In *Silico* Approach for Prediction of Antifungal Peptides

This Project Report is in partial fulfillment of Requirements of subject

BIG DATA MINING IN HEALTHCARE- BIO 543

At

INDRAPRASTHA INSTITUTE of INFORMATION TECHNOLOGY, DELHI

MEGHAL DANI (MT17144)

SHUBHI TIWARI (MT17057)

30.04.2018

DECLARATION CERTIFICATE

This is to certify that the work presented in the project entitled “In Silico Approach for Prediction of Antifungal Peptides” is an authentic work carried out under supervision and guidance of our instructor Prof. G.P.S. Raghava
The project involves work done earlier by authors of the paper and additional work done by us as a requirement to complete the credits of the subject.

Date: Apr 30,2018 Instructor: Prof. G.P.S.
Raghava
Place: Delhi Submitted by : Meghal Dani
Shubhi Tiwari

ACKNOWLEDGEMENT

We would like to acknowledge and extend our heartfelt gratitude to Prof. G.P.S Raghava, INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY, DELHI who guided us through this project. His keen and vital encouragement, superb guidance, and constant support are the motive force behind this project work.
We would like to show our warm thanks to all lab members and friends for their continuous motivation and encouragement during this project.

Meghal Dani
Shubhi Tiwari

INDEX

INTRODUCTION

Despite of tremendous advances in the field of antibiotics; morbidity and mortality is quite high due to invasive fungal infections. Drug resistance is one of major causes of millions of death worldwide per year due to antifungal infections.In last decade number of peptide based therapeutics have been developed.

One of the major classes of peptide based therapeutics comes from antimicrobial peptides(AMPs). These AMPs can be classified into various kinds of peptides viz., antibacterial, antiviral, antifungal, antiparasitic, etc. Though AMPs cn be used to treat fungal infection but lack specificity which in turn reduces its effectiveness. This led to a strong need to design antifungal peptides (AFPs) to treat fungal infections. AFPs have the potential to kill fungus as it disrupts membrane physiology of fungus.

In current study, an attempt has been made to develop models using machine learning techniques for discriminating antifungal peptides(AFPs) from non-AFPs.

MATERIALS AND METHODS

1. Datasets Preparation:

   We utilised the dataset available:
   http://webs.iiitd.edu.in/raghava/antifp/algo.php.

   It contains sequences of 1459 unique AFPs with 3 datasets, first being main dataset termed as “Antifp_Main” and two alternate datasets termed as “Antifp_DS1” and “Antifp_DS2”. The length of sequences range from 4 to 100.

2. Internal and External Validation

   The datasets were randomly divided into two parts (i) training dataset, which comprises of 80% data (1168 positive and negative sequences) and (ii) validation dataset with 20% data (291 positive and negative sequences).

In case of internal validation, we developed and evaluate prediction models using fivefold cross validation techniques. Here, sequences present in the dataset are divided randomly into five different sets, out of which any four sets out of five are used for training and the remaining fifth set is used for testing. In the process, each set is used once for testing by repeating the process five times, and the final result is calculated by averaging the performance of all the five sets. The validation of any prediction method plays a very significant role in its evaluation. We evaluated the performance of all the models on validation dataset, termed as external validation.

PROCEDURE

1. **Dataset preparation:** Our data consist of 3 datasets, Antifp_Main, Antifp_DS1, Antifp_DS2 each having a positive and negative set of sequences both for training and validation as explained earlier.
2. **Feature Generation**: The following features were generated using python scripts developed by us. Initially, the sequences in .txt format were converted to FASTA format.
    a. Amino Acid Composition: We used Biopython for composition calculation. The amino acid composition tells us about the fraction of each amino acid type within a peptide. The vector of dimension 20 was obtained when the amino acid composition for both AFPs and non-AFPs was calculated by using the following equation:

    Composition(i)=(Ri/N)*100

    Here, Composition (i) is the percent composition of amino acid (i); Ri is the number of residues of type i, and N represents the total number of peptide's residues.

    b. Dipeptide Composition: The dipeptide composition provides the composition of the residues present in a pair (e.g., A-A, A-L, etc.) in the peptide, and used to convert the variable length of peptides to fixed length feature vector size of 400. It summarizes information about the amino acid's fraction as well as their local order. Dipeptide composition is calculated using following equation:

Dipeptide fraction(i)= (TotalnumberofDipeptide(i)/Totalnumberofallpossibledipeptides)*100

Where dipeptide (i) is 1 out of 400 dipeptides.

c. Split Composition: We also compute amino acid and dipeptide composition of N-terminus and C-terminus residues; first 5, 10, and 15 residues from N-terminus and the last 5, 10, and 15 residues from the C-terminus. Also, we joined the terminal residues like N5C5, N10C10, and N15C15 and checked the performance of combination

d. Binary Profile: In this study, length of antifungal and non-AFP is variable, thus it is difficult to generate fixed length pattern. Thus we extract fixed length segment from either N-terminus or C-terminus of the peptide to generate fixed length binary profile.A vector of dimension 20 represented each amino acid in segment obtained from terminal residues. We generated binary profiles for first 5, 10, and 15 N-terminus residues and for the last 5, 10, and 15 residues from the peptide C-terminus. We also created the binary profile for the N5C5, N10C10, and N15C15 residues of peptides by combining N- and C-terminus residues.

e. PhysicoChemical Properties:Default parameters were used for calculation of mass, charge, and pI values and the values were used as features along with the amino acid composition on which the best performance was found. We wanted to check whether adding these properties would help in further increasing the performance of a model.

3. **Machine Learning** : Broadly, the performance of any classification is measured using two type measures call threshold-dependent and threshold-independent. In this study, we used both types of measure to evaluate the performance of models. In case of threshold-dependent parameters, we compute performance of a model in following terms; Sensitivity (Sen), Specificity (Spc), Accuracy (Acc) and ROC-AUC.

   The models used by authors, and us (both considered and disregarded) are

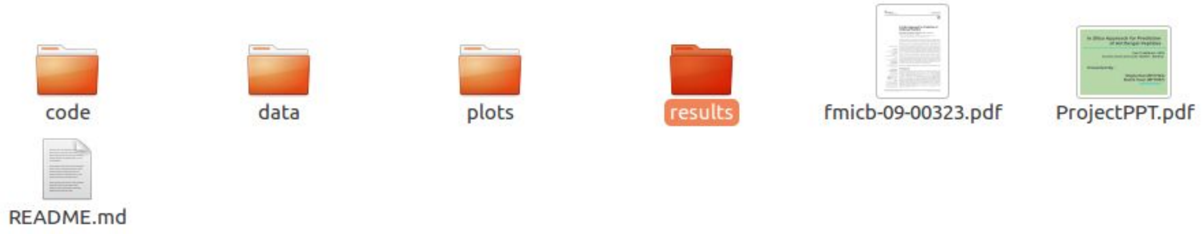
   As given in the table below.

	Techniques	**Applied in paper**	**Applied by Us**	
			Selected	Rejected
01	SVM Radial	Yes	Yes	-
02	Random Forest	Yes	Yes	-
03	J48	Yes	Yes	-
04	Naive Bayes	Yes	Yes	-
05	ELM	-	-	Yes
06	Adaboost.M1	-	Yes	-
07	KNN	-	Yes	-
08	Neural Network	-	-	Yes
09	Rotation Forest	-	-	Yes
10	Ensemble (SVM radial, RF, J48, Adaboost.M1)	-	Yes	-

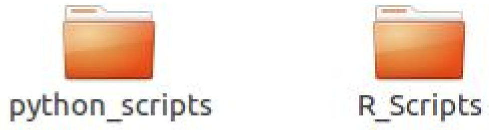
NAVIGATING THE PROJECT FOLDER

The project folder is organized as below:

It is available on github :
https://github.com/tiwariShubhi/Prediction_Antifungal_Peptide



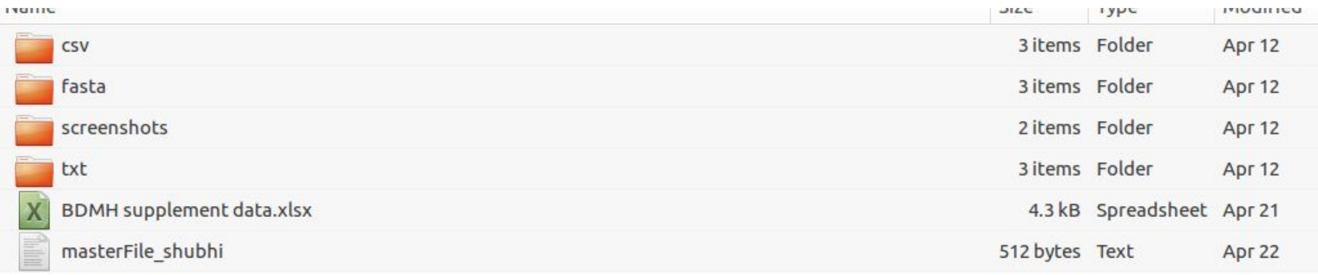
Code folder contains all python and R scripts



S.No.	Code File	Description
1	Convert to fasta	To convert txt to fasta
2	utility.py	Python utility to run all models on number of train and test files in one go
3	pep.py	Generate peptide composition from sequences
4	dipep.py	Generate dipeptide composition from sequence
5	C_k.py	Generate c-k terminus composition from sequence
6	C_k_binary.py	Generate c-k terminus composition in binary from sequence

7	C_k_bin-dipep.py	Generate c-k terminus composition for dipeptide in binary from sequence
8	C_k_dipep.py	Generate c-k terminus composition for dipeptide from sequence
9	N_k.py	Generate n-k terminus composition from sequence
10	N_k_binary.py	Generate n-k terminus composition in binary from sequence
11	N_k_bin-dipep.py	Generate n-k terminus composition for dipeptide in binary from sequence
12	N_k_dipep.py	Generate n-k terminus composition for dipeptide from sequence
13	N_C_k.py	Generate n-k c-k terminus composition from sequence
14	N_C_k_binary.py	Generate n-k c-k terminus composition in binary from sequence
15	N_C_k_bin-dipep.py	Generate n-k c-k terminus composition for dipeptide in binary from sequence
16	N_C_k_dipep.py	Generate n-k c-k terminus composition for dipeptide from sequence
17	mass_charge_pi.py	Generate physicochemical composition for peptide from sequence
18	model_project_res.R	Trains and validates all models on a input test and train file and store results in CSV
19	model_ensemble.R	Trains and validates ensemble model on a input test and train file.

Data can be found in the data folder

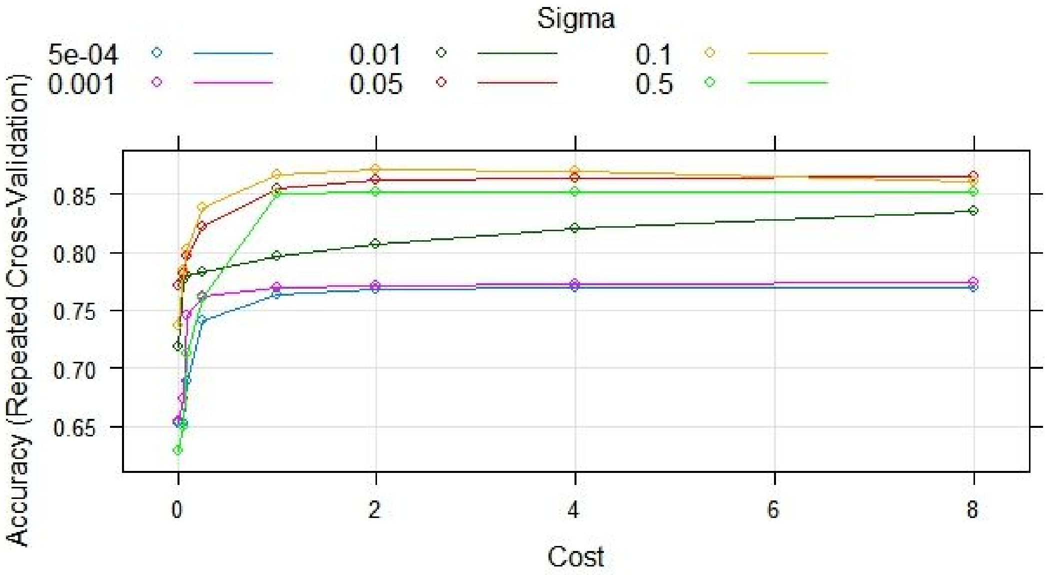


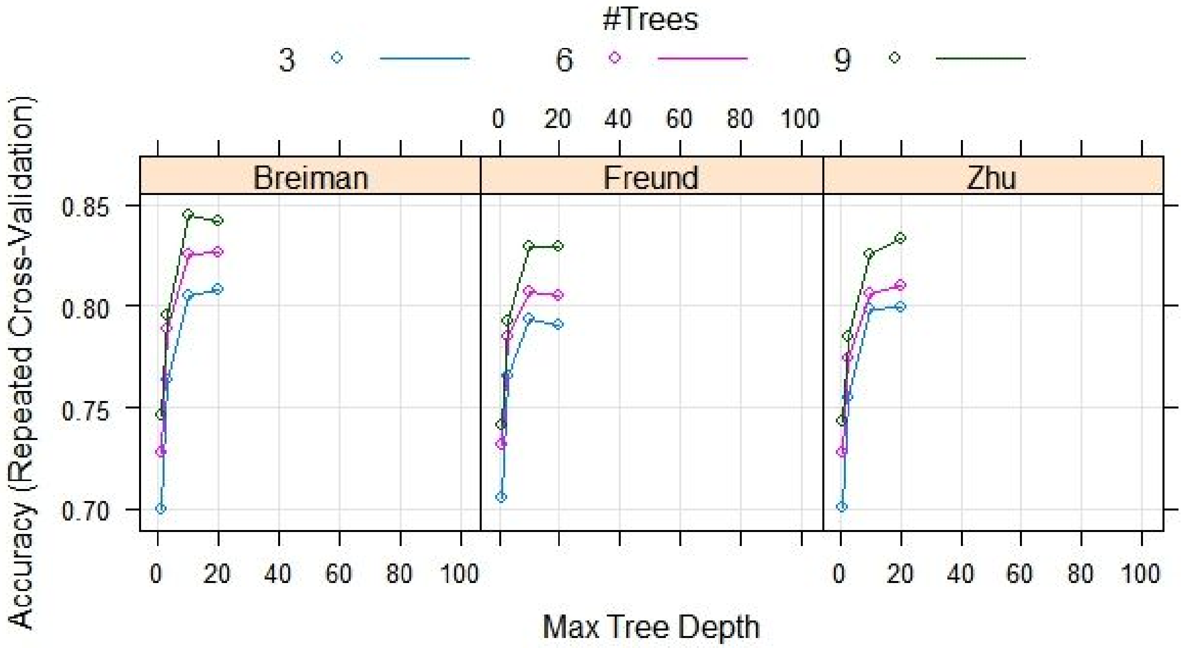
Name	Size	Type	Modified
csv	3 items	Folder	Apr 12
fasta	3 items	Folder	Apr 12
screenshots	2 items	Folder	Apr 12
txt	3 items	Folder	Apr 12
BDMH supplement data.xlsx	4.3 kB	Spreadsheet	Apr 21
masterFile_shubhi	512 bytes	Text	Apr 22

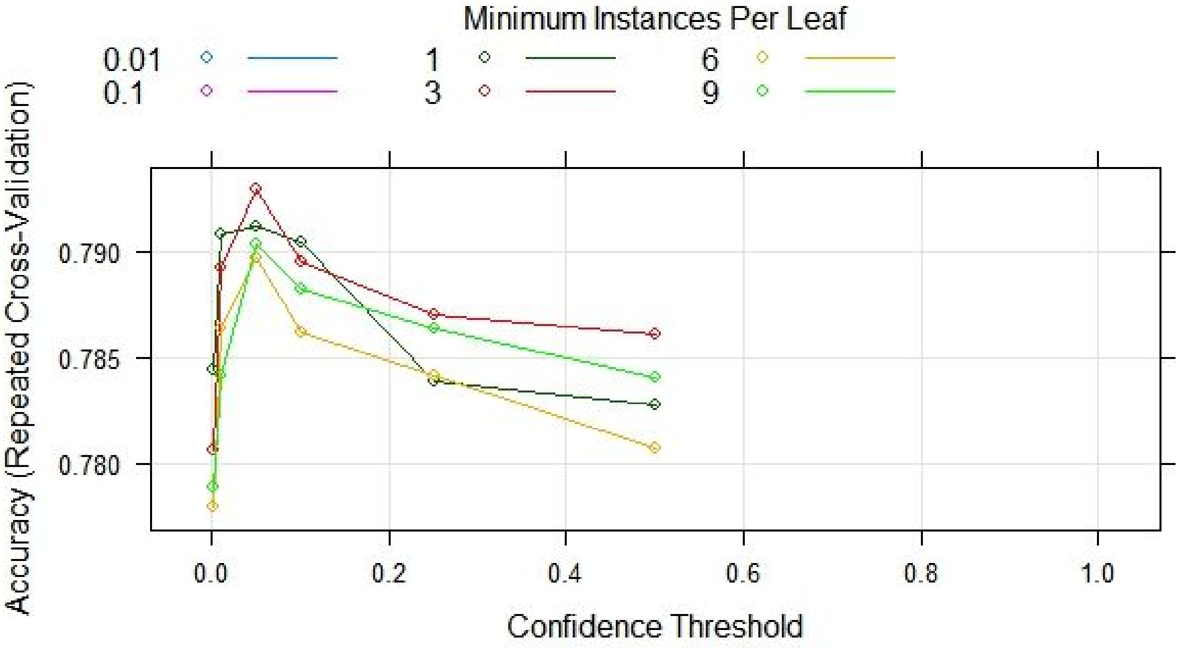
RESULTS

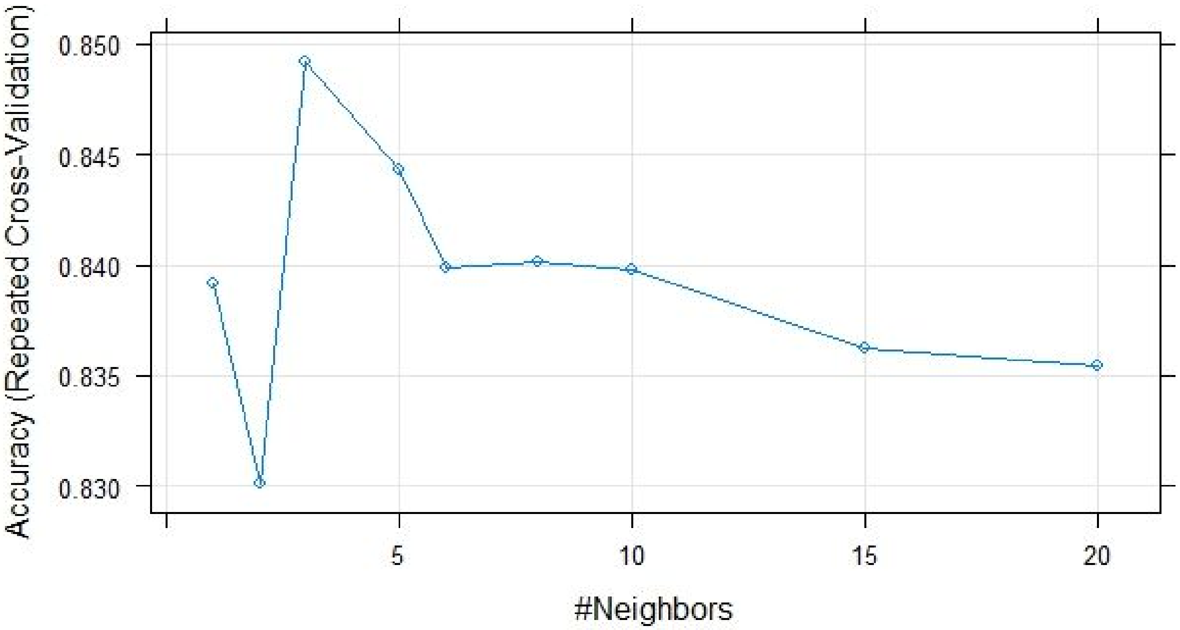
This section includes results obtained as follows:

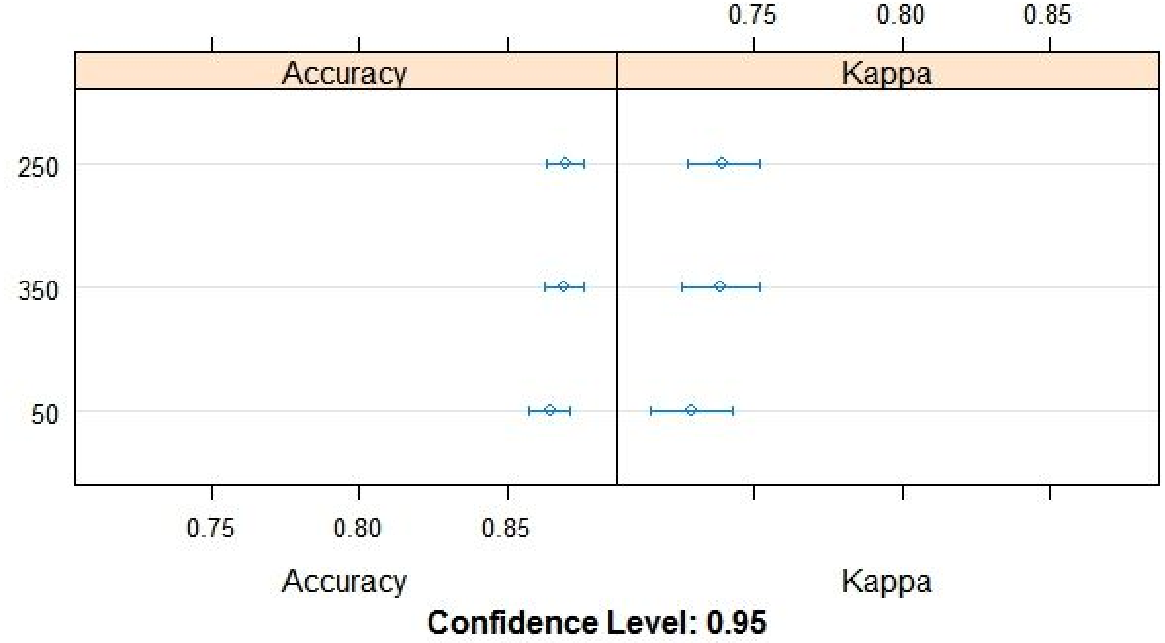
1. Parameter tuning of Models : SVM , Adaboost, J48, KNN, Random Forest






Minimum Instances Per Leaf
0.01
0.1
1
3
6
9
Accuracy (Repeated Cross-Validation)
0.790
0.785
0.780
0.0
0.2
0.4
0.6
0.8
1.0
Confidence Threshold

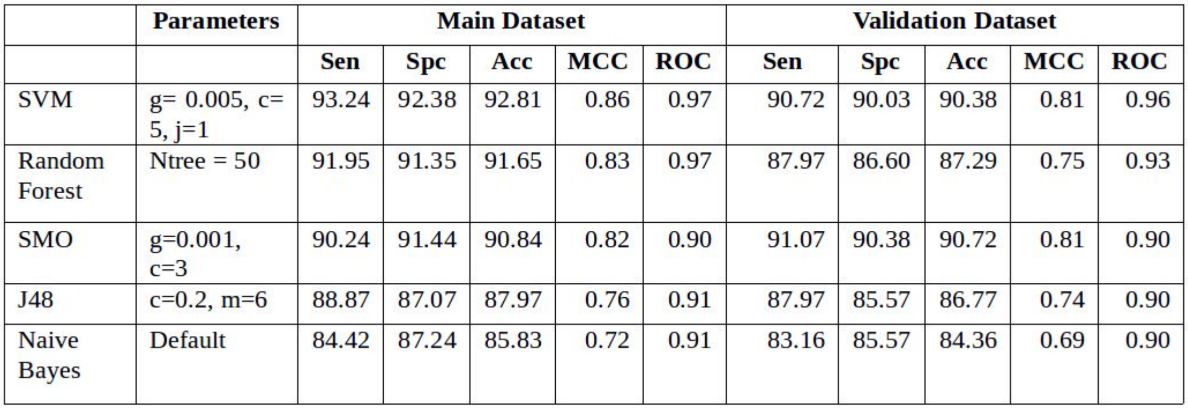

Accuracy (Repeated Cross-Validation)
0.850
0.845
0.840
0.835
0.830
5
10
15
20
#Neighbors



2. Models trained on Peptide Sequences

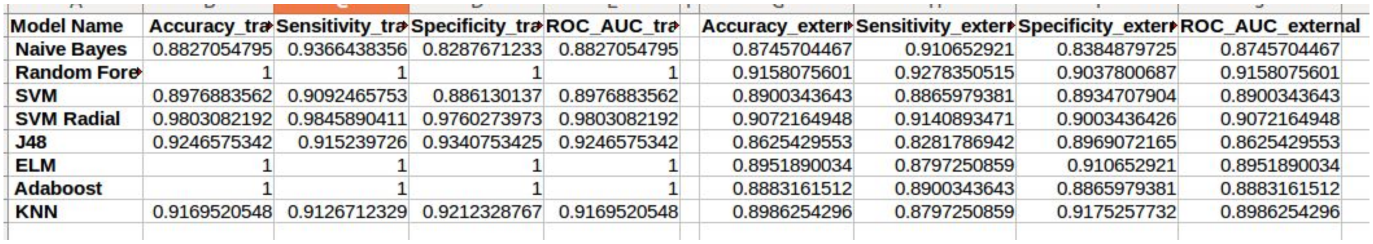
Data set : Antifp_DS2

Results in paper



	Parameters	Main Dataset					Validation Dataset				
		Sen	Spc	Acc	MCC	ROC	Sen	Spc	Acc	MCC	ROC
SVM	g= 0.005, c= 5, j=1	93.24	92.38	92.81	0.86	0.97	90.72	90.03	90.38	0.81	0.96
Random Forest	Ntree = 50	91.95	91.35	91.65	0.83	0.97	87.97	86.60	87.29	0.75	0.93
SMO	g=0.001, c=3	90.24	91.44	90.84	0.82	0.90	91.07	90.38	90.72	0.81	0.90
J48	c=0.2, m=6	88.87	87.07	87.97	0.76	0.91	87.97	85.57	86.77	0.74	0.90
Naive Bayes	Default	84.42	87.24	85.83	0.72	0.91	83.16	85.57	84.36	0.69	0.90

Results obtained by us



Model Name	Accuracy_tra	Sensitivity_tra	Specificity_tra	ROC_AUC_tra		Accuracy_extern	Sensitivity_extern	Specificity_extern	ROC_AUC_external
Naive Bayes	0.8827054795	0.9366438356	0.8287671233	0.8827054795		0.8745704467	0.910652921	0.8384879725	0.8745704467
Random Fore	1	1	1	1		0.9158075601	0.9278350515	0.9037800687	0.9158075601
SVM	0.8976883562	0.9092465753	0.886130137	0.8976883562		0.8900343643	0.8865979381	0.8934707904	0.8900343643
SVM Radial	0.9803082192	0.9845890411	0.9760273973	0.9803082192		0.9072164948	0.9140893471	0.9003436426	0.9072164948
J48	0.9246575342	0.915239726	0.9340753425	0.9246575342		0.8625429553	0.8281786942	0.8969072165	0.8625429553
ELM	1	1	1	1		0.8951890034	0.8797250859	0.910652921	0.8951890034
Adaboost	1	1	1	1		0.8883161512	0.8900343643	0.8865979381	0.8883161512
KNN	0.9169520548	0.9126712329	0.9212328767	0.9169520548		0.8986254296	0.8797250859	0.9175257732	0.8986254296

Ensemble Results

Accuracy - Training

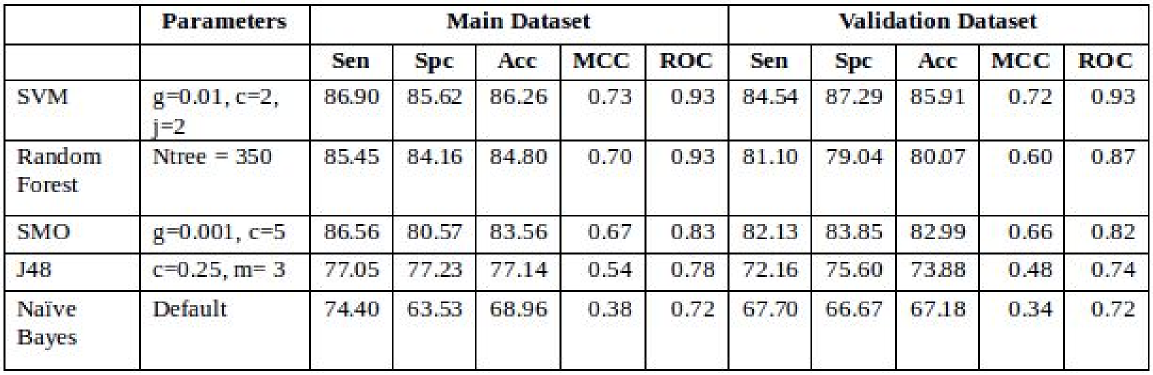
	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
svmRadial	0.89	0.90	0.91	0.91	0.92	0.93	
rf	0.90	0.91	0.92	0.92	0.92	0.95	
J48	0.85	0.86	0.87	0.87	0.88	0.91	
AdaBoost.M1	0.89	0.90	0.91	0.91	0.92	0.95	

Accuracy - Validation

SVM Radial	**RF**	**J48**	**AdaBoost.M1**	**Ensemble**
96.22	96.78	87.89	95.32	**96.58**

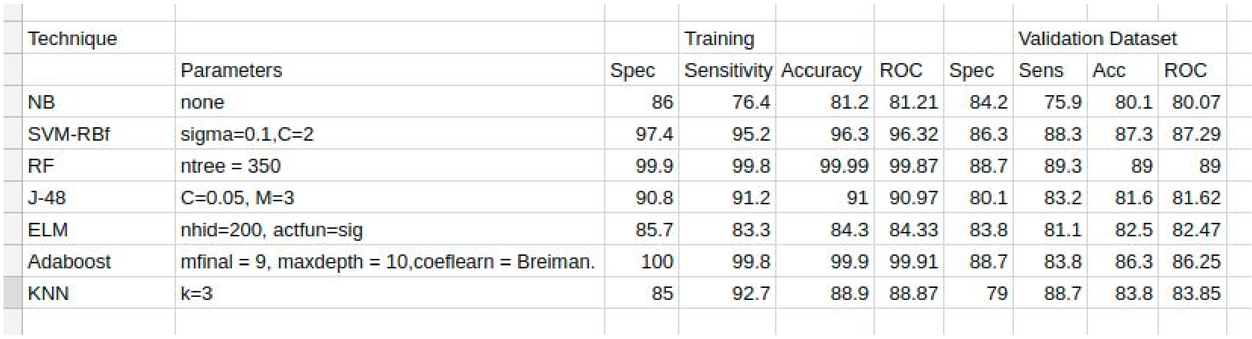
Data set : Antifp_Main

Results in paper



	Parameters	Main Dataset					Validation Dataset				
		Sen	Spc	Acc	MCC	ROC	Sen	Spc	Acc	MCC	ROC
SVM	g=0.01, c=2, j=2	86.90	85.62	86.26	0.73	0.93	84.54	87.29	85.91	0.72	0.93
Random Forest	Ntree = 350	85.45	84.16	84.80	0.70	0.93	81.10	79.04	80.07	0.60	0.87
SMO	g=0.001, c=5	86.56	80.57	83.56	0.67	0.83	82.13	83.85	82.99	0.66	0.82
J48	c=0.25, m= 3	77.05	77.23	77.14	0.54	0.78	72.16	75.60	73.88	0.48	0.74
Naïve Bayes	Default	74.40	63.53	68.96	0.38	0.72	67.70	66.67	67.18	0.34	0.72

Results obtained by us



Technique			Training				Validation Dataset		
	Parameters	Spec	Sensitivity	Accuracy	ROC	Spec	Sens	Acc	ROC
NB	none	86	76.4	81.2	81.21	84.2	75.9	80.1	80.07
SVM-RBf	sigma=0.1,C=2	97.4	95.2	96.3	96.32	86.3	88.3	87.3	87.29
RF	ntree = 350	99.9	99.8	99.99	99.87	88.7	89.3	89	89
J-48	C=0.05, M=3	90.8	91.2	91	90.97	80.1	83.2	81.6	81.62
ELM	nhid=200, actfun=sig	85.7	83.3	84.3	84.33	83.8	81.1	82.5	82.47
Adaboost	mfinal = 9, maxdepth = 10,coeflearn = Breiman.	100	99.8	99.9	99.91	88.7	83.8	86.3	86.25
KNN	k=3	85	92.7	88.9	88.87	79	88.7	83.8	83.85

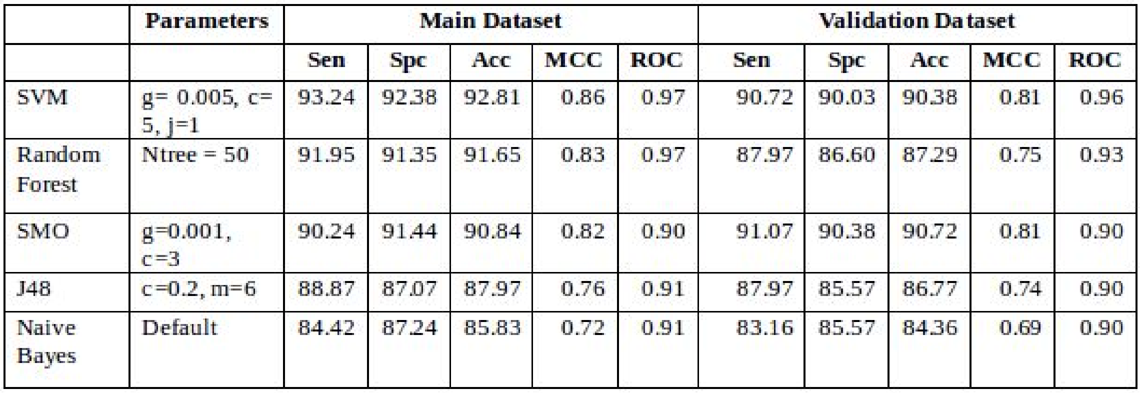
Ensemble Results

Accuracy

```
             Min.     1st Qu.  Median    Mean    3rd Qu.    Max. NA's
svmRadial   0.828    0.843    0.856     0.855   0.86       0.890
rf          0.852    0.869    0.88      0.8781  0.888      0.9014
J48         0.783    0.817    0.829     0.829   0.841      0.86
AdaBoost.M1 0.826    0.852    0.867     0.86    0.871      0.89
```

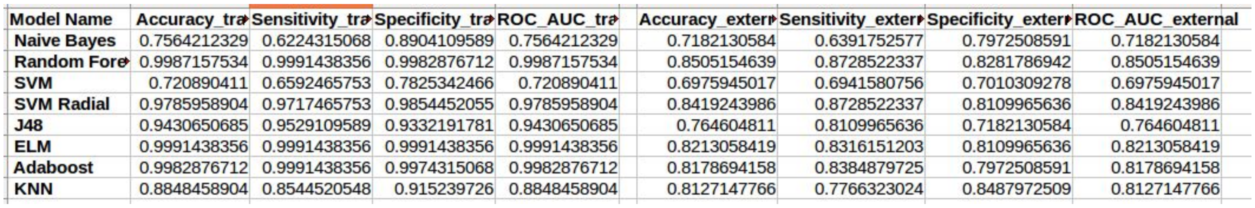
Data set : Antifp_DS1

Results in paper



	Parameters	Main Dataset					Validation Dataset				
		Sen	Spc	Acc	MCC	ROC	Sen	Spc	Acc	MCC	ROC
SVM	g= 0.005, c= 5, j=1	93.24	92.38	92.81	0.86	0.97	90.72	90.03	90.38	0.81	0.96
Random Forest	Ntree = 50	91.95	91.35	91.65	0.83	0.97	87.97	86.60	87.29	0.75	0.93
SMO	g=0.001, c=3	90.24	91.44	90.84	0.82	0.90	91.07	90.38	90.72	0.81	0.90
J48	c=0.2, m=6	88.87	87.07	87.97	0.76	0.91	87.97	85.57	86.77	0.74	0.90
Naive Bayes	Default	84.42	87.24	85.83	0.72	0.91	83.16	85.57	84.36	0.69	0.90

Results obtained by us



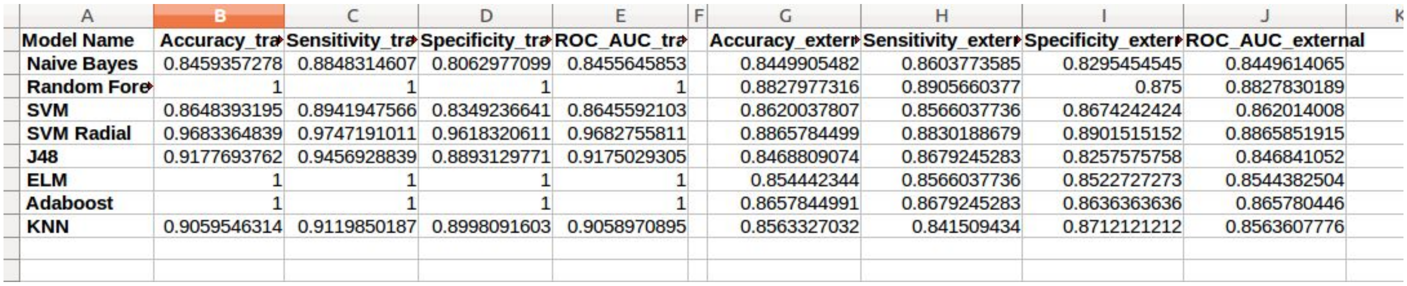
Model Name	Accuracy_tra	Sensitivity_tra	Specificity_tra	ROC_AUC_tra	Accuracy_extern	Sensitivity_extern	Specificity_extern	ROC_AUC_external
Naive Bayes	0.7564212329	0.6224315068	0.8904109589	0.7564212329	0.7182130584	0.6391752577	0.7972508591	0.7182130584
Random Fore	0.9987157534	0.9991438356	0.9982876712	0.9987157534	0.8505154639	0.8728522337	0.8281786942	0.8505154639
SVM	0.720890411	0.6592465753	0.7825342466	0.720890411	0.6975945017	0.6941580756	0.7010309278	0.6975945017
SVM Radial	0.9785958904	0.9717465753	0.9854452055	0.9785958904	0.8419243986	0.8728522337	0.8109965636	0.8419243986
J48	0.9430650685	0.9529109589	0.9332191781	0.9430650685	0.764604811	0.8109965636	0.7182130584	0.764604811
ELM	0.9991438356	0.9991438356	0.9991438356	0.9991438356	0.8213058419	0.8316151203	0.8109965636	0.8213058419
Adaboost	0.9982876712	0.9991438356	0.9974315068	0.9982876712	0.8178694158	0.8384879725	0.7972508591	0.8178694158
KNN	0.8848458904	0.8544520548	0.915239726	0.8848458904	0.8127147766	0.7766323024	0.8487972509	0.8127147766

3. Models trained on Split Composition

Data set : Antifp_DS2

Features : n15 c15

Results

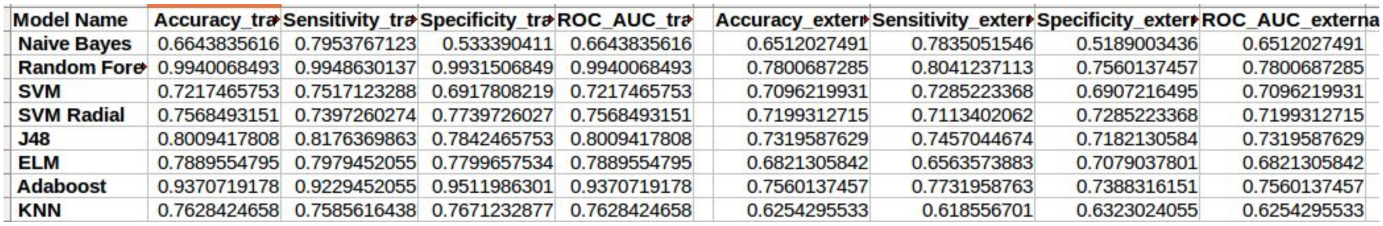


A	B	C	D	E	F	G	H	I	J	K
Model Name	Accuracy_tra	Sensitivity_tra	Specificity_tra	ROC_AUC_tra		Accuracy_extern	Sensitivity_extern	Specificity_extern	ROC_AUC_external	
Naive Bayes	0.8459357278	0.8848314607	0.8062977099	0.8455645853		0.8449905482	0.8603773585	0.8295454545	0.8449614065	
Random Fore	1	1	1	1		0.8827977316	0.8905660377	0.875	0.8827830189	
SVM	0.8648393195	0.8941947566	0.8349236641	0.8645592103		0.8620037807	0.8566037736	0.8674242424	0.862014008	
SVM Radial	0.9683364839	0.9747191011	0.9618320611	0.9682755811		0.8865784499	0.8830188679	0.8901515152	0.8865851915	
J48	0.9177693762	0.9456928839	0.8893129771	0.9175029305		0.8468809074	0.8679245283	0.8257575758	0.846841052	
ELM	1	1	1	1		0.854442344	0.8566037736	0.8522727273	0.8544382504	
Adaboost	1	1	1	1		0.8657844991	0.8679245283	0.8636363636	0.865780446	
KNN	0.9059546314	0.9119850187	0.8998091603	0.9058970895		0.8563327032	0.841509434	0.8712121212	0.8563607776	

4. Models Trained on Physicochemical properties

Data set : Antifp_DS3

Results

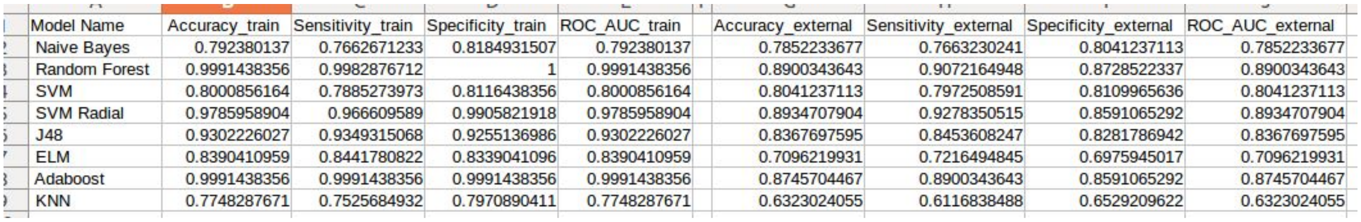


Model Name	Accuracy_tra	Sensitivity_tra	Specificity_tra	ROC_AUC_tra	Accuracy_extern	Sensitivity_extern	Specificity_extern	ROC_AUC_externa
Naive Bayes	0.6643835616	0.7953767123	0.533390411	0.6643835616	0.6512027491	0.7835051546	0.5189003436	0.6512027491
Random Fore	0.9940068493	0.9948630137	0.9931506849	0.9940068493	0.7800687285	0.8041237113	0.7560137457	0.7800687285
SVM	0.7217465753	0.7517123288	0.6917808219	0.7217465753	0.7096219931	0.7285223368	0.6907216495	0.7096219931
SVM Radial	0.7568493151	0.7397260274	0.7739726027	0.7568493151	0.7199312715	0.7113402062	0.7285223368	0.7199312715
J48	0.8009417808	0.8176369863	0.7842465753	0.8009417808	0.7319587629	0.7457044674	0.7182130584	0.7319587629
ELM	0.7889554795	0.7979452055	0.7799657534	0.7889554795	0.6821305842	0.6563573883	0.7079037801	0.6821305842
Adaboost	0.9370719178	0.9229452055	0.9511986301	0.9370719178	0.7560137457	0.7731958763	0.7388316151	0.7560137457
KNN	0.7628424658	0.7585616438	0.7671232877	0.7628424658	0.6254295533	0.618556701	0.6323024055	0.6254295533

5. Models trained on Amino and physicochemical properties

Data set : Antifp_Main

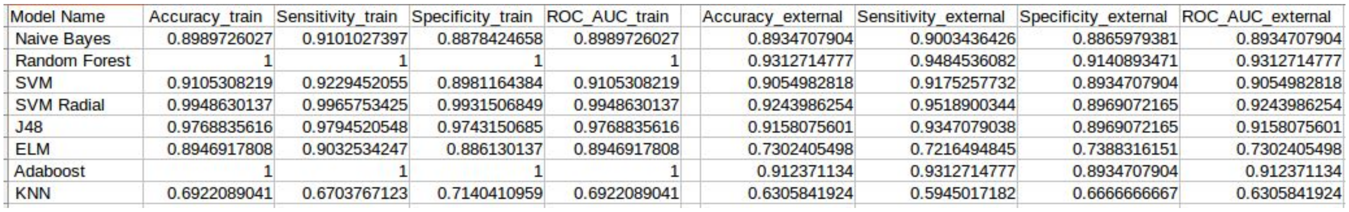
Results



Model Name	Accuracy_train	Sensitivity_train	Specificity_train	ROC_AUC_train	Accuracy_external	Sensitivity_external	Specificity_external	ROC_AUC_external
Naive Bayes	0.792380137	0.7662671233	0.8184931507	0.792380137	0.7852233677	0.7663230241	0.8041237113	0.7852233677
Random Forest	0.9991438356	0.9982876712	1	0.9991438356	0.8900343643	0.9072164948	0.8728522337	0.8900343643
SVM	0.8000856164	0.7885273973	0.8116438356	0.8000856164	0.8041237113	0.7972508591	0.8109965636	0.8041237113
SVM Radial	0.9785958904	0.966609589	0.9905821918	0.9785958904	0.8934707904	0.9278350515	0.8591065292	0.8934707904
J48	0.9302226027	0.9349315068	0.9255136986	0.9302226027	0.8367697595	0.8453608247	0.8281786942	0.8367697595
ELM	0.8390410959	0.8441780822	0.8339041096	0.8390410959	0.7096219931	0.7216494845	0.6975945017	0.7096219931
Adaboost	0.9991438356	0.9991438356	0.9991438356	0.9991438356	0.8745704467	0.8900343643	0.8591065292	0.8745704467
KNN	0.7748287671	0.7525684932	0.7970890411	0.7748287671	0.6323024055	0.6116838488	0.6529209622	0.6323024055

Data set : Antifp_DS2

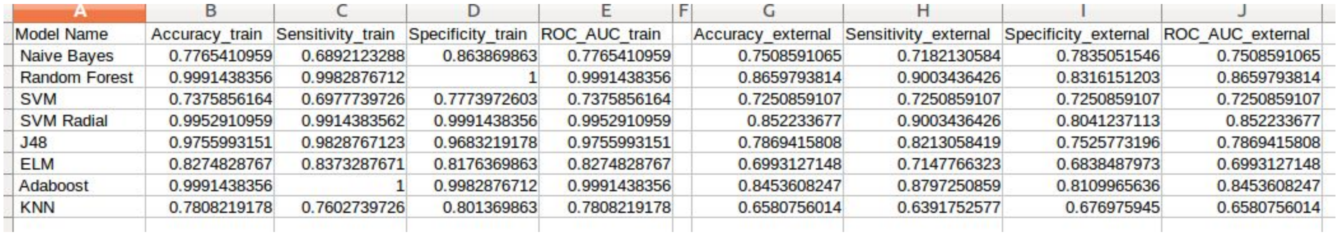
Results



Model Name	Accuracy_train	Sensitivity_train	Specificity_train	ROC_AUC_train	Accuracy_external	Sensitivity_external	Specificity_external	ROC_AUC_external
Naive Bayes	0.8989726027	0.9101027397	0.8878424658	0.8989726027	0.8934707904	0.9003436426	0.8865979381	0.8934707904
Random Forest	1	1	1	1	0.9312714777	0.9484536082	0.9140893471	0.9312714777
SVM	0.9105308219	0.9229452055	0.8981164384	0.9105308219	0.9054982818	0.9175257732	0.8934707904	0.9054982818
SVM Radial	0.9948630137	0.9965753425	0.9931506849	0.9948630137	0.9243986254	0.9518900344	0.8969072165	0.9243986254
J48	0.9768835616	0.9794520548	0.9743150685	0.9768835616	0.9158075601	0.9347079038	0.8969072165	0.9158075601
ELM	0.8946917808	0.9032534247	0.886130137	0.8946917808	0.7302405498	0.7216494845	0.7388316151	0.7302405498
Adaboost	1	1	1	1	0.912371134	0.9312714777	0.8934707904	0.912371134
KNN	0.6922089041	0.6703767123	0.7140410959	0.6922089041	0.6305841924	0.5945017182	0.6666666667	0.6305841924

Data set : Antifp_DS1

Results



Model Name	Accuracy_train	Sensitivity_train	Specificity_train	ROC_AUC_train	Accuracy_external	Sensitivity_external	Specificity_external	ROC_AUC_external
Naive Bayes	0.7765410959	0.6892123288	0.863869863	0.7765410959	0.7508591065	0.7182130584	0.7835051546	0.7508591065
Random Forest	0.9991438356	0.9982876712	1	0.9991438356	0.8659793814	0.9003436426	0.8316151203	0.8659793814
SVM	0.7375856164	0.6977739726	0.7773972603	0.7375856164	0.7250859107	0.7250859107	0.7250859107	0.7250859107
SVM Radial	0.9952910959	0.9914383562	0.9991438356	0.9952910959	0.852233677	0.9003436426	0.8041237113	0.852233677
J48	0.9755993151	0.9828767123	0.9683219178	0.9755993151	0.7869415808	0.8213058419	0.7525773196	0.7869415808
ELM	0.8274828767	0.8373287671	0.8176369863	0.8274828767	0.6993127148	0.7147766323	0.6838487973	0.6993127148
Adaboost	0.9991438356	1	0.9982876712	0.9991438356	0.8453608247	0.8797250859	0.8109965636	0.8453608247
KNN	0.7808219178	0.7602739726	0.801369863	0.7808219178	0.6580756014	0.6391752577	0.676975945	0.6580756014

CONCLUSION AND DISCUSSION

The in silico method developed can in advance predict whether a peptide sequence can be AFP or not ,would definitely help experimental biologists for a speedy screening of AFPs before synthesis and thus, fasten the AFP based research. Development of a computational method for AFP prediction is challenging due to various reasons since (i) AFPs have a lot of flexibility in size (4–100 amino acids) and fixed length pattern is required as input by machine learning methods to develop a model (ii) due to lack of experimentally validated AFPs.

In order to discriminate AFPs from non-AFPs with higher precision, we have developed machine learning models (SVM, Random Forest, Naive Bayes, KNN, Adaboost, J48 and Ensemble) based on features like amino acid composition, dipeptide composition, amino acid composition along with mass, charge and pI value, binary profile, N and C-terminal residue hybrid. The performance of the models developed was found to be quite impressive when features like amino acid composition, amino acid composition along with mass, charge, and pI value and dipeptide composition were used as input.

Discriminating two sequences with high identity but different activity is a challenging task for most of the prediction methods. To address this issue, we calculated the euclidean distance between our positive and negative peptides and selected the negative peptides with minimum distance. We tested the performance of our composition based model as well as N15C15 binary profile based model and observed that composition model didn’t perform well in discriminating two sequences very accurately. However, our binary profile based model was able to discriminate the two sequences with good accuracy, suggesting that binary profile feature can be used in discriminating such sequences where sequences are very similar to each other but possess different activity.

REFERENCES

1. https://www.frontiersin.org/articles/10.3389/fmicb.2018.00323/full
2. http://webs.iiitd.edu.in/raghava/antifp/
3. https://www.analyticsvidhya.com/blog/2016/12/practical-guide-to-implement-machine-learning-with-caret-package-in-r-with-practice-problem/
4. http://topepo.github.io/caret/train-models-by-tag.html
5. https://www.analyticsvidhya.com/blog/2015/12/improve-machine-learning-results/
6. https://blog.statsbot.co/ensemble-learning-d1dcd548e936