Mathematical Biology and Bioinformatics
2023. V. 18. № 1. P. 113–127. doi: 10.17537/2023.18.113

========== BIOINFORMATICS ==========

Exploiting ensemble learning and negative sample space for predicting extracellular matrix receptor interactions

Abhigyan Nath[1], Sudama Rathore[1], Pangambam Sendash Singh*[2]

[1]*Department of Biochemistry, Pt. Jawahar Lal Nehru Memorial Medical College, Raipur, India*
[2]*Department of Computer Science, Banaras Hindu University, Varanasi, India*

Abstract. The extracellular matrix (ECM) is best described as a dynamic three-dimensional mesh of various macromolecules. These include proteoglycans (e.g., perlecan andagrin), non-proteoglycan polysaccharides (e.g., hyaluronan), and fibrous proteins (e.g., collagen, elastin, fibronectin, and laminin). ECM proteins are involved in various biological functions and their functionality is largely governed by interaction with other ECM proteins as well as trans-membrane receptors including integrins, proteoglycans such assyndecan, other glycoproteins, and members of the immunoglobulin superfamily. In the present work, a machine learning approach is developed using sequence and evolutionary features for predicting ECM protein-receptor interactions. Two different feature vector representations, namely fusion of feature vectors and average of feature vectors are used within corporation of the best representation employing feature selection. The current results show that the feature vector representation is an important aspect of ECM protein interaction prediction, and that the average of feature vectors performed better than the fusion of feature vectors. The best prediction model with boosted random forest resulted in 72.6 % overall accuracy, 74.4 % sensitivity and 70.7 % specificity with the 200 best features obtained using the ReliefF feature selection algorithm. Further, a comparative analysis was performed for negative sample subset selection using three sampling methods, namely random sampling, *k*-Means sampling, and Uniform sampling. *k*-Means based representative sampling resulted in enhanced accuracy (75.5 % accuracy with 80.8 % sensitivity, 68.1 % specificity and 0.801 AUC) for the prediction of ECM protein-receptor interactions in comparison to the other sampling methods. On comparison with other three state of the art PPI predictors, it is observed that the latter displayed low sensitivity but higher specificity. The current work presents the first machine learning based prediction model specifically developed for ECM protein-receptor interactions.

Key words: *ECM receptor interaction, Boosting; Boosted Random Forest, ReliefF, Random Sampling, k-Means, Uniform Sampling.*

INTRODUCTION

Extracellular matrix (ECM) proteins play many important roles in cell mechanics, cytoskeletal organization, cell growth, cell differentiation, cell migration, tissue development, and other cellular processes [1, 2]. As such, they are directly involved in the transmission of information from the extracellular environment to the intracellular environment and vice-versa [3]. The ECM consist of three major classes of molecules: proteoglycans (e.g., perlecans, agrins, etc.); non-proteoglycan polysaccharides, such as hyaluronic acid; and fibrous proteins such as elastin, collagens, laminin, and fibronectin, out of which collagen is the most abundant [4]. Some ECM proteins bind only single specific ECM proteins while others bind several ECM proteins.

*sendashpangambam@gmail.com

Among the proteins that are part of the ECM and link it to cells are integrins and syndecans, two families of cell surface transmembrane receptors. Integrins are composed of two non-covalently associated transmembrane glycoprotein subunits called α and β. 24 types of α and 9 types of β subunits yield adversity of human integrins further increased via alternative splicing, governing specificity for different ligands. Integrins specifically interact with proteins containing the ARG-GLY-ASP motif [5]. Integrins function as adhesion molecules that connect ECM proteins such as fibronectin and laminin to the cell's actin cytoskeleton [6] and are involved in various biological functions [5]. Integrin-ligand interactions have been found to play important roles in many signal transduction pathways, in cell proliferation, apoptosis [4, 6], cell migration, cell adhesion and in diseases such as LAD1 (leukocyte adhesion deficiency) [5]. The role of integrins in cancer progression has also been reported [3]. Syndecans are proteoglycans enriched in dibasic sequences and peculiar in that they are using a transmembrane domain instead of glycosyl-phophatidyl inositol linkage for their attachment to the plasma membrane [7, 8]. Based on the presence and location of GAG binding sites, syndecans are sub classified into four families: syndecans 1 to 4 [9].

Laminin and tenascin fall under the category of adhesive glycoproteins. Laminins are mainly located in basement membranes and play a salient role in the interaction between cells and extracellular matrix. Some studies have found association of laminins with a number of diseases such as angiogenesis and cancer [4].

The ECM and ECM-receptor interactions are also involved in various diseases. For instance, recently, the role of ECM-receptor interactions in cardiac fibrosis and osteoporosis was reported [10, 11]. Also, the ECM can undergo extensive remodeling during certain pathological conditions [12, 13, 14, 15] such as cancer, where it can drive progression [16]. Consequently, understanding ECM protein-receptor interactions can facilitate our understanding of disease progression and mechanisms [17]. For that reason and because ECM proteins are involved in various signaling pathways, they are also perceived as potential druggable targets. Successful development of drugs against specific ECM proteins holds the promise to facilitate the regulation and treatment of many diseases [5, 18, 19].

Recently a dedicated database, MatrixDB [20], was created providing information about ECM and GAGs (glycosaminoglycans) interactions. Despite the advances of many successful protein-protein interaction (PPI) prediction programs, no specific method has been developed for ECM protein-receptor interactions. In the present work, machine learning (ML) algorithms are utilized using sequence and evolutionary features for predicting ECM protein-receptor interactions. Various sequence and evolutionary features are included along with two different feature vector representations. The best feature vector representation was selected and used along with a feature selection algorithm to obtain enhanced prediction accuracy. The schematic representation of the employed methodology is shown in Figure 1.

In addition, a comparative analysis is performed evaluating the representativeness of negative samples in the training set using three different sampling methods: Random sampling, *k*-Means sampling, and Uniform sampling [21]. Lastly, a comparison is made for the current approach with three available state-of-the-art PPI algorithms.

MATERIALS AND METHODS

Dataset

The dataset is constructed using the KO-pathway K004512 (map04512) obtained from Kyoto Encyclopedia of Genes and Genomes [22]. The interaction map consisted of 13 proteins and 31 receptors with 82 known interactions (positive interactions). Accordingly, there are 403 $(= 13 \times 31)$ possible combinations of such extracellular proteins and receptors. Among those are 82 known interacting pairs that form the positive training dataset (see Table S1 in Supplementary

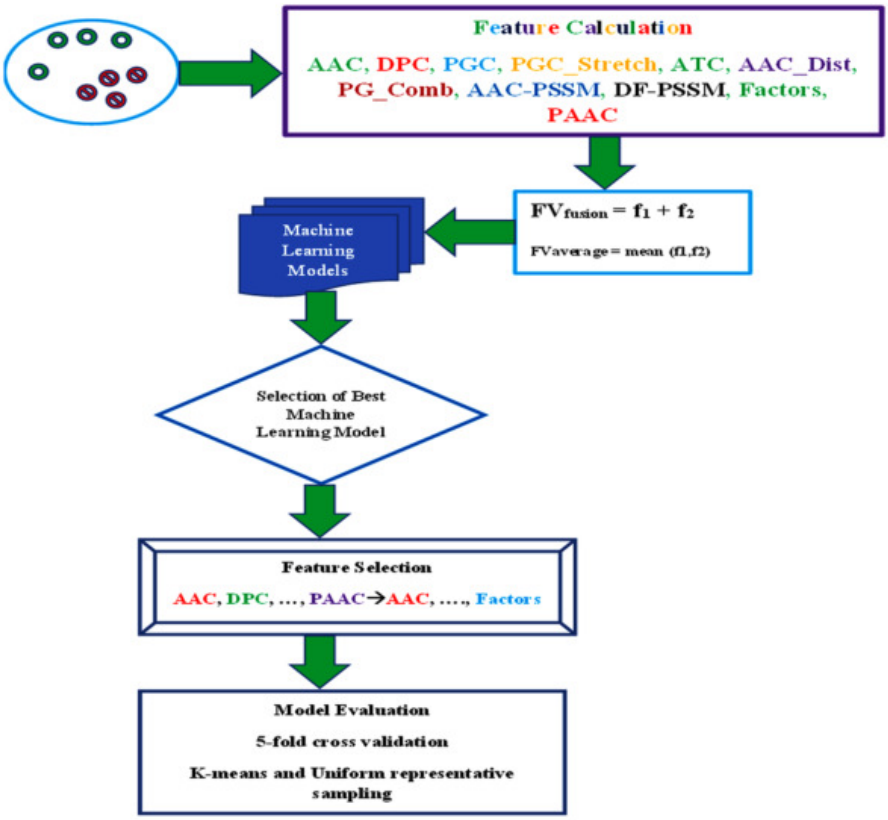

Fig. 1. Schematic representation of the present methodology.

Data). As per the "closed world association" [23], all those interaction pairs that are not present in the positive dataset are potentially negative. After eliminating the 82 interacting pairs, there are 321 pairs, which are not known to interact. Consequently, the negative training data is prepared by randomly selecting 82 pairs from the pool of 321 non-interacting pairs. The protein-protein interaction network for the ECM interaction pathway is shown in Figure 2.

Feature Extraction

A number of sequence-based features are calculated to include the physicochemical properties, sequence residue order, and coupling effects:

- Amino Acid Composition (AAC): The frequencies of twenty amino acid residues constitute the first component of the feature vector, having a dimension of 20.

- Dipeptide Composition (DPC): As simple AAC is deficient in capturing any sequence order effect, dipeptide counts which can capture the coupling of amino acid residues is considered as the second constituent of the feature vector, having a dimension of 400.

- Property Group Composition (PGC): The reduced representation of amino acid residues (see Table 1) have been used previously for a number of protein prediction/classification tasks [24, 25, 26]. This gives rise to a feature vector of 11 dimensions.

- Property Group Stretch (PGC_Stretch): To take into account the conservation of amino acid residues with respect to their physicochemical properties, PGC_Stretch [21] is used as another component of the feature vector. PGC_Stretch takes into account the conservation of stretches of physicochemical properties. A window length of 2 is used and within this

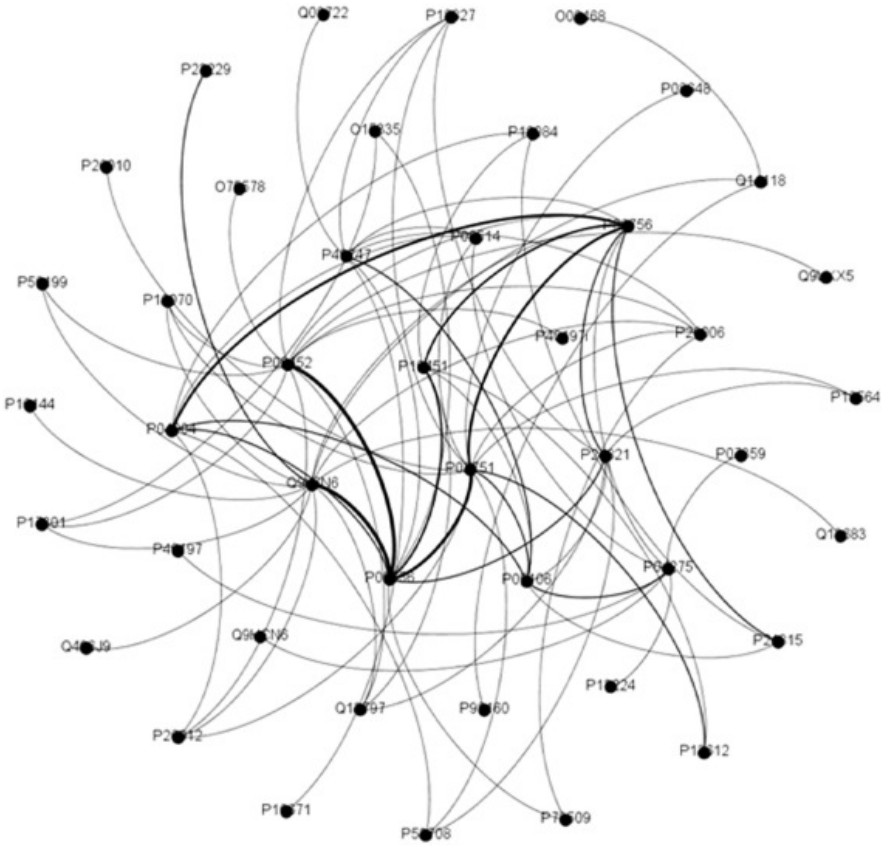

Fig. 2. Protein-protein interaction network for the ECM interaction pathway.

window length only the identical physicochemical groups of the amino acid residues are counted.

- Atomic Features (ATC): The composition of atoms is incorporated as the next feature vector component, consisting of counts of Carbon (C), Hydrogen (H), Oxygen (O), Nitrogen (N) and Sulphur (S), resulting in a 5 dimensional feature vector.

- Amino Acid Distance (AAC_Dist): The distance feature calculates the average distance of a particular amino acid residue with all other amino acid residues present in the sequence, for instance, for the sequence: "QWNFAGIEAAAS", the average distance between Q and W in the above sequence is $QW_{average} = 1$ (as Q and W are in consecutive positions), while the average distance between Q and A in the above sequence is $QA_{average} = 7.75$ (Q $\rightarrow$A = 4, 8, 9, and 10: are the positions of A from Q and $QA_{average}$ is the average value of all the positions). The dimension of this feature is 400.

- Property Group Combination (PG_Comb): This feature is similar to PGC_Stretch as it uses the same 11 different amino acid residue groups for its generation. It incorporates the sequence order information of consecutive physicochemical groups in a window of 2, resulting in a feature vector of 121 dimensions (11 amino acid property groups $\times$ 11 amino acid property groups).

- Amino Acid Factors (Factors): The amino acid factors are included as proposed in [27]. It consists of five factors resulting from multidimensional analysis of 500 amino acid attributes incorporating the amino acid variability. Pseudo Amino Acid Composition (PAAC): The pseudo amino acid composition feature is used to incorporate a higher order

Table 1. Amino acid residues categorised into 11 different overlapping property groups

S.No.	Amino Acid Group	Amino Acids in the Specific Group
1	Tiny group	Ala, Cys, Gly, Ser, Thr
2	Small group	Ala, Cys, Asp, Gly, Asn, Pro, Ser, Thr and Val
3	Aliphatic group	Ile, Leu and Val
4	Non-polar groups	Ala, Cys, Phe, Gly, Ile, Leu, Met, Pro, Val, Trp and Tyr
5	Aromatic group	Phe, His, Trp and Tyr
6	Polar group	Asp, Glu, His, Lys, Asn, Gln. Arg, Ser, and Thr
7	Charged group	Asp, Glu, His, Arg, Lys
8	Basic group	His, Lys and Arg
9	Acidic group	Asp and Glu
10	Hydrophobic group	Ala, Cys, Phe, Ile, Leu, Met, Val, Trp, Tyr
11	Hydrophilic group	Asp, Glu, Lys, Asn, Gln,Arg

sequence correlation that exists in each protein sequence. It is calculated with the help of iFeature server [28] giving rise to feature vector of 50 dimensions.

- Evolutionary Features: Apart from sequence features, to incorporate the evolutionary information into the prediction model, two types of features are calculated: AAC_PSSM and DF_PSSM using POSSUM server [29]. The database selected for POSSUM is Uniref50 with 3 iterations and an E-value threshold of 0.001 for BLAST search [30].

Feature Representation

Representation of protein sequences using some mathematical formulation and its presentation to a particular machine learning algorithm is an important factor in the development of an accurate predictor. Different feature vector representations present the different aspects of the input data space. Here, two types of representation are used for feeding the interacting/non-interacting pairs into the machine learning algorithms:

- Fusion of Feature Vectors ($\mathbf{FV_{fusion}}$): The two feature vectors of the protein and its receptor, each of length 1063, are concatenated to form the final feature vector of dimension 2126.

$$\mathbf{FV_{fusion}} = (\mathbf{f_1},\mathbf{f_2})$$

- Average of feature vectors ($\mathbf{FV_{average}}$): For each protein-receptor pair the mean of every feature is calculated, resulting in a feature vector of length 1063.

$$\mathbf{FV_{average}} = \text{mean}(\mathbf{f_1},\mathbf{f_2})$$

Classification Algorithms

Generally, an ensemble of classifiers performs better in a classification task than individual classifiers [31]. Boosting [32, 33] and Bagging [34] are two prominent ensemble learning methods. Boosting employs a sequential learning strategy where during each iteration, the base learners are refined to correctly classify the hard-to-classify examples. Bagging consists of bootstrap sampling, where each base learner is provided with a different set of training samples. The final step consists of fusing of the classification decisions from all of the base learners. The random forest [35] algorithm also involves bagging with random selection of features at

each node of the base classifiers (decision trees). The decision trees are grown without pruning and the final step consists of combining the decisions from all of the base classifiers. A boosted version of random forest is implemented for the classification of the ECM interacting proteins, with Real Adaboost as the meta classifier and random forest as the base classifier [36].

Further a comparison is made for the performance of RARF with six other machine learning algorithms: Naïve Bayes (NB) [51], Random Forest (RF) [52], Bagging [53], *k*-Nearest Neighbors (*k*-NN) [54], Support Vector Machines [55] implemented with sequential minimization optimization and Radial Basis Function Kernel (SMO-RBF) [56, 57] and Rotation Forest (ROF) [58].

Representative Sampling Methods

Previous works [26, 37, 38] have emphasized the importance of having representative samples in the training dataset. Ideally, the training data set should cover the entire input space, so that the classification algorithm can learn all the different types of patterns. Three sampling methods are implemented: *k*-Means sampling, Uniform sampling, and Random sampling methods for inclusion of representative negative training data.

k-Means Clustering

The purpose of the *k*-Means algorithm [39, 40] is to find natural groupings among the samples which can be further used for representative sampling (i.e. selection of representative samples). *k*-Means clustering is used to cluster the non-interacting proteins and select representative samples for the construction of the training set. The value for *k* (a parameter determining the number of clusters/groups) is selected equivalent to the number of positive samples (i.e. the number of interacting proteins). From each cluster, one sample is selected for the training set. The clustering algorithm is used to create a representative training dataset covering the full input space.

Uniform Sampling

The Uniform sampling implemented in the present work is based on the Kennard-Stone algorithm [41]. The aim of Uniform sampling is the selection of representative samples uniformly covering the input space. First, the mean of the data is calculated and the sample, which has the minimum distance (Euclidean distance) to the data mean, is drawn into the representative set. Next, all other samples are selected in an iterative manner, selecting those into the representative sample set that are farthest away from the samples already selected.

Feature Selection

Feature selection reduces the original full feature set, retaining only highly informative and non-redundant features. Most of the times implementation of feature selection algorithms results in enhanced prediction performance. The ReliefF [42, 43] feature selection algorithm is used in a stepwise manner (increasing the number of features in steps of 100) for obtaining a more accurate model. All the machine learning algorithms and the feature selection method are simulated using the Java-based machine learning platform-WEKA [44].

MODEL PERFORMANCE VALIDATION

Five-fold cross validation is employed for evaluating the trained prediction models. In five-fold cross validation, a single fold is reserved as a testing set while the remaining four are used as the training set. The process continues until all the folds are used once as a testing set.

Model Performance Evaluation Metrics

- Sensitivity: It is the percentage of correctly predicted ECM interacting proteins.

$$Sensitivity = \frac{TP}{TP + FN} \times 100 \tag{1}$$

- Specificity: It is the percentage of correctly predicted ECM non-interacting proteins.

$$Specificity = \frac{TN}{TN + FP} \times 100 \tag{2}$$

- Accuracy: It is the percentage of correctly predicted ECM interacting proteins and non-interacting proteins.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \times 100 \tag{3}$$

- Area under ROC (*AUC*): *AUC* [45, 46] is used to describe ROC (receiver operating characteristic) curves and it is a threshold dependent metric. It can take values from 0 to 1. The closer its value to 1, the better the prediction model.

- Mathews Correlation Coefficient (*MCC*): It is a performance evaluation metric for two class classification scenarios, where its value ranges from -1 to $+1$. Obtaining *MCC* values closer to 1 is deemed to be better for trained prediction models.

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \tag{4}$$

RESULTS AND DISCUSSION

Table 2. Performance evaluation metrics of different machine learning algorithms using fusion of feature vectors

All Features (Fusion of feature vectors)					
	Sensitivity	**Specificity**	**Accuracy**	***MCC***	***AUC***
NB	82.9	43.9	63.4	0.291	0.660
RF	59.8	69.5	64.6	0.294	0.718
Bagging	63.4	65.9	64.6	0.293	0.669
IBK	72.0	54.9	63.4	0.272	0.626
SMO-RBF	43.9	58.5	51.2	0.025	0.512
ROF	58.5	69.5	64.0	0.282	0.683
RARF	61.0	73.2	**67.1**	0.344	0.730
Avg. of all classifiers	63.0	62.2	62.6	0.257	0.656

Table 2 presents the performances of various machine learning algorithms using the fusion of feature vector strategy ($\mathbf{FV_{fusion}}$). In terms of accuracy, specificity, *MCC* and *AUC*, boosted random forest (RARF) performed much better than all other learning algorithms.

In Table 3, the performances of the various machine learning algorithms using the average of feature vector strategy ($\mathbf{FV_{average}}$) is presented. The highest accuracy of 70.7 % with 0.767

Table 3. Performance evaluation metrics of different machine learning algorithms using average of feature vectors

All Features (Average of feature vectors)					
	Sensitivity	**Specificity**	**Accuracy**	***MCC***	***AUC***
NB	82.9	28.0	55.5	0.131	0.614
RF	69.5	65.9	67.7	0.354	0.739
Bagging	73.2	61.0	67.1	0.344	0.680
IBK	78.0	56.1	67.1	0.342	0.741
SMO-RBF	56.1	76.8	66.5	0.337	0.665
ROF	68.3	65.9	67.1	0.342	0.739
RARF	69.5	72.0	**70.7**	0.415	0.767
Avg. of all classifiers	71.0	60.8	67.1	0.344	0.730

AUC is obtained by RARF as the classifier. In both feature vector representations ($\mathbf{FV_{fusion}}$ and $\mathbf{FV_{average}}$), RARF performed superior in comparison to all other algorithms.

As per the "No free lunch theorem", no classification algorithm is the most suitable for all the datasets [50] . As the representation of the training data varies, so do the performances of the classification algorithms. Higher accuracies are achieved by some classification algorithms using fusion representation and AAC, PGC, PGC_Stretch, AC, AAC_PSSM, DF_PSSM, Factors and PAAC feature sets, while $\mathbf{FV_{average}}$ performed better with AAC_Dist and PG_Comb feature sets (see Tables S2 and S3 in Supplementary File). However, when the full feature set is used, better performance metrics are obtained for all the classification algorithms using $\mathbf{FV_{average}}$. Also, $\mathbf{FV_{average}}$ resulted in better accuracy than $\mathbf{FV_{fusion}}$. In terms of accuracy, the RARF classifier obtained the highest accuracies in both feature representation schemes (67.1 % and 70.7 % using $\mathbf{FV_{fusion}}$ and $\mathbf{FV_{average}}$ respectively). On the full feature set, all classification algorithms except NB performed better using $\mathbf{FV_{average}}$. Of note, SMO-RBF achieved higher specificity than RARF. A gain of 8 %, 3.3 %, 6.6 %, and 5 % are observed for mean sensitivity, mean accuracy, mean *MCC*, and mean *AUC*, respectively, when using $\mathbf{FV_{average}}$.

Table 4. Performance evaluation metrics of RARF on ReliefF based feature selection using average of feature vectors

RARF+ReliefF					
No. of features	**Sensitivity**	**Specificity**	**Accuracy**	***MCC***	***AUC***
10	68.3	61.0	64.6	0.293	0.707
100	72.0	68.3	70.1	0.403	0.729
200	**74.4**	**70.7**	**72.6**	**0.452**	**0.752**
300	73.2	69.5	71.3	0.427	0.762
400	74.4	68.3	71.3	0.428	0.764
500	73.2	69.5	71.3	0.427	0.764
600	70.7	68.3	69.5	0.390	0.768
700	72.0	70.7	71.3	0.427	0.772
800	70.7	67.1	68.9	0.378	0.772
900	70.7	70.7	70.7	0.415	0.768
1000	72.0	67.1	69.5	0.391	0.772

With the full feature set, RARF obtained the highest accuracy using $\mathbf{FV_{average}}$. Therefore, RARF is also used to identify the most relevant features using the ReliefF algorithm. The performances of RARF with incremental feature sets are presented in Table 4. When increasing the number of features from 10 to 200, an increase in all performance evaluation metrics is observed. Beyond 200 features, a decreasing trend in performance evaluation metrics is observed. The best performance evaluation metrics is obtained with the top 200 features achieving 72.6 % accuracy, 74.4 % sensitivity and 70.7 % specificity.

To evaluate the representativeness of negative samples in the training set three different sampling methods are implemented: Random Sampling, *k*-Means Sampling and Uniform Sampling. Table 5 presents the performances of RARF on 5 runs of 5-fold cross validation for Random Sampling and *k*-Means Sampling. The top 200 features are used as the feature set. As uniform sampling does not involve any random seed generation step, it is performed for a single run.

Using *k*-Means Sampling, RARF achieved 75.5 % accuracy as compared to 72.9 % on random sampling and 73.2 % on uniform sampling, using 5 runs of 5-fold cross validation. Except for specificity, *k*-Means based sampling resulted in an enhanced performance for RARF as compared to random and uniform sampling methods. In terms of sensitivity, accuracy, *MCC*, and *AUC*, RARF achieved higher values when trained on representative training sets obtained using *k*-Means and uniform sampling as compared to random sampling. However, when considering only specificity, Random Sampling performed better than *k*-Means and Uniform Sampling.

Table 5. Performance evaluation metrics for RARF on 5 × 5 fold cross validation for Random sampling and *k*-Means sampling and single run of 5 fold cross validation for Uniform sampling

RARF+ReliefF					
Random Sampling					
Run	**Sensitivity**	**Specificity**	**Accuracy**	***MCC***	***AUC***
1	74.4	70.7	72.6	0.452	0.752
2	82.9	74.4	78.7	0.575	0.834
3	76.8	70.7	73.8	0.476	0.780
4	70.7	68.3	69.5	0.390	0.742
5	72.0	68.3	70.1	0.403	0.755
Average	**75.3**	**70.4**	**72.9**	**0.459**	**0.772**
***k*-Means Sampling**					
Run	**Sensitivity**	**Specificity**	**Accuracy**	***MCC***	***AUC***
1	82.9	65.9	74.4	0.495	0.803
2	82.9	65.9	74.4	0.495	0.803
3	76.8	67.1	72.0	0.411	0.773
4	86.6	70.7	78.7	0.581	0.813
5	85.4	70.7	78.0	0.567	0.813
Average	**80.8**	**68.1**	**75.5**	**0.598**	**0.801**
Uniform Sampling					
	Sensitivity	**Specificity**	**Accuracy**	***MCC***	***AUC***
	79.3	**67.1**	**73.2**	**0.467**	**0.785**

The performance evaluation metrics of all the machine learning algorithms tested here with individual feature sets are summarized in Tables S2 ($\mathbf{FV_{fusion}}$) and S3 ($\mathbf{FV_{average}}$). The ROCs of

the ML algorithms on the 200 best features is presented in Figure 3.

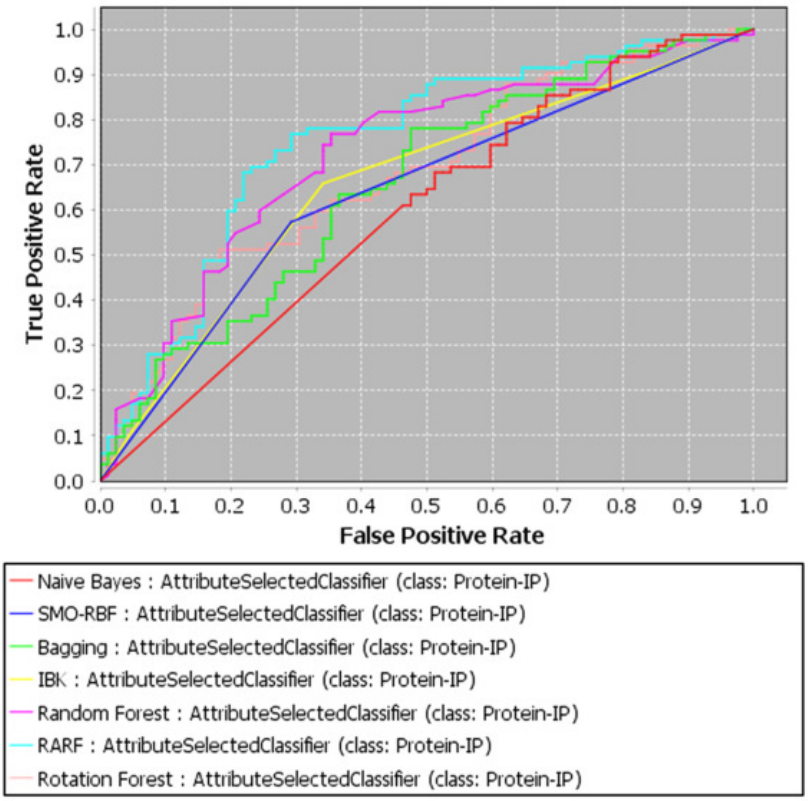

Fig. 3. ROCs of different machine learning algorithms on best feature subset (200 features) using average of feature vectors.

Further three PPI predictors are used to test their accuracy for predicting the ECM PPIs: PSOPIA* [47], TRI_tool† [48] and iLoops [49]

The PSOPIA server was trained on 43,060 high confidence direct (physical) PPIs and 33,098,951 negative PPIs. Both the positive and negative interactions are used for testing the server. Of note, this server is unable to process sequences of more than 3000 residues, but the positive dataset consisted of two proteins which are over 3000 residues in length. Consequently, three positive PPI interactions were not processed (see supplementary material 2).

The following confusion matrix is obtained from which sensitivity, specificity and accuracy values are calculated:

	T	**F**
T	9 (TP)	70 (TP)
F	4 (TP)	78 (TP)

Therefore, sensitivity = 11 %, specificity = 95 %, and accuracy = 54 %.

The TRI_tool gave the following confusion matrix:

	T	**F**
T	9 (TP)	73 (TP)
F	3 (TP)	79 (TP)

*mizuguchilab.org/PSOPIA

†www.vin.bg.ac.rs/180/tools/tfpred.php

Thus, sensitivity = 10.9 %, specificity = 96.3 % and accuracy = 53.65 % (see supplementary material 3).

From the above results, it can concluded that both of these PPI predictors are better at predicting non-interactions than ECM interactions. Overall, the accuracy is skewed towards non-interacting protein pairs.

Further, another PPI prediction method iLoops server is also evaluated using the 82 positive and negative PPIs. Out of those positive interactions, the server returned prediction results for 54 PPIs. For the remaining 28 PPIs the server returned "No interaction signatures/ no protein features". Out of those 54 PPIs, 38 were predicted correctly (true positives). Further, out of 82 non interacting PPIs (true negatives), the server returned results for 34 interactions. For the remaining 48 PPIs, the server returned "No interaction signatures/ no protein features". For the 82 true negatives, only 8 were correctly predicted.

The resulting confusion matrix is shown below:

	T	**F**
T	38 (TP)	16 (TP)
F	26 (TP)	8 (TP)

Thus, sensitivity = 70.3 %, specificity = 23.5 % and accuracy = 52.2 %.

Although the sensitivity of iLoops (70.3 %) is better than the sensitivity of PSOPIA (11 %) and TRI_tool (10.9 %), significant numbers of interacting and non-interacting PPIs were not classified, effectively resulting in accuracies of 46.34 % and 9.7 % for the interacting and non-interacting classes, respectively.

In comparison, the current specialized ECM predictor provides an acceptable sensitivity with balanced accuracy. ECM-receptor interaction are only a very small fraction of all PPIs and hence it is well possible that in larger PPI datasets, the small subgroup of ECM-receptor interactions does not get proper consideration, i.e. their representation in the training sets gets diluted, such that ML algorithms don't have an opportunity to learn this specialized group of PPIs.

One of the limitations of any protein-protein interaction prediction problem is the selection of true negatives, as the experimental non-interaction data is not readily available. Further the dimension of feature vector can be reduced using representative learning techniques such as autoencoders which may facilitate in obtaining better performance evaluation metrics with reduced training time.

CONCLUSION

Understanding the ECM-receptor interaction can enable and speed up our understanding of cancer progression as well as ECM-mediated mechanisms of other diseases. Sample representation is an important parameter in classification tasks. The way we feed the sequences using some mathematical formulation presents a different aspect of the prediction tool and plays an important role in acquiring higher predictive accuracy. In the present work, two different types of feature representation are compared: fusion of feature vectors ($\mathbf{FV_{fusion}}$) and average of feature vectors ($\mathbf{FV_{average}}$) for predicting interacting versus non-interacting protein pairs. $\mathbf{FV_{average}}$ performed comparatively better than $\mathbf{FV_{fusion}}$. Boosted random forest performed best among all classifiers. For both feature vector representations ($\mathbf{FV_{fusion}}$ and $\mathbf{FV_{average}}$), the RARF algorithm performed better than the base classifier RF, more so in terms of specificity and accuracy. The step wise increase of the number of features using the ReliefF feature selection algorithm resulted in improved prediction accuracy. A comparative analysis evaluating the effectiveness of representativeness of negative training samples when using random sampling, *k*-Means sampling, and Uniform sampling showed that *k*-Means sampling and the subsequent generation

of a diversified training set is advantageous in comparison to random sampling and Uniform sampling as it resulted in enhanced prediction accuracy for ECM-receptor interaction prediction. On comparison with three state of the art PPI predictors, it is observed that PPI predictors lack accuracy in identifying ECM-receptor interaction (sensitivity) but show relatively high accuracy in identifying non-interactions (specificity).The current work presents the first machine learning based prediction model specifically developed for ECM protein-receptor interactions.

REFERENCES

1. Gullberg D., Heldin P., Liliana S., Ruggero T., Achilleas T., Jan-Olof W. *Extracellular matrix: pathobiology and signaling.* Walter de Gruyter, 2012.
2. Manou D., Caon I., Bouris P., Triantaphyllidou I.-E., Giaroni C., Passi A., Karamanos N.K., Vigetti D., Theocharis A.D. *The Complex Interplay Between Extracellular Matrix and Cells in Tissues*. 2019. P. 1–20.
3. Jinka R., Kapoor R., Sistla P.G., Raj T.A., Pande G. Alterations in Cell-Extracellular Matrix Interactions during Progression of Cancers. *International Journal of Cell Biology.* 2012. V. 2012. P. 1–8.
4. Bosman F.T., Stamenkovic I. Functional structure and composition of the extracellular matrix. *The Journal of Pathology.* 2003. V. 200. № 4. P. 423–428.
5. Kim S.-H., Turnbull J., Guimond S. Extracellular matrix and cell signalling: the dynamic cooperation of integrin, proteoglycan and growth factor receptor. *Journal of Endocrinology.* 2011. V. 209. № 2. P. 139–151.
6. van der Flier A., Sonnenberg A. Function and interactions of integrins. *Cell and Tissue Research.* 2001. V. 305. № 3. P. 285–298.
7. David G., Lories V., Decock B., Marynen P., Cassiman J.J., Van den Berghe H. Molecular cloning of a phosphatidylinositol-anchored membrane heparan sulfate proteoglycan from human lung fibroblasts. *Journal of Cell Biology.* 1990. V. 111. № 6. P. 3165–3176.
8. Stipp C.S., Litwack E.D., Lander A.D. Cerebroglycan: an integral membrane heparan sulfate proteoglycan that is unique to the developing nervous system and expressed specifically during neuronal differentiation. *Journal of Cell Biology.* 1994. V. 124. № 1. P. 149–160.
9. Elenius K., Jalkanen M. Function of the syndecans - a family of cell surface proteoglycans. *Journal of Cell Science.* 1994. V. 107. № 11. P. 2975–2982.
10. Shi Yan, Yunpeng Zhang, Dai-Feng Lu, Feng Dong, Yongyun Lian. *ECM-receptor interaction as a prognostic indicator for clinical outcome of primary osteoporosis*. 2016.
11. Buttner P., Ueberham L., Shoemaker M.B., Roden D.M., Dinov B., Hindricks G., Bollmann A., Husser D. Identification of Central Regulators of Calcium Signaling and ECM–Receptor Interaction Genetically Associated With the Progression and Recurrence of Atrial Fibrillation. *Frontiers in Genetics.* 2018. V. 9.
12. Karamanos N.K. Extracellular matrix: key structural and functional meshwork in health and disease. *The FEBS Journal.* 2019. V. 286. № 15. P. 2826–2829.
13. Mavrogonatou E., Pratsinis H., Papadopoulou A., Karamanos N.K., Kletsas D. Extracellular matrix alterations in senescent cells and their significance in tissue homeostasis. *Matrix Biology.* 2019. V. 75–76. P. 27–42.
14. Theocharis A.D., Manou D., Karamanos N.K. The extracellular matrix as a multitasking player in disease. *The FEBS Journal.* 2019. V. 286. № 15. P. 2830–2869.
15. Urbanczyk M., Layland S.L., Schenke-Layland K. The role of extracellular matrix in biomechanics and its impact on bioengineering of cells and 3D tissues. *Matrix Biology.* 2020. V. 85–86. P. 1–14.
16. Pupa S.M., Menard S., Forti S., Tagliabue E. New insights into the role of extracellular matrix during tumor onset and progression. *Journal of Cellular Physiology.* 2002. V. 192.

№ 3. P. 259–267.
17. Jung J., Ryu T., Hwang Y., Lee E., Lee D. Prediction of Extracellular Matrix Proteins Based on Distinctive Sequence and Domain Characteristics. *Journal of Computational Biology.* 2010. V. 17. № 1. P. 97–105.
18. Hanna E., Quick J., Libutti S.K. The tumour microenvironment: a novel target for cancer therapy. *Oral Diseases.* 2009. V. 15. № 1. P. 8–17.
19. Desgrosellier J.S., Cheresh D.A. Integrins in cancer: biological implications and therapeutic opportunities. *Nature Reviews Cancer.* 2010. V. 10. № 1. P. 9–22.
20. Launay G., Salza R., Multedo D., Thierry-Mieg N., Ricard-Blum S. MatrixDB, the extracellular matrix interaction database: updated content, a new navigator and expanded functionalities. *Nucleic Acids Research.* 2015. V. 43. № D1. P. D321–D327.
21. Nath A., Leier A. Improved cytokine–receptor interaction prediction by exploiting the negative sample space. *BMC Bioinformatics.* 2020. V. 21. № 1. P. 493.
22. Kanehisa M. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research.* 2000. V. 28. № 1. P. 27–30.
23. Roy S., Martinez D., Platero H., Lane T., Werner-Washburne M. Exploiting Amino Acid Composition for Predicting Protein-Protein Interactions. *PLoS ONE.* 2009. V. 4. № 11. Article No. e7813.
24. Nath A., Chaube R., Subbiah K. An insight into the molecular basis for convergent evolution in fish antifreeze Proteins. *Computers in Biology and Medicine.* 2013. V. 43. № 7. P. 817–821.
25. Nath A. Insights into the sequence parameters for halophilic adaptation. *Amino Acids.* 2016. V. 48. № 3. P. 751–762.
26. Nath A., Subbiah K. The role of pertinently diversified and balanced training as well as testing data sets in achieving the true performance of classifiers in predicting the antifreeze proteins. *Neurocomputing*. 2018. V. 272. P. 294–305.
27. Atchley W.R., Zhao J., Fernandes A.D., Drüke T. Solving the protein sequence metric problem. *Proceedings of the National Academy of Sciences.* 2005. V. 102. № 18. P. 6395–6400.
28. Chen Z., Zhao P., Li F., Leier A., Marquez-Lago T.T., Wang Y., Webb G.I., Smith A.I., Daly R.J., Chou K.-C., Song J. iFeature: a Python package and web server for features extraction and selection from protein and peptide sequences. *Bioinformatics.* 2018. V. 34. № 14. P. 2499–2502.
29. Wang J., Yang B., Revote J., Leier A., Marquez-Lago T.T., Webb G., Song J., Chou K.-C., Lithgow T. POSSUM: a bioinformatics toolkit for generating numerical sequence feature descriptors based on PSSM profiles. *Bioinformatics.* 2017. V. 33. № 17. P. 2756–2758.
30. Altschul S.F., Gish W., Miller W., Myers E.W., Lipman D.J. Basic local alignment search tool. *Journal of Molecular Biology.* 1990. V. 215. № 3. P. 403–410.
31. Polikar R. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine.* 2006. V. 6. № 3. P. 21–45.
32. Freund Y., Schapire R.E. Experiments with a New Boosting Algorithm. In: *In proceedings of the thirteenth International Conference on Machine Learning.* Morgan Kaufmann, 1996. P. 148–156.
33. Schapire R.E. *The Boosting Approach to Machine Learning: An Overview*. 2003. P. 149–171.
34. Breiman L. Bagging predictors. *Machine Learning.* 1996. V. 24. № 2. P. 123–140.
35. Breiman L. Random Forests. *Machine Learning.* 2001. V. 45. № 1. P. 5–32.
36. Nath A., Subbiah K. Maximizing lipocalin prediction through balanced and diversified training set and decision fusion. *Computational Biology and Chemistry.* 2015. V. 59.

P. 101–110.
37. de Groot P.J., Postma G.J., Melssen W.J., Buydens L.M.C. Selecting a representative training set for the classification of demolition waste using remote NIR sensing. *Analytica Chimica Acta.* 1999. V. 392. № 1. P. 67–75.
38. Li D.-C., Hu S.C., Lin L.-S., Yeh C.-W. Detecting representative data and generating synthetic samples to improve learning accuracy with imbalanced data sets. *PLoS ONE.* 2017. V. 12. № 8. Article No. e0181853.
39. Jain A.K., Murty M.N., Flynn P.J. Data clustering. *ACM Computing Surveys.* 1999. V. 31. № 3. P. 264–323.
40. Larose D.T., Larose C.D. *Discovering Knowledge in Data.* Hoboken, NJ, USA: John Wiley and Sons, Inc., 2014.
41. Daszykowski M., Walczak B., Massart D.L. Representative subset selection. *Analytica Chimica Acta.* 2002. V. 468. № 1. P. 91–103.
42. Kira K., Rendell L.A. A Practical Approach to Feature Selection. In: *Proceedings of the Ninth International Workshop on Machine Learning.* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1992. P. 249–256.
43. Urbanowicz R.J., Meeker M., La Cava W., Olson R.S., Moore J.H. Relief-based feature selection: Introduction and review. *Journal of Biomedical Informatics.* 2018. V. 85. P. 189–203.
44. Hall M., Frank E., Holmes G., Pfahringer B., Reutemann P., Witten I.H. The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter.* 2009. V. 11. № 1. P. 10–18.
45. Ling C.X., Huang J., Zhang H. AUC: A Better Measure than Accuracy in Comparing Learning Algorithms. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* 2003. V. 2671. P. 329–341.
46. Jin H., Ling C.X. Using AUC and accuracy in evaluating learning algorithms. *IEEE Transactions on Knowledge and Data Engineering.* 2005. V. 17. № 3. P. 299–310.
47. Murakami Y., Mizuguchi K. PSOPIA: Toward more reliable protein-protein interaction prediction from sequence information. In: *2017 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS).* IEEE, 2017. P. 255–261.
48. Perovic V., Sumonja N., Gemovic B., Toska E., Roberts S.G., Veljkovic N. TRI_tool: a web-tool for prediction of protein–protein interactions in human transcriptional regulation. *Bioinformatics.* 2017. V. 33. № 2. P. 289–291.
49. Planas-Iglesias J., Marin-Lopez M.A., Bonet J., Garcia-Garcia J., Oliva B. iLoops: a protein–protein interaction prediction server based on structural features. *Bioinformatics.* 2013. V. 29. № 18. P. 2360–2362.
50. Wolpert, D. & Macready W. No free lunch theorems for optimization. *IEEE Transactions On Evolutionary Computation*. 1997. V. 1. P. 67–82.
51. Murphy K. Naive bayes classifiers. *University Of British Columbia*. 2006. V. 18. P. 1–8.
52. Breiman L. Random forests. *Machine Learning*. 2001. V. 45. P. 5–32.
53. Breiman L. Bagging predictors. *Machine Learning*. 1996. V. 24. P. 123–140.
54. Peterson L. K-nearest neighbor. *Scholarpedia*. 2009. V. 4. Article No. 1883.
55. Cortes C., Vapnik V. Support-vector networks. *Machine Learning*. 1995. V. 20. P. 273–297.
56. Platt J. Sequential minimal optimization: A fast algorithm for training support vector machines. Microsoft Research, 1998. Technical Report No. msr-tr-98-14.
57. Keerthi S., Shevade S., Bhattacharyya C., Murthy K. Improvements to Platt's SMO algorithm for SVM classifier design. *Neural Computation*. 2001. V. 13. P. 637–649.
58. Rodriguez J., Kuncheva L., Alonso C. Rotation forest: A new classifier ensemble method.

IEEE Transactions On Pattern Analysis And Machine Intelligence. 2006. V. 28. P. 1619–1630.

Accepted 19.01.2023.
Revised 24.02.2023.
Published 24.04.2023.