Pseudo Amino Acid Feature-Based Protein Function Prediction using Support Vector Machine and K-Nearest Neighbors

Anjna Jayant Deen[1], Manasi Gyanchandani[2]
Department of Computer Science and Engineering
Maulana Azad National Institute of Technology, Bhopal, India

Abstract—**Bioinformatics facing the vital challenge in protein function prediction due to protein data are available in primary structure, an amino acid sequence. Every protein cell sequence length and size are in different sequence order. Protein is available in 20 amino acid sequence alphabetic order; however, the corresponding information of the membrane protein sequence is insufficient to capture the function and structures of a protein from primary sequence datasets. A challenging task to correctly identify protein structure and function from amino acid sequence. The basic principle of PseAAC (Pseudo Amino Acid Composition) is to generate a discrete number of every protein samples. In each protein, sequence length varies due to protein functions. Some protein sequence length is less than 50, and some are large. Due to this, different sizes of the amino acid sample are chances to lose sequence order information. PseAAC feature generates a fixed size descriptor value in vector space to overcome sequence information loss and is used to further systematic evolution. Therefore machine learning computational tool synthesizes accurate identification of structure and function class of membrane protein. In this study, SVM (Support Vector Machine) and KNN (K-nearest neighbors) based prediction classifier used to identifying membrane protein and their types.**

Keywords*—*Membrane protein types; classifiers; SVM (RBF); KNN; Random Forest; PseAAC

I. Introduction

Bioinformatics is a different field of combination to solve biological problems with computational techniques dealing remarkably in extensive scale information of system biology. The amino acid residue is a part of macromolecules. The membrane protein is the type of protein residing around a cell membrane, or their subcellular locations are also defined as various types of protein. The most genome encodes a membrane protein, during the encoding process to finding genes cell membrane function perform a wide range of synthesis. Membrane cell identification hard due to lack of stability found a more flexible and hydrophobic surface part [8],[9]. However, protein structure finding is still challenging. Learning outer and non-outer membrane cells necessary to develop computational tools for new drug design and genome sequencings [10][26]. The Protein cell determines its energy and several functions; every cell component depends on protein molecules functions, the cell responsible for signalling cell system, and mobilize an intracellular response. The cell membrane of functional and structural properties targets to find disease, drug design, and novel research [15]. Membrane cell misfold work as monitors, changing their shape and action in response to metabolic signals or information from outside the cell causes various disease like Alzheimer's disease, cardiovascular diseases, neurological disorders, and cancer [1],[20],[21]. Membrane proteins sequence combination, 45-55%, is used in protein legend docking and drug design [10]. Membrane functions are the essential element to discover new drugs and genomes [9-10]. Now capturing the features of membrane functions is responsible for the distribution of cell systems and their role. Conventional techniques used in biological experimental to predict the membrane types are costly and tedious [19]. However, a fast, automated, and effective method must be needed to identify unknown protein types. Analysis of the membrane proteins is hard, and most of them will not dissolve in ordinary solvents. Hence, very few structures of membrane protein have been found so far. Many authors were reports [11],[18],[21],[22] have shown NMR to be a powerful instrument for the detection of membrane protein structures; it is expensive and time processing. Therefore demand to construct computational methods that can predict membrane protein characteristics based on their primary sequence would be very helpful. The membrane protein is classified into mainly transmembrane or anchored protein attached to inside and outside the cell. These membrane cells are further classified into eight subtypes: Type-1, Type-2, Type-3, Type-4, Multi-pass are transmembrane protein and were Peripheral, Lipid chain, and GIP are anchored protein [2], [5]. Currently, different kinds of feature extractions and classification methods have been built to be used to predict membrane types. ACC (Amino acid composition) is used in predicting membrane protein types [3],[5],[13], first used by the article [3], but sequence order of information can't store during implementing amino acid composition. Therefore Chou's suggests PseAAC (Pseudo Amino Acid Composition) evaluates the composition value of amino acid in fixed combinations and saves them. Further, many authors [1],[4],[5],[6],[18] have been processed and suggested different techniques to depict protein samples to overcome. PseAAC, feature extraction method, followed by many latest article [13],[18],[5]. Various computational methods based on learning classifiers and ensemble methods have been used for predicting cell membranes in high-performance accuracy. In this study, a novel feature-based machine learning technique to identify the membrane cell. The proposed objective model was to enhance the accuracy of the classification. Each sequence chain of protein features has mapped into a vector space. And, the multiclass membrane to recognize, the better performing multiclass classifier was chosen. Patterns match and similarity were calculated by using the standard test conducted on high dimensional multiclass protein data.

II. Materials and Method

A. Data Sets

The protein data bank has manually annotated proteins collected from Swiss-Prot PDB [14],[16],[17]. In this study, 560459 protein was obtained from a form data source. Datasets are further preprocessed for identifying a non-membrane and membrane protein correctly [19]. Here, 62029 membrane proteins are captured. For finding its types which is in eight classes, are: (i) GPI-anchored, (ii) lipid chain-anchored, (iii) multipass transmembrane, (iv) peripheral, (v) Type-1, (vi) Type-2, (vii) Type-3 and, (viii) Type-4. Further classification of the 62029 membrane proteins data sequence split into 43418 training and 18611 test samples. Table I have shown the sample details [2].

TABLE I. Total Samples in the Dataset

Membrane protein (types)	No. of instances
GPI-anchored	651
Lipid chain anchored	3032
Multipass transmembrane	35480
peripheral	17319
Type-1	2948
Type-2	2194
Type-3	211
Type-4	194
Total	62029

B. Feature Extraction Methods

Feature selection is the main part of the machine learning process [4]. Specific knowledge is useful for identifying membrane types. Without knowing the sequence order, a sequence's composition loses the information and not used further evaluation. PseAAC (pseudo amino acid composition) to prevent the protein sequence order and pattern data. [29]. PseAAC has to generate ordered 50-dimensional vector space for each sequence data to be involved in computational proteomics [20], and sequence length generate 1 dimensional vector space each samples. In [30] suggest that it is feasible to predict membrane protein type when the features are derived directly from the amino acid sequence. A python-based toolkit iFeature integrates and calculating an extracting feature encode into specific properties of amino acid for generating 51 numerical descriptor value.

III. Proposed Methodology

A practical method develops for predicting the function and structure of protein class from its discrete dimensional vector value. For doing this, the main steps are followed by step 1. Collect protein benchmark data, step 2. Establish a well-built prediction algorithm and step 3. Valuable intrinsic relates as an emphasis for the membrane data samples that can match their desire object to predict.

This study is focusing on the 3rd step, a necessity. In this regard, various methods for formulating protein samples. Therefore, they can categorize into various representations as to the discrete value for sequential description. The flow diagram is shown in Fig. 1. In this study, a significant improvement as an order to,

- SVM (RBF) to expand functional parameters reflect in high-dimensional membrane cell descriptors protein and,
- Enhanced predicting results merits the use of further enriched training data samples and identify different types of membrane cell descriptors.
- Integrated features of PseAAC and sequence length are used for analysis to evaluate membrane. The learning model is based on kernel SVM for functional prediction and similarity matches in sequence to a query membrane.
- Unique multiclass are in eight batches—each sample descriptor value is 51D space for supporting multiple membrane types.
- Machine learning classifiers K nearest neighbors and Random Forest was added for simplifying the collective samples via computation of protein functions by multiple types.

Cross-validation is one method to overcome the class imbalance problem. Therefore, in this study, we use k-fold cross-validation. Membrane protein data consisting of N tuple has divided in k=10 folds (D1, D2, D3...D10), and if the N tuple is not divisible by k, then the last part is considered as a (k-1). Here in our estimation, learning using 10-fold cross-validation. A sequence of k = 10 runs is carried out with the decomposition and i^{th} = iteration, and D_i use as test data and other fold as training data. Thus, each tuple uses the same amount of time for training samples, and once for testing. The overall average of each iteration is estimated.

A. PseAAC (Pseudo Amino Acid Composition)

Membrane protein information senses its molecular action. Its process is in a molecules system that allows organisms to endure these basic life processes—various inherited diseases caused by mutations and changes observed in a protein sequence result. The amino acid sequence is a pattern of 20 unique amino acid residues. As per chemical composition, amino acid 20 sets are further categorized into four groups: polar, nonpolar, positive charged, and negatively charged [12]. These are comparable and a varying side chain. Each amino acid has distinct chemical properties due to the different groups' side chains. The 20 amino acids composition computed as in eq. (1), [3].

$$P = [p_1, p_2 \dots p_{20}, p_{20+1} \cdots p_{20+\lambda}]^T \quad (1)$$

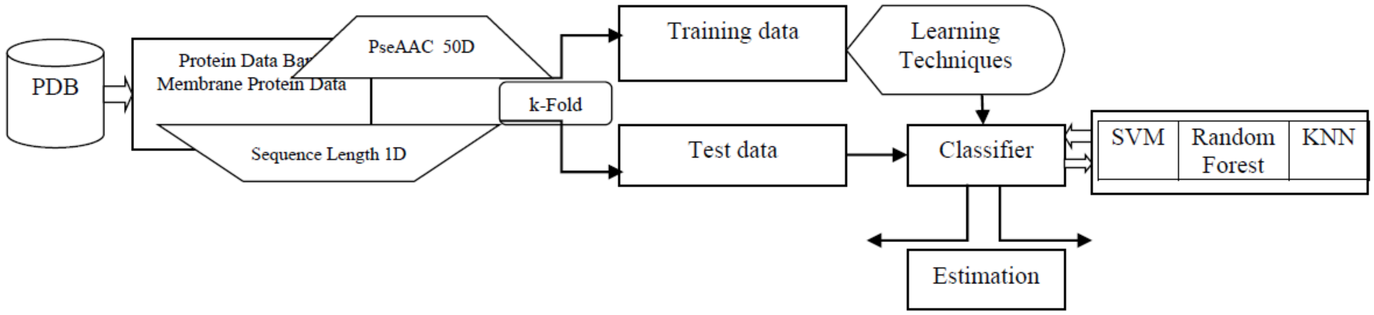

Fig. 1. Proposed Methodology.

The composition portions of amino acids are evaluated by using the mass of 20 amino acids, hydrophilic value and hydrophilic value. $p_1, p_2 \ldots p_{20+\lambda}$ are calculated by eq. (2):

$$p_u = \begin{cases} \frac{f_u}{\sum_{i=1}^{20} f_i + \omega \sum_{k=1}^{\lambda} \tau_k}, & (1 \leq u \leq 20) \\ \frac{\omega . \tau_{u-20}}{\sum_{i=1}^{20} f_i + \omega \sum_{k=1}^{\lambda} \tau_k}, & (20 + 1 \leq u \leq 20 + \lambda) \end{cases} \quad (2)$$

Integrating features descriptor value has produced a verity of amino acid patterns on regular occurrence in the protein sequence. The length of PseAAC depends on the descriptor value. This study uses a set of 30 amino acid composition. Thus the feature dimension from PseAAC is 50 (20+30=50D) descriptor vector space [2],[23].

B. Sequence to Integer Encoding

This method is configured for a particular integer value (range from 1-20) with 20 amino acid residues made from the protein sequence. A protein sequence can be translated to an integer sequence by replacing each letter with a corresponding mapping integer value. The sum of residues in the sequence is proportional to its weight. For instance, in a protein sequence, AJKJLMLLK, L, is seen three times. The weight of L is then measured as 3/9=0.33. Then. We use the following formula in eq. (3) to find the weight of a residue:

$$w_i = \frac{n_i}{L} \quad (3)$$

where, w_i is the weight of ith residue, n_i is the number of occurrence of ith residue in the protein sequence and L is the length of the protein sequence. A weighted total volume of each residue represents the required protein sequence and is performed by measuring each residue's weight. The numerical value encoded is then found as follows.

$$SEQ_{encoded} = k_1.w_1 + k_2.w_2 + \cdots + k_{20}w_{20} \quad (4)$$

Where $k_i$is the ith residue's mapped integer value and w_i is the corresponding weight of the residue got from equation (4).the resultant values gives one dimensional data for protein sequences. Where, weight factor $is\ \omega$ (set to 0.05) and τ_k is the kth tier correlation factor that represents all correlation order of the kth-most continuous residues.

C. Classification Algorithm

Feature-based classification algorithm mapping the input data samples into the desired class, model build to predict class labels for unseen samples, the main part of machine learning applied in all fields are in bioinformatics and data science. These were training techniques used to train most 70% data samples, and classifier testing test data sets 30%, respectively. This study classifier model based on SVM, KNN, and RF (Random Forest) used to classify the membrane features pseudo amino acid composition and sequence length descriptors into eight types.

1) Support Vector Machine (SVM): In the bioinformatics data source, protein information has generally gathered in an amino acid sequence. However, a knowledge-based learning system dealing with homogeneous and heterogeneous datasets still needed some basic models based upon classification and clustering techniques. To implement the classifier support vector machine trendy and powerful for predicting protein structure and function. SVM (support vector machine) classification techniques have been used for dual-mode separation as a binary or multiclass. SVM outline draw hyper-plane, which separates the decision surface data into two different class [31]. The use of the SVM learning model in high dimensional datasets creates a multiclass problem, so resolving that need to build a modified classification technique. In this study, unique features are based on a novel classifier model design on predicting membrane protein of achieving high accuracy for multiclass in the high-dimensional protein data source.

SVM transforms the given data first into a large vector space and then draws the maximum hyperplane margin to separate non-linear datasets, represented in Fig. 2. The functions for Radial Basic Function (RBF) in the SVM algorithm were used: Step 1. The built a feature vector from the input sequence. It can represent classes based on PseAAC and Sequence length properties. Step 2. RBF kernel selects to predict function while training eq. (5-16). Step 3 Selection of the prime parameter during training kernel function fit data to get maximum accuracy eq. (17-22)[2]. When the SVMs are using for the classification, the known set ({+1, -1}), marked training data is segregated by a hyperplane, which is as far as possible distant from positive negative samples. [See "optimal separating plant" (OSH) in Fig. 2]. The test data 'plot' then defines the positive or negative OSH for the high-dimensional sphere. The kernel model enables SVMs to work in combination with the nonlinear mapping into a function space to classify membrane protein types. For these problems, SVM is not linearly detachable. The SVM's optimal separating hyperplane within functional space is a nonlinear decision limit within the input space.

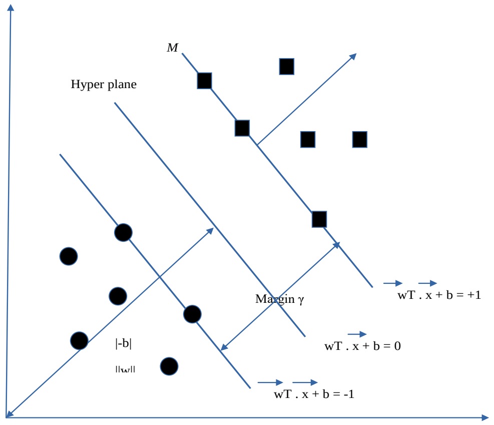
M
Hyper plane
Margin γ
$\overrightarrow{w}T . \overrightarrow{x} + b = +1$
$\overrightarrow{w}T . x + b = 0$
$\overrightarrow{w}T . \overrightarrow{x} + b = -1$
|-b|
||w||

Fig. 2. Hyperplane and Margin Description. Samples of Class -1 and Class +1 are Represented Respectively by the Circular Dots and Square Dots.

Linear SVM classification

where $\vec{w} = (w_1, w_2, \dots, w_n)^T$ is a vector of n elements.Consider, the training data of two groups of n instance $(\overrightarrow{x_1}, y_1), (\overrightarrow{x_2}, y_2), \dots, (\overrightarrow{x_n}, y_n)$, i = 1, 2, . . . , n, on each instances, y (i=1..n), i = 1, 2, . . . , n, where specified a weight vector $\vec{w}$ and bias b,and $\overrightarrow{x_i} \epsilon R^N$ is an N dimensional space, and $y_i \epsilon \{-1, +1\}$ is the class index.

$$\overrightarrow{w}^T.\overrightarrow{x_i} + b \geqslant 1, y_i = +1, \quad (5)$$

$$\vec{w}^T.\overrightarrow{x_i} + b \leqslant -1, y_i = -1, \quad (6)$$

The vector of n elements is where $\vec{w} = (w_1, w_2, \dots, w_n)^T$. Uniformities (1) , (2) can be fused into one.

$$y_i(\vec{w}^T.\overrightarrow{x_i} + b) \geqslant 1, \qquad i = 1, 2, . . . , n. \quad (7)$$

For each training group, there are a number of hyperplanes. SVM's classification aims to create an optimum weight $\overrightarrow{w_0}$ and an optimal bias b0 to achieve the maximum margin between the training data and the chosen hyperplane. The defined hyperplane by $(\overrightarrow{w_0})$ and b0 is optimal separating hyperplane.Any hyperplane can be represented as equated in eq. (8).

$$\vec{w}^T.\overrightarrow{x_i} + b = 0 \quad (8)$$

and the difference between the two margins is in eq. (9)

$$\gamma(\vec{w}, b) = \min_{\{\vec{x}|y=+1\}} \frac{\vec{x}^T.\vec{w}}{\|\vec{w}\|} - \max_{\{\vec{x}|y=-1\}} \frac{\vec{x}^T.\vec{w}}{\|\vec{w}\|}. \quad (9)$$

The optimum separating hyperplane is being identified by raising the distance above or reducing the norm of $\|\vec{w}\|$ by trying to restrict discrimination eq. (7), and

$$\gamma_{max} = \gamma(\overrightarrow{w_0}, b_0) = \frac{2}{\|\overrightarrow{w_0}\|}. \quad (10)$$

The following Lagrangean saddle point provides solutions to the above problems with optimization

$$L(\vec{w}, b, \alpha) = \frac{1}{2}\vec{w}^T.\vec{w} - \sum_{i=1}^{n} \alpha_i \left[y_i(\vec{w}^T.\vec{x}_i + b) - 1\right], \quad (11)$$

where $\alpha \geqslant 0$ are Lagrange multipliers.To solve the quadratic programming problem, the gradient of $L(\vec{w})$ to $L(\vec{w}, b, \alpha)$ disappears in respect of $\vec{w}$ and b., which gives a calculation of the following terms:

$$\frac{\delta L}{\delta \vec{w}}|_{\vec{w}=\overrightarrow{w_0}} = 0, \text{and} \frac{\delta L}{\delta \vec{w}}|_{\vec{w}=\overrightarrow{w_0}} = 0$$

$$\overrightarrow{w_0} = \sum_{i=1}^{n} \alpha_i y_i \overrightarrow{x_\imath}, \quad (12)$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0. \quad (13)$$

Via replacement of Eqs. (12, and 13) into (11), Maxing the following expression becomes the quadratic programming (QP) problem:

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \left(\vec{x}_i^T.\vec{x}_j\right) \quad (14)$$

in the constraints $\sum \alpha_i y_i = 0$ and $\alpha_i \geqslant 0, i = 1,2, \dots, n$.

Non-zero α_i coefficients are among Eq. (14) solutions at the two optimal margins, and is known as vectors support (SV). The bias b0 can be estimated accordingly:

$$b_0 = -\frac{1}{2}\left(\min_{\{\overrightarrow{x_\imath}|y_i=+1\}} \overrightarrow{w}_0^T.\overrightarrow{x_\imath} + \max_{\{\overrightarrow{x_\imath}|y_i=-1\}} \overrightarrow{w}_0^T.\overrightarrow{x_\imath}\right) \quad (15)$$

The decision function that divides the two groups can be written as after evaluating the support vector and bias

$$f(\vec{x}) = \text{sign}[\sum_{i=1}^{n} \alpha_i y_i \vec{x}_i^T.\vec{x} + b_0] = \text{sign}[\sum_{SV} \alpha_i y_i \vec{x}_i^T.\vec{x} + b_0] \quad (16)$$

Non-linear SVM classification

Since membrane protein types are typically nonlinear, these problems have been implemented in the SVM [30]. In the input space X, the original training data $\vec{x}$ are translated into a high-dimensional F-function through the operator kernel Mercer K [34], in which the optimum separating hyperplane is formed. The set of classifiers will be converted into the form in mathematical terms.

$$f(\vec{x}) = \text{sign}\left[\sum_{i\epsilon\{SV\}} \alpha_i y_i K(\vec{x}_i, \vec{x}) + b_0\right], \quad (17)$$

Where K is a symmetric positive function that fulfills the conditions of Mercer.

$$K(\vec{x}, \vec{y}) = \sum_{m=1}^{\infty} \alpha_m \phi(\vec{x}^T).\phi(\vec{y}), \quad \alpha_m \geqslant 0$$

$$\iint K(\vec{x}, \vec{y})g(\vec{x})g(\vec{y})d\vec{x}d\vec{y} > 0, \int g^2(\vec{x})d\vec{x} < \infty \quad (18)$$

The kernel is a valid internal product in the input field

$$K(\vec{x}, \vec{y}) = \phi(\vec{x}^T).\phi(\vec{y}). \quad (19)$$

The dual Lagrangian in the F space, given in Eq. [14].

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K\left(\vec{x}_i, \vec{x}_j\right) - \lambda \sum_{i=1}^{n} \alpha_i y_i \quad (20)$$

subject to $\sum_{i=1}^{n} \alpha_i y_i = 0$and $\qquad \alpha \geqslant 0, i = 1,2, \dots, n$

and the decision function is

$$f(\vec{x}) = sign\left[\sum_{i\epsilon\{SV\}} \alpha_i y_i K(\vec{x}_i, \vec{x}) + b_0\right], \quad (21)$$

Where

$$b_0 = -\frac{1}{2}\left\{\min_{\{\overrightarrow{x_\iota}|y_i=+1\}}\left(\sum_{j\epsilon\{SV\}}\alpha_j\, y_j K\left(\overrightarrow{x_\iota},\overrightarrow{x_J}\right)\right) + \max_{\{\overrightarrow{x_\iota}|y_i=-1\}}\left(\sum_{j\epsilon\{SV\}}\alpha_j\, y_j K\left(\overrightarrow{x_\iota},\overrightarrow{x_J}\right)\right)\right\} \tag{22}$$

SVM has employed a variety of candidate kernel functions, including poly-nominal $K(\vec{x},\vec{y}) = (1 + \vec{x}.\vec{y})^d$, Gaussian RBF $K(\vec{x},\vec{y}) = exp\left(-\frac{\|\vec{x}-\vec{y}\|^2}{2\sigma^2}\right)$, exponential RBF $K(\vec{x},\vec{y}) = exp\left(-\frac{\|\vec{x}-\vec{y}\|}{2\sigma^2}\right)$, and their kernel summing combinations of kernel coefficients products [33]. The Gaussian RBF kernel function is employed in this work to predict membrane protein types.

2) K-Nearest Neighbor (KNN): K- nearest neighbor classifier, input data based on instance-based learner, into its feature space. KNN is based on the neighbor set that will be found near k object. KNN locate on majority voting among the k-data samples. Which store all value of the training data and wait till new data arrived to be classified on similarity measures or as a pattern matching techniques [2]. K-nearest finding based on Euclidean distance eq. (23). To classify membrane proteins, predicting the functional types of membrane proteins is indispensable [24]. Therefore similarity measures formula as the Euclidean distance (E_{Dis}) phrase between two points(y_1, y_2) [27].

$$E_{Dis}(y_1, y_2)=\sum_{r=1}^{N}\sqrt{(yr1-yr2)^2} \tag{23}$$

The next steps are to generalize K-nearest neighbor classifier innovations. Metric distance and functions are measured to measure the distance between characteristics. The k-parameter must be designed for training data.

IV. Results and Discussion

In this study, two different types of feature extraction techniques, namely PseAAC and sequence to integer encoding, are used, giving a feature vector of 62029 instances in a row and 51-dimension in a column classified by the proposed model, were 43418×51 training and 18611×51 test samples are implemented. For getting best accuracy, various type of classifiers such as SVM(Support vector machine), KNN (K Nearest neighbor), RF (Random Forest), classifiers are used and based on the result obtained from them, and the model is built [2],[16],[22],[32].

A. Accuracy

The number of instances rightfully predicted out of total number of instances in eq. (24).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{24}$$

where TP is total number of true positive, FN is total number of false-positives, $TN$is total number of true negatives and $FP$is total number of false positives [3],[4]. The overall accuracy as eq. (24) of different classifiers are shown in Table II.

TABLE II. Overall Accuracy of different Classifiers

Classifier	Accuracy
Random Forest	89.38
KNN	93.24
SVM (RBF)	85.86

B. Specificity

Specificity of classifiers are good as the true negatives are correctly identified as calculated by eq. (25).

$$Specificity = \frac{TN}{TN+FP} \tag{25}$$

The specificity of classifiers is shown in Table III. The specificity range is 85% to 99% because TNR (true negative rate) is good. Specificity results noticed that wrongly classified samples are significantly less in KNN classifier [2].

C. Sensitivity

The classifier can correctly predict in eq. (26) the true positives shown in Table IV.

$$Sensitivity = \frac{TP}{TP+FN} \tag{26}$$

D. F-measure

Every model design to handle various kinds of multiclass problem to look at the accuracy of that model as the number of samples corrects predicted and misclassified from all prediction. Confusion Matrix gives detailed information about the failure in predictions for an unseen dataset sample. The F1 measures mathematically computed in eq. (27-28) recorded precision and recall balance values.

$$Precision = \frac{TP}{TP+FP} \tag{27}$$

$$F_{measure} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{28}$$

TABLE III. Specificity

Types	Specificity of classifiers		
	Random Forest	KNN	SVM
GPI	0.88	0.99	0.99
Lipid	0.93	0.95	0.97
Multi-pass	0.95	0.96	0.96
Peripheral	0.97	0.96	0.96
Type1	0.94	0.97	0.98
Type2	0.94	0.95	0.85
Type3	0.96	0.95	0.93
Type4	0.93	0.93	0.94

TABLE IV. Sensitivity

Types	Sensitivity of classifiers		
	Random Forest	KNN	SVM
GPI	0.06	0.25	0.05
Lipid	0.53	0.65	0.47
Multi-pass	0.97	0.96	0.99
Peripheral	0.93	0.95	0.82
Type1	0.59	0.64	0.35
Type2	0.43	0.63	0.51
Type3	0.49	0.64	0.53
Type4	0.15	0.35	0.26

F1 balanced accuracy are used as a better metrics for a multi class imbalanced dataset classification task. F1-measure of various classifiers are shown in Table V.

E. Mathew's Correlation Coefficient (MCC)

Standard measure in machine learning MCC was suggested in 1975 by Brain W. Matthews [28]. Matthew's correlation coefficient is balanced in binary classifications into true and false positives and negatives classes [25]. It found a degree of correlation in the predicted level. It returns a value between -1 and +1.were + represents a perfect prediction, and -1 represents the entire disqualifying range between predicting and observation eq. (29), shown in Table VI. If D datasets and N is the total number of the outcome of true and false positives and negatives views from a single instance, the Matthews correlation coefficient best such measures in larger dataset achieves a high proportion of correct predictions from the confusion matrix [11].

$$\mathrm{MCC} = \frac{\mathrm{TP}\times\mathrm{TN}-\mathrm{FP}\times\mathrm{FN}}{\sqrt{(\mathrm{TP+FP})(\mathrm{TP+FN})(\mathrm{TN+FP})(\mathrm{TN+FN})}} \quad (29)$$

From confusion matrix observation, is found that all classifiers perform well on multiclass datasets, KNN measure better as compared to other classifiers on the various parameter such as precision, recall, specificity, accuracy and F1-measure in MCC value.

Prediction result indicated that shown in Fig. 3, the proposed method achieved high prediction accuracy for the independent datasets. The different classifier prediction performance measures in confusion matrix results are represented in Table VII, VIII, and IX. Statistical Problem handed through machine learning is known as a confusion matrix. The proposed learning model, field error defined in the matrix table, also describes the classifier's efficiency to testing data samples, the actual value visualizing true identity on an algorithm. The confusion matrix makes easy identification of confusion between classes or mislabeled class of others in performance on various scales.

F. Confusion Matrix Results

The classfiers output was examined using independent tests [7]. The ensemble classifiers such as Random Forest 89.38% value, SVM 85.86%, and KNN value is improved, i.e., a maximum of 93.24%. Membrane Protein is a multilabel dataset. The classification results of models are shown in the confusion matrix, the total number of passable similarity matching with other multi-class functions. The confusion matrix is a critical way to summarize machine learning classifiers' performance, like SVM, RF (Random Forest), and KNN classifiers. This Square matrix consists of based on features PseAAC and Sequence Length encoding. There are 62029 rows (43418 training rows and 18611 test rows in datasets) in total protein sequence and 51D descriptor size in columns. Moreover, this is listing the number of instances as absolute or relative actual class vs. predicted class ratio. The confusion matrix results demonstrate a major role in prediction identification in terms of accuracy, precision, recall, and F-1 score. SVM, KNN, and RF three learning techniques were analyzed based on outcome comparisons to find model performance. Parameter of the confusion matrix observed that the learning model KNN performs well in all eight membrane protein types. Overall, classifier performance observed a high boosting rate for large data training samples. Multipass large data sample observed F1-Score 95% in RF, 96% in KNN, and 89% in SVM where GPI class score poorly 7% in RF, 25% in KNN, and 5% in SVM. Peripheral and multipass transmembrane class are more sensitive in all classifiers, where GPI and Type-4 found a less sensitive class, with 99% GIP specificity found in KNN and SVM. In all classifiers observation found, the integrated 51D features of protein sequences and different patterns length, KNN classifier, provide better performance for membrane protein types, as shown in Fig. 4.

TABLE V. F1 SCORE

Types	**F1**	**Score**	
	Random Forest	**KNN**	**SVM**
GPI	0.07	0.25	0.05
LIPID	0.60	0.65	0.56
MULTI-PASS	0.95	0.96	0.91
PERIPHERAL	0.91	0.92	0.89
TYPE1	0.68	0.69	0.51
TYPE2	0.61	0.69	0.67
TYPE3	0.66	0.74	0.70
TYPE4	0.27	0.49	0.28

TABLE VI. MCC VALUE OF DIFFERENT CLASSIFIERS

Classifier	**MCC Value**
Random Forest	80.6
KNN	85.4
SVM(RBF)	82.2

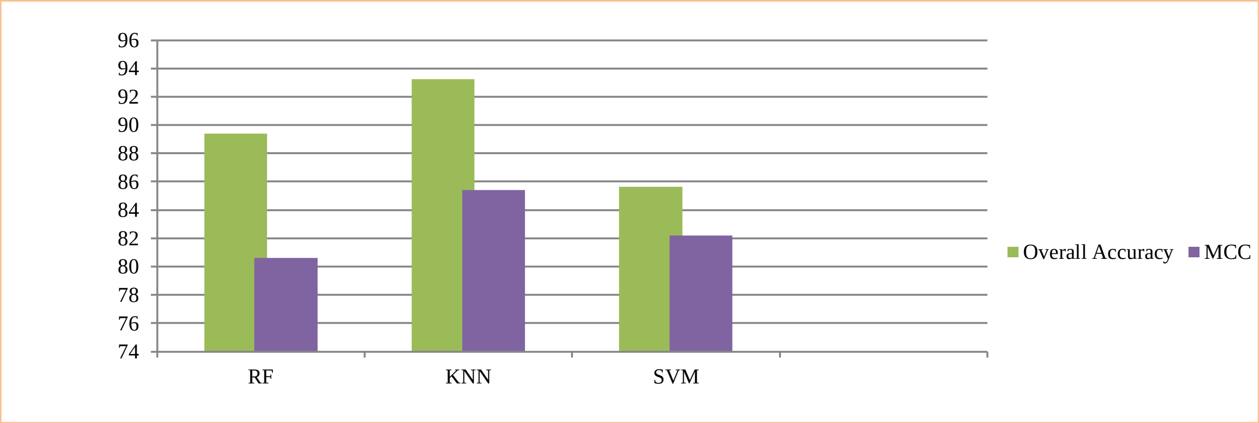

Fig. 3. Overall Accuracy and MCC Performance Scale in Bar-Chart.

TABLE VII. RESULT OF SVM RBF CONFUSION MATRIX; OVERALL ACCURACY: 85.86018485678972

	Gpi	**Lipid**	**Multi-pass**	**Peripheral**	**Type1**	**Type2**	**Type3**	**Type4**
Gpi	9	139	37	1	2	0	0	0
Lipid	158	418	262	44	1	2	0	0
Multi-pass	0	10	10554	39	5	5	0	0
Peripheral	0	19	901	4294	2	2	0	0
Type1	1	4	570	20	317	2	0	0
Type2	0	4	291	32	2	340	0	0
Type3	0	2	15	11	0	0	32	0
Type4	0	0	38	14	0	0	0	10

TABLE VIII. RESULT OF KNN CLASSIFIER CONFUSION MATRIX; OVERALL ACCURACY: 93.24188725885324

	Gpi	**Lipid**	**Multi-pass**	**Peripheral**	**Type1**	**Type2**	**Type3**	**Type4**
Gpi	53	123	9	9	14	3	0	0
Lipid	111	586	41	121	33	14	0	1
Multipass	14	62	10184	262	62	42	3	1
Peripheral	13	56	119	4962	34	41	2	1
Type1	21	24	145	99	545	14	0	1
Type2	3	21	71	127	28	422	1	0
Type3	0	4	4	7	4	2	38	0
Type4	2	7	1	19	1	4	0	18

TABLE IX. RESULT OF RANDOM FOREST CLASSIFIER CONFUSION MATRIX; OVERALL ACCURACY: 89.38606749422322

	Gpi	**Lipid**	**Multi-pass**	**Peripheral**	**Type1**	**Type2**	**Type3**	**Type4**
Gpi	13	152	26	7	9	4	0	0
Lipid	132	502	95	153	13	12	0	0
Multi-pass	1	41	10355	193	36	4	0	0
Peripheral	3	38	294	4863	25	5	0	0
Type1	3	20	299	32	489	6	0	0
Type2	1	21	186	139	20	303	0	0
Type3	0	4	13	9	4	0	29	0
Type4	0	0	11	30	2	0	0	8

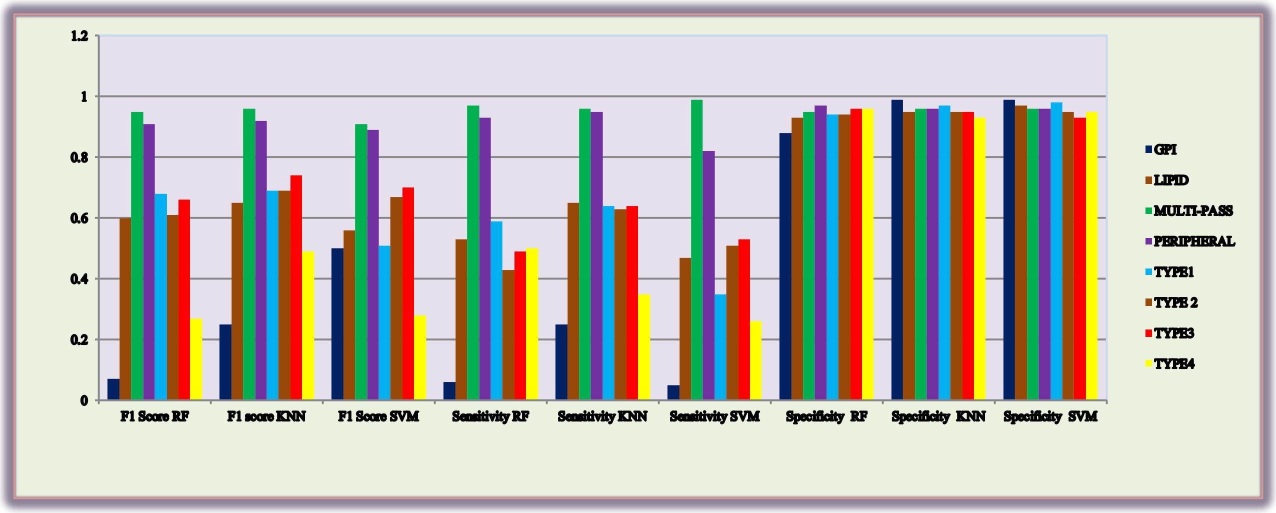

Fig. 4. F-1 Score, Sensitivity and Specificity Bar Chart of Membrane Protein Types.

V. Conclusion

The proposed model objective is to score proper functions based on PseAAC and Sequence length of 51 descriptor features. This study confirmed a large sample size and fine-tuning techniques enforcement provides to build superior models that allow integrations of variant feature levels. In KNN, learning strategies based on the nearest neighbor's weight vector exploit the overall membrane protein types in a biological cell network to find the correct eight types of membrane protein. Prediction based on 51D feature vectors is used to learn three classifiers Random forest, K-nearest neighbors, and SVM. Python programming is supported by many machine learning techniques potent today. Python library provides many functions to learn about the Specify-Compile-Fit workflow that will be easy to make predictions. It can build simple necessary tools for various learning methods and generate predictions with them. Real classification results show that the proposed model achieves the desired goal significantly.

References

[1] Ali, F., Hayat, M., "Classification of membrane protein types using voting feature interval in combination with Chou's pseudo amino acid composition". J. Theor, Biol. 384 78-83,2015.

[2] Anjna J.Deen, Manasi Gyanchandani, "Improved Machine Learning using Adaptive Boosting algorithm in Membrane Protein Prediction", International Journal of Innovative Technology and Exploring Engineering Vol.8(12), page 3131-3137,2019.

[3] Cai, Y.D., Chou, K.C., "Predicting membrane protein type by functional domain composition and pseudo amino acid composition". J. Theor, Biol. 238, 395-400,2006.

[4] Cai, Y.D., Ricardo, P.W., Jen, C.H., Chou, K.C., "Application of SVM to predict membrane protein types". J. Theor. Biol. 226 (4), 373-376,2004.

[5] Chen, W., Ding, H., Feng, P., "iACP: a sequence-based tool for identifying anti-cancer peptides", Oncotarget 7, 16895-16909,2016.

[6] Chen, W., Lin. H., "Pseudo nucleotide composition or PseKNC: an effective formulation for analyzing genomic sequences". Mol. Biosyst. 11, 2620-2634,2015.

[7] Chen, Y.K., Li, K.B., "Predicting membrane protein types by incorporating protein topology, domains, signal peptides, and physiochemical properties into the general form of Chou's pseudo amino acid composition". J. Theor, Biol. 318, 1-12,2013.

[8] Elisabeth P carpenter, Konsatinos Beis, Alexander D , So Iwata., Overcoming the challenges of membrane protein crystallography. Curr Opin Struct Biol.PMC ID. 18(5), 581-586,2008.

[9] [9] Chen Z, Zhao P, Li F, Leier A, Marquez LogoTT, Ang Y, Webb GI, Smith AI, Daly RJ, Chou KC, Song J." iFeature: a python package and web server for feature extraction and selection from protein and peptide sequence".Bioinformatics Volume 34 issue 14,15 page2499-2502, 2018.

[10] Qiao Ning, Zhiqiang Ma, Xiaewi Zhao. "dformKNN -PseAAC detecting formylation site from protein sequence using K-nearest neighbor algorithm via chou's 5-step rule and pseudo component", Journal of Theoretical Biology.470,43-49,2019.

[11] Xiao-Sheng, Run-Jing Zhan. "clustering based subset ensemble learning method for imbalance data", proceeding 13, ICMLC ;35-39,2013.

[12] Marco Punta, Lucy R. Forrest, Henry Bigelow, Andrew Kernytsky, Jinfeng Liu, and BurkhardRost. "membrane protein prediction methods" NIH Public access PMC ; 41(4): 460–474,2007.

[13] Cheng, X., Zhao, S.G., Xiao, X., "iATC-mISF: a multi-label classifier for predicting the classes of anatomical therapeutic chemicals". Bioinformatics 33, 341-346,2017.

[14] Chou, K.C.,"Insights from modelling three-dimensional structures of the human potassium and sodium channels". J. Proteome Res. 3, 856-861,2004.

[15] Chou, K.C., "Impacts of bioinformatics to medicinal chemistry". Med. Chem. 11, 218-234,2015.

[16] Chou, K.C., Elrod, D.W., "Prediction of membrane protein types and sub cellular locations". Proteins Struct. Funct. Bioinf. 34 (1), 137-153,1999.

[17] Chou, K.C., Shen, H.B., "MemType-2L: a web server for predicting membrane proteins and their types by incorporating evolution information through Pse-PSSM". Biochem. Biophys. Res. Commun. 360 (2), 339-345,2007.

[18] Gao, Q.B., Ye, X.F., Jin, Z.C., He, J., "Improving discrimination of outer membrane proteins by fusing different forms of pseudo amino acid composition". J. Anal. Biochem. 398, 52-59,2010.

[19] Golmohammadi, K.S.K., Crowley, L., Reformat, M., "Classification of cell membrane proteins". Front, Convergence Biosci. Inf. Technol. 153-158,2007.

[20] Golmohammadi, S.K., Kurgan, L., Crowley, B. Reformat, M., "Amino acid sequence based method for prediction of cell membrane protein types". Int. J. Hybrid Inf. Technol. 1 (1), 95-109,2008.

[21] Hayat M., Khan, A., "Predicting membrane protein types by fusing composite protein sequence features into pseudo amino acid composition". J. Theor. Biol. 262, 10-17,2011.

[22] Hayat M., Khan, A., Yeasin, M., "Prediction of membrane proteins using split amino acid and ensemble classification". Amino Acids 42 (6), 2447-2460,2012.

[23] Wang, S.Q., Yang, J., Chou, K.C., "Using stacked generalization to predict membrane protein types based on pseudo-amino acid composition". J. Theor. Biol. 242 (4), 941-946,2006.

[24] Wan, S., M.W., Kung, S.Y., "Mem-mEN: predicting multi-functional types of membrane proteins by interpretable elastic nets". IEEE/ACM Trans. Comput. Biol. Bioinf. Doi: 10.1109/TCBB.2015. 2474407,2015.

[25] Jia, J., Zhang, L., Liu, Z.,Psumo-CD: "predicting sumoylation sites in proteins with covariance discriminate algorithm by incorporating sequence-coupled effects into general PseAAC", Bioinformatics 32, 3133-3141,2016.

[26] Cheng, X., Zhao, S.G., Xiao, X., "iATC-mHyb: a hybrid multi-label classifier for predicting the classification of anatomical therapeutic chemicals". Oncotarget doi:10.18632/oncotarget.17028,2017.

[27] Shen, H., Chou, K.C., "Using optimized evidence-theoretic K-nearest neighbour classifier and pseudo-amino acid composition to predict membrane protein types". Biochem. Biophys. Res. Commun. 334 (1), 288-292,2005.

[28] B.W. Matthews, "Comparison of the predicted and observed secondary structure of T4 phage lysozyme", Protein Structure. 405 (2),pp.442–451,1975.

[29] Mahdavi, A., Jahandideh, S., "Application of density similarities to predict membrane protein types based on pseudo-amino acid composition". J. Theor. Biol. 276, 132-137,2011.

[30] Nanni, L., Lumini, A., "An ensemble of support vector machines for predicting the membrane protein type directly from the amino acid sequence". Amino Acids 35 (3), 573-580,2008.

[31] Wang, M., Yang, J., Liu, G.P., Xu, Z.J., Chou, K.C., "Weighted-support vector machines for predicting membrane protein types based on pseudo-amino acid composition". Protein Eng. Des. Sel. 17 (6), 509-516,2004.

[32] Shen, H.S., Chou, K.C., "Using ensemble classifier identify membrane protein types". Amino Acids 32, 483-488,2007.

[33] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines, Cambridge University, Cambridge, 2000.

[34] V.N.Vapnik, The Nature of Statistical Learning Theory, second ed., Springer, New York, 1999.