Minimal Learning Machine: A New Distance-Based Method for Supervised Learning

Amauri Holanda de Souza Junior[1,⋆], Francesco Corona[3], Yoan Miche[3], Amaury Lendasse[3], Guilherme A. Barreto[2], and Olli Simula[3]

[1] Federal Institute of Ceará, Department of Computer Science
Av. Contorno Norte, 10 - Maracanaú, Ceará, Brazil
[2] Federal University of Ceará, Department of Teleinformatics Engineering
Av. Mister Hull, S/N - Campus of Pici, Center of Technology, Fortaleza, Ceará, Brazil
[3] Aalto University, Department of Information and Computer Science
Konemiehentie 2, Espoo, Finland

Abstract. In this work, a novel supervised learning method, the Minimal Learning Machine (MLM), is proposed. Learning a MLM consists in reconstructing the mapping existing between input and output distance matrices and then estimating the response from the geometrical configuration of the output points. Given its general formulation, the Minimal Learning Machine is inherently capable to operate on nonlinear regression problems as well as on multidimensional response spaces. In addition, an intuitive extension of the MLM is proposed to deal with classification problems. On the basis of our experiments, the Minimal Learning Machine is able to achieve accuracies that are comparable to many *de facto* standard methods for regression and it offers a computationally valid alternative to such approaches.

1 Introduction

In this paper, we present a new supervised method, the Minimal Learning Machine (MLM). The basic idea behind the Minimal Learning Machine is the existence of a mapping between the geometric configurations of points in the input and output space. On the basis of our experiments, such a mapping can be accurately reconstructed by learning a multi-response linear regression model between distance matrices. Under these conditions, for an input point with known configuration in the input space, its corresponding configuration in the output space can be easily estimated after learning a simple linear model between input and output distance matrices. The resulting estimate is then used to locate the output point and thus provide an estimate for the response. In its basic formulation, the MLM closely resembles a classical unsupervised dimensionality reduction method, Multidimensional Scaling (MDS, [1]), and more specifically its variant known as Landmark MDS [2], the main difference being that the output configuration in MLM is known beforehand.

⋆ The author would like to thank the financial support received from the Brazilian Agency of Post-Graduate Studies (CAPES) under the grant number 9147-12-8.

I. Rojas, G. Joya, and J. Cabestany (Eds.): IWANN 2013, Part I, LNCS 7902, pp. 408–416, 2013.

The remainder of the paper is organized as follows. In Section 2, the Minimal Learning Machine is presented; the MLM is formulated (Section 2.1), its properties discussed (Section 2.2) and two illustrative examples presented (Section 2.3) along with a simple extension of MLM that renders it suitable also for classification tasks. In Section 3, a thorough experimental assessment of the Minimal Learning Machine is conducted to evaluate its performance and to compare it with state-of-the-art approaches in regression.

2 Minimal Learning Machine

We are given a set of N input points $X = \{\mathbf{x}_i\}_{i=1}^N$, with $\mathbf{x}_i \in \mathbb{R}^D$, and the set of their corresponding outputs $Y = \{\mathbf{y}_i\}_{i=1}^N$, with $\mathbf{y}_i \in \mathbb{R}^S$. Assuming the existence of a continuous mapping $f : \mathcal{X} \to \mathcal{Y}$ between the input and the output space, we want to estimate it from data using a multi-response model

$$\mathbf{Y} = f(\mathbf{X}) + \mathbf{R},$$

where the columns of the matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_D]$ correspond to the input variables and the rows to the observations, analogously the columns of the matrix $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_S]$ correspond to the output variables and the rows to the observations. The $N \times S$ matrix $\mathbf{R}$ denotes the output residual vectors.

2.1 Formulation

Provided that the input space $\mathcal{X}$ is well sampled and f is smooth, we expect that for each pair of input points $(\mathbf{x}_i, \mathbf{x}_j)$ and for every $\varepsilon_y > 0$, there exists a $\varepsilon_x > 0$ such that for $d(\mathbf{x}_i, \mathbf{x}_j) < \varepsilon_x$ we have that $\delta(f(\mathbf{x}_i)), \delta(f(\mathbf{x}_j)) < \varepsilon_y$, where $d(\cdot, \cdot)$ and $\delta(\cdot, \cdot)$ are distance functions in $\mathcal{X}$ and $\mathcal{Y}$, respectively. Under this condition, we are interested in reconstructing the mapping $g : \mathcal{D}_\mathcal{X} \to \mathcal{D}_\mathcal{Y}$ between input distance matrices $\mathbf{D}_x$ and the corresponding output distance matrices $\boldsymbol{\Delta}_y$. The availability of the geometrical configurations of the points in the input and in the output space is then used to estimate the response $\mathbf{y}$ of a query input $\mathbf{x}$.

Distance Regression. For a selection of reference input points $R = \{\mathbf{m}_k\}_{k=1}^K$ with $R \subseteq X$ and corresponding outputs $T = \{\mathbf{t}_k\}_{k=1}^K$ with $T \subseteq Y$, we define $\mathbf{D}_x \in \mathbb{R}^{N \times K}$ in such a way that its kth column contains the distances $d(\mathbf{x}_i, \mathbf{m}_k)$ between the $i = 1, \ldots, N$ input points $\mathbf{x}_i$ and the kth reference point $\mathbf{m}_k$. Analogously, we define $\boldsymbol{\Delta}_y \in \mathbb{R}^{N \times K}$ in a way that its kth column contains the distances $\delta(\mathbf{y}_i, \mathbf{t}_k)$ between the output points $\mathbf{y}_i$ and the output $\mathbf{t}_k$ of the kth reference point. The associated multi-response regression model for estimating g is thus

$$\boldsymbol{\Delta}_y = g(\mathbf{D}_x) + \mathbf{E}, \tag{1}$$

where the columns of the matrix $\mathbf{D}_x$ correspond to the K input vectors and the columns of the matrix $\boldsymbol{\Delta}_y$ correspond to the K response vectors. As usual, the K columns of the $N \times K$ matrix $\mathbf{E}$ correspond to the residuals.

We assume that the mapping g between input and output distances is linear, thus the multi-response regression model between distance matrices becomes

$$\boldsymbol{\Delta}_y = \mathbf{D}_x\mathbf{B} + \mathbf{E}, \tag{2}$$

where the regression matrix $\mathbf{B} \in \mathbb{R}^{K\times K}$ has to be solved from data. Under the normal conditions where the number of equations in (2) is larger to the number of unknowns, the problem is overdetermined and, usually, with no solution. This corresponds to the case where the number of selected reference points is smaller than the number of points available for solving the model (i.e., $K < N$) and we have to rely on the approximate solution provided by the least squares estimate:

$$\hat{\mathbf{B}} = (\mathbf{D}'_x\mathbf{D}_x)^{-1}\mathbf{D}'_x\boldsymbol{\Delta}_y. \tag{3}$$

On the other hand, if in (2) the number of equations equals the number of unknowns (i.e., all the learning points are also selected as reference points and $K = N$), the problem is uniquely determined and, usually, with a single solution $\hat{\mathbf{B}} = (\mathbf{D}_x)^{-1}\boldsymbol{\Delta}_y$. Clearly less interesting is the case where in (2) the number of equations is smaller than then number of unknowns (i.e., for $K > N$, corresponding to the situation where, after selecting the reference points, only a smaller number of learning points is used), for it leads to an underdetermined problem with, usually, infinitely many solutions.

Given the possibility for $\mathbf{B}$ to be either uniquely solvable or estimated (Equation 3), for a test point $\mathbf{x} \in \mathbb{R}^D$ whose distances from the K reference input points $\{\mathbf{m}_k\}_{k=1}^K$ are collected in the vector $\mathbf{d}(\mathbf{x}, R) = [d(\mathbf{x}, \mathbf{m}_1), \ldots, d(\mathbf{x}, \mathbf{m}_K)]$, the corresponding distances between its unknown output $\mathbf{y}$ and the known outputs of the reference points, the vector $\boldsymbol{\delta}(\mathbf{y}, T) = [\delta(\mathbf{y}, \mathbf{t}_1), \ldots, \delta(\mathbf{y}, \mathbf{t}_K)]$, are

$$\hat{\boldsymbol{\delta}}(\mathbf{y}, T) = \mathbf{d}(\mathbf{x}, R)\hat{\mathbf{B}}. \tag{4}$$

The vector $\hat{\boldsymbol{\delta}}(\mathbf{y}, T)$ provides an estimate of the geometrical configuration in $\mathcal{D}_\mathcal{Y}$ of $\mathbf{y}$ with respect to all the reference points $\{\mathbf{t}_k\}_{k=1}^K$ and thus can be used to estimate its location in $\mathcal{Y}$.

Output Estimation. Estimating $\mathbf{y}$ is equivalent to solve the overdetermined set of nonlinear equations corresponding to the K $(S+1)$-dimensional hyperspheres centered in $\mathbf{t}_k$ and all passing through $\mathbf{y}$, that is with a radius equal to $\hat{\delta}(\mathbf{y}, \mathbf{t}_k)$:

$$(\mathbf{y} - \mathbf{t}_k)'(\mathbf{y} - \mathbf{t}_k) = \hat{\delta}^2(\mathbf{y}, \mathbf{t}_k), \quad \forall k = 1, \ldots, K. \tag{5}$$

The problem in Equation 5 can be formulated as an optimization problem where an estimate $\hat{\mathbf{y}}$ can be obtained by the following minimization:

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmin}} \sum_{k=1}^{K} \Big((\mathbf{y} - \mathbf{t}_k)'(\mathbf{y} - \mathbf{t}_k) - \hat{\delta}^2(\mathbf{y}, \mathbf{t}_k) \Big)^2. \tag{6}$$

The objective has a minimum equal to 0 that can be achieved if and only if $\mathbf{y}$ is the solution of Equation 5. If it exists, such a solution is global and unique.

Due to the uncertainty introduced by the estimates $\hat{\delta}(\mathbf{y}, \mathbf{t}_k)$, an optimal solution to Equation 6 can still be achieved using gradient descent methods or the Levenberg-Marquardt algorithm. This method is used in our experiments.

2.2 Parameters and Computational Complexity

On the basis of the aforementioned overview, the number of reference points K is virtually the only hyper-parameter that the user needs to select in order to optimize a Minimal Learning Machine. As always, a selection based on standard resampling methods for cross-validation could be adopted for the task and thus optimize the MLM against over-fitting. Two figures of merit can be used for selecting K; one for the distance regression step and another one for the output estimation. In this work, we use the Average Mean Squared Error for the output distances ($AMSE(\boldsymbol{\delta}) = 1/K \sum_{k=1}^{K} (1/N \sum_{i=1}^{N} (\delta(y_i, t_k) - \hat{\delta}(y_i, t_k))^2)$) and the Mean Squared Error for the responses ($MSE(y) = 1/N \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$).

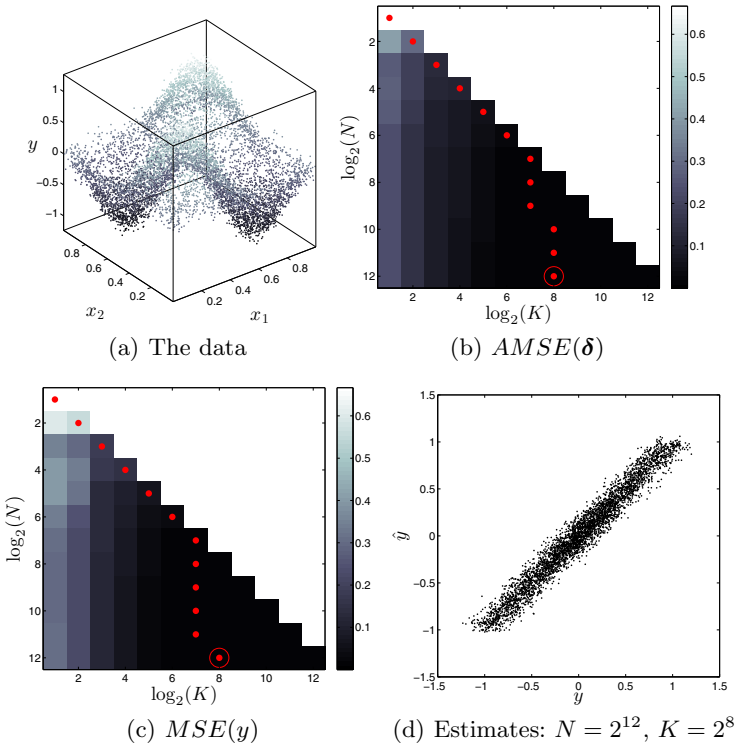
The computation for learning a MLM can be decomposed into two steps: i) calculations of the pairwise distance matrices in the output and input space and ii) calculation of the least-square solution for the multi-response linear regression problem on distance matrices (Equation 3). The first procedure takes $\Theta(KN)$ time. The computational cost of the second step is driven by the calculation of the Moore-Penrose pseudo-inverse matrix. One of the most used method for the task is the Singular Value Decomposition, which runs in $\Theta(K^2N)$ time. The time complexity of the overall learning phase is thus driven by the computation of the Moore-Penrose matrix and then it is given by $\Theta(K^2N)$. However, because the optimal number of reference points might not grow at the same rate of the number of learning points, then such complexity can be reduced to $\mathcal{O}(N)$ if one considers $K = \mathcal{O}(1)$, or $\mathcal{O}(N^2)$ if $K = \mathcal{O}(N^{0.5})$. In addition, large pairwise distance matrices could be approximated using Nyström methods and matrix multiplication operations could be parallelized using multicore architectures.

2.3 Two Illustrative Examples

In this section, we illustrate the effectiveness of the Minimal Learning Machine using two synthetic problems. The first one is related to nonlinear regression (the smoothed parity function) and the second one (the Tai Chi) is used to introduce an intuitive extension that allows the MLM to deal with classification problems.

The Smoothed Parity. To illustrate the behavior of the Minimal Learning Machine for regression, we generated 2^{13} bidimensional input points uniformly distributed in the unit-square, $\mathbf{x} \in [0, 1]^2$, and built the response using the model $y = f(x) + \varepsilon$ with $f = \sin(2\pi x_1)\sin(2\pi x_2)$ and $\varepsilon \sim \mathcal{N}(0, 0.1^2)$, Figure 1(a).

We analyzed the performance of the MLM for N learning points ranging from 2^1 to 2^{12} and K reference points such that always $K \leq N$. A common set of $N_v = 2^{12}$ independent points is used for validating the MLM in terms of its



(a) The data (b) $AMSE(\boldsymbol{\delta})$

(c) $MSE(y)$ (d) Estimates: $N = 2^{12}$, $K = 2^8$

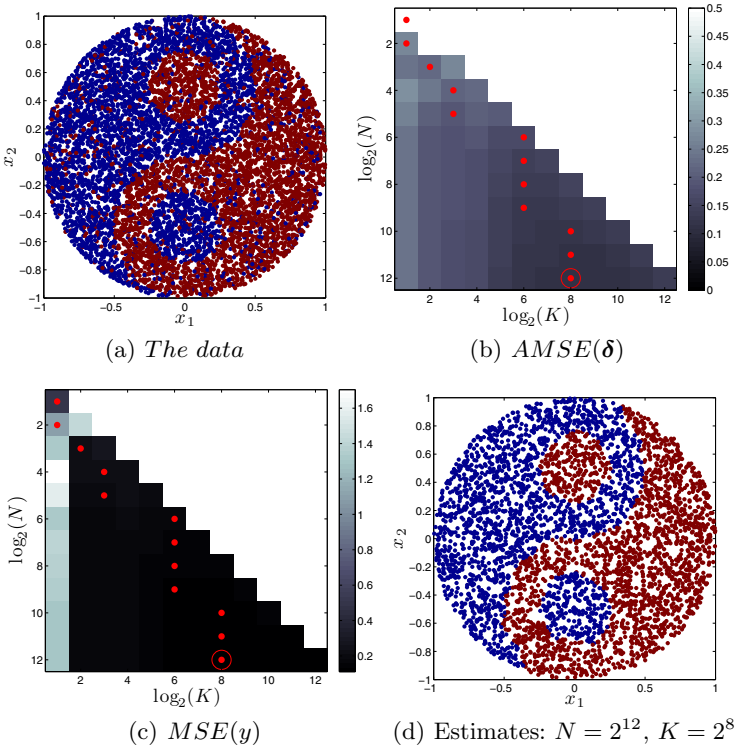
Fig. 1. A regression example: The *smoothed* parity function

hyper-parameter K. Two figures of merit are considered for selecting K, the $AMSE(\boldsymbol{\delta})$ for the output distances and the $MSE(y)$ for the response.

As expected, for each size of the learning set it is possible to select an optimal number of reference points that minimizes the validation error. Figure 1(b) and 1(c) depicts such optimal models with red dots. In these two figures, the circle is used to depict the best model overall ($N = 2^{12}$ and $K = 2^8$, for both $AMSE(\boldsymbol{\delta})$ and $MSE(y)$). Figure 1(d) illustrates the validation results when estimating the response with such a model. Interestingly, the MSE achieved by this MLM is 0.011, which tends to the variance of the noise (0.010) and thus also to the smallest MSE that any regression model can achieve without over-fitting.

The Tai Chi. Since the Minimal Learning Machine is able to deal with multidimensional response spaces, it can be easily extended to multi-class classification problems by representing the classes through binary output encoding schemes.

For compactness, here we illustrate the behavior of the MLM for binary classification. We generated 2^{13} bidimensional input points uniformly distributed in the Tai Chi symbol and, after assigning the class labels to the Yin an Yang areas, we purposely mislabeled 10% of the observations, Figure 2(a). A binary output



(a) *The data* (b) $AMSE(\boldsymbol{\delta})$

(c) $MSE(y)$ (d) Estimates: $N = 2^{12}$, $K = 2^8$

Fig. 2. A classification example: The Tai Chi

encoding that assigns $y_i = 0$ to the points in the class 'Yin' and $y_i = 1$ to those in the class 'Yang' is used. The output distances are calculated accordingly.

The performance of the MLM for classification with N and K ranging from 2^1 to 2^{12} and $K \leq N$ is presented. For validation purposes, a set of 2^{12} observations is used. The results in terms of $AMSE(\boldsymbol{\delta})$ and $MSE(y)$ are reported in Figure 2(b) and 2(c), respectively. The red dots denote the best model per number of learning points and the circle depicts the best model overall. Figure 2(d) shows the estimated classes in validation using the best model; the accuracy is 88%.

3 Experiments

In this section, we present results obtained with six real-world regression datasets used for benchmarking purposes (UCI Repository: `www.ics.uci.edu/~mlearn/`). The datasets have been chosen to object heterogeneity in the number of samples and inputs: 1) Breast Cancer (32 inputs, 194 samples); 2) Boston Housing (13 inputs, 506 samples); 3) Servo (4 inputs, 167 samples); 4) Abalone (8 inputs, 4177 samples); 5) Stocks (9 inputs, 950 samples); and 6) Auto Price (15 inputs, 159 samples). For each problem, ten different random permutations of the whole dataset

are taken, two thirds are used for learning and the rest for testing. The learning sets are normalized to have zero mean and unit variance, and the test sets are normalized using the corresponding mean and variance from the learning set.

The Minimal Learning Machine is compared to five other methods: The Extreme Learning Machine (ELM, [3]), the Optimally Pruned ELM (OP-ELM, [4]), the Support Vector Machine for Regression (SVM, [5]), Gaussian Processes (GP, [6]) and the MultiLayer Perceptron (MLP, [7]).

The hyper-parameters for the SVM and the MLP are selected using 10-fold cross-validation. The SVM is learned using the SVM toolbox [8] with default settings for the hyper-parameters and grid search, with a radial basis kernel. The MLP is optimized using Levenberg-Marquardt and validated on a range of hidden units from 1 to 20. The learning of GP is based on the default settings in the Matlab Toolbox [6]. The ELM and OP-ELM have been validated using sigmoid, gaussian and linear kernels, and a maximum number of 100 hidden units. The only hyper-parameter of the Minimal Learning Machine (the number of reference points) has also been selected through 10-fold cross-validation, for a K ranging from 5% to 100% (with a step size of 5%) of the learning samples.

Table 1. Test results: MSE, standard deviations (below the MSE) and t-test results (✓ for accept, × for reject and p-values). The best performing models are in bold.

Datasets	**Models**					
	MLM	ELM	OP-ELM	SVM	GP	MLP
Breast Cancer	**1.1**e+**3**	7.7e+3	1.4e+3	1.2e+3	1.3e+3	1.5e+3
	2.1e+2	2.0e+3	3.6e+2	7.2e+1	1.9e+2	4.4e+2
		× (e-9)	✓ (.955)	✓ (.913)	✓ (.109)	× (.027)
Boston	2.3e+1	1.2e+2	1.9e+1	3.4e+1	**1.1**e+**1**	2.2e+1
	1.2e+1	2.1e+1	2.9	3.1e+1	3.5	8.8
		× (e-10)	✓ (.374)	✓ (.314)	× (.008)	✓ (.851)
Servo	4.9e−1	7.1	8.0e−1	6.9e−1	**4.8**e−**1**	6.0e−1
	2.9e−1	5.5	3.3e−1	3.2e−1	3.5e−1	3.2e−1
		× (.001)	× (.037)	✓ (.164)	✓ (.961)	✓ (.427)
Abalone	4.6	8.3	4.9	**4.5**	**4.5**	4.6
	2.9e−1	7.5e−1	6.6e−1	2.7e−1	2.4e−1	5.0e−1
		× (e-11)	✓ (.353)	✓ (.378)	✓ (.206)	✓ (.844)
Stocks	**4.1**e−**1**	3.4e+1	9.8e−1	5.1e−1	4.4e−1	8.8e−1
	5.8e−2	9.35	1.1e−1	9.8e−2	5.0e−2	2.1e−1
		× (e-9)	× (e-11)	× (.016)	✓ (.329)	× (e-6)
Auto Price	5.1e+7	7.9e+9	9.5e+7	9.8e+7	2.0e+7	**1.0**e+**7**
	7.4e+7	7.2e+9	4.0e+6	8.4e+6	1.0e+7	3.9e+6
		× (.003)	✓ (.096)	✓ (.346)	✓ (.205)	✓ (.103)

All the models are evaluated using the mean and standard deviation of the resulting MSE over 10 independently drawn test sets. We also carried out a statistical evaluation of the MLM performance against those achieved by the other models using the two-sample t-test [9] with a significance level equal to

5%. The null hypothesis is that the MSE distributions are independent random samples from normal distributions with equal means and equal but unknown variances, against the alternative that the means are not equal. On the basis of the experimental results (Table 1), we can observe that the state-of-the-art models seem to be able to achieve similar accuracies. In this regard, also the MLM achieves performances that are comparable to such methods. The table also shows that most of the models are equivalent for the Auto Price and Abalone datasets, except of the ELM. The MLM is not among the best models only for the Boston dataset where the GP is the most reliable option. In addition, the most similar performances to MLM are those achieved by the SVM and GP models, whose null hypotheses were accepted for five different datasets.

4 Conclusions

This work presents a novel method for supervised learning, the Minimal Learning Machine, MLM. Learning a MLM consists in reconstructing the mapping existing between input and output distance matrices and then exploiting the geometrical arrangement of the output points for estimating the response. Based on our experiments, a multiresponse linear regression model is capable to reconstruct the mapping existing between the aforementioned distance matrices. The MLM has only one hyper-parameter to be optimized. Given its general formulation, the Minimal Learning Machine is also inherently capable to operate on multidimensional responses and it can be extended to classification problems.

On a large number of real-world problems, the Minimal Learning Machine has achieved accuracies that are comparable to what is obtained using state-of-the art nonlinear regression methods. For compactness, we have reported the performances on a selection of six datasets from the UCI Repository and comparisons with five reference regression approaches. The results highlight the potentiality of the MLM and we are currently further investigating its properties and the ties with classical dimensionality reduction methods based on distances.

References

1. Cox, T., Cox, M.: Multidimensional Scaling. Chapman & Hall, London (1994)
2. de Silva, V., Tenenbaum, J.B.: Global versus local methods in nonlinear dimensionality reduction. In: Advances in Neural Information Processing Systems, vol. 15, pp. 705–712 (2002)
3. Huang, G.B., Zhu, Q.Y., Ziew, C.K.: Extreme Learning Machine: Theory and applications. Neurocomputing 70(1-3), 489–501 (2006)
4. Miche, Y., Sorjamaa, A., Bas, P., Simula, O., Jutten, C., Lendasse, A.: OP-ELM: Optimally Pruned Extreme Learning Machine. IEEE Transactions on Neural Networks 21(1), 158–162 (2010)
5. Smola, A.J., Schölkopf, B.: A tutorial on support vector regression. Statistics and Computing 14(3), 199–222 (2004)

6. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. The MIT Press, Cambridge (2006)
7. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, New York (1995)
8. Chang, C.-C., Lin, C.-J.: LIBSVM: A Library for Support Vector Machines. ACM Transactions on Intelligent Systems and Technology 2(27), 1–27 (2011)
9. Lehmann, E.L., Romano, J.P.: Testing statistical hypotheses. Springer, New York (2005)