7	3	0.529	0.444	0.405	0.308	0.459
pittsburg-bridges-REL-L	103	7	3	0.561	0.498	0.624	0.485	0.519
pittsburg-bridges-SPAN	92	7	3	0.522	0.504	0.494	0.319	0.461
pittsburg-bridges-T-OR-D	102	7	2	0.397	0.299	0.376	0.320	0.462
pittsburg-bridges-TYPE	105	7	6	0.795	0.713	0.741	0.658	0.634
planning	182	12	2	0.340	0.337	0.362	0.315	0.295
plant-margin	1600	64	100	0.575	0.565	0.692	0.323	0.258
plant-shape	1600	64	100	0.685	0.662	0.757	0.519	0.475
plant-texture	1599	64	100	0.599	0.590	0.704	0.300	0.323
post-operative	90	8	3	0.375	0.375	0.230	0.329	0.368
primary-tumor	330	17	15	0.787	0.777	0.683	0.717	0.712
ringnorm	7400	20	2	0.335	0.329	0.269	0.040	0.111
seeds	210	7	3	0.521	0.447	0.453	0.282	0.284
semeion	1593	256	10	0.524	0.514	0.698	0.083	0.224
soybean	683	35	18	0.542	0.476	0.418	0.282	0.164
spambase	4601	57	2	0.307	0.282	0.290	0.170	0.228
spect	265	22	2	0.503	0.487	0.381	0.425	0.451
spectf	267	44	2	0.446	0.386	0.395	0.216	0.392
statlog-australian-credit	690	14	2	0.465	0.446	0.408	0.408	0.406
statlog-german-credit	1000	24	2	0.414	0.410	0.373	0.320	0.378
statlog-heart	270	13	2	0.427	0.406	0.351	0.288	0.435
statlog-image	2310	18	7	0.531	0.437	0.465	0.220	0.100
statlog-landsat	6435	36	6	0.428	0.393	0.551	0.317	0.306
statlog-shuttle	58000	9	7	0.542	0.294	0.527	0.214	0.183
statlog-vehicle	846	18	4	0.531	0.515	0.304	0.376	0.347
steel-plates	1941	27	7	0.520	0.511	0.528	0.446	0.430
synthetic-control	600	60	6	0.595	0.453	0.396	0.031	0.076
teaching	151	5	3	0.612	0.578	0.503	0.614	0.568
thyroid	7200	21	3	0.433	0.397	0.148	0.116	0.178
tic-tac-toe	958	9	2	0.606	0.473	1.000	0.132	0.461
titanic	2201	3	2	0.386	0.385	0.599	0.372	0.386
trains	10	29	2	0.625	0.625	0.875	0.750	0.625
twonorm	7400	20	2	0.358	0.345	0.394	0.152	0.278
vertebral-column-2clases	310	6	2	0.462	0.456	0.302	0.345	0.402
vertebral-column-3clases	310	6	3	0.509	0.478	0.279	0.460	0.366
wall-following	5456	24	4	0.416	0.373	0.299	0.204	0.243
waveform	5000	21	3	0.463	0.419	0.327	0.353	0.390
waveform-noise	5000	40	3	0.423	0.394	0.318	0.227	0.322
wine	178	13	3	0.815	0.669	0.606	0.056	0.413
wine-quality-red	1599	11	6	0.530	0.518	0.537	0.489	0.486
wine-quality-white	4898	11	7	0.539	0.527	0.625	0.532	0.521
yeast	1484	8	10	0.565	0.557	0.506	0.502	0.493
zoo	101	16	7	0.888	0.744	0.752	0.183	0.473

Continued from previous page.

Table 8: Comparison between RED and Counterparts Using AUPR-Success

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
abalone	4177	8	3	0.852	0.850	0.821	0.844	0.844
acute-inflammation	120	6	2	NA	NA	NA	NA	NA
acute-nephritis	120	6	2	NA	NA	NA	NA	NA
adult	48842	14	2	0.968	0.967	0.944	0.964	0.961
annealing	898	31	5	0.976	0.903	0.942	0.907	0.914
arrhythmia	452	262	13	0.813	0.806	0.806	0.645	0.659
audiology-std	196	59	18	0.946	0.925	NA	0.609	0.725
balance-scale	625	4	3	0.995	0.999	0.997	0.971	0.983
balloons	16	4	2	0.742	0.736	0.861	0.690	0.736
bank	4521	16	2	0.975	0.980	0.976	0.968	0.936
bioconcentration	779	9	3	0.743	0.723	0.779	0.712	0.722
blood	748	4	2	0.891	0.891	0.872	0.901	0.882
breast-cancer	286	9	2	0.810	0.814	0.757	0.788	0.767
breast-cancer-wisc	699	9	2	0.998	0.997	0.998	0.994	0.984
breast-cancer-wisc-diag	569	30	2	0.995	0.997	0.997	0.969	0.980
breast-cancer-wisc-prog	198	33	2	0.853	0.855	0.787	0.817	0.828
breast-tissue	106	9	6	0.903	0.869	0.893	0.855	0.751
car	1728	6	4	0.999	0.999	0.999	0.992	0.988
cardiotocography-10clases	2126	21	10	0.956	0.959	0.927	0.889	0.871
cardiotocography-3clases	2126	21	3	0.986	0.991	0.988	0.971	0.963
chess-krvk	28056	6	18	0.900	0.890	0.934	0.867	0.837
chess-krvkp	3196	36	2	1.000	1.000	0.995	0.990	0.994

Continued on next page.

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
climate	540	20	2	0.991	0.989	0.976	0.937	0.975
congressional-voting	435	16	2	0.701	0.701	0.706	0.704	0.694
conn-bench-sonar-mines-rocks	208	60	2	0.961	0.963	0.980	0.806	0.898
conn-bench-vowel-deterding	990	11	11	1.000	1.000	1.000	0.979	0.984
connect-4	67557	42	2	0.969	0.969	0.938	0.942	0.958
contrac	1473	9	3	0.715	0.713	0.578	0.680	0.672
credit-approval	690	15	2	0.941	0.941	0.915	0.908	0.936
cylinder-bands	512	35	2	0.865	0.860	0.870	0.784	0.804
dermatology	366	34	6	0.999	0.999	0.997	0.975	0.980
echocardiogram	131	10	2	0.936	0.934	0.922	0.913	0.920
ecoli	336	7	8	0.959	0.955	0.969	0.908	0.874
energy-y1	768	8	3	0.989	0.993	0.992	0.975	0.973
energy-y2	768	8	3	0.984	0.986	0.985	0.961	0.945
fertility	100	9	2	0.947	0.923	0.946	0.914	0.924
flags	194	28	8	0.725	0.707	0.812	0.551	0.599
glass	214	9	6	0.848	0.804	0.912	0.767	0.725
haberman-survival	306	3	2	0.844	0.843	0.851	0.815	0.828
hayes-roth	160	3	3	0.933	0.880	0.965	0.882	0.854
heart-cleveland	303	13	5	0.854	0.856	0.832	0.792	0.818
heart-hungarian	294	12	2	0.924	0.905	0.926	0.841	0.912
heart-switzerland	123	12	5	0.461	0.440	0.429	0.399	0.414
heart-va	200	12	5	0.482	0.470	0.438	0.413	0.480
hepatitis	155	19	2	0.968	0.964	0.974	0.925	0.888
hill-valley	1212	100	2	0.756	0.702	0.554	0.650	0.739
horse-colic	368	25	2	0.924	0.916	0.917	0.827	0.911
ilpd-indian-liver	583	9	2	0.877	0.873	0.811	0.859	0.862
image-segmentation	2310	18	7	0.997	0.997	0.998	0.983	0.984
ionosphere	351	33	2	0.990	0.988	0.976	0.897	0.973
iris	150	4	3	0.986	0.991	0.997	0.935	0.912
led-display	1000	7	10	0.888	0.888	0.773	0.877	0.849
lenses	24	4	3	0.982	0.947	0.944	0.831	0.845
letter	20000	16	26	0.994	0.996	0.999	0.980	0.888
libras	360	90	15	0.954	0.942	0.955	0.802	0.800
low-res-spect	531	100	9	0.976	0.968	0.952	0.905	0.889
lung-cancer	32	56	3	0.731	0.687	0.572	0.461	0.703
lymphography	148	18	4	0.936	0.922	0.956	0.797	0.851
magic	19020	10	2	0.966	0.966	0.954	0.960	0.959
mammographic	961	5	2	0.922	0.918	0.881	0.909	0.910
messidor	1151	19	2	0.859	0.861	0.762	0.805	0.829
miniboone	130064	50	2	0.992	0.992	0.983	0.988	0.986
molec-biol-promoter	106	57	2	0.890	0.882	0.951	0.702	0.873
molec-biol-splice	3190	60	3	0.953	0.953	0.926	0.832	0.845
monks-1	556	6	2	0.997	0.997	0.982	0.988	0.989
monks-2	601	6	2	0.879	0.855	0.925	0.806	0.807
monks-3	554	6	2	0.999	0.999	0.992	0.980	0.993
mushroom	8124	21	2	NA	NA	NA	NA	NA
musk-1	476	166	2	0.983	0.983	0.986	0.910	0.956
musk-2	6598	166	2	1.000	1.000	0.998	0.995	0.995
nursery	12960	8	5	1.000	1.000	1.000	0.999	0.999
oocytes_merluccius_nucleus_4d	1022	41	2	0.938	0.929	0.881	0.891	0.903
oocytes_merluccius_states_2f	1022	25	3	0.991	0.991	0.991	0.967	0.970
oocytes_trisopterus_nucleus_2f	912	25	2	0.939	0.931	0.907	0.870	0.922
oocytes_trisopterus_states_5b	912	32	3	0.991	0.990	0.985	0.951	0.967
optical	5620	62	10	0.999	0.999	0.999	0.981	0.978
ozone	2536	72	2	0.994	0.996	0.993	0.988	0.966
page-blocks	5473	10	5	0.998	0.997	0.993	0.993	0.989
parkinsons	195	22	2	0.993	0.992	0.995	0.936	0.942
pendigits	10992	16	10	0.999	0.999	1.000	0.992	0.992
phishing	1353	9	3	0.973	0.973	0.958	0.959	0.945
pima	768	8	2	0.898	0.893	0.873	0.845	0.871
pittsburg-bridges-MATERIAL	106	7	3	0.971	0.968	0.950	0.916	0.928
pittsburg-bridges-REL-L	103	7	3	0.775	0.766	0.832	0.741	0.725
pittsburg-bridges-SPAN	92	7	3	0.831	0.793	0.766	0.649	0.734
pittsburg-bridges-T-OR-D	102	7	2	0.949	0.941	0.919	0.924	0.945
pittsburg-bridges-TYPE	105	7	6	0.803	0.658	0.718	0.612	0.596
planning	182	12	2	0.715	0.709	0.723	0.714	0.713
plant-margin	1600	64	100	0.950	0.949	0.966	0.827	0.816
plant-shape	1600	64	100	0.860	0.849	0.879	0.715	0.668
plant-texture	1599	64	100	0.953	0.950	0.958	0.807	0.780
post-operative	90	8	3	0.647	0.642	0.719	0.656	0.660
primary-tumor	330	17	15	0.752	0.746	0.646	0.627	0.623
ringnorm	7400	20	2	0.998	0.999	0.989	0.982	0.977
seeds	210	7	3	0.988	0.986	0.991	0.962	0.960
semeion	1593	256	10	0.994	0.994	0.996	0.926	0.949
soybean	683	35	18	0.996	0.995	0.993	0.970	0.940
spambase	4601	57	2	0.989	0.988	0.983	0.965	0.966
spect	265	22	2	0.803	0.795	0.762	0.721	0.783

Continued on next page.

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
spectf	267	44	2	0.954	0.951	0.941	0.911	0.917
statlog-australian-credit	690	14	2	0.696	0.687	0.649	0.662	0.672
statlog-german-credit	1000	24	2	0.888	0.890	0.872	0.835	0.848
statlog-heart	270	13	2	0.939	0.934	0.904	0.878	0.914
statlog-image	2310	18	7	0.997	0.998	0.998	0.986	0.979
statlog-landsat	6435	36	6	0.985	0.987	0.991	0.963	0.959
statlog-shuttle	58000	9	7	1.000	1.000	1.000	0.998	0.907
statlog-vehicle	846	18	4	0.965	0.967	0.927	0.928	0.912
steel-plates	1941	27	7	0.918	0.916	0.928	0.881	0.848
synthetic-control	600	60	6	1.000	1.000	0.999	0.985	0.988
teaching	151	5	3	0.656	0.625	0.569	0.630	0.625
thyroid	7200	21	3	0.998	0.999	0.996	0.992	0.984
tic-tac-toe	958	9	2	1.000	0.999	1.000	0.999	0.998
titanic	2201	3	2	0.895	0.894	0.889	0.890	0.891
trains	10	29	2	0.625	0.625	0.875	0.750	0.625
twonorm	7400	20	2	0.998	0.999	0.998	0.991	0.991
vertebral-column-2clases	310	6	2	0.977	0.977	0.952	0.952	0.940
vertebral-column-3clases	310	6	3	0.977	0.977	0.943	0.958	0.931
wall-following	5456	24	4	0.980	0.980	0.972	0.928	0.949
waveform	5000	21	3	0.968	0.966	0.947	0.947	0.940
waveform-noise	5000	40	3	0.962	0.962	0.935	0.880	0.900
wine	178	13	3	0.999	0.999	0.998	0.945	0.990
wine-quality-red	1599	11	6	0.733	0.726	0.750	0.702	0.692
wine-quality-white	4898	11	7	0.672	0.662	0.745	0.648	0.654
yeast	1484	8	10	0.755	0.749	0.704	0.716	0.689
zoo	101	16	7	0.999	0.997	0.997	0.919	0.958

Continued from previous page.

Table 9: Comparison between RED and Counterparts Using AUROC

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
abalone	4177	8	3	0.733	0.730	0.680	0.719	0.726
acute-inflammation	120	6	2	NA	NA	NA	NA	NA
acute-nephritis	120	6	2	NA	NA	NA	NA	NA
adult	48842	14	2	0.841	0.837	0.759	0.828	0.830
annealing	898	31	5	0.866	0.699	0.776	0.724	0.723
arrhythmia	452	262	13	0.707	0.715	0.727	0.505	0.517
audiology-std	196	59	18	0.885	0.848	NA	0.439	0.605
balance-scale	625	4	3	0.973	0.987	0.909	0.510	0.621
balloons	16	4	2	0.704	0.667	0.722	0.556	0.667
bank	4521	16	2	0.849	0.861	0.840	0.822	0.750
bioconcentration	779	9	3	0.622	0.607	0.660	0.572	0.595
blood	748	4	2	0.705	0.705	0.634	0.716	0.687
breast-cancer	286	9	2	0.673	0.677	0.587	0.637	0.609
breast-cancer-wisc	699	9	2	0.943	0.936	0.945	0.873	0.759
breast-cancer-wisc-diag	569	30	2	0.914	0.925	0.921	0.558	0.682
breast-cancer-wisc-prog	198	33	2	0.644	0.644	0.521	0.566	0.595
breast-tissue	106	9	6	0.868	0.815	0.850	0.788	0.651
car	1728	6	4	0.972	0.968	0.934	0.717	0.615
cardiotocography-10clases	2126	21	10	0.835	0.835	0.756	0.670	0.622
cardiotocography-3clases	2126	21	3	0.890	0.901	0.882	0.804	0.761
chess-krvk	28056	6	18	0.778	0.759	0.868	0.722	0.685
chess-krvkp	3196	36	2	0.964	0.963	0.777	0.573	0.642
climate	540	20	2	0.896	0.858	0.760	0.449	0.744
congressional-voting	435	16	2	0.601	0.600	0.562	0.607	0.595
conn-bench-sonar-mines-rocks	208	60	2	0.832	0.833	0.912	0.488	0.643
conn-bench-vowel-deterding	990	11	11	0.989	0.979	0.995	0.582	0.624
connect-4	67557	42	2	0.835	0.831	0.744	0.727	0.816
contrac	1473	9	3	0.662	0.658	0.556	0.619	0.642
credit-approval	690	15	2	0.738	0.741	0.683	0.649	0.703
cylinder-bands	512	35	2	0.706	0.702	0.713	0.579	0.637
dermatology	366	34	6	0.982	0.974	0.934	0.589	0.766
echocardiogram	131	10	2	0.746	0.709	0.655	0.634	0.687
ecoli	336	7	8	0.793	0.785	0.806	0.670	0.556
energy-y1	768	8	3	0.877	0.886	0.848	0.691	0.650
energy-y2	768	8	3	0.890	0.880	0.874	0.777	0.668
fertility	100	9	2	0.564	0.469	0.563	0.382	0.485
flags	194	28	8	0.729	0.696	0.791	0.603	0.618
glass	214	9	6	0.717	0.659	0.823	0.648	0.560
haberman-survival	306	3	2	0.694	0.692	0.651	0.645	0.671
hayes-roth	160	3	3	0.847	0.746	0.909	0.785	0.749
heart-cleveland	303	13	5	0.805	0.811	0.783	0.758	0.769
heart-hungarian	294	12	2	0.754	0.724	0.742	0.555	0.730
heart-switzerland	123	12	5	0.591	0.545	0.580	0.545	0.496

Continued on next page.

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
heart-va	200	12	5	0.617	0.606	0.544	0.522	0.603
hepatitis	155	19	2	0.833	0.797	0.845	0.703	0.619
hill-valley	1212	100	2	0.703	0.636	0.512	0.585	0.669
horse-colic	368	25	2	0.742	0.731	0.749	0.530	0.717
ilpd-indian-liver	583	9	2	0.719	0.709	0.650	0.686	0.695
image-segmentation	2310	18	7	0.933	0.936	0.946	0.770	0.742
ionosphere	351	33	2	0.912	0.886	0.779	0.504	0.786
iris	150	4	3	0.926	0.928	0.963	0.553	0.471
led-display	1000	7	10	0.762	0.762	0.545	0.749	0.717
lenses	24	4	3	0.927	0.875	0.863	0.677	0.740
letter	20000	16	26	0.926	0.940	0.977	0.795	0.526
libras	360	90	15	0.843	0.812	0.865	0.524	0.535
low-res-spect	531	100	9	0.861	0.837	0.791	0.618	0.559
lung-cancer	32	56	3	0.756	0.686	0.573	0.444	0.700
lymphography	148	18	4	0.803	0.753	0.825	0.503	0.622
magic	19020	10	2	0.810	0.807	0.753	0.788	0.796
mammographic	961	5	2	0.764	0.761	0.650	0.729	0.742
messidor	1151	19	2	0.695	0.690	0.565	0.609	0.662
miniboone	130064	50	2	0.905	0.903	0.806	0.878	0.882
molec-biol-promoter	106	57	2	0.741	0.727	0.885	0.414	0.719
molec-biol-splice	3190	60	3	0.811	0.810	0.722	0.517	0.547
monks-1	556	6	2	0.959	0.936	0.619	0.715	0.779
monks-2	601	6	2	0.745	0.698	0.814	0.616	0.632
monks-3	554	6	2	0.972	0.966	0.845	0.750	0.872
mushroom	8124	21	2	NA	NA	NA	NA	NA
musk-1	476	166	2	0.862	0.860	0.863	0.449	0.724
musk-2	6598	166	2	0.975	0.981	0.794	0.513	0.627
nursery	12960	8	5	0.990	0.955	0.925	0.601	0.603
oocytes_merluccius_nucleus_4d	1022	41	2	0.779	0.758	0.627	0.652	0.717
oocytes_merluccius_states_2f	1022	25	3	0.907	0.909	0.907	0.763	0.795
oocytes_trisopterus_nucleus_2f	912	25	2	0.786	0.766	0.693	0.619	0.759
oocytes_trisopterus_states_5b	912	32	3	0.897	0.895	0.834	0.638	0.744
optical	5620	62	10	0.969	0.964	0.973	0.561	0.534
ozone	2536	72	2	0.895	0.905	0.851	0.773	0.568
page-blocks	5473	10	5	0.950	0.942	0.905	0.906	0.855
parkinsons	195	22	2	0.916	0.881	0.928	0.574	0.561
pendigits	10992	16	10	0.954	0.948	0.972	0.681	0.635
phishing	1353	9	3	0.855	0.852	0.788	0.798	0.762
pima	768	8	2	0.755	0.751	0.700	0.673	0.730
pittsburg-bridges-MATERIAL	106	7	3	0.819	0.771	0.776	0.586	0.706
pittsburg-bridges-REL-L	103	7	3	0.635	0.611	0.739	0.612	0.587
pittsburg-bridges-SPAN	92	7	3	0.671	0.634	0.631	0.403	0.574
pittsburg-bridges-T-OR-D	102	7	2	0.766	0.740	0.694	0.669	0.750
pittsburg-bridges-TYPE	105	7	6	0.798	0.713	0.743	0.657	0.636
planning	182	12	2	0.483	0.468	0.509	0.492	0.473
plant-margin	1600	64	100	0.845	0.844	0.888	0.605	0.559
plant-shape	1600	64	100	0.794	0.780	0.823	0.647	0.587
plant-texture	1599	64	100	0.853	0.848	0.883	0.580	0.560
post-operative	90	8	3	0.435	0.433	0.414	0.433	0.446
primary-tumor	330	17	15	0.765	0.765	0.676	0.689	0.673
ringnorm	7400	20	2	0.950	0.952	0.708	0.587	0.565
seeds	210	7	3	0.917	0.890	0.913	0.746	0.724
semeion	1593	256	10	0.929	0.926	0.953	0.499	0.662
soybean	683	35	18	0.946	0.942	0.916	0.765	0.598
spambase	4601	57	2	0.864	0.856	0.809	0.710	0.724
spect	265	22	2	0.667	0.660	0.581	0.562	0.620
spectf	267	44	2	0.796	0.788	0.740	0.650	0.727
statlog-australian-credit	690	14	2	0.587	0.569	0.543	0.537	0.539
statlog-german-credit	1000	24	2	0.715	0.720	0.679	0.620	0.676
statlog-heart	270	13	2	0.761	0.746	0.693	0.624	0.731
statlog-image	2310	18	7	0.937	0.947	0.941	0.818	0.683
statlog-landsat	6435	36	6	0.887	0.887	0.919	0.795	0.770
statlog-shuttle	58000	9	7	0.994	0.919	0.998	0.763	0.589
statlog-vehicle	846	18	4	0.853	0.856	0.719	0.759	0.716
steel-plates	1941	27	7	0.795	0.790	0.805	0.736	0.695
synthetic-control	600	60	6	0.984	0.980	0.956	0.497	0.547
teaching	151	5	3	0.635	0.596	0.555	0.618	0.574
thyroid	7200	21	3	0.954	0.963	0.849	0.782	0.646
tic-tac-toe	958	9	2	0.987	0.933	1.000	0.876	0.894
titanic	2201	3	2	0.673	0.671	0.508	0.662	0.671
trains	10	29	2	0.500	0.500	0.833	0.667	0.500
twonorm	7400	20	2	0.951	0.952	0.941	0.817	0.812
vertebral-column-2clases	310	6	2	0.860	0.857	0.770	0.771	0.765
vertebral-column-3clases	310	6	3	0.877	0.873	0.715	0.819	0.710
wall-following	5456	24	4	0.850	0.847	0.793	0.641	0.706
waveform	5000	21	3	0.845	0.834	0.761	0.773	0.791
waveform-noise	5000	40	3	0.823	0.819	0.727	0.601	0.692

Continued on next page.

Dataset	N	M	K	RED	MCP baseline	Trust Score	ConfidNet	Introspection-Net
wine	178	13	3	0.989	0.975	0.954	0.345	0.799
wine-quality-red	1599	11	6	0.646	0.638	0.644	0.615	0.607
wine-quality-white	4898	11	7	0.621	0.612	0.677	0.604	0.605
yeast	1484	8	10	0.683	0.678	0.620	0.627	0.611
zoo	101	16	7	0.986	0.970	0.958	0.659	0.779

Continued from previous page.