Are Neural Networks Really the Holy Grail? A Comparison of Multivariate Calibration for Low-Cost Environmental Sensors

Xinwei Fang(✉), Iain Bate, and David Griffin

Department of Computer Science, University of York, York, UK
{Xinwei.Fang,Iain.Bate,David.Griffin}@york.ac.uk

Abstract. Data obtained from low-cost environmental sensors can have various issues such as low precision and accuracy and incompleteness. A calibration process is often applied to address such issues. With the recent advances in artificial intelligence, we have seen an increased number of applications that starts to use an artificial neural network (ANN) to calibrate the sensors, and their results are promising. In this work, we used a six-months worth of real hourly data to demonstrate that the ANN may not always be the best choice of a calibration method. Our evaluation compares an ANN-based method with a simple regression-based method in various aspects. The result shows that the ANN-based method does not consistently outperform the regression-based method. More interestingly, in the comparison, our results suggest that the performance of a calibration can be more sensitive to some of the factors (e.g. training and testing data, model parameters) than the use of different calibration methods. Even though the results may not be generalised in other sensors or datasets, our evaluation provides evidence showing that inappropriate use of a calibration method can compromise the calibration result, and the use of the ANN will not magically solve that problem.

Keywords: Low-cost sensors · Sensor calibration

1 Introduction

Low-cost environmental sensors have been widely used in monitoring of urban environment as they can provide much better spatial and temporal resolutions than the regulatory monitoring instruments [3,4,9,15]. However, the low-cost sensors are prone to temporary failure and are sensitive to the environmental interference, which results in the obtained data being much less structural in term of size, completeness and integrity [11]. More importantly, the data quality from these low-cost sensors is often reported to be insufficient and requires pre-processing [3,13,19,24].

Sensor calibration is one of a process to improve data quality. In this paper, sensor calibration is to determine a model that transfers the data of low-cost

S. Paiva et al. (Eds.): SmartCity360° 2020, LNICST 372, pp. 442–461, 2021.
https://doi.org/10.1007/978-3-030-76063-2_30

sensors to minimise the difference with the data from the co-located reference instruments. According to the literature, the state-of-the-art in-field sensor calibrations often use multiple variables to calibrate a sensor, which is referred to as multivariate calibration [7–10]. Multivariate calibration means the calibration model is constructed using not only the parameter of interest but also other supporting parameters, e.g. including temperature when calibrating NO_2 [18]. The intuition is if the response of NO_2 is related to or affected by the temperature, a more accurate calibration of NO_2 can be determined if it includes the temperature and accounts for the related effects.

Multivariate calibration can be accomplished in many ways, and the two most prominent methods seen in current literature are a simple regression-based method and and artificial neural networks (ANNs) based method [7,9,10,12,19]. With the recent advance in machine learning, we have seen an increased number of applications that starts to use an ANN-based method to calibrate the sensors, and their results are promising. This makes us wonder if the ANN-based methods can also work better on the *imperfect* data (e.g. the small size, noisy data) and whether the ANN-based method should always be the first choice when comes to the selection of a calibration method.

This paper presents a systematic comparison of those two calibration techniques (i.e. a regression-based method and an ANN-based method) using a real dataset, and focuses on determining how their calibration results can be affected under various conditions. This work not only compares the calibration accuracy but also analyses the sensitivity of each method to different settings of training and testing dataset. This gives us an evidence and insight to reason weather the ANN-based method is really the holy grail in the calibration of low-cost sensors.

Main contribution: Even though a few existing works have demonstrated the comparison of the calibration methods for calibrating low-cost sensors, to the best of our knowledge, this paper is the first work that focus on the sensitivity of the calibration methods with respect to various scenarios (e.g. *imperfect* data). With the main contribution, the following additional contributions are made:

- **Reality:** Real hourly data from 6-month worth of deployment was used to simulate the calibration of sensors for a short deployment.
- **Practicality:** The selection of model parameters are demonstrated to show the variability of the calibration process, which are often ignored in the existing comparison.
- **Sensitivity:** Both models are trained and tested under different settings to gain an in-depth knowledge on the sensitivity of the methods.

After the review of the existing comparison of multivariate calibrations in Sects. 2, 3 explains how the calibration models can be constructed using both approaches and what the model parameters need to be determined; Sect. 4 illustrates the determination of the model parameters for both approaches; Sects. 5, 6, and 7 compare the approaches in conditions of model generation, varying training and testing dataset and varying data characteristic respectively. Section 8 concludes the paper.

2 Related Work

We have seen an increasing number of sensor calibration starts to use an ANN-based method to calibrate a low-cost sensor [19,24]. However, to the best of our knowledge, a little work has done to demonstrate a systematic comparison of different calibration methods, especially when calibrated data are *imperfect*.

A prominent existing comparison, such as [24], is limited to comparing the calibration result in terms of calibration accuracy, which is often represented as the averaged error between the model predictions and the reference, e.g. root-mean-squared error (RMSE) or mean-absolute error (MAE). Since two identical averaged errors may represent different error distributions, using an averaged error as the only metric for the comparison would not help us to gain an insight of the performance. Further, while a focus on aggregate measures allows us to characterise the mean error of different methods, such metrics can be skewed by outliers and hence are insufficient to determine which method is most likely to give the best results.

To solve that issue, authors in [7,10] provided a more detailed comparison for multivariate calibration approaches. In their work, the approaches were cross-compared not only for the calibration accuracy (determined by the mean absolute error) but also for the capability of dealing with different training scenarios. In work [10], the calibration result was compared by varying a different number of training and testing samples with more than 40000 instances in total. However, it is noted that the variation of the training and testing samples were divided by a cut-off value. Since a cut-off value can change the size of both the training and testing dataset, it difficult to determine which changes are responsible for the variation of the result.

Devito et al. [7] analysed how the calibration accuracy was affected by using different model parameters. For example, Devito et al. compared the calibration accuracy by varying the certain model parameters in the ANN network. In that case, Devito et al. would have to assume that the model parameters of the ANN are independent or partially dependent. However, this assumption does not hold in our evaluation as demonstrated in Sect. 4.3.

3 Determine the Calibration Models

Sensor calibration is a process of finding a calibration model that minimises the difference between the model output and the reference. In this section, we demonstrate how calibration models can be constructed using both methods, and discuss what model parameters are important for each of the methods. For the demonstration, we assume that calibrating X_1 to its reference $\tilde{X}_1$ requires X_2 and X_3 as the supporting parameters. Then, we further assume that all the parameters, including the reference, have the same number of samples taken during the same time window.

3.1 Calibration Using an ANN-Based Method

A mathematical representation of an ANN can be extremely complicated, as discussed in [5]. Hence, instead of a detailed mathematical construct, we use more abstract notations of ANN's. The training process determines the calibration model. The model would provide an approximation of the calibrated $\tilde{X}_1$ given the uncalibrated inputs, X_1, X_2 and X_3. According to the literature, there are a number of model parameters are important for the ANN-based method [22], which are summarised in Table 1.

Table 1. Model parameters to be needed for an ANN-based method

Model parameters	Examples
Activation function	Sigmoid, ReLU [21], SeLU [17], etc.
Number of neurons	1 to $+\infty$
Number of layers	1 to $+\infty$
Type of neurons	Dense, LSTM, etc.
Batch size	1 to the total number of training samples
Epoch	1 to $+\infty$
Loss function	Mean squared error, mean absolute error, etc.
Optimisation method	Gradient descent, Adam, etc.

3.2 Calibration Using a Regression-Based Method

In contrast to the ANN-based method, a regression-based method is easier to be presented mathematically, and it does not require many pre-determined parameters [12,19]. For example, a linear calibration model to calibrate X_1 using the corresponding coefficients β_i can be constructed based on Eq. 1.

$$\tilde{X}_1(i) = \beta_0 + \beta_1 \cdot X_1(i) + \beta_2 \cdot X_2(i) + ... + \beta_n \cdot X_n(i) + \varepsilon(i) \tag{1}$$

In Eq. 1, ε stands for the error term and the i indicates that the measurements are taken from the same time frame. $\tilde{X}_1$ is the reference of X_1; X_2 to X_n are the supporting parameters of the calibration. The calibration model is then to determine the coefficient β based on the Eq. 2.

$$\mathcal{E} = minimise \sum_{i=1}^{N} \varepsilon(i)^2 \tag{2}$$

Note that the example in Eq. 1 uses a linear combination of first order terms to describe the relationship between the inputs variables and output (i.e. linear). If a more complex non-linear relationship needs to be utilised in the model, a pre-determination of the model is required (e.g. including non-linear terms or applying a non-linear transformation). Therefore, we consider the relationship between input variables and output as the only model parameter for the regression based method.

4 Determining the Model Parameters

In this section, we demonstrate the determination of the model parameters, and discuss the practical issues encountered during the process. Firstly, we present the data and programming environment used for this experiment. Then, we demonstrate the process for both methods respectively.

4.1 Data and Programming Environment

An ELM unit, a product from Perkin Elmer [23], is used as low-cost sensor in this work. The unit was situated at York, UK, next to a busy junction. It measures multiple parameters: nitrogen dioxide (NO_2), ozone (O_3), nitrogen oxide (NO), temperature (T), humidity (H). The ELM unit was co-located with a regulatory monitoring instrument from [6], and the hourly NO_2 data from this instrument was used as the reference for the sensor calibration in this work. Due to restrictions on reporting the data, we are unable to provide information on the exact quantities, including units, but all data are comparable.

The collected data is pre-processed in advance, which aggregates the ELM data into the same temporal resolution as the reference (hourly) and excludes data gaps in the averaged data. The process ensures the consistent samples in the dataset and it is required by the method. After the process, the dataset has 4000 samples with a temporal resolution of one hour.

The regression based method was programmed in Matlab, and the ANN-based method was programmed in Python using *Keras* library [16] and TensorFlow [26].

For the selection of model parameters, the entire dataset was divided sequentially into two equally sized partitions. The first 2000 samples are used as training (i.e. the first half of sensor's operative time span) and the rest of the samples are used as testing. This is to simulate the situation where only 2000 samples were available (i.e. a short development with 3-month worth of data) for training the model. Furthermore, as calibrating NO_2 is often reported to be problematic and would require multivariate calibration to compensate [18,20,24], the calibration of NO_2 is used as an example for this paper.

4.2 Model Parameters for a Regression Based Method

Most of the existing works for the regression based method utilise the linear relationship to construct the calibration model [12,19]. This experiment is to determine whether using a more complex relationship (e.g. non-linear) would improve the calibration accuracy. The complex relationship is referred to as adding higher order terms into the existing linear model.

For the experiment, the calibration errors from using different models are illustrated in Fig. 1. The calibration error is defined as the difference between the model output (y) and the reference (Y), given by Eq. 3. It is noted that i indicates the number of samples.

$$error(i) = Y(i) - y(i) \tag{3}$$

In the figure, the number in the X-axis differentiate calibration models. The first model uses a linear combination of first order terms, which is identical to Eq. 1, and expressed as $f(NO_2, O_3, NO, T, H)$. The following models are constructed by gradually including a second order term into the existing model as well as their interactions [14]. We express the second model as $f(NO_2, O_3, NO, T, H, (NO_2)^2)$ and the last model as $f(NO_2, O_3, NO, T, H, (NO_2)^2, (O_3)^2, (NO)^2, (T)^2, (H)^2)$. The experiment tests all the possible combinations, i.e. $\binom{0}{5} + \binom{1}{5} + \binom{2}{5} + \binom{3}{5} + \binom{4}{5} + \binom{5}{5}$, which have 32 models in total. In the figure, X-axis (1) indicates the linear model; whereas X-axis (2) to (32) indicates non-linear model in which one or more higher order terms were introduced.

Figure 1 shows that utilising a more complex relationship in the calibration model does not appear to improve the calibration result. Therefore, a linear relationship is used for the model of the regression based method.

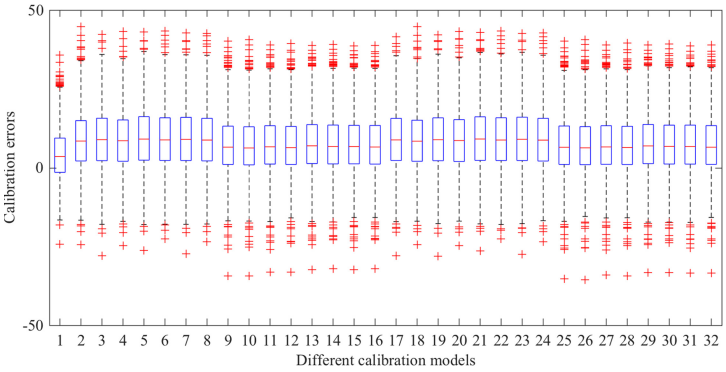

Fig. 1. The error distributions for different model settings

4.3 Model Parameters for an ANN-Based Method

It is noted that the determination of the model parameters for an ANN-based method, which is also known as hyperparameter optimisation [2], is still an open challenge, and the trial by error is currently the best practice for this purpose [1]. Since the parameters often need to be selected from a large parameter space, it would be impractical to test all possible combinations. Therefore, the selection of the parameters in this paper is tested in a certain range only, for which the decision is made based on either existing works or expert knowledge. This also reflects the disadvantage of using ANN.

Activation function. The Sigmoid, RuLU, and SeLU are tested. It is clear that each neuron can have a different activation function. However, since it is impractical to test the combination of activation functions, the same activation function is applied to all neurons in a network setting.

Type of neurons. Dense and LSTM are tested. As above, due to the exponential cost of varying each neuron, the same type of neurons are used in all neurons in the network.

Number of neurons and layers. We vary the number of neurons in each layer as [5 20 35] and the number of layers in [1 2 3 4 5]. The same number of neurons are used in each layer. These test ranges was chosen as the similar range of the parameters was used in [7].

Batch size and epoch. We test the number of batch size in [1 6 11 16 21 26] and epoch in [1 6 11 16 21 26], which is 1 to 26 with an increment of 5, as no significant different in results can be determined with further increase of the batch and epoch sizes.

Loss function. We test the Mean Absolute Error (MAE), Poisson and Mean Squared Error (MSE) as the loss function in the experiment as they are often used as the evaluation of sensor calibrations.

Optimization method. Gradient descent, RMSprop and Adam are tested in the experiment.

In the experiment, we vary all eight model parameters. As a result, the model parameters would be selected from eight dimensional parameter space. The selection is based on the Root Mean Squared Error (RMSE) between the reference and the model output. We use determination of the loss function as example to demonstrate how the model parameter is selected.

We classify all networks into three groups with respect to the use of the loss functions. Then, we determine the percentage of the model in each group that the error in terms of RMSE is below an RMSE threshold. The RMSE threshold varies from small to large, and the process is applied to all three groups. The result would indicate the difference between the loss function. We consider the optimal parameters as the one that has the highest percentage with the lowest RMSE threshold. The result is showing in Fig. 2.

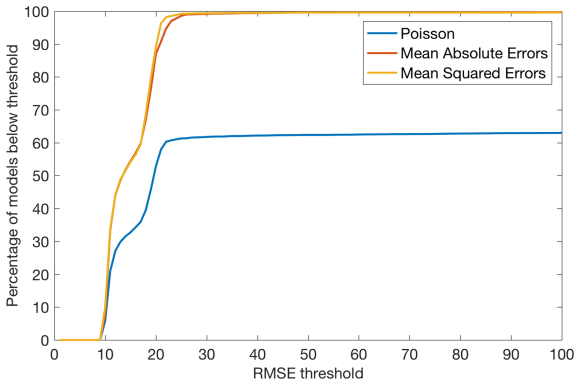

Fig. 2. Percentage of model below threshold for different loss functions

In addition, we perform a statistical test to determine the probability that one method is more likely to produce a better result than another. This is accomplished by fixing parameters of a given test apart from the parameter of

interest. Once the data is gathered, a null hypothesis test [25] is conducted with the null hypothesis being that the varying the parameter between two values has no effect. If the evidence supports the alternative hypothesis, that one of the parameter values has consistently better performance, we can reject the null hypothesis for this configuration. By repeating this experiment across all possible values of other parameters, we can derive an estimate for the probability that a parameter is more likely to produce a better result; this is shown in Table 2 for the Loss function, where we can conclude Mean Squared Error has highest chance of producing the best result from the three loss functions. The result is also in-line with Fig. 2.

Table 2. Probability of dominance when varying loss function

	Mean squared error	Mean absolute error	Poisson
Mean squared error		42%	63%
Mean absolute error	37%		61%
Poisson	27%	24%	

The same process is applied to all eight parameters. We summarise the parameters used in this work in Table 3.

Table 3. The parameters used in the ANN-based method

Model parameters	Parameter used
Activation function	ReLU
Number of neurons	20
Number of layers	1
Type of neurons	LSTM
Batch size	26
Epoch	21
Loss function	MSE
Optimisation method	Adam

5 Variability of Model Generation

In this section, we want to understand how the model output would be affected by the model generation process. Hence, we train the model with identical parameters multiple times and compare their model outputs, with the only difference being the random seed used for training the ANNs.

The training and testing datasets used in this experiment are identical to the previous experiment, which have been discussed in Sect. 4.1. The RMSE of the calibrated results is shown in Fig. 3.

Figure 3-b shows the 100 model outputs obtained from the regression based method. It is clear that the regression based method provides a consistent result as long as the model settings and the use of the data are identical as the RMSE for

the regression based method shows no variation over the 100 iterations. Figure 3-a presents the 100 model outputs obtained from the ANN-based method. In comparison to the regression based method, Fig. 3-a indicates that the ANN-based method is sensitive to the model generation process as the variation of the model output can be observed. While the ANN-based method does produce a slightly better RMSE, it does not produce a significant advantage.

While there is a clear variation in the RMSE of the ANN-based method, it is comparatively small when compared to the variation in the RMSE from changing the model parameters. Given this, we assume that provided the model parameters are set correctly, the variation in ANN models due to the training process is largely insignificant. Hence for the remainder of this work we will ignore the random effects of training the ANN model.

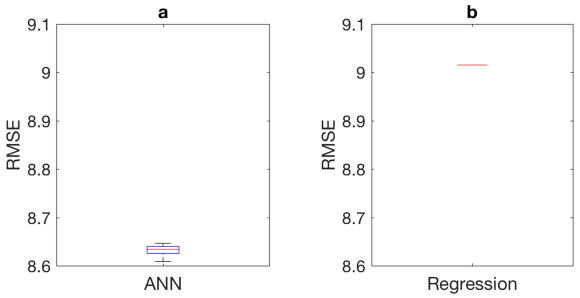

Fig. 3. Comparing the variation of RMSE over 100 repetitions

This section demonstrate that the regression based method would provide a consistent calibration result for the model generation; however, the model generation would introduce a variation in the calibration result for the ANN-based method.

6 Comparing the Model Under Different Training and Testing Scenarios

This section compares the difference between in performance of the methods using different settings of training and testing dataset. The first experiment varies the size of training dataset, and then with varying the size of testing dataset.

6.1 Varying the Training Dataset

This experiment is designed to understand how the increasing size of training dataset would affect the calibration result for both methods. In the experiment, the same dataset used previously is divided sequentially and evenly into to 10 partitions with each partition having 10% of the data and following the temporal dimension. The calibration model is determined by using the training dataset

that gradually increasing the data size; and the result of the calibration is evaluated in the same testing dataset. This could help us to understand how to size of training data plays in the calibration process. The classification and the use of the training and testing dataset are illustrated in Fig. 4.

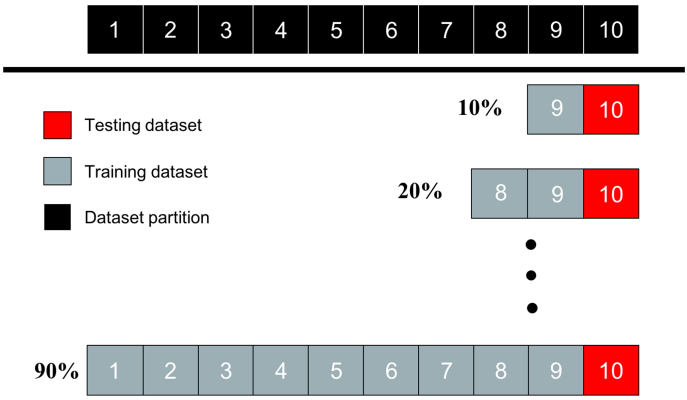

Fig. 4. Varying the training datasets

Figure 4 shows how the data is divided into ten equal partitions, numbered from (1) to (10). For the testing dataset, the last partition (10) is used; and for the training dataset, different combinations of the partitions are applied. As illustrated in Fig. 4, the training dataset steadily increase from 10% of the data to 90% of the data with each step being 10%. In order to preserve the temporal dependencies of the data, the first experiment uses Partition (9) for the training dataset (to preserve the dependencies with Partition (10)). More data is added to the later experiments by going backwards from Partition (9) e.g. the second experiment uses Partitions (8) and (9). We label the use of the different training datasets as 10% to 90% to simplify the labelling in the later plots.

The calibration errors from using the different training datasets are illustrated in the boxplots in the Fig. 5. Comparing the boxplots in the figure, the difference between the methods as well as the effect of increasing size the training dataset is not obvious. Therefore, we plot the mean value of the errors with the confidence interval in Fig. 6 to analyse it further.

In Fig. 6, the bars show the mean of the errors, the error bars indicate the confidence level of the mean. The color of the bar differentiates the calibration methods. The figure shows that the regression-based method would over predict when the training dataset is relatively small, and under predict when the training dataset is relatively large. Whereas, the ANN-based method over predict in all circumstance. The result suggests that the change of the training dataset does have significant impact on the calibration results.

While the averaged errors allow us to determine a general accuracy of the calibration, they do not indicate which method is most likely to give the best result. To accomplish this we perform a null-hypothesis test [25] on the output data of the models, using the null-hypothesis that the two methods are equal -

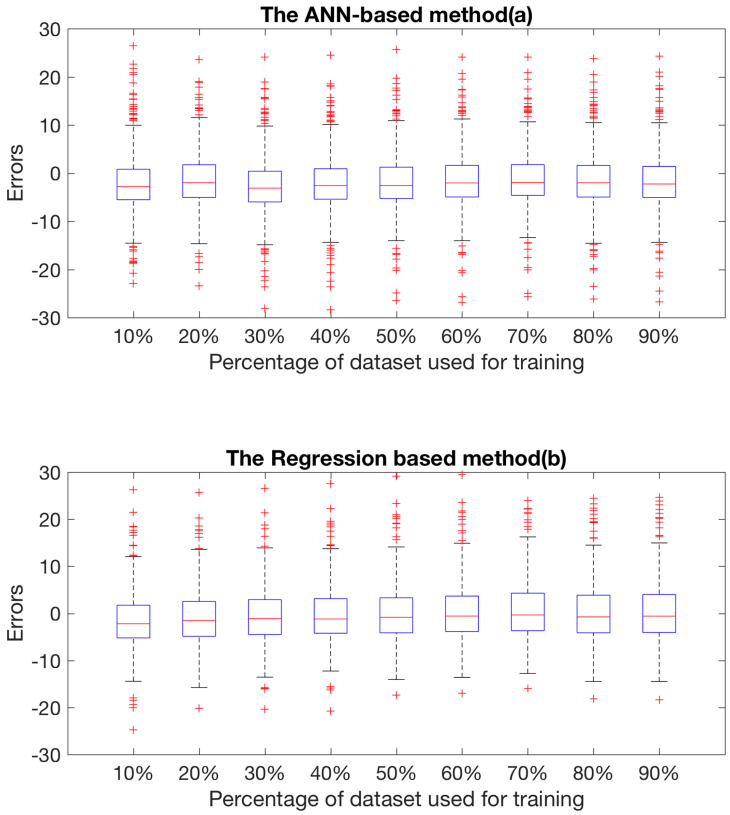

Fig. 5. The errors when using different training datasets

i.e. that for a given input vector, the regression-based method has a 50% chance of producing a lower error than the ANN-based method. We then compute the probability of the actual result of the experiment under the null-hypothesis, and if this probability is sufficiently unlikely, we can reject the null-hypothesis. This method allows us to have statistical confidence in our claim of which method is most likely to produce the lowest error. Our degree of confidence is derived by the standard method of determining how many standard deviations (σ) from the mean of the null-hypothesis deviation the observed result is [25]. At confidence levels above 3σ, we can claim that a method is better, and that at 5σ, we are certain that a method is better. The result is shown in Table 4.

Table 4 shows that the ANN-based method provide consistent better results when a larger training dataset is used. It suggests that an ANN-based method would potentially benefit from using a larger training dataset.

6.2 Varying the Testing Dataset

This experiment is designed to understand how the calibration result is affected by increasing the size of the testing dataset, which may reflect on how long a

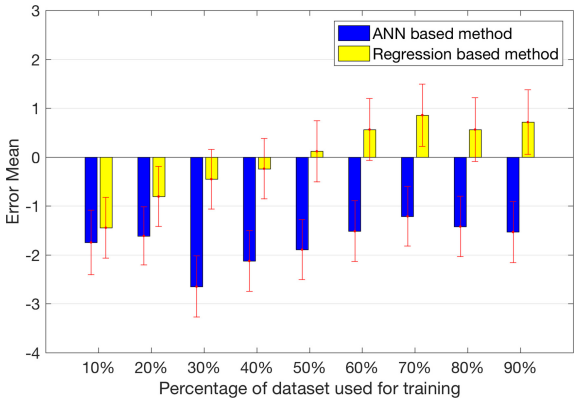

Fig. 6. Error mean with the confidence interval (Color figure online)

Table 4. Significance between calibration results when varying training dataset

Training dataset	P-value	Significance of P-value
10%	0.0976	No significant difference (1σ)
20%	0.0554	ANN potentially better (2σ)
30%	0.9708	Regression better (3σ)
40%	0.5000	No significant difference (0σ)
50%	0.0815	No significant difference (1σ)
60%	0.0063	ANN better (3σ)
70%	0.0083	ANN potentially better (2σ)
80%	0.0035	ANN better (3σ)
90%	0.0010	ANN better (3σ)

Low P-values indicate ANN is better, High P-values indicate regression is better. Results given to 4 decimal places

calibration function can hold. For this experiment, the same dataset is divided into the partitions as in the previous experiment in Sect. 6.1, but the training and testing datasets are utilised differently as illustrated in Fig. 7.

The errors between the model output and the reference when utilising different testing datasets are illustrated in Fig. 8. The boxplots represent the error distribution and x-axis indicate the testing dataset increases from 10% to 90% of the dataset according to Fig. 7. In the figure, using 10% of the data shows the best result for both methods in comparison to using other testing datasets. It suggests that the calibration function would obtain a better result if the testing dataset and the training dataset are close in time and have a similar data size. Furthermore, comparing Fig. 8-a to b, the errors for the ANN-based method contain more extreme values than the regression-based method.

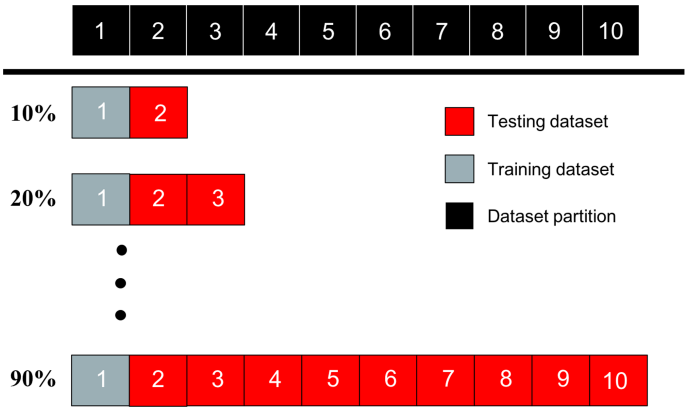

Fig. 7. Training models by varying the testing datasets

Table 5. Significance between calibration results when varying testing dataset

Testing dataset	P-value	Significance of P-value
10%	0.4210	No significant difference (0σ)
20%	0.9880	No significant difference (1σ)
30%	0.9998	Regression better (3σ)
40%	0.9999	Regression better (4σ)
50%	1.0000	Regression certainly better (5σ)
60%	1.0000	Regression certainly better (5σ)
70%	1.0000	Regression certainly better (5σ)
80%	1.0000	Regression certainly better (5σ)
90%	1.0000	Regression certainly better (5σ)

Low P-values indicate ANN is better, High P-values indicate Regression is better. Results given to 4 decimal places

We further plot the mean of the errors with the confidence interval in Fig. 9, which show the error mean and 95% confidence interval from the experiment. The figure shows that the error mean for both methods gradually increase with more testing data used. It suggests that both calibrations would degrade over time with a similar tend, and the performance of the calibration can be more sensitive to the testing dataset than the calibration method.

We also apply the statistical analysis for this experiment, again using the null-hypothesis that the methods are equal. The result is shown in Table 5. The table shows the regression-based method is consistently better than the ANN-based method with the increasing size of the testing dataset. This implies that the degradation of the calibration for the regression-based method is much less significant than the ANN-based method.

7 Influence from the Data Characteristics

In previous section, we have seen that the size of training and testing dataset can have a large impact on the calibration result for both methods. In this section, we investigate how the performance of the calibration methods is sensitive to the change of data characteristics. The experiment was performed using the same dataset as the previous experiments. However, the training dataset was selected based on indices that randomly selected from 50% of the data, and the rest of the data are used for testing. This process is to ensure that the data characteristics between the training and testing datasets are consistent (e.g. training and testing data are from the same distribution). Then, we artificially manipulate the characteristics of the testing datasets to create a different data characteristics. It is clear that the data characteristics can be different in many ways. In this experiment, we consider three properties as they are commonly observed in the low-cost sensors [13]: (1) a sensor outputing a constant value, (2) a sensor outputing an offset value and (3) a sensor outputing values with a greater spread (represented as a higher standard deviation).

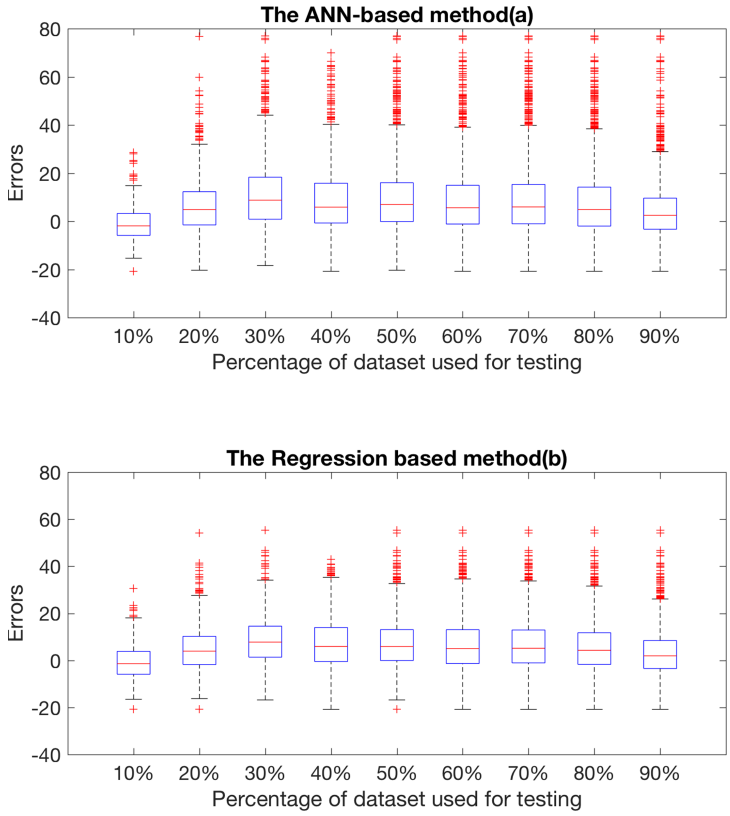

Fig. 8. The errors of both methods when using different testing datasets

Table 6. Different configurations for varying the test data characteristics

Constant value	STD	= 0
	Mean	Not changed
Offset mean	STD	Not changed
	Mean	2*mean
Higher standard deviation	STD	2*STD
	Mean	Not changed

The modification of the testing dataset was performed according to Table 6. The changes of mean and standard deviation are with respect to the original testing data. For the constant value, all samples in the testing dataset are replaced by the mean value of the testing dataset. The offset mean doubles the mean value of the the testing dataset but the standard deviation of the data remains the same. The higher standard deviation changes the standard deviation of the testing dataset but the mean remains. Since different parameters may contribute to the calibration result differently, the modification was performed on all parameters. It is noted that there only one parameter being modified for every calibration.

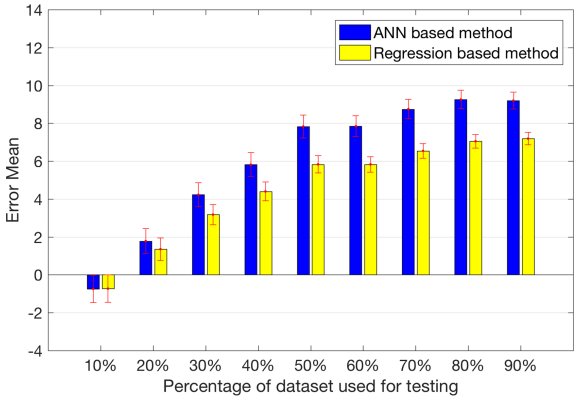

Fig. 9. Error mean with the confidence interval

Table 7. Significance of the calibration result (constant value)

Constant value		
Modified parameter	P-value	Significance of P-value
NO_2	0.9999	Regression better (4σ)
O_3	0.4115	No significant confidence (0σ)
Humidity	0.0006	ANN better (3σ)
Temperature	0.0006	ANN better (3σ)
NO	1.0000	Regression certainly better (5σ)

Low P-values indicate ANN is better, High P-values indicate Regression is better. Results given to 4 decimal places

Table 8. The calibration results when using the testing dataset with constant value

Constant value							
		Original	NO_2	O_3	H	T	NO
ANN based method	RMSE	9.10	8.51	7.17	6.53	6.53	11.74
	Mean	5.33 ± 0.47	−2.22 ± 0.45	−2.55 ± 0.38	−1.33 ± 0.37	−1.33 ± 0.37	−4.61± 0.58
Regression based method	RMSE	6.65	7.86	7.12	6.94	6.94	9.28
	Mean	−0.12 ± 0.37	−0.12 ± 0.42	−0.12 ± 0.39	−0.12 ± 0.38	−0.12 ± 0.38	−0.12 ± 0.52

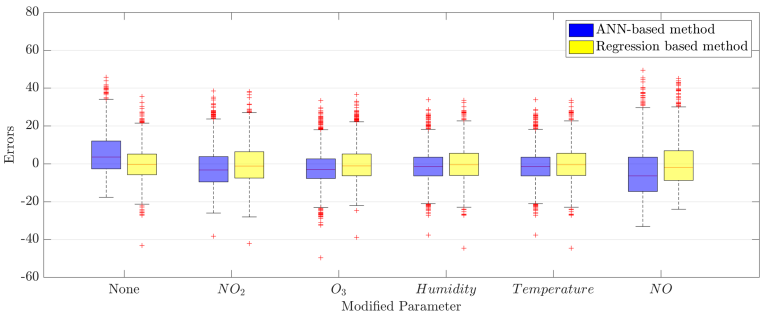

Fig. 10. The calibration errors when using the testing dataset with constant value

Table 9. Significance of the calibration result (offset mean)

Offset mean		
Training dataset	P-value	Significance of P-value
NO_2	1.0000	Regression certainly better (5σ)
O_3	0.9463	ANN potentially better (2σ)
Humidity	1.0000	Regression certainly better (5σ)
Temperature	1.0000	Regression certainly better (5σ)
NO	1.0000	Regression certainly better (5σ)

Low P-values indicate ANN is better, High P-values indicate Regression is better. Results given to 4 decimal places

Table 10. Significance of the calibration result (higher standard deviations)

Higher standard deviation		
Training dataset	P-value	Significance of P-value
NO_2	0.9780	Regression potentially better (2σ)
O_3	0.9239	No significant difference (1σ)
Humidity	0.8683	No significant difference(1σ)
Temperature	0.9413	No significant difference (1σ)
NO	0.9999	Regression better (3σ)

Low P-values indicate ANN is better, High P-values indicate Regression is better. Results given to 4 decimal places

Figures 10, 11 and 12 show the calibration results when the testing dataset of one parameter is modified according to Table 6. The figures differentiate the different modifications, i.e. offset, constant value and higher standard deviation. The boxplots in each figure represent the calibration errors, with the label on the X-axis indicating which parameter (if any) is modified (Table 10).

Figure 10 and Table 8 presents the results when one parameter of the testing data becomes constant. Figure 10 shows no observable difference in terms of the errors, which suggests that the constant value would only have a small impact on both calibration methods. The table indicates that the constant value only causes a small variation in RMSE, and it has even less impact on the error mean, especially for the regression-based method.

The result of the statistical analysis is summarised in Table 7, which shows that the ANN-based method is more sensitive to the NO_2 and NO readings becoming constant, and regression-based method is more sensitive to Humidity and Temperature becoming constant. This indicates that both calibrations may assign different weight to the input parameters when constructing a calibration model.

Figure 11 and Table 11 illustrate the calibration result when the mean value of one parameter is doubled than the original testing dataset. Figure 11 shows a large variation in the errors when the mean value of the testing dataset is modified, which suggests the change of the mean value would have significantly higher impact on the calibration result. Table 11 shows that most of the RMSE and error mean are significantly worse than the result using the unmodified data. The results suggests the change of mean value of the testing dataset would have a great impact on the calibration result. The statistical test is shown in Table 9,

Table 11. The calibration results when using the testing dataset with offset mean

Offset mean							
		Original	NO_2	O_3	H	T	NO
ANN based method	RMSE	9.10	14.68	7.53	11.02	18.34	13.04
	Mean	5.33 ± 0.47	−12.23 ± 0.53	−3.4 ± 0.41	−9.02 ± 0.40	−17.91 ± 0.43	−7.36 ± 0.59
Regression based method	RMSE	6.65	9.76	7.64	6.71	13.67	9.88
	Mean	−0.12 ± 0.37	−6.36 ± 0.42	4.53 ± 0.39	1.39 ± 0.37	13.18 ± 0.38	−2.80 ± 0.15

Table 12. The calibration results when using the testing dataset with a large standard deviation

Higher standard deviation							
		Original	NO_2	O_3	H	T	NO
ANN based method	RMSE	9.10	7.87	6.95	6.87	7.21	8.94
	Mean	5.33 ± 0.47	−1.83 ± 0.45	−1.78 ± 0.38	−1.52 ± 0.38	−2.05 ± 0.39	−1.56 ± 0.53
Regression based method	RMSE	6.65	7.55	6.96	6.66	7.01	8.53
	Mean	−0.12 ± 0.37	−0.13 ± 0.44	−0.13 ± 0.40	−0.12± 0.37	−0.13 ± 0.39	−0.12 ± 0.53

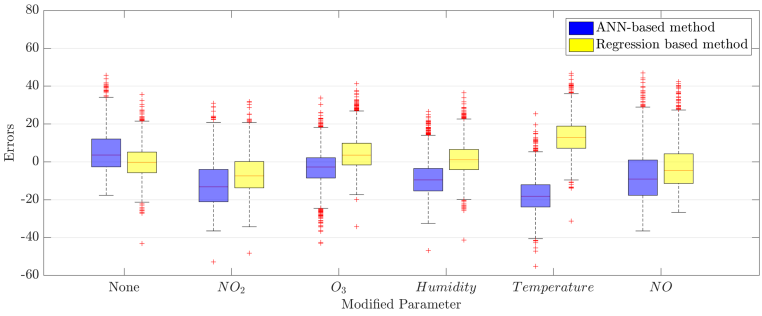

Fig. 11. The calibration errors when using the testing dataset with offset mean

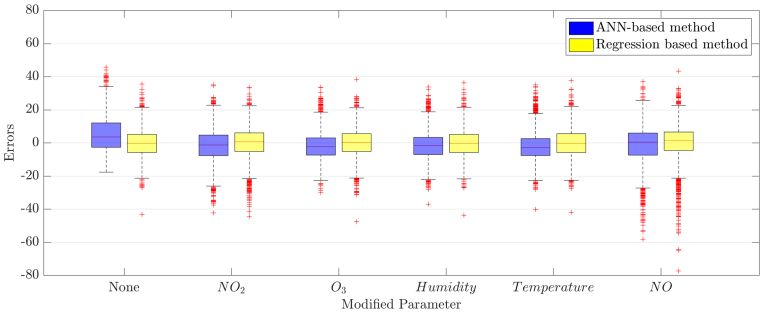

Fig. 12. The calibration errors when using the testing dataset with higher standard deviation

which suggests that the regression-based method is significantly better than the ANN-based method most of the time. It implies that both methods can have different tolerance to drift of mean, and the regression-based method seems to have a better tolerance based on our result.

Figure 12 and Table 12 show the calibration results when one parameter in the testing data have a higher standard deviation. The figure shows that the higher standard deviation in the testing dataset would also have a small impact on the calibration result as the variation of the errors between using modified data and non-modified data is not significant.

Cross-comparing the results above, we conclude that the difference between the training and testing dataset in terms of data characteristics does have a higher impact on the calibration result than the methods itself. However, in general, the ANN-based method is more sensitive to these influences than the regression-based method. Among the different data characteristics, both methods can cope well with the constant value and the higher data standard deviations, but not for the offset mean. This implies that a re-calibration of sensors may be needed if actual training and testing dataset are significantly different in the mean value.

8 Conclusions

This paper provided a systematic comparison between two of the most popular calibration methods, regression-based method and ANN-based method, with detail sensitivity analysis under various conditions.

The comparison shows that the calibration results are extremely sensitive to some of the factors such as the use of hyperparameters in the calibration models or different training and testing datasets. The calibration result can be more sensitive to some of those factors than the use of different calibration methods. In addition, in our comparison, the ANN-based method did not consistently show a better calibration result compared to the regression-based method, and in some of the conditions, it performed much worse than the regression-based method. The result suggests that the ANN-based method may not always be the best option for calibrating a low-cost sensor as the its performance is sensitive to many factors.

Even though some of the results obtained in this study may not be generalised in or directly applied to other sensors or datasets, we have a good reason to believe based on our evaluation that the performance of a sensing calibration is not only dependent on the use of a method but also heavily related to many associated factors (e.g. the training and testing data, the selection of model parameter, the characteristic of the monitored data). Therefore, understanding the key factors and their influence can be important for selecting an appropriate calibration method.

Acknowledgement. This work is funded by the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 608014 (CAPACITIE).

References

1. Bashiri, M., Geranmayeh, A.: Tuning the parameters of an artificial neural network using central composite design and genetic algorithm. Scientia Iranica **18**(6), 1600–1608 (2011)
2. Claesen, M., De Moor, B.: Hyperparameter search in machine learning (2015)
3. Castell, N., et al.: Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates? Environ. Int. **99**, 293–302 (2017)
4. Cheng, Y., et al.: Aircloud: a cloud-based air-quality monitoring system for everyone. In: Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems, pp. 251–265. ACM (2014)
5. Coolen, A.: A beginners guide to the mathematics of neural networks. In: Landau, L.J., Taylor, J.G. (eds.) Concepts for Neural Networks, pp. 13–70. Springer, London (1998). https://doi.org/10.1007/978-1-4471-3427-5_2
6. Department for Environment Food & Rural Affairs: Monitoring networks (2017). https://uk-air.defra.gov.uk/networks/
7. Devito, S., et al.: Calibrating chemical multisensory devices for real world applications: an in-depth comparison of quantitative machine learning approaches. Sens. Actuators, B Chem. **255**, 1191–1210 (2018)

8. Devito, S., Piga, M., Martinotto, L., Difrancia, G.: co, no_2 and no_x urban pollution monitoring with on-field calibrated electronic nose by automatic Bayesian regularization. Sens. Actuators, B Chem. **143**(1), 182–191 (2009)
9. Esposito, E., Devito, S., Salvato, M., Bright, V., Jones, R., Popoola, O.: Dynamic neural network architectures for on field stochastic calibration of indicative low cost air quality sensing systems. Sens. Actuators, B Chem. **231**, 701–713 (2016)
10. Esposito, E., De Vito, S., Salvato, M., Fattoruso, G., Di Francia, G.: Computational intelligence for smart air quality monitors calibration. In: Gervasi, O., et al. (eds.) ICCSA 2017. LNCS, vol. 10406, pp. 443–454. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-62398-6_31
11. Fang, X., Bate, I.: Issues of using wireless sensor network to monitor urban air quality. In: Proceedings of the First ACM International Workshop on the Engineering of Reliable, Robust, and Secure Embedded Wireless Sensing Systems (FAILSAFE). ACM (2017)
12. Fang, X., Bate, I.: Using multi-parameters for calibration of low-cost sensors in urban environment. In: International Conference on Embedded Wireless Systems and Networks (EWSN) (2017)
13. Fang, X., Bate, I.: An improved sensor calibration with anomaly detection and removal. Sens. Actuators, B Chem. **307**, 127428 (2020)
14. Hayes, A.: Introduction To Mediation, Moderation, and Conditional Process Analysis: A Regression-Based Approach. Guilford Press, New York (2013)
15. Heimann, I., et al.: Source attribution of air pollution by spatial scale separation using high spatial density networks of low cost air quality sensors. Atmos. Environ. **113**, 10–19 (2015)
16. Keras: The Python deep learning library. https://keras.io (2017)
17. Klambauer, G., Unterthiner, T., Mayr, A., Hochreiter, S.: Self-normalizing neural networks. arXiv e-prints (2017)
18. Lewis, A., et al.: Evaluating the performance of low cost chemical sensors for air pollution research. Faraday Discuss. **189**, 85–103 (2016)
19. Maag, B., Saukh, O., Hasenfratz, D., Thiele, L.: Pre-deployment testing, augmentation and calibration of cross-sensitive sensors, pp. 169–180. ACM (2016)
20. Mueller, M., Meyer, J., Hueglin, C.: Design of an ozone and nitrogen dioxide sensor unit and its long-term operation within a sensor network in the city of Zurich. Atmos. Meas. Tech. **10**(10), 3783–3799 (2017)
21. Nair, V., Hinton, G.: Rectified linear units improve restricted Boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning, pp. 807–814 (2010)
22. Nielsen, M.A.: Neural Networks and Deep Learning. Determination Press (2015). http://neuralnetworksanddeeplearning.com/
23. Perkin Elmer: ELM sensor. https://elm.perkinelmer.com/map/ (2015)
24. Spinelle, L., Gerboles, M., Villani, M., Aleixandre, M., Bonavitacola, F.: Field calibration of a cluster of low-cost available sensors for air quality monitoring part a: ozone and nitrogen dioxide. Sens. Actuators, B Chem. **215**, 249–257 (2015)
25. Stephens, L.J.: Schaum's Outlines: Beginning Statistics, 2nd edn. McGraw-Hill, New York (2006)
26. Tensorflow: An open-source software library for machine intelligence. https://www.tensorflow.org (2017)