RESEARCH ARTICLE OPEN ACCESS

Missing Value Imputation Based on MINNS-SVM

XueLing Chan[1], YanjunZhong[2],YujuanQuan[3]
*(Computer Science, JiNan University, China)

----------**********----------

Abstract:

Missing value processing is an unavoidable problem of data pre-processing in the field of Machine learning. Most traditional missing value imputation methods are based on probability distribution and the likes, which might not be suitable for high-dimensional data. Inspired by many unique advantages of Support Vector Machine (SVM) in the high-dimensional model, this paper proposes the missing value imputation based on the nearest neighbors similarity and support vector machine. Four commonly used data sets in the UCI machine learning database are adopted in the experiment, with experimental results showing that MINNS-SVM is effective.

Keywords **—missing value, data imputation, support vector machine.**

----------**********----------

I. INTRODUCTION

A lot of factors in real life lead to missing value, such as a failed data acquisition in experiment, loss of data during transmission, incomplete data because of unanswered questions in the survey, etc. Even the most famous UCI database in the field of machine learning also has more than 40% of the data sets that contains missing value [1]. Missing value is usually divided into two categories—missing completely at random (MCAR) and missing at random (MAR), in which the missing value of MCAR does not depend on all the observed data, while MAR is just the opposite[2,3]. Since MAR meets the requirements of reality to the largest extent, most of the researches on imputation algorithms are based on MAR data.

Faced with the inevitable MAR missing value, existing research has proposed a large number of commonly used methods for missing value imputation, which can be divided into the following three categories: simple processing, probability theory imputation and imputation based on neighborsimilarity. Simple processing method mainly includes the simple discarding method and the mean value imputation method. These two methods can not estimate whether the missing value will affect the results of machine learning, so simply discarding or substituting the missing value by mean value is likely to affect the objectivity of data and reduce the accuracy of machine learning outcomes[4,5]; probability theory imputation includes the missing value imputation methods based on Bayesian probability and probability-weighted[6,7]. According to a priori probability of data distribution, Chiu and Sedransk put forward a new Bayesian method for predicting the missing value[8]. The probability imputation method has a good anti-jamming capability; neighbor similarity imputation-based principle is: things of the same kind or similar nature are similar in attributes, so neighbor similarity-based imputation method can effectively complement the missing value.

As the most classic in neighbor similarity-based imputation methods, K-means clustering proposed by Amanda is to find out K samples nearest to the missing value firstly, followed by a weighted average of the missing value corresponding to K samples, and then take calculated results as the imputation of the missing value[9]. Difficulty of this method is to select similarity and K value between samples, different similarities and K values will affect the accuracy of results, and the selection of these two parameters is also more difficult in the high-dimensional model. Support vector machine exhibits many unique advantages in the problem of high-dimensional pattern recognition, and every item of the real missing value has a lot of attributes,

forming a high-dimensional model. Therefore, a MINNS-SVM (Missing value Imputation base on Nearest Neighbors Similarity and Support Vector Machine) is proposed in this paper.

In order to verify the effectiveness of MINNS-SVM algorithm, the experimental investigation is conducted from two aspects. On the one hand, four complete data sets are chosen from UCI database, and then a certain percentage of random data is extracted as missing value, using MINNS-SVM algorithm for missing value imputation to verify the validity of the algorithm through comparison of the error between the original value and imputation value; on the other hand, the data set usually needs to be classified in the field of machine learning. In the experiment, commonly used classification algorithms of machine learning are used to classify the complete data set for verification. Then missing value and imputed value are classified with the identical classification algorithm to contrast the accuracy of classification. The experimental analysis shows that imputed value of MINNS-SVM algorithm has a better quality.

II. MINNS-SVM ALGORITHM DESIGN

MINNS-SVM algorithm is a mix of the ideas of nearest neighbor similarity and the methods of support vector machine classification. The dataset is divided into two sets—with missing value and without missing value. The dataset without missing value serves as the training set, and the one with missing value serves as the test set, finally missing value is imputed with the use of the ideas of nearest neighbor similarity.

data sets can expressed as a matrix.

$$D=(D_1,D_2,....,D_n)=\begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1j} & \cdots & d_n \\ d_{21} & d_{22} & \cdots & d_{2j} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & & & \vdots \\ d_{i1} & d_{i2} & & d_{ij} & & d_{in} \\ \vdots & \vdots & & & \ddots & \vdots \\ d_{m1} & d_{m2} & \cdots & d_{mj} & \cdots & d_{mn} \end{bmatrix}$$

$D_i=(\mathrm{d}_{i1},\mathrm{d}_{i2},...,\mathrm{d}_{in})$ represents the i-th data attribute, d_{ij} is the value of j attributes. $D_j=(\mathrm{d}_{1j},\mathrm{d}_{2j},...,\mathrm{d}_{mj})^T$ isthe setofvalues of all the j-th attributes. $d_{ij}="*"$ indicates that the attributes of this value is missing.

Dataset D is divided into two parts: $D=M\cup C$ and $C=(C_1,C_2,....,C_n)$ with complete attributes. $M=(M_1,M_2,....,M_n)$ is the dataset withmissing attributes.

A. Configuration of SVM

Support vector machine is an algorithm of machine learningbased on the Statistical Learning Theory (SLT). Principle of this algorithm is to find a hyperplane, so that points belonging to different categories are located on both sides of the hyperplanewith the largest empty region [10, 11].

1) Linear SVM

In the dataset with missing value, $m_{ip_i}="*"$. p_i represents the location of missing attribute in the missing value, while the training sample set deletes this attribute of the same category, shown as follows:

$$\left(C_i*E^{'},T_i\right),\mathrm{i}=1,2,3,\ldots,k,T\in\{+1,-1\}$$

$$E^{'}=\begin{bmatrix} E_{p_i-1} & 0 \\ 0 & E_{n-p_i} \end{bmatrix} \quad (1)$$

k isthe line width of matrix C. T_i isthe value of classification, so hyperplane is $\boldsymbol{\omega}\cdot(C_i*E^{'})+b=0$, which should satisfy the condition of $T_i\left[\boldsymbol{\omega}\cdot(C_i*E^{'})+\mathrm{b}\right]\geq 1,\mathrm{i}=1,2,3,\ldots,k$. At this point, interval of classification is $\frac{2}{\|\boldsymbol{\omega}\|}$, so the problem of finding the optimal hyperplane is transformed to that of finding the minimum for constrained optimization:

$$\min\frac{1}{2}\|\boldsymbol{\omega}\|^2$$

$$\text{subject to } T_i[\boldsymbol{\omega}\cdot(C_i*E^{'})+\mathrm{b}]-1\geq 0\ \ i=1,2,...,\mathrm{k} \quad (2)$$

In order to solve constrained optimization problem, Lagrange function is introduced:

$$\mathrm{L}(\boldsymbol{\omega},\mathrm{b},\mathrm{a})=\frac{1}{2}\boldsymbol{\omega}^2-\sum_{i=1}^{n}a_i\left(T_i\left(\boldsymbol{\omega}\cdot(C_i*E^{'})+b\right)-1\right) \quad (3)$$

According to Saddle Point Theorem, the result isa$^* = (a_1^*, a_2^*, a_3^*, \dots, a_n^*)^T$.

$$\boldsymbol{\omega}^* = \sum_{j=1}^{n} a_j^* (C_j * E')T_j \quad (4)$$

Where the subscript $j \in \{j | a_j^* > 0\}$. Therefore, the optimal hyperplane is $\left(\boldsymbol{\omega}^* \cdot \left(C * E'\right)\right) + b^* = 0$, and the optimal classification function is:

$$f(x) = \text{sgn}\left\{\left(\boldsymbol{\omega}^* \cdot x\right) + b^*\right\}, x \in M * E' \quad (5)$$

2) Non-linear SVM

For the case of non-linear data, SVM is to map the training set to a higher-dimensional vector space, thereby solving the maximum interval hyperplane in the higher-dimensional vector space. In the corresponding nonlinear case, x will be mapped to a high-dimensional space H, then

$$x \to \Phi(x) = (\Phi_1(x), \Phi_2(x), ..., \Phi_n(x))^T \quad (6)$$

The SVM decision function is obtained by solving the above function under nonlinear situation:

$$f(x) = \text{sgn}(\sum_{i=1}^{n} a_i T_i \Phi(x_i) \cdot \Phi(x) + b) \quad (7)$$

Seen from the above formula, objective function and decision function only relate to the inner product operation between data, so support vector machine is dominant in the high-dimensional data calculation.

B. Configuration of MINNS-SVM

With the use of the nearest neighbors similarity idea, missing value imputation is achieved on the basis of the determination of missing value category.

Similarity refers that the specific number of relative indicators is on a unified scale, using the principles of fuzzy comprehensive evaluation to determine the value of the evaluation criteria and draw an item's difference values betweenthe standard value and those of different indexes. Similarity is to measure the degree of similarity between two data, which is often calculated by the distance between the two data. The most common formula for calculating the distance is the Euclidean distance:

$$\text{dist}(C_i, M_i) = \sqrt{\sum_{j=1}^{n} (c_{ij} - m_{ij})^2} \quad (8)$$

Where k represents the data's dimension, xi, yi are the i-th components of data x, y.

Therefore, the transformational relation between similarity and distance can be expressed as:

$$\text{sim}(C_i, M_i) = g\left(\text{dist}(C_i, M_i)\right)$$
$$\text{subject to} \sum_{i=1}^{m-k} \text{sim}(C_i, M_i) = 1 \quad (9)$$

Where g: $R \to R$ is the one-to-one mapping from distance $\text{dist}(C_i, M_i)$ to similarity $\text{sim}(C_i, M_i)$, with the existence of an inverse mapping g^{-1}, make $\text{dist}(C_i, M_i) = g^{-1}(\text{sim}(C_i, M_i))$. Since the distance is the vector displacement from the starting point to the end, the relationship between distance $\text{dist}(C_i, M_i)$ and similarity $\text{sim}(C_i, M_i)$ is monotonically decreasing.

According to the nearest neighbors similarity idea, the following function can be got:

$$m_{ip_i} = \sum_{i=1}^{m-k} c_{ip_i} \cdot sim(C_i, M_i) / (m - k) \quad (10)$$

C. MINNS-SVM algorithm flow

This algorithm is a combination of support vector machine algorithm with the nearest neighbors similarity idea. On account of missing value imputation, the first step is to scan the data set to classify the complete value and missing value. Next, get data out from the missing dataset circularly and record the location of lost attributes in the acquired missing value. Take the complete dataset out, and then extract the lost attributes as a basis for classification and the remaining data as a training set to be input in the support vector machine. Maximum interval of hyperplane is taken as a classification basis of the support vector machine, and only the inner product between data is used for operation, so multidimensional data have a good fitting ability. Then calculate the similarity of missing value and similar values based on the classification result, as well as the value after imputation based on K neighbors which are

obtained in accordance with the nearest neighbors similarity.

Core pseudo-code of the algorithm:

Inputs:

$X^{original}$: it contains the n × d dataset of the missing value

Outputs:

$X^{predicted}$: n × d dataset after imputation

Step 1: The missing value is separated from the dataset

$X^{original} = X^{complete} \cup X^{missing}$, $X^{complete} \cap X^{missing} = \Phi$

$X^{complete}$: n_c × d matrix without missing value

$X^{missing}$: Each row has a n_m × d matrix of the missing value

Step 2: The missing value of each row is imputedbased on nearest neighbors similarity and support vector machine

fori=1 to n_c

X_i = i-th row of $X^{complete}$ Matrix

$I_{missing}$ = location of the missing value in $X^{missing}$

X_{inputi} = value of X_i without the location of $I_{missing}$

SVM_{train}.add(X_{inputi})

end for

for j=1 to n_m

X_j=i-th row of $X^{missing}$ Matrix

X_j^{target} = SVM.predict(X_j)

for k = 1 ton_c

if$X_k^{target} = X_j^{target}$

Dis_{jk} = STD(X_j ,X_K)

end if

end for

$Sum_{dis} = \sum 1/Dis_{jk}$

$X_j^{predicted} = \sum X_k [I_{missing}]/(Dis_{jk} \times Sum_{dis})$

end for

III. EXPERIMENT

In order to verify the effectiveness of MINNS-SVM algorithm, two comparative experiments are designed in this paper. At the beginning of the experiment, a certain amount of data is randomly extracted from the complete data set, assuming that one or more attribute values is missing, the data set is regarded as the missing value. The classic algorithm—k-means algorithm which is also based on nearest neighbors similarity, and Bayesian network algorithm which is commonly used in probability theory are selected for comparison.

The original value of the missing data is known to assess the continuous effectiveness of the missing value imputation algorithm. One of the important indicators is the error between the value after imputation and the original value, so the most commonly used root-mean-square error is selected as evaluation criteria for the first group of comparative experiments in this paper.

On the other hand, missing value imputation aims to improve the accuracy of machine learning, in which the classification algorithm is extensively used, so the accuracy improvement achieved by the classification algorithm can be taken as evaluation criteria. Through a classification of the data set after imputation and then a comparison with the data without imputation, ACC improvement (simple classification accuracy improvement) percentage can be calculated [12].

$$ACC = \frac{1}{n}\sum_{i=1}^{n} I(\hat{y}_i = y_i) \qquad (11)$$

$$\text{ACC improvement}(\%) = 100 \times \left(\frac{\text{ACC with predicted}}{\text{ACC without predicted}} \right) \qquad (12)$$

Where I () is a judgment function. If the condition is true, return to 1, otherwise return to 0.

In order to fully test the effectiveness of the algorithm for missing value imputation in this paper, five percent of data is firstly extracted from four data sets as the missing value, with five percent increments until the missing ratio reaches 90%. For each data set, each missing value is randomly selected in order to test the accuracy of the results, and the same experiment is repeated a thousand times to take the average value as the result. Experimental hardware environment is: Intel (R) Core ™ i3-3240 @ 3.4GHZ, 4GB of memory; software environment is: Windows8 operating system.

A. Datasets

Four real-life data sets are selected from the UCI machine learning library. In order to have an accurate test data for comparison, the selected data sets are complete without missing value, but a certain proportion of missing values will be randomly selected in the experiment. Four data sets are respectively: Iris Dataset, Mushroom Dataset, Steel Plates Faults Dataset and Breast Cancer EisconsinDataset. Table 1 shows the basic information of four data sets, all of which are selectedfrom different perspectives of consideration. From the perspective of the sample size of datasets, Iris, Breast Cancer Eisconsin have a smaller data size, while Mushroom has a larger data size; from the perspective of the dimension of datasets, the dimension of Iris is lower, while the dimensions of Mushroom, Steel Plates Faults, Breast Cancer Eisconsin are higher. These data are collected from reality with authenticity.

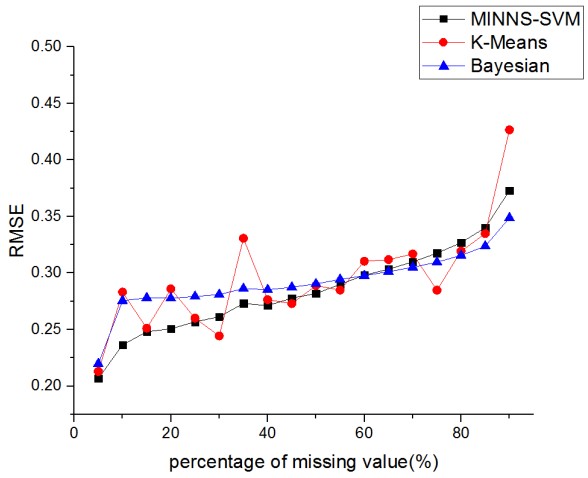
TABLE 1 BASIC INFORMATION OF DATASETS

Dataset	Sample number	Dimension
Iris	150	5
Mushroom	8124	22
Steel Plates Faults	1941	27
Breast Cancer Eisconsin	569	32

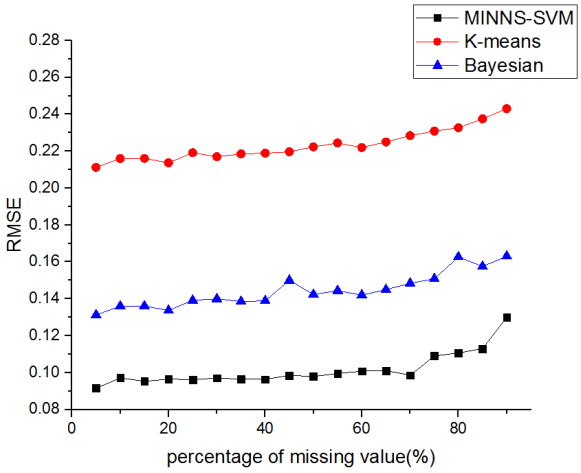
B. RMSE Assessment Results

Root-mean-square error formula:

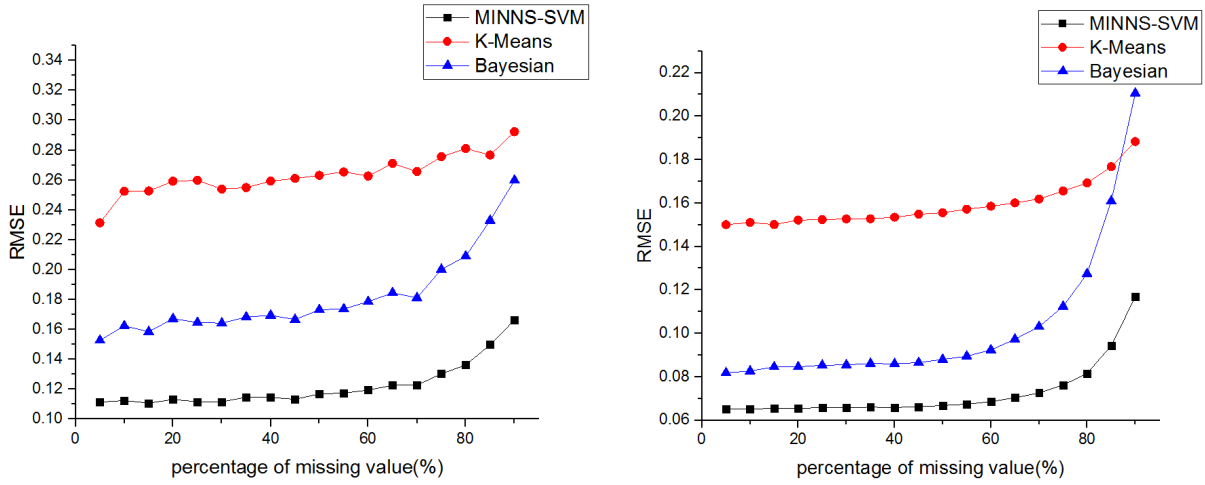
$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{n}\left(\frac{\hat{y}_i - y_i}{y_i}\right)^2}$ Where N = n-1, n is the number of missing value, y_i is the original value, and $\hat{y}_i$ is the imputation value.



Iris Dataset



Mushroom Dataset



Steel Plates Faults Dataset

Breast Cancer Eisconsin Dataset

Fig.1 comparison chart of four data sets' root-mean-square error after completion

It can clearly be seen from Fig.1 that the algorithm proposed in this paper is better than the k-means and Bayesian missing value imputation in most cases, and the accuracy of k-means algorithm even shows instability in some specific data sets; while MINNS-SVM algorithm is stable, particularly in the three high-dimensional datasets, the root-mean-square error of MINNS-SVM algorithm is smaller obviously, indicating that MINNS-SVM algorithm is feasible and effective for missing value imputation.

C. *Classification Algorithm Assessment Results*

Common K-Nearest neighbor regression and Artificial Neural Network (ANN) in the field of machine learning are adopted by the classification algorithm [13].

Core idea of K-NN algorithm is to select k nearest neighbors from the feature space as the sample, if a majority of the selected neighbors belong to certain category, the sample can be divided into this category:

$$\hat{y}_{n+1} = \arg\max_j \sum_{y_i=j} w_i \qquad (11)$$

ANN is one of the most widely used machine learning algorithms. Classification is determined based on the input attributes and their weights:

$$y_k = \sum_{q=1}^{h} w_{kq}^{(2)} g\left(\sum_{r=1}^{d} w_{qr}^{(1)} x_r\right), k = 1,2,\dots,d \qquad (12)$$

TABLE 2 ACC IMPROVEMENT (%) OF K-NN CLASSIFICATION ALGORITHM

DataSet	Imputation	5	10	15	20	25	30	35	40	45	50
iris	k-means	0.90	0.88	0.95	0.96	1.08	1.11	1.13	1.15	1.31	1.42
	Bayesian	1.03	1.09	1.16	1.19	**1.29**	1.35	1.49	1.54	1.61	1.67
	MINNS-SVM	**1.03**	**1.10**	**1.17**	**1.25**	**1.29**	**1.42**	**1.51**	**1.61**	**1.75**	**1.88**
Mushroom	k-means	0.83	0.91	0.92	0.92	0.95	1.01	1.21	1.31	1.38	1.42
	Bayesian	1.02	1.04	1.07	1.09	1.13	1.16	1.21	1.26	1.32	1.39
	MINNS-SVM	**1.03**	**1.07**	**1.10**	**1.17**	**1.25**	**1.39**	**1.36**	**1.51**	**1.71**	**1.76**
Steel Plates Faults	k-means	0.81	0.82	0.82	0.85	0.82	0.85	0.92	0.90	0.93	0.92
	Bayesian	1.00	1.02	1.04	1.03	1.05	1.06	1.09	1.15	1.14	1.20
	MINNS-SVM	**1.04**	**1.10**	**1.15**	**1.20**	**1.28**	**1.39**	**1.46**	**1.59**	**1.74**	**1.87**
Breast Cancer Eisconsin	k-means	1.04	1.08	**1.14**	1.19	1.25	1.31	1.40	1.46	1.57	**1.72**
	Bayesian	0.95	0.92	0.95	0.98	1.02	1.10	1.18	1.32	1.31	1.42
	MINNS-SVM	**1.05**	**1.09**	**1.14**	**1.21**	**1.27**	**1.32**	**1.41**	**1.48**	**1.60**	**1.72**

TABLE 3 ACC IMPROVEMENT (%) OF ANN CLASSIFICATION ALGORITHM

DataSet	Imputation	5	10	15	20	25	30	35	40	45	50
iris	k-means	0.92	0.88	0.92	0.96	1.02	1.11	1.16	1.18	1.33	1.52
	Bayesian	**1.04**	1.10	1.16	**1.25**	1.32	1.40	1.45	1.61	**1.78**	1.88
	MINNS-SVM	**1.04**	**1.11**	**1.17**	**1.25**	**1.33**	**1.43**	**1.52**	**1.67**	1.77	**1.97**
Mushroom	k-means	0.88	0.91	0.91	0.90	0.93	1.02	1.15	1.33	1.38	1.42
	Bayesian	1.01	1.04	1.06	1.09	1.17	1.18	1.19	1.27	1.36	1.34
	MINNS-SVM	**1.04**	**1.08**	**1.13**	**1.19**	**1.25**	**1.34**	**1.42**	**1.53**	**1.63**	**1.79**
Steel Plates Faults	k-means	0.81	0.83	0.85	0.85	0.83	0.86	0.99	0.95	0.95	0.95
	Bayesian	1.01	1.04	1.03	1.04	1.07	1.10	1.24	1.13	1.14	1.42
	MINNS-SVM	**1.04**	**1.09**	**1.15**	**1.20**	**1.31**	**1.40**	**1.41**	**1.57**	**1.70**	**1.79**
Breast Cancer Eisconsin	k-means	1.04	**1.08**	**1.14**	1.19	**1.26**	**1.33**	1.40	1.49	1.58	**1.72**
	Bayesian	0.90	0.92	0.95	0.93	1.01	1.12	1.18	1.35	1.36	1.52
	MINNS-SVM	**1.05**	**1.08**	**1.14**	**1.20**	**1.26**	**1.33**	**1.41**	**1.52**	**1.60**	1.71

As can be seen from Table 2 and Table 3, all the methods for missing value imputation can improve the accuracy of classification. MINNS-SVM is better than traditional k-means and Bayesian missing value imputation in most cases. On the low-dimensional Iris Dataset, compared to k-means algorithm, MINNS-SVM algorithm has increased 5% on average; compared to Bayesian algorithm, the accuracy has increased 3%. But the average increasing rate of high-dimensional datasets of Mushroom, Steel Plates Faults and Breast Cancer Eisconsin reaches 15%, indicating that MINNS-SVM algorithm is effective for the high-dimensional missing value imputation.

IV. CONCLUSIONS

incomplete datasets are common in the practical application of machine learning, and data preprocessing is inseparable from the handling of missing value. Therefore, this paper proposes MINNS-SVM. In order to test the effectiveness of the proposed MINNS-SVM algorithm, four hot data sets in the UCI machine learning database are adopted and used MINNS-SVM algorithm compared with the traditional K-means algorithm and Bayesian algorithm in the experiment. The experimental results show that, the error between the original value and MINNS-SVM algorithm-based value after imputation is smaller. Compared to the data without imputation, the one after imputation has a higher success rate of classification when being used for the classification algorithm in the field of machine learning, particularly for the high-dimensional missing value.

This algorithm still remains several issues to be further studied. First, attributes of the data sets selected in the experiment are totally numerical, so the algorithm in this paper can not handle the datasets which have non-numeric attributes. Second, the use of SVM algorithm significantly increases the running time of algorithm in the case of numerous data sets. Therefore, parallelization is considered to be put into practice with distributed systems to handle large amounts of data in future research work.

ACKNOWLEDGMENT

The work was supported by the research program funded by the **Smart City** Research Program through the Province Research Foundation of GuangDong,China (No. 2013B090500030).

REFERENCES

[1] P. J. García-Laencina, J.-L. Sancho-Gómez, A. R. Figueiras-Vidal,M. Verleysen.K nearest neighbours with mutual information for simultaneous classification and missing data imputation [J]. Neurocomputing, 2009, 72(7):1483-1493.

[2] P. R. Rosenbaum,D. B. Rubin.The central role of the propensity score in observational studies for causal effects [J]. Biometrika, 1983, 70(1):41-55.

[3] BU Fan-yu,CHENZhi-kui,ZHANG Qing-che..Incomplete Big Data Imputation Algorithm Based On Deep Learning [J]. Microelectronics & Computer.2014(12):173-176

[4] R. J. Little,D. B. Rubin.Statistical analysis with missing data [J]. 2002.

[5] A. R. T. Donders, G. J. van der Heijden, T. Stijnen,K. G. Moons.a gentle introduction to imputation of missing values [J]. Journal of clinical epidemiology, 2006, 59(10):1087-1091.

[6] H.Chiu,J.Sedransk.A Bayesian procedure for imputing missing values in sample surveys [J]. Journal of the American Statistical Association, 1986, 81(395):667-676.

[7] DeyS.Bayesian Estimation of the Parameter and Reliability Function of an Inverse Rayleigh Distribution. Malaysian J.of Mathematical Sciences . 2012.

[8] S. J. Rizvi,J. R. Haritsa.Maintaining data privacy in association rule mining:Proceedings of the 28th international conference on Very Large Data Bases,2002[C].VLDB Endowment:682-693

[9] A. N. Baraldi,C. K. Enders.An introduction to modern missing data analyses [J]. Journal of School Psychology, 2010, 48(1):5-37.

[10] DAI Liang,XU Hong-ke,CHENTing,QIAN Chao. Least squares support vector machine regression model based on MapReduce [J]. Application Research of Computers,2014(12): 1060-1064

[11] DING Shi-fei, QI Bing-juan,TAN Hong-yan.An Overview on Theory and Algorithm of Support Vector Machines[J]. Journal of University of Electronic Science and Technology of China,, 2011(01):2-10.

[12] P. Kang.Locally linear reconstruction based missing value imputation for supervised learning [J]. Neurocomputing, 2013, 118(65-78).

[13] T. Cover,P. Hart.Nearestneighbor pattern classification [J]. Information Theory, IEEE Transactions on, 1967, 13(1):21-27.