CHAPTER 3

HIDDEN MORKOV MODEL AS A POST CLASSIFIER FOR CLASSIFICATION OF EPILEPSY RISK LEVELS

3.1 INTRODUCTION

The Hidden Markov Model (HMM) has been intensively studied during past three decades. Major applications in speech signal processing have been benefited from the utilization of HMM. Theoretical approaches on HMM helped researchers to understand its mechanisms. This chapter lucidly presents the application of HMM for classification of epilepsy risk levels from dimensionally reduced EEG signal features. HMM is a powerful statistical tool for modelling generative sequences that can be characterized by an underlying process generating an observable sequence explained by Penny et al (1998). Hidden Markov model (HMM) is basically a non-deterministic probabilistic finite state machine, which can be constructed inductively. It has been widely used in speech recognition and DNA modelling. It is found that hidden Markov models are very suitable for epilepsy detection and classification problems and that they are able to model accurately segmented EEG signals conferred by Easwaramoorthy et al (2010). The general aim of any signal segmentation method is to partition a given signal into consecutive regions of interest. In the context of the EEG signal, the role of dimensionality reduction is to determine the possible onset and offset boundaries, as well as the peakedness of the segments. One of the advantages of probabilistic models over traditional methods is that a
confidence measure for each segmented signal is given by the log likelihood of the observed signal.

The conventional method of confirmation of Epilepsy is by detecting spikes and sharp waveforms in acquired EEG signals. However, the extent of existence of epilepsy is undetermined. A patient might have no epilepsy where as other might be critical. Hence, it is essential to classify the risk level of an epileptic EEG. As the EEG signal is non-stationary in nature, probabilistic models for signal analysis are essential and these are achieved by Hidden Markov Model (HMM). Epileptic EEG is acquired and dimensionality reduction is performed by using Singular Value Decomposition (SVD), Principal Component Analysis (PCA) and Independent Component Analysis (ICA). However, reduced matrix would still lead to computational complexity. A further reduction in the input is achieved by Vector Quantization (VQ). The final risk level classification is done by Hidden Markov Model (HMM).

3.2 REVIEW OF HIDDEN MARKOV MODELS

Penny et al (1998) discovered that HMMs are capable of detecting non-stationary changes and are suitable for speech signal analysis. However they concluded that operating HMMs on AR coefficients is flawed and the state and state transitions in HMM model are estimated incorrectly due to the windowing procedure used in AR models. Graja & Boucher (2003) described that spatio-temporal data reduction and soft computing techniques like the Bayesian statistics can also be used in order to reduce the computational loads. Works on EEG classification usually apply HMMs to the time changing feature vectors extracted by an AR model or by some other digital signal processing techniques. Huang et al (1996) used the mean frequency features, calculated from FFT spectrum, for detecting the arousal state changes. But application of FFT for signals that have high probabilistic variation may not produce good
results. Obermaier, Guger, and Pfurtscheller (1999) compared LDA (linear discrimination analysis) and HMMs on band pass-filtered feature vectors and experiment with the structure parameters of HMMs. Later Graja and Boucher (2003) have investigated the use of hidden Markov tree models for segmenting ECG signals encoded with the discrete wavelet transform.

More recently Easwarmoorthy et al (2010) focused on the method which indicated the state of illness of epileptic patient from EEG recording. Analysis based wavelet decomposition through DWT was performed. Though DWT is a good tool for signal analysis it does not produce effective results as that of HMM. Leon, D, Iasemidis et al (2003) proposed Early Seizure Detection in which involved the intracranial measurement of EEG signals. However, it is considered as a method of complexity as the electrodes could be placed only in the course of an operation. Hidden Markov Model (HMM) is one of the most popular techniques in recognition tasks. HMM models are very rich in mathematical, theoretical basis for use in wide range of application (Choudrey et al 2003). HMM based recognition systems that use a Vector Quantization (VQ) front-end process constitute very useful and inexpensive solutions.

3.3 VECTOR QUANTIZATION (VQ)

Vector Quantization is a method of automatically partitioning a feature space into different clusters based on training data (Makhoul et al 1985). It has been used to generate features for pattern classification problems. Computer vision has used Vector Quantization (VQ) as a tool for constructing higher level data representation. Vector Quantization offers speed-ups in situations where arithmetic accuracy is not crucial. Kekre et al (2010) used Vector Quantization as a technique for approximate nearest neighbour search. They powerfully estimated the Euclidean distance between two vectors from their codes. It works by dividing a large set of points (vectors) into groups having approximately the same number of points closest
to them. Each group is represented by its centroid point, as in k-means and some other clustering algorithms. The density matching property of Vector Quantization is powerful, especially for identifying the density of large and high-dimensional data. Since, the data points are represented by the index of their closest centroid, commonly occurring data have low error, and rare data have high error. This can be used for data reduction (mapping a large number of feature points to a much smaller number of clusters).

Vector Quantization (VQ) is a procedure that encodes a vector of input (e.g. a segment of waveform or a parameter vector that represents the segment spectrum) into an integer (index) that is associated with an entry of a collection (codebook) of reproduction vectors. The reproduction vector chosen is the one that is closest to the input vector in a specified distortion sense. The performance of the Vector Quantizer, however, depends on whether the set of reproduction vectors, which are often called code words, is properly chosen such that the incurred distortion to be minimum on an average (Rabiner et al 1993). It maps K-dimensional vector space to a finite set CB= \{C1, C2, C3, ..., CN\}. The set CB is called as codebook consisting of N number of code vectors and code vector Ci = \{ci1, ci2, ci3, ..., cik\} is of dimension K. Good code book design leads to high accuracy in pattern classification.

3.3.1 Elements of a Vector Quantization

To build a VQ codebook and implement a VQ analysis procedure, the following are needed

i. A large set of analysis of dimensional component vectors, v1, v2, ..., vL, which form a training set. The training set is used to create the “optimal” set of codebook vectors for representing the component variability observed in training set. If the size of the VQ codebook is denoted as M=2^B vectors, then \( L \gg M \) so as to able to find the best set of \( M \) codebook vectors in a
robust manner. L should be at least 10M in order to train a VQ codebook that works reasonably well.

ii. A measure of similarity, or distance, between a pair of dimensional component vectors so as to cluster the training set vectors as well as to associate or classify arbitrary component vectors into unique codebook entries.

iii. A Centroid Computation Procedure: On the basis of the partitioning that classifies the L training vectors into M clusters, M codebook vectors are chosen as the centroid of each of the M clusters.

iv. A classification procedure for arbitrary EEG signal analysis in dimensional component vectors that chooses the codebook vector closest to the input vector and uses the codebook index as the resulting component representation, this is often referred as the nearest neighbor labeling or optimal encoding procedure. The classification procedure is essentially a Quantizer that accepts, as input, a dimensional component vector and provides, as output, the codebook index of the codebook vector that best matches the input. Figure 3.1 shows the block diagram of the basic VQ training and classification structure.

![Diagram](image-url)

**Figure 3.1 Block Diagram of VQ Training and Classification Structure**
3.3.2 VQ Training Set

To properly train the codebook, the training set vectors should span the anticipated range of the following:

i. Patients, including ranges in age, gender epileptic history, spike rate, risk levels, and other variables

ii. Transducers and transmission systems, including wideband EEG amplifier system, audio handsets, wideband channel, and other recording devices

iii. EEG signal units including specific recognition tasks (e.g. epileptic spikes) and audio and photo stimulus.

The more narrowly focused the training set (i.e., dimensionally reduced EEG signals), the smaller the quantization error in representing the dimensional component information with a fixed-size codebook. However, for applicability to a wide range, the training set should be as broad, in each of the above dimensions, as possible.

3.3.3 The Distance Measure

Euclidean distance measure is used for comparing component vectors \( v_i \) and \( v_j \) and is given by the following formula

\[
d(v_i, v_j) = \sqrt{\sum_{i=1}^{n} (v_i - v_j)^2}
\]  

(3.1)

3.3.4 Clustering the Training Vectors

K-means clustering algorithm is used for clustering \( L \) training vectors into a set of \( M \) codebook vectors. The algorithm is as follows
i. Initialization: Arbitrarily choose M vectors from L training
vectors (typically $M=2^B$) as the initial set of code words in the
codebook.

ii. Search: For each training vector, find the code word in the
codebook that is closest, and assign that vector to the
Corresponding cell.

iii. Centroid Update: Update the code word in each cell using the
centroid of the training vectors assigned to that cell.

iv. Repeat steps ii and iii until the average distance falls below a
preset threshold.

Although the above iterative procedure works well, it is
advantageous to design a M-vector codebook in stages-i.e., by first designing
a 1-vector codebook, and then using a splitting technique on the code words to
initialize the search for a 2-vector codebook, and continuing the splitting
process until the desired M-vector codebook is obtained. This procedure is
called the binary split algorithm and is formally implemented by the
following procedure:

1) Design 1-vector codebook; this is the centroid of the entire set
of training vectors. (hence, no iteration is required here).

2) Double the size of the codebook by splitting each codebook $y_n$
according to the rule:

$$y_n^* = y_n(1+\epsilon)$$  \hspace{1cm} (3.2)

$$y_n^- = y_n(1-\epsilon)$$  \hspace{1cm} (3.3)

where $1 \leq n \leq N$, and $\epsilon$ is a splitting parameter ($0.01 \leq \epsilon \leq 0.05$).
3) Use K-means algorithm to get the best set of centroids for the split codebook.

4) Repeat steps 2 and 3 until desired codebook of size M is obtained.

Figure 3.2 shows, in a flowchart, the detailed steps of the binary split VQ codebook generation technique.

Figure 3.2 Flowchart of Binary Split Codebook Generation Algorithm
3.3.5 Vector Classification Procedure

The classification procedure for dimensional component arbitrary vectors is basically a full search through the codebook to find the best match. If the code book vectors of an M-vector codebook are denoted as $y_m, 1 \leq m \leq M$, and the spectral vector to be classified as $v$, then the index, $m^*$, of the best codebook entry is

$$n^* = \arg\min_{1 \leq n \leq N} d(v, y_n)$$  \hspace{1cm} (3.4)

3.3.6 Advantages of Vector Quantization

i. Reduced storage for dimensionally reduced EEG signal analysis information

ii. Reduced computation for determining similarity of component analysis vectors in epilepsy risk detection a major component of the computation is the determination of component similarity between pair of vectors. Based on the VQ representation, this component similarity computation is often reduced to a table look-up of similarities between pairs of codebook vectors.

iii. Discrete representation of dimensionally reduced EEG components. By associating a risk level label with each codebook vector, the process of choosing a best codebook vector to represent a given epilepsy risk level vector becomes equivalent to assigning a risk level label to each dimensionally reduced segments of EEG signal.
Table 3.1  Vector Quantized Values for Dimensionally Reduced EEG Signal in Patient 1

<table>
<thead>
<tr>
<th>Sl.No</th>
<th>Vector Quantized Value of SVD</th>
<th>Vector Quantized Value of PCA</th>
<th>Vector Quantized Value of ICA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.8218</td>
<td>0.2586</td>
<td>0.2019</td>
</tr>
<tr>
<td>2</td>
<td>0.3071</td>
<td>0.1032</td>
<td>0.0177</td>
</tr>
<tr>
<td>3</td>
<td>0.0401</td>
<td>0.0921</td>
<td>0.0522</td>
</tr>
<tr>
<td>4</td>
<td>0.0249</td>
<td>0.0015</td>
<td>0.0212</td>
</tr>
<tr>
<td>5</td>
<td>0.0112</td>
<td>0.000725</td>
<td>0.009</td>
</tr>
<tr>
<td>6</td>
<td>0.0091</td>
<td>0.000163</td>
<td>0.2019</td>
</tr>
<tr>
<td>7</td>
<td>0.0058</td>
<td>0.0001073</td>
<td>0.0177</td>
</tr>
<tr>
<td>8</td>
<td>0.00043</td>
<td>0.000026</td>
<td>0.0522</td>
</tr>
</tbody>
</table>

A sample of Vector Quantized Values for Dimensionally Reduced EEG signal in patient 1 is shown in Table 3.1. Vector Quantized values shown in table 3.1 exhibits the possible onset and offset boundaries of dimensionally reduced (SVD, PCA and ICA) EEG signals. The epilepsy risk levels for EEG segments are represented as components labels and entries in vector code book. They are further processed by HMM classifier to obtain singleton classification of epilepsy.

3.4  HIDDEN MARKOV MODEL

Hidden Markov Model (HMM) is a well-known and widely used statistical method of characterizing the component properties of the EEG segments. HMM is a doubly embedded stochastic process with an underlying stochastic process that is not directly observable but can be observed only through another set of stochastic processes that produce the sequence of observations. Discrete Hidden Markov Model is a type of HMM that models EEG segment components based on VQ technique to produce the
observations. It is an efficient and reliable technique in which the sub 
segments, epileptic spikes, and epilepsy risk levels can be modelled easily. 
Hidden Markov Models (HMM) are broadly used in Automatic epilepsy 
detection. The dynamic properties of EEG segments and especially 
interleaving epileptic fragments can be modeled with them quite well.

3.4.1 Elements of HMM

The elements of the HMM are,

1. N, the number of states in the model. Although the states are 
   hidden, for many practical applications there is often some 
   physical significance attached to the states or to sets of states 
   of the model.

2. M, the number of distinct observation symbols per state- i.e., 
   the discrete component size. The observation symbols 
   correspond to the physical output of the system being 
   modelled.

3. The state-transition probability distribution A= {a_{ij}} where

\[ a_{ij} = P[q_{t+1} = j \mid q_t = i], 1 \leqslant i, j \leqslant N \]  \hspace{1cm} (3.5)

For the special case in which any state can reach any other 
state in a single step \( a_{ij} \geq 0 \) for all \( i, j \). For other types of 
HMMs, \( a_{ij} = 0 \) for one or more \( i, j \) pairs.

4. The observation symbol probability distribution, \( B = \{b_j(k)\} \),
in which

\[ b_j(k) = P[o_t = v_k \mid q_t = j], 1 \leqslant k \leqslant M, \]  \hspace{1cm} (3.6)

defines the symbol distribution in state \( j, j=1,2,3,..N \).
5. The initial state distribution $\pi = \{ \pi_i \}$ in which

$$\pi = P[q_1 = i], 1 \leq i \leq N. \quad (3.7)$$

A complete specification of an HMM requires specification of two model parameters, $N$ and $M$, specification of observation symbols, and the specification of the three sets of probability measures $A$, $B$, and $\pi$. Complete parameter set of the model is given by the notation

$$\lambda = (A, B, \pi) \quad (3.8)$$

### 3.4.2 HMM Constraints

HMM could have different constraints depending on the nature of the problem that wanted to be modelled. The main constraints needed in the implementation of epilepsy risk level classification can be summarized in the following assumptions:

1) **First order Markov chain**

In this assumption the probability of transition to a state depends only on the current state

$$P(q_{t+1} = S_1, q_{t-1} = S_k; Q_{t-2})$$

$$= S_1, \ldots, q_{t-1} = S_z \approx P(q_{t+1} = S_1 / q_t = S_1) \quad (3.9)$$

2) **Stationary States’ Transition**

This assumption testifies that the states transition are time independent, and accordingly

$$a_{ij} = P(q_{t+1} = S_j / q_t = S_i) \ for \ all \ t \quad (3.10)$$
3) **Observations Independence**

This assumption presumes that the observations come out within certain state depend only on the underlying Markov chain of the states, without considering the effect of the occurrence of the other observations. Although this assumption is a poor one and deviates from reality but it works fine in modelling EEG signal.

This assumption implies that:

\[
P\left(\frac{O_t, q_{t-2}, \ldots, q_{t-p}, q_{t-1}, q_{t-2}, \ldots, q_{t-p}}{O_{t-1}}\right) = P(O_t / q_t, q_{t-1}, q_{t-2}, \ldots, q_{t-p})
\]  

(3.11)

where \( p \) represents the considered history of the observation sequence.

Then

\[
b_j(O_t) = P(O_t / q_t = j)
\]

(3.12)

4) **Left-Right Topology Constraint**

\[
\alpha_{ij} = 0 \quad \text{for all} \quad j > i + 2 \quad \text{and} \quad j < i
\]

(3.13)

\[
\pi_i = P(q_1 = S_i) = \begin{cases} 
1 & \text{for } i = 1 \\
0 & \text{for } 1 < i \leq N
\end{cases}
\]

(3.14)

5) **Probability Constraints**

\[
\sum_{j=1}^{N} a_{ij} = 1
\]

(3.15)

\[
\sum_{j=1}^{N} \pi_{ij} = 1
\]

(3.16)

\[
\sum_{j=1}^{N} b_i(O_t) = 1
\]

(3.17)
3.4.3 Three Main Cases of HMM

There are three main cases to be dealt with to formulate a successful HMM. These are:

Case 1: Evaluation

Given:

1. A model $\lambda = (A, B, \pi)$ ready to be used.
2. Testing observation sequence $O = O_1, O_2, O_3, \ldots, O_{T-1}, O_T$.

Action: Compute $P(O/\lambda)$; the probability of the observation sequence given the model.

Case 2: Decoding

Given:

1. A model $\lambda = (A, B, \pi)$ ready to be used.
2. Testing or training observation sequence $O = O_1, O_2, O_3, \ldots, O_{T-1}, O_T$.

Action: Track the optimum state sequence $Q = q_1, q_2, q_3, \ldots, q_{T-1}, q_T$ that most likely produce the given observations, using the given model.

Case 3: Training

Given:

1. A model $\lambda = (A, B, \pi)$ ready to be used.
2. Training observation sequence
\[ O^k = C_1^k, C_2^k, C_3^k, \ldots, C_{T-1}^k, O_T^k \]

where \( k \) is the number of examples for training the model.

**Action:** Tune the model parameters to maximize \( P(O/\lambda) \).

Case 1 is an evaluation procedure to find the probability of producing given observation \( O \) by a given model \( \lambda \). This is used to find out the best model among many which produces the given observation.

Case 2 is a decoding procedure to detect or unhide the state sequence of a given observation. The observations could be training examples if the behaviour of each state from different aspects to be studied, such as states’ duration or spectral characteristics of each state. Some techniques utilize the state duration in their evaluation procedure and in this case the observation will be the test example to detect the states duration.

Case 3 is the training procedure to optimize the model parameters to obtain the best model that represents certain set of observations belonging to one group epilepsy risk level entity. The way is paved now to tackle an important goal of our task, namely derivation of the mathematical formulas to the three previous cases.

### 3.4.4 Forward Procedure

Initially consider a new forward probability variable \( \alpha_t(i) \), at instant \( t \) and state \( i \), has the following formula:

\[
\alpha_t(i) = P(O_1, O_2, O_3, \ldots, O_t, q_t = S_t/\lambda) \quad (3.18)
\]

This probability function could be solved for \( N \) states and \( T \) observations iteratively:
1) Initialization

\[ \alpha_t(i) = \pi_i b_i(O_j) \]  (3.19)

2) Induction

\[ a_{t+1}(j) = \left[ \sum_{i=1}^{N} \alpha_t(i) \cdot \alpha_{ij} \right] b_j(O_{t+1}) \quad 1 \leq t \leq T - 1 \]

\[ 1 \leq j \leq N \]  (3.20)

Figure 3.3 shows the induction step graphically. It is clear from this figure how state \( S_j \) at instant \( t+1 \) reached from \( N \) possible states at instant \( t \).

3) Termination

\[ P(O/\lambda) = \sum_{t=1}^{N} \alpha_T(t) \]  (3.21)

This stage is just a sum of all the values of the probability function \( \alpha_T(t) \) over all the states at instant \( T \). This sum will represent the probability of the given observations to be driven from the given model.

![Figure 3.3 Forward Probability Function Representation](image)
3.4.5 Backward Procedure

This procedure is similar to the forward procedure but it takes into consideration the state flow as if in backward direction from the last observation entity, instant $T$, till the first one, instant 1. That means that the access to any state will be from the states that are coming just after that state in time is shown in Figure 3.3. The backward probability function $\beta_t(i)$ is defined as:

$$\beta_t(i) = P(O_{t+1}, O_{t+2}, \ldots, O_T | q_t = S_i; \lambda)$$  \hspace{1cm} (3.22)

In analogy to the forward procedure for $\beta_t(i)$ it can be solved in the following two steps:

1) Initialization

$$\beta_T(i) = 1, \hspace{0.5cm} 1 \leq i \leq N$$  \hspace{1cm} (3.23)

These initial values for $\beta$’s of all states at instant $T$ is arbitrarily selected.

2) Induction

$$\beta_t(i) = \sum_{j=1}^{N} a_i b_j (O_{t+1}) \beta_{t+1}(j), \hspace{0.5cm} t = T - 1, T - 2, \ldots, 1 \hspace{0.5cm} 1 \leq N$$  \hspace{1cm} (3.24)

Equation (3.24) can be well understood with help of Figure 3.4. it shows that in order to have state $S_i$ at time $t$, and to account for the observation sequence from time $t+1$ on, all possible states $S_j$ at time $t+1$ are considered accounting for the transition from $S_i$ to $S_j$ as well as the observation $O_{t+1}$ in state $j$. 
Figure 3.4 Backward probability function representation

The probability function of the model $P(O/\lambda)$ can be computed from both $\alpha$ and $\beta$ functions. Figure 3.5 demonstrates this computation graphically. At instant $t$, the event of being in state $q_i$ and moving to state $q_j$ at instant $t+1$ is calculated by $\alpha_t(i)$ accounts for the path termination in state $i$. The transition to state $j$ is weighted by the product $\alpha_t(i) \times \beta_{t+1}(j)$. At instant $t+1$ the event of observation sequence to the instant $T$ starting from state $j$, while being at state $j$ during instant $t$, is represented by the backward probability function $\beta_{t+1}(j)$.

Then $P(O/\lambda)$ is directly concluded to be:

\begin{align*}
(3.25) \\
(3.26)
\end{align*}
3.4.6 Baum Welch Algorithm

The most difficult, problem of HMMs is to determine a method to adjust the model parameters \((A, B, \pi)\) to satisfy a certain optimization criterion. There is no known way to analytically solve for the model parameter set that maximizes the probability of the observation sequence in a closed form. A model \(\lambda=(A, B, \pi)\) is chosen such that its likelihood, \(P(O/\lambda)\), is locally maximized using an iterative procedure called Baum-Welch method.

The procedure for re-estimation of HMM parameters is as follows, \(\xi_t(i,j)\), the probability of being in state \(i\) at time \(t\), and state \(j\) at time \(t+1\), given the model and the observation sequence is

\[
\xi_t(i,j) = P(q_t = i, q_{t+1} = j | O, \lambda)
\]  

(3.27)
\( \xi_t(i,j) \) in terms of forward and backward variables is given by,

\[
\xi_t(i,j) = \frac{P(q_t=i,q_{t+1}=j|\lambda)}{P(q_t=i|\lambda)}
\]

(3.28)

\[
\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}b_j(c_{t+1}|b_{t+1}(j))}{P(\theta|\lambda)}
\]

(3.29)

\[
\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}b_j(c_{t+1}|b_{t+1}(j))}{\sum_{k=1}^{N_p} \sum_{p=1}^{N} \alpha_t(k)c_{tp}k_{tp}(c_{t+1}|b_{t+1}(p))}
\]

(3.30)

\( \gamma_t(i) \) defined as the probability of being in state \( i \) at time \( t \), given the entire observation sequence and the model is related to \( \gamma_t(i) \) to \( \xi_t(i,j) \) by summing over \( j \), giving

\[
\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j)
\]

(3.31)

Summation of \( \gamma_t(i) \) over the time index \( t \), gives the expected number of transitions made from state \( i \). Similarly, summation of \( \xi_t(i,j) \) over \( t \) gives the expected number of transitions from state \( i \) to state \( j \).

\[
\sum_{t=1}^{T} \gamma_t(i) = \text{expected number of times in state } i \text{ for } O
\]

(3.32)

\[
\sum_{t=1}^{T} \gamma_t(i,j) = \text{expected number of transitions from state } i \text{ to state } j \text{ for } O
\]

(3.33)

From the behaviour of \( \gamma_t(i) \) and \( \xi_t(i,j) \), the following re-estimations of the model parameters could be deduced

\( \hat{\pi}_i = \text{expected number of instants the starting state is } i \)

(3.35)

\[
\bar{\pi}_j = \gamma_t(j)
\]

(3.36)
\[ \bar{a}_{ij} = \frac{\text{expected number of transition from state } i \text{ to state } j}{\text{expected number of transition from state } i} \]  \hspace{1cm} (3.37)

\[ \bar{c}_{ij} = \frac{\sum_{t=1}^{T-1} c_t(i,j)}{\sum_{t=1}^{T-1} c_t(i,i)} \]  \hspace{1cm} (3.38)

\[ \bar{b}_j(k) = \frac{\text{expected number of times in state } j \text{ and observing symbol } v_k}{\text{expected number of times in state } j} \]  \hspace{1cm} (3.39)

\[ \bar{b}_j(k) = \frac{\sum_{t=1}^{T-1} c_t(v_k)}{\sum_{t=1}^{T-1} c_t(v_k)} \text{ such that } O_t = v_k \]  \hspace{1cm} (3.40)

After the re-estimation of the model parameters there is another model \( \hat{\lambda} \) which is more likely, than model \( \lambda \), producing observation sequence \( O \). This means that

\[ P(O/\hat{\lambda}) > P(O/\lambda) \]  \hspace{1cm} (3.41)

This process of re-estimation can be continued till no improvement in \( P(O/\lambda) \) is reached.

### 3.4.7 Scaling

The scaling factor is a major issue in implementing the HMM because of the underflow that may easily occur when calculating the probability function \( P(O/\lambda) \). This is due to the long sequence of multiplications of less than one values probability functions.

The straightforward technique of scaling is started by defining the scaling coefficient \( c(t) \).
\[ c(t) = \frac{1}{\sum_{i=1}^{N} \alpha_i \epsilon_i} \]  

(3.42)

Now let us compute \( \alpha_i(t) \) from Equation 3.20 and then multiply it by \( c(t) \). This will lead to:

\[ \frac{1}{\sum_{i=1}^{N} \frac{1}{\beta_i} \alpha_i(t) \epsilon_i} \times \frac{1}{\sum_{i=1}^{N} \frac{1}{\beta_i} \alpha_i \epsilon_i} \]  

(3.43)

where

\[ T_C = \prod_{i=1}^{T} t_T \]  

(3.44)

and

\[ t_{-1} = \prod_{i=1}^{T} t_{-1} \]  

(3.45)

The denominator of Equation (3.43) consist of the product \( t_{-1} t_{-1} = \prod_{i=1}^{T} \tau_T \) which can be factored out and retained the original equation (3.36). This scaling technique can also be applied successfully to Equation (3.37). The scaling coefficients can be used to find \( \log O(\lambda/\gamma) \) by the following method:

\[ \text{Consider that there is } c_i \text{ for } t=1,2,3,\ldots, T \text{ and } T_C \text{ is obtained from the equation (3.44), then from the equation (3.42)} \]

\[ T_C = \prod_{i=1}^{T} t_T = \left[ \sum_{i=1}^{T} T_T \right]^{1C} \]  

(3.46)

sing [21] will have:

\[ T_C = \prod_{i=1}^{T} \frac{t_T}{\gamma \lambda} \]  

(3.47)

Take the log of the last two terms
\[ \log(\prod_{t=1}^{T} C_t) = -\log[ P(O/\lambda)] \]  

(3.48)

By using log properties it can be obtained

\[ \log[P(O/\lambda)] = -\sum_{t=1}^{T}\log( C_t) \]  

(3.49)

Equation (3.49) shows that \( \log P(O/\lambda) \) can be computed but not \( P(O/\lambda) \) as the later will be out of the dynamic range of the computer.

**3.4.8 Training and Testing Procedures for the Selection of Optimal Architecture in HMM Model**

The primary aim of developing a HMM is to generalize the detection of Epilepsy risk levels from dimensionally reduced EEG segments. The training and testing procedure for epilepsy risk level classification using HMM is displayed in Figure 3.6.

**Figure 3.6 Training and Testing Procedure in Epilepsy Risk Level Classifications from EEG Signals using HMMs**
In this research, twenty patients are used in nine groups with three types of Dimensionality Reduction (SVD, PCA and ICA) Techniques. Therefore, twenty seven HMM models are selected. The training process is controlled by monitoring of the Mean Square Error (MSE) which is defined as (Vitabile et al 2004, Drazen et al 2004)

\[ \text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (O_i - T_j)^2 \]  \hspace{1cm} (3.50)

where \( O_i \) is the observed value at time \( i \), \( T_j \) is the target value at model \( j \); \( j=1-9 \), and \( N \) is the total number of observations per epoch in our case it is 16.

This research uses all EEG segmented data, both for training as well as testing for HMM classifiers. The training was progressed regressively and the MSE Values of HMMs were decreased to minimum. HMM classifiers are trained with zero training error of MSE. Therefore, the testing procedure was initiated and obtained average MSE values for twenty patients which are shown in Table 3.2. Based on the estimation of testing MSE in HMM classifier as in Table 3.2 better HMM classifier will be identified. The value of MSE indicates the training and testing intensity of HMM classifier. Lower the testing MSE will be the better HMM classifier. HMM classifier response under noisy environment also exhibits by MSE parameter.

Compared to the average MSE of 0.00339 and 0.00318 for PCA and ICA Dimensionality Reduction method along with HMM classifier, SVD Dimensionality Reduction method followed by HMM classifier scores high with minimum average MSE of 0.00277. Good performance shown by SVD indicates good training and testing of data for HMM classifier. One problem with the PCA is that it is a kind of unsupervised learning procedure and does not consider the correlation between target outputs and input features. In
addition, PCA is a linear dimension reduction technique. Hence, PCA performs poorly in terms of MSE when compared to SVD and ICA methods.

Table 3.2 Estimation of Mean Square Error (MSE) in HMM Classifier under Testing Condition

<table>
<thead>
<tr>
<th>Patient</th>
<th>Target Value</th>
<th>SVD</th>
<th>PCA</th>
<th>ICA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Observed Value</td>
<td>MSE</td>
<td>Observed Value</td>
<td>MSE</td>
</tr>
<tr>
<td>1</td>
<td>0.85</td>
<td>0.79</td>
<td>0.0036</td>
<td>0.77</td>
</tr>
<tr>
<td>2</td>
<td>0.85</td>
<td>0.81</td>
<td>0.0016</td>
<td>0.82</td>
</tr>
<tr>
<td>3</td>
<td>0.65</td>
<td>0.6</td>
<td>0.0025</td>
<td>0.59</td>
</tr>
<tr>
<td>4</td>
<td>0.45</td>
<td>0.39</td>
<td>0.0036</td>
<td>0.41</td>
</tr>
<tr>
<td>5</td>
<td>0.25</td>
<td>0.19</td>
<td>0.0036</td>
<td>0.16</td>
</tr>
<tr>
<td>6</td>
<td>0.35</td>
<td>0.32</td>
<td>0.0009</td>
<td>0.29</td>
</tr>
<tr>
<td>7</td>
<td>0.65</td>
<td>0.61</td>
<td>0.0016</td>
<td>0.6</td>
</tr>
<tr>
<td>8</td>
<td>0.25</td>
<td>0.2</td>
<td>0.0025</td>
<td>0.2</td>
</tr>
<tr>
<td>9</td>
<td>0.05</td>
<td>0.02</td>
<td>0.0009</td>
<td>0.02874</td>
</tr>
<tr>
<td>10</td>
<td>0.1</td>
<td>0.07</td>
<td>0.0009</td>
<td>0.06</td>
</tr>
<tr>
<td>11</td>
<td>0.85</td>
<td>0.78</td>
<td>0.0049</td>
<td>0.8</td>
</tr>
<tr>
<td>12</td>
<td>0.35</td>
<td>0.28</td>
<td>0.0049</td>
<td>0.28</td>
</tr>
<tr>
<td>13</td>
<td>0.45</td>
<td>0.41</td>
<td>0.0016</td>
<td>0.39</td>
</tr>
<tr>
<td>14</td>
<td>0.65</td>
<td>0.61</td>
<td>0.0016</td>
<td>0.61</td>
</tr>
<tr>
<td>15</td>
<td>0.2</td>
<td>0.14</td>
<td>0.0036</td>
<td>0.14</td>
</tr>
<tr>
<td>16</td>
<td>0.15</td>
<td>0.1</td>
<td>0.0025</td>
<td>0.11</td>
</tr>
<tr>
<td>17</td>
<td>0.2</td>
<td>0.13</td>
<td>0.0049</td>
<td>0.14</td>
</tr>
<tr>
<td>18</td>
<td>0.65</td>
<td>0.57</td>
<td>0.0064</td>
<td>0.58</td>
</tr>
<tr>
<td>19</td>
<td>0.25</td>
<td>0.21</td>
<td>0.0016</td>
<td>0.19</td>
</tr>
<tr>
<td>20</td>
<td>0.85</td>
<td>0.81</td>
<td>0.0016</td>
<td>0.82</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>0.00277</td>
<td>-</td>
<td>0.00339</td>
</tr>
</tbody>
</table>
3.5 RESULTS AND DISCUSSION

To study the relative performance of HMM classifiers, we measure four parameters such as sensitivity, specificity, average detection and Good Detection rate (GDR) are measured. These parameters are calculated for each set of twenty patients and compared.

3.5.1 Performance Measures

Performance measure of HMM classifier is associated with the terminologies like Perfect Classification (PC), Missed Classification (MC) and False Alarm (FA). In our case, perfect classification represents when the physicians and HMM classifier agree with the epilepsy risk level. Missed classification represents a true negative of HMM classifier in reference to the physician and shows High level as Low level. False alarm represents a false positive of HMM classifier in reference to the physician and shows Low level as High level.

Terminology is also an important issue when performance of methods is compared. It is important to differentiate between the two terms of risk level prediction and risk level predictability. Predictability is necessary but not a sufficient condition for risk level prediction. Risk level predictability has to do with the sensitivity, whereas risk level prediction with both the sensitivity and specificity of the proposed and prospective methods. Hence, it is necessary to present the sensitivity and specificity of epilepsy risk levels classifier with HMM method. These two precursors are defined by Celement et al (2003),

\[
\text{Sensitivity} = \frac{\text{PC}}{\text{PC} + \text{FA}} \times 100
\]

(3.51)
Specificity = \frac{PC}{PC + MC} \times 100 \quad (3.51)

Clement et al (2003) also defines Average detection of the classifier as

Average Detection = 0.5(\text{Sensitivity} + \text{Specificity}) \times 100 \quad (3.52)

Good detection Rate: The most important criterion of a detector. This represents the ability of a detector in successful detection as defined by RezaSang (2006)

\[ GDR = \left( \frac{PC-MC}{PC+MC} \right) \times 100 \quad (3.53) \]

A sample of performance measure in HMM classifiers for SVD, PCA and ICA Dimensionality Reduction methods are shown in Table 3.3, 3.4 and 3.5. From table 3.3 it was found that the performance measures of HMM classifiers with SVD dimensionally reduction technique is attained at higher level and exhibits better sensitivity, good GDR, better average detection and lower specificity. From Table 3.4 it was observed that the performance measures of HMM classifiers with PCA dimensionally reduction technique is placed at lower level of all methods with low GDR, good sensitivity, better average detection of 94% and poor specificity. From Table 3.5 it was identified that the performance measures of HMM classifiers with ICA dimensionally reduction technique is placed at middle level among the methods with good GDR, good sensitivity, better average and lower specificity of 90%. 
<table>
<thead>
<tr>
<th>Patient</th>
<th>Perfect Classification (%)</th>
<th>Missed Classification (%)</th>
<th>False Alarm (%)</th>
<th>Sensitivity (%)</th>
<th>Specificity (%)</th>
<th>Average Detection (%)</th>
<th>Good Detection Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>86</td>
<td>14</td>
<td>0</td>
<td>100</td>
<td>86</td>
<td>93</td>
<td>83.7</td>
</tr>
<tr>
<td>2</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>3</td>
<td>94</td>
<td>6</td>
<td>0</td>
<td>100</td>
<td>94</td>
<td>97</td>
<td>93.6</td>
</tr>
<tr>
<td>4</td>
<td>86</td>
<td>14</td>
<td>0</td>
<td>100</td>
<td>86</td>
<td>93</td>
<td>83.7</td>
</tr>
<tr>
<td>5</td>
<td>86</td>
<td>14</td>
<td>0</td>
<td>100</td>
<td>86</td>
<td>93</td>
<td>83.7</td>
</tr>
<tr>
<td>6</td>
<td>94</td>
<td>6</td>
<td>0</td>
<td>100</td>
<td>94</td>
<td>97</td>
<td>93.6</td>
</tr>
<tr>
<td>7</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>8</td>
<td>87</td>
<td>13</td>
<td>0</td>
<td>100</td>
<td>87</td>
<td>93.5</td>
<td>85.1</td>
</tr>
<tr>
<td>9</td>
<td>90</td>
<td>10</td>
<td>0</td>
<td>100</td>
<td>90</td>
<td>95</td>
<td>88.8</td>
</tr>
<tr>
<td>10</td>
<td>90</td>
<td>10</td>
<td>0</td>
<td>100</td>
<td>90</td>
<td>95</td>
<td>88.8</td>
</tr>
<tr>
<td>11</td>
<td>85</td>
<td>15</td>
<td>0</td>
<td>100</td>
<td>85</td>
<td>92.5</td>
<td>82.3</td>
</tr>
<tr>
<td>12</td>
<td>85</td>
<td>15</td>
<td>0</td>
<td>100</td>
<td>85</td>
<td>92.5</td>
<td>82.3</td>
</tr>
<tr>
<td>13</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>14</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>15</td>
<td>85</td>
<td>15</td>
<td>0</td>
<td>100</td>
<td>85</td>
<td>92.5</td>
<td>82.4</td>
</tr>
<tr>
<td>16</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>17</td>
<td>83</td>
<td>17</td>
<td>0</td>
<td>100</td>
<td>83</td>
<td>91.5</td>
<td>79.5</td>
</tr>
<tr>
<td>18</td>
<td>83</td>
<td>17</td>
<td>0</td>
<td>100</td>
<td>83</td>
<td>91.5</td>
<td>79.5</td>
</tr>
<tr>
<td>19</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>20</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>Average</td>
<td>89.95</td>
<td>10.05</td>
<td>0</td>
<td>100</td>
<td>89.95</td>
<td>94.97</td>
<td>88.52</td>
</tr>
</tbody>
</table>
### Table 3.4 Performance Measures of HMM Classifier with PCA Dimensionality Reduction

<table>
<thead>
<tr>
<th>Patient</th>
<th>Perfect Classification (%)</th>
<th>Missed Classification (%)</th>
<th>False Alarm (%)</th>
<th>Sensitivity (%)</th>
<th>Specificity (%)</th>
<th>Average Detection (%)</th>
<th>Good Detection Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>85</td>
<td>15</td>
<td>0</td>
<td>100</td>
<td>85</td>
<td>92.5</td>
<td>82.3</td>
</tr>
<tr>
<td>2</td>
<td>85</td>
<td>15</td>
<td>0</td>
<td>100</td>
<td>85</td>
<td>92.5</td>
<td>82.4</td>
</tr>
<tr>
<td>3</td>
<td>92</td>
<td>8</td>
<td>0</td>
<td>100</td>
<td>92</td>
<td>96</td>
<td>91.3</td>
</tr>
<tr>
<td>4</td>
<td>88</td>
<td>12</td>
<td>0</td>
<td>100</td>
<td>88</td>
<td>94</td>
<td>86.4</td>
</tr>
<tr>
<td>5</td>
<td>85</td>
<td>15</td>
<td>0</td>
<td>100</td>
<td>85</td>
<td>92.5</td>
<td>82.3</td>
</tr>
<tr>
<td>6</td>
<td>92</td>
<td>8</td>
<td>0</td>
<td>100</td>
<td>92</td>
<td>96</td>
<td>91.3</td>
</tr>
<tr>
<td>7</td>
<td>92</td>
<td>8</td>
<td>0</td>
<td>100</td>
<td>92</td>
<td>96</td>
<td>91.3</td>
</tr>
<tr>
<td>8</td>
<td>92</td>
<td>8</td>
<td>0</td>
<td>100</td>
<td>92</td>
<td>96</td>
<td>91.3</td>
</tr>
<tr>
<td>9</td>
<td>92</td>
<td>8</td>
<td>0</td>
<td>100</td>
<td>92</td>
<td>96</td>
<td>91.3</td>
</tr>
<tr>
<td>10</td>
<td>90</td>
<td>10</td>
<td>0</td>
<td>100</td>
<td>90</td>
<td>95</td>
<td>88.8</td>
</tr>
<tr>
<td>11</td>
<td>86</td>
<td>14</td>
<td>0</td>
<td>100</td>
<td>86</td>
<td>93</td>
<td>83.7</td>
</tr>
<tr>
<td>12</td>
<td>86</td>
<td>14</td>
<td>0</td>
<td>100</td>
<td>86</td>
<td>93</td>
<td>83.7</td>
</tr>
<tr>
<td>13</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>14</td>
<td>92</td>
<td>8</td>
<td>0</td>
<td>100</td>
<td>92</td>
<td>96</td>
<td>91.3</td>
</tr>
<tr>
<td>15</td>
<td>92</td>
<td>8</td>
<td>0</td>
<td>100</td>
<td>92</td>
<td>96</td>
<td>91.3</td>
</tr>
<tr>
<td>16</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>17</td>
<td>82</td>
<td>18</td>
<td>0</td>
<td>100</td>
<td>82</td>
<td>91</td>
<td>78.1</td>
</tr>
<tr>
<td>18</td>
<td>82</td>
<td>18</td>
<td>0</td>
<td>100</td>
<td>82</td>
<td>91</td>
<td>78.1</td>
</tr>
<tr>
<td>19</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>20</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>Average</td>
<td>89.65</td>
<td>10.35</td>
<td>0</td>
<td>100</td>
<td>89.65</td>
<td>94.82</td>
<td>88.19</td>
</tr>
</tbody>
</table>
Table 3.5 Performance Measures of HMM Classifier with ICA Dimensionality Reduction

<table>
<thead>
<tr>
<th>Patient</th>
<th>Perfect Classification (%)</th>
<th>Missed Classification (%)</th>
<th>False Alarm (%)</th>
<th>Sensitivity (%)</th>
<th>Specificity (%)</th>
<th>Average Detection (%)</th>
<th>Good Detection Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>84</td>
<td>16</td>
<td>0</td>
<td>100</td>
<td>84</td>
<td>92</td>
<td>80.9</td>
</tr>
<tr>
<td>2</td>
<td>96</td>
<td>4</td>
<td>0</td>
<td>100</td>
<td>96</td>
<td>98</td>
<td>95.8</td>
</tr>
<tr>
<td>3</td>
<td>92</td>
<td>8</td>
<td>0</td>
<td>100</td>
<td>92</td>
<td>96</td>
<td>91.3</td>
</tr>
<tr>
<td>4</td>
<td>89</td>
<td>11</td>
<td>0</td>
<td>100</td>
<td>89</td>
<td>94.5</td>
<td>87.6</td>
</tr>
<tr>
<td>5</td>
<td>84</td>
<td>16</td>
<td>0</td>
<td>100</td>
<td>84</td>
<td>92</td>
<td>80.9</td>
</tr>
<tr>
<td>6</td>
<td>90</td>
<td>10</td>
<td>0</td>
<td>100</td>
<td>90</td>
<td>95</td>
<td>88.8</td>
</tr>
<tr>
<td>7</td>
<td>94</td>
<td>6</td>
<td>0</td>
<td>100</td>
<td>94</td>
<td>97</td>
<td>93.6</td>
</tr>
<tr>
<td>8</td>
<td>87</td>
<td>13</td>
<td>0</td>
<td>100</td>
<td>87</td>
<td>93.5</td>
<td>85.1</td>
</tr>
<tr>
<td>9</td>
<td>92</td>
<td>8</td>
<td>0</td>
<td>100</td>
<td>92</td>
<td>96</td>
<td>91.3</td>
</tr>
<tr>
<td>10</td>
<td>88</td>
<td>12</td>
<td>0</td>
<td>100</td>
<td>88</td>
<td>94</td>
<td>86.4</td>
</tr>
<tr>
<td>11</td>
<td>88</td>
<td>12</td>
<td>0</td>
<td>100</td>
<td>88</td>
<td>94</td>
<td>86.4</td>
</tr>
<tr>
<td>12</td>
<td>85</td>
<td>15</td>
<td>0</td>
<td>100</td>
<td>85</td>
<td>92.5</td>
<td>82.4</td>
</tr>
<tr>
<td>13</td>
<td>94</td>
<td>6</td>
<td>0</td>
<td>100</td>
<td>94</td>
<td>97</td>
<td>93.6</td>
</tr>
<tr>
<td>14</td>
<td>94</td>
<td>6</td>
<td>0</td>
<td>100</td>
<td>94</td>
<td>97</td>
<td>93.6</td>
</tr>
<tr>
<td>15</td>
<td>85</td>
<td>15</td>
<td>0</td>
<td>100</td>
<td>85</td>
<td>92.5</td>
<td>82.4</td>
</tr>
<tr>
<td>16</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>100</td>
<td>95</td>
<td>97.5</td>
<td>94.7</td>
</tr>
<tr>
<td>17</td>
<td>90</td>
<td>10</td>
<td>0</td>
<td>100</td>
<td>90</td>
<td>95</td>
<td>88.8</td>
</tr>
<tr>
<td>18</td>
<td>85</td>
<td>15</td>
<td>0</td>
<td>100</td>
<td>85</td>
<td>92.5</td>
<td>82.4</td>
</tr>
<tr>
<td>19</td>
<td>90</td>
<td>10</td>
<td>0</td>
<td>100</td>
<td>90</td>
<td>95</td>
<td>88.8</td>
</tr>
<tr>
<td>20</td>
<td>96</td>
<td>4</td>
<td>0</td>
<td>100</td>
<td>96</td>
<td>98</td>
<td>95.8</td>
</tr>
<tr>
<td>Average</td>
<td>89.9</td>
<td>10.1</td>
<td>0</td>
<td>100</td>
<td>89.9</td>
<td>94.95</td>
<td>88.55</td>
</tr>
</tbody>
</table>
Table 3.6 demonstrates the average performance measures of HMM classifiers for all twenty patients. It is found from Table 3.6 that HMM classifier is characterized by 90% perfect classification, Nil False alarm and 10% missed classification. Table 3.6 presents good sensitivity and lower specificity of the system. Average detection of the HMM classifier is placed around 95% and 90% GDR shows the better classification rate under dynamic condition.

Table 3.6 Average Performance Measures in HMM Classifier

<table>
<thead>
<tr>
<th>Parameters (%)</th>
<th>Dimensionality Reduction Techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SVD</td>
</tr>
<tr>
<td>Perfect Classification</td>
<td>89.95</td>
</tr>
<tr>
<td>Missed Classification</td>
<td>10.05</td>
</tr>
<tr>
<td>False Alarm</td>
<td>0.00</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>100.00</td>
</tr>
<tr>
<td>Specificity</td>
<td>89.95</td>
</tr>
<tr>
<td>Average Detection</td>
<td>94.98</td>
</tr>
<tr>
<td>Good Detection Rate (GDR)</td>
<td>88.52</td>
</tr>
</tbody>
</table>

The sensitivity and specificity parameters for twenty epilepsy patients in classification of epilepsy risk levels through HMM classifier for three Dimensionality reduction techniques is shown in Figures 3.7, 3.8 and 3.9. High sensitivity and nil false alarm demonstrates the robustness of HMM classifiers. It is found from Figures 3.7, 3.8 and 3.9 that HMM classifiers suffered from severe missed classification. This leads to poor specificity and under performance of the classifier. HMM classifiers with larger Missed classification are known as high threshold and poor response one.
Figure 3.7 Sensitivity and Specificity Measures of HMM Classifier using SVD Dimensionality Reduction

Figure 3.8 Sensitivity and Specificity Measures of HMM Classifier using PCA Dimensionality Reduction
Figure 3.9  Sensitivity and Specificity Measures of HMM Classifier using ICA Dimensionality Reduction

Figure 3.10 GDR and Average Detection Measures of HMM Classifier
The parameter Good Detection Rate (GDR) shows the performance of the classifier in the noisy environment. Better the GDR will increase the robustness of the classifier. Good Detection Rate (GDR) and Average Detection parameters for twenty epilepsy patients in classification of epilepsy risk levels through HMM classifier for three Dimensionality Reduction Techniques is shown in Figure 3.10. From Figure 3.10 it is found that all HMM classifiers are enriched with classification rate of more than 80%. HMM classifiers suffered from severe missed classification and the same is reflected in the Figure 3.10.

![GDR and Perfect Classification Measures of HMM Classifier](image)

**Figure 3.11 GDR and Perfect Classification Measures of HMM Classifier**

Good Detection Rate (GDR) and Average Detection parameters for twenty epilepsy patients in classification of epilepsy risk levels through HMM classifier for three Dimensionality Reduction Techniques are shown in Figure 3.11. Figure 3.11 exhibits that all HMM classifiers which are enriched with Perfect classification of more than 85%.
3.6 CONCLUSION

The dimensionally reduced features of the classification of epilepsy risk level through HMM classifier from EEG signals are discussed in this chapter. The HMM is specially designed for probabilistic distributions and hence it is considered as a good classifier for EEG signal which has large transitions and probabilistic variation and is used for final risk level classification. The system is trained and tested using twenty patients. HMM classifier demonstrates good sensitivity and total removal of false alarm which subsequently increases the robustness. The missed classification is not fully removed from this type of classifier. Therefore, it is recommended to use neural network models as a comprehensive post processing method to classify the epilepsy risk levels.