CHAPTER 8

COMPRESSION ENTROPY ESTIMATION OF HEART RATE VARIABILITY AND COMPUTATION OF ITS RENORMALIZED ENTROPY

8.1 INTRODUCTION

Klimontovich’s S-theorem offers an approach to compare two different states of energy. According to the assumption that the mean effective energy of these two states has to be equal. The spectral distribution of the reference person is renormalised to an equal energy state. The last step of the procedure is the calculation of the difference of Shannon entropies of both frequency distributions which yields the Renormalized entropy (Lau et al 2007; Niels Wessel et al 2000). The obtained results suggest that renormalised entropy (Wessel et al 1994; Saparin et al 1994) is a suitable method for the detection of patients with high risk for sudden cardiac death.

In the following sections, compression entropy estimation of HRV is given.

LZ77 compression technique is introduced. GZIP compression technique is introduced. BZIP2 compression technique is introduced. Implementation flow charts are given. Estimation of Tachogram and Reference selection are given. Reference tachogram selection is mentioned. Autoregressive modeling is indicated. Burg’s or Maximum Entropy Method
(MEM) algorithm for spectral estimation is described. Principle of calculating renormalized entropy is given.

LZ77 compression technique is a sequential dictionary method. Ziv and Lempel developed an universal algorithm for lossless data compression.

GZIP compression technique (Gailly 1993) is a variant of LZ77 technique i.e., it combines the LZ77 algorithm with a Huffman coding.

BZIP2 compression technique uses Burrow Wheelers Transform (BWT), move to front transform (MTF) and Huffman encoding.

In this chapter entropy of heart rate variability using LZ77 compression technique, GZIP compression technique and BZIP2 compression technique is estimated.

Renormalized entropy clearly indicates all transitions from periodic to chaotic behaviour as well as between different types of chaos, such as band-merging and intermittency (Saparin et al 1994).

### 8.1.1 An Example of LZ77 Compression

The column Step indicates the number of the encoding step. It completes each time the encoding algorithm makes an output.

- The column Pos indicates the coding position. The first character in the input stream has the coding position 1.
- The column Match shows the longest match found in the window.
- The column Char shows the first character in the look ahead buffer after the match.
The column Output presents the output in the format (B,L) C:

- (B,L) is the pointer (P) to the Match. This gives the following instruction to the decoder: "Go back B characters in the window and copy L characters to the output";

- C is the explicit Character.

Input stream for encoding and encoding process:

The input stream for encoding and encoding process are shown in Table 8.1.

**Table 8.1 The encoding process**

<table>
<thead>
<tr>
<th>Pos</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Char</td>
<td>A</td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>B</td>
<td>B</td>
<td>A</td>
<td>B</td>
<td>C</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step</th>
<th>Pos</th>
<th>Match</th>
<th>Char</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>1</td>
<td>--</td>
<td>A</td>
<td>(0,0) A</td>
</tr>
<tr>
<td>2.</td>
<td>2</td>
<td>A</td>
<td>B</td>
<td>(1,1) B</td>
</tr>
<tr>
<td>3.</td>
<td>4</td>
<td>--</td>
<td>C</td>
<td>(0,0) C</td>
</tr>
<tr>
<td>4.</td>
<td>5</td>
<td>B</td>
<td>B</td>
<td>(2,1) B</td>
</tr>
<tr>
<td>5.</td>
<td>7</td>
<td>A B</td>
<td>C</td>
<td>(5,2) C</td>
</tr>
</tbody>
</table>

**8.1.2 Sub Algorithms In BZIP2 Compression**

The three sub-algorithms used in Bzip2 compression are:

Algorithm A performs the reversible transformation (BWT) that is applied to a block of text before compressing it.

Algorithm B Move To Front encoding (MTF) is a method for compressing the transformed block of text.

Algorithm C Huffman encoding for better compression.
8.1.2.1 Compression Entropy Computation for Bzip2

The overall steps needed to perform the Bzip2 compression are depicted in the Figure 8.1.

8.1.3 Implementation of Flow Charts

LZ77 implementation flow chart is shown in Figure 8.2. GZIP implementation flow chart is shown in Figure 8.3. In Figure 8.4, flow chart for generating Huffman tables is shown. BZIP2 implementation flow chart is shown in Figure 8.5.
Figure 8.3 GZIP implementation

Figure 8.4 For generating Huffman tables
8.2 METHODOLOGY

8.2.1 LZ77 Compression Technique

One of the most important all-purpose compression methods (Baumert et al 2004; David Salomon 2006; Mark Nelson and Jean-Loup Gailly 1995), called LZ77, was introduced by Ziv and Lempel (Ziv and Lempel 1977). It is a sequential dictionary method. The key idea is to encode future sections of the input through a reference to an earlier occurrence of the same section in the input. Always the longest match is used. The output consists of code words. These are triplets of the address of the earlier occurrence, the length and the next symbol. The next symbol is required for the case that no match is found. The entropy is estimated by

$$E_{LZ77} = \frac{\# \text{ Code words}}{\# \text{ input symbols}}$$

(8.1)
The ratio of code words to input symbols is used as estimation of the entropy.

In 1977 Ziv and Lempel developed an universal algorithm for lossless data compression (LZ77) using string-matching on a sliding window that is implemented in many tools including ‘zip’ and ‘stacker’.

The principle of this method is to use part of the previously-seen input stream as the dictionary. The encoder maintains a window to the input stream and shifts the input in that window from right to left as strings of symbols are being encoded. Thus, the method is based on a sliding window. The window below is divided into two parts. The part on the left is the search buffer. This is the current dictionary, and it includes symbols that have been input and encoded. The part on the right is the look-ahead buffer, containing text yet to be encoded. In practical implementations the search buffer is some thousands of bytes long, while the look-ahead buffer is only tens of bytes long.

### 8.2.2 GZIP Compression Technique

GZIP (Gailly 1993) is a variant of LZ77 technique i.e., it combines the LZ77 algorithm with a Huffman coding in the back end. The Figure 8.6 shows the steps involved.

![Figure 8.6 GZIP compression scheme](image)

GZIP (GNUzip) provides an optimized implementation of LZ77. Although the LZ77 technique produces a symbol sequence that is shorter than
the original one, the code words are still not stored in an optimal way. Each
code word would have a fixed size. In order to achieve maximum
compression, the output has to be optimally coded. This is done by GZIP. It
encodes the output of LZ77, a variant of it, in two Huffman trees, one for the
pointers and one for the lengths of the earlier occurrences. The next symbol is
only used when no match is found. This is done for blocks of the LZ77 output.

8.2.2.1 Huffman encoding

Huffman coding is a statistical technique which attempts to reduce
the amount of bits required to represent a string of symbols. In order to reduce
the amount of bits required to represent a string of symbols, symbols are
allowed to be of varying lengths. Shorter codes are assigned to the most
frequently used symbols, and longer codes to the symbols which appear less
frequently in the string.

Building a Huffman Tree

The Huffman code for an alphabet (set of symbols) may be
generated by constructing a binary tree with nodes containing the symbols to
be encoded and their probabilities of occurrence.

The tree may be constructed as follows:

Step1. Select the two parentless nodes with the lowest probabilities.

Step2. Create a new node which is the parent of the two lowest probability
nodes.
Step 3. Assign the new node a probability equal to the sum of its children's probabilities.

Step 4. Repeat Step 1 until there is only one parentless node left.

The code for each symbol may be obtained by tracing a path to the symbol from the root of the tree. A 1 is assigned for a branch in one direction and a 0 is assigned for a branch in the other direction. For example a symbol which is reached by branching right twice, left once may be represented by the pattern '110'. The Figure 8.7 depicts codes for nodes of a sample tree.

```
* \\
/ \ 
(0) (1) \\
/ \ 
(10)(11) \\
/ \ 
(110)(111)
```

**Figure 8.7 Huffman tree building**

Each byte is considered a symbol. Therefore each beat-to-beat interval needs to be encoded as one byte = 8 bit in the file, even though only 7 = \( \log_2(128) \) bit are required due to the quantization. The file is then compressed using GZIP. The entropy per bit of the input signal is estimated by:
Entropy = \frac{\text{Compressed file size}}{\# \text{ input symbols} \times \# \text{bit per symbol}} \text{[bit]} \quad (8.2)

The compressed file size divided by the number of input symbols, the intervals, yields the entropy per interval. Because each interval was encoded with 7 bit during quantization, normalization is done by dividing by 7. The resulting value can then be interpreted as proportion of information in the quantized tachogram.

### 8.2.3 BZIP2 Compression Technique

![BZIP2 Diagram](image)

**Figure 8.8 Basic steps in BZIP2 algorithm**

Bzip2 uses the Burrows-Wheeler transform (Burrows and Wheeler 1994) to convert frequently recurring character sequences into strings of identical letters, and then applies a move-to-front transform and finally Huffman coding. Basic steps in BZIP2 algorithm (Seward 2001) are shown in Figure 8.8. The algorithm described here was discovered by one of the authors (Wheeler) in 1983 while he was working at AT&T Bell Laboratories, though it has not previously been published.

The algorithm works by applying a reversible transformation to a block of input text. The transformation does not itself compress the data, but records it to make it easy to compress with simple algorithms such as move-to-front coding. Bzip2 algorithm achieves speed comparable to algorithms based on the techniques of Lempel and Ziv, but obtains compression close to
the best statistical modeling techniques. The size of the input block must be large (a few kilobytes) to achieve good compression.

This algorithm transforms a string $S$ of $N$ characters by forming the $N$ rotations (cyclic shifts) of $S$, sorting them lexicographically, and extracting the last character of each of the rotations. A string $L$ is formed from these characters, where the $i$th character of $L$ is the last character of the $i$th sorted rotation. In addition to $L$, the algorithm computes the index $i$ of the original string $S$ in the sorted list of rotations. Surprisingly, there is an efficient algorithm to compute the original string $S$ given only $L$ and $i$. The sorting operation brings together rotations with the same initial characters. Since the initial characters of the rotations are adjacent to the final characters, consecutive characters in $L$ are adjacent to similar strings in $S$. If the context of a character is a good predictor for the character, $L$ will be easy to compress with a simple locally-adaptive compression algorithm.

8.2.4 Entropy Using BZIP2 Compression Technique

Each byte is considered a symbol. The file is then compressed using Bzip2. The entropy per bit of the input signal is estimated by:

$$\text{Entropy} = \frac{\text{Compressed file size}}{\# \text{ input symbols} \times \# \text{ bit per symbol}} \text{[bit]}$$  \hspace{1cm} (8.3)

Bzip2 uses a different compression algorithm from gzip, which results in some advantages and some disadvantages.

8.2.5 Estimation of the Tachogram and Reference Selection

Calculation of the renormalised entropy requires estimating the tachogram distributions. One uses an autoregressive spectral estimation of tachogram. Applications of renormalised entropy to heart rate data based on
the Fast Fourier Transform were previously introduced but to overcome the potential lack of reproducibility and time instability of this measure, the autoregressive method REAR was developed. Early spectral analyses of HRV based on the fast Fourier transform as it was widely used in many applications and was computationally efficient. However, the FFT-based method requires equidistant sampling of HRV data, needs data windowing to decrease the edge effects and subsequent spectral leakage. Recent studies use autoregressive (AR) algorithms, which avoid the previous difficulty (i.e., requirement for stationarity of the signal in FFT calculations.). In AR modeling the condition of stationarity is much easier to satisfy for short time series.

A known problem of autoregressive spectral estimations is the bias which might appear even in idealized circumstances. To overcome this problem a sinusoidal oscillation with a fixed amplitude and frequency was added to the time series. The amplitude at 40 ms was chosen to obtain a dominant peak in the spectral estimation, and the frequency was set to 0.4 Hz, which is the upper limit of the high frequency band. Spectral density estimation in the interval \([0, 0.42]\) Hz was used to include all physiological modulations, and the calibration peak.

8.2.5.1 Reference tachogram selection

The healthy person with the most disordered spectral distribution ie, the highest peak in the ‘VLF’-band (< 0.05 Hz) has been considered. This distribution was the reference for all further analysis. The choice of the reference state is very important for the classification of HRV time series. Using a reference tachogram from a healthy subject with normal low and high frequency modulations the REAR method is designed so that either a decreased HRV or a pathological spectrum leads to positive values of renormalized entropy.
The low and high frequency oscillations (0.05-0.4 Hz) are rather low in comparison to the very low frequency peak (<0.05 Hz). An autoregressive spectral estimation is useful to emphasize the different spectral domains.

### 8.2.5.2 Auto regressive modeling

A model based on spectral estimation procedure, such as AR model is more suitable for HRV analysis because of its lesser sensitivity to the instationarities.

Various spectral methods for the analysis of the RR tachogram have been applied since the late 1960’s. Power spectral density (PSD) analysis provides the basic information of how power (i.e. variance) distributes as a function of frequency. Independent of the method employed, only an estimate of the true PSD of the signals can be obtained by proper mathematical algorithms.

Methods for the calculation of PSD may be generally classified as non-parametric and parametric. In most instances, both methods provide comparable results.

The advantages of the non-parametric methods are:

(a) the simplicity of the algorithm employed (Fast Fourier Transform—FFT— in most of the cases) and

(b) the high processing speed,

whilst the advantages of parametric methods are:

(a) smoother spectral components which can be distinguished independently of preselected frequency bands,
(b) easy post-processing of the spectrum with an automatic calculation of low and high frequency power components and easy identification of the central frequency of each component, and

(c) an accurate estimation of PSD even on a small number of samples on which the signal is supposed to maintain stationarity.

### 8.2.6 Burg’s or MEM Algorithm for Spectral Estimation

The MEM is also known as the Maximum Entropy Spectral Estimation (MESE) and the Maximum Entropy Spectrum Analysis (MESA). This method devised by Burg (1968) for estimating the AR parameters can be viewed as an order-recursive least-squares lattice method, based on the minimization of the forward and backward errors in linear predictors, with the constraint that the AR parameters satisfy the Levinson-Durbin recursion. The method was originally proposed for geophysical applications and was since dealt with by many researchers; several special problems have been investigated and attention has also been placed on the application of the original method have also been suggested.

The MEM PSD estimation can be posed as follows:

Given $p+1$ consecutive estimates of the correlation coefficients of the process $\{x(t)\}$, $r_x(i)$, $i = 0, 1, \ldots, p$, estimate the PSD of the process. Clearly what is needed for the estimation are the unknown correlation coefficients $r_x(i); i > p$. Burg has suggested these autocorrelation coefficients be extrapolated so that the time series characterized by the correlation has maximum entropy. Out of all time series having the $p + 1$ given first autocorrelation coefficient, the time series that yields maximum
entropy will be the most random one or, in other words, the estimated PSD will be the flattest among all PSDs having the given p+1 correlation coefficients.

The entropy is a measure of the amount of information (or “ignorance”) one has on a process. Consider a discrete process with J states each having the probability of occurrence pj, j=1, 2, ….., J. Assume that initially no knowledge is available on the probabilities pj. The information one possesses, on the system, is limited. If now one is given the value of a certain pj, one has gained a certain amount of information on the system or one’s state of ignorance has been reduced.

In the deterministic case, where one knows that event i has happened with probability one, all other pj’s are zero. In the random case, the entropy is always positive. The entropy is thus a measure for the randomness of the system.

In the continuous case the relative entropy is given by

\[ H = -\int_{s} p(s) \log_2 p(s) ds \]  \hspace{1cm} (8.4)

where p(s) is the amplitude distribution of the signal s(t). One would like to write down a relation between the estimated spectrum of the signal \( \hat{S}(w) \) and the entropy. Assume the signal s(t) was generated by passing a white signal through a linear filter having a transfer function S(w). It can be shown that the difference in entropies, \( \Delta H \), between that of the signal s(t) and the input is

\[ \Delta H = \int_{-\infty}^{\infty} \log_2 S(w) dw \]  \hspace{1cm} (8.5)
Since there are an infinite number of signals with white spectrum, the exact input is unknown. However, one knows that one wants to maximize the entropy subject to the constraints that

\[ r(n) = \int S(w) \exp(j2\pi w \Delta t) \, dw \]  

(8.6)

\[ n = 0, 1, \ldots, P \]

This constraint maximization will ensure that the estimated spectrum is the spectrum of a process having the flatness spectrum of all processes with the given \( P + 1 \) correlation coefficients.

The Lagrange Multiplier technique results in

\[ \hat{S}(w) = \frac{\sigma_p^2 A}{1 + \sum_{i=1}^{P} a_i \exp(-j2\pi wi)} \]  

(8.7)

where \( \sigma_p \) and \( a_i, i = 1, \ldots, P \) are determined from the data.

This method is also known as an order-recursive least-squares lattice method, based on the minimization of the forward and backward errors in linear predictors, with the constraint that the AR parameters \( A \) satisfy the Levinson-Durbin recursion.

So to derive the estimator, suppose that one is given the data \( x(n), n=0,1,\ldots,N-1 \), and let us consider the forward and backward prediction estimates of order \( m \), given as

\[ x(n) = -\sum_{k=1}^{m} a_m(k) x(n-k) \]
and the corresponding forward and backward errors $f_m(n)$ and $g_m(n)$ defined as $f_m(n) = x(n) - a_m^*(n)$ and $g_m(n) = x(n-m) - a_m^*(n-m)$ where $a_m(k), 0 \leq k \leq m-1, m=1,2,\ldots,p$, are the prediction coefficients. The least-squares error is

$$\mathcal{E}_m = \sum [ | f_m(n) |^2 + | g_m(n) |^2 ]$$

(8.9)

This error is to be minimized by selecting the prediction coefficients, subject to the constraint that they satisfy the Levinson-Durbin recursion by

$$a_m(k) = a_m(k) + K_m a_{m-1}(m-k) \quad (1 \leq k \leq m-1 \& 1 \leq m \leq p)$$

(8.10)

where $K_m = a_m(m)$ is the $m$th reflection in the lattice filter realization of the predictor.

When Equation (8.7) is substituted into the expressions for $f_m(n)$ and $g_m(n)$, the result is the pair of order-recursive equations for the forward and backward prediction errors given by

$$f_0(n) = g_0(n) = x(n)$$

(8.11)

$$f_m(n) = f_{m-1}(n) + K_m g_{m-1}(n-1) \quad m=1,2,\ldots,p$$

$$g_m(n) = K_m^* f_{m-1}(n) + g_{m-1}(n-1) \quad m=1,2,\ldots,p$$

If one substitutes the above equations in Equation (8.7) and perform the minimization of $\mathcal{E}_m$ with respect to the complex-valued reflection coefficient $K_m$, one obtains the result
\[ K_m = \frac{-\sum_{n=m}^{N-1} f_{m-1}(n) g_{m-1}^*(n-1)}{\frac{1}{2} \sum_{n=m}^{N-1} [|f_{m-1}(n)|^2 + |g_{m-1}(n-1)|^2]} \quad m=1,2, \ldots, p \]  

(8.12)

The term in the numerator of the above equation is an estimate of cross-correlation between the forward and backward prediction errors. With the normalization factors in the denominator of the equation, it is apparent that \(|K_m|<1\), so that the all-pole model obtained from the data is stable.

One notes that the denominator in Equation (8.12) is the least-squares estimate of the forward and backward errors, \(E_{m-1}^f\) and \(E_{m-1}^b\), respectively. Hence Equation (8.12) can be expressed as

\[ K_m = \frac{-\sum_{n=m}^{N-1} f_{m-1}(n) g_{m-1}^*(n-1)}{\frac{1}{2} [\hat{E}_{m-1}^f + \hat{E}_{m-1}^b]} \quad m=1,2, \ldots, p \]  

(8.13)

where \(\hat{E}_{m-1}^f + \hat{E}_{m-1}^b\) is an estimate of the total squared error \(E_m\).

To summarize, the Burg algorithm computes the reflection coefficients in the equivalent lattice structure and the Levinson-Durbin algorithm is used to obtain the AR model parameters. From the estimates of the AR parameters, one forms the Power spectrum estimate

\[ P_{xx}^{BU} = \frac{\hat{E}_p}{|1 + \sum_{k=1}^{p} \hat{\alpha}_p(k)e^{-j2\pi k}|^2} \]  

(8.14)

where

\[ \hat{\alpha}_p(k) \] are prediction coefficients of predictor of order \(=p\).
and $E_p$ is the least mean square error of p-stage lattice filter.

Advantages:-

1) High frequency resolution.

2) AR model developed using Burg’s method is stable (because all the reflection coefficients obtained have a value less than ‘1’).

Disadvantages:-

1) For High SNR signals, one gets frequency splitting.

2) For sinusoidal signals with short data length, one gets a shift in the frequency spectrum which is dependant on the initial phase. This property is known as phase dependent frequency bias.

8.2.7 Principle of Calculating Renormalized Entropy

\[
f_0 = (p_1, p_2, \ldots, p_N)
\]  
\[f_1 = (q_1, q_2, \ldots, q_N)
\] (8.15) (8.16)

These are the absolute AR coefficients obtained from the power spectra of RR time series of a healthy (reference) $f_0$ person and a cardiac patient $f_1$. $N$ is the order of the AR model chosen.

1. Renormalization:

\[
\bar{p}_k = \frac{1}{\sum_{i=1}^{N} p_i T}, \quad k=1,2,\ldots,
\] (8.17)

2. Find parameter $T$ such that the following conditions holds:
\[
\prod_{k=1}^{N} p_k^{p_k} = \prod_{k=1}^{N} p_k^{q_k}
\]

ie,

\[
\prod_{k=1}^{N} \left( \frac{1}{p_k} \right) = \prod_{k=1}^{N} p_k^{q_k}, k=1,2,\ldots,N
\]

3. Calculation of Shannon entropies:

\[
\overline{S}_0 = -\sum_{i=1}^{N} p_i \ln p_i, \text{for the reference subject tachogram}
\]

\[
S_1 = -\sum_{i=1}^{N} q_i \ln q_i, \text{for the cardiac patient tachogram}
\]

4. Renormalized entropy:

\[
\Delta \overline{S} = S_1 - \overline{S}_0
\]

The normal range of renormalized entropy is between -1.5 to 1.5.

This value is negative for healthy subjects (usually between 0 to -0.33) and positive for pathological cases (Voss A. 1993).

For healthy subjects, \( S_1 \) is positive. So renormalized entropy, \( \Delta \hat{S} \) is negative (because \( \hat{S}_0 \) is less than \( S_1 \)).

For arrhythmia patients, \( S_1 \) is negative and \( \Delta \hat{S} \), normalized entropy is positive.
Entropy-based algorithms for measuring the complexity of physiologic time series have been used.

Advantages:

The renormalised entropy as a measure of complexity can distinguish between two groups of patients:

1. patients with normal HRV (renormalised entropy less than 0 )
2. patients with reduced HRV - increased risk (renormalised entropy greater than 0 )

Therefore renormalised entropy as a measure of the relative degree of order is a suitable method for the detection of high risk patients for sudden cardiac death.

Disadvantage:

The main drawback of using Renormalised entropy is the selection of the reference tachogram. For a given control group one can do it easily but when analyzing new data sets, the problem arises. Whether the most disordered spectrum of all control group data sets should be taken or for each study an own reference state must be selected for reference is to be decided.

8.3 RESULTS

Nonlinear parameters can be used to analyse the health of the subjects. Dynamics would be relevant both to physiologic and pathologic cardiac function. Use of methods of nonlinear dynamics to study HRV is considered. There is possible relation of nonlinear indices and autonomic regulation of cardiovascular function. The methods quantify some nonlinear
characteristic of HRV. The nonlinear methods represent potentially promising tools for HRV. Nonlinear chaotic systems can produce very irregular data.

Table 8.2 Compression entropies of NSR subjects

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Age</th>
<th>Sex</th>
<th>LZ77 Entropy (bit)</th>
<th>GZIP Entropy (bit)</th>
<th>BZIP2 Entropy (bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>35</td>
<td>M</td>
<td>0.654</td>
<td>0.78</td>
<td>0.77</td>
</tr>
<tr>
<td>2</td>
<td>65</td>
<td>F</td>
<td>0.667</td>
<td>0.789</td>
<td>0.746</td>
</tr>
<tr>
<td>3</td>
<td>29</td>
<td>M</td>
<td>0.657</td>
<td>0.855</td>
<td>0.845</td>
</tr>
<tr>
<td>4</td>
<td>59</td>
<td>M</td>
<td>0.687</td>
<td>0.778</td>
<td>0.768</td>
</tr>
<tr>
<td>5</td>
<td>64</td>
<td>F</td>
<td>0.765</td>
<td>0.882</td>
<td>0.878</td>
</tr>
</tbody>
</table>

Avg.: 0.686  Avg.: 0.817  Avg.: 0.801  
SD : 0.0411  SD : 0.0432  SD : 0.0509

Table 8.3 CHF : Compression entropies of CHF patients

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Age</th>
<th>Sex</th>
<th>LZ77 Entropy (bit)</th>
<th>GZIP Entropy (bit)</th>
<th>BZIP2 Entropy (bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>22</td>
<td>M</td>
<td>0.356</td>
<td>0.345</td>
<td>0.137</td>
</tr>
<tr>
<td>2</td>
<td>71</td>
<td>M</td>
<td>0.456</td>
<td>0.356</td>
<td>0.157</td>
</tr>
<tr>
<td>3</td>
<td>63</td>
<td>F</td>
<td>0.4789</td>
<td>0.356</td>
<td>0.106</td>
</tr>
<tr>
<td>4</td>
<td>53</td>
<td>M</td>
<td>0.4267</td>
<td>0.456</td>
<td>0.045</td>
</tr>
<tr>
<td>5</td>
<td>54</td>
<td>F</td>
<td>0.3678</td>
<td>0.379</td>
<td>0.146</td>
</tr>
</tbody>
</table>

Avg.: 0.417  Avg.: 0.378  Avg.: 0.118  
SD : 0.0481  SD : 0.0404  SD : 0.0403
Table 8.4 AF : Compression entropies of AF patients

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Age</th>
<th>Sex</th>
<th>LZ77 Entropy (bit)</th>
<th>GZIP Entropy (bit)</th>
<th>BZIP2 Entropy (bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>81</td>
<td>F</td>
<td>0.412</td>
<td>0.56</td>
<td>0.356</td>
</tr>
<tr>
<td>2</td>
<td>79</td>
<td>M</td>
<td>0.356</td>
<td>0.52</td>
<td>0.246</td>
</tr>
<tr>
<td>3</td>
<td>79</td>
<td>M</td>
<td>0.356</td>
<td>0.435</td>
<td>0.278</td>
</tr>
<tr>
<td>4</td>
<td>81</td>
<td>F</td>
<td>0.456</td>
<td>0.43</td>
<td>0.3023</td>
</tr>
<tr>
<td>5</td>
<td>76</td>
<td>M</td>
<td>0.489</td>
<td>0.51</td>
<td>0.354</td>
</tr>
</tbody>
</table>

Avg.: 0.419  Avg.: 0.491  Avg.: 0.307
SD: 0.0531   SD: 0.0506  SD: 0.0429

For NSR subjects, one male aged 59 has entropy of 0.687 using LZ77 algorithm. One female aged 65 has entropy of 0.667. Both are correct. One male aged 29 has entropy of 0.855 using GZIP algorithm. One female aged 65 has entropy of 0.789. Both are correct. One male aged 29 has entropy of 0.845 using BZIP2 algorithm. One male aged 35 has entropy of 0.77. Both are correct. In Table 8.2, compression entropies of NSR subjects is shown.

For CHF patients, one male aged 53 has entropy of 0.4267 using LZ77 algorithm. One female aged 54 has entropy of 0.3678. Both are correct. One female aged 54 has entropy of 0.379 using GZIP algorithm. A male aged 53 has entropy of 0.456. Both are correct. One female aged 63 has entropy of 0.106 using BZIP2 algorithm. One male aged 22 has entropy of 0.137. Both are correct. In Table 8.3, compression entropies of CHF patients is shown.

For AF patients, one female aged 81 has entropy of 0.412 using LZ77 algorithm. One female aged 81 has entropy of 0.456. Both are correct. One male aged 76 has entropy of 0.51 using GZIP algorithm. A male aged 79 has entropy of 0.435. Both are correct. A female aged 81 has entropy of
0.3023 using BZIP2 algorithm. A male aged 76 has entropy of 0.354. In Table 8.4, compression entropies of AF patients is shown.

Power Spectral Density (PSD) of a subject versus frequency is shown in Figure 8.9. Power spectral density of reference person is similar (Voss et al 1994). HRV spectrum of Normal is considered. Power spectral density (PSD) of a normal subject versus frequency is shown in Figure 8.9.

![HRV spectrum of a subject](image)

**Figure 8.9** HRV spectrum of a subject

Renormalized entropy (RE):

Final renormalized entropy is shown in Table 8.5.

**Table 8.5 Final renormalized entropy**

<table>
<thead>
<tr>
<th>S. No.</th>
<th>( f_0 )</th>
<th>( f_1 )</th>
<th>Final Renormalized Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>NSR (Healthy)</td>
<td>NSR</td>
<td>-0.254403</td>
</tr>
<tr>
<td>2.</td>
<td>NSR (Healthy)</td>
<td>CHF</td>
<td>0.063733</td>
</tr>
<tr>
<td>3.</td>
<td>NSR (Healthy Arrhythmia)</td>
<td></td>
<td>0.186612</td>
</tr>
<tr>
<td>4.</td>
<td>NSR (Healthy)</td>
<td>Atrial Fibrillation</td>
<td>0.117650</td>
</tr>
</tbody>
</table>
8.4 DISCUSSION

For Arrhythmia the average value of entropy of L277 is 0.512 and the average value of entropy of GZIP is 0.930 and the average value of entropy of BZIP2 is 1.443. For normal sinus Rhythm, the average value of entropy of LZ77 is 0.348 and the average value of entropy of GZIP in 0.6 and the average value of entropy of BZIP2 is 0.6.

The renormalized entropy of healthy subjects (NSR subjects) is typically less than zero. Only heavy physical or mental stress cause values greater than zero. High risk patients like CHF, Arrhythmia and Atrial Fibrillation patients show a renormalized entropy greater than zero during the whole day (except during instationarities). Using maximum entropy method power spectral density of a subject versus frequency is obtained.

8.5 CONCLUSIONS

Entropy estimation is not carried out for SCD patients data.

Bzip2, which implements Burrows-Wheeler compression, produced the highest compression rate and thus the closest entropy estimation.

The renormalised entropy as a measure of complexity was able to distinguish between two groups of patients:

- patients with normal HRV (renormalised entropy less than 0 (-0.33<RE<0) )
- patients with reduced HRV - increased risk (renormalised entropy greater than 0 )

The Renormalized entropy value is negative for a healthy patient most of the times except during instationarities and mostly positive for a
pathological person. The RE parameter can be used to identify the complexity variations in pathological states. Thus, Renormalised entropy as a measure of the relative degree of order is a suitable method for the detection of high risk patients for sudden cardiac death.

Using maximum entropy method, power spectral density of a subject versus frequency is obtained. Renormalized entropy is negative for the combination $f_0 : \text{NSR}$ and $f_1 : \text{NSR}$. Renormalized entropy is positive for the combinations $f_0 : \text{NSR}$ and $f_1 : \text{CHF}$, $f_0 : \text{NSR}$ and $f_1 : \text{arrhythmia}$ and $f_0 : \text{NSR}$ and $f_1 : \text{Atrial fibrillation}$.

In the next chapter, next nonlinear parameters, wavelet coefficient entropy and wavelet sample entropy are computed.