Computational Materials Science 99 (2015) 316–325

Contents lists available at ScienceDirect

Computational Materials Science

journal homepage: www.elsevier.com/locate/commatsci

Comprehensive mechanical property classification of vapor-grown carbon nanofiber/vinyl ester nanocomposites using support vector machines

O. Abuomar [a,*], S. Nouranian [a,1], R. King [a,b], T.M. Ricks [c], T.E. Lacy [c]

[a] Center for Advanced Vehicular Systems (CAVS), Mississippi State, MS 39762, USA
[b] Department of Electrical and Computer Engineering, Mississippi State University, Mississippi State, MS 39762, USA
[c] Department of Aerospace Engineering, Mississippi State University, MS 39762, USA

ARTICLE INFO

Article history:
Received 30 September 2014
Received in revised form 17 December 2014
Accepted 23 December 2014

Keywords:
Support vector machines
Data mining
Knowledge discovery
Vapor-grown carbon nanofiber/vinyl ester
Confusion matrix
Cross validation

ABSTRACT

In the context of data mining and knowledge discovery, a large dataset of vapor-grown carbon nanofiber (VGCNF)/vinyl ester (VE) nanocomposites was thoroughly analyzed and classified using support vector machines (SVMs) into ten classes of desired mechanical properties. These classes are high true ultimate strength, high true yield strength, high engineering elastic modulus, high engineering ultimate strength, high flexural modulus, high flexural strength, high impact strength, high storage modulus, high loss modulus, and high tan delta. Resubstitution and 3-folds cross validation techniques were applied and different sets of confusion matrices were used to compare and analyze the classifier's resulting classification performance. The designed SVMs model is resourceful for materials scientists and engineers, because it can be used to qualitatively assess different nanocomposite mechanical responses associated with different combinations of the formulation, processing, and environmental conditions. In addition, the lead time required to develop VGCNF/VE nanocomposites for particular engineering application will be significantly reduced using the designed SVMs classifier. This work specifically present a framework for a fast and reliable classification of a large material dataset with respect to desired mechanical properties, and can be used for all materials within the context of materials science and engineering.

1. Introduction

The support vector machines (SVMs) methodology [1] is considered a widely used technique by the artificial intelligence community. It can be employed to design classifiers using datasets of different sizes and dimensions and from different knowledge domains. SVMs can be used for both supervised and unsupervised learning methodologies [1]. Supervised learning can be implemented using a relatively small number of data vectors (points). However, some prior knowledge of the problem is needed to assist the SVMs model in generalizing to unknown data vectors and thus predicting the correct quantity. Unsupervised learning typically requires a large number of data vectors within a particular dataset to adequately discover the relationship between the dataset's dimensions and to model the problem appropriately without over-training (over-fitting) the model [1]. The development of SVMs involves theory first, then implementation and experiments take place whereas other classifiers, like ANN follow a heuristic path, with applications and extensive experimentation preceding theory [1]. A significant advantage of SVMs is that other classifiers like ANN can suffer from multiple local minima whereas the solution to an SVM is global and unique. Two more advantages of SVMs are that that have a simple geometric interpretation and give a sparse solution. Unlike other techniques, the computational complexity of SVMs does not depend on the dimensionality of the input space. In addition, ANNs for example use empirical risk minimization, whereas SVMs use structural risk minimization [1]. The reason that SVMs often outperform other classifiers in practice, especially ANNs, is that they are less prone to overfitting [1].

SVMs classifiers generally perform poorly on highly unbalanced datasets because they are designed to generalize from sample data and output the simplest hypothesis that best fits the data, based on the principle of Occam's razor. This principle is embedded in the inductive bias of many machine learning algorithms including decision trees, which favor shorter trees over longer ones [2]. With

[*] Corresponding author at: Center for Advanced Vehicular Systems (CAVS), 200 Research Blvd., Mississippi State University, Mississippi State, MS 39762, USA. Tel.: +1 662 325 5576.

E-mail address: abuomar@cavs.msstate.edu (O. Abuomar).

[1] Present address: Department of Chemical Engineering, The University of Mississippi, University, MS 38677, USA.

http://dx.doi.org/10.1016/j.commatsci.2014.12.029
0927-0256/© 2014 Elsevier B.V. All rights reserved.

unbalanced data, the simplest hypothesis is often the one that classifies almost all instances as negative or all instances as positive. In addition, highly unbalanced datasets will have a negative effect on the designed classifier by making it too sensitive to noise and more prone to learn an erroneous hypothesis [2].

These problems are encountered on highly unbalanced datasets. According of what has mentioned in literature, an imbalance of 100 samples of one class to 1 sample of another class exists in fraud detection domains, even approaching 100,000 samples of one class to 1 sample of another class in other applications [2].

SVMs can also classify linearly and nonlinearly separable data into two or more classes [1]. SVMs have recently been employed as an application of data mining and knowledge discovery techniques in the context of materials science and engineering to facilitate the discovery of new knowledge [3–6]. For example, materials scientists can use SVMs to interpret experimental data. This activity can not only accelerate research, but also aid in the development of new materials with desired mechanical properties. In short, data mining approaches are being fueled by dynamic growth in the information technology sector and is driving the interest in SVMs, machine learning, information retrieval, and other knowledge representation in different engineering disciplines [7]. Abuomar et al. [8] applied an artificial neural network (ANN) technique to a dataset associated with the viscoelastic response of a vapor-grown carbon nanofiber (VGCNF)/vinyl ester (VE) nanocomposite material system. The ANN was trained using the resubstitution method and the 3-fold cross validation (CV) technique to predict the responses (*i.e.* storage modulus, loss modulus, and tan delta) with the minimal mean square error [8]. Nunes et al. [9] evaluated the efficiency and accuracy of artificial intelligence techniques to classify ultrasound signals, raw data and feature selection methods, background echo and backscattered signals acquired at frequencies of 4 and 5 MHz to characterize the microstructural kinetics of phase transformations on a Nb-base alloy, thermally aged at 650 and 950 °C for 10, 100 and 200 h. Papa et al. [10] implemented SVMs, Bayesian and Optimum-Path Forest (OPF) based classifiers, and also the Otsu's method for automatic characterization of particles in metallographic images. De Albuquerque et al. [11] presented an ANN model to automatically segment and quantify material phases from SEM metallographic images and then the results were compared to a commercial software used for quantifying material phases from metallographic images. The results of the new ANN model were precise, reliable and more accurate and faster than the commercial software [11]. In addition, De Albuquerque et al. [12,13] presented a comparative analysis between backpropagation multilayer perceptron and self-organizing maps (SOMs) topologies applied to segment microstructures from metallographic images as well as they applied an ANN computational solution to segment and quantify the constituents of metallic materials from images. As another application of radiographic images segmentation task, an ANN model was employed to evaluate the delamination in laminate plates due to drilling operation [14]. Roberts et al. [15] presented a model that classifies different materials based on their microstructure. Based on microstructural characteristics such as Haralick variables [16], the Euler parameter [15], and the fractal dimension [15], the designed SVMs classifier identifies the appropriate class of given material sample [15]. Swaddiwudhipong et al. [17] utilized and implemented least squares support vector machines (LS-SVMs) [18]. Four LS-SVMs models that simulate the relationship between the elasto-plastic material properties and indentation load–displacement characteristics were designed; it was determined that the LS-SVMs technique was robust in determining the power hardening parameters given the fact that no iterative approaches were used [18]. Hu et al. [19] used knowledge discovery to resolve the problem of materials science image data sharing. Different annotations for non-structured materials science data were developed that utilize a complete ontology-based approach with the aid of semantic web technologies [19]. Sabin et al. [20] used a Gaussian process framework as a statistical technique to predict the output (*i.e.* the mean logarithm of grain size (D)) based on a probability distribution function over the training dataset. This framework was trained based on the available input variables (*i.e.* Strain, temperature (°C), and annealing time (s)) and tested to make the corresponding predictions and estimations.

In this work, a specific class of advanced engineering materials was studied, *i.e.*, polymer nanocomposites [21]. These materials have multifunctional properties and are extensively being used for fuel cell, aerospace, automotive, catalysis, biomedical, and other engineering applications. For example, nano-enhanced polymer composites meet the requirements of improved stiffness properties and energy absorption characteristics in automotive structural applications [22]. They have been the subject of extensive research recently [23,24]. Abuomar et al. [25] applied data mining and knowledge discovery techniques in order to analyze a thermosetting viscoelastic VGCNF/VE nanocomposite material system [26–29]. These techniques included SOMs, which are sometimes referred to as Kohonen maps [30,31] and fuzzy C-means (FCM) clustering [32,33]. The SOMs were used to determine the VGCNF/VE nanocomposite systems that produce the same storage and loss modulus for the lowest cost [25]. The FCM algorithm has also been used to discover some of the mechanical and physical patterns in the VGCNF/VE nanocomposite behavior after using the principal component analysis (PCA) technique to reduce the number of dimensions in the original dataset [34].

The current knowledge of the influence of formulation, processing, and environmental factors on the mechanical behavior of VGCNF/VE nanocomposites has been expanded in this study. This was accomplished by including a wider range of measured mechanical properties, *i.e.*, viscoelastic [26], compressive and tensile [35], flexural [36], and impact strengths [29]. Abuomar et al. [37] implemented this idea initially for a smaller dataset, where SVMs technique was used to analyze and classify a VGCNF/VE dataset, including viscoelastic data, compressive and tensile property data, and flexural property data into three classes of desired mechanical properties, i.e., high storage modulus, high true ultimate strength, and high flexural modulus. This new study, however, provides a more general and comprehensive insight into the mechanical behavior of VGCNF/VE nanocomposites for data mining purposes by including the VGCNF/VE impact strengths data as well as classifying and analyzing ten desired mechanical properties instead of three. The application of data mining and knowledge discovery techniques to a comprehensive dataset of mechanical responses of polymer nanocomposites is unprecedented and novel. The SVMs technique is used in this work to separate the new VGCNF/VE nanocomposite test data into ten different desired mechanical property classes. Thus, an unknown VGCNF/VE sample whose configuration is not represented by the current dataset can be easily identified, analyzed, and classified into its corresponding VGCNF/VE mechanical class without the need to conduct expensive and time-consuming experiments. Materials scientists and engineers can use the results of this study as a guideline to efficiently design or optimize a material system for a certain engineering application. The lead time required to develop a new material system for a specific engineering application can be significantly reduced using this study's fast and reliable qualitative assessment.

2. Materials and methods

The majority of data samples used in this work were generated using various statistics-based designed experiments, utilizing a general mixed level full factorial and central composite designs

[26–29,35,36]. Different datasets were merged into a larger one incorporating a total of 583 data points, *i.e.* 240 viscoelastic, 60 flexural, 172 compression, 93 tension, and 18 impact strength data points for VGCNF/VE nanocomposites treated at different formulation and processing conditions. Therefore, the VGCNF/VE dataset used in this study is not highly unbalanced. Each data point corresponds to combinations of nine input design factors and ten output responses. The input factors of the new VGCNF/VE dataset are curing environment (air vs. nitrogen), use or absence of a dispersing agent, strain rate, mixing method (ultrasonication, high-shear mixing, combination of both), VGCNF weight fraction, VGCNF type (pristine vs. oxidized), high-shear mixing time, ultrasonication time, and temperature. The output factors (*i.e.*, measured properties) are true ultimate strength, true yield strength, engineering elastic modulus, engineering ultimate strength, flexural modulus, flexural strength, impact strength, storage modulus, loss modulus, and tan delta. These nine inputs and ten output factors (classes) correspond to the different experimental formulation and processing conditions of the new merged VGCNF/VE nanocomposites system. Therefore, the effectiveness of the SVMs technique implemented in this study is that materials scientists and engineers can select the optimal manufacturing combination of input factors that yield the desired mechanical property responses. For example, high loss modulus and tan delta are desired in automotive applications, where crash situations necessitate a high energy dissipation capability exhibited by the material system.

The inputs levels and ranges are given in Table 1. This table includes also the outputs' ranges of high mechanical property responses. These values were used to train and develop the SVMs model implemented in this study.

The choice of optimal input level combinations is based on several industrial measures, among which are minimum fabrication cost, fastest or most time-efficient fabrication, and achievement of highest mechanical properties for the resulting VGCNF/VE nanocomposites. Often a combination of two or more of these measures are desired.

In order to replace some of the missing and unknown data fields in the new dataset, different data interpolation techniques were implemented in this work [38]. These techniques include linear interpolation and spline interpolation. However, spline interpolation not only avoids the problem of Runge's phenomenon [39] but it also yields a low interpolation error regardless of the polynomial degree used for the spline [38].

3. Theory/calculation

This study incorporates nine input and ten output design factors. Therefore, the dataset represents a nineteen-dimensional (19-D) analysis case. Curing environment, use or absence of dispersing agent, mixing method, and VGCNF type are considered qualitative factors, so they are represented by a numeric code for analysis purposes. Other dimensions in the dataset are quantitative. Therefore, their values were normalized using standardized scores since the original values have different ranges.

Before applying these techniques, a brief explanation of the SVMs operations, resubstitution, and cross validation techniques is given in the next section.

3.1. SVMs operations

The goal of an SVMs classifier is to define a separating hyperplane between the points belonging to two distinct classes and maximize the distance between these points and the hyperplane. This maximum distance is referred to as the margin. This concept is illustrated in Fig. 1 [1] for linearly separable data. For nonlinearly separable data, the resulting hyperplane and margin has a complex, nonlinear form as shown in Fig. 2 [1].

3.1.1. SVMs operations for two-class linearly separable data

The margin (m) (indicated in Fig. 1) is given by the relation:

$$m = \frac{|g(x)|}{\|w\|} \tag{1}$$

where $g(x)$ is the discriminant function used to separate and classify the data vectors into corresponding classes and w is the weight vector used by SVMs model. The weight vector is scaled so that the value of $g(x)$ at the closest point to the separating hyperplane is

Table 1
The experimental design factors, their levels and the high ranges of the mechanical property responses [26–29,35,36].

Factors	Level/range			
Inputs	1	2	3	4
Curing environment	Air	Nitrogen		
Use of dispersing agent	Yes	No	–	–
Strain rate (/second)	0.0001–2537.00			
Mixing method	US[a]	HS[b]	HS/US	–
VGCNF fiber loading (phr[c])	0.00 – 1.00			
VGCNF type	Pristine	Oxidized	–	–
High shear mixing time (min)	0.00–100.00			
Sonication time (min)	0.00–60.00			
Temperature (°C)	30 °C	60 °C	90 °C	120 °C
Outputs (Mechanical property responses)				
High true ultimate strength (MPa)	223.20–255.6			
High true yield strength (MPa)	180.00–198.30			
High engineering elastic modulus (GPa)	3.51–4.31			
High engineering ultimate strength (MPa)	68.2–84.7			
High flexural modulus (GPa)	3.15–3.69			
High flexural strength (MPa)	80.90–117.10			
High impact strength (J/m[d])	13.83–18.00			
High storage modulus (GPa)	2.58–2.77			
High loss modulus (MPa)	164.0–207.7			
High tan delta	0.14–0.31			

[a] Ultrasonication.
[b] High-shear mixing.
[c] Parts per hundred parts of resin.
[d] Joule per meter.

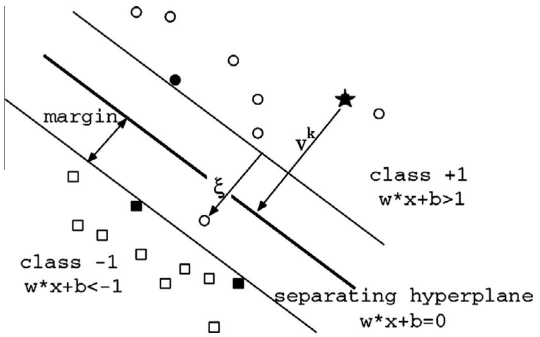

Fig. 1. The SVMs model: the separating hyperplane along with the maximum margin for linearly separable data [1].

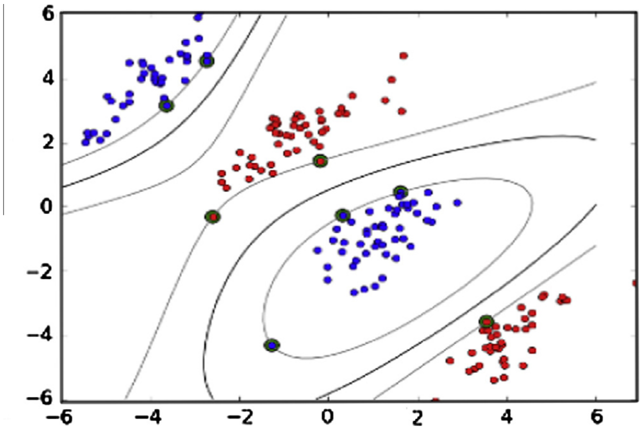

Fig. 2. An example of the SVMs model for nonlinearly separable data. This case involves the introduction of a new set of "slack" variables (ξ) [1].

equal to 1 for class one and −1 for class two. Alternatively, in the case of two-class SVMs model, the goal is to have a margin such that:

$$m = \frac{1}{\|w\|} + \frac{1}{\|w\|} = \frac{2}{\|w\|} \tag{2}$$

and this requires that:

$$\begin{aligned} &w^T x + w_0 \geqslant 1 \ldots \text{ for } x \in \text{class } 1 \\ &w^T x + w_0 \leqslant -1 \ldots \text{ for } x \in \text{class } 2 \end{aligned} \tag{3}$$

where w_0 is the weight bias used to specify how much the margin is away from the origin and x is the matrix for all data points (*i.e.*, data vectors).

The values of support vectors λ are the data points located on the margin borders, and their values are greater than zero. Since λ values that are less than zero are not considered support vectors, the corresponding data points belong to either class one or class two.

The theory of SVMs states that for each data vector x_i, there must be a class indicator (say y_i). The task is to find w and w_0 such that the cost function J is minimized:

$$J(w, w_0) = \frac{1}{2}\|w\|^2 \tag{4}$$

subject to

$$y_i(w^T x + w_0) \geqslant 1 \quad \text{for } i = 1, 2, \ldots, N \tag{5}$$

This nonlinear optimization task can be solved using a quadratic programming optimization algorithm whereby a quadratic function of some real-valued variables is maximized subject to linear constraints [1].

In SVMs, the primal form of the Lagrangian function L_p can be used:

$$L_p(w, w_0, \lambda) = \frac{1}{2} w^T w - \sum_{i=1}^{N} \lambda_i \left[y_i (w^T x_i + w_0) - 1 \right] \tag{6}$$

where λ is the Lagrange multiplier vector (*i.e.*, support vector) and λ_i is the Lagrangian multiplier.

The Lagrangian function is subject to a set of constraints defined by the Karush–Kuhn–Tucker (KKT) conditions [1]:

$$\frac{\partial}{\partial w} L(w, w_0, \lambda) = 0 \tag{7}$$

$$\frac{\partial}{\partial w_0} L(w, w_0, \lambda) = 0 \tag{8}$$

$$\lambda_i \geqslant 0 \quad \text{for } i = 1, 2, \ldots, N$$

$$\lambda_i [y_i (w^T x_i + w_0) - 1] \quad \text{for } i = 1, 2, \ldots, N \tag{9}$$

If Lagrangian function is combined with Eqs. (7) and (8), the SVMs optimization task is to minimize $L_p(w, w_0, \lambda)$, subject to the following constraints:

$$w = \sum_{i=1}^{N} \lambda_i y_i x_i \tag{10}$$

$$\sum_{i=1}^{N} \lambda_i y_i = 0 \tag{11}$$

$$\lambda_i \geq 0$$

If the equalities above are substituted into $L_p(w, w_0, \lambda)$, the final form of SVMs optimization task for the two-class linearly separable case will be to maximize the dual form of the Lagrangian, $L_D(w, w_0, \lambda)$, with respect to λ:

$$\max \ L_D(w, w_0, \lambda) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_i \lambda_j y_i y_j x_i^T x_j \tag{12}$$

subject to the following constraints in Eq. (11).

The weight bias term can be calculated from Eq. (5). For example, if $\lambda_1 > 0$, the corresponding w_0 can be found from the relation:

$$y_1 (w^T x_1 + w_0) = 1 \tag{13}$$

Therefore,

$$w_0 = \frac{1}{y_1} - w^T x_1 \tag{14}$$

3.1.2. SVMs operations for two-class non-linearly separable data

A visualization of the SVMs implementation of nonlinearly separable data is illustrated in Fig. 2.

A new set of "slack" variables, ξ_i, is introduced, such that:

$$y_i [w^T x + w_0] \geqslant 1 - \xi_i \tag{15}$$

In this context, the following scenarios must be taken into account:

- correct classification of the data points x_i is obtained if $\xi_i = 0$
- x_i will be inside the band (inside the margin) if $0 \leqslant \xi_i \leqslant 1$
- x_i is misclassified (the SVMs model classifies x_i in a different class than what it actually should belong to) if $\xi_i > 1$

The closely related cost function in this case (in *primal form*) is to minimize $J(w, w_0, \xi)$ such that:

$$J(w, w_0, \xi) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \tag{16}$$

subject to the constraints:

$$y_i[w^T x + w_0] \geqslant 1 - \xi_i \tag{17}$$
$$\xi_i \geqslant 0 \tag{18}$$

for $i = 1, 2, \ldots, N$, where C is a positive constant that balances between the margin size and the misclassification instances. The choice for C also determines the number of support vectors and the overall performance of the SVMs model.

The corresponding Lagrangian function in this case becomes:

$$L(w, w_0, \xi, \lambda, \mu) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\mu_i\xi_i - \sum_{i=1}^{N}\lambda_i[y_i(w^T x_i + w_0) - 1 + \xi_i] \tag{19}$$

where λ and μ are the Lagrangian vectors. The corresponding KKT conditions are:

$$\frac{\partial}{\partial w}L(w, w_0, \xi, \lambda, \mu) = 0 \tag{20}$$
$$\frac{\partial}{\partial \xi_i}L(w, w_0, \xi, \lambda, \mu) = 0 \tag{21}$$
$$\frac{\partial}{\partial w_0}L(w, w_0, \xi, \lambda, \mu) = 0 \tag{22}$$
$$\mu_i\xi_i = 0 \tag{23}$$
$$\lambda_i[y_i(w^T x_i + w_0) - 1 + \xi_i] = 0 \tag{24}$$

$$\lambda_i \geq 0$$

$$\mu_i \geq 0$$

for $i = 1, 2, \ldots, N$

The goal for the non-linearly separable case is to make the margin as large as possible, but at the same time, make the number of data points with $\xi > 0$ as small as possible. In this case, the misclassification mistakes and encountering cases where there are data points inside the margin even though the classification is correct will be avoided. Therefore, the Lagrangian function $L(w, w_0, \xi, \lambda, \mu)$ (in *primal form*) can be minimized subject to the following constraints:

$$w = \sum_{i=1}^{N}\lambda_i y_i x_i \tag{25}$$

$$\sum_{i=1}^{N}\lambda_i y_i = 0 \tag{26}$$

$$\begin{aligned} &C - \mu_i - \lambda_i = 0 \\ &\lambda_i \geqslant 0,\ \mu_i \geqslant 0 \end{aligned} \tag{27}$$

If the above equality constraints are substituted in the Lagrangian, the final *dual form* format of nonlinearly separable data $L_D(w, w_0, \xi, \lambda, \mu)$ is maximized with respect to λ such that:

$$\max\ L_D(w, w_0, \xi, \lambda, \mu) = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j x_i^T x_j \tag{28}$$

subject to the constraints:

$$0 \leqslant \lambda_i \leqslant C, \quad \text{for } i = 1, 2, \ldots, N \tag{29}$$

$$\sum_{i=1}^{N}\lambda_i y_i = 0 \tag{30}$$

Another important aspect of SVMs development is the kernel function, which takes the optimization problem from a lower space to a higher space. This kernel function is a function of x_i and x_j, shown above for the dual form. So, the SVMs optimization task reduces to the minimization of $L(w, w_0, \xi, \lambda, \mu)$ with respect to λ such that:

$$L_D(w, w_0, \xi, \lambda, \mu) = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j K(x_i, x_j) \tag{31}$$

subject to the constraints in Eqs. (22) and (23). $K(x_i,x_j)$ is the kernel function.

The following are some of the typical kernels:

- Polynomial:

$$K(x, z) = (x^T z + 1)^Q \quad Q > 0 \tag{32}$$

where Q is the polynomial degree;

- Radial basis function (RBF):

$$K(x, z) = \exp\left(-\frac{\|x - z\|^2}{2\sigma^2}\right) \tag{33}$$

where σ^2 is the standard deviation;

- Hyperbolic Tangent:

$$K(x, z) = \tanh(\beta x^T z + \gamma) \tag{34}$$

where β, γ are constants;

- Dot product:

$$K(x, z) = x^T z \tag{35}$$

The following strategy can be used to assign each data point (data vector) to the corresponding class:

$$\begin{aligned} &g(x) = \sum_{i=1}^{NS}\lambda_i y_i K(x_i, x) + w_0 > 0, \text{then} \\ &x_i \in \text{class 1} \\ &g(x) = \sum_{i=1}^{NS}\lambda_i y_i K(x_i, x) + w_0 \leqslant 0, \text{then} \\ &x_i \in \text{class 2} \end{aligned} \tag{36}$$

where $g(x)$ is the discriminant function, NS is the number of support vectors, and x_i is a data vector in the dataset being optimized.

In order to tackle the problem of unbalanced VGCNF/VE dataset (*i.e.* 240 viscoelastic, 60 flexural, 172 compression, 93 tension, and 18 impact strength data points), when designing the SVMs classifier a greater penalty to misclassification errors related with the less likely instance (impact strength data points in our case) was assigned, rather than assigning equal error weight which results in an undesirable classifier that assigns everything to the majority samples (viscoelastic data points).

Another approach to solve this problem is to preprocess the data by oversampling the majority class or undersampling the minority class in order to create a balanced dataset [2].

However, recent studies combine both of these approaches together to improve the performance of the SVMs classifier compared to applying any one approach. Specifically, this new approach has the following steps [2]:

1. Not undersampling the majority instances since they lead to loss of information.
2. Using different error costs for different classes to push the boundary of the SVMs hyperplane away from the minority instances.

Table 2
Description of VGCNF/VE mechanical property classes characterized in the study.

Class designation	Classes
C1	Specimens with high true ultimate strength
C2	Specimens with high true yield strength
C3	Specimens with high engineering elastic modulus
C4	Specimens with high engineering ultimate strength
C5	Specimens with high flexural modulus
C6	Specimens with high flexural strength
C7	Specimens with high impact strength
C8	Specimens with high storage modulus
C9	Specimens with high loss modulus
C10	Specimens with high tan delta

3. Using Synthetic Minority Oversampling Technique (SMOTE) to make the minority instances more densely distributed in order to make the boundary better defined.

Furthermore, some studies suggest using Granular Support Vector Machines Repetitive Undersampling algorithm (GSVMRU) and consider it the best in terms of both effectiveness and efficiency [40]. GSVM-RU is effective as it can minimize the negative effect of information loss while maximizing the positive effect of data cleaning in the undersampling process. GSVM-RU is efficient by extracting much less support vectors, and hence greatly speeding up SVM prediction [40].

Resubstitution and 3-fold cross validation techniques were used after the SVMs technique to characterize the specimens that have desired VGCNF/VE properties. Each specimen was separated into an appropriate VGCNF/VE mechanical property class. These classes are shown in Table 2 where C1 denotes class one, C2 denotes class two, and so on.

3.2. Resubstitution method

In resubstitution method [41], the entire dataset is used to train the SVMs model and the same dataset is used for testing (validation). This method is computationally efficient and ensures that the SVMs model generalizes well to correctly classify unknown classes based on combinations of inputs and outputs. A good generalization is achieved when the apparent error (AE) is minimized [32]. The AE is defined as:

$$\mathrm{AE} = \frac{1}{N}\sum_{i=1}^{N}|t_i - a_i| \qquad (37)$$

where N is the total number of samples, t_i is the targeted class of the sample in binary classification (*i.e.*, 1 if the sample belongs to one class and 0 if it belongs to the other class), and a_i is the actual SVMs binary classification value (0 or 1).

Although several SVMs architectures and training algorithms are available, the SVMs classifier for two nonlinearly separable data is the most commonly used one and was utilized in this study [1]. However, the designed SVMs model was implemented in ten stages using a one-against-all (OAA) strategy [42] because this study deals with separating the VGCNF/VE specimens into ten different distinct property classes. For example, in the first stage, specimens belonging to C2–C10 were combined and compared against specimens belonging to C1. Similarly, in the second stage, specimens belonging to C1 and C3–C10 were combined in one class and compared against specimens belonging to C2, and so on with all stages. Finally, the classification information from these stages was combined in order to determine the ten distinct property classes. This SVMs model assumed a non-linear relationship between the input and output variables and the corresponding class associated with each data point.

3.3. Cross validation technique

Cross validation (CV) techniques [41] use available data to train the SVMs classifier. First, the dataset is randomly partitioned into a training set and a test set. The training samples are further partitioned into two disjoint subsets: (1) the estimation subset, which is used to select the SVMs, and (2) the validation subset, which is used to test or validate the developed SVMs classifier [43]. Therefore, several candidate SVMs classifiers are obtained and then the "best" one is selected [43]. Currently, there are four different CV methods: holdout CV, early-stopping method of training, multifold CV, and leave-one-out CV. The following is a brief explanation of each of these methods; further details can be found in [44].

(1) **Holdout CV**: If a random number, r, is defined in the interval $[0, 1]$, then $(1 - r)N$ samples are allotted to the estimation subset, and the remaining rN samples are used for validation, where N is the total number of samples. The final SVMs model is the one yielding the minimum classification error. However, this method is computationally expensive.

When the complexity of the target function (mapping of input–output and the corresponding classes) is small compared to the sample size N, the validation performance is relatively insensitive to the choice of r. When the target function becomes more complex relative to the sample size N, the choice of r has a more pronounced effect on cross-validation performance. However, a single fixed value of r (*e.g.*, 0.2) is nearly optimal for a wide range of target functions.

(2) **Early-stopping method of training**: The training procedure can be stopped earlier before the classification error becomes too low in order to yield good generalization. The best point to stop training can be determined by the periodic "estimation-followed-by-validation" process as shown in Fig. 3. After some periods of training, say five epochs (*i.e.*, five step iterations for all training samples), the classification error based on validation sample is then measured. When the validation phase is completed, the training is resumed for another epoch(s). Finally, when the classification error based on validation sample starts to increase, the training process is terminated even if the classification error for the training samples continues to decrease.

(3) **Multifold CV**: A disadvantage of the holdout method is that not all samples are used for validation. Instead, in multifold validation, the N samples are divided into K subsets. At each

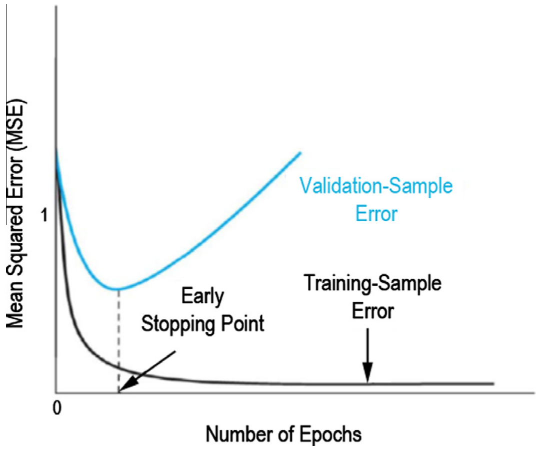

Fig. 3. Illustration of the early-stopping rule based on cross validation [44].

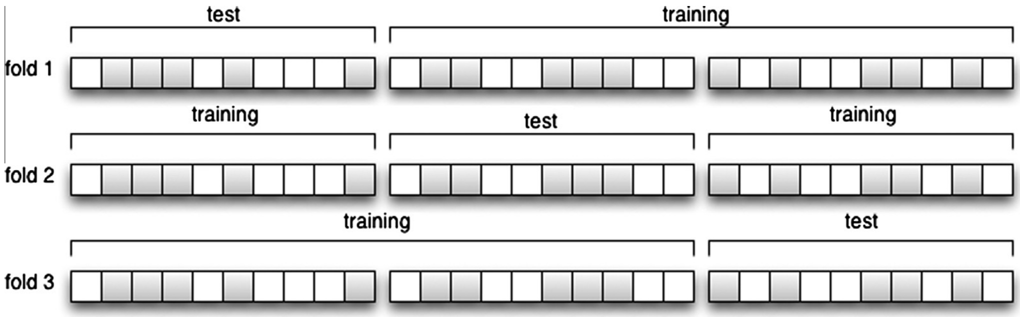

Fig. 4. Illustration of the 3-fold method of cross validation on a data sample of 30 elements [43].

fold, one subset is used for validation and the remaining $K - 1$ subsets for training. This process is continued until each subset is used for validation once. In this study, 3-fold CV was implemented and the performance was assessed by averaging the classification performance over all the trials. In Fig. 4, an illustration of a 3-fold cross validation is shown.

(4) **Leave-one-out CV**: When the available number of samples, N, is severely limited, an extreme form of multifold validation known as *leave-one-out* validation can be used. In each trial, $N - 1$ samples are used for training and the one left out can be used for testing. The process is repeated N times until each sample is used for validation exactly once.

4. Results and discussion

The classification process starts by dividing the VGCNF/VE data into training and test sets. These sets are applied to the developed SVMs model. That is, materials scientists and engineers can provide any new formulations or fabrication conditions as inputs combinations to the SVMs model and the corresponding desired mechanical property response(s) these combinations were classified into. An illustration of the model's operation is given in Fig. 5.

After choosing a combination of nine inputs (processing parameters) and supplying this input vector to the designed SVMs classifier, the model will classify the input vector into the corresponding desired mechanical property response(s)(*i.e.* classes) as shown in Table 2. In this particular example, the chosen input vector yielded high true ultimate strength (C1) and high engineering ultimate strength (C4).

Two techniques were used for the performance evaluation of the SVMs classifier: the resubstitution method and the 3-fold CV. In essence, the classifier's ability to identify the percentage of test samples that belong to each of the desired mechanical property classes, *i.e.*, high true ultimate strength, high true yield strength, high engineering elastic modulus, high engineering ultimate strength, high flexural modulus, high flexural strength, high impact strength, high storage modulus, high loss modulus, and high tan delta was evaluated and analyzed.

In the SVMs analyses, confusion matrices (contingency tables) [45] were used to compare and analyze the resulting classifications. Three values of C were used in this study: (*i.e.* 0.5, 10, and 100), and the developed SVMs model was run using three kernel functions: a polynomial of degree two, a dot product, and a hyperbolic tangent kernels. As previously indicated in Section 3, the choice for the positive constant C determines the number of support vectors and the overall performance of the SVMs model [1].

The overall classification rates and apparent error rates (or false negative values) for the three different kernels using the resubstitution are shown in Tables 3–5, respectively where these rates are identically equal for each value of the constant C. In this case, the performance of the SVMs model was good and was able to correctly classify about 99% of the VGCNF/VE specimens into ten different distinct property classes when the dot product kernel was used regardless of the constant C (Table 4). Although a classification error of about 13% resulted when the dot product kernel was used for C3, this error was considered to be acceptable as it did not significantly affect the overall classification accuracy of the model. The polynomial kernel (degree 2) achieved a 93% average classification rate (Table 3). When the hyperbolic tangent kernel was implemented, the classification performance was degraded down to about 82% (Table 5). These high classification rates (in case of dot product and polynomial kernels) are due to the fact that all samples were used for training and testing in order to minimize the AE (apparent error) rate. When the 3-fold CV technique was used,

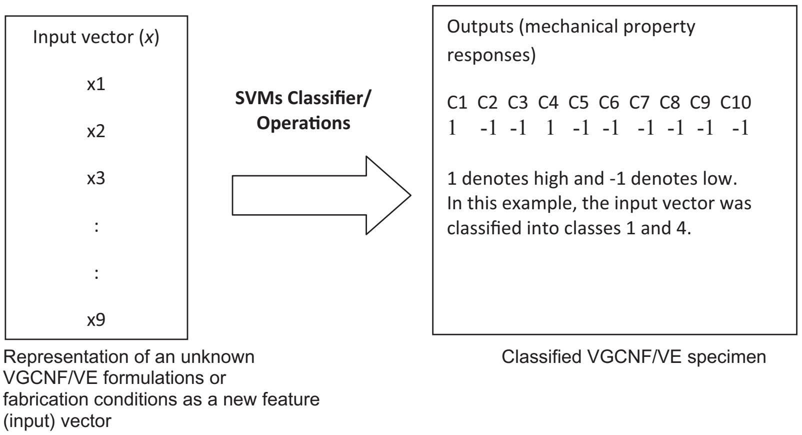
Input vector (x)
x1
x2
x3
:
:
x9
Representation of an unknown VGCNF/VE formulations or fabrication conditions as a new feature (input) vector
SVMs Classifier/ Operations
Outputs (mechanical property responses)
C1 C2 C3 C4 C5 C6 C7 C8 C9 C10
1 -1 -1 1 -1 -1 -1 -1 -1 -1
1 denotes high and -1 denotes low.
In this example, the input vector was classified into classes 1 and 4.
Classified VGCNF/VE specimen

Fig. 5. Representation of the SVMs model used in this study. The input vector x is processed through the SVMs classifier to create a mapping to the corresponding desired mechanical property response(s).

Table 3
Classification information of the SVMs model when a polynomial kernel of degree 2 was implemented using the resubstitution method.

Polynomial kernel (degree 2) and $C = 0.5$, 10, 1000	Resubstitution method										
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	Average
Correct classification rate	80%	93%	100%	100%	93%	97%	100%	87%	100%	80%	93%
Apparent error rate/false negative value	20%	7%	0%	0%	7%	3%	0%	13%	0%	20%	7%

Table 4
Classification information of the SVMs model when a dot product kernel was implemented using the resubstitution method.

Dot product kernel and $C = 0.5$, 10, 1000	Resubstitution method										
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	Average
Correct classification rate	100%	100%	87%	100%	100%	100%	100%	100%	100%	100%	99%
Apparent error rate/false Negative value	0%	0%	13%	0%	0%	0%	0%	0%	0%	0%	1%

Table 5
Classification information of the SVMs model when a hyperbolic tangent kernel was implemented using the resubstitution method.

Hyperbolic tangent kernel and $C = 0.5$, 10, 1000	Resubstitution method										
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	Average
Correct classification rate	63%	100%	70%	67%	80%	100%	94%	70%	77%	100%	82%
Apparent error rate/false negative value	37%	0%	30%	33%	20%	0%	6%	30%	23%	0%	18%

the selected sizes of training and testing sets for each of the three folds were 80 and 40 data samples, respectively.

Despite the fact that the 3-fold CV technique yields lower computational cost than the resubstitution method, the classification performance of the 3-fold CV technique was inferior to that of the resubstitution method as the AE rates were higher than those of the resubstitution method. This is due to the fact that the sizes of classes 1–10 for the three folds in all stages were significantly lower than such classes when the resubstitution method was applied. Furthermore, unlike the resubstitution method, the same samples were not used for training and testing in each fold, resulting in some additional misclassification error. Therefore, the AE rate of the resubstitution method is actually the error rate obtained from training data and this explains why the AE rate is low using different kernels. Consequently, resubstitution AE rate indicates only how good (or bad) are the results (using SVMs classifier in this case) on the TRAINING data and it expresses some knowledge about the algorithms used. In other words, it is used as a performance measure of the designed SVMs classifier as it reflects the imprecision of the training results; the lower the AER, the better precision the classifier has. 3-fold CV method however, is used to prevent the overlap of the test sets by first splitting data into x subsets of equal size and then using each subset in turn for testing and the remainder for training. Therefore, in CV method, The AE estimates are averaged to yield an overall error, called the predictive accuracy estimates, whereas in resubstitution method, the estimated AE is the performance measure of the designed classifier. This explains that fact that the AE rates using the CV method are higher than that of the resubstitution method.

For example, when the resubstitution method was implemented using the dot product kernel, the SVMs model was able to identify all samples (100%) that have the highest output mechanical properties, except for those that have the highest engineering elastic modulus (C3) (Table 4), where it was able to identify about 87% of samples. When the 3-fold CV was implemented, the SVMs model was able to identify 100% of test samples that have the highest tan delta value, and 0% of test samples that have the highest loss modulus. The identification of the test specimens with respect to the other mechanical property classes fell in between these two percentages (Table 7). Even though the model performed poorly in identifying the test samples that have the highest loss modulus in case of the 3-fold CV when dot product kernel was implemented, the model performed really well in identifying the samples that have the highest tan delta (100% classification rate) and it was able to identify about 48% of test samples that have the highest storage modulus. Given the fact that tan delta is the ratio of loss modulus to storage modulus, the samples that have the highest loss modulus can be determined by comparing the test samples that have the highest tan delta value and the test samples that have the highest storage modulus.

The overall classification rates and apparent error rates (or false negative values) for the three different kernels using the 3-fold CV techniques are shown in Tables 6–8. Based on these results, while 3-fold CV technique was able to correctly classify specimens into C1, C2, and C10, mixed results were obtained when classifying specimens into other classes. This behavior was observed to be strongly dependent on the kernel function used. For example, when a dot product kernel was implemented, the correct classification rate for C5 was observed to be 95% (Table 7). This value dropped to 52% when the hyperbolic tangent kernel was used (Table 8). Generally, the dot product kernel performed the best, yielding an average classification rate of about 71% (Table 7).

However, the resulting confusion matrices proved that the SVMs classifier performed well for fold 3 samples for dot product and degree 2 polynomial kernel functions at 99% and 95% classification rates, respectively. In addition, reasonable classification rates were achieved when the hyperbolic tangent kernel was implemented for fold 3 samples at 67% and for fold 2 samples at 61% and 59%, when the polynomial (degree 2) and dot product kernels were implemented, respectively. The classification rates were lower for other cases. In addition, the classification results were independent of the value of the constant C, similar to the resubstitution method analyses.

Another observation is that the SVMs model was able to more correctly classify specimens belonging to classes 1 and 10 than other classes when the 3-fold CV technique was implemented in case of polynomial and hyperbolic tangent kernels.

CV method has the advantage of producing an effectively unbiased error estimate, but the estimate is highly variable. However, in order to mitigate this, extensive experiments in literature

Table 6
Classification information of the SVMs model when a polynomial kernel of degree 2 was implemented using the 3-fold CV method.

Polynomial kernel (degree 2) and $C = 0.5$, 10, 1000	3-fold CV method										
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	Average
Correct classification rate	100%	95%	62%	67%	95%	93%	33%	33%	0%	100%	68%
Apparent error rate/false negative value	0%	5%	38%	33%	5%	7%	67%	67%	100%	0%	32%

Table 7
Classification information of the SVMs model when a dot product kernel was implemented using the 3-fold CV method.

Dot product kernel and $C = 0.5$, 10, 1000	3-fold CV method										
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	Average
Correct classification rate	90%	90%	83%	82%	95%	58%	63%	48%	0%	100%	71%
Apparent error rate/false negative value	10%	10%	17%	18%	5%	42%	37%	52%	100%	0%	29%

Table 8
Classification information of the SVMs model when a hyperbolic tangent was implemented using the 3-fold CV method.

Hyperbolic tangent kernel and $C = 0.5$, 10, 1000	3-fold CV method										
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	Average
Correct classification rate	75%	70%	67%	58%	52%	50%	18%	27%	0%	100%	52%
Apparent error rate/false negative value	25%	30%	33%	42%	48%	50%	82%	73%	100%	0%	48%

[46,47] have shown that the more folds used in the CV method, the better and more stable the predictive estimate of the designed SVMs classifier will be.

In addition, by choosing particular input level combinations, based on one of the industrial measures mentioned in Section 2, the SVMs model is able to best identify a desired mechanical property among the ten mechanical response classes. Section 5 elaborates more on the usefulness of the developed SVMs model to the VGCNF/VE manufacturing process and its mechanical performance characterization.

5. Industrial application to the SVMs model

The resulting SVMs model can be used to effectively anticipate the mechanical response for arbitrary input level combinations associated with the formulation and fabrication of VGCNF/VE nanocomposites. The presented methodology in this work can also be generalized to include other engineering material systems. This fast and reliable qualitative assessment of a material's response significantly reduces the need to perform additional extensive and expensive experiments. In addition, complex modeling procedures required to arrive at a suitable material that would meet the design and performance criteria can be significantly eliminated using the developed model. Furthermore, the cost estimation can directly be correlated with the material design and performance upfront, shortening the lead time for new material and component design. As an example, if a high impact strength is desired for the component made of the VGCNF/VE nanocomposite, the optimal high shear mixing time can be determined for a given cost-effective VGCNF/VE formulation, which is likely located at one of the untested (unknown) input level combinations. Since VGCNF/VE formulations are often expensive and their fabrication is time-consuming, a range of optimal VGCNF weight fractions and mixing times can be established over which desired properties are obtained by using the results of the SVMs model developed in this work. Once a suitable, cost-effective VGCNF/VE formulation with optimal fabrication time and performance is identified to give the desired mechanical response in one or more of the ten mechanical property classes presented in this work, quantitative tests can be performed to fully characterize it.

6. Summary and conclusions

A support vector machines (SVMs) technique was applied to a large vapor-grown carbon nanofiber (VGCNF)/vinyl ester (VE) nanocomposite dataset, consisting of 583 different design points: 172 compression, 93 tension, 60 flexure, 18 impact strength, and 240 viscoelastic data points. Each input level combination consisted of nineteen feature dimensions corresponding to nine input and ten output design factors. The nine input factors of the VGCNF/VE dataset were curing environment (air vs. nitrogen), use or absence of a dispersing agent, strain rate, mixing method (ultrasonication, high-shear mixing, and combination of both), VGCNF weight fraction, VGCNF type (pristine vs. oxidized), high-shear mixing time, sonication time, and temperature. The outputs (*i.e.*, measured properties) were true ultimate strength, true yield strength, engineering elastic modulus, engineering ultimate strength, flexural modulus, flexural strength, impact strength, storage modulus, loss modulus, and tan delta. Using the resubstitution and the 3-fold cross validation (CV) methods, the SVMs classifier was trained to classify each VGCNF/VE sample into one of ten optimal property classes that represent the high values for the above-mentioned outputs. The model was implemented in ten stages using a *one-against-all* strategy. A set of confusion matrices was used to compare the sets of analyses after exploring three kernels functions: a polynomial kernel of degree two, a dot product kernel, and a hyperbolic tangent kernel.

In general, the SVMs model using the resubstitution method was able to better predict the optimal property classes with a minimal apparent error (AE) rate using the dot product and degree two polynomial kernels than the 3-fold CV method. Nevertheless, an overall good classification result was obtained using the 3-fold CV method when the dot product kernel was implemented and also the model was able to accurately predict which data points belonged to the high true ultimate strength, high true yield strength, and high tan delta classes in case of the other kernel functions (*i.e.*, polynomial of degree two and hyperbolic tangent). However, although 3-fold CV method yields less computational cost, the SVMs model using this method had significant AEs for other mechanical property classes used in this study.

Most importantly, the developed SVMs model is able to identify the mechanical property response value resulting from a selected untested combination of the nine input factors mentioned in this study. The choice of the input level combinations is commensurate with particular optimal measure(s) considered by materials scientists and engineers. This includes, but is not limited to, the input level combinations that would yield the least time and cost to fabricate the specimens, while providing the highest mechanical properties (one or several of the mechanical property classes) of the VGCNF/VE nanocomposites. In other words, if a given input level combination is fed to the SVMs model, the unknown (*i.e.*, untested) output mechanical response(s) can be easily determined.

The model's ability to identify these desired mechanical property responses based on a particular combination of input factors will result in a faster and more reliable VGCNF/VE nanocomposites manufacturing lead time without the need to rely on extensive and time-consuming experiments or complex modeling and simulations. The SVMs classifier applied in this study demonstrates the usefulness of data mining and knowledge discovery techniques in materials science and engineering. It is expected that more such techniques will be employed within this rising field in near future.

References

[1] S. Theodoridis, K. Koutroumbas, Pattern Recognition, fourth ed., Academic Press, Massachusetts, 2008.

[2] R. Akbani, S. Kwek, N. Japkowicz, Applying support vector machines to imbalanced datasets, in: Mach. Learn: ECML, Springer, Berlin, Heidelberg, 2004, pp. 39–50.

[3] K. Rajan, Mater. Today. 8 (2005) 38–45.

[4] K.F. Ferris, L.M. Peurrung, J.M. Marder, Adv. Mater. Process. 165 (2007) 50–51.

[5] C. Suh, K. Rajan, B.M. Vogel, B. Narasimhan, S.K. Mallapragada, Comb. Mater. Sci. 5 (2006) 109–119.

[6] Q. Song, Chin. Sci. Bull. 49 (2004) 210–214.

[7] J. Rodgers, D. Cebon, MRS Bull. 31 (2006) 975–980, http://dx.doi.org/10.1557/mrs2006.223.

[8] O. Abuomar, S. Nouranian, R. King, Artificial neural network modeling of the viscoelastic properties of Vapor-Grown Carbon Nanofiber/Vinyl Ester nanocomposites, in: The 19th International Conference on Composite Materials (ICCM19), Montreal, Quebec, Canada, 2013, July 28–August 2.

[9] T.M. Nunes, V.H.C. De Albuquerque, J.P. Papa, C.C. Silva, P.G. Normando, E.P. Moura, J.M.R. Tavares, Expert Syst. Appl. 40 (8) (2013) 3096–3105.

[10] J.P. Papa, R.Y. Nakamura, V.H.C. De Albuquerqu, A.X. Falcao, J.M.R. Tavares, Expert Syst. Appl. 40 (2) (2013) 590–597.

[11] V.H.C. De Albuquerque, C.C. Silva, T.I.D.S. Menezes, J.P. Farias, J.M.R. Tavares, Microsc. Res. Tech. 74 (1) (2011) 36–46.

[12] V.H.C. De Albuquerque, A.R. De Alexandria, P.C. Cortez, J.M.R. Tavares, NDT&E Int. 42 (7) (2009) 644–651.

[13] V.H.C. De Albuquerque, P.C. Cortez, A.R. De Alexandria, J.M.R. Tavares, Nondestruct. Testing Eval. 23 (4) (2008) 273–283.

[14] V.H.C. De Albuquerque, J.M.R. Tavares, L.M. Durão, J. Compos. Mater. 44 (9) (2010) 1139–1159.

[15] K. Roberts, F. Muchlich, R. Schenkel, G. Weikum, An information system for material microstructures, in: International Conference on Scientific and Statistical Database Management (SSDBM'04), vol. 04, IEEE, 2004, pp. 1099–3371.

[16] R.M. Haralick, IEEE Trans. Syst., Man, Cyber. (1973) 610–621

[17] S. Swaddiwudhipong, K. Tho, Z.S. Liu, J. Hua, N.S.B. Ooi, Modell. Simul. Mater. Sci. Eng. 13 (2005) 993–1004.

[18] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. DeMoor, J. Vandewalle, Least Squares Support Vector Machines, World Scientific, 2002. ISBN 981-238-151-1.

[19] C. Hu, C. Ouyang, J. Wu, X. Zhang, C. Zhao, Data Sci. 8 (2009) 52–61.

[20] T. Sabin, C. Bailer-Jones, P. Withers, Modell. Simul. Mater. Sci. Eng. 8 (2000) 687–706.

[21] J.H. Koo, Polymer Nanocomposites: Processing, Characterization, and Applications, first ed., McGraw-Hill, New York, 2006.

[22] J. Garces, D.J. Moll, J. Bicerano, R. Fibiger, D.G. McLeod, Adv. Mater. 12 (2000) 1835–1839.

[23] F. Hussain, M. Hojjati, M. Okamoto, R.E. Gorga, J. Compos. Mater. 40 (2006) 1511–1575.

[24] E.T. Thostenson, C. Li, T.W. Chou, Compos. Sci. Technol. 65 (2005) 491–516.

[25] O. Abuomar, S. Nouranian, R. King, J.L. Bouvard, H. Toghiani, T.E. Lacy, C.U. Pittman Jr, Adv. Eng. Inform. 27 (2013) 615–624, http://dx.doi.org/10.1016/j.aei.2013.08.002.

[26] S. Nouranian, Vapor-grown Carbon Nanofiber/vinyl ester nanocomposites: designed Experimental Study of Mechanical Properties and Molecular Dynamics Simulations, Mississippi State University, PhD Dissertation, Mississippi State, MS USA, 2011.

[27] S. Nouranian, H. Toghiani, T.E. Lacy, C.U. Pittman, J. Dubien, J. Compos. Mater. 45 (2011) 1647–1657.

[28] S. Nouranian, T.E. Lacy, H. Toghiani, C.U. Pittman Jr, J.L. Dubien, J. Appl. Polym. Sci. (2013), http://dxdoi.org/10.1002/app.39041.

[29] G.W. Torres, S. Nouranian, T.E. Lacy, H. Toghiani, C.U. Pittman Jr, J. Dubien, J. Appl. Polym. Sci. 128 (2013) 1070–1080.

[30] R.L. King, A. Rosenberger, L. Kanda, Folia Primatol. 76 (2005) 303–324.

[31] T. Kohonen, Self-organization and Associative Memory, Springer-Verlag, 1988.

[32] S. Miyamoto, H. Ichihashi, K. Honda, Algorithms for Fuzzy Clustering: Methods in c-Means Clustering with Applications, Springer, 2008.

[33] J.C. Bezdek, R. Ehrlich, Comput. Geosci. 10 (1984) 191–203.

[34] I.T. Jolliffe, Principal Component Analysis, Springer, 2002.

[35] J. Lee, The Effect of Material and Processing on the Mechanical Response of Vapor-Grown Carbon Nanofiber/Vinyl Ester Nanocomposites, M.S. Thesis, Mississippi State University, Mississippi State, MS, 2010.

[36] J. Lee, S. Nouranian, G.W. Torres, T.E. Lacy, H. Toghiani, C.U. Pittman, J.L. DuBien, J. Appl. Polym. Sci. 130 (2013) 2087–2099, http://dx.doi.org/10.1002/app.39380.

[37] O. Abuomar, S. Nouranian, R. King, T.M. Ricks, T.E. Lacy, Mechanical property classification of vapor-grown carbon nanofiber/vinyl ester nanocomposites using support vector machines, in: The 10th International Conference on Data Mining, Las Vegas, USA, 2014 July 21–24.

[38] MATLAB Mathematics and Interpolation, Release 2012a, The MathWorks, Inc., Natick, Massachusetts, United States, 2012.

[39] B. Fornberg, J. Zuev, Comput. Math. Appl. 54 (2007) 379–398.

[40] Y. Tang, Y.Q. Zhang, N.V. Chawla, S. Krasser, IEEE Trans. Syst., Man., Cybern. 39 (1) (2009) 281–288.

[41] J. Twomey, A. Smith, IEEE Trans. Syst., Man., Cybern. 28 (1998) 417–430.

[42] J. Milgram, M. Cheriet, R. Sabourin, "One Against One" or "One Against All": which one is better for handwriting recognition with SVMs?", in: 10th International Workshop on Frontiers in Handwriting Recognition, 2006.

[43] S. Haykin, Neural Networks and Learning Machines, third ed., Prentice-Hall, 2009.

[44] S. Hayken, Neural Networks: A Comprehensive Foundation, second ed., Prentice-Hall, Englewood Cliffs (NJ), 1999.

[45] R. Kovari, F. Provost, J. Mach. Learn. 30 (1998) 271–274.

[46] R. Kohavi, IJCAI 14 (2) (1995) 1137–1145.

[47] U. Braga-Neto, R. Hashimoto, E.R. Dougherty, D.V. Nguyen, R.J. Carroll, Bioinformatics 20 (2) (2004) 253–258.