

# An Improved Imputation Algorithm Based on QENNI <sup>★</sup>

Zhibo Chen <sup>1,\*</sup>, JianXin Wang <sup>2</sup>, Zhaoyu Zhang <sup>3</sup>, Qiuwu Yuan <sup>4</sup>,

<sup>1</sup>*School of Information Science And Technology, Beijing Forestry University, Beijing 100083, China*

<sup>2</sup>*School of Information Science And Technology, Beijing Forestry University, Beijing 100083, China*

<sup>3</sup>*School of Information Science And Technology, Beijing Forestry University, Beijing 100083, China*

<sup>4</sup>*School of Information Science And Technology, Beijing Forestry University, Beijing 100083, China*

## Abstract

Missing data imputation is an important research aspect in data mining. Data quality is a major concern in Machine Learning and other correlated areas such as Knowledge Discovery from Databases (KDD). Many imputation methods of missing data have been designed to resolve the problem. More or less, they have some deficiencies. As the K-Nearest Neighbor Imputation ( $k$ NNI) Algorithm is often biased in choosing the  $k$  nearest neighbors of missing data. A new imputation method is put forward, Quadrant Encapsidated Nearest Neighbor based Imputation method (QENNI). QENNI uses the quadrant nearest neighbors around a missing datum to impute the missing data value. It is not biased in selecting nearest neighbors. Experiments demonstrate that QENNI is much better than the  $k$ NNI method in imputation accuracy. But, as the experiments proceed, we find out the density of points in each quadrant and the distance between the two point affect the missing data value seriously. So, we improved the QENNI algorithm and put forward Density and Distance Weighted Quadrant Encapsidated Nearest Neighbor based Imputation method algorithm (DDWQ). The experimental result demonstrates that our DDWQ method has a higher imputation accuracy than QENNI.

*Keywords:* Imputation of missing data; Quadrant; Shell-neighbors;  $k$ NNI; QENNI; DDWQ

## 1 Introduction

Data mining is an interdisciplinary subfield of computer science. It is the computational process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems. The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use. Data mining is a powerful technology with great potential. Pre-processing as one of the indispensable step of data mining can seriously affect the accuracy of the conclusion. That is why missing data imputation has been an inevitably and challenging research.

---

<sup>★</sup>Project supported by the National Nature Science Foundation of China (No. \*\*\*).

\*Corresponding author.

Email address: [zhibo@bjfu.edu.cn](mailto:zhibo@bjfu.edu.cn) (Zhibo Chen).

Due to its importance in data mining, missing data imputation has received considerable attention during the past decades. A large percentage of studies[1, 2, 3, 4, 5] have been done to develop procedures to deal with missing values. Recently, no matter in which field of study,  $k$ NNI[7, 8, 9] imputation method has been researched and applied widely because of its easy operating, high efficiency and accuracy. It is an excellent imputation algorithm, but those choosing nearest neighbors of missing data is very likely biased on one side. So it's not the best choice to use them for missing data imputation. In addition, the parameter  $k$  is the key factor for the  $k$ NNI algorithm. In the experiments, if the  $k$  sets a larger value, it brings seriously randomness; if the  $k$  sets a smaller value, it will lose large sample size standard of statistics. Before very experiment of  $k$ NNI, lots of calculation should be token to get the appropriate value of  $k$ . It makes the algorithm more complex. In response to these problems, Shichao Zhang put forward the Shell-neighbors[6] viewpoint of missing data imputation and realize a new missing data imputation algorithm, quadrant encapsidated nearest neighbor based imputation (QENNI). QENNI algorithm imputes the missing data by finding all of the quadrant encapsidated nearest neighbors of the missing data. Exactly to say, it assumes the incomplete data as the center, the complete data sets are distributed to each quadrant. Because of this feature, it can void the heavily depending on the parameter  $k$  of  $k$ NNI. Experimental results show QENNI has a higher accuracy than  $k$ NNI.

But, according to the analysis of the incomplete data, we find it also seriously affected by density of points in each quadrant and the distance between the incomplete data and complete data. So, based on QENNI, we take the density and distance's weight into account and propose a new missing data imputation algorithm, Density and Distance Weighted Encapsidated Nearest Neighbor based Imputation method (DDWQ). This imputation algorithm overcomes the above-mentioned limitations and has a good performance than QENNI.

The rest of this paper is organized as follows: Section 2 introduces the details of the proposed imputation method and give the corresponding algorithm. Section 3 gives the experiments and results. Section 4 discusses the result and draws a conclusion.

## 2 DDWQ Algorithm

On the basis of the above discussion, in this part, we give the definition and implementation of the DDWQ algorithm and point out the improvement of DDWQ algorithm. We also discuss the shortcomings of  $k$ NNI and QENNI algorithm.

### 2.1 Algorithm background

Suppose  $X$  is an  $M$ -dimensional random vector,  $Y$  is the dependent variable affected by  $X$ . In practice, if a missing data random sample (size is  $n$ ) can be get, it can be expressed as  $(X_i, Y_i, \delta_i), i = 1, 2, \dots, n$ . In which, all of the  $X_i$  vector is observable, when  $Y_i$  is missing,  $\delta_i = 1$ , otherwise  $\delta_i = 0$ . If the data set  $T$  contains  $n$  data, each data has  $m + 1$  attributes (contains  $m$  condition attributes and 1 decision attribute), keep:  $T_i = (X_{i1}, X_{i2}, \dots, X_{im}, Y)$  (missing values are generated only in decision attribute  $Y$ ).  $T = I \cap C$ , let  $r = \sum_{k=1}^n \delta_i$ ,  $I = T_1, \dots, T_r, r \leq n$  are the data sets which decision attributes are missing, referred to missing datasets;  $C = T_{r+1}, \dots, T_n$  is the complete datasets.

## 2.2 $k$ NNI algorithm

K-Nearest Neighbor Imputation ( $k$ NNI) Algorithm imputes the missing value by the  $k$  nearest neighbors of the incomplete data. It bases on the theory that the closer the distance, the closer the relation. If a data loses one attribute, to find out the  $k$  nearest neighbors in complete data sets and use the average value of them to impute the missing value.

As previously mentioned,  $k$ NNI is praised by the majority of researchers because of its simple operation, low time complexity and high imputation accuracy. But there are still some drawbacks. The  $k$  nearest neighbors selected by  $k$ NNI algorithm may occur preferences, which makes the filling result relatively inefficient. In fact, from the distribution of complete data which surrounds the missing data, the distribution of its nearest neighbors and overall data may be inconsistent. For instance, as shown in Fig.1,  $O$  represents missing data, the other points represents complete data, we use  $k$ NNI algorithm to impute the missing data value. According to the way of  $k$ NNI, we need to find three nearest neighbors and use them to impute the missing data value. So  $A, B, C$  are selected. Nevertheless, from Fig.1 we can clearly see that the complete data surrounding the missing data are evenly distributed, but the three selected nearest neighbors bias in the first quadrant. So the three nearest neighbors may not be the best choice and we can't get the best results by using them to impute the missing value.

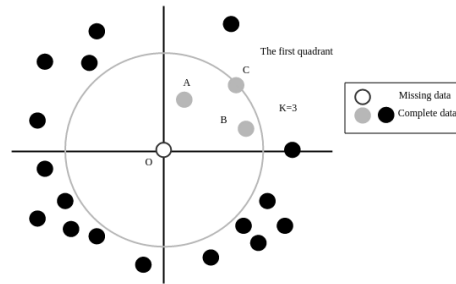


Fig. 1: Nearest neighbors chosen by  $k$ NN

To choose the  $k$  parameter of  $k$ NNI algorithm is difficult. Each time, we use  $k$ NNI to impute the missing data, we need to repeat the experiment so many times to obtain the value  $k$ . Once the value of  $k$  occurs deviation, the performance of  $k$ NNI will be significantly lower. In conclusion, if the imputation algorithm can eliminate the dependence on the parameter  $k$ , it will be the best choice.

## 2.3 QENNI algorithm

We propose the hypothesis that the complete data which is used to impute missing data must be the nearest neighbors and in the first encirclement of the incomplete data. we also need to eliminate the dependence on the parameter  $k$ . Let's realize the idea below.

- (1) First, take data  $(X_1, X_2, \dots, X_m)$  which contains  $m$  condition attributes as a point of  $m$ -dimensional space, establish a coordinate system and the missing data is the center. Through the axis to divide the  $m$ -dimensional into  $2^m$  quadrants.
  - a. When  $m = 2$ , the condition attributes  $(X_1, X_2)$  can be regarded as a point of plane. As shown in Fig.2, the axis divides the plane into  $2^2$  quadrants. If one point is just between

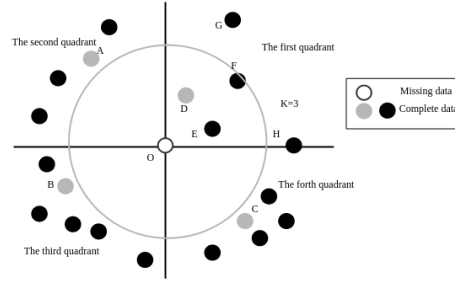


Fig. 2: Nearest neighbors chosen by QENNI

the two quadrants, we classify it to one of its nearby quadrant. According to our definition, each of the complete data can be located in the only certainty quadrant.

- b. Similarly, when  $m = 3$ , the condition attributes of data is  $(X_1, X_2, X_3)$ . The formed spatial coordinate system divides the space into  $2^3$  quadrants and each of the complete data is also in the only identified quadrant.
  - c. Extended to the general case, when  $m = m$ , the condition attributes of data is  $(X_1, X_2, \dots, X_m)$  and the space is divided into  $2^m$  quadrants.
- (2) Based on the dividing of  $m$ -dimensional space, the Euclidean distance of each point (complete data) from its own to the center (which is the incomplete data) is calculated. To find out the nearest one of each quadrant (if not exists, ignore it) and use decision attributes of them to impute the missing data value. With  $m = 2$ , for example, as shown in Fig.2. In each quadrant, we select  $A, B, C, D$  as the nearest neighbors and use the decision attributes  $Y$  of them to impute the missing data value of center  $O$ . It also weighted by the distance of each selected point. Obviously, the result of QENNI to choose the nearest neighbors is different from the  $k$ NNI algorithm.

$$\text{dist}(T_i, T_j) = \sqrt{\sum_{k=1}^m (X_{ik} - X_{jk})^2} \quad (1)$$

- (3) In order to analyze the effectiveness of the algorithm better, for each missing data  $T_i (T_i \in I)$ , on the basis of QENNI algorithm, the following definition can be given:

**Definition 1** The coordinate system centered at  $T_i$  divides the space into  $2^m$  quadrants and the complete dataset  $C$  based on quadrant is divided into  $2^m$  subset  $C = \{D_1, D_2, \dots, D_q, \dots, D_{2^m}\}$ . Each complete data set  $D_q (q = 1, 2, \dots, 2^m)$  is the  $q$  quadrant's data of  $T_i$ .

**Definition 2**  $\forall T_j \in D_q$ , satisfy  $\text{Near}_q = \arg \min_{T_j \in D_q} \text{dist}(T_i, T_j)$ ,  $\text{Near}_q$  is the nearest neighbor of  $T_i$  in  $q$  quadrant. As shown in Fig.2, the first quadrant data of  $T_o$  is  $D_1 = \{T_D, T_E, T_F, T_G, T_H\}$ . So  $\text{Near}_q = T_D$  which is the nearest neighbor of the first quadrant.

**Definition 3** In the  $q$  quadrant, take  $T_i$  as the center of the sphere (or hypersphere),  $\text{dist}(\text{Near}_q, T_i)$  as the radius to ensure the  $\text{Shell}_q$  of  $T_i$  in  $q$  quadrant.

**Definition 4** All of the  $Shell_q (q = 1, 2, \dots, 2^m)$  and axis constitute the  $m$ -dimensional subspace which is Shell of  $T_i$ .

**Definition 5** All the of nearest neighbors  $\{Near_1, Near_2, \dots, Near_{2^m}\}$  of  $T_i$  in each quadrant are called the points of Shell.

**Corollary 1** If  $D_q \neq \emptyset$ , the Shell of  $T_i$  must exists in  $q$  quadrant.

**Corollary 2**  $\forall T_j \in D_q, \exists dist(T_j, T_i) \geq dist(Near_q, T_i)$ .

- (4) In summary, the QENNI algorithm can overcome the shortcomings of  $kNNI$  in choosing the  $k$  nearest neighbors of missing data. The selected complete data is not biased in any side and the Shell of the missing data is smallest. That is to say it does not exist any other complete data on the Shell of the missing data. It is thus clear that QENNI algorithm can find out the most satisfied complete data without any preferences. They can represent the missing data value better than  $kNNI$ .

## 2.4 DDWQ aglorithm

Even though the QENNI algorithm has a better performance than  $kNNI$ , it does not take the density of complete data in each quadrant into account. As shown in Fig.3, if we just think about the distance of complete data  $A$  and  $D$ , the  $D$  has a higher effect of the result. But it is clearly that just the distance can not represents the real effect of the complete data  $A$ . If the density of complete data surrounding  $A$  is token into account, it may have a higher weight than  $D$ . Based on the above considerations, our new algorithm takes both the weight of distance and density into account. The following definitions are given.

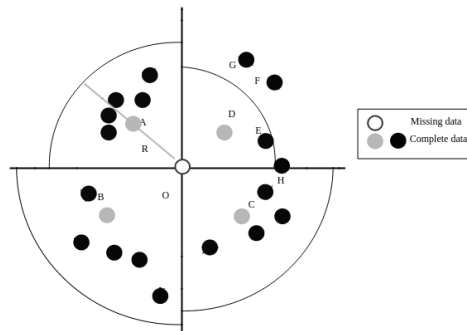


Fig. 3: Nearest neighbors chosen affected by Density

- (1) Weight of Distance : The different length of distance has different size of influence. The more closer, the higher weight. So we keep weight of each nearest neighbor  $Near_i (i = 1, 2, \dots, 2^m)$  in every quadrant as:

$$W(D)_i = \frac{1}{dist(Near_i, T_i)^2} \quad (2)$$

- (2) Weight of Density : The density represents the number of complete data in per unit volume and it can indicate the wight of the quadrant's density. We use Eq. (3) to calculate the volume of n-sphere. The Eq. (5) is used to calculate the density which can represents the weight. Each quadrant's number of complete data in the limited volume of n-sphere which has  $R \leq 2 * dist(Near_i, I_i)$  is counted as  $N = \{N_1, \dots, N_i, \dots, N_{2^m}\}$ .

$$V_n(R) = \frac{\pi^{\frac{n}{2}} R^n}{\Gamma(\frac{n}{2} + 1)} \quad (3)$$

$$\Gamma(\frac{n}{2} + 1) = \begin{cases} (\frac{n}{2})! & n \text{ is even,} \\ \sqrt{\pi} \frac{n!!}{2^{\frac{n+1}{2}}} & n \text{ is odd.} \end{cases} \quad (4)$$

$$W(\rho)_i = \frac{N_i}{\frac{V_n(R)_i}{2^m}} \quad (5)$$

- (3) Core algorithm : The algorithm takes both density and distance's weight into consideration and bring in the factor  $\beta(0.0 \leq \beta \leq 1.0)$  to decide the percentage of density and distance. The followings are the detailed steps of DDWQ :

- a. Calculate the Euclidean distance of missing data set  $I$  and complete data set  $C$  by Eq. (1).
- b. Iterate data set  $I$  and  $C$ , calculate the nearest neighbors, total number of complete data in each quadrant where  $R \leq 2 * dist(Near_i, I_i)$  and volume of each hyperspace by Eq. (2, 3, 5).
- c. The key about how to figure the complete data for each quadrant is to translate the data vector into a number and use it to mark the quadrant. First, translate the coordinate into a vector consisting of one and zero. if the value of vector greater than zero, translate it into one, otherwise keep it as zero. Than, translate the binary number into decimal number and use it to mark the quadrant.

$$\begin{pmatrix} 2 & -1 & 0 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \Rightarrow (1 * 2^2 + 0 * 2^1 + 0 * 2^0) \Rightarrow 4 \quad (6)$$

- d. Take the previous results into Eq. (7) to calculate the missing data value ( $v_i$ ). The equation has token the weight of density and distance into account.

$$v_i = \frac{\sum_{i=1}^n ((1 - \beta)W(D)_i + \beta W(\rho)_i) * D_i}{\sum_{i=1}^n ((1 - \beta)W(D)_i + \beta W(\rho)_i)} \quad (7)$$

## 3 Experiments and Results

### 3.1 Dataset

In order to test our algorithm, we choose the open datasets Abalone which is from UCI and Delta\_ailerons which is from weka. We do the same experiments as QENNI does. The sex

attribute is excluded and left the data whose sex value is  $M$ . It has 8 attributes and 1528 records in total. The Diameter attribute is chosen as decision attribute and missing data is generated on it. The Delta\_ailerons dataset has 6 attributes and 7129 records in total. The last attribute is chosen as decision attribute and the missing data is generated on it. The imputation accuracy is used as evaluation index. In general, the Root Mean Square Error (RMSE)[10] is used.  $e_i$  is the original value,  $e'_i$  is the imputation value and  $m$  is the number of missing data. The smaller the RMSE, the higher the imputation accuracy.

$$RMSE = \frac{1}{m} \sum_{i=1}^m (e_i - e'_i)^2 \quad (8)$$

### 3.2 Experiments and results analysis

In order to get more objective results, both datasets randomly generate the missing data. For evaluation, the missing rate sets 5% and 200 times of random missing experiments are done on it to calculate the average of RMSE. The following is the comparison of QENNI and WWDQ when  $\beta \in \{0.0, 0.1, \dots, 0.5, \dots, 1.0\}$ .

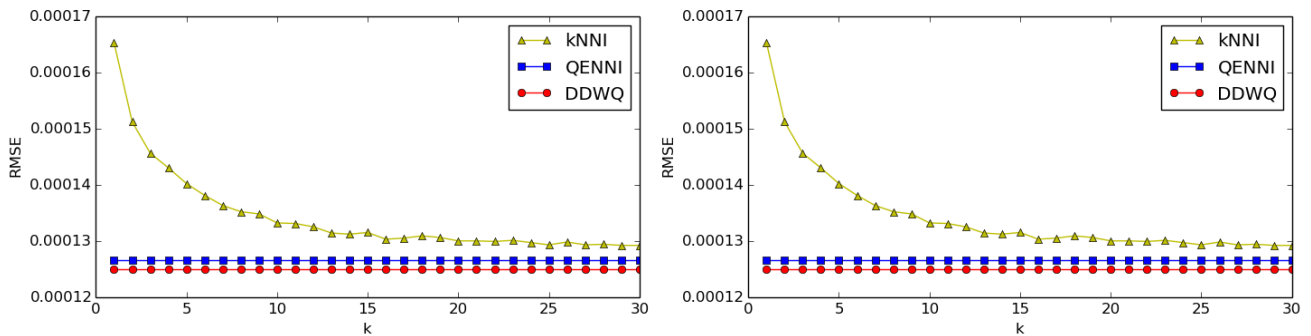


Fig. 4: Results of Abalone and Delta\_ailerons datasets

Compared from the Fig. 4, we can draw a conclusion that WWDQENNI algorithm has better imputation accuracy than QENNI no matter what's the value of  $\beta$ .

## 4 Conclusion

For improving the accuracy of missing data imputation, DDWQ imputation algorithm has been put forward. The algorithm is able to overcome the limitations of  $k$ NNI and QENNI. The innovation of our algorithm is to take the density of points in each quadrant and distance between the complete data and the missing data into consideration based on the Shell-neighbors thought. So, the result of missing data value can be more closer to the ideal one. The experimental results indicate that DDWQ algorithm has a better performance than QENNI.

## References

- [1] Zou G H, Li Y F, Zhu R, et al. Imputation of mean of ratios for missing data and its application to PPSWR sampling[J]. *Acta Mathematica Sinica, English Series*, 2010, 26(5):863-874.
- [2] Zhang S, S Z, Zhang S, et al. Missing Value Imputation Based on Data Clustering[J]. *Lecture Notes in Computer Science*, 2008:128-138.
- [3] Luengo J, Saez J A, Herrera F. Missing data imputation for fuzzy rule-based classification systems[J]. *Soft Computing*, 2012, 16(5):863-881.
- [4] Zhang S, S Z, Zhang S, et al. Missing data imputation by utilizing information within incomplete instances[J]. *Journal of Systems and Software*, 2011, 84:452-459.
- [5] P. Smaragdis, B. Raj, and M. Shashanka, Missing data imputation for spectral audio signals, in *IEEE Workshop in Machine Learning for Signal Processing (MLSP)*, 2009.
- [6] Zhang, S. (2011). Shell-neighbor method and its application in missing data imputation. *Applied Intelligence*, 35, 123-133.
- [7] Ling, Wang, Fu Dong-Mei. Estimation of Missing Values Using a Weighted K-Nearest Neighbors Algorithm. *Environmental Science and Information Application Technology*, 2009. ESIAT 2009. International Conference on. IEEE, 2009:660-663.
- [8] BATISTA G E, MONARDMC . A study of k-nearest neighbor as a model-based method to treat missing data[C] *Proceedings of the Argentine Symposium on Artificial Intelligence*. Bering Germany : Springer, 2001, 30: 1-9.
- [9] Kung Y, et al. An optimal k-nearest neighbor for density estimation[J]. *Statistics & Probability Letters*, 2012, (10):1786-1791.
- [10] Aptula A, Jeliaskova N, Schultz T, et al. The Better Predictive Model: High  $q \geq 2$  for the Training Set or Low Root Mean Square Error of Prediction for the Test Set?[J]. *Qsar & Combinatorial Science*, 2005, 24(3):385-396.