

# Predicting Moonlighting Proteins from Protein Sequence

Jing Hu\* and Yihang Du

*Department of Computer Science, Franklin & Marshall College, Lancaster, PA, U.S.A.*

**Keywords:** K-Nearest Neighbors Method, Bit-Score Weighted Euclidean Distance, Feature Selection, Moonlighting Proteins.


**Abstract:** High-throughput proteomics projects have resulted in a rapid accumulation of protein sequences in public databases. For the majority of these proteins, limited functional information has been known so far. Moonlighting proteins (MPs) are a class of proteins which perform at least two physiologically relevant distinct biochemical or biophysical functions. These proteins play important functional roles in enzymatic catalysis process, signal transduction, cellular regulation, and biological pathways. However, it has been proven to be difficult, time-consuming, and expensive to identify MPs experimentally. Therefore, computational approaches which can predict MPs are needed. In this study, we present MPKNN, a K-nearest neighbors method which can identify MPs with high efficiency and accuracy. The method is based on the bit-score weighted Euclidean distance, which is calculated from selected features derived from protein sequence. On a benchmark dataset, our method achieved 83% overall accuracy, 0.64 MCC, 0.87 F-measure, and 0.86 AUC.

## 1 INTRODUCTION

Due to high throughput sequencing technologies, huge number of protein sequences have been accumulated in public databases, waiting to be analyzed. The study of protein functions is important for the understanding of the cellular mechanism and biological pathways of organisms. Most proteins as we know only exhibit single function. However, there are a class of proteins named moonlighting proteins (MPs) in which a single protein performs multiple physiologically relevant distinct biochemical or biophysical functions that are not result of gene fusions, alternative RNA splicing, multiple domains, DNA rearrangement, or proteolytic fragments (Jeffery, 2015; Jeffery, 2018; Weaver, 1998; Jain, 2018; Shirafkan, 2021). Since the first MP was detected in 1980s (Piatigorsky, 1989), scientists have found MPs in many types of species, including bacteria, archaea, mammals, reptiles, birds, fish, worms, insects, plants, fungi, protozoans and even viruses (Jeffery, 2018). The datasets of these proteins are stored in public databases such as MoonProt (Chen, 2018), MultitaskProtDB-II (Franco-Serrano, 2018), and MoonDB (Ribeiro, 2019). Recent studies

have found that a large number of MPs play important roles in diseases, infection, virulence, or immune responses, and some can be potential vaccination targets (Jeffery, 2015). Therefore, identification and study of MPs is crucial for research on diseases and drug-target discovery.

However, it is time-consuming and expensive to identify MPs experimentally in laboratory. Therefore, computational methods which can predict MPs with high performance are in urgent need. As to our knowledge, there are only a limited number of methods published so far. Chapple et al. has developed MoonGO (Chapple, 2015), a method based on features derived from protein-protein interaction networks and gene ontology (GO) information to predict MPs. MPFit (Khan, 2016) is a machine learning method which utilizes features derived from gene ontology (GO), protein-protein interactions, gene expression, phylogenetic profiles, genetic interactions and network-based graph properties to protein structural properties to predict MPs. Although they reported a high accuracy of 98% in identifying MPs when GO information was incorporated, the prediction performance was degraded dramatically to 75% accuracy when GO

\* <https://www.fandm.edu/jing-hu>

annotations were removed. Jain et al. proposed a text mining method called DextMP to predict MPs from several different information sources, database entries, literature, and large-scale omics data (Jain, 2018). However, the lack of GO annotations and literature-based information for the majority of proteins restricted the application of the methods mentioned above. Recently, Shirafkan et al. attempted to predict MPs using features extracted only from protein sequences (Shirafkan, 2021). After assessing 8 different machine learning methods with each of 37 distinct feature vectors to detect moonlighting proteins, they found that the Support Vector Machine (SVM) model based on the split amino acid composition (SAAC) feature set has the highest accuracy of 77% in classifying between MPs and non-MPs

In this paper, we propose MPKNN, a K-nearest neighbors (KNN) method to predict MPs. Our method relies on the bit-score weighted Euclidean distance, which is calculated from selected features including compositions of certain amino acids, extended pseudo-amino acids (Chou, 2003) (Du, 2006), and FECS features (Mu, 2021). The final method achieved 83% overall accuracy, 0.64 MCC, 0.87 F-measure, and 0.86 AUC on a benchmark dataset used in previous study (Shirafkan, 2021).

## 2 METHODS AND MATERIAL

### 2.1 Dataset

We used the benchmark dataset that was used in the previous study (Shirafkan, 2021). In their study, all moonlighting proteins were experimentally verified and collected from MoonProt (Chen, 2018) database. After removal of sequence redundancy using CD-hit (Fu, 2012) with a 40% mutual sequence similarity threshold, there were 315 proteins left in the final dataset, among which 215 were moonlighting proteins and 136 were non-moonlighting proteins from species including *Mus Musculus*, *Human*, *E. coli*, *Yeast*, *Rat*, *Drome*, *Arath*, and others.

### 2.2 Feature Extraction

We have investigated feature sets such as amino acid composition, extended pseudo-amino acid composition (Chou, 2003) (Du, 2006), and FECS features (Mu, 2021) in this study.

#### 2.2.1 Amino Acid Composition

The amino acid composition of a protein sequence was calculated by

$$x_i = n_i / \sum_{j=1}^{20} n_j \quad (1)$$

where  $n_i$  and  $n_j$  are the numbers of amino acid  $i$  and  $j$  in the protein sequence.

#### 2.2.2 Extended Pseudo-Amino Acid Composition

The original model of pseudo-amino acid composition (Chou, 2003) consists of compositions of 20 amino acids in a protein and  $\lambda$  different ranks of sequence-order correlation factors (i.e., delta function set). In this study, we extended the definition of pseudo-amino acid composition by including 9 more sets of various physicochemical properties that were investigated in the previous study of Du and Li (Du, 2006).

In this study, the delta function set was calculated as in (Chou, 2003). Suppose a protein  $X$  with a sequence of  $L$  amino acid residues:  $R_1 R_2 \dots R_i \dots R_L$ , where  $R_i$  represents the amino acid at sequence position  $i$ . The first set, delta-function set, consisted of  $\lambda$  sequence-order-correlated factors, which were given by

$$\delta_i = \frac{1}{L-i} \sum_{j=1}^{L-i} \Delta_{j,j+i} \quad (2)$$

where  $i = 1, 2, 3 \dots \lambda, \lambda < L$  and  $\Delta_{j,j+i} = \Delta(R_j, R_{j+i}) = 1$  if  $R_j = R_{j+i}$ , 0 otherwise. These features were named as  $\{\delta_1, \delta_2, \dots, \delta_\lambda\}$ .

The remaining 9 sets of physicochemical properties were based on AAindex values (Kawashima, 2000). Similar as (Du, 2006), the following AAindex indices were used: BULH740101 (transfer free energy to surface), EISD840101 (consensus normalized hydrophobicity), HOPT810101 (hydrophilicity value), RADA880108 (mean polarity), ZIMJ680104 (isoelectric point), MCMT640101 (refractivity), BHAR880101 (average flexibility indices), CHOC750101 (average volume of buried residue), COSI940101 (electron-ion interaction potential values). For each of 9 AAindex indices, we obtained  $\mu$  sequence-order-correlated factors by

$$h_i = \frac{1}{L-i} \sum_{j=1}^{L-i} H_{j,j+i} \quad (3)$$

where  $i = 1, 2, 3, \dots, \mu, \mu < L$ , and  $H_{i,j} = H(R_i) \cdot H(R_j)$ . In this study,  $H(R_i)$  and  $H(R_j)$  are the normalized AAindex values of residues  $R_i$  and  $R_j$  respectively. The normalized AAindex value of each amino acid was calculated by applying

$$H(AA_i) = \frac{H^o(AA_i) - \overline{H^o}}{\sqrt{\{\sum_{j=1}^{20} (H^o(AA_j) - \overline{H^o})^2\} / 20}} \quad (4)$$

where  $i = 1, 2, 3, \dots, 20$ .  $H^o(AA_i)$  is the original AAindex value of amino acid  $i$ , and  $\overline{H^o}$  is the average AAindex value of 20 amino acids. For each of 9 AAindex types (i.e., BULH740101, EISD840101, etc.), we obtained  $\mu$  features using (3) and (4). In total there are  $9\mu$  features. We named these features as {BULH740101\_1, BULH740101\_2, ..., BULH740101\_μ, EISD840101\_1, EISD840101\_2, ..., EISD840101\_μ, ..., COSI940101\_1, COSI940101\_2, ..., COSI940101\_μ}. Therefore, the pseudo-amino acid compositions consist of  $\lambda$  (delta-function factors) +  $9\mu$  (9 sets of physicochemical factors) numbers. In this study, both  $\lambda$  and  $\mu$  were set to 10.

### 2.2.3 FECS Features

FECS (Feature Extraction based on Graphical and Statistical features) is a protein feature extraction model by integrating the graphical representation of protein sequences based on the physicochemical properties of amino acids and statistical features of the protein sequences (Mu, 2021). Using the MATLAB script provided by the authors, a set of 578 features was extracted for each protein sequence in our dataset.

In total, we have investigated 698 sequence-derived features (i.e., 20 amino acid compositions, 10 delta-function factors, 9 sets of physicochemical factors with 10 features in each set, and 578 FECS features).

## 2.3 Bit-Score Weighted Euclidean Distance

For each query protein  $t$ , its distance to a training protein  $T$  is calculated as

$$D_{t,T} = \sqrt{\sum_{i=1}^N (t_i - T_i)^2} / BS(t, T) \quad (5)$$

where  $t_i$  and  $T_i$  are the  $i^{th}$  features (i.e., amino acid composition, pseudo-amino acid composition, FECS features) of the query protein  $t$  and the training protein  $T$  respectively, and  $N$  is the number of features used.  $BS(t, T)$  is the bit score computed by the *blastp* program of Blast package (Altschul, 1997) when comparing the local sequence similarity between protein sequences  $t$  and  $T$ . It is a normalized score which is independent of query sequence length and database. A higher bit score indicates two protein sequences are more similar, and vice versa. Notice that  $\sqrt{\sum_{i=1}^N (t_i - T_i)^2}$  gives the Euclidean distance between two proteins. Here, the distance is weighted by a factor (i.e., bit score). Therefore, the distance is referred to as the bit-score weighted Euclidean distance (BS-WED). As can be seen from the equation (5), the more similar the corresponding features, the higher the sequence similarity between two proteins, the smaller the distance (as measured by BS-WED).

## 2.4 K-Nearest Neighbors Method

A traditional K-nearest neighbors method (KNN) method classifies the query sample by the majority voting strategy. For each query sample, KNN finds its  $k$  nearest neighbors in the training dataset, and then assigns it to the class to which most of its neighbors belong. The KNN method used in this study differs in that it finds  $k$  nearest neighbors to the query sample from each class of training samples. The average of these  $k$  distances is calculated as the *adjacency value* of the query sample to that class. The adjacency values between the query sample and all classes are compared and the query sample is assigned to the class to which it has the smallest adjacency value. Specifically, for each test protein, its distance (measured by BS-WED) to every moonlighting protein in the training set was calculated using (5). The  $k$  shortest BS-WEDs were chosen, namely  $d_{mp-1}, d_{mp-2}, \dots, d_{mp-k}$ . Then, the adjacency value between the test protein and the MP class (denoted as  $A_{mp}$ ) was given by

$$A_{mp} = \sum_{i=1}^k d_{mp-i} / k \quad (6)$$

The adjacency value between the query protein and non-MP class ( $A_{non-mp}$ ) was calculated in a similar way. Then the query protein was predicted as moonlighting protein if  $A_{mp} / A_{non-mp} < \theta$ ; non-moonlighting protein otherwise. The default value of  $\theta$  was set to 1.

## 2.5 Performance Measurement

Ten-fold cross-validation was used to evaluate the performance of the algorithm. In this study, MPs were treated as the positive samples, while non-MPs were treated as the negative samples.

Performance was measured using recall, precision, Acc (overall accuracy), MCC (Matthews Correlation Coefficient), F-Measure, and AUC (area under the ROC curve). F-measure and MCC are balanced measurements of the performance on both positive and negative samples. AUC is the area under the ROC curve when adjusting true positive rate vs. false positive rate by tuning the parameter  $\theta$ .

$$Recall = TP / (TP + FP) \quad (7)$$

$$Precision = TP / (TP + FN) \quad (8)$$

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

$$F - Measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (10)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (11)$$

## 2.6 Heuristic Feature Selection Process

For each protein, there were 20 (classic amino acid compositions) +  $\lambda$  (delta-function factors) +  $9\mu$  (9 sets of physicochemical factors) + 578 (FEGS) extracted features. In this study,  $\lambda$  and  $\mu$  were both set to 10. Therefore, there were 698 features investigated in total. To identify the most useful subset of features, a previously described greedy feature selection algorithm by our group (Hu, 2012) was employed. The greedy search was based on the wrapper approach and it started with a feature set that included 20 amino acids. Let  $n$  be the size of the feature set. Then  $n = 20$  at the beginning. The algorithm could be divided into two stages: reduction and growth. In the reduction stage, the size of the feature set was gradually reduced. First, one amino acid was removed and the composition of the remaining  $n-1$  amino acids were used to calculate WEDs. Ten-fold cross-validation was used to evaluate the performance of the method. This step was repeated  $n$  times, so that every combination of  $n-1$  amino acids was tried. The combination that improved the performance most was chosen. Thus, the size of the feature set was reduced from  $n$  to  $n-1$ . This reduction process was continued until removing any amino acid from the feature set would reduce the performance. At the end of the reduction stage, we reached a feature set that included the composition of  $N$  amino acids ( $N \leq 20$ ). Then,

we used a growth stage to increase the size of the feature set by adding pseudo-amino acid compositions. One pseudo-amino acid composition was added at a time and the resulting feature set was used to calculate WEDs. Ten-fold cross-validation was used to evaluate the performance of the method. The pseudo-amino acid composition that brought the biggest improvement in performance was chosen and added into the feature set. Thus, the size of the feature set was increased to  $N+1$ . This growth process continued until adding any more pseudo-amino acid composition would decrease the performance. Then we began the process of adding FEGS feature one at a time until adding any more FEGS feature into the set would decrease the prediction performance. In the end, we obtained a set of 18 features that include compositions of 12 amino acids {A, N, D, C, E, G, I, L, K, F, S, T}, 4 delta-function factors  $\{\delta_1, \delta_3, \delta_5, \delta_7\}$ , and 2 physicochemical factors {EISD840101\_2 and HOPT810101\_10}.

## 3 RESULTS

The proposed MPKNN method was first used to classify between MPs and non-MPs using only the compositions of 20 amino acids. We then assessed the prediction ability of each feature set individually and also that of all combined features. Finally, we seek to improve the prediction performance by applying the heuristic feature selection process mentioned above to choose the more relevant features. Our final method was also compared with previously published method (Shirafkan, 2021) on the same benchmark dataset.

Table 1: Prediction performance of each feature set.

Methods	Acc	MCC
Amino acid compositions	80.3%	0.579
delta-function factors	73.8%	0.432
BULH740101	76.4%	0.490
EISD840101	80.1%	0.572
HOPT810101	77.5%	0.522
RADA880108	77.8%	0.523
ZIMJ680104	73.5%	0.429
MCMT640101	72.1%	0.397
BHAR880101	70.7%	0.360
CHOC750101	72.6%	0.412
COSI940101	71.2%	0.374
EGS features	79.5%	0.560
All features combined	78.3%	0.535



### 3.1 Predicting MPS Using Only Amino Acid Composition

First only compositions of 20 amino acids were used to calculate the distance (i.e., BS-WED) of each test protein to each training protein. Ten-fold cross-validation was used to evaluate the performance on the dataset. The parameter  $\theta$  was set to 1. Various  $k$  values ranging from 1 to 30 were tried. For comparison, we also searched for the best performance of traditional KNN method using the standard Euclidean distance for various  $k$  values in the same range (i.e., 1-30). As can be seen from Table 1 (row 2), MPKNN achieved the performance of 80.3% accuracy and 0.579 MCC when  $k = 20$ . In comparison, standard KNN has the most optimal performance of 76.4% accuracy and 0.490 MCC when  $k = 19$ . Therefore, it is clear that BS-WED is a better distance measurement than standard Euclidean distance when measuring the relationship between the query protein and the training proteins (i.e., MPs or non-MPs). The detailed prediction performance of MPKNN using only compositions of 20 amino acids is shown in Table 2 (Column 2).

### 3.2 Prediction Performance of Each Feature Set

The proposed KNN method based on BS-WED was tested on each feature set individually to assess their usefulness in predicting MPs. The parameter  $\theta$  was set to the default value (i.e., 1). Various  $k$  values ranging from 1 to 30 were tried and the best performance was kept. We also evaluated the prediction performance of all features combined. The results are listed in Table 1. It is obvious that the prediction performance couldn't be further improved by simply combining all features investigated in this study. This could be due to the fact that not all features were useful for the prediction. Also, some features might be correlated with each other, which could impair the prediction performance.

Table 2: Comparison of different methods.

Method	MPKNN +20 AAs	MPKNN + 18 selected features	Shirafkan et al.'s method
Recall	91.6%	91.6%	-
Precision	79.4%	<b>82.4%</b>	74%
Acc	80.3%	<b>82.9%</b>	77%
MCC	0.579	0.635	-
F-Measure	0.851	0.868	0.75
AUC	0.750	<b>0.862</b>	0.75

### 3.3 Performance After Feature Selection

We then seek to improve the prediction performance by applying the heuristic feature selection process as described in Methods and Materials to search for a subset of features that was (almost) most useful for the prediction. In the end a set of 18 features that include compositions of 12 amino acids {A, N, D, C, E, G, I, L, K, F, S, T}, 4 delta-function factors  $\{\delta_1, \delta_3, \delta_5, \delta_7\}$ , and 2 physicochemical factors {EISD840101\_2 and HOPT810101\_10} were chosen. Adding more features did not improve the prediction performance. As can be seen from Table 2 (Column 3), using selected features, MPKNN improved its performance to 91.6% recall, 82.4% precision, 82.9% accuracy, 0.635 MCC, 0.868 F-measure, and 0.862 AUC.

We also compared our final MPKNN (i.e., using 18 selected features) with previously published method by (Shirafkan, 2021) on the same benchmark dataset. The prediction performance of Shirafkan et al.'s method was directly obtained from their report (as shown in Table 2, Column 4). Table 2 clearly shows that our method has achieved far more superior performance than that of Shirafkan et al.

## 4 CONCLUSIONS

In this study we present MPKNN, a KNN method which can predict MPs with 91.6% recall, 82.4% precision, 82.9% accuracy, 0.635 MCC, 0.868 F-measure, and 0.862 AUC. The method is based on a bit-score weighted Euclidean distance (BS-WED) to measure the similarity between proteins. Compared to the standard Euclidean distance, BS-WED takes account of both compositions and sequence similarity.

The benchmark dataset used in this study was relatively small, and therefore feature selection (wrapper method) and cross-validation were performed on the same dataset to avoid insufficient training. To better estimate the generalization ability of our method, it would be preferred to carry out these two processes on two separate non-overlapping datasets. For the future work, we plan to curate a more comprehensive dataset with a larger number of MP and non-MP proteins from various protein databases so that we may split the dataset into two parts, with the first part reserved for feature selection and the second part for cross-validation (i.e., training and test) using the selected features. We also plan to investigate the possibility of improving the prediction

performance by including other features such as evolutionary features (PSSM profiles.), predicted structural information of each protein, feature vectors calculated by the ftrCOOL library (Amerifar, 2020), etc., and applying our method to identify potential MPs in proteomes of human and other species.

## ACKNOWLEDGEMENTS

This work was partially supported by the FR/PDF and Hackman grant from Franklin & Marshall College.

## REFERENCES

- Altschul, S., et al. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, vol. 25, pp. 3389-3402. doi: 10.1093/nar/25.17.3389.
- Amerifar, S. and Zahiri, J. (2020). ftrCOOL: Feature extraction from biological sequences. CRAN - Package ftrCOOL - R Project.
- Chapple, C., et al. (2015). Extreme multifunctional proteins identified from a human protein interaction network. *Nat Commun.*, 6:7412. doi: 10.1038/ncomms8412.
- Chen, C., et al. (2018). MoonProt 2.0: an expansion and update of the moonlighting proteins database. *Nucleic Acids Res.*, 46(D1): D640-D644. doi: 10.1093/nar/gkx1043.
- Chou, K. and Cai, Y. (2003). Prediction and classification of protein subcellular location-sequence-order effect and pseudo amino acid composition. *J. Cell Biochem.*, vol. 90, pp. 1250-1260. doi: 10.1002/jcb.10719.
- Du, P. and Li, Y. (2006). Prediction of protein submitochondria locations by hybridizing pseudo-amino acid composition with various physicochemical features of segmented sequence. *BMC bioinformatics*, vol. 7 518. 30. doi:10.1186/1471-2105-7-518.
- Franco-Serrano, L., et al. (2018) MultitaskProtDB-II: an update of a database of multitasking/moonlighting proteins. *Nucleic Acids Res.*, 46(D1): D645-D648. doi: 10.1093/nar/gkx1066.
- Fu, L., et al. (2012). CD-HIT: accelerated for clustering the next generation sequencing data. *Bioinformatics*, 28 (23): 3150-3152. doi: 10.1093/bioinformatics/bts565.
- Hu, J. and Ng, P. (2012). Predicting the effects of frameshifting indels. *Genome Biol.*, vol. 9, pp. R9. DOI: 10.1186/gb-2012-13-2-r9.
- Jain, A., Gali, H., and Kihara, D. (2018). Identification of Moonlighting Proteins in Genomes Using Text Mining Techniques. *Proteomics*, vol. 18, pp. 21-22. doi:10.1002/pmic.201800083.
- Jeffery, C. (2015). Why study moonlighting proteins? *Front Genet.* 6:211. doi: 10.3389/fgene.2015.00211.
- Jeffery, C. (2018). Protein moonlighting: what is it, and why is it important? *Philos Trans R Soc Lond B Biol Sci.*, 373 (1738): 20160523. doi: 10.1098/rstb.2016.0523.
- Kawashima, S. and Kanehisa, M. (2000). AAindex: amino acid index database. *Nucleic Acids Res.*, vol. 28, pp. 374. doi: 10.1093/nar/28.1.374.
- Khan, I., et al. (2016). Genome-scale prediction of moonlighting proteins using diverse protein association information. *Bioinformatics*, 32(15): 2281-2288. doi:10.1093/bioinformatics/btw166.
- Mu, Z., et al. (2021). FEGS: a novel feature extraction model for protein sequences and its applications. *BMC bioinformatics*, vol. 22,1 297. doi:10.1186/s12859-021-04223-3.
- Piatigorsky, J., Wistow, G. (1989). Enzyme/crystallins: gene sharing as an evolutionary strategy. *Cell*, 57(2): 197-9. doi: 10.1016/0092-8674(89)90956-2.
- Ribeiro, D., et al. (2019) MoonDB 2.0: an updated database of extreme multifunctional and moonlighting proteins. *Nucleic Acids Res.*, 47(D1): D398-D402. doi: 10.1093/nar/gky1039.
- Shirafkan, F., et al. (2021). Moonlighting protein prediction using physico-chemical and evolutionary properties via machine learning methods. *BMC bioinformatics* vol. 22,1 261. 24. doi:10.1186/s12859-021-04194-5.
- Weaver, D. (1998). Telomeres: moonlighting by DNA repair proteins. *Curr. Biol.* vol. 8, R492-R494. doi: 10.1016/S0960-9822(98)70315-X.