CSCanada
Advances in Petroleum Exploration and Development
Vol. 9, No. 1, 2015, pp. 1-8
DOI:10.3968/6573

ISSN 1925-542X [Print]
ISSN 1925-5438 [Online]
www.cscanada.net
www.cscanada.org

Economic Evaluation of Waterflood Using Regression and Classification Algorithms

ZHANG Qian[a]; SHI Guangren[a],*

[a] Research Institute of Petroleum Exploration and Development, PetroChina, China.
*Corresponding author.

Supported by the Research Institute of Petroleum Exploration and Development (RIPED) and PetroChina.

Received 28 January 2015; accepted 10 March 2015
Published online 30 March 2015

Abstract

Three regression algorithms and three classification algorithms have been applied to forecast economics of waterflood. The three regression algorithms are the regression of support vector machine (*R*-SVM), the back-propagation neural network (BPNN), and the multiple regression analysis (MRA), while the three classification algorithms are the classification of support vector machine (*C*-SVM), the naïve Bayesian (NBAY), and the Bayesian successive discrimination (BAYSD). In general, when all these six algorithms are used to solve a real-world problem, they often produce different solution accuracies. In this paper, the solution accuracy is expressed with the total mean absolute relative residual for all samples, $R(\%)$, and it is proposed that an algorithm is applicable if $R(\%) \leq 10$. A case study at the Nebraska Panhandle has been used to validate the proposed approach. This case study consists of two problems: regression and classification. The only difference between these two problems is the predicted variable in regression problem is real number, while the predicted variable in classification problem is integer number. And the integer number is determined from the real number by using proposed convertion rules. For the regression problem, *R*-SVM, BPNN and MRA are inapplicable because their $R(\%)$ values are 140, 51 and 293, respectively. For the classification problem, however, *C*-SVM, NBAY and BAYSD are all applicable since their $R(\%)$ values are all 0. From the case study, it is concluded that: a) For classification problems, the preferable algorithm is *C*-SVM, NBAY, or BAYSD, and BAYSD can also serve as a promising dimension-reduction tool; b) for regression problems, the preferable algorithm is BPNN, but MRA can serve as a promising dimension-reduction tool only when the studied problems are linear; c) if BPNN is inapplicable for a regression problem because its $R(\%) > 10$, it is proposed to change this problem from regression to classification by reasonable conversion rules, then apply *C*-SVM, NBAY, or BAYSD; and d) comparing with *C*-SVM, BAYSD is conditionally better than *C*-SVM.

Key words: Regression; Classification; Solution accuracy; Conversion rules; Dimensionality reduction; Nebraska Panhandle

Zhang, Q., & Shi, G. R. (2015). Economic evaluation of waterflood using regression and classification algorithms. *Advances in Petroleum Exploration and Development*, *9*(1), 1-8. Available from: URL: http://www.cscanada.net/index.php/aped/article/view/6573
DOI: http://dx.doi.org/10.3968/6573

INTRODUCTION

Correlations Company (2001) adopted fuzzy ranking and BPNN for the economic evaluation of waterflood[1].

This paper discusses the economic evaluation of waterflood using the following three regression algorithms and three classification algorithms. The three regression algorithms are the regression of support vector machine (*R*-SVM), the back-propagation neural network (BPNN), and the multiple regression analysis (MRA), while the three classification algorithms are the classification of support vector machine (*C*-SVM), the naïve Bayesian (NBAY), and the Bayesian successive discrimination (BAYSD). In general, when all these six algorithms are used to solve a real-world problem, they often produce different solution accuracies. In this paper, the solution accuracy is expressed with the total mean absolute relative residual for all samples, $R(\%)$. In general, it is proposed that an algorithm is applicable if $R(\%) \leq 5$, otherwise this

algorithm is inapplicable. In this paper, however, it is proposed that an algorithm is applicable if $R(\%) \leq 10$, otherwise this algorithm is inapplicable. This is because the subsurface geoscience is different from the other fields, with miscellaneous data types, huge quantity, different measuring precision, and lots of uncertainties to data processing results[2-3]. The case study at the Nebraska Panhandle below has been used to validate the proposed approach.

1. METHODOLOGY

The methodology consists of the following three major parts: definitions commonly used by regression and classification algorithms; methods of six algorithms; dimensionality reduction.

1.1 Definitions Commonly Used by Regression and Classification Algorithms

The aforementioned regression and classification algorithms share the data of samples. The essential difference between the two types of algorithms is that the output of regression algorithms is real-type value and in general differs from the real number given in the corresponding learning sample, whereas the output of classification algorithms is integer-type value and must be one of the integers defined in the learning samples. In the view of dataology, the integer-type value is called as discrete attribute, while the real-type value is called as continuous attribute.

The six algorithms (*R*-SVM, BPNN, MRA, *C*-SVM, NBAY, BAYSD) use the same known parameters, and also share the same unknown that is predicted. The only difference between them is the approach and calculation results.

Assume that there are n learning samples, each associated with $m + 1$ numbers ($x_1, x_2, \ldots, x_m, y^*$) and a set of observed values ($x_{i1}, x_{i2}, \ldots, x_{im}, y^*_i$), with $i = 1, 2, \ldots, n$ for these numbers. In principle, $n > m$, but in actual practice $n >> m$. The n samples associated with $m + 1$ numbers are defined as n vectors:

$$\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{im}, y^*_i) \quad (i = 1, 2, \ldots, n), \tag{1}$$

where n is the number of learning samples; m is the number of independent variables in samples; $\boldsymbol{x}_i$ is the i^{th} learning sample vector; x_{ij} is the value of the j^{th} independent variable in the i^{th} learning sample, $j = 1, 2, \ldots, m$; and y^*_i is the observed value of the i^{th} learning sample.

Equation (1) is the expression of learning samples.

Let $\boldsymbol{x}_0$ be the general form of a vector of ($x_{i1}, x_{i2}, \ldots, x_{im}$). The principles of BPNN, MRA, NBAY and BAYSD are the same, that is, try to construct an expression, $y = y(\boldsymbol{x}_0)$, such that Equation (2) is minimized. Certainly, these four different algorithms use different approaches and obtain calculation results in differing accuracies.

$$\sum_{i=1}^{n} \left[y\left(\boldsymbol{x}_{0i}\right) - y^*_i \right]^2, \tag{2}$$

where $y = y(\boldsymbol{x}_{0i})$ is the calculation result of the dependent variable in the i^{th} learning sample; and the other symbols have been defined in Equation (1).

However, the principles of *R*-SVM and *C*-SVM algorithms are to try to construct an expression, $y = y(\boldsymbol{x}_0)$, such that to maximize the margin based on support vector points so as to obtain the optimal separating line.

This $y = y(\boldsymbol{x}_0)$ is called the fitting formula obtained in the learning process. The fitting formulas of different algorithms are different. In this paper, y is defined as a single variable.

The flowchart is as follows: The 1st step is the learning process, using n learning samples to obtain a fitting formula; the 2nd step is the learning validation, substituting n learning samples ($x_{i1}, x_{i2}, \ldots, x_{im}$) into the fitting formula to get prediction values ($y_1, y_2, \ldots, y_n$), respectively, so as to verify the fitness of an algorithm; and the 3rd step is the prediction process, substituting k prediction samples expressed with Equation (3) into the fitting formula to get prediction values ($y_{n+1}, y_{n+2}, \ldots, y_{n+k}$), respectively.

$$\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{im}) \quad (i = n + 1, n + 2, \ldots, n + k), \tag{3}$$

where k is the number of prediction samples; $\boldsymbol{x}_i$ is the i^{th} prediction sample vector; the other symbols have been defined in Equation (1).

Equation (3) is the expression of prediction samples.

In the six algorithms, only MRA is a linear algorithm whereas the other five are nonlinear algorithms, this is due to the fact that MRA constructs a linear function whereas the other five construct nonlinear functions, respectively.

To express the calculation accuracies of the prediction variable y for learning and prediction samples when the six algorithms are used, the following four types of residuals are defined.

The absolute relative residual for each sample, $R(\%)_i$ ($i = 1, 2, \ldots, n, n + 1, n + 2, \ldots, n + k$), is defined as

$$R(\%)_i = \left| (y_i - y^*_i) / y^*_i \right| \times 100, \tag{4}$$

where y_i is the calculation result of the dependent variable in the i^{th} sample; and the other symbols have been defined in Equations (1) and (3). $R(\%)_i$ is the fitting residual to express the fitness for a sample in learning or prediction process.

It is noted that zero must not be taken as a value of y^*_i to avoid floating-point overflow. Therefore, for regression algorithm, delete the sample if its $y^*_i = 0$; and for classification algorithm, positive integer is taken as values of y^*_i.

The mean absolute relative residual for all learning samples, $R_1(\%)$, is defined as

$$R_1(\%) = \sum_{i=1}^{n} R(\%)_i / n, \tag{5}$$

where all symbols have been defined in Equations (1) and (4). $R_1(\%)$ is the fitting residual to express the fitness of learning process.

The mean absolute relative residual for all prediction samples, $R_2(\%)$, is defined as

$$R_2(\%) = \sum_{i=n+1}^{k} R(\%)_i / k \ , \quad (6)$$

where all symbols have been defined in Equations (3) and (4). $R_2(\%)$ is the fitting residual to express the fitness of prediction process.

The total mean absolute relative residual for all samples, $R(\%)$, is defined as

$$R(\%) = \sum_{i=1}^{n+k} R(\%)_i / (n+k) \ , \quad (7)$$

where all symbols have been defined in Equations (1), (3) and (4). If there are no prediction samples, $k = 0$, then $R(\%) = R_1(\%)$.

$R(\%)$ is the fitting residual to express the fitness of learning and prediction processes.

When the six algorithms (*R*-SVM, BPNN, MRA, *C*-SVM, NBAY, BAYSD) are used to solve a real-world problem, they often produce different solution accuracies. In this paper, the solution accuracy is expressed with $R(\%)$ shown in Equation (7), and it is proposed that an algorithm is applicable if $R(\%) \le 10$, otherwise this algorithm is inapplicable.

1.2 Methods of Six Algorithms

The methods of the six algorithms (*R*-SVM, BPNN, MRA, *C*-SVM, NBAY, BAYSD) are not detailedly described here because readers can refer to the relevant articles and books (For example: [2-6]).

Through the learning process, each algorithm constructs its own function $y = y(\boldsymbol{x})$. It is noted that $y = y(\boldsymbol{x})$ created by BPNN is an implicit expression, that is, which cannot be expressed as a usual mathematical formula; whereas that of the other five algorithms are explicit expressions, that is, which are expressed as a usual mathematical formula.

In the case study below, (a) in *C*-SVM and *R*-SVM, the kernel function used is the RBF (radial basis function), and the termination of calculation accuracy TCA is fixed to 10^{-3}; and the insensitive function ε in *R*-SVM is fixed to 0.1. (b) in BPNN, $N_{\text{hidden}} = 2(N_{\text{input}}+N_{\text{output}})-1$ where N_{hidden} is the number of hidden nodes, N_{input} is the number of input nodes and N_{output} is the number of output nodes; TCA is fixed to 10^{-4}; And in each iteration, the error takes the root mean square error[2, 3, 7] is

$$\text{RMSE}(\%) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i^*)^2} \times 100 \ , \quad (8)$$

where y_i and y^*_i are under the conditions of normalizations in the learning process. RMSE(%) is used in the conditions for terminating network learning.

1.3 Dimensionality Reduction

The definition of dimensionality reduction is to reduce the number of dimensions of a data space as small as possible but the results of studied problem are unchanged. The benefits of dimensionality reduction are to reduce the amount of data can enhance the calculating speed, to reduce the independent variables can extend applying ranges, and to reduce misclassification ratio of prediction samples can enhance processing quality.

Among the aforementioned six algorithms, each of MRA and BAYSD can serve as a promising dimension-reduction tool, respectively, because the two algorithms all can give the dependence of the predicted value (y) on independent variables ($x_1, x_2, \ldots, x_m$), in decreasing order. However, because MRA belongs to data analysis in linear correlation whereas BAYSD is in nonlinear correlation, in applications the preferable tool is BAYSD, whereas MRA is available only when the studied problems are linear. The called "promising tool" is whether it succeeds or not needs a high-class nonlinear tool (e.g., BPNN for regression problem, *C*-SVM for classification problem) for the validation, so as to determine how many independent variables can be reduced. For instance, the classification problem in the case study below indicates that a 7-D problem ($x_1, x_2, x_3, x_4, x_5, x_6, y$) can be reduced to 5-D problem (x_2, x_3, x_4, x_6, y).

2. CASE STUDY: ECONOMIC EVALUATION OF WATERFLOOD AT THE NEBRASKA PANHANDLE

This case study consists of two problems: regression and classification. The objective of this case study is to calculate the ratio of secondary to primary oil recovery (S/P), and to determine the S/P classification (SPC) for oilfields, which has practical value when the waterflood has not been installed in oilfields.

Using data of 18 samples from the Nebraska Panhandle of the Denver-Julesberg Basin, USA[1], and each sample contains 6 independent variables (x_1 = lateral area, x_2 = average porosity, x_3 = average permeability, x_4 = original bottom hole pressure, x_5 = cumulative water oil ratio, x_6 = cumulative gas oil ratio) and one variable (y^* = S/P), Correlations Company (2001) adopted fuzzy ranking and BPNN for the prediction of S/P[1]. In the case study, among these 18 samples, 17 are taken as learning samples and one as prediction sample (Table 1) for the prediction of both S/P and SPC, in which for S/P using *R*-SVM, BPNN and MRA, and for SPC using *C*-SVM, NBAY and BAYSD. It is noted that this SPC is figured out from S/P by using the conversion rules given in Table 2. In Table 2, if $S/P \le 0.25$ the waterflood is expected to be marginally economic[1].

Table 1
Input Data for Economic Evaluation of Waterflood at the Nebraska Panhandle

Sample type	Sample No.		Six parameters related to y[a]					y^*	
		x_1 (acres)	x_2 (%)	x_3 (10^{-3} μm^2)	x_4 (psi)	x_5 (bbl/bbl)	x_6 (mcf/bbl)	S/P[b]	SPC[c]
Learning samples	1	560	17	63	1,328	8.37	0.02	0.56	3
	2	2,080	21	212	1,400	6.86	0.92	0.68	3
	3	960	20	100	1,300	252.55	0.03	0.96	3
	4	1,840	16.8	42	1,240	1.62	0.05	1.17	3
	5	24,000	22	400	1,115	0.82	0.00	6.98	3
	6	12,000	23.2	44	1,300	2.05	0.55	1.57	3
	7	840	17.5	139	1,100	2.15	0.15	0.52	3
	8	960	17.4	430	1,000	1.84	2.08	0.02	1
	9	1,920	10.7	10	1,590	14.92	10.88	0.02	1
	10	1,100	17.5	86	1,546	5.97	0.40	0.32	2
	11	800	18.1	60	1,640	2.00	0.26	0.26	2
	12	480	15.2	62.2	1,600	2.79	0.01	0.37	2
	13	440	19	100	1,240	2.97	0.04	1.18	3
	14	1,120	20	150	1,375	6.20	0.03	1.01	3
	15	160	16	72	1,510	1.60	0.10	0.4	2
	16	1,760	24	400	1,200	37.40	0.03	0.64	3
	17	640	21.8	15	1,350	3.57	0.00	0.53	3
Prediction samples	18	2,320	21.8	15	1,350	3.47	0.01	(1.24)	(3)

[a] x_1 = lateral area, x_2 = average porosity, x_3 = average permeability, x_4 = original bottom hole pressure, x_5 = cumulative water oil ratio, x_6 = cumulative gas oil ratio.
[b] S/P = the ratio of secondary to primary oil recovery, number in parenthesis is not input data, but is used for calculating $R(\%)_i$.
[c] SPC = the S/P classification (1−economic, 2−uneconomic, 3−very uneconomic) determined by Table 2, number in parenthesis is not input data, but is used for calculating $R(\%)_i$.

Table 2
***S/P* Classification Based on the Ratio of Secondary to Primary Oil Recovery**

Economic evaluation of waterfloods	S/P (The ratio of secondary to primary oil recovery)	SPC (S/P classification)
Economic	≤ 0.25	1
Uneconomic	(0.25, 0.5]	2
Very uneconomic	> 0.5	3

2.1 Regression Problem for Calculating the Ratio of SECOndary to Primary Oil Recovery (*S/P*)

Using the 17 learning samples with $y^* = S/P$ (Table 1) and by R-SVM, BPNN and MRA, the following three functions of S/P (y) with respect to 6 independent variables (x_1, x_2, x_3, x_4, x_5, x_6) have been constructed.

Using R-SVM[2, 3, 4], the result is an explicit nonlinear function:

$$y = R\text{-}SVM(x_1, x_2, x_3, x_4, x_5, x_6), \quad (9)$$

with the penalty factor $C = 1$, the regularization parameter $\gamma = 0.166667$, and 13 free vectors $\boldsymbol{x}_i$.

The BPNN[2, 3] used consists of 6 input layer nodes, 1 output layer node and 13 hidden layer nodes. The result is an implicit nonlinear function:

$$y = \text{BPNN}(x_1, x_2, x_3, x_4, x_5, x_6), \quad (10)$$

with the optimal learning time count t_{opt} = 168,640, and RMSE(%) = 0.424×10^{-2}.

Using MRA[2, 3], the result is an explicit linear function:

$$y = -2.43 + 0.000255x_1 - 0.0986x_2 + 0.000872x_3 - 0.000291x_4 + 0.00227x_5 - 0.15x_6, \quad (11)$$

Equation (11) yields a residual variance of 0.092 and a multiple correlation coefficient of 0.953. From the regression process, S/P (y) is shown to depend on the 6 independent variables in decreasing order: x_1, x_6, x_2, x_5, x_3, and x_4.

Substituting the values of 6 independent variables (x_1, x_2, x_3, x_4, x_5, x_6) given by the 17 learning samples and one prediction sample (Table 1) in Equations (9), (10) and (11), respectively, the S/P (y) of each sample is obtained (Table 3).

From Table 4, R-SVM, BPNN and MRA are inapplicable because their $R(\%)$ values are 140, 51 and 293, respectively.

Table 3
Prediction Results of *S*/*P* at the Nebraska Panhandle

Sample type	Sample No.	The ratio of secondary to primary oil recovery (*S*/*P*)						
			Regression algorithm					
		y^*	*R*-SVM		BPNN		MRA	
			y	$R(\%)_i$	y	$R(\%)_i$	y	$R(\%)_i$
Learning samples	1	0.56	0.645	15.2	0.669	19.4	0.585	4.5
	2	0.68	0.712	4.67	0.466	31.4	0.549	19.2
	3	0.96	0.86	10.4	1.37	43	0.985	2.64
	4	1.17	0.762	34.9	1.77	50.9	0.919	21.5
	5	6.98	1.57	77.5	6.98	0	6.41	8.14
	6	1.57	1.35	14.1	5.46	248	2.79	77.7
	7	0.52	0.741	42.5	0.28	46.1	0.706	35.8
	8	0.02	0.335	1572	0.02	0	0.74	360
	9	0.02	0.12	501	0.02	0	-0.179	995
	10	0.32	0.441	37.8	0.533	66.4	0.568	77.5
	11	0.26	0.36	38.4	0.585	125	0.394	51.6
	12	0.37	0.309	16.6	0.413	11.5	0.651	76
	13	1.18	0.764	35.2	1.49	26.5	0.4	66.1
	14	1.01	0.707	30	1.68	66.4	0.488	51.7
	15	0.4	0.407	1.65	0.348	13	0.509	27.3
	16	0.64	0.74	15.6	0.59	7.88	0.596	6.81
	17	0.53	0.818	54.3	1.36	156	0.076	85.7
Prediction samples	18	1.24	0.883	28.8	1.15	7.15	0.503	59.5

Table 4
Comparison Among the Applications of Regression Algorithms (*R*-SVM, BPNN and MRA) to *S*/*P* at the Nebraska Panhandle

Algorithm	Fitting formula	Mean absolute relative residual			Dependence of the predicted value (y) on independent variables ($x_1, x_2, x_3, x_4, x_5, x_6$), in decreasing order	Time consuming on PC (Intel Core 2)	Results availability
		R_1(%)	R_2(%)	R(%)			
R-SVM	Nonlinear, explicit	147	28.8	140	N/A	3 s	Inapplicable
BPNN	Nonlinear, implicit	53.6	7.15	51	N/A	30 s	Inapplicable
MRA	Linear, explicit	306	59.5	293	$x_1, x_6, x_2, x_5, x_3, x_4$	<1 s	Inapplicable

2.2 Dimension-Reduction Failed by Using MRA and BPNN

MRA gives the dependence of the predicted value (y) on 6 independent variables, in decreasing order: x_1, x_6, x_2, x_5, x_3, x_4 (Table 4). According to this dependence order, at first, deleting x_4 and running BPNN, it is found the results of BPNN are changed, that is, R(%) = 84 which is greater much than previous R(%) = 51 (Table 4). Thus the 7-D problem (x_1, x_2, x_3, x_4, x_5, x_6, y) cannot become 6-D problem (x_1, x_2, x_3, x_5, x_6, y). This is due to the fact that this regression problem is a nonlinear problem according to R(%) values of R-SVM and MRA are 140 and 293, respectively (Table 4). Therefore, MRA can serve as a promising dimension-reduction tool only when the studied problems are linear.

2.3 Classification Problem for Determining the *S*/*P* Classification (*SPC*)

Using the 17 learning samples with $y^* = SPC$ (Table 1) and by C-SVM, NBAY and BAYSD, the following three functions of SPC (y) with respect to 6 independent variables ($x_1, x_2, x_3, x_4, x_5, x_6$) have been constructed.

Using C-SVM[2, 3, 4], the result is an explicit nonlinear function:

$$y = C\text{-}SVM(x_1, x_2, x_3, x_4, x_5, x_6), \quad (12)$$

with C = 32, γ = 0.03125, 9 free vectors $\boldsymbol{x}_i$, and the cross validation accuracy CVA = 88.26%.

Using NBAY[2, 3, 5, 6], the result is an explicit nonlinear discriminate function:

$$N_l(\boldsymbol{x}) = \prod_{j=1}^{6}\left\{\frac{1}{\sigma_{jl}\sqrt{2\pi}}\exp\left(\frac{-(x_j-\mu_{jl})^2}{2\sigma_{jl}^2}\right)\right\}, \quad (l=1,2,3) \tag{13}$$

where for l = 1, σ_{j1} = 480, 3.35, 210, 295, 6.54, 4.4, μ_{j1} = 1,440, 14.1, 220, 1,295, 8.38, 6.48; for l = 2, σ_{j2} = 351, 1.16, 10.26, 49.8, 1.72, 0.15, μ_{j2} = 635, 16.7, 70.1, 1,574, 3.09, 0.192; For l = 3, σ_{j3} = 7,008, 2.35, 129, 94.9, 71.2, 0.282, μ_{j3} = 4,203, 20.2, 151, 1,268, 29.5, 0.165.

Once Equation (13) is created, the values of 6 independent variables (x_1, x_2, x_3, x_4, x_5, x_6) of any sample (Table 1) can be substituted in Equation (13) to obtain 3 values: N_1, N_2, N_3. If $N_{l_b} = \max_{1\le l\le 3}\{N_l\}$ then

$$y = l_b, \tag{14}$$

for this sample.

Using BAYSD[2, 3], the result is an explicit nonlinear discriminate function:

$$\begin{cases} B_1(\boldsymbol{x}) = \ln(0.118) - 116 + 0x_1 + 1.15x_2 + \\ \qquad 0.083x_3 + 0.178x_4 - 0.019x_5 - 5.17x_6 \\ B_2(\boldsymbol{x}) = \ln(0.235) - 246 + 0.001x_1 - 0.632x_2 + \\ \qquad 0.085x_3 + 0.317x_4 - 0.024x_5 - 18.2x_6 \\ B_3(\boldsymbol{x}) = \ln(0.647) - 171 + 0.001x_1 + 1.34x_2 + \\ \qquad 0.057x_3 + 0.241x_4 - 0.015x_5 - 12.9x_6 \end{cases}, \tag{15}$$

Once Equation (15) is created, the values of 6 independent variables (x_1, x_2, x_3, x_4, x_5, x_6) of any sample (Table 1) can be substituted in Equation (15) to obtain 3 values: B_1, B_2, B_3. If $B_{l_b} = \max_{1\le l\le 3}\{B_l\}$ then

$$y = l_b, \tag{16}$$

for this sample.

From the successive process, SPC (y) is shown to depend on the 6 independent variables in decreasing order: x_6, x_4, x_2, x_3, x_1, and x_5.

Substituting the values of 6 independent variables (x_1, x_2, x_3, x_4, x_5, x_6) given by the 17 learning samples and one prediction samples (Table 1) in Equations (12), (13) (and then use Equation (14)), and (15) (and then use Equation (16)), respectively, the *SPC* (y) of each sample is obtained (Table 5).

Table 5
Prediction Results of SPC at the Nebraska Panhandle

Sample type	Sample No.	The *S*/*P* classification (*SPC*)					
			Classification algorithm				
		y^*	*C*-SVM		NBAY		BAYSD	
			y	$R(\%)_i$	y	$R(\%)_i$	y	$R(\%)_i$
Learning samples	1	3	3	0	3	0	3	0
	2	3	3	0	3	0	3	0
	3	3	3	0	3	0	3	0
	4	3	3	0	3	0	3	0
	5	3	3	0	3	0	3	0
	6	3	3	0	3	0	3	0
	7	3	3	0	3	0	3	0
	8	1	1	0	1	0	1	0
	9	1	1	0	1	0	1	0
	10	2	2	0	2	0	2	0
	11	2	2	0	2	0	2	0
	12	2	2	0	2	0	2	0
	13	3	3	0	3	0	3	0
	14	3	3	0	3	0	3	0
	15	2	2	0	2	0	2	0
	16	3	3	0	3	0	3	0
	17	3	3	0	3	0	3	0
Prediction samples	18	3	3	0	3	0	3	0

Table 6
Comparison Among the Applications of Regression Algorithms (*C*-SVM, NBAY and BAYSD) to SPC at the Nebraska Panhandle

Algorithm	Fitting formula	Mean absolute relative residual			Dependence of the predicted value (y) on independent variables (x_1, x_2, x_3, x_4, x_5, x_6), in decreasing order	Time consuming on PC (Intel Core 2)	Results availability
		R_1(%)	R_2(%)	R(%)			
C-SVM	Nonlinear, explicit	0	0	0	N/A	5 s	Applicable
NBAY	Nonlinear, explicit	0	0	0	N/A	< 1 s	Applicable
BAYSD	Nonlinear, explicit	0	0	0	x_6, x_4, x_2, x_3, x_1, x_5	1 s	Applicable

From Table 6, *C*-SVM, NBAY and BAYSD are applicable since their R(%) values are all 0.

2.4 Dimension-Reduction From 7-D to 5-D Problem by Using BAYSD and *C*-SVM

BAYSD gives the dependence of the predicted value (y) on 6 independent variables, in decreasing order: x_6, x_4, x_2, x_3, x_1, x_5 (Table 6). According to this dependence order, at first, deleting x_5 and running *C*-SVM, it is found the results of *C*-SVM are the same as before, that is, R(%) = 0, thus 7-D problem (x_1, x_2, x_3, x_4, x_5, x_6, y) can become 6-D problem (x_1, x_2, x_3, x_4, x_6, y). In the same way, it is found that this 6-D problem can become 5-D problem by deleting x_1, but deleting x_2 is failed since the results of *C*-SVM are changed, that is, R(%) = 11.1. For this classification problem, therefore, the 7-D problem (x_1, x_2, x_3, x_4, x_5, x_6, y) at last can become 5-D problem (x_2, x_3, x_4, x_6, y).

2.5 Summary of the Case Study

From Tables 4 and 6, Table 7 summarizes the applicability of each algorithm in the case study.

Table 7
Summary of the Case Study at the Nebraska Panhandle

Algorithm type	Algorithm	Mean absolute relative residual			Dependence of the predicted value (y) on independent variables (x_1, x_2, x_3, x_4, x_5, x_6), in decreasing order	Time consuming on PC (Intel Core 2)	Results availability
		R_1(%)	R_2(%)	R(%)			
Regression Algorithm (see Table 4)	*R*-SVM	147	28.8	140	N/A	3 s	Inapplicable
	BPNN	53.6	7.15	51	N/A	30 s	Inapplicable
	MRA	306	59.5	293	x_1, x_6, x_2, x_5, x_3, x_4	< 1 s	Inapplicable
Classification Algorithm (see Table 6)	*C*-SVM	0	0	0	N/A	5 s	Applicable
	NBAY	0	0	0	N/A	< 1 s	Applicable
	BAYSD	0	0	0	x_6, x_4, x_2, x_3, x_1, x_5	1 s	Applicable

Comparing with *C*-SVM, the major advantages of BAYSD are: (a) BAYSD runs much faster than *C*-SVM, (b) it is easy to code the BAYSD program whereas very complicated to code the *C*-SVM program, and (c) BAYSD can serve as a promising dimension-reduction tool. So BAYSD is conditionally better than *C*-SVM.

CONCLUSION

The purpose of this paper is how to select a proper algorithm in three algorithms (*C*-SVM, NBAY, BAYSD) for regression problems and three algorithms (*R*-SVM, BPNN, MRA) for regression problems. From the aforementioned case study at the Nebraska Panhandle, five major conclusions can be drawn as follows:

The definition of solution accuracy R(%), the threshold of applicability (R(%) ≤ 10) for an algorithm, and the rules of conversion from real number to integer number are practical;

For classification problems, the preferable algorithm is *C*-SVM, NBAY, or BAYSD, and BAYSD can also serve as a promising dimension-reduction tool;

For regression problems, the preferable algorithm is BPNN, but MRA can serve as a promising dimension-reduction tool only when the studied problems are linear;

If BPNN is inapplicable for a regression problem because its R(%) > 10, it is proposed to change this problem from regression to classification by reasonable conversion rules, then apply *C*-SVM, NBAY, or BAYSD;

And comparing with *C*-SVM, BAYSD is conditionally better than *C*-SVM.

REFERENCES

[1] Correlations Company. (2001). *Data mining at the Nebraska oil & gas commission* (Final technical report) [Online forum comment]. Retrieved from http://www.netl.doe.gov/kmd/cds/disk37/C%20-%20Independent%20Producers%20Program/15255%20final.pdf/

[2] Shi, G. R. (2015). Optimal prediction in petroleum geology by regression and classification methods. *Sci. J. Information Engineering*, *5*(2), 14-32.

[3] Shi, G. R. (2013). *Data mining and knowledge discovery for geoscientists*. USA: Elsevier Inc.

[4] Chang, C. C., & Lin, C. J. (2011). *LIBSVM: A library for support vector machines* (Version 3.1) [Online forum comment]. Retrieved from http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[5] Han, J., & Kamber, M. (2006). *Data mining*: *Concepts and techniques* (2nd ed.). San Francisco, CA, USA: Morgan Kaufmann.

[6] Tan, P., Steinbach, M., & Kumar, V. (2005). *Introduction to data mining.* Boston, MA, USA: Pearson Education.

[7] Hush, D. R., & Horne, B. G. (1993). Progress in supervised neural networks. *IEEE Sig. Proc. Mag*., *10*(1), 8-39.