SKNN Algorithm for Filling Missing Oil Data Based on KNN

Mei Wang1,a, Dong Li1,b, Kaiyuan Qi1,c, Chenglong Xue1,d, Erlong Yang2,*

1School of Computer and Information Technology, Northeast Petroleum University
China
2Key Laboratory of Enhanced Oil Recovery in Ministry of Education, Northeast
Petroleum University Daqing, Heilongjiang, China

*Corresponding author e-mail: yangerlong@nepu.edu.cn, awangmei@nepu.edu.cn,
blidong_nepu@163.com, cnepu1960@yeah.net, d1578444251@qq.com

Abstract. Along with the rapid development of science and technology information in
contemporary society, massive data has become a common phenomenon in information
processing in various industries, and various data quality problems have also followed.
Among them, data loss is a common problem. In the process of oilfield production, the
dynamic data of production wells is increasing continuously every day, and data missing
problems often occur. Aiming at the missing data, this paper proposes an improved data
filling algorithm SKNN based on KNN. Based on the KNN algorithm, the algorithm
uses the SMOTE algorithm to randomly generate the fill data range, and uses the multi-
filling idea to find the weighted average as the fill data. The experiment with the
production data of a well area in Daqing oilfield as the sample set verified that the
SKNN algorithm could not only fill the data, but also improve the accuracy.

1. Introduction

With the repudiation, diversification and automation of data collection methods in various industries,
the amount of data is rapidly increasing, and the data calculation can reach the PB level. The increase
in the amount of data also produces various data quality problems, including data loss is a serious
problem cannot be ignored. Manual entry errors, improper operation of operators, constraints on
collection conditions, improper measurement methods, etc, will result in data loss. The proportion of
data missing in some areas is even large, such as insurance industry, communication industry and
medicine industry, etc. Due to the irregular storage of data records in these industries, the data loss rate
can reach more than 60% [1]. Similarly, data loss is very serious in the huge amount of data generated
by the oil industry.

Data loss is an urgent problem in data analysis of various industries. High-quality data source is the
primary condition for data analysis success. Data missing is the missing data source data tuples or
attributes. The processing of missing data has delete tuples, ignoring processing and filling method [2-
4]. With the rise of the big data era, the extraction of potential information from a large amount of data,
or the classification and clustering, etc, all require a complete data source. Missing data is not only
related to the use of the user and data analysis, but also related to data consistency, data integrity and
data accuracy. Therefore, in the process of data quality management, filling the missing data is the best
missing value processing solution.
Traditional missing data filling methods include special value filling or mean filling [5]. Researchers have proposed a series of complex and accurate methods for various data sources, such as filling missing values based on non-parametric density estimation kernel method Algorithm (DAIM) [5], Bayesian network based probability filling method [6], optimal hybrid replacement filling algorithm (OMI) [7], etc. These methods are based on specific data from different angles using different techniques to solve the problem of missing data. In addition, there is a more common filling processing method, namely KNN filling method.

The KNN algorithm belongs to the local-based single-fill algorithm. The principle is that the two most recent samples have the most similar properties. If a certain attribute data of a sample is missing, the same attribute value of the K-nearest neighbor sample can be used to infer the missing value. However, the KNN algorithm does not consider the positional relationship between each sample point, and the accuracy of the selected K-nearest neighbor samples is not high. There are many researches and improvements on the shortcomings of the KNN algorithm, such as the CNN method proposed by Huang Liangchang et al. [8], the APT_KNN method proposed by Xu Yu Ming et al. [9] and the MKNN method proposed by Yang Tao et al. [10]. These algorithms all improve the distance of KNN algorithm to select neighbor samples and obtain better results.

In data processing, there are few ideal data sets for equalization, most of which are unbalanced data sets. SMOTE algorithm is a new type of oversampling technology, which synthesizes new minority samples through linear interpolation through existing minority samples. It is different from the traditional oversampling technique based on simple copying of a few samples, which avoids the over-fitting problem caused by the small sample distribution area [11-12]. However, the SMOTE method has the disadvantages of classification boundary blurring and noise error generation [13]. There are many improvements to the shortcomings of SMOTE, such as the Improved Smote algorithm and the R-Smote algorithm proposed by Yi Wei et al. [14], and the SD-ISMOTE algorithm proposed by GU Ping Et Al. [15]. These algorithms improve the SMOTE algorithm selection area interpolation and boundary definition, which solves the shortcomings of SMOTE algorithm.

In this paper, based on KNN algorithm and SMOTE algorithm, an improved SKNN algorithm is proposed. In the SKNN algorithm, the K-nearest neighbor samples of KNN algorithm are firstly selected. Then, the SMOTE algorithm is used to randomly generate neighbor samples, and the missing values are filled by the multi-filling idea. The position distribution of neighbor samples and the overall effect of single-filling are improved.

Section 2 of this paper describes the idea and steps of SKNN algorithm in detail, section 3 proves the effectiveness of the algorithm through experiments, and section 4 summarizes the whole paper and proposes the next step.

2. SKNN ALGORITHM
SKNN algorithm is a combination of KNN algorithm for filling the missing data set and SMOTE algorithm for dealing with unbalanced data set, and adopts the idea of multiple filling to fill the missing data. Suppose the given training data set is \( T \)

\[
T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}.
\]

\( x_i \in \mathcal{X} \subseteq \mathbb{R}^m \) is the input vector that the first and second dimensions of the input vector are position information. \( y_i \in \mathcal{Y} \subseteq \mathbb{R} \) Is the output. \( i = 1, 2, \cdots, n \).

2.1. Neighborhood Determination of KNN Algorithm
The traditional KNN algorithm [16-17] finds the \( k \) neighbors of the sample \( x \) in the data set \( T \) according to the given distance metric, and the \( k \) neighbor fields of \( x \) are denoted as \( N_k(x) \). Identify the category \( y \) of \( x \) in \( N_k(x) \) according to the majority decision principle. The distance metric is generally Euclidean distance, which is a special \( L_p \) distance.
Feature space \( \mathcal{X} \subset \mathbb{R}^m, x_i, x_j \in \mathcal{X} \), \( L_p \) distance is defined as

\[
L_p(x_i, x_j) = \left( \sum_{l=1}^{m} |x_i^{(l)} - x_j^{(l)}|^p \right)^{\frac{1}{p}},
\]

And it's called the Euclidean distance while \( p = 2 \).

The disadvantage of the KNN algorithm is that it does not consider the distance between the missing point and the nearest neighbor point and the positional relationship between them. This may result in the accuracy of the selected neighbor samples is not high. In view of this shortcoming, the random interpolation idea of SMOTE algorithm is introduced. Meanwhile, only the first two dimensional attributes \((x_1, x_2)\) of the missing sample \( x \) are selected for calculating Euclidean distance. Picking \( k \) neighbors is within a given limit distance: In the data set \( T \), calculate the Euclidean distance using the missing data sample position information \((x_1, x_2)\) and other samples position information \((x_i^{(1)}, x_i^{(2)})\)

\[
L = \sqrt{(x_1^{(i)} - x_1)^2 + (x_2^{(i)} - x_2)^2}
\]

The upper limit of the distance given by the empirical theorem is denoted by \( M \), and \( k \) neighbors of \( L < M \) are selected from the calculated Euclidean distance set from small to large.

2.2. Interpolation Calculation Based on SMOTE

The SMOTE algorithm [18-23] is an oversampling method proposed by Chawla for unbalanced data sets. The main idea is to randomly generate \( k \) new minority samples by linear interpolation between a minority class samples and \( k \) nearest neighbor samples to achieve the purpose of balancing the data set.

For minority classes \( T_s \) in the data set \( T \), calculate the Euclidean distance from the sample \( x \) in \( T_s \) to other minority samples, and obtain the \( k \) nearest neighbor samples with the smallest distance. Set a sampling rate \( w \) based on the sample imbalance ratio. For each minority sample \( x \), randomly select \( w \) samples from its \( k \) nearest neighbors, denoted as \( x_1, x_2, \ldots, x_w \). Random linear interpolation between minority classes \( x \) and \( x_i (i = 1, 2, \ldots, w) \) to construct a new minority sample \( t \)

\[
t_i = x + \text{rand}(0,1) \times (x_i - x), i = 1, 2, \ldots, w.
\]

\( \text{rand}(0,1) \) Represents a random number in the interval \((0, 1)\). Adding a newly synthesized minority sample \( T_i \) to the data set \( T \) will result in a new and more balanced training set. A new sample of minority class is generated as shown in Fig.1.
This paper is based on the neighborhood determination of the KNN algorithm and is improved by the SMOTE algorithm. Firstly, \( w (0 < w \leq k) \) neighboring points are randomly selected from \( k \) neighbor samples, and the data of neighboring points are read corresponding to the missing values, which are denoted as \( A_1, A_2, \ldots, A_w \), and the missing value is \( A_i \). Then, using the SMOTE linear random interpolation method to generate \( w \) new neighbor data points value, \n
\[
value_i = A_i + \text{rand}(0,1) \times (A_i - A_i), i = 1, 2, \ldots, w, A_i = 0. \tag{4}
\]

Calculate the weighted average \( W \) of \( w \) new neighbor data points as candidate for missing data

\[
W = \frac{\sum_{i=1}^{w} value_i}{w} \tag{5}
\]

Repeat the above random selection interpolation process \( w \) times to obtain \( w \) missing value candidate values \( W \). Finally, using the multi-filling idea, the weighted average of \( w \) candidate values

\[
V = \frac{\sum_{i=1}^{w} W_i}{w} \tag{6}
\]

Is calculated as the last missing padding value.

2.3. SKNN Algorithm Steps

The input of the SKNN algorithm is a data missing set \( T \) containing \( n \) rows and \( m \) columns of test data, an initial value \( k \) of the neighbor, an empirical distance \( M \) given by the expert, and finally a padding value of the missing data \( V \).

First of all, to initialize. Initialize the sample location table \( p(\text{key}, PX, PY) \), where \( \text{key} \) is the sample, \( PX \) is the X-axis of the sample, and \( PY \) is the Y-axis of the sample. Initializes array \( h[1] \) to store the location of the sample \( x \). Initialize array \( d[1,n] \), which \( n \) is the total number of samples in data set \( T \), store the Euclidean distance from sample \( x \) to each sample, and note that the first value in \( d[1,n] \) is the distance from sample \( x \) to itself, and the value is 0, which does not participate in the subsequent
calculation process. Initialize array \(\text{list}[k-1]\), store and randomly select \(k - 1\) neighbor points in \(k\) neighbor points. Initializes the sum of the nearest neighbor points corresponding to the missing values \(\text{sumitem} = 0\). Initializes the number of neighbor points \(\text{nzeronum} = 0\) corresponding to missing values that are not empty. Initialize the sum of missing candidate values \(\text{sumitemall} = 0\).

The main steps of the SKNN algorithm include:

1) Read each row and column of data in the dataset. If the data is missing as NULL, skip to step (2). Otherwise, continue to loop until the dataset is traversed.

2) Read all sample position information in the data set \(T\) to form a table \((, ,)\)

3) Calculate the Euclidean distance \(L\) between sample \(x\) and other samples in table \(P\), and sort the samples in the order from near to far: Store the position coordinates of the sample \(x\), \(h = [x^{(1)}, x^{(2)}]\).

4) According to the given empirical distance \(M\), the first \(k\) neighbors with the Euclidean distance \(L < M\) and \(L = 0\) are selected from the samples ordered by \(d[1, n]\), and \(k - 1\) of the \(k\) nearest neighbors are randomly selected and stored in the array \(\text{list}[k-1]\).

5) The data corresponding to the missing position of the sample in \(\text{list}[k-1]\) is read from the data set \(T\), denoted as \(A_1, A_2, \cdots A_{k-1}\), and \(A_i\) is the sample \(x\) missing value. Using the SMOTE random interpolation idea, each nearest neighbor computes a missing approximation \(\text{value}_i\)

6) Calculate the candidate values of the missing data, and use the weighted average of the \(k - 1\) neighbors data \(\text{value}_i\) as the candidate for the missing data

7) Loop steps (4), (5), (6) and get missing data candidate values.

8) The weighted average of the missing data candidate values \(W_i\) of \(k - 1\) neighbors is used as the padding data for the missing data

9) Fill the calculated missing data \(V = \text{sumitemall}\) into the data set table \(T\).

The SKNN algorithm performs multiple random padding generation on the KNN algorithm to fill the missing data, and then weights the data generated multiple times as the padding value of the last missing data. The SMOTE algorithm idea and the multi-filling idea make up for the shortcomings of the KNN algorithm for the inaccurate filling data caused by the uneven distribution of sample points.

2.4. Algorithm Complexity Analysis

The number of samples in the dataset is \(n\). The number of attributes is \(m\), and the time complexity of reading all data is \(O(n \times m)\). Euclidean distance time complexity is \(O(n)\). Euclidean distance sorting time
complexity is $O(n)$. The time complexity of the fill candidate generated by the loop calculation is $O((k-1)*(k-1))$. Time complexity of weighted average of missing candidate values is $O(k)$.

In summary, the time complexity of the algorithm used in this paper is $O(n^2\cdot (\mathcal{O}(n)+\mathcal{O}(n)+\mathcal{O}(k-1)*(k-1)+\mathcal{O}(k-1)))$.

The time complexity is $O(n^2)$ while $n >> k$ and $n >> m$.

3. Experimental results and analysis

3.1. Data Set

Numerical data of dynamic production wells are generally discontinuous. In this paper, partial production data of a well area in Daqing oilfield are selected as experimental data, and it is found that frequently missing values include top depth of sandstone, various pressure values, water content values and ground elevation, etc. These data must have a presence value when reading the analysis. At the same time, the data cannot be filled in at will, otherwise it will have an error effect on the subsequent operations. The data of oil wells and wells in the two blocks in the above well area are selected, including 910 wells, including 603 oil wells and 297 wells. Each well has about 30 layers.

The three attribute sets of fracture pressure, simulated water production and ground elevation in the dynamic well production data of the oilfield as the data set to be tested. The data table information is shown in TABLE 1.

### Table 1. Data set description of the well table

| Well  | Block | Layer | X   | Y   | Fracture Pressure | Simulated Water Production | Ground Elevation |
|-------|-------|-------|-----|-----|-------------------|---------------------------|-----------------|
| G237-33 | a     | G12a  | 22753.0 | 55655.0 | 13.66             | 2.7                       | 150.33          |
|       | a     | G12b  | 22975.0 | 55649.0 | 13.66             | 2.9                       | 150.23          |
| G231-37 | a     | G35   | 23222.0 | 55661.0 | 13.67             | 2.7                       | 150.81          |
|       | a     | G36   | 23777.0 | 55449.0 | 13.67             | 0.3                       | 150.36          |
| G238-36 | b     | G21   | 24298.0 | 55203.0 | 13.67             | 0.2                       | 151.26          |
| G231-38 | b     | G32   | 24631.0 | 55151.0 | 13.68             | 0.1                       | 150.95          |
| G133-25 | a     | G33   | 24836.0 | 55206.0 | 13.68             | 0.1                       | 150.47          |
| G129-28 | b     | G32   | 25901.0 | 54956.0 | 13.68             | 1.3                       | 150.08          |
| ...    | ...   | ...   | ...    | ...    | ...               | ...                       | ...             |

In order to avoid the calculated filling data without reference, the selected data set does not have missing data, which is a complete data set. In the experiment, different proportions of missing data are artificially produced on different test attribute sets.

3.2. Experimental Method

For the complete production data of the above wells, artificially randomly manufactured different proportions of missing data for each test attribute set, the ratio is from 10% to 40%, and keeps other information that is not the test attribute data intact. In this experiment, KNN algorithm and SKNN algorithm are used to fill the missing data in different proportions of data sets. It is known from expert
experience that in the same well group, the well spacing is generally maintained at about 125 m, and the number of wells is generally maintained at about 3 wells. For each experiment of the KNN algorithm, the wells with missing data select \( k \) neighboring wells in the same well zone according to the Euclidean distance of the location, and the value of \( k \) is 3. For each experiment of SKNN algorithm, each well has the position coordinate \((x^{(1)}, x^{(2)})\) of the well area. Calculate the Euclidean distance \( L \) of the missing data well and its surrounding wells and sort them, and find \( k \) nearest neighbors according to the Euclidean distance \( L < 125 \text{ m} \), \( k \) is taken as 5. SMOTE is then used to generate random linear interpolation from \( k \)-neighbor and select \( w \) samples. The corresponding data of 3 wells are randomly interpolated from 5 neighboring wells. Each time the data is calculated, and it is denoted as equal to the data in the selected neighbor well. \( \text{rand}(0,1) = 1 \) is calculated to get value \( i \), \( \text{value}_i = A \). Finally, 3 values close to the missing data were obtained, and the weighted average of the obtained 3 values was used as candidate value for filling data. This process is repeated 3 times, and then 3 candidate fill values are obtained. The weighted average of 3 values is calculated as the filling value of the missing data. This calculation avoids the uncertainty of a single fill and increases the accuracy of the fill value.

The well spacing of dynamic Wells in the same well group in the well area is not always within 125 m, because of geographical location constraints, there are a few Wells with the well spacing outside 125 m. In this case, the nearest well to the missing data well is selected and the corresponding data of the nearest well is taken as the filling data. This method is called the nearest neighbor algorithm (1NN). Both 1NN algorithm and SKNN algorithm have poor results, which need to be improved in the future.

3.3. Experimental Results
For KNN and SKNN both missing value filling methods, filling accuracy is used to calculate the filling effect, and the calculation method is as follows

\[
TP = \frac{tn}{tm} \quad (tm > 0)
\]

Where \( tn \) is the correct number of filled values, \( tm \) is the number of missing data, and \( TP \) is the filling accuracy. The fill value error range is controlled at \([-1,1]\). For the Wells with dynamic well spacing beyond 125 m in the well group in the experimental well area, the expert introduced that this situation is rare, so it is also included in the calculation range. The experimental results of missing data filling accuracy in different proportions in different attribute sets are shown in TABLE II and from Fig. 2 to Fig. 4.

| Filling Algorithm | Missing Proportion | Fracture Pressure Accuracy | Simulated Water Accuracy | Ground Elevation Accuracy |
|-------------------|--------------------|----------------------------|--------------------------|--------------------------|
| KNN               | 10%                | 89%                        | 23%                      | 76%                      |
|                   | 20%                | 87%                        | 37%                      | 63%                      |
|                   | 30%                | 85%                        | 26%                      | 57%                      |
|                   | 40%                | 86%                        | 34%                      | 49%                      |
| SKNN              | 10%                | 90%                        | 67%                      | 89%                      |
|                   | 20%                | 91%                        | 71%                      | 93%                      |
|                   | 30%                | 95%                        | 75%                      | 91%                      |
|                   | 40%                | 95%                        | 83%                      | 95%                      |
Figure 2. Filling accuracy of the two algorithms in the fracture pressure attribute set

Figure 3. Filling accuracy of two algorithms in simulated water production attribute set

Figure 4. Filling accuracy of the two algorithms in the ground elevation attribute set
Fig. 2 to Fig. 4 shows the filling accuracy of the two filling algorithms under different missing proportions of each attribute set of dynamic well production data. According to the filling accuracy of the two filling algorithms in each attribute set, it can be seen that the accuracy of the improved algorithm SKNN is higher than that of the KNN algorithm.

4. Conclusion
Based on the KNN algorithm, this paper introduces the SMOTE algorithm and multiple filling ideas to form a new missing data filling algorithm SKNN. The SKNN algorithm can solve the continuous and discrete data problems and effectively improve the accuracy of the KNN algorithm. Experimental results show that for the problem of data missing in dynamic well of oil field, the accuracy of SKNN algorithm data filling is generally better than that of the classical algorithm KNN when the data missing ratio of various attribute data is different. However, the time complexity of the SKNN algorithm is complicated. As the amount of data increases, the algorithm runs for a long time. At the same time, the SKNN algorithm cannot solve the non-numeric data well. This is a problem to be solved in the future.

Acknowledgments
We acknowledge support from the National Natural Science Foundation of China under Grants 51774090 and 51574085 and the Science Foundation of Heilongjiang Province under Grants E2016008 and F2016002.

References
[1] Xing Zhao, Xun Wang, Shucheng Huang, “Missing value imputation algorithm based on distance maximization and missing data clustering,” Electronic Design Engineering, 2018, 26 (01): 20 - 24+28.
[2] Xingyi Liu, Guocai Nong, “Comparing Several Popular Missing Data Imputation Methods,” JOURNAL OF NANNING TEACHERS COLLEGE, 2007 (03): 148 - 150.
[3] Sen Wu, Xiaodong Feng, Zhiguang Shan, “Missing Data Imputation Approach Based on Incomplete Data Clustering,” CHINESE JOURNAL OF COMPUTERS, 2012, 35 (08): 1726 - 1738.
[4] Zhaoqiang Chen, Jiajun Li, Chuanjiang, Hailong Liu, Qun Chen, Zhanhuai Li, “A Context-Aware Entity Ranking Method for Web-Based Data Imputation,” CHINESE JOURNAL OF COMPUTERS, 2015, 38 (09): 1755 - 1766.
[5] Xiaofeng Zhu, ”Studies on Missing Data Imputation,” Guangxi normal university, 2007.
[6] Jin, ”RESEARCH ON MISSING VALUE IMPUTATION OF INCOMPLETE DATA,” Harbin Institute of Technology, 2013.
[7] Liling Xia, Yuelong Zhu, “Research on Approach Optimization of Missing Water-Quality Data,” Water Resources and Power, 2018, 36 (04): 158 - 161+85.
[8] Liangchang Huang, “the Analysis and Improvement research of KNN-Imputation algorithm,” Guangxi normal university, 2010.
[9] Yuming Xu, Cheng Chen, Yun Xiong, Yangyong Zhu, “APT-KNN: AN EFFICIENT MISSING VALUE IMPUTATION METHOD ORIENTED TOWARD CLASSIFICATION ISSUE,” Computer Applications and Software, 2011, 28 (04): 135 - 139.
[10] Tao Yang, Jiawei Luo, Yan Wang, Junhao Wu, “Missing value estimation for gene expression data based on Mahalanobis distance,” Computer Applications, 2005 (12): 152 - 155.
[11] Zhiying Lu, Yimo Ren, Xiaolei Sun, Huizhen Jia, “Recognition of Short-Time Heavy Rainfall Based on Deep Learning,” Journal of Tianjin University (Science and Technology), 2018, 51 (02): 111 - 119.
[12] Yajun Xu,“ Research on Automatic Recognition of Layer of Neutron Lifetime Logging,” Northeast Petroleum University, 2014.
[13] Longshen Zhong, Xuejun Gao, Zhenyou Wang, “A New Kind of Improving SOMTE Algorithm Based on K-means in Imbalanced Datasets,” Mathematics in Practice and Theory, 2015, 45
(19): 198 - 206.

[14] Wei Yi, Li Mao, Jun Sun, Linhai Wu, “Research on Classification of Improved Smote Algorithm on Imbalanced Datasets,” JISUANJI YU XIANDAIHUA, 2018 (03): 83 - 88.

[15] Ping Gu, Yang Yang, “Oversampling Algorithm Oriented to Subdivision of Minority Class in Imbalanced Data Set,” Computer Engineering, 2017, 43 (02): 241 - 247.

[16] Li Hang. Statistical Learning Method, Beijing: Tsinghua University Press, 2012.

[17] I-Ling Chen, Kai-Chih Pai, Bor-chen Kuo, Cheng-Hsuan Li, “An Adaptive Rule Based on Unknown Pattern for Improving K-Nearest Neighbor Classifier,” Technologies and Applications of Artificial Intelligence (TAAI), 2010 International Conference on, 2010.

[18] Chawla, Nitesh V., et al. "SMOTE: synthetic minority over-sampling technique." Journal of Artificial Intelligence Research 16.1 (2002): 321 - 357.

[19] Zhang, Yongqing, et al. "Improved SMOTE Bagging and its application in imbalanced data classification." IEEE Conference Anthology 2014.

[20] Blagus, Rok, and L. Lus, "Evaluation of SMOTE for High-Dimensional Class-Imbalanced Microarray Data." International Conference on Machine Learning & Applications 2013.

[21] Lina Zhang, Chengjin Zhang, Rui Gao, Runtao Yang, Qing Song, “Using the SMOTE technique and hybrid features to predict the types of ion channel-targeted conotoxins,” Journal of Theoretical Biology, 2016, 403.

[22] Mengjie Zhang. “SMOTE algorithm applying imbalanced data in higher education,” Proceedings of 2nd Information Technology and Mechatronics Engineering Conference (ITOEC 2016), 2016: 4.

[23] Blagus, Rok, and L. Lusa. "SMOTE for high-dimensional class-imbalanced data." BMC Bioinformatics 14.1 (2013): 1 - 16.