Different Learning Functions for Weighted Kernel Regression in Solving Small Sample Problem with Noise

Zuwairie Ibrahim, Nurul Wahidah Arshad
Faculty of Electrical and Electronic Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia

Mohd Ibrahim Shapiai
Malaysia-Japan International Institute of Technology Universiti Teknologi Malaysia, 81310 Johor Bahru, Malaysia

Norrima Mokhtar
Applied Control and Robotics (ACR) Laboratory, Department of Electrical Engineering, Faculty of Engineering, University of Malaya, 50603 Kuala Lumpur, Malaysia

E-mail: zuwairie@ump.edu.my

Abstract

Previously, weighted kernel regression (WKR) for solving small samples problem has been reported. In the original WKR, the simple iterative learning technique and the formulated learning function in estimating weight parameters are designed only to solve non-noisy and small training samples problem. In this study, an extension of WKR in solving noisy and small training samples is investigated. The objective of the investigation is to extend the capability and effectiveness of WKR when solving various problems. Therefore, four new learning functions are proposed for estimating weight parameters. In general, the formulated learning functions are added with a regularization term instead of error term only as in the existing WKR. However, one free parameter associated to the regularization term has firstly to be predefined. Hence, a simple cross-validation technique is introduced to estimate this free parameter value. The improvement, in terms of the prediction accuracy as compared to existing WKR is presented through a series of experiments.

Keywords: Learning functions, Small sample problem, Regression.

1. Introduction

In general, the kernel based regression aims at regressing the unknown function based on the available training samples. In real world applications, to obtain sufficient training samples is too expensive as when dangerous real measurements have to be performed [1]. There are numerous techniques in machine learning for regression. However, all the available techniques mainly focus in solving sufficient training samples problem. As most existing techniques perform well under sufficiently large training samples, the performance of those techniques degrades as the size of samples decreases.

Weighted Kernel Regression (WKR) [2] has proved to solve small sample problems with high accuracy by assuming all the available training samples are free from error. An example of regression using WKR is shown in Fig. 1. To design a WKR, one must estimate the weight parameters, *W*, before it can be used to predict unseen samples. The estimation of the weight parameters depends on the learning functions and learning techniques.

In general, the ability of WKR is only restricted to solve non-noisy training samples. Hence, the formulated learning function and simple iterative learning technique in WKR may fail in estimating weight parameters if the observed training samples are corrupted by noise. An example of regression using WKR in the presence of noise is shown in Fig. 2. Therefore, in this study, four learning functions are considered to particularly solve small and noisy samples problems.

2. Weighted Kernel Regression

Given training samples, $\{x_i, y_i\}_{i=1}^{n}$, where *n* is the number of training samples, input is denoted as $x_i \in \Re^d$, and $y_i \in \Re$ is the target output. WKR is the technique to regress the output space by mapping the input space $\Re^d$ to $\Re$. In general, WKR is a modified Nadaraya-Watson kernel regression (NWKR) [3] by expressing the weight based on the observed samples through a kernel function.

The existing WKR relies on the Gaussian kernel function as given in Eq. (1).

$$K(X,X_i)=\frac{1}{\sqrt{2\pi}}\exp\frac{\left(-\|X-X_i\|^2\right)}{h} \tag{1}$$

where h is the smoothing parameter. As in NWKR, the selection of smoothing parameter, h, is important to compromise between smoothness and fitness. As in existing WKR, Eq. (2) is employed to determine the value of h.

$$h=\max\left(\|X_{k+1}\|^2-\|X_k\|^2\right) \tag{2}$$

where $1 < k < n\text{-}1$ and $\|X_{k+1}\|^2 > \|X_k\|^2$.

The kernel matrix $K = [K_{ij}]$, where $i = j = 1,..., n$, with a generalised kernel matrix based on the Gaussian kernel, is given in Eq. (3). The matrix K transforms the linear observed samples to non-linear problems by mapping the data into a higher dimensional feature space.

$$K_{ij}=\begin{cases}\dfrac{\prod_{p=1}^{d}K\left(X_i^p,X_j^p\right)}{\sum_{l=1}^{n}\left[\prod_{p=1}^{d}K\left(X_{i\vee j}^p,X_j^p\right)\right]} & i\neq j\\ \dfrac{1}{\sum_{l=1}^{n}\left[\prod_{p=1}^{d}K\left(X_{i\vee j}^p,X_j^p\right)\right]} & i=j\end{cases} \tag{3}$$

In WKR, the most popular function for regression problems is used which to minimize the RSS to estimate the weight parameters, W.

$$\min\ f(W)\Leftrightarrow\min\|Kw-y\|^2 \tag{4}$$

Once the optimum weight is estimated, the model is ready to predict any unseen samples (test samples). The test samples can be predicted by using Eq. (5).

$$\hat{y}\left(X,\hat{W}\right)=\frac{\sum_{i=1}^{n}\hat{w}_i\left(\prod_{p=1}^{d}K\left(X^p,X_i^p\right)\right)}{\sum_{i=1}^{n}\left(\prod_{p=1}^{d}K\left(X^p,X_i^p\right)\right)} \tag{5}$$

3. Extension of Weighted Kernel Regression

In general, minimizing the error term only may lead to numerical instabilities and bad generalization performance. The instability yields a high variance model which potentially produces large differences of weight parameter values given different training samples, even minor perturbation of the same training samples. In general, this instability can be addressed by restricting the class of permissible solution by introducing the regularization term to the formulated learning function. The regularization term is usually of the form of a penalty term for complexity, such as restrictions for smoothness.

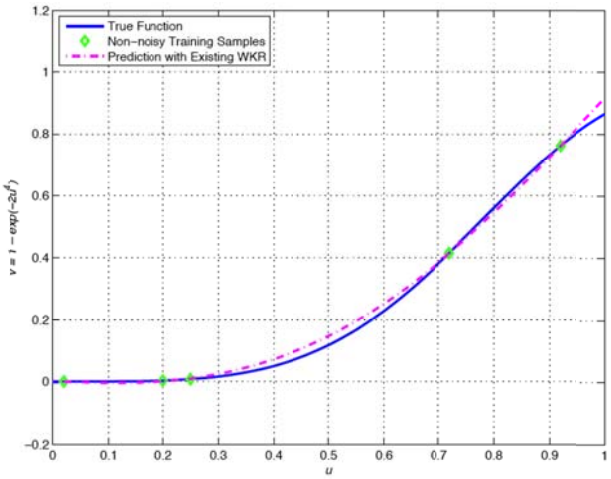

Fig 1. Regression using WKR

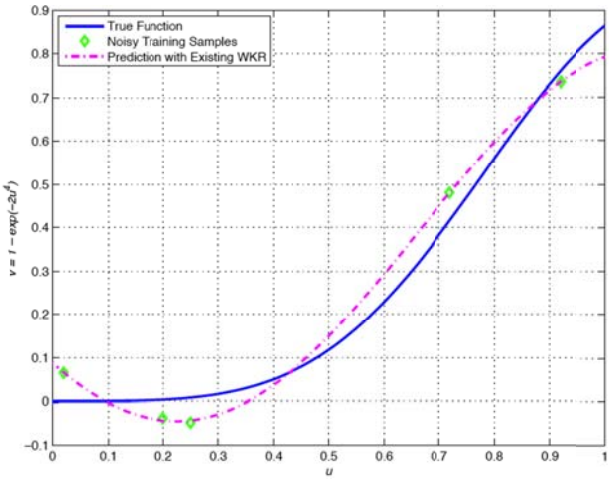

Fig 2. Regression using WKR in the presence of noise

Therefore, all the investigated learning functions will comprise not only the error term but also the regularization term as given in Eq. (6).

$$f_{learning}=f_e+f_r \tag{6}$$

where f_e refers to error term and f_r refers to regularization term.

The addition of regularization term is to avoid the magnitude of estimated weight parameters to be very large which may lead to over-fitting problem. Hence, the addition of regularization term gives advantages to the regression quality. In general, the error term and regularization term can be formulated either with L_1 norm or L_2 norm function. Therefore, the four learning functions are formulated with combination of L_1 and L_2 as error term and regularization term based on the WKR concept are proposed in this study.

Table 1. Parameter setting of GA

Generation	300
Population size	100
Probability of cross-over	0.7
Probability of mutation	0.001

The error term with L_2 norm is a popular choice and most widely used for linear model but it is less robust. Meanwhile, error term in L_1 norm form is not sensitive and produces robust estimation as compared to L_2 norm but L_1 norm function is not differentiable. For regularization term, the L_2 norm offers lower variances model by shrinking the estimated weight parameter values as compared to learning function without regularization term. The L_2 norm regularization term is also known as Tikhonov regularization [4] or Ridge regularization [5] for solving matrix inverse problem to learning problems with good generalization. L_1 norm regularization not only offers lower variances model but also produces sparseness solution which offers a better generalization. The sparseness solution forces some estimated weight parameter value equals to zero. Also, L_1 norm regularization term creates accurate predictive models that also have interpretable or parsimonious representations. The proposed formulated learning functions, which are based on learning function in the existing WKR, are given in Eq. (7) to Eq. (10).

$$f_{L_2R_2}(W) = \underset{W}{\operatorname{argmin}} \left(\|KW - Y\|^2 + \lambda \|W\|^2 \right) \tag{7}$$

$$f_{L_2R_1}(W) = \underset{W}{\operatorname{argmin}} \left(\|KW - Y\|^2 + \lambda \|W\|_1 \right) \tag{8}$$

$$f_{L_1R_2}(W) = \underset{W}{\operatorname{argmin}} \left(\|KW - Y\|_1 + \lambda \|W\|^2 \right) \tag{9}$$

$$f_{L_1R_1}(W) = \underset{W}{\operatorname{argmin}} \left(\|KW - Y\|_1 + \lambda \|W\|_1 \right) \tag{10}$$

where K is kernel matrix, W is weight parameter to be estimated, Y is observed output domain values, $\|.\|_1$ is L_1 norm function, $\|.\|^2$ is L_2 norm function, and λ is a free parameter that control the generalization of the regressed function.

In general, the formulated learning functions can be categorized into two types; closed form solution function and non-closed form solution function. Closed form solution function can be derived analytically as compared to non-closed form solution function when estimating the weight parameters. For non-closed form function, there is no analytical solution can be obtained as the function is nondifferentiable. As evolutionary computing offers an effective way to optimize i.e. estimate the weight parameters for non-differentiable function, hence, Genetic Algorithm (GA) is introduced as a learning technique. The formulated learning function with L_1 norm term either as error term or regularization term is considered as non-closed form solution function.

Prior to estimating the weight parameters based on new formulated learning functions, an associated λ value has firstly to be estimated. Cross-validation is a technique to evaluate model in order to generalize the predictive performance when predicting unseen samples. The need of cross-validation is important in model selection as some models parameters, such as λ value has to be estimated. In general, cross-validation separates the available training samples into two sets, called the training set and validation set. Training set is used to build the model and validation set is used to evaluate the model based on the selected models parameter with respect to the cross-validation error. Typically, the cross-validation error is measured based on MSE performance criterion. The model with the lowest cross-validation error is then used as a final model which possibly offers a better generalization performance.

There are various cross-validation techniques available in literatures such as hold-out method, K-fold cross-validation, and leave-one-out cross-validation (LOOCV). In general, LOOCV is very expensive to compute but it is able to retrieve a lot of information from the available training samples. As the focus of the study is to solve small and limited training samples problem, LOOCV is found to offer several advantages in terms of information retrieval and computational time.

In general, LOOCV separates the available n training samples into a training set of size n-1 and a validation of size 1. For every selected models parameter, there are n different combinations of training and validation set. The lowest cross-validation on the validation set is used as an indicator to select the final model.

4. Experiment, Result, and Discussion

Since the learning function which possesses L_1 norm term is considered as non-closed form solution function, an analytic form solution cannot be obtained when minimizing the corresponding learning function in estimating the weight parameter. This drawback has led to the use of GA. The parameter settings of GA are summarised in Table 1. A specific function is employed with three different Gaussian noise distributions, N~(0,0.1), N~(0,0.3), and N~(0,0.5).

The quality of prediction for every learning function for three different problems is tabulated in Table 2, Table 3, and Table 4. In general, the learning function with L_2 norm of error term offers a better generalization as compared to the learning function with L_1 norm of error term.

Table 2. Results of 100 experiments to predict 1- $exp(-2u)^4$ with $n = 5$ and contaminated by Gaussian noise, N~(0,0.1) with various learning functions

Learning Function	Average MSE	Standard Deviation	Min MSE	Max MSE
L_2R_2	0.00707	0.00517	0.00184	0.01834
L_2R_1	0.00704	0.00474	0.00221	0.01831
L_1R_2	0.01201	0.00926	0.00199	0.02916
L_2R_1	0.00930	0.00681	0.00181	0.02440

Table 3. Results of 100 experiments to predict 1- $exp(-2u)^4$ with $n = 5$ and contaminated by Gaussian noise, N~(0,0.3) with various learning functions

Learning Function	Average MSE	Standard Deviation	Min MSE	Max MSE
L_2R_2	0.04832	0.04122	0.01180	0.16416
L_2R_1	0.05048	0.04831	0.00630	0.17170
L_1R_2	0.06403	0.05314	0.00871	0.19740
L_2R_1	0.05940	0.05353	0.00862	0.19563

Table 4. Results of 100 experiments to predict 1- $exp(-2u)^4$ with $n = 5$ and contaminated by Gaussian noise, N~(0,0.5) with various learning functions

Learning Function	Average MSE	Standard Deviation	Min MSE	Max MSE
L_2R_2	0.11101	0.10993	0.03022	0.4474
L_2R_1	0.10767	0.11666	0.00402	0.45964
L_1R_2	0.14788	0.14201	0.02225	0.56058
L_2R_1	0.13753	0.13729	0.02376	0.56251

5. Conclusions

An extension of WKR is investigated to address regression problem with noisy training samples. The investigation emphasized on formulation of learning functions. Prior to these two investigations, the free parameter, λ value has firstly to be estimated. The improvement, in terms of quality of prediction is experimented and presented. In general, the selection of learning technique must be based on the formulated learning function, which implies the dependency of problem being solved. Also, the quality of prediction is mainly determined by the selection of L_2 norm as error term regardless of norm type of regularization term. Hence, the inclusion of regularization term is a must in formulating the learning function.

Acknowledgement

This work is financially supported by the RAGS Grant Scheme (RDU131416) awarded by the Ministry of Higher Education (MOHE) to Universiti Malaysia Pahang (UMP).

References

1. C. Huang and C. Moraga, A diffusion-neural-network for learning from small samples, *International Journal of Approximate Reasoning* **35**(2) (2004) 137-161.
2. M. I. Shapiai, Z. Ibrahim, M. Khalid, L. W. Jau, and V. Pavlovic, Enhanced Nadaraya Watson Kernel Regression: Surface Approximation for Extremely Small Samples, *The 5th Asia Modeling Symposium* (2011) 7-12.
3. E. A. Nadaraya, On Estimating Regression, *Theory of Probability and its Applications* **9**(1) (1964) 141-142.
4. A. Tikhonov and V. Arsenin, *Solutions of Ill-posed Problems* (1977).
5. A. E. Hoerl and R. W. Kennard, Ridge Regression: Biased Estimation for Nonorthogonal Problems, *Technometrics* **12**(1) (1970) 55-67.