www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 6 Issue 3 March 2017, Page No. 20765-20769
Index Copernicus value (2015): 58.10 DOI: 10.18535/ijecs/v6i3.65

A Comparative Study of SVM Kernel Functions Based on Polynomial Coefficients and V-Transform Coefficients

***D.Shanthini*[1], *M.Shanthi*[2], *Dr.M.C.Bhuvaneswari*[3]**
[1]Kumaraguru College of Technology, Department of ECE,
Coimbatore - 641 049, Tamil Nadu, India
shanthini.devaraaj@gmail.com
[2]Kumaraguru College of Technology, Department of ECE,
Coimbatore – 641 049, Tamil Nadu, India
shanthi.m.ece@kct.ac.in
[3]PSG College of Technology, Department of EEE,
Coimbatore – 641 004, Tamil Nadu, India
mcb@eee.psgtech.ac.in

Abstract: ***This paper presents a comparative study between Polynomial and V-transform coefficients for the fault classification in analog filter circuit using different kernel functions of Support Vector Machines (SVM). V-transform is a non-linear transformation which increases the sensitivity of polynomial coefficients with respect to circuit component's variation by three to five times. It makes the original polynomial coefficients monotonic. Support Vector Machine is used for fault classification in polynomial and V-transform coefficients. The classification accuracy in both Polynomial and V-transform coefficients are increased by varying the kernel parameters c and epsilon associated with the use of SVM algorithm for the different kernel functions. The SVM's are estimated in comparisons with the varied kernel functions by applying to the two feature sets. It is shown that the Pearson VII kernel function (PUK) provides good classification accuracy for the two feature sets compared to the other kernel functions such as Polynomial kernel function (POLY kernel) and the Radial Basis Function (RBF) kernel functions.***

Keywords: Polynomial Coefficients, V-transform coefficients, Support Vector Machines (SVM), Pearson VII kernel function (PUK), Polynomial kernel function (POLY kernel), Radial Basis Function (RBF).

1. Introduction

Machine learning is a type of Artificial Intelligence (AI) used in various fields which allows complex tasks to be solved. Information about spatially localized basis functions evolved as a popular paradigm in machine learning community. Kernel methods provide a powerful framework motivating algorithms that acts on general types of data and provides general relations such as classifications, regressions etc [1]. Kernel based methods were first applied in Support Vector Machines (SVM) [2].

In machine learning, support vector machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. In today's world, support vector machines along with kernel-based algorithms provide good classification results than Artificial Neural Networks (ANN's) for most of the benchmark problems [3]. Kernel methods used in SVM were applied to a variety of problems such as classification and regression [4]. Support vector machines have a malleable structure and they do not build an arbitrary model [5].

In machine learning, classification is one of the most important operation. Classification using SVMs are defined as Quadratic Programming (QP) problems which are solved by using many optimization algorithms [6] [7]. The neural networks are considered as the universal classifications of the measured data. The two classifications of the neural networks are the global and the local classifications [8]. The example of global neural network is the Multi-Layer Perception (MLP). The example of the local neural network is the Support Vector Machine utilizing different kernel functions. Choice of different kernel functions will result in different SVMs and different performances [9].

The different kernel functions available in SVMs are the Polynomial kernel, Radial Basis kernel, Pearson VII kernel functions. On selection of the kernel function, the different parameters has to be varied in order to obtain a higher classification accuracy. Choosing the best kernel function for a specific problem is still an ongoing research issue. Tsang et al. [10] proposed an idea for identifying a suitable kernel for the given data set.

Maji et al. [11] proposed a method for the correct estimation of intersection kernel SVMs which is almost simple and has a good classification accuracy but resulted in increase in runtime compared to RBF and POLY kernels due to large number of support vectors for each classifier [12].

Regardless of the ability of classification, problems still remain, particularly in choosing the efficient kernels of support vector machines for a specific application [13]. Examining new techniques and efficient methods for constructing an effective kernel function for plotting SVMs in a distinct application is an important research work in SVMs.

In this study, different kernel methods of SVMs such as RBF, POLY kernel and PUK are introduced to obtain higher classification accuracy by varying the kernel parameters. A comparison is made between the different kernel functions for the varied kernel parameters. It is shown that the PUK kernel function gives good classification accuracy in both the given feature sets.

The rest of the paper is organized as follows: Section 2 describes the construction of feature sets. Section 3 deals with

DOI: 10.18535/ijecs/v6i3.65

the description of SVM with varied kernel functions. Section 4, discuses the comparison of results between the kernel functions. Conclusion is presented in Section 5.

2. Feature Set Construction

The comparisons between the different kernel functions are performed for the two types of feature sets. The feature sets used for the proposed work consists of Polynomial coefficients and the V-transform coefficients. The feature sets are obtained from the benchmark circuit, the biquad filter which is used as the circuit under test (CUT) for the proposed work. Figure 1 illustrates the flowchart for the proposed work.

2.1. Circuit Under Test (CUT)

The benchmark circuit used as the CUT for the proposed work is the biquad filter. Biquad filters are typically active filters and implemented with a two integer-loop topology. The filter produces two types of output responses such as the band-pass output and the low-pass output. The final output of the biquad filter is the low-pass response which is used to obtain the coefficients for the feature set. Figure 2 presents the circuit of the biquad filter.

2.1.1 Transfer Function

The biquad filter is a type of linear filter that implements a transfer function that is the ratio of the two quadratic functions. The transfer function is the ratio of the output voltage to the input voltage which varies for various types of output. Using the transfer function of the biquad filter, the phase and magnitude of the frequency response of the circuit

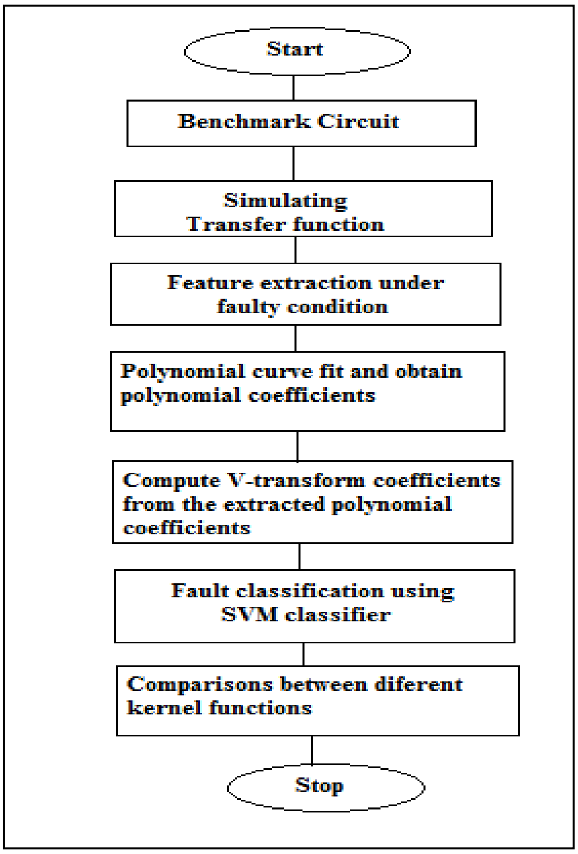

Figure 1: Flowchart for the proposed work

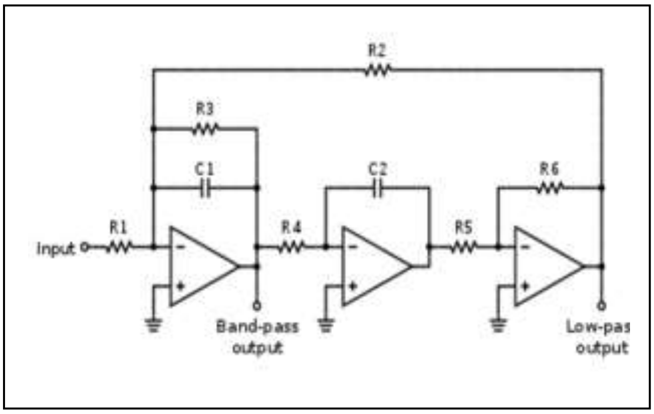

Figure 2: Biquad Filter Circuit

is obtained. The transfer function of the biquad filter with respect to its low-pass output is given by

$$\frac{V_0Lp(s)}{V_{in}(s)} = -\frac{\frac{R_4}{C_1R_2R_5R_6R_1}}{S^2+S\frac{1}{C_1R_2}+\frac{R_4}{C_1C_2R_3R_5R_6}} \qquad (1)$$

The Resistors R1, R2, R3, R4, R5, R6 and the capacitors C1 and C2 are the components of the CUT where the nominal values of the components are given by R1=R3=2.7$K\Omega$, R2=1.5$K\Omega$, R4=12$K\Omega$, R5=1$K\Omega$, R6=10$K\Omega$, C1=C2=10nF.

2.2. Fault Dictionary Creation

Fault dictionary belongs to the Simulation Before Test (SBT) technique. The fault dictionary for the biquad filter is obtained by simulating the transfer function of the circuit by injecting faults to the components. The fault is injected with a deviation of about $\pm50\%$ from the nominal value. The fault dictionary is constructed by using two types of feature sets.

2.2.1 Polynomial Coefficients

A method for identifying the parametric faults using Polynomial coefficients was proposed by Sindia S [14]. For the different faults injected on each component, different frequency response graphs will be attained. The obtained frequency response graphs are curve fitted with 9^{th} order polynomial curve fitting tool in MATLAB to produce ten polynomial coefficients. A feature set is then constructed with the obtained polynomial coefficients with their corresponding component values under fault condition.

2.2.2 V-Transform Coefficients

The significance of V-transform coefficients was proposed by Sindia S [15]. V-transform coefficients (VTC) are described as exponential functions of the modified polynomial coefficients. V-transform coefficients are explained as follows: if $C_1, C_2 \dots C_n$ are polynomial coefficients of the circuit under test, then their V-transform coefficients are expressed as $V_{c1}, V_{c2} \dots V_{cn}$ which are given by

$$V_{ci} = e^{\gamma C_i'} \; \forall \; 0 \le i \le n \qquad (2)$$

where C_i' are the modified polynomial coefficients defined as follows

$$\frac{dC_i'}{dp_j} = \left|\frac{dC_i}{dp_j}\right| \ \forall\, 0 \le i \le n \qquad (3)$$

The modified polynomial coefficients C_i' in (3) ensures that they are monotonic with the polynomial coefficients. γ is a sensitivity parameter which can be chosen according to the desired sensitivity. The second feature set is constructed with the obtained V-transform coefficients.

Fault dictionary is constructed for the two feature sets consisting of the Polynomial and the V-transform coefficients.

3. Support Vector Machines (SVM)

Support Vector Machines (SVMs) was proposed by Vapnik [6] for solving classification problems. SVMs were initially created to perform binary classification problems [7]. Later on it was extended to solve multi-class classification problems. The multi-class classification problems are generally dissolved into a number of binary problems, so that the standard SVMs are directly applied.

SVMs belong to the class of supervised learning algorithms in which the learning machine is given with the set of input values with their associated labels. SVMs construct a hyperplane that separates two classes which can also be extended to multi-class problems. Construction of the hyperplane makes SVM to achieve maximum separation between the classes.

Separation of the classes with a large margin minimizes the generalization error. The planes that are parallel to the hyperplane and which passes through one or more points in the data set are called bounding planes and the distance between these bounding planes is called the margin. SVM aims at finding the best hyperplane which maximizes this margin. Figure 3 shows the SVM with an optimal hyperplane.

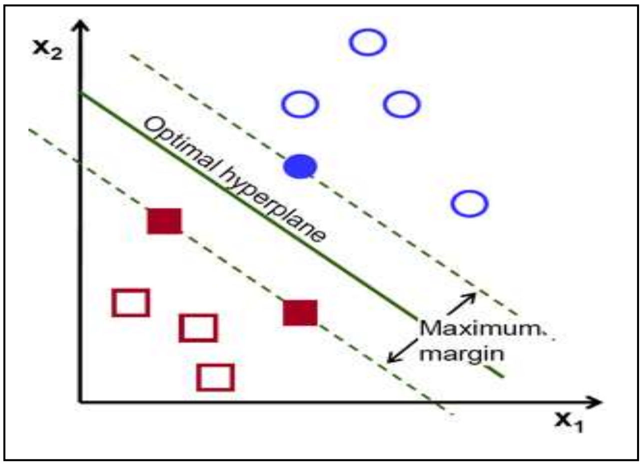

Figure 3: SVM with optimal hyperplane

The points falling on the bounding planes are called support vectors. The separating hyperplane can be written as

$$\omega.x + \beta = 0 \qquad (4)$$

where ω is a weight vector and β is a bias. The maximal margin is denoted mathematically by the formula

$$M = \frac{2}{\|W\|} \qquad (5)$$

where $\|W\|$ is the Euclidean norm of w. The notation to define the hyperplane is given by

$$f(X) = \beta_0 + \beta^T \times \qquad (6)$$

where β is known as the weight vector and β_0 the bias.

SVM classification is performed with different kernel functions such as Polykernel function, Radial basis kernel function and Pearson VII kernel function.

3.1 Kernel Functions

There are various kernel functions used with SVMs, but the choice of a particular kernel function to map the non-linear input space into a linear feature space depends highly on the nature of the data. As the nature of the data is unknown, the finest mapping function must be resolved experimentally by applying and validating various kernel functions producing the highest generalization performance. Therefore by adjusting the kernel parameters, the best kernel function can be determined.

3.1.1 Polynomial kernel (POLY kernel)

Polynomial kernels are commonly used with support vector machines which represents the similarity of vectors in the feature space over polynomials of the original variables. In a POLY kernel, K corresponds to an inner product in a feature space based on the mapping φ:

$$K(x,y) = \langle \varphi(x), \varphi(y) \rangle \qquad (7)$$

where x and y are inputs in the vector space.

3.1.2 Radial Basis kernel (RBF)

RBF is a popular kernel function used commonly with SVM classification. The RBF kernel on two samples $\times$ and $\times'$, represented as feature vectors in some input space is defined as

$$K(\times,\times') = \exp\left(-\frac{\|\times - \times'\|^2}{2\sigma^2}\right) \qquad (8)$$

where $\|\times - \times'\|^2$ is the squared Euclidean distance between the two feature vectors and σ is a free parameter.

3.1.3 Pearson VII kernel (PUK)

PUK kernel is an Universal kernel function generally applied to SVM [16]. PUK is very flexible and has possibility to change easily by adapting its parameters. Therefore it is possible to use Pearson VII kernel function as a general kernel which can replace the other kernel functions.

4. Experimental Results

The set of 1870 samples from the fault dictionary are used from which 1470 are used as training samples and the remaining 400 are used as testing samples. SVM classification is performed for the two feature sets consisting of Polynomial and V-transform coefficients using different kernel functions by varying the kernel parameters. The value of the exponent in the Polynomial kernel is chosen to be 1. The value of gamma

DOI: 10.18535/ijecs/v6i3.65

in the Radial basis function is set as 0.01 and the values of omega and sigma in the PUK kernel function are chosen as 1. Therefore by varying the complexity parameter C in the ranges between 10^{-6} to 10^{6} and choosing the values for the insensitive loss function ε as 0.1 and 10^{-12}, the accuracies for the kernel functions are measured.

Confusion matrix is used from which different performance measures between the kernel functions are compared. The following tables from 1 to 6 shows the improved results for the varied parameters.

Table 1: Simulation results for the Polynomial kernel function with $\varepsilon = 0.1$

Performance Measures	***ε = 0.1***					
	C = 0.001		*C = 1*		*C =10³*	
	PC	VC	PC	VC	PC	VC
Accuracy	12.5 %	12.5 %	82.5 %	84.25 %	93%	91.25 %
Time	0.01s	0.11s	0.02s	0.001s	0.01s	0.01s
Sensitivity	0.125	0.125	0.825	0.843	0.93	0.913
Specificity	0.125	0.125	0.025	0.023	0.01	0.013
Precision	0.016	0.016	0.844	0.881	0.94	0.928
F-measure	0.028	0.028	0.816	0.834	0.93	0.911

From the table PC refers to the Polynomial Coefficients and VC refers to V-transform Coefficients

Table 2: Simulation results for the Polynomial kernel function with $\varepsilon = 10^{-12}$

Performance Measures	***ε = 10⁻¹²***					
	C = 0.001		*C = 1*		*C =10³*	
	PC	VC	PC	VC	PC	VC
Accuracy	29%	31%	59%	84.5%	**95.5 %**	**93.75 %**
Time	0.05s	0.01s	0.01s	0.02s	0.02s	0.01s
Sensitivity	0.290	0.310	0.59	0.845	0.955	0.938
Specificity	0.101	0.099	0.059	0.022	0.006	0.009
Precision	0.196	0.235	0.597	0.891	0.967	0.951
F-measure	0.213	0.247	0.558	0.835	0.953	0.936

Table 3: Simulation results for the RBF kernel function with $\varepsilon = 0.1$

Performance Measures	***ε = 0.1***					
	C = 0.001		*C = 1*		*C =10³*	
	PC	VC	PC	VC	PC	VC
Accuracy	12.5 %	12.5 %	41.25 %	39%	82.75 %	91.75 %
Time	0.01s	0.02s	0.39s	0.38s	0.09s	0.06s
Sensitivity	0.125	0.125	0.413	0.39	0.828	0.918
Specificity	0.125	0.125	0.084	0.087	0.025	0.012
Precision	0.016	0.016	0.519	0.440	0.857	0.934
F-measure	0.028	0.028	0.406	0.333	0.816	0.916

Table 4: Simulation results for RBF kernel function with $\varepsilon = 10^{-12}$

Performance Measures	***ε = 10⁻¹²***					
	C = 0.001		*C = 1*		*C =10³*	
	PC	VC	PC	VC	PC	VC
Accuracy	29%	31%	36.5 %	38.75 %	**92.25 %**	**93.25 %**
Time	0.63s	0.44s	0.39s	0.38s	0.06s	0.06s
Sensitivity	0.290	0.310	0.365	0.388	0.923	0.933
Specificity	0.101	0.099	0.091	0.088	0.011	0.010
Precision	0.196	0.235	0.336	0.322	0.946	0.936
F-measure	0.213	0.247	0.319	0.327	0.916	0.932

Table 5: Simulation results for PUK kernel function with $\varepsilon = 0.1$

Performance Measures	***ε = 0.1***					
	C = 0.001		*C = 1*		*C =10³*	
	PC	VC	PC	VC	PC	VC
Accuracy	12.5 %	12.5 %	96.25 %	96.75 %	98.5 %	97.5 %
Time	0.03s	0.02s	0.23s	0.14s	0.13s	0.11s
Sensitivity	0.125	0.125	0.963	0.968	0.985	0.975
Specificity	0.125	0.125	0.005	0.005	0	0.003
Precision	0.016	0.016	0.966	0.973	1	0.982
F-measure	0.028	0.028	0.962	0.967	0.992	0.977

Table 6: Simulation results for PUK kernel function with $\varepsilon = 10^{-12}$

Performance Measures	***ε = 10⁻¹²***					
	C = 0.001		*C = 1*		*C =10³*	
	PC	VC	PC	VC	PC	VC
Accuracy	41.5 %	48.5 %	93%	97%	**100%**	**97.75 %**
Time	1s	0.59s	0.13s	0.08s	0.05s	0.03s
Sensitivity	0.415	0.485	0.93	0.97	1	0.978
Specificity	0.084	0.074	0.010	0.004	0	0.003
Precision	0.328	0.358	0.953	0.974	1	0.981
F-measure	0.328	0.38	0.925	0.969	1	0.977

The following two figures show the variation in accuracies obtained for the different kernel functions for the values of c = 10^3 and with epsilon values equal to 0.1 and 10^{-12} between the two feature sets.

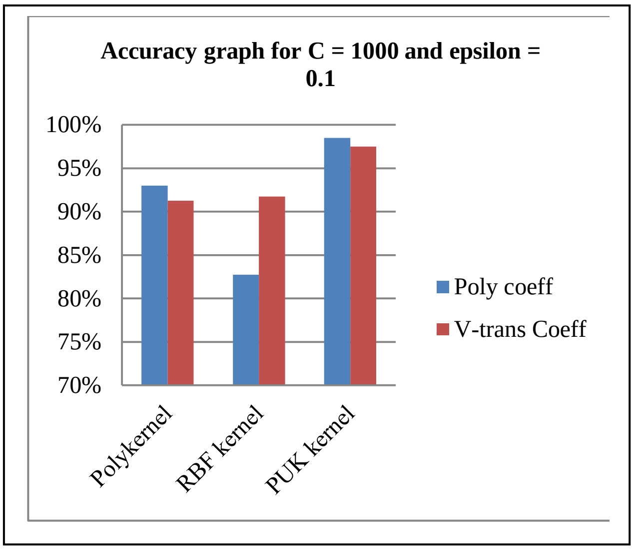

Figure 4: Accuracy plot for Polynomial Vs V-transform Coefficients with c = 10^3 and epsilon = 0.1

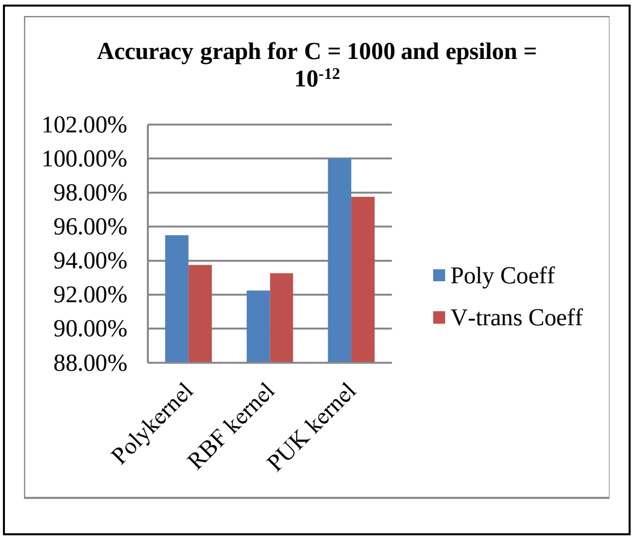

Figure 5: Accuracy plot for Polynomial Vs V-transform Coefficients with c = 10^3 and epsilon = 10^{-12}

5. Conclusion

It is observed that the SVM based on Polynomial kernel, RBF kernel and PUK kernel shows the similar performance on mapping the relation between the input and the output data. A comparison of simulation results for SVM classification based on POLY, RBF and PUK kernels show that the accuracy increases for both the Polynomial coefficients and the V-transform coefficients by choosing the complexity parameter value as 10^3 and the insensitive loss function ε value as 10^{-12}. This shows that for a maximum value of C and a minimum value of ε, the support vectors obtained will be maximum. Also from the comparison, PUK kernel produces higher accuracy compared to the Polynomial kernel and RBF kernel functions.

References

[1] Shawe-Taylor, J., & Cristianini, N. (2004). Kernel methods for pattern analysis. Cambridge university press.

[2] Burges, C. J. (1998). A tutorial on support vector machines for pattern recognition. Data mining and knowledge discovery, 2(2), 121-167.

[3] Meyer, D., Leisch, F., & Hornik, K. (2003). The support vector machine under test. Neurocomputing, 55(1), 169-186.

[4] Baudat, G., & Anouar, F. (2001). Kernel-based methods and function approximation. In Neural Networks, 2001. Proceedings. IJCNN'01. International Joint Conference on (Vol. 2, pp. 1244-1249). IEEE.

[5] Huang, T. M., Kecman, V., & Kopriva, I. (2006). Kernel based algorithms for mining huge data sets (Vol. 1). Heidelberg: Springer.

[6] Vapnik, V. (2013). The nature of statistical learning theory. Springer science & business media.

[7] Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine learning, 20(3), 273-297.

[8] Mohammad, M. N., Sulaiman, N., & Khalaf, E. T. (2011). A novel local network intrusion detection system based on support vector machine. Journal of Computer Science, 7(10), 1560.

[9] Williamson, R. C., Smola, A. J., & Schölkopf, B. (1999, March). Entropy numbers, operators and support vector kernels. In European Conference on Computational Learning Theory (pp. 285-299). Springer Berlin Heidelberg.

[10] Tsang, I. W., Kwok, J. T., & Cheung, P. M. (2005). Core vector machines: Fast SVM training on very large data sets. Journal of Machine Learning Research, 6(Apr), 363-392.

[11] Maji, S., Berg, A. C., & Malik, J. (2008, June). Classification using intersection kernel support vector machines is efficient. In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on (pp. 1-8). IEEE.

[12] Haasdonk, B. (2005). Feature space interpretation of SVMs with indefinite kernels. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(4), 482-492.

[13] Ektefa, M., Sidi, F., Ibrahim, H., Jabar, M. A., & Memar, S. (2011). A comparative study in classification techniques for unsupervised record linkage model. Journal of Computer Science, 7(3), 341.

[14] Sindia, S., Singh, V., & Agrawal, V. D. (2010, January). Parametric fault diagnosis of nonlinear analog circuits using polynomial coefficients. In VLSI Design, 2010. VLSID'10. 23rd International Conference on (pp. 288-293). IEEE.

[15] Sindia, S., Agrawal, V. D., & Singh, V. (2011, May). Non-linear analog circuit test and diagnosis under process variation using V-transform coefficients. In VLSI Test Symposium (VTS), 2011 IEEE 29th (pp. 64-69). IEEE.

[16] Üstün, B., Melssen, W. J., & Buydens, L. M. (2006). Facilitating the application of Support Vector Regression by using a universal Pearson VII function based kernel. Chemometrics and Intelligent Laboratory Systems, 81(1), 29-40.