Comparative Analysis of Incomplete Business Data Clustering

Rongxuan Wang *, Longao Weng

Jinhe Center For Economic Research, Xi'an Jiaotong University, Xi'an, China

*Corresponding Author Email: 2203712694@stu.xjtu.edu.cn

Abstract. Incomplete values can significantly reduce the accuracy and usability of missing data. In particular, in analyzing commercial data sets, missing values often lead to the dilemma of data selection. It means that a common way to deal with missing data is to delete the sample that contains the missing attribute. However, this can lead to biased and invalidated conclusions, as some data are too critical to be omitted. Therefore, we should use some method to fill the data set rather than delete the data with missing values. The filling of missing data is divided into supervised learning and unsupervised learning. This paper compares six benchmark business datasets by adopting several different data imputation methods and supplementing the missing data with a clustering approach (unsupervised learning). The results are guided to dealing with incomplete business data.

Keywords: Clustering Algorithms, Business Analysis, Incomplete Data.

1. Introduction

There are often missing values in some dataset samples during data analysis. In many data sets collected from real life, there is often the problem of incomplete data. It has puzzled researchers in many fields. *Scalable tensor factorizations for incomplete data* [1] illustrate that in social sciences and engineering disciplines such as computer vision, biological systems, and remote sensing, the problem of incomplete data can hinder scientific research. *What is the expectation maximization algorithm* [2] depicts that the only data available for training a probabilistic model is often incomplete. Moreover, *an effective and efficient approach to classification with incomplete data* [3] depicts 45% of the datasets in the UCI dataset, a commonly used benchmark database for machine learning, containing missing values. The reasons for missing values vary. For instance, data loss occurs in various situations due to information loss, errors during data collection, or costly experiments detected in *Scalable tensor factorizations for incomplete data* [1]. Meanwhile, in the application of remote sensing, improper sensor placement results in incomplete data.

Incomplete data is also very common in finance, business, and other fields. *A fuzzy artificial neural network (p, d, q) model for incomplete financial time series forecasting* [8] indicates that financial institutions will face significant value loss problems in obtaining data to obtain the ability of borrowers to repay loans on time. In the face of incomplete data, it is an essential and challenging task for forecasters to improve the prediction accuracy, especially the time series prediction accuracy. According to recent studies, existing methods for dealing with incomplete data can be divided into deletion and imputation. The delete method is one of the easiest ways to deal with incomplete data, which means that we only need to delete examples that contain missing values. However, the deletion of some samples may lead to the loss of valuable and important information because some data containing missing attributes also contains helpful information. Deletion is generally desirable when only a small fraction of samples contain missing values, and the biggest advantage of these methods is that they are efficient and time-saving [3]. However, deletion is not always justified, especially when many samples have missing values according to *best practices for missing data management in counseling psychology* [5].

Data imputation refers to filling in the missing values with specific values to complete the data set. Uncertainty in the data is ignored because calculations treat the data as known values after populating. [1] One of the easiest ways to do this is to fill in the missing value with a constant, and we usually use zero or average value. These methods are simple to deal with missing data and save time, so they have outstanding efficiency. Another area of data imputation methods is more complex, such as K-

nearest neighbor imputation (KNNI) and random forest imputation [6], which are usually based on the correlation between missing variables [3]. These relatively complex data calculation methods are often more accurate, but the efficiency is relatively low [6], which takes much more time than the previous methods. Therefore, it is often difficult to choose the most appropriate method.

In business analysis, clustering has a wide range of application scenarios [7]. For example, different customers have different consumption characteristics when consuming the same kind of goods or services. The commonly used customer classification methods mainly include traditional and non-traditional statistical methods, namely simple statistical methods based on customer attributes and non-numerical methods based on artificial intelligence technology. Cluster analysis has the characteristics of the latter two methods and can effectively complete the customer segmentation process.

Specifically, customers' purchase motivation is generally determined by internal factors such as need, cognition, and learning and external factors such as culture, society, family, small groups, and reference groups. To divide customers according to different purchase motives, the factors mentioned above can be taken as analysis variables, and then the cluster analysis method can be used for classification. The common point of the above analyses is that they are classified according to multiple variables, which is just in line with the characteristics of cluster analysis to solve problems [7].

In this paper, we compare the performance of four different data imputation methods on six benchmark business and financial datasets. First, we artificially randomly removed some attributes of the dataset and then used different methods to calculate the data. Then, K-nearest neighbor and fuzzy C-means are used to cluster the filled data [7]. After that, we tested the clustering performance using three criteria: Silhouette Coefficient, balance score (F1score), and fuzzy partition coefficient (VPC function), respectively. This result gives us some insight into how to deal with actual data.

2. Our Methods

This paper uses four data imputation methods: zero imputation, mean imputation, K-nearest neighbor-based imputation, and random forest imputation. We will explain the principles in which these methods work. Then, K-means and fuzzy C-means clustering methods are used to evaluate and compare the performance of these data imputation methods. These four data imputation and clustering methods are chosen because they have been widely used. Each data imputation method has its unique advantages and can be adapted to some specific data sets.

2.1. Zero Imputation and Mean Imputation

Zero imputation and mean imputation operate similarly: they use a specific value to replace the missing data. Zero imputation uses zero to fill in all missing values, regardless of the differences between missing data on different attributes, which is the easiest way to deal with incomplete data.

Mean imputation chooses to use the average value of the attributes that contain missing values to fill in the missing values. The mean imputation method is the most commonly used data imputation method because it is efficient and easy to understand. When variables are nominal, we use modal value calculations.

The common feature of these two methods is their simplicity and ease of adoption. Therefore, they are widely used. However, these simple data calculation methods are not accurate enough in many cases. The main limitation of these simple methods is that replacing missing values with mean or mode or some constant leads to distorted distribution function estimates and degrades the quality of data mining results [4].

2.2. k-nearest neighbor-based imputation

The k-nearest neighbor-based (KNN) imputation is based on the k-nearest neighbor-based (KNN) algorithm, which is a more accurate imputation method to deal with complex data sets. We will introduce the KNN method.

The basic principle of the KNN algorithm is that a sample belongs to a particular category. KNN is applied when this happens: most of the k nearest neighbors of this sample in the feature space are affiliated to a specific category. If so, there is a high probability that the missing data will also fall into this category. In other words, the method determines the class of samples to be classified based only on the class of the most recent sample or samples. The algorithm flow chart of KNN is shown in Fig. 1.

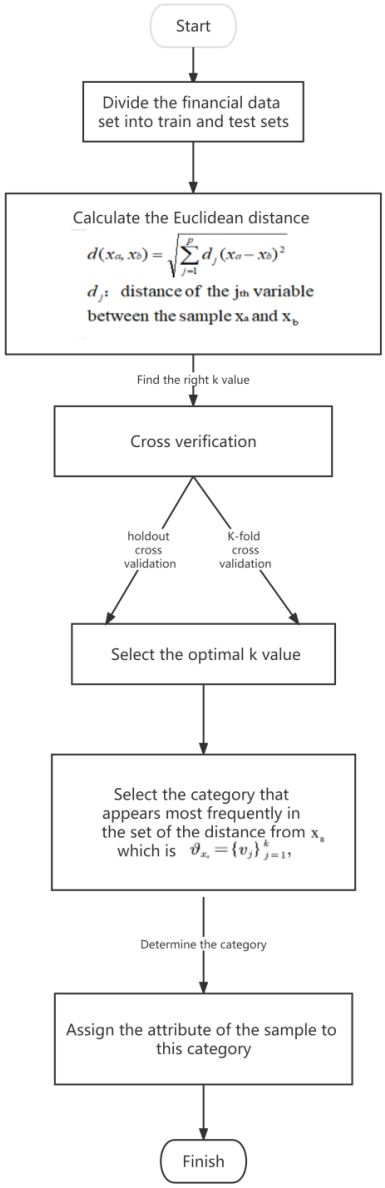

Figure 1. The process of KNN

It is found that the KNN algorithm can achieve relatively accurate classification, which provides an idea for researchers to supplement missing values when dealing with data containing missing values. The classification result of the KNN algorithm is filled with the value of the missing item in this attribute. Therefore, KNNI (k-nearest neighbor-based imputation) was created based on the KNN algorithm.

KNNI also uses k-nearest neighbors of the sample containing missing values to calculate a value to fill in the missing value. The k-nearest neighbors of the sample are its most resemble samples from

the dataset. The most common value among all neighbors is taken for nominal values, while the average value is used for numerical values [6]. The algorithm flow chart of KNNI is shown in Fig. 2.

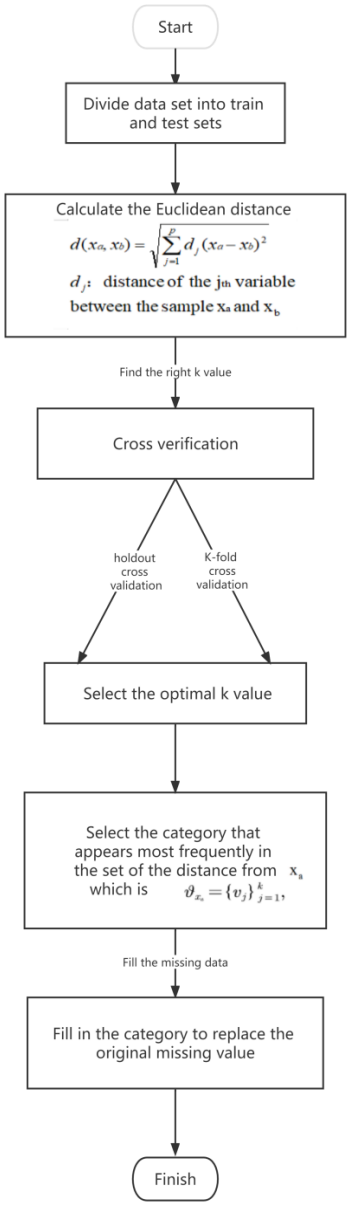

Figure 2. The process of KNNI

KNNI method is more efficient than zero imputation and mean imputation. In addition, the principle of KNNI is easy to realize. However, KNNI has its limitations. First, when the number of samples is large, it can be computationally expensive because, for each sample, it needs to search k-nearest neighbors. More importantly, we use Euclidean distance to find the most similar neighbors in samples with missing values in the paper. However, in the academic community, there is still controversy as to which method we should use to generate the k most similar samples [4]. In addition, in high-dimensional data sets, the difference between nearest and farthest neighbors is small. Therefore, the accuracy of KNN will be reduced.

2.3. Random forest imputation

The principles of random forest inference (RFI) are much more complex than those mentioned earlier. In practical research, RFI is not widely used. However, as a supplement and contrast to previous methods, we still use this method for imputation.

Random Forest (RF) algorithm is another classification algorithm, and the idea is that the probability of error of a large number of decision trees is much lower than the probability of error of a single decision tree. By creating multiple decision trees (forest) and training them without pruning, the classification of a particular attribute of the data is predicted.

Considering the pattern of prediction results of many decision trees as the overall prediction result, the accuracy of RF classification is high.

The algorithm flow chart of RF is shown in Fig. 3.

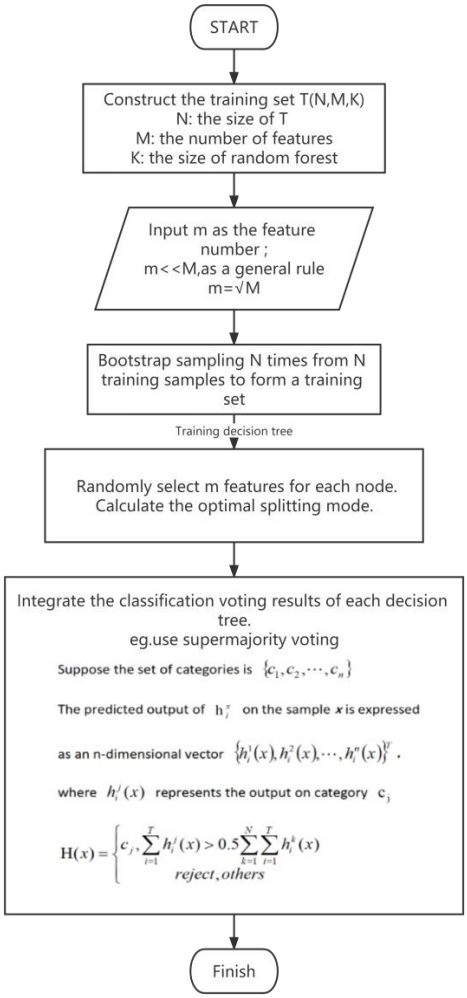

Figure 3. The process of RF classification

RFI is proposed because of the satisfactory classification accuracy brought by the RF algorithm. The classification results of the RF algorithm are added to the dataset as fill-in values. In front of multiple data sets with missing attributes, RFI will select the attribute with the least missing values to start interpolation and take the interpolation result as new information to fill the subsequent attributes with more missing values. The algorithm flow chart of RFI is shown in Fig. 4.

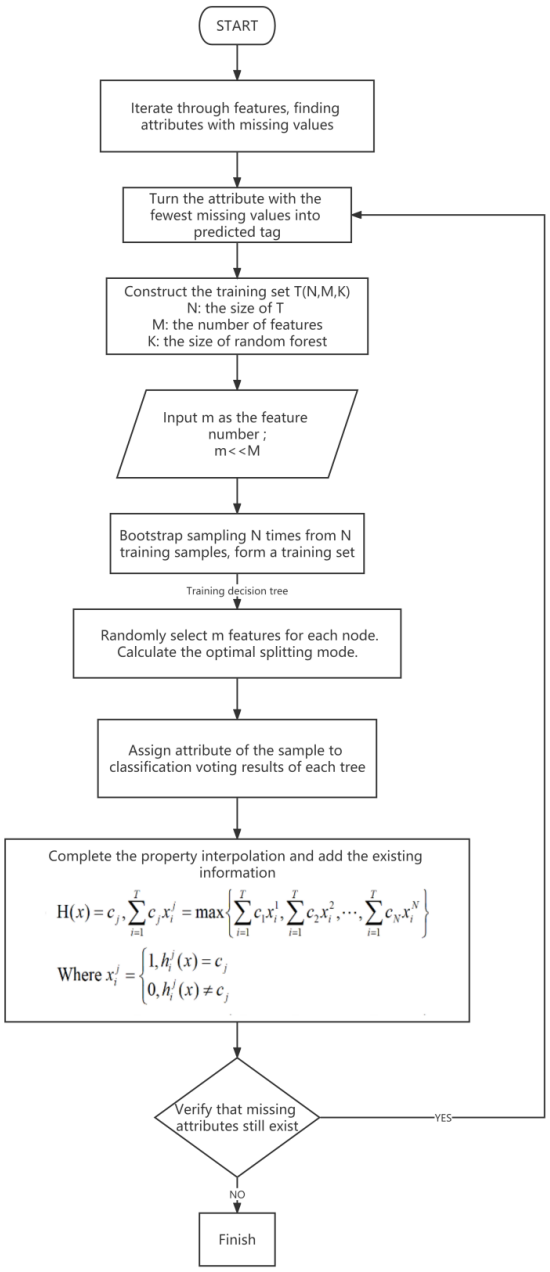

Figure 4. The process of RFI

As high as its classification performance is, RFI is also very efficient. However, the drawback of this algorithm is the repeated use of random forest modeling in the algorithm operation, which leads to a large amount of computation. Therefore, the RFI algorithm always takes a long time to work.

2.4. K-means clustering algorithm

Kmeans algorithm is the most commonly used clustering algorithm. The main idea is to give the values of K and K initial cluster center, under the condition of each sample, to its nearest cluster center, which represents a class cluster. After all, point distribution, according to all points within a class cluster, recounts the class (average) at the center of the cluster. Then, the steps of allocating points and updating the center point of the class cluster are iteratively carried out until the change of the center point of the class cluster is very small, or the specified number of iterations is reached. The algorithm flow chart of Kmeans is shown in Fig. 5.

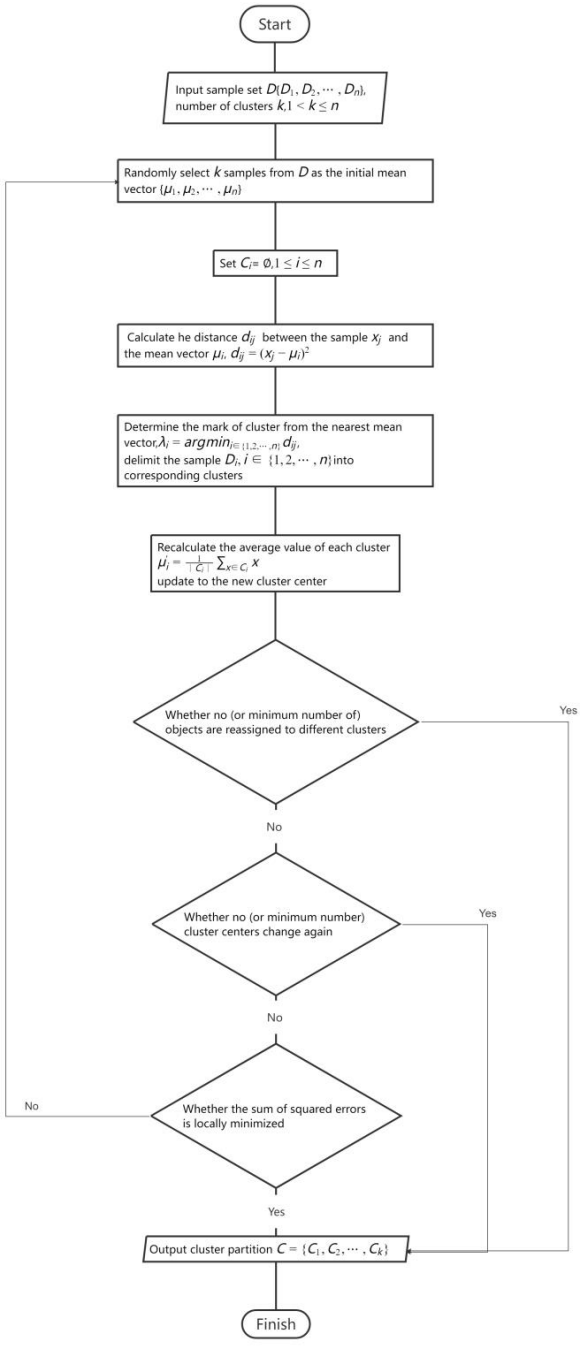

Figure 5. The process of Kmeans

K-means clustering is a Gaussian Mixture Model (GMM) solved by an expectation-maximization algorithm. The covariance of the normal distribution is the identity matrix. The case is obtained when the posterior distribution of the implicit variable is a set of Dirac delta functions.

The k-means algorithm is a complicated clustering algorithm. That is, clustering can be divided into hard clustering and soft clustering [7]. If a point can belong to only one class cluster, it is hard clustering; if a point can belong to multiple class clusters, it is soft clustering, as mentioned in the Fuzzy C-means (FCM) algorithm later. In practice, hard clustering is the main method.

Kmeans is representative of the typical prototype-based objective function clustering method. It is a certain distance between data points and prototype as the optimization objective function, and the method of finding the extreme value of the function is used to obtain the adjustment rules of iterative operation.

In addition, the Kmeans algorithm has the following features: Because it is distance-based clustering, it is suitable for samples with clique characteristics. The characteristics of the heuristic

make the Kmeans algorithm extremely dependent on initial points, which are randomly selected, so even if the original samples have good classification characteristics, the final results may be unsatisfactory, and the model is unstable. Moreover, because it takes the mean, the algorithm is sensitive to outliers.

Moreover, the crucial point is that although the calculation process is simple, the time complexity and the number of iterations are very high, and the time cost is very high. Its convergence is step by step. The speed of convergence is not fast.

2.5. The fuzzy c-means clustering algorithm

Fuzzy C-means Algorithm (FCMA or FCM). The fuzzy C-means (FCM) algorithm is the most widely used and successful algorithm among many fuzzy clustering algorithms. It obtains the membership degree of each sample point to all class centers by optimizing the objective function to determine the class of the sample points and achieve the purpose of automatic classification of the sample data [7].

Fuzzy clustering analysis is one of the main technologies of unsupervised machine learning, and it is an important data analysis using the theory of the fuzzy modeling method. It is established that the description belongs to the uncertainty of the sample, which can more objectively reflect the real world. It has effectively been used in large-scale data analysis, data mining, vector quantization, image segmentation, pattern recognition, and other fields. It has important theoretical and practical application value.

The algorithm flow chart of FCM is shown in Fig. 6.

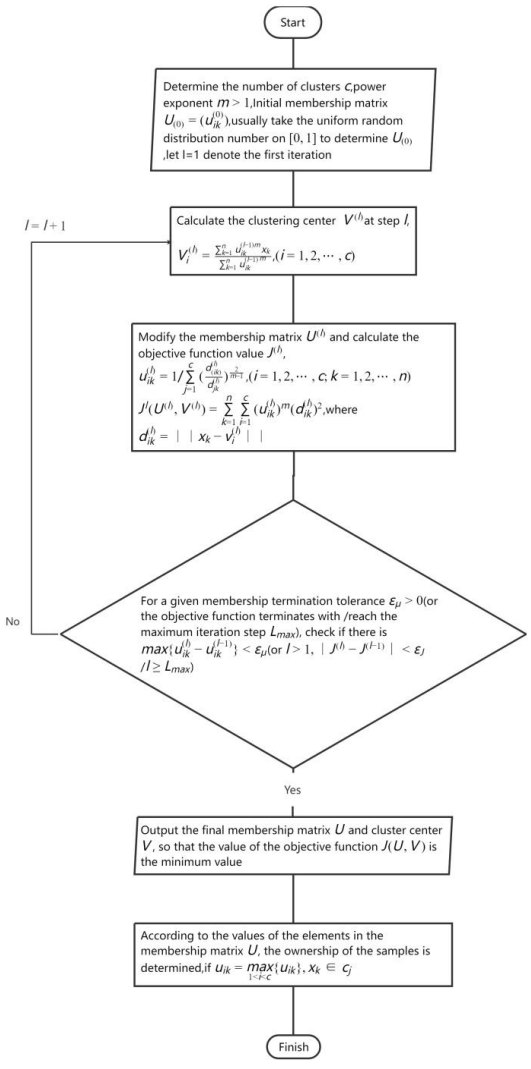

Figure 6. The process of FCM

FCM algorithm determines the degree to which each sample belongs to a particular cluster by membership degree, which is the unique feature of the soft clustering algorithm. Compared with k-means and center point algorithms, the computation amount is significantly reduced, and the efficiency will be improved because it saves the repeated computation process of multiple iterations. At the same time, fuzzy clustering analysis can form a fuzzy similarity matrix according to the relevant data in the database. After forming the similarity matrix, we can directly process the similarity matrix, avoiding the waste of time and cost caused by repeatedly scanning the database [7].

The *m* value is set dynamically according to the experimental requirements to meet the needs of different types of data mining tasks. It is suitable for the processing of high-latitude data and has good scalability, which is easy to find outliers. However, the *m* value is obtained from experience or experiment, which is uncertain and may affect the experimental results. Moreover, the search direction better than the gradient method is always along the direction of energy reduction, which makes the algorithm easy to fall into local minima and sensitive to initialization. In order to overcome the above shortcomings, the optimization method can be introduced into the FCM algorithm to eliminate the local minima that may fall during FCM clustering operation to optimize the clustering effect.

3. The experiment

3.1. Description of the Dataset

We choose six benchmark business datasets, five of which are from the UCI database, which is a well-known machine learning database. The last dataset is about the factors that influence corporate bankruptcy. They are South German Credit, Australian Credit Approval, Bank Marketing, the default of credit card clients, Online Shoppers Purchasing Intention, and Bankrupt. For datasets that contain too many samples, we only select part of them to do the analysis. A detailed description of the datasets is shown in Table I.

Table 1. The Description of Data

dataset	*observations*	*Features*	*Clusters*
South German Credit	1000	21	2
Australian Credit Approval	691	15	2
Bank Marketing	4521	17	2
default of credit card clients	30000	24	2
Online Shoppers' Purchasing Intention	6855	18	2
Bankrupt	1359	96	2

3.1.1 The Preprocessing of the data

We use Jupyter in the Python environment for programming and data analysis. In order to test the performance of unsupervised learning, we directly randomly removed some features of the samples from each dataset instead of dividing the training set and the test set. More specifically, for a property of some sample in each dataset, we will assign it a coordinate *(i, j)*. I will be strictly less than the total number of features in each dataset, and j will be strictly less than the total number of samples in each dataset. Then, the coordinates m and n times were extracted by the two-dimensional random number method, and the *m*n* data of the data set were randomly deleted. At the same time, the interference caused by the same number of deleted features of different samples on the clustering results was avoided.

3.1.2 The clustering of the Missing Values

As mentioned above, we use four data imputation methods to replenish the missing values of each of the deleted data sets. We then wrote code to run two clusters, k-means and FCM, to see their performance on deleted datasets. By comparing the actual classes of samples and the clustering

classes, we can evaluate the accuracy of the data imputation and clustering. The data calculation method becomes more feasible with the accuracy of data clustering.

3.1.3 The Evaluation of clustering

Firstly, we structure the confusion matrix for the evaluation of the clustering. The confusion matrix is defined in table II:

Table 2. The Confusion Matrix

	Positive	***Negative***
Positive	***TP***	***FN***
Negative	***FP***	***TN***

Through the confusion matrix, we can construct the index of our valuation. Accuracy is precisely defined as the ratio of successful predictions to the total number of samples:

$$Accuracy = (TP + TN)/Total \tag{1}$$

Where $Total = TP + TN + FP + FN$

$Precision$ is defined as:

$$Precision = TP/(TP + FP) \tag{2}$$

$Recall$ is defined as:

$$Recall = TP/(TP + FN) \tag{3}$$

However, $Recall$ will decrease as the $Precision$ increases, and these two indicators are always negatively correlated. We can adopt $F1\ Score$, which is a combination of $Precision$ and $Recall$, to evaluate the classification performance.

Then the $F1\ Score$ is defined based on the $Precision$ and $Recall$:

$$F1\ Score = 2 \times Precision \times Recall/(Precision + Recall) \tag{4}$$

Thus, $F1\ Score$ can be regarded as $Precision$ and $Recall$. Using $F1\ Score$ means that these two indicators are seen as equally important. $F1\ Score$ ranges from 0 to 1, and the closer it is to 1, the better the classification.

Moreover, we introduced the $Silhouette\ Coefficient$ as another index to evaluate the clustering effect. Peter J. Rousseeuw first proposed it in 1986.

It combines the two factors, which are cohesion degree and separation degree, $Silhouette\ Coefficient$ can be used to evaluate the impact of different algorithms or operation modes of algorithms on the clustering results based on the same original data.

It is supposed that we have clustered the data to be classified through a particular algorithm. For each vector in the cluster, their contour coefficients were calculated respectively:

For a point, *i*, let *n* be the number of elements in the cluster of *i*, *m* be the number of clusters, and d_{ij} is the distance between *i* and another *j* elements in clusters containing *i*.

Calculate *a(i)* as:

$$a(i) = \frac{\sum_{j=1}^{n-1} \left| d_{ij} \right|}{n\text{-}1} \tag{5}$$

In the same way, the average distance between the vector *i* and all points in the cluster that does not contain *i*, respectively named as $b_1(i), b_2(i), \ldots, b_{m-1}(i)$.

Calculate *b(i)* as:

$$b(i) = min\{b_1(i), b_2(i), \ldots, b_{m-1}(i)\} \tag{6}$$

Then the $Silhouette\ Coefficient$ of *i* vector is:

$$S(i) = \frac{b(i)-a(i)}{max\{a(i),b(i)\}} \tag{7}$$

Therefore, it can be seen that the value of the contour coefficient is between [-1, 1], and the closer it is to 1, the better cohesion and separation degree are.

In particular, to evaluate the effect of fuzzy clustering, we introduce a particular evaluation index for measuring FCM. In fuzzy clustering, many scholars consider the information of membership degree and data set geometry structure and define the validity function. The classical ones include Bezdek's V_{PC} function, Fukuyama-Sugeno's V_{FS} function, and Xie-Beni's V_{XB} function. Here we use the V_{PC} function to evaluate it. It is described as:

$$V_{\mathrm{PC}}(U,c) = \frac{1}{n}\sum_{i=1}^{c}\sum_{j=1}^{n}\mu_{ij}^{2} \tag{8}$$

In the formula, n is the number of samples in the data set, c is the number of clustering centers, μ_{ij} is the membership degree of the j_{th} sample belonging to the ith class. U is the fuzzy partition matrix.

The V_{PC} function is a continuous convex function in the defined interval. Therefore, the function has a unique maximum value in the interval, and the c value corresponding to the maximum value is the number of clusters. The closer it is to 1, the better the effectiveness of fuzzy clustering is.

We choose $Silhouette\ Coefficient$ and $F1\ Score$ as the indexes for all of our clustering, and V_{PC} is specially used to evaluate the effect of fuzzy c-means clustering. From our data, we calculate these indexes, and the result is shown in Fig. 7, Fig. 8, and Fig.9:

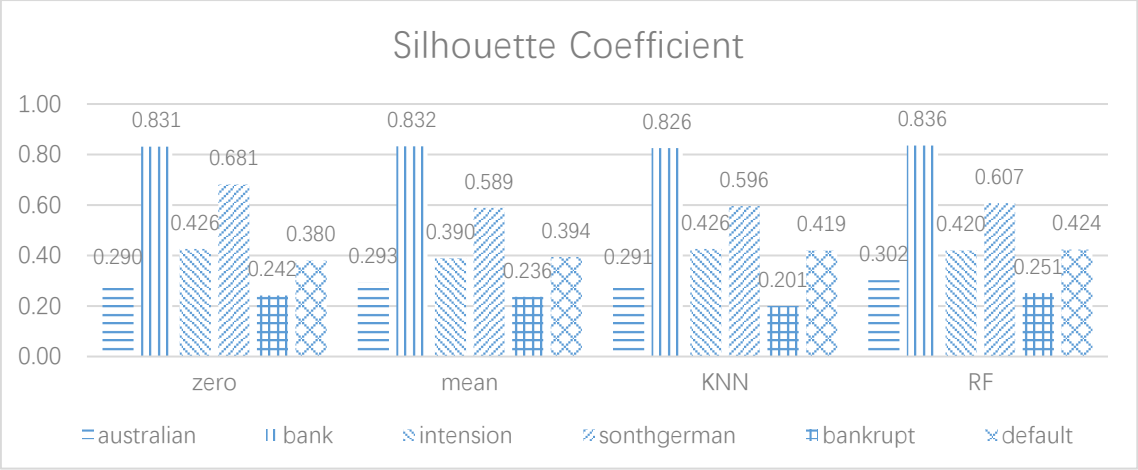

Figure 7. Results of Silhouette Coefficient after clustering

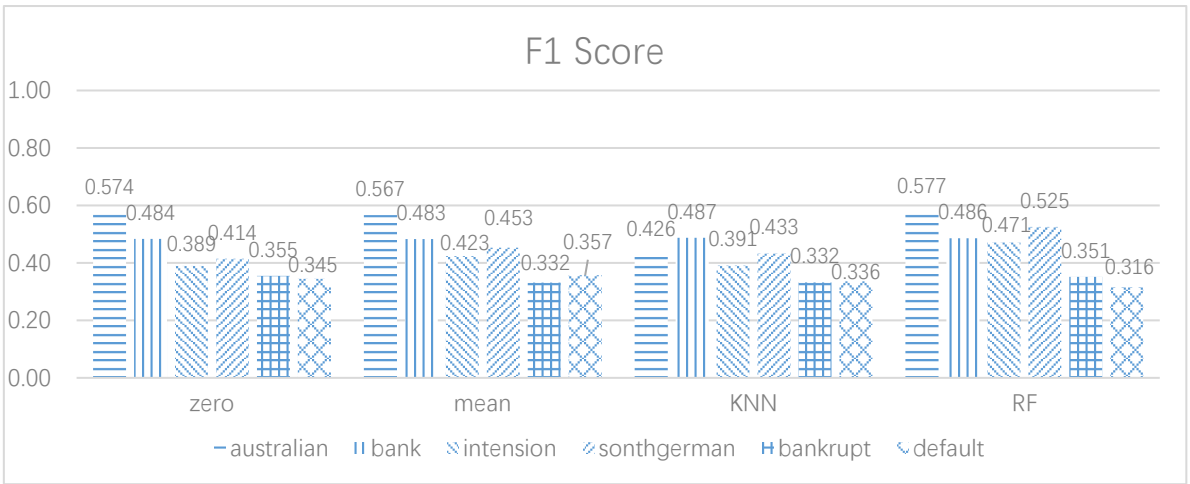

Figure 8. Results of $F1\ Score$ after clustering

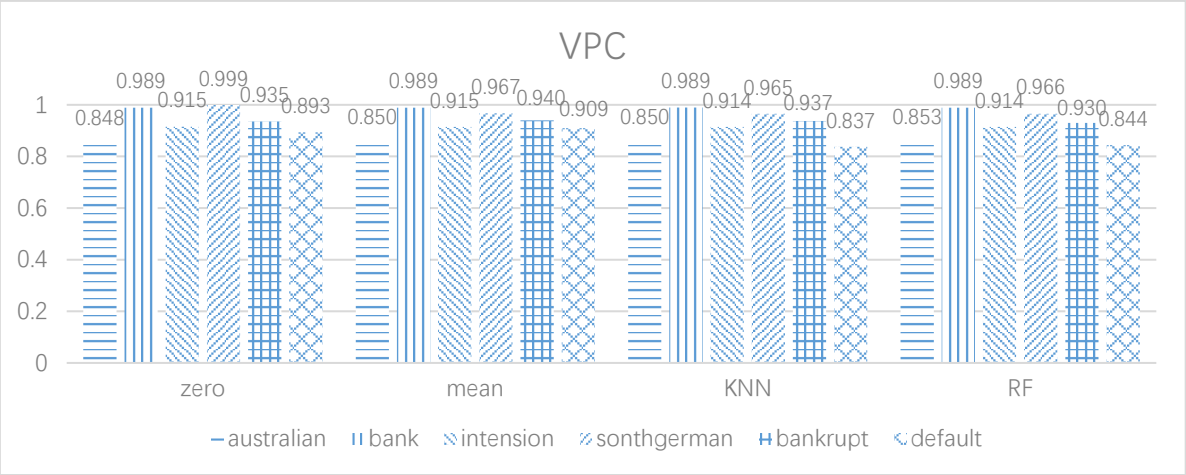

Figure 9. VPC of different imputation methods

Since the data sets we use are different in the number of samples, attributes, and data types, we can find that different interpolation method perform differently on different types of data sets.

For example, according to the distribution of the F1 score, compared with the multivariate case, the calculation cost of adding missing values to a dataset composed of simple data types is much cheaper under the premise of the same accuracy. From the results, we can see that in clustering, the accuracy of each data interpolation method is roughly similar in each data set. Specifically, we note that Zero and Mean, two relatively simple and easy to adopt methods, hardly lag behind other relatively complex methods, suggesting that we have no reason to use computationally expensive methods in most cases.

KNNI and RFI algorithms' effectiveness is very close, indicating that the performance of KNNI and RFI on our dataset is also similar. However, RFI takes much more time than KNNI. Therefore, we conclude that RFI is unnecessary in most cases due to its high complexity and unintuitive principles. Further evidence is shown that KNN is still the most widespread interpolation method available.

However, we find that the effectiveness of different clustering methods is not so close in selecting clustering methods. Specifically, the rationality of using soft clustering (FCM) is much higher than that of using hard clustering algorithm in the index of testing the model's validity. FCM is not only more affordable in terms of time cost but also has higher effectiveness and rationality. Especially in the face of multi-attribute and large sample data sets, FCM performs far better than the complicated clustering algorithm, which is good news for the popularization and extensive use of the soft clustering algorithm.

4. Conclusion

This paper uses four imputation methods to deal with six business datasets with artificial missing data. Our experiments show that the most suitable imputation methods differ only on some unique datasets when the accuracy of filling incomplete data is taken as the evaluation index. However, when selecting the clustering method after filling the dataset, it can be found that the stability of the clustering result of soft clustering (FCM) is much better than that of hard clustering, which is consistent with the fact that fuzzy clustering is widely used and the field is developing rapidly.

References

[1] Acar, E., Dunlavy, D. M., Kolda, T. G., & Mørup, M. (2011). Scalable tensor factorizations for incomplete data. Chemometrics and Intelligent Laboratory Systems, 106(1), 41-56.

[2] Do C B, Batzoglou S. What is the expectation maximization algorithm? [J] Nature biotechnology, 2008, 26(8): 897-899.

[3] Cao Truong Tran, Zhang Mengjie, Andreae Peter, Xue Bing, Lam Thu Bui. An effective and efficient approach to classification with incomplete data [J]. Knowledge-Based Systems, 2018, 154(AUGa15): 1-16.

[4] Lan Qiujun, Xu Xuqing, Ma Haojie, Li Gang. Multivariable data imputation for the analysis of incomplete credit data [J]. Expert Systems with Application, 2020, 141(Mara): 112926.1-112926.12

[5] Schlomer G L, Bauman S, Card N A. Best practices for missing data management in counseling psychology [J]. Journal of Counseling psychology, 2010, 57(1): 1.

[6] Julián Luengo, Salvador García, Francisco Herrera. On the choice of the best imputation methods for missing values considering three groups of classification methods [J]. Knowledge and Information Systems, 2012, 32(1): p.77-108

[7] Kaufman L, Rousseeuw P J. Finding groups in data: an introduction to cluster analysis [M]. John Wiley & Sons, 2009.

[8] Khashei, Mehdi, and Mehdi Bijari. Fuzzy artificial neural network (p, d, q) model for incomplete financial time series forecasting. Journal of Intelligent & Fuzzy Systems 26.2 (2014): 831-845.

[9] Belgiu, Dragut. Random forest in remote sensing: A review of applications and future directions [J]. ISPRS J PHOTOGRAMM, 2016, 2016, 114(-): 24-31.

[10] Sinaga K P, Yang M S. Unsupervised K-means clustering algorithm [J]. IEEE access, 2020, 8: 80716-80727.

[11] Wang S, Li M, Hu N, et al. K-means clustering with incomplete data [J]. IEEE Access, 2019, 7: 69162-69171.

[12] Ghosh S, Dubey S K. Comparative analysis of k-means and fuzzy c-means algorithms [J]. International Journal of Advanced Computer Science and Applications, 2013, 4(4).

[13] Nie F, Wang X, Huang H. Clustering and projected clustering with adaptive neighbors [C] // Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014: 977-986.