Selection of important input variables for RBF network using partial derivatives

Jarkko Tikka and Jaakko Hollmén

Helsinki University of Technology
Department of Information and Computer Science
P.O. Box 5400, FI-02015 HUT, Finland
tikka@mail.cis.hut.fi, http://www.cis.hut.fi/tikka/

Abstract. In regression problems, making accurate predictions is often the primary goal. Also, relevance of inputs in the prediction of an output would be valuable information in many cases. A sequential input selection algorithm for Radial basis function (SISAL-RBF) networks is presented to analyze importances of the inputs. The ranking of inputs is based on values, which are evaluated from the partial derivatives of the network. The proposed method is applied to benchmark data sets. It yields accurate prediction models, which are parsimonious in terms of the input variables.

1 Introduction

The goal of a regression problem is to learn an input-output relationship from data. Dependencies between the inputs and the output are typically nonlinear, and the exact functional form is unknown. Neural networks are widely utilized in regression problems, since they are relatively fast to train [11] and capable to approximate a wide class of functions accurately [5]. The disadvantage of neural networks is, that they include all the input variables and importances of inputs are unclear. We propose a backward input selection algorithm for RBF networks. The inputs are dropped one at a time from the model based on the ranking calculated from the partial derivatives. The resulting subsets of inputs are assessed using leave-one-out (LOO) error. The rejection of unimportant inputs increases the interpretability of the network, it may improve the generalization capability, and it also decreases the computational complexity of the final network [2]. The proposed algorithm can be seen as a wrapper input selection method [3]. Another approaches are filter [13] and embedded methods [12].

2 Radial basis function networks

Let us assume that we have N measurements from an output y_j and d inputs $\boldsymbol{x}_j = [x_{j1}, \ldots, x_{jd}],\ j = 1, \ldots, N$. The output of RBF network with a Gaussian basis functions is

$$\hat{y}_j = \sum_{m=1}^{M} \alpha_m K(\boldsymbol{c}_m, \boldsymbol{x}_j) + \alpha_0, \text{ where } K(\boldsymbol{c}_m, \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{c}_m - \boldsymbol{x}_j\|^2}{\sigma_m^2}\right), \quad (1)$$

and M, $\boldsymbol{c}_m$, and σ_m are the number, the centers, and the widths of the basis functions [4], respectively. The model can also be written in the matrix form as

$\hat{\boldsymbol{y}} = \boldsymbol{K}\boldsymbol{\alpha}$, where the elements of matrix $\boldsymbol{K}$ are defined as $\boldsymbol{K}_{jm} = K(\boldsymbol{c}_m, \boldsymbol{x}_j)$ and the $(M+1)^{\text{th}}$ column is the vector of ones corresponding to the bias term α_0.

We place the Gaussian basis function on each training data point $\boldsymbol{x}_j$ and set the widths of the basis functions to an equal value $\sigma_m = \sigma$. The parameters $\boldsymbol{\alpha}$ are estimated by minimizing the regularized mean squared error (MSE)

$$J = \frac{1}{N}\sum_{j=1}^{N}(y_j - \hat{y}_j)^2 + \gamma \sum_{m=0}^{N} \alpha_m^2 \ , \tag{2}$$

where the second term controls smoothness of the nonlinear mapping.

The generalization capability of the model is measured using the LOO error. For the fixed value of the width σ, the LOO error is the function of the regularization parameter γ

$$\text{MSE}_{\text{LOO}}(\gamma) = \frac{1}{N}\boldsymbol{y}^T \boldsymbol{P}\left(\text{diag}(\boldsymbol{P})\right)^{-2}\boldsymbol{P}\boldsymbol{y} \ , \tag{3}$$

where $\boldsymbol{P} = \boldsymbol{I}_N - \boldsymbol{K}(\boldsymbol{K}^T\boldsymbol{K} + \gamma \boldsymbol{I}_m)^{-1}\boldsymbol{K}^T$ and $\text{diag}(\boldsymbol{P})$ is of the same size and has the same diagonal as $\boldsymbol{P}$ but is zero off the diagonal [4]. In the optimization of γ, we use the golden section line search method [1]. The parameters $\boldsymbol{\alpha}$ are found by minimizing Eq. (2) using the parameters (σ, γ), which minimize Eq. (3).

3 Input variable selection algorithm

We propose a relevance measure to rank importance of each input variable in the model in Eq. (1). Relevance of the inputs $x_i,\ i = 1, \ldots, d$, can be measured using the partial derivatives of the output $\hat{y}$ with respect to x_i [10, 9, 8]. The derivatives of the most relevant inputs vary most through the range of input values. The partial derivative of the RBF network with respect to x_i is

$$d_{ji} = \frac{\partial \hat{y}_j}{\partial x_{ji}} = \frac{2}{\sigma^2}\sum_{m=1}^{M} \alpha_m K(\boldsymbol{c}_m, \boldsymbol{x}_j)(c_{mi} - x_{ji}),\ i = 1, \ldots, d,\ j = 1, \ldots, N \ . \tag{4}$$

We use an Add10 data set as an example. It includes inputs $x_i,\ i = 1, \ldots, 10$, which are sampled independently from an uniform distribution $\mathcal{U}(0, 1)$. The output is $y = 10\sin(2x_1x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 + \varepsilon$, where ε is the Gaussian noise with zero mean and unit variance. All the variables were scaled to have zero mean and unit variance before training of a RBF network using $N = 250$ samples. On the first column of Fig. 1, the partial derivatives of the RBF network with respect to the inputs x_3 and x_4 are shown. The values d_{j4} are almost constant, thus the dependency between x_4 and $\hat{y}$ is linear. The dependency between $\hat{y}$ and x_3 is quadratic, since x_3 and d_{j3} are linearly dependent.

The median of the values d_{j3} is nearly zero, because of the cancellations between negative and positive values. Thus, the absolute values $|d_{j3}|$ and $|d_{j4}|$ might be more representative, which are shown as a function of x_{j3} and x_{j4} on the second column of Fig. 1. The histograms of $|d_{j3}|$ and $|d_{j4}|$ are presented on

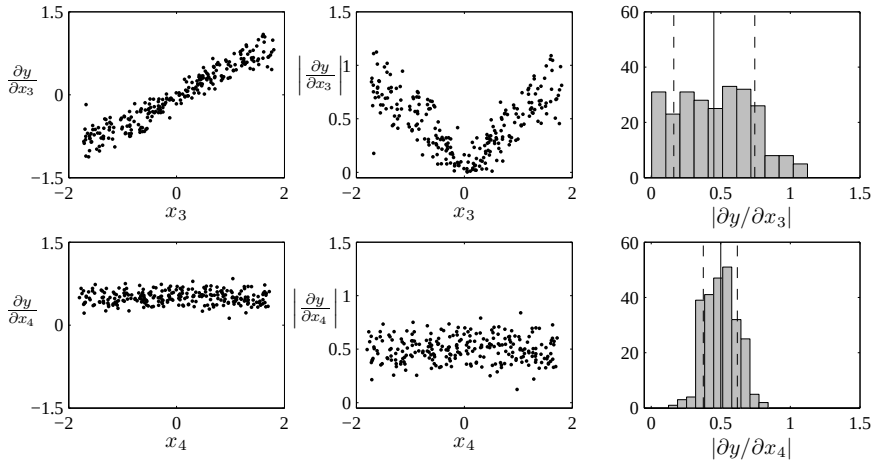

Fig. 1: Scatter plots of partial derivatives d_{j3} and d_{j4} (*left*) and absolute values $|d_{j3}|$ and $|d_{j4}|$ (*middle*) as a function of x_{j3} and x_{j4}, the histograms of the absolute values $|d_{j3}|$ and $|d_{j4}|$ (*right*), the solid lines are the medians and between the dashed lines is the central 67% quantile (*right*).

the third column. However, the medians of $|d_{j3}|$ and $|d_{j4}|$ are nearly equal and the relevance of the variables cannot be distinguished based on them. Thus, we propose to define a relevance measure of the input x_i as follows

$$r_i = m_{d_i} + \Delta_{d_i} \ , \tag{5}$$

where m_{d_i} is the median of the absolute values $|d_{ji}|,\ j = 1, \ldots, N$. The second term Δ_{d_i} measures the variablity of the values $|d_{ji}|$. It is defined as a difference $\Delta_{d_i} = |d_{ji}|^{high} - |d_{ji}|^{low}$, where $|d_{ji}|^{high}$ and $|d_{ji}|^{low}$ are the $0.835N^{\text{th}}$ and $0.165N^{\text{th}}$ values in the ordered list of the N absolute values $|d_{ji}|$. With previous choices, the difference Δ_{d_i} is twice as large as the standard deviation in the case of the normal distribution. The larger the variation Δ_{d_i} the more sensitive the output is to the corresponding input. The relevant inputs should also have clearly nonzero medians m_{d_i}. Thus, the most relevant input has the highest value for the relevance measure r_i. Both the median m_{d_i} and the difference Δ_{d_i} are insensitive to the outliers in the data. Here, we use equal weighting for the two terms in (5), but unequal weighting could be used as well. Other relevance measures based on the partial derivatives are presented in [10, 14, 6].

We propose a backward input selection algorithm based on the relevance measure r_i. The algorithm starts by evaluating the RBF network with all the available input variables $x_i,\ i = 1, \ldots, d$. The hyperparameters σ^2 and γ are selected by minimizing the LOO error. After that, the parameters $\boldsymbol{\alpha}$ are obtained as a solution to a system of linear equations in minimization of Eq. (2). The next step is to delete the least relevant input x_i, which is the input having the smallest value for the relevance measure r_i. The previous steps are repeated using the

Algorithm 1 SISAL-RBF

1: Let $\mathcal{L}$ be the set of the inputs $x_i,\ i = 1, \ldots, d$

2: Minimize the LOO error using the inputs in $\mathcal{L}$
- width of the basis functions σ fixed
- optimize regularization parameter γ by minimizing (3)

3: Repeat step 2. with various values of σ. Select the pair $(\sigma_{\mathcal{L}}, \gamma_{\mathcal{L}})$, which minimize Eq. (3)

4: Use the pair $(\sigma_{\mathcal{L}}, \gamma_{\mathcal{L}})$, minimize the function in Eq. (2) with respect to $\boldsymbol{\alpha}$

5: Evaluate relative importances of the inputs $r_i,\ i \in \mathcal{L}$

6: Delete the input x_i, which has the smallest value for the relevance measure r_i, from the set of inputs $\mathcal{L}$

7: If $\mathcal{L} \neq \varnothing$ go to step 2, otherwise go to step 8

8: Select the set of inputs $\mathcal{L}_v$, which gives the smallest value for the LOO error

Name	Training (N_t)	Test (N_{test})	inputs	range of σ^2
Add10[1]	250	9542	10	$[1, 500]$
Bank[1]	500	7692	32	$[1, 10^4]$
Boston housing[1]	400	106	13	$[1, 500]$
Wine[2]	94	30	256	$[5, 10^6]$

Table 1: Properties of the data sets.

remaining inputs, which results to the evaluation of d subsets of inputs. The final set of inputs minimize the LOO error. The sequential input selection algorithm for the RBF network (SISAL-RBF) is summarized in detail in Algorithm 1.

4 Experiments

SISAL-RBF was applied to four benchmark data sets (Add10, Bank, Boston housing, and Wine). In the case of Add10 data, the assessment of input selection results is straightforward, since the correct inputs are known. The data sets were randomly divided to the training and test sets. The sample sizes and the number of inputs are reported in Table 1. LOO errors were evaluated using 50 values of σ^2, which were equally spaced on a logarithmic scale in the ranges shown in Table 1. All the inputs and the outputs were scaled to have zero mean and unit variance to make the relevance measures comparable.

A forward selection (FS) algorithm was used as a baseline method to compare the performance of the proposed input selection strategy, since it is known that FS is robust against overfitting [7]. In the case of d inputs, $(d + 1)d/2$ subsets of inputs have to be evaluated. FS could be stopped before all the inputs are

[1] Available from: http://www.cs.toronto.edu/~delve/data/datasets.html
[2] Available from: http://www.dice.ucl.ac.be/mlg/index.php?page=DataBases

Data set	Ordinary RBF	SISAL-RBF	RBF with FS
Add10	0.170 | 0.164 | 10	0.069 | 0.060 | 5	0.069 | 0.060 | 5
Bank	0.261 | 0.245 | 32	0.217 | 0.222 | 10	0.211 | 0.229 | 12
Boston housing	0.110 | 0.148 | 13	0.104 | 0.133 | 9	0.093 | 0.127 | 10
Wine	0.004 | 0.014 | 256	0.002 | 0.004 | 39	0.001 | 0.003 | 36

Table 2: LOO error (the 1st value), MSE for the test set (the 2nd value), and the number of selected inputs (the 3rd value).

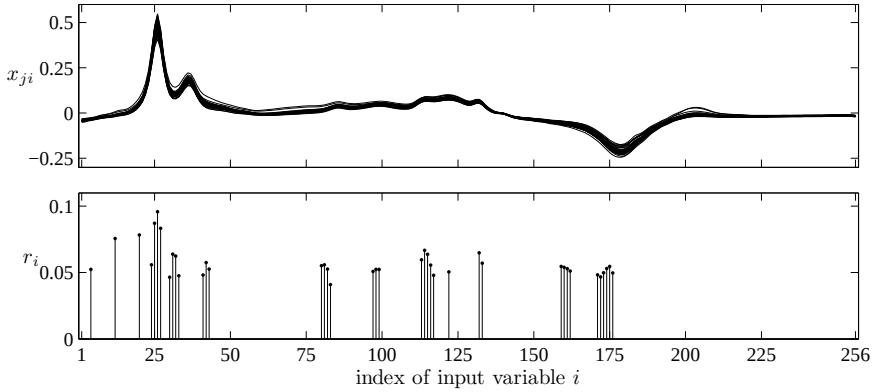

Fig. 2: The input vectors $\boldsymbol{x}_j$ in the test set of the Wine data (*top*), the selected variables and their relative importances (*bottom*).

selected, but that point might not minimize the estimate of generalization error. The prediction accuracies are also reported for the ordinary RBF networks, i.e. for the networks including all the inputs.

The prediction errors of the models are reported in Table 2. The ordinary RBF networks have the highest MSEs. In practice, there are no differences between the errors of SISAL-RBF and FS. Both algorithms found all the five correct inputs in the case of Add10 data. In the cases of Bank, Boston housing, and Wine data, the numbers of the same inputs were ten, nine, and thirteen, respectively. Most of the selected inputs were different in the case of Wine data. However, they are highly correlated, and thus they contain similar information. The input vectors $\boldsymbol{x}_j$ of Wine data in the test set are shown in the top panel of Fig. 2. The inputs x_i selected by SISAL-RBF and their relative importances are illustrated in the bottom panel. Only 14% of the available inputs are included to the final network.

5 Summary and conclusions

The sequential input selection algorithm for RBF networks was presented. The relevance of input variables were measured using the partial derivatives of the

network. The final subset of inputs was selected based on the minimum LOO error. The advantage of the LOO error is that the regularization parameter can be optimized using a line search method, which does not restrict the evaluation of error to the predefined values. It was proposed to place the basis function on each training data point, which is infeasible with large data sets. In such a case, the locations of basis functions could be selected, for example, using some unsupervised clustering technique. After that, the centers would be kept fixed and SISAL-RBF could be applied as it was presented.

Experiments showed that SISAL-RBF was competitive in comparison with FS in terms of prediction accuracy and selected inputs. Nevertheless, the computational complexity of SISAL-RBF is linear with respect to number of input variables, whereas the complexity of FS is quadratic. The networks constructed using SISAL-RBF were also more accurate than the networks using all the inputs, which indicates that the most important variables were detected by SISAL-RBF.

References

[1] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, Nonlinear Programming Theory and Algorithms. John Wiley & Sons, 1979.

[2] I. Guyon and A. Elisseeff, An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157-1182, The MIT Press, 2003.

[3] R. Kohavi and G. H. John, Wrappers for feature subset selection. *Artificial Intelligence*, 97:273–324, Elsevier, 1997.

[4] M. J. Orr, Introduction to Radial Basis Functions Networks. Technical Report, Edinburgh University, Edinburgh, Scotland, UK, April 1996.

[5] J. Park and I. W. Sandberg, Approximation and Radial-Basis-Function Networks. *Neural Computation*, 5:305-316, The MIT Press, 1993.

[6] A.-P. N. Refenes and A. D. Zapranis, Neural model identification, variable selection and model adequacy. *Journal of Forecasting*, 18:299-332, John Wiley & Sons, 1999.

[7] J. Reunanen, Overfitting in making comparisons between variable selection methods, *Journal of Machine Learning Research* 3:1371–1382, The MIT Press, 2003.

[8] M. La Rocca and C. Perna, Variable selection in neural network regression models with dependent data: a subsampling approach. *Computational Statistics & Data Analysis*, 48:415-429, Elsevier, 2005

[9] F. Rossi, Attribute suppression with multi-layer perceptron. In *Proceedings of CESA Multiconference, volume Symposium on Robotics and Cybernetics*, IMACS, pages 542-547, 1996

[10] D.W. Ruck, S.K. Rogers, and M. Kabrisky, Feature selection using a multilayer perceptron, *Journal of Neural Network Computing*, 2:40–48, Auerbach Publishers, 1990.

[11] D.F. Specht, A general regression neural network. *Transactions on Neural Networks*, 2:568–576, IEEE, 1991.

[12] J. Tikka, Input selection for radial basis function networks by constrained optimization. In Proceedings of the 17^{th} *International Conference on Artificial Neural Networks* (ICANN 2007), Lecture Notes in Computer Science 4668, pages 239-248, Springer-Verlag, 2007.

[13] J. Tikka and J. Hollmén, Sequential input selection algorithm for long-term prediction of time series. *Neurocomputing*, In press, Elsevier.

[14] J. M. Zurada, A. Malinowski, and S. Usui, Perturbation method for deleting redundant inputs of perceptron networks. *Neurocomputing*, 14:177-193, Elsevier, 1997.