

## HMMs in Protein Fold Classification

**Christos Lampros, Costas Papaloukas, Themis Exarchos,  
and Dimitrios I. Fotiadis**

### Abstract

The limitation of most HMMs is their inherent high dimensionality. Therefore we developed several variations of low complexity models that can be applied even to protein families with a few members. In this chapter we present these variations. All of them include the use of a hidden Markov model (HMM), with a small number of states (called reduced state-space HMM), which is trained with both amino acid sequence and secondary structure of proteins whose 3D structure is known and it is used for protein fold classification. We used data from Protein Data Bank and annotation from SCOP database for training and evaluation of the proposed HMM variations for a number of protein folds that belong to major structural classes. Results indicate that the variations have similar performance, or even better in some cases, on classifying proteins than SAM, which is a widely used HMM-based method for protein classification. The major advantage of the proposed variations is that we employed a small number of states and the algorithms used for training and scoring are of low complexity and thus relatively fast. The main variations examined include a version of the reduced state-space HMM with seven states (7-HMM), a version of the reduced state-space HMM with three states (3-HMM) and an optimized version of the reduced state-space HMM with three states, where an optimization process is applied to its scores (optimized 3-HMM).

**Keywords** Fold classification, Hidden Markov model, Optimization

---

## 1 Introduction

Today we possess a huge amount of sequence information concerning proteins. The problem of determining the function of these proteins requires the knowledge of their structures [1–3]. Many experimental methods for structure determination exist, such as nuclear magnetic resonance (NMR) spectroscopy [4] and X-ray crystallography. Nevertheless, the exact structure determination remains expensive and time consuming. On the other hand, there is an indirect way to make predictions for both structural and functional attributes of a protein with unknown structure. Its amino acid sequence can be compared with proteins in annotated databases for which the 3D structure is known [1], since similarity

in the structure of proteins typically results in similarity of their function.

Moreover, proteins have similar structure if they share a common fold. Specifically, if proteins belong to the same fold category they have the same major secondary structures in the same arrangement and with the same topological connections. The fold of two common proteins can be the same even when there is a very low sequence similarity among them [5, 6]. The task of classifying the proteins into the appropriate fold category is called fold classification [7].

Fold classification can directly lead to the determination of the tertiary (3D) structure of a protein. There are two main methodological approaches in tertiary structure prediction, the template free methods [8–12] and the template-based methods [6, 13–29]. Template free methods try to directly approach the specific 3D structure of a protein using energy minimization based on physical principles rather than on previously identified structures. Template-based methods use already known structures (templates) as candidate structures of the protein. They are further divided into two categories. The first category consists of the 3D-1D fold recognition methods (otherwise called threading methods), which use a scoring function that assesses the compatibility of the amino acid sequence of unknown structure to already known structures [13–18]. The second category consists of the comparative modeling methods, where sequence alignment is used to discern the relationship between target sequence and template [6, 19–29].

Comparative modeling approaches are also known as sequence-based approaches.

There are many different machine learning methods that have been used for that purpose, such as HMMs [6, 19–21], genetic algorithms [22], artificial neural networks [23], support vector machinesSupport Vector Machines (SVMs) [24, 25], data mining [26, 27], similarity networks [28], and FRAN and RBFRadial basis function (RBF) networks [29].

Among these, HMMs have been widely used and also achieve high performance. HMMs have initially been applied for protein fold classification in the context of HMMER [19] and of the sequence alignment and modeling (SAM) system [20]. Moreover, secondary structure information was incorporated into HMMs [6] to increase their fold recognition performance. A similar approach was also used together with the evaluation of several backbone geometry alphabets [21].

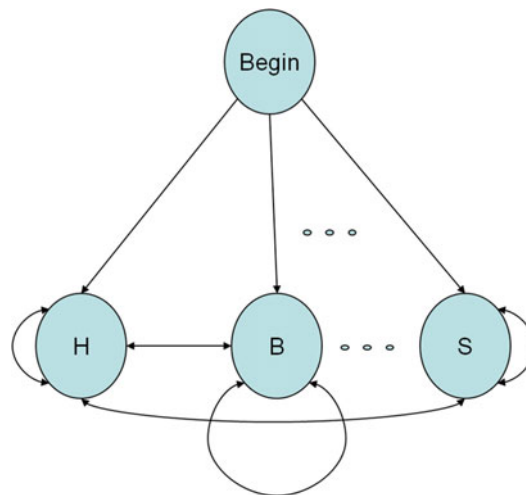
Nevertheless, the most important disadvantage of HMMs was the use of models with many states, which demand much data and significant computational effort for training [30]. Thus, it is necessary to reduce the parameters of HMMs and also employ HMMs that can be trained even with a small amount of data, while their performance in fold classification is maintained. For this purpose

we introduced different variations related to a HMM with a small number of states (reduced state-space HMM or simply reduced HMM), which is trained with both amino acid sequence and secondary structure for proteins whose 3D structure is known and it is used for protein fold classification. In this chapter we discuss these HMM models and a comparative analysis of these models in detail.

## 2 7-HMM

In the seven-state HMM model we propose the use of a simple architecture with a small number of states and a training algorithm with low complexity. We also incorporate secondary structure information into the model, which is employed in such a manner that enables us to use the low complexity algorithm and also improves its performance. The emission states are equal in number to the total of possible formations of secondary structure like helix (H),  $\beta$  strands (B), and loops (S) (Fig. 1), and there is also a begin state. The model is trained for each candidate fold. Its major advantage is that the computational complexity of the training procedure of the model is significantly smaller than the complexity of other current methods based on larger HMMs [30].

The topology of the model employs the mathematical background of a typical HMM. It models a sequence of amino acids through a hypothesized (hidden) procedure. The model contains a number of states  $S$  (nodes) and a number of possible transitions  $T$  (arrows) between them (Fig. 1). Every emission state emits an amino acid based upon a set of emission probabilities. Then transition to some other emission state takes place with a probability



**Fig. 1** Topology of the reduced state-space HMM. H, B, ..., S are the letters of DSSP secondary structure alphabet

depending on the previous state. This process continues until all of the amino acids of the protein sequence are emitted. There are also transition probabilities from the begin state, where the process starts, to each emission state. The sum of these probabilities is equal to one and so does the sum of emission probabilities of all possible amino acids in each state and the sum of transition probabilities from each state to each next possible state. Another basic feature of the model is the Markov property, which assumes that the current state  $S_t$  and all future states are independent of all the states prior to the previous state  $S_{t-1}$  [31].

The most important characteristic of the reduced state-space HMM, which makes it different from other HMMs, is that it incorporates the secondary structure information in such a manner that each state of the model corresponds to a different type of secondary structure. The correct assignment of the protein to its structural group will be done not only with its primary structure information but also with the use of its secondary structure information. We use secondary structure sequences which are incorporated in the context of our HMM as hidden state sequences. This allows the benefit of employing a HMM with a number of states equal to the number of the different letters ( $\{H, B, E, G, I, T, S\}$ , see Table 1) in the definition of secondary structure of proteins (DSSP) alphabet [32], which means a small number of states. These letters represent the possible secondary structure formations of each amino acid. Moreover, the sequence of states for each amino acid sequence produced by the model is known and that fact enables us to employ a low complexity training algorithm based on likelihood maximization [31]. So we can avoid the complicated Baum–Welch algorithm [33], which is an iterative learning process commonly used in other HMMs.

**Table 1**  
**Correspondence between letters of the DSSP alphabet and the letters of the DSSP-EHL alphabet**

DSSP	Type	Corresponding letter of the DSSP-EHL alphabet
H	Alpha-helix	H
G	$3_{10}$ -helix	H
I	$\Pi$ -helix	H
E	Extended( $\beta$ -strand)	E
B	Residue in isolated $\beta$ -bridge	E
T	Turn	L
S	Bend	L

All possible transitions between the states of the model are permitted. There is also one to one correspondence between the amino acids and the secondary structure formations in the training set, which means that for each state we should estimate the distribution over all possible amino acids. There are 21 possible amino acids which are the variables in each distribution. Among them, 20 correspond to the standard amino acids that exist and one more symbolizes amino acids of unknown origin. The total number of the model parameters is 203, which is the sum of  $7 \times 21$  parameters for the possible emissions,  $7 \times 7$  for the possible transitions between emission states and  $1 \times 7$  for the transitions from the begin state [30].

The likelihood maximization algorithm is used for the learning procedure of the 7-HMM. By implementing this algorithm, the emission and transition parameters are estimated in one step with the use of the maximum likelihood estimators. The estimators are calculated by the following equations, which are the estimation equations in the HMMs when the state sequences are known [31]:

$$t_{kl} = \frac{T_{kl}}{\sum_l T_{kl'}} \quad (1)$$

and

$$e_k(a) = \frac{E_k(a)}{\sum_{a'} E_k(a')}, \quad (2)$$

where  $t_{kl}$  is the transition probability from state  $k$  to another state  $l$  and  $e_k(a)$  the emission probability of the amino acid  $a$  in the state  $k$ . Also  $T_{kl}$  is the number of times that the transition from  $k$  to  $l$  takes place and  $E_k(a)$  is the number of times the emission of  $a$  from  $k$  takes place in the training set of sequences. Whenever there is a state  $k$  that has never been used in the training set, then the estimation equations are undefined for that state (numerator and denominator will be equal to zero). In order to avoid this problem it is better to add predetermined pseudocounts to the  $T_{kl}$  and  $E_k(a)$  before using Eqs. (1) and (2), thus we have:

$$T_{kl} = (\text{number of transitions } k \text{ to } l \text{ in training data}) + p_{kl}, \quad (3)$$

$$E_k(a) = (\text{number of emissions of } a \text{ from } k) + p_k(a). \quad (4)$$

The pseudocounts  $p_{kl}$  and  $p_k(a)$  should reflect our prior beliefs about the probability values. In our case we do not actually use any prior knowledge, which means that  $p_{kl} = p_k(a) = 1$ . This practically means that both the prior distribution of amino acids in each state and the prior distribution of transitions from each state are considered to be

the uniform distributions. Thus, the use of pseudocounts here simply aims to avoid definition problems and does not include the incorporation of some specific prior knowledge [30].

The posterior probability scores are used for assessing 7-HMM and they are otherwise called log-likelihood scores. These scores are logarithmic probabilities of a test sequence, given the model of each candidate fold. The test sequence is assigned to that fold whose model gives the maximum posterior probability score compared to the scores produced from all the other models of the candidate folds. The sequences used for the assessment are divided into training and test sets. The test set contains only amino acid sequences, as all other forms of data related to the structure of a protein are considered unknown. Moreover, the likelihood score for a sequence against a model is divided with the score of that sequence against the so-called *null* model. The *null* model is based on the assumption that the amino acids are statistically independent at each position. It also gives fixed emission probabilities based on the uniform distribution over the possible amino acids. Therefore, the log-likelihood score of a sequence against the null model is given as:

$$score(x_i) = \log_z \frac{P_m(x_i)}{P_\emptyset(x_i)}, \quad (5)$$

where  $P_m(x_i)$  corresponds to the probability that a sequence  $x_i$  has been produced by model  $m$  and  $P_\emptyset(x_i)$  to the probability that the same sequence has been produced by the *null* model. We have chosen this type of scoring in order to limit the effect of length variations among sequences. The criterion for selecting the fold category, where a particular protein most possibly belongs, is to select the model that gives the highest posterior probability score for that protein's amino acid sequence. Thus, the fold selected as the best classification for protein  $x_i$  would be model  $m_i$  for which  $P_{m_i}(x_i) > P_{m_j}(x_i)$  for all  $i \neq j$ , or equivalently  $score_{m_i}(x_i) > score_{m_j}(x_i)$ . The posterior probability scores correspond to the log-likelihood scores against the *null* model and are calculated with the use of the forward algorithm [31].

The forward algorithm is necessary for calculating the probability of a protein sequence  $x$  given the model, when we do not know the exact path  $\pi$  that produced the sequence. That probability is given by the following equation:

$$P(x) = \sum_{\pi} P(x, \pi). \quad (6)$$

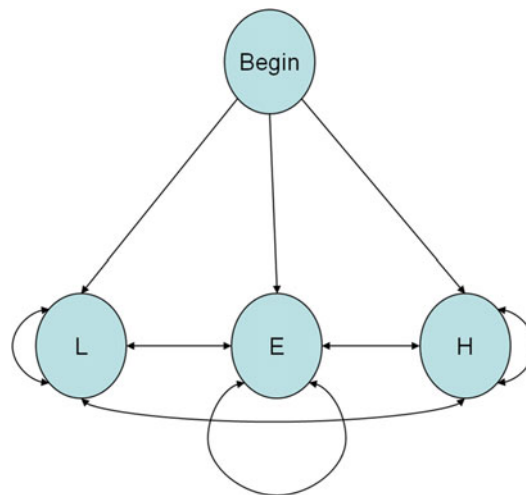
If the number of states is  $K$  and the length of the test sequence is  $N$ , the time complexity of the forward algorithm is  $O(K^2 N)$ . It is also necessary to use logarithms in order to avoid underflow problems, because such problems appear when there is the need to calculate a long product of probabilities.

### 3 3-HMM

In the 3-HMM version of the model we introduce specific improvements in the reduced state-space model which lead to a substantial increase in its ability to classify proteins in the correct fold category. In the first improvement we change the topology of the model while in the second the test proteins are scored against the 3-HMM in a different manner [34].

More specifically, in the first improvement we use a different alphabet in order to encode the secondary structure of the proteins. The different possible secondary structure formations of each amino acid are represented by the three-letter alphabet DSSP-EHL instead of the seven-letter DSSP alphabet. As it is shown in [21], the DSSP-EHL alphabet is a reduced representation of the DSSP alphabet. Specifically, H, G and I correspond to H, E and B to E, while T and S to L. Those amino acids which were considered of unknown structure in the DSSP representation are considered as loops (L) in the DSSP-EHL representation. The correspondence of letters between the two alphabets is shown in Table 1. So, now we employ three states in the model that correspond to the three different possible formations of the underlying secondary structure that each amino acid may have according to the DSSP-EHL alphabet (Fig. 2).

In the 3-HMM model there are  $3 \times 21$  emission parameters,  $3 \times 3$  transition parameters between the states and three parameters for the transitions from the starting state. Therefore, the total number of parameters is 75, which is much less than the 203 parameters which were used in the 7-HMM. In the first



**Fig. 2** Topology of the HMM with three states. Each state corresponds to one of the letters of the DSSP-EHL alphabet

improvement the training and scoring procedures of the 3-HMM model are the same with those of the 7-HMM.

In the second improvement, we maintain the model with the three states for training, but we use a different way of scoring the test proteins against the model. Our goal is to use also the secondary structure information of the test set proteins, which is not possible when we employ the forward algorithm. When the forward algorithm is used for scoring, the probabilities of all possible paths of the amino acid sequence through the model are added. Alternatively, we can compute the probability across the sequence of states (path) that corresponds to the secondary sequence of each protein. The use of the forward algorithm for scoring is necessary only when the secondary structure of the test protein is unknown. But if the secondary sequence is known, the path that produces the amino acid sequence through the model is also known, so the calculation of the score of each test protein becomes computationally less expensive. So this time we use the secondary sequence in scoring in addition to the primary sequence of test proteins and at the same time scoring becomes simpler [34].

Specifically, we avoid the iterative procedure of the forward algorithm in scoring each amino acid sequence. Here the path  $\pi^*$  of the amino acid sequence of the model is known, so the probability of the protein sequence  $x$  is:

$$P(x) = P(x, \pi^*). \quad (7)$$

Therefore, one step is needed for the calculations and the complexity depends only on the length of the sequence. Thus the time complexity is  $O(N)$ .

---

## 4 Optimized 3-HMM

We apply the following optimization approach to the scores of the 3-HMM. First, the scores of the model are normalized between 0 and 1 for each protein classification. Then, a set of parameters are added to the model, each one as a multiplier to the scores of each category (fold). These parameters are initially set to 1, which corresponds to the actual scores calculated using the 3-HMM. The parameters are subsequently introduced in the following optimization function:

$$F(D, p) = size(D) - trace(CM), \quad (8)$$

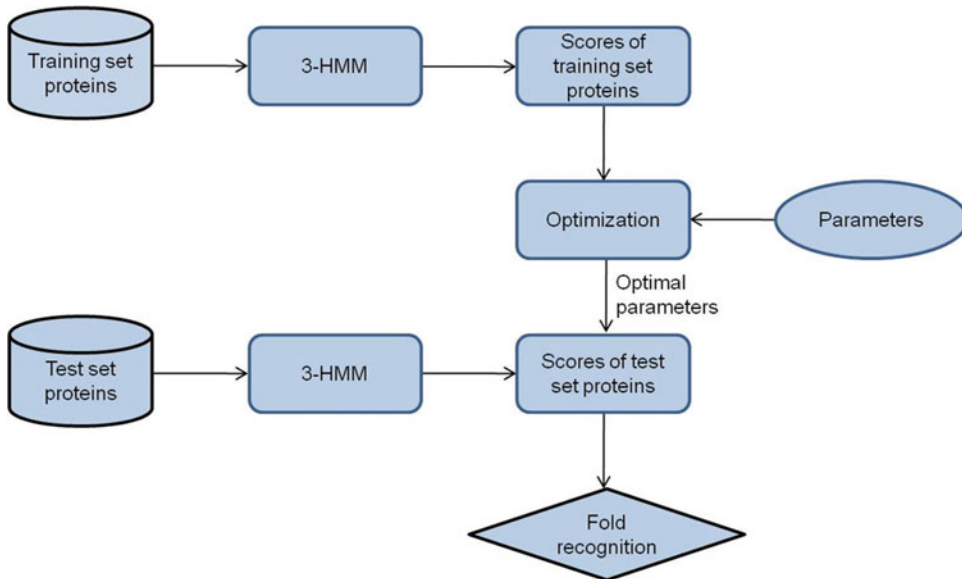
where  $D$  is the training dataset of proteins,  $p$  are the parameters,  $size(D)$  is the number of training proteins and  $trace(CM)$  is the correct predictions of the model, measured as the sum of the diagonal elements of the produced confusion matrix  $CM$  created for the training set  $D$ . Therefore Eq. (8) can be formulated as an



optimization problem. Different values for the parameters have an impact to the value of the optimization function, since they affect the confusion matrix. The result of the optimization procedure is the optimal set of parameters  $p^*$  [7].

The minimum value of  $F$  is 0 and this value is obtained if all proteins are correctly classified. Thus, the minimization of the above function increases the accuracy of the model by optimizing the parameters for the folds. The local optimization strategy was chosen, since an initial point is available (values of all parameters are equal to 1,  $p = 1$ ) while the analytical calculation of the derivatives of the objective function is computationally expensive. Based on the above, we employed the Nelder–Mead simplex search method [35] to solve the optimization problem. This is a direct search method for multidimensional unconstrained minimization which does not use numerical or analytic gradients.

The whole procedure is depicted in Fig. 3. The scores for each fold and for each protein of the training set were entered into the optimization procedure, along with the initial parameters (which are all set equal to 1). The optimization procedure identifies the optimal parameters, which are multiplied with the scores of the test set proteins against the 3-HMM. The final scores were used for the fold classification of the test proteins. It should be noted that the training proteins correspond to the primary and the true secondary structure of the proteins, while the test proteins correspond to the primary and the predicted secondary structure of the test proteins.



**Fig. 3** The optimization procedure applied to the 3-HMM for the identification of the optimal parameters for fold classification

## 5 Results and Discussion

### 5.1 7-HMM

The 7-HMM was assessed with the ASTRAL40 [36] 1.69 dataset, where only proteins with less than 40 % similarity are included. The primary and secondary structure information was extracted from the PDB database [37]. Moreover, the 34 most populated SCOP [38] folds (those with at least 30 proteins in this case) of the four major structural classes were used for the fold recognition experiment [30]. In total, 1513 proteins were used for training and 758 for testing.

Two experiments took place. In the first experiment, 34 reduced HMMs were trained for each one of the 34 most populated folds. The test sequences were scored against all folds of every class and the prediction accuracy was calculated for all folds. The prediction accuracy is the number of test proteins uniquely recognized as belonging to a specific fold divided by the total number of test proteins belonging to that fold. In the second experiment, 1513 reduced HMMs were trained for each one of the training sequences of all folds. In that case the prediction accuracy is calculated in the same way, but the test sequences are first scored against the models of all proteins of every fold. Then each protein was classified to the fold whose member was the protein the model of which gave the maximum probability score among all 1513 models of sequences used for training. The results of the experiments are shown in Table 2.

The 7-HMM performance deteriorates slightly when different models are trained for every sequence of the training set (it falls from 17.9 % to 16.1 %), indicating the ability of 7-HMM to learn families with fewer members. These performances were also compared with SAM, which is considered as the most effective method

**Table 2**

**Comparison of 7-HMM for the 34 SCOP folds when different models were trained for each fold and when different models were trained for each sequence in the training set**

Folds <sup>a</sup>	7-HMM accuracy (fold models)		7-HMM accuracy (sequence models)	
Overall class A	31/131	23.7 %	18/131	13.7 %
Overall class B	61/203	30.1 %	33/203	16.3 %
Overall class C	34/329	10.3 %	61/329	18.5 %
Overall class D	10/95	10.5 %	10/95	10.5 %
Overall	136/758	17.9 %	122/758	16.1 %

<sup>a</sup>A = all alpha proteins; B = all beta proteins; C = alpha and beta ( $a/b$ ) proteins (mainly parallel beta sheets); D = alpha and beta ( $a + b$ ) proteins (mainly antiparallel beta sheets)

that employs HMM for protein classification [20, 39]. In the first case the performance of SAM was 23.5 % while the second case it dropped to 11.9 % [30]. These results indicate that models like SAM, cannot perform adequately with insufficient data, in contrast to the proposed 7-HMM architecture.

## 5.2 3-HMM

The sequences used in the assessment of the improvements introduced with the 3-HMM come from the newer ASTRAL40 SCOP 1.71 dataset, where again no proteins with more than 40 % similarity are included [34]. Moreover, the SCOP folds of classes A, B, C, and D, and specifically those which have at least 30 members, were included. There were 38 such folds. In total, the training set contained 1727 proteins and the test set 864.

Predicted secondary structure information was also needed for our dataset. More specifically, it was necessary in the test set so that we would be able to score the test proteins against the model not only with their amino acid sequence but also using their secondary structure sequence. The predictions were obtained from PSIPRED [40], which provides reliability indices for all three secondary states {E,H,L} for each residue in the query sequence.

The 3-HMM model was compared with the 7-HMM model in order to assess the novelties introduced either in its structure (first improvement) or in the way of scoring (second improvement). Four experiments took place for this purpose using the same dataset which includes data from the 38 folds. The comparison among the results of those four experiments is shown in Table 3.

**Table 3**

**Comparison among the 7-HMM, the 3-HMM when only primary structure is used in scoring, the 3-HMM when predicted secondary structure is also used in scoring and the 3-HMM when true secondary structure is used for the 38 SCOP folds**

Folds	7-HMM	3-HMM (primary only) (o primary) recognition accuracy	3-HMM (predicted secondary)	3-HMM (true secondary)
Overall class A	39/140(27.9 %)	39/140(27.9 %)	54/140(38.6 %)	50/140(35.7 %)
Overall class B	70/243(28.8 %)	74/243(30.5 %)	94/243(38.7 %)	115/243 (47.3 %)
Overall class C	38/363(10.5 %)	53/363(14.6 %)	80/363(22 %)	97/363(26.7 %)
Overall class D	18/118(15.3 %)	24/118(20.3 %)	32/118(28 %)	38/118(32.2 %)
Overall	165/864 (19.1 %)	190/864(22 %)	260/864(30.1 %)	300/864 (34.7 %)

In the first experiment, we trained the 7-HMM for the 38 candidate folds. Then, the test set proteins were scored against all models of candidate folds and were assigned to that fold whose model gave the maximum posterior probability score. In the second experiment, we trained the 3-HMM for the 38 candidate folds, while the scoring and classification procedure remains the same. In both cases, forward algorithm was used for scoring only the primary sequences against the model.

In the third experiment, we trained again the 3-HMM for all candidate folds but we scored each test set protein against the model by taking into account its predicted secondary sequence. In the fourth experiment we replaced the predicted secondary sequences of the test set proteins with the correct ones given by PDB [37] and everything else remained the same as in the third one. In that way we could theoretically assess the maximum performance of the 3-HMM, given that the secondary structure predictor is 100 % accurate.

We can see that the 3-HMM outperforms 7-HMM in the same dataset as the performance increased by 2.9 % (from 19.1 % to 22 %). When we used also the predicted secondary structure in scoring, the fold classification performance was further improved, from 22 % to 30 %. Finally, when the correct secondary structure was used in scoring, the fold classification accuracy in the fourth experiment reached its maximum of 34.7 %.

### **5.3 Optimized 3-HMM**

The performance of the optimized 3-HMM was compared against the performance of the most effective version of 3-HMM, as well as against the latest version of SAM (3.5). As a result two experiments took place for this purpose.

The sequences employed in the experiments come from the newest ASTRAL40 SCOP 2.03 dataset, where we did not include proteins with more than 40 % similarity [7]. Moreover, the SCOP folds of classes A, B, C, D, and G, and specifically those which have at least 50 members, were included. We employed only the folds with more than 50 members so that there would be enough data for proper training and testing (especially in the case of SAM). This yielded a set of 42 such folds with 5024 sequences in total.

We also used tenfold cross validation for the evaluation procedure, so the sequences were separated ten times in training and test sets. Each time 90 % of the sequences of each fold were used as the training set of the fold and the rest 10 % were used as the test set. Furthermore, secondary structure predictions were needed for the whole dataset, so predictions were obtained from SymPsiPred [41]. SymPsiPred is considered as a more effective method for secondary structure prediction compared to PSIPRED which was used earlier for assessing the 3-HMM [7].

In the first experiment, we trained the improved version of the 3-HMM for the 42 candidate folds, using both primary and true

secondary structures of the proteins in the training set. Then the test set proteins were scored against all models of candidate folds and were assigned to that fold for which the model gives the maximum posterior probability score. The test set proteins were scored against the 3-HMM by employing not only their primary structure but also their predicted secondary structure. Overall, the 3-HMM achieves a 37.8 % accuracy.

In the second experiment, we employed the optimization method described above (Subheading 4) for the identification of the optimal parameters for the 3-HMM. When optimization was performed to the 3-HMM, the test set accuracy of the model increased by almost 10 % (from 37.8 % to 41.4 %). Moreover, we compared the above results with the classification results of SAM, which was applied in the same dataset. The classification accuracy of SAM reached 38 % which was approximately 8 % lower than the accuracy obtained by the optimized 3-HMM. Results for the above procedure for both experiments in the dataset are shown in Table 4.

Finally, in order to evaluate the robustness of the optimized 3-HMM, the receiver operating characteristics (ROCReceiver operating characteristics (ROC)) analysis was performed following the class reference formulation, where each class (fold) is considered separately against all others [7]. The attained area under the curve (AUC) was 0.88, indicating the robustness of our method. It is worth mentioning that when ROC analysis was performed for the 7-HMM, the AUC was found to be 0.76 when different models were trained for each fold and 0.73 when different models were trained for each sequence [30].

**Table 4**  
**Classification results for the two experiments**

<b>Folds</b>	<b>3-HMM (%)</b>	<b>3-HMM optimized (%)</b>	<b>SAM 3.5 (%)</b>
Overall class A	39.3	44.9	31.5
Overall class B	48.3	50.0	33.0
Overall class C	32.7	37.2	42.9
Overall class D	28.4	31.1	41.8
Overall class G	75.6	76.2	50.0
Overall	37.8	41.4	38.0

In the first column the classes that correspond to the folds used are shown. The results from the 3-HMM when it is used alone and when its scores are optimized are shown in the second and third column, respectively. The classification results of SAM 3.5 are shown in the fourth column

## 6 Conclusions

In this chapter, we discuss methods for building HMMs with reduced number of states for predicting the fold of a protein given its sequence and assess the performances of different variations of these HMMs including 7-HMM, 3-HMM, and optimized 3-HMM. 7-HMM and optimized 3-HMM were also compared with the latest version of SAM, a widely used HMM-based method. 3-HMM was more effective than 7-HMM and optimized 3-HMM achieved the maximum performance. The results indicate that the variations of the model were equally or more effective than SAM while they use a small architecture and require less data for training. Moreover, all of them use a low complexity training algorithm while the 3-HMM also uses a testing algorithm less complex than the forward algorithm which is used in the 7-HMM. Other possible improvements that could be incorporated to the context of the proposed reduced state-space HMM architecture may exploit the use of more structural features, apart from the secondary structure. The aim would be to further enhance the classification efficiency of the model and at the same time maintain its low size and complexity.

## References

1. Whitford D (2005) *Proteins: structure and function*. John Wiley & Sons, NJ, USA
2. Lee SY, Lee JY, Jung KS, Ryu KH (2009) A 9-state hidden Markov model using protein secondary structure information for protein fold recognition. *Comp Biol Med* 39(6):527–534
3. Camproux A, Guyon F, Gautier R, Laffray J, Tuffery P (2005) A hidden Markov model applied to the analysis of protein 3D-structures. in: *Proc. int. symp. applied stochastic models and data analysis*
4. Orengo CA, Jones DT, Thornton JM (2003) *Bioinformatics: genes, proteins and computers*. Bios Scientific Pub. Ltd, Oxford
5. Zhang Y, Skolnick J (2005) The protein structure prediction problem could be solved using the current PDB library. *Proc Natl Acad Sci U S A* 102(4):1029–1034
6. Hargbo J, Elofsson A (1999) Hidden Markov models that use predicted secondary structures for fold recognition. *Proteins* 36(1):68–76
7. Lampros C, Simos T, Exarchos TP, Exarchos KP, Papaloukas C, Fotiadis DI (2014) Assessment of optimized Markov models in protein fold classification. *J Bioinform Comput Biol* 12(4):1450016
8. Murzin AG (1999) Structure classification based assessment of CASP3 predictions for the fold recognition targets. *Proteins (Suppl 3)*:88–103
9. Orengo CA, Bray JE, Hubbard T, LoConte L, Sillitoe I (1999) Analysis and assessment of ab initio three-dimensional prediction, secondary structure, and contacts prediction. *Proteins* 37:149–170
10. Zhang Y (2008) Progress and challenges in protein structure prediction. *Curr Opin Struct Biol* 18(3):342–348
11. Zhou Y, Duan Y, Yang Y, Farragi E, Lei H (2011) Trends in template/fragment-free protein structure prediction. *Theor Chem Acc* 128(1):3–16
12. Maurice KJ et al (2014) SSThread: template-free protein structure prediction by threading pairs of contacting secondary structures followed by assembly of overlapping pairs. *J Comput Chem* 35(8):644–656
13. Bowie JU, Luthy R, Eisenberg D (1991) A method to identify protein sequence that fold into a known three-dimensional structure. *Science* 253:164–170

14. Flockner H, Domingues F, Sippl MJ (1997) Proteins folds from pair interactions: a blind test into fold recognition. *Proteins* 1:129–133
15. Xu J (2005) Fold recognition by predicted alignment accuracy. *IEEE/ACM Trans Comput Biol Bioinform* 2(2):157–165
16. Sander O, Sommer I, Lengauer T (2006) Local protein structure prediction using discriminative models. *BMC Bioinformatics* (7):14
17. Hu Y, Dong X, Wu A, Cao Y, Tian L, Jiang T (2011) Incorporation of local structural preference potential improves fold recognition. *PLoS One* 6(2):e17215
18. Mahajan S, De Brevern AG, Sanejouand YH, Srinivasan N, Offmann B (2015) Use of a structural alphabet to find compatible folds for amino acid sequences. *Protein Sci* 24(1):145–153
19. Finn RD, Clements J, Eddy SR (2011) HMMER web server: interactive sequence similarity searching. *Nucleic Acids Res* 39(Suppl 2):W29–W37
20. Karplus K, Karchin R, Shackelford G, Hughey R (2005) Calibrating E-values for hidden Markov models using reverse-sequence null models. *Bioinformatics* 21:4107–4115
21. Karchin R, Cline M, Mandel-Gutfreund Y, Karplus K (2003) Hidden Markov models that use predicted local structure for fold recognition: alphabets of backbone geometry. *Proteins* 51:504–514
22. Dandekar T, Argos P (1996) Identifying the tertiary fold of small proteins with different topologies from sequence and secondary structure using the genetic algorithm and extended criteria specific for strand regions. *J Mol Biol* 256:645–660
23. Zangoei MH, Jalili S (2013) Protein fold recognition with a two-layer method based on SVM-SA, WP-NN and C4. 5 (TLM-SNC). *Int J Data Mining Bioinform* 8(2):203–223
24. Deschavanne P, Tuffery P (2009) Enhanced protein fold recognition using a structural alphabet. *Proteins* 76:129–137
25. Chmielnicki W, Stapor K (2012) A hybrid discriminative/generative approach to protein fold recognition. *Neurocomputing* 75(1):194–198
26. Exarchos TP, Papaloukas C, Lampros C, Fotiadis DI (2008) Mining sequential patterns for protein fold recognition. *J Biomed Inform* 41(1):165–179
27. Tsai CY, Chen CJ, (2015) A PSOAB classifier for solving sequence classification problems. *Appl Soft Comput* 27(C):11–27
28. Valavanis I, Spyrou G, Nikita K (2010) A similarity network approach for the analysis and comparison of protein sequence/structure sets. *J Biomed Inform* 43(2):257–267
29. Abbasi E, Mehdi G, Shiri ME (2013) FRAN and RBF-PSO as two components of a hyper framework to recognize protein folds. *Comput Biol Med* 43(9):1182–1191
30. Lampros C, Papaloukas C, Exarchos TP, Goletsis Y, Fotiadis DI (2007) Sequence-based protein structure prediction using a reduced state-space hidden Markov model. *Comput Biol Med* 37:1211–1224
31. Durbin R, Eddy S, Krogh A, Mitchison G (1998) Biological sequence analysis: probabilistic models of proteins and nucleic acids. Cambridge University Press, New York
32. Kabsch W, Sander C (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* 22:2577–2637
33. Baum LE (1972) An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. *Inequalities* 3:1–8
34. Lampros C, Papaloukas C, Exarchos K, Fotiadis DI, Tsalikakis D (2009) Improving the protein fold recognition accuracy of a reduced state-space hidden Markov model. *Comput Biol Med* 39:907–914
35. Lagarias JC, Reeds JA, Wright MH, Wright PE (1998) Convergence properties of the Nelder-Mead simplex method in low dimensions. *SIAM J Optim* 9(1):112–147
36. Chandonia JM, Hon G, Walker NS, Lo Conte L, Koehl P, Levitt M, Brenner SE (2004) The ASTRAL compendium in 2004. *Nucleic Acids Res* 32(Database issue):D189–D192
37. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE (2000) The protein data bank. *Nucleic Acids Res* 28:235–242
38. Andreeva A, Howorth D, Brenner SE, Hubbard TJ, Chothia C, Murzin AG (2004) SCOP database in 2004: refinements integrate structure and sequence family data. *Nucleic Acids Res* 32(Database issue):D226–D229
39. Machado-Lima A, Kashiwabara AY, Durham AM (2010) Decreasing the number of false positives in sequence classification. *BMC Genomics* 22(11 Suppl 5):S10
40. Jones DT (1999) Protein secondary structure prediction based on position specific scoring matrices. *J Mol Biol* 292:195–202
41. Lin HN, Sung TY, Ho SY, Hsu WL (2010) Improving protein secondary structure prediction based on short subsequences with local structure similarity. *BMC Genomics* 2(Suppl 4):S4

Hidden Markov Models

Methods and Protocols

Westhead, D.; Vijayabaskar, M.S. (Eds.)

2017, X, 221 p. 59 illus., 17 illus. in color., Hardcover

ISBN: 978-1-4939-6751-3

A product of Humana Press