The Protein Family Classification in Protein Databases via Entropy Measures

R.P. Mondaini, S.C. de Albuquerque Neto

Abstract

In the present work, we review the fundamental methods which have been developed in the last few years for classifying into families and clans the distribution of amino acids in protein databases. This is done through functions of random variables, the Entropy Measures of probabilities of the occurrence of the amino acids. An intensive study of the Pfam databases is presented with restriction to families which could be represented by rectangular arrays of amino acids with $m$ rows (protein domains) and $n$ columns (amino acids). This work is also an invitation to scientific research groups worldwide to undertake the statistical analysis with arrays of different numbers of rows and columns, since we believe in the mathematical characterization of the distribution of amino acids as a fundamental insight on the determination of protein structure and evolution.

1 Introduction and Motivation

DESIDERATA: “To translate the information contained on protein databases in terms of random variables in order to model a dynamics of folding and unfolding of proteins”.

The information on the planetary motion has been annotated on Astronomical almanacs (Ephemerides) along centuries and can be derived and analyzed by Classical Dynamics and Deterministic Chaos as well as confirmed and corrected by General Relativity. The information which is accumulated on Biological almanacs (Protein databases) on the last decades, is still waiting for its first description to be done by a successful theory of folding and unfolding of proteins. We think that this study should be started from the evolution of protein families and its association into Family clans as a necessary step of their development.

The first fundamental idea to be developed here is that proteins do not evolve independently. We introduce several arguments and we have done many calculations to span the bridge over the facts about protein evolution in order
to emphasize the existence of a protein family formation process (PFFP), a successful pattern recognition method and a coarse-grained protein dynamics are driven by optimal control theory \[1, 2, 3, 4, 5, 6\]. Proteins, or their “intelligent” parts, protein domains, evolve together as a family of protein domains. We then realize that the exclusion of the evolution of an “orphan” protein is guaranteed by the probabilistic approach to be introduced in the present contribution. We think that the elucidation of the nature of intermediate stages of the folding/unfolding dynamics, in order to circumvent the Levinthal “paradox” \[7, 8\] as well as the determination of initial conditions, should be found from a detailed study of this PFFP process. A byproduct of this approach is the possibility of testing the hypothesis of agglutination of protein families into Clans by rigorous statistical methods like ANOVA \[9, 4\].

We take many examples of Entropy Measures as the generalized functions of random variables on our modelling. These are the probabilities of occurrence of amino acids in rectangular arrays which are the representatives of families of protein domains. In section 2, the sample space which is adequate for this statistical approach is described in detail. We start from the definition of probabilities of occurrence and the restrictions imposed on the number of feasible families by the structure of this sample space. Section 3 introduces the set of Sharma-Mittal Entropy Measures \[10, 4\] to be adopted as the functions of probabilities of occurrence in the statistical analysis to be developed. The Mutual Information measures associated with the Sharma-Mittal set, as well as the normalized Jaccard distance measures, are also introduced in this section. In section 4, we present a naive sketch of assessing Protein database, to set the stage for a more efficient approach of the following sections. In section 5, we point out the inconvenience of the Maple computing system for the statistical calculations to be done, by displaying tables with all CPU and real times necessary to perform all necessary calculations. We have also provided in this section, some adaptation of our methods in order to be used with the Perl computing system and we compare the new times of calculation with those by using Maple at the beginning of the section. We also include some comments on the use of Perl, especially on its oddness to calculate with the input data given in arrays and the way of circumventing this. However, we also stress that despite the fact that joint probabilities and their powers could be usually calculated, the output will come randomly distributed and the CPU and real times will increase too much to favour the calculation of the entropy measures. This is due to the intrinsic “Hash” structure \[11\] of the Perl computing system. We then introduce a modified array structure in order to calculate with Perl.

2 The Sample Space for a Statistical Treatment

We consider a rectangular array of \(m\) rows (protein domains) and \(n\) columns (amino acids). These arrays are organized from the protein database whose domains are classified into families and clans by the professional expertise of senior biologists \[12, 13\].
The random variable is the probability of occurrences of amino acids, $p_j(a)$, $j = 1, 2, \ldots, n$, $a = A, C, D, \ldots, W, Y$ (one-letter code for amino acids), to be given by

$$p_j(a) \equiv \frac{n_j(a)}{m} \quad (1)$$

where $n_j(a)$ is the number of occurrences of the amino acid $a$ in the $j$-th column.

Eq.(1) could be also interpreted as the components of $n$ vectors of 20 components each

$$\begin{pmatrix} p_1(A) \\ \vdots \\ p_1(Y) \end{pmatrix} \begin{pmatrix} p_2(A) \\ \vdots \\ p_2(Y) \end{pmatrix} \cdots \begin{pmatrix} p_n(A) \\ \vdots \\ p_n(Y) \end{pmatrix} \quad (2)$$

and we have

$$\sum_a n_j(a) = m, \ \forall j \Rightarrow \sum_a p_j(a) = 1, \ \forall j \quad (3)$$

Analogously, we could also introduce the joint probability of occurrence of a pair of amino acids $a, b$ in columns $j, k$, respectively $P_{jk}(a, b)$ as the random variables. These are given by

$$P_{jk}(a, b) = \frac{n_{jk}(a, b)}{m} \quad (4)$$

where $n_{jk}(a, b)$ is the number of occurrences of the pair of amino acids $a, b$ in columns $j, k$, respectively.

A convenient interpretation of these joint probabilities could be the elements of $\frac{n(n-1)}{2}$, square matrices of $20 \times 20$ elements, to be written as

$$P_{jk} = \begin{pmatrix} P_{jk}(A, A) & \cdots & P_{jk}(A, Y) \\ \vdots & \ddots & \vdots \\ P_{jk}(Y, A) & \cdots & P_{jk}(Y, Y) \end{pmatrix} \quad (5)$$

where $j = 1, 2, \ldots, (n-1); k = j + 1, \ldots, n$.

We can also write,

$$P_{jk}(a, b) = P_{jk}(a|b)p_k(b), \quad (6)$$

This equation can be also taken as another definition of joint probability. $P_{jk}(a|b)$ is the Conditional probability of occurrence of the amino acid $a$ in column $j$ if the amino acid $b$ is already found in column $k$. We then have,

$$\sum_a P_{jk}(a|b) = 1 \quad (7)$$

From eqs.(6), (7), we have:

$$\sum_a P_{jk}(a, b) = p_k(b) \quad (8)$$
and from eq.(8),
\[ \sum_a \sum_b P_{jk}(a,b) = 1 \] (9)

which is an identity since \( P_{jk}(a,b) \) is also a probability.

Eqs.(8) and (9) can be also derived from
\[ \sum_a n_{jk}(a,b) = n_k(b); \quad \sum_a \sum_b n_{jk}(a,b) = m \] (10)

and the definitions, eqs.(1), (4).

We now have from Bayes’ law:
\[ P_{jk}(a|b) p_k(b) = P_{kj}(b|a) p_j(a) \] (11)

and from eq.(11), the property of symmetry,
\[ P_{jk}(a,b) = P_{kj}(b,a) \] (12)

The matrices \( P_{jk} \) can be organized in a triangular array:
\[
P = \begin{bmatrix}
P_{12} & P_{13} & P_{14} & \cdots & P_{1(n-2)} & P_{1(n-1)} & P_{1n} \\
P_{23} & P_{24} & \cdots & P_{2(n-2)} & P_{2(n-1)} & P_{2n} \\
P_{34} & \cdots & P_{3(n-2)} & P_{3(n-1)} & P_{3n} \\
& \ddots & & & & \vdots \\
P_{n-3,n-2} & \cdots & P_{n-3,n-1} & P_{n-3,n} \\
P_{n-2,n-1} & \cdots & P_{n-2,n} \\
P_{n-1,n} & & & & & & 
\end{bmatrix}
\] (13)

The number of matrices until the \( P_{jk} \)-th one is given by
\[ C_{jk} = j(n - 1) - \frac{j(j - 1)}{2} - (n - k) \] (14)

These numbers can be also arranged as a triangular array:
\[
C = \begin{bmatrix}
1 & 2 & 3 & 4 & 5 & 6 & \cdots & (n-3) & (n-2) & (n-1) \\
n & (n+1) & (n+2) & (n+3) & (n+4) & \cdots & (2n-5) & (2n-4) & (2n-3) \\
(2n-2) & (2n-1) & 2n & (2n+1) & \cdots & (3n-8) & (3n-7) & (3n-6) \\
(3n-5) & (3n-4) & (3n-3) & \cdots & (4n-12) & (4n-11) & (4n-10) \\
(4n-9) & (4n-8) & \cdots & (5n-17) & (5n-16) & (5n-15) \\
& \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
(5n-14) & \cdots & (6n-23) & (6n-22) & (6n-21) \\
& \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
& & \frac{1}{2}(n^2 - n - 10) & \frac{1}{2}(n^2 - n - 8) & \frac{1}{2}(n+2)(n-3) \\
& & \frac{1}{2}(n^2 - n - 4) & \frac{1}{2}(n+1)(n-2) \\
& & \frac{1}{2}n(n-1) 
\end{bmatrix}
\] (15)
Eq.(13) should be used for the construction of a computational code to perform all necessary calculations. We postpone to other publication the presentation of some interesting results on the analysis of eq.(15).

The calculation of the matrix elements $P_{jk}(a, b)$ from a rectangular array $m \times n$ of amino acids is done by the “concatenation” process which is easily implemented on computational codes. We choose a pair of columns $j = \bar{j}, k = \bar{k}$ from the strings, $a = A, C, \ldots, W, Y, b = A, C, \ldots, W, Y$ and we look for the occurrence of the combinations $ab = AA, AC, \ldots, AW,AY, CA, CC, \ldots, CW, CY, \ldots, WA, WC, \ldots, WW, WY, \ldots, YA, YC, \ldots, YW, YY$. We then calculate their numbers of occurrences $n_{\bar{j}\bar{k}}(A,A), n_{\bar{j}\bar{k}}(A,C), \ldots, n_{\bar{j}\bar{k}}(Y,W), n_{\bar{j}\bar{k}}(Y,Y)$ and the corresponding probabilities $P_{\bar{j}\bar{k}}(A,A), P_{\bar{j}\bar{k}}(A,C), \ldots, P_{\bar{j}\bar{k}}(Y,W), P_{\bar{j}\bar{k}}(Y,Y)$ from eq.(4). We do the same for the other $n^2 - n - 2$ pairs of columns.

As an example, let us suppose that we have the $3 \times 4$ array:

|   | A | C | A | D |
|---|---|---|---|---|
| 1 | C | A | D | D |
| 2 | D | A | C | C |

Figure 1: An example of a $3 \times 4$ array with amino acids A, C, D.

Let us choose the pair of columns 1,2. We look for the occurrence of the combinations $AA, AC, AD, CA, CC, CD, DA, DC, DD$ on the pair of columns 1,2 of the array above and we found $n_{12}(A,C) = 1, n_{12}(C,A) = 1, n_{12}(D,A) = 1$. The others $n_{12}(a,b) = 0$. From eq.(4) we can write for the matrices $P_{jk}$ of eq.(5):

$$
P_{12} = \begin{pmatrix} 0 & 1/3 & 0 \\ 1/3 & 0 & 0 \\ 1/3 & 0 & 0 \end{pmatrix} ; P_{13} = \begin{pmatrix} 1/3 & 0 & 0 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \end{pmatrix} ; P_{14} = \begin{pmatrix} 0 & 0 & 1/3 \\ 0 & 1/3 & 0 \\ 0 & 1/3 & 0 \end{pmatrix}
$$

$$
P_{23} = \begin{pmatrix} 0 & 1/3 & 0 \\ 1/3 & 0 & 0 \\ 1/3 & 0 & 0 \end{pmatrix} ; P_{24} = \begin{pmatrix} 0 & 1/3 & 1/3 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \end{pmatrix} ; P_{34} = \begin{pmatrix} 0 & 0 & 1/3 \\ 1/3 & 0 & 0 \\ 0 & 0 & 1/3 \end{pmatrix}
$$

(16)

The Maple computing system “recognizes” the matricial structure through its Linear Algebra package. The Perl computing system “operates” only with “strings”. The results above are easily obtained in Maple, but in Perl we have to find alternative ways of calculating the joint probabilities. The first method is to calculate the probabilities per row of the $3 \times 4$ array. We have for the first
row:

\[ \Pi^{(1)}_{12} = \begin{pmatrix} 0 & 1/3 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(1)}_{13} = \begin{pmatrix} 1/3 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(1)}_{14} = \begin{pmatrix} 0 & 0 & 1/3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \]

\[ \Pi^{(2)}_{23} = \begin{pmatrix} 0 & 0 & 0 \\ 1/3 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(2)}_{24} = \begin{pmatrix} 0 & 0 & 1/3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(2)}_{34} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \end{pmatrix} \]  (17)

For the second row:

\[ \Pi^{(2)}_{12} = \begin{pmatrix} 0 & 0 & 0 \\ 1/3 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(2)}_{13} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(2)}_{14} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \end{pmatrix} \]

\[ \Pi^{(3)}_{23} = \begin{pmatrix} 0 & 0 & 1/3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(3)}_{24} = \begin{pmatrix} 0 & 0 & 1/3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(3)}_{34} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \end{pmatrix} \]  (18)

For the third row:

\[ \Pi^{(3)}_{12} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1/3 & 0 & 0 \end{pmatrix}; \quad \Pi^{(3)}_{13} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1/3 \end{pmatrix}; \quad \Pi^{(3)}_{14} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1/3 \end{pmatrix} \]

\[ \Pi^{(3)}_{23} = \begin{pmatrix} 0 & 0 & 1/3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(3)}_{24} = \begin{pmatrix} 0 & 0 & 1/3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \quad \Pi^{(3)}_{34} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \end{pmatrix} \]  (19)

We stress that Perl does not recognize these matrix structures. This is just our arrangement in order to make comparison with Maple calculations. However, Perl “knows” how to sum the calculations done per rows to obtain:

\[ \Pi^{(1)}_{12} + \Pi^{(2)}_{12} + \Pi^{(3)}_{12} = \begin{pmatrix} 0 & 1/3 & 0 \\ 1/3 & 0 & 0 \\ 1/3 & 0 & 0 \end{pmatrix} \equiv P_{12} \]  (20)

\[ \Pi^{(1)}_{13} + \Pi^{(2)}_{13} + \Pi^{(3)}_{13} = \begin{pmatrix} 1/3 & 0 & 0 \\ 0 & 0 & 1/3 \\ 0 & 1/3 & 0 \end{pmatrix} \equiv P_{13} \]  (21)

\[ \Pi^{(1)}_{14} + \Pi^{(2)}_{14} + \Pi^{(3)}_{14} = \begin{pmatrix} 0 & 0 & 1/3 \\ 0 & 0 & 1/3 \\ 0 & 1/3 & 0 \end{pmatrix} \equiv P_{14} \]  (22)

\[ \Pi^{(1)}_{23} + \Pi^{(2)}_{23} + \Pi^{(3)}_{23} = \begin{pmatrix} 1/3 & 1/3 & 0 \\ 1/3 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \equiv P_{23} \]  (23)

\[ \Pi^{(1)}_{24} + \Pi^{(2)}_{24} + \Pi^{(3)}_{24} = \begin{pmatrix} 0 & 1/3 & 1/3 \\ 0 & 0 & 1/3 \\ 0 & 0 & 0 \end{pmatrix} \equiv P_{24} \]  (24)
\[ \Pi_{34}^{(1)} + \Pi_{34}^{(2)} + \Pi_{34}^{(3)} = \begin{pmatrix} 0 & 0 & 1/3 \\ 0 & 1/3 & 0 \\ 0 & 0 & 1/3 \end{pmatrix} \equiv P_{34} \]  

We are then able to translate the Perl output in “matrix language”. However, this output does not come as an ordered set of joint probabilities, as we have done by arranging the output in the form of the matrices \( \Pi_{jk}^{(l)} \), \( j = 1, 2, 3 \), \( k = 2, 3, 4 \), \( l = 1, 2, 3 \). In order to calculate functions of the probabilities as the entropy measures, it will take too much time for the Perl computing system to collect the necessary probability values. This is due to the “Hash” structure of the Perl as compared to the usual “array” structure of the Maple. A new form of arranging the strings to favour an a priori ordination will circumvent this inconvenience of the “hash” structure. Let us then write the following extended string associated to the \( m \times n \) rectangular array:

\[
\begin{pmatrix}
(A\,C\,D) & (C\,A\,A) & (A\,D\,C) & (D\,D\,C)
\end{pmatrix}
\]

We then get

\[
P_{12}(A, C) = 1/3, \quad P_{22}(C, A) = 1/3, \quad P_{12}(D, A) = 1/3
\]

\[
P_{13}(A, A) = 1/3, \quad P_{23}(C, D) = 1/3, \quad P_{13}(D, C) = 1/3
\]

\[
P_{14}(A, D) = 1/3, \quad P_{24}(C, D) = 1/3, \quad P_{14}(D, C) = 1/3
\]

\[
P_{23}(C, A) = 1/3, \quad P_{33}(A, D) = 1/3, \quad P_{23}(A, C) = 1/3
\]

\[
P_{24}(C, D) = 1/3, \quad P_{34}(A, D) = 1/3, \quad P_{24}(A, C) = 1/3
\]

\[
P_{34}(A, D) = 1/3, \quad P_{44}(D, D) = 1/3, \quad P_{34}(C, C) = 1/3
\]

All the other joint probabilities \( P_{jk}(a, b) \) are equal to zero.

This is a feasible treatment for the “hash” structure. In the example solved above the probabilities will come already ordered in triads. This will save time in the calculations with the Perl system.

It should be stressed that the Perl computing system does not recognize any formal relations of Linear Algebra. However, it does quite well if these relations are converted into products and sums. In order to give an example of working with the Perl system, we take a calculation with the usual Shannon Entropy measure. The calculation of the Entropy for the columns \( j, k \) is done by

\[
S_{jk} = - \sum_a \sum_b P_{jk}(a, b) \log P_{jk}(a, b) = - \text{Tr} \left( P_{jk}(\log P_{jk})^T \right) \]

where \( P_{jk} \) is the matrix given in eq.(5) and \( \text{Tr}, T \) stands for the operations of taking the trace and transposing a matrix, respectively. The matrix \((\log P_{jk})^T\) is given by

\[
(\log P_{jk})^T = \begin{pmatrix}
\log P_{jk}(A, A) & \ldots & \log P_{jk}(Y, A) \\
\vdots & \ddots & \vdots \\
\log P_{jk}(A, Y) & \ldots & \log P_{jk}(Y, Y)
\end{pmatrix}
\]
we also include for a useful reference, the matrix

\[ p_j(p_k)^T = \begin{pmatrix} p_j(A)p_k(A) & \ldots & p_j(A)p_k(Y) \\ \vdots & \ddots & \vdots \\ p_j(Y)p_k(A) & \ldots & p_j(Y)p_k(Y) \end{pmatrix} \]

Since from eqs.(20)–(25), we have

\[ P_{jk} = \sum_{l=1}^{m} \Pi_{jk}^{(l)} \]  \hspace{1cm} (28)

we can write:

\[ S_{jk} = -\text{Tr} \left( \left( \sum_{l=1}^{m} \Pi_{jk}^{(l)} \right) (\log P_{jk})^T \right) = -\sum_{l=1}^{m} \text{Tr} \left( \Pi_{jk}^{(l)} (\log P_{jk})^T \right) \]  \hspace{1cm} (29)

There is no problem for calculating in Perl, if we prepare eq.(28) by expressing previously all the products and sums to be done. The real problem with Perl calculations is the arrangement of the output of values \( P_{jk}(a,b) \), due to the “hash” structure as have been stressed above.

3 Entropy Measures. The Sharma-Mittal set and the associated Jaccard Entropy measure

We start this section with the definition of the two-parameter Sharma-Mittal entropies [10, 1, 2]

\[ (SM)_{jk}(r,s) = -\frac{1}{1-r} \left( 1 - \left( \sum_a \sum_b (P_{jk}(a,b))^s \right)^{\frac{1}{1-s}} \right) \]  \hspace{1cm} (30)

\[ (SM)_{j}(r,s) = -\frac{1}{1-r} \left( 1 - \left( \sum_a (p_j(a))^s \right)^{\frac{1}{1-s}} \right) \]  \hspace{1cm} (31)

where \( p_j(a) \) and \( P_{jk}(a,b) \) are the simple and joint probabilities of occurrence of amino acids as defined on eqs.(1) and (4), respectively. \( r, s \) are non-dimensional parameters.

We can associate to the entropy measures above their corresponding one parameter forms to be given by the limits:

\[ H_{jk}(s) = \lim_{r \to s} (SM)_{jk}(r,s) = -\frac{1}{1-s} \left( 1 - \sum_a \sum_b (P_{jk}(a,b))^s \right) \]  \hspace{1cm} (32)

\[ H_{j}(s) = \lim_{r \to s} (SM)_{j}(r,s) = -\frac{1}{1-s} \left( 1 - \sum_a (p_j(a))^s \right) \]  \hspace{1cm} (33)
These are the Havrda-Charvat Entropy Measures and they will be specially emphasized in the present work. Other alternative proposals for the single parameter entropies are given by

The Renyi’s Entropy measures:

\[ R_{jk}(s) = \lim_{r \to 1} (SM)_{jk}(r, s) = \frac{1}{1-s} \log \left( \sum_a \sum_b (P_{jk}(a, b))^s \right) \]  

(34)

\[ R_j(s) = \lim_{r \to 1} (SM)_j(r, s) = \frac{1}{1-s} \log \left( \sum_a (p_j(a))^s \right) \]  

(35)

The Landsberg-Vedral Entropy measures:

\[ L_{jk}(s) = \lim_{r \to 2-s} (SM)_{jk}(r, s) = \frac{1}{1-s} \left( 1 - \left( \sum_a \sum_b (P_{jk}(a, b))^s \right)^{-s} \right) \]  

(36)

\[ L_j(s) = \lim_{r \to 2-s} (SM)_j(r, s) = \frac{1}{1-s} \left( 1 - \left( \sum_a (p_j(a))^s \right)^{-s} \right) \]  

(37)

All these Entropy measures have the free-parameter Shannon entropy in the limit \( s \to 1 \).

\[ \lim_{s \to 1} H_{jk}(s) = \lim_{s \to 1} R_{jk}(s) = \lim_{s \to 1} L_{jk}(s) = S_{jk} \]  

(38)

\[ \lim_{s \to 1} H_j(s) = \lim_{s \to 1} R_j(s) = \lim_{s \to 1} L_j(s) = S_j \]  

(39)

where

\[ S_{jk} = - \sum_a \sum_b P_{jk}(a, b) \log P_{jk}(a, b) \]  

(27)

\[ S_j = - \sum_a p_j(a) \log p_j(a) \]  

(40)

are the Shannon entropy measures [6].

We now introduce a convenient version of a Mutual Information measure:

\[ M_{jk}(r, s) = \frac{1}{1-r} \left( 1 - \left( \frac{\sum_a \sum_b (P_{jk}(a, b))^s}{\sum_a \sum_b (P_j(a)p_k(b))^s} \right)^{1-s} \right) \]  

(41)

We can see that \( M_{jk}(r, 0) = 0 \) and if \( \exists j, k \) such that \( P_{jk}(a, b) = p_j(a)p_k(b) \Rightarrow M_{jk}(r, s) = 0 \). We also have,

\[ M_{jk}(1, s) = \lim_{r \to 1} M_{jk}(r, s) = - \frac{1}{1-s} \log \left( \frac{\sum_a \sum_b (P_{jk}(a, b))^s}{\sum_a \sum_b (p_j(a)p_k(b))^s} \right) \]  

(42)
and in the limit \( s \to 1 \)

\[
M_{jk} = \lim_{s \to 1} M_{jk}(1, s) = \sum_{a} \sum_{b} P_{jk}(a, b) \log P_{jk}(a, b) - \sum_{a} \sum_{b} p_{j}(a)p_{k}(b) \log (p_{j}(a)p_{k}(b))
\]  

(43)

and from the identities:

\[
\sum_{a} p_{j}(a) = 1, \forall j; \quad \sum_{b} p_{k}(b) = 1, \forall k
\]

\[
\sum_{a} P_{jk}(a, b) = p_{k}(b), \forall j; \quad \sum_{b} P_{jk}(a, b) = p_{j}(a), \forall k
\]

obtained from eqs. (3), (4), (6), (7), we can also write instead eq.(43):

\[
M_{jk} = \sum_{a} \sum_{b} P_{jk}(a, b) \log P_{jk}(a, b) - \sum_{a} \sum_{b} P_{jk}(a, b) \log (p_{j}(a)p_{k}(b))
\]

(44)

It should be stressed that we are not assuming that \( P_{jk}(a, b) \equiv p_{j}(a)p_{k}(b) \) above. This equality is assumed to be valid only for \( j = j, k = k \).

Eq.(43) or (44) can be also written as:

\[
M_{jk} = -S_{jk} + S_{j} + S_{k}
\]

(45)

where \( S_{jk} \) and \( S_{j}, S_{k} \) are the Shannon entropy measures for joint and single probabilities, respectively, eqs.(27), (40).

As an additional topic, we emphasize that the Mutual Information measure can be also derived from the Kullback-Leibler divergence \([6]\) which is written as

\[
(KL)_{jk}(b) = \sum_{a} P_{jk}(a|b) \log \left( \frac{P_{jk}(a|b)}{p_{j}(a)} \right)
\]

(46)

where \( P_{jk}(a|b) \) is the Conditional probability, eq.(8). We then have,

\[
(KL)_{jk}(b) = \sum_{a} \frac{P_{jk}(a, b)}{p_{k}(b)} \log \left( \frac{P_{jk}(a, b)}{p_{j}(a)p_{k}(b)} \right)
\]

(47)

and the \( M_{jk} \) mutual information measure will be given by

\[
M_{jk} = \sum_{b} p_{k}(b) (KL)_{jk}(b) = \sum_{a} \sum_{b} P_{jk}(a, b) \log \left( \frac{P_{jk}(a, b)}{p_{j}(a)p_{k}(b)} \right)
\]

(48)

which is the same as eq.(44), q.e.d.

As the last topic of this section, we now introduce the concept of Information Distance and we then derive the Jaccard Entropy measure as an obvious consequence. Let us write:

\[
d_{jk}(r, s) = H_{jk}(r, s) - M_{jk}(r, s)
\]

(49)
Since we are working with Entropy measures, we have to satisfy the non-negativeness criteria:

\[ H_{jk}(r, s) \geq 0; \quad M_{jk}(r, s) \geq 0; \quad H_{jk}(r, s) - M_{jk}(r, s) \geq 0 \]  

(50)

This means that by satisfying the inequalities (50), restrictions on the \( r, s \) parameters should be discovered and considered for the description of the protein databases by Entropy measures like \( H_{jk}(r, s) \).

From inequalities (50), we can write,

\[ 0 \leq d_{jk}(r, s) = H_{jk}(r, s) - M_{jk}(r, s) \leq H_{jk}(r, s) \]  

(51)

and

\[ 0 \leq J_{jk}(r, s) \leq 1 \]  

(52)

where

\[ J_{jk}(r, s) = 1 - \frac{M_{jk}(r, s)}{H_{jk}(r, s)} \]  

(53)

is the normalized Jaccard Entropy Measure as obtained from the normalized Information Distance. We then give below the results of checking the inequalities (50) for some families of the Pfam database. We shall take the limit \( r \to s \) and we work with the corresponding one-parameter Entropy measures: \( H_j(s), H_{jk}(s), M_{jk}(s), J_{jk}(s) \). We then have to check:

\[ H_{jk}(s) \geq 0, \quad M_{jk}(s) \geq 0, \quad H_{jk}(s) - M_{jk}(s) \geq 0, \]

\[ 0 \leq J_{jk}(s) = 1 - \frac{M_{jk}(s)}{H_{jk}(s)} \leq 1 \]

Table 1: Study of the non-negativeness of \( H_{jk}(s), M_{jk}(s) \) and \( d_{jk}(s) \) values for the protein family PF06850.

| s  | \( H_{jk}(s) \) | \( M_{jk}(s) \) | \( d_{jk}(s) \) |
|----|----------------|----------------|----------------|
| 0.1 | 0              | 0              | 0              |
| 0.3 | 0              | 0              | 0              |
| 0.5 | 0              | 0              | 0              |
| 0.7 | 0              | 0              | 0              |
| 0.9 | 0              | 0              | 0              |
| 1.0 | 0              | 718            | 16             |
| 1.2 | 0              | 1708           | 38             |
| 1.7 | 0              | 2351           | 61             |
| 1.9 | 0              | 2898           | 192            |
| 2.0 | 0              | 3139           | 369            |

The \( s \)-values corresponding to negative \( M_{jk}(s) \) values do not lead to a useful characterization of the Jaccard Entropy measure according to the inequality on
eq.(51) which is violated in this case and these \( s \)-values will not be taken into consideration. Other studies of the Entropy values and specially those of the behaviour of the association of entropies, will give additional restrictions on the feasible \( s \)-range. The scope of the present work does not allow an intensive study of these techniques of entropy association [2] which will then appear on a forthcoming contribution.

Table 2: Study of the non-negativeness of \( H_{jk}(s) \), \( M_{jk}(s) \) and \( d_{jk}(s) \) values for the protein family PF00135.

| \( s \) | \( H_{jk}(s) \) | \( M_{jk}(s) \) | \( d_{jk}(s) \) |
|---|---|---|---|
| 0.1 | 0 | 0 | 0 |
| 0.3 | 0 | 0 | 0 |
| 0.5 | 0 | 0 | 0 |
| 0.7 | 0 | 0 | 0 |
| 0.9 | 0 | 0 | 0 |
| 1.0 | 0 | 0 | 0 |
| 1.2 | 0 | 0 | 0 |
| 1.5 | 0 | 0 | 467 |
| 1.7 | 0 | 0 | 14509 |
| 1.9 | 0 | 0 | 19026 |
| 2.0 | 0 | 0 | 19451 |

Table 3: Study of the non-negativeness of \( H_{jk}(s) \), \( M_{jk}(s) \) and \( d_{jk}(s) \) values for the protein family PF00005.

| \( s \) | \( H_{jk}(s) \) | \( M_{jk}(s) \) | \( d_{jk}(s) \) |
|---|---|---|---|
| 0.1 | 0 | 0 | 0 |
| 0.3 | 0 | 0 | 0 |
| 0.5 | 0 | 0 | 0 |
| 0.7 | 0 | 0 | 0 |
| 0.9 | 0 | 0 | 0 |
| 1.0 | 0 | 0 | 0 |
| 1.2 | 0 | 0 | 0 |
| 1.5 | 0 | 33 | 4741 |
| 1.7 | 0 | 55 | 9679 |
| 1.9 | 0 | 65 | 12442 |
| 2.0 | 0 | 69 | 13203 |

The results on the previous three tables will clarify the idea of restriction of the \( s \)-values of entropy measures for obtaining a sound classification of families and clans on the Pfam database. We now announce that the non-negativeness of the values of \( H_{jk}(s) \), \( M_{jk}(s) \) and \( d_{jk}(s) \) is actually guaranteed if we restrict to \( s \leq 1 \) for all 1069 families which are classified into 68 clans and already characterized at section 2. In figures 2, 3, 4, we present the histograms of the Jaccard Entropy measures for some \( s \leq 1 \) values of the \( s \)-parameter.
Figure 2: Histograms of Jaccard Entropy for family PF06850.
Figure 3: Histograms of Jaccard Entropy for family PF00135.
Figure 4: Histograms of Jaccard Entropy for family PF00005.
We also present the curves corresponding to the Average Jaccard Entropy Measure (formula) for 99 families, a well-posed measure, with the restriction $s \leq 1$, which is given by

$$J(s, f) = \frac{2}{n(n-1)} \sum_{j} \sum_{k} J_{jk}(s, f)$$

Figure 5: Curves of the Average Jaccard Entropy measures for families PF00005, PF05673, PF13481, PF00135, PF06850, PF11339, PF02388, PF09924, PF13718.

4 A First Assessment of Protein Databases with Entropy Measures

As a motivation for future research to be developed in sections 6, 7, we now introduce the first application of the formulae derived on the previous sections in terms of a naive analysis of averages and standard deviations of Entropy measure distributions. This will be also the first attempt at classifying the
distribution of amino acids in a generic protein database. A robust approach to this research topic will be introduced and intensively analyzed on sections 6, 7 with the introduction of ANOVA statistics and the corresponding Hypothesis testing.

We then consider a Clan with $F$ families. The Havrda-Charvat entropy measure associated to a pair of columns on the representative $m \times n$ array of each family with a specified value of the $s$ parameter is given by

$$H_{jk}(s;f) = -\frac{1}{1-s} \left( 1 - \sum_a \sum_b (P_{jk}(a,b;f))^s \right)$$  \hspace{1cm} (54)

We can then define an average of these entropy measures for each family by

$$\langle H(s;f) \rangle = \frac{2}{n(n-1)} \sum_j \sum_k H_{jk}(s;f)$$  \hspace{1cm} (55)

We also consider the average value of the averages over the set of $F$ families:

$$\langle H(s) \rangle_F = \frac{1}{F} \sum_{f=1}^F \langle H(s;f) \rangle$$  \hspace{1cm} (56)

The Standard deviation of the Entropy measures $H_{jk}(s;f)$ with relation to the given average in eq.(55) can be written as:

$$\sigma(s;f) = \left( \frac{1}{n(n-1)} - 1 \right) \sum_j \sum_k \left( H_{jk}(s;f) - \langle H(s;f) \rangle \right)^2 \right)^{1/2} \hspace{1cm} (57)$$

and finally, the Standard deviation of the average $\langle H(s;f) \rangle$ with respect to the average $\langle H(s) \rangle_F$:

$$\sigma_F(s) = \left( \frac{1}{F-1} \sum_{f=1}^F \left( \langle H(s;f) \rangle - \langle H(s) \rangle_F \right) \right)^{1/2}$$  \hspace{1cm} (58)

We present in figs.6, 7 below the diagrams corresponding to formulae (55) and (57). We should stress that only Clans with a minimum of five families are considered.
Figure 6: The Average values of the Havrda-Charvat Entropy measures for the families of a selected set of Clans and eleven values of the $s$-parameter, eq.(55).
Figure 7: The standard deviation of the Havrda-Charvat Entropy measures with relation to the averages of these entropies for each family and eleven values of the $s$-parameter, eq.(57).
We now present the values of \( \langle H(s) \rangle_F \) and \( \sigma_F(s) \) for a selected number of Clans and eleven values of the \( s \)-parameter, according to eqs.(56) and (58).

Table 4: The average values and the standard deviation of the Average Havrda-Charvat Entropy measures for eleven values of the \( s \)-parameter and a selected set of 8 Clans.

| Clan number | \( s \) | \( \langle H(s) \rangle_F \) | \( \sigma_F(s) \) | Clan number | \( s \) | \( \langle H(s) \rangle_F \) | \( \sigma_F(s) \) |
|-------------|------|----------------|----------------|-------------|------|----------------|----------------|
| CL0020 (38 families) | 0.1 | 44.212 | 6.396 | CL0023 (119 families) | 0.1 | 40.401 | 4.667 |
|               | 0.3 | 23.375 | 3.057 |               | 0.3 | 22.831 | 2.295 |
|               | 0.5 | 12.992 | 1.492 |               | 0.5 | 12.785 | 1.161 |
|               | 0.7 | 7.645  | 0.746 |               | 0.7 | 7.546  | 0.605 |
|               | 0.9 | 4.784  | 0.384 |               | 0.9 | 4.741  | 0.326 |
|               | 1.0 | 3.875  | 0.278 |               | 1.0 | 3.846  | 0.242 |
|               | 1.2 | 2.661  | 0.150 |               | 1.2 | 2.648  | 0.137 |
|               | 1.5 | 1.680  | 0.063 |               | 1.5 | 1.676  | 0.062 |
|               | 1.7 | 1.311  | 0.038 |               | 1.7 | 1.309  | 0.038 |
|               | 1.9 | 1.062  | 0.023 |               | 1.9 | 1.061  | 0.024 |
|               | 2.0 | 0.967  | 0.018 |               | 2.0 | 0.966  | 0.019 |
| CL0020 (38 families) | 0.1 | 40.345 | 10.175 | CL0023 (119 families) | 0.1 | 46.084 | 6.790 |
|               | 0.3 | 20.908 | 4.984 |               | 0.3 | 24.260 | 3.224 |
|               | 0.5 | 11.733 | 2.510 |               | 0.5 | 13.417 | 1.561 |
|               | 0.7 | 6.982  | 1.305 |               | 0.7 | 7.535  | 0.772 |
|               | 0.9 | 4.422  | 0.704 |               | 0.9 | 4.886  | 0.392 |
|               | 1.0 | 3.604  | 0.525 |               | 1.0 | 3.949  | 0.282 |
|               | 1.2 | 2.504  | 0.392 |               | 1.2 | 2.701  | 0.149 |
|               | 1.5 | 1.606  | 0.144 |               | 1.5 | 1.696  | 0.080 |
|               | 1.7 | 1.263  | 0.093 |               | 1.7 | 1.320  | 0.035 |
|               | 1.9 | 1.030  | 0.063 |               | 1.9 | 1.067  | 0.020 |
|               | 2.0 | 0.940  | 0.053 |               | 2.0 | 0.970  | 0.016 |

In figs. 8a and 8b below, we present the graphs corresponding to Table 4. These results just point out a more elaborate formulation of the problem.

20
We now proceed to analyze a proposal (naive) for testing the robustness of the Clan concept. We will check if Pseudo-Clans which have the same number of families (a minimum of 05 families) of the corresponding Clans will have essentially different values of $\langle H(s) \rangle_F$ and $\sigma_F(s)$. The families to be associated with a Pseudo-Clan are obtained by sorting on the set of 1069 families and by withdrawal of the families already sorted. In Table 5 below we present the values $\langle H(s) \rangle_F$ and $\sigma_F(s)$ for the Pseudo-Clans obtained by the procedure described above.

The Figures 9a, 9b, do correspond to the comparison of data of Table 4 (Clans) with those of Table 5 (Pseudo-Clans). Clans are in red, Pseudo-Clans in blue.
Table 5: The average values and the standard deviation of the Average Havrda-Charvat Entropy measures for eleven values of the α-parameter and a selected set of 8 Pseudo-Clans.

| Pseudo-Clan number | s  | (H(α))F | σ(α)F |
|------------------|----|--------|--------|
| PCL0020 (38 families) | 0.1 | 42.381 | 8.224 |
|                   | 0.3 | 22.480 | 3.979 |
|                   | 0.5 | 12.542 | 1.972 |
|                   | 0.7 | 7.410  | 1.065 |
|                   | 0.9 | 4.657  | 0.528 |
|                   | 1.0 | 3.781  | 0.389 |
|                   | 1.2 | 2.807  | 0.216 |
|                   | 1.5 | 1.655  | 0.096 |
|                   | 1.7 | 1.295  | 0.059 |
|                   | 1.9 | 1.051  | 0.037 |
|                   | 2.0 | 0.958  | 0.030 |
| PCL0023 (119 families) | 0.1 | 41.023 | 9.007 |
|                   | 0.3 | 21.825 | 4.360 |
|                   | 0.5 | 12.219 | 2.183 |
|                   | 0.7 | 7.247  | 1.194 |
|                   | 0.9 | 4.573  | 0.582 |
|                   | 1.0 | 3.714  | 0.427 |
|                   | 1.2 | 2.573  | 0.240 |
|                   | 1.5 | 1.640  | 0.109 |
|                   | 1.7 | 1.286  | 0.068 |
|                   | 1.9 | 1.046  | 0.044 |
|                   | 2.0 | 0.954  | 0.037 |
| PCL0028 (41 families) | 0.1 | 42.716 | 7.433 |
|                   | 0.3 | 21.658 | 3.573 |
|                   | 0.5 | 12.641 | 1.756 |
|                   | 0.7 | 7.468  | 0.886 |
|                   | 0.9 | 4.692  | 0.460 |
|                   | 1.0 | 3.808  | 0.335 |
|                   | 1.2 | 2.625  | 0.183 |
|                   | 1.5 | 1.864  | 0.079 |
|                   | 1.7 | 1.302  | 0.048 |
|                   | 1.9 | 1.056  | 0.030 |
|                   | 2.0 | 0.962  | 0.024 |
| PCL0063 (92 families) | 0.1 | 41.870 | 8.134 |
|                   | 0.3 | 22.252 | 3.914 |
|                   | 0.5 | 12.440 | 1.929 |
|                   | 0.7 | 7.366  | 0.977 |
|                   | 0.9 | 4.638  | 0.510 |
|                   | 1.0 | 3.771  | 0.375 |
|                   | 1.2 | 2.603  | 0.207 |
|                   | 1.5 | 1.654  | 0.092 |
|                   | 1.7 | 1.295  | 0.057 |
|                   | 1.9 | 1.052  | 0.036 |
|                   | 2.0 | 0.959  | 0.030 |

| Pseudo-Clan number | s  | (H(α))F | σ(α)F |
|------------------|----|--------|--------|
| PCL0123 (96 families) | 0.1 | 42.042 | 4.484 |
|                   | 0.3 | 22.359 | 2.175 |
|                   | 0.5 | 12.508 | 1.082 |
|                   | 0.7 | 7.409  | 0.555 |
|                   | 0.9 | 4.667  | 0.293 |
|                   | 1.0 | 3.791  | 0.216 |
|                   | 1.2 | 2.618  | 0.120 |
|                   | 1.5 | 1.663  | 0.053 |
|                   | 1.7 | 1.301  | 0.032 |
|                   | 1.9 | 1.056  | 0.020 |
|                   | 2.0 | 0.962  | 0.016 |
| PCL0186 (29 families) | 0.1 | 40.888 | 7.179 |
|                   | 0.3 | 21.790 | 3.768 |
|                   | 0.5 | 12.220 | 1.891 |
|                   | 0.7 | 7.258  | 0.981 |
|                   | 0.9 | 4.585  | 0.529 |
|                   | 1.0 | 3.730  | 0.394 |
|                   | 1.2 | 2.582  | 0.227 |
|                   | 1.5 | 1.646  | 0.109 |
|                   | 1.7 | 1.290  | 0.071 |
|                   | 1.9 | 1.048  | 0.049 |
|                   | 2.0 | 0.956  | 0.041 |
| PCL0192 (26 families) | 0.1 | 42.756 | 9.361 |
|                   | 0.3 | 22.666 | 4.535 |
|                   | 0.5 | 12.635 | 2.253 |
|                   | 0.7 | 7.458  | 1.151 |
|                   | 0.9 | 4.681  | 0.608 |
|                   | 1.0 | 3.797  | 0.448 |
|                   | 1.2 | 2.615  | 0.250 |
|                   | 1.5 | 1.667  | 0.113 |
|                   | 1.7 | 1.296  | 0.070 |
|                   | 1.9 | 1.051  | 0.046 |
|                   | 2.0 | 0.958  | 0.037 |
| PCL0236 (21 families) | 0.1 | 41.551 | 7.476 |
|                   | 0.3 | 22.090 | 4.591 |
|                   | 0.5 | 12.357 | 1.763 |
|                   | 0.7 | 7.322  | 0.887 |
|                   | 0.9 | 4.614  | 0.459 |
|                   | 1.0 | 3.752  | 0.334 |
|                   | 1.2 | 2.595  | 0.181 |
|                   | 1.5 | 1.652  | 0.077 |
|                   | 1.7 | 1.294  | 0.046 |
|                   | 1.9 | 1.051  | 0.028 |
|                   | 2.0 | 0.959  | 0.022 |

From the figures and tables above we can see that the region $s \leq 1$ leads to a better characterization of the Entropy measures distributions on protein databases.

For completeness, we list some useful formulae obtained from eqs. (55), (56),
which help to predict the profile of the curves above:

\[
\langle H(s; f) \rangle = -\frac{1}{1-s} \left( 1 - \frac{2}{n(n-1)} \sum_{a,b,j,k} e^{-s|\log P_{jk}(a,b,f)|} \right)
\]

\[
\langle H(s) \rangle_F = -\frac{1}{1-s} \left( 1 - \frac{2}{Fn(n-1)} \sum_{f,a,b,j,k} e^{-s|\log P_{jk}(a,b,f)|} \right)
\]

\[
\sigma_F(s) = \frac{2(F-1)^{1/2}}{Fn(n-1)} \left( \sum_{f=1}^{F} \left( \sum_{a,b,j,k} e^{-s|\log P_{jk}(a,b,f)|} \right)^2 \right)^{1/2}
\]

5 The treatment of data with the Maple Computing system and its inadequacy for calculating Joint probabilities of occurrence. Alternative systems.

In this section, we specifically study the performance of the Maple system and an example of alternative computing system, the Perl system, for calculating the simple and joint probabilities of occurrences of amino acids. We also use these two systems for calculating 19 s-power values of these probabilities. We now select the family PF06850 in order to get an idea of the CPU and real times which are necessary for calculating the probabilities and their powers for the set of 1069 families. We start the calculation by adopting the Maple system version 18. There are some comments to be made on the construction of a computational code for calculating joint probabilities. This will be done in detail at the end of CPU and real times for the calculation of the simple and joint probabilities by using the developed code. The table below will repeat the times for calculating \(200 \times 20 = 4 \times 10^3\) and \(200 \times \frac{399}{2} \times 20 \times 20 = 7.96 \times 10^6\) of simple and joint probability values, respectively, for the PF06850 Pfam family.

| Maple System, version 18 | \(t_{CPU} (sec)\) | \(t_R (sec)\) |
|-------------------------|-----------------|----------------|
| Simple probabilities    | 0.527           | 0.530          |
| Joint probabilities     | 5073.049        | 4650.697       |

After calculating all values of the probabilities \(p_j(a)\) and \(P_{jk}(a,b)\), we can proceed to evaluate the powers \((p_j(a))^s\) and \((P_{jk}(a,b))^s\) for 19 s-values. Our aim will be to use these values for calculating the Entropy Measures according to eqs.(32), (33). It should be noticed that the values of \(p_j(a)\) and \(P_{jk}(a,b)\), have to be calculated only once by using a specific computational code already
referred on this work. Nevertheless, the use of the code for calculating the joint probabilities associated to 1069 protein families is the hardest of all calculations to be undertaken and it takes too much time. These probabilities once calculated should be grouped in sets of 400 values each corresponding to a pair of columns \( j, k \) among the \( \frac{n(n-1)}{2} \) feasible ones and the calculating of entropy value associated to this pair of columns \( j, k \). Given a \( s \)-value and after calculating the entropy of this first pair \( H_{12} \) as a function of the 400 variables \((P_{jk}(a,b))^s\), \( j \neq 1, k \neq 2 \) and he/she will proceeds to calculate again all values of \( \frac{n(n-1)}{2} \times (20)^2 \) in order to extract another value of joint probability for calculating the corresponding entropy value. This seems to be associated to the unknowing of the concepts of a function of several variables, unfortunately. After circumventing these mistakes coming from a bad educational formation, we succeed at keeping all calculated values of the probabilities and we then proceed to the calculation of the powers \((p_j(a))^s\), \((P_{jk}(a,b))^s\) of these values and the corresponding entropy measures. In tables 7, 8, 9, 10 below, we report all these calculations for 19 values of the \( s \)-parameter.

Table 7: CPU and real times for the calculation of 19 \( s \)-powers of simple probabilities of occurrence associated with the protein family PF06850.

| Maple System, version 18, \( s \)-powers of probability \((p_j(a))^s\) | \( t_{CPU} \) (sec) | \( t_R \) (sec) |
|---|---|---|
| 0.1 | 0.263 | 0.358 |
| 0.2 | 0.137 | 0.145 |
| 0.3 | 0.268 | 0.277 |
| 0.4 | 0.139 | 0.153 |
| 0.5 | 0.240 | 0.219 |
| 0.6 | 0.144 | 0.157 |
| 0.7 | 0.276 | 0.254 |
| 0.8 | 0.144 | 0.157 |
| 0.9 | 0.264 | 0.235 |
| 1.0 | 0.088/0.151 | 0.095/0.307 |
| 2.0 | 0.153 | 0.095 |
| 3.0 | 0.128 | 0.131 |
| 4.0 | 0.148 | 0.141 |
| 5.0 | 0.096 | 0.144 |
| 6.0 | 0.148 | 0.167 |
| 7.0 | 0.148 | 0.155 |
| 8.0 | 0.181 | 0.094 |
| 9.0 | 0.104 | 0.092 |
| 10.0 | 0.104 | 0.100 |
| Total | 3.173 | 3.164 |

The last row in tables 7, 8, includes the times necessary for calculating the probabilities of table 6.
Table 8: CPU and real times for the calculation of 19 $s$-powers of joint probabilities of occurrence associated with the protein family PF06850.

| $s$ | $t_{CPU}$ (sec) | $t_{R}$ (sec) |
|-----|----------------|---------------|
| 0.1 | 390.432 | 206.646 |
| 0.2 | 382.887 | 202.282 |
| 0.3 | 401.269 | 210.791 |
| 0.4 | 416.168 | 216.993 |
| 0.5 | 427.572 | 221.541 |
| 0.6 | 430.604 | 223.227 |
| 0.7 | 421.904 | 218.484 |
| 0.8 | 434.888 | 224.267 |
| 0.9 | 431.948 | 223.023 |
| 1.0 | 442.933/482.612 | 224.731/259.301 |
| 2.0 | 176.212 | 147.455 |
| 3.0 | 234.100 | 174.853 |
| 4.0 | 289.184 | 181.552 |
| 5.0 | 327.740 | 178.117 |
| 6.0 | 334.800 | 194.691 |
| 7.0 | 349.064 | 195.258 |
| 8.0 | 361.304 | 195.437 |
| 9.0 | 386.217 | 197.150 |
| 10.0 | 397.276 | 197.868 |
| Total | 7036.502 | 3834.366 |

We are then able to proceed to the calculation of the corresponding Havrda-Charvat entropy measures, $H_j(s)$, $H_{jk}(s)$: The results for 19 $s$-values are given in tables 9, 10 below.

The total time for calculating all the Havrda-Charvat Entropy measure content of probabilities of occurrence of amino acids on a specific family is given in table 11 below.

From inspections of table 11, we realize that the Total CPU and real times for calculating the Havrda-Charvat entropies $H_j(s)$, of simple probabilities of occurrence $p_j(a)$ are obtained by summing up the total time results from tables 6, 7 and 9. For the Havrda-Charvat entropies $H_{jk}(s)$, we have to sum up the total times at table 6, 8 and 10. We take for granted that the times for calculating the Entropy Measure content of each family will not differ too much and the results 3rd and 5th rows of table 11 are obtained by multiplying by 1069 — the number of families in the sample space.

The results of table 11 suggest the inadequacy of the Maple computing system for analyzing the Entropy measure content of an example of protein database. We have restricted ourselves to operate with usual operating sys-
Table 9: CPU and real times for the calculation of the Entropy measures $H_j(s)$ for the protein family PF06850.

| s  | $t_{CPU}$ (sec) | $t_R$ (sec) |
|----|----------------|-------------|
| 0.1| 0.148          | 0.261       |
| 0.2| 0.084          | 0.153       |
| 0.3| 0.120          | 0.189       |
| 0.4| 0.124          | 0.198       |
| 0.5| 0.160          | 0.299       |
| 0.6| 0.092          | 0.139       |
| 0.7| 0.150          | 0.199       |
| 0.8| 0.137          | 0.175       |
| 0.9| 0.120          | 0.166       |
| 1.0| 0.192/0.175    | 0.339/0.159 |
| 2.0| 0.144          | 0.099       |
| 3.0| 0.147          | 0.105       |
| 4.0| 0.084          | 0.101       |
| 5.0| 0.136          | 0.070       |
| 6.0| 0.096          | 0.119       |
| 7.0| 0.115          | 0.078       |
| 8.0| 0.120          | 0.109       |
| 9.0| 0.133          | 0.080       |
|10.0| 0.132          | 0.133       |
|Total| 2.443         | 3.012       |

tems, Linux or OSX, on laptops. We have also worked with the alternative Perl computing system. The Maple computing system has an “array” structure which is very effective for doing calculations which require a knowledge of mathematical methods. On the contrary, the alternative Perl computing system has a “hash” structure as was emphasized in the 2nd section and it operates very well elementary operations with very large numbers. It is essential the comparison of a senior erudite which is largely conversant with a large amount of mathematical methods versus a “genius” brought to fame by media, who is able only to multiply in a very fast way, numbers of many digits.

In order to specify the probabilities of computational configurations with the usual desktops and laptops, we list below some of them which have been used in the present work. The computing systems were the Maple ($M$) and Perl ($P$), the operating systems, the Linux ($L$) and Mac OSX ($O$) and the structures: The Array I ($A_I$), Array II ($A_{II}$) and Hash ($H$) (page 8, section 2). The available computational configurations to undertake the task of assessment of protein databases with Entropy measures could be listed as:

1. $MLA_I$ — Maple, Linux, Array I
2. **POH** — Perl, OSX, Hash

3. **PLA** — Perl, Linux, Array II

4. **POA** — Perl, OSX, Array II

Table 10: CPU and real times for the calculation of the Entropy measures $H_{jk}(s)$ for the protein family PF06850.

| s   | $t_{CPU}$ (sec) | $t_R$ (sec) |
|-----|----------------|-------------|
| 0.1 | 156.332        | 133.242     |
| 0.2 | 160.797        | 136.706     |
| 0.3 | 169.024        | 140.960     |
| 0.4 | 176.824        | 147.853     |
| 0.5 | 184.120        | 150.163     |
| 0.6 | 190.304        | 154.058     |
| 0.7 | 196.633        | 157.750     |
| 0.8 | 205.940        | 164.101     |
| 0.9 | 215.559        | 169.549     |
| 1.0 | 253.648/124.501| 204.634/148.993|
| 2.0 | 141.148        | 184.030     |
| 3.0 | 158.536        | 167.173     |
| 4.0 | 173.136        | 181.282     |
| 5.0 | 197.680        | 238.723     |
| 6.0 | 215.000        | 111.476     |
| 7.0 | 145.257        | 115.221     |
| 8.0 | 156.848        | 122.957     |
| 9.0 | 157.300        | 126.233     |
| 10.0| 166.399        | 135.080     |
| Total| 3420.485       | 2941.791    |

Table 11: Total CPU and real times for calculating the Entropy measure content of a family PF06850 and approximations for Grand Total of all sample space.

| Maple System, version 18, Entropy Measures $H_{jk}(s)$ | $H_{jk}(s)$ — 19 $s$-values | $H_{jk}(s)$ — 19 $s$-values |
|------------------------------------------------------|-----------------------------|-----------------------------|
| Total CPU time (family PF06850)                       | 0.527±3.173±2.443           | 5.073±9.036±5.502+           |
|                                                        | =6.143 sec                  | 3.420±5.530±0.036 sec        |
| Grand Total CPU time (1069 families)                  | 6.566.867 sec               | 16.601.608.484 sec          |
|                                                        | =1.824 hs                   | =192.148 days               |
| Total Real time (family PF06850)                      | 0.530±3.164±3.012           | 4.650±6.97±3.834±366+       |
|                                                        | =6.766 sec                  | 2.941.791±11.426.854 sec    |
| Grand Total Real time (1069 families)                 | 7.168.714 sec               | 12.215.306.926 sec          |
|                                                        | =1.991 hs                   | =141.381 days               |
The following table will display a comparison of the CPU and real times for the calculation of 19 s-values of the joint probabilities \( (P_{jk}(a,b))^s \) for the protein family PF06850 by the four configurations nominated above. It should be stressed that we are here comparing the times for calculating the s-power with the values of the probabilities themselves previously calculated and kept on a file.

We can check that the times on table 12 seem to be generically ordered as

\[
    t_{MLA_I} > t_{POA_H} > t_{PLA_H} > t_{POH} \tag{62}
\]

From this ordering of computing times, we are then able to consider that the inconvenience of using the “hash” structure which has been emphasized on section 2, was not circumvented by working with a modified array structure \((A_H\) instead of \(H\)), at least for the Mac Pro machine used in these calculations. We do not also know if this machine has been even used with an “overload” of programs from the assumed part-time job of the experimenter (maybe 99.99% time job?). Anyhow, the usual Hash structure of Perl computing system has delayed the calculation of the Entropy Measures and even with the help of the modified \(A_H\) structure, it does not succeed at computing with this structure if operated on a OSX computing system.

On the other hand, the configuration \(MLA_I\) could be chosen for parallelizing the respective adopted code in a work to be done with supercomputer facilities. If we try to avoid this kind of computational facility, in the belief that the problem of classifying the distribution of amino acids of a protein database in terms of Entropy Measures could be treated with less powerful but very objective “weapons”, we should try to look for very fast laptop machines instead, by working with a Linux operating system, a Perl computing system and a modified array structure. This means that it would be worthwhile the continuation of the present work with the \(PLA_H\) configuration. This is now in progress and will be published elsewhere.

We summarize the conclusions commented above on tables 13–16 below for the calculation of CPU and Real times of 19 s-powers of joint probabilities \((P_{jk}(a,b))^s\) and the corresponding values of Havrda-Charvat entropy measures. The necessary times for calculating the joint probabilities themselves has not been taken into consideration. It would be very useful to make a comparison of the results of table 10 with those on tables 14, 16, and table 12 with tables 13, 15 as well.

As a last remark of this section, we shall take into consideration, the restrictions of \(s \leq 1\) for working with Jaccard Entropy measures and we calculate the total CPU and real times for the set of s-values: \(s = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\). The results are presented in table 17 below for the \(PLA_H\) configuration and the calculation of the Havrda-Charvat entropies.
Table 12: A comparison of calculation times (CPU and real) for 19 s-powers of joint probabilities only of PF06850 protein family, using the four configurations MLA, POAII, PLAII, POH.

| s  | t_CPU (sec) | t_R (sec) |
|----|-------------|-----------|
|    | MLAI | POAI | PLAII | POH | MLAI | POAI | PLAII | POH |
| 0.1 | 390.432 | 31.478 | 41.633 | 24.849 | 206.646 | 88.527 | 83.139 | 26.862 |
| 0.2 | 382.887 | 31.711 | 26.483 | 25.090 | 202.282 | 80.624 | 53.384 | 26.994 |
| 0.3 | 401.269 | 31.726 | 25.286 | 24.751 | 210.791 | 80.519 | 50.849 | 27.385 |
| 0.4 | 416.168 | 31.448 | 26.138 | 26.345 | 216.993 | 79.032 | 51.409 | 27.087 |
| 0.5 | 427.572 | 32.860 | 25.822 | 26.726 | 221.541 | 93.048 | 51.334 | 27.528 |
| 0.6 | 430.604 | 33.444 | 27.013 | 25.021 | 223.227 | 102.317 | 52.849 | 25.408 |
| 0.7 | 421.904 | 31.053 | 25.814 | 25.011 | 218.484 | 79.255 | 51.466 | 26.414 |
| 0.8 | 434.888 | 31.526 | 26.725 | 25.183 | 224.267 | 80.469 | 53.388 | 25.668 |
| 0.9 | 431.948 | 31.482 | 26.895 | 25.409 | 223.023 | 80.002 | 53.579 | 25.536 |
| 1.0 | 442.933 | 32.056 | 25.990 | 25.090 | 224.731 | 80.917 | 51.687 | 25.640 |
| 2.0 | 176.212 | 32.638 | 27.089 | 26.012 | 147.454 | 80.751 | 54.384 | 26.960 |
| 3.0 | 234.100 | 31.892 | 25.766 | 24.498 | 174.853 | 85.853 | 51.843 | 24.717 |
| 4.0 | 284.184 | 31.662 | 26.515 | 25.251 | 181.552 | 91.837 | 52.636 | 25.718 |
| 5.0 | 327.740 | 32.295 | 27.516 | 24.925 | 178.117 | 87.486 | 54.908 | 25.814 |
| 6.0 | 334.800 | 32.674 | 28.126 | 25.440 | 194.691 | 86.569 | 54.611 | 25.847 |
| 7.0 | 349.064 | 31.674 | 23.908 | 26.389 | 195.258 | 86.215 | 49.262 | 27.745 |
| 8.0 | 361.304 | 33.105 | 26.020 | 25.100 | 195.437 | 116.601 | 51.889 | 26.735 |
| 9.0 | 386.217 | 31.881 | 26.208 | 24.783 | 197.150 | 81.372 | 53.114 | 25.155 |
| 10.0 | 397.276 | 32.269 | 26.125 | 24.979 | 197.868 | 87.963 | 52.541 | 26.504 |
| Total | 7030.502 | 411.374 | 515.072 | 480.867 | 3834.365 | 1689.357 | 1028.655 | 508.147 |
| Total (1069 families) | 7,522,020.640 | 653,588.806 | 550,011.968 | 514,046.813 | 4,098,396.190 | 1,763,162.630 | 1,099,632.200 | 554,157.143 |

CPU: 87.067 days, 7.565 days, 6.373 days, 5.950 days, 47.441 days, 20.407 days, 12.727 days, 6.188 days.
Table 13: Calculation of CPU and Real times of 19 $s$-powers of joint probabilities $(P_{jk}(a,b))^s$ measures for 06 families from 03 Clans with the POA$_H$ configuration.

| s  | CL0028       | PF00850 | t$_{CPU}$ [sec] | t$_g$ [sec] | CL00135 | PF00005 | t$_{CPU}$ [sec] | t$_g$ [sec] | CL0023      | PF13481 | t$_{CPU}$ [sec] | t$_g$ [sec] | CL0257      | PF02388 | t$_{CPU}$ [sec] | t$_g$ [sec] | CL0924      | PF09924 | t$_{CPU}$ [sec] | t$_g$ [sec] |
|----|--------------|---------|----------------|-------------|---------|---------|----------------|-------------|--------------|---------|----------------|-------------|--------------|---------|----------------|-------------|--------------|---------|----------------|-------------|
| 0.1| 35.478       | 88.527  | 46.747         | 130.014     | 41.475  | 169.679 | 44.187         | 171.636     | 44.716       | 242.376 | 46.177         | 288.259     | 0.2         | 31.711  | 80.624         | 42.157       | 90.687       | 36.893       | 78.893       | 38.567       | 88.754       | 37.088       | 88.091       | 40.709       | 148.371     |
| 0.2| 31.726       | 80.519  | 39.957         | 82.240      | 36.920  | 97.270  | 36.800         | 81.613      | 37.150       | 88.664  | 38.986         | 119.781     | 0.3         | 31.448  | 79.032         | 41.737       | 87.934       | 36.252       | 80.428       | 38.500       | 78.757       | 35.592       | 80.337       | 36.773       | 87.217       |
| 0.5| 32.860       | 93.048  | 41.130         | 89.997      | 38.203  | 93.595  | 37.618         | 80.179      | 35.117       | 78.476  | 35.534         | 82.528      | 0.6         | 33.944  | 102.317        | 41.417       | 120.420      | 37.143       | 81.289       | 38.387       | 81.667       | 45.900       | 501.222      | 35.830       | 83.439       |
| 0.7| 31.053       | 79.255  | 40.122         | 78.531      | 36.386  | 84.421  | 43.216         | 562.659     | 43.452       | 259.961 | 35.261         | 72.605      | 0.8         | 31.526  | 80.469         | 41.516       | 118.519      | 36.955       | 79.543       | 41.862       | 148.028      | 35.047       | 78.469       | 35.718       | 85.142       |
| 0.9| 31.482       | 80.002  | 41.386         | 81.747      | 36.811  | 81.025  | 35.518         | 78.567      | 35.540       | 74.570  | 40.534         | 204.673     | 1.0         | 32.056  | 80.917         | 40.724       | 80.958       | 37.234       | 80.587       | 36.610       | 87.848       | 36.353       | 78.767       | 39.095       | 125.292      |
| 2.0| 32.638       | 80.751  | 40.701         | 79.667      | 38.452  | 79.316  | 36.916         | 111.199     | 36.658       | 81.415  | 40.415         | 182.552     | 3.0         | 31.892  | 85.853         | 40.822       | 79.293       | 38.223       | 80.688       | 37.356       | 84.312       | 36.658       | 81.415       | 39.774       | 157.090      |
| 4.0| 31.662       | 91.837  | 41.208         | 79.825      | 37.936  | 98.905  | 37.000         | 86.906      | 36.012       | 77.741  | 38.794         | 140.857     | 5.0         | 32.295  | 85.486         | 41.059       | 82.191       | 38.293       | 83.289       | 36.245       | 80.873       | 35.308       | 77.225       | 39.927       | 147.968      |
| 6.0| 32.764       | 86.569  | 41.215         | 86.311      | 37.601  | 82.375  | 36.395         | 97.307      | 35.748       | 76.573  | 39.327         | 154.829     | 7.0         | 31.674  | 86.215         | 41.290       | 88.714       | 38.341       | 89.991       | 36.724       | 95.148       | 36.194       | 75.341       | 39.826       | 153.204      |
| 8.0| 33.105       | 116.604 | 40.984         | 83.157      | 37.756  | 81.073  | 40.524         | 105.466     | 36.716       | 82.921  | 41.056         | 175.943     | 9.0         | 31.881  | 81.372         | 41.469       | 113.561      | 38.748       | 82.499       | 37.874       | 94.981       | 40.341       | 121.311      | 39.847       | 144.595      |
| 10.0| 32.269       | 87.963  | 41.466         | 89.364      | 37.807  | 81.043  | 37.134         | 86.065      | 40.053       | 136.243 | 40.333         | 173.395     | Total       | 611.374 | 1649.355       | 787.447      | 1743.132     | 717.438      | 1676.929     | 729.433      | 2303.885     | 719.843      | 2381.018     | 743.916      | 2725.140     |
Table 14: Calculation of CPU and Real times of Havrd-Charvat Entropy measures for 06 families from 03 Clans with the POA configuration.

| s   | CL028 | PF00050 | CL023 | PF13481 | PF02888 | CL0257 | PF09924 |
|-----|-------|---------|-------|---------|---------|--------|---------|
|     | t_{CPU} (sec) | t_R (sec) | t_{CPU} (sec) | t_R (sec) | t_{CPU} (sec) | t_R (sec) | t_{CPU} (sec) | t_R (sec) |
| 0.1 | 19.451 | 26.196 | 22.630 | 56.287 | 25.411 | 121.508 | 18.968 | 32.706 | 23.340 | 80.826 | 22.502 | 77.826 |
| 0.2 | 19.245 | 24.901 | 22.851 | 69.992 | 23.362 | 65.508 | 18.810 | 26.815 | 23.346 | 87.098 | 23.330 | 90.684 |
| 0.3 | 19.582 | 26.091 | 21.962 | 60.960 | 22.598 | 72.363 | 18.602 | 26.028 | 20.012 | 43.734 | 21.929 | 48.947 |
| 0.4 | 20.194 | 31.418 | 22.211 | 54.562 | 23.448 | 63.309 | 19.176 | 30.482 | 21.218 | 67.817 | 22.182 | 45.480 |
| 0.5 | 20.162 | 31.653 | 23.163 | 67.391 | 22.153 | 55.173 | 19.281 | 34.495 | 21.037 | 55.237 | 23.200 | 62.384 |
| 0.6 | 20.941 | 34.979 | 23.961 | 62.158 | 22.342 | 57.081 | 20.794 | 44.558 | 21.087 | 54.990 | 24.477 | 62.421 |
| 0.7 | 20.805 | 34.057 | 23.679 | 82.669 | 19.587 | 35.750 | 19.982 | 41.314 | 20.375 | 32.699 | 22.398 | 66.438 |
| 0.8 | 20.900 | 34.146 | 22.787 | 60.924 | 19.081 | 34.116 | 18.890 | 28.054 | 20.167 | 32.685 | 23.056 | 61.376 |
| 0.9 | 20.909 | 34.568 | 22.808 | 54.890 | 19.030 | 33.258 | 18.894 | 29.712 | 19.422 | 26.914 | 23.543 | 65.099 |
| 1.0 | 21.353 | 35.024 | 22.860 | 51.308 | 19.789 | 32.218 | 19.869 | 37.922 | 21.076 | 35.774 | 22.528 | 52.209 |
| 2.0 | 20.505 | 32.920 | 21.080 | 46.921 | 19.672 | 33.778 | 19.083 | 62.061 | 22.530 | 82.336 | 21.783 | 51.656 |
| 3.0 | 23.923 | 58.998 | 22.020 | 47.298 | 18.788 | 31.926 | 19.097 | 35.389 | 24.210 | 83.371 | 21.686 | 69.905 |
| 4.0 | 24.622 | 68.692 | 21.954 | 56.909 | 18.556 | 26.512 | 19.208 | 32.014 | 24.300 | 112.613 | 21.458 | 69.901 |
| 5.0 | 24.221 | 60.107 | 22.131 | 58.975 | 19.424 | 33.185 | 18.231 | 25.808 | 24.199 | 95.807 | 22.011 | 55.109 |
| 6.0 | 24.475 | 64.843 | 22.911 | 72.004 | 20.582 | 38.006 | 18.330 | 25.959 | 22.741 | 61.516 | 21.857 | 67.976 |
| 7.0 | 24.593 | 70.336 | 22.779 | 51.309 | 20.564 | 38.390 | 19.583 | 30.719 | 23.222 | 88.997 | 23.018 | 85.054 |
| 8.0 | 25.613 | 83.423 | 20.058 | 34.861 | 20.460 | 35.589 | 19.977 | 37.130 | 23.825 | 110.933 | 24.474 | 90.322 |
| 9.0 | 24.139 | 73.973 | 21.418 | 44.709 | 19.274 | 32.129 | 18.871 | 29.089 | 24.207 | 106.471 | 23.349 | 74.724 |
| 10.0 | 25.785 | 101.370 | 22.183 | 46.878 | 23.625 | 87.442 | 22.652 | 64.895 | 23.779 | 86.225 | 21.802 | 57.444 |
| Total | 421.418 | 927.405 | 427.442 | 1075.005 | 397.746 | 927.301 | 368.298 | 675.260 | 424.153 | 1345.973 | 428.533 | 1204.985 |
Table 15: Calculation of CPU and Real times of 19 $s$-powers of joint probabilities $(P_{ij}(a,b))^s$ measures for 06 families from 03 Clans with the \(P|A\mid_B\) configuration.

|          | PP00650 | PP00135 | PP00005 | PP13451 | PP02588 | PP09924 | CL0028 | CL0035 | CL0023 | CL0257 | CL0257 |
|----------|---------|---------|---------|---------|---------|---------|--------|--------|--------|--------|--------|
| $s$      | t_CPU | t_R  | t_CPU | t_R  | t_CPU | t_R  | t_CPU | t_R  | t_CPU | t_R  | t_CPU | t_R  |
|----------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0.1      | 41.633 | 83.139 | 43.135 | 83.901 | 44.008 | 84.655 | 26.613 | 52.365 | 27.472 | 53.350 | 43.132 | 83.249 |
| 0.2      | 26.483 | 53.384 | 26.373 | 50.442 | 26.610 | 52.904 | 26.599 | 51.822 | 43.199 | 83.497 | 25.762 | 50.203 |
| 0.3      | 25.286 | 50.899 | 26.644 | 50.992 | 29.965 | 53.250 | 24.966 | 48.846 | 25.498 | 50.303 | 25.904 | 50.737 |
| 0.4      | 26.138 | 51.409 | 25.255 | 48.576 | 26.686 | 52.482 | 27.769 | 53.837 | 25.753 | 51.013 | 25.684 | 50.371 |
| 0.5      | 25.822 | 51.837 | 26.041 | 50.492 | 26.648 | 52.171 | 25.026 | 48.927 | 25.727 | 49.887 | 25.513 | 48.793 |
| 0.6      | 27.013 | 52.889 | 25.778 | 50.324 | 24.912 | 49.340 | 26.062 | 51.312 | 26.066 | 51.417 | 25.435 | 49.086 |
| 0.7      | 25.814 | 51.496 | 25.490 | 49.954 | 25.284 | 50.368 | 23.698 | 48.134 | 25.723 | 50.760 | 25.822 | 51.122 |
| 0.8      | 27.725 | 53.388 | 27.274 | 50.865 | 25.456 | 50.870 | 25.816 | 51.196 | 26.883 | 52.511 | 29.837 | 57.367 |
| 0.9      | 26.895 | 53.579 | 23.749 | 47.296 | 26.237 | 51.725 | 28.441 | 54.743 | 25.237 | 50.532 | 27.951 | 54.289 |
| 1.0      | 25.990 | 51.687 | 27.225 | 54.384 | 26.014 | 51.969 | 27.148 | 52.791 | 23.672 | 48.547 | 26.314 | 51.932 |
| 2.0      | 27.890 | 54.384 | 26.237 | 50.335 | 27.756 | 54.761 | 26.424 | 52.503 | 24.811 | 49.859 | 25.846 | 51.125 |
| 3.0      | 25.766 | 51.843 | 27.342 | 52.508 | 25.135 | 50.954 | 27.753 | 53.650 | 25.072 | 49.142 | 25.727 | 50.990 |
| 4.0      | 26.515 | 52.636 | 25.716 | 50.531 | 25.106 | 50.449 | 24.871 | 50.279 | 25.674 | 51.536 | 26.897 | 52.227 |
| 5.0      | 27.516 | 54.908 | 26.002 | 50.390 | 24.666 | 49.463 | 24.973 | 47.554 | 25.675 | 51.032 | 27.810 | 53.620 |
| 6.0      | 28.126 | 54.611 | 27.441 | 53.032 | 26.014 | 52.209 | 25.676 | 50.426 | 26.235 | 51.375 | 26.180 | 51.346 |
| 7.0      | 23.908 | 49.262 | 26.812 | 51.556 | 24.696 | 49.783 | 25.519 | 50.741 | 25.863 | 50.995 | 25.593 | 50.611 |
| 8.0      | 26.020 | 51.889 | 25.431 | 49.135 | 26.757 | 52.518 | 26.369 | 49.668 | 26.401 | 52.682 | 25.084 | 50.596 |
| 9.0      | 26.200 | 53.114 | 25.856 | 50.093 | 25.803 | 51.233 | 24.628 | 47.880 | 25.379 | 51.014 | 27.949 | 52.891 |
| 10.0     | 26.125 | 52.561 | 24.963 | 47.402 | 24.369 | 48.065 | 42.270 | 53.201 | 24.878 | 49.926 | 24.553 | 48.016 |
| **Total**| 515.072 | 1028.655 | 511.741 | 992.205 | 569.127 | 1069.049 | 509.601 | 999.865 | 505.198 | 999.378 | 516.103 | 1009.971 |
| s  | CL0028 | CL00135 | CL0005 | CL13481 | CL0257 | CL0023 |
|----|--------|---------|--------|----------|--------|--------|
| 0.1| 21.219 | 26.195  | 18.698 | 23.442   | 19.383 | 24.675 |
| 0.2| 22.694 | 28.014  | 21.109 | 28.536   | 20.338 | 25.665 |
| 0.3| 19.076 | 24.139  | 20.692 | 25.665   | 21.934 | 27.466 |
| 0.4| 21.552 | 27.442  | 19.941 | 25.021   | 18.794 | 24.142 |
| 0.5| 31.952 | 88.406  | 21.161 | 26.374   | 19.123 | 24.309 |
| 0.6| 20.195 | 25.362  | 21.090 | 26.371   | 21.426 | 30.303 |
| 0.7| 19.168 | 23.924  | 20.980 | 26.482   | 21.440 | 27.129 |
| 0.8| 21.982 | 27.362  | 19.760 | 24.681   | 20.662 | 25.948 |
| 0.9| 21.356 | 27.235  | 21.251 | 26.550   | 19.518 | 24.880 |
| 1  | 20.206 | 25.608  | 20.914 | 26.173   | 20.726 | 25.877 |
| 2  | 19.435 | 24.882  | 20.690 | 25.675   | 20.798 | 26.326 |
| 3  | 20.549 | 25.646  | 19.988 | 25.479   | 20.629 | 25.516 |
| 4  | 19.528 | 24.693  | 20.649 | 25.610   | 19.426 | 24.617 |
| 5  | 19.698 | 24.824  | 21.076 | 26.468   | 19.865 | 24.753 |
| 6  | 20.809 | 26.319  | 20.352 | 25.914   | 19.507 | 24.962 |
| 7  | 20.287 | 25.911  | 21.073 | 26.459   | 19.482 | 24.861 |
| 8  | 21.427 | 26.885  | 19.494 | 24.421   | 20.107 | 25.228 |
| 9  | 21.623 | 27.335  | 21.554 | 26.517   | 19.239 | 24.529 |
| 10 | 20.815 | 26.379  | 21.127 | 26.630   | 19.927 | 25.340 |

Total: 403.461, 557.294, 391.524, 490.466, 382.324, 485.946, 388.906, 489.634, 383.716, 484.707, 392.368, 493.143
Table 17: Calculation of CPU and Real times of Havrda-Charvat Entropy measures for 06 families from 03 Clans with parameters $0 < s \leq 1$ and the $PLA_H$ configuration.

| $s$ | PF06850 | PF00135 | PF00005 | PF13481 | PF02388 | PF09924 |
|-----|---------|---------|---------|---------|---------|---------|
|     | CPU     | R       | CPU     | R       | CPU     | R       | CPU     | R       | CPU     | R       | CPU     | R       |
| 0.1 | 21.219  | 26.895  | 18.698  | 23.442  | 19.383  | 24.675  | 21.351  | 26.966  | 19.059 | 24.560  | 21.451  | 26.829  |
| 0.2 | 22.604  | 28.014  | 21.109  | 26.536  | 20.336  | 25.665  | 19.555  | 24.562  | 19.370  | 24.681  | 22.504  | 27.855  |
| 0.3 | 19.056  | 24.130  | 20.692  | 25.663  | 21.934  | 27.466  | 21.084  | 26.236  | 19.317  | 24.934  | 19.684  | 24.732  |
| 0.4 | 21.552  | 27.442  | 19.981  | 25.021  | 18.794  | 24.142  | 21.065  | 26.485  | 20.878  | 26.291  | 21.947  | 26.979  |
| 0.5 | 31.952  | 88.406  | 21.161  | 26.374  | 19.123  | 24.309  | 22.534  | 27.851  | 19.729  | 25.011  | 21.888  | 27.168  |
| 0.6 | 20.195  | 25.362  | 21.006  | 26.371  | 21.426  | 30.303  | 20.640  | 26.050  | 19.424  | 24.279  | 20.750  | 26.175  |
| 0.7 | 19.168  | 23.924  | 20.989  | 26.482  | 21.440  | 27.129  | 19.370  | 24.287  | 19.274  | 24.779  | 21.545  | 26.830  |
| 0.8 | 21.982  | 27.362  | 19.760  | 24.681  | 20.662  | 25.488  | 21.324  | 26.648  | 21.085  | 26.994  | 21.644  | 27.160  |
| 0.9 | 21.356  | 27.235  | 21.251  | 26.559  | 19.518  | 24.880  | 21.276  | 26.736  | 21.283  | 26.981  | 19.217  | 23.910  |
| 1.0 | 20.206  | 25.608  | 20.914  | 26.173  | 20.726  | 25.877  | 21.430  | 26.625  | 20.102  | 25.198  | 19.810  | 25.348  |
| Total| 219.290 | 524.380 | 205.561 | 257.290 | 203.342 | 260.194 | 209.629 | 262.446 | 200.121 | 253.678 | 210.477 | 262.986 |
| Grand Total| 1284.343 | 1895.873 | 1199.098 | 1727.300 | 1131.878 | 1693.572 | 1066.375 | 1511.927 | 1124.465 | 1681.087 | 1211.808 | 1772.106 |
The corresponding times for calculating the joint probabilities and s-powers of these have been added up to report the results for the Havrda-Charvat entropies of the Grand total row. We think that these times are very well affordable indeed.

Table 18: Total CPU and real times for calculating the Entropy measure content of a family PF06850 and approximations for Grand Total of all sample space.

| POA | Entropy Measures | Entropy Measures |
|-----|------------------|------------------|
|     | $H_j(s) - 19$ s-values | $H_{jk}(s) - 19$ s-values |
| Total CPU time (family PF06850) | $0.358 + 2.562 + 0.292$ | $550.129 + 611.374$ |
|     | $= 3.212$ sec | $+ 421.418 = 1,582.921$ sec |
| Grand Total CPU time (1069 families) | $3,433.628$ sec | $1,692,142.549$ sec |
|     | $= 0.954$ hs | $= 19,585$ days |
| Total Real time (family PF06850) | $1.062 + 9.300 + 0.332$ | $593.848 + 1,649.357$ |
|     | $= 10.694$ sec | $+ 927.405 = 3.170.61$ sec |
| Grand Total Real time (1069 families) | $11,431.886$ sec | $3,389,382.090$ sec |
|     | $= 3.175$ hs | $= 39,229$ days |

Table 19: Total CPU and real times for calculating the Entropy measure content of a family PF06850 and approximations for Grand Total of all sample space.

| PLA | Entropy Measures | Entropy Measures |
|-----|------------------|------------------|
|     | $H_j(s) - 19$ s-values | $H_{jk}(s) - 19$ s-values |
| Total CPU time (family PF06850) | $0.291 + 1.801 + 0.640$ | $787.254 + 515.072$ |
|     | $= 2.732$ sec | $+ 403.461 = 1,705.787$ sec |
| Grand Total CPU time (1069 families) | $2,920.508$ sec | $1,823,486.303$ sec |
|     | $= 0.811$ hs | $= 21,105$ days |
| Total Real time (family PF06850) | $0.642 + 7.261 + 1.291$ | $1,068.026 + 1,028.655$ |
|     | $= 9.194$ sec | $+ 557.294 = 2,603.975$ sec |
| Grand Total Real time (1069 families) | $9,828.386$ sec | $2,783,649.275$ sec |
|     | $= 2.730$ hs | $= 32,218$ days |

6 Concluding Remarks and Suggestions for Future Work

The treatment of the distributions of probability of occur in protein databases is a twofold procedure. We intend to find a way of characterizing the protein database by values of Entropy Measures in order to provide a sound discussion to be centered on the maximization of a convenient average Entropy Measure to represent the entire protein database. We also intend to derive a partition function in order to derive a thermodynamical theory associated to the temporal evolution of the database. If the corresponding evolution of the protein families
is assumed to be registered on the subsequent versions of the database (Table 17), we will then be able to describe the sought thermodynamical evolution from this theory as well as to obtain from it the convenient description of all intermediate Levinthal’s stages which seem to be necessary for describing the folding/unfolding dynamical process.

We summarize this approach by the need of starting from a thermodynamical theory of the evolution of protein databases via Entropy measures to the construction of a successful dynamical theory of protein families. In other words, from the thermodynamics of evolution of a protein database, we will derive a statistical mechanics to give us physical insight on the construction of a successful dynamics of protein families.

References

[1] R.P. Mondaini — A Survey of Geometric Techniques for Pattern Recognition of probability of occurrence of Amino acids in Protein Families — BIOMAT 2016, 304 – 326, (2017), World Scientific Co. Pte. Ltd.

[2] R.P. Mondaini, S.C. de Albuquerque Neto — The Pattern Recognition of Probability Distributions of Amino Acids in Protein Families – BIOMAT 2016, 29–50, (2017) World Scientific Co. Pte. Ltd.

[3] R.P. Mondaini, S.C. de Albuquerque Neto – Pattern Recognition of Amino acids by a Poisson Statistical Approach – 31st International Colloquium on Group Theoretical Methods in Physics, Lecture Notes in Physics, Springer Verlag (2017).

[4] R.P. Mondaini, S.C. de Albuquerque Neto — Entropy Measures and the Statistical Analysis of Protein Family Classifications – BIOMAT 2015,193–210, (2016), World Scientific Co. Pte. Ltd.

[5] R.P. Mondaini, S.C. de Albuquerque Neto — Optimal Control of a Coarse-grained Model for Protein Dynamics — BIOMAT 2014, 11–25, (2015), World Scientific Co. Pte. Ltd.

[6] R.P. Mondaini — Entropy Measures based method for the classification of Protein Domains into Families and Clans – BIOMAT 2013, 209 - 218, (2014), World Scientific Co. Pte. Ltd.

[7] C. Levinthal — Are there Pathways for Protein Folding? — Journal de Chimie Physique et Physico-Chimie Biologique, 65(1968)44–45.

[8] M. Karplus — The Levinthal Paradox, Yesterday and Today — Folding and Design 2(1)(1997) 569–575.

[9] M.H. De Groot, M.J. Schervish — Probability and Statistics, 4th edition - Addison-Wesley, 2012.
[10] B.D. Sharma, D.P. Mittal — New Non-additive Measures of Entropy for a Discrete Probability Distribution — J. Math. Sci. 10(1975) 28–40.

[11] T.H. Cormen, C.B. Leiserson, L.R. Rivest, C. Stein — Introduction to Algorithms — MIT Press, 2001.

[12] R.D. Finn, A. Bateman, J. Clements, P. Coggill, R.Y. Eberhardt, S.R. Eddy, A. Heger, K. Hetherington, L. Holm, J. Mistry, E.L.L. Sonnhammer, J. Tate, M. Punta — The Pfam Protein Families Database — Nucleic Acids Research 42 (2014) D222–D230.

[13] R.D. Finn, P. Coggill, R. Y. Eberhardt, S.R. Eddy, J. Mistry, A.L. Mitchell, S.C. Potter, M. Punta, M. Qureshi, A. Sangrador-Veras, G.A. Salazar, J. Tate, A. Bateman — The Pfam Protein Families Database: Towards a Sustainable Future — Nucleic Acids Research, 44 (2016) D279 –D285.