International Journal of Computer Mathematics
Vol. 84, No. 5, May 2007, 599–617

Protein sequence analysis using relational soft clustering algorithms

PRADIPTA MAJI* and SANKAR K. PAL

Center for Soft Computing Research, Indian Statistical Institute, 203 B.T. Road,
Kolkata, 700 108, India

(Received 14 September 2006; revised version received 26 December 2006; accepted 07 January 2006)

To recognize functional sites within a protein sequence, the non-numerical attributes of the sequence need encoding prior to using a pattern recognition algorithm. The success of recognition depends on the efficient coding of the biological information contained in the sequence. In this regard, a bio-basis function maps a non-numerical sequence space to a numerical feature space, based on an amino acid mutation matrix. In effect, the biological content in a sequence can be maximally utilized for analysis. One of the important issues for the bio-basis function is how to select a minimum set of bio-bases with maximum information. In this paper, we present two relational soft clustering algorithms, named rough c-medoids and fuzzy-possibilistic c-medoids, to select the most informative bio-bases. While both fuzzy and possibilistic memberships of fuzzy-possibilistic c-medoids avoid the noise sensitivity defect of fuzzy c-medoids and the coincident clusters problem of possibilistic c-medoids, the concept of lower and upper boundaries of rough c-medoids deals with uncertainty, vagueness, and incompleteness in class definition of biological data. The concept of 'degree of resemblance', based on non-gapped pairwise homology alignment score, circumvents the initialization and local minima problems of both c-medoids algorithms. In effect, it enables efficient selection of a minimum set of most informative bio-bases. The effectiveness of the algorithms, along with a comparison with other algorithms, has been demonstrated on HIV (human immunodeficiency virus) protein datasets.

Keywords: Fuzzy sets; Pattern recognition; Relational clustering; Rough sets; Sequence analysis

AMS Subject Classifications: 03E72; 6ZH30

1. Introduction

The problem with using most pattern recognition algorithms to analyse biological sequences is that they cannot recognize non-numerical features such as the biochemical codes of amino acids. Therefore, they need encoding prior to input. Investigating a proper encoding process prior to modelling the amino acids is then critical. The successful analysis of biological sequences relies on the efficient coding of the biological information contained in the sequences.

The most commonly used method for coding a subsequence is distributed encoding, in which each of 20 amino acids is encoded using a 20-bit binary vector [1]. However, in this method the input space is expanded unnecessarily. Also, this method may not be able to encode biological content in sequences efficiently. Different distances for different amino acid pairs have been defined by various mutation matrices [2–4]. But, they cannot be used directly for encoding an amino acid to a unique numerical value.

With this background, the concept of bio-basis function has been proposed in [5–7] for analysing biological subsequences. It uses a kernel function to transform biological subsequences to feature vectors directly. Bio-bases consist of sections of a biological sequence that code for a feature of interest in the study, and are responsible for the transformation of biological data to high dimensional feature space. Transformation of input data to high dimensional feature space is performed based on the similarity of an input subsequence to a bio-basis with reference to a biological similarity matrix. Thus, the biological content in the sequences can be maximally utilized for accurate modelling. The use of similarity matrices to map features allows the bio-basis function to analyse biological sequences without the need for encoding. The concept of bio-basis function has been successfully applied in different applications [5–10].

The most important issue for a bio-basis function is how to select a minimum set of bio-bases with maximum information. Berry *et al.* [6] used genetic algorithms for bio-bases selection considering the Fisher ratio as the fitness function. Yang and Thomson [7] proposed another method to select bio-bases using mutual information. In principle, the bio-bases in non-numerical sequence space should be such that the degree of resemblance between pairs of bio-bases would be as small as possible. Each of them would then represent a unique feature in numerical feature space. As this is a feature selection problem, a clustering method can be used which partitions the given biological sequences into subgroups around each bio-basis, each of which should be as homogeneous (informative) as possible. However, the methods proposed in [6, 7] have not adequately addressed this problem.

In biological sequences, the only available information is the numerical values that represent the degrees to which pairs of sequences in the dataset are related. Algorithms that generate partitions of that type of relational data are usually referred to as relational or pairwise clustering algorithms. A well-known relational clustering algorithm is Kaufman and Rousseeuws' *c*-medoids [11, 12]. The *c*-medoids algorithm is applicable to situations where the objects to be clustered cannot be represented by numerical features, rather, only represented with similarities or dissimilarities between pairs of objects. So, the relational clustering algorithms can be used to cluster biological subsequences if one can come up with a similarity measure to quantify the degree of resemblance between pairs of subsequences.

One of the main problems with biological subsequence analysis is the uncertainty. Some of the sources of this uncertainty include incompleteness and vagueness in class definitions of biological data. In this background, two major soft computing techniques, fuzzy sets theory [13] and rough sets theory [14], have gained popularity in modelling and propagating uncertainty. Both fuzzy sets and rough sets provide a mathematical framework to capture uncertainties associated with the data [15–18]. Ruspini [19] and Diday [20] proposed two of the early fuzzy relational clustering algorithms. Other notable algorithms include Roubens' fuzzy non-metric model [21], Windham's association prototype model [22], Hathaway and Bezdek's relational fuzzy *c*-means (RFCM) [23], and Kaufman and Rousseeuws' fuzzy analysis (FANNY) [12]. A more recent algorithm is Krishnapuram's fuzzy *c*-medoids [24]. It offers the opportunity to deal with the data that belong to more than one cluster at the same time. Also, it can handle the uncertainties arising from overlapping cluster boundaries. However, it is very sensitive to noise and outliers as the memberships of a pattern in fuzzy *c*-medoids are inversely related to the relative distances of the pattern to the clusters prototypes. The possibilistic *c*-medoids [24] is an extension of fuzzy c-medoids, which efficiently handles datasets containing noise and outliers considering typicalities or compatibilities of the patterns with the cluster prototypes. But, it sometimes generates coincident clusters. Moreover, typicalities can be very sensitive to the choice of additional parameters needed by possibilistic c-medoids.

In this paper, we propose two relational soft clustering algorithms, rough c-medoids and fuzzy-possibilistic c-medoids, to select the most informative bio-bases. In rough c-medoids, the judicious integration of c-medoids and the concept of lower and upper approximations of rough sets efficiently deals with uncertainty, vagueness, and incompleteness in class definition. Each partition is represented by a medoid (bio-basis), a lower approximation, and a boundary region. The medoid (bio-basis) depends on the weighting average of the lower approximation and boundary. Similarly, the fuzzy-possibilistic c-medoids attempts to exploit the benefits of both fuzzy and possibilistic c-medoids. Both fuzzy and possibilistic memberships of fuzzy-possibilistic c-medoids enable efficient handling of overlapping partitions. It also avoids the noise sensitivity defect of fuzzy c-medoids and the coincident clusters problem of possibilistic c-medoids. Each partition is represented by a medoid (bio-basis), and a cluster, which is a combination of fuzzy and possibilistic partitions. The medoid (bio-basis) depends on both fuzzy and possibilistic memberships. The concept of ‘degree of resemblance’, based on non-gapped pairwise homology alignment score, automatically circumvents the initialization and local minima problems of both rough c-medoids and fuzzy-possibilistic c-medoids. In effect, it enables efficient selection of a minimum set of most informative bio-bases. Some quantitative measures are introduced based on mutual information and homology alignment score to evaluate the quality of selected bio-bases. The effectiveness of the proposed algorithms, along with a comparison with hard c-medoids, fuzzy c-medoids, possibilistic c-medoids, Berry *et al.*’s method [6], and Yang and Thomsons’ method [7], has been demonstrated on five whole HIV protein datasets.

The structure of the rest of this paper is as follows. Section 2 briefly introduces the necessary notions of a bio-basis function. In section 3, the fuzzy-possibilistic c-medoids algorithm is proposed based on the fuzzy and possibilistic memberships for bio-bases selection. Section 4 presents rough c-medoids along with an introduction to rough sets. Some quantitative performance measures are introduced in section 5 to select most informative bio-bases. A few case studies and a comparison with other methods are presented in section 6. Concluding remarks are given in section 7.

2. Protein sequence analysis using bio-basis kernel

The most successful method of sequence analysis is homology alignment [25, 26]. In this method, the function of a sequence is annotated through aligning a novel sequence with known sequences. If the homology alignment between a novel sequence and a known sequence gives a very high similarity score, the novel sequence is believed to have the same or similar function as the known sequence. In homology alignment, an amino acid mutation matrix is commonly used. Each mutation matrix has 20 columns and 20 rows. A value at the nth row and mth column is a probability or a likelihood value that the nth amino acid mutates to the mth amino acid after a particular evolutionary time [3, 4]. However, the principle of homology alignment cannot be used directly for subsequence analysis. Because, a subsequence may not contain enough information for conventional homology alignment. A high homology alignment score between a novel subsequence and a known subsequence cannot assert that two subsequences have the same function. However, it can be assumed that they may have the same function statistically.

The design of a bio-basis function is based on the principle of conventional homology alignment. The homology alignment score is calculated using an amino acid mutation matrix. Using a table look-up technique, a homology alignment score as a similarity value can be obtained for a pair of subsequences. The non-gapped homology alignment method is used to calculate this similarity value, where no deletion or insertion is used to align two subsequences. The definition of a bio-basis function is as follows [5–7]:

$$f(x_j, v_i) = \exp\left\{\gamma \frac{h(x_j, v_i) - h(v_i, v_i)}{h(v_i, v_i)}\right\} \tag{1}$$

where $h(x_j, v_i)$ is the pairwise homology alignment score between a subsequence x_j and a bio-basis v_i [2–4], $h(v_i, v_i)$ denotes the maximum homology alignment score of the ith bio-basis v_i and γ is a constant. Supposing both x_j and v_i have m residues, the non-gapped pairwise homology alignment score is defined as

$$h(x_j, v_i) = \sum_{k=1}^{m} M(x_{jk}, v_{ik}) \tag{2}$$

where $M(x_{jk}, v_{ik})$ can be obtained from an amino acid mutation matrix through a table look-up method. Note that $x_{jk}, v_{ik} \in A$ and A is a set of 20 amino acids. The function value is high if two subsequences are similar or close to each other. The function value is small if two subsequences are distinct. The function value is one if two subsequences are identical.

Each bio-basis is a feature dimension in a numerical feature space. It needs a subsequence as a support. If we use A to denote a collection of 20 amino acids, an input space of all potential subsequences with m residues is A^m. Then, a collection of c bio-bases formulates a numerical feature space $\mathbb{R}^c$, to which a non-numerical sequence space A^m is mapped for analysis. More importantly, the bio-basis function can transform various homology alignment scores to a real number as a similarity within the interval [0, 1]. After the mapping using bio-bases, a non-numerical subsequence space A^m will be mapped to a c-dimensional numerical feature space $\mathbb{R}^c$, i.e. $A^m \rightarrow \mathbb{R}^c$.

3. Fuzzy-possibilistic *c*-medoids algorithm

Three relational clustering algorithms – hard c-medoids, fuzzy c-medoids, and possibilistic c-medoids – are described first for selection of bio-bases. Next, we propose a new relational soft clustering algorithm, termed as fuzzy-possibilistic c-medoids, integrating both fuzzy and possibilistic membership functions.

3.1 *Hard c-medoids*

The hard c-medoids algorithm [11, 12] uses the most centrally located object in a cluster, which is termed the medoid. A medoid is essentially one of the actual data points from the cluster, which is closest to the mean of the cluster. The objective of the hard c-medoids algorithm for selection of bio-bases is to assign n subsequences to c clusters. Each of the clusters β_i is represented by a bio-basis v_i, which is the medoid for that cluster. The process begins by randomly choosing c subsequences as the bio-bases. The subsequences are assigned to one of the c clusters based on the similarity between the subsequence x_j and the bio-basis v_i. The similarity is assessed through the non-gapped pairwise homology alignment score $h(x_j, v_i)$ between the subsequence x_j and the bio-basis v_i. The score $h(x_j, v_i)$ can be calculated as per equation (2). After the assignment of all the subsequences to various clusters, the new bio-bases are calculated as follows:

$$v_i = x_q; \quad \text{where } q = \arg\min(h(x_j, x_j) - h(x_k, x_j)); \quad x_j \in \beta_i; \;\; x_k \in \beta_i. \tag{3}$$

The basic steps are outlined as follows:

(i) Arbitrarily choose c subsequences as the initial bio-bases v_i, $i = 1, 2, \ldots, c$.
(ii) Assign each remaining subsequences to the cluster for the closest bio-basis.
(iii) Compute the new bio-basis as per equation (3).
(iv) Repeat steps (ii) and (iii) until no more new assignments can be made.

3.2 *Fuzzy c-medoids*

This provides a fuzzification of the hard c-medoids algorithm [24]. For bio-bases selection, it minimizes

$$J_{\mathrm{F}} = \sum_{j=1}^{n}\sum_{i=1}^{c}(\mu_{ij})^{\acute{m}_1}(h(v_i, v_i) - h(x_j, v_i)) \tag{4}$$

where $1 \leq \acute{m}_1 < \infty$ is the fuzzifier, v_i is the ith bio-basis, $\mu_{ij} \in [0, 1]$ is the fuzzy membership of the subsequence x_j to cluster β_i, such that

$$\mu_{ij} = \sum_{l=1}^{c}\left\{\frac{(h(v_i, v_i) - h(x_j, v_i))}{(h(v_l, v_l) - h(x_j, v_l))}\right\}^{-1/\acute{m}_1 - 1} \tag{5}$$

subject to

$$\sum_{i=1}^{c}\mu_{ij} = 1, \; \forall j, \quad \text{and} \quad 0 < \sum_{j=1}^{n}\mu_{ij} < n, \; \forall i.$$

The new bio-bases are calculated as:

$$v_i = x_q; \quad \text{where} \quad q = \arg\min\sum_{k=1}^{n}(\mu_{ik})^{\acute{m}_1}(h(x_j, x_j) - h(x_k, x_j)); \quad 1 \leq j \leq n. \tag{6}$$

The algorithm proceeds as follows:

(i) Assign initial bio-bases v_i, $i = 1, 2, \ldots, c$.
(ii) Choose values for the fuzzifier $\acute{m}_1$ and threshold ϵ_1 and set iteration counter $t = 1$.
(iii) Compute μ_{ij} by equation (5) for c clusters and n subsequences.
(iv) Update bio-basis v_i by equation (6).
(v) Repeat steps (iii)–(v), by incrementing t, until $|\mu_{ij}(t) - \mu_{ij}(t-1)| > \epsilon_1$.

3.3 *Possibilistic c-medoids*

In possibilistic c-medoids [24], the objective function can be formulated as

$$J_{\mathrm{P}} = \sum_{i=1}^{c}\sum_{j=1}^{n}(\nu_{ij})^{\acute{m}_2}(h(v_i, v_i) - h(x_j, v_i)) + \sum_{i=1}^{c}\eta_i\sum_{j=1}^{n}(1 - \nu_{ij})^{\acute{m}_2} \tag{7}$$

η_i represents the bandwidth or resolution or scale parameter. The membership matrix ν generated by the possibilistic c-medoids is not a partition matrix in the sense that it does not satisfy the constraint

$$\sum_{i=1}^{c} \nu_{ij} = 1. \tag{8}$$

The membership update equation in the possibilistic c-medoids is

$$\nu_{ij} = \frac{1}{1+\mathrm{D}}; \quad \text{where } \mathrm{D} = \left\{ \frac{(h(v_i, v_i) - h(x_j, v_i))}{\eta_i} \right\}^{1/(\acute{m}_2 - 1)} \tag{9}$$

subject to

$$\nu_{ij} \in [0, 1], \ \forall i, j; \quad 0 < \sum_{j=1}^{n} \nu_{ij} \leq n, \ \forall i; \quad \text{and} \quad \max_i \nu_{ij} > 0, \ \forall j.$$

The scale parameter η_i represents the zone of influence or size of the cluster β_i and its update equation is

$$\eta_i = \mathrm{K} \cdot \frac{\mathrm{P}}{\mathrm{Q}}; \quad \text{where} \quad \mathrm{P} = \sum_{j=1}^{n} (\nu_{ij})^{\acute{m}_2} (h(v_i, v_i) - h(x_j, v_i)); \quad \text{and} \quad \mathrm{Q} = \sum_{j=1}^{n} (\nu_{ij})^{\acute{m}_2}. \tag{10}$$

Typically K is chosen to be 1. From the standpoint of 'compatibility with the bio-basis', the membership ν_{ij} of a subsequence x_j in a cluster β_i should be determined solely by how close it is to the bio-basis v_i of the class, and should not be coupled with its similarity with respect to other classes. Thus, in each iteration, the updated value of ν_{ij} depends only on the similarity between the subsequence x_j and the bio-basis v_i. The resulting partition of the biological data can be interpreted as a possibilistic partition, and the membership values may be interpreted as degrees of possibility of the subsequences belonging to the classes, i.e. the compatibilities of the subsequences with the bio-bases. The updating of the bio-bases proceeds exactly the same way as in the case of the fuzzy c-medoids algorithm.

3.4 *Fuzzy-possibilistic c-medoids*

Incorporating both fuzzy and possibilistic membership functions into hard c-medoids algorithm, we propose fuzzy-possibilistic c-medoids algorithm. It avoids the noise sensitivity defect of fuzzy c-medoids and the coincident clusters problem of possibilistic c-medoids. For bio-bases selection, it minimizes

$$J_{\mathrm{FP}} = \sum_{j=1}^{n} \sum_{i=1}^{c} \{a(\mu_{ij})^{\acute{m}_1} + b(\nu_{ij})^{\acute{m}_2}\}(h(v_i, v_i) - h(x_j, v_i)) + \sum_{i=1}^{c} \eta_i \sum_{j=1}^{n} (1 - \nu_{ij})^{\acute{m}_2}. \tag{11}$$

The constants a and b define the relative importance of fuzzy and possibilistic memberships in the objective function and $a + b = 1$. Note that, μ_{ij} has the same meaning of membership as that in fuzzy c-medoids. Similarly, ν_{ij} has the same interpretation of typicality as in possibilistic

c-medoids and is given by

$$\nu_{ij} = \frac{1}{1 + \mathrm{D}}; \quad \text{where} \quad \mathrm{D} = \left\{ \frac{b(h(v_i, v_i) - h(x_j, v_i))}{\eta_i} \right\}^{1/(\acute{m}_2 - 1)}. \tag{12}$$

The new bio-bases are calculated as:

$$v_i = x_q; \quad \text{where} \quad q = \arg\min \sum_{k=1}^{n} \{a(\mu_{ik})^{\acute{m}_1} + b(\nu_{ik})^{\acute{m}_2}\}(h(x_j, x_j) - h(x_k, x_j)); \quad 1 \leq j \leq n. \tag{13}$$

The main steps of the algorithm are as follows:

(i) Assign initial bio-bases v_i, $i = 1, 2, \ldots, c$.
(ii) Choose values for a, b, $\acute{m}_1$, $\acute{m}_2$, and threshold ϵ_1; and set iteration counter $t = 1$.
(iii) Compute μ_{ij} and ν_{ij} by equations (5) and (12) respectively for c clusters and n subsequences.
(iv) Estimate η_i using equation (10).
(v) Update bio-basis v_i by equation (13).
(vi) Repeat steps (iii)–(vi), by incrementing t, until $|\nu_{ij}(t) - \nu_{ij}(t-1)| > \epsilon_1$.

4. Rough sets and rough *c*-medoids algorithm

This section presents another version of c-medoids algorithm, known as rough c-medoids, based on rough sets. For ease of subsequent discussions, next we present the basic notions in the theory of rough sets.

4.1 *Rough sets*

The theory of rough sets begins with the notion of an approximation space, which is a pair $\langle U, R \rangle$, where U is a non-empty set (the universe of discourse) and R an equivalence relation on U, i.e. R is reflexive, symmetric, and transitive. The relation R decomposes the set U into disjoint classes in such a way that two elements x, y are in the same class iff $(x, y) \in R$. Let us denote by U/R the quotient set of U by the relation R, and

$$U/R = \{X_1, X_2, \ldots, X_m\}$$

where X_i is an equivalence class of R, $i = 1, 2, \ldots, m$. If two elements x, y in U belong to the same equivalence class $X_i \in U/R$, we say that x and y are indistinguishable. The equivalence classes of R and the empty set $\emptyset$ are the elementary sets in the approximation space $\langle U, R \rangle$. Given an arbitrary set $X \in 2^U$, in general it may not be possible to describe X precisely in $\langle U, R \rangle$. One may characterize X by a pair of lower and upper approximations defined as follows [14]:

$$\underline{R}(X) = \bigcup_{X_i \subseteq X} X_i; \quad \bar{R}(X) = \bigcup_{X_i \cap X \neq \emptyset} X_i.$$

That is, the lower approximation $\underline{R}(X)$ is the union of all the elementary sets which are subsets of X, and the upper approximation $\bar{R}(X)$ is the union of all the elementary sets which have a non-empty intersection with X. The interval $[\underline{R}(X), \bar{R}(X)]$ is the representation of an ordinary set X in the approximation space $\langle U, R\rangle$ or simply called the rough set of X. The lower (resp. upper) approximation $\underline{R}(X)$ (resp. $\bar{R}(X)$) is interpreted as the collection of those elements of U that definitely (resp. possibly) belong to X. Further, we can define:

- a set $X \in 2^U$ is said to be definable (or exact) in $\langle U, R\rangle$ iff $\underline{R}(X) = \bar{R}(X)$;
- for any $X, Y \in 2^U$, X is said to be roughly included in Y, denoted by $X \tilde{\subset} Y$, iff $\underline{R}(X) \subseteq \underline{R}(Y)$ and $\bar{R}(X) \subseteq \bar{R}(Y)$;
- X and Y is said to be roughly equal, denoted by $X \simeq_R Y$, in $\langle U, R\rangle$ iff $\underline{R}(X) = \underline{R}(Y)$ and $\bar{R}(X) = \bar{R}(Y)$.

In [14], Pawlak discusses two numerical characterizations of imprecision of a subset X in the approximation space $\langle U, R\rangle$: accuracy and roughness. Accuracy of X, denoted by $\alpha_R(X)$, is simply the ratio of the number of objects in its lower approximation to that in its upper approximation; namely

$$\alpha_R(X) = \frac{|\underline{R}(X)|}{|\bar{R}(X)|}.$$

The roughness of X, denoted by $\rho_R(X)$, is defined by subtracting the accuracy from 1:

$$\rho_R(X) = 1 - \alpha_R(X) = 1 - \frac{|\underline{R}(X)|}{|\bar{R}(X)|}.$$

Note that the lower the roughness of a subset, the better is its approximation. Further, the following observations are easily obtained:

(i) As $\underline{R}(X) \subseteq X \subseteq \bar{R}(X)$, $0 \leq \rho_R(X) \leq 1$.
(ii) By convention, when $X = \emptyset$, $\underline{R}(X) = \bar{R}(X) = \emptyset$ and $\rho_R(X) = 0$.
(iii) $\rho_R(X) = 0$ if and only if X is definable in $\langle U, R\rangle$.

4.2 *Rough c-medoids*

Let $\underline{A}(\beta_i)$ and $\bar{A}(\beta_i)$ represent the lower and upper approximations of cluster β_i, and $B(\beta_i) = \bar{A}(\beta_i) - \underline{A}(\beta_i)$ denote the boundary region of cluster β_i (figure 1). In the rough c-medoids algorithm, the concept of c-medoids algorithm is extended by viewing each cluster β_i as an interval or rough set. However, it is possible to define a pair of lower and upper bounds $[\underline{A}(\beta_i), \bar{A}(\beta_i)]$ or a rough set for every set $\beta_i \subseteq U$, U is the set of objects of concern [14]. The family of upper and lower bounds is required to follow some of the basic rough set properties such as:

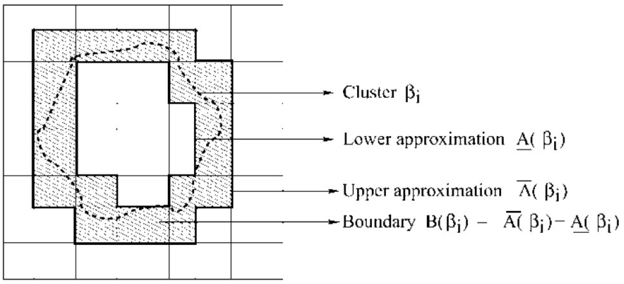
Cluster β_i
Lower approximation $\underline{A}(\beta_i)$
Upper approximation $\overline{A}(\beta_i)$
Boundary $B(\beta_i) - \overline{A}(\beta_i) - \underline{A}(\beta_i)$

Figure 1. Rough c-medoids: cluster β_i is represented by lower and upper bounds $[\underline{A}(\beta_i), \bar{A}(\beta_i)]$.

(i) an object x_j can be part of at most one lower bound;
(ii) $x_j \in \underline{A}(\beta_i) \Rightarrow x_j \in \bar{A}(\beta_i)$; and
(iii) an object x_j is not part of any lower bound $\Rightarrow$ x_j belongs to two or more upper bounds.

Incorporating rough sets into the c-medoids algorithm, we propose a rough c-medoids algorithm for generating bio-bases. It adds the concept of lower and upper bounds to the c-medoids algorithm. It classifies the subsequence space into two parts – lower approximation and boundary region. The bio-basis (medoid) is calculated based on the weighted average of the lower bound and boundary region. All the subsequences in the lower approximation take the same weight w while all the subsequences in the boundary take another weighting index $\tilde{w}$ uniformly. Calculation of the bio-bases is modified to include the effects of lower as well as upper bounds. The modified bio-bases calculation for rough c-medoids algorithm is given by:

$$v_i = x_q \tag{14}$$

where q is given by

$$q = \arg\min \begin{cases} w \times \mathcal{A} + \tilde{w} \times \mathcal{B} & \text{if } \underline{A}(\beta_i) \neq \emptyset, B(\beta_i) \neq \emptyset \\ \mathcal{A} & \text{if } \underline{A}(\beta_i) \neq \emptyset, B(\beta_i) = \emptyset \\ \mathcal{B} & \text{if } \underline{A}(\beta_i) = \emptyset, B(\beta_i) \neq \emptyset \end{cases}$$

$$\mathcal{A} = \sum_{x_k \in \underline{A}(\beta_i)} (h(x_j, x_j) - h(x_k, x_j)); \quad \text{and} \quad \mathcal{B} = \sum_{x_k \in B(\beta_i)} (h(x_j, x_j) - h(x_k, x_j)).$$

β_i represents the ith cluster associated with the bio-basis v_i. $\underline{A}(\beta_i)$ and $B(\beta_i)$ represent the lower bound and the boundary region of cluster β_i. The parameters w and $\tilde{w}$ correspond to the relative importance of the lower bound and boundary region, and $w + \tilde{w} = 1$. The main steps of rough c-medoids are as follows:

(i) Assign initial bio-bases v_i, $i = 1, 2, \ldots, c$. Choose a value for threshold ϵ_2.
(ii) For each subsequence x_j, calculate the homology alignment score $h(x_j, v_i)$ between itself and the bio-basis v_i of cluster β_i.
(iii) If $h(x_j, v_i)$ is maximum for $1 \le i \le c$ and $h(x_j, v_i) - h(x_j, v_k) \le \epsilon_2$, then $x_j \in \bar{A}(\beta_i)$ and $x_j \in \bar{A}(\beta_k)$. Furthermore, x_j is not part of any lower bound.
(iv) Otherwise, $x_j \in \underline{A}(\beta_i)$ such that $h(x_j, v_i)$ is the maximum for $1 \le i \le c$. In addition, by the properties of rough sets, $x_j \in \bar{A}(\beta_i)$.
(v) Compute new bio-basis (medoid) as per equation (14).
(vi) Repeat steps (ii)–(v) until no more new assignments can be made.

5. Selection of initial bio-basis and quantitative measure

This section provides the methodology to select initial bio-bases for different c-medoids algorithms. Some quantitative measures are also presented to evaluate the quality of selected bio-bases.

5.1 *Selection of initial bio-basis*

A limitation of the c-medoids algorithm is that it can only achieve a local optimum solution that depends on the initial choice of the bio-bases. Consequently, computing resources may be wasted in that some initial bio-bases get stuck in regions of the input space with a scarcity of data points and may therefore never have the chance to move to new locations where they are needed. To overcome this limitation of the c-medoids algorithm, next we propose a method to select initial bio-bases, which is based on a similarity measure using amino acid mutation matrix. It enables the algorithm to converge to an optimum or near optimum solutions (bio-bases).

Prior to describing the proposed method for selecting initial bio-bases, next we provide a measure to quantify the similarity between two subsequences in terms of pairwise homology alignment score.

- **Degree of resemblance (DOR):** The DOR between two subsequences x_i and x_j is defined as

$$\mathrm{DOR}(x_j, x_i) = \frac{h(x_j, x_i)}{h(x_i, x_i)}. \tag{15}$$

It is the ratio between the non-gapped pairwise homology alignment score of two input subsequences x_i and x_j based on an amino acid mutation matrix to the maximum homology alignment score of the subsequence x_i. It is used to quantify the similarity in terms of the homology alignment score between pairs of subsequences. If functions of two subsequences are different, the DOR between them is small. A high value of the $\mathrm{DOR}(x_i, x_j)$ between two subsequences x_i and x_j asserts that they may have the same function statistically. If two subsequences are same, the DOR between them is maximum, that is, $\mathrm{DOR}(x_i, x_i) = 1$. Thus, $0 < \mathrm{DOR}(x_i, x_j) \leq 1$. Also, $\mathrm{DOR}(x_i, x_j) \neq \mathrm{DOR}(x_j, x_i)$.

Based on the concept of the DOR, next we describe the method for selecting initial bio-bases. The main steps of this method proceed as follows.

(i) For each subsequence x_i, calculate the $\mathrm{DOR}(x_j, x_i)$ between itself and the subsequence $x_j, \forall_{j=1}^{n}$.

(ii) Calculate the similarity score between subsequences x_i and x_j

$$\mathrm{S}(x_j, x_i) = \begin{cases} 1 & \text{if } \mathrm{DOR}(x_j, x_i) > \epsilon_3 \\ 0 & \text{Otherwise.} \end{cases}$$

(iii) For each x_i, calculate the total number of similar subsequences of x_i as

$$\mathrm{N}(x_i) = \sum_{j=1}^{n} \mathrm{S}(x_j, x_i).$$

(iv) Sort n subsequences according to their values of $\mathrm{N}(x_i)$ such that $\mathrm{N}(x_1) > \mathrm{N}(x_2) > \cdots > \mathrm{N}(x_n)$.

(v) If $\mathrm{N}(x_i) > \mathrm{N}(x_j)$ and $\mathrm{DOR}(x_j, x_i) > \epsilon_3$, then x_j cannot be considered as a bio-basis, resulting in a reduced set of subsequences to be considered for initial bio-bases.

(vi) Let there be $\acute{n}$ subsequences in the reduced set having $\mathrm{N}(x_i)$ values such that $\mathrm{N}(x_1) > \mathrm{N}(x_2) > \cdots > \mathrm{N}(x_{\acute{n}})$. A heuristic threshold function can be defined as [27]

$$\mathrm{Tr} = \frac{\mathrm{R}}{\epsilon_4}; \quad \text{where } \mathrm{R} = \sum_{i=1}^{\acute{n}} \frac{1}{\mathrm{N}(x_i) - \mathrm{N}(x_{i+1})}$$

where ϵ_4 is a constant (=0.5, say), so that all subsequences in the reduced set having $\mathrm{N}(x_i)$ value higher than it are regarded as the initial bio-bases.

The value of Tr is high if most of the $\mathrm{N}(x_i)$ are large and close to each other. The above condition occurs when a small number of large clusters are present. On the other hand, if the $\mathrm{N}(x_i)$ have wide variation among them, then the number of clusters with smaller size increases. Accordingly, Tr attains a lower value.

Note that the main motive for introducing this threshold function lies in reducing the number of bio-bases. We attempt to eliminate noisy bio-bases (subsequence representatives having lower values of $\mathrm{N}(x_i)$) from the whole subsequences. The whole approach is, therefore, data dependent.

5.2 *Quantitative measure*

In this section we present some quantitative indices to evaluate the quality of selected bio-bases incorporating the concepts of non-gapped pairwise homology alignment score and mutual information.

5.2.1 Using homology alignment score. Based on the non-gapped pairwise homology alignment score, next we introduce two indices, β and γ, for evaluating quantitatively the quality of selected bio-bases.

- β Index: This is defined as

$$\beta = \frac{1}{c}\sum_{i=1}^{c}\frac{1}{n_i}\sum_{x_j\in\beta_i}\frac{h(x_j, v_i)}{h(v_i, v_i)}; \quad \text{i.e. } \beta = \frac{1}{c}\sum_{i=1}^{c}\frac{1}{n_i}\sum_{x_j\in\beta_i}\mathrm{DOR}(x_j, v_i) \tag{16}$$

  where n_i is the number of subsequences in the ith cluster β_i and $h(x_j, v_i)$ is the non-gapped pairwise homology alignment score between subsequence x_j and bio-basis v_i. The β index is the ratio between the normalized average homology alignment scores of input subsequences and their corresponding bio-bases. A good clustering procedure for bio-bases selection should make all input subsequences as similar to their bio-bases as possible. The β index increases with increase in homology alignment scores within a cluster. Therefore, for a given dataset and c value, the higher the homology alignment scores within the clusters, the higher would be the β value. The value of β also increases with c. In an extreme case when the number of clusters is maximum, i.e. $c = n$, the total number of subsequences in the dataset, we have $\beta = 1$. Thus, $0 < \beta \leq 1$.
- γ Index: This can be defined as

$$\gamma = \max_{i,j}\frac{1}{2}\left\{\frac{h(v_j, v_i)}{h(v_i, v_i)} + \frac{h(v_i, v_j)}{h(v_j, v_j)}\right\}; \quad \text{i.e.}$$

$$\gamma = \max_{i,j}\frac{1}{2}\{\mathrm{DOR}(v_j, v_i) + \mathrm{DOR}(v_i, v_j)\}; \quad 0 < \gamma < 1 \tag{17}$$

  The γ index calculates the maximum normalized homology alignment score between bio-bases. A good clustering procedure for bio-bases selection should make the homology alignment score between all bio-bases as low as possible. The γ index minimizes the between-cluster homology alignment score.

5.2.2 Using mutual information. Using the concept of mutual information, one can measure the within-cluster and between-cluster shared information. In principle, mutual information is regarded as a non-linear correlation function and can be used to measure the mutual relation between a bio-basis and the subsequences as well as the mutual relation between each pair of bio-bases. It is used to quantify the information shared by two objects. If two independent objects do not share much information, the mutual information value between them is small, while two highly non-linearly correlated objects will demonstrate a high mutual information value. In the present case, the objects can be the bio-bases and the subsequences.

- Based on the mutual information, the β index would be as follows.

$$\bar{\beta} = \frac{1}{c}\sum_{i=1}^{c}\frac{1}{n_i}\sum_{x_j \in \beta_i}\frac{\mathrm{MI}(x_j, v_i)}{\mathrm{MI}(v_i, v_i)}. \tag{18}$$

  $\mathrm{MI}(x_i, x_j)$ is the mutual information between subsequences x_i and x_j, and is defined as

$$\mathrm{MI}(x_i, x_j) = \mathrm{H}(x_i) + \mathrm{H}(x_j) - \mathrm{H}(x_i, x_j) \tag{19}$$

  with $\mathrm{H}(x_i)$ and $\mathrm{H}(x_j)$ being the entropy of subsequences x_i and x_j respectively, and $\mathrm{H}(x_i, x_j)$ their joint entropy. $\mathrm{H}(x_i)$ and $\mathrm{H}(x_i, x_j)$ are defined as

$$\mathrm{H}(x_i) = -\mathrm{p}(x_i)\mathrm{ln}\mathrm{p}(x_i); \quad \mathrm{H}(x_i, x_j) = -\mathrm{p}(x_i, x_j)\ln \mathrm{p}(x_i, x_j). \tag{20}$$

  $\mathrm{p}(x_i)$ and $\mathrm{p}(x_i, x_j)$ are the *a priori* probability of x_i and joint probability of x_i and x_j respectively. The $\bar{\beta}$ index is the ratio between the normalized average mutual information of input subsequences to their corresponding bio-bases. A bio-bases selection procedure should make the shared information between all input subsequences and their bio-bases as high as possible. The $\bar{\beta}$ index increases with increase in mutual information within a cluster. Therefore, for a given dataset and c value, the higher the mutual information within the clusters, the higher would be the $\bar{\beta}$ value. The value of $\bar{\beta}$ also increases with c. When $c = n$, $\bar{\beta} = 1$. Thus, $0 < \bar{\beta} \leq 1$.
- Similarly, the γ index is

$$\bar{\gamma} = \max_{i,j}\frac{1}{2}\left\{\frac{\mathrm{MI}(v_i, v_j)}{\mathrm{MI}(v_i, v_i)} + \frac{\mathrm{MI}(v_i, v_j)}{\mathrm{MI}(v_j, v_j)}\right\}. \tag{21}$$

  The $\bar{\gamma}$ index calculates the maximum normalized mutual information between bio-bases. A good clustering procedure for bio-bases selection should make the shared information between all bio-bases as low as possible. The $\bar{\gamma}$ index minimizes the between-cluster mutual information.

6. Experimental results

The performance of rough c-medoids (RCMdd) and fuzzy-possibilistic c-medoids (FPCMdd) is compared extensively with that of various other related ones. The algorithms compared are (i) hard c-medoids (HCMdd) [11, 12], (ii) fuzzy c-medoids (FCMdd) [24], (iii) possibilistic c-medoids (PCMdd) [24], (iv) method proposed by Yang and Thomson [7] using mutual information (MI), and (v) method proposed by Berry *et al.* [6] using genetic algorithms and the Fisher ratio (GAFR).

To analyse the performance of the proposed methods, we have used a real dataset of HIV (human immunodeficiency virus) protein sequences. The initial bio-bases $\{v_i\}$ for c-medoids algorithms, which represent crude clusters in the non-numerical space, have been generated by the methodology described in section 5.1. The Dayhoff amino acid mutation matrix is used to calculate the non-gapped pairwise homology alignment score between two subsequences [2–4]. In all the experiments, the parameters used are as follows:

Fuzzifiers: $\acute{m}_1 = 2.0$ and $\acute{m}_2 = 2.0$; Constants: $a = 0.5$ and $b = 0.5$
Parameters: $\epsilon_1 = 0.001$, $\epsilon_2 = 0.2$, and $\epsilon_4 = 0.5$

The parameters are held constant across all runs. All the experiments are implemented in C and run in the LINUX® environment having machine configuration Pentium® IV, 3.2 GHz, 1 MB cache, and 1 GB RAM.

6.1 *Description of dataset*

HIV protease belongs to the family of aspartyl proteases, which have been well-characterized as proteolytic enzymes. The catalytic component is composed of carboxyl groups from two aspartyl residues located in both NH_2- and COOH-terminal halves of the enzyme molecule in HIV protease [28]. They are strongly substrate-selective and cleavage-specific demonstrating their capability of cleaving large, virus-specific polypeptides called polyproteins between a specific pair of amino acids. Miller *et al.* [29] showed that the cleavage sites in HIV polyprotein can extend to an octapeptide region. The amino acid residues within this octapeptide region are represented by P_4-P_3-P_2-P_1-$P_{1'}$-$P_{2'}$-$P_{3'}$-$P_{4'}$, where P_4-P_3-P_2-P_1 is the NH_2-terminal half and $P_{1'}$-$P_{2'}$-$P_{3'}$-$P_{4'}$ the COOH-terminal half. Their counterparts in HIV protease are represented by S_4-S_3-S_2-S_1-$S_{1'}$-$S_{2'}$-$S_{3'}$-$S_{4'}$ [30]. The HIV protease cleavage site is exactly between P_1 and $P_{1'}$.

The five whole HIV protein sequences have been downloaded from NCBI (the National Center for Biotechnology Information, http://www.ncbi.nlm.nih.gov). The accession numbers are AAC82593, AAG42635, AAO40777, NP_057849, and NP_057850. Details of these five sequences are included in table 1. Note that MA, CA, NC, TF, PR, RT, RH, and IN are matrix protein, capsid protein, nucleocapsid core protein, transframe peptide, protease, reverse transcriptase, RNAse, and integrase, respectively. They are all cleavage products of HIV protease. p1, p2, and p6 are also cleavage products [31, 32]. For instance, 132 (MA/CA) means that the cleavage site is between the residues 132 (P_1) and 133 ($P_{1'}$) and the cleavage split the polyprotein producing two functional proteins, the matrix protein and the capsid protein. The subsequences from each of five whole protein sequences are obtained through moving a sliding window with eight residues. Once a subsequence is produced, it is considered as functional (Class A) if there is a cleavage site between P_1-$P_{1'}$, otherwise it is labelled as non-functional (Class B).

Table 1. Details of five whole HIV protein sequences from NCBI.

Accession number	Sequence length	Cleavage sites at P_1
AAC82593	500	132(MA/CA), 363(CA/p2), 377(p2/NC), 432(NC/p1), 448(p1/p6)
AAG42635	498	132(MA/CA), 363(CA/p2), 376(p2/NC), 430(NC/p1), 446(p1/p6)
AAO40777	500	132(MA/CA), 363(CA/p2), 377(p2/NC), 432(NC/p1), 448(p1/p6)
NP_057849	1435	488(TF/PR), 587(PR/RT), 1027(RT/RH), 1147(RH/IN)
NP_057850	500	132(MA/CA), 363(CA/p2), 377(p2/NC), 432(NC/p1), 448(p1/p6)

Table 2. Performance of different algorithms on NP_057849.

Parameters	Algorithms	β	γ	$\bar{\beta}$	$\bar{\gamma}$
	HCMdd	0.621	0.827	0.803	1.000
	FCMdd	0.746	0.828	0.811	0.996
$\epsilon_3 = 0.70$	PCMdd	0.750	0.825	0.814	0.991
$\acute{n} = 84$, $\mathrm{Tr} = 16.58$	FPCMdd	0.752	0.819	0.816	0.988
$c = 27$	RCMdd	0.635	0.829	0.812	1.000
	MI	0.625	0.913	0.801	1.000
	GAFR	0.618	0.902	0.810	1.000
	HCMdd	0.643	0.751	0.807	1.000
	FCMdd	0.767	0.701	0.823	0.956
$\epsilon_3 = 0.75$	PCMdd	0.773	0.713	0.826	0.953
$\acute{n} = 223$, $\mathrm{Tr} = 35.32$	FPCMdd	0.782	0.703	0.825	0.937
$c = 36$	RCMdd	0.651	0.751	0.822	1.000
	MI	0.637	0.854	0.802	1.000
	GAFR	0.646	0.872	0.811	1.000
	HCMdd	0.605	0.938	0.807	1.000
	FCMdd	0.667	0.941	0.805	1.000
$\epsilon_3 = 0.80$	PCMdd	0.670	0.941	0.806	1.000
$\acute{n} = 594$, $\mathrm{Tr} = 28.05$	FPCMdd	0.674	0.938	0.810	1.000
$c = 6$	RCMdd	0.604	0.941	0.805	1.000
	MI	0.611	0.938	0.811	1.000
	GAFR	0.608	0.957	0.803	1.000

HCMdd: hard c-medoids; FCMdd: fuzzy c-medoids; PCMdd: possibilistic c-medoids;
FPCMdd: fuzzy-possibilistic c-medoids; RCMdd: rough c-medoids;
MI: mutual information (Yang and Thomson [7]); GAFR: genetic algorithms and fisher ratio (Berry [6])

6.2 *Performance analysis*

The experimental results on five whole HIV protein datasets, reported in table 1, are presented in tables 2–6. Subsequent discussions analyse the results with respect to β, γ, $\bar{\beta}$, $\bar{\gamma}$, and execution time.

6.2.1 Optimum value of ϵ_3. Table 2 reports the values of β, γ, $\bar{\beta}$, and $\bar{\gamma}$ of different algorithms for the dataset NP_057849. Results are presented for different values of ϵ_3. The parameters generated from the dataset NP_057849 are shown in table 2. The value of c is computed using the method described in section 5.1. It may be noted that the optimal choice of c is a function of the value ϵ_3. The best result is achieved at $\epsilon_3 = 0.75$. For the purpose of comparison, c bio-bases are generated using GAFR and MI.

It is seen from the results of table 2 that the FPCMdd achieves consistently better performance than other algorithms with respect to the values of β, γ, $\bar{\beta}$, and $\bar{\gamma}$ for different values of ϵ_3. The best performance with respect to the values of β, γ, $\bar{\beta}$, and $\bar{\gamma}$, is achieved with $\epsilon_3 = 0.75$. At $\epsilon_3 = 0.75$, the values of $\mathrm{N}(x_i)$ for most of the subsequences in reduced dataset are large and close to each other. So, the threshold Tr attains a higher value compared to that of other values of ϵ_3. In effect, the subsequences selected as initial bio-bases with $\epsilon_3 = 0.75$, have higher values of $\mathrm{N}(x_i)$. Hence, the quality of generated clusters using different c-medoids algorithms are better compared to other values of ϵ_3.

6.2.2 Random versus DOR based Initialization. Tables 3 and 4 provide comparative results of different c-medoids algorithms with random initialization of bio-bases and the DOR based initialization method considering $\epsilon_3 = 0.75$. The DOR based initialization is

Table 3. Performance of different *c*-medoids algorithms.

Dataset	Algorithms	Bio-bases	β	γ	$\bar{\beta}$	$\bar{\gamma}$
AAC82593	HCMdd	Random	0.615	0.817	0.809	1.000
		Proposed	0.719	0.702	0.852	1.000
	FCMdd	Random	0.655	0.791	0.821	1.000
		Proposed	0.814	0.680	0.901	0.956
	PCMdd	Random	0.644	0.772	0.805	1.000
		Proposed	0.815	0.677	0.904	0.949
	FPCMdd	Random	0.698	0.757	0.832	1.000
		Proposed	0.821	0.677	0.909	0.952
	RCMdd	Random	0.674	0.813	0.825	1.000
		Proposed	0.815	0.677	0.872	0.983
AAG42635	HCMdd	Random	0.657	0.799	0.803	1.000
		Proposed	0.714	0.664	0.853	1.000
	FCMdd	Random	0.698	0.706	0.818	1.000
		Proposed	0.807	0.674	0.892	0.924
	PCMdd	Random	0.701	0.689	0.824	1.000
		Proposed	0.811	0.672	0.897	0.937
	FPCMdd	Random	0.704	0.683	0.828	1.000
		Proposed	0.811	0.659	0.894	0.928
	RCMdd	Random	0.685	0.709	0.812	1.000
		Proposed	0.768	0.681	0.882	1.000
AAO40777	HCMdd	Random	0.651	0.864	0.837	1.000
		Proposed	0.794	0.723	0.881	1.000
	FCMdd	Random	0.718	0.804	0.842	1.000
		Proposed	0.817	0.634	0.912	0.977
	PCMdd	Random	0.726	0.801	0.846	1.000
		Proposed	0.821	0.630	0.911	0.962
	FPCMdd	Random	0.729	0.796	0.850	1.000
		Proposed	0.824	0.629	0.914	0.972
	RCMdd	Random	0.717	0.791	0.847	1.000
		Proposed	0.809	0.633	0.879	0.977
NP_057849	HCMdd	Random	0.601	0.882	0.801	1.000
		Proposed	0.643	0.751	0.807	1.000
	FCMdd	Random	0.606	0.802	0.811	1.000
		Proposed	0.767	0.701	0.823	0.956
	PCMdd	Random	0.614	0.802	0.817	1.000
		Proposed	0.773	0.713	0.826	0.953
	FPCMdd	Random	0.651	0.799	0.805	1.000
		Proposed	0.782	0.703	0.825	0.937
	RCMdd	Random	0.600	0.811	0.801	1.000
		Proposed	0.651	0.751	0.822	1.000
NP_057850	HCMdd	Random	0.611	0.913	0.792	1.000
		Proposed	0.714	0.719	0.801	1.000
	FCMdd	Random	0.648	0.881	0.796	1.000
		Proposed	0.784	0.692	0.886	0.983
	PCMdd	Random	0.657	0.837	0.799	1.000
		Proposed	0.801	0.692	0.889	0.983
	FPCMdd	Random	0.662	0.831	0.801	1.000
		Proposed	0.807	0.688	0.890	0.971
	RCMdd	Random	0.639	0.895	0.794	1.000
		Proposed	0.758	0.702	0.826	0.993

Table 4. Execution time (milliseconds) of different c-medoids algorithm.

Algorithms	Bio-bases	AAC82593	AAG42635	AAO40777	NP_057849	NP_057850
HCMdd	Random	2359	2574	2418	8728	2164
	Proposed	535	534	532	4397	529
FCMdd	Random	7349	16342	11079	293264	13217
	Proposed	5898	11998	9131	240834	9174
PCMdd	Random	8217	13691	10983	295990	14372
	Proposed	5982	10311	9618	241033	9713
FPCMdd	Random	9353	15892	12669	295874	15307
	Proposed	6437	12133	12561	250963	10521
RCMdd	Random	6108	13816	8053	268199	10318
	Proposed	5691	8015	5880	160563	5895

Table 5. Performance of different c-medoids algorithms.

Dataset	Algorithms	β	γ	$\bar{\beta}$	$\bar{\gamma}$
AAC82593	HCMdd	0.719	0.702	0.852	1.000
	FCMdd	0.814	0.680	0.901	0.956
	PCMdd	0.815	0.677	0.904	0.949
	FPCMdd	0.821	0.677	0.909	0.952
	RCMdd	0.815	0.677	0.872	0.983
	MI	0.764	0.788	0.906	0.977
	GAFR	0.736	0.814	0.826	1.000
AAG42635	HCMdd	0.714	0.664	0.853	1.000
	FCMdd	0.807	0.674	0.892	0.924
	PCMdd	0.811	0.672	0.897	0.937
	FPCMdd	0.811	0.659	0.894	0.928
	RCMdd	0.768	0.681	0.882	1.000
	MI	0.732	0.637	0.829	0.989
	GAFR	0.707	0.713	0.801	1.000
AAO40777	HCMdd	0.794	0.723	0.881	1.000
	FCMdd	0.817	0.634	0.912	0.977
	PCMdd	0.821	0.630	0.911	0.962
	FPCMdd	0.824	0.629	0.914	0.972
	RCMdd	0.809	0.633	0.879	0.977
	MI	0.801	0.827	0.890	0.982
	GAFR	0.773	0.912	0.863	1.000
NP_057849	HCMdd	0.643	0.751	0.807	1.000
	FCMdd	0.767	0.701	0.823	0.956
	PCMdd	0.773	0.713	0.826	0.953
	FPCMdd	0.782	0.703	0.825	0.937
	RCMdd	0.651	0.751	0.822	1.000
	MI	0.637	0.854	0.802	1.000
	GAFR	0.646	0.872	0.811	1.000
NP_057850	HCMdd	0.714	0.719	0.801	1.000
	FCMdd	0.784	0.692	0.886	0.983
	PCMdd	0.801	0.692	0.889	0.983
	FPCMdd	0.807	0.688	0.890	0.971
	RCMdd	0.758	0.702	0.826	0.993
	MI	0.736	0.829	0.833	1.000
	GAFR	0.741	0.914	0.809	1.000

Table 6. Execution time (milliseconds) of different methods.

Algorithms	AAC82593	AAG42635	AAO40777	NP_057849	NP_057850
HCMdd	535	534	532	4397	529
FCMdd	5898	11998	9131	240834	9174
PCMdd	5982	10311	9618	241033	9713
FPCMdd	6437	12133	12561	250963	10521
RCMdd	5691	8015	5880	160563	5895
MI	8617	13082	12974	250138	9827
GAFR	12213	12694	11729	291413	10873

found to improve the performance in terms of β, γ, $\bar{\beta}$, and $\bar{\gamma}$ as well as reduce the time requirement of all c-medoids algorithms. It is also observed that HCMdd with the DOR based initialization performs similar to FPCMdd with random initialization, although it is expected that FPCMdd is superior to HCMdd in partitioning subsequences. While in random initialization, the c-medoids algorithms get stuck in local optima, the DOR based scheme enables the algorithms to converge to an optimum or near optimum solution. In effect, the execution times required for different c-medoids are shorter in the DOR based initialization compared to random initialization.

6.2.3 Performance on five protein datasets. Finally, table 5 provides the comparative results of different algorithms for five whole HIV protein datasets. It is seen that the FPCMdd with the DOR based initialization produces bio-bases having the highest β and $\bar{\beta}$ values and lowest γ and $\bar{\gamma}$ values for all the cases. Table 6 provides comparative results of different algorithms in terms of execution time for five datasets. The execution time required for FPCMdd is comparable to MI and GAFR. For the HCMdd, although the execution time is less, the performance is significantly poorer than the RCMdd and FPCMdd.

The following conclusions can be drawn from the results reported in tables 2–6:

(i) It is observed that FPCMdd is superior to HCMdd both with random and the DOR based initialization. However, HCMdd requires considerably less time compared to FPCMdd. But, the performance of HCMdd is significantly poorer than FPCMdd. The performance of FCMdd, PCMdd, and RCMdd is intermediate between FPCMdd and HCMdd.

(ii) The DOR based initialization is found to improve the values of β, γ, $\bar{\beta}$, and $\bar{\gamma}$ as well as reduce the time requirement substantially for all c-medoids algorithms.

(iii) Use of fuzzy and possibilistic memberships and rough sets adds a small computational load to the HCMdd algorithm; however the corresponding integrated methods (FCMdd, PCMdd, FPCMdd, and RCMdd) show a definite increase in β and $\bar{\beta}$ values and decrease in γ and $\bar{\gamma}$ values.

(iv) Integration of soft computing and c-medoids, in RCMdd, FCMdd, PCMdd, and FPCMdd, produces a minimum set of most informative bio-bases in least computation time compared to GAFR and MI.

(v) It is observed that the RCMdd algorithm requires significantly less time compared to MI and GAFR having comparable performance. Reduction in time is achieved due to the DOR based initialization. The DOR based initialization reduces the convergence time of the RCMdd algorithm considerably compared to random initialization.

The best performance of the proposed algorithms is achieved due to the following three reasons:

(i) the DOR based initialization of bio-bases enables the algorithm to converge to an optimum or near optimum solution;
(ii) the concept of lower and upper bounds of rough sets in RCMdd deals with uncertainty, vagueness, and incompleteness in class definition; and
(iii) both fuzzy and possibilistic memberships in FPCMdd handle overlapping partitions efficiently and deal with uncertainty, vagueness, and incompleteness in class definition.

In effect, the minimum set of most informative bio-bases are obtained using proposed algorithms.

7. Conclusion

The main contribution of this paper is to develop a methodology integrating the merits of soft computing (rough sets and fuzzy sets), *c*-medoids algorithm, and amino acid mutation matrix for bio-bases selection. Some quantitative measures are introduced to evaluate quantitatively the quality of selected bio-bases. The effectiveness of the proposed algorithms has been demonstrated, along with a comparison with other algorithms, on five whole HIV protein datasets. The concept of 'degree of resemblance' is found to be successful in effectively circumventing the initialization and local minima problems of iterative refinement clustering algorithms like *c*-medoids. In addition, this concept enables efficient selection of a minimum set of most informative bio-bases compared to existing methods.

Although the methodology of integrating *c*-medoids algorithm and soft computing (rough sets and fuzzy sets) has been efficiently demonstrated for protein sequence analysis, the concept can be applied to other bioinformatics problems. The integration of *c*-medoids, rough sets, fuzzy sets, and evolutionary algorithm, may be used for generating a minimum set of bio-bases with maximum information.

Acknowledgement

The authors would like to thank the anonymous referees for providing helpful comments and valuable criticisms which have greatly improved the presentation of the paper.

References

[1] Qian, N. and Sejnowski, T.J., 1988, Predicting the secondary structure of globular proteins using neural network models. *Journal of Molecular Biology*, **202**, 865–884.
[2] Dayhoff, M.O., Schwartz, R.M. and Orcutt, B.C., 1978, A model of evolutionary change in proteins. Matrices for detecting distant relationships. *Atlas of Protein Sequence and Structure*, **5**, 345–358.
[3] Henikoff, S. and Henikoff, J.G., 1992, Amino acid substitution matrices from protein blocks. *PNSA*, **89**, 10915–10919.
[4] Johnson, M.S. and Overington, J.P., 1993, A structural basis for sequence comparisons: an evaluation of scoring methodologies. *Journal of Molecular Biology*, **233**, 716–738.
[5] Thomson, R., Hodgman, C., Yang, Z.R. and Doyle, A.K., 2003, Characterising proteolytic cleavage site activity using bio-basis function neural network. *Bioinformatics*, **19**, 1741–1747.
[6] Berry, E.A., Dalby, A.R. and Yang, Z.R., 2004, Reduced bio-basis function neural network for identification of protein phosphorylation sites: Comparison with pattern recognition algorithms. *Computational Biology and Chemistry*, **28**, 75–85.
[7] Yang, Z.R. and Thomson, R., 2005, Bio-basis function neural network for prediction of protease cleavage sites in proteins. *IEEE Transactions on Neural Networks*, **16**, 263–274.
[8] Yang, Z.R., 2005, Prediction of caspase cleavage sites using Bayesian bio-basis function neural networks. *Bioinformatics*, **21**, 1831–1837.

[9] Yang, Z.R., Thomson, R., McNeil, P. and Esnouf, R., 2005, RONN: Use of the bio-basis function neural network technique for the detection of natively disordered regions in proteins. *Bioinformatics*, **21**, 3369–3376.

[10] Yang, Z.R. and Chou, K.C., 2004, Predicting the O-linkage sites in glycoproteins using bio-basis function neural networks. *Bioinformatics*, **20**, 903–908.

[11] Kaufman, L. and Rousseeuw, P.J., 1987, Clustering by means of medoids. In: Y. Dodge (Ed.) *Statistical Data Analysis Based on the L_1 Norm* (Amsterdam: North Holland/Elsevier), pp. 405–416.

[12] Kaufman, L. and Rousseeuw, P.J., 1990, *Finding Groups in Data, An Itroduction to Cluster Analysis* (Brussels: John Wiley).

[13] Zadeh, L.A., 1965, Fuzzy sets. *Information and Control*, **8**, 338–353.

[14] Pawlak, Z., 1991, *Rough Sets, Theoretical Aspects of Resoning About Data* (Dordrecht: Kluwer).

[15] Dubois, D. and Prade, H., 1990, Rough fuzzy sets and fuzzy rough sets. *International Journal of General Systems*, **17**, 191–209.

[16] Saad, P., Shamsuddin, S.M., Deris, S. and Mohamad, D., 2002, Rough set on trademark images for neural network classifier. *International Journal of Computer Mathematics*, **79**, 789–796.

[17] Lee, H.-S. and Chou, M.-T., 2004, Fuzzy forecasting based on fuzzy time series. *International Journal of Computer Mathematics*, **81**, 781–789.

[18] Skowron, A., Swiniarski, R.W. and Synak, P., 2005, Approximation spaces and information granulation. *Transactions on Rough Sets*, **3**, 175–189.

[19] Ruspini, E.H., 1970, Numerical methods for fuzzy clustering. *Information Science*, **2**, 319–350.

[20] Diday, E., 1975, La methode des nuees dynamiques. *Revue de Statistique Appliquée*, **XIX**, 19–34.

[21] Roubens, M., 1978, Pattern classification problems and fuzzy sets. *Fuzzy Sets System*, **1**, 239–253.

[22] Windham, M.P., 1985, Numerical classification of proximity data with assignment measures. *Journal of Classification*, **2**, 157–172.

[23] Hathaway, R.J. and Bezdek, J.C., 1994, NERF C-Means: Non-Euclidean relational fuzzy clustering. *Pattern Recognition*, **27**, 429–437.

[24] Krishnapuram, R., Joshi, A., Nasraoui, O. and Yi, L., 2001, Low complexity fuzzy relational clustering algorithms for web mining. *IEEE Transactions on Fuzzy System*, **9**, 595–607.

[25] Altschul, S.F., Gish, W., Miller, W., Myers, E. and Lipman, D.J., 1990, Basic local alignment search tool. *Journal of Molecular Biology*, **215**, 403–410.

[26] Altschul, S.F., Boguski, M.S., Gish, W. and Wootton, J.C., 1994, Issues in searching molecular sequence databases. *Nature Genetics*, **6**, 119–129.

[27] Banerjee, M., Mitra, S. and Pal, S.K., 1998, Rough fuzzy MLP: knowledge encoding and classification. *IEEE Transactions on Neural Networks*, **9**, 1203–1216.

[28] Pearl, L.H. and Taylor, W.R., 1987, A structural model for the retroviral proteases. *Nature*, **329**, 351–354.

[29] Miller, M., Schneider, J., Sathayanarayana, B.K., Toth, M.V., Marshall, G.R., Clawson, L., Selk, L., Kent, S.B.H. and Wlodawer, A., 1989, Structure of complex of synthetic HIV-1 protease with substrate-based inhibitor at 2.3 Å resolution. *Science*, **246**, 1149–1152.

[30] Chou, K.C., 1993, Vectorised sequence-coupling model for predicting HIV protease cleavage sites in proteins. *Journal of Biological Chemistry*, **268**, 16938–16948.

[31] Barre-Sinoussi, F., Chermann, J.C., Rey, F., Nugeyre, M.T., Chamaret, S., Gruest, J., Dauguet, C., Axler-Blin, C., Vezinet-Brun, F., Rouzioux, C., Rozenbaum, W. and Montagnier, L., 1983, Isolation of a T-lymphotropic retrovirus from a patient at risk for Acquired Immune Deficiency Syndrome (AIDS). *Science*, **220**, 868–871.

[32] Chou, K.C., 1996, Prediction of human immunodeficiency virus protease cleavage sites in proteins. *Analytical Biochemistry*, **233**, 1–14.