Journal of Machine Learning Research 22 (2021) 1-66 Submitted 6/19; Revised 8/20; Published 1/21

A Unified Sample Selection Framework for Output Noise Filtering: An Error-Bound Perspective

Gaoxia Jiang JIANGGAOXIA@SXU.EDU.CN
School of Computer and Information Technology
Shanxi University
Taiyuan, 030006, PR China

Wenjian Wang* WJWANG@SXU.EDU.CN
School of Computer and Information Technology, Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education
Shanxi University
Taiyuan, 030006, PR China

Yuhua Qian JINCHENGQYH@126.COM
Institute of Big Data Science and Industry, Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education
Shanxi University
Taiyuan, 030006, PR China

Jiye Liang LJY@SXU.EDU.CN
Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education
Shanxi University
Taiyuan, 030006, PR China

Editor: Isabelle Guyon

Abstract

The existence of output noise will bring difficulties to supervised learning. Noise filtering, aiming to detect and remove polluted samples, is one of the main ways to deal with the noise on outputs. However, most of the filters are heuristic and could not explain the filtering influence on the generalization error (GE) bound. The hyper-parameters in various filters are specified manually or empirically, and they are usually unable to adapt to the data environment. The filter with an improper hyper-parameter may overclean, leading to a weak generalization ability. This paper proposes a unified framework of optimal sample selection (OSS) for the output noise filtering from the perspective of error bound. The covering distance filter (CDF) under the framework is presented to deal with noisy outputs in regression and ordinal classification problems. Firstly, two necessary and sufficient conditions for a fixed goodness of fit in regression are deduced from the perspective of GE bound. They provide the unified theoretical framework for determining the filtering effectiveness and optimizing the size of removed samples. The optimal sample size has the adaptability to the environmental changes in the sample size, the noise ratio, and noise variance. It offers a choice of tuning the hyper-parameter and could prevent filters from overcleansing. Meanwhile, the OSS framework can be integrated with any noise estimator

*. Corresponding author.

and produces a new filter. Then the covering interval is proposed to separate low-noise and high-noise samples, and the effectiveness is proved in regression. The covering distance is introduced as an unbiased estimator of high noises. Further, the CDF algorithm is designed by integrating the cover distance with the OSS framework. Finally, it is verified that the CDF not only recognizes noise labels correctly but also brings down the prediction errors on real apparent age data set. Experimental results on benchmark regression and ordinal classification data sets demonstrate that the CDF outperforms the state-of-the-art filters in terms of prediction ability, noise recognition, and efficiency.

Keywords: output noise, generalization error bound, optimal sample selection, covering distance filtering, supervised learning

1. Introduction

In real-world applications, the performance of the learning model highly depends on data quality. The output noise has a negative impact on data quality. It might result in an unreasonable model and inaccurate prediction if the noise is ignored. From the perspective of statistical learning theory, the bound of *generalization error*(GE) or risk can be expressed as an equation increasing with the empirical risk (Bartlett and Mendelson, 2007; Rigollet, 2007; Oneto et al., 2015; Zhang et al., 2017). However, the true empirical risk might be underestimated due to noisy outputs or labels in supervised learning. Then the GE bound will be raised to some extent.

The outputs are usually polluted due to insufficient information, inexperienced labelers, errors in encoding or communication processes, etc (Wang et al., 2018; Han et al., 2019). The presence of noise may lead to various consequences, such as weakening the generalization ability of models (Tian and Zhu, 2015; Han et al., 2019), increasing the complexity of models (Sluban et al., 2014; Segata et al., 2010), and interfering with feature selection (Shanab et al., 2012; Gerlach and Stamey, 2007). The noise on output is considered to be more important than that on input because there are many features but only one output (few outputs in multi-label learning) which has a decisive impact on learning (Frenay and Verleysen, 2014).The output noise denotes the mislabeling in classification, and it means that the real output has a deviation from the true output in regression.

Noise-robust models and noise filtering are dominant techniques in dealing with the output noise. The model which is robust to label noise can be constructed by robust losses, sample weighting, and ensemble methods (Patrini et al., 2017; Shu et al., 2019; Sabzevari et al., 2018). Deep learning and transfer learning are also verified to be effective in learning with label noises (Li et al., 2018; Lee et al., 2018). Researches indicate that many losses are not completely robust to label noise, and the performance of the noise-robust model is still influenced by output noise (Nettleton et al., 2010; Yao et al., 2018).

Noise filtering is a popular way to deal with output noise by removing noisy samples (Frenay and Verleysen, 2014). From the perspective of the output type, filtering algorithms are mainly designed for output noise in classification, and some of them are extended to the regression scenario. From the perspective of designing idea, most filters are based on nearest neighbor and ensemble learning. The main idea of a nearest neighbor-based filter is that a label is probably to be noisy if it is different from its neighbors' (Cao et al., 2012; Sáez et al., 2013). To obtain a more reliable detection result, ensemble-based filters employ base models to vote for each sample according to whether their predictions are consistent with

the real label (Khoshgoftaar and Rebours, 2007; Sluban et al., 2010; Yuan et al., 2018). In addition, there are a few filters based on model complexity (Gamberger et al., 1999; Sluban et al., 2014).

Although kinds of filters have been proposed to detect noisy samples, they might also make wrong recognitions and remove many noise-free samples (Frenay and Verleysen, 2014; Garcia et al., 2015). Then the original data distribution may be destroyed and the prediction ability of models trained on the filtered set might be weakened to some extent. Besides, most of the filters are heuristic and lack of theoretical foundations. Moreover, filters for regression have not yet been widely studied because the noise problem is more complex than that in classification. Specifically, the number of values to be predicted in classification is usually very low, whereas the output variable in regression is continuous, such that the number of possible values to predict is unlimited (Kordos and Blachnik, 2012; Kordos et al., 2013).

1.1 Related Work

Noise filtering approaches can be classified into nearest neighbor-based filter, ensemble-based filter and model complexity-based filter.

As k-nearest neighbor (kNN) classifier is sensitive to label noise (García et al., 2012), various nearest neighbor-based filters, such as edited nearest neighbor (ENN), all kNN (ANN) (Cao et al., 2012), and mutual nearest neighbors, were presented (Barandela and Gasca, 2000; Liu and Zhang, 2012). Samples misclassified by kNN is known as noises in ENN, and the noise recognition procedure is repeated for $k = 1, \cdots, K$ in ANN. A sample x_1 is a mutual nearest neighbor of x_2, if x_1 belongs to the k nearest neighbors of x_2, and x_2 is also one of the k nearest neighbors of x_1 at the same time. All samples with an empty MNN set are deleted from the original data set.

Ensemble-based filter employs various classifiers to vote for samples. The removing criterion has two choices: a majority vote and a consensus vote (Frenay and Verleysen, 2014). The majority vote classifies a sample as mislabeled if a majority of these classifiers misclassified it, and the consensus vote requires that all classifiers have misclassified it. Classification filtering (CF) may be the simplest among them. The cross-validation procedure is executed on a data set for a given model and misclassified samples are recognized as noises by CF. The noise is determined by multiple classifiers in majority voting filter (MVF) (Brodley and Friedl, 1999). Iterative-partitioning filter (IPF) partitions a data set into several subsets and removes samples by the ensemble criterion (Khoshgoftaar and Rebours, 2007). This process is repeated on the filtered data set until all the quantities of continuous three removing are less than a threshold. Samples are recognized as noises by high agreement random forest (HARF) if the predictions of all decision trees do not reach a high agreement level (Sluban et al., 2010). In order to improve the prediction performance of base classifier, iterative noise filter based on the fusion of classifiers (INFFC) builds base models on a data set with fewer noises and scores for all misclassified samples (Sáez et al., 2016). Samples with scores beyond a threshold will be deleted by INFFC. In probabilistic sampling filter, clean samples have more chances to be selected than noisy ones. The degree of cleanliness is measured by the label confidence (Yuan et al., 2018).

It is known that the complexity of a model is likely to be raised if noises are added to the training set. Conversely, it might become lower if partial noisy samples are removed.

Saturation-based filter (SF) exhaustively looks for examples which could make the maximum reduction of complexity (Gamberger et al., 1999). However, it takes a lot of time to complete the SF procedure because of the huge searching space and the complicated calculation of model complexity. Prune saturation filter simplifies SF by replacing the complexity indicator with the nodes number of random forest, and all decision trees have been pruned before saturation filtering (Sluban et al., 2014).

Above filters are empirically compared in relevant studies. The results from Sáez et al. (2013) indicate that there is a notable relationship between the characteristics of the data and the efficacy of filters. Li et al. (2016) shows that filters can significantly reduce the noise ratio and enhance the model's generalization ability. It confirms that CF, MVF, and IPF outperform ENN and ANN. Sluban et al. (2014) and Garcia et al. (2016) examined the performances of ensemble filters. The former's result shows ensemble filters usually have better precision than elementary filters. The latter studied many possible ensembles and found that the use of ensembles increases the predictive performance in the noise identification, especially for the ensemble of HARF and MVF. However, the ensemble filters are prone to overcleansing and have a higher computational cost than the elementary filter.

The filtering for regression gets less attention than that for classification. Inspired by the feature selection, a filter based on mutual information (MI) is presented to decide which samples should belong to the training data set (Guillen et al., 2010). But its performance is not satisfactory and it takes a huge amount of time. The ENN evolves into a new version, ENN for regression (RegENN), which is to remove any sample if its neighbors' output is far away from the prediction of a model trained on all samples except itself (Kordos et al., 2013). Another evolution is named as the discrete ENN (DiscENN) (Arnaiz-González et al., 2016). It transforms the continues variables into classes by the discretization of the numerical output variable, then the filter for classification is applied to the transformed data set.

Existing filters for classification or regression focus on the noise (value or probability) estimation by means of various data partitions, models, and ensemble strategies. However, two basic problems are often missing or solved intuitively. (1) The influence of filtering on GE bound. Many filters may be good at estimating the noise, but they did not explain the influence of filtering on the generalization ability. There is no theoretical guarantee that a filter could reduce the GE bound. (2) The adaptive sample selection problem. It mainly refers to how to regulate the number of kept or removed samples according to the noise environment. Specifically, the filtering thresholds are empirically selected by the simulated results in many filters, including HARF, INFFC, MI, and RegENN. Then the filtering with the suggested threshold may be unsuitable for a new noisy data set.

1.2 Summary of Contributions

From the perspective of generalization error (GE) bound, three essential problems need to be answered in output noise filtering.

1. Whether a filter works or whether it could reduce the GE bound?

2. How many samples should be removed so as to obtain the least GE bound?

3. Which samples should be removed?

The first two problems are often out of consideration or discussed empirically in related works. We propose a unified framework of optimal sample selection (OSS) for effectiveness determination and optimal sample selection in output noise filtering from the perspective of GE bound. A noise estimator and a novel filter are presented to deal with noisy outputs in regression and ordinal classification problems. In summary, the main contributions of this paper are as follows.

- For Problem 1, the decision rule is presented to decide whether a filter could reduce the GE bound. It implies that not all filters are beneficial to the generalization ability although they could reduce the noise level. The decision rule applies to any filter when the noise estimation and model errors are prepared.

- For Problem 2, the proposed OSS framework provides the optimal sample size for filtering so as to obtain the least GE bound. It implies that not all noisy samples need to be removed even though noises almost have been estimated accurately. Essentially, the optimal sample size is the result of a comprehensive balance of multiple factors, including the sample size, model errors, and the noise level. Hence the optimal sample size has the adaptability to environmental changes in the sample size, the noise ratio, and noise variance. Meanwhile, the OSS framework can be integrated with any noise estimator and then produces a new filter. It offers a choice of tuning the filtering threshold and could prevent filters from overcleansing.

- For Problem 3, the proposed OSS framework suggests that samples with larger noises should be removed first. Further, high-noise and low-noise samples can be broadly separated by the covering interval, and the covering distance (CD) provides a practical estimation of symmetric output noise. The characteristics of the CD estimator, including unbiasedness, absolute and relative deviations, are explored under popular noise distributions. Then the covering distance filtering (CDF) approach is proposed by integrating the CD with the OSS framework.

- It is empirically verified that the covering interval and the CD estimator are also applicable to asymmetric mixed noise. Experimental results indicate that the proposed CDF filter is effective in regression and ordinal classification problems, and it outperforms the state-of-the-art filters in terms of noise recognition, prediction ability, and efficiency. In the real problem of apparent age estimation, the CDF algorithm not only recognizes noisy age labels correctly but also brings down the prediction errors.

1.3 Organization

The rest of this paper is organized as follows. Section 2 describes the unified filtering framework in determining the effectiveness and optimizing the sample size from the perspective of GE bound in regression. The properties and applications are analyzed subsequently. In Section 3, the covering interval is introduced to separate low-noise samples from high-noise ones. The covering distance is proposed for estimating the output noise, and it is integrated with the OSS framework, generating the CDF filtering algorithm. All the proofs of Section 2 and Section 3 can be found in the appendix. In Section 4, the CDF filter is applied to a real apparent age data set in order to identify noisy age labels and improve the prediction

ability. Section 5 shows the results of numerical experiments on benchmark data sets in regression and ordinal classification, and Section 6 concludes.

2. Optimal Sample Selection Framework for Noise Filtering

A general structural regression problem can be denoted by $D = \{(x_i, y_i), i = 1, 2, \ldots, n\}$, where x_i, y_i are input and output of the i-th sample or instance, respectively. If output noise exists in the problem, y_i may be unequal to the true output y_i^0. Let $y = m(x)$ be a model trained on data set D.

Definition 1 (Noise) *The noise on the i-th sample*

$$e_i = y_i - y_i^0. \tag{1}$$

Definition 2 (Error) *The model error (or residual) on the i-th sample*

$$r_i = m(x_i) - y_i. \tag{2}$$

Let D_F be the filtered data set from D. The size of filtered data set n_F is less than n. $y = m_F(x)$ denotes the model trained on data set D_F.

Definition 3 (Relative size) *The relative size of the filtered data set to the initial one is defined as*

$$\rho = \frac{n_F}{n}. \tag{3}$$

If all noises have been well estimated, they can be sorted by the absolute values. An intuitive idea is to remove samples whose noises are over a threshold. Indeed, the threshold corresponds to a relative size. In another word, the filtered data set with relative size ρ consists of $n\rho$ samples with (estimated) noises less than the threshold.

2.1 Main Results for Sample Selection

2.1.1 Effective Noise Filtering

This subsection analyzes the necessary and sufficient condition of *effective noise filtering* which brings down the generalization error (GE) bound by removing noisy samples.

Lemma 1 *(Cherkassky et al., 1999) For regression problems with squared loss, the following GE bound holds with probability $1 - \eta$:*

$$\mathcal{R}(m, D) \leq \mathcal{R}_{emp}(m, D) \cdot \epsilon(h, n, \eta), \tag{4}$$

where $\mathcal{R}(m, D)$ is the prediction risk of leaner m trained on data set D, the empirical risk

$$\mathcal{R}_{emp}(m, D) = \frac{1}{n} \sum_{x_i \in D} [m(x_i) - y_i^0]^2, \tag{5}$$

and

$$\epsilon(h, n, \eta) = \left(1 - \sqrt{\frac{h\left(\ln \frac{n}{h} + 1\right) - \ln \eta}{n}}\right)_+^{-1} \tag{6}$$

is a function about the VC-dimension h, the sample size n, and the probability η.

Let $\epsilon(D) = \epsilon(h, n, \eta)$ and $\epsilon(D_F) = \epsilon(h, n_F, \eta)$. The following theorem provides a necessary and sufficient condition of effective noise filtering, and it can serve as a decision rule for effective noise filtering.

Theorem 1 (A necessary and sufficient condition for effective noise filtering) *The model trained on D_F has a lower GE bound than that on D if and only if*

$$\frac{\mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D(e_i^2)} < \frac{\epsilon(D)}{\epsilon(D_F)}(1 + C) - C \tag{7}$$

holds for a fixed goodness of fit ($1 - \frac{\sum_i [m(x_i) - y_i]^2}{\sum_i (y_i - \sum y_i/n)^2}$), where $\mathbb{E}(\cdot)$ denotes expectation function, the coefficient

$$C = \frac{\mathbb{E}_D(r_i^2)}{\mathbb{E}_D(e_i^2)} > 0, \tag{8}$$

and

$$\epsilon(D_F) = \epsilon(h, n\rho, \eta) = \left(1 - \sqrt{\frac{h\left(\ln \frac{n\rho}{h} + 1\right) - \ln \eta}{n\rho}}\right)_+^{-1}. \tag{9}$$

It means that the ratio of noise levels $\frac{\mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D(e_i^2)}$ is required to be under a corresponding bound for effective noise filtering. Conversely, the model trained on the filtered data set will have a higher risk. The proof of Theorem 1 can be found in Appendix A.1.

For simplicity, two definitions are given based on (7).

Definition 4 (Ratio of noise levels and its bound) *The ratio of noise levels with respect to the relative size ρ is defined as*

$$T(\rho) = \frac{\mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D(e_i^2)}, \tag{10}$$

and the (upper) bound of $T(\rho)$ is defined by

$$\mathcal{B}_T(\rho) = \frac{\epsilon(D)}{\epsilon(D_F)}(1 + C) - C, \tag{11}$$

where $\rho = \#\{D_F\}/\#\{D\} = n_F/n$.

According to Theorem 1, the relationship between the ratio of noise levels and its bound decides whether a filter works or whether it could reduce the GE bound.

2.1.2 Optimal Sample Selection

This subsection proposes a necessary and sufficient condition for the optimal sample selection which minimizes the GE bound of the model trained on filtered data set.

Theorem 2 (Optimal sample selection for effective noise filtering) *For a fixed goodness of fit in regression,*

$$\min \mathcal{R}_{emp}(m_F, D_F) \cdot \epsilon(D_F) \Leftrightarrow \max \left[\mathcal{B}_T(\rho) - T(\rho)\right] \cdot \epsilon(D_F), \tag{12}$$

where the three components are defined in (9), (10) and (11).

It indicates that the GE bound after filtering depends on the *margin* between $T(\rho)$ and its bound $\mathcal{B}_T(\rho)$ and the coefficient $\epsilon(D_F)$. The proof of Theorem 2 can be found in Appendix A.2. Note that the error bound in Lemma 1 depends on the probability η, i.e., the probability η is arbitrary. And so, the conclusions in Theorems 1 and 2 hold for any value of η. By (12), the *objective function* of effective noise filtering becomes

$$\mathcal{F}(\rho) = [\mathcal{B}_T(\rho) - T(\rho)] \cdot \epsilon(D_F), \tag{13}$$

and the *optimal relative size*

$$\rho^* = \arg\max \mathcal{F}(\rho) = \arg\max [\mathcal{B}_T(\rho) - T(\rho)] \cdot \epsilon(D_F). \tag{14}$$

According to Theorem 2, the number of kept samples should maximize the objective function $\mathcal{F}(\rho)$ so as to obtain the least GE bound.

In Theorem 2, only the component $T(\rho)$ is related to the noise in D_F, and a smaller $T(\rho)$ is desired by the objective function. From (10), retaining low-noise samples means a small $T(\rho)$. Therefore, high-noise samples should be removed so as to obtain a smaller $T(\rho)$ and a lower GE bound for a given relative size.

Remark 1 *The goodness of fit is assumed to be fixed in both Theorems 1 and 2. However, it is expected to be slightly enlarged after filtering as some hard-to-learn (noisy or specific) samples are removed in reality. In another word, the model error is likely to be reduced by the filtering. Strictly speaking, this reduction will bring a small positive shift to $\mathcal{B}_T(\rho)$, i.e., $\mathcal{B}_T(\rho)$ in (11) is an underestimated bound for $T(\rho)$ if the assumption is removed in the proof of Theorem 1(see Proof A.1).*

The true noise of retained samples is not always less than that of the removed ones due to the deviation of noise estimation. It implies that the estimated $T(\rho)$ is usually optimistic (underestimated) in reality. Then the deviation of $\mathcal{B}_T(\rho)$ will be counteracted by that of $T(\rho)$ to some extent. In addition, the robustness of some regression models might reduce the variation of the goodness of fit in filtering, especially for large-scale data sets. Overall, the violation of the assumption about the goodness of fit is considered to have no essential influence on the theoretical results.

Theorem 1 means that not all filters are beneficial to the error bound although they can reduce noise level ($T(\rho) < \mathcal{B}_T(\rho) < 1$). If there does not exist an effective filtering for a data set, the optimal relative size ρ^* will be equal to 1, i.e., no sample needs to be removed. The optimal relative size is determined by the three components related to the sample size, model errors, and the noise level. Hence the optimal sample size is the result of a comprehensive balance of multiple factors (not just minimize the noise level). This implies that not all noisy samples need to be removed even though noises almost have been estimated accurately.

Theorems 1 and 2 provide theoretical foundations for the determination of effective filtering and the optimal sample selection. They are directly available for noise filtering when the noise estimation and model errors are ready. From a broad perspective, the two theorems can be integrated with any noise estimator. In another word, they provide a unified framework, named as the optimal sample selection (OSS) framework, for the output noise filtering in regression. In addition, the effectiveness of the OSS framework partially depends on the accuracy of noise estimation in reality.

2.2 Properties of the three components

This subsection studies the properties of the three components in objective function $\mathcal{F}(\rho)$. They are the basis of the instructions of the proposed framework in Subsection 2.3.

2.2.1 Properties of $T(\rho)$

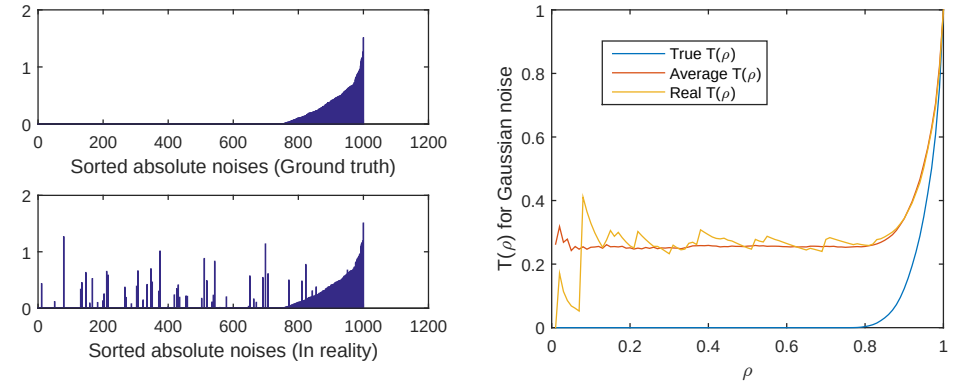
By (10), the $T(\rho)$ curve depends on the noise and the sequence of sample removing. Assume that the true noises are known in the ideal situation, then all samples can be sorted by their absolute noise values in ascending order (the last is removed first). However, a few samples may be in the wrong order in reality due to the deviation in noise estimation. So $T(\rho)$ can be calculated in different ways.

- The **true** $T(\rho)$ from the God's-eye view. All true noises are known and can be sorted in ascending order completely. It means that the true noise value and completely correct sequence are employed in this scheme.
- The **real** $T(\rho)$ in a trial. The removing sequence based on the estimated noises should be partially consistent with the truth. The true noise value is adopted to validate the effectiveness of the proposed theorems. In other words, the partially correct sequence based on noise estimation and the true noise value are employed in this scheme.
- The real **average** $T(\rho)$ in multiple trials. It is obtained by averaging the real $T(\rho)$ values over many trials in order to analyze the property in the sense of expectation.
- The **estimated** $T(\rho)$ in experiment. The estimated noise values and the corresponding sequence (partially correct) have to be used in real filtering algorithms.

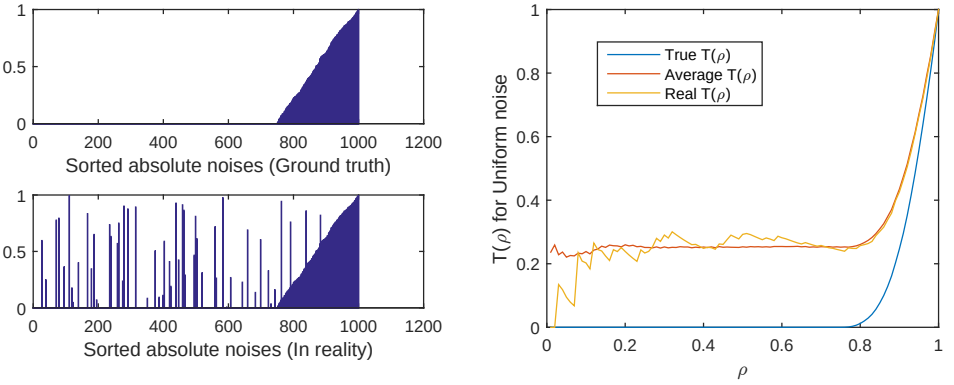
It is easy to find that both the true and estimated $T(\rho)$ monotonously increase with ρ. The real $T(\rho)$ in a trial should have a larger variation than the average $T(\rho)$. The first three kinds of $T(\rho)$ are analyzed in this section, and the last is utilized in the real filtering algorithm in Section 3.

For simplicity, the noise for exploring the property of $T(\rho)$ is randomly generated from a predefined distribution. Indeed, it is independent of any data set and model in this subsection. Simulated results of $T(\rho)$ are shown in Figure 1. It can be observed that the true $T(\rho)$ is equal to 0 for $\rho < 1-NR$ and monotonously increases with ρ when $\rho > 1-NR$, where NR denotes the noise ratio. The average $T(\rho)$ is similar to the true version in shape, but the former is larger. In addition, the average $T(\rho)$ is more smooth than the real $T(\rho)$. It is obvious that the true $T(\rho)$ is less than the other two versions for any ρ. Further, the averaged $T(\rho)$ is an increasing and convex function about ρ for the predefined noise distributions. The true $T(\rho)$ curves in two sub-figures are similar in terms of monotonicity and concavity. The difference in noise distribution is reflected by the slope of the $T(\rho)$ curve.

The true $T(\rho)$ may be affected by the noise levels, including the noise ratio NR and the noise variance σ^2, for a given type of noise distribution (the noise expectation is usually assumed to be zero). Note that σ^2 is a general variance notation but not specialized for the Gaussian distribution. The true $T(\rho)$ is theoretically analyzed under the assumption that the sample size $n \to +\infty$ so as to describe the properties in the form of partial derivative. The proportion relationship implied by the partial derivative holds for any sample size.



(a) $T(\rho)$ for Gaussian noises $N(0, 0.5^2)$



(b) $T(\rho)$ for uniform noises $U(-1, 1)$

Figure 1: The ratio of noise levels $T(\rho)$ is simulated in two situations. In the ideal (ground truth) situation, all (100%) noises are in the correct order, and the sorted noises are displayed in the top left of each sub-figure. In the real situation, most (75%) of the noises are in the correct order and the others (25%) are randomly permuted. The real order for sorted noises are displayed in the bottom left of each sub-figure. The noises in the two sub-figures are from the Gaussian distribution $N(0, 0.5^2)$ and the uniform distribution $U(-1, 1)$, respectively. The *noise ratio* (number of noisy samples/n) is set to be $NR = 25\%$ and the sample size $n = 1000$. The real $T(\rho)$ is the result in a trial and the average $T(\rho)$ is from 100 independent trials.

Property 1 *The true $T(\rho)$ has the following characteristics:*

(1)

$$T(\rho < 1 - NR) = 0, T(\rho > 1 - NR) > 0. \tag{15}$$

(2)

$$\frac{\partial T(\rho)}{\partial \rho} \geq 0. \tag{16}$$

(3) If there do not exist two equal noises,

$$\frac{\partial^2 T(\rho)}{\partial \rho^2} > 0. \tag{17}$$

(4) For a given noise distribution,

$$\frac{\partial T(\rho)}{\partial NR} \geq 0. \tag{18}$$

(5) If the noise is from a symmetric Gaussian(or uniform, or Laplace) distribution,

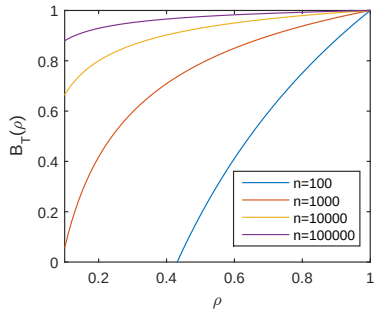
$$\frac{\partial T(\rho)}{\partial \sigma} = 0. \tag{19}$$

The proof of Property 1 can be found in Appendix A.3. The first two properties can be observed from Figure 1. Equation (16) means that the true $T(\rho)$ (non-strictly) monotonously increases with ρ. Equation (17) shows the convexity of $T(\rho)$. Actually, the true $T(\rho)$ might be non-convex when there are many equal noises. Equation (18) indicates that $T(\rho)$ increases with the noise ratio NR. Although the noise level is related to the noise ratio and the noise variance, Equation (19) shows that $T(\rho)$ is independent of the variance for usual symmetric noise distributions. The reason is that both the numerator and denominator in the definition of $T(\rho)$ (Equation 10) are proportional to the variance and the variance can be eliminated at the same time. Note that (19) might not hold for some asymmetric noise distributions.

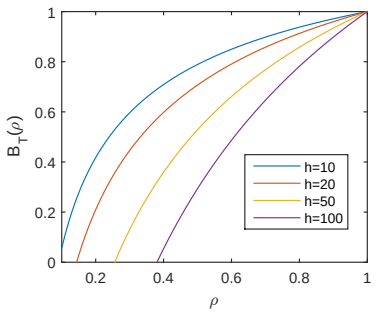
As the average $T(\rho)$ is similar to the true $T(\rho)$ in shape, it should have the same proportional relationship about the factors including ρ, NR, and σ in most cases. In addition, the estimated $T(\rho)$ associates with noise estimation apart from the noise level.

2.2.2 Properties of $\mathcal{B}_T(\rho)$

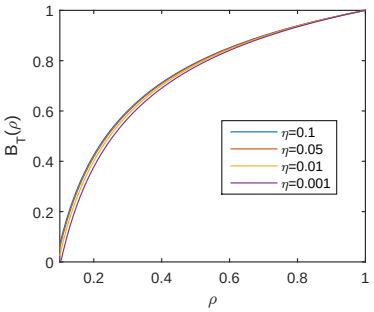
From (11), the upper bound $\mathcal{B}_T(\rho)$ is related to the relative size ρ, the sample size n, the VC-dimension h, the probability η, and the coefficient C. These relationships are shown in Figure 2. It can be observed that $\mathcal{B}_T(\rho)$ increases with n, η and decreases with h, C. There is no significant difference among the considered η values. Besides, $\mathcal{B}_T(\rho)$ monotonously increases with ρ for other fixed factors (proved in Equation 21). The simulated results indicate that $\mathcal{B}_T(\rho)$ is a concave function about ρ. According to Theorem 1, $\mathcal{B}_T(\rho = 0.6|n = 1000, h = 10, \eta = 0.05, C = 1) = 0.85$ indicates that the noise level should be reduced by more that 15% $(1 - 0.85)$ in the filtering of removing 40% $(1 - \rho)$ of the samples so as to obtain a lower GE bound with probability 0.95 $(1 - \eta)$.



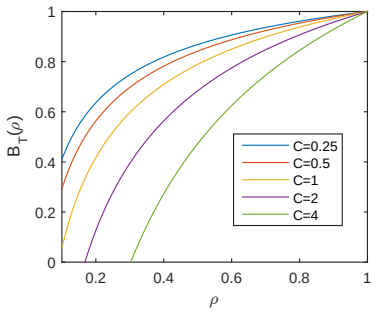
(a) Influence of the sample size n



(b) Influence of VC-dimension h



(c) Influence of the probability η



(d) Influence of the coefficient C

Figure 2: Influences on the upper bound $\mathcal{B}_T(\rho)$ are simulated with the following settings. Sub-figure (a): $n = 10^2, 10^3, 10^4, 10^5, h = 10, \eta = 0.05, C = 1$; sub-figure (b): $n = 1000, h = 10, 20, 50, 100, \eta = 0.05, C = 1$; sub-figure (c): $n = 1000, h = 10, \eta = 0.1, 0.05, 0.01, 0.001, C = 1$; sub-figure (d): $n = 1000, h = 10, \eta = 0.05, C = 0.25, 0.5, 1, 2, 4$. The independent variable ρ is in [0.1,1] for each sub-figure.

Property 2 *$\mathcal{B}_T(\rho)$ has the following characteristics:*

(1)

$$\frac{\partial \mathcal{B}_T(\rho)}{\partial C} < 0. \tag{20}$$

(2)

$$\frac{\partial \mathcal{B}_T(\rho)}{\partial \rho} > 0. \tag{21}$$

(3) When $n \gg h$,

$$\frac{\partial \mathcal{B}_T(\rho)}{\partial n} > 0. \tag{22}$$

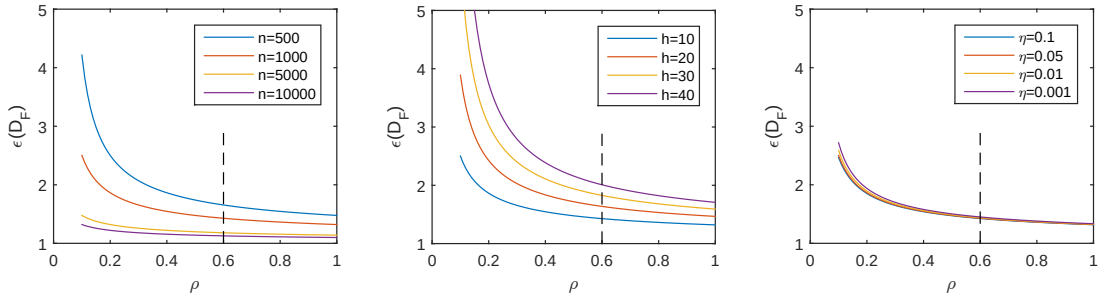
(4)

$$\frac{\partial^2 \mathcal{B}_T(\rho)}{\partial \rho \partial C} = \frac{\partial^2 \mathcal{B}_T(\rho)}{\partial C \partial \rho} > 0. \tag{23}$$

The proof can be found in Appendix A.4. Note that C is inversely proportional to the initial noise level $\mathbb{E}_D(e_i^2)$ in (8), Equation (20) implies that $\mathcal{B}_T(\rho)$ increases with $\mathbb{E}_D(e_i^2)$ for any $\rho < 1$. For example, $\mathcal{B}_T(\rho = 0.8|\mathbb{E}_D(e_i^2) = 2) > \mathcal{B}_T(\rho = 0.8|\mathbb{E}_D(e_i^2) = 1)$. Equation (23) means that the slope of $\mathcal{B}_T(\rho)$ with regard to ρ will increase with C or decrease with $\mathbb{E}_D(e_i^2)$. Equations (21) and (23) indicate that the bound $\mathcal{B}_T(\rho)$ has a smaller increase for the larger ρ when the initial noise level is enlarged. For example, assume that $\mathbb{E}_D(r_i^2) = 1$ and $\mathbb{E}_D(e_i^2)$ is enlarged from 0.5 to 1, then C will be reduced from 2 to 1 for a fixed goodness of fit. It can be deduced that $0 < [\mathcal{B}_T(\rho|\mathbb{E}_D(e_i^2) = 1) - \mathcal{B}_T(\rho|\mathbb{E}_D(e_i^2) = 0.5)]|_{\rho=0.8} < [\mathcal{B}_T(\rho|\mathbb{E}_D(e_i^2) = 1) - \mathcal{B}_T(\rho|\mathbb{E}_D(e_i^2) = 0.5)]|_{\rho=0.6}$, i.e., $0 < [\mathcal{B}_T(0.8) - \mathcal{B}_T(0.6)]|_{C=1} < [\mathcal{B}_T(0.8) - \mathcal{B}_T(0.6)]|_{C=2}$. These properties can be clearly observed from Figure 2.

2.2.3 Properties of $\epsilon(D_F)$

Since $\epsilon(D_F) = \epsilon(h, n\rho, \eta)$ in (9), $\epsilon(D_F)$ is related to the relative size ρ, the sample size n, VC-dimension h, and the probability η. The relationships between $\epsilon(D_F)$ and its factors are shown in Figure 3.



(a) Influence of the sample size n (b) Influence of VC-dimension h (c) Influence of the probability η

Figure 3: Influences on $\epsilon(D_F)$ are simulated with the following settings. Sub-figure (a): $n = 500, 1000, 5000, 10000, h = 10, \eta = 0.05$; sub-figure (b): $n = 1000, h = 10, 20, 30, 40, \eta = 0.05$; sub-figure (c): $n = 1000, h = 10, \eta = 0.1, 0.05, 0.01, 0.001$.

It can be observed that $\epsilon(D_F)$ increases with h and decreases with n, η for fixed ρ. There is no significant difference among the considered η values. Besides, $\epsilon(D_F)$ monotonously decreases with ρ when other factors are fixed. More importantly, the $\epsilon(D_F)$ curve is almost flat when $\rho \in [0.6, 1]$. It implies that $\epsilon(D_F)$ may have a smaller impact on the objective function $\mathcal{F}(\rho)$ than the margin between $T(\rho)$ and $\mathcal{B}_T(\rho)$ when the relative size ρ is large enough.

Property 3 *$\epsilon(D_F)$ has the following characteristics:*

(1) When $n \gg h$,

$$\frac{\partial \epsilon(D_F)}{\partial n} < 0. \tag{24}$$

(2)

$$\frac{\partial \epsilon(D_F)}{\partial \rho} < 0. \tag{25}$$

(3)

$$\frac{\partial \epsilon(D_F)}{\partial h} > 0. \tag{26}$$

(4)

$$\frac{\partial \epsilon(D_F)}{\partial \eta} < 0. \tag{27}$$

These relationships can be easily obtained by (9), and the proof is omitted. In addition, the relations are verified by Figure 3.

The proposed properties are summarized in Table 1. $T(\rho)$ refers to the true version here. The relationships between the component and its factors are denoted by different signals. The correspondences are as follows: $\uparrow$-proportional relationship, $\downarrow$-inversely-proportional relationship, $\nearrow$-insignificant proportional relationship, $\searrow$-insignificant inversely-proportional relationship, $\times$-independent. The signal with a star means the relationship is just supported by simulation results, and the others have been proved.

			Noise				
	ρ	n	NR	σ	C	h	η
$T(\rho)$	$\uparrow$	$\times$	$\uparrow$	$\times$	$\times$	$\times$	$\times$
$\mathcal{B}_T(\rho)$	$\uparrow$	$\uparrow$	$\uparrow$	$\uparrow$	$\downarrow$	$\downarrow^*$	$\nearrow^*$
$\epsilon(D_F)$	$\searrow$	$\downarrow$	$\times$	$\times$	$\times$	$\uparrow$	$\searrow$

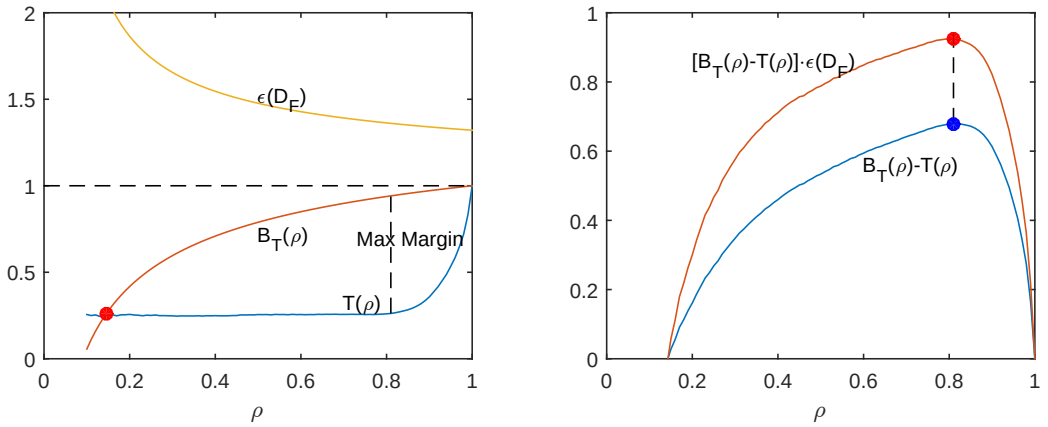
Table 1: Summary of the properties

2.3 Practical Instructions for OSS Framework

This part describes practical instructions for the OSS framework, and explains its adaptability based on previous simulations and properties.

2.3.1 Applications of OSS Framework

Figure 4 shows the simulation results of relevant functions for determining the effective noise filtering and optimizing the relative size ρ. From Figure 4(a), the filtering is effective (gets a lower GE bound) when the relative size is on the right of the red dot ($\rho > 0.145$). From Figure 4(b), the maximum of the objective function (red dot) is on the upper left of the maximum of the margin curve (blue dot) since $\epsilon(D_F)$ decreases with ρ and $\epsilon(D_F) > 1$. Besides, the two dots have very close horizontal coordinates, and it means the optimal relative size mainly depends on the margin.



(a) Discrimination of effective noise filtering

(b) Optimization of the relative size

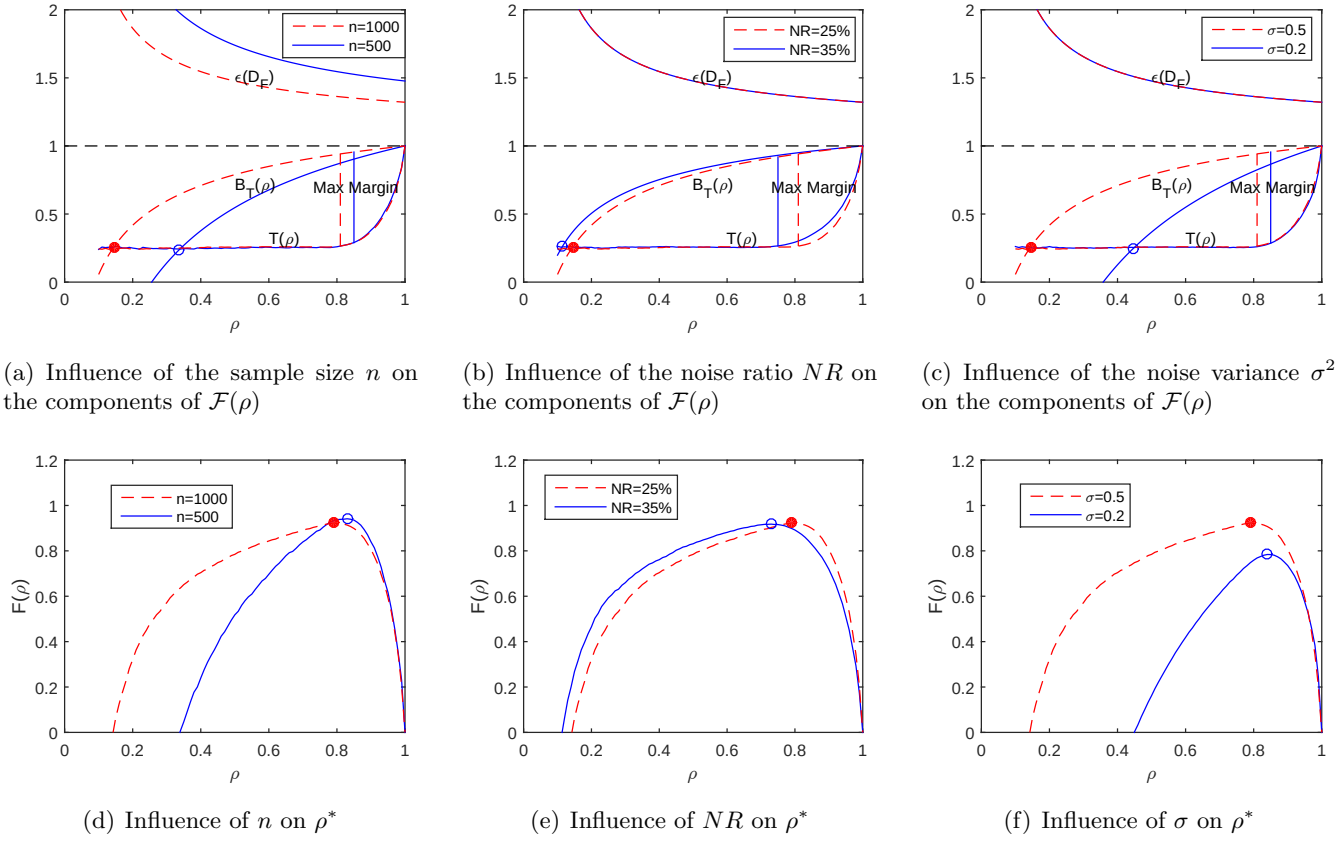
Figure 4: The sample selection is simulated based on the proposed OSS framework. The left sub-figure shows the three components in the objective function. Note that the average $T(\rho)$ is implemented in the same way as Figure 1. The maximum margin between $T(\rho)$ and $\mathcal{B}_T(\rho)$ is marked by a black dashed line, and a red dot is plotted at the crossing position. The right sub-figure shows the margin curve (blue) and the objective function (red), and their maximums are marked by solid dots. A vertical dashed line is added at the maximum margin. The parameter setting for both sub-figures is: the sample size $n = 1000$, noise ratio $NR = 25\%$, noise distribution $N(0, 0.5^2)$, VC-dimension $h = 10$, the probability $\eta = 0.05$, the coefficient $C = 1$.

2.3.2 Adaptability of OSS Framework from a Qualitative Perspective

Intuitively speaking, the optimal relative size ρ^* in (14) should have the adaptability to the data quality and noise level, so we explored the influences of the sample size n and the noise levels, including the noise ratio NR and noise variance σ^2 (for a given type of noise distribution), on ρ^*.

The simulation results are shown in Figure 5. The paired top and bottom sub-figures are analyzed together in order to make a clear explanation.

(1) In the left pair (Figure 5(a) and (d)), the first new setting has a smaller sample size than the baseline setting ($n = 500 < 1000$). From the baseline setting to the new setting, $\mathcal{B}_T(\rho)$ has a reduction due to (22). More importantly, sub-figure (a) indicates that the smaller the ρ value is, the more $\mathcal{B}_T(\rho)$ is reduced. As a result, the relative size with the maximum margin becomes larger when the sample size is reduced. So does the relative size at the crossing position as shown in Figure 5(a). Although the new setting for comparison has a larger $\epsilon(D_F)$ due to (24), it has an insignificant influence on the maximum of the objective function. Thus the optimal relative size in Figure 5(d) also has a growing tendency. Overall, the optimal relative size ρ^* decreases with the sample size n in the sense of expectation when the other conditions are fixed.



(a) Influence of the sample size n on the components of $\mathcal{F}(\rho)$

(b) Influence of the noise ratio NR on the components of $\mathcal{F}(\rho)$

(c) Influence of the noise variance σ^2 on the components of $\mathcal{F}(\rho)$

(d) Influence of n on ρ^*

(e) Influence of NR on ρ^*

(f) Influence of σ on ρ^*

Figure 5: Influences of the sample size n, noise ratio NR, and noise variance σ^2 on the three components of the objective function $\mathcal{F}(\rho)$ and on the optimal relative size ρ^* are displayed. All components of $\mathcal{F}(\rho)$ are shown in the top sub-figures. The bottom sub-figures show corresponding objective functions, and the maximums are marked with dots. Note that component $T(\rho)$ is implemented in the average version similar to Figure 1, and it can be considered as a derivable function about ρ in the sense of expectation. Three new settings are compared with a baseline setting. The results under the baseline setting are denoted by the red dashed lines, and those with the new settings for comparison correspond to the blue solid lines. The baseline setting is: $n = 1000$, $NR = 25\%$, the noise variance $\sigma^2 = 0.5^2$ (Gaussian distribution), $h = 10$, $\eta = 0.05$, $C = 1$. The new settings for comparison are as follows. The left sub-figures: $n = 500$; the middle sub-figures: $NR = 35\%$, $C = 25\%/35\% = 5/7$; the right sub-figures: $\sigma^2 = 0.2^2$, $C = 0.5^2/0.2^2 = 25/4$. Since C is inversely proportional to the noise level $\mathbb{E}_D(e_i^2)$, it is adjusted according to the noise levels of the new settings. Omitted variables are the same as those of the baseline setting.

(2) In the middle pair (Figure 5(b) and (e)), the second new setting has a larger noise ratio than the baseline setting ($NR = 35\% > 25\%$). By (8), the C value should become smaller under the new setting. It can be observed from Figure 5(b) that both $\mathcal{B}_T(\rho)$ and $T(\rho)$ are affected by the change of the noise ratio. $\mathcal{B}_T(\rho)$ is larger under the new setting because of (20). Moreover, Equation (23) implies that the smaller the ρ values is, the more the increment of $\mathcal{B}_T(\rho)$ is. The average $T(\rho)$ function with the new setting is larger than or equal to that with the baseline setting because of (18). The final result is that both the relative size at the maximum margin and the optimal relative size become smaller from the baseline setting to the new setting. Overall, the optimal relative size ρ^* decreases with the noise ratio in the sense of expectation when the other conditions are fixed.

(3) In the right pair (Figure 5(c) and (f)), the third new setting has a smaller noise variance than the baseline setting ($\sigma^2 = 0.2^2 < 0.5^2$). For a given type of noise distribution, both $T(\rho)$ and $\epsilon(D_F)$ are not affected by the variance due to Equations (19) and (9). Whereas $\mathcal{B}_T(\rho)$ becomes smaller under the new setting because of Equations (20) and (8). Moreover, both Equation (23) and Figure 5(c) support that the reduction of $\mathcal{B}_T(\rho)$ is more evident for a smaller ρ value. The final result is that both the relative size at the maximum margin and the optimal relative size become larger when the noise variance is reduced. Overall, the optimal relative size ρ^* decreases with the noise variance in the sense of expectation when the other conditions are fixed. In addition, the crossing point for determining effective noise filtering has the same tendency as ρ^*.

The following property provides a strict description of the relationship between the optimal relative size and noise variance.

Property 4 (Adaptability of ρ^* to the noise variance) *Assume the noise is from a symmetric Gaussian (or Laplace, or uniform) distribution. Then*

$$\frac{\partial \rho^*}{\partial(\sigma^2)} < 0 \tag{28}$$

holds for a fixed goodness of fit and the true $T(\rho)$, where σ^2 is the noise variance.

The proof can be found in Appendix A.5. It indicates that the optimal relative size should be reduced, or more samples could be removed, when the noise variance is enlarged. This is consistent with the result in Figure 5(c) and (f). Actually, they provide the same suggestion in tuning the relative size from the theoretical and simulated perspectives, respectively. They have the same assumptions including the fixed goodness of fit and the symmetry of the noise distribution. The difference between Figure 5 and Property 4 lies in the version of $T(\rho)$. The former utilizes the average $T(\rho)$, while Property 4 adopts the true $T(\rho)$. The main reason for the same suggestion is that the average $T(\rho)$ and the true version have similar properties.

In sum, the optimal relative size ρ^* decreases with the sample size, the noise ratio, and noise variance in the sense of expectation when the other conditions are fixed. So do the relative size at the crossing position and the one with the maximum margin. In another word, more samples are allowed to be removed when the sample size and the noise level

(NR and σ^2) are large enough. On the contrary, more samples should be retained to avoid overcleansing. This is accordant with our intuition. Therefore, the proposed OSS framework has the adaptability to the data quality and noise level.

3. Covering Distance Filtering under OSS Framework

Two proposals, i.e. the covering interval and covering distance, are presented to distinguish noisy samples in regression. A novel filtering algorithm is designed by integrating them with the proposed OSS framework.

3.1 Covering Interval: a Selector for Low-noise Samples

Definition 5 (Covering interval) *$[u, v]$ is a covering interval of the true output y_i^0 if $y_i^0 \in [u, v]$. The center of the interval $c = \frac{u+v}{2}$, and the radius $r = \frac{v-u}{2}$.*

Low-noise samples can be identified by the covering interval according to the rule: falling in the covering interval corresponds to a smaller noise, and getting out of the interval indicates a larger noise. This is supported by the following propositions.

Let $F(e|\mu, \sigma), f(e)$ be the cumulative distribution function (CumDF) and probability density function (PDF) of the noise e, respectively. μ denotes the mean and σ^2 is the variance. Note that $F(e|\mu, \sigma)$ is a general notation for any distribution but not specialized for the Gaussian distribution.

Proposition 1 (A necessary condition for low-noise samples) *Assume that the noise e_i on y_i^0 is from a symmetric distribution with CumDF $F(e|\mu = 0, \sigma)$. $e^{(1)}, e^{(2)}$ are two sets of noises with variances σ_1^2, σ_2^2 on y_i^0. $[u, v]$ is any covering interval of y_i^0. If $\frac{\partial F(e|\mu=0,\sigma)}{\partial \sigma} < 0$ for $e > 0$ and $\sigma_1^2 < \sigma_2^2$, then*

$$\mathbb{P}\{y_i \in [u, v] \big| e^{(1)}\} > \mathbb{P}\{y_i \in [u, v] \big| e^{(2)}\}, \tag{29}$$

where $\mathbb{P}(\cdot)$ denotes the probability function.

Corollary 1 *Assume that the noise e_i on y_i^0 is from a Gaussian(or uniform, or Laplace) distribution with the CumDF $F(e|\mu = 0, \sigma)$. $e^{(1)}, e^{(2)}$ are two sets of noises with variances σ_1^2, σ_2^2 on y_i^0. $[u, v]$ is any covering interval of y_i^0. Then (29) holds for $\sigma_1^2 < \sigma_2^2$.*

Proposition 1 means that the samples with low noises are more likely to fall in the covering interval than those with large noises. Corollary 1 shows that the conclusion still holds for usual distributions at fewer preconditions. Their proofs can be found in Appendix A.6 and A.7.

Proposition 2 (A sufficient condition for low-noise samples) *Assume that the noise e on the true output is from a symmetric distribution with the PDF $f(e)$, i.e. $f(-e) = f(e)$. $[u, v]$ is any covering interval of y_i^0. $e \in \{e_i, i = 1, 2, \cdots, n\}$. Then*

$$\mathbb{E}(|e_i|^p \big| y_i^0 + e_i \in [u, v]) < \mathbb{E}(|e_i|^p \big| y_i^0 + e_i \notin [u, v]) \tag{30}$$

holds for any $p \in \mathbb{N}^+$.

The proof can be found in Appendix A.8. It means that the samples within any covering interval have a lower average noise than those out of the interval. Proposition 2 is applicable to some special symmetric noise distributions, including the zero-mean Gaussian, uniform, and Laplace distributions. The conclusion is still true if the noise distribution is a mixture of multiple symmetric distributions.

The covering interval is required for distinguishing between the low noise and high noise. Unfortunately, it is difficult to decide whether a given interval is the covering interval since the true output is usually unknown in reality. It is helpful if we can deduce that a specific interval covers the true output with a relatively large probability. Actually, this kind of covering interval can be prepared by means of model predictions.

Definition 6 *The prediction interval of J base models $\{y = m_j(x), j = 1, 2, \cdots, J\}$ for y_i is defined as*

$$[u_i, v_i] = [\min_j m_j(x_i), \max_j m_j(x_i)]. \tag{31}$$

Note that whether the above interval covers the true value is unknown yet.

If the base models $\{m_j(x), j = 1, 2, \cdots, J\}$ are trained on different data subsets, we can assume the independence of $m_j(x)$. Also, the events $y_i^0 < m_j(x_i)$ for different j values are independent. Similarly, the independence of $y_i^0 > m_j(x_i)$ with different j values holds, too. According to the principle of indifference (Peters, 2014), we can assume $\mathbb{P}\{m_j(x_i) < y_i^0\} = \mathbb{P}\{m_j(x_i) > y_i^0\} = 1/2$. By the above independence and indifference, the *covering probability*

$$\begin{aligned}
\mathbb{P}\{y_i^0 \in [u_i, v_i]\} &= \mathbb{P}\{\min_j m_j(x_i) \le y_i^0 \le \max_j m_j(x_i)\} \\
&= 1 - \mathbb{P}\{y_i^0 < \min_j m_j(x_i)\} - \mathbb{P}\{y_i^0 > \max_j m_j(x_i)\} \\
&= 1 - \mathbb{P}\{y_i^0 < m_j(x_i), \forall j\} - \mathbb{P}\{y_i^0 > m_j(x_i), \forall j\} \\
&= 1 - \prod_{j=1}^{J} \mathbb{P}\{y_i^0 < m_j(x_i)\} - \prod_{j=1}^{J} \mathbb{P}\{y_i^0 > m_j(x_i)\} \\
&= 1 - \prod_{j=1}^{J} 1/2 - \prod_{j=1}^{J} 1/2 \\
&= 1 - 2^{1-J}.
\end{aligned} \tag{32}$$

It indicates that the larger J produces a higher covering probability. However, increasing J will enlarge the interval radius, and then raise the deviation of noise estimation (see Property 6). In order to obtain a good balance, the parameter J is selected in a trade-off situation when the uncovering probability is around the popular significant levels, such as 0.1, 0.05 and 0.01. As $\mathbb{P}\{y_i^0 \in [u_i, v_i]\} = 0.9 \Rightarrow J \approx 4.3$ and $\mathbb{P}\{y_i^0 \in [u_i, v_i]\} = 0.99 \Rightarrow J \approx 7.6$, $J = 5, 6, 7$ is considered in the construction of covering interval.

Considering that the independence is required, the model predictions are generated in the *subsets* scheme in reality. The original data set is randomly partitioned into J subsets. Then the regression model is trained on each subset and tested on the whole data set.

3.2 Covering Distance: an Approach of Noise Estimation

The covering distance (CD) is proposed to estimate the absolute noise. On the basis of the noise estimation, a noisy data set can be filtered under the proposed OSS framework.

3.2.1 Definition of the Covering Distance

Definition 7 (Covering distance, CD) *The covering distance from y_i to the covering interval $[u, v]$ is defined as*

$$R_i = \frac{1}{2}\left(\min_{y_i^0 \in [u,v]} |y_i - y_i^0| + \max_{y_i^0 \in [u,v]} |y_i - y_i^0|\right). \tag{33}$$

By the noise definition in (1), the absolute noise has the following bounds:

$$\begin{aligned} \inf |e_i| &= \min_{y_i^0 \in [u,v]} |y_i - y_i^0| \\ &= \begin{cases} 0 & \text{if } y_i \in [u, v] \\ \min\{|y_i - u|, |y_i - v|\} & \text{otherwise} \end{cases}, \\ &= \begin{cases} 0 & \text{if } y_i \in [u, v] \\ |y_i - c| - r & \text{otherwise} \end{cases}, \end{aligned} \tag{34}$$

$$\begin{aligned} \sup |e_i| &= \max_{y_i^0 \in [u,v]} |y_i - y_i^0| \\ &= \max\{|y_i - u|, |y_i - v|\} \\ &= |y_i - c| + r. \end{aligned} \tag{35}$$

where c, r denote the center and radius of the covering interval, respectively. Substituting (34) and (35) into (33), we get a practical definition of the covering distance

$$R_i = \begin{cases} \frac{1}{2}\left(|y_i - c| + r\right) & \text{if } y_i \in [u, v] \\ |y_i - c| & \text{otherwise} \end{cases}, \tag{36}$$

where the interval center $c = \frac{u+v}{2}$ and the interval radius $r = \frac{v-u}{2}$. Besides, the covering distance can be simplified to be the *absolute covering distance* (ACD)

$$R_i^A = |y_i - c|. \tag{37}$$

Figure 6 shows the mapping from y_i to the absolute noise. One possible situation of y_i^0 is plotted in the covering interval. The true absolute noise is zero when $y_i = y_i^0$, and it linearly increases with the distance $|y_i - y_i^0|$ (see the blue dotted line). The area between the lower and upper bounds from (34) and (35) is gray-colored. And the covering distance lies in the center of the gray area from the vertical view.

3.2.2 Theoretical Property

As an estimator of the true absolute noise, the covering distance has the following properties.

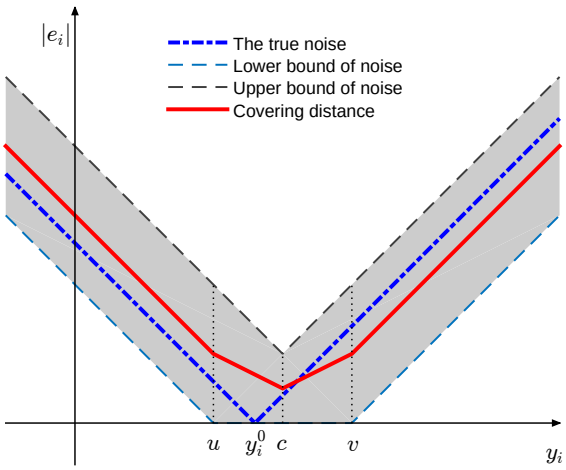


Figure 6: Cover distance and the true absolute noise

Property 5 (Unbiasedness) *Assume the probability density function (PDF) of the interval center $f_c(\cdot)$ is symmetric about the true output y_i^0, i.e. $f_c(y_i^0 - c) = f_c(y_i^0 + c)$ and $y_i^0 \in [u, v]$, then $\mathbb{E}_c(R_i) = |e_i|, \forall y_i \notin [u, v]$.*

The proof can be found in Appendix A.9. It indicates that the covering distance is an unbiased estimation of $|e_i|$ for any $y_i \notin [u, v]$ if the distribution of the interval center c is symmetric about y_i^0. In another word, the average estimated noise is nearby the true noise as long as the centers of covering intervals are around the true output. Note that the unbiasedness of the CD does not hold for $y_i \in [u, v]$.

The *expected absolute deviation (EAD)* of the CD is defined as the expectation of the distance from $|e_i|$ with respect to c:

$$EAD_{CD} = \mathbb{E}_c\big|R_i - |e_i|\big| = \int_{-\infty}^{+\infty} \big|R_i - |e_i|\big| \cdot f_c(c)dc. \tag{38}$$

The *expected relative deviation (ERD)* of the CD is defined as

$$ERD_{CD} = \mathbb{E}_c\frac{\big|R_i - |e_i|\big|}{|e_i|} = \int_{-\infty}^{+\infty} \frac{\big|R_i - |e_i|\big|}{|e_i|} \cdot f_c(c)dc. \tag{39}$$

The EADs of the two bounds in (34) and (35) have similar equations:

$$EAD_L = \mathbb{E}_c\big|\inf|e_i| - |e_i|\big|, EAD_U = \mathbb{E}_c\big|\sup|e_i| - |e_i|\big|.$$

The ERDs of the two bounds are

$$ERD_L = \mathbb{E}_c\frac{\big|\inf|e_i| - |e_i|\big|}{|e_i|}, ERD_U = \mathbb{E}_c\frac{\big|\sup|e_i| - |e_i|\big|}{|e_i|}.$$

Property 6 *Assume the PDF of the interval center $f_c(\cdot)$ is symmetric about y_i^0 and $y_i^0 \in [u, v]$. For any $y_i \notin [u, v]$,*

(1) $EAD_{CD} < EAD_L = EAD_U \equiv r$.

(2) $ERD_{CD} < ERD_L = ERD_U$.

(3) $\frac{\partial EAD_{CD}}{\partial r} > 0$.

The proof can be found in Appendix A.10. It indicates the covering distance outperforms the two bounds in estimating the noise in terms of the EAD and ERD when the noise is relatively large ($y_i \notin [u, v]$). Besides, the deviation of the CD increases with the interval radius r. In another word, the shorter the covering interval is, the more accurate the estimator is. Particularly, it is easy to prove $EAD_{CD} = r/2$ when the interval center c is from the uniform distribution $U(y_i^0 - r, y_i^0 + r)$. In addition, the ACD in (37) has the same characteristics as the CD in Properties 5 and 6 since $R_i^A \equiv R_i$ for any $y_i \notin [u, v]$.

3.2.3 Empirical Property

It is assumed $y_i^0 \in [u, v]$ in Properties 5 and 6. However, Equation (32) implies that y_i^0 may be out of the constructed covering interval. So the previous properties are required to be reexamined in a more realistic situation. Assume the interval center c is from three distributions: the uniform distribution $U(y_i^0 - 1.25r, y_i^0 + 1.25r)$, the Gaussian distribution $N(\mu = y_i^0, \sigma = r/2)$ and the Laplace distribution $Lp(\mu = y_i^0, \sigma = r)$, where μ denotes the mean and σ is the standard deviation. Their covering probabilities ($\mathbb{P}\{y_i^0 \in [u_i, v_i]\} = \mathbb{P}\{y_i^0 \in [c - r, c + r]\}$) are 0.8, 0.95 and 0.76, respectively.

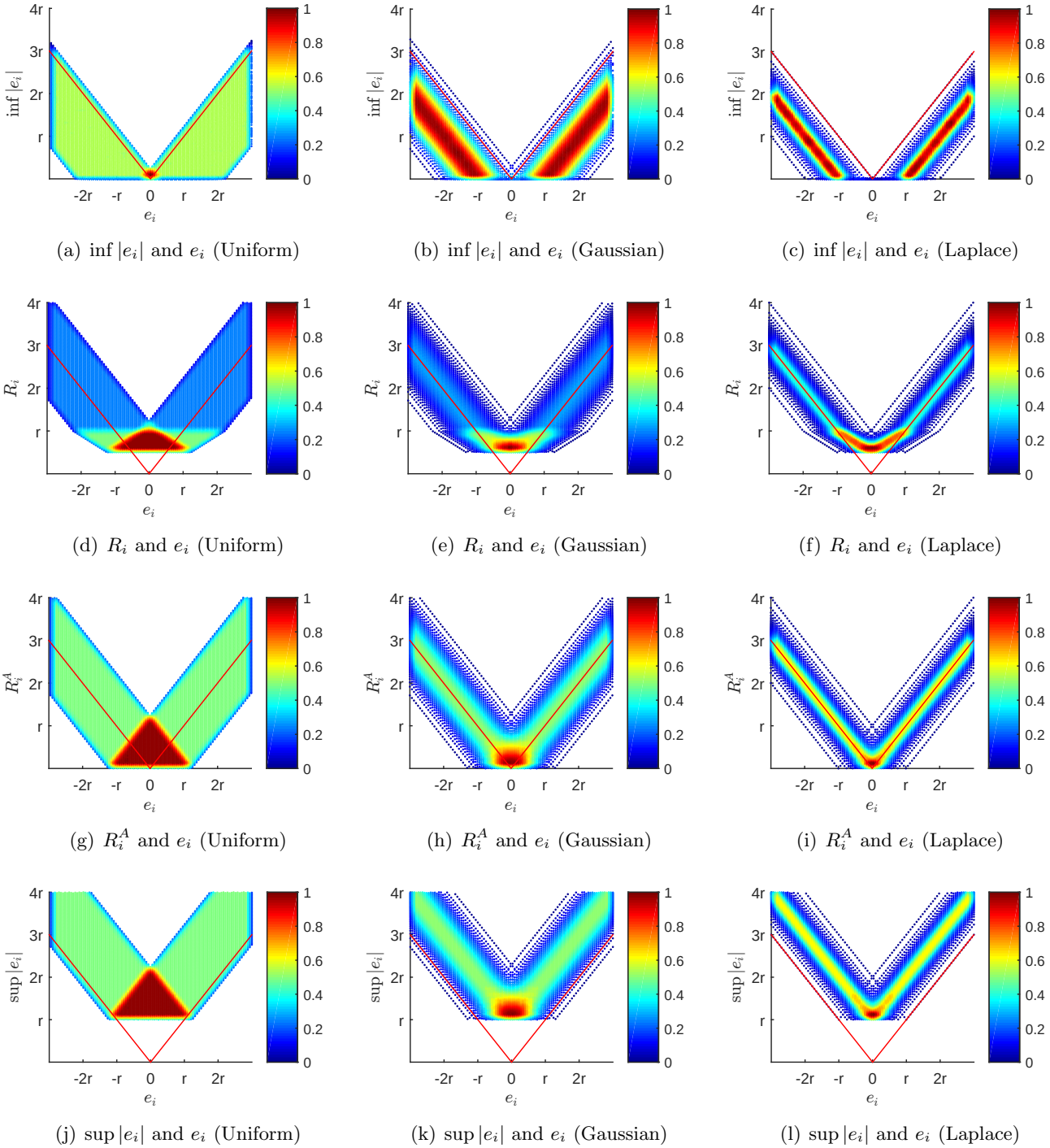
Figure 7 shows the density plots of the true noise and four noise approximations, including $\inf|e_i|$ in (34), $\sup|e_i|$ in (35), R_i in (36), and R_i^A in (37). Generally, the lower bound $\inf|e_i|$ underestimates the noise, and the upper bound $\sup|e_i|$ overestimates for all predefined PDFs. From Figure 7(d), (e) and (f), the covering distance R_i has an unbiased estimation for $|e_i| > 2r$. Indeed, $|e_i| > 2r$ implies $y_i \notin [u, v]$ with a large probability (at least the covering probability),

$$\begin{aligned}
\mathbb{P}(y_i \notin [u, v]) &= \mathbb{P}(|y_i - c| > r) \\
&\geq \mathbb{P}(|y_i - y_i^0| - |y_i^0 - c| > r) \\
&= \mathbb{P}(|e_i| - |y_i^0 - c| > r) \\
&\geq \mathbb{P}(2r - |y_i^0 - c| > r) \\
&= \mathbb{P}(|y_i^0 - c| < r) \\
&= \mathbb{P}(y_i^0 \in [u, v]).
\end{aligned}$$

It means the unbiasedness of Property 5 is verified in the simulation. Whereas R_i usually makes an overestimation when $|e_i| < r$. This bias is mainly from the interference of the upper bound ($R_i = (\inf|e_i| + \sup|e_i|)/2$) as the least estimation of $\sup|e_i|$ is the radius r but not zero in (35). In a word, the covering distance is unbiased for high noises, and it produces an overestimation for low noises. Compared with the covering distance, the ACD estimator performs better for low noises as it gets rid of the minimum limit of $\sup|e_i|$.

Figure 8 shows the EAD and ERD curves under three predefined PDFs of c.

- It can be observed from Figure 8 that $EAD_{CD} \leq EAD_{ACD} < EAD_L \leq EAD_U$ and $ERD_{CD} \leq ERD_{ACD} < ERD_L \leq ERD_U$ when $|e_i| > r$. It means that the



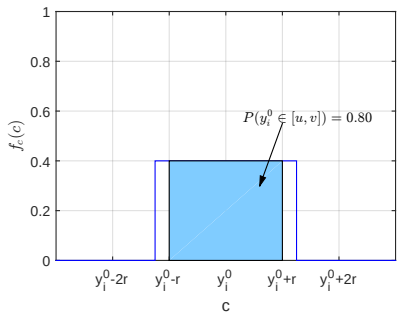
(a) inf $|e_i|$ and e_i (Uniform) (b) inf $|e_i|$ and e_i (Gaussian) (c) inf $|e_i|$ and e_i (Laplace)

(d) R_i and e_i (Uniform) (e) R_i and e_i (Gaussian) (f) R_i and e_i (Laplace)

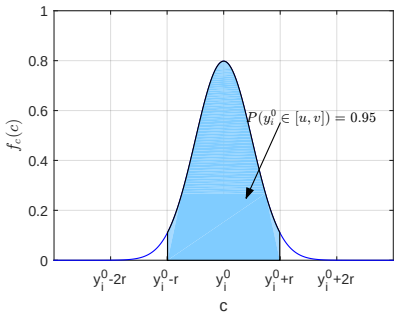
(g) R_i^A and e_i (Uniform) (h) R_i^A and e_i (Gaussian) (i) R_i^A and e_i (Laplace)

(j) sup $|e_i|$ and e_i (Uniform) (k) sup $|e_i|$ and e_i (Gaussian) (l) sup $|e_i|$ and e_i (Laplace)

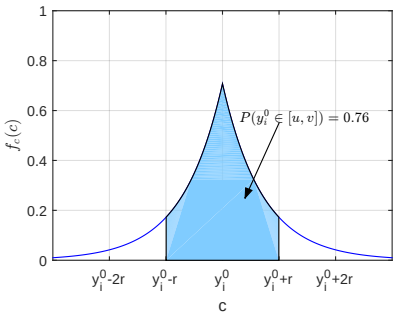
Figure 7: The density maps of the true noise e_i and four approximations (inf $|e_i|$, R_i, R_i^A, sup $|e_i|$) are displayed. In order to obtain a stable result, the interval center c is set to be the equally spaced percentiles $(0.005 : 0.01 : 0.995)$ of the predefined PDFs. One hundred noise values are uniformly taken from the interval $[-3r, 3r]$. Thus there are 10000 dots in each density plot. A red solid line representing the accurate estimation is added to each sub-figure.



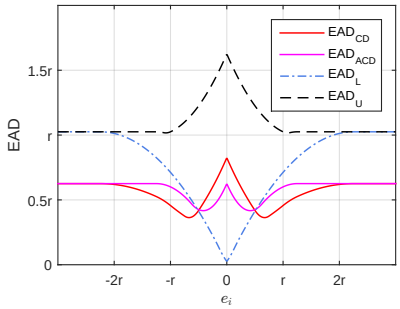
(a) PDF of c (Uniform)



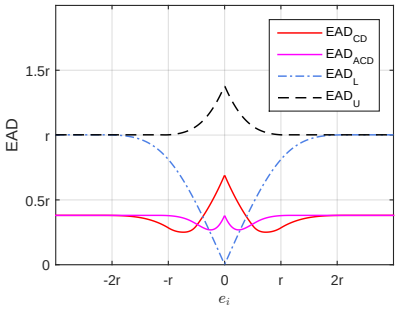
(b) PDF of c (Gaussian)



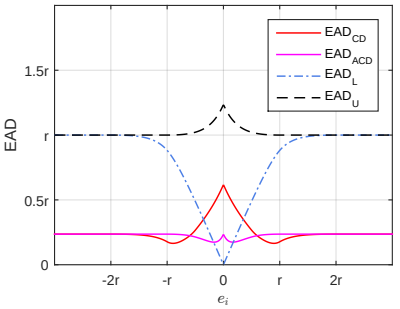
(c) PDF of c (Laplace)



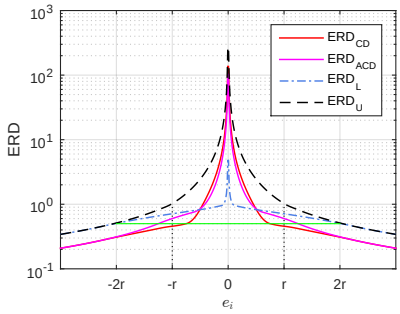
(d) EAD for the Uniform distribution



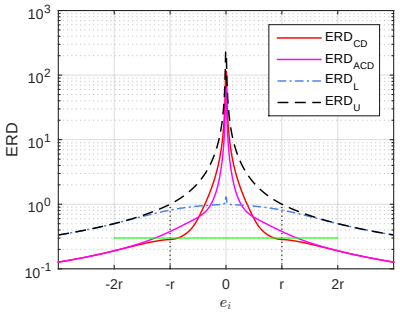
(e) EAD for the Gaussian distribution



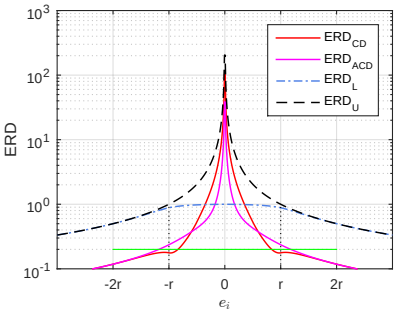
(f) EAD for the Laplace distribution



(g) ERD for the Uniform distribution



(h) ERD for the Gaussian distribution



(i) ERD for the Laplace distribution

Figure 8: The top sub-figures show the predefined PDFs of the interval center c. The middle and bottom sub-figures display the EAD and ERD curves, respectively. In the top sub-figures, the area of $y_i^0 \in [u, v]$ ($u < y_i^0 < v \Leftrightarrow c - r < y_i^0 < c + r \Leftrightarrow y_i^0 - r < c < y_i^0 + r$) is blue-colored. The EAD and ERD curves are obtained in the same way as the density plot in Figure 7, i.e., the variable of integration c is taken from 100 equally spaced percentiles ($0.005 : 0.01 : 0.995$) of the predefined PDFs. Since R_i depends on y_i, c, r and $y_i = y_i^0 + e_i$, EAD_{CD} and ERD_{CD} can be considered as the functions with regard to e_i for fixed y_i^0, r (c is the variable of integration). Also, the independent variable can be replaced with y_i ($e_i \in [-3r, 3r] \Leftrightarrow y_i \in [y_i^0 - 3r, y_i^0 + 3r]$).

CD outperforms the others for large noises in terms of the EAD and ERD. This is generally consistent with that in Property 6. Their difference lies in the independent variable. Property 6 describes the results with regard to y_i ($y_i \notin [u, v]$), whereas Figure 8 displays the results with respect to e_i ($|e_i| > r$) for the convenience of calculation. Actually, both $y_i \notin [u, v]$ and $|e_i| > r$ refer to the large noise.

- When $|e_i| < r$, EAD_L and EAD_{ACD} may be less than EAD_{CD}. That is because the lower bound in (34) usually underestimates the noise and caters for the low noise, while the upper bound in (35) overestimates and then interferes with the low noise estimation of the covering distance ($R_i = (\inf |e_i| + \sup |e_i|)/2$).

- From Figure 8 (d), (e) and (f), all EADs become stable when $|e_i| > 2r$. The reason is that $|e_i| > 2r$ could ensure $y_i \notin [u, v]$ with a relatively large probability (at least the covering probability). According to Property 6, the EAD is a constant for given $f_c(c)$ and r when $y_i \notin [u, v]$, and so, all EAD curves become stable when $|e_i| > 2r$. Besides, both EAD_L and EAD_U are very close to r for all predefined probability density functions when $|e_i| > 2r$.

- All EAD_{CD} and EAD_{ACD} curves are wavy when $|e_i| < r$. It is from the instability of $\big|R_i - |e_i|\big|$. For example, $\big|R_i - |e_i|\big|$ in Figure 6 increases with y_i in $[u, y_i^0]$, and decreases with y_i in $[y_i^0, c + \delta]$, then increases again in $[c + \delta, v]$ (δ is the horizontal distance of the crossing point of $|e_i|$ and R_i from the interval center c). In contrast, the two bounds have monotone EAD curves with regard to $|e_i|$ as their deviations are simpler than the CD. For example, $\big|\inf |e_i| - |e_i|\big|$ decreases with y_i in $[u, y_i^0]$ and increases in $[y_i^0, v]$ in Figure 6. Indeed, the EAD_{CD} curves are based on the density maps in Figure 7(d), (e) and (f), and the instability of $\big|R_i - |e_i|\big|$ also can be observed from Figure 7.

- From Figure 8 (g), (h) and (i), all ERDs generally decrease with $|e_i|$ since they are inversely proportional to the absolute noise. $ERD_{CD} < 50\%$ for the uniform distribution, $ERD_{CD} < 30\%$ for the Gaussian distribution and $ERD_{CD} < 20\%$ for the Laplace distribution when $|e_i| > r$. It means the covering distance has small ERDs in estimating large noises.

Too large ERD (e.g. $ERD > 50\%$) may bring confusion to noise comparison or ranking. For example, there are two cases of noise comparison based on a noise estimator: case 1 $\{A{=}50 \pm 2$ vs. $B{=}30 \pm 2\}$; case 2 $\{A{=}5 \pm 2$ vs. $B{=}3 \pm 2\}$. Although both cases have the same absolute deviation (± 2), they differ greatly in the relative deviation. In case 1, it is clear that noise A is larger than noise B. While there exists an obvious uncertainty in the noise comparison of case 2. That is because the relative deviation in case 2 is significantly larger than that in case 1. The noise estimator with a large ERD may lead to an unreliable or wrong result in noise comparison and it should be avoided wherever possible.

In a word, the covering distance (CD) would be an accurate and reliable estimator, especially for high noises. Meanwhile, the absolute covering distance (ACD) is comparable with the covering distance and it has a lower deviation for low noises.

3.3 Covering Distance Filtering for Regression

This part shows the filtering difference between the CD and ACD, and presents the filtering algorithm based on CD.

3.3.1 Filtering based on CD and ACD

The optimal sample selection in (14) provides a unified framework for noise filtering, and the (absolute) covering distance in (36) and (37) can be embedded in the framework. Specifically, the joint points are in the estimations of $T(\rho)$ and the coefficient C.

$T(\rho)$ can be estimated by replacing e_i^2 with R_i^2,

$$\hat{T}(\rho) = \frac{\mathbb{E}_{D_F}(R_i^2)}{\mathbb{E}_D(R_i^2)} = \frac{\sum_{D_F} R_i^2/(n\rho)}{\sum_D R_i^2/n}, \tag{40}$$

where R_i is the CD estimator in (36).

$$\hat{C} = \frac{\max\limits_{j=1,2,\cdots,J} \sum_{i=1}^n \left(r_i^{(j)}\right)^2/n}{\sum_{i=1}^n R_i^2/n}, \tag{41}$$

where the model error $r_i^{(j)} = m_j(x_i) - y_i$. Compared with the definition of C in (8), $\hat{C}$ takes the maximum on J sets of prediction errors. The reason is that the covering distance is overestimated for low noises, and then the denominator term $\sum_{i=1}^n R_i^2/n$ in (41) is an overestimation of $\mathbb{E}_D(e_i^2)$. This modification in (41) aims to weaken the negative impact of noise estimation. In addition, $T(\rho)$ and C also can be similarly estimated by the absolute covering distance R_i^A in (37).

By (11) and(13), the estimated objective function

$$\hat{\mathcal{F}}(\rho) = \left[\frac{\epsilon(D)}{\epsilon(D_F)}(1+\hat{C}) - \hat{C} - \hat{T}(\rho)\right] \cdot \epsilon(D_F) \tag{42}$$

$$= (\hat{C}+1)\cdot\epsilon(D) - (\hat{C}+\hat{T}(\rho))\cdot\epsilon(D_F), \tag{43}$$

where $\epsilon(D), \epsilon(D_F)$ depend on n, ρ, h, η in (6) and (9).

Considering that the CD and ACD may be unreliable in estimating and ranking low noises (large ERDs), low-noise samples are directly retained in noise filtering. According to Proposition 2, they can be identified through the covering interval. The sample selection is optimized on high-noise samples ($y_i \notin [u, v]$) under the proposed OSS framework,

$$\hat{\rho^*} = \arg\max_{\rho > n_c/n} \hat{\mathcal{F}}(\rho), \tag{44}$$

where $\hat{\rho^*}$ denotes the estimated optimal relative size, and n_c is the number of low-noise samples ($y_i \in [u, v]$).

It is obvious that $\hat{\rho^*}$ satisfies $\frac{\partial \hat{\mathcal{F}(\rho)}}{\partial \rho} = 0$ when $n \to \infty$, where

$$\begin{aligned}\frac{\partial \hat{\mathcal{F}(\rho)}}{\partial \rho} &= (\hat{C} + \hat{T}(\rho)) \cdot \left(-\frac{\partial \epsilon(D_F)}{\partial \rho}\right) - \frac{\partial \hat{T}(\rho)}{\partial \rho} \cdot \epsilon(D_F) \\ &= \left(\frac{\max_j \sum_{i=1}^{n} (r_i^{(j)})^2/n}{\sum_{i=1}^{n} R_i^2/n} + \frac{\sum_{D_F} R_i^2/(n\rho)}{\sum_D R_i^2/n}\right) \cdot \left(-\frac{\partial \epsilon(D_F)}{\partial \rho}\right) - \frac{\partial \hat{T}(\rho)}{\partial \rho} \cdot \epsilon(D_F) \quad (45) \\ &= \left(\frac{\max_j \sum_{i=1}^{n} (r_i^{(j)})^2/n + \sum_{D_F} (R_i^2)/(n\rho)}{\sum_D R_i^2/n}\right) \cdot \left(-\frac{\partial \epsilon(D_F)}{\partial \rho}\right) - \frac{\partial \hat{T}(\rho)}{\partial \rho} \cdot \epsilon(D_F).\end{aligned}$$

Note that the ACD in (37) provides a lower estimation than the CD in (36) for low noises in D_F, i.e., $R_i^A < R_i$ for $y_i \in [u, v]$, we have

$$\frac{\sum_{D_F} (R_i^A)^2/(n\rho)}{\sum_D (R_i^A)^2/n} < \frac{\sum_{D_F} (R_i^2)/(n\rho)}{\sum_D R_i^2/n}$$

and

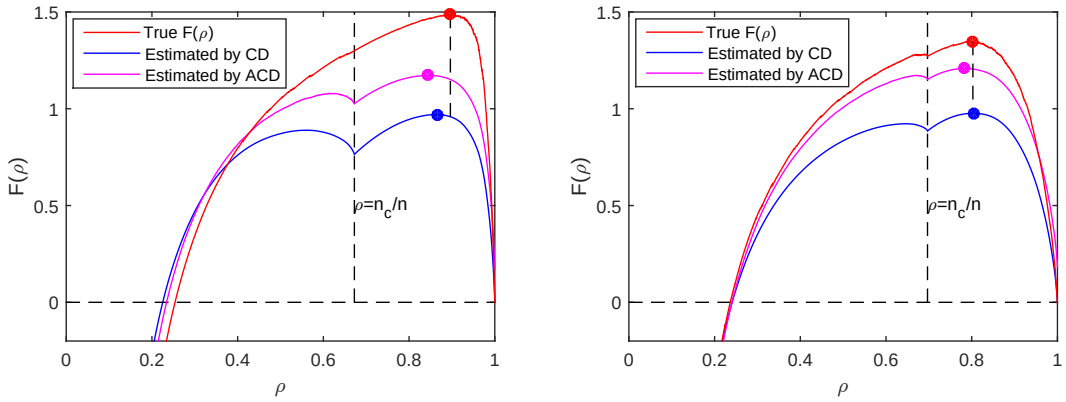
$$\frac{\max_j \sum_{i=1}^{n} (r_i^{(j)})^2/n + \sum_{D_F} (R_i^A)^2/(n\rho)}{\sum_D (R_i^A)^2/n} < \frac{\max_j \sum_{i=1}^{n} (r_i^{(j)})^2/n + \sum_{D_F} (R_i^2)/(n\rho)}{\sum_D R_i^2/n}.$$

By (25) and (16), we know $\left(-\frac{\partial \epsilon(D_F)}{\partial \rho}\right) > 0$ and $\frac{\partial \hat{T}(\rho)}{\partial \rho} > 0$. Then $\frac{\partial \hat{\mathcal{F}(\rho)}}{\partial \rho}$ based on the ACD will be less than that based on the CD. Considering that the slope of $\hat{\mathcal{F}(\rho)}$ generally decreases with ρ, we could deduce that the estimated relative size $\hat{\rho^*}$ from the ACD is smaller than that from the CD. In another word, the filtering based on the ACD will remove more samples than that based on the CD, and the ACD takes a higher risk of overcleansing than the CD in noise filtering.

The CD and ACD are compared in the following simulation.

Let D_{in}, D_{out} be the subsets consisting of samples in or out of the covering interval, respectively. Their sample sizes are $n_c = \#\{D_{in}\}$ and $n_p = \#\{D_{out}\}$. Considering that the sorted sequences of D_{in} from both CD and ACD may be unreliable (large ERDs), samples in D_{in} and D_{out} are sorted separately and those in D_{out} are removed first.

Figure 9 shows the filtering results based on the CD and ACD on benchmark data set *Space_ga* (detailed data information is shown in Table 5). From Figure 9(a), the optimal relative size ρ^* has the following relation: $\rho^*_{ACD} < \rho^*_{CD} < \rho^*$. The result in Figure 9(b) becomes $\rho^*_{ACD} < \rho^*_{CD} \approx \rho^*$. It means the filtering from ACD removes more samples than the CD and the true ρ^*. It verifies the previous deduction that the ACD takes a higher risk of overcleansing than the CD. Clearly, the optimal relative size from the CD is more closer to the true one. When the noise ratio is large enough, the filtering based on the CD is comparable with the true ρ^*.



(a) Objective function for $NR = 10\%$

(b) Objective function for $NR = 30\%$

Figure 9: The sample selections based on the CD and ACD are compared under the OSS framework. The Gaussian noises from $N(0, 0.5^2)$ are artificially added to the data set *Space_ga* (the sample size $n = 3107$, the noise ratio $NR = 10\%, 30\%$). The model predictions are produced by the kNN model ($k = 3$) in the subsets scheme ($J = 5$). Note that the sequence of sample removing based on the CD is the same as that from the ACD, only one true objective function is displayed as the baseline. In other words, the three objective functions in each sub-figure have the same sequence of sample removing. The true $\mathcal{F}(\rho)$ is computed by the true noise, while the others are based on the noise estimations in (36) and (37). The reference line marked with $\rho = n_c/n$ separates the low-noise (D_{in}) and high-noise (D_{out}) samples.

It is worth noting in Figure 9 that a valley occurs at $\rho = n_c/n$ for each curve, especially for those from the CD and ACD. The reason is that samples in D_{in} and D_{out} are sorted separately and the estimated noises of a few samples in D_{in} are larger than the least of D_{out}. These samples are named as the overflowing samples. As mentioned before, the noise for D_{in} is usually overestimated by the CD and ACD (see Figure 8), while the estimation for D_{out} is unbiased (Property 5). The truth is that the overflowing samples based on the true noise is not as many as those based on the CD or ACD. This is verified by the fact that the valley in the true $\mathcal{F}(\rho)$ curve is insignificant. In addition, $\hat{\mathcal{F}}(\rho)$ in reality is a smooth function with respect to ρ, while the true $\mathcal{F}(\rho)$ is slightly rough as the sequence of sample removing is based on the noise estimation but not the true value.

The objective function estimated by the ACD is closer to the true curve because it has a lower deviation for some low noises. In estimating the objective function, the ACD wins in the distance from the true function, while the CD wins in the shape and the trend. Compared with the ACD, the CD estimator allows more samples to be retained. As a result, it has a lower risk of overcleansing and is closer to the true optimal sample selection. Thus the CD estimator is adopted in the filtering algorithm.

3.3.2 Filtering Algorithm based on CD

Let D_{in}, D_{out} be the subsets consisting of samples in or out of the covering interval, respectively. Their sample sizes are $n_c = \#\{D_{in}\}$ and $n_p = \#\{D_{out}\}$. The noise filtering is executed in the following way. Firstly, the samples in D_{in} are considered as low-noise and they are retained directly as the covering distance (CD) has large relative deviations in estimating low noises. Secondly, the noise of each sample is estimated by the CD. Then the samples in D_{out} can be removed one by one according to the absolute noise (large noise first). When a new sample is removed, the objective function $\mathcal{F}(\rho)$ for a smaller ρ can be estimated by (43). Finally, the removing operation is stopped at the maximum $\mathcal{F}(\rho)$.

Algorithm 1 Covering distance filtering (CDF) algorithm for regression.

Input:

Regression data set $D = \{(x_i, y_i), i = 1, 2, \cdots, n\}$

Base models $y = m_j(x), j = 1, 2, \cdots, J$

Output:

Filtered data set D_F

1: Train and test the base models in the subsets scheme, then each sample has J model predictions and errors.
2: Compute the covering interval for each sample by (31).
3: Compute the CD value for each sample by (36). Sort the samples in D_{out} by the CD in ascending order and obtain a new set $D'_{out} = \{(x_{i'}, y_{i'})\}_{i'=1}^{n_p}$.
4: Estimate the coefficient C by (41), compute $\epsilon(D)$ by (6).
5: **for** $s = 1$ **to** n_p **do**
6: $n_F = n_c + s$, $\rho_s = n_F/n$, D_F=$D_{in} \bigcup \{(x_{i'}, y_{i'}) \in D'_{out}\}_{i'=1}^{s}$;
7: Compute $\hat{T}(\rho_s)$ by (40); Compute $\epsilon(D_F)$ by (9).
8: Estimate the objective function $\mathcal{F}(\rho_s)$ by (43).
9: **end for**
10: $s^* = \arg\max_s \mathcal{F}(\rho_s)$, D_F=$D_{in} \bigcup \{(x_{i'}, y_{i'}) \in D'_{out}\}_{i'=1}^{s^*}$.

Algorithm 1 shows the steps of the proposed covering distance filtering (CDF). Model predictions and errors are generated in the subsets scheme in step 1. In steps 2-3, the covering interval and CD are obtained based on the model predictions and real outputs. The samples in D_{out} are sorted to decide the removing sequence. Step 4 computes the fixed items in (43), including coefficient C and $\epsilon(D)$. In steps 5-9, the objective function $\mathcal{F}(\rho)$ is estimated for each possible filtering from the removing sequence. Step 10 finds the filtering such that $\mathcal{F}(\rho)$ achieves the maximum ($\rho^* = (n_c + s^*)/n$). The traversal calculation of $\mathcal{F}(\rho_s)$ (steps 5-9) is efficiently executed through vector operations (vectors $\hat{T}(\rho_s)$ and $\epsilon(D_F)$) in our experiments.

The relative size is limited in $\rho = (n_c + 1 : n)/n$, and the maximum objective value must exist. If any filtering is determined to be ineffective by (7), the optimal relative size will be equal to 1. In most cases, $T(\rho)$ is convex about ρ and $\mathcal{B}_T(\rho)$ is concave, the estimated $\mathcal{F}(\rho)$ is usually a concave function with respect to $\rho \in [(n_c + 1)/n, 1]$. Thus the optimization procedure can also be implemented by other optimizing methods such as the binary search and gradient-based search.

Assume that $\mathcal{C}_j(n)$ is the time complexity function of the j-th base model. Only the first step of Algorithm 1, computing the model predictions, is the complexity of $\mathcal{C}_j(n)$ and the other steps have a linear complexity. Hence the total complexity is $T(CDF) = \sum_{j=1}^{J} \mathcal{C}_j(n) + n_p \cdot n$, where n_p denotes the number of samples out of the covering interval. If the kNN model is employed in the subsets scheme, it becomes $T(CDF) = (\log(n) + n_p) \cdot n$.

4. A Real Example: Noise Filtering on Apparent Age Data Set

4.1 Competition Data Set Information

Apparent age estimation has attracted more and more researchers since its potential applications in the real world such as in forensics or social media (Rothe et al., 2018). In apparent age estimation, each facial image is labeled by multiple individuals. The mean age (rather than the real age) is set to be the ground truth age and the uncertainty is introduced by the standard deviation (Std). Briefly speaking, face age estimation consists of two crucial stages: age feature representation and age estimator learning (Liu et al., 2015). Huo et al. (2016) provide four sets of feature representations for images in the age estimation competition data set based on fine-tuned deep models. The original images are from ICCV 2015 (training data set: 2463×2; validation data set: 1136×2) and CVPR 2016 (training data set: 4113×2; validation data set: 1500×2) (Escalera et al., 2015). Images are turned over as the image augmentation (×2). The data set contains 90 features and 18424 samples in each feature representation. The following age noise identification is based on the given features and labels.

As the apparent age is labeled by multiple individuals, the age value may be inconsistent with the facial image. The noise filtering on apparent age data set aims to find the most inconsistent label(s) and improve the prediction ability of the model trained on it. The proposed CDF ($J = 5$) is independently executed on the data sets with four sets of different feature representations. Predictions are obtained by kNN ($k = 5$) regressor in the subsets scheme. Considering the randomness of data partition, each noise filtering round is repeated five times and the covering distance (CD) is averaged.

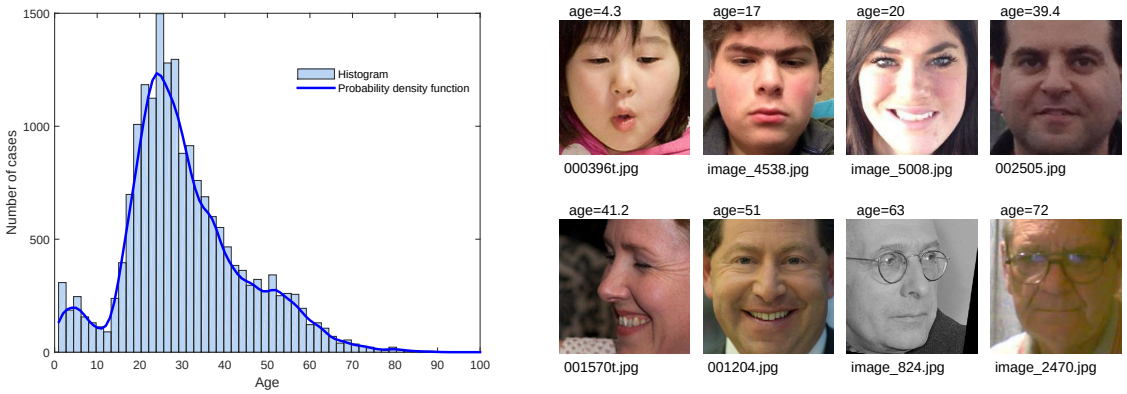
Figure 10(a) displays the age distribution by means of the density curve and histogram. Figure 10(b) shows some confident examples (with the least average CD) in different age intervals. The apparent age and image name are given over and under the image, respectively.

4.2 Noise Filtering Results

It is known that high-noise samples usually have large CDs. Table 2 lists the images with the 20 largest average CDs (on the four sets and five repetitions) and those with the maximum CD over 4. Considering that the facial image is similar to its mirror image (such as $005052.jpg$ and $005052t.jpg$), they are listed adjacently and only the one with the larger average is shown. The CD values beyond 4 have a light blue background, and the average CDs over 4 are bolded. The label bias denotes the relationship between the age label and the covering interval. The symbol ↑ means the label is larger than the interval center, i.e., the label seems to be overestimated. While the symbol ↓ has the opposite meaning.

Image	No.	Name (Source)	Age label	Std.	Label bias	CD from different features Fea. 1	Fea. 2	Fea. 3	Fea. 4	Aver. CD
	1	005152.*jpg*	11.3	14.1	↑	8.82	8.62	6.80	7.39	**7.91**
	2	005152*t.jpg*	11.3	14.1	↑	7.88	10.17	4.84	7.88	**7.69**
		(CVPR2016 validation dataset)								
	3	000115*t.jpg*	89.2	6.5	↑	6.36	6.29	5.18	5.50	**5.83**
	4	000115.*jpg*	89.2	6.5	↑	5.94	6.27	5.15	5.25	**5.65**
		(CVPR2016 training dataset)								
	5	005165.*jpg*	87.9	6.6	↑	4.94	4.72	3.78	4.13	4.39
	6	005165*t.jpg*	87.9	6.6	↑	5.01	5.13	3.79	3.97	4.47
		(CVPR2016 validation dataset)								
	7	005565.*jpg*	25.2	10.7	↑	3.30	4.24	4.20	3.44	3.79
	8	005565*t.jpg*	25.2	10.7	↑	3.29	4.55	3.58	2.29	3.43
		(CVPR2016 validation dataset)								
	9	000962*t.jpg*	74.1	11.2	↑	4.06	4.04	3.23	3.26	3.65
	10	000962.*jpg*	74.1	11.2	↑	4.06	3.85	2.14	3.43	3.37
		(CVPR2016 training dataset)								
	11	*augmentation_image_*2084.*jpg*	7	5.2	↑	10.05	0.82	1.96	1.60	3.61
	12	*image_*2084.*jpg*	7	5.2	↑	8.73	0.97	1.49	1.63	3.21
		(ICCV2015 validation dataset)								
	13	002914.*jpg*	7.1	5.2	↑	8.64	0.30	1.58	1.54	3.02
	14	002914*t.jpg*	7.1	5.2	↑	9.97	0.24	1.87	1.52	3.40
		(CVPR2016 training dataset)								
	15	003260.*jpg*	86.9	5.5	↑	3.96	4.85	1.16	2.32	3.07
	16	003260*t.jpg*	86.9	5.5	↑	3.96	4.60	1.16	2.27	3.00
		(CVPR2016 training dataset)								
	17	*image_*432.*jpg*	51	4.7	↓	2.06	1.88	2.01	2.79	2.18
	18	*augmentation_image_*432.*jpg*	51	4.7	↓	1.94	1.98	1.95	2.23	2.02
		(ICCV2015 training dataset)								
	19	002376*t.jpg*	55.1	7.7	↓	2.27	2.18	2.16	1.93	2.14
	20	002376.*jpg*	55.1	7.7	↓	2.09	2.29	2.11	1.57	2.01
		(CVPR2016 training dataset)								
	21	002827*t.jpg*	50.5	8.4	↑	1.22	1.83	2.16	2.53	1.93
		(CVPR2016 training dataset)								
	22	*image_*4725.*jpg*	31	2.2	↑	0.94	5.30	0.03	0.02	1.57
	23	*augmentation_image_*4725.*jpg*	31	2.2	↑	0.41	5.35	0.01	0.04	1.45
		(ICCV2015 training dataset)								
	24	001265*t.jpg*	31.1	2.2	↑	0.37	5.26	0.03	0.02	1.42
	25	001265.*jpg*	31.1	2.2	↑	0.63	5.19	0.17	0.02	1.50
		(CVPR2016 training dataset)								

Table 2: Noisy age labels in competition data set



(a) Age distribution

(b) Confident age labels

Figure 10: Information of age estimation data set from ICCV2015 and CVPR2016

In Table 2, it might be difficult to decide whether the age labels have deviations for the elderly people (Nos. 3-6, 9-10, 15-16), whereas it becomes easier for the children. For example, the age for the first image (Nos. 1, 2, age=11.3) should intuitively between 2 and 5, and the label is inaccurate. Although the 9-th and 10-th images (Nos. 17-20) are identical, their features have minor differences among the four sets of representations. Moreover, the subset partition is random, hence these images (Nos. 17-20) do not exactly have the same CD. So do images Nos. 11-14 and Nos. 22-25. The apparent age of the 6-th and 7-th images (Nos. 11-12, age=7; Nos. 13-14, age=7.1) should be no more than 5. It is inappropriate to assign age=31.1 to the last two images (No. 21-25). The above images have notable overestimated labels. While the ages for images Nos. 17-20 seem to be over 60 and the labels (51 and 55.1) should be underestimated. The results indicate that the proposed CDF filter can effectively identify inaccurate labels in apparent age data set on the basis of proper feature representations.

The relative size of the filtered data set is about 82% (15108 samples). The age distribution changes are shown in Figure 11. The number of removed samples for each age label (1-100) is plotted in the figure, and the horizontal dotted line denotes threshold 30. The age labels losing more than 30 samples range from 16 to 61. From the smoothed density curves, the distribution difference mainly lies in the same range. It means that the CDF is more likely to drop samples in the high-density area. Thus the CDF would not destroy the initial age distribution and it is a safe filter.

For an imbalanced data set, the filtering also can be executed on the subsets separately. For example, the age data set can be partitioned into three subsets according to the initial age density: [0-15], [16-60], [60-100]. As described in Subsection 2.3.2, the optimal relative size decreases with the sample size. Then more samples will be removed by CDF on the second subset ([16-60]) which has many more samples than the other two.

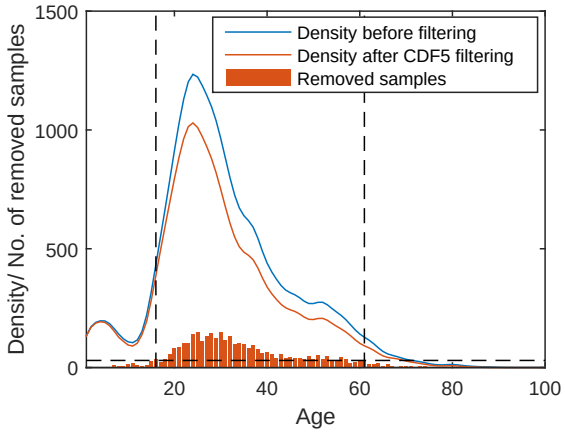
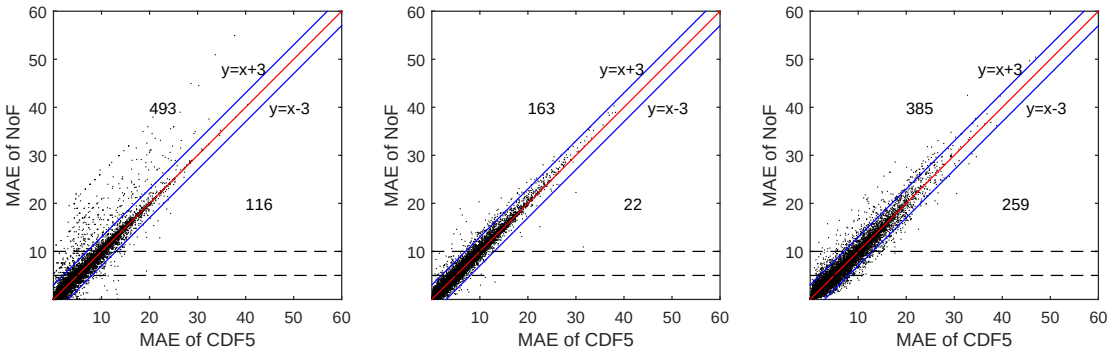


Figure 11: Change of the age distribution in filtering

4.3 Age Prediction on Real Data Set *wiki*

To evaluate the effectiveness of filtering, the model trained on the competition data set (ICCV2015 and CVPR2016) is tested on real data set *wiki* based on Wikipedia (Rothe et al., 2018). The raw *wiki* data set is preprocessed by the face and label validity detections. The cascaded convolutional networks is employed to detect the face and landmarks (Huang et al., 2018). The feature representation for images in data set *wiki* is obtained by the fine-tuned deep models the same as that for the training data set (Huo et al., 2016). Ages in [0,100] are considered as valid labels. The final test set has 29930 samples.



(a) MAE of kNN (b) MAE of RF (c) MAE of NNet

Figure 12: Prediction error comparison

Three widely used models, kNN ($k = 5$), random forest (RF, 100 trees) and neural networks (NNet, 30 neurons), are trained on the data sets before (no filtering, NoF) and after CDF5 (CDF,$J = 5$) filtering, and the test error is measured by the mean absolute error (MAE). The test errors of NoF and CDF5 are compared in Figure 12. The red diagonal means equal errors and the blue lines denote the error difference within 3. It can be observed that most scatters are around the diagonal and most MAE values are less than 20. The

number of samples with the error difference over 3 is marked in the corresponding area. For example, there are 493 samples whose NoF error is beyond the CDF error+3 in Figure 12(a), while only 116 samples whose NoF error is less than the CDF error-3. It means that CDF5 outperforms NoF for samples with the error difference over 3. Results for the RF and NNet models are similar to that of kNN (163>22, 385>259).

The detailed error comparison of NoF and CDF5 is implemented on three test sets. The first set consists of all available testing samples, the second set contains samples whose MAE of NoF is over 5 (above the lower dotted line in each sub-figure of 12), and the last contains those with MAE over 10 (above the upper dotted line in Figure 12). Note that the last two sets may vary with the testing model. Table 3 lists the results including the win-tie-lose ratio, the average error (± Std) on testing samples, the relative reduction, and the P-value of the rank-sum test. The relative reduction is calculated by the formula: $1 - MAE_{CDF5}/MAE_{NoF}$. The reductions over 1% and P-values under 0.05 are bolded.

Test set	Model	#Samples	CDF5 vs. NoF			MAE		Relative reduction	P-value
			win	tie	lose	NoF	CDF5		
All available	kNN	29930	38.28%	23.72%	38.00%	5.38±4.43	5.29±4.19	**1.63%**	0.561
	RF	29930	50.07%	0.00%	49.93%	5.39±4.30	5.35±4.20	0.66%	0.934
	NNet	29930	50.93%	0.00%	49.07%	5.41±4.43	5.40±4.36	0.03%	0.641
MAE(NoF)>5	kNN	13342	41.84%	22.96%	35.20%	9.07±4.12	8.79±3.80	**2.97%**	**0.000**
	RF	13516	52.74%	0.00%	47.26%	9.00±3.83	8.87±3.73	**1.37%**	**0.046**
	NNet	13369	50.91%	0.00%	49.09%	9.10±4.13	9.02±4.03	0.81%	0.800
MAE(NoF)>10	kNN	3745	46.33%	23.60%	30.07%	14.12±4.50	13.28±4.14	**5.99%**	**0.000**
	RF	3807	61.15%	0.00%	38.85%	13.77±3.96	13.43±3.87	**2.50%**	**0.000**
	NNet	3819	57.29%	0.00%	42.71%	14.11±4.43	13.76±4.29	**2.44%**	**0.004**

Table 3: MAE comparison of NoF and CDF5 on *wiki* data set

It is clear from Table 3 that the average error of CDF5 is less than that of NoF in all cases. For all samples, the two sets of errors have no significant difference. For the second subset, the CDF5 error is significantly less than the NoF error for kNN and RF. For the last subset, the error difference between CDF5 and NoF is significant for all models. Generally speaking, the CDF filter could significantly reduce the prediction error for hard-to-learn samples.

Furthermore, the CDF filter is superior to NoF in efficiency as it has a smaller data size and a shorter training time. For example, the total time of CDF5 in filtering and testing (9.1+219.3 seconds) is less than that of NoF (0+351.5 seconds) for the NNet model.

5. Experiments and Analysis

In this section, we empirically study the performances of the proposed CDF filter on benchmark data sets. We present our experimental framework, empirical results, and analysis.

5.1 Experimental Framework

For each data set and filter, the overall process is shown in Figure 13. Firstly, the original data set D is randomly partitioned into two parts D_A, D_B whose size ratio is 8:2. The polluted set D_{Ap} is obtained by adding noises artificially to the first part D_A. Then D_{Ap} is filtered and a cleaner set D_{Af} is obtained. Finally, the model is trained on D_{Af} and tested

on the second part D_B. The above steps are repeated ten times owing to the randomness in data partition and adding noises.

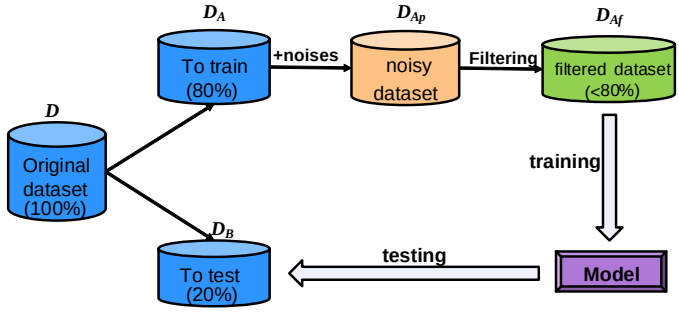


Figure 13: The overall framework of experiments

Task	Regression	Ordinal Classification/Regression
Data	10 data sets (Table 5)	17 data sets (Table 6)
Noise distribution /Noise rule	Uniform 1: $U(-0.5, 0.5)$; Uniform 2: $U(-0.8, 0.8)$; Gaussian 1: $N(\mu = 0, \sigma = 0.5)$; Gaussian 2: $N(\mu = 0, \sigma = 0.8)$; Laplace 1: $Lp(\mu = 0, \sigma = 0.5)$; Laplace 2: $Lp(\mu = 0, \sigma = 0.8)$; Mixture 1: $N(\mu = 1, \sigma = 0.3) + N(\mu = -1, \sigma = 0.3)$; Mixture 2: $N(\mu = 1, \sigma = 0.2) + N(\mu = -0.8, \sigma = 0.4)$.	Label $y_i \to y_j$ $\cdot y_i \neq y_j$ $\cdot P(y_i \to y_j) \propto \#\{y_j\}/n$
Noise ratio	0%,10%,20%,30%,40%	0%,10%,20%,30%
CDF (par.)	CDF5($J = 5$), CDF6($J = 6$), CDF7($J = 7$)	CDF5($J = 5$), CDF6($J = 6$), CDF7($J = 7$)
Competitors(par.)	NoF (No filtering), MI (threshold $\alpha = 0.05$, number of neighbors: 6), RegENN/Reg (threshold $\alpha = 5$, number of neighbors: 9), DiscENN/Disc (number of neighbors: 9).	NoF (No filtering), RegENN/Reg (threshold $\theta = 5$, number of neighbors: 9), DiscENN/Disc (number of neighbors: 9), ENN (number of neighbors: 3), ANN (number of neighbors: 3), CF (number of folds: 10), MVF (number of folds: 4), IPF (number of subsets: 5), HARF (agreement level: 70%), INFFC (threshold: 0).
Testing model	kNN, NNet, RF	SVC1V1, NNOP
Hyper-parameters	No. of neighbors (kNN): $k \in \{1, 3, 5, 7, 9\}$ No. of hidden neurons (NNet): $H \in \{10, 20, 30, 40\}$ No. of trees (RF): $T \in \{50, 100, 150, 200\}$	Kernel width(SVC1V1): $\sigma \in \{10^{-3}, 10^{-2}, \cdots, 10^{3}\}$ No. of hidden neurons (NNOP): $H \in \{10, 20, 30, 40\}$
Evaluation	MSE	MAE, MZE

Table 4: Experimental settings

The detailed experimental settings are listed in Table 4.

- **Data sets**. 27 benchmark data sets are employed for supervised learning. All data sets in Table 5 are regression problems (Dua and Graff, 2017; Chang and Lin, 2016), and those in Table 6 are prepared for ordinal classification/regression (Gutierrez et al., 2016; Sanchez-Monedero et al., 2019). Note that the first ten data sets in Table 6 are derived from regression data sets by discretizing the output. All numerical features have been scaled to $[-1, 1]$.

No.	Data set	#Sample	#Feature
1	Yacht Hydrodynamics	308	7
2	Housing	506	14
3	Energy efficiency	768	8
4	Concrete	1030	9
5	Geographical Original of Music	1059	68
6	MG	1385	6
7	Airfoil self-noise	1503	6
8	Space_ga	3107	6
9	Skill Craft Master Table	3395	20
10	Parkinsons Telemonitoring	5875	26

Table 5: Regression data sets

No.	Data set	Type	#Sample	#Feaure	#Class	Class distribution
1	Housing5	Discretised	506	14	5	≈101 per class
2	Stock5	Discretised	700	9	5	140 per class
3	Abalone5	Discretised	4177	11	5	≈836 per class
4	Bank5	Discretised	8192	8	5	≈1639 per class
5	Bank5′	Discretised	8192	32	5	≈1639 per class
6	Computer5	Discretised	8192	12	5	≈1639 per class
7	Computer5′	Discretised	8192	21	5	≈1639 per class
8	Cal.housing5	Discretised	20640	8	5	4128 per class
9	Census5	Discretised	22784	8	5	≈4557 per class
10	Census5′	Discretised	22784	16	5	≈4557 per class
11	Balance-Scale	Real	625	4	3	288-49-288
12	SWD	Real	1000	10	4	32-352-399-217
13	Car	Real	1728	21	4	211-384-69-65
14	Eucalyptus	Real	736	91	5	180-107-130-214-105
15	LEV	Real	1000	4	5	93-280-403-197-27
16	Wine quality-Red	Real	1599	11	6	10-53-681-638-199-18
17	ERA	Real	1000	4	9	92-142-181-172-158-118-88-31-18

Table 6: Ordinal classification data sets

- **Noises**. Assume that all the original data sets are unpolluted. And noises are artificially added to the output. There are 8 kinds of noise distributions for regression. Note that the last two are mixed by Gaussian distributions, and 50% of the noises are from a single Gaussian distribution in each mixture. The last mixed distribution is asymmetric. In ordinal classification, the label is randomly changed to other labels according to the noise ratio. And the transforming probability is proportional to the label frequency.

- **Testing models**. There are three testing models in regression: kNN, Neural networks (NNet), and random forest (RF). Support vector classifier with OneVsOne (SVC1V1) and Neural network with ordered partitions (NNOP) are adopted in ordinal classification (Gutierrez et al., 2016; Sanchez-Monedero et al., 2019). All model hyper-parameters are selected by five-fold cross-validation over the training set after filtering.

- **Filters**. In regression, the proposed CDF algorithm is compared with existing filters including MI (Guillen et al., 2010), RegENN (Kordos et al., 2013), and DiscENN (Arnaiz-González et al., 2016). The competitors for ordinal classification include ENN (Barandela and Gasca, 2000), ANN (Barandela and Gasca, 2000), CF (Sluban et al., 2014), MVF (Brodley and Friedl, 1999), IPF (Khoshgoftaar and Rebours, 2007), HARF (Sluban et al., 2010), and INFFC (Sáez et al., 2016). Considering that the ordinal classification can be seen as a special regression problem, RegENN and DiscENN are utilized in the filtering of ordinal classification data sets. The data set without any filtering (NoF) is also examined as a baseline. In addition, the covering interval in CDF is obtained by the kNN ($k = 3$) predictions in the subsets scheme. In the stage of optimizing the relative size, the parameters are specified as: the probability constant $\eta = 0.05$, the VC-dimension $h = 100$. The parameters for the other filters are set to be the suggested values in references.

- **Evaluators**. Mean square error (MSE) is utilized to measure the generalization ability in regression. Mean absolute error (MAE) and mean zero-one error (MZE) are the most common indicators for evaluating ordinal classification models (Gutierrez et al., 2016).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2,$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\mathcal{O}(\hat{y}_i) - \mathcal{O}(y_i)|,$$

$$MZE = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}(\hat{y}_i \neq y_i),$$

  where y_i is the real output/label, $\hat{y}_i$ is the predicted output/label and $\mathcal{O}(\cdot)$ denotes the rank function. MZE considers a zero-one loss for misclassification, while MAE uses an absolute cost.

5.2 Results and Analysis in Regression

The covering interval and covering distance are designed for identifying and estimating the noise. Their performances are studied in this section. Then the CDF algorithm is compared with other filters in terms of the prediction error after filtering.

5.2.1 Performance of Covering Interval

The covering interval is constructed by (31) and it may not cover the true output. So it is necessary to study the covering probability.

Figure 14 shows the average covering probability and interval length (± standard deviation/2) on eight noise distributions and five repetitions for each data set. Each marker corresponds to a regression data set and a filter. The results of CDF5, CDF6, and CDF7 for the same data set are connected by a dashed line. It is clear that the CDF5 has a smaller covering probability and a shorter interval for each data set. Whereas the CDF7 has the longest interval and the largest covering probability among three filters. Property 6 indicates that the deviation of the CD increases with the interval radius or length under the assumption of covering the true output. So a short interval and a large covering probability are required for a good noise estimator. From Figure 14, there is no dominant covering interval from the aspects of interval length and covering probability. In another word, the advantage changes from the interval length to the covering probability when the number of base models increases from 5 to 7.

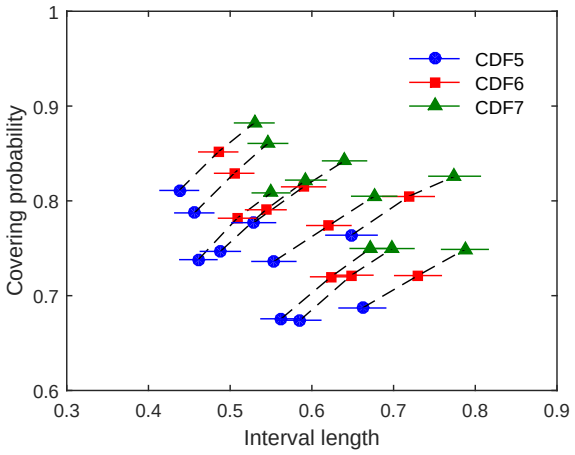
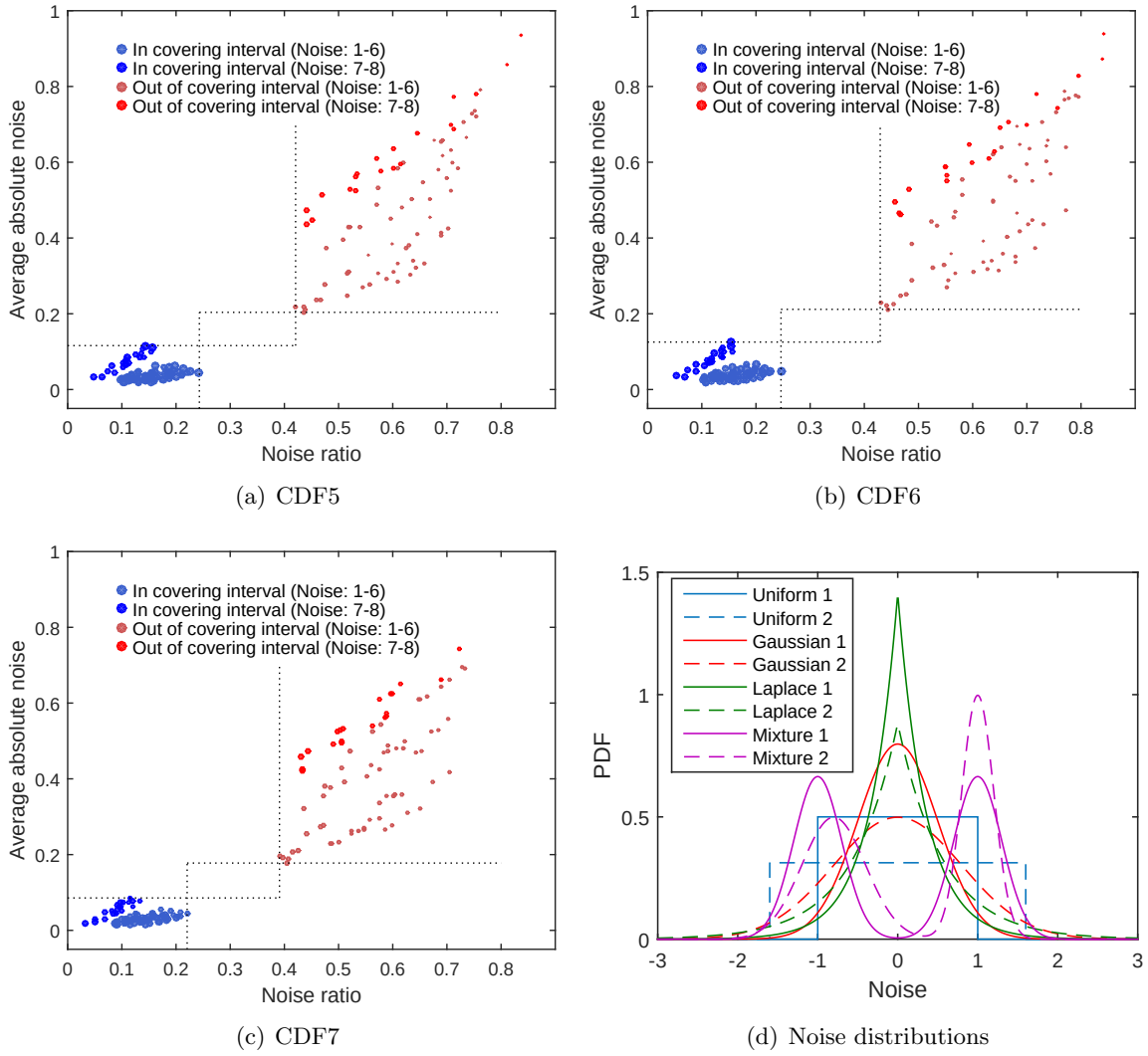


Figure 14: The interval length and covering probability

The original data set can be divided into two subsets according to whether the real output is in the covering interval (D_{in} and D_{out}). Figure 15 shows the noise characteristics of the two subsets. From Figure 15(a), (b) and (c), the noise ratio of D_{in} is below 25% and the average absolute noise of D_{in} is no more than 0.12 for all CDF filters. It is obvious that D_{in} has a lower noise ratio and a smaller average absolute noise than D_{out} for each data set and each noise distribution. There exists a clear gap between the scatters of D_{in} and D_{out}. This indicates that the covering interval could well separate low-noise samples from high-noise ones.



(a) CDF5 (b) CDF6

(c) CDF7 (d) Noise distributions

Figure 15: Performance of the covering interval is shown in the first three sub-figures, and predefined noise distributions are shown in the last sub-figure. There are 160 scatters (10 data sets $\times$ 8 noise distributions $\times$ 2 subsets, noise ratio=30%) in any one of the first three sub-figures. The center of a scatter is determined by the noise ratio and the average absolute noise of a subset (D_{in} or D_{out}). The radius of the red scatter denotes the size ratio of D_{out} to the original set, and the blue scatter represents that of D_{in}. The scatters in the legend have the maximum size ratio of 1. Besides, those for the two mixed noise distributions are in bright colors. The last sub-figure shows the probability density functions of all noise distributions.

As shown in Figure 15(d), the probability density functions (PDFs) of the last two mixed noise distributions (Noise: 7-8) are "low-head" and "high-shoulder" in shape, and it is opposite to the others' (Noise: 1-6). So their results are shown in different colors. Compared with the other noise distributions, the last two have larger noise variances. As a result, more samples will be out of the covering interval.

In Figure 15(a), (b) and (c), the scatters in bright red color have slightly larger radii than those in dark red color. Specifically, the size ratios of D_{out} for the first six noise distributions and the last two are 32.06%(±1.58%) and 40.20%(±0.82%) in CDF5, respectively. They are 29.67%(±1.43%) and 37.75%(±0.65%) in CDF6, 38.03%(±1.63%) and 47.43%(±0.86%) in CDF7. It is clear that D_{out} contains more samples for the last two noise distributions. More importantly, the gap between the scatters of D_{in} and D_{out} is larger for the two distributions. Thus the covering interval is applicable for complicated or asymmetric noise distributions.

It is worth noting that the size ratio of D_{in} in CDF5 is less than that of CDF6 (65.90%<68.31%) because the latter usually has a longer interval as shown in Figure 14. However, the CDF7 has a smaller D_{in} size (59.6%) than both CDF5 and CDF6. Although the covering interval in CDF7 is longer, the subset has fewer samples ($n/7$), and then the output noise might have more severe negative impacts on the constructed covering interval in CDF7. The noise estimation in CDF7 would also be affected for the same reason.

5.2.2 Performance of Covering Distance

The covering distance is a noise estimator, and the performance is evaluated by means of the real absolute deviation (RAD) and real relative deviation (RRD), where $RAD = \sum_{i=1}^{n} ||e_i| - R_i| / n$, $RRD = \sum_{i=1}^{n} \frac{||e_i|-R_i|}{n|e_i|}$. It is clear that they are the practical versions of the EAD in (38) and ERD in (39) .

Considering that the estimation performances on all data sets are similar to each other, only the result of CDF5 on data set *Parkinsons* is shown in Figure 16. More complete and detailed results can be found in Table 7.

- It can be observed from the first column of Figure 16 that scatters are around the ideal line except for the last two mixed noise distributions. The covering distance makes an under-estimation under the two mixed distributions. The reason is that the subsets scheme for constructing the covering interval is not very suitable for the two distributions. Specifically, most noises in the two mixtures have large values ("low-head" and "high-shoulder" PDF). The training and testing sets are partially overlapped in the subsets scheme. Then the covering interval and its center c deriving from kNN predictions are prone to get an evident shift towards the real output when there are too many large noises. As a result, the covering distance, proportional to $|y_i - c|$, under-estimates the noise. While the proportion of the large noise is small in other distributions, and it has an insignificant effect on the noise estimation.

- From the first column of Figure 16, the correlation coefficient has the following results: (1) The correlation coefficient R^2 in uniform 1 is less than that in uniform 2, and it is similar for the Gaussian or Laplace distribution. (2) R^2 in Gaussian 1 is less than that in Laplace 1, and it has the same result for Gaussian 2 and Laplace 2. The main reason is that the CD is better at estimating the large noise (unbiased). Specifically speaking,

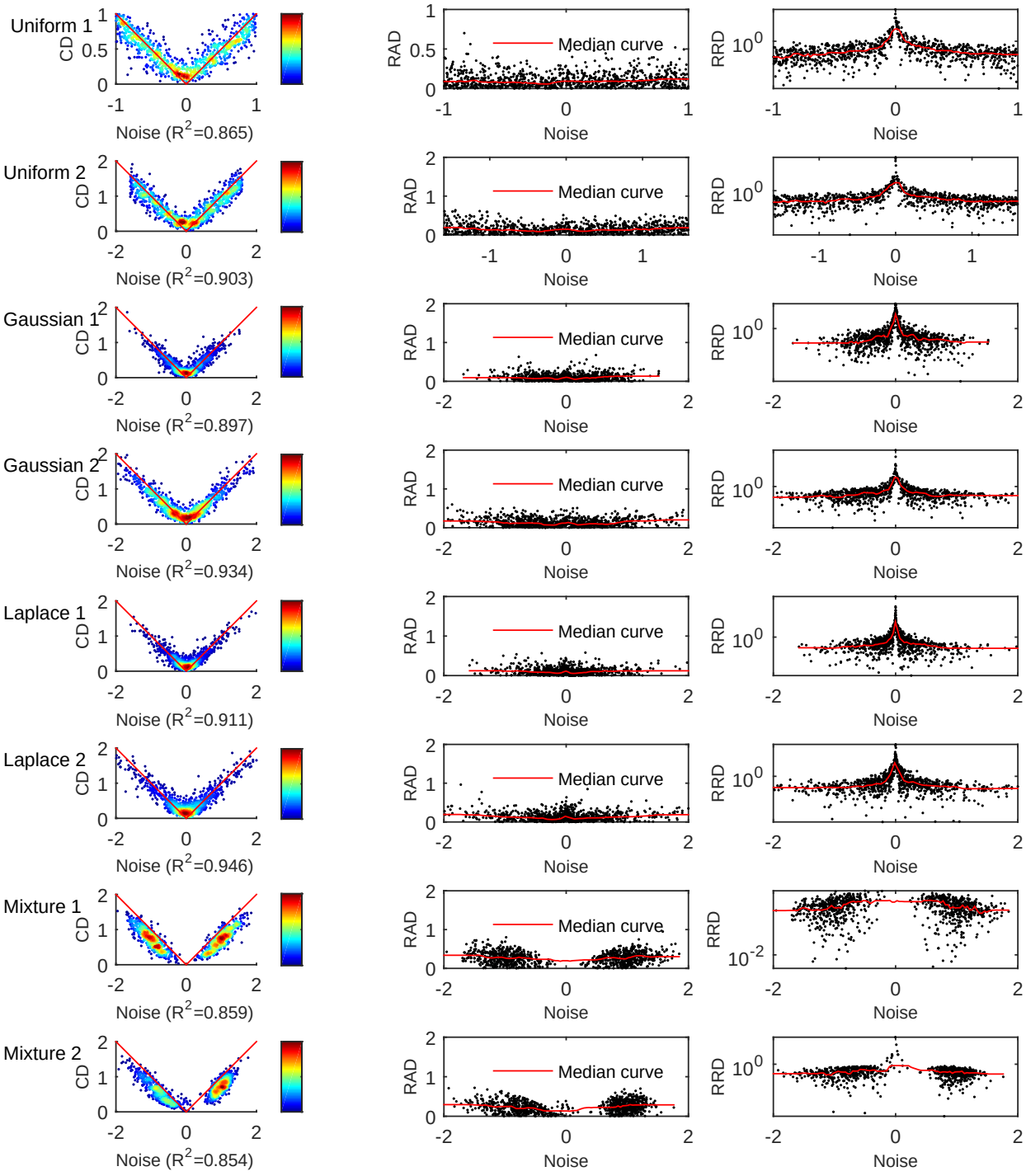


Figure 16: Performance of the covering distance in CDF5. The results for all noise distributions (noise ratio $NR = 30\%$) are displayed in rows. The first column shows the density plot of the noise vs. CD. The ideal line of $CD = |e_i|$ is added and R^2 denotes their correlation coefficient here. The second column shows the scatters of the noise and RAD. The distribution of the noise and RRD is displayed in the third column. The red curves in RAD and RRD denote the median of 99-nearest neighbors.

(1) the variance of the first uniform/Gaussian/Laplace distribution is less than that of the second, and more large noises should be generated from the distribution with a larger variance, so the correlation is more significant for a higher noise level. (2) For a fixed variance, the Laplace distribution has a heavier tail than the Gaussian distribution, and more large noises are expected to appear in the former.

- From the second column of Figure 16, the median curve is almost flat. That means the deviation of the CD estimator is stable about the noise. In addition, the median curve has a small peak in Laplace 1 and 2. These characteristics are generally consistent with the simulation results in Figure 8. The third column shows that the RRD generally decreases with the absolute noise for each distribution. This verifies the ERD results in Figure 8.

Table 7 lists the detailed results of the CDF noise estimation on data set *Parkinsons*. All values are averaged over five repetitions. The best value among the CDF filters is in bold font. CDF5 generally outperforms CDF6 and CDF7 in terms of the correlation coefficient, RAD and RRD. Rank-sum tests show that the difference between CDF5 and CDF6 is insignificant in both the correlation and RAD, but they have a significant difference in ERD. CDF6 and CDF7 have no significant difference in all indicators.

From the perspective of noise distribution, the RRD values for Gaussian and Laplace noises are larger than the others. This is because small noises, more likely to have large RRDs, account for a large proportion of the two kinds of noises. In addition, the last two mixtures have smaller correlation coefficients and larger RADs. It is related to the larger noise variance and the estimation bias induced by the subsets scheme.

Noise	Correlation coefficient (↑)			RAD (↓)			RRD (↓)		
distribution	CDF5	CDF6	CDF7	CDF5	CDF6	CDF7	CDF5	CDF6	CDF7
Uniform 1	**0.871**	0.869	0.865	**0.1124**	0.1140	0.1142	**71.26%**	81.45%	85.49%
Uniform 2	0.911	**0.914**	0.911	0.1545	**0.1499**	0.1514	**67.44%**	73.28%	69.52%
Gaussian 1	**0.890**	0.884	0.884	**0.1047**	0.1069	0.1080	**88.50%**	93.15%	98.66%
Gaussian 2	**0.931**	0.929	0.929	**0.1424**	0.1442	0.1440	**121.46%**	128.24%	136.90%
Laplace 1	**0.915**	0.914	0.909	**0.1086**	0.1108	0.1140	**210.58%**	240.49%	282.61%
Laplace 2	**0.943**	0.942	0.938	**0.1453**	0.1478	0.1520	**175.46%**	177.67%	197.29%
Mixture 1	**0.854**	0.852	**0.854**	0.2816	0.2816	**0.2788**	**22.95%**	25.40%	24.76%
Mixture 2	**0.858**	0.852	0.853	0.2497	**0.2468**	0.2491	**24.87%**	25.67%	26.64%

Table 7: Performance of CDF in noise estimation

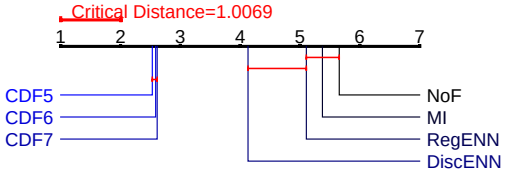
5.2.3 Prediction Error

Table 8 lists the average and the standard deviation (Std) of the prediction error MSE on ten data sets, five repetitions and eight noise distributions. The least error in each row (among filters) is bolded, and the least three errors have a light blue background. From the table, the best error is most likely to appear in the column of CDF5. Although CDF6 and CDF7 reach the minimum in three rows, their errors are very close to that of CDF5. Obviously, all three implements of the CDF algorithm have smaller errors than the others in most cases. From the perspective of the noise level, the prediction error usually increases with the noise ratio (NR) for any filter and any model. And the higher the noise ratio is, the larger the MSE reduction of CDF5 from NoF is. It means the filter is more effective

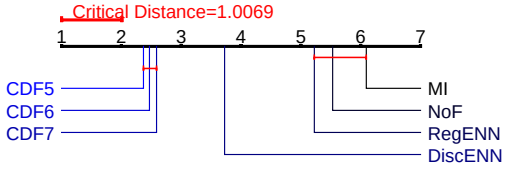
for the data set with a larger noise ratio. From the model perspective, kNN has the largest error and it is the most sensitive to noises. In addition, the error reductions of kNN and NNet are more significant than that of the RF model.

NR	Model	NoF	MI	RegENN	DiscENN	CDF5	CDF6	CDF7
10%	kNN	0.0837 (0.0142)	0.0784 (0.0153)	0.0738 (0.0146)	0.0742 (0.0239)	**0.0653** (0.0183)	0.0659 (0.0177)	0.0685 (0.0191)
	NNet	0.0636 (0.0251)	0.0517 (0.0259)	0.0511 (0.0263)	0.0525 (0.0180)	**0.0420** (0.0157)	0.0424 (0.0155)	0.0456 (0.0158)
	RF	0.0511 (0.0112)	0.0491 (0.0112)	0.0486 (0.0112)	0.0507 (0.0191)	**0.0458** (0.0137)	0.0459 (0.0135)	0.0504 (0.0147)
20%	kNN	0.0903 (0.0125)	0.0888 (0.0135)	0.0874 (0.0130)	0.0789 (0.0179)	**0.0671** (0.0148)	0.0680 (0.0147)	0.0717 (0.0157)
	NNet	0.0895 (0.0327)	0.0712 (0.0317)	0.0701 (0.0319)	0.0564 (0.0180)	**0.0469** (0.0157)	0.0474 (0.0160)	0.0493 (0.0154)
	RF	0.0525 (0.0109)	0.0503 (0.0111)	0.0495 (0.0107)	0.0516 (0.0150)	**0.0477** (0.0125)	0.0478 (0.0127)	0.0537 (0.0132)
30%	kNN	0.1174 (0.0137)	0.1146 (0.0146)	0.1133 (0.0140)	0.0854 (0.0211)	**0.0743** (0.0163)	0.0751 (0.0160)	0.0762 (0.0173)
	NNet	0.1139 (0.0370)	0.0955 (0.0428)	0.0944 (0.0437)	0.0698 (0.0210)	0.0593 (0.0173)	**0.0590** (0.0178)	0.0597 (0.0172)
	RF	0.0630 (0.0110)	0.0615 (0.0112)	0.0550 (0.0113)	0.0562 (0.0174)	**0.0513** (0.0137)	0.0515 (0.0136)	0.0547 (0.0143)
40%	kNN	0.1477 (0.0121)	0.1225 (0.0125)	0.1221 (0.0121)	0.0942 (0.0191)	0.0856 (0.0160)	0.0877 (0.0156)	**0.0848** (0.0165)
	NNet	0.1490 (0.0482)	0.1068 (0.0529)	0.1064 (0.0525)	0.0914 (0.0235)	0.0740 (0.0167)	0.0766 (0.0161)	**0.0726** (0.0162)
	RF	0.0772 (0.0084)	0.0640 (0.0087)	0.0677 (0.0085)	0.0600 (0.0157)	**0.0576** (0.0124)	0.0589 (0.0123)	0.0604 (0.0132)

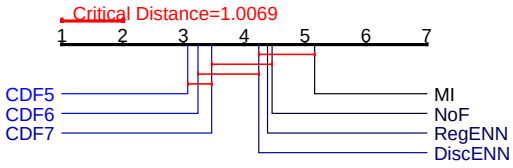
Table 8: MSE (Std) under different noise ratios



(a) MSE of kNN



(b) MSE of NNet



(c) MSE of RF

Figure 17: Critical difference diagram of MSE

Figure 17 compares the MSEs of all filters by means of the critical difference diagram. For kNN and NNet, the three CDF filters significantly outperform the others in terms of MSE. The ranks of filters in the RF model are closer to each other (range from 3 to 5), and

both CDF6 and CDF7 have no significant difference with a few existing filters. Generally, CDF5 reduces the prediction errors of all models and performs better.

The relative error reduction (RER) of CDF5 is defined by $1 - MSE_{CDF5}/MSE_{NoF}$. Table 9 lists the average RER over ten data sets, five repetitions and four noise ratios. The RF model has the lowest RER and the NNet has the largest RER. It means that the NNet is more sensitive to the artificial noise on benchmark data sets. The performance of NNet here differs from that on apparent age data set (Table 3). It might be because of the noise level and sample size. The apparent age set has more than ten thousands of training data with a low noise level. Whereas the result in Table 9 is based on the benchmark data sets with high noise ratios and most of them have no more than 2000 samples.

Noise	Noise	Model		
distribution	variance	kNN	NNet	RF
Uniform 1	0.5	10.2%	23.3%	4.8%
Uniform 2	0.8	43.7%	51.7%	8.3%
Gaussian 1	0.5	8.7%	28.2%	6.8%
Gaussian 2	0.8	34.1%	50.2%	12.4%
Laplace 1	0.5	9.7%	32.0%	5.9%
Laplace 2	0.8	36.3%	61.7%	14.3%
Mixture 1	1.0	44.2%	52.0%	33.3%
Mixture 2	1.0	39.9%	44.5%	29.0%

Table 9: Relative error reduction of CDF5

From Table 9, the RERs of all models increase with the noise variance for a given type of noise distribution. It means the effectiveness of CDF5 is more evident on the data set with a high noise level. The last two noise distributions have the same variance. The mixture 1 is symmetric and the mixture 2 is asymmetric. The RER in mixture 2 is less than that in mixture 1 for the same model. It indicates the CDF filter is more suitable for the noise from a symmetric distribution.

Furthermore, the RER differences with respect to the model, the noise distribution and variance are examined by the statistical test. The result of a three-way analysis of variance indicates that the RER mainly depends on the noise variance (P-value=0.00) and the model (P-value=0.00). The type of noise distribution has no significant impact on the RER (P-value=0.69). It means that CDF5 is effective for some complicated and asymmetric noise distributions.

5.3 Results and Analysis in Ordinal Classification

The experimental results are analyzed in the aspects of noise recognition, prediction error and efficiency.

5.3.1 Noise Recognition

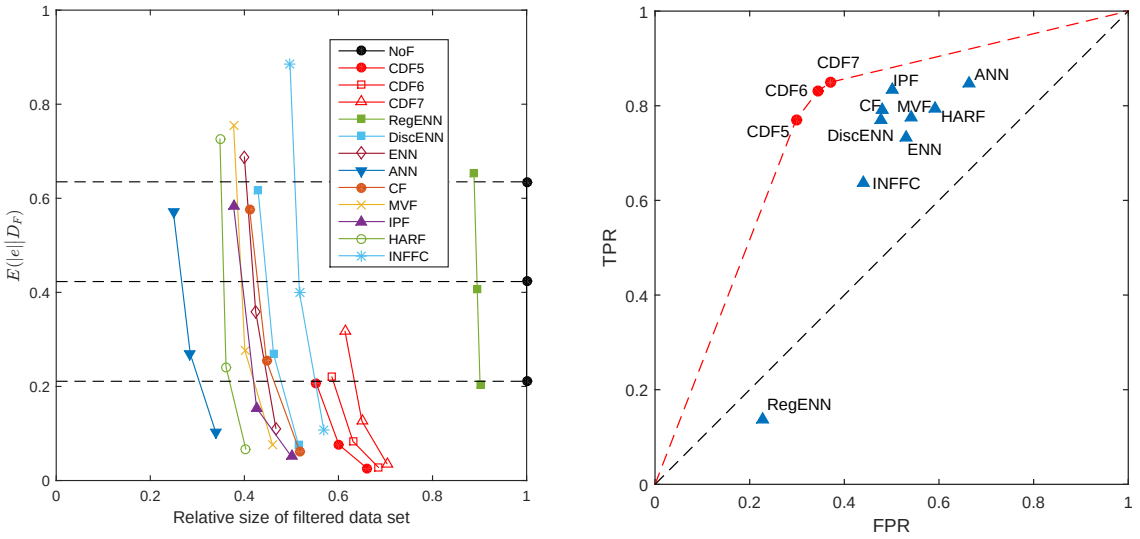
The performance in noise recognition is evaluated by the average absolute noise of filtered data set and the ROC space. The average absolute noise is calculated by

$$\mathbb{E}(|e|\big|D_F) = \frac{1}{n_F}\sum_{i=1}^{n_F}|\mathcal{O}(y_i) - \mathcal{O}(y_i^0)|,$$

where $\mathcal{O}(\cdot)$ is the rank function. As the noise recognition problem can be considered as a binary classification problem, a new confusion matrix can be constructed similarly, then all filters can be compared in ROC space. The confusion matrix for noise filtering is shown in Table 10. The true positive rate $TPR = \frac{TP}{TP+FN}$ and false positive rate $FPR = \frac{FP}{FP+TN}$.

		Predicted	
		noisy	clean
Actual	noisy	True Positive (TP)	False negative (FN)
	clean	False positive (FP)	True negative (TN)

Table 10: Confusion matrix in noise filtering



(a) Noise level and relative size

(b) Noise recognition in ROC space

Figure 18: Performance in noise recognition

Figure 18 shows the performances of filters in noise recognition. Figure 18(a) displays the average absolute noise and relative size on 17 filtered data sets and five repetitions. The original noise levels (NoF) are denoted by the black solid dots. The dot moves from the bottom left to the top right corner for each filter when the noise ratio (NR) varies from 10% to 30% ($NR = 10\%, 20\%, 30\%$). Generally, the relative size decreases with the NR for all filters. It indicates that more samples will be dropped for a higher NR. From Figure 18(a), all relative sizes are less than the original size, but not all noise levels become lower. HARF, MVF, ENN INFFC, and RegENN are above the highest dotted line (the highest dot of NoF), and it means they do not reduce the noise level when $NR = 30\%$. Although ANN brings down the noise levels for all NRs, it removes about 70% of the samples and may destroy the original data distribution. On the contrary, RegENN retains about 90% of the samples but it almost does not change the noise level. Compared with the two extreme filters, IPF, CF, and DiscENN not only obtain evident noise reductions but also retain more samples than ANN. In another word, they are good choices among existing filters in

noise recognition. Besides, all CDF filters get lower noise levels and larger relative sizes than existing ones except RegENN. It indicates that the CDF could reduce the noise level significantly at a lower cost of data removing.

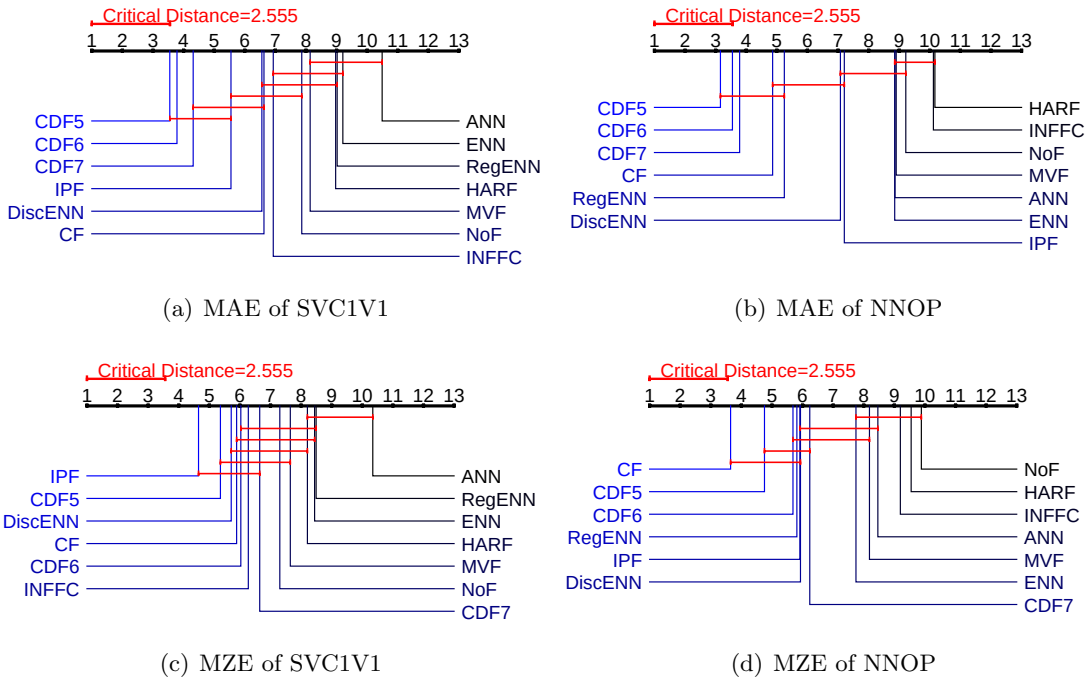
Figure 18(b) shows the average TPR and FPR values on 17 data sets, five repetitions and three noise ratios. The existing and proposed filters are marked by blue triangles and red dots, respectively. From Figure 18(b), RegENN is under the diagonal. This means the RegENN filter is inferior to a random filtering if the noise quantity ($|\mathcal{O}(y_i) - \mathcal{O}(y_i^0)|$) is out of consideration. ANN has the biggest FPR because of a large amount of false positive samples, i.e., many clean samples are removed in the ANN filtering. It is known that the ideal filter should be near the upper left corner of the ROC space. Hence IPF, CF, and DiscENN should be good choices among existing filters. These are consistent with the results of noise level reduction in Figure 18(a). Besides, CDF5 is the filter with the least FPR except for RegENN, and CDF7 has the largest TPR. Three CDF filters are located at the left of IPF, CF, and DiscENN which are outstanding among existing filters. It means that the CDFs have smaller false positive rates and lower risks of overcleansing. Therefore, the CDF filters outperform the others in the noise recognition of ordinal classification.

5.3.2 Prediction Error

Two indicators (MAE, MZE) are employed for evaluating the prediction performance of the ordinal classification model (SVC1V1, NNOP) after filtering. Figure 19 shows the critical difference diagram of the prediction error in ordinal classification. From Figure 19(a) and (b), the CDF filters are superior to the others in terms of MAE both for SVC1V1 and NNOP. IPF, CF, and DiscENN obtain small MAE values among existing filters owing to their good performances in noise recognition. In addition, the CDF filters have no significant difference with IPF (SVC1V1) or CF (NNOP) in terms of MAE.

From Figure 19(c) and (d), IPF, CF, and DiscENN are good choices among existing filters from the aspect of MZE. It can be observed that both CDF5 and CDF6 are inferior to IPF in SVC1V1 or CF in NNOP, but the four filters have no obvious difference for both models. Clearly, the MAE ranks of the CDF filters are different from their MZE ranks. The main reason is that MAE uses an absolute cost, while MZE considers a zero-one loss. Specifically, the CDF algorithm focuses on the noise quantity (the rank distance of y_i from y_i^0), and the MAE indicator measures the error quantity (the rank distance of $\hat{y}_i$ from y_i). Both CDF and MAE adopt the quantitative analysis, so the CDF algorithm outperforms IPF and CF in terms of MAE. Whereas the filters for classification, including IPF and CF, care about whether it is a noise label ($y_i \neq y_i^0$), and the MZE indicator represents the probability of correct predictions($\hat{y}_i \neq y_i$). They consider the qualitative relation but not the quantity, so IPF and CF perform better in terms of MZE.

It is worth noting that many filters are inferior to NoF in terms of both indicators of SVC1V1 in Figure 19(a) and (c). It means that not all filters are effective in improving the prediction ability. Although RegENN does not perform well in noise recognition, it is superior to IPF and DiscENN in terms of both indicators of NNOP. It might because RegENN keeps most of the samples (90% in Figure 18(a)) and the NNOP model is less sensitive to the label noise when there are enough samples.



(a) MAE of SVC1V1

(b) MAE of NNOP

(c) MZE of SVC1V1

(d) MZE of NNOP

Figure 19: Critical difference diagram of model errors in ordinal classification

5.3.3 Efficiency

Figure 20 shows the average runtime of all filters on five repetitions for each sample size. Note that there are 11 sample sizes in 17 data sets as some sets have the same size. The runtime of the CDF filter includes all steps in Algorithm 1. The runtime generally increases with the sample size for all filters. The wave in the curve is mainly from the variation of the feature number. The comparison result is clearer when the sample size is over 3000. Generally, the filters can be divided into three sets. Neighbor-based filters, including ANN, ENN, RegENN, DiscENN, usually have low efficiencies. Ensemble-based filters, including INFFC, HARF, MVF, IPF, and CF, are more efficient than neighbor-based filters. The proposed CDF filters have smaller runtime and could complete the filtering on tens of thousands of samples within one second.

5.4 Filtering on Real Benchmark Data Set

In order to explore the effectiveness of the CDF algorithm in real problems, the original data set without artificial noise ($NR = 0\%$) is processed by CDF5. The discretized data sets in Table 6 (Nos. 1-10) are not considered here. The filtering procedure is repeated 20 times for each data set owing to the randomness in the subsets scheme. The prediction error (MSE in regression, MAE in ordinal classification) after the CDF5 filtering is compared with that of NoF (no filtering). Table 11 lists the winning probability (WinP) of CDF5, i.e., CDF5 testing error<NoF testing error. The probabilities over 0.8 are bolded. If there exists a significant difference between the two sets of errors (rank-sum test, significant level

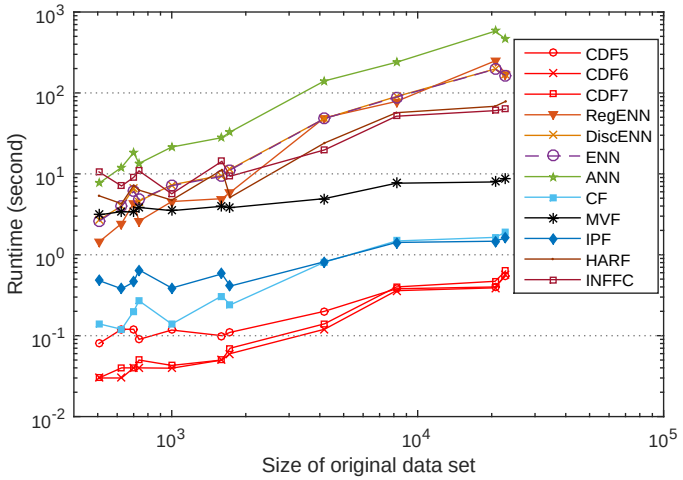


Figure 20: Runtime comparison

0.05), a bullet is added after the probability. The relative size of filtered data set is also displayed. The relative time denotes the ratio of the total time with filtering (including the CDF5 filtering on D_{Ap}, training on D_{Af} and testing on D_B in Figure 13) to that without filtering (including training on D_{Ap} and testing on D_B).

Task (Model)	Data	Sample size	Relative size	Model 1 WinP	Model 1 Relative time	Model 2 WinP	Model 2 Relative time
Regression (RF,NNet)	1	308	97%	0.50	97%	**0.85•**	94%
	2	506	96%	0.55	94%	0.50	96%
	3	768	95%	**0.80**	94%	**0.95•**	73%
	4	1030	92%	0.50	94%	0.70•	74%
	5	1059	91%	**0.90•**	93%	0.45	98%
	6	1385	92%	0.60	90%	0.70	79%
	7	1503	90%	0.65	92%	**0.85•**	80%
	8	3107	92%	**1.00•**	94%	0.65	90%
	9	3395	94%	**0.95•**	90%	0.65	91%
	10	5875	90%	**1.00•**	90%	0.55	80%
Ordinal classification (SVC1V1,NNOP)	11	625	95%	0.50	99%	0.60	96%
	12	1000	86%	0.45	93%	0.40	93%
	13	1728	95%	0.55	97%	0.55	97%
	14	736	85%	0.45	93%	0.60	95%
	15	1000	87%	**0.95**	92%	0.50	91%
	16	1599	87%	**0.80**	85%	0.60	91%
	17	1000	80%	**1.00•**	84%	0.55	87%

Table 11: Effectiveness of CDF5 on real data sets

It is clear from Table 11 that most bullets correspond to large WinP values. For the first model (RF in regression, SVC1V1 in classification), CDF5 is effective for data sets Nos. 5, 8-10, 17. It means the filter is likely to be effective for large-size data sets in regression and those with more categories in ordinal classification. For the neural networks models (NNet in regression, NNOP in classification), CDF5 is effective for data sets Nos. 1, 3, 4, 7. The proposed filter does not improve the prediction ability of NNOP significantly. It implies

that the neural networks model could work with low-noise data sets in classification, while it does not perform well for some regression data sets with a small sample size.

From Table 11, there is no bullet on two regression data sets (Nos. 2, 6) and most ordinal classification data sets. It means the CDF is better at filtering on the regression problem. For the cases without the bullet, the rank-sum test shows that there is no significant difference before and after the CDF5 filtering. It indicates that CDF5 does not reduce the model prediction ability significantly, and thus it is a safe filtering algorithm.

The relative size is over 0.95 when the sample size is less than 1000 in regression. It verifies the adaptability of the proposed OSS framework with respect to the sample size. Besides, all relative time is less than 100%. It means the strategy of training and testing with filtering is more efficient than that without filtering.

The above results indicate that the CDF filtering always could significantly improve the prediction ability on benchmark data sets with artificial noises, even though the model is less sensitive to noise. Whereas it is not effective for all original benchmark data sets. It means the proposed filter is more suitable for the data set with a large noise ratio, such as raw samples collected from crowdsourcing systems or search engines. In other words, the improvement of prediction ability after filtering partially depends on the noise level and the robustness of the learning model apart from the accuracy of noise estimation. Compared with the CDF filtering, many noise-robust models are less efficient and require certain prior knowledge and manually tuned hyper-parameters.

In brief, the CDF filter (CDF5 or CDF6 is advisable) could significantly improve the prediction ability of the model trained on a data set unless the data set is low-noise or noise-free and the model is robust enough. Thus it can serve as an important auxiliary tool to improve both data quality and model prediction.

6. Conclusion

Although various noise filters have been presented to deal with the output noise, the effectiveness and the influence of the filtering have not been studied carefully from the perspective of error bound. This paper answers three essential problems in noise filtering: whether a filter works, how many and which samples should be filtered. The theoretical foundations for the determination of effective noise filtering and the optimal sample selection are provided, and then a unified framework of the output noise filtering, which can be integrated with any noise estimator, is built. More importantly, the OSS framework is adaptable to noisy environments and could prevent filters from overcleansing. It may provide a novel way to optimize the hyper-parameter in other filters. Experimental results on real-world image data and benchmark data sets demonstrate that the proposed CDF filter could significantly reduce the noise level and outperforms the state-of-the-art filters in prediction ability and efficiency. These results indicate that the OSS framework and the CDF can be used as important auxiliary tools to improve data quality and model prediction.

In the CDF filtering, the noise estimation for the mixed or asymmetric noise distributions seems to be slightly biased, and the CD estimator could be improved. The covering distance is a quantitative measure of the output quality. It may be beneficial to some label-related tasks like crowdsourcing learning, self-paced learning, and semi-supervised learning.

Besides, the filtering for a general classification data set with the label noise may be reconsidered from the error-bound perspective. All these deserve further research.

Acknowledgments

The authors greatly thank the handling associate editor and all anonymous reviewers for their valuable comments. We also thank Men Changqian and Liang Yudong for their valuable suggestions. This work was partially supported by the National Natural Science Foundation of China (Nos. 62076154, 61906113, U1805263, 61673249), National Key R&D Program of China (Nos. 2020AAA0106100, 2018YFB1004300), Key R&D program of Shanxi Province (International Cooperation, 201903D421050), and Scientific and Technological Innovation Programs of Higher Education Institutions in Shanxi (No. 2020L0007).

A. Appendices

A.1 Proof of Theorem 1

Proof Let D_F be the filtered data set from the original set $D = \{(x_i, y_i), i = 1, 2, \ldots, n\}$. y_i is the real output and y_i^0 is the true output. $m(x)$ and $m_F(x)$ are the models trained on them. The noise $e_i = y_i - y_i^0$, the error $r_i = m(x_i) - y_i$.

In learning problems with accurate outputs, the empirical risk measures the difference between the model predictions and real outputs. The true empirical risk should represent the loss of model predictions about the true outputs in the learning with noisy outputs,

$$\begin{aligned}\mathcal{R}_{emp}(m, D) &= \frac{1}{n}\sum_{x_i \in D}[m(x_i) - y_i^0]^2 \\ &= \frac{1}{n}\sum_{x_i \in D}[(m(x_i) - y_i) + e_i]^2 \\ &= \frac{1}{n}\sum_{x_i \in D}\left\{[m(x_i) - y_i]^2 + e_i^2 + 2e_i \cdot r_i\right\}.\end{aligned}$$

By the symmetry of noise distribution, we have $\mathbb{E}_D(e_i) = 0$. Then

$$\begin{aligned}\mathcal{R}_{emp}(m, D) &= \frac{1}{n}\sum_{x_i \in D}\left[(m(x_i) - y_i)^2\right] + \mathbb{E}_D(e_i^2) + 2\mathbb{E}_D(e_i)\mathbb{E}_D(r_i) \\ &= \mathbb{E}_D\left[(m(x_i) - y_i)^2\right] + \mathbb{E}_D(e_i^2).\end{aligned}$$

Similarly, we have $\mathcal{R}_{emp}(m_F, D_F) = \mathbb{E}_{D_F}\left[(m_F(x_i) - y_i)^2\right] + \mathbb{E}_{D_F}(e_i^2)$. Then

$$\begin{aligned}\mathcal{R}_{emp}(m_F, D_F) \cdot \epsilon(D_F) < \mathcal{R}_{emp}(m, D) \cdot \epsilon(D) &\Leftrightarrow \frac{\mathcal{R}_{emp}(m_F, D_F)}{\mathcal{R}_{emp}(m, D)} < \frac{\epsilon(D)}{\epsilon(D_F)} \\ &\Leftrightarrow \frac{\mathbb{E}\left[(m_F(x_i) - y_i)^2 \middle| D_F\right] + \mathbb{E}(e_i^2 \middle| D_F)}{\mathbb{E}\left[(m(x_i) - y_i)^2 \middle| D\right] + \mathbb{E}(e_i^2 | D)} < \frac{\epsilon(D)}{\epsilon(D_F)},\end{aligned}$$

where $\epsilon(D)$ and $\epsilon(D_F)$ are defined in (6) and (9), respectively.

If the outputs in D_F and D are from the same distribution, they should have the same variation, i.e., $\mathbb{E}_{D_F}[(y_i - \sum y_i/n)^2] = \mathbb{E}_D[(y_i - \sum y_i/n)^2]$. For any given data set and a fixed goodness of fit $(1 - \frac{\sum_i [m(x_i)-y_i]^2}{\sum_i (y_i - \sum y_i/n)^2})$, it can be assumed that $\mathbb{E}_{D_F}[(m_F(x_i) - y_i)^2] = \mathbb{E}_D[(m(x_i) - y_i)^2] = C \cdot \mathbb{E}_D(e_i^2)$, where C is a positive coefficient. Then we have

$$\begin{aligned}
\mathcal{R}_{emp}(m_F, D_F) \cdot \epsilon(D_F) < \mathcal{R}_{emp}(m, D) \cdot \epsilon(D) \Leftrightarrow & \frac{\mathbb{E}_{D_F}[(m_F(x_i) - y_i)^2] + \mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D[(m(x_i) - y_i)^2] + \mathbb{E}_D(e_i^2)} < \frac{\epsilon(D)}{\epsilon(D_F)} \\
\Leftrightarrow & \frac{C \cdot \mathbb{E}_D(e_i^2) + \mathbb{E}_{D_F}(e_i^2)}{C \cdot \mathbb{E}_D(e_i^2) + \mathbb{E}_D(e_i^2)} < \frac{\epsilon(D)}{\epsilon(D_F)} \\
\Leftrightarrow & \frac{C \cdot \mathbb{E}_D(e_i^2) + \mathbb{E}_{D_F}(e_i^2)}{(1+C) \cdot \mathbb{E}_D(e_i^2)} < \frac{\epsilon(D)}{\epsilon(D_F)} \\
\Leftrightarrow & C + \frac{\mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D(e_i^2)} < \frac{\epsilon(D)}{\epsilon(D_F)}(1+C) \\
\Leftrightarrow & \frac{\mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D(e_i^2)} < \frac{\epsilon(D)}{\epsilon(D_F)}(1+C) - C,
\end{aligned}$$

where the coefficient $C = \frac{\mathbb{E}_D(r_i^2)}{\mathbb{E}_D(e_i^2)} > 0$. ■

A.2 Proof of Theorem 2

Proof As proved in A.1, $\mathcal{R}_{emp}(m, D) = \mathbb{E}_D\left[(m(x_i) - y_i)^2\right] + \mathbb{E}_D(e_i^2)$ and $\mathcal{R}_{emp}(m_F, D_F) = \mathbb{E}_{D_F}\left[(m_F(x_i) - y_i)^2\right] + \mathbb{E}_{D_F}(e_i^2)$. $\mathbb{E}_{D_F}[(m_F(x_i) - y_i)^2] = \mathbb{E}_D[(m(x_i) - y_i)^2] = C \cdot \mathbb{E}_D(e_i^2)$ hold for a fixed goodness of fit, where C is a positive coefficient. Then

$$\begin{aligned}
& \mathcal{R}_{emp}(m_F, D_F) \cdot \epsilon(D_F) - \mathcal{R}_{emp}(m, D) \cdot \epsilon(D) \\
= & \left(\frac{\mathcal{R}_{emp}(m_F, D_F)}{\mathcal{R}_{emp}(m, D)} - \frac{\epsilon(D)}{\epsilon(D_F)} \right) \cdot \mathcal{R}_{emp}(m, D) \cdot \epsilon(D_F) \\
= & \left(\frac{\mathbb{E}_{D_F}\left[(m_F(x_i) - y_i)^2\right] + \mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D\left[(m(x_i) - y_i)^2\right] + \mathbb{E}_D(e_i^2)} - \frac{\epsilon(D)}{\epsilon(D_F)} \right) \cdot \mathcal{R}_{emp}(m, D) \cdot \epsilon(D_F) \\
= & \left(\frac{C \cdot \mathbb{E}_D(e_i^2) + \mathbb{E}_{D_F}(e_i^2)}{C \cdot \mathbb{E}_D(e_i^2) + \mathbb{E}_D(e_i^2)} - \frac{\epsilon(D)}{\epsilon(D_F)} \right) \cdot \mathcal{R}_{emp}(m, D) \cdot \epsilon(D_F) \\
= & \left(\frac{C + \mathbb{E}_{D_F}(e_i^2)/\mathbb{E}_D(e_i^2)}{1+C} - \frac{\epsilon(D)}{\epsilon(D_F)} \right) \cdot \mathcal{R}_{emp}(m, D) \cdot \epsilon(D_F) \\
= & \left(C + \frac{\mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D(e_i^2)} - \frac{\epsilon(D)}{\epsilon(D_F)}(1+C) \right) \cdot \frac{\mathcal{R}_{emp}(m, D) \cdot \epsilon(D_F)}{1+C} \\
= & \left(\frac{\mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D(e_i^2)} - \left[\frac{\epsilon(D)}{\epsilon(D_F)}(1+C) - C \right] \right) \cdot \frac{\mathcal{R}_{emp}(m, D) \cdot \epsilon(D_F)}{1+C} \\
= & \left[T(\rho) - \mathcal{B}_T(\rho) \right] \cdot \epsilon(D_F) \cdot \frac{\mathcal{R}_{emp}(m, D)}{1+C},
\end{aligned}$$

where $T(\rho)$ and $\mathcal{B}_T(\rho)$ are defined in (10) and (11), respectively.

For any given data set and model, the empirical risk $\mathcal{R}_{emp}(m, D)$, $\epsilon(D)$, and the coefficient C are fixed. So we have

$$\begin{aligned}\min \mathcal{R}_{emp}(m_F, D_F) \cdot \epsilon(D_F) &\Leftrightarrow \min \mathcal{R}_{emp}(m_F, D_F) \cdot \epsilon(D_F) - \mathcal{R}_{emp}(m, D) \cdot \epsilon(D) \\ &\Leftrightarrow \min \left[T(\rho) - \mathcal{B}_T(\rho)\right] \cdot \epsilon(D_F) \cdot \frac{\mathcal{R}_{emp}(m, D)}{1 + C} \\ &\Leftrightarrow \min \left[T(\rho) - \mathcal{B}_T(\rho)\right] \cdot \epsilon(D_F) \\ &\Leftrightarrow \max \left[\mathcal{B}_T(\rho) - T(\rho)\right] \cdot \epsilon(D_F).\end{aligned}$$

■

A.3 Proof of Property 1

Proof (1) Note that all noises are sorted by the absolute value in ascending order in the calculation of the true $T(\rho)$. We know that all samples in D_F are noise-free if the relative size $\rho < 1 - NR$ (NR denotes the noise ratio). So we have $\mathbb{E}_{D_F}(e_i^2) = 0$ and $T(\rho < 1 - NR) = 0$. When the relative size $\rho > 1 - NR$, there must exist at least one noisy sample in D_F. So $\mathbb{E}_{D_F}(e_i^2) > 0$ and $T(\rho > 1 - NR) > 0$.

(2) Let $f(e)$ be the probability density function (PDF) of the noise, and $f_a(|e|)$ denotes the PDF of the absolute noise. It is known that $T(\rho) = 0$ and $\frac{\partial T(\rho)}{\partial NR} = 0$ when $\rho < 1 - NR$.

For $\rho > 1 - NR$,

$$\begin{aligned}T(\rho) &= \frac{\mathbb{E}_{D_F}(e_i^2)}{\mathbb{E}_D(e_i^2)} \\ &= \frac{(1 - NR) \cdot 0 + (\rho + NR - 1) \cdot \int_{|e|<Q_\rho} f(e) \cdot e^2 de}{(1 - NR) \cdot 0 + NR \cdot \int_{|e|<+\infty} f(e) \cdot e^2 de} \\ &= \frac{(\rho + NR - 1) \cdot \int_{|e|<Q_\rho} f(e) \cdot e^2 de}{NR \cdot \int_{|e|<+\infty} f(e) \cdot e^2 de} \\ &= \frac{(\rho + NR - 1) \cdot \int_0^{Q_\rho} f_a(|e|) \cdot |e|^2 d|e|}{NR \cdot \int_{|e|<+\infty} f(e) \cdot e^2 de} \\ &= \frac{1}{\int_{|e|<+\infty} f(e) \cdot e^2 de} \cdot \frac{NR + \rho - 1}{NR} \cdot \int_0^{Q_\rho} f_a(|e|) \cdot |e|^2 d|e| \\ &= \frac{1}{U} \cdot \frac{NR + \rho - 1}{NR} \cdot U_\rho,\end{aligned}$$

where $U = \int_{|e|<+\infty} f(e) \cdot e^2 de > 0, U_\rho = \int_0^{Q_\rho} f_a(|e|) \cdot |e|^2 d|e| > 0$. Q_ρ is the $\rho + NR - 1$ percentile of $f_a(|e|)$, and Q_ρ increases with ρ and NR, i.e., $\frac{\partial Q_\rho}{\rho} > 0, \frac{\partial Q_\rho}{NR} > 0$.

Note that U is independent of ρ, we have

$$\frac{\partial T(\rho)}{\partial \rho} = \frac{1}{U} \cdot \left[\frac{1}{NR} \cdot U_\rho + \frac{NR + \rho - 1}{NR} \cdot \frac{\partial U_\rho}{\partial \rho}\right]$$

$$= \frac{1}{U} \cdot \left[\frac{1}{NR} \cdot U_\rho + \frac{NR+\rho-1}{NR} \cdot \frac{\partial U_\rho}{\partial Q_\rho} \cdot \frac{\partial Q_\rho}{\partial \rho} \right]$$
$$= \frac{1}{U} \cdot \left[\frac{1}{NR} \cdot U_\rho + \frac{NR+\rho-1}{NR} \cdot f_a(Q_\rho) \cdot Q_\rho^2 \cdot \frac{\partial Q_\rho}{\partial \rho} \right] > 0.$$

(3) Let $\{|e'_1|, |e'_2|, \cdots, |e'_{i-1}|, |e'_i|, |e'_{i+1}|, \cdots, |e'_n|\}$ be the set of sorted absolute noises ($|e'_i| < |e'_{i+1}|, \forall i$). Then $T(\rho = i/n) = \frac{\sum_{t=1}^{i} |e'_t|^2}{\sum_{t=1}^{n} |e'_t|^2}$.

For any $i = 2, 3, \cdots, n-1$,

$$\begin{aligned} T\left(\rho = \frac{i+1}{n}\right) + T\left(\rho = \frac{i-1}{n}\right) &= \frac{\sum_{t=1}^{i+1} |e'_t|^2 + \sum_{t=1}^{i-1} |e'_t|^2}{\sum_{t=1}^{n} |e'_t|^2} \\ &= \frac{2\sum_{t=1}^{i-1} |e'_t|^2 + |e'_i|^2 + |e'_{i+1}|^2}{\sum_{t=1}^{n} |e'_t|^2} \\ &> \frac{2\sum_{t=1}^{i-1} |e'_t|^2 + |e'_i|^2 + |e'_i|^2}{\sum_{t=1}^{n} |e'_t|^2} \\ &= \frac{2\sum_{t=1}^{i} |e'_t|^2}{\sum_{t=1}^{n} |e'_t|^2} \\ &= 2T\left(\rho = \frac{i}{n}\right). \end{aligned}$$

Similarly, $\lambda T(\rho_1) + (1-\lambda)T(\rho_2) > T[\lambda\rho_1 + (1-\lambda)\rho_2]$ holds for $0 < \rho < 1$ when $n \to +\infty$. So we get the desired result by the definition of convex function.

(4) When $\rho > 1 - NR$,

$$\begin{aligned} \frac{\partial T(\rho)}{\partial NR} &= \frac{1}{U} \cdot \left[\frac{1-\rho}{NR^2} \cdot U_\rho + \frac{NR+\rho-1}{NR} \cdot \frac{\partial U_\rho}{\partial NR} \right] \\ &= \frac{1}{U} \cdot \left[\frac{1-\rho}{NR^2} \cdot U_\rho + \frac{NR+\rho-1}{NR} \cdot \frac{\partial U_\rho}{\partial Q_\rho} \cdot \frac{\partial Q_\rho}{\partial NR} \right] \\ &= \frac{1}{U} \cdot \left[\frac{1-\rho}{NR^2} \cdot U_\rho + \frac{NR+\rho-1}{NR} \cdot f_a(Q_\rho) \cdot Q_\rho^2 \cdot \frac{\partial Q_\rho}{\partial NR} \right] > 0. \end{aligned}$$

When $\rho < 1 - NR$, $\frac{\partial T(\rho)}{\partial NR} = 0$. Therefore, $\frac{\partial T(\rho)}{\partial NR} \geq 0$.

(5) Note that $\mathbb{E}(e) = 0$ holds for any symmetric noise distribution, then the noise variance $\sigma^2 = \mathbb{E}(e^2) - \mathbb{E}^2(e) = \mathbb{E}(e^2) = U$.

For $\rho > 1 - NR$,

$$T(\rho) = \frac{1}{U} \cdot \frac{NR+\rho-1}{NR} \cdot U_\rho = \frac{NR+\rho-1}{NR \cdot \sigma^2} \cdot U_\rho,$$

where $U = \int_{|e|<+\infty} f(e) \cdot e^2 de > 0, U_\rho = \int_0^{Q_\rho} f_a(|e|) \cdot |e|^2 d|e| > 0$, and Q_ρ is the $\rho + NR - 1$ percentile of $f_a(|e|)$.

(a) The PDF of a symmetric Gaussian distribution $f(e) = \frac{1}{\sqrt{2\pi}\sigma}\exp(-\frac{e^2}{2\sigma^2})$, and $f_a(|e|) = 2f(e), e > 0$. Then we have

$$\begin{aligned}
T(\rho) &= \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot U_\rho \\
&= \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot\int_0^{Q_\rho} f_a(|e|)\cdot|e|^2 d|e| \\
&= \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot\int_0^{Q_\rho} 2f(e)\cdot e^2 de \\
&= \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot\int_0^{Q_\rho} \frac{2}{\sqrt{2\pi}\sigma}\exp(-\frac{e^2}{2\sigma^2})\cdot e^2 de \\
&\overset{t=e/\sigma}{=\!=\!=} \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot\int_0^{Q_\rho/\sigma} \frac{2}{\sqrt{2\pi}\sigma}\exp(-t^2/2)\cdot\sigma^2 t^2\cdot\sigma dt \\
&= \frac{NR+\rho-1}{NR}\cdot\int_0^{Q_\rho/\sigma}\sqrt{\frac{2}{\pi}}e^{-t^2/2}\cdot t^2 dt,
\end{aligned}$$

where Q_ρ is the $\rho+NR-1$ percentile of $f_a(|e|)$.

By the symmetry of the noise distribution and the relationship between $f(e)$ and $f_a(|e|)$, we could deduce that Q_ρ, the $\rho+NR-1$ percentile of $f_a(|e|)$, is equal to the $(\rho+NR)/2$ percentile of the distribution $f(e) = \frac{1}{\sqrt{2\pi}\sigma}\exp(-\frac{e^2}{2\sigma^2})$. Then Q_ρ/σ is also the $(\rho+NR)/2$ percentile of the distribution $f(e) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{e^2}{2})$.

It is clear that any variable in $T(\rho)$ is irrelative to σ, so we get the desired result. The result for the symmetric Laplace distribution can be obtained in a similar way.

(b) The PDF of a symmetric uniform distribution $f(e) = \frac{1}{2a_0}, (-a_0 < e < a_0)$ where a_0 is a positive constant. The variance $\sigma^2 = \frac{(2a_0)^2}{12} = \frac{a_0^2}{3}$.

The PDF of the absolute noise distribution $f_a(|e|) = \frac{1}{a_0}, 0 < |e| < a_0$. It is obvious that the $\rho+NR-1$ percentile of $f_a(|e|)$ is $Q_\rho = (\rho+NR-1)\cdot a_0$. Then we have

$$\begin{aligned}
T(\rho) &= \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot U_\rho \\
&= \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot\int_0^{Q_\rho} f_a(|e|)\cdot|e|^2 d|e| \\
&= \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot\int_0^{Q_\rho} \frac{e^2}{a_0} de \\
&= \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot\frac{Q_\rho^3}{3a_0} \\
&= \frac{NR+\rho-1}{NR\cdot\sigma^2}\cdot\frac{(\rho+NR-1)^3\cdot a_0^3}{3a_0} \\
&= \frac{(NR+\rho-1)^4}{NR}.
\end{aligned}$$

It is clear that $T(\rho)$ is irrelative to σ, so we get the desired result. ■

A.4 Proof of Property 2

Proof (1) It is obvious that $\epsilon(D) < \epsilon(D_F)$ since $\epsilon(D)$ decreases with n. By the definition of $\mathcal{B}_T(\rho)$ in (11), we have

$$\frac{\partial \mathcal{B}_T(\rho)}{\partial C} = \frac{\epsilon(D)}{\epsilon(D_F)} - 1 < 0$$

for any $\rho < 1$.

(2) By Equations (6) and (3), we have $\frac{\partial \epsilon(D_F)}{\partial \rho} < 0$. Considering that only $\epsilon(D_F)$ is related to ρ in the definition of $\mathcal{B}_T(\rho)$, we get

$$\frac{\partial \mathcal{B}_T(\rho)}{\partial \rho} = -\frac{\epsilon(D)}{\epsilon(D_F)^2}(1+C) \cdot \frac{\partial \epsilon(D_F)}{\partial \rho} > 0.$$

(3) From the definition $\mathcal{B}_T(\rho) = \frac{\epsilon(D)}{\epsilon(D_F)}(1+C) - C$, we have

$$\begin{aligned}
\frac{\partial \mathcal{B}_T(\rho)}{\partial n} &= (1+C) \cdot \frac{\partial}{\partial n}\left(\frac{\epsilon(D)}{\epsilon(D_F)}\right) \\
&= (1+C) \cdot \frac{1}{\epsilon(D_F)^2} \cdot \left[\frac{\partial \epsilon(D)}{\partial n} \cdot \epsilon(D_F) - \epsilon(D) \cdot \frac{\partial \epsilon(D_F)}{\partial n}\right] \\
&= \frac{(1+C)}{\epsilon(D_F)^2} \cdot \left[\frac{\partial \epsilon(D)}{\partial n} \cdot \epsilon(D_F) - \frac{\partial \epsilon(D_F)}{\partial n} \cdot \epsilon(D)\right].
\end{aligned}$$

Let $V(n) = (1 - 1/\sqrt{n})^{-1}$. Then we have

$$\frac{\partial V(n)}{\partial n} = -\frac{1}{2\sqrt{n}(\sqrt{n}-1)^2}, \ \frac{\partial^2 V(n)}{\partial n^2} = \frac{3n+1-4\sqrt{n}}{2n(\sqrt{n}-1)^2}.$$

So $\frac{\partial V(n)}{\partial n} < 0$ and $\frac{\partial^2 V(n)}{\partial n^2} > 0$ hold for $n > 1$.

Compared with (6), $V(n)$ can be seen as a simplification of the $\epsilon(\cdot)$ function. In the fraction of $\epsilon(\cdot)$, the denominator term n plays a more important role than the numerator term $[h(\ln \frac{n}{h} + 1) - \ln \eta]$, especially for a large n. Thus $\epsilon(\cdot)$ has the property similar to $V(n)$ when n is large enough, i.e., $\frac{\partial \epsilon(D)}{\partial n} < 0$ and $\frac{\partial^2 \epsilon(D)}{\partial n^2} > 0$ hold for $n \gg h$.

Since $\epsilon(D_F) = \epsilon(h, n\rho, \eta), \epsilon(D) = \epsilon(h, n, \eta)$ and $n\rho < n$, we have $\epsilon(D_F) > \epsilon(D)$ and $\frac{\partial \epsilon(D)}{\partial n} > \frac{\partial \epsilon(D_F)}{\partial n}$ when n is large enough. Then we have $\left[\frac{\partial \epsilon(D)}{\partial n} \cdot \epsilon(D_F) - \frac{\partial \epsilon(D_F)}{\partial n} \cdot \epsilon(D)\right] > 0$. Therefore, $\frac{\partial \mathcal{B}_T(\rho)}{\partial n} > 0$ holds for $n \gg h$.

(4)

$$\frac{\epsilon(D)}{\epsilon(D_F)^2} > 0, \frac{\partial \epsilon(D_F)}{\partial \rho} < 0 \Rightarrow \frac{\partial^2 \mathcal{B}_T(\rho)}{\partial \rho \partial C} = \frac{\partial^2 \mathcal{B}_T(\rho)}{\partial C \partial \rho} = -\frac{\epsilon(D)}{\epsilon(D_F)^2} \cdot \frac{\partial \epsilon(D_F)}{\partial \rho} > 0.$$

■

A.5 Proof of Property 4

Proof It is clear that $\mathbb{E}(e^2) = \sigma^2$ holds for any symmetric noise distribution, where σ^2 denotes the variance. Then

$$C = \frac{\mathbb{E}_D(r_i^2)}{\mathbb{E}_D(e_i^2)} = \frac{\mathbb{E}_D(r_i^2)}{(1-NR)\cdot 0 + NR\cdot \mathbb{E}(e^2)} = \frac{\mathbb{E}_D(r_i^2)}{NR\cdot \sigma^2},$$

where NR denotes the noise ratio.

For a fixed goodness of fit, we have $\frac{dC}{d(\sigma^2)} < 0$.

From (9), (11) and (19), we know that only the component $\mathcal{B}_T(\rho)$ in $\mathcal{F}(\rho)$ is affected by the noise variance σ^2 when the noise is from a usual symmetric distribution. Then we have

$$\frac{\partial \mathcal{F}(\rho)}{\partial(\sigma^2)} = \epsilon(D_F)\cdot \frac{\partial \mathcal{B}_T(\rho)}{\partial(\sigma^2)}.$$

Moreover,

$$\begin{aligned}\frac{\partial^2 \mathcal{F}(\rho)}{\partial\rho\partial(\sigma^2)} &= \frac{\partial^2 \mathcal{F}(\rho)}{\partial(\sigma^2)\partial\rho}\\ &= \frac{\partial \epsilon(D_F)}{\partial\rho}\cdot\frac{\partial \mathcal{B}_T(\rho)}{\partial(\sigma^2)} + \epsilon(D_F)\cdot\frac{\partial^2 \mathcal{B}_T(\rho)}{\partial(\sigma^2)\partial\rho}\\ &= \frac{\partial \epsilon(D_F)}{\partial\rho}\cdot\frac{\partial \mathcal{B}_T(\rho)}{\partial C}\cdot\frac{dC}{d(\sigma^2)} + \epsilon(D_F)\cdot\frac{\partial^2 \mathcal{B}_T(\rho)}{\partial\rho\partial C}\cdot\frac{dC}{d(\sigma^2)}.\end{aligned}$$

From (25), (20) and (23), we have

$$\frac{\partial^2 \mathcal{F}(\rho)}{\partial\rho\partial(\sigma^2)} < 0.$$

It means that $\frac{\partial \mathcal{F}(\rho)}{\partial\rho}$ decreases with the variance σ^2.

From (16), (21) and (25), the objective function $\mathcal{F}(\rho)$ is derivable about ρ. According to $\rho^* = \arg\max \mathcal{F}(\rho)$, we know that $\frac{\partial \mathcal{F}(\rho)}{\partial\rho}\Big|_{\rho=\rho^*} = 0$, $\frac{\partial \mathcal{F}(\rho)}{\partial\rho}\Big|_{\rho<\rho^*} > 0$, $\frac{\partial \mathcal{F}(\rho)}{\partial\rho}\Big|_{\rho>\rho^*} < 0$.

Let ρ_0^* be the initial optimal relative size. When the variance σ^2 becomes larger, $\frac{\partial \mathcal{F}(\rho)}{\partial\rho}$ is reduced for any $\rho < 1$. Then the new ρ^* will appear in the interval where $\frac{\partial \mathcal{F}(\rho)}{\partial\rho} > 0$ holds for the initial variance, i.e. in the interval $(0, \rho_0^*)$. It means that ρ^* decreases with the noise variance and the desired result is obtained. ■

A.6 Proof of Proposition 1

Proof Let $\delta_1 = y_i^0 - u > 0$, $\delta_2 = v - y_i^0 > 0$. Without loss of generality, suppose $\delta_1 \le \delta_2$. By the definition of the cumulative distribution function (CumDF) and the symmetry of the error distribution,

$$\begin{aligned}\mathbb{P}\{y_i^0 + e \in [u, v]\} &= \mathbb{P}\{e \in [u - y_i^0, v - y_i^0]\}\\ &= \mathbb{P}\{-\delta_1 \le e \le \delta_2\}\\ &= F(\delta_2|\mu = 0, \sigma) - F(-\delta_1|\mu = 0, \sigma)\\ &= F(\delta_2|\mu = 0, \sigma) + F(\delta_1|\mu = 0, \sigma) - 1.\end{aligned}$$

As $\frac{\partial F(e|\mu=0,\sigma)}{\partial \sigma} < 0$ for $e > 0$ and $\sigma_1^2 < \sigma_2^2$, $F(e|\mu = 0, \sigma_1) > F(e|\mu = 0, \sigma_2)$ holds when $e > 0$. Considering that both δ_1 and δ_2 are positive, $F(\delta_1|\mu = 0, \sigma_1) > F(\delta_1|\mu = 0, \sigma_2)$, $F(\delta_2|\mu = 0, \sigma_1) > F(\delta_2|\mu = 0, \sigma_2)$. Therefore,

$$\begin{aligned}\mathbb{P}\{y_i^0 + e^{(1)} \in [u, v]\} &= F(\delta_2|\mu = 0, \sigma_1) + F(\delta_1|\mu = 0, \sigma_1) - 1 \\ &> F(\delta_2|\mu = 0, \sigma_2) + F(\delta_1|\mu = 0, \sigma_2) - 1 \\ &= \mathbb{P}\{y_i^0 + e^{(2)} \in [u, v]\}.\end{aligned}$$

■

A.7 Proof of Corollary 1

A.7.1 Gaussian distribution

Proof The CumDF of a symmetric Gaussian distribution with mean zero and variance σ^2

$$\begin{aligned}F(e|\mu = 0, \sigma) &= \int_{-\infty}^{e} \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{x^2}{2\sigma^2})dx \\ &\overset{t=x/(\sqrt{2}\sigma)}{=\!=\!=} \int_{-\infty}^{\frac{e}{\sqrt{2}\sigma}} \frac{1}{\sqrt{\pi}} \exp(-t^2)dt \\ &= \int_{-\infty}^{0} \frac{1}{\sqrt{\pi}} \exp(-t^2)dt + \int_{0}^{\frac{e}{\sqrt{2}\sigma}} \frac{1}{\sqrt{\pi}} \exp(-t^2)dt \\ &= \frac{1}{2} + \frac{1}{\sqrt{\pi}} \int_{0}^{\frac{e}{\sqrt{2}\sigma}} \exp(-t^2)dt \\ &= \frac{1}{2} + \frac{1}{2} \cdot \{\frac{2}{\sqrt{\pi}} \int_{0}^{\frac{e}{\sqrt{2}\sigma}} \exp(-t^2)dt\} \\ &= \frac{1}{2} + \frac{1}{2} \cdot erf(\frac{e}{\sqrt{2}\sigma}),\end{aligned}$$

where $\exp(\cdot)$ denotes exponential function, and error function $erf(x) = \frac{1}{\sqrt{\pi}} \int_{-x}^{x} \exp(-t^2)dt = \frac{2}{\sqrt{\pi}} \int_{0}^{x} \exp(-t^2)dt$.

Let $s = \frac{e}{\sqrt{2}\sigma}$, $F(e|\mu = 0, \sigma) = \frac{1}{2}[1 + erf(s)]$. It is known that $\frac{d}{dx} erf(x) = \frac{2}{\sqrt{\pi}} \exp(-x^2)$. Then we have

$$\begin{aligned}\frac{\partial F(e|\mu = 0, \sigma)}{\partial \sigma} &= \frac{\partial F(e|\mu = 0, \sigma)}{\partial s} \cdot \frac{\partial s}{\partial \sigma} \\ &= \frac{1}{2} \cdot \frac{2}{\sqrt{\pi}} \exp(-s^2) \cdot (-\frac{e}{\sqrt{2}\sigma^2}) \\ &= -\frac{e}{\sqrt{2\pi}\sigma^2} \exp(-s^2).\end{aligned}$$

When $e > 0$,

$$\frac{\partial F(e|\mu=0,\sigma)}{\partial \sigma} = -\frac{e}{\sqrt{2\pi}\sigma^2}\exp(-s^2) < 0.$$

It gives the desired result by applying Proposition 1. ■

A.7.2 Uniform distribution

Proof Assume that the probability density function (PDF) of a symmetric uniform distribution is

$$f(x) = \begin{cases} \frac{1}{b_0-a_0} & \text{if } a_0 < x < b_0 \\ 0 & \text{otherwise} \end{cases}$$

where a_0, b_0 are constants. The mean $\frac{a_0+b_0}{2} = \mu = 0$, so $b_0 = -a_0 > 0$. The variance $\mathbb{V}(x) = \frac{(b_0-a_0)^2}{12} = \sigma^2$, so $\sigma = \frac{b_0-a_0}{\sqrt{12}}$.

The CumDF of the uniform distribution $F(e|\mu=0,\sigma) = \frac{e-a_0}{b_0-a_0} = \frac{e-a_0}{\sqrt{12}\sigma}$. Then we have

$$\frac{\partial F(e|\mu=0,\sigma)}{\partial \sigma} = -\frac{e+b_0}{\sqrt{12}\sigma^2}.$$

When $0 < e < b_0$, $\frac{\partial F(e|\mu=0,\sigma)}{\partial \sigma} < 0$. It gives the desired result by applying Proposition 1. ■

A.7.3 Laplace distribution

Proof Assume that the PDF of a symmetric Laplace distribution is

$$f(x|\mu=0,\sigma) = \frac{1}{2b_1}\exp(-\frac{|x|}{b_1})$$

where b_1 is a constant. The variance $\mathbb{V}(x) = 2b_1^2 = \sigma^2$, so $\sigma = \sqrt{2}b_1$.

When $e > 0$, $F(e|\mu=0,\sigma) = 1 - \frac{1}{2}\exp(-\frac{e}{b_1}) = 1 - \frac{1}{2}\exp(-\frac{\sqrt{2}e}{\sigma})$. Then

$$\frac{\partial F(e|\mu=0,\sigma)}{\partial \sigma} = -\frac{1}{2}\exp(-\frac{\sqrt{2}e}{\sigma})\cdot(\frac{\sqrt{2}e}{\sigma^2}) < 0.$$

It gives the desired result by applying Proposition 1. ■

A.8 Proof of Proposition 2

Proof Let $\delta_1 = y_i^0 - u > 0$, $\delta_2 = v - y_i^0 > 0$. Without loss of generality, suppose $\delta_1 \le \delta_2$. Let $e_A \in \{e_i | y_i^0 + e_i \in [u,v]\}, e_B \in \{e_i | y_i^0 + e_i \notin [u,v]\}$.

The PDF of e_A

$$f_A(e) = \begin{cases} \frac{f(e)}{\int_{u-y_i^0}^{v-y_i^0} f(e)de} & \text{if } u \le y_i^0 + e \le v \\ 0 & \text{otherwise} \end{cases} = \begin{cases} \frac{f(e)}{\int_{-\delta_1}^{\delta_2} f(t)dt} & \text{if } -\delta_1 \le e \le \delta_2 \\ 0 & \text{otherwise} \end{cases}.$$

For any $p \in \mathbb{N}^+$,

$$\begin{aligned} \mathbb{E}(|e_A|^p) &= \int_{-\infty}^{+\infty} |e_A|^p \cdot f_A(e)de \\ &= \int_{-\delta_1}^{\delta_2} |e_A|^p \cdot \frac{f(e)}{\int_{-\delta_1}^{\delta_2} f(t)dt}de \\ &= \frac{\int_{-\delta_1}^{\delta_2} |e_A|^p \cdot f(e)de}{2\int_0^{\delta_1} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \\ &= \frac{2\int_0^{\delta_1} |e_A|^p \cdot f(e)de + \int_{\delta_1}^{\delta_2} |e_A|^p \cdot f(e)de}{2\int_0^{\delta_1} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \\ &= \frac{2\int_0^{\delta_1} |e_A|^p \cdot f(e)de}{2\int_0^{\delta_1} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} + \frac{\int_{\delta_1}^{\delta_2} |e_A|^p \cdot f(e)de}{2\int_0^{\delta_1} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \\ &= \frac{2\int_0^{\delta_1} f(e)de}{2\int_0^{\delta_1} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \cdot \frac{2\int_0^{\delta_1} |e_A|^p \cdot f(e)de}{2\int_0^{\delta_1} f(e)de} \\ &\quad + \frac{\int_{\delta_1}^{\delta_2} f(e)de}{2\int_0^{\delta_1} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \cdot \frac{\int_{\delta_1}^{\delta_2} |e_A|^p \cdot f(e)de}{\int_{\delta_1}^{\delta_2} f(e)de}. \end{aligned}$$

Let $a = \int_0^{\delta_1} f(t)dt$, $b = \int_{\delta_1}^{\delta_2} f(t)dt$, $c = \int_{\delta_2}^{+\infty} f(t)dt$, and $H = \frac{\int_{\delta_1}^{\delta_2} |e_A|^p \cdot f(t)dt}{\int_{\delta_1}^{\delta_2} f(t)dt} = \frac{\int_{-\delta_2}^{-\delta_1} |e_B|^p \cdot f(t)dt}{\int_{\delta_1}^{\delta_2} f(t)dt} = \frac{\int_{\delta_1}^{\delta_2} |e|^p \cdot f(t)dt}{\int_{\delta_1}^{\delta_2} f(t)dt}$.

It is obvious that $H \ge \frac{\int_{\delta_1}^{\delta_2} \delta_1^p \cdot f(t)dt}{\int_{\delta_1}^{\delta_2} f(t)dt} = \delta_1^p$ and $H \le \frac{\int_{\delta_1}^{\delta_2} \delta_2^p \cdot f(t)dt}{\int_{\delta_1}^{\delta_2} f(t)dt} = \delta_2^p$. Then we have

$$\begin{aligned} \mathbb{E}(|e_A|^p) &= \frac{2a}{2a+b} \cdot \frac{2\int_0^{\delta_1} |e_A|^p \cdot f(e)de}{2\int_0^{\delta_1} f(e)de} + \frac{b}{2a+b} \cdot \frac{\int_{\delta_1}^{\delta_2} |e_A|^p \cdot f(e)de}{\int_{\delta_1}^{\delta_2} f(e)de} \\ &\le \frac{2a}{2a+b} \cdot \frac{2\int_0^{\delta_1} \delta_1^p \cdot f(e)de}{2\int_0^{\delta_1} f(e)de} + \frac{b}{2a+b} \cdot \frac{\int_{\delta_1}^{\delta_2} |e_A|^p \cdot f(e)de}{\int_{\delta_1}^{\delta_2} f(e)de} \end{aligned}$$

$$
\begin{aligned}
&= \frac{2a}{2a+b} \cdot \delta_1^p + \frac{b}{2a+b} \cdot H \\
&\leq \frac{2a}{2a+b} \cdot H + \frac{b}{2a+b} \cdot H \\
&= H,
\end{aligned}
$$

with equality if and only if $|e_A| \equiv \delta_1$. According to the symmetry of the noise distribution, $|e_A| \equiv \delta_1$ means $\mathbb{P}\{e_A = -\delta_1\} = \mathbb{P}\{e_A = \delta_1\} = 1/2$.

The PDF of e_B

$$
\begin{aligned}
f_B(e) &= \begin{cases} \frac{f(e)}{\int_{-\infty}^{u-y_i^0} f(e)de + \int_{v-y_i^0}^{+\infty} f(e)de} & \text{if } y_i^0 + e \notin [u, v] \\ 0 & \text{otherwise} \end{cases} \\
&= \begin{cases} \frac{f(e)}{2\int_{\delta_2}^{+\infty} f(e)de + \int_{\delta_1}^{\delta_2} f(e)de} & \text{if } e \notin [-\delta_1, \delta_2] \\ 0 & \text{otherwise} \end{cases}.
\end{aligned}
$$

For any $p \in \mathbb{N}^+$,

$$
\begin{aligned}
\mathbb{E}(|e_B|^p) &= \int_{-\infty}^{+\infty} |e_B|^p \cdot f_B(e)de \\
&= \frac{\int_{-\infty}^{-\delta_1} |e_B|^p \cdot f(e)de + \int_{\delta_2}^{+\infty} |e_B|^p \cdot f(e)de}{2\int_{\delta_2}^{+\infty} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \\
&= \frac{2\int_{\delta_2}^{+\infty} |e_B|^p \cdot f(e)de + \int_{-\delta_2}^{-\delta_1} |e_B|^p \cdot f(e)de}{2\int_{\delta_2}^{+\infty} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \\
&= \frac{2\int_{\delta_2}^{+\infty} |e_B|^p \cdot f(e)de + \int_{\delta_1}^{\delta_2} |e|^p \cdot f(e)de}{2\int_{\delta_2}^{+\infty} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \\
&= \frac{2\int_{\delta_2}^{+\infty} |e_B|^p \cdot f(e)de}{2\int_{\delta_2}^{+\infty} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} + \frac{\int_{\delta_1}^{\delta_2} |e|^p \cdot f(e)de}{2\int_{\delta_2}^{+\infty} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \\
&= \frac{2\int_{\delta_2}^{+\infty} f(t)dt}{2\int_{\delta_2}^{+\infty} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \cdot \frac{2\int_{\delta_2}^{+\infty} |e_B|^p \cdot f(e)de}{2\int_{\delta_2}^{+\infty} f(t)dt} \\
&\quad + \frac{\int_{\delta_1}^{\delta_2} f(t)dt}{2\int_{\delta_2}^{+\infty} f(t)dt + \int_{\delta_1}^{\delta_2} f(t)dt} \cdot \frac{\int_{\delta_1}^{\delta_2} |e|^p \cdot f(e)de}{\int_{\delta_1}^{\delta_2} f(t)dt}.
\end{aligned}
$$

By $H \leq \delta_2^p$, we have

$$\begin{aligned}
\mathbb{E}(|e_B|^p) &= \frac{2c}{2c+b} \cdot \frac{2\int_{\delta_2}^{+\infty} |e_B|^p \cdot f(e)de}{2\int_{\delta_2}^{+\infty} f(t)dt} + \frac{b}{2c+b} \cdot \frac{\int_{\delta_1}^{\delta_2} |e|^p \cdot f(e)de}{\int_{\delta_1}^{\delta_2} f(t)dt} \\
&\geq \frac{2c}{2c+b} \cdot \frac{\int_{\delta_2}^{+\infty} \delta_2^p \cdot f(e)de}{\int_{\delta_2}^{+\infty} f(t)dt} + \frac{b}{2c+b} \cdot H \\
&= \frac{2c}{2c+b} \cdot \delta_2^p + \frac{b}{2c+b} \cdot H \\
&\geq \frac{2c}{2c+b} \cdot H + \frac{b}{2c+b} \cdot H \\
&= H,
\end{aligned}$$

with equality if and only if $|e_B| \equiv \delta_2$. Note that $e_B \notin [-\delta_1, \delta_2]$ and $\delta_1 \leq \delta_2$, $\mathbb{P}\{e_B = -\delta_2\} = 1$. By the symmetry of noise distribution, $|e_B| \equiv \delta_2$ means $\mathbb{P}\{e_A = \delta_2\} > 0$.

Hence

$$\mathbb{E}(|e_A|^p) \leq \mathbb{E}(|e_B|^p)$$

holds with equality if and only if $|e_A| \equiv \delta_1$ and $|e_B| \equiv \delta_2$. The two conditions imply $\delta_1 = \delta_2$. Then $e_B \in \{e_i | e_i \notin [-\delta_1, \delta_2]\} = \varnothing$. And thus $\mathbb{E}(|e_i|^p | y_i^0 + e_i \in [u, v]) < \mathbb{E}(|e_i|^p | y_i^0 + e_i \notin [u, v])$ holds for any $p \in \mathbb{N}^+$. ■

A.9 Proof of Property 5

Proof Let $f_\theta(\theta)$ be the probability density function (PDF) of variable $\theta = y_i^0 - c$, where c is the center of the covering interval. Then $f_c(c) = f_\theta(\theta) = f_\theta(y_i^0 - c)$. Substituting c with $y_i^0 - c$ gives us $f_c(y_i^0 - c) = f_\theta(c)$. Substituting c with $y_i^0 + c$ gives us $f_c(y_i^0 + c) = f_\theta(-c)$. By the symmetry of $f_c(c)$, i.e.,$f_c(y_i^0 - c) = f_c(y_i^0 + c)$, we obtain $f_\theta(c) = f_\theta(-c)$. In other words, θ is from a symmetric distribution. So we have $\int_{-r}^{r} \theta \cdot f_\theta(\theta)d\theta = 0$ for any interval radius r.

When $y_i \notin [u, v]$, $R_i = |y_i - c| = |y_i - y_i^0 + \theta| = |e_i + \theta|$. Then

$$\mathbb{E}_c(R_i) = \int_{y_i^0 - r}^{y_i^0 + r} |y_i - c| \cdot f_c(c)dc = \int_{-r}^{r} |e_i + \theta| \cdot f_\theta(\theta)d\theta \triangleq \mathbb{E}_\theta(R_i).$$

Considering that $y_i \notin [u, v] \Rightarrow |e_i| > r > 0$, the unbiasedness is analyzed in the following two cases.

If $e_i > r > 0$, $R_i = |e_i + \theta| = e_i + \theta$. Then

$$\begin{aligned}
\mathbb{E}_\theta(R_i) &= \int_{-r}^{r} (e_i + \theta) \cdot f_\theta(\theta)d\theta \\
&= \int_{-r}^{r} e_i \cdot f_\theta(\theta)d\theta + \int_{-r}^{r} \theta \cdot f_\theta(\theta)d\theta \\
&= e_i \cdot \int_{-r}^{r} f_\theta(\theta)d\theta + \int_{-r}^{r} \theta \cdot f_\theta(\theta)d\theta \\
&= e_i \cdot 1 + 0 = |e_i|.
\end{aligned}$$

If $e_i < -r < 0$, $R_i = |e_i + \theta| = -(e_i + \theta)$. Then

$$\begin{aligned}
\mathbb{E}_\theta(R_i) &= -\int_{-r}^{r} (e_i + \theta) \cdot f_\theta(\theta) d\theta \\
&= -\int_{-r}^{r} e_i \cdot f_\theta(\theta) d\theta - \int_{-r}^{r} \theta \cdot f_\theta(\theta) d\theta \\
&= -e_i \cdot \int_{-r}^{r} f_\theta(\theta) d\theta - \int_{-r}^{r} \theta \cdot f_\theta(\theta) d\theta \\
&= -e_i \cdot 1 - 0 \\
&= -e_i = |e_i|.
\end{aligned}$$

Therefore, $\mathbb{E}_c(R_i | y_i \notin [u, v]) = \mathbb{E}_\theta(R_i | y_i \notin [u, v]) = |e_i|$. ■

A.10 Proof of Property 6

Proof (1) From Figure 6, we have

$$\big|R_i - |e_i|\big| = |c - y_i^0|,$$

$$\big|\inf|e_i| - |e_i|\big| = \begin{cases} y_i^0 - u & \text{if } y_i < u \\ v - y_i^0 & \text{if } y_i > v \end{cases},$$

$$\big|\sup|e_i| - |e_i|\big| = \begin{cases} v - y_i^0 & \text{if } y_i < u \\ y_i^0 - u & \text{if } y_i > v \end{cases},$$

By $y_i^0 \in [u, v]$ and $c = (u + v)/2$, we have $|c - y_i^0| \le r$. Then

$$EAD_{CD} = \mathbb{E}_c\big|R_i - |e_i|\big| = \int_{-\infty}^{+\infty} |c - y_i^0| \cdot f_c(c) dc \le \int_{-\infty}^{+\infty} r \cdot f_c(c) dc = r,$$

with equality if and only if $|c - y_i^0| \equiv r$. Considering that $|c - y_i^0| \equiv r$ does not hold in reality, we have $EAD_{CD} < r$ for any $y_i \notin [u, v]$.

When $y_i < u$,

$$\begin{aligned}
EAD_L &= \mathbb{E}_c\big|\inf|e_i| - |e_i|\big| \\
&= \int_{-\infty}^{+\infty} (v - y_i^0) \cdot f_c(c) dc \\
&= \int_{-\infty}^{+\infty} (c + r - y_i^0) \cdot f_c(c) dc \\
&= \int_{-\infty}^{+\infty} (c - y_i^0) \cdot f_c(c) dc + r \cdot \int_{-\infty}^{+\infty} f_c(c) dc.
\end{aligned}$$

By the symmetry of $f_c(c)$, we have $\int_{-\infty}^{+\infty} (c - y_i^0) \cdot f_c(c) dc = 0$, then $EAD_L = r$.

Similarly, $EAD_L = r$ holds for $y_i > v$. Thus $EAD_L \equiv r$ holds for any $y_i \notin [u, v]$. It can be proved in the same way for $EAD_U \equiv r$.

(2) Since the absolute noise $|e_i|$ is independent of the interval center c, we have $ERD_{CD} = \frac{EAD_{CD}}{|e_i|}, ERD_L = \frac{EAD_L}{|e_i|}, ERD_U = \frac{EAD_U}{|e_i|}$. Then $EAD_{CD} < EAD_L = EAD_U \Rightarrow ERD_{CD} < ERD_L = ERD_U$ for any $y_i \notin [u, v]$.

(3) Note that $u < y_i^0 < v \Leftrightarrow y_i^0 - r < c < y_i^0 + r$, we have

$$EAD_{CD} = \int_{y_i^0 - r}^{y_i^0 + r} |c - y_i^0| \cdot f_c(c) dc \overset{t=c-y_i^0}{=\!=\!=} \int_{-r}^{r} |t| \cdot f_c(t + y_i^0) dt.$$

By the symmetry of $f_c(c)$,

$$\frac{\partial EAD_{CD}}{\partial r} = r \cdot f_c(r + y_i^0) + r \cdot f_c(-r + y_i^0) = 2r f_c(r + y_i^0) > 0.$$

■

References

A. Arnaiz-González, J. F. Díez-Pastor, J. J. Rodríguez, and C. I. García-Osorio. Instance selection for regression by discretization. *Expert Systems with Applications*, 54:340–350, 2016.

R. Barandela and E. Gasca. Decontamination of training samples for supervised pattern recognition methods. In *Joint International Workshops on Advances in Pattern Recognition (IAPR)*, pages 621–630, 2000.

P. L. Bartlett and S. Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2007.

C. E. Brodley and M. A. Friedl. Identifying mislabeled training data. *Journal of Artificial Intelligence Research*, 11(1):131–167, 1999.

J. Cao, S. Kwong, and R. Wang. A noise-detection based AdaBoost algorithm for mislabeled data. *Pattern Recognition*, 45(12):4451–4465, 2012.

C. C. Chang and C. J. Lin. LIBSVM data: Classification, regression, and multi-label. *http://www.csie.ntu.edu.tw/˜cjlin/libsvmtools/datasets/*, 2016.

V. Cherkassky, X. Shao, F. M. Mulier, and V. Vapnik. Model complexity control for regression using VC generalization bounds. *IEEE Transactions on Neural Networks*, 10 (5):1075–1089, 1999.

D. Dua and C. Graff. UCI machine learning repository. university of california, Irvine, School of information and computer science. *http://archive.ics.uci.edu/ml*, 2017.

S. Escalera, J. Fabian, P. Pardo, X. Baro, and I. Guyon. ChaLearn looking at people 2015: Apparent age and cultural event recognition datasets and results. In *IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2015.

B. Frenay and M. Verleysen. Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 25(5):845–869, 2014.

D. Gamberger, N. Lavrač, and C. Grošelj. Experiments with noise filtering in a medical domain. In *International Conference on Machine Learning (ICML)*, pages 143–151, 1999.

L. P. F. Garcia, J. A. Sáez, J. Luengo, A. C. Lorena, A. F. de Carvalho, and F. Herrera. Using the one-vs-one decomposition to improve the performance of class noise filters via an aggregation strategy in multi-class classification problems. *Knowledge-Based Systems*, 90:153–164, 2015.

L. P. F. Garcia, A. C. Lorena, S. Matwin, and A. de Carvalho. Ensembles of label noise filters: a ranking approach. *Data Mining and Knowledge Discovery*, 30(5):1192–1216, 2016.

S. García, J. Derrac, and F. Herrera. Prototype selection for nearest neighbor classification: taxonomy and empirical study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(3):417–435, 2012.

R. Gerlach and J. Stamey. Bayesian model selection for logistic regression with misclassified outcomes. *Statistical Modelling*, 7(3):255–273, 2007.

A. Guillen, L. J. Herrera, G. Rubio, H. Pomares, A. Lendasse, and I. Rojas. New method for instance or prototype selection using mutual information in time series prediction. *Neurocomputing*, 73(10):2030–2038, 2010.

P. A. Gutierrez, M. Perez-Ortiz, J. Sanchez-Monedero, F. Fernandez-Navarro, and C. Hervas-Martinez. Ordinal regression methods: Survey and experimental study. *IEEE Transactions on Knowledge and Data Engineering*, 28(1):127–146, 2016.

B. Han, I. Tsang, L. Chen, J. Zhou, and C. Yu. Beyond majority voting: A coarse-to-fine label filtration for heavily noisy labels. *IEEE Transactions on Neural Networks and Learning Systems*, 30(12):3774–3787, 2019.

Z. Huang, W. Zhou, and H. Li. Cascaded deep convolutional neural network for robust face alignment. In *IEEE International Conference on Image Processing (ICIP)*, pages 1218–1222, 2018.

Z. Huo, X. Yang, C. Xing, Y. Zhou, P. Hou, J. Lv, and X. Geng. Deep age distribution learning for apparent age estimation. In *IEEE conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 722–729, 2016.

T. Khoshgoftaar and P. Rebours. Improving software quality prediction by noise filtering techniques. *Journal of Computer Science and Technology*, 22(3):387–396, 2007.

M. Kordos and M. Blachnik. Instance selection with neural networks for regression problems. In *International Conference on Artificial Neural Networks*, pages 263–270, 2012.

M. Kordos, S. Bialka, and M. Blachnik. Instance selection in logical rule extraction for regression problems. In *International Conference on Artificial Intelligence and Soft Computing*, pages 167–175, 2013.

K. Lee, X. He, L. Zhang, and L. Yang. CleanNet: Transfer learning for scalable image classifier training with label noise. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5447–5456, 2018.

C. Li, V. S. Sheng, L. Jiang, and H. Li. Noise filtering to improve data and model quality for crowdsourcing. *Knowledge-Based Systems*, 107:96–103, 2016.

S. Li, H. Wang, and M. U. Rafique. A novel recurrent neural network for manipulator control with improved noise tolerance. *IEEE Transactions on Neural Networks and Learning Systems*, 29(5):1908–1918, 2018.

H. Liu and S. Zhang. Noisy data elimination using mutual-nearest neighbor for classification mining. *Journal of Systems and Software*, 85(5):1067–1074, 2012.

X. Liu, S. Li, M. Kan, J. Zhang, S. Wu, W. Liu, H. Han, S. Shan, and X. Chen. AgeNet: Deeply learned regressor and classiffier for robust apparent age estimation. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 16–24, 2015.

D. F. Nettleton, A. Orriolspuig, and A. Fornells. A study of the effect of different types of noise on the precision of supervised learning techniques. *Artificial Intelligence Review*, 33(4):275–306, 2010.

L. Oneto, A. Ghio, S. Ridella, and D. Anguita. Local Rademacher complexity: Sharper risk bounds with and without unlabeled samples. *Neural Networks*, 65:115–125, 2015.

G. Patrini, A. Rozza, A. K. Menon, R. Nock, and L. Qu. Making deep neural networks robust to label noise: A loss correction approach. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2233–2241, 2017.

G. Peters. Rough clustering utilizing the principle of indifference. *Information Sciences*, 277(2):358–374, 2014.

P. Rigollet. Generalization error bounds in semi-supervised classification under the cluster assumption. *Journal of Machine Learning Research*, 8:1369–1392, 2007.

R. Rothe, R. Timofte, and L. Van Gool. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, 126 (2):144–157, 2018.

M. Sabzevari, G. Martínez-Muñoz, and A. Suárez. Vote-boosting ensembles. *Pattern Recognition*, 83:119–133, 2018.

J. A. Sáez, J. Luengo, and F. Herrera. Predicting noise filtering efficacy with data complexity measures for nearest neighbor classification. *Pattern Recognition*, 46(1):355–364, 2013.

J. A. Sáez, M. Galar, J. Luengo, and F. Herrera. INFFC: An iterative class noise filter based on the fusion of classifiers with noise sensitivity control. *Information Fusion*, 27: 19–32, 2016.

J. Sanchez-Monedero, P. A. Gutierrez, and M. Perez-Ortiz. ORCA: A matlab/octave toolbox for ordinal regression. *Journal of Machine Learning Research*, 20(125):1–5, 2019.

N. Segata, E. Blanzieri, S. J. Delany, and P. Cunningham. Noise reduction for instance-based learning with a local maximal margin approach. *Journal of Intelligent Information Systems*, 35(2):301–331, 2010.

A. A. Shanab, T. Khoshgoftaar, and R. Wald. Robustness of threshold-based feature rankers with data sampling on noisy and imbalanced data. In *International Florida Artificial Intelligence Research Society Conference*, pages 1–6, 2012.

J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, and D. Meng. Meta-Weight-Net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1919–1930, 2019.

B. Sluban, D. Gamberger, and N. Lavrač. Advances in class noise detection. In *European Conference on Artificial Intelligence*, pages 1105–1106, 2010.

B. Sluban, D. Gamberger, and N. Lavrač. Ensemble-based noise detection: noise ranking and visual performance evaluation. *Data Mining and Knowledge Discovery*, 28(2):265–303, 2014.

T. Tian and J. Zhu. Max-margin majority voting for learning from crowds. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1621–1629, 2015.

R. Wang, T. Liu, and D. Tao. Multiclass learning with partially corrupted labels. *IEEE Transactions on Neural Networks and Learning System*, 29(6):2568–2580, 2018.

J. Yao, J. Wang, I. W. Tsang, Y. Zhang, J. Sun, C. Zhang, and R. Zhang. Deep learning from noisy image labels with quality embedding. *IEEE Transactions on Image Processing*, 28(4):1909–1922, 2018.

W. Yuan, D. Guan, T. Ma, and A. M. Khattak. Classification with class noises through probabilistic sampling. *Information Fusion*, 41:57–67, 2018.

L. Zhang, T. Yang, and R. Jin. Empirical risk minimization for stochastic convex optimization: O(1/n)- and O(1/n2)-type of risk bounds. In *Annual Conference on Learning Theory (COLT)*, pages 1954–1979, 2017.