Scientific
Research

Support Vector Machine-Based Fault Diagnosis of Power Transformer Using *k* Nearest-Neighbor Imputed DGA Dataset

Zahriah Binti Sahri[1], Rubiyah Binti Yusof[2]

[1]Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia
[2]Malaysia Japan International Institute of Technology, Universiti Tekonologi Malaysia, Kuala Lumpur, Malaysia
Email: szahriah2511@yahoo.com.my, rubiyah@ic.utm.my

Abstract

Missing values are prevalent in real-world datasets and they may reduce predictive performance of a learning algorithm. Dissolved Gas Analysis (DGA), one of the most deployable methods for detecting and predicting incipient faults in power transformers is one of the casualties. Thus, this paper proposes filling-in the missing values found in a DGA dataset using the k-nearest neighbor imputation method with two different distance metrics: Euclidean and Cityblock. Thereafter, using these imputed datasets as inputs, this study applies Support Vector Machine (SVM) to built models which are used to classify transformer faults. Experimental results are provided to show the effectiveness of the proposed approach.

Keywords

Missing Values, Dissolved Gas Analysis, Support Vector Machine, *k*-Nearest Neighbors

1. Introduction

Power transformers are used for transmitting and distributing electricity from plant to customer in utility companies worldwide. Therefore, it is important to always ensure good operating condition of power transformers to provide reliable and continuous supply of electrical power, a necessity in this modern world. Acting on this fact, utility companies have implemented various condition assessment and maintenance measures, and Dissolved Gas Analysis (DGA) is one of them. DGA is a method that detects and predict faults found in oil-filled transformers by: a) analyzing the concentration of certain gases dissolved in the insulating oil, and their gassing rates, and gas ratios, b) identification of fault using diagnostic tools such as Key Gas [1], IEC ratios, Rogers ratios [1], Dornenburg ratios [1] and Duval Triangle [2]. However, these tools face some shortcomings. In some cases, the calculated gas ratios fall outside the established ratios codes of the aforementioned tools. As a result, faults that occur inside transformers may not be identifiable [3]. In addition, these tools can give different analysis results for the same dissolved gas record, and it is difficult for engineers to conclude a final assessment when faced with so much diverse information [4].

Those drawbacks have motivated many researchers to develop faults diagnostics tools embedded with machine learning techniques that learn from historic DGA data to predict new or unknown faults. In recent years, Support Vector Machine (SVM) has been widely applied for classification of power transformer faults. Such interest is justified by: 1) SVM excellent generalization ability to new knowledge; 2) SVM requires limited effort for architecture design (*i.e.*, it involves few control parameters); and 3) SVM ability to classify non-linear problems. Capitalizing on these strengths, this paper adopts SVM to learn from historic DGA data to predict faults of power transformer.

Alas, those strengths of SVM are not the only deciding factor for achieving higher predictive accuracy of its learning task. The representation and quality of the training data is first and foremost [5]. One factor that affects data quality is the presence of missing values in a dataset. Unfortunately, it is a common fact that missing values persistently appear in most real-world data sources and DGA datasets are no exception as documented by [6]. It has been proven [7] [8] that as missing values increase in a dataset, the predictive accuracy of an algorithm that learns from this dataset decreases in tandem. As many statistical and learning methods cannot deal with missing values directly, examples with missing values are often deleted. However, deleting cases can result in the loss of a large amount of valuable data. Thus, much previous research has focused on filling-in the missing values with estimated values ("imputing") before learning and testing is applied to.

In this paper, we propose imputing the missing values in a DGA dataset before the learning process of SVM takes place with the main objective of increasing SVM performance in classifying power transformers' faults. At present, there are many established imputations methods such as mean/mode, regression, expectation maximization and multiple imputation to choose from. These techniques, however, require a *priori* knowledge of data distribution in order to produce as accurate estimation values as possible. In view of this hard-to-realize prerequisite, this paper adopts a simpler well-known method that is the k-nearest neighbour (kNN) to estimate plausible values to fill-in the missing values in DGA dataset. The successful attempts by [9] [10] in using kNN for imputing missing values have also motivated us to do the same.

In Section 2 we briefly described imputation methods and application of SVM. The proposed combination of kNN imputation method and SVM is elaborated in Section 3. Section 4 presents and analysed experimental results. Section 5 concludes our findings and future work.

2. Related Work

2.1. Imputation Methods

Various methods have emerged over the years to determine and assign replacement values for missing data items. These methods can be categorised into statistical [11] [12] and machine learning methods [13]. Linear regression, multiple imputation, parametric imputation, and non-parametric imputation fall into the first category; whilst neural networks, decision tree imputation, and kNN fall into the latter.

The kNN algorithm fills in missing data by taking values from other observations in the same data set. This method searches the k-nearest neighbours of the case with missing value(s) and replaces the missing value(s) by the mean or mode value of the corresponding feature values of the k-nearest neighbours. The advantages of the kNN imputation are:

1) It does not require creating a predictive model for each feature with missing data.
2) It can treat both continuous and categorical values.
3) It can easily deal with cases with multiple missing values.
4) It takes into account the correlation structure of the data.

Most notably, kNN is also a non-parametric imputation which makes it practically

2.2. Application of Support Vector Machine

In the area of pattern recognition, SVM has been applied to recognize offline and online handwritten characters. After extracting important features such as chain code, density, and number of lines, SVM was trained on these features to build a model to recognize offline handwritten numbers in [14]. The authors reported 98.99% recognition rate. Meanwhile, researchers in [15] combined two classifiers which were Convolutional Neural Network (CNN) and SVM to solve the handwritten digit recognition problem. Their hybrid model was evaluated in two aspects: the recognition accuracy and the reliability performance. Experimental results on the MNIST digit da-

tabase showed the proposed hybrid model surpassed CNN and human subjects.

SVM has also gained attention in the image recognition field. The authors in [16] automated the process of bacterial recognition and counting during the process of detection of food contamination. Compared with the results recognized by human eye, SVM can effectively distinguish the bacteria from non-bacteria in the image, and greatly reduce the detection time of each sample. SVM was applied in [17] to recognize facial expression by Gabor features which separate facial region from images and reported satisfied results.

In the context of faults prediction of power transformers, some experimental investigations pointed out the effectiveness of SVM in classification of fault despite the small size of DGA training and testing samples. Researchers in [18]-[20] trained and tested their SVM using less than 100 samples for each investigated fault and reported accuracy between 80%-100%. Their SVM classifier also performed better when compared to ratio-based DGA diagnostic tools and NN and fuzzy logic

3. The Proposed Method

Figure 1 depicts the system architecture for classifying transformer faults. As an input is the incomplete (dataset that contains missing values) DGA dataset, then, all missing values are filled-in with estimated values obtained from the kNN algorithm which results in a complete DGA dataset (dataset with no missing values). Using this complete dataset, SVM learns and predicts the transformer faults.

3.1. Impute Missing Values Using kNN

As an imputation method, the kNN algorithm is very efficient and simple to execute. In this method, the missing values of a sample are imputed considering k samples that are most similar to the sample of interest. The similarity of two samples is determined using a distance metric. This study uses three well-known distance metrics to gauge similarities among samples.

1) City Block Distance (*CB*): It is based on Taxicab geometry, first considered by Hermann Minkowski in the 19th century, is a form of geometry in which in which the distance between two points is the sum the absolute differences of their coordinates defined using the following equation:

$$d_{st} = \sum_{j=1}^{n} \left| x_{sj} - y_{tj} \right| \tag{1}$$

2) Euclidan Distance (*EU*): This is the most usual way of computing a distance between two objects. It examines the root of square differences between coordinates of a pair of objects and is defined using the following equation:

$$d_{st} = \sqrt{\sum_{j=1}^{n} \left(x_{sj} - y_{tj} \right)^2} \tag{2}$$

Generally, the steps of k-NN are as follows:

a) choose k, the number of nearest neighbours to be selected.

b) calculate the distance between the sample with the to-be-imputed missing value with an another sample using a distance metric. Let $X_i = \{x_{i1}, x_{i2}, \cdots, x_{im}\}$ denotes the instance that contains the to-be-imputed missing values and $X_q = \{x_{q1}, x_{q2}, \cdots, x_{qm}\}$ be the other sample. If the metric is CB then the distance between X_i and X_q is:

$$d\left(X_i, X_q \right) = \sum_{j=1}^{m} \left| x_{ij} - x_{qj} \right| \tag{3}$$

where m is the number of features in X_i and X_q, and x_{ij} is the jth feature of sample X_i and x_{qj} is the jth feature of sample X_q.

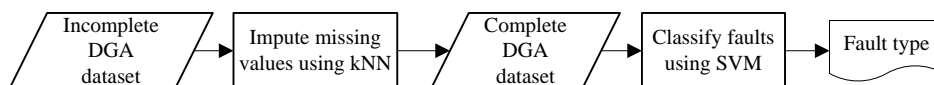| Incomplete DGA dataset | → | Impute missing values using kNN | → | Complete DGA dataset | → | Classify faults using SVM | → | Fault type |

Figure 1. The proposed method for classifying transformer faults.

c)   Repeat step 2 to compute the distance between X_i with each remaining sample in the dataset.

d)   sort in ascending order (based on the calculated distance values) all X_q excluding X_i.

e)   select the top k samples from the sorted list as the k-nearest neighbours to X_i. These k-nearest neighbours are $X_{kNN} = \{X_1, X_2, \cdots, X_k\}$.

f)   Let x_{ij} be the to-be-imputed missing value in X_i. Then the estimated value is obtained from

$$x_{ij} = \frac{\sum_{l=1}^{k} x_{lj}}{k} \tag{4}$$

where k is the number of nearest neighbours, x_{lj} is the jth feature of sample x_{lj}, and $X_l \in X_{kNN}$.

3.2. Classify Using SM

Figure 2 depicts the implementation of SVM for classifying transformer faults using DGA datasets. In this phase, the dataset used as input is the imputed and complete dataset from the imputation phase executed earlier.

a)   Normalization: A common characteristics of DGA datasets is the wide range of values contained in some of the attributes. To avoid the possible domination of attributes with greater numeric ranges, this study adopts a preprocessing technique called normalization, specifically the min-max normalization. All attributes are normalized to [0,1] as follows:

$$v' = \frac{v - \min_A}{\max - \min} \tag{5}$$

where v is the actual value, min is the minimum value of an attribute A, max is the maximum value of attribute A, v' is the normalized value.

b)   Training and Testing Data: Since SVM is a supervised learning algorithm, the original DGA dataset is randomly split into two parts: a training dataset for deciding the hyperplane that can separate the samples into different classes (*i.e.* different fault types) and a testing dataset for verifying the classification accuracy of the algorithms. Note that the samples distribution among different classes in both training and testing dataset are kept as the same as that in the original dataset.

c)   Model Selection: In view of the possibility that classifying fault using DGA dataset is a non-linear problem, the study chooses three different kernels that helps SVM to solve non-linear classification. They are Radial Basis Function (RBF), Polynomial and Sigmoid kernels which equations are as follow:

- Polynomial: $K(x_i, x_j) = (ax_i^T x_j + r)^d$
- Radial Basis Function: $K(x_i, x_j) = e^{-\gamma \|x_i - x_j\|_2}$
- Sigmoid: $K(x_i, x_j) = \tanh(ax_i^T x_j + r)$

d)   1-vs-1 SVM: SVM is originally designed for binary classification. However, some datasets contain multi-class labels to learn from and DGA is an example of such datasets. Fortunately, a few extensions to SVM have been developed such as multi-level, one-against-all, one-against-one, and DAGSVM to solve this multi-category
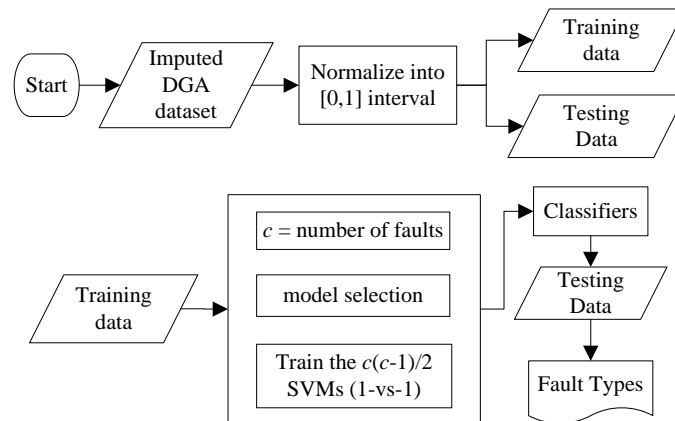

Figure 2. Flowchart of SVM.

issue. Following the recommendations made by a few researchers [21] [22] on the benefits of one-against-one method, this study adopts it to diagnose different fault types in DGA datasets. Based on the idea of "divide and conquer", one-against-one method decomposes the multi-class problem into $c(c\text{-}1)/2$ binary SVMs where c is the number of classes in the experimented dataset. Upon learning from the training set, a classifier is built which is used to classify faults in the testing set.

4. Experimental Designs and Results

4.1. Dataset

DGA datasets, with different percentages of actual missing values, are imputed and classified in this study. The first DGA dataset (named MAL) is obtained from a local Malaysian utility company which manage various transformers located throughout Malaysia, whilst IECDB10 [6] is the second. The characteristics of the datasets are shown in **Table 1**. A sample of DGA data consists of a number of dissolved gases in oil and the corresponding fault type as shown in **Table 2**. Dashes in **Table 2** represents missing values (missing gases).

4.2. Experimental Setup

One of the advantages of the kNN method in imputing missing values is that it requires only few parameters to tune: k and the distance metric, for achieving high estimation accuracy. Because both datasets in **Table 1** are quite small in size, this study chooses $k = \{1, 3, 5, 7, 9\}$. Two different distance metrics mentioned in Section 3.1 are compared. Therefore, each incomplete dataset will be filled-in using kNN for different values of k and for three different distance metrics.

After filling in the missing values in DGA datasets, using the imputed datasets as input, SVM was trained and its model predicted the fault types. The effectiveness of SVM is measured using accuracy defined as follows:

$$\text{Accuracy} = \frac{n_c}{n} \times 100\% \tag{7}$$

where n_c is the number of instances whose class labels are predicted correctly and n is the total number of instances in a test set. SVM with three different kernels were used as mentioned in Section 3.2.

As we want to estimate how accurately a predictive model will perform in practice, this study performed a 5-fold cross-validation where a dataset is divided into 5 disjoint sets (folds), 4 folds are used for training and the last fold is used for evaluation. This process is repeated 5 times, leaving one different fold for evaluation each time. Further, to reduce variability, 100 runs of this 5-fold cross-validation were carried out and the accuracy of

Table 1. The characteristics DGA datasets used in this study.

	IEC10DB	MAL
Number of samples	167	1228
Number of dissolved gases	7	9
Number of fault type	6	6
Instances with missing values (%)	27.54	76.07
Missing values (%)	7.96	14.21

Table 2. A DGA dataset with missing values.

H_2	CH_4	CO	CO_2	C_2H_4	C_2H_6	C_2H	Fault
9	5	403	1316	12	1	-	PD
61	-	217	2039	1	-	6712	PD
-	2504	1251	566	640	18	3251	Arcing
11	3	624	2043	56	3	5129	Arcing

(1) equals to the mean of the accuracies of all run.

This paper used a commercial software package MATLAB [23] to impute the missing values as well as classify the faults. The three kernels of SVM require different parameters which values affect the performance of SVM. However, optimizing of these parameters is not the purpose of this study, therefore the default parameter values provided by MATLAB were adopted. The effectiveness of our proposed method to diagnose power transformer faults was validated using the "before-and-after" experiment where the accuracies of each kernel learned on the original incomplete datasets and that learned on the imputed datasets were compared. Because MATLAB is one of the software that cannot deal with datasets with missing values unless they are deleted, we filled-in the missing values with zero instead, to enable MATLAB to perform classification task. However, we would like to remind that zero is not a missing value. As such, the before-and-after experiment is reduced to comparison between zero-filled datasets with imputed dataset using kNN imputation method.

4.3. Case Study 1: IEC10DB Dataset

Figure 3 and **4** shows the comparative performances of SVM_RBF, SVM_POLY, and SVM_SIG that predict fault using imputed IEC10DB dataset that was imputed using different values of k for two distance metrics. Using *CB* as the distance metric, the individual performance of the three kernels over different values of k were pretty much the same as evident in **Figure 3**. Among the three kernels, SVM_RBF registered the highest accuracy at $k = 5$. whilst SVM_SIG performed the worst. In the case of *EU* (**Figure 4**), similar observations as *CB* were recorded in terms of the influence of the different values of k to the individual performance of each kernel. Again, SVM_RBF outperformed the other methods and SVM_SIG performed the worst.

Table 3 shows the values of k that helped achieved highest accuracy for each SVM kernel using two distant
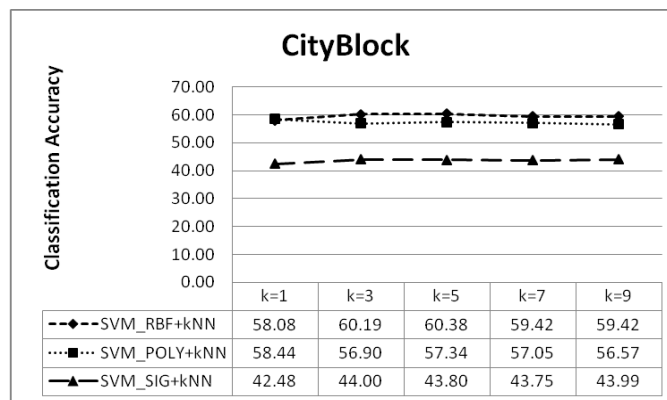

	k=1	k=3	k=5	k=7	k=9
SVM_RBF+kNN	58.08	60.19	60.38	59.42	59.42
SVM_POLY+kNN	58.44	56.90	57.34	57.05	56.57
SVM_SIG+kNN	42.48	44.00	43.80	43.75	43.99

Figure 3. SVMs performances on the IEC10DB imputed datasets using Cityblock.

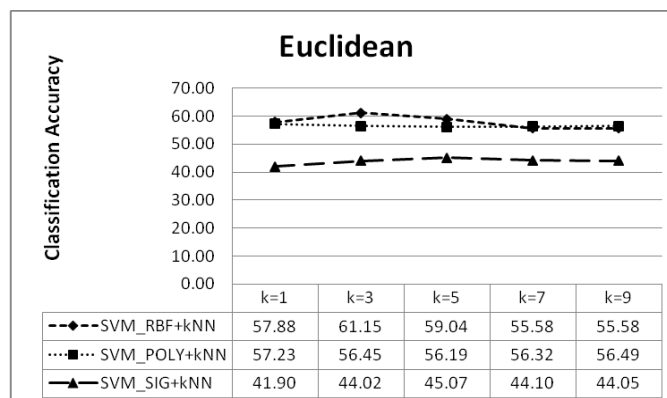|	k=1	k=3	k=5	k=7	k=9
SVM_RBF+kNN	57.88	61.15	59.04	55.58	55.58
SVM_POLY+kNN	57.23	56.45	56.19	56.32	56.49
SVM_SIG+kNN	41.90	44.02	45.07	44.10	44.05

Figure 4. SVMs performances on the IEC10DB imputed datasets using Euclidean.

metrics. When $k = 1$, SVM_POLY performed the best, whilst the other two kernels predicted better over higher k. It can be said, the choice of distance metric and the kernel determine the best k. The results of the before-and-after experiment for this dataset are shown in **Figure 5**. For this comparison, only the highest accuracy for each kernel over each distance metric was taken for comparison. It is noted that two kernels, namely SVM_RBF and SVM_SIG had better performance using imputed datasets than learning from zero-filled dataset. All imputed datasets obtained using all of the three kernels improved, albeit slightly, these two kernels. The opposite was reported by SVM_POLY.

4.4. Case Study 2: MAL

Figure 6 and **7** show the comparative performances of SVM_RBF, SVM_POLY, and SVM_SIG on each imputed dataset using the two distant metrics. It can be seen that, in the case of CB (see **Figure 6**), higher values of k increased the performances of all of the kernels. However, the performance of each kernel varies greatly among each other. SVM_RBF outperformed the other two kernels with big differences, and reached the highest accuracy at $k = 9$. SVM_SIG performed second after SVM_RBF, and the highest accuracy was obtained when $k = 9$. The worst performer was SVM_POLY which was at its most accurate when $k = 7$.

Similar observations as *CB* were seen when *EU* was the distance metric as shown in **Figure 7**. However, the effect of higher k to the individual performance of each kernel was more pronounced and better using *EU* as the distance metric. Again, SVM_RBF was the most effective of all and achieved the highest accuracy when $k = 9$. Next was SVM_SIG, which recorded the highest accuracy when $k = 5$. SVM_POLY was the least effective and it performed the best when $k = 5$. It can be seen that, *EU* improved the performance of each kernel better than *CB* for the MAL dataset.

Table 4 shows the values of k that help achieved highest prediction rate by the kernels. For this dataset, it is clear that all kernels predicted better over higher values of k. In fact SVM_RBF worked best when $k = 9$ for the two distance metrics. The results of the before-and-after experiment are shown in **Figure 8**. For this comparison, only the highest accuracy for each kernel over each distance metric was taken for comparison. For this dataset, all of the kernels had better performance using imputed datasets than learning from zero-filled dataset. Although SVM_POLY was the least effective, it benefited the most when missing values were imputed before learning took place.

4.5. Analysis

a) The value of the best k, the number of nearest neighbor is determined by the individual dataset and the choice of distance metric. However, larger values of k increase the kernels performance when high amount of missing values are found in a dataset as in the case of the MAL dataset.

b) For both of the datasets, imputed datasets using *EU* provide better performance for two kernels (SVM_RBF and SVM_SIG) than *CB*. For SVM_POLY, it works better with *CB* than *EU* on the small dataset IEC10DB. The opposite is true for the large dataset MAL.

Table 3. The best value of k for each kernel for IEC10DB dataset.

	Cityblock	Euclidean
SVM_RBF	$k = 5$	$k = 3$
SVM_POLY	$k = 1$	$k = 1$
SVM_SIG	$k = 3$	$k = 5$

Table 4. The best value of k for each kernel for the MAL dataset.

	Cityblock	Euclidean
SVM_RBF	$k = 9$	$k = 9$
SVM_POLY	$k = 7$	$k = 5$
SVM_SIG	$k = 9$	$k = 5$
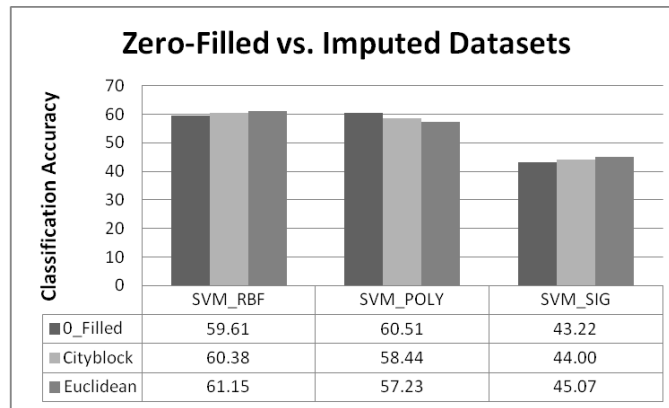
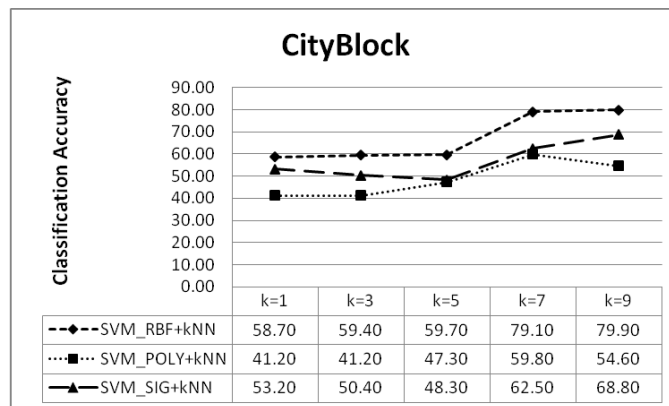**Figure 5.** The before-and-after comparative performances on the IEC10DB dataset.

	SVM_RBF	SVM_POLY	SVM_SIG
0_Filled	59.61	60.51	43.22
Cityblock	60.38	58.44	44.00
Euclidean	61.15	57.23	45.07

Figure 6. SVMs performances on the MAL imputed datasets using Cityblock.

	k=1	k=3	k=5	k=7	k=9
SVM_RBF+kNN	58.70	59.40	59.70	79.10	79.90
SVM_POLY+kNN	41.20	41.20	47.30	59.80	54.60
SVM_SIG+kNN	53.20	50.40	48.30	62.50	68.80
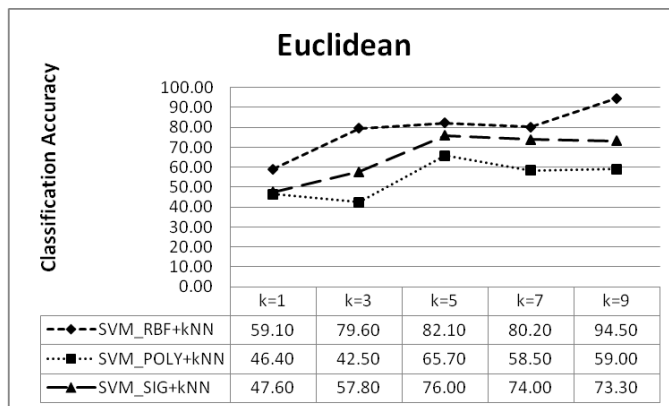
Figure 7. SVMs performances on the MAL imputed datasets using Euclidean.

	k=1	k=3	k=5	k=7	k=9
SVM_RBF+kNN	59.10	79.60	82.10	80.20	94.50
SVM_POLY+kNN	46.40	42.50	65.70	58.50	59.00
SVM_SIG+kNN	47.60	57.80	76.00	74.00	73.30

c) For both of the datasets, SVM_RBF is the most effective among the three SVM kernels. However, SVM_POLY and SVM_SIG are dataset-dependent. SVM_POLY performs better on the IEC10DB dataset than SVM_SIG. The opposite is true for the MAL dataset.

d) For most of the experimental settings, imputing missing values in a DGA dataset improve the performance of SVM kernels compared to learning from a zero-filled datasets. SVM_RBF and SVM_SIG performed better
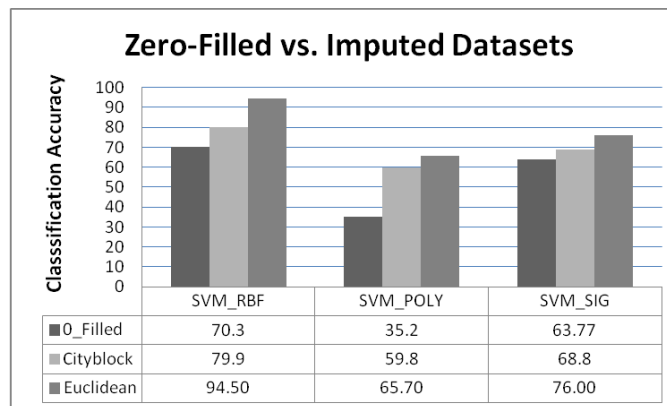
Figure 8. The before-and-after comparative performances on the MAL datasets.

on both of imputed datasets than on zero-filled dataset. Only SVM_POLY has opposite results in one of the dataset.

e) For DGA dataset having a large number of missing values like the MAL dataset, imputing missing values with kNN does improve the performance of all the kernels.

f) The best kernel is SVM_RBF which consistently outperforms the other two kernels and the best distant metric is *EU*. The combination of these two also produces the highest accuracy for both DGA dataset of different sizes and different percentages of missing values.

5. Conclusions

This paper proposes imputing the missing values found in a DGA dataset using the well-known kNN imputation method before letting SVM—a widely applied classification algorithm—learns and builds classifier to predict transformer faults. The experiments conducted using this proposed combination show the significant improvements in classifying the faults of power transformer especially when the percentage of missing values in DGA dataset is high. Moreover, by imputing missing values in a dataset enable some software to perform statistical analysis or machine learning task to be carried out without having to omit the samples that contain the missing values.

For future research, this study intents to conduct experiments using other combination of classification algorithms and/or imputation methods.

Acknowledgements

References

[1]   (2009) Guide for the Interpretation of Gases Generated in Oil-Immersed Transformers. IEEE Std C57.104-2008 (Revision of IEEE Std C57.104-1991).

[2]   Duval, M. Dissolved Gas Analysis and the Duval Triangle. http://www.avo.co.nz

[3]   Yang, Z., Tang, W.H., Shintemirov, A. and Wu, Q.H. (2009) Association Rule Mining-Based Dissolved Gas Analysis for Fault Diagnosis of Power Transformers. *Transactions on Systems*, *Man*, *and Cybernetics C*: *Applied Review*, **39**, 597-610.

[4]   Tang, W.H., Spurgeon, K., Wu, Q.H. and Richardson, Z.J. (2004) An Evidential Reasoning Approach to Transformer Condition Assessment. *IEEE Transactions on Power Delivery*, **19**, 1696-1703.
http://dx.doi.org/10.1109/TPWRD.2003.822542

[5]   Hall, M.A. (1999) Correlation-Based Feature Selection for Machine Learning. Ph.D. Thesis, University of Waikato, Waikato.

[6]   Duval, M. and de Pabla, A. (2001) Interpretation of Gas-in-Oil Analysis Using New IEC Publication 60599 and IEC TC 10 Databases. *IEEE on Electrical Insulation Magazine*, **17**, 31-41. http://dx.doi.org/10.1109/57.917529

[7]   Acuña, E. and Rodriguez, C. (2004) The Treatment of Missing Values and Its Effect on Classifier Accuracy. In: Banks, D., *et al.*, Eds., *Classification*, *Clustering*, *and Data Mining Applications*, Springer, Berlin Heidelberg, 639-647. http://dx.doi.org/10.1007/978-3-642-17103-1_60

[8]   Peng, L., Lei, L. and Naijun, W. (2005) A Quantitative Study of the Effect of Missing Data in Classifiers. *Proceedings of the Fifth International Conference on Computer and Information Technology*, 21-23 September 2005, 28-33. http://dx.doi.org/10.1109/CIT.2005.41

[9]   García-Laencina, P., Sancho-Gomes, J., Figueiras-Vidal, A. and Verleysen, M. (2009) K-Nearest Neighbours with Mutual Information for Simultaneous Classification and Missing Data Imputation. *Neurocomputing*, **72**, 1483-1493. http://dx.doi.org/10.1016/j.neucom.2008.11.026

[10]   Song, Q., Shepperd, M., Chen, X. and Liu, J. (2008) Can k-NN Imputation Improve the Performance of C4.5 with Small Software Project Data Sets? A Comparative Evaluation. *Journal of Systems and Software*, **81**, 2361-2370. http://dx.doi.org/10.1016/j.jss.2008.05.008

[11]   Schafer, J.L. and Graham, J.W. (2002) Missing Data: Our View of the State of the Art. *Psychological Methods*, **7**, 147-177. http://dx.doi.org/10.1037/1082-989X.7.2.147

[12]   Tsikriktsis, N. (2005) A Review of Techniques for Treating Missing Data in OM Survey Research. *Journal of Operations Management*, **24**, 53-62. http://dx.doi.org/10.1016/j.jom.2005.03.001

[13]   Jerez, J.M., Molina, I., García-Laencina, P.J., Alba, E., Ribelles, N., Martín, M. and Franco, L. (2010) Missing Data Imputation Using Statistical and Machine Learning Methods in a Real Breast Cancer Problem. *Artificial Intelligence in Medicine*, **50**, 105-115.

[14]   Shen-Wei, L. and Hsien-Chu, W. (2012) Effective Multiple-Features Extraction for Off-Line SVM-Based Handwritten Numeral Recognition. *Proceedings of the International Conference on Information Security and Intelligence Control* (*ISIC*), 14-16 August 2012, 194-197.

[15]   Niu, X.-X. and Suen, C.Y. (2012) A Novel Hybrid CNN-SVM Classifier for Recognizing Handwritten Digits. *Pattern Recognition*, **45**, 1318-1325. http://dx.doi.org/10.1016/j.patcog.2011.09.021

[16]   Rongbiao, Z., Zhao, S., Jin, Z., Zhenjun, Y., Ning, K. and Kang, H.J. (2010) Application of SVM in the Food Bacteria Image Recognition and Count. *Proceedings in the* 3*rd International Congress in Image and Signal Processing* (*CISP*), 16-18 October 2010, 1819-1823.

[17]   Li, W., Ruifeng, L. and Ke, W. (2013) Automatic Facial Expression Recognition Using SVM Based on AAMs. *Proceeding of the* 5*th International Conference on Intelligent Human-Machine Systems and Cybernetics* (*IHMSC*), 26-27 August 2013, 330-333.

[18]   Dong-Hui, L., Jian-Peng, B. and Xiao-Yun, S. (2008) The Study of Fault Diagnosis Model of DGA for Oil-Immersed Transformer Based on Fuzzy Means Kernel Clustering and SVM Multi-Class Object Simplified Structure. *Proceedings of the International Conference on Machine Learning and Cybernetics*, 12-15 July 2008, 1505-1509.

[19]   Bacha, K., Souahlia, S. and Gossa, M. (2012) Power Transformer Fault Diagnosis Based on Dissolved Gas Analysis by Support Vector Machine. *Electric Power Systems Research*, **83**, 73-79. http://dx.doi.org/10.1016/j.epsr.2011.09.012

[20]   Lv, G.Y., Cheng, H.Z., Zhai, H.B. and Dong, L.X. (2005) Fault Diagnosis of Power Transformer Based on Multi-Layer SVM Classifier. *Electric Power Systems Research*, **75**, 1-7. http://dx.doi.org/10.1016/j.epsr.2004.07.013

[21]   Chih-Wei, H. and Chih-Jen, L. (2002) A Comparison of Methods for Multiclass Support Vector Machines. *IEEE Transactions on Neural Networks*, **13**, 415-425. http://dx.doi.org/10.1109/72.991427

[22]   Allwein, E.L., Schapire, R.E. and Singer, Y. (2001) Reducing Multiclass to Binary: A Unifying Approach for Margin Classifier. *Journal of Machinery Learning and Research*, **1**, 113-141.

[23]   (2009) MATLAB, Version 7.9.0 (R2009b). The Math Works Inc.