iPHLoc-ES: Identification of bacteriophage protein locations using evolutionary and structural features

Dear author,

Please note that changes made in the online proofing system will be added to the article before publication but are not reflected in this PDF.

We also ask that this file not be used for submitting corrections.

Journal of Theoretical Biology xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Journal of Theoretical Biology

journal homepage: www.elsevier.com/locate/jtbi

iPHLoc-ES: Identification of bacteriophage protein locations using evolutionary and structural features

Q2 Swakkhar Shatabda[a,*], Sanjay Saha[a], Alok Sharma[b,c], Abdollah Dehzangi[d]

[a] Department of Computer Science and Engineering, United International University, House 80, Road 8A, Dhanmondi, Dhaka-1209, Bangladesh
[b] Institute for Integrated and Intelligent Systems, Griffith University, Australia
[c] School of Engineering and Physics, University of the South Pacific, Fiji
[d] Department of Computer Science, School of Computer, Mathematical, and Natural Sciences, Morgan State University, United States

ARTICLE INFO

Article history:
Received 20 July 2017
Revised 18 September 2017
Accepted 20 September 2017
Available online xxx

MSC:
00-01
99-00

Keywords:
Proteins
Locations
Phage
Classification
Feature selection

ABSTRACT

Bacteriophage proteins are viruses that can significantly impact on the functioning of bacteria and can be used in phage based therapy. The functioning of Bacteriophage in the host bacteria depends on its location in those host cells. It is very important to know the subcellular location of the phage proteins in a host cell in order to understand their working mechanism. In this paper, we propose iPHLoc-ES, a prediction method for subcellular localization of bacteriophage proteins. We aim to solve two problems: discriminating between host located and non-host located phage proteins and discriminating between the locations of host located protein in a host cell (membrane or cytoplasm). To do this, we extract sets of evolutionary and structural features of phage protein and employ Support Vector Machine (SVM) as our classifier. We also use recursive feature elimination (RFE) to reduce the number of features for effective prediction. On standard dataset using standard evaluation criteria, our method significantly outperforms the state-of-the-art predictor. iPHLoc-ES is readily available to use as a standalone tool from: https://github.com/swakkhar/iPHLoc-ES/ and as a web application from: http://brl.uiu.ac.bd/iPHLoc-ES/.

1. Introduction

The term 'bacteriophage' means 'bacteria eaters' in Latin. Bacteriophage or informally called phage proteins are viruses that can kill the bacteria by infection and replication. History of phage goes back 100 years back in 1910s when phages were used to cure dysentery (Keen, 2012; Lederberg, 1996). With the emergence of antibiotics, phage therapy somehow lost its popularity (Keen, 2012). However, in recent years due to continuous abuse of anti-bacterial drug by inappropriate prescription practices and poor drug access control (Liljeqvist et al., 2012) and evolving capability of the microbes, the commercial viability of new antibiotics is in decline (Hughes, 2011). The overuse of antibiotics have also been detrimental to the communities of beneficial bacteria (Buffie et al., 2012). In contrast, the phages are very precise in nature and the scientists are again looking back to these bacteriophages to treat the intractable bacterial infections (Deresinski, 2009; Sorokulova et al., 2014).

An injected bacteriophage transcribed by host cell polymerase typically has two life cycles: lytic and lysogenic. In lysogenic or temperate phase, the phage continues replication along with the host cell. However, lysis instigated typically by enzymes breaks open the host cell membrane and destroys it (Sass and Bierbaum, 2007). Phage proteins are either extra-cellular or not located in host cells or located in host cells. Extra cellular phages often take help of receptor for adsorption whose location are pivotal among other factors (Rakhuba et al., 2010). Subcellular localization of phage proteins are mostly distributed in host membrane or in host cytoplasm. Knowledge of the location of bacteriophage proteins are fundamental to the understanding of the mechanism of the virion and development of anti-bacterial therapy. Electron microscopy is generally used to find the locations of phage proteins in host cell (Altman et al., 1985; Casjens and Hendrix, 1988). However, the experimental methods are still time consuming and expensive.

Many computational methods have been developed to study and analyze phage proteins (Cheng et al., 2017a; 2017c; Chou and Shen, 2006; Ding et al., 2014; 2016a; 2016b; Khan et al., 2017; Seguritan et al., 2012; Shen and Chou, 2007a; 2007b; 2009; 2010a; 2010b; Wu et al., 2012; Xiao et al., 2011a; 2011b; Zhou et al., 2011). PHAST was introduced in Zhou et al. (2011) to identify and

* Corresponding author.
E-mail addresses: swakkhar@cse.uiu.ac.bd (S. Shatabda), sanjay@cse.uiu.ac.bd (S. Saha), alok.sharma@griffith.edu.au (A. Sharma), abdollah.dehzangi@morgan.edu (A. Dehzangi).

https://doi.org/10.1016/j.jtbi.2017.09.022
0022-5193/

annotate prophage sequences within bacterial genomes. Among other phage finding tools are PHASTER (Arndt et al., 2016), Phage_finder (Fouts, 2006). Another successful phage prediction tool was PhiSpy (Akhter et al., 2012) that used similarity and composition based strategies.

Several classification algorithms are used to predict phage or phage locations including Artificial Neural Network (ANN) (Galiez et al., 2015; Seguritan et al., 2012), Support Vector Machine (SVM) (Ding et al., 2016b), Random Forest (RF) (McNair et al., 2012) and Naive Bayesian Classifier (NBC) (Feng et al., 2013). Subcellular localization of proteins (Emanuelsson et al., 2000) and bacteriophages (Chou and Shen, 2007; Ding et al., 2014) are of interest for a long time in the research field. In a very recent work, a prediction methodology was proposed to identify phage locations in protein in Ding et al. (2016a) using feature selection method. They have used Support Vector Machine (SVM) classifier to solve two subcellular localization problems on a verified benchmark dataset.

In this paper we tackle two types of localization problems. The first problem we denote as *PH vs non-PH* discrimination problem, where the aim is to classify whether a given phage protein is a host located phage (PH) or a extra-cellular phage (non-PH). The second problem is denoted by *PHM vs PHC* classification where the aim is to classify between two types of host located phages, whether they are located in cell membrane (PHM) or in cell cytoplasm (PHC). We propose iPHLoc-ES for prediction of subcellular locations of phage proteins. iPHLoc-ES is also able to discriminate between host located phages and extra-cellular phages. Our predictor is based on extracting a set of evolutionary and structural features and using a Support Vector Machine (SVM) classifier along with recursive feature elimination (RFE) as feature selection technique. On the standard benchmark dataset of phage proteins our method significantly outperforms the state-of-the-art predictor. We have also made iPHLoc-ES available as a stand-alone tool that is freely available to use (https://github.com/swakkhar/iPHLoc-ES/). We have also made it available as a web application from: http://brl.uiu.ac.bd/iPHLoc-ES/.

In this paper, we follow the guidelines in compliance with Chou's 5-step rule (Chou, 2011) to establish a useful statistical predictor for a biological system. The rest of the paper is organized accordingly: (a) description of the benchmark dataset and construction of train and test sets for the predictor; (b) mathematical formulation of the biological sequence samples that can reflect their intrinsic correlation with the target to be predicted; (c) a powerful model for feature selection and classification algorithm; (d) proper experimentation with cross-validation tests; (e) a user-friendly web-server for the predictor that is accessible to the public.

2. Materials and methods

In this section, we describe the materials and methods required to develop iPHLoc-ES. We call our system **i**dentification of bacterio**PH**age protein **Loc**ations using **E**volutionary and **S**tructural Features (iPHLoc-ES). A system flow-chart of our prediction model is given in Fig. 1.

Phage protein sequences from the benchmark dataset are first fed to PSI-BLAST (Altschul et al., 1997) and SPIDER2 (Heffernan et al., 2015; Yang et al., 2017). PSI-BLAST produces a position specific scoring matrix (PSSM) file and SPIDER2 predicts structural information and generates a SPD file that is used by the feature generation module to generate a set of features. Features are generated belonging to three different groups: composition based evolutionary features, PSSM based evolutionary features and SPD based structural features. After the feature generation a feature selection method selects only a small subset of features to train the dataset. With the help of this selected small set of features the original dataset is transformed and trained using a classification model. We used Support Vector Machine (SVM) (Cortes and Vapnik, 1995) in this paper due to superiority over other methods (Ding et al., 2016b). The trained model is saved for prediction phase. Whenever a new sequence is given, it goes through the same process and given the instance with selected features, the trained model predicts its label. For both of the problems (PH vs non-PH and PHM vs PHC), we follow the same procedure.

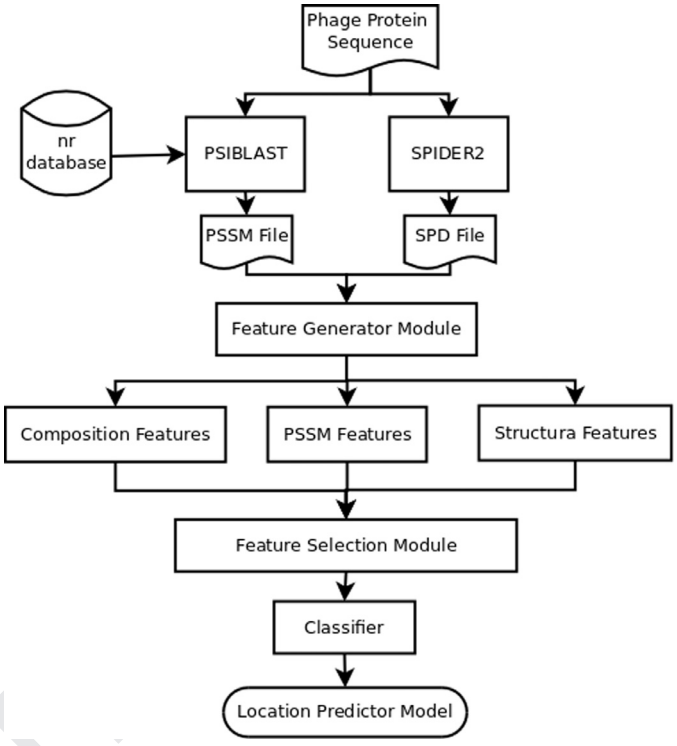

Fig. 1. System flowchart of iPHLoc-ES.

Table 1
Summary of bacteriophage protein dataset for pH vs non-PH prediction.

Phage Type	Number of Samples
Host-Located Proteins (PH)	144
Extra-Cellular Proteins (non-PH)	134

2.1. Benchmark dataset

The description of the datasets used in this paper for pH vs non-PH problem is given in Table 1. There are total 278 instances out of which 144 are positive instances or host-located proteins and 134 are extra-cellular proteins or negative samples. This dataset is similar to the one used in Ding et al. (2016a). All the protein sequences are collected from UniProt Database (Consortium, 2014). All these subcellular locations are experimentally validated. Subphages that are part of other phage proteins or the phages with non-standard amino-acids were discarded to generate the dataset. This dataset excludes the redundant sequences with similarity threshold set to 30%.

From the host located protein dataset, a second dataset was derived for PHC vs PHM problem. The description is given in Table 2. In total, 68 phages are location in cell membrane and 76 phages are located in cell cytoplasm.

Table 2
Summary of host located bacteriophage protein dataset for PHC vs PHM prediction.

Location Type	Number of Samples
Cell Membrane (PHM)	68
Cell Cytoplasm (PHC)	76

2.2. Feature generation

Various types of feature extraction techniques are used in the literature for subcellular localization of protein and particularly phage proteins. Among them are PSSM-based features (Sharma et al., 2015; Wang et al., 2017), g-gap dipeptide composition (Ding et al., 2016a), gene ontology based features (Wang et al., 2016), pseudo amino acid composition (Chen et al., 2016), physico-chemical based features (Dehzangi et al., 2015) etc.

With the explosive growth of biological sequences in the post-genomic era, one of the most important but also most difficult problems in bioinformatics and system biology is how to express a biological sequence with a discrete model or a vector, yet still keep considerable sequence-order information or key pattern characteristic. This is because all the existing machine-learning algorithms can only handle vector but not sequence samples, as elucidated in a recent review (Chou, 2015). However, a vector defined in a discrete model may completely lose all the sequence-pattern information. To avoid completely losing the sequence-pattern information for proteins, the pseudo amino acid composition was proposed (Chou, 2001; 2004). Ever since then, the approach of PseAAC has penetrated into nearly all the computational proteomics (Chou, 2017; Khan et al., 2017; Meher et al., 2017; Nanni et al., 2012; Rahimi et al., 2017). Because it has been widely and increasingly used, recently three powerful open access soft-wares, called 'PseAAC-Builder', 'propy', and 'PseAAC-General', were established: the former two are for generating various modes of Chou's special PseAAC; while the 3rd one for those of Chou's general PseAAC (Chou, 2009), including not only all the special modes of feature vectors for proteins but also the higher level feature vectors such as "Functional Domain" mode, "Gene Ontology" mode, and "Sequential Evolution" or "PSSM" mode. Encouraged by the successes of using PseAAC to deal with protein/peptide sequences, similar web-servers (Chen et al., 2014) were developed for generating various feature vectors for DNA/RNA sequences as well. Particularly, an extremely powerful web-server called Pse-in-One (Liu et al., 2017) and its very recently updated version Pse-in-One 2.0 (Liu et al., 2017) have been established that can be used to generate any desired feature vectors for protein/peptide and DNA/RNA sequences according to the need of users' studies.

In this study, we have used three types of features. They are amino-acid sequence based features, PSSM based features and structure based features. First, the PSSM files generated for the phage sequences by PSI-BLAST are used to create a consensus sequence that contains evolutionary information (Sharma et al., 2015). Then, other set of features are extracted from the PSSM file and the SPD file generated by SPIDER. This section presents a brief overview of the features. A summary of all the features used in this paper is given in Table 3.

2.2.1. Sequence based features

A consensus sequence generated by the multiple sequence alignment by PSI-BLAST is used to generate this features. The first group is called the amino-acid composition which is the count or frequency in the given consensus sequenced normalized by the length of the protein. Formally,

$$\mathrm{AAC}(i) = \frac{1}{L}\sum_{j=1}^{L} c_{ij}, 1 \le i \le 20 \quad (1)$$

Here, L is the length of the protein and

$$c_{ij} = \begin{cases} 1, & \text{if } s_j = a_i \\ 0, & \text{else} \end{cases}$$

where s_j is an amino acid in the protein sequence and a_i is one of the 20 different amino-acid symbols (Dehzangi et al., 2014b). Another group of features called Dubchuck features (Dubchak et al., 1999) are also generated using this sequence based information depending on the physico-chemical properties of the amino acids residues, such as polarity, solvability, hydro-phobicity etc.

2.2.2. PSSM based features

PSSM files were generated using three iterations of the PSI-BLAST Algorithm (Altschul et al., 1997) using the non-redundant database (nr) provided by NCBI. The threshold cut-off value of E was set to 0.001. PSSM file returns the log-odds of the substitution probabilities of a given protein at each position for all possible amino-acid symbols after the alignment (Chou and Shen, 2007). This is a $L \times 20$ matrix which we refer in this paper as *PSSM matrix*. We first normalize the pssm matrix using the same technique as proposed in Sharma et al. (2015). After normalization, we generated five groups of features from the normalized PSSM matrix. We will denote the normalized matrix throughout this section as N which is a two dimensional matrix of dimension $L \times 20$. They are enumerated as bellow:

1. **PSSM Bigram:** Bigram features from PSSM matrix are well used in the literature of subcellular localization Sharma et al. (2013); 2015) and defined as below:

$$\text{PSSM-bigram}(k,l) = \frac{1}{L}\sum_{i=1}^{L-1} N_{i,k}N_{i+1,l} (1 \le k \le 20, 1 \le l \le 20) \quad (2)$$

2. **PSSM 1-lead Bigram:** PSSM 1-lead bigram is defined in a similar way to PSSM bigram:

$$\text{PSSM-1-lead-bigram}(k,l) = \frac{1}{L}\sum_{i=1}^{L-2} N_{i,k}N_{i+2,l} \quad (1 \le k \le 20, 1 \le l \le 20) \quad (3)$$

3. **PSSM Composition:** PSSM composition is created by taking the normalized sum of the column wise values in the PSSM matrix Sharma et al. (2015). It is defined as:

$$PSSM - Composition(k,l) = \frac{1}{L}\sum_{i=1}^{L-1} N_{i,j} (1 \le j \le 20) \quad (4)$$

4. **PSSM Auto-Covariance:** Auto-Covariance of PSSM is a feature Dehzangi et al. (2014a); Sharma et al. (2015) depending of a distance factor, DF as parameter. In this study we used, DF = 10. The feature is formally defined as:

$$\text{PSSM-Auto-Covariance}(k,j) = \frac{1}{L}\sum_{i=1}^{L-k} N_{i,j}N_{i+k,j} \quad (1 \le j \le 20, 1 \le k \le DF) \quad (5)$$

5. **PSSM Segmented Distribution:**
Previously, the segmented distribution of the PSSM matrix proposed in Dehzangi and Phon-Amnuaisuk (2011) was used as feature for subcellular localization of proteins in Dehzangi et al. (2015). The idea is to find the distribution

Table 3
Summary of evolutionary and structural features used.

Feature group	Number of features	Reference
Amino-acid composition	20	Sharma et al. (2015)
Dubchuck features	105	
PSSM bigram	400	Sharma et al. (2015)
PSSM 1-lead bigram	400	Dehzangi and Phon-Amnuaisuk (2011)
PSSM composition	20	Sharma et al. (2015)
PSSM auto-covariance	200	Sharma et al. (2015)
PSSM segmented distribution	200	Dehzangi et al. (2015)
Secondary structure occurence	3	This paper
Secondary structure composition	3	
Accessible surface area composition	1	
Torsional angles composition	8	
Structural probabilities composition	3	
Torsional angles bigram	64	
Structural probabilities bigram	9	
Torsional angles auto-covariance	80	
Structural probabilities auto-covariance	30	
Total	1546	

of the values in the PSSM matrix column wise by calculating the partial sums column wise starting from the first row and the last row and iterating until the partial running sum is F_p % of the total sum. The details of the procedure for this feature generation can be found in Dehzangi et al. (2013); 2015), Dehzangi and Sattar (2013). In this paper, we used $F_p = 5, 10, 25$.

2.2.3. Structure based features

We hypothesize that along with the sequential and evolutionary information, structural information also can affect the subcellular localization of phage proteins. Therefore, we extract a novel set of features generated using the SPD files produced by SPIDER2 software (Heffernan et al., 2015; Yang et al., 2017). The SPD files generated by SPIDER2 contains, secondary structural motif and their probabilities, accessible surface area and torsional angles for each amino-acid residue. All the feature groups generated from SPIDER2 are enumerated here:

1. **Secondary structure occurence:** This feature is the count or frequencies of the structural motifs present in amino-acid residue positions. There are three types of motifs: α-helix (H), β-sheet (E) and random coil (C).
2. **Secondary structure composition:** This feature is the normalized secondary structure occurrence by the length of the phage protein length. This is similar to the amino-acid composition except that here we are taking the count of motif symbols in stead of amino-acid symbols.

$$\text{SS-Composition}(i) = \frac{1}{L}\sum_{j=1}^{L} c_{ij}, 1 \le i \le 3 \quad (6)$$

here, L is the length of the protein and

$$c_{ij} = \begin{cases} 1, & \text{if } SS_j = f_i \\ 0, & \text{else} \end{cases}$$

where SS_j is the structural motif at position j of the protein sequence and f_i is one of the 3 different motif symbols.

3. **Accessible surface area composition:** The accessible surface area composition is the normalized sum of accessible surface area defined by:

$$\text{ASA-Composition} = \frac{1}{L}\sum_{i=1}^{L} ASA(i) \quad (7)$$

4. **Torsional angles composition:** For four different types of torsional angles: ϕ, ψ, τ and θ we first convert each of them into radians from degree angles and then take sign and cosine of the angles at each residue position. Thus we get a matrix of dimension $L \times 8$. We denote this matrix by T is this section for torsional angles. Torsional angles composition is defined as:

$$\text{Torsional-Angles-Composition(k)} = \frac{1}{L}\sum_{i=1}^{L} T_{i,k} (1 \le k \le 8) \quad (8)$$

5. **Structural probabilities composition:** Structural probabilities for each position of the amino-acid residue are given in spd3 file as a matrix of dimension $L \times 3$. We denote it by P. Structural probabilities composition is defined as:

$$\text{Structural-Probabilities-Composition(k)} = \frac{1}{L}\sum_{i=1}^{L} P_{i,k} (1 \le k \le 3) \quad (9)$$

6. **Torsional angles bigram:** Bigram for the torsional angles is similar to that of PSSM matrix and defined as:

$$\text{Torional-angles-bigram}(k, l) = \frac{1}{L}\sum_{i=1}^{L-1} T_{i,k} T_{i+1,l} \quad (1 \le k \le 8, 1 \le l \le 8) \quad (10)$$

7. **Structural probablities bigram:** Bigram of the structural probabilities is similar to that of PSSM matrix and defined as:

$$\text{Structural-Probabilities-bigram}(k, l) = \frac{1}{L}\sum_{i=1}^{L-1} P_{i,k} P_{i+1,l} \quad (1 \le k \le 3, 1 \le l \le 3) \quad (11)$$

8. **Torsional angles auto-covariance:** This feature is also derived from torsional angles and defined as:

$$\text{Torsional-Angles-Auto-Covariance}(k, j) = \frac{1}{L}\sum_{i=1}^{L-k} T_{i,j} T_{i+k,j} \quad (1 \le j \le 8, 1 \le k \le DF) \quad (12)$$

9. **Structural probablities auto-covariance:** This feature is also derived from structural probabilities and defined as:

$$\text{Structural-Probabilities-Auto-Covariance}(k, j) = \frac{1}{L}\sum_{i=1}^{L-k} P_{i,j} P_{i+k,j} (1 \le j \le 3, 1 \le k \le DF) \quad (13)$$

2.3. Recursive feature elimination

For both of the problems, the total number of features generated is higher than the number of instances. This possibly can lead to the curse of dimensionality (Friedman, 1997; Keogh and Mueen, 2011). Therefore, we adopt a feature selection technique to reduce the number features and avoid potential curse of dimensionality. Several techniques are reported in the literature for feature selection or dimensionality reduction for classification problems (Saeys et al., 2007). Among them are genetic programming (Nanni and Lumini, 2008), recursive feature elimination (Guyon et al., 2002), tree based method (Deng and Runger, 2012), randomized sparse elimination (Bach; Meinshausen and Bühlmann, 2010), and incremental forward selection algorithm (Ding et al., 2016a). To select the most effective feature reduction method, we choose several of most popular techniques and compared their performance for our problems. Among these methods using recursive feature elimination technique attained better results compared to the other methods. Therefore, we use this method as our main feature selection scheme.

Recursive feature elimination (RFE) was first proposed in Guyon et al. (2002). The idea of the algorithm is depicted as pseudo-code in Algorithm 1 . It starts with a given dataset and iteratively classifies the dataset given a classifier and then rank the all the features following a given criteria. It then removes the feature with lowest rank from the feature set and transforms the dataset accordingly and continues the whole process again and again until the dataset is reduced to k features.

Algorithm 1: RecursiveFeatureElimination(*dataset*, *classifier*, k).

```
dataset' ← dataset;
FeatureSet = {All features};
while |FeatureSet| < k do
    classifier.train(dataset');
    FeatureSet.computeRanks();
    f_r ← Featureset.selectLowestRank();
    FeatureSet ← FeatureSet − {f_r};
    dataset' =transform(dataset, FeatureSet);
end
return dataset'
```

Usually an external estimator used used to assign weights to the features. For example if a linear estimator is used then the weights are the coefficients of the linear model.

2.4. Support vector machine

In this study, we use Support Vector Machine (SVM) (Cortes and Vapnik, 1995) as classification model for both of the problems: pH vs non-PH and PHC vs PHM. During the last few years, a wide range of classification techniques have been used to tackle these problems. Among them, SVM attained the best results (Dehzangi et al., 2014a; Ding et al., 2016a; Sharma et al., 2015). Therefore, we use this classifier to build our model. SVM is non parametric classifier that aims at finding the marginal hyperplane with maximum distance from different classes to achieve the lowest error and highest generality. A comparison of the performance of our model with different classifiers to solve the two problems are presented in the results section of this paper.

2.5. Performance evaluation

A wide varieties of comparison matrices has been used in the literature of supervised learning to evaluate the performances of different prediction algorithms Powers. In this paper, we used several of them as defined in the following equation:

$$\begin{cases} Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \\ Sensitivity = \frac{TP}{TP+FN} \\ Specificity = \frac{TN}{TN+FP} \\ MCC = \frac{(TP\times TN)-(FP\times FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \end{cases} \quad (14)$$

For each of the problem, the dataset is considered as a set containing positive and negative samples.

$$\mathbb{S} = \mathbb{S}^- \cup \mathbb{S}^+ \quad (15)$$

In a typical binary classification problem, one of the classes is considered as negative and the other as positive. Now, TP is the number of positive examples correctly classified, TN is the number of negative samples correctly predicted, FP is the number of positive examples incorrectly classified and FN is the number of negative examples incorrectly classified examples.

In addition to these measures, we also used area under Receiver Operating Characteristic (auROC) and area under precision recall curve (auPR) to measure the performance of the algorithms. The set of metrics is valid only for the single-label systems. For the multi-label systems whose existence has become more frequent in system biology (Cheng et al., 2017b; 2017c; 2017d) and system medicine (Cheng et al., 2016; Qiu et al., 2016), a completely different set of metrics as defined in Cheng et al. (2017b) and Chou (2013) is needed.

Several sampling methods (Efron and Gong, 1983) are used in the literature to assess the performance of the classification algorithms for supervised learning. Among them jackknife and cross-validation are the most popular ones. In this paper, we employed both k-fold cross-validation with $k = 10$ and jack knife test to be able to directly compare our method with the previous studies found in the literature. It is very important to test the predictors using any of these acceptable sampling methods to tackle the bias-variance trade-off (Friedman, 1997).

3. Results and discussion

In this section, we present the results of the experiments that were carried in this study. All the methods were implemented in Python. Each of the experiments were carried 5 times and only the average is reported as results.

3.1. Feature selection method

The first challenge to solve these two problems were the large number of features that we extracted that potentially can cause curse of dimensionality (Friedman, 1997; Keogh and Mueen, 2011). Several candidate feature reduction methods are available in literature. To see the effect of the different feature selection methods, we applied them on the dataset for pH vs non-pH problem. Three different methods were tried: recursive feature elimination (RFE) (Guyon et al., 2002), tree based method (Deng and Runger, 2012) and randomized sparse elimination (Meinshausen and Bühlmann, 2010), Bach. For each of these methods, we ran the algorithms using 10-fold cross validation on the dataset. Those results are shown in Table 4. As it is shown in Table 4, Recursive feature elimination show superior performance compared to other two feature selection methods in terms of all the measures. We also plot Receiver Operating Characteristic (ROC) curve to see the effectiveness of the feature selection methods. The plot of the ROC curve is given in Fig. 2. The area under ROC curve value is maximum for the recursive feature elimination method which is 0.9623 with accuracy 89.92%.

Table 4
Comparison of performance of different types of feature elimination techniques on pH vs non-pH classification.

pH vs no-pH Classification						
Method	Accuracy	Sensitivity	Specificity	MCC	auROC	auPR
RFE	89.92%	0.8805	0.9166	0.8044	0.9623	0.9195
Tree Based Classifier	66.54%	0.7164	0.6180	0.3548	0.75354	0.6330
Sparse Elimination	74.10%	0.7462	0.7361	0.4872	0.8010	0.7437

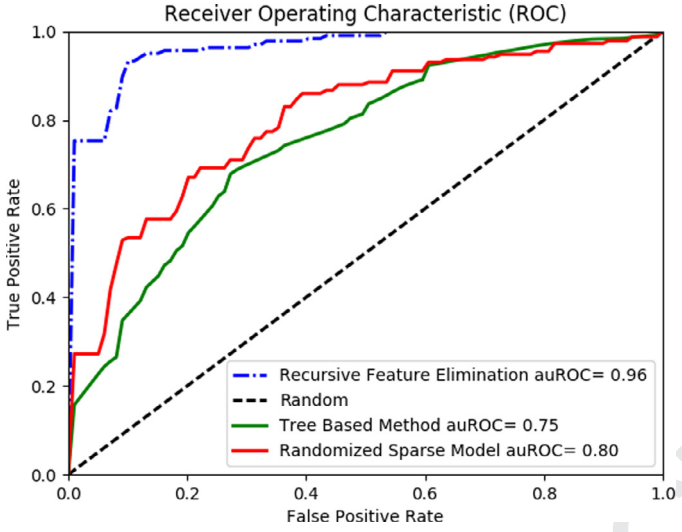

Fig. 2. Receiver Operating Characteristic curves for different feature selection methods.

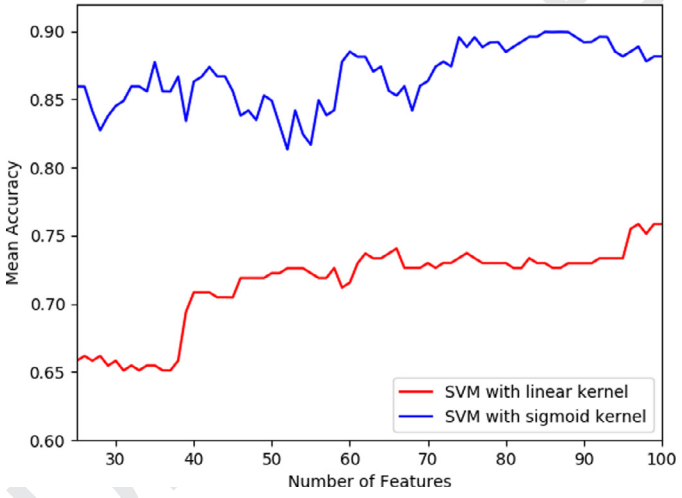

Fig. 3. Mean accuracy achieved for different number of selected features using different kernels of SVM using recursive feature selection algorithm.

For the same dataset, we performed another set of experiments to find the optimal number of features required for the classification problem of pH vs non-pH problem. We varied the number of features to be selected by the RFE algorithm and performed 10-fold cross fold validation on the data. We tried two different classifiers in this setting: support vector machine with linear kernel and sigmoid kernel with the parameters, $C = 1000$ and $\gamma = 0.01$. Mean accuracy obtained in the experiments are shown in Fig. 3. The number of features were exhaustively tried in the range [25, 100]. The highest accuracy was found when the number of features set in Algorithm 1 was set to 85.

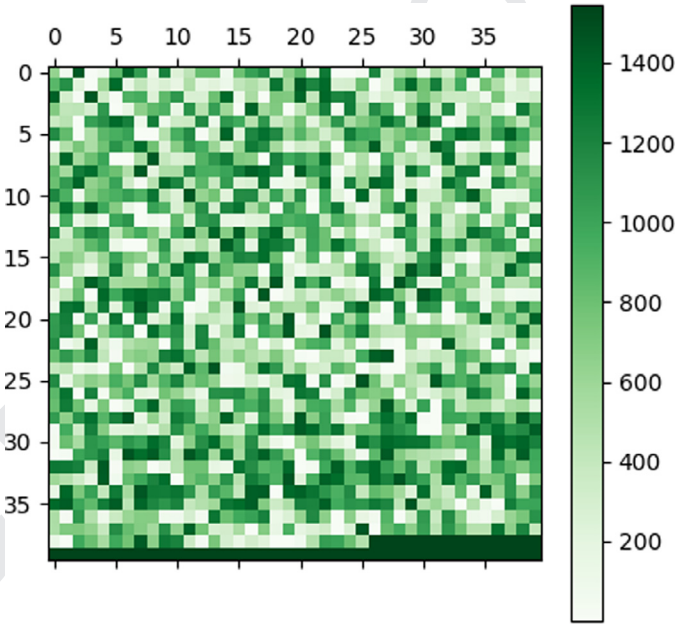

Fig. 4. Ranking of all 1546 features shown in a color map showing the importance of the features, the darker the color is, less important the feature.

Color map of the rankings of the features as ranked by the RFE algorithm is given in 4. This map shows the distribution of selected features over all the features. Selected features include Dubchuck features, PSSM bigram, PSSM Auto-Covariance, PSSM 1-lead bigram and PSSM segmented distribution from the evolutionary group of features extracted for PSSM and the rest of the features were structural features generated by SPIDER3. It reveals the importance of both type of features: evolutionary and structural. We used the same number and set of features also for the PHM vs PHC problem. The selected features are given as supporting information with the paper.

3.2. Classifiers

To see the effect of the different classification algorithms, we applied different types of supervised learning algorithms on the dataset of pH vs non-pH classification problem. We tried six classifiers in our experiments. They were: Support Vector Machine with linear kernel, Support Vector Machine with rbf kernel, Support Vector Machine with sigmoid kernel, Random Forest Classifier, Naive Bayes Classifier and Logistic Regression Classifier. We used 10-fold cross validation in the experiments and mean values of performance metrics are reported in Table 5.

From the values reported in Table 5, it is clearly noticed that the best classification algorithm for the pH vs non-pH problem is SVM with linear kernel. In this experiments, we used the same features that were selected in the feature selection phase using RFE algorithm. Logistic Regression algorithm was the second best with 85.97% accuracy and auROC value of 0.9326. We have also

Table 5
Comparison of performance of prediction of different types of classification algorithms.

Classifier	Accuracy	Sensitivity	Specificity	MCC	auROC	auPR
SVM (linear kernel)	89.92%	0.8805	0.9166	0.8044	0.9623	0.9195
SVM (rbf kernel)	79.13%	0.8134	0.7708	0.5896	0.8641	0.7779
SVM (sigmoid kernel)	57.91%	0.5671	0.5902	0.1571	0.6351	0.5925
Random Forest	69.06%	0.7388	0.6458	0.4034	0.7764	0.6589
Naive Bayes	59.35%	0.4626	0.7152	0.2054	0.6708	0.7249
Logistic Regression	85.97%	0.8582	0.8611	0.7267	0.9326	0.8752

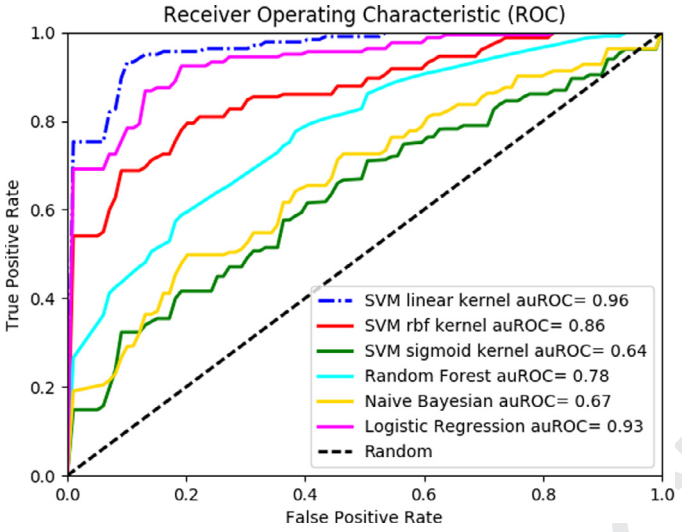

Fig. 5. Receiver Operating Characteristic curves for different classification algorithms.

Table 6
Comparison of results achieved by iPHLoc-ES with other predictors.

	PH vs non-PH		PHM vs PHC	
Method Name	Accuracy	auROC	Accuracy	auROC
PHPred	84.2%	0.872	92.4%	0.970
iPHLoc-ES (10-fold)	89.92%	0.962	100%	0.994
iPHLoc-ES (Jack Knife)	88.48%	0.952	100%	0.992

plot the Flase Positive Rate vs True Positive Rate or Receiver Operating Characteristic (ROC) curve for all these classifiers on the dataset. The plot is given in Fig. 5. From this analysis we selected the SVM classifier for our predictor with linear kernel.

3.3. Comparison with other methods

In this section, we analyze the performance of our method with that of the other state-of-the-art prediction PHPred (Ding et al., 2016a). For a fair comparison, we performed jack knife test on our datasets and reported mean accuracy and mean area under ROC curve in Table 6. We have used the selected features and the classification algorithm from the previous experiments and applied it on both of the problems and the respective datasets. In case of the pH vs non-PH problem, the jack knife test was able to produce results with 88.48% accuracy and area under ROC curve of 0.952 compared to the accuracy of PHPred of 84.2% and area under ROC curve of 0.872. Evaluating our results using 10-fold cross validation, we achieved similar and slightly better results for our prediction algorithm iPHLoc-ES.

In the case of PHM vs PHC classification, our algorithm was able to predict all the subcellular localization of host located proteins correctly. The accuracy was perfect (100%) with area under ROC value 0.994 compared to the 92.4% accuracy and 0.970 area under ROC curve value of PHPred. Thus, for both of the problems and their datasets, iPHLoc-ESis able to significantly outperform PHPred, which is the current best known predictor for the problem.

3.4. Discussion

In this study, We have developed a method named iPHLoc-ES that significantly outperformed the previously proposed methods for prediction of subcellular localization of bacteriophage proteins. The performance of iPHLoc-ES was superior than PHPred as the most accurate predictor that was recently developed in terms of all the comparison metrics used in this paper. The accuracy of the first problem of discrimination of host located phage proteins from the extra-cellular phage proteins (PH vs non-PH) was improved from 84% accuracy to 88.48% accuracy using jack knife test. The improvement in the other problems ware even higher. We achieved the classification accuracy of 100% compared to that of 92.4% for PHPred. Similar improvements are noticed in Table 6 for other metrics as well.

The receiver operating characteristic graph which is a plot of false positive rate against true positive rate is very important when considered balanced data. In terms of imbalanced data, often area under Precision-Recall Curve and balanced accuracies are often considered for performance consideration. In our case, the datasets were quite balanced as shown in Tables 1 and 2. Hence the measure of area under ROC curve is sufficient to compare the performance of the algorithms or methods. At the same time iPHLoc-ES achieve very high sensitivity and specificity as well. For the second problem we achieve to 100% prediction performance. Note that we admit that the number of samples present in the dataset is very small which may cause very high performance and hard to generalize. However, this is due to the lack of experimentally validated phage locations available. Moreover, a number of phages were discarded for several reasons including sequence similarity and others. We aim at employing iPHLoc-ES for larger benchmarks as soon it is made available.

One of the main success of iPHLoc-ES is due to the efficient feature selection. It is important to note that most of the features were previously used in the literature for protein subcellular localization except the structural features. It was very important to reduce the number of features and remove the curse of dimensionality and hence select only effective and discriminatory features for classification. It is also important to note that logistic regression classifier and SVM with linear kernel were among the best performing classification algorithms.

3.5. Web server implementation

To make our method available as a web application we implemented an web application and made it publicly available from: http://brl.uiu.ac.bd/iPHLoc-ES/. The web application was developed using PHP and python language. It is very simple to use. This predictor can be used to find two types of prediction results: pH vs non-PH and PHM vs PHC. This can be selected using the option button. For the prediction one need to provide two files to the

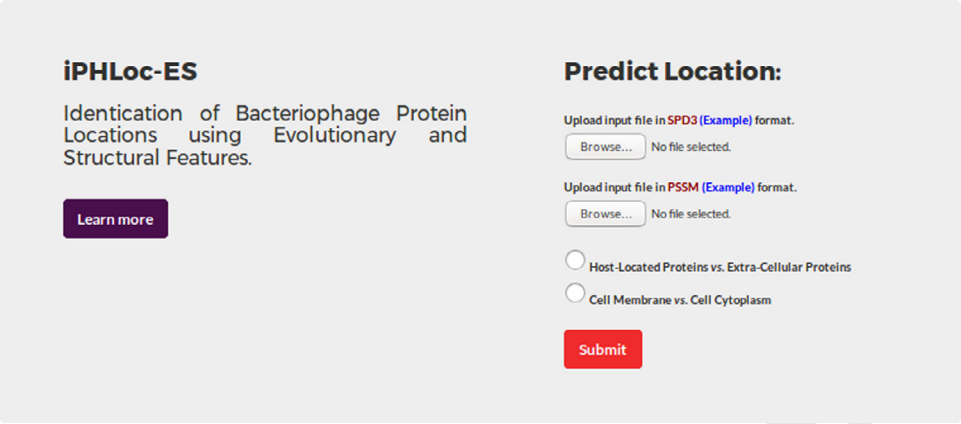

Fig. 6. Screen shot of the web application implemented for the iPHLoc-ES predictor.

predictor: a pssm file generated from PSI-BLAST and a SPD file generated from SPIDER2 software. After that one might expect an instantaneous prediction of the location of the given protein based on the option. A typical screen shot of the system in given in Fig. 6.

4. Conclusion

In this paper, we have proposed a prediction method for subcellular localization of bacteriophage proteins. Two problems were addressed in this regard on an experimentally validated dataset. The features generated from the phage protein sequences were based on evolutionary and structural information and were proven to be successful in predicting locations of phage proteins in the host cell. We also used Recursive feature selection to reduce the number of features and that drastically improved the performance of the classifier. Furthermore, we implemented our model (iPHLoc-ES) as a publicly available web server. However, one limitation to the proposed work is that the dataset is small. All these sample phage proteins are taken from latest protein database. However, since the field of phage therapy is getting popular day by day, we believe the number of experimentally validated phage locations will be increased and hence prediction models will be enhanced.

References

Akhter, S., Aziz, R.K., Edwards, R.A., 2012. Phispy: a novel algorithm for finding prophages in bacterial genomes that combines similarity-and composition-based strategies. Nucleic Acids Res. 40 (16), e126.
Altman, E., Young, K., Garrett, J., Altman, R., Young, R., 1985. Subcellular localization of lethal lysis proteins of bacteriophages lambda and phix174.. J. Virol. 53 (3), 1008–1011.
Altschul, S.F., Madden, T.L., Schäffer, A.A., Zhang, J., Zhang, Z., Miller, W., Lipman, D.J., 1997. Gapped blast and psi-blast: a new generation of protein database search programs. Nucleic Acids Res. 25 (17), 3389–3402.
Arndt, D., Grant, J.R., Marcu, A., Sajed, T., Pon, A., Liang, Y., Wishart, D.S., 2016. Phaster: a better, faster version of the phast phage search tool. Nucleic Acids Res. 44 (W1), W16–W21.
Bach, F., Model-consistent sparse estimation through the bootstraparxiv:0901.3202.
Buffie, C.G., Jarchum, I., Equinda, M., Lipuma, L., Gobourne, A., Viale, A., Ubeda, C., Xavier, J., Pamer, E.G., 2012. Profound alterations of intestinal microbiota following a single dose of clindamycin results in sustained susceptibility to clostridium difficile-induced colitis. Infect. Immun. 80 (1), 62–73.
Casjens, S., Hendrix, R., 1988. Control mechanisms in dsdna bacteriophage assembly. In: The Bacteriophages. Springer, pp. 15–91.
Chen, W., Lei, T.-Y., Jin, D.-C., Lin, H., Chou, K.-C., 2014. Pseknc: a flexible web server for generating pseudo k-tuple nucleotide composition. Anal. Biochem. 456, 53–60.
Chen, X.-X., Tang, H., Li, W.-C., Wu, H., Chen, W., Ding, H., Lin, H., 2016. Identification of bacterial cell wall lyases via pseudo amino acid composition. Biomed. Res. Int..
Cheng, X., Xiao, X., Chou, K.-C., 2017. ploc-mEuk: Predict subcellular localization of multi-label eukaryotic proteins by extracting the key go information into general PseAAC. Genomics doi:10.1016/j.ygeno.2017.08.
Cheng, X., Xiao, X., Chou, K.-C., 2017. ploc-mplant: Predict subcellular localization of multi-location plant proteins by incorporating the optimal go information into general pseaac. Mol. Biosyst. 13, 1722–1727.
Cheng, X., Xiao, X., Chou, K.-C., 2017. ploc-mvirus: Predict subcellular localization of multi-location virus proteins via incorporating the optimal go information into general pseaac. Gene 628, 315–321.
Cheng, X., Zhao, S.-G., Lin, W.-Z., Xiao, X., Chou, K.-C., 2017. ploc-mAnimal: Predict subcellular localization of animal proteins with both single and multiple sites. Bioinformatics. btx476
Cheng, X., Zhao, S.-G., Xiao, X., Chou, K.-C., 2016. iatc-misf: A multi-label classifier for predicting the classes of anatomical therapeutic chemicals. Bioinformatics 33 (3), 341–346.
Chou, K., 2017. An unprecedented revolution in medicinal chemistry driven by the progress of biological science. Curr. Top. Med. Chem.
Chou, K.-C., 2001. Prediction of protein cellular attributes using pseudo-amino acid composition. Proteins Struct. Funct. Bioinf. 43 (3), 246–255.
Chou, K.-C., 2004. Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. Bioinformatics 21 (1), 10–19.
Chou, K.-C., 2009. Pseudo amino acid composition and its applications in bioinformatics, proteomics and system biology. Curr. Proteomics 6 (4), 262–274.
Chou, K.-C., 2011. Some remarks on protein attribute prediction and pseudo amino acid composition. J. Theor. Biol. 273 (1), 236–247.
Chou, K.-C., 2013. Some remarks on predicting multi-label attributes in molecular biosystems. Mol. Biosyst. 9 (6), 1092–1100.
Chou, K.-C., 2015. Impacts of bioinformatics to medicinal chemistry. Med. Chem. (Los Angeles) 11 (3), 218–234.
Chou, K.-C., Shen, H.-B., 2006. Large-scale predictions of gram-negative bacterial protein subcellular locations. J. Proteome Res. 5 (12), 3420–3428.
Chou, K.-C., Shen, H.-B., 2007. Recent progress in protein subcellular location prediction. Anal. Biochem. 370 (1), 1–16.
Consortium, U., 2014. Uniprot: a hub for protein information. Nucleic Acids Res. gku989.
Cortes, C., Vapnik, V., 1995. Support-vector networks. Mach. Learn. 20 (3), 273–297.
Dehzangi, A., Paliwal, K., Lyons, J., Sharma, A., Sattar, A., 2013. Enhancing Protein Fold Prediction Accuracy Using Evolutionary and Structural Features. In: IAPR International Conference on Pattern Recognition in Bioinformatics. Springer, pp. 196–207.
Dehzangi, A., Paliwal, K., Lyons, J., Sharma, A., Sattar, A., 2014. A segmentation-based method to extract structural and evolutionary features for protein fold recognition. IEEE/ACM Trans. Comput. Biol. Bioinf. 11 (3), 510–519.
Dehzangi, A., Phon-Amnuaisuk, S., 2011. Fold prediction problem: the application of new physical and physicochemical-based features. Protein Pept. Lett. 18 (2), 174–185.
Dehzangi, A., Sattar, A., 2013. Protein fold recognition using segmentation-based feature extraction model. In: Asian Conference on Intelligent Information and Database Systems. Springer, pp. 345–354.
Dehzangi, A., Sharma, A., Lyons, J., Paliwal, K.K., Sattar, A., 2014. A mixture of physicochemical and evolutionary–based feature extraction approaches for protein fold recognition. Int. J. Data Min. Bioinf. 11 (1), 115–138.
Dehzangi, A., Sohrabi, S., Heffernan, R., Sharma, A., Lyons, J., Paliwal, K., Sattar, A., 2015. Gram-positive and gram-negative subcellular localization using rotation forest and physicochemical-based features. BMC Bioinf. 16 (4), S1.
Deng, H., Runger, G., 2012. Feature selection via regularized trees. In: Neural Networks (IJCNN), The 2012 International Joint Conference on. IEEE, pp. 1–8.
Deresinski, S., 2009. Bacteriophage therapy: exploiting smaller fleas. Clin. Infect. Dis. 48 (8), 1096–1101.
Ding, H., Feng, P.-M., Chen, W., Lin, H., 2014. Identification of bacteriophage virion proteins by the anova feature selection and analysis. Mol. Biosyst. 10 (8), 2229–2235.
Ding, H., Liang, Z.-Y., Guo, F.-B., Huang, J., Chen, W., Lin, H., 2016. Predicting bacteriophage proteins located in host cell with feature selection technique. Comput. Biol. Med. 71, 156–161.

Ding, H., Yang, W., Tang, H., Feng, P.-M., Huang, J., Chen, W., Lin, H., 2016. Phypred: a tool for identifying bacteriophage enzymes and hydrolases. Virol. Sin. 31 (4), 350.

Dubchak, I., Muchnik, I., Mayor, C., Dralyuk, I., Kim, S.-H., 1999. Recognition of a protein fold in the context of the scop classification. Proteins Struct. Funct. Bioinf. 35 (4), 401–407.

Efron, B., Gong, G., 1983. A leisurely look at the bootstrap, the jackknife, and cross-validation. Am. Stat. 37 (1), 36–48.

Emanuelsson, O., Nielsen, H., Brunak, S., Heijne, G.V., 2000. Predicting subcellular localization of proteins based on their n-terminal amino acid sequence. J. Mol. Biol. 300 (4), 1005–1016.

Feng, P.-M., Ding, H., Chen, W., Lin, H., 2013. Naive Bayes classifier with feature selection to identify phage virion proteins. Comput. Math. Methods Med.

Fouts, D.E., 2006. Phage_finder: automated identification and classification of prophage regions in complete bacterial genome sequences. Nucleic Acids Res. 34 (20), 5839–5851.

Friedman, J.H., 1997. On bias, variance, 0/1loss, and the curse-of-dimensionality. Data Min. Knowl. Discovery 1 (1), 55–77.

Galiez, C., Magnan, C., Coste, F., Baldi, P., 2015. Viralpro: A New Suite for Identifying Viral Capsid and Tail Sequences.

Guyon, I., Weston, J., Barnhill, S., Vapnik, V., 2002. Gene selection for cancer classification using support vector machines. Mach. Learn. 46 (1), 389–422.

Heffernan, R., Paliwal, K., Lyons, J., Dehzangi, A., Sharma, A., Wang, J., Sattar, A., Yang, Y., Zhou, Y., 2015. Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. Sci. Rep. 5, 11476.

Hughes, J.M., 2011. Preserving the lifesaving power of antimicrobial agents. JAMA 305 (10), 1027–1028.

Keen, E.C., 2012. Phage therapy: concept to cure. Front. Microbiol. 3, 238.

Keogh, E., Mueen, A., 2011. Curse of dimensionality. In: Encyclopedia of Machine Learning. Springer, pp. 257–258.

Khan, M., Hayat, M., Khan, S.A., Iqbal, N., 2017. Unb-dpc: Identify mycobacterial membrane protein types by incorporating un-biased dipeptide composition into Chou's general pseaac. J. Theor. Biol. 415, 13–19.

Lederberg, J., 1996. Smaller fleas··· ad infinitum: therapeutic bacteriophage redux. Proc. Natl. Acad. Sci. 93 (8), 3167–3168.

Liljeqvist, T.G., Andresen, D., Zuo, Y., Weston, C., 2012. Antimicrobial resistance: moving forward to the past. N. S. W. Public Health Bull. 23 (2), 37.

Liu, B., Wu, H., Chou, K.-C., 2017. Pse-in-one 2.0: an improved package of web servers for generating various modes of pseudo components of dna, rna, and protein sequences. Nat. Sci. (Irvine) 9 (04), 67.

McNair, K., Bailey, B. A., Edwards, R. A., 2012. Phacts, a computational approach to classifying the lifestyle of phages. Bioinformatics, 28, 5, 614–618.

Meher, P.K., Sahu, T.K., Saini, V., Rao, A.R., 2017. Predicting antimicrobial peptides with improved accuracy by incorporating the compositional, physico-chemical and structural features into Chou's general PseAAC. Sci. Rep. 7.

Meinshausen, N., Bühlmann, P., 2010. Stability selection. J. R. Stat. Soc. 72 (4), 417–473.

Nanni, L., Lumini, A., 2008. Genetic programming for creating Chous pseudo amino acid based features for submitochondria localization. Amino Acids 34 (4), 653–660.

Nanni, L., Lumini, A., Gupta, D., Garg, A., 2012. Identifying bacterial virulent proteins by fusing a set of classifiers based on variants of Chou's pseudo amino acid composition and on evolutionary information. IEEE/ACM Trans. Comput. Biol. Bioinf. 9 (2), 467–475.

Powers, D. M.,. Evaluation: from Precision, Recall and f-Measure to ROC, Informedness, Markedness and Correlation.

Qiu, W.-R., Sun, B.-Q., Xiao, X., Xu, Z.-C., Chou, K.-C., 2016. iptm-mlys: identifying multiple lysine ptm sites and their different types. Bioinformatics 32 (20), 3116–3123.

Rahimi, M., Bakhtiarizadeh, M.R., Mohammadi-Sangcheshmeh, A., 2017. Oogenesis_pred: a sequence-based method for predicting oogenesis proteins by six different modes of chou's pseudo amino acid composition. J. Theor. Biol. 414, 128–136.

Rakhuba, D., Kolomiets, E., Dey, E. S., Bacteriophage receptors, mechanisms of phage adsorption and penetration into host cell, Pol. J. Microbiol., Novik, G., 2010. 59, 3, 145–155.

Saeys, Y., Inza, I., Larrañaga, P., 2007. A review of feature selection techniques in bioinformatics. Bioinformatics 23 (19), 2507–2517.

Sass, P., Bierbaum, G., 2007. Lytic activity of recombinant bacteriophage φ11 and φ12 endolysins on whole cells and biofilms of staphylococcus aureus. Appl. Environ. Microbiol. 73 (1), 347–352.

Seguritan, V., Alves Jr, N., Arnoult, M., Raymond, A., Lorimer, D., Burgin, A.B., Salamon, P., Segall, A.M., 2012. Artificial neural networks trained to detect viral and phage structural proteins. PLoS Comput. Biol. 8 (8), e1002657.

Sharma, A., Lyons, J., Dehzangi, A., Paliwal, K.K., 2013. A feature extraction technique using bi-gram probabilities of position specific scoring matrix for protein fold recognition. J. Theor. Biol. 320, 41–46.

Sharma, R., Dehzangi, A., Lyons, J., Paliwal, K., Tsunoda, T., Sharma, A., 2015. Predict gram-positive and gram-negative subcellular localization via incorporating evolutionary information and physicochemical features into Chou's general PseAAC. IEEE Trans. Nanobiosci. 14 (8), 915–926.

Shen, H.-B., Chou, K.-C., 2007. Gpos-ploc: an ensemble classifier for predicting subcellular localization of gram-positive bacterial proteins. Protein Eng. Des. Sel. 20 (1), 39–46.

Shen, H.-B., Chou, K.-C., 2007b. Virus-ploc: a fusion classifier for predicting the subcellular localization of viral proteins within host and virus-infected cells. Biopolymers, 85, 3, 233–240.

Shen, H.-B., Chou, K.-C., 2009. Gpos-mploc: a top-down approach to improve the quality of predicting subcellular localization of gram-positive bacterial proteins. Protein Pept. Lett. 16 (12), 1478–1484.

Shen, H.-B., Chou, K.-C., 2010. Gneg-mploc: a top-down strategy to enhance the quality of predicting subcellular localization of gram-negative bacterial proteins. J. Theor. Biol. 264 (2), 326–333.

Shen, H.-B., Chou, K.-C., 2010. Virus-mploc: a fusion classifier for viral protein subcellular location prediction by incorporating multiple sites. J. Biomol. Struct. Dyn. 28 (2), 175–186.

Sorokulova, I., Olsen, E., Vodyanoy, V., 2014. Bacteriophage biosensors for antibiotic-resistant bacteria. Expert Rev. Med. Devices 11 (2), 175–186.

Wang, J., Yang, B., Revote, J., Leier, A., Marquez-Lago, T. T., Webb, G., Song, J., Chou, K.-C., Lithgow, T.. 2017Possum: a bioinformatics toolkit for generating numerical sequence feature descriptors based on pssm profiles. Bioinformatics.

Wang, X., Li, H., Zhang, Q., Wang, R., 2016. Predicting subcellular localization of apoptosis proteins combining go features of homologous proteins and distance weighted knn classifier. Biomed. Res. Int.

Wu, Z.-C., Xiao, X., Chou, K.-C., 2012. Iloc-gpos: a multi-layer classifier for predicting the subcellular localization of singleplex and multiplex gram-positive bacterial proteins. Protein Pept. Lett. 19 (1), 4–14.

Xiao, X., Wu, Z.-C., Chou, K.-C., 2011. iloc-virus: A multi-label learning classifier for identifying the subcellular localization of virus proteins with both single and multiple sites. J. Theor. Biol. 284 (1), 42–51.

Xiao, X., Wu, Z.-C., Chou, K.-C., 2011. A multi-label classifier for predicting the subcellular localization of gram-negative bacterial proteins with both single and multiple sites. PLoS ONE 6 (6), e20592.

Yang, Y., Heffernan, R., Paliwal, K., Lyons, J., Dehzangi, A., Sharma, A., Wang, J., Sattar, A., Zhou, Y., 2017. Spider2: a package to predict secondary structure, accessible surface area, and main-chain torsional angles by deep neural networks. Prediction Protein Secondary Struct. 55–63.

Zhou, Y., Liang, Y., Lynch, K.H., Dennis, J.J., Wishart, D.S., 2011. Phast: a fast phage search tool. Nucleic Acids Res. 39 (suppl_2), W347–W352.

Please cite this article as: S. Shatabda et al., iPHLoc-ES: Identification of bacteriophage protein locations using evolutionary and structural features, Journal of Theoretical Biology (2017), https://doi.org/10.1016/j.jtbi.2017.09.022