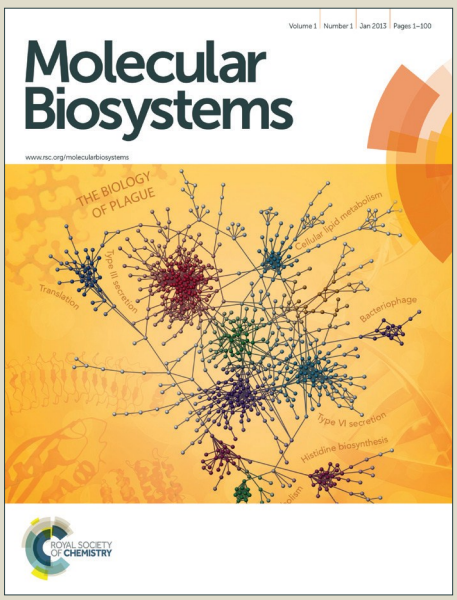
Molecular BioSystems

Accepted Manuscript



ROYAL SOCIETY OF CHEMISTRY

www.rsc.org/molecularbiosystems

AAFreqCoil: A new classifier to distinguish parallel dimeric and trimeric coiled coils

Xiaofeng Wang[a,] *, Yuan Zhou[b], Renxiang Yan[c,] *

[a]*School of Mathematics and Computer Science, Shanxi Normal University, Linfen 041004, China*

[b]*College of Biological Sciences, China Agricultural University, Beijing 100193, China*

[c]*Institute of Applied Genomics, School of Biological Sciences and Engineering, Fuzhou University, Fuzhou 350108, China*

*Corresponding authors (yanrenxiang@fzu.edu.cn and nongdaxiaofeng@126.com)

Running title: *Prediction of coiled coil oligomerizations*

ABSTRACT

Coiled coils are characteristic rope-like protein structures, constituted by one or more heptad repeats. Native coiled-coil structures play important roles in various biological processes, while the designed ones are widely employed in medicine and industry. To date, two major oligomeric states (i.e. dimeric and trimeric states) of a coiled-coil structure have been observed, plausibly exerting different biological functions. Therefore, exploration of the relationship between heptad repeat sequence and coiled coil structures is highly important. In this paper, we develop a new method named AAFreqCoil to classify parallel dimeric and trimeric coiled coils. Our method demonstrated its competitive performance when benchmarked based on 10-fold cross validation and jackknife cross validation. Meanwhile, the rules that can explicitly explain the prediction results of the test coiled coil can be extracted from the AAFreqCoil model for a better explanation of user predictions. A web server and stand-alone programs implementing the AAFreqCoil algorithm are freely available at http://genomics.fzu.edu.cn/AAFreqCoil/.

1 INTRODUCTION

In the early 1950s, Crick and Pauling independently found that the structure of α-keratin is likely to be coiled coils[1, 2]. As a characteristic protein motif, coiled coils are frequently presented in text books. They have a specific sequence pattern, unique structure and diverse biological functions[3]. Coiled coil structures have been found in many kinds of native proteins and have been widely investigated by the community. The coiled coil is a kind of rope-like structure, which contains two or more alpha-helices winding around each other[4] (Figure 1). The particular structure of the coiled coil result from its specific sequence pattern, "*HXXHCXC*" repeats, where *H* represents hydrophobic amino acid residues, *C* stands for charged residues and *X* denotes any residue locating in the position. This pattern is usually called heptad repeat and the seven positions in one repeat are consecutively labeled as ***abcdefg*** registers. In general, residues at ***a*** and ***d*** positions are buried inside the coiled coil structure to form the hydrophobic core, while residues at ***e*** and ***g*** positions are exposed outside[5] in favor of the formation of salt bridge and electrostatic interactions. Therefore, these residues (i.e. residues at ***a***, ***d***, ***e***, and ***g*** positions) are essential to maintain the stability and specificity of the coiled coil structural motif[6, 7].

Coiled coils are a versatile motif in organisms[8]. They mainly exist in transcription factors, structural proteins, membrane proteins, and enzymes, performing various cellular functions such as metabolism regulation, muscle contraction, transmembrane transport, and molecular chaperones and so on. The property of coiled coils to mediate solid interactions between two or more proteins has attracted protein designers. Meanwhile, the design of coiled coil structures has been widely applied in the fields of biology, industry and medicine[9, 10]. For example, through designing a polypeptide chain which forms a hexameric coiled coil structure with the membrane receptor in the target cell, the invasion of the human immunodeficiency virus (HIV) can be successfully prevented[11].

Due to the importance of coiled coils, a series of computational methods has been elegantly developed to study coiled coils. As examples, SOCKET[12] and Twister[13] are two structure-seeded methods for coiled coil prediction. SOCKET is designed to

identify coiled coil motifs from a protein three-dimensional structure. It also defines the heptad registers, orientation and oligomeric states of the coiled coil. Later, a coiled coil database CC+[14] was constructed by employing this program. Similarly, Twister can also compute the structure parameters of coiled coils. Since structural data are not always available, there are several effective methods, including PCOILS[15], MARCOIL[16], PAIRCOIL[17] and CCHMM-PROF[18] to predict coiled coil region from a protein sequence. Finally, recent methods elaborate more on the prediction of an important property of the coiled coil, i.e. its oligomeric states. PrOCoil[19], SCORER2.0[20], and RFCoil[4] focus on the discrimination between parallel dimeric and trimeric coiled coils. PrOCoil uses amino acid pair pattern as input to train a support vector machine based model. SCORER2.0 employs amino acid frequencies combined with a Bayes factor method. RFCoil is a random forest based predictor. In addition, the development of more comprehensive predictors has also been enabled. LOGICOIL[21] is a predictor for both coiled-coils' oligomeric states and their orientations, while MultiCoil2[22] can predict the coiled-coil region and its oligomeric state at once. CCBuilder probably is the latest interactive web server-based tool to build, design and assess coiled coil models[23]. Although several methods have been proposed, the performance of coiled coil prediction is not yet very satisfactory. Especially, as an avenue connecting protein sequence and coiled coil structure design, new algorithms for coiled coil oligomeric state prediction are still desirable. Previously, we developed RFCoil, in which the random forest method and amino acid index features were combined to predict coiled coil oligomeric states. But the dimension of its input features was too large (3,703 features) and the prediction results were not very stable when compared with other methods.

In this paper, we address this issue by employing not only state-of-the-art random forest method but also elegant amino acid frequency-based encoding. Our novel predictor, AAFreqCoil, was strictly benchmarked and the results demonstrate that it is very competitive when compared with other methods. Furthermore, the hidden and complex oligomeric formation mechanism of the coiled coils was investigated by extracting explainable rules from the predictive model. These unambiguous rules

could tell us how amino acid propensities of certain positions in the heptad repeats jointly influence the final oligomeric state of coiled coils.

2 MATERIALS AND METHODS

2.1 Datasets

To train and validate our method, a training dataset was compiled as follows. First, all the parallel dimeric and trimeric coiled coils, which contain eight or more residues, were downloaded from the CC+ database (http://coiledcoils.chm.bris.ac.uk/ccplus/search/, version of 26 April, 2013). Second, protein structures were downloaded from PDB database (released after 26 April, 2013)[24], and the SOCKET program was used to identify coiled coils from these protein structures. The sequences and registers of the identified parallel dimeric and trimeric coiled coils were extracted from the SOCKET outputs. Then, CD-HIT[25] was used to cull the coiled coil sequences. We noted that in many research works[26-29], a cutoff threshold of 25% or 30% was imposed to exclude sequences that have equal to or greater than 25% or 30% sequence identity with any other in the same dataset. However, in this study, such a stringent criterion was not used because the small amount of data does not allow us to do so. Otherwise, the number of culled sequences would be too few to evaluate statistical significance. Instead, like LogiCoil[21], a 50% sequence identity threshold was used to cull the non-redundant coiled coil sequences in the dataset. As a result, our training dataset includes 496 dimers and 100 trimers (listed in supplementary material 1).

2.2 Statistical significance of amino acid frequency difference

In order to test whether there was a significant difference in amino acid frequencies between dimers and trimers, we computed the 20 amino acid frequencies at each heptad register (position) of the dimeric and trimeric coiled coils in the training dataset, respectively. Then, the Z statistic[30-32] was used to estimate the statistical significance of the hypothesis $\boldsymbol{p}_{2,r,a} - \boldsymbol{p}_{3,r,a} \neq 0$ (i.e. the frequencies of amino acid a at

register *r* differ between dimers and trimers):

$$Z = \frac{| p_{2,r,a} - p_{3,r,a} |}{\sqrt{\frac{p_{2,r,a}(1-p_{2,r,a})}{n_{2,r}} + \frac{p_{3,r,a}(1-p_{3,r,a})}{n_{3,r}}}} \quad (1)$$

In the above formula, $n_{2,r}$ and $n_{3,r}$ represent the counts of amino acids at heptad register *r* in dimers and trimers respectively. $p_{2,r,a}$ and $p_{3,r,a}$ are the frequencies of amino acid *a* at register *r* in the dimeric and trimeric coiled coils respectively. Under the null hypothesis that $p_{2,r,a} - p_{3,r,a} = 0$, *p*-value $= P\{| Z | \geq z\}$ represents the significance level. The smaller *p*-value is, the more credible that the alternative hypothesis $p_{2,r,a} - p_{3,r,a} \neq 0$ is.

2.3 Random forest

The random forest[33] algorithm is an ensemble learning[34-36] method that integrates multiple classifiers to improve the prediction accuracy. It's known that for a supervised classifier, the classification error is partly attributed to the divergence between the distributions of the training samples and those of unknown ones. The ensemble learning method doesn't train a set of classifiers on the same training set. In contrast, it performs a series of perturbations to the training set, which can learn more general predictive model from the perturbed datasets and remove the single classifier's bias[37, 38].

A random forest consists of many decision trees and the final classification is decided by the votes from all of the trees. Both theoretical and experimental researches have proved that random forest can effectively improve the generalization ability[39-41]. Usually, three steps are required to build a random forest classification model. First, prepare a dataset with *N* samples, and randomly select *N* samples with replacement to obtain a new dataset. Repeat this procedure *n* times to obtain *n* different sub-datasets. Second, construct one decision tree based on one sub-dataset, therefore *n* decision trees can be obtained in total. One decision tree is consisted of nodes and branches. At the root node and the intermediate nodes, *m* features are

randomly selected from all of the features, and the feature enabling best split of the sub-dataset is selected to deduce the decision criterion. Each leaf node of one decision tree is the classification outputs based on the criteria represented by a series of nodes from the root node to this leaf node. Lastly, after obtaining *n* decision trees, count the votes for classification outputs by these trees and the most approved one is chosen as the final prediction result.

In practice, to build the random forest model, 4,000 trees were grown, which was found enough to ensure stable prediction results. In order to get the optimal performance, we tuned the value of parameter *m* from 1 to 50. In addition, considering the unbalance of dimers and trimers in the training dataset, the weight for dimer class and trimer class, which is denoted as *w*, was tuned from 1:1 to 1:10. The prediction performances with different combinations of the parameters on 10-fold cross validation are listed in the supplementary material file 2. When the value of *m* and *w* were set as 33 and 1:2 respectively, the prediction performance AUC (area under the ROC curve) will reach its maximum value of 0.875.

In the process of building the random forest model, the importance of each feature for classification can be estimated. When a feature is split at one node of the decision tree, the descendent nodes should have less gini impurity[42] than this parent node. The sum of the gini decreases for the feature over all the trees can be exploited to measure the importance of the feature. In this paper, we used the Randomforest package[43] to build our predictive model.

2.4 Encoding

We used the 20 amino acid frequencies at 7 heptad registers to encode the coiled coil, which can be denoted by the following equation:

$$(\boldsymbol{r}, \mathbf{A}) = \frac{\boldsymbol{n}(\boldsymbol{r}, \mathbf{A})}{\boldsymbol{n}(\boldsymbol{r})}$$

In the above formula, $\boldsymbol{n}(\boldsymbol{r}, \mathrm{A})$ represents the counts of heptad register $\boldsymbol{r}$ whose amino acid residue is A and $\boldsymbol{n}(\boldsymbol{r})$ is the number of heptad register $\boldsymbol{r}$ in a coiled coil sequence. Then, a coiled coil can be represented by a 140-dimensional vector. The

encoding is simple but robust. When combined with random forest, the relation between amino acids at different heptad registers can be readily captured by the decision trees of a random forest. We also noted that many protein prediction issues have extended single amino acid composition to dipeptide[44, 45], tripeptide[46, 47], or tetrapeptide[48, 49] compositions. When using dipeptide frequencies as the input to random forest, the average AUC on the 10-fold cross validation was 0.809, which is not better than that could be achieved by using single amino acid frequencies.

2.5 Rule-extracting

For a decision tree, each path from the root node to the leaf node represents a rule. So the random forest model can be considered as a series of rule sets. In our previous work, we devised a novel and easy method to extract the minimum set of rules from the random forest model covering all the training data[4]. The rule-extracting method can be also applied to AAFreqCoil. Briefly, the following procedure was used to extract dimer rules. First, suppose the initial dimer rule set is empty. Extract all the rules from the model that correctly predict dimers but don't wrongly predict a trimer to be a dimer. Second, find the rule that covers most dimeric coiled coils, put this rule to the rule set and delete dimers from the training set that conform to this rule. Third, repeat the second step until there is no dimer in the training dataset. Using a similar procedure, the trimer rules can also be extracted.

For one testing coiled coil, the predictive model can tell the probabilities of being dimeric and trimeric. But for a protein designer, the reason why this coiled coil is predicted as dimeric or trimeric is more cared. Recall the fact that to build the AAFreqCoil model, 4,000 trees were grown, and each tree used one rule to decide its classification output. Taken together, there are 4,000 rules that give rise to one testing coiled coil's classification. Since there are many rules available, we employed accuracy and coverage to measure a rule's contribution to the final classification. Coverage is the number of coiled coils whose classification can be deduced by this rule, while accuracy is the fraction of correct classification to all classification based on this rule. In our work, accuracy is considered prior to coverage.

2.6 Performance evaluation

In statistical prediction, cross validation test is often used to examine a predictor for its effectiveness in practical application[50, 51]. In *k*-fold cross validation, the dataset was randomly partitioned into *k* equal size subsets. One subset was retained to test the predictive model trained by the remaining *k*-1 subsets, and this process was repeated *k* times. Here, we used popular 10-fold cross validation and more stable but time-consuming jackknife cross validation for performance evaluation. Jackknife cross validation is a special type of *k*-fold cross validation where *k* is equal to the number of samples in the dataset. That is to say, every sample will be retained once as the testing sample. The advantage of jackknife cross validation is that it can always yield a unique result for a given benchmark dataset, as elucidated in[52] and demonstrated by the work of Chou and Shen[53]. Therefore, the jackknife cross validation has been increasingly and widely adopted by investigators to test the power of various predictors[54-59].

More specifically, 10-fold cross validation and jackknife cross validation were used to evaluate the performance of AAFreqCoil and other methods including SCORER2.0, PrOCoil and RFCoil. Because the source code of LOGICOIL only contains the trained model, we submitted the training coiled coil sequences and registers to the LOGICOIL server and got the prediction results. LOGICOIL can output the score of a coiled coil being antiparallel dimeric, parallel dimeric, trimeric and tetremeric, respectively. We took parallel dimeric score divided by the sum of parallel dimeric score and trimeric score as the final predictive score of LOGICOIL.

Although MultiCoil2 can also distinguish coiled coil oligomeric states, it is different from the aforementioned five predictors. It predicts each residue of a sequence to be non-coiled-coil, dimeric or trimeric. We submitted the coiled coil sequences and registers of the training dataset to the online webserver of MultiCoil2, and the probability for each residue to be dimeric was recorded. The probability for a coiled coil to be dimeric is the average probability of its residues being in dimeric state.

2.7 Performance measure

We used the ROC curve[60, 61] to measure the performance of different predictors. The ROC curve is often used to measure the performance of binary classification methods. It plots the true positive rate (TPR) against the false positive rate (FPR) as the discrimination threshold varies. In this paper, dimeric coiled coils are defined as positive samples, and trimeric coiled coils are defined as negative samples. TPR is the ratio of correctly predicted positive samples to all the positive samples. FPR is the ratio of the negative samples falsely predicted to be positive samples to all the negative samples. The area under a ROC curve (AUC) quantifies the overall prediction ability of a classifier[62, 63]. So AUC is usually used as a measure to compare two models and the model whose ROC curve closer to the top left (i.e. with larger AUC) is considered to be better. In our paper, the ROCR[64] package was used to plot the ROC curves.

3 RESULTS AND DISCUSSIONS

3.1 The difference of amino acid frequencies between dimeric and trimeric coiled coils

For exploring whether there is a significant difference in amino acid occurrences between parallel dimers and trimers, we computed the frequencies of 20 amino acids at each heptad register and used Z-test for significance evaluation. Figure 2 illustrates the distribution of 20 amino acids at 7 heptad registers in trimers and dimers. From Figure 2, we can see that G (Gly) occurs very few at registers ***a***, ***d***, ***e,*** and ***g***, perhaps due to the fact that G is too small to form stable hydrophobic interfaces or electrostatic interactions. P (Pro) rarely occurs at all 7 registers, as proline is of exceptional conformational rigidity compared to other amino acids and often acts as a structural disruptor in the middle of alpha helices. It can also be observed that at positions ***a*** and ***d***, hydrophobic residues L (Leu), A (Ala), V (Val) and I (Ile) occur most frequently, which agree well with the fact that in the coiled coil structure, ***a*** and ***d*** positions are buried interior to form the hydrophobic core. In addition, these amino

acids have larger frequencies at ***a*** in trimers than in dimers, which demonstrates that higher-order oligomeric states need stronger hydrophobic force[65]. At positions ***e*** and ***g***, polar residues E (Glu), K (Lys), Q (Gln), and R (Arg) occur most. These residues can mediate the electrostatic interactions between ***e*** and ***g***. It can be seen that for most amino acids, there is a marginal difference in frequencies at each register between dimers and trimers. Indeed, we merely found K (Lys) at register ***a***, P (Pro) at register ***a***, ***b***, ***c***, and ***d***, W (Trp) at register ***b***, Y (Tyr) at register ***a*** are significantly different in occurrences by the Z-test with a p-value<0.05. But methods like SCORER2.0 and RFCoil, which use single amino acid information, could still achieve good performance. This demonstrates that the discrimination between parallel dimeric and trimeric coiled coils depend on the joint divergence of each amino acid in the coiled-coil and machine learning method should be suitable to summarize these marginal frequency differences and accurately predict the oligomeric state of coiled coils. .

3.2 Prediction performance based on 10-fold cross validation

We performed 10-fold cross validation on the training dataset. When *m* is 33, and the class weight is set 1:2 for the dimer and the trimer, AAFreqCoil gets the best performance. Here, the ROC curve is used to describe the performance of predictive methods. The values of AUC for different methods based on 10-fold cross validation are listed in Table 1. From the table, we can see that the average AUC for SCORER2.0, PrOCoil, RFCoil and AAFreqCoil are 0.838, 0.863, 0.841 and 0.875 respectively, among which AAFreqCoil obtains the highest AUC value. We also computed the standard error of AUC across the 10-fold cross validation. For the above four predictors, the values of the standard error are 0.084, 0.066, 0.077 and 0.049, respectively, which demonstrates that AAFreqCoil is more stable than other methods.

3.3 Prediction performance on jackknife cross validation

In addition to 10-fold cross validation, jackknife cross validation was used to further examine the prediction performance of different methods. The ROC curves for

different methods are plotted in Figure 3 and the values of AUC for SCORER2.0, PrOCoil, RFCoil and AAFreqCoil are 0.851, 0.860, 0.836 and 0.882, where AAFreqCoil still achieves the best performance.

We submitted the training coiled coil sequences and registers to the web servers of LOGICOIL and MultiCoil, and plotted the ROC curves based on their prediction results (see Materials and Methods for details). As a result, we found that the AUC of LOGICOIL can reach 0.849. MultiCoil2 predicted that only 112 coiled coils contained coiled coil residues. So we plotted its ROC curve based on these 112 coiled coils, and the corresponding value of AUC is 0.781. Taken together, our results demonstrated that AAFreqCoil can achieve competitive performance when benchmarked on 10-fold cross validation and jackknife cross validation.

3.4 Rules extracted from AAFreqCoil model

In order to investigate the mechanism of the coiled coil oligomeric state formation, we devised a method to extract informative rules from AAFreqCoil model that cover all the training data (see Materials and Methods for details). We finally obtained 17 rules (supplementary material 3) to predict dimers and 13 rules to predict trimers (supplementary material 4).

The rule that covers most dimers is:

(***g***, R)<=1.5 & (***c***, T)<=0.5 & (***a***, V)<=2.5 & (***e***, C)<=0.5 & (***c***, D)<=0.5 & (***c***, E)<=1.5 & (***e***, E)<=0.5 & (***a***, I)<=0.5.

In the above rule, (***r***, A) denotes the number of amino acids A at register ***r***. "&" denotes the logical conjunction. 211 dimers in the training dataset conform to this rule.

The rule that covers most trimers is:

(***a***, Y)<=0.5 & (***d***, Y)<=0.5 & (***e***, E)<=4.5 & (***e***, E)>0.5 & (***g***, E)<=2.5 & (***d***, A)<=0.5 & (***a***, K)<=0.5 & (***d***, K)<=0.5 & (***d***, L)>0.5 & (***e***, M)<=0.5 & (***g***, M)<=1.5 & (***a***, N)<=0.5 & (***b***, P)<=0.5 & (***a***, Q)<=0.5 & (***a***, R)<=0.5.

29 trimers conform to the above rule.

The RFCoil model was built based on the training dataset. 12 dimeric rules and

10 trimeric rules were extracted from the RFCoil model (see supplementary material 5 and 6). Rules from RFCoil model and AAFreqCoil model can both correctly distinguish dimeric and trimeric coiled coils in the training dataset. Compared with rules from RFCoil model, rules from AAFreqCoil model are more applicable, because the amino acid frequencies at each register are very easy to calculate and interpret. In contrast, RFCoil considers the average values of the 529 amino acid indices at each register, which is more complicated and not straightforward to understand.

3.5 Gini importance of each feature

A random forest can give gini importance of each input feature. Here, we summed the gini importance of 20 amino acids for each heptad register and listed them in Table 2. As shown in Table 2, the hydrophobic positions ***a*** and ***d*** are more important than other positions for classification, and position ***a*** is of the most importance. The importance of other 5 positions decreases in the order of ***e***, ***g***, ***c***, ***f*** and ***b***.

Meanwhile, the 10 most important individual features are listed in Table 3. Among them, only K at ***a*** and Y at ***a*** have been approved by the aforementioned Z-test, showing significant frequency differences between dimers and trimmers (Figure 2). This suggests that the discrimination of parallel dimeric and trimeric coiled coils is not only dependent on the single amino acid frequency, but also dependent on other factors, such as the correlation between amino acids at different heptad registers and the accumulation of amino acid differences.

3.6 Case study

To provide a realistic example, we applied AAFreqCoil to a coiled coil protein Bicaudal-D (PDB entry 4BL6)[66]. For clarity, we focused on the prediction results of chain B in this structure.

The amino acids and heptad registers of the coiled coil region of the protein 4BL6_B is given in Figure 4. Its real oligomeric state is parallel dimeric. AAFreqCoil reports the probability for it to be dimeric is as high as 0.963, while the probability to be trimeric is 0.037. This means that among the 4,000 trees, 96.3% of them classify

the coiled coil as a dimer, while only 3.7% of them classify the coiled coil as a trimer. The three best rules for it to be dimeric and trimeric, respectively, can also be extracted to explain the prediction results. The rules and their coverage are listed in Table 4. The best rule for the coiled coil to be dimeric is that if there are at least 1 R at register ***b***, less than 2 V at register a and less than 2 R at register g, and there are no S at register ***g***, nor T at register ***b***, the coiled coil is parallel dimeric. According to the fourth row in Table 4, the coiled coil is a trimer because there exist more than 1 L at position ***d***, E at position ***g***, K at position ***c***, and L at position ***c***, and there are no E at position ***d***, nor D at position ***d***. Apparently, the coverage of dimer rules are much more higher than trimer rules, indicating that the coiled coil region of this protein fits well to the general dimer rules extracted from the AAFreqCoil model (and thus finally be predicted as an dimer), but show little similarity to known trimeric coiled coils in the training dataset.

It should be noted that we took this case study to show that understandable prediction results can be drawn by applying AAFreqCoil. We did not aim at comparing different methods in this case. Indeed, except for LOGICOIL, other methods all gave a reliable prediction results on this particular case (Table 5).

3.7 Web server

To aid the research community, a user-friendly web server implementing our method is developed and is publicly available at http://genomics.fzu.edu.cn/AAFreqCoil/. Users can submit a protein sequence and its corresponding heptad registers of the coiled coil to the server to get the prediction results. Meanwhile, the predominant rules explaining the result are also shown in the result page. The server was designed using programming languages of R, PHP and HTML. The computational time for finishing a job depends on the length of the query sequence. It is estimated that a typical job will be finished within 3 minutes if the protein contains less than 150 amino acids. To facilitate batch predictions, the source code and stand-alone program of AAFreqCoil can also be downloaded from the web server.

4 Conclusions

In this paper, we used a state-of-the-art machine learning method, random forest, to develop a new tool called AAFreqCoil for parallel dimeric and trimeric coiled coils discrimination. AAFreqCoil demonstrated its competitive performance in comparison to the existing methods. Meanwhile, we counted the frequencies of 20 amino acids at 7 heptad registers and found that except a few amino acids at some registers, most of the amino acids at certain heptad registers show no great difference in frequencies between dimeric and trimeric coiled coils. So the efficient classification between parallel dimeric and trimeric coiled coils should depend not only on the single amino acid frequencies, but also on the combination of amino acids at different positions. Furthermore, we devised a method to extract explainable rules from AAFreqCoil model, which trained by all the trimers and dimers in the dataset. For each test coiled coil, the understandable rules can be extracted from AAFreqCoil model to explain the prediction results. Finally, a web server, all source codes and the stand-alone program of AAFreqCoil are publicly available at http://genomics.fzu.edu.cn/AAFreqCoil/ to serve the biological community.

Acknowledgements

This work was supported by Start-up fund of Shanxi Normal University (83358), the Education and Science Foundation for Young teachers of Fujian (650056), Start-Up Fund of Fuzhou University (510046). More importantly, we are grateful to Professor Ziding Zhang in China Agricultural University for helpful comments. Last but not least, we also gratefully thank all those who make their experimental data and bioinformatics programs publicly available.

Supplementary material Information

Supplementary material 1: The training dataset.

Supplementary material 2: Prediction performances of AAFreqCoil using different combined parameters on 10-fold cross validation.

Supplementary material 3: Informative dimer rules obtained from the AAFreqCoil model.

Supplementary material 4: Informative trimer rules obtained from the AAFreqCoil model.

Supplementary material 5: Dimer rules obtained from the RFCoil model.

Supplementary material 6: Trimer rules obtained from the RFCoil model.

References

1. F. Crick, *Acta Crystallographica*, 1953, **6**, 689-697.
2. L. Pauling and R. B. Corey, *Nature*, 1953, **171**, 59-61.
3. A. N. Lupas and M. Gruber, *Advances in protein chemistry*, 2005, **70**, 37-78.
4. C. Li, X. F. Wang, Z. Chen, Z. Zhang and J. Song, *Molecular bioSystems*, 2015, **11**, 354-360.
5. Y. B. Yu, *Advanced drug delivery reviews*, 2002, **54**, 1113-1129.
6. G. Grigoryan and A. E. Keating, *Current opinion in structural biology*, 2008, **18**, 477-483.
7. J. M. Mason and K. M. Arndt, *ChemBioChem*, 2004, **5**, 170-176.
8. P. Burkhard, J. Stetefeld and S. V. Strelkov, *Trends in cell biology*, 2001, **11**, 82-88.
9. H. Chao, D. L. Bautista, J. Litowski, R. T. Irvin and R. S. Hodges, *Journal of chromatography. B, Biomedical sciences and applications*, 1998, **715**, 307-329.
10. G. De Crescenzo, P. L. Pham, Y. Durocher and M. D. O'Connor-McCourt, *Journal of molecular biology*, 2003, **328**, 1173-1183.
11. N. R. Kilgore, K. Salzwedel, M. Reddick, G. P. Allaway and C. T. Wild, *Journal of virology*, 2003, **77**, 7669-7672.
12. J. Walshaw and D. N. Woolfson, *Journal of molecular biology*, 2001, **307**, 1427-1450.
13. S. V. Strelkov and P. Burkhard, *Journal of structural biology*, 2002, **137**, 54-64.
14. O. D. Testa, E. Moutevelis and D. N. Woolfson, *Nucleic acids research*, 2009, **37**, D315-322.
15. M. Gruber, J. Soding and A. N. Lupas, *Journal of structural biology*, 2006, **155**, 140-145.
16. M. Delorenzi and T. Speed, *Bioinformatics*, 2002, **18**, 617-625.
17. B. Berger, D. B. Wilson, E. Wolf, T. Tonchev, M. Milla and P. S. Kim, *Proceedings of the National Academy of Sciences of the United States of America*, 1995, **92**, 8259-8263.
18. L. Bartoli, P. Fariselli, A. Krogh and R. Casadio, *Bioinformatics*, 2009, **25**, 2757-2763.
19. C. C. Mahrenholz, I. G. Abfalter, U. Bodenhofer, R. Volkmer and S. Hochreiter, *Molecular & cellular proteomics : MCP*, 2011, **10**, M110 004994.
20. C. T. Armstrong, T. L. Vincent, P. J. Green and D. N. Woolfson, *Bioinformatics*, 2011, **27**, 1908-1914.
21. T. L. Vincent, P. J. Green and D. N. Woolfson, *Bioinformatics*, 2013, **29**, 69-76.
22. J. Trigg, K. Gutwin, A. E. Keating and B. Berger, *PloS one*, 2011, **6**, e23519.
23. C. W. Wood, M. Bruning, A. A. Ibarra, G. J. Bartlett, A. R. Thomson, R. B. Sessions, R. L. Brady and D. N. Woolfson, *Bioinformatics*, 2014, **30**, 3029-3035.
24. H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov and P. E. Bourne, *Nucleic acids research*, 2000, **28**, 235-242.
25. W. Li and A. Godzik, *Bioinformatics*, 2006, **22**, 1658-1659.

26. K. C. Chou and H. B. Shen, *PloS one*, 2010, **5**, e9931.
27. P. P. Zhu, W. C. Li, Z. J. Zhong, E. Z. Deng, H. Ding, W. Chen and H. Lin, *Molecular bioSystems*, 2015, **11**, 558-563.
28. H. Lin, W. Chen and H. Ding, *PloS one*, 2013, **8**, e75726.
29. J. B. Leikin, L. J. Frateschi, D. A. Boston, P. J. Eckenrode, R. J. Morris, L. J. Konczyk and D. O. Hryhorczuk, *The Journal of emergency medicine*, 1990, **8**, 545-550.
30. J. L. Loveland, Utah State University, 2011.
31. N. A. Weiss, 1989.
32. G. Dahlberg, *Statistical Methods for Medical and Biological Students.*, 1940.
33. L. Breiman, *Machine Learning*, 2001, **45**, 5-32.
34. R. Maclin and D. Opitz, *Journal of Artificial Intelligence Research*, 1999.
35. R. Polikar, *Circuits and systems magazine, IEEE*, 2006, **6**, 21-45.
36. L. Rokach, *Artificial Intelligence Review*, 2010, **33**, 1-39.
37. G. Brown, J. Wyatt, R. Harris and X. Yao, *Information Fusion*, 2005, **6**, 5-20.
38. J. J. G. Adeva, U. Beresi and R. Calvo, *CLEI Electronic Journal*, 2005, **9**, 1-12.
39. X. F. Wang, Z. Chen, C. Wang, R. X. Yan, Z. Zhang and J. Song, *PloS one*, 2011, **6**, e26767.
40. N. Zhang, B. Q. Li, S. Gao, J. S. Ruan and Y. D. Cai, *Molecular bioSystems*, 2012, **8**, 2946-2955.
41. Z. C. Li, Y. H. Lai, L. L. Chen, Y. Xie, Z. Dai and X. Y. Zou, *Molecular bioSystems*, 2014, **10**, 514-525.
42. H. Deng, G. Runger and E. Tuv, in *Artificial Neural Networks and Machine Learning–ICANN 2011*, Springer, 2011, pp. 293-300.
43. A. Liaw and M. Wiener, *R news*, 2002, **2**, 18-22.
44. Y. F. Liou, P. Charoenkwan, Y. Srinivasulu, T. Vasylenko, S. C. Lai, H. C. Lee, Y. H. Chen, H. L. Huang and S. Y. Ho, *BMC bioinformatics*, 2014, **15 Suppl 16**, S4.
45. H. Ding, P. M. Feng, W. Chen and H. Lin, *Molecular bioSystems*, 2014, **10**, 2229-2235.
46. W. X. Liu, E. Z. Deng, W. Chen and H. Lin, *International journal of molecular sciences*, 2014, **15**, 12940-12951.
47. S. Reumann, D. Buchwald and T. Lingner, *Frontiers in plant science*, 2012, **3**, 194.
48. Y. Feng and L. Luo, *Amino acids*, 2008, **35**, 607-614.
49. H. Ding, H. Lin, W. Chen, Z. Q. Li, F. B. Guo, J. Huang and N. Rao, *Interdisciplinary sciences, computational life sciences*, 2014, **6**, 235-240.
50. K. C. Chou and C. T. Zhang, *Critical reviews in biochemistry and molecular biology*, 1995, **30**, 275-349.
51. R. Kohavi, Ijcai, 1995.
52. K. C. Chou and H. B. Shen, *Nature protocols*, 2008, **3**, 153-162.
53. K. C. Chou and H. B. Shen, *Analytical biochemistry*, 2007, **370**, 1-16.
54. S. P. Shi, X. Chen, H. D. Xu and J. D. Qiu, *Molecular bioSystems*, 2015, **11**, 819-825.
55. W.-C. Li, E.-Z. Deng, H. Ding, W. Chen and H. Lin, *Chemometrics and Intelligent Laboratory Systems*, 2015, **141**, 100-106.
56. C. Li, Y. Yang, W. Fei, P. A. He, X. Yu, D. Zhang, S. Yi, X. Li, J. Zhu, C. Wang and Z. Wang, *Journal of theoretical biology*, 2015, **369**, 51-58.
57. Z. Liu, X. Xiao, W. R. Qiu and K. C. Chou, *Analytical biochemistry*, 2015, **474**, 69-77.
58. H. Lin, E. Z. Deng, H. Ding, W. Chen and K. C. Chou, *Nucleic acids research*, 2014, **42**, 12961-12972.

59. H. Ding, E. Z. Deng, L. F. Yuan, L. Liu, H. Lin, W. Chen and K. C. Chou, *BioMed research international*, 2014, **2014**, 286419.
60. T. Fawcett, *Pattern Recognition Letters*, 2006, **27**, 861-874.
61. J. A. Swets, *Signal detection theory and ROC analysis in psychology and diagnostics: Collected papers*, Psychology Press, 2014.
62. S. Mason and N. Graham, *Quarterly Journal of the Royal Meteorological Society*, 2002, **128**, 2145-2166.
63. J. A. Hanley and B. J. McNeil, *Radiology*, 1982, **143**, 29-36.
64. T. Sing, O. Sander, N. Beerenwinkel and T. Lengauer, *Bioinformatics*, 2005, **21**, 3940-3941.
65. D. N. Woolfson, G. J. Bartlett, M. Bruning and A. R. Thomson, *Current opinion in structural biology*, 2012, **22**, 432-441.
66. Y. Liu, H. K. Salter, A. N. Holding, C. M. Johnson, E. Stephens, P. J. Lukavsky, J. Walshaw and S. L. Bullock, *Genes & development*, 2013, **27**, 1233-1246.

TABLES

Table 1. Comparison of AUC scores for different methods based on the 10-fold cross validation

fold	SCORER2.0	PrOCoil	RFCoil	AAFreqCoil
1	0.737	0.829	0.718	0.827
2	0.848	0.768	0.900	0.848
3	0.840	0.920	0.804	0.875
4	0.898	0.930	0.834	0.936
5	0.896	0.812	0.837	0.896
6	0.726	0.762	0.776	0.786
7	0.952	0.940	0.966	0.910
8	0.788	0.888	0.841	0.935
9	0.943	0.898	0.947	0.896
10	0.753	0.886	0.787	0.839
average	0.838	0.863	0.841	0.875
standard error	0.084	0.066	0.077	0.049

Table 2. Gini importance of each heptad register

heptad register	a	b	c	d	e	f	g
gini importance	67.8	22.3	31.1	44.0	40.9	26.3	31.8

Table 3. The 10 most important features for classification

(***a***, I)	(***e***, E)	(***a***, K)	(***d***, L)	(***a***, N)	(***d***, K)	(***a***, V)	(***a***, Y)	(***c***, D)	(***g***, S)
14.6	14.1	11.1	9.5	9.0	6.2	6.0	5.7	5.1	4.8

Table 4. The 3 best rules for the case study

rules	coverage	
	dimer	trimer
(***a***,V)<=1.5&(***g***,S)<=0.5&(***g***,R)<=1.5&(***b***,R)>0.5&(***b***,T)<=0.5, then dimer	78	0
(***a***,K)>0.5, then dimer	63	0
(***a***,V)<=2.5&(***a***,K)>0.5, then dimer	62	0
(***d***,E)<=0.5&(***d***,L)>1.5&(***g***,E)>0.5&(***c***,K)>0.5&(***c***,L)>0.5&(***d***,D)<= 0.5, then trimer	1	5
(***a***,I)<=0.5&(***d***,L)>1.5&(***b***,K)>0.5&(***e***,V)>0.5, then trimer	1	4
(***d***,K)<=0.5&(***g***,R)<=1.5&(***g***,K)<=1.5&(***d***,I)<=0.5&(***f***,D)<=0.5&(***e***, N)>0.5&(***b***,Y)<=0.5&(***a***,A)>0.5, then trimer	1	3

"&" denotes the logical conjunction. (***r***,A) denotes the number of amino acids A at register ***r***.

Table 5. Comparison of AUC scores for different methods on the case study

Method	SCORER2.0	PrOCoil	RFCoil	AAFreqCoil	LOGICOIL	MultiCoil2
AUC	9.200	-1.144	0.982	0.963	0.517	0.990

RFCoil, AAFreqCoil and LOGICOIL use 0.5 as threshold and SCORER2.0 uses 0 as threshold to discriminate dimers and trimers. For them, greater score means greater possibility for a coiled coil to be dimeric. PrOCoil uses 0 as cutoff. The smaller the predictive score from PrOCoil is, the more possible for a coiled coil to be dimeric.

Figures

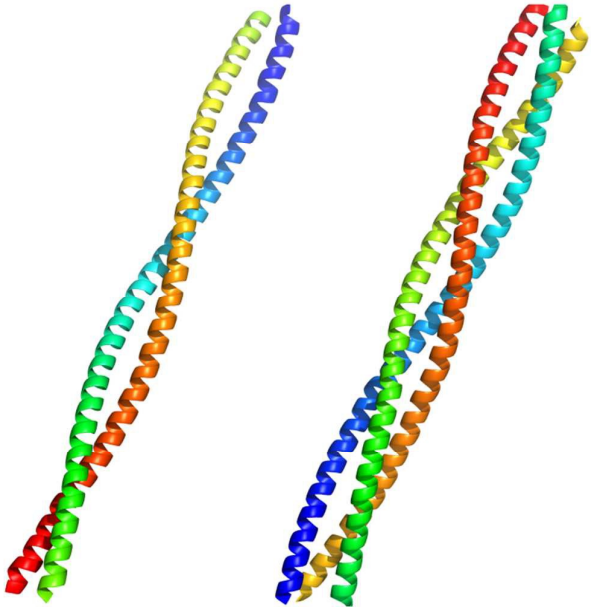


Figure 1. Structures of coiled coils. Cartoon presentation of a dimeric coiled coil and a trimeric coiled coil[4]

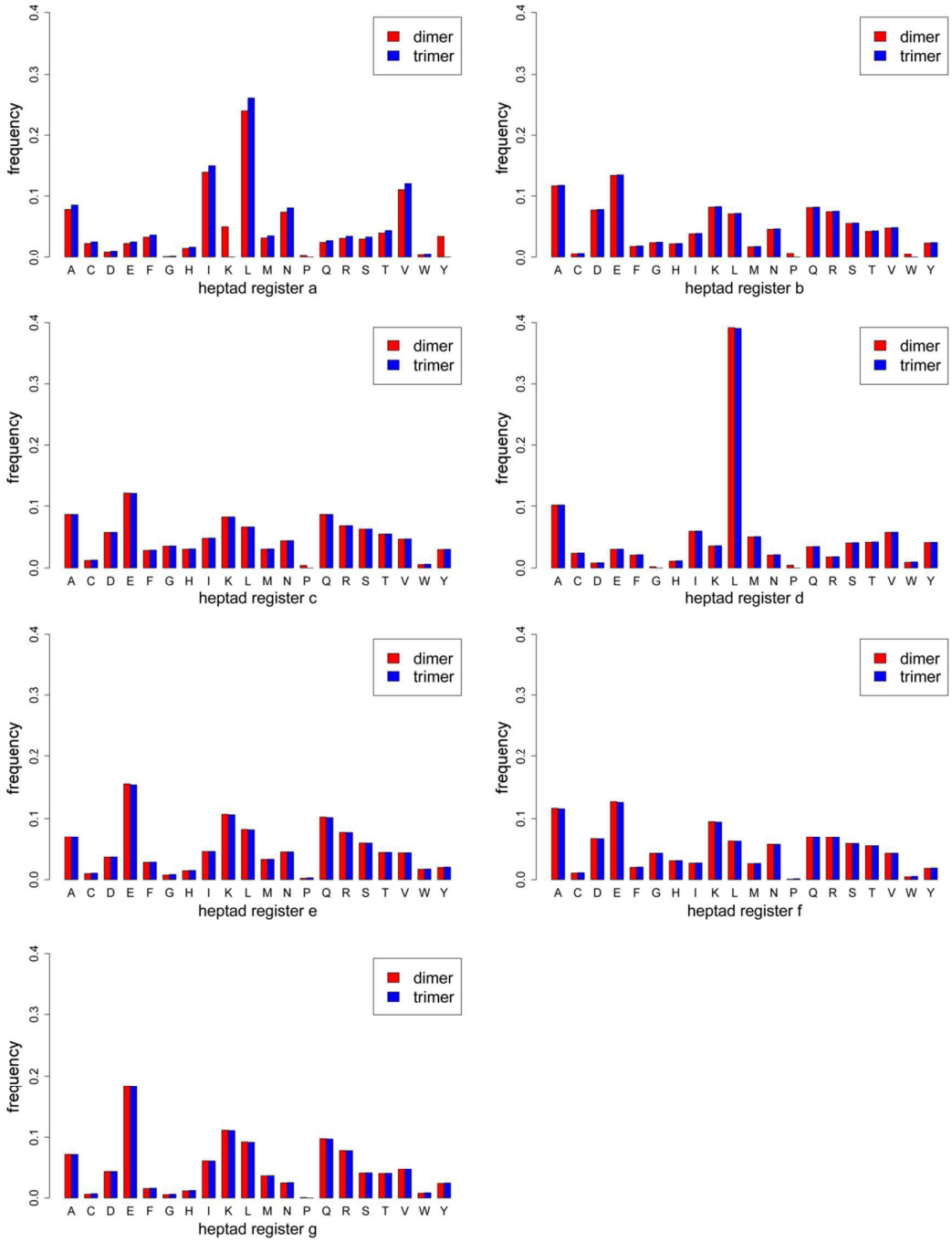


Figure 2. Amino acid frequencies at 7 heptad registers in dimers and trimers

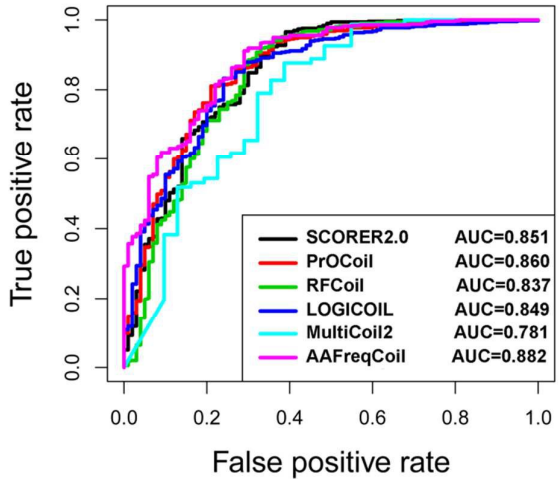


Figure 3. ROC curves for different methods on jackknife cross validation

```
16 MSKLRNELRLLKEDAATFSSLRAVFAARCEEYVTQVDDLNRQLEAAEEEKKTLNQLLRLA 76
16 abcdefgabcdefgabcdefgabcdefgabcdefgabcdefgabcdefgabcdefgabcd 76
```

Figure 4. The amino acids (upper cases) and heptad registers (lower cases) of the coiled coil region of protein 4BL6_B. The covering positions of the coiled coil region in the 4BL6_B protein are also given.