Protein function prediction: Some lessons for classifier performance evaluation

WONG Limsoon

Plan, 5/1/2026

Classifier performance

Definitions

Interpretation & correct usage

Test sets

Purpose & design

Protein function prediction

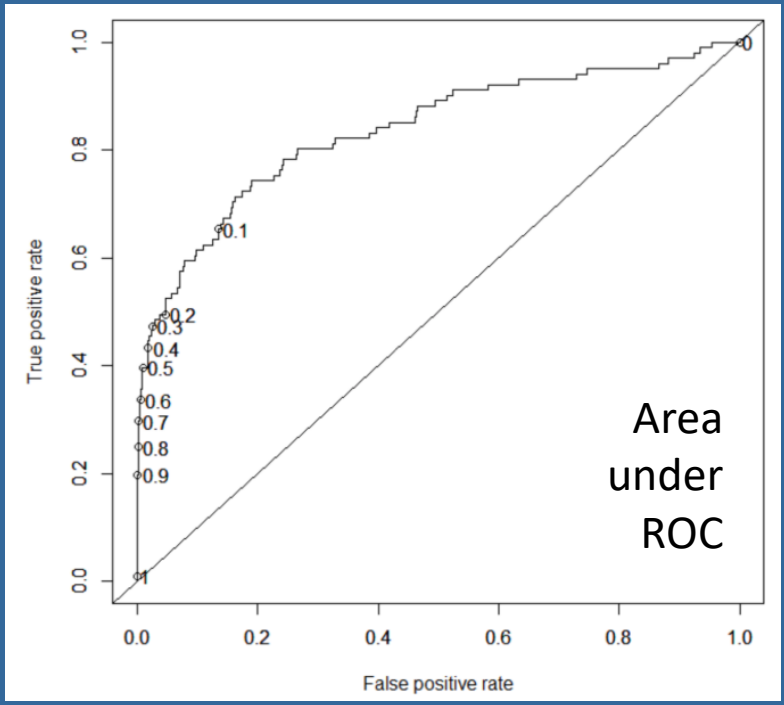
Guilt by association

Deep learning

Illuminating the twilight zone of protein function prediction assessment

Illuminating the twilight zone of protein function prediction

Classifier performance measures



True positives (TP): actuals are positives and are predicted as positives. *You predicted that a woman is pregnant and she actually is.*	**False positives (FP)** - Type 1 Error: actuals are negatives and are predicted as positives. *You predicted that a man is pregnant but he actually is not.*
False negatives (FN) - Type 2 Error: actuals are positives and are predicted as negatives. *You predicted that a woman is not pregnant but she actually is.*	**True negatives (TN):** actuals are negatives and are predicted as positives. *You predicted that a man is not pregnant and he actually is not.*

Accuracy:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

Recall:

$$Recall = \frac{TP}{TP + FN}$$

Precision:

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

F_1 score:

$$F_1 = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$$

Image credit: Clare Liu's post on KDnuggets, 20/9/2022

A common practice in machine learning

Optimize and evaluate based on cross-validation accuracy

Is this sound?

Exercise

You have a classifier. On a test set having 20% +ve and 80% -ve cases, the classifier's recall and precision are both 80%

Suppose you test it on a new test set having 80% +ve and 20% -ve cases. What do you expect its accuracy to be?

You may assume that the +ve (resp. –ve) cases in both test sets are equally sufficiently representative of the +ve (resp. –ve) real-world population

Calculations for the two scenarios

Given

- +ve : -ve in test set = 200 : 800
- Recall = 80%, precision = 80%

Thus

- TP = 200 * recall = 160
- FP + TP = TP / precision = 200
- TN = 800 – FP = 800 – (200 – 160) = 760
- Specificity = TN / 800 = 95%
- Accuracy = (TP + TN) / (200 + 800) = 92%

Given

- +ve : -ve in test set = 800 : 200

Thus

- Recall = …, specificity = …
- TP = 800 * recall = …
- TN = 200 * specificity = …
- FP = 200 – TN = …
- Precision = TP / (TP + FP) = …
- Accuracy = (TP + TN) / (800 + 200) = …

Class proportion of test set may not be fidel to real life

Accuracy determined from test set may not give the right picture of real-life performance

Test set:

20% +ve, 80% -ve

recall = 80%, precision = 80%

∴specificity = 95%, accuracy = 92%

New test set “real life”:

80% +ve, 20% -ve

By “representativeness”,

recall = …, specificity = …

∴ accuracy = …, precision = …

Exercise

Accuracy & precision are not robust classifier performance measures as they are sensitive to changes in class proportions in test sets

Is F1 more robust to this problem?

KDnuggets Blog Topics Datasets Education Resources

F-Measure provides a single score that balances both the concerns of precision and recall in one number. A good F1 score means that you have low false positives and low false negatives, so you're correctly identifying real threats, and you are not disturbed by false alarms. An F1 score is considered perfect when it's 1, while the model is a total failure when it's 0.

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

Image credit: Clare Liu's post on KDnuggets, 20/9/2022

Exercise

You have a classifier

On a test set having 90% +ve and 10% -ve cases, the classifier's precision is 99% and recall is 90%

On a test set having 10% +ve and 90% -ve cases, the classifier's precision is 80% and recall is 90%

Do you think anything is wrong? Why?

Calculations for the two scenarios

Given

- +ve : -ve = 900 : 100
- Recall = 90%, precision = 99%

Thus

- TP = 900 * recall = 810
- FP / (TP + FP) = (1 – precision) = 1%

$\Rightarrow$ FP = 810 * 1% + FP * 1%

$\Rightarrow$ FP * 99% = 8

$\Rightarrow$ FP = 8

- TN = 100 – FP = 92
- Specificity = TN / 100 = 92%

Given

- +ve : -ve = 100 : 900
- Recall = 90%,

Assume (if nothing is wrong)

- Specificity = ...

Thus

- TP = 100 * recall = 90
- FP = 900 * (1 – specificity) = ...
- Precision = TP / (TP + FP) = ...

Accuracy from a test set must be calibrated for interpretability

Also, it is probably better to optimize wrt sqrt(recall * specificity), as these are independent of class proportion

A common practice in machine learning

Optimize and evaluate based on a test set without considering the properties of the test set, in particular, without checking whether the test set is well designed

Is this sound?

How do serious professors set final exam?

This is how I set exams

Some easy questions

Enough hard questions

And often some surprising questions

Students don't get "A" answering easy questions

Plan, 5/1/2026

Classifier performance

Definitions

Interpretation & correct usage

Test sets

Purpose & design

Protein function prediction

Guilt by association

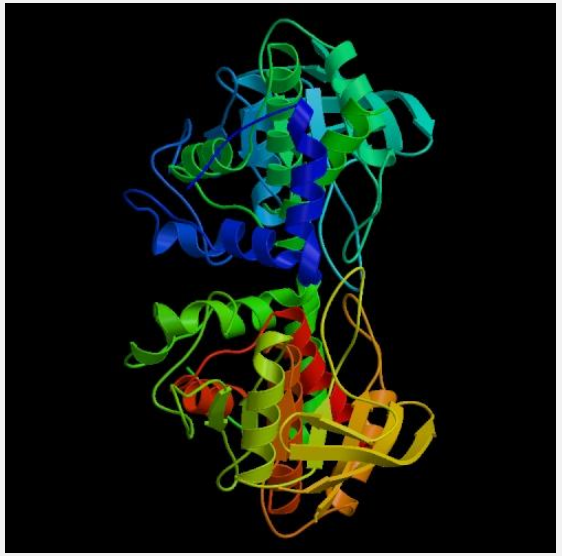
Deep learning

Illuminating the twilight zone of protein function prediction assessment

Illuminating the twilight zone of protein function prediction

Protein function assignment

A protein is a large complex molecule made up of one or more chains of amino acids



Usually, only the sequence of amino acid is known

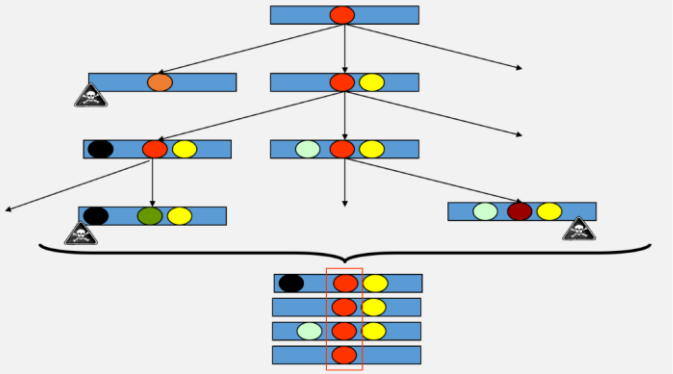
```
SPSTNRKYPPLPVDKLEEEINRRMADDNKLFREEFNALPACPIQATCEAASKEENKEKNR
YVNILPYDHSRVHLTPVEGVPDSDYINASFINGYQEKNKFIAAQGPKEETVNDFWRMIWE
QNTATIVMVTNLKERKECKCAQYWPDQGCWTYGNVRVSVEDVTVLVDYTVRKFCIQQVGD
VTNRKPQRLITQFHFTSWPDFGVPFTPIGMLKFLKKVKACNPQYAGAIVVHCSAGVGRTG
TFVVIDAMLDMMHSERKVDVYGFVSRIRAQRCQMVQTDMQYVFIYQALLEHYLYGDTELE
VT
```

Proteins perform a wide variety of activities in the cell

How do we predict the function of a protein?

A standard postulate based on evolution

In the course of evolution…



Evolution takes time …

Let a = AFPHQHRVP
Let b = PQVYNIMKE

Suppose each generation differs from the previous by 1 residue

What is the max difference between the 2nd generation of a

What is the min difference between the 2nd generation of a and b?

Two proteins (not) inheriting their function from a common ancestor (do not) have similar amino acid sequences

Guilt by association

Compare *T* with seqs of known function in a db

Poor Sequence Alignment

- Poor seq alignment shows few matched positions

⇒ The two proteins are not likely to be homologous

Alignment by FASTA of the sequences of amicyanin and domain 1 of ascorbate oxidase

```
                    60        70        80        90       100
Amicyanin           MPHNVHFVAGVLGEAALKGPMMKKEQAYSLFFTEAGTYDYHCTPHPFMRGKVVV
                                                :.:  . ::. ::
Ascorbate Oxidase   ILQRGTPWADGTASISQCAINPGETFFYNFTVDNPGTFFYHGHLGMQRSAGLYG
                      70        80        90       100       110
```

No obvious match between
Amicyanin and Ascorbate Oxidase

Discard this function as a candidate

Good Sequence Alignment

- Good alignment usually has clusters of extensive matched positions

⇒ The two proteins are likely to be homologous

```
>gi|13476732|ref|NP_108301.1|  unknown protein [Mesorhizobium loti]
gi|14027493|dbj|BAB53762.1|  unknown protein [Mesorhizobium loti]
          Length = 105

 Score =  105 bits (262), Expect = 1e-22
 Identities = 61/106 (57%), Positives = 73/106 (68%), Gaps = 1/106 (0%)

Query: 1   MKPGRLASIALAIIFLPWAVPAHAATIEITMENLVISPTEVSAKVGDTIRWVNKDVFAHT 60
           MK G L  ++       MA PA AATIE+T++ LV SP  V AKVGDTI WVN DV AHT
Sbjct: 1   MKAGALIRLSVLAALALMAAPAAAATIEVTIDKLVFSPATVEAKVGDTIEWVNNDVVAHT 60
```

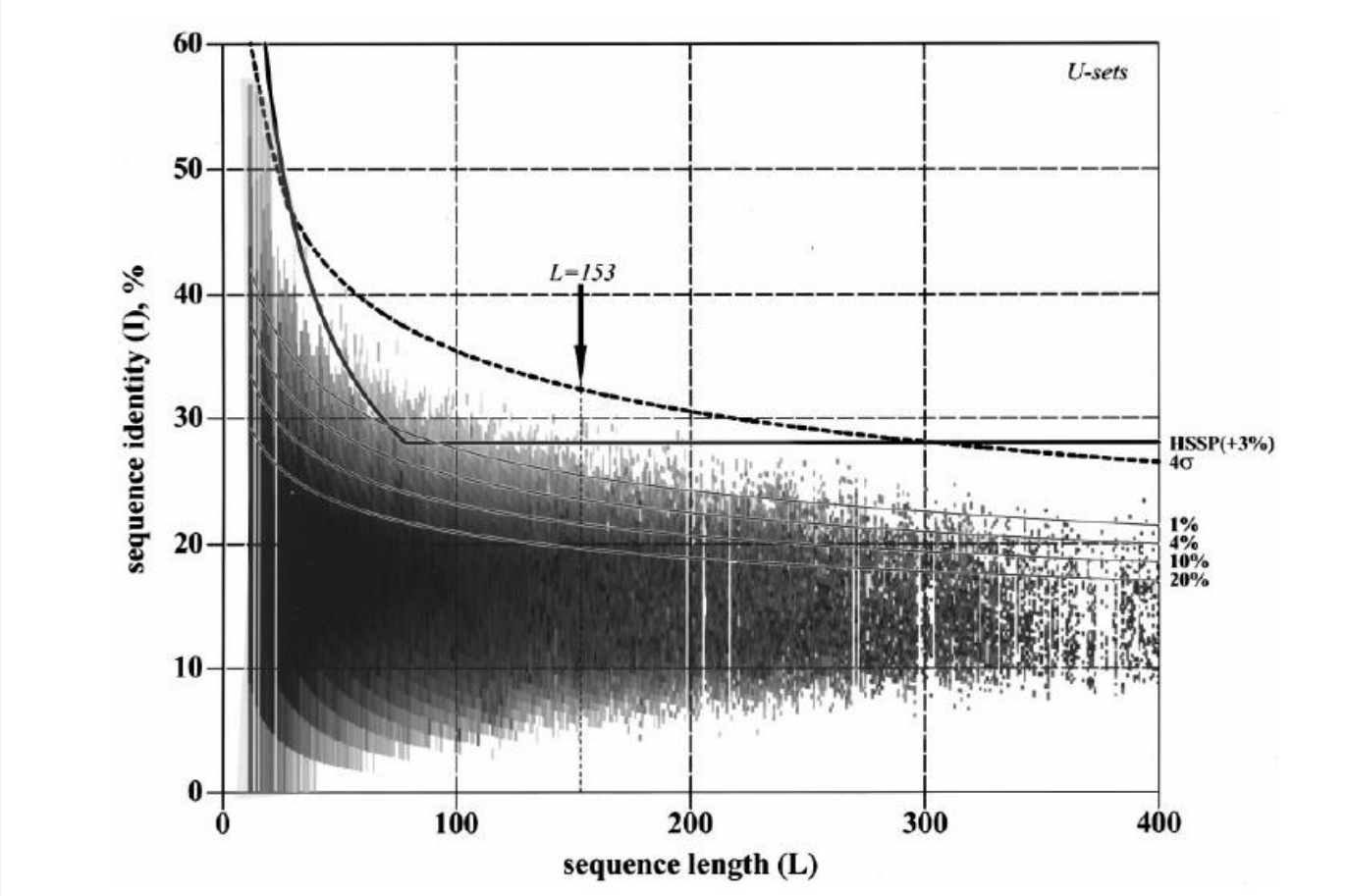
good match between
Amicyanin and unknown M. loti protein

Assign to *T* same function as homologs

Confirm with suitable wet experiments

Twilight zone: Limit of sequence similarity-based protein function assignment

So, need clever methods for the twilight zone



Abagyan RA, Batalov S. *J Mol Biol.,* 273(1):355-68, 1997

Many deep learning models to the rescue?

DeepFam (2018, CNN)

DeepGO (2018, CNN + DNN)

DeepPred (2019, hierarchical DNN)

DeepGoPlus (2019, CNN + DNN)

UDSMProt (2020, LSTM)

MultiPredGO (2020, multimodal DL)

TALE+ (2021, transformer)

DeepGraphGO (2021, CNN + DNN, multimodal)

DeepGoZero (2022, zero-shot learning), …

DeepFam, deep learning for protein family prediction

This looks good

Really?

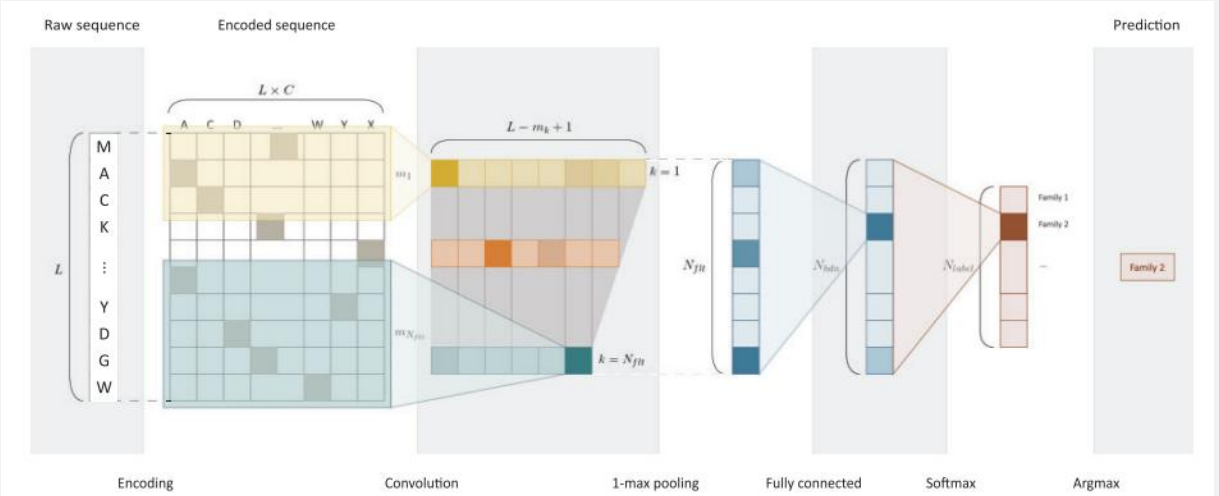


Fig. 1. The overview of DeepFam model. It is a feedforward convolution neural network whose last layer represents the probabilities of each family. convolution layer and 1-max pooling layer calculate a score (activation) of the existence of a conserved regions. The next layer is fully-connected neural network which can detect longer or complex sites. In order to infer the probability of each family, the last layer is designed as softmax layer (multinomial logistic regression), generally used for multi-class classification

Table 2. Prediction accuracy (%) comparison of COG dataset

Dataset	COG-500-1074	COG-250-1796	COG-100-2892
DeepFam	**95.40**	**94.08**	91.40
pHMM	91.75	91.78	**91.67**
3-mer LR	85.59	81.15	75.44
Protvec LR	47.34	41.76	37.05

Bold indicates the best performance for each dataset.

Seo et al., *Bioinformatics,* 34(13):i254-i262, 2018

Plan, 5/1/2026

Classifier performance

Definitions

Interpretation & correct usage

Test sets

Purpose & design

Protein function prediction

Guilt by association

Deep learning

Illuminating the twilight zone of protein function prediction assessment

Illuminating the twilight zone of protein function prediction

How do I set final exams?

This is how I set exams

Some easy questions

Enough hard questions

And often some surprising questions

Students don't get "A" answering easy questions

Table 2. Prediction accuracy (%) comparison of COG dataset

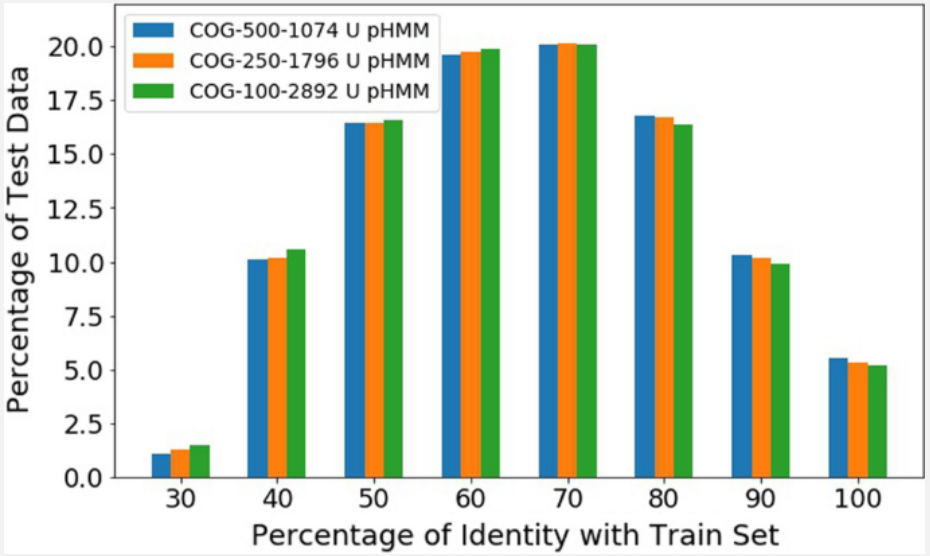
Dataset	COG-500-1074	COG-250-1796	COG-100-2892
DeepFam	**95.40**	**94.08**	91.40
pHMM	91.75	91.78	**91.67**
3-mer LR	85.59	81.15	75.44
Protvec LR	47.34	41.76	37.05

Bold indicates the best performance for each dataset.

Is this a good assessment of DeepFam? Do the test sets have enough hard questions and surprising questions?

What can be used as hard questions?
What can be used as surprising questions?

DeepFam's good accuracy is largely due to "easy" proteins



Dataset	Method	predCount = 1	predCount = 2	predCount = 3	predCount = 4	predCount = 5	predCount > 5
Identity: $0 < x \leq 30$							
COG-500-1074	EnsembleFam	**72.07**	**81.00**	**82.82**	**84.96**	**85.33**	**85.27**
	pHMM	69.54	73.75	55.51	70.62	70.85	73.55
	DeepFam	57.14	54.52	49.90	46.92	43.64	35.94
COG-250-1796	EnsembleFam	72.84	**77.07**	**81.02**	**82.14**	**84.66**	**86.45**
	pHMM	**75.39**	73.82	73.84	71.02	67.44	72.43
	DeepFam	32.44	32.54	30.24	29.53	30.02	28.68
COG-100-2892	EnsembleFam	**75.24**	**79.55**	**81.21**	**80.63**	**82.05**	**88.95**
	pHMM	63.44	59.69	53.45	48.16	47.42	57.57
	DeepFam	27.30	26.13	25.54	27.62	24.83	25.36

Neamul Kabir & Wong, *BMC Bioinformatics,* 23:90, 2022

There are few twilight zone proteins in the reference database. DeepFam's poor twilight zone performance is thus ok

What do you think?

Don’t be fooled by high accuracy on easy test sets

Need to stratify accuracy wrt easy and hard test instances

Do the test sets contain surprising questions?

How do DeepFam perform on these?

What do you think?

Don’t be fooled by high accuracy on test sets without surprises

Real world is full of surprises!

Plan, 26/1/2026

Classifier performance

Definitions

Interpretation & correct usage

Test sets

Purpose & design

Protein function prediction

Guilt by association

Deep learning

Illuminating the twilight zone of protein function prediction assessment

Illuminating the twilight zone of protein function prediction

Take-home message

Prediction model assessment needs careful test set design

Representativeness of test set

Easy questions, hard questions, & surprise questions in test set

Extrapolation of test metrics to real-world

Real-world utility considerations

References

Seokjun Seo, et al., “DeepFam: Deep learning based alignment-free method for protein family modeling and prediction”, *Bioinformatics*, 34(13):i254-i262, 2018

Neamul Kabir & Wong, “EnsembleFam: Towards more accurate protein family prediction in the twilight zone”, *BMC Bioinformatics*, 23:90, 2022

Homework, due 10/1/2026

submit by email to dcswls@nus.edu.sg

Read this paper, focusing on the way it evaluated the performance of the proposed classifier:

Yu et al., "Accurate prediction and key protein sequence feature identification of cyclins", Briefings in Functional Genomics, 22:411-419, 2023

Read the review report (attached below) produced by ChatGPT on the performance evaluation done by this paper.

Answer the following questions:

1/ Which parts of the ChatGPT report, if any, do you agree with?

2/ Which parts of the ChatGPT report, if any, do you disagree with? Why?

3/ Are there any major flaws that you think should be pointed out but are missed by ChatGPT?

Make 5-minutes presentation to class on 12/1/2026

Review Report (Evaluation Focus)

This paper proposes a cyclin/non-cyclin classifier based on ACC-PSSM feature representation, MRMD-based feature selection, and a random forest model. My review focuses exclusively on the **evaluation methodology** used to assess the proposed classifier.

The paper adopts **10-fold cross-validation** as the sole evaluation framework. This is a standard choice for small datasets and is appropriate in principle, but the paper does not clarify whether folds were **stratified**, nor whether feature selection (MRMD1.0 and MRMD2.0) was performed **inside each training fold** or once globally before cross-validation. If feature selection used the full dataset prior to CV, then the reported accuracies (up to 98.2%) may be **optimistically biased** due to information leakage. A clarification on this point is essential, especially given the modest dataset size (333 sequences).

The performance metrics reported—ACC, SN, SP, MCC and AUC—are appropriate for a binary classification task with roughly balanced classes. The inclusion of **MCC** and **AUC** is commendable, as they provide robustness beyond accuracy alone. However, the paper does not report **variance estimates** (e.g., per-fold standard deviations), making it difficult to judge the stability of performance across folds.

The model comparison section evaluates RF against SVM, KNN, NB, AdaBoost and J48 using the same 10-fold CV setup. This provides a useful baseline comparison, although hyperparameter tuning is not described for any model. Without systematic tuning (and without a validation framework separate from test folds), the comparisons may not reflect each model's actual capability. The ROC curves shown are informative, but again appear to be averaged or pooled without details on computation.

The feature-selection evaluation (MRMD1.0 and MRMD2.0) is assessed exclusively within the same CV framework. The results suggest that MRMD1.0 with ACC-PSSM improves accuracy and that MRMD2.0 produces a seven-feature subset with strong performance. However, because there is **no independent test set**, it is difficult to know how these selected features generalize. The reliance on t-SNE plots for qualitative validation is not equivalent to rigorous performance testing and can be misleading due to the stochastic and non-metric nature of t-SNE.

The comparison with three previous studies is useful but limited. Since all methods are evaluated on the same dataset, the absence of an **external validation set** means the improvement may partly reflect model tuning to a widely reused benchmark rather than true generalization.

Overall, while the evaluation includes appropriate metrics and reasonable baseline comparisons, the assessment would be substantially strengthened by (i) ensuring feature-selection steps are nested within CV, (ii) reporting variance across folds, (iii) performing principled hyperparameter tuning, and (iv) testing on an independent hold-out set. Without these, the reported ~98% accuracy should be interpreted cautiously.

Presentations & discussion on …

OXFORD

Briefings in Functional Genomics, 2023, **22**, 411–419

https://doi.org/10.1093/bfgp/elad014
Advance access publication date 28 April 2023
Review Paper

Accurate prediction and key protein sequence feature identification of cyclins

Shaoyou Yu, Bo Liao*, Wen Zhu, Dejun Peng and Fangxiang Wu

*Corresponding author: Bo Liao, Key Laboratory of Computational Science and Application of Hainan Province, Haikou, China. Tel.: 0898-65883210;
E-mail: dragonbw@163.com