

Self-Advising SVM for Sleep Apnea Classification

Yashar Maali¹, Adel Al-Jumaily¹ and Leon Laks²

¹University of Technology, Sydney (UTS)
Faculty of Engineering and IT, Sydney, Australia

Yashar.Maali@student.uts.edu.au, Adel.Al-Jumaily@uts.edu.au

²Respiratory Physician, Concord Hospital
Sydney, Australia

leonlaks@bigpond.com

Abstract. In this paper Self-Advising SVM, a new proposed version of SVM, is investigated for sleep apnea classification. Self-Advising SVM tries to transfer more information from training phase to the test phase in compare to the traditional SVM. In this paper Sleep apnea events are classified to central, obstructive or mixed, by using just three signals, airflow, abdominal and thoracic movement, as inputs. Statistical tests show that self-advising SVM performs better than traditional SVM in sleep apnea classification.

Keywords: Sleep apnea, Support vector machines, Particle swarm optimization.

1 Introduction

Sleep disorders are common and sleep apnea (SA) is one of the most common and critical types of sleep disorders. SA can be recognized by the repeated temporary cessation of breathing during sleep [1]. More precisely, apnea is defined as the total or near-total absence of airflow. This becomes significant once the reduction of the breathing signal amplitude is at least around 75% with respect to the normal respiration and occurs for a period of 10 seconds or longer[2]. A sleep apnea event can also be classified into three groups as: central sleep apnea, obstructive sleep apnea, and mixed sleep apnea. In case of the first, sleep apnea is originated by the central nervous system. In the case of the second, the reason for the pauses in the breathing lie in a respiratory tract obstruction, while in the third case, both of these reasons may be present.

The manual scoring of sleep apnea is costly and time-consuming. Therefore, many efforts have been made to develop systems that score the records automatically [11-13]. For this reason several Artificial Intelligent (AI) algorithms are used in this area such as fuzzy rule-based system [14], genetic SVM [15], and PSO-SVM [16] which have been proposed in our previous works. Classification of apneic events to apnea or hypopnea is also so important for severity calculation of the sleep disorder. The classification of apneic events is also considered in many studies, such as [17-19].

In this study an improved version of SVM, named self-advising SVM, is used to classify sleep apnea to central, obstructive and mixed. The second section of this work

covers some preliminaries about SVM and partial swarm optimization. We introduce the self-advising SVM algorithm in the third section of this paper; fourth section covers proposed methodology for classifying sleep apnea by the self-advising SVM which is followed by experimental results in section five, and the conclusion in section six.

2 Preliminaries

2.1 Support vector machine

Support vector machine (SVM) is a machine learning method proposed by Vapnik in 1995 [20]. The idea of SVM is to construct a maximized separating hyperplane. The optimization criterion of SVM is the width of the margin between the classes, i.e., the empty area around the decision boundary defined by the distance to the nearest training patterns. SVM shows its ability for classification in many applications even with high dimension. In addition, SVMs avoid over fitting by choosing a specific hyperplane among the many that can separate the data in the feature space.

The brief math description can be shown as follows. For a binary classification, from a training set of N samples $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_i, y_i), \dots, (\mathbf{x}_N, y_N) \in \mathcal{R}^n \times \{\pm 1\}$, where \mathbf{x}_i is the input vector corresponding to the i th sample and labeled by y_i depending on its class. SVM aim is, separating the binary labeled training data with a hyperplane that has maximum distance from them, known as maximum margin hyperplane. Figure 1 shows the basic idea of the SVM graphically. The pair (\mathbf{w}, b) defines the hyperplane with equation $\langle \mathbf{w}, \mathbf{x} \rangle + b = 0$. So, this hyperplane can linearly separate the train data if

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, N. \quad (1)$$

Distance of each training data \mathbf{x}_i from the hyperplane is given by

$$d_i = \frac{\mathbf{w} \cdot \mathbf{x}_i + b}{\|\mathbf{w}\|}, \quad (2)$$

combining inequality (1) and (2), for all \mathbf{x}_i result in

$$y_i d_i \geq \frac{1}{\|\mathbf{w}\|}. \quad (3)$$

Therefore, $\frac{1}{\|\mathbf{w}\|}$ is the lower bound on the distance between the training data \mathbf{x}_i and the separating hyperplane.

The maximum margin hyperplane can be considered as the solution of the problem of maximizing the $\frac{1}{\|\mathbf{w}\|}$ subject to the constraint (1), or equivalently by solving the following problem

$$\begin{aligned} \text{Minimize} \quad & z = \frac{1}{2} \mathbf{w} \cdot \mathbf{w} \\ \text{s. t.} \quad & y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, N. \end{aligned} \quad (4)$$

If we denote $(\alpha_1, \alpha_2, \dots, \alpha_N)$ the N nonnegative Lagrange multipliers associated with the constraints (1), and without considering few steps the resulting decision function is given by [21],

$$f(\mathbf{x}) = \text{sign} \left(\sum_{\alpha_i > 0} y_i \alpha_i \langle \mathbf{x}, \mathbf{x}_i \rangle + b \right), \quad (5)$$

Note that the nonzero α_i are those for which the constraints (1) are satisfied with the equality sign. This has an important consequence. Since most of the α_i are usually zero the vector \mathbf{w} is a linear combination of a relatively small percentage of the training data \mathbf{x}_i . These points are termed *support vectors* because they are the closest points from the separating hyperplane and the only points needed to determine the hyperplane. The support vectors are the training patterns that lie on the margin boundaries. An advantage of SVM is this fact that only small subset of the training samples, support vectors, is finally retained for the classifier.

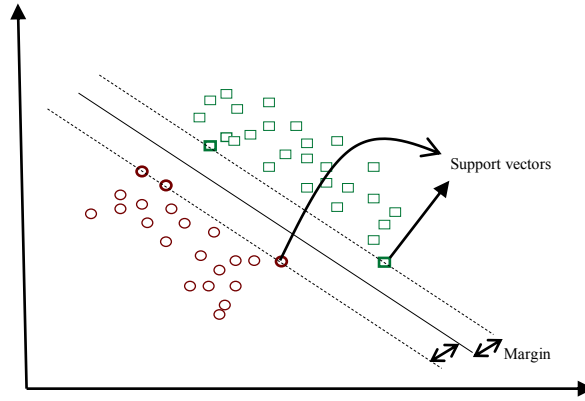


Fig.1. Basic ideas of support vector machines.

In order to use the SVM to produce nonlinear decision functions, the training data is projected to a higher-dimensional inner product space F , called feature space, using a nonlinear map $\phi(\mathbf{x}): \mathcal{R}^n \rightarrow \mathcal{R}^d$. In the feature space the optimal linear hyperplane is computed. Nevertheless, by using kernels it is possible to make all the necessary operations in the input space by using $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$ as $k(\mathbf{x}_i, \mathbf{x}_j)$ is an inner product in the feature space. The decision function can be written in terms of these kernels as follows:

$$f(\mathbf{x}) = \text{sign} \left(\sum_{\alpha_i > 0} y_i \alpha_i k(\mathbf{x}, \mathbf{x}_i) + b \right). \quad (6)$$

There are 3 common kernel functions in SVM:

Polynomial kernel : $K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + 1)^q$

RBF kernel : $K(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$

Sigmoid kernel : $K(x_i x_j) = \tanh(\gamma x_i^T x_j + c)$
Here q, γ, c are kernel parameters.

2.2 Particle Swarm Optimization

Particle Swarm Optimization (PSO), was introduced by Kennedy and Eberhart in 1995 [22, 23] based on the movement of swarms and inspired by the social behaviors of birds or fishes. Similar to the genetic algorithm, PSO is a population-based stochastic optimization technique. In the PSO, each member is named particle, and each particle is a potential solution to the problem. In comparison with the genetic algorithms, PSO updates the population of particles by considering their internal velocity and position, which are obtained by the experience of all the particles.

In this study, we use the constriction coefficient PSO [24]. In this approach, the velocity update equation is as (1),

$$v_{ij}(t+1) = \chi \left[v_{ij}(t) + \phi_1 (y_{ij}(t) - x_{ij}(t)) + \phi_2 (\hat{y}_{ij}(t) - x_{ij}(t)) \right], \quad (7)$$

where y_{ij} is the particle best and \hat{y}_{ij} is the global best particles. And,

$$\chi = \frac{2k}{|2 - \phi - \sqrt{\phi(\phi - 4)}|}, \quad (8)$$

with,

$$\phi = \phi_1 + \phi_2, \quad \phi_i = c_i r_i \quad i = 1, 2.$$

Equation (7) is used under the constraints that $\phi \geq 4$ and $k, r_i \in [0, 1]$.

The parameter k in the equation (8) controls the exploration and exploitation. For $k \sim 0$, fast convergence is expected and for $k \sim 1$ we can expect slow convergence with a high degree of exploration [24].

The constriction approach has several advantages over traditional PSO model such as; we do not need velocity clamping for constriction model and this model guarantees convergence under the given constraints[25].

3 Self-Advising SVM

In current SVM methods, the only information that is used in the test phase from the training is the hyperplane positions or SVs. Subsequent knowledge can be any more information about the SVs, such as their distribution, and or the knowledge extracted from the misclassified data in the training phase.

Self-advising SVM tries to generate subsequent knowledge from the misclassified data of the training phase of the SVM. This misclassified data can come from 2 potential sources as outliers or as data that have not been linearly separated by using any type of kernels. Classic SVM ignores the training data that has not been separated linearly by kernels in the training phase. Self-advising SVM intended to deal with the

ignoring of the knowledge that can be extracted from the misclassified data. This can be done by generating advice weights based on using of misclassified training data, if possible, and use these weights together with decision values of the SVM in the test phase. These weights help the algorithm to eliminate the outlier data.

To benefit from the misclassified data of the training phase, we must first find them. Let's define the misclassified data sets, MD, in the training phase as follows:

$$MD = \bigcup_{i=1}^n \mathbf{x}_i | y_i \neq \text{sign} \left(\sum_{\alpha_j > 0} y_j \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) + b \right) \quad (9)$$

It must be considered that on the right hand side of the equation (9), we can use any SVM decision function and kernel. The MD set can be null, but experimental results revealed that the occurrence of misclassified data in training phase is common.

For each \mathbf{x}_i of MD the neighborhood length (NL) is defined as:

$$NL(\mathbf{x}_i) = \text{minimum}_{\mathbf{x}_j} (|\mathbf{x}_i - \mathbf{x}_j| | y_i \neq y_j). \quad (10)$$

where $\mathbf{x}_j, j = 1, \dots, n$, are the training data.

Note: if the training data is mapped to a higher dimension by using a mapping function, then the distance between \mathbf{x}_i and \mathbf{x}_j can be computed according to the following equation with reference to the related kernel k ,

$$\|\boldsymbol{\theta}(\mathbf{x}_i) - \boldsymbol{\theta}(\mathbf{x}_j)\| = (k(\mathbf{x}_i, \mathbf{x}_i) + k(\mathbf{x}_j, \mathbf{x}_j) - 2k(\mathbf{x}_i, \mathbf{x}_j))^{0.5}. \quad (11)$$

Finally, based on finding of NL, for each \mathbf{x}_k from the test set, the advised weight (AW) is computed as follows,

$$\begin{cases} 0, & \forall \mathbf{x}_i \in MD, \|\mathbf{x}_k - \mathbf{x}_i\| > NL(\mathbf{x}_i) \text{ or } MD = NUL \\ 1 - \frac{\sum_{\mathbf{x}_i} \|\mathbf{x}_k - \mathbf{x}_i\|}{\sum_{\mathbf{x}_i} NL(\mathbf{x}_i)}, & \mathbf{x}_i \in MD, \|\mathbf{x}_k - \mathbf{x}_i\| \leq NL(\mathbf{x}_i) \end{cases} \quad (12)$$

These AWs are between 0 and 1, and they represent how close the test data are to the misclassified data. To conclude the above, the self-advising SVM (SA-SVM) is as follows:

Training phase:

- 1- Finding the hyperplane by solving problem of equation (6) or related problem, it means normal SVM training.
- 2- Find the MD set using equation (9).
- 3- If the MD is null, go to the testing phase else compute NL for each member of MD using equation (10).

Testing phase:

- 1- Compute the AW(\mathbf{x}_k) for each \mathbf{x}_k from the test set

- 2- Compute the absolute value of the SVM decision values for each x_k from the test set and scale the values to $[0,1]$.
- 3- For each x_k from the test set,
 - If $AW(x_k) < \text{decision value}(x_k)$ then $y_k = \text{sign}\left(\sum_{\alpha_j > 0} y_j \alpha_j k(x_k, x_j) + b\right)$, this means normal SVM labeling.
 - Else, $y_k = y_i$ ($\|x_k - x_i\| \leq NL(x_i)$ and $x_i \in MD$).

Note: If the testing and training data are mapped to a higher dimension, then $\|x_k - x_i\|$ in step 3 of the test phase should be computed by equation (11); further, as mentioned previously, any SVM methods and kernels can be used in this algorithm.

4 Approach and Method

In this section, we present the proposed algorithm for the classification of the sleep apnea events into central, obstructive or mixed. The proposed methodology is as follows:

- Feature generation: this stage generates several statistical features for each event from the wavelet packet coefficients.
- PSO-SVM classifier: In this stage PSO is used to select a best features subset interactively with the SA SVM. PSO also is used for tuning the parameters of the SA SVM. In the process, SA SVM is used as the fitness evaluator.
- Final classification: the selected pattern is used for classification of the unseen validation data in this stage. The accuracy of this step is assumed as the final performance of the algorithm.

The details of these steps are as follows:

4.1 Feature generation

Feature extraction plays an important role in recognition systems, in this paper features are generated from wavelet coefficients. In the first step, 3 levels "Haar" wavelet packet applied on input signals, airflow, abdominal and thoracic movements. Then several statistical measures are computed by attention to the coefficients related to each apnea events, and considered as features of that event. These features represent the inputs of proposed PSO-SVM algorithm in the next step. Full list of proposed features are included in Table I.

Table I: List of statistical features, x is coefficients of wavelet.

$\log(\text{mean}(x^2))$	$\text{kurtosis}(x^2)$	$\text{geomean}(x)$
$\text{std}(x^2)$	$\text{var}(x^2)$	$\text{mad}(x)$
$\text{skewness}(x^2)$	$\text{mean}(x)$	$\text{mean}(x^2)$
$\text{skewness}(x)$	$\text{kurtosis}(x)$	$\text{var}(x)$
$\text{geomean}(x^2)$	$\text{mad}(x^2)$	$\text{std}(x)$

4.2 Particle representation

In this study, each particle consists of two arrays; the length of the first array is equal to the number of features. Each cell can get a real number between 0 and 1 as importance of the relevant feature. Features, which their corresponding cells have values higher than 0.5, are selected for classification. The second array is related to the gamma and cost as parameters of the SVM, which can get a value between 2^{-5} to 2^5 .

5 Results and discussion

Experimental data consist of 20 samples which events of them are annotated by an expert were provided by the Concord hospital in Sydney. We run the algorithm 5 different times; in each run 10 samples are chosen as the training, 5 samples as validation and 5 samples as the test set. RBF kernel is selected for the both of the Self-advising and traditional SVM. In the constriction coefficient PSO structure, k considered as 0.8 and $c_1 = 2$, $c_2 = 4$ and swarm contain 20 particles

Table 2 tabulates the number of central, obstructive and mixed events in each of the validation set, train and test for these 5 runs.

Accuracy and also f-score of the self-advising and traditional SVM in classification of these apnea events are as Table 3.

Table 2. Number of obstructive, central and mixed apnea in 5 different runs

	Obstructive	central	Mixed
#1	931	375	312
#2	879	463	276
#3	913	478	227
#4	870	494	254
#5	894	453	271

Table 3. Accuracies and f-score of self-advising and traditional SVM in classification of apnea events

	SVM		Self-Advising SVM	
	Accuracy	f-score	Accuracy	f-score
#1	85.02	0.79	87.32	0.84
#2	86.31	0.83	86.81	0.84
#3	78.44	0.74	83.28	0.81
#4	77.45	0.79	79.95	0.80
#5	76.44	0.77	82.29	0.84
Total	80.732	0.784	83.93	0.826

The average accuracies for these two methods are as 80.73 and 83.93, respectively. Also, for more reliable evaluation between results of these two methods pair t-test is used. The p value of t-test is as 0.028. These statistical tests show that the results obtained by the self-advising SVM are significantly better than the results of traditional SVM.

Also, we consider the f-score as another performance measure to compare these methods. Table 3, tabulated the f-scores for these two methods. The average f-score for traditional SVM and self-advisable SVM are 0.78 and 0.83, respectively. Also paired t-test shows that self-advisable is significantly better than traditional SVM by considering the f-score. The p value of t-test is as for the f-score is as 0.036.

6 Conclusion

In this study we proposed a new version of SVM named self-advising SVM for classification of sleep apnea events into obstructive, central or mixed. This study shows that self-advising SVM has advantage over traditional SVM in apnea classification problem. More investigation of the proposed SVM algorithm in apnea detection or other classification problems must be study in future works.

References

1. Guilleminault, C., J. van den Hoed, and M. Mitler, *Overview of the sleep apnea syndromes*. In: C. Guilleminault and Wc Dement, Editors, *Sleep apnea syndromes*, Alan R Liss, New York. 1978: p. 1-12.
2. Flemons, W.W.; Buysse, D; Redline, S, *Sleep-related breathing disorders in adults: Recommendations for syndrome definition and measurement techniques in clinical research*. *Sleep*, 1999. 22(5): p. 667-689.
3. Chokroverty, S., *Sleep deprivation and sleepiness*, in *Sleep Disorders Medicine (Third Edition)2009*, W.B. Saunders: Philadelphia. p. 22-28.
4. Chokroverty, S., *Overview of sleep & sleep disorders*. *Indian Journal of Medical Research*, 2010. 131(2): p. 126-140.
5. Ball, E.M.; Simon RD Jr; Tall AA; Banks MB; Nino-Murcia G; Dement WC, *Diagnosis and treatment of sleep apnea within the community - The Walla Walla project*. *Archives of Internal Medicine*, 1997. 157(4): p. 419-424.
6. Kryger, M.H.; Roos L, Delaive K, Walld R, Horrocks J, *Utilization of health care services in patients with severe obstructive sleep apnea*. *Sleep*, 1996. 19(9): p. S111-S116.
7. Stradling, J.R. and J.H. Crosby, *Relation between systemic hypertension and sleep hypoxemia or snoring- analysis in 748 men drawn from general-practice*. *British Medical Journal*, 1990. 300(6717): p. 75-78.
8. Hoffstein, V., *Snoring*, in *Principles and practice of sleep medicine*, M.H. Kryger, T. Roth, and D.W. C., Editors. 2000, Saunders: Philadelphia, PA: W.B. p. 813-826.
9. Penzel, T.; McNames J; Murray A; de Chazal P; Moody G; Raymond B., *Systematic comparison of different algorithms for apnoea detection based on electrocardiogram recordings*. *Medical & Biological Engineering & Computing*, 2002. 40(4): p. 402-407.
10. Kryger, M.H., *management of obstructive sleep-apnea*. *Clinics in Chest Medicine*, 1992. 13(3): p. 481-492.
11. Cabrero-Canosa, M., E. Hernandez-Pereira, and V. Moret-Bonillo, *Intelligent diagnosis of sleep apnea syndrome*. *Ieee Engineering in Medicine and Biology Magazine*, 2004. 23(2): p. 72-81.
12. Cabrero-Canosa, M; Castro-Pereiro, M; Graña-Ramos, M; Hernandez-Pereira, E; Moret-Bonillo, V; Martin-Egaña, M; Vereas-Hernando, H, *An intelligent system for the detection and interpretation of sleep apneas*. *Expert Systems with Applications*, 2003. 24(4): p. 335-349.
13. de Chazal, P.; Heneghan C; Sheridan E; Reilly R; Nolan P; O'Malley M, *Automated processing of the single-lead electrocardiogram for the detection of obstructive sleep apnoea*. *Ieee Transactions on Biomedical Engineering*, 2003. 50(6): p. 686-696.
14. Maali, Y. and A. Al-Jumaily, *Genetic Fuzzy Approach for detecting Sleep Apnea/Hypopnea Syndrome*. 2011 3rd International Conference on Machine Learning and Computing (ICMLC 2011), 2011.

15. Maali, Y. and A. Al-Jumaily. Automated detecting sleep apnea syndrome: A novel system based on genetic SVM. in *Hybrid Intelligent Systems (HIS), 2011 11th International Conference on*. 2011.
16. Maali Y. and A. Al-Jumaily, A Novel Partially Connected Cooperative Parallel PSO-SVM Algorithm Study Based on Sleep Apnea Detection, in *Accepted in IEEE Congress on Evolutionary Computation 2012: Brisbane, Australia*.
17. Schluter, T. and S. Conrad, An approach for automatic sleep stage scoring and apnea-hypopnea detection. *Frontiers of Computer Science*, 2012. 6(2): p. 230-241.
18. Aksahin, M; Aydin, S.; Firat, H.; Erogul, O, Artificial Apnea Classification with Quantitative Sleep EEG Synchronization. *Journal of Medical Systems*, 2012. 36(1): p. 139-144.
19. Guijarro-Berdinas, B., E. Hernandez-Pereira, and D. Peteiro-Barral, A mixture of experts for classifying sleep apneas. *Expert Systems with Applications*, 2012. 39(8): p. 7084-7092.
20. Vapnik, V., *Statistical Learning* 1998: Wiley.
21. Lauer, F. and G. Bloch, Incorporating prior knowledge in support vector machines for classification: A review. *Neurocomputing*, 2008. 71(7-9): p. 1578-1594.
22. Eberhart, R. and J. Kennedy. A new optimizer using particle swarm theory. in *Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium on*. 1995.
23. Kennedy, J., R. Eberhart, and Ieee, Particle swarm optimization. 1995 Ieee International Conference on Neural Networks Proceedings, Vols 1-6 1995. 1942-1948.
24. Clerc, M. The swarm and the queen: towards a deterministic and adaptive particle swarm optimization. in *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*. 1999.
25. Engelbrecht, A.P., *Fundamentals of Computational Swarm Intelligence* 2005: Wiley.