Preprint and Prereview Version
Published in: Proceedings of the Annual Conference of the North American Fuzzy Information Processing Society (NAFIPS 2021)

Evaluation Criteria for Noise Resilience in Regression Algorithms

Javier Viaña[1[0000-0002-0563-784X]] and Kelly Cohen[2[0000-0002-8655-1465]]

[1,2] University of Cincinnati, Cincinnati OH 45219, USA
vianajr@mail.uc.edu
cohenky@ucmail.uc.edu

Abstract. Noise resilience is a popular attribute among machine learning algorithms. In regression problems, it refers to the ability to keep high performance even when the data is noisy. Surprisingly, there is no standard figure of merit to quantify it. This theoretical research leverages the variance of the residuals to determine objectively the performance of a regression algorithm in the presence of noisy data. The two main contributions of the paper are the locality conditions of noise resilience and the noise resilience score, NR.

Keywords: Approximation, Fuzzy AI, Noisy Data, Noise Resilience, Local Conditions of Noise Resilience, Global Noise Resilience Score

1 Introduction

The presence of noisy data can lead to overfitting in regression problems. Deciding the best approximation is not a trivial task; ultimately, the algorithm should find the pattern rather than fitting accurately the datapoints. In robotics for example, noise is frequently present due to the vibrations of the mechanical systems. For that matter, the classical control theory is often replaced by more flexible and realistic methods. Fuzzy logic has proven notable performance in this field, which in many cases, [1 - 4], exhibits the property of noise resilience.

Regression problems have several figures of merit to evaluate the accuracy of the predictions, [5 - 9]: Root Mean Squared Error (RMSE), Mean Average Error (MAE) or Mean Absolute Percentage Error (MAPE) for example. Nevertheless, the generalized technique to evaluate noise resilience entails experimental testing. In fact, there is a lack of updated literature to quantify theoretically the noise resilience of a regression algorithm. Which is undoubtedly relevant, as many methods have claimed such property in the past decade: [10 - 16] for instance. In contrast, other branches of AI, such as associative memory and information retrieval problems, already have standard criteria to measure the amount of affordable noise.

2 Research Objectives

This paper provides a criteria to measure theoretically the noise resilience of a regression algorithm. With the growing interest in understanding noisy datasets, it becomes more prominent to obtain a technique that allows for the quantification of an approximation's resilience to noise. Such an evaluation method would provide higher credibility to the techniques in question. Specifically, the realm of soft computing artificial intelligence could be substantially benefitted from it (which encompasses the tolerance of precision, partial truth, and uncertainty).

In section 3.1 the framework of the problem is explained in detail, where the development makes use of the locality hypothesis. This will later serve to find the boundaries that determine if an algorithm is resilient or not, within a given region of the domain. 3.2 contains the theoretical development of the local conditions and boundaries of noise resilience. It should be remarked that the property of noise resilience is often associated to a maximum value of affordable corruption in the data, above which, the algorithm no longer provides a useful prediction. Thus, the mentioned boundaries must depend on the amount of noise present in the data. Finally, section 3.3 provides the definition of the NR metric.

3 Methodology

3.1 Framework

The following development will be particularized for a one input one output case. For simplicity, the noise will be present only in the output variable. Note that if there was noise in the input features those inputs could be considered as true (as long as they belong to the domain of the dimension) and the noise would be transmitted to the output. In order to evaluate the impact of the noise, for each q instance of the dataset (formed by a total of Q points), three important output variables will be studied; $\hat{y}^q$, y^q, and μ^q.

The first variable, $\hat{y}^q$, is the predicted output for a given input x^q. This will be the outcome generated by the algorithm.

The second is the observation, y^q, a real output obtained in the data acquisition process, which has a certain amount of noise. In other words, if the acquisition was carried out again for the same input, x^q, its value will most likely differ from the previous one, ${}^1y^q \neq {}^2y^q$.

The third variable, μ^q, is the unknown ground truth. This can be seen as the average of all the possible output values obtained, for a given input x^q, when the number of tests, K, tends to infinity.

$$\mu^q = \lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^{K} {}^k y^q \tag{1}$$

In a real problem there is no access to this value, unless the noise is intentionally introduced or there exists a tailored theoretical formulation for the system which represents accurately the reality. In the majority of the datasets that is not the case. The goal of this AI task is not only to correctly fit the data but also to infer the value of μ^q as accurately as possible. Note that these are two very different things. Obviously, the system will train using only the information available, x^q and y^q, but it will never be exposed to μ^q (at least during the training). While the algorithm learns, it will gradually minimize the difference between $\hat{y}^q$ and y^q, but $\hat{y}^q$ should never match y^q, unless y^q is exactly μ^q. Here resides one of the current biggest issues in AI, overtraining. If the training time is longer than the optimal, $\hat{y}^q$ can suddenly start converging towards y^q, creating unexpected shapes in the continuum of predictions. In the presence of noise, this is more likely to happen, which relates to the topic of the present research. Ultimately, $\hat{y}^q$ should be an approximation of μ^q, not of y^q.

In a regression problem of continuous variables, if the range of the inputs is infinitesimally small (dx) the curve converges to a linear function. This is a fundamental concept of calculus, which can be seen in Fig. 1 (where a nonlinear function with a single input variable has been considered).

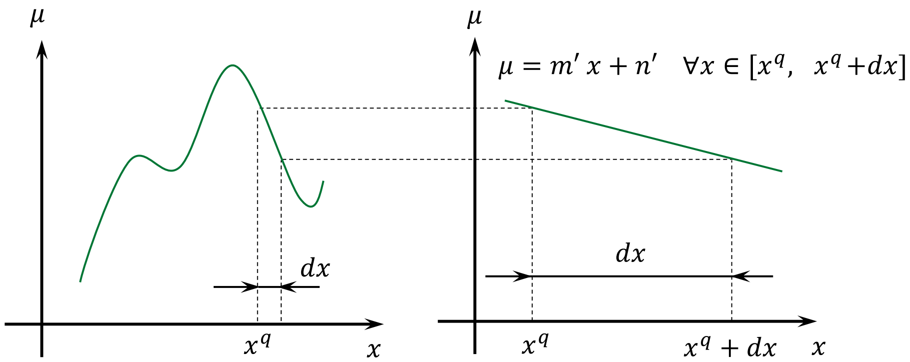

Fig. 1. Representation of the aforementioned fundamental concept of calculus.

This concept will serve as the basis of the hypothesis of locality: the truth (instances with no noise) and the prediction should both follow approximately a linear regression, provided that the range of the input variable considered is sufficiently small (Fig. 2).

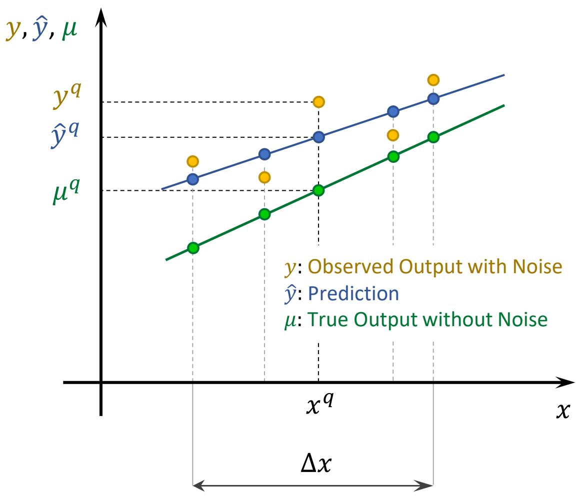

Fig. 2. Observations, predictions, and ground truth.

Thus, the prediction can be modeled as

$$\hat{y}^q \approx m\, x^q + n \tag{2}$$

Where m and n are the slope and the intercept, respectively, of the closest linear regression to the prediction function within Δx. To model the ground truth, the hypothesis of locality (for a narrow Δx range) is also considered. Thus, μ^q can be approximated to

$$\mu^q \approx m'\, x^q + n' \tag{3}$$

Ideally both the prediction and the ground truth should follow the same equation, but in a general case, m' and n' are different from m and n. To account for such difference, the residuals ${\varepsilon_y}^q$, ${\varepsilon_{\hat{y}}}^q$ and λ^q are defined as

$${\varepsilon_y}^q = y^q - \mu^q \tag{4}$$

$${\varepsilon_{\hat{y}}}^q = \hat{y}^q - \mu^q \tag{5}$$

$$\lambda^q = y^q - \hat{y}^q = {\varepsilon_y}^q - {\varepsilon_{\hat{y}}}^q \tag{6}$$

The expression of ${\varepsilon_{\hat{y}}}^q$ can be further developed,

$${\varepsilon_{\hat{y}}}^q = \hat{y}^q - \mu^q = m\, x^q + n_i - m' x^q - n' = (m - m')x^q + (n - n') \tag{7}$$

Which means that it also follows the equation of a line

$${\varepsilon_{\hat{y}}}^q = \alpha\, x^q + \beta \tag{8}$$

The representation of these residuals has been incorporated in Fig. 3.

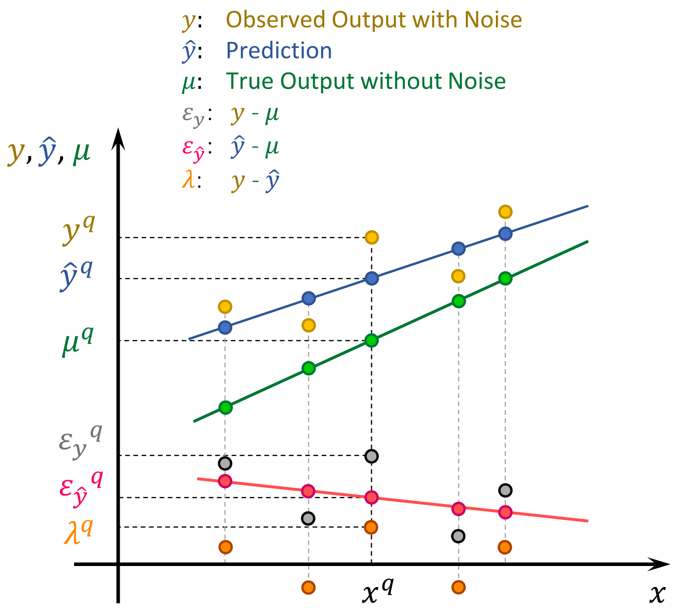

Fig. 3. Observations, predictions, ground truth and residuals.

Where α can be obtained by comparing the first and last points of the range studied,

$$\alpha = m - m' = \frac{\hat{y}^f - \hat{y}^0}{x^f - x^0} - \frac{\mu^f - \mu^0}{x^f - x^0} = \frac{\hat{y}^f - \mu^f + \hat{y}^0 - \mu^0}{x^f - x^0} = \frac{\varepsilon_{\hat{y}}{}^f - \varepsilon_{\hat{y}}{}^0}{\Delta x} \tag{9}$$

Note that there are data points that are closer to the ground truth than others. Those that have more noise, jeopardize the prediction, and those that have less, improve it. The goal is to define under what conditions the prediction is better than the real data when compared to the ground truth.

3.2 Conditions of Noise Resilience

The last statement of the previous section is the bottom line of this development. The question that follows is: under what conditions does a regression algorithm performs better than the noisy data when its outcomes are compared to the ground truth?

To answer this, the mean and the variances of both residuals $\varepsilon_y{}^q$ and $\varepsilon_{\hat{y}}{}^q$ will be compared. Since the prediction has been generated from the same instances, it is expected that the means will be the same, but the variances might not be equal. A favorable case would be when the variance of $\varepsilon_{\hat{y}}{}^q$ is smaller than the variance of $\varepsilon_y{}^q$. That would imply that the slope of the function (8) is small. Obviously, the prediction would have less noise than the data itself. Thus, if there exists any particular instance whose confidence in the output is higher, then it could be used to further improve the prediction. In the event that there are several points with higher output confidence than the average, the compensation, Ƈ, can be calculated as the weighted mean of those. In such case, both the confidence value, γ^q, and the distance to the center of the Δx range, x_c, should define the weights, w^q. For example, the authors suggest the definition (10) for the compensation, where the weights range from 0 to 1. Let N represent the number of datapoints that belong to the range considered, then

$$Ƈ = \sum_{q=1}^{N} w^q \, \lambda^q = \sum_{q=1}^{N} \frac{(\gamma^q - \gamma^{min})}{(\gamma^{max} - \gamma^{min})(1 + \mathrm{abs}(x_c - x^q))} \, \lambda^q \tag{10}$$

To retake the development, the first step is the calculation of the means. The mean value of $\varepsilon_y{}^q$ is

$$\bar{\varepsilon}_y = \frac{1}{N}\sum_{q=1}^{N} \varepsilon_y{}^q = \frac{1}{N}\sum_{q=1}^{N} (y^q - \mu^q) = \frac{1}{N}\sum_{q=1}^{N} y^q - \frac{1}{N}\sum_{q=1}^{N} \mu^q = \bar{y} - \bar{\mu} \tag{11}$$

Similarly, the mean value of $\varepsilon_{\hat{y}}{}^q$

$$\begin{aligned} \bar{\varepsilon}_{\hat{y}} &= \frac{1}{N}\sum_{q=1}^{N} \varepsilon_{\hat{y}}{}^q = \frac{1}{N}\sum_{q=1}^{N} (\hat{y}^q - \mu^q) = \frac{1}{N}\sum_{q=1}^{N} \hat{y}^q - \frac{1}{N}\sum_{q=1}^{N} \mu^q = \cdots \\ &= \frac{1}{N}\sum_{q=1}^{N} (m\, x^q + n) - \bar{\mu} = \bar{\hat{y}} - \bar{\mu} \end{aligned} \tag{12}$$

The mean of the predictions, $\bar{\hat{y}}$, can be obtained considering the input feature x, and its mean $\bar{x}$,

$$\bar{\hat{y}} = m\frac{1}{N}\sum_{q=1}^{N} x^q + n\frac{1}{N}\sum_{q=1}^{N} 1 = m\bar{x} + n \tag{13}$$

(14) holds for a linear regression, and for a generic approximator it should also be a fair representation of the reality.

$$m\bar{x} + n = \bar{y} \tag{14}$$

$$\bar{y} = \bar{\hat{y}} \tag{15}$$

Thus, the final value of the $\bar{\varepsilon}_{\hat{y}}$ residual is

$$\bar{\varepsilon}_{\hat{y}} = \bar{y} - \bar{\mu} \tag{16}$$

From (11) and (16) can be seen that $\bar{\varepsilon}_y$ and $\bar{\varepsilon}_{\hat{y}}$ are equal, as expected.

$$\bar{\varepsilon}_y = \bar{\varepsilon}_{\hat{y}} \tag{17}$$

Now, the calculation of the variances follows, they will be represented by s. Since the preliminary development is the same for both ε_y and $\varepsilon_{\hat{y}}$, the generic symbol ε will be used instead.

$$\begin{aligned} s_\varepsilon &= \frac{1}{N-1}\sum_{q=1}^{N}(\varepsilon^q - \bar{\varepsilon})^2 = \frac{1}{N-1}\sum_{q=1}^{N}\varepsilon^{q\,2} - \frac{2}{N-1}\sum_{q=1}^{N}\varepsilon^q\,\bar{\varepsilon} + \frac{1}{N-1}\sum_{q=1}^{N}\bar{\varepsilon}^{\,2} = \cdots \\ &= \frac{1}{N-1}\sum_{q=1}^{N}\varepsilon^{q\,2} - \frac{2\,\bar{\varepsilon}}{N-1}\sum_{q=1}^{N}\varepsilon^q + \frac{N}{N-1}\bar{\varepsilon}^{\,2} \end{aligned} \tag{18}$$

Simplifying,

$$\begin{aligned} s_\varepsilon\frac{N-1}{N} &= \frac{1}{N}\sum_{q=1}^{N}\varepsilon^{\,q\,2} - \frac{2\,\bar{\varepsilon}_y}{N}\sum_{q=1}^{N}\varepsilon^{\,q} + \bar{\varepsilon}^{\,2} = \frac{1}{N}\sum_{q=1}^{N}\varepsilon_y{}^{q\,2} - 2\bar{\varepsilon}^{\,2} + \bar{\varepsilon}^{\,2} = \cdots \\ &= \frac{1}{N}\sum_{q=1}^{N}\varepsilon^{q\,2} - \bar{\varepsilon}^{\,2} \end{aligned} \tag{19}$$

Let φ be the factor $\frac{N-1}{N}$,

$$\varphi\, s_\varepsilon = \frac{1}{N}\sum_{q=1}^{N}\varepsilon^{q\,2} - \bar{\varepsilon}^{\,2} \tag{20}$$

Particularizing for ε_y and $\varepsilon_{\hat{y}}$,

$$\varphi\, s_{\varepsilon_y} = \frac{1}{N}\sum_{q=1}^{N}\varepsilon_y{}^{q\,2} - \bar{\varepsilon}_y{}^2 = \frac{1}{N}\sum_{q=1}^{N}(y^q - \mu^q)^2 - \bar{\varepsilon}_y{}^2 \tag{21}$$

and

$$\varphi\, s_{\varepsilon_{\hat{y}}} = \frac{1}{N}\sum_{q=1}^{N}\varepsilon_{\hat{y}}{}^{q\,2} - \bar{\varepsilon}_{\hat{y}}{}^2 = \frac{1}{N}\sum_{q=1}^{N}(\hat{y}^q - \mu^q)^2 - \bar{\varepsilon}_{\hat{y}}{}^2 \tag{22}$$

As stated, the goal is to prove that

$$s_{\varepsilon_{\hat{y}}} \le s_{\varepsilon_y} \tag{23}$$

Considering λ^q,

$$\varphi\, s_{\varepsilon_y} = \frac{1}{N}\sum_{q=1}^{N}(\hat{y}^q - \mu^q + \lambda^q)^2 - \bar{\varepsilon}_y{}^2 = \cdots$$

$$= \frac{1}{N}\sum_{q=1}^{N}(\hat{y}^q - \mu^q)^2 + \frac{2}{N}\sum_{q=1}^{N}(\hat{y}^q - \mu^q)\,\lambda^q + \frac{1}{N}\sum_{q=1}^{N}\lambda^{q\,2} - \bar{\varepsilon}_y{}^2 \qquad (24)$$

The first and last terms can be grouped using (22),

$$\varphi\, s_{\varepsilon_y} = \varphi\, s_{\varepsilon_{\hat{y}}} + \frac{2}{N}\sum_{q=1}^{N}(\hat{y}^q - \mu^q)\,\lambda^q + \frac{1}{N}\sum_{q=1}^{N}\lambda^{q\,2} = \cdots$$

$$= \varphi\, s_{\varepsilon_{\hat{y}}} + \frac{2}{N}\sum_{q=1}^{N}\varepsilon_{\hat{y}}{}^q\,\lambda^q + \frac{1}{N}\sum_{q=1}^{N}\lambda^{q\,2} \qquad (25)$$

If the equation (8) is incorporated,

$$\varphi\, s_{\varepsilon_y} = \varphi\, s_{\varepsilon_{\hat{y}}} + \frac{2}{N}\sum_{q=1}^{N}(\alpha\, x^q + \beta)\,\lambda^q + \frac{1}{N}\sum_{q=1}^{N}\lambda^{q\,2} = \cdots$$

$$= \varphi\, s_{\varepsilon_{\hat{y}}} + \alpha\frac{2}{N}\sum_{q=1}^{N} x^q\,\lambda^q + \beta\frac{2}{N}\sum_{q=1}^{N}\lambda^q + \frac{1}{N}\sum_{q=1}^{N}\lambda^{q\,2} \qquad (26)$$

Let the last term be calculated separately,

$$\frac{1}{N}\sum_{q=1}^{N}\lambda^q = \frac{1}{N}\sum_{q=1}^{N}(y^q - \hat{y}^q) = \frac{1}{N}\sum_{q=1}^{N}y^q - \frac{1}{N}\sum_{q=1}^{N}\hat{y}^q = \bar{y} - \bar{\hat{y}} = 0 \qquad (27)$$

Thus β has no influence in the prove of $s_{\varepsilon_{\hat{y}}} \le s_{\varepsilon_y}$,

$$\varphi\, s_{\varepsilon_y} = \varphi\, s_{\varepsilon_{\hat{y}}} + \alpha\frac{2}{N}\sum_{q=1}^{N} x^q\,\lambda^q + \frac{1}{N}\sum_{q=1}^{N}\lambda^{q\,2} \qquad (28)$$

The relation $s_{\varepsilon_{\hat{y}}} \le s_{\varepsilon_y}$ *is* the same as $\varphi\, s_{\varepsilon_{\hat{y}}} \le \varphi\, s_{\varepsilon_y}$ as long as $\varphi \ge 0$ (which in this case is). Thus, as long as

$$\alpha\frac{2}{N}\sum_{q=1}^{N} x^q\,\lambda^q + \frac{1}{N}\sum_{q=1}^{N}\lambda^{q\,2} \ge 0 \qquad (29)$$

then $s_{\varepsilon_{\hat{y}}} \le s_{\varepsilon_y}$ holds. The second term is always positive, but the first can be negative (it might alternate for each datapoint). In the end, what determines the veracity of the expression (29) is the slope of the difference of $\hat{y}$ and μ, that is, α:

$$2\,\alpha\sum_{q=1}^{N} x^q\,\lambda^q + \sum_{q=1}^{N}\lambda^{q\,2} \ge 0 \qquad (30)$$

Using the final values of the series, S_1 and S_2,

$$2\,\alpha\, S_1 + S_2 \ge 0 \qquad (31)$$

$$\alpha\, S_1 \geq -\frac{S_2}{2} \tag{32}$$

Table 1 shows all the possible scenarios,

Table 1. Possible scenarios.

α	S_1	Constrain for $s_{\varepsilon_{\hat{y}}} \leq s_{\varepsilon_y}$
≥ 0	≥ 0	$\forall\, \alpha \geq 0$
≥ 0	≤ 0	$\alpha \leq -\frac{S_2}{2\, S_1} \;\rightarrow\; \alpha \leq \frac{S_2}{2\, \lvert S_1 \rvert}$
≤ 0	≥ 0	$\alpha \geq -\frac{S_2}{2\, S_1}$
≤ 0	≤ 0	$\forall\, \alpha \leq 0$

The variable B is used to group these boundaries,

$$B = \frac{S_2}{2\, \lvert S_1 \rvert} = \frac{\sum_{q=1}^{N} {\lambda^q}^2}{2\, \left\lvert \sum_{q=1}^{N} x^q \lambda^q \right\rvert} = \frac{\sum_{q=1}^{N} (y^q - \hat{y}^q)^2}{2\, \left\lvert \sum_{q=1}^{N} x^q\, (y^q - \hat{y}^q) \right\rvert} \tag{33}$$

Thus, the conditions of noise resilience, using the hypothesis of locality, can be expressed as

$$\begin{cases} -B \leq \alpha \leq B & if\;\; \mathrm{sgn}(\alpha) \neq \mathrm{sgn}(S_1) \\ \forall\, \alpha & if\;\; \mathrm{sgn}(\alpha) = \mathrm{sgn}(S_1) \end{cases} \tag{34}$$

It should be reminded that the parameters α and β represent the amount of noise that is still present after doing the prediction. It has been shown that α has a direct influence in the variance of the distribution, and β can be understood as an offset which has no impact in $s_{\varepsilon_{\hat{y}}}$. This offset can be compensated when α satisfies the conditions of noise resilience, which may result in further improvement of the approximation. Finally, from the intersection of (34), it follows that as long as α is bounded between $-B$ and B (i.e., as long as the noise is bounded between these two values), the expression $s_{\varepsilon_{\hat{y}}} \leq s_{\varepsilon_y}$ always holds.

3.3 NR, Global Noise Resilience Score

In this final section, the authors suggest a figure of merit to quantitatively determine the resilience to noise of a given approximator. This would consider both $s_{\varepsilon_{\hat{y}}}$ and s_{ε_y}. Previously the objective was to provide under what circumstances the prediction was better than the noisy data. This next task should focus on the calculation of an accumulated measurement of resiliency. In fact, that figure could be simply the difference between s_{ε_y} and $s_{\varepsilon_{\hat{y}}}$. Let (28) be retaken, then,

$$s_{\varepsilon_y} = s_{\varepsilon_{\hat{y}}} + \alpha \frac{2}{N\,\varphi} \sum_{q=1}^{N} x^q\,\lambda^q + \frac{1}{N\,\varphi} \sum_{q=1}^{N} {\lambda^q}^2 \tag{35}$$

$$s_{\varepsilon_y} - s_{\varepsilon_{\hat{y}}} = \alpha\, \frac{2}{N-1} \sum_{q=1}^{N} x^q\,\lambda^q + \frac{1}{N-1} \sum_{q=1}^{N} {\lambda^q}^2 \tag{36}$$

Let the dataset have a total of J different regions that can be approximated linearly, and let NR be the noise resilience score (the metric suggested).

$$NR = \frac{1}{J} \sum_{j=1}^{J} \frac{s_{\varepsilon_{y_j}} - s_{\varepsilon_{\hat{y}_j}}}{s_{\varepsilon_{y_j}}} \tag{37}$$

Each jth region would have different values of $s_{\varepsilon_{y_j}}$, $s_{\varepsilon_{\hat{y}_j}}$, N_j and α_j, thus,

$$NR = \frac{1}{J} \sum_{j=1}^{J} \frac{1}{s_{\varepsilon_{y_j}}} \frac{1}{N_j - 1} \left(2\,\alpha_j \sum_{q=1}^{N_j} x^q\,\lambda^q + \sum_{q=1}^{N_j} {\lambda^q}^2 \right) \tag{38}$$

From (21),

$$s_{\varepsilon_{y_j}} = \frac{1}{N_j - 1} \sum_{q=1}^{N_j} {\varepsilon_y^q}^2 - \frac{N_j}{N_j - 1}\, {\bar{\varepsilon}_{y_j}}^2 \tag{39}$$

$$NR = \frac{1}{J} \sum_{j=1}^{J} \frac{2\,\alpha_j \sum_{q=1}^{N_j} x^q\,\lambda^q + \sum_{q=1}^{N_j} {\lambda^q}^2}{\sum_{q=1}^{N_j} {\varepsilon_y^q}^2 - N_j\, {\bar{\varepsilon}_{y_j}}^2} \tag{40}$$

The resilience score ranges from $(-\infty, 1]$. For a given amount of noise, a value of 1 in NR would mean that the approximator has pure noise resilience. A case in which for all the regions, the variance $s_{\varepsilon_{\hat{y}_j}}$ is null.

The partitioning of the domain was done such that the truth of each small region could be approximated by a linear function, where each datapoint is only considered once. Nevertheless, the score could be further improved with non-exclusive regions of datapoints, using a sliding window of variable width for example (if the input is one-dimensional). This way, each point would be considered multiple times, while the NR would still be defined by (40).

4 Conclusion

This development can serve as a possible method to measure noise resilience in regression algorithms. It might as well be useful to quantify overtraining, although a specific figure of merit should be developed for that phenomenon.

Future work should also focus on the generalization of the conditions of noise resilience and the resilience score to multidimensional problems. Additionally, the application of NR is encouraged as a metric to quantify and compare objectively the performance in the presence of noise. Finally, Table 2 shows the definition and attributes of some of the popular metrics in regression together with the score presented in this research.

Table 2. Overview of popular metrics and the attributes they evaluate from the regression algorithm, together with the proposed NR score.

Metric	Definition	Attribute
MSE	$\frac{1}{Q}\sum_{q=1}^{Q}(y^q - \hat{y}^q)^2$	Accuracy • Penalizes large errors
$RMSE$	$\sqrt{\frac{1}{Q}\sum_{q=1}^{Q}(y^q - \hat{y}^q)^2}$	Accuracy • Same information as the MSE
MAE	$\frac{1}{Q}\sum_{q=1}^{Q}\lvert y^q - \hat{y}^q\rvert$	Accuracy • Penalizes overprediction
$MAPE$	$\frac{100\%}{Q}\sum_{q=1}^{Q}\left\lvert\frac{y^q - \hat{y}^q}{y^q}\right\rvert$	Accuracy • Same info as MAE in percentage
$SMAPE$	$\frac{100\%}{Q}\sum_{q=1}^{Q}2\frac{\lvert y^q - \hat{y}^q\rvert}{\lvert y^q\rvert + \lvert\hat{y}^q\rvert}$	Accuracy
R^2	$1 - \frac{\sum_{q=1}^{Q}(y^q - \hat{y}^q)^2}{\sum_{q=1}^{Q}(y^q - \bar{y})^2}$	Accuracy • Per unit explained variation
ME	$\frac{1}{Q}\sum_{q=1}^{Q}y^q - \hat{y}^q$	Bias
MPE	$\frac{100\%}{Q}\sum_{q=1}^{Q}\frac{y^q - \hat{y}^q}{y^q}$	Bias • Same info as ME in percentage • Penalizes overprediction
NR	$\frac{1}{J}\sum_{j=1}^{J}\frac{2\,\alpha_j\sum_{q=1}^{N_j}x^q\,\lambda^q + \sum_{q=1}^{N_j}{\lambda^q}^2}{\sum_{q=1}^{N_j}{\varepsilon_y}^{q\,2} - N_j\,{\bar{\varepsilon}_{y_j}}^2}$	Noise Resilience

Acknowledgements

The project that generated these results was supported by a grant from the ”la Caixa” Banking Foundation (ID 100010434), whose code is LCF / BQ / AA19 / 11720045.

References

1. Liu, J., Feng, K., Qu, Y. et al.: Stability analysis of T–S fuzzy coupled oscillator systems influenced by stochastic disturbance. Neural Comput & Applic, (2020).
2. Ontiveros-Robles, E., Melin P., Castillo, O.: Comparative analysis of noise robustness of type 2 fuzzy logic controllers. Kybernetika 54, 175 – 201 (2018).
3. Hoseini, S. A., Labibi, B.: Robust fuzzy controller design with bounded control effort for nonlinear systems with parametric uncertainties. In: 2009 International Conference on Networking, Sensing and Control, pp. 118-123. IEEE, Okayama, Japan (2009).
4. Sathyan, A., Cohen, K.: Development of a Genetic Fuzzy Controller and its Application to a Noisy Inverted Double Pendulum. Fuzzy Logic Based in Optimization Methods and Control Systems and its Applications, IntechOpen, pp. 27–40 (2018).
5. Aman, S., Simmhan, Y., Prasanna, V. K.: Holistic measures for evaluating prediction models in smart grids. IEEE Transactions on Knowledge and Data Engineering, 27(2), 475- 488 (2015).
6. Chai, T., Draxler, R. R.: Root mean square error (RMSE) or mean absolute error (MAE)?–Arguments against avoiding RMSE in the literature. Geoscientific Model Development, 7(3), 1247-1250 (2014).
7. Kitchenham, B. A., Pickard, L. M., MacDonell, S. G., & Shepperd, M. J. (2001). What accuracy statistics really measure? IEE Proceedings-Software, 148(3), 81-85.
8. Makridakis, S. (1993). Accuracy measures: theoretical and practical concerns. International Journal of Forecasting, 9(4), 527-529.
9. Morley, S. K., Brito, T. V., & Welling, D. T. (2018). Measures of model performance based on the log accuracy ratio. Space Weather, 16(1), 69-88.
10. Likassa, H. T., Fang, W.: Robust Regression for Image Alignment via Subspace Recovery Techniques. In: Proceedings of the 2018 VII International Conference on Network, Communication and Computing (ICNCC 2018), pp. 288–293. Association for Computing Machinery, New York, NY, USA (2018).
11. Anubi. O. M., Konstantinou, C.: Enhanced Resilient State Estimation Using Data-Driven Auxiliary Models. IEEE Transactions on Industrial Informatics 16(1), 639-647 (2020).
12. Brandman, D. M., Burkhart, M. C., Kelemen, J., Franco, B., Harrison, M. T., Hochberg L. R.: Robust Closed-Loop Control of a Cursor in a Person with Tetraplegia using Gaussian Process Regression. Neural Computation 30(11), 2986-3008 (2018).
13. Li, J., Chen, Q., Leng, J. et al. : Probabilistic robust regression with adaptive weights — a case study on face recognition. Front. Comput. Sci. 14(145314) (2020).
14. Both, G., Choudhury, S., Sens, P., Kusters, R.: DeepMoD: Deep learning for model discovery in noisy data. Journal of Computational Physics 428 (2021).
15. Li, M., Soltanolkotabi, M., Oymak, S.: Gradient Descent with Early Stopping is Provably Robust to Label Noise for Overparameterized Neural Networks. Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, PMLR 108, 4313-4324 (2020).
16. Dick, K., Russell, L., Souley Dosso, Y., Kwamena, F., Green, J. R.: Deep Learning for Critical Infrastructure Resilience. Journal of Infrastructure Systems 25(2), (2019).