2008

SUPPORT VECTOR MACHINE FOR HIGH THROUGHPUT RODENT SLEEP BEHAVIOR CLASSIFICATION

Shantilal
University of Kentucky, shan_grad@yahoo.com

Click here to let us know how access to this document benefits you.

Recommended Citation
Shantilal, "SUPPORT VECTOR MACHINE FOR HIGH THROUGHPUT RODENT SLEEP BEHAVIOR CLASSIFICATION" (2008). University of Kentucky Master's Theses. 506.
https://uknowledge.uky.edu/gradschool_theses/506
SUPPORT VECTOR MACHINE FOR HIGH –THROUGHPUT RODENT SLEEP BEHAVIOR CLASSIFICATION

This thesis examines the application of a Support Vector Machine (SVM) classifier to automatically detect sleep and quiet wake (rest) behavior in mice from pressure signals on their cage floor. Previous work employed Neural Networks (NN) and Linear Discriminant Analysis (LDA) to successfully detect sleep and wake behaviors in mice. Although the LDA was successful in distinguishing between the sleep and wake behaviors, it has several limitations, which include the need to select a threshold and difficulty separating additional behaviors with subtle differences, such as sleep and rest. The SVM has advantages in that it offers greater degrees of freedom than the LDA for working with complex data sets. In addition, the SVM has direct methods to limit overfitting for the training sets (unlike the NN method). This thesis develops an SVM classifier to characterize the linearly non separable sleep and rest behaviors using a variety of features extracted from the power spectrum, autocorrelation function, and generalized spectrum (autocorrelation of complex spectrum). A genetic algorithm (GA) optimizes the SVM parameters and determines a combination of 5 best features. Experimental results from over 9 hours of data scored by human observation indicate 75% classification accuracy for SVM compared to 68% accuracy for LDA.

KEYWORDS: Rodent Sleep Behavior Characterization, Piezoelectric Sensors, Pattern Classification, Linear Classifiers, SVM.

Shantilal

December 14, 2007
SUPPORT VECTOR MACHINE FOR HIGH–THROUGHPUT RODENT SLEEP BEHAVIOR CLASSIFICATION

By

Shantilal

Dr. Kevin D. Donohue
Director of Thesis

Dr. Bruce F. O’Hara
Co-Director of Thesis

Dr. Yu Ming Zhang
Director of Graduate Studies

December 14, 2007
RULES FOR THE USE OF THESIS

Unpublished theses submitted for the Masters degree and deposited in the University of Kentucky Library are as a rule open for inspection, but are to be used only with due regard to the rights of the authors. Bibliographical references may be noted, but quotations or summaries of parts may be published only with the permission of the author, and with the usual scholarly acknowledgments.

Extensive copying or publication of the thesis in whole or in part also requires the consent of the Dean of the graduate School of the University of Kentucky.

A library that borrows this dissertation for use by its patrons is expected to secure the signature of each user.

Name                                      Date
________________________________________
________________________________________
________________________________________
________________________________________
________________________________________
________________________________________
________________________________________
________________________________________
SUPPORT VECTOR MACHINE FOR HIGH-THROUGHPUT RODENT SLEEP BEHAVIOR CLASSIFICATION

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in the College of Engineering at the University of Kentucky

By
Shantilal
Lexington, Kentucky

Director: Dr. Kevin D. Donohue, Professor of Electrical and Computer Engineering
Co-Director: Dr. Bruce F. O'Hara, Associate Professor of Biology

Lexington, Kentucky
2007

Copyright © Shantilal 2007
DEDICATION

To my parents, siblings, and Sangita
ACKNOWLEDGEMENTS

I am grateful to my advisor, Dr. Kevin D. Donohue, for his guidance and support throughout my thesis. Dr. Donohue inspired me with his work ethic and commitment to excellence. I had several discussions with him which helped me to sharpen my technical skills and gain good insight into the topic. Thank you Sir, it was privilege working with you. Next, I would like to thank my co-director, Dr. Bruce F. O’Hara, for his help with the experiment and financial assistance. I would also like to thank Dr. Yu Ming Zhang for agreeing to take part in my thesis committee.

Lastly, I would like to thank my lab mates Aaron Daniels, David Williams, Anand Ramamurthy, Arul Kumaran, and all my friends for their help and support at various stages of my project.
# Table of Contents

**Acknowledgements** .............................................................................................................. III

**Table of Contents** .................................................................................................................. IV

**List of Tables** .......................................................................................................................... VI

**List of Figures** .......................................................................................................................... VII

**List of Files** .............................................................................................................................. VIII

1. **Introduction** .......................................................................................................................... 1
   1.1 Motivation: .......................................................................................................................... 1
   1.2 Background: ....................................................................................................................... 2
   1.3 Hypothesis: .......................................................................................................................... 4
   1.4 Organization of the Thesis: ............................................................................................... 5

2. **Support Vector Machine, LDA and MD** ............................................................................. 6
   2.1 SVM and Linearly Separable Datasets: ................................................................. 6
   2.2 Soft Margin Hyperplane: ......................................................................................... 12
   2.3 Non Linear SVM: ........................................................................................................... 15
       2.4.1 The LDA Method: ......................................................................................... 20
       2.4.2 The MD Method: ......................................................................................... 21

3. **Data Acquisition, Processing and Feature Extraction** ...................................................... 22
   3.1 The PVDF Sensor: ........................................................................................................... 22
   3.2 Signal Amplification: ....................................................................................................... 23
LIST OF TABLES

Table 3.1 Different Features and their description.............................................. 40
Table 4.1 Performance of the GA for 25 generations ........................................ 46
Table 4.2 Indices for Best Feature Combinations and performance for LDA and MD...................................................................................................................... 48
Table 5.1 Comparison of SVM performance w.r.t. other classifiers for Sleep vs. Rest behaviors.................................................................................................... 57
Table 5.2 Comparison of SVM performance w.r.t other classifiers for Sleep-rest vs. Wake behaviors. ........................................................................................... 61
LIST OF FIGURES

Figure 2.1 Maximal margin classifier ............................................................... 7
Figure 2.2 Linear SVM for Noisy Data ............................................................ 13
Figure 2.3 Scatter Plot for the Sleep and Rest Behaviors .............................. 16
Figure 2.4 An Example for Feature Space Enlargement. ............................... 17
Figure 3.1 Electrical equivalent of the sensor ............................................... 22
Figure 3.2 Signal Amplifier ........................................................................... 24
Figure 3.3 Signal Amplifier Frequency Response ......................................... 25
Figure 3.4 A six seconds sleep segment ....................................................... 30
Figure 3.5 A six seconds rest segment ........................................................... 30
Figure 3.6 Comparison of PS for sleep and rest behaviors ......................... 33
Figure 3.7 Comparison of AC for sleep and rest behaviors .......................... 34
Figure 3.8 CA plot for sleep and rest behaviors .......................................... 36
Figure 3.9 PS plot for 50 sleep and rest behaviors ....................................... 37
Figure 3.10 AC plot for 50 sleep and rest behaviors ..................................... 38
Figure 3.11 CA plot for 50 sleep and rest behaviors ..................................... 39
Figure 4.1 Crossover Operation ................................................................. 43
Figure 4.2 GA to simultaneously optimize feature subset & SVM parameters 44
Figure 4.3 GA performance for different generations ................................. 47
Figure 5.1 Experimental setup for data acquisition ..................................... 50
Figure 5.2 SVM Classification for the Sleep and Rest Behaviors ................. 54
Figure 5.3 Signal Transform comparison of sleep-rest (blue) and wake signals for (a) PS, (b) AC, and (c) CA. ................................................................. 58
Figure 5.4 SVM Classification for the Sleep-Rest and Wake Behaviors ......... 60
LIST OF FILES

ETD_Shan_Thesis.pdf (1.13 MB)
1. INTRODUCTION

1.1 Motivation:

Humans spend a third of their lives sleeping yet the basic functions of sleep are poorly understood. Around 50-70 million Americans and many more across the world are afflicted by sleep disturbances, and a major cause of industrial accidents is sleepiness [1, 46]. If the genes that are responsible for both normal and unusual sleep behaviors are identified, then many fundamental questions regarding sleep function and sleep regulation can be better understood. Such knowledge may assist in the development of better drug and non-drug treatments to help people with sleep disorders, or others who must operate at peak performance with insufficient sleep. This group includes shift workers, military personnel who must operate at nights, pilots, and other trans-meridian travelers who may suffer jet lag.

It is found that a vast majority of human genes have essentially identical homologs in mouse [2] and the mouse brain shares many functions with human brain including sleep behaviors [3]. Therefore, if mouse genes responsible for sleep behavior are determined, these are almost certainly the same genes important for human sleep. The commonly used methods for sleep studies in mice and humans are Electroencephalographic (EEG) and Electromyographic (EMG) recordings [4]. Although these methods accurately distinguish between sleep and wake in mice, they require electrodes to be implanted into the skull and muscle of the mice. This invasive procedure requires significant effort, and recovery time before any analysis can be commenced. In addition, the waveforms must be stored in a semi automatic fashion, requiring additional human effort. Consequently, these methods cannot be utilized for high throughput genetic studies in mice as they are both expensive and time consuming.
This thesis considers a non-invasive, high-throughput system coupled with an advanced classification scheme to automatically classify sleep and rest behaviors in mice.

1.2 Background:

Due to the disadvantages associated with the invasive method for analyzing sleep in rodents, other methods have been explored for non or minimum invasive methods with automatic classification. This Section reviews previous attempts to develop automatic and high throughput systems for rodent sleep studies and other behaviors. These methods are either based on PVDF sensors or require digital video analysis.

Megens, et al. [8], utilized a combination of an optical scanning technique and piezoelectric principle to monitor motor activity in rodents treated with different drug dosages. Here, the piezoelectric signals were mostly due to the locomotion of rodents on top of two PVDF sensors that were placed side by side. Respiratory movements or heart-beats were not considered for analysis and were filtered along with noise. In this thesis, however, respiratory movements, along with locomotion activities, are considered as they provide vital information to distinguish between sleep and rest behaviors in mice.

The first method to analyze sleep in rodents using PVDF sensor was presented in [9] where the pattern recognition system was up to 95% accurate in classifying sleep and wake piezoelectric signals. However, because of certain limitations, it was not suitable for large scale genetic studies. The major limitations are as follows:

1. Most of the features that were extracted for classification were computationally expensive as a result of which the classification process was time consuming.
2. Some of the features that were employed for classification were amplitude dependent. Although these features provide good information to the classifier to distinguish sleep and wake behaviors, the classifier might perform badly due to differences in mouse weight and contact with the PVDF sensor.

3. Neural networks were employed for classification between sleep and wake behaviors. Neural networks don’t provide direct methods to reduce overfitting and training neural networks is computationally expensive.

[5] addressed the limitations in [9] by:
1. Extracting features from three simple transforms, namely, autocorrelation, power spectrum, and generalized spectrum.
2. Making the features scale independent by normalizing the piezoelectric signals and thus making the classifier robust.
3. Employing a simple linear classifier, based on LDA approach, to distinguish between sleep and wake behaviors in mice.

Although the linear classifier was about 94% accurate in distinguishing between sleep and wake behaviors in mice, it required determination of a threshold and the selection of the threshold vastly influenced the performance of the classifier. Also, because of the linear nature of decision boundary, it was not able to learn complex patterns and consequently performed badly in distinguishing the linearly non separable sleep and rest behaviors.

The non-invasive methods described in [5], [8], and [9] were based on PVDF sensors. Another novel non-invasive way of assessing sleep and wake behaviors in mice is by digital video analysis or by evaluating breaking of infra red beams by mouse movements [10]. The method employed in [10] was based on the fact that an inactivity of $\geq 40$ seconds was predicted as sleep. Although this method achieved an average accuracy of 92% in classifying sleep and wake, its performance on the more difficult sleep and rest was unknown. The other limitations associated with this method were as follows:
1. Duration of inactivity that predicts sleep may vary across different strains of mice.
2. Short sleep behavior (<40 seconds) cannot be assessed.
3. During sleep bouts (≥ 40 seconds), the mice may perform intermittent stirrings which can be ignored and considered as part of sleep. But this method considers those stirrings as a break in inactivity and considers the duration after those stirrings as a different sleep bout.

1.3 Hypothesis:

This thesis utilizes the non-invasive method that was employed for high throughput monitoring of sleep and wake behaviors in mice [5] and extends it to successfully classify the linearly non-separable sleep and rest (quiet wake) behaviors. Both the behaviors were very similar in characteristics. The relationship between these behaviors is described in [13]. During the sleep behavior, the mouse was still with its eyes closed (or half open) or had its head tucked under its body. During the rest behavior, the mouse was still with its eyes open and with little to no head motion. In both cases, the mouse was lying on the sensor and the signal that was sensed by the PVDF sensor was mostly due to breathing of the mouse. Hence breathing signal was the primary signal that was utilized for classification.

Initial analyses in this thesis showed that linear classifiers like Linear Discriminant Analysis (LDA) and Mahalanobis Distance (MD) performed poorly to distinguish between the sleep and rest behaviors. An advanced classification scheme known as SVM was employed for classifying the sleep and rest behaviors and it performed the classification with reasonable accuracy.

For all three classifiers, the features were extracted from autocorrelation, power spectrum, and generalized spectrum. For the linear classifiers, an optimal subset of 5 features was obtained through a combination of bootstrapping approach and
Monte Carlo simulation. A genetic algorithm was employed to determine an optimal subset of 5 features and optimal SVM parameters for the SVM method of classification.

1.4 Organization of the Thesis:

The organization of the thesis is as follows. Chapter 2 describes the SVM classifier for linear and non-linear datasets, and presents a brief overview of the linear classifiers, LDA and MD. Chapter 3 provides details about the PVDF sensor, the amplifier, data processing and feature extraction. Chapter 4 presents the feature selection methods that were employed for SVM and the linear classifiers. Chapter 5 describes the experiment used to assess the performance and presents classification rates for SVM, LDA, and MD. It also summarizes the results and explains the advantages and limitations of the SVM classifier with regards to the LDA and MD. Lastly, Chapter 6 presents the conclusion and future work.
2. Support Vector Machine, LDA and MD

This chapter provides an overview of SVM theory. It provides details about how SVMs can be utilized to classify datasets which are easily (linearly) separable and how SVMs can be extended to classify datasets that cannot be separated by a linear boundary. The chapter also presents a brief overview of the linear classifiers, LDA and MD.

2.1 SVM and Linearly Separable Datasets:

SVM, developed by Vapnik and his colleagues, is a learning strategy for solving binary classification problems and is widely used in text categorization, image classification, bioinformatics, hand-written character recognition, and face detection [23, 24, 25, 26, and 27]. The simplest SVM model is called as maximal margin classifier and is employed for datasets that can be separated in a linear manner. Most of the times, datasets are not linearly separable (as in this thesis) but still it is important to have an understanding of the maximal margin classifier in order to have good insight of non linear SVM.

Given two classes, the overall goal in training an SVM is to find a hyperplane that is at a maximum distance from the training vectors of both the classes. In general, there might be many hyperplanes that linearly separate both the classes, but the optimal hyperplane would be one which maximizes the separation between the two classes or one that results in maximum margin. Figure 2.1 illustrates the maximal margin classifier for two classes (diamonds and circles) which are easily separable.
In the Figure 2.1, the two classes considered are easily separable. The optimal hyperplane is represented by the solid line and is perpendicular to the shortest line that joins the convex hulls (dotted lines), also known as margins, of both the classes [24]. Also, it can be noted that the optimal hyperplane intersects the shortest line in a manner such that it is equidistant from both the classes.

Denote the two classes (diamonds and circles) as [26]:

\[ \{ x_i, y_i \}, \quad i = 1, \ldots, l, \quad x_i \in \mathbb{R}^n, \quad y_i \in \{-1, 1\} \]  

(2.1)
In the above equation, vectors corresponding to diamonds are labeled as +1 and the vectors corresponding to circles are labeled as -1, \( l \) indicates the total number of vectors, and \( n \) indicates the dimensionality of each vector.

Let there be a hyperplane which separates the diamonds and circles. The vectors that lie on the hyperplane satisfy [26]:

\[
\mathbf{w} \cdot \mathbf{x} + b = 0
\]  
(2.2)

where \( \mathbf{w} \) is a vector normal to the hyperplane, as shown in Figure 2.1, \( ||\mathbf{w}|| \) is its Euclidean norm, and \( |b|/||\mathbf{w}|| \) is the perpendicular distance from the separating hyperplane to the origin.

The goal in training an SVM is to find the optimal hyperplane and this can be formulated such that all the input (training) vectors satisfy the following constraints [26]:

\[
\mathbf{x}_i \cdot \mathbf{w} + b \geq +1 \quad \text{for} \quad y_i = +1
\]  
(2.3)

\[
\mathbf{x}_i \cdot \mathbf{w} + b \leq -1 \quad \text{for} \quad y_i = -1
\]  
(2.4)

Eqns. 2.3 and 2.4 can be combined to obtain:

\[
y_i (\mathbf{x}_i \cdot \mathbf{w} + b) - 1 \geq 0 \quad \forall i \quad (2.5)
\]

If \( \mathbf{x}_1 \) is a training vector that lies on the hyperplane represented by Eqn. 2.3, then

\[
\mathbf{x}_1 \cdot \mathbf{w} + b = +1 \quad (2.6)
\]

Similarly, if \( \mathbf{x}_2 \) is a training vector that lies on the hyperplane represented by Eqn. 2.4, then

\[
\mathbf{x}_2 \cdot \mathbf{w} + b = -1 \quad (2.7)
\]
From Eqns. 2.6 and 2.7, we can write

\[
\left( \frac{\mathbf{w}}{||w||} \right) \cdot (\mathbf{x}_1 - \mathbf{x}_2) = \frac{2}{||w||} \quad (2.8)
\]

In the Eqn. 2.8, the term \(2/||w||\) represents the width of the margin, maximizing this is same as minimizing \(||w||/2\), so the optimization problem can be stated as:

\[
\text{Minimize} \quad 0.5 ||\mathbf{w}||^2 \quad (2.9)
\]

such that \(y_i (\mathbf{x}_i \cdot \mathbf{w} + b) \geq 1 \quad \forall_i \quad (2.10)

When this optimization problem is converted to Lagrangian form it offers two advantages [26]. The first advantage is that the constraints in the optimization problem will be converted to constraints on the Lagrangian multipliers, which can be easily taken care off. The other advantage is that when the Lagrangian form is converted to its dual, input vectors will only appear as dot products between vectors. The dual form will convert the optimization problem in a form such that it would allow kernel trick to be used in case of non linear SVM, which would be explained in Section 2.3.

The primal form is obtained by multiplying the constraint equations of Eqn. 2.10 with positive Lagrange multipliers and subtracting the resultant equations from the objective function. The primal form is expressed as [26]:

\[
L_p \equiv 0.5 \ || \mathbf{w} \ ||^2 - \sum_{i=1}^{l} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i \quad (2.11)
\]

In Eqn. 2.11, \(\alpha_i \geq 0\) are Lagrange multipliers (LMs). The primal form can be converted to dual by first differentiating Eqn. 2.11 with respect to the original variables \(\mathbf{w}\) and \(b\), and imposing stationarity to obtain the following equations.
\[
\frac{\partial L_p}{\partial w} = w - \sum_{i=1}^{l} y_i \alpha_i x_i = 0 \quad (2.12)
\]
\[
\frac{\partial L_p}{\partial b} = \sum_{i=1}^{l} y_i \alpha_i = 0 \quad (2.13)
\]

On solving Eqns. 2.12 and 2.13, the relations thus obtained are expressed as:
\[
w = \sum_{i=1}^{l} y_i \alpha_i x_i = 0 \quad (2.14)
\]
\[
\sum_{i=1}^{l} y_i \alpha_i = 0 \quad (2.15)
\]

These relations are substituted in the primal form (Eqn. 2.11), to obtain dual form that is devoid of the original variables and has to be maximized. The dual form is expressed as:
\[
\text{Maximize} \quad L_D = \sum_{i=1}^{l} \alpha_i - 0.5 \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i \cdot x_j \quad (2.16)
\]
\[
\text{such that} \quad \sum_{i=1}^{l} \alpha_i y_i = 0, \quad \alpha_i \geq 0 \quad (2.17)
\]

If \(\alpha_i^*, i=1,\ldots,l\), are the LMs (the asterisk indicates that the LM obtained is optimal) that are obtained on solving the dual form then the weight vector for the optimal hyperplane is
\[
w^* = \sum_{i=1}^{l} y_i \alpha_i^* x_i = 0 \quad (2.18)
\]

The value of \(b^*\) can be found by utilizing constraints expressed in Eqn. 2.10.
The values of \(w^*\) and \(b^*\) thus obtained should satisfy Karush-Kuhn-Tucker (KKT) conditions in order to ensure that they are optimal [23, 26, 27]. On applying the KKT conditions to the primal problem the following relations are obtained.
\[
\frac{\partial L_p(w^*, b^*, \alpha^*)}{\partial w} = 0
\]  \hspace{1cm} (2.19)

\[
\frac{\partial L_p(w^*, b^*, \alpha^*)}{\partial b} = 0
\]  \hspace{1cm} (2.20)

\[
\alpha_i^* (y_i (x_i \cdot w^* + b^*) - 1) = 0 \quad \forall_i
\]  \hspace{1cm} (2.21)

Eqn. 2.21 is called as the complimentarity condition. The LMs in this relation provide an idea about the dependence of the objective function on the constraints. In this relation, if \( \alpha_i^* = 0 \), the constraint is inactive and if \( \alpha_i^* \) is non-zero the constraint is active. For the latter case it implies that \( \alpha_i^* \) with non-zero values correspond to input vectors \( x_i \) that satisfy the relation

\[
y_i (x_i \cdot w^* + b^*) = 1
\]  \hspace{1cm} (2.22)

This means that only input vectors with non-zero LMs have a role in the determination of the optimal hyperplane. These vectors lie near the optimal hyperplane and are called as “support vectors”. It can also be noted that the value for \( b^* \) can be explicitly determined by using Eqn. 2.22.

The optimal hyperplane can now be expressed as follows [23]:

\[
f(x, \alpha^*, b^*) = \sum_{i \in \text{sv}} y_i \alpha_i^* \langle x_i, x \rangle + b^*
\]  \hspace{1cm} (2.23)

where \( \text{sv} \) stands for support vectors and \( \langle, \rangle \) indicates inner product between the different training vectors

The determination of the optimal hyperplane completes the training part of the SVM for the linearly separable datasets.
For testing, the decision function can be expressed as:

\[
f(x_i) = \text{sgn} \left( \sum_{i \in SV} y_i \alpha_i^* \langle x_i, x_i \rangle + b^* \right)
\]  

(2.24)

where \( x_i \) is the test vector. Eqn. 2.24 would result in +1 if the test vector belongs to the class of diamonds and would result in -1 if the test vector belongs to the class of circles.

### 2.2 Soft Margin Hyperplane:

The maximal margin classifier described in Section 2.1 (Eqn. 2.23) cannot deal with noisy data and non-linearly separable data as the optimization problem results in empty feasible region (which would indicate that the optimization problem is not feasible) for the primal case and an unbounded objective function for the dual case (Eqn. 2.16) [23]. Section 2.3 describes how SVMs can be utilized to deal with non linearly separable data. This Section discusses the SVM’s approach for noisy data. The problem of noisy data is dealt with by introducing nonnegative slack variables \( \xi_i, \ i=1,..,l \), as illustrated in Figure 2.2, to allow the for possible violation of constraints represented by Eqns. 2.3 and 2.4. \( \xi_i \) is the distance of a misclassified vector to its correct plane.
The optimization problem can now be expressed as [26]:

\[
\begin{align*}
\text{Minimize} & \quad 0.5 \| \mathbf{w} \|^2 + C \sum_{i=1}^{l} \xi_i \\
\text{such that} & \quad y_i (x_i \cdot \mathbf{w} + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad \forall i
\end{align*}
\]

The parameter C is called as the penalty parameter or trade-off parameter. It is a trade-off between number of training errors and margin width. If C is too large, it is equivalent to assigning a very high penalty for non separable points and hence results in fewer training errors (and a narrow margin) [22]. This may lead to overfitting. A small value of C results in more training errors as a wide margin allows more number of training points to be located inside it. To obtain a good generalization performance, an appropriate value for C should be chosen.
The dual form for the above optimization problem is obtained in a manner similar to that obtained for the maximal margin classifier in Section 2.1. The dual form is expressed as:

\[ \text{Maximize } L_D = \sum_{i=1}^{l} \alpha_i - 0.5 \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i \cdot x_j \quad (2.27) \]

such that \[ \sum_{i=1}^{l} \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C \quad (2.28) \]

The constraint in Eqn. 2.28 is similar to the constraint associated with the dual for the maximal margin classifier in Eqn. 2.17. The only difference is that the LMs in Eqn. 2.28 have an upper limit.

In order to find the hyperplane that solves the optimization problem, the LMs have to be determined such that Eqn. 2.27 is maximized subject to the constraints expressed in Eqn. 2.28 If \( \alpha_i^*, i=1,\ldots,l \), are the LMs that are obtained on solving the dual form then the weight vector for the hyperplane is

\[ w^* = \sum_{i=1}^{l} y_i \alpha_i^* x_i = 0 \quad (2.29) \]

The value of \( b^* \) is easily determined by utilizing KKT complementarity conditions, according to which [23],

\[ \text{If } 0 < \alpha_i^* < C, \text{ then } y_i (x_i \cdot w^* + b^*) - 1 + \xi_i^* = 0, \quad (2.30) \]

and \( \xi_i = 0 \quad (2.31) \)

From the above conditions it is clear that only those input input vectors with \( \xi_i = 0 \) and \( 0 < \alpha_i^* < C \) play a role in the determination of the threshold \( b^* \). Input vectors with \( 0 < \alpha_i^* < C \), are the support vectors that lie on the margin. Hence it can be concluded that lie on the margin have a role in the determination of the bias term.
The generalized optimal hyperplane thus obtained is similar to the maximal margin hyperplane and is expressed as [23]:

\[ f(x) = \sum_{i \in SV} y_i \alpha_i^* \langle x_i, x \rangle + b^* \tag{2.32} \]

where \( sv \) stands for support vectors.

2.3 Non Linear SVM:

This thesis considered non linear SVM to separate the sleep and rest behaviors as both behaviors were not linearly separable as illustrated in Figure 2.3. Even for non-linearly separable datasets the SVM principle remains the same, that of finding an optimal hyperplane as illustrated in Figure 2.4. The SVM separates non-linearly separable datasets by performing feature space enlargement followed by linear classification. For this, the SVM combines features to effectively map the features vectors to a very high dimensional feature space where a hyperplane is determined to maximize the separation between the two behaviors sleep and rest [25]. This results in nonlinear decision boundaries in the original space. With the help of a kernel function, the computations are performed in the input space without explicitly performing the mapping.
Figure 2.3 Scatter Plot for the Sleep and Rest Behaviors

In the above figure, the sleep behavior is represented by diamonds and the rest behavior is represented by circles. The scatter plot was plotted for two of the better performing features from a set of 26 features. It can be noted that there is a great amount of overlap between the two behaviors due to which they cannot be separated by a linear decision boundary, forcing us to consider an advanced classification scheme like SVM.
Figure 2.4 An Example for Feature Space Enlargement.

Figure Adapted from [24]. In the above figure (top), it can be noted that the two classes, cross and circles, are not linearly separable in the original 2-D space. With the help of the non-linear function $\Phi$ feature space enlargement is performed such that the two classes can be separated in a linear fashion in the new space. This is equivalent to separating the two classes in a non-linear manner in the original space as illustrated in the bottom portion of the figure.
The formulation of the optimization problem for the datasets which are non-linearly separable is done in a manner similar to that discussed in Section 2.2. The dual of the optimization problem is expressed as:

Maximize \[ L_D = \sum_{i=1}^{l} \alpha_i - 0.5 \sum_{i,j} \alpha_i \alpha_j y_i y_j (\phi(x_i) \cdot \phi(x_j)) \] (2.33)

such that \[ \sum_{i=1}^{l} \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C \] (2.34)

In Eqn. 2.33, \( \Phi \) is the function which performs the non-linear map from the input space to feature space. If the mapping is performed to a very high dimensional feature space, the dot product between the input vectors in Eqn. 2.33 may lead to extensive computations. Similar dot product computations will be required to be performed in the computation of equation of the hyperplane (between different support vectors) and consequently the decision function (between support vectors and the test vector). In order to avoid this, the dot product between different vectors in the high dimensional space is replaced with a kernel function \( K(x_i, x_j) \). The dual is now expressed as:

Maximize \[ L_D = \sum_{i=1}^{l} \alpha_i - 0.5 \sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \] (2.35)

such that \[ \sum_{i=1}^{l} \alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C \] (2.36)

The corresponding decision function can be expressed as:

\[ f(x_i) = \text{sgn} \left( \sum_{i=1}^{N} \alpha_i^* y_i K(s_i, x_j) + b^* \right) \] (2.37)
In Eqn. 2.37, $s_i$ are the training vectors with non-zero LMs (support vectors), obtained on solving Eqn. 2.35. $x_i$ is the test vector, and the value of threshold is computed by utilizing KKT complementarity conditions as discussed in Section 2.2. The kernel function that was employed in this work was Exponential Radial Basis Function (ERBF) kernel, which is expressed as:

$$K(x, y) = e^{-\sqrt{((x-y)^T(x-y))/2\sigma^2}} \quad (2.38)$$

where kernel parameter $\sigma$ is the width of the kernel [28].

This kernel was chosen after experimenting with different kernels namely, Polynomial kernel, Radial Basis Function kernel (RBF), and Sigmoid Kernel, and evaluating classifier performance.

It is important to note here that a kernel can be considered as an equivalent of dot product in some high dimensional space only if satisfies Mercer’s condition [23].

According to Mercer’s condition, if $g(x)$ is a square integrable function such that

$$\int g(x)^2 dx \text{ is finite then } \quad (2.39)$$

$$\int K(x, y)g(x)g(y)dxdy \geq 0 \quad (2.40)$$

The above condition implies that a kernel function ($K(x, y)$) can be considered as inner product in some space if it is a symmetric positive definite function.
2.4.1 The LDA Method:

This Section provides a brief overview of how the LDA method was utilized to perform the classification between the sleep and rest behaviors. In the LDA method, given training vectors belonging to two different classes, the first step involves computing feature weights for these training vectors based on Fisher’s Discriminant which can be expressed as [32]:

\[
v = \sum^{-1} \left( m_s - m_r \right) \quad (2.41)
\]

where \( m_s \) is the mean of all feature vectors belonging to the class sleep, \( m_r \) is the mean of all feature vectors belonging to the class rest, and \( \Sigma^{-1} \) is the covariance matrix. The Fisher’s Discriminant allocates weighting to a feature depending on the information the feature provides to the classifier.

The decision rule is expressed as:

Decide Sleep if \( v^T x_t > v_0 \) \quad (2.42)
Decide Rest if \( v^T x_t < v_0 \)

where \( v^T \) is the transpose of the weight vector obtained from Eqn. 2.41, \( x_t \) is test vector, and \( v_0 \) is threshold.

Thus, given a test vector, its inner product with the weight vector is computed and if the resultant scalar is greater than certain threshold, the test vector is classified as sleep. If the inner product results in a value lower than the threshold the test vector is termed as rest. Numerous thresholds that can discriminate the two classes are considered before settling down on a threshold that results in least classification error.
2.4.2 The MD Method:

Apart from LDA, the other linear classifier that was considered was the MD method. It's a minimum distance classifier that is similar to the Euclidean distance except for the fact that it corrects the correlation between different features apart from automatically scaling each feature to have unit variance [18]. An advantage that MD offers over LDA is that it does not require computation of a threshold.

The MD is expressed as:

\[
 d^2 = (x_t - m_i) \Sigma_i^{-1} (x_t - m_i) \quad (2.43)
\]

where \(x_t\) is a test vector, \(m_i\) is the mean for the \(i^{th}\) class and \(\Sigma_i^{-1}\) is the covariance for the \(i^{th}\) class.

A test vector \(x_t\) is said to belong to a class \(i\) if it is nearest to the template or observation vector of that class.
3. DATA ACQUISITION, PROCESSING AND FEATURE EXTRACTION

A classifier’s performance is largely influenced by the signal quality from the data acquisition system and the transforms used to extract features for classification. The first part of this chapter describes the PVDF sensor, and explains signal amplification. The second part of this chapter deals with the processing that was performed on data segments, the different signal transforms used to characterize the sleep and rest behaviors, and feature extraction.

3.1 The PVDF Sensor:

The PVDF sensor or piezo film, made by Measurement Specialties, Inc. (Hampton-VA), converts the pressure or force from the mouse activity to proportionate electric signals. The sensor covered the entire cage bottom and consisted of a PVDF dielectric sheet that had silver ink sputtered on either side. The sheet was 110 microns thick and was a square with each side being 7”. To allow easy cleaning and to protect the sensor from moisture, a plastic film (2 mm thick) was placed on top of the sensor. An electrical equivalent of the sensor is shown in the Figure 3.1 [12].

![Figure 3.1 Electrical equivalent of the sensor](image)
The model is a voltage source in series with a parasitic capacitor where the voltage source $V_s$ is the voltage that is generated by the application of pressure on the film and $C$ is the capacitance of the PVDF sensor [12]. The capacitance of the sensor is calculated using the equation 3.1.

$$C = \varepsilon \frac{A}{t} \quad (3.1)$$

where $A$ is the area the PVDF sheet, $t$ is the thickness of the PVDF sheet, $C$ is the unknown capacitance, $\varepsilon$ is the permittivity which is expressed as:

$$\varepsilon = \varepsilon_r \varepsilon_0 \quad (3.2)$$

where $\varepsilon_r$ is the relative permittivity of the PVDF which is equal to 12 [12] and $\varepsilon_0$ permittivity of free space which is equal to $8.854 \times 10^{-12}$ F/m

The capacitance of the sensor is determined as:

$$C = (12 \times 8.854 \times 10^{-12} \text{ F/m}) \frac{(0.177 m \times 0.177 \text{ m})}{(110 \times 10^{-6} \text{ m})} = 30.24 \text{nF} \quad (3.3)$$

3.2 Signal Amplification:

The amplifier employed for signal amplification was designed in a manner so that it amplified signals in a particular frequency range. As illustrated in Figure 3.2, the amplifier mainly consisted of a differential amplifier, low pass filter, and non inverting amplifier. The voltages from the two leads of the PVDF sensor served as inputs to the differential amplifier. The AC power supply was regulated to $\pm 10$ V DC power supply and delivered to the two amplifiers.
As shown in the Figure, the frequency response of this circuit had -3dB points at 0.74 Hz and 27.1 Hz. It was highly desirable to have a frequency response of this kind for the following reasons:

1. It was reported in [13] that across different strains of mice, the breath frequency varies from 2.1 Hz through 3.68 Hz respectively during REM and non REM sleep.

2. For any strain of mice, the frequency range should include the fundamental frequency of breathing, which may vary between 2.1 Hz through 3.68 Hz [13], and the corresponding harmonics.

3. It was found in [5] that the 60 Hz power supply signal interfered with the weak piezo signal resulting in a significant spike in the frequency spectrum. The amplifier designed should be able to attenuate such high frequency signal.
4. The lower end of the frequency range (0.74 Hz) ensured that low frequency human interference, which may drive the amplifier into saturation for extended periods of time, and DC components were filtered out. Filtering the DC components was important as the signal of interest was of low frequency with amplitude in the order of 10 mV.

Figure 3.3 Signal Amplifier Frequency Response
3.3 Amplifier Design

The piezo film constitutes the first stage of the signal amplifier and is modeled as a voltage source ($V_s$) in series with a capacitor ($C_1$) [12]. The capacitance associated with $C_1$ keeps varying, depending on the area of contact between the mouse and the sensor, and this may shift the desired frequency range. In order to minimize this, the capacitance $C_2$, as shown in Figure 3.1, is connected in parallel to the sensor capacitance. Consequently, the resulting equivalent capacitance would be less dependent on the sensor capacitance.

The first op-amp in Figure 3.2 acts as a differential amplifier. The differential amplifier in combination with the piezo film and $C_2$ acts as a high pass filter. It attenuates very low frequency signals like common mode noise and amplifies the differential inputs. This stage of the signal amplifier plays a crucial role in preventing the amplifier from going into saturation for extended periods of time and in filtering out DC components. An important point to note here is that the differential amplifier offers large input impedance and thus betters the sensor from variable loading effects of the rest of the circuit.

The gain offered by the differential amplifier is [14]:

$$Gain = \frac{R_4}{R_1} \quad (3.4)$$

where $R_4$ is the feedback resistance, and $R_1$ is the input resistance to the differential amplifier

From Eqn. 3.4, the gain for the differential amplifier was found to be 12.5 or 21.9 dB.

The signal from the differential amplifier was passed through the low pass filter basically to attenuate high frequency noise like that from the 60 Hz supply.
The high pass filter followed by the low pass filter provided a band pass effect in the frequency range 0.74Hz to 27.1 Hz

The actual signal amplification was performed in the final stage of the signal amplifier with the help of a non-inverting amplifier. The feedback resistance $R_6$ was variable from 1 K$\Omega$ to 100 K$\Omega$ by means of a potentiometer. A high value of gain was required especially when the piezo signal corresponded to sleep or rest. The gain offered by the non-inverting amplifier was variable and was determined as [14]:

$$Gain = 1 + \frac{R_6}{R_7}$$  \hspace{1cm} (3.5)

where $R_6$ is the feedback resistance, and $R_7$ is the input resistance to the non-inverting amplifier.

### 3.4 Analog to Digital Conversion:

The amplified signal was fed to a NI PCI-6224 multi channel data acquisition board with the help of a SCB-68 shielded I/O Connector Block. The data card was controlled with LabVIEW 8.2 (National Instruments, Austin-Texas). In order to meet the Nyquist criteria, the sampling frequency was chosen in a manner such that it was more than twice the highest possible significant frequency in the frequency spectrum. As reported in [5], the highest significant frequency was due to 60 Hz power supply. Consequently, the signal was sampled at 128 samples/sec. The low sampling frequency ensured that complications with respect to storage and processing were minimized. The sampled signal was quantized with 16 bits and was stored in binary format before it was read into MATLAB 7.4 (The MathWorks, Natick-MA) for performing data processing and further analysis.
3.5 Data Processing:

Before any analysis can be performed on the raw data it is important that the data is processed such that noise and artifacts are attenuated. Once processing is performed on the data, features can be estimated with less error from the noise and artifact variations.

Initially, the data segments were band pass filtered such that frequencies outside the spectral range of 0.5 Hz through 19 Hz were attenuated. The frequency range ensured that fundamental frequency of sleep and the important harmonics were enhanced. Band-pass filtering was implemented with an FIR filter of order 511. The filter was designed using the window method where a hamming window was used. In order to ensure that there is no phase distortion due to filtering the \texttt{filtfilt} command in MATLAB was used to perform zero phase filtering. The \texttt{filtfilt} command FIR filters the given sequence in forward direction, the sequence obtained is reversed and again FIR filtered. The resulting sequence is reversed to represent the actual filtered sequence. Each contiguous segment was modified to extract segments of fixed length of 8 seconds from which features can be computed. The segment length was chosen based on preliminary analyses which showed low classification error for segments of duration 8 seconds. A logarithmic compression was performed on each extracted data segment to have its compressed version. The compression was performed as per the following equation [6]:

\[
d[n] = \begin{cases} 
  x[n] \left( \frac{v[n]}{T} \right)^{1-\rho} & \text{for } v[n] > T \\
  x[n] & \text{for } v[n] \leq T
\end{cases}
\]

(3.6)

where \( x[n] \) is the data segment that has to be compressed

\( v[n] \) is the envelope of \( x[n] \)

\( T \) is the compression threshold
\[ \rho \text{ is the compression factor} \]

The envelope of the data segment was determined with Hilbert transform magnitude; the median of the envelope was used to determine the compression threshold. The compression factor was chosen as 0.1.

### 3.6 Signal Transforms:

The main signals of interest were sleep and rest behaviors of mice. As mentioned earlier, during sleep or rest behavior, the mouse was lying on the sensor and the signal was mostly due to the breathing of the mouse. It was reported in [13] that the mean frequency of breathing for sleep behavior often overlapped with that of rest behavior. The differences between these two behaviors, though subtle, are illustrated through Figures 3.4 and 3.5. Figure 3.4 shows a six seconds sleep segment. In this Figure, 128 samples correspond to a second, as the sampling frequency was 128 samples/s. It can be noted from the Figure that the mouse breathes about 3 times a second. It can also be noted that the waveform is characterized by regularity in the period and envelope. The rest behavior often precedes or succeeds sleep. Figure 3.5 shows a six seconds rest segment and it can be noted that the breathing pattern (number of breaths per second) or period and the amplitude lacks the regularity that was associated with the sleep behavior.
Figure 3.4 A six seconds sleep segment

Figure 3.5 A six seconds rest segment
The signal transforms chosen should allow for the simple extraction of features that provide vital information to the classifier. In order to characterize the sleep and rest behaviors the signal transforms that were employed are as follows:

1. Power spectrum (PS)
2. Autocorrelation (AC)
3. Generalized spectrum (GS)

### 3.6.1 Power Spectrum:

The PS of a signal determines its power content at various frequencies. Both PS and CA required computation of DFT, the following paragraph provides details about preprocessing that was done before the DFT was computed.

When a finite window is extracted from a contiguous segment the resultant segment can have sharp discontinuities at endpoints. The spectrum of this segment will have undesired high frequencies as a result of these discontinuities. This results in a smeared version of the actual spectrum. This is called as spectral leakage. To minimize spectral leakage, the window chosen was such that its ends were tapered to zero in a smooth manner. Although windows like Hamming, Hanning, Bartlett, and Blackman have their ends tapered to zero, Kaiser window was chosen for windowing as it provided better flexibility over main lobe width and side lobe area [19]. Preliminary analyses with different window lengths and Kaiser $\beta$ values revealed that a window length of 8 seconds or 128*8 samples and $\beta$ value of 3 yields optimum classifier performance. Thus, the compressed data segment was windowed with a Kaiser window with $\beta$ value 3. In order to have more number of grid points or interpolation in frequency domain, the windowed segment was zero padded such that the DFT would be computed over double the number of windowed samples and rounded up to power of 2. The DFT of the windowed segment is given by:
\[ Y[k] = \sum_{n=0}^{N-1} w_\beta[n]d[n]\exp \left( -\frac{j2\pi nk}{N_{\text{FFT}}} \right) \quad (3.7) \]

for \( 0 \leq k < N_{\text{FFT}} \)

where \( N \) is the data segment length in samples, \( k \) is the frequency sample index, and \( N_{\text{FFT}} \) is number of grid points for the computed spectrum.

The PS computed is normalized by signal energy as is expressed as follows:

\[ P[k] = \frac{\left| Y[k] \right|^2}{\sum_{k=0}^{N_{\text{FFT}}-1} \left| Y[k] \right|^2} \quad (3.8) \]

The PS for an 8 second segment of both sleep and rest behaviors is shown in Figure 3.6. The sleep behavior is characterized by peaks at the fundamental frequency of sleep and corresponding harmonics. The rest behavior also has peaks at positions identical to that of sleep behavior. However, these peaks are not strong enough, especially the ones in the lower spectral region.
3.6.2 Autocorrelation:

The autocorrelation is the time domain version of the PS. The AC function decays rapidly for a process that rapidly changes with time and decays slowly for a process that change slowly with time [15]. If a process includes periodic components, the AC for that process will also have periodic components. These properties of AC come in handy in characterizing the sleep and rest behaviors. The AC was computed as follows:

\[
r[\lambda] = \frac{1}{s} \sum_{n=0}^{N-1} d[n]d[n + \lambda]
\]  

(3.9)
where $\lambda$ is the sample lag, $s$ is a normalization factor to normalize the zero lag AC value. Figure 3.7 shows the AC plot for 8 second segment of the sleep and rest behaviors.

![Comparison of AC for Sleep and Rest states](image)

**Figure 3.7 Comparison of AC for sleep and rest behaviors**

From the figure, both the sleep and rest behaviors appear to be periodic. The strong peak at a lag of around 0.3 seconds for sleep behavior distinguishes it from the rest behavior. The AC plot for sleep behavior is a low frequency one when compared to that for rest behavior.
3.6.3 Generalized Spectrum:

Most of the spectral analysis techniques work well for analysis of stationary signals but their performance on non-stationary signals may lead to ambiguous results [16]. The GS can be considered as an extension of PS that utilizes phase information of a signal to characterize special forms of non-stationarities. The Collapsed Average (CA) of the GS detects correlation between different spectral bands in the frequency domain [5]. This is similar to AC which detects correlations in the time domain. The expected value of CA for a stationary random process is zero for distinct frequency components [17]. Non stationary activity results in an increase in the CA values over a frequency range. A harmonic process results in peaks in the CA. The CA was computed as follows:

\[
C[\lambda] = \frac{1}{s} \left| \sum_{k=1}^{(N_{FFT}/2)-2\lambda} Y[k] Y^*[k + \lambda] \right| \quad (3.10)
\]

where \(s\) is normalization factor to normalize CA value at zero frequency lag, the superscript * denotes the complex conjugate

From the above eqn., it can be noted that the summation is evaluated only over half the spectrum. This was done to ensure that CA exhibits desired properties [17]. Artifacts may exist near zero lag due to zero padding and windowing and hence the frequency lag ranges should be selected in a manner such that these artifacts are avoided. For this reason, areas near zero lag (below 0.4 Hz) in the CA plots were excluded from analysis. Figure 3.8 shows CA plots for sleep and rest segments.
In the Figure 3.8, the peaks for sleep behavior are strong when compared to the rest behavior because of more regularity in the time envelope of the sleep signal as well as its consistently stronger periodicity. The other difference that can be noted is the energy distribution over different frequency lags.

**3.7 Feature Extraction:**

From Figures 3.6, 3.7, and 3.8, suggest that the differences between sleep and rest behaviors can be exploited to classify these behaviors by extracting appropriate features from the three transforms. However, these differences were not so obvious when 50 random segments from both the behaviors were selected and the mean values for each of the three transforms were plotted. This was due to the fact that there was lot of similarity between the two behaviors.
This resulted in sleep behavior being scored as rest and vice versa on several occasions. These results are illustrated in Figures 3.9, 3.10 and 3.11. The broken boxes in these figures describe the variability that was consistently observed between the two behaviors.

![Power Spectrum for 50 random segments](image)

**Figure 3.9 PS plot for 50 sleep and rest behaviors**

From the Figure 3.9, it is evident that the repeatability in breathing, which was observed for the sleep segment in Figure 3.6, is not witnessed. The features that were extracted after observing different PS plots for 50 random segments were mostly based on the magnitude of the maximum PS peaks or the position of these peaks under different frequency ranges. An important frequency range considered was sleep range which varied from 1.5 Hz -4.5 Hz. This broad frequency range was chosen based on the fact that breathing rate in mice varies between 2.1 Hz through 3.68 Hz [13]. The other frequency ranges considered were based on observation of different PS plots. These frequency ranges covered harmonics that would have resulted due to the fundamental frequency of breathing. A majority of the 26 features extracted were from PS.
The AC peak that was prominent at a lag of around .3 sec in Figure 3.7 was not observed when AC for 50 random sleep and rest segments was considered. The different AC features considered includes position and magnitude of maximum AC peak in the sleep range. Sleep range considered for AC included the lag region of 0.22 - 0.66 sec as illustrated in Figure 3.10. An interesting AC feature, based on previous work, was determining the distance of maximum AC peak in the sleep range from the mean period of sleep (0.34 sec). This feature would result in a zero value when the maximum AC peak in the sleep range coincides with the peak at a lag of 0.34 sec. A similar feature was also considered for PS. Other AC features included determining the area under the AC plot in the sleep range and overall lag range, after computing the absolute value of AC.
Figure 3.11 CA plot for 50 sleep and rest behaviors

The CA plot for sleep segment in Figure 3.8 was characterized by peaks at the fundamental frequency of breathing and the corresponding harmonics indicating the periodic nature of the sleep segment. But these peaks were not so dramatic when CA for 50 random sleep segments was considered. After analyzing various CA plots, the features considered were low frequency lag area under the CA from 0.4 – 2 Hz, CA area under 4 – 6 Hz and 6 – 8 Hz frequency sub-bands, and maximum of CA area under different frequency sub-bands. In the Figure 3.11, it appears that CA area under 10 – 12 Hz frequency sub-band may be a good feature. However this feature was not considered for analysis as the variability provided by this feature was not consistent across different CA plots.

A detailed list of all the 26 features that were extracted from the 3 transforms is presented in Table 3.1.
<table>
<thead>
<tr>
<th>No.</th>
<th>Feature name</th>
<th>Feature Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ca.lfp</td>
<td>Low frequency lag area under the CA from .4 to 2 Hz</td>
</tr>
<tr>
<td>2</td>
<td>abs(sp.sp-3)</td>
<td>Distance of max PS peak in the sleep range from 3 Hz</td>
</tr>
<tr>
<td>3</td>
<td>Ac.sm</td>
<td>Max AC peak magnitude in the sleep range</td>
</tr>
<tr>
<td>4</td>
<td>Abs(ac.sp -.34)</td>
<td>Distance of the max AC peak in the sleep range from the mean period of sleep</td>
</tr>
<tr>
<td>5</td>
<td>Ac.spt</td>
<td>Position of max AC peak in the global range of interest</td>
</tr>
<tr>
<td>6</td>
<td>Ac.smt</td>
<td>Mag. of max AC peak in the global range of interest</td>
</tr>
<tr>
<td>7</td>
<td>Amp</td>
<td>Amplitude of the segment</td>
</tr>
<tr>
<td>8</td>
<td>Ac.sp ll</td>
<td>Position of max AC peak in the lower lag region</td>
</tr>
<tr>
<td>9</td>
<td>ac.sm ll</td>
<td>Max AC peak in the lower lag region-0.05-.22 sec</td>
</tr>
<tr>
<td>10</td>
<td>Sp.smlt</td>
<td>Max SP peak in .5-1.5Hz range</td>
</tr>
<tr>
<td>11</td>
<td>Sp.smlt1</td>
<td>Max SP peak in 4.5-7 Hz range</td>
</tr>
<tr>
<td>12</td>
<td>Sp.smlt2</td>
<td>Max SP peak in 7-10 Hz range</td>
</tr>
<tr>
<td>13</td>
<td>Sp.smlt3</td>
<td>Position of max SP peak in 10-12 Hz range</td>
</tr>
<tr>
<td>14</td>
<td>Sp.smlt3</td>
<td>Max SP peak in 10-12 Hz range</td>
</tr>
<tr>
<td>15</td>
<td>Sp.smlt4</td>
<td>Max SP peak in 12-16 Hz range</td>
</tr>
<tr>
<td>16</td>
<td>Sp.sp</td>
<td>Position of max SP peak in 1.5-4.5 Hz range</td>
</tr>
<tr>
<td>17</td>
<td>Sp.sm</td>
<td>Max SP peak in 1.5-4.5 Hz range</td>
</tr>
<tr>
<td>18</td>
<td>Sp.spt</td>
<td>Position of max SP peak in .5-19 Hz range</td>
</tr>
<tr>
<td>19</td>
<td>Sp.smt</td>
<td>Max SP peak in 0.5-19 Hz range</td>
</tr>
<tr>
<td>20</td>
<td>Ca.sfb1</td>
<td>CA area under 4-6 Hz frequency sub band</td>
</tr>
<tr>
<td>21</td>
<td>Ca.sfb2</td>
<td>CA areas under 6-8 Hz frequency sub band</td>
</tr>
<tr>
<td>22</td>
<td>Ca.mx</td>
<td>Max of CA area under different frequency sub-bands</td>
</tr>
<tr>
<td>23</td>
<td>Ca.ps</td>
<td>Location of Max of CA area under diff. freq. sub-bands</td>
</tr>
<tr>
<td>24</td>
<td>nac.a</td>
<td>AC area in the range of interest (.5-19 Hz)</td>
</tr>
<tr>
<td>25</td>
<td>nac.a2</td>
<td>AC area in the lag region (.34-.41 sec)</td>
</tr>
<tr>
<td>26</td>
<td>Sp.smlt5</td>
<td>Max SP peak in 16-19 Hz range</td>
</tr>
</tbody>
</table>
4. FEATURE SELECTION

Once all the features are extracted, an optimal feature subset that can characterize the two behaviors has to be determined. This helps in exclusion of redundant features and speeds up the process of classification. It also results in better generalization performance when compared to utilizing all the features at once and evaluating classifier performance. This chapter provides details about the feature selection methods that were adopted for the SVM classifier and the linear classifiers.

4.1 Optimal feature subset determination for SVM

To determine optimal feature subset for SVM an exhaustive algorithm like Monte Carlo simulations may have been considered. But the SVM method also required determination of optimal SVM parameters. The SVM parameters that had to be optimized were the trade-off parameter \((C)\) and the kernel parameter \((\sigma)\). The problem was solved by employing a Genetic Algorithm (GA). The GA simultaneously determined the optimal feature subset (5 features) and values of SVM parameters \(C\) (penalty) and \(\sigma\) (kernel parameter).

GAs are powerful stochastic search and optimization techniques based on principles of natural selection and natural genetics [20]. The GA employed in this work is a modified and extended version of the one used in [21], where its purpose was to determine optimal feature subset.

The flowchart for the GA utilized in this work is illustrated in Figure 4.2. Possible candidate solutions, known as chromosomes, were first randomly created to form an initial population of 100 chromosomes \((N)\). A chromosome consisted of 3 genes, \(C, \sigma\), and a subset of randomly chosen 5 features (from the original 26). \(C\) and \(\sigma\) values in each chromosome were represented in terms of a fixed
number of bits. The maximum number of bits allocated for these parameters was
dependent on the range of their decimal values. Based on the suggestions of
[22], the maximum values for $C$ and $\sigma$ were respectively 10000 and 30. Hence
the chromosome had 14 bits allocated for $C$, 5 bits allocated for $\sigma$. The
chromosomes were then evaluated to determine their fitness. The fitness value
for a chromosome was the classification error due to it. The classification error
was computed by performing Monte Carlo simulations for 10 runs. For each run
bootstrapping was performed to randomly select 300 training and 60 test
segments for each class. The goal of the GA was to determine a chromosome
that would result in least classification error. In the second step, the top N/2
chromosomes were retained based on their fitness value, and N/2 new
individuals were generated by a crossover between randomly selected
chromosomes and mutations of randomly chosen chromosomes from the initial
population. For crossover operation, pair of chromosomes was merged based
on randomly chosen crossover points to form two off-springs. For each cross
over operation, there were two cross-over points, one for the bit combination of
$C$, $\sigma$, and the other for the decimal combination of features. The number of
parents selected for cross over were 40 % of the initial population. The crossover
operation is illustrated in Figure 4.1.
Figure 4.1 Crossover Operation

During mutation, the bits of a chromosome were flipped whenever the mutation probability (0.2) for a bit was greater than a randomly determined probability for that bit. Similarly, a feature was replaced by a random feature whenever the mutation probability for that feature was greater than a randomly determined probability. The number of chromosomes selected for mutation was 10% of the initial population.
Figure 4.2 GA to simultaneously optimize feature subset & SVM parameters

Start

Form initial population (N) and determine fitness values for individual chromosomes

Retain N/2 individuals based on their fitness values (N1)

Form new N/2 individuals through cross-over and mutation (N2)

New Gen. = N1+N2, Eval. fitness

Is the stopping criterion met?

No

Yes

End
The last step in the algorithm merged the new N/2 individuals generated by the crossover and mutations with most fit N/2 individuals to form a new generation of population. In this new generation, the chromosome with the least classification error was determined, and if the classification error was less than the desired classification error, 5%, the algorithm was terminated. In case the classification error was greater than the desired value, the algorithm was repeated from step 2 onward until the desired classification error was achieved or when the problem converged (the classification error remains constant for many generations). Due to time constraints the maximum number of generations for the GA was set to 25.

The results obtained for different generations of the GA are presented in Table 4.1. These results are illustrated through Figure 4.3. The feature names for the features listed in the Table can be obtained from Table 3.1.

From the GA, the 5 best features were:
1. maximum AC peak magnitude in the sleep range
2. maximum AC peak in the lag region of 0.05s-0.22s
3. maximum PS peak magnitude in 10 Hz-12 Hz range
4. position of maximum PS peak in the sleep range
5. maximum amplitude of the signal

The optimal values for $C$ and $\sigma$ were 2,228 and 18 respectively.

One of the best features obtained was amplitude dependent. This feature may hurt the classifier performance because every time the sensitivity changes the classifier has to be re-trained and tested. However, this feature emerged as one of the best features for different trials of the GA and hence it was retained.
Table 4.1 Performance of the GA for 25 generations

<table>
<thead>
<tr>
<th>Generation Number</th>
<th>C</th>
<th>σ</th>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>F4</th>
<th>F5</th>
<th>Min. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3379</td>
<td>20</td>
<td>3</td>
<td>7</td>
<td>13</td>
<td>14</td>
<td>16</td>
<td>28.91%</td>
</tr>
<tr>
<td>2</td>
<td>2390</td>
<td>9</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>18</td>
<td>27.67%</td>
</tr>
<tr>
<td>3</td>
<td>2390</td>
<td>9</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>18</td>
<td>27.67%</td>
</tr>
<tr>
<td>4</td>
<td>2390</td>
<td>9</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>18</td>
<td>27.67%</td>
</tr>
<tr>
<td>5</td>
<td>2390</td>
<td>9</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>18</td>
<td>27.67%</td>
</tr>
<tr>
<td>6</td>
<td>5907</td>
<td>20</td>
<td>3</td>
<td>7</td>
<td>13</td>
<td>14</td>
<td>16</td>
<td>27.33%</td>
</tr>
<tr>
<td>7</td>
<td>9492</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>7</td>
<td>14</td>
<td>16</td>
<td>26.92%</td>
</tr>
<tr>
<td>8</td>
<td>9492</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>7</td>
<td>14</td>
<td>16</td>
<td>26.925</td>
</tr>
<tr>
<td>9</td>
<td>1238</td>
<td>23</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>25.25%</td>
</tr>
<tr>
<td>10</td>
<td>1238</td>
<td>23</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>25.25%</td>
</tr>
<tr>
<td>11</td>
<td>1238</td>
<td>23</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>25.25%</td>
</tr>
<tr>
<td>12</td>
<td>1238</td>
<td>23</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>25.25%</td>
</tr>
<tr>
<td>13</td>
<td>1238</td>
<td>23</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>25.25%</td>
</tr>
<tr>
<td>14</td>
<td>1238</td>
<td>23</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>25.25%</td>
</tr>
<tr>
<td>15</td>
<td>1238</td>
<td>23</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>25.25%</td>
</tr>
<tr>
<td>16</td>
<td>1238</td>
<td>23</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>25.25%</td>
</tr>
<tr>
<td>17</td>
<td>1238</td>
<td>23</td>
<td>3</td>
<td>7</td>
<td>8</td>
<td>14</td>
<td>16</td>
<td>25.25%</td>
</tr>
<tr>
<td>18</td>
<td>4802</td>
<td>26</td>
<td>3</td>
<td>7</td>
<td>14</td>
<td>16</td>
<td>24</td>
<td>24.75%</td>
</tr>
<tr>
<td>19</td>
<td>2228</td>
<td>18</td>
<td>3</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td>16</td>
<td>24.67%</td>
</tr>
<tr>
<td>20</td>
<td>2228</td>
<td>18</td>
<td>3</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td>16</td>
<td>24.67%</td>
</tr>
<tr>
<td>21</td>
<td>2228</td>
<td>18</td>
<td>3</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td>16</td>
<td>24.67%</td>
</tr>
<tr>
<td>22</td>
<td>2228</td>
<td>18</td>
<td>3</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td>16</td>
<td>24.67%</td>
</tr>
<tr>
<td>23</td>
<td>2228</td>
<td>18</td>
<td>3</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td>16</td>
<td>24.67%</td>
</tr>
<tr>
<td>24</td>
<td>2228</td>
<td>18</td>
<td>3</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td>16</td>
<td>24.67%</td>
</tr>
<tr>
<td>25</td>
<td>2228</td>
<td>18</td>
<td>3</td>
<td>7</td>
<td>9</td>
<td>14</td>
<td>16</td>
<td>24.67%</td>
</tr>
</tbody>
</table>
From the GA plot it is clear that as the number of generations increases the classification error drops down. As the desired classification was not achieved even after 25 generations, the GA was terminated at that point.
4.2 Optimal feature subset determination for LDA and MD

For SVM, the best combination of 5 features was determined with GA. However, for the LDA and MD classifiers a simple approach was used to obtain the best combination that characterizes the sleep and rest behaviors in mice. In order to obtain the best combination of 5 features from a set of 26 features, a total of 65780 combinations were considered. Each of these 65780 combinations was evaluated 100 times. For each run, bootstrapping was performed to randomly select 300 training and 60 test segments for each class. At the end of the 100 Monte Carlo simulations average probability of error and 95% confidence limits for a combination were computed. After evaluating the average probability of error in a similar manner for all the combinations, the combination with the least average probability of error was considered as the best combination. For both the classifiers, the best feature combinations along with their average probability of error are listed in Table 4.2. The features names for the features listed in the table can be obtained from the Table 3.1.

Table 4.2 Indices for Best Feature Combinations and performance for LDA and MD

<table>
<thead>
<tr>
<th>CLASSIFIER</th>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>F4</th>
<th>F5</th>
<th>MIN. ERROR</th>
<th>95% CONFIDENCE LIMITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA</td>
<td>11</td>
<td>16</td>
<td>17</td>
<td>25</td>
<td>26</td>
<td>31%</td>
<td>± 0.67</td>
</tr>
<tr>
<td>MD</td>
<td>21</td>
<td>19</td>
<td>17</td>
<td>25</td>
<td>26</td>
<td>35%</td>
<td>± 0.86</td>
</tr>
</tbody>
</table>

From the table it is clear that a majority of features for both LDA and MD are similar. This can be attributed to the fact that both the classifiers are linear in nature. However, these feature combinations are different from the one that was obtained for SVM.
5. EXPERIMENT AND RESULTS

This chapter describes the experimental setup, the training of the SVM and other classifiers, and the performance of these classifiers in characterizing the sleep and rest behaviors in mice. The chapter also provides details about the method that was employed to validate the performance of the SVM.

5.1 Experiment:

The experiment was performed as part of our research project to track multiple behaviors which were identified based on different behavioral studies done on rodents. The different behaviors considered for tracking were sleep, rest, active, alert, and grooming. These behaviors were observed as follows:

- **Sleep**: The mouse remains still with eyes closed (or half open) or has its head tucked under its body.
- **Rest**: The mouse is still with eyes open and with little to no head motion. It often preceded or succeeded sleep.
- **Alert**: The mouse is stationary but actively observing the environment with head movements and sniffing.
- **Active**: The mouse is walking, running, eating, drinking and leaning against the walls of the cage.
- **Grooming**: The mouse scratches its face and sometimes its tail or the back portion of its body.

However for this thesis, the signals of interest were sleep and rest. This was due to the fact that sleep studies on rodents generally involve characterizing sleep activity from wake activity or rest (quiet wake) behavior, or characterizing different stages during sleep cycle. The classification between sleep and wake behaviors was performed successfully in [5]. For this thesis, classification was to be performed between the linearly non separable sleep and rest behaviors using
some advanced classification scheme as the classification method employed in [5] (LDA) failed to give desired results.

From the description of sleep and rest behaviors, it can be noted that there is some ambiguity about the type of behavior when the mouse is still and has its eyes open. Such ambiguity about behavior would be tough to resolve even by using EEG/EMG recordings. Hence it is possible that sleep behavior can be scored as rest or vice versa by an observer and thus leading to potential errors in data collection.

Figure 5.1 illustrates the experimental setup. An interface was developed to allow observers on laptop stations to label the different behaviors in terms of voltage levels. This was implemented with the help of menu interfaces to DC voltage generators. When an observer selected a menu button corresponding to a
particular behavior, an appropriate voltage level was sent from the voltage generator to the data acquisition card along with the behavior. Thus a database was formed that had different mice behaviors and their corresponding voltage levels. This allowed for extraction of signal segments corresponding to either sleep or rest for training or testing the classifier.

The experiment involved 4 C57BL/6J male, a commonly used strain of mice for sleep studies. The four mice were placed individually in separate PVDF-sensor cages. There were 4 observer stations consisting of a laptop to control the voltage generator sending DC voltages levels to the PCI 6224 multi-channel data card. The voltages from the voltage generator varied from -1.5V to 2V with -1.5 being assigned to the sleep behavior. A total of 8 channels were used, 4 for the sensor signals and 4 for the voltage levels coming from the observer stations. The observers were required to choose one of the six options that were available from a menu on their laptop screens. The first 5 options were the 5 behaviors that were described at the beginning of this Section. The sixth option, called “undefined” was that of refusing to make a decision about the observed mice behavior. The observers were instructed to select the sixth option when they had difficulty in making a decision on the 5 behaviors, or when the mice performed other activities like sitting on the rod of the water bottle, etc. In addition, the observers selected the sixth option when they experienced fatigue or when the data collection had to be stopped temporarily. Another instruction for the observers was to ignore intermittent stirrings during sleep behavior. This allowed for recording of long sleep bouts.

The experiment was conducted for 7 days resulting in 8.2 hours of observed sleep and 1.6 hours of observed rest. A total of 9 observers helped in data collection. The data was collected at different times of the day to ensure that different behaviors are adequately represented.
Labeled contiguous segments were parsed into non-overlapping segments of 8 seconds for feature extraction. A contiguous segment was considered for analysis only when its size was greater than a guard band (10 seconds) and one segment (8 seconds), for a segment a guard band was formed by excluding 5 seconds from the ends of the segment (as an observer needs few seconds to respond when behavior transition occurs). This resulted in an accurate representation of the observed mice behavior but reduced the overall number of segments. The trade-off was acceptable as the resulting segments were still large in numbers to provide an adequate representation of the two behaviors from which features can be extracted.

5.1.1 The SVM Approach:

The GA determined an optimal feature subset of 5 features and the SVM parameters. The first step of the SVM classification was to use bootstrap method to randomly selected 300 segments from each class to train and 60 segments from each class to test (excluding those already chosen for training) the classifier. Feature normalization was performed such that each component of the feature vector had zero mean and unit variance [29] (For this, the feature vectors belonging to both the classes were combined and mean and standard deviation of individual features were computed. The next step was to subtract the computed mean from its corresponding feature and divide the difference by the computed standard deviation.). This ensured equal importance of all 5 features. The next step was to label the training vectors corresponding to sleep as +1 and those corresponding to rest as -1. The third step involved computing the kernel matrix of training vectors as per Eqn. 2.38, and constraints as per Eqn. 2.36. As mentioned in Chapter 2, the kernel function considered in this thesis was ERBF. The optimization problem of Eqn. 2.35 was solved using the quadprog command in MATLAB to obtain LMs and hence support vectors. The command finds minimum of a quadratic programming problem, however, the dual form of the
optimization problem was to be maximized. This was addressed by inverting signs of the input arguments that were part of the objective function (Eqn. 2.35). In order to solve the dual optimization problem, the command employs active set method and obtains initial feasible solution from a linear programming problem [30]. To obtain reliable results using the *quadprog* command, it is important to not to rely on the default iteration settings of MATLAB, as noted in [31]. For this work, the number of iterations was set to 10000.

The bias term in the decision function (Eqn. 2.37) was computed by considering KKT complementarity conditions as discussed in Chapter 2. As all the unknowns for the decision function were now determined, it was easy to make a decision on a test vector. A test vector was considered to be sleep if Eqn.2.37 resulted in +1 and rest when the value was -1. This was done for all the 60 test vectors that were selected by the bootstrap approach and the error in classification was noted.

Monte Carlo simulations were carried out from step 1 through last step for 100 times to determine mean classification error and 95% confidence limits.

The results obtained with the SVM approach are illustrated in Figure 5.2 and are also shown in Table 5.1. The figure shows how the linearly non separable (as noted in Figure 2.3 and from the plots of signal transforms) sleep and rest can be classified using the SVM method. In order to generate the figure, 35 training vectors from each class were considered. Each vector consisted of two of the best performing features. From the figure, it can be noted that the SVM results in a non linear decision boundary between the sleep and rest behaviors. Also, because of similarity in behaviors some of sleep vectors were classified as rest and vice versa. In the Figure, vectors with white markers in them are the support vectors. A majority of vectors in this plot turned out to be support vectors. This suggests that most of the training vectors influenced decision making. The white dotted lines represent margins (as discussed in chapter 2). These margins are
non linear as they were mapped to the original (5 D) space from very high (infinite) dimension space (where they were linear).

Figure 5.2 SVM Classification for the Sleep and Rest Behaviors

Figure obtained by utilizing software of [28]
5.1.2 The LDA Method:

This Section describes how the LDA method was utilized to perform the classification between the sleep and rest behaviors. The optimal feature subset for the LDA method was determined with an algorithm described in Section 4.2 of Chapter 4. The first step was to use bootstrap method to randomly selected 300 segments from each class to train and 60 segments from each class to test (excluding those already chosen for training) the classifier. The next step involved computing feature weights for the training vectors based on Fisher’s Discriminant (Eqn. 2.41). The covariance matrix was obtained by combining the feature vectors belonging to both the classes and computing the covariance of the resultant matrix. The next step was to obtain the threshold $\nu_0$ that would efficiently discriminate sleep from rest. For this, the 60 sleep test vectors were multiplied with the weight vector obtained from Eqn. 2.41 to form class 1 and similarly, the 60 rest test vectors were multiplied with the weight vector to form class 2. A set of 200 thresholds were determined that ranged from the maximum and minimum values of the above two classes. The threshold that resulted in least classification error was considered as optimum threshold. Once the threshold was determined the classification error was computed.

Monte Carlo simulations were carried out from step 1 through last step for 100 times to determine mean classification error and 95% confidence limits. The results obtained with the LDA approach are shown in Table 5.1.

5.1.3 The MD Method:

Like the previous classifiers, the first step was to use bootstrap method to randomly selected 300 segments from each class to train and 60 segments from each class to test (excluding those already chosen for training) the classifier. The next step was forming templates or observation vectors for the sleep and rest behaviors. The template for sleep behavior was formed by taking the mean of all
its training feature vectors. The template for rest behavior was also formed in a similar manner. In the third step, the inverse covariance matrix was formed by taking inverse of the mean of the covariances of sleep and rest behaviors.

The next step was to consider sleep test vectors, calculate the MD with the templates and count the number of sleep test vectors that were misclassified. A sleep test vector was considered to be misclassified, if its MD from sleep template vector was greater than that of rest template vector. In a similar manner, the number of rest test vectors that were misclassified was computed. Finally, the average of the sleep classification error and rest classification error was taken to determine the overall classification error.

Monte Carlo simulations were carried out from step 1 through last step for 100 times to determine mean classification error and 95% confidence limits. The results obtained with the MD approach are shown in Table 5.1 along with the results of the other classifiers.

5.2 Results:

Table 5.1 illustrates the results that were obtained by employing different classifiers for detecting sleep and rest behaviors in mice. From the table it is obvious that the SVM classifier performs much better when compared to the LDA and MD. The reason for this can be attributed to the fact the linear classifiers resulted in a linear decision boundary and consequently had difficulty in learning the sleep and rest behaviors which had subtle differences between them.
Table 5.1 Comparison of SVM performance w.r.t. other classifiers for Sleep vs. Rest behaviors.

<table>
<thead>
<tr>
<th>CLASSIFIER</th>
<th>CLASSIFICATION ERROR</th>
<th>95% CONFIDENCE LIMITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>25.45 %</td>
<td>± 0.88</td>
</tr>
<tr>
<td>LDA</td>
<td>32.12 %</td>
<td>± 2.78</td>
</tr>
<tr>
<td>MD</td>
<td>34.57 %</td>
<td>± 0.85</td>
</tr>
</tbody>
</table>

The accuracy obtained with the SVM can be termed as reasonable as both the behaviors were very similar in characteristics. The observers scoring these behaviors had difficulty deciding on a behavior, when the mouse was still and had its eyes open or closed. This would have lead to sleep behaviors being scored as rest or vice versa. The other point that can be noted from the table is that the performance of LDA and MD was similar. Unlike LDA, the SVM performance was not influenced by the value of chosen threshold.

5.3 Cross Validation:

This section describes cross validation method that was used to validate the correctness of the SVM approach. For this, the SVM was employed to classify two easily separable behaviors.

From the experimental data, the sleep and rest behaviors of mice were considered as class1 called “sleep-rest” and different mice activities like grooming, active (eating, jumping, running, etc), and alert (sniffing) were considered as class2 called “wake”. The signal generated from the PVDF sensor was regular when the mouse was in sleep-rest behavior and was highly erratic during the wake behavior. Due to this, when the three signal transforms were applied on the sleep-rest and wake segments, the differences between these behaviors were pretty obvious as illustrated in Figure 5.3.
Figure 5.3 Signal Transform comparison of sleep-rest (blue) and wake signals for (a) PS, (b) AC, and (c) CA.
The SVM classification was performed on the sleep-rest and wake behaviors in a manner similar to that described in Section 5.1. But before performing the classification, GA was used to determine optimal feature subset and SVM parameters.

For this classification problem, the 5 best features were:
1. distance of maximum PS peak magnitude in the sleep range from the fundamental frequency of breathing (3 Hz)
2. maximum AC peak magnitude in the sleep range
3. maximum amplitude of the segment
4. maximum PS peak magnitude in 7Hz-10 Hz range
5. position of maximum PS peak magnitude in 10Hz-12 Hz range

The optimal values for $C$ and $\sigma$ were 3565 and 29 respectively.

Figure 5.4 explains the performance of SVM on the easily separable sleep-rest and wake behaviors. The plots were generated in a manner similar to the ones that were generated for sleep and rest behaviors. An important thing that can be noted from the plot is that very few training vectors play a role in decision making as there are very few support vectors.
Figure 5.4 SVM Classification for the Sleep-Rest and Wake Behaviors

Bottom portion of the Figure obtained by utilizing software of [28].
Table 5.2 illustrates the performance of the SVM classifiers with respect to other linear classifiers, LDA and MD, in classifying sleep-rest and wake behaviors.

Table 5.2 Comparison of SVM performance w.r.t other classifiers for Sleep-rest vs. Wake behaviors.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Classification Error</th>
<th>95% Confidence Limits</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>4.32 %</td>
<td>± 0.38</td>
</tr>
<tr>
<td>LDA</td>
<td>4.11%</td>
<td>± 0.55</td>
</tr>
<tr>
<td>MD</td>
<td>4.95 %</td>
<td>± 0.58</td>
</tr>
</tbody>
</table>

From the table, it is evident that the performance of SVM was on par with that of the linear classifiers. This validates the SVM algorithm that was utilized to characterize the more difficult sleep and rest behaviors.
6. CONCLUSION AND FUTURE WORK

This thesis developed a robust method to classify the linearly non separable sleep and rest behaviors in mice by employing the powerful SVM technique for classification. The salient features of this thesis can be summarized as:

1. The SVM method showed significant improvement in accurately classifying behaviors with subtle differences like sleep and rest when compared to the previously employed LDA.
2. The SVM technique offered direct methods to control overfitting and the optimization problem converged to a unique solution, unlike Neural networks.
3. The method did not require determination of an optimal threshold to separate two classes, unlike LDA.
4. The features employed for classification were extracted from simple transforms. Through genetic algorithm, an optimal feature subset of 5 features was determined from a total of 26 features. These 5 best features were able to provide information to the SVM classifier that was not possible with the optimal feature set for linear classifiers The GA also determined optimal SVM parameters.
5. The overall system is of significant value for experiments required for identifying sleep related genes that typically require large numbers of mice for genetic mapping and statistical analyses [33].

Some of the limitations of the SVM method include, determining the right type of kernel and its parameters, the value for the trade-off parameter. Also, the optimization problem is time consuming.
6.1 Future Work:

In this thesis, the linearly non separable sleep and rest behaviors were characterized with 75% classification accuracy. The accuracy can be further improved by developing own kernels and trying new signal transforms. Once this is achieved, the SVM method can be potentially extended to classify the different stages of sleep cycle namely REM and non-REM. The system can also be utilized to classify different behaviors in mice which may help studying the impact of different drugs on humans.
REFERENCES


<http://www.mathworks.com/access/helpdesk/help/toolbox/optim/>


<http://www.engr.uky.edu/~donohue/>


[38] Huang TM, Kecman V, “Bias Term b in SVMs Again”, Proceedings of European Symposium on Artificial Neural Networks, Burges (Belgium), 2004, pp. 441-448.


VITA

Shantilal was born on 1st March 1981 in Kitalsar, Rajasthan, India. He received his Bachelor’s Degree in Electronics and Communications Engineering from Osmania University, India, in the year 2005. He is expected to earn his Master’s degree in Electrical and Computer Engineering from University of Kentucky, Lexington, in the year 2007. His research interests are in Signal Processing and Pattern Recognition. As a graduate research assistant at University of Kentucky, he worked in tandem with the Department of Electrical and Computer Engineering and Department of Biology.