A New Method for Predicting the Subcellular Localization of Eukaryotic Proteins with Both Single and Multiple Sites: Euk-mPLoc 2.0

Kuo-Chen Chou[1,2]*, **Hong-Bin Shen**[1,2]

[1] Gordon Life Science Institute, San Diego, California, United States of America, [2] Institute of Image Processing & Pattern Recognition, Shanghai Jiaotong University, Shanghai, China

Abstract

Information of subcellular locations of proteins is important for in-depth studies of cell biology. It is very useful for proteomics, system biology and drug development as well. However, most existing methods for predicting protein subcellular location can only cover 5 to 12 location sites. Also, they are limited to deal with single-location proteins and hence failed to work for multiplex proteins, which can simultaneously exist at, or move between, two or more location sites. Actually, multiplex proteins of this kind usually posses some important biological functions worthy of our special notice. A new predictor called "**Euk-mPLoc 2.0**" is developed by hybridizing the gene ontology information, functional domain information, and sequential evolutionary information through three different modes of pseudo amino acid composition. It can be used to identify eukaryotic proteins among the following 22 locations: (1) acrosome, (2) cell wall, (3) centriole, (4) chloroplast, (5) cyanelle, (6) cytoplasm, (7) cytoskeleton, (8) endoplasmic reticulum, (9) endosome, (10) extracell, (11) Golgi apparatus, (12) hydrogenosome, (13) lysosome, (14) melanosome, (15) microsome (16) mitochondria, (17) nucleus, (18) peroxisome, (19) plasma membrane, (20) plastid, (21) spindle pole body, and (22) vacuole. Compared with the existing methods for predicting eukaryotic protein subcellular localization, the new predictor is much more powerful and flexible, particularly in dealing with proteins with multiple locations and proteins without available accession numbers. For a newly-constructed stringent benchmark dataset which contains both single- and multiple-location proteins and in which none of proteins has $\geq 25\%$ pairwise sequence identity to any other in a same location, the overall jackknife success rate achieved by **Euk-mPLoc 2.0** is more than 24% higher than those by any of the existing predictors. As a user-friendly web-server, Euk-mPLoc 2.0 is freely accessible at http://www.csbio.sjtu.edu.cn/bioinf/euk-multi-2/. For a query protein sequence of 400 amino acids, it will take about 15 seconds for the web-server to yield the predicted result; the longer the sequence is, the more time it may usually need. It is anticipated that the novel approach and the powerful predictor as presented in this paper will have a significant impact to Molecular Cell Biology, System Biology, Proteomics, Bioinformatics, and Drug Development.

Citation: Chou K-C, Shen H-B (2010) A New Method for Predicting the Subcellular Localization of Eukaryotic Proteins with Both Single and Multiple Sites: Euk-mPLoc 2.0. PLoS ONE 5(4): e9931. doi:10.1371/journal.pone.0009931

Editor: Darren P. Martin, Institute of Infectious Disease and Molecular Medicine, South Africa

Received February 1, 2010; **Accepted** March 8, 2010; **Published** April 1, 2010

Funding: This work was supported by the National Natural Science Foundation of China (Grant No. 60704047), Science and Technology Commission of Shanghai Municipality (Grant No. 08ZR1410600, 08JC1410600), sponsored by Shanghai Pujiang Program and Innovation Program of Shanghai Municipal Education Commission (10ZZ17). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: kcchou@gordonlifescience.org

Introduction

With the avalanche of protein sequences generated in the post-genomic era, numerous efforts have been made to develop various methods for predicting protein subcellular localization based on the sequence information (see, e.g., [1,2,3,4,5,6,7,8] as well as a long list of references cited in two comprehensive review articles [9,10]). However, relatively much less efforts have been made to address those proteins which may simultaneously exist at, or move between, two or more different subcellular locations. Actually, proteins with multiple locations or dynamic feature of this kind are particularly interesting because they may have some very special biological functions worthy of our notice [11,12]. Particularly, as pointed out by Millar et al. [13], recent evidences indicate that an increasing number of proteins have multiple locations in the cell.

About two years ago, a web-server predictor [14] was developed for dealing with the eukaryotic systems that contain both single-location and multiple-location proteins. The predictor is called **Euk-mPLoc**, where "m" stands for "multiple" meaning it can be used to deal with multiplex proteins as well. The **Euk-mPLoc** predictor was established by hybridizing the "higher-level" GO (gene ontology [15]) approach and PseAAC (pseudo amino acid composition [16,17]) approach. Its power mainly came from the GO approach because proteins formulated in the GO database space would be clustered in a manner much better reflecting the distribution of their subcellular locations, as elucidated in [18].

However, the existing version of **Euk-mPLoc** has the following shortcomings. **(1)** In order to make the prediction engine able to use the advantage of the GO approach, the accession number for a query protein is required as a part of input; many proteins, such as synthetic and hypothetical proteins, or newly-discovered sequences without being deposited into databanks yet, do not have accession numbers, and hence cannot be treated with the GO approach. **(2)** Even though their accession numbers are available, it is not always certain for them to be meaningfully formulated in a GO space because the current GO database is far from complete yet. **(3)** Although the PseAAC approach, a complement to the GO approach in **Euk-mPLoc**, can take into account some partial sequence order effects, the original PseAAC [16,19] missed the functional domain and sequential evolution information that may considerably affect the prediction quality.

The present study was devoted to develop a new and more powerful predictor for predicting eukaryotic protein subcellular localization by addressing the above three problems.

Materials and Methods

Protein sequences were collected from the Swiss-Prot database at http://www.ebi.ac.uk/swissprot/. The detailed procedures are basically the same as described in [14]; the only difference is: in order to establish a more updated benchmark dataset, instead of version 50.7 of the Swiss-Prot database released on 9-Sept-2006, the version 55.3 released on 29-Apr-2008 was adopted. After strictly following the procedures as described in [14], we finally obtained a benchmark dataset $\mathbb{S}$ containing 7,766 different protein sequences that are distributed among 22 subcellular locations (**Fig. 1**); i.e.,

$$\mathbb{S}=\mathbb{S}_1\cup\mathbb{S}_2\cup\mathbb{S}_3\cup\mathbb{S}_4\cup\mathbb{S}_5\cup\mathbb{S}_6\cup\cdots\cup\mathbb{S}_{22} \qquad (1)$$

where $\mathbb{S}_1$ represents the subset for the subcellular location of "acrosome", $\mathbb{S}_2$ for "cell membrane", $\mathbb{S}_3$ for "cell wall", and so forth; while $\cup$ represents the symbol for "union" in the set theory. A breakdown of the 7,766 eukaryotic proteins in the benchmark dataset $\mathbb{S}$ according to their 22 location sites is given in **Table 1**. To avoid redundancy and homology bias, none of the proteins in $\mathbb{S}$ has $\geq 25\%$ pairwise sequence identity to any other in a same subset. The corresponding accession numbers and protein sequences are given in Online Supporting Information S1.

Because the system investigated now contains both the single-location and the multiple-location proteins, some of the proteins in $\mathbb{S}$ may occur in two or more location sites. Therefore, it is instructive to introduce the concept of "virtual sample", as illustrated as follows. A protein sample coexisting at two different location sites will be counted as 2 virtual samples even though they have an identical sequence; if coexisting at three different sites, 3 virtual samples; and so forth. Accordingly, the total number of the different virtual protein samples is generally greater than that of the total different sequence samples. Their relationship can be formulated as follows

$$N(\mathrm{vir})=N(\mathrm{seq})+\sum_{L=1}^{M}(L-1)N(\varphi) \qquad (2)$$

where $N(\mathrm{vir})$ is the number of total different virtual protein

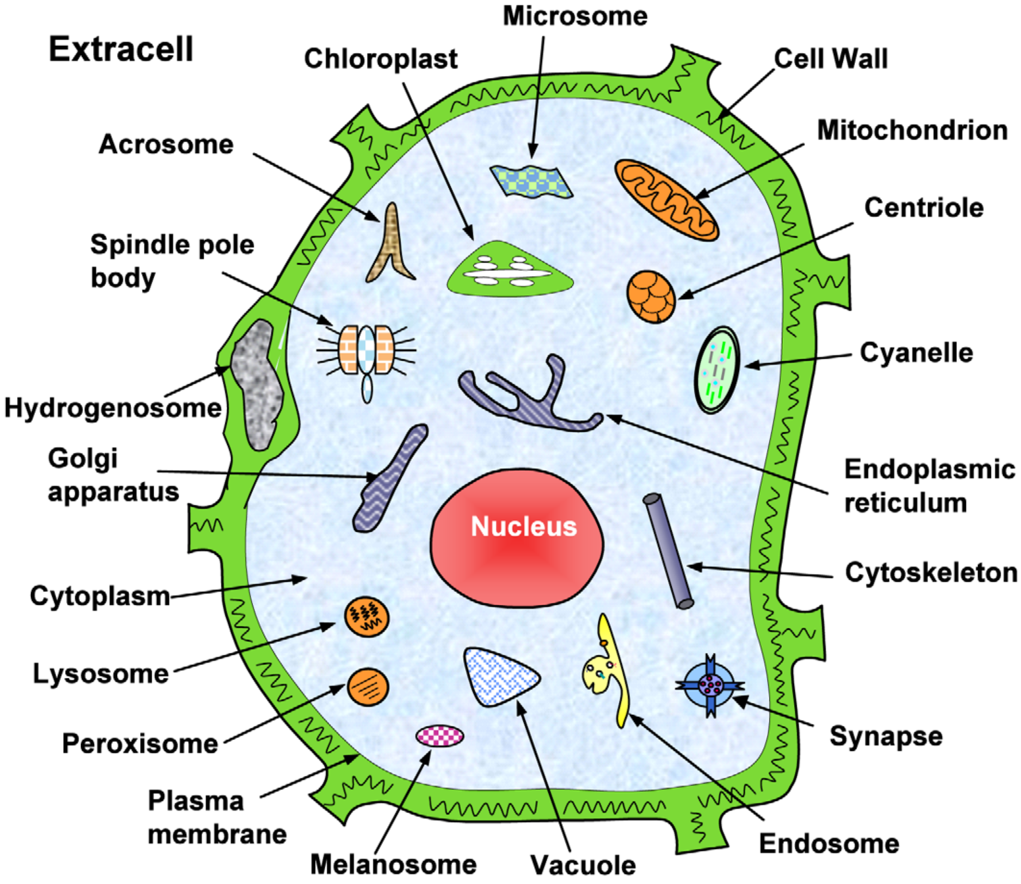


Figure 1. Illustration to show the 22 subcellular locations of eukaryotic proteins. The 22 location sites are: (1) acrosome, (2) cell wall, (3) centriole, (4) chloroplast, (5) cyanelle, (6) cytoplasm, (7) cytoskeleton, (8) endoplasmic reticulum, (9) endosome, (10) extracell, (11) Golgi apparatus, (12) hydrogenosome, (13) lysosome, (14) melanosome, (15) microsome (16) mitochondria, (17) nucleus, (18) peroxisome, (19) plasma membrane, (20) plastid, (21) spindle pole body, and (22) vacuole. Reprinted from [14] with permission.
doi:10.1371/journal.pone.0009931.g001

Table 1. Breakdown of the eukaryotic protein benchmark dataset $\mathbb{S}$ derived from Swiss-Prot database (release 55.3) according to the procedures described in the Materials section.

Subset[a]	Subcellular location	Number of proteins
$\mathbb{S}_1$	Acrosome	14
$\mathbb{S}_2$	Cell membrane	697
$\mathbb{S}_3$	Cell wall	49
$\mathbb{S}_4$	Centrosome	96
$\mathbb{S}_5$	Chloroplast	385
$\mathbb{S}_6$	Cyanelle	79
$\mathbb{S}_7$	Cytoplasm	2186
$\mathbb{S}_8$	Cytoskeleton	139
$\mathbb{S}_9$	Endoplasmic reticulum	457
$\mathbb{S}_{10}$	Endosome	41
$\mathbb{S}_{11}$	Extracell	1048
$\mathbb{S}_{12}$	Golgi apparatus	254
$\mathbb{S}_{13}$	Hydrogenosome	10
$\mathbb{S}_{14}$	Lysosome	57
$\mathbb{S}_{15}$	Melanosome	47
$\mathbb{S}_{16}$	Microsome	13
$\mathbb{S}_{17}$	Mitochondrion	610
$\mathbb{S}_{18}$	Nucleus	2320
$\mathbb{S}_{19}$	Peroxisome	110
$\mathbb{S}_{20}$	Spindle pole body	68
$\mathbb{S}_{21}$	Synapse	47
$\mathbb{S}_{22}$	Vacuole	170
Number of total virtual proteins $N(\mathrm{vir})$		8,897[b]
Number of total different proteins $N(\mathrm{seq})$		7,766[c]

None of the proteins included here has $\geq 25\%$ sequence identity to any other in a same subcellular location.

[a] See Fig. 1 and Eq.1 as well as the relevant text for the definitions of the subsets listed in this table.

[b] See Eqs.2–3 for the definition about the number of virtual proteins, and its relation with the number of different proteins.

[c] Of the 7,766 different proteins, 6,687 belong to one subcellular location, 1,029 to two locations, 48 to three locations, and 2 to four locations. See Online Supporting Information S1 for the protein sequences.

doi:10.1371/journal.pone.0009931.t001

samples in $\mathbb{S}$, $N(\mathrm{seq})$ the number of total different protein sequences, $N(1)$ the number of proteins with one location, $N(2)$ the number of proteins with two locations, and so forth; while M is the number of total subcellular location sites (for the current case, $M=22$ as shown in **Fig. 1** and **Table 1**).

For the current 7,766 different protein sequences, 6,687 occur in one subcellular location, 1,029 in two locations, 48 in three locations, 2 in four locations, and none in five or more locations. Substituting these data into **Eq.2**, we have

$$\begin{aligned} N(\mathrm{vir}) &= N(\mathrm{seq}) + (1-1)\times 6687 + (2-1)\times 1029 \\ &\quad + (3-1)\times 48 + (4-1)\times 2 + \sum_{L=5}^{22}(L-1)\times 0 \qquad (3) \\ &= 7766 + 0 + 1029 + 96 + 6 + 0 = 8897 \end{aligned}$$

which is fully consistent with the figures in **Table 1** and the data in Online Supporting Information S1.

As stated in a recent comprehensive review [20], to develop a powerful method for statistically predicting protein subcellular localization, one of the most important things is to formulate the sample of a protein with the core features that have intrinsic correlation with its localization in a cell. Since the concept of pseudo amino acid composition (PseAAC) was proposed [16], it has provided a very flexible mathematical frame for investigators to incorporate their desired information into the representation of protein samples. According to its original definition, the PseAAC is actually formulated by a set of discrete numbers [16] as long as it is different from the classical amino acid composition (AAC) and that it is derived from a protein sequence that is able to harbor some sort of its sequence order and pattern information, or able to reflect some physicochemical and biochemical properties of the constituent amino acids. Since the concept of PseAAC was proposed, it has been widely used to deal with many protein-related problems and sequence-related systems (see, e.g., [21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42] and a long list of PseAAC-related references cited in a recent review [20]). As summarized in [20], until now 16 different PseAAC modes have been used to represent the samples of proteins for predicting their attributes. Each of these modes has its own advantage and disadvantage. In this study, we are to formulate the protein samples by hybridizing the following three different modes of PseAAC.

1. GO (Gene Ontology) Representation Mode

GO database [15] was established according to the molecular function, biological process, and cellular component. Accordingly, protein samples defined in a GO database space would be clustered in a way better reflecting their subcellular locations [10,18]. However, the way of using GO mode to represent a protein sample in the original **Euk-mPLoc** predictor [14] was derived through its accession number from the GO database [43]. Thus, when using **Euk-mPLoc** to perform prediction, the accession number of a query protein would be indispensable. To avoid such a requirement, the following different procedures are proposed to derive the GO representation mode.

Step 1. Use BLAST [44] to search the homologous proteins of the query protein **P** from the Swiss-Prot database (version 55.3), with the expect value $E \leq 0.001$ for the BLAST parameter.

Step 2. Those proteins which have $\geq 60\%$ pairwise sequence identity with the query protein **P** are collected into a set, $\mathbb{S}^{\mathbf{P}\text{-homo}}$, called the "homology set" of **P**. All the elements in $\mathbb{S}^{\mathbf{P}\text{-homo}}$ can be deemed as the "representative proteins" of **P**. Because they were retrieved from the Swiss-Prot database, these representative proteins must each have their own accession numbers.

Step 3. Search each of these accession numbers collected in Step 2 against the GO database at http://www.ebi.ac.uk/GOA/ to find the corresponding GO numbers [43].

Step 4. The current GO database (version 70.0 released 10 March 2008) contains 60,020 GO numbers, thus the query protein **P** can be expressed via its representative proteins in $\mathbb{S}^{\mathbf{P}\text{-homo}}$ by the following formulation

$$\mathbf{P}_{\mathrm{GO}} = \begin{bmatrix} \Delta_1^{\mathrm{G}} & \Delta_2^{\mathrm{G}} & \cdots & \Delta_i^{\mathrm{G}} & \cdots & \Delta_{60020}^{\mathrm{G}} \end{bmatrix}^{\mathbf{T}} \qquad (4)$$

where **T** is the transposing operator, and

$$\Delta_i^{\mathrm{G}} = \begin{cases} 1, & \text{if a hit is found against the } i\text{-th GO number} \\ & \text{for any of the proteins in } \mathbb{S}^{\mathbf{P}\text{-homo}} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

Through the above steps, we can use the GO information derived from its representative proteins in $\mathbb{S}^{\mathbf{P}\text{-homo}}$ to formulate the query protein **P**. The rationale of so doing is based on the fact that homology proteins generally share similar attributes, such as structural conformations and biological functions [45,46,47]. Thus, the accession number is no longer indispensable for the input of the query protein even if using the high-level GO approach to predict its subcellular localization as required in **Euk-mPLoc** [14].

The above homology-based GO extraction method is particularly useful for studying those proteins which do not have UniProt accession numbers. However, it would still fail to work under any one of the following situations: **(1)** the query protein does not have significant homology to any protein in the Swiss-Prot database, i.e., $\mathbb{S}^{\mathbf{P}\text{-homo}} = \varnothing$ meaning the homology set is an empty one; **(2)** its representative proteins do not contain any useful GO information for statistical prediction based on a given training dataset.

Therefore, it is necessary to consider the following representation modes for those proteins which fail to be meaningfully defined in the GO space.

2. FunD (Functional Domain) Representation Mode

FunD is the core of a protein that plays the major role for its function. That is why in determining the 3-D (dimensional) structure of a protein by experiments (see, e.g., [48,49]) or by computational modeling (see, e.g., [47,50]) the first priority was always focused on its FunD. Actually, using the FunD information to formulate protein samples for statistical predictions was originally proposed in [51,52], and quite encouraged results were achieved. In that time, the 2005 FunDs in the SBASE-A database [53] were used as bases to formulate the protein samples. Since then, a series of follow-up protein FunD databases were established, such as COG [54], KOG [54], SMART [55], Pfam [56], and CDD [57]. Of these databases, CDD contains the domains imported from COG, Pfam and SMART, and hence is relatively much more complete [57]. The version 2.11 of CDD contains 17,402 characteristic domains. Using each of these domains as a base vector, we can define a FunD space with 17,402 dimensions. Thus, by following the similar procedures in [51], a protein sample can be uniquely defined through the steps described below:

Step 1. Use RPS-BLAST (Reverse PSI-BLAST) program [44] to conduct sequence alignment of the protein sequence with each of the 17,402 domain sequences in the CDD database.

Step 2. If the significance threshold value (expect value) is ≤ 0.001 for the i-th domain meaning that a "hit" is found, then the i-th component of the protein in the 17402-D space is assigned 1; otherwise, 0.

Step 3. The protein sample **P** in the FunD space can thus be formulated as

将蛋白质表示为17402维的向量

$$\mathbf{P}_{\mathrm{FunD}} = \begin{bmatrix} \Delta_1^{\mathrm{D}} & \Delta_2^{\mathrm{D}} & \cdots & \Delta_i^{\mathrm{D}} & \cdots & \Delta_{17402}^{\mathrm{D}} \end{bmatrix}^{\mathbf{T}} \tag{6}$$

where **T** is the transpose operator, and

$$\Delta_i^{\mathrm{D}} = \begin{cases} 1, & \text{when a hit is found for } \mathbf{P} \text{ in CDD} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

Defined this way, the protein sample becomes corresponding to a 17402-D vector $\mathbf{P}_{\mathrm{FunD}}$ with each of the 17402 functional domain sequences as a base for the vector space. By using such a representation, not only some sequence-order effects but also some functional information is included. Since the function of a protein is closely related to its subcellular location, the FunD formulation of Eq.6 would naturally incorporate those factors that might be directly correlated with the protein subcellular location.

3. SeqEvo (Sequential Evolution) Representation Mode

Since biology is a natural science with historic dimension, all biological species have actually developed continuously starting out from a very limited number of ancestral species. It is quite typical for protein sequences [47]. Their evolution involves changes of single residues, insertions and deletions of several residues, gene doubling, and gene fusion. With such changes accumulated for a long period of time, many similarities between initial and resultant amino acid sequences are eliminated, but the corresponding proteins may still share many common attributes, such as their location site in a cell. Therefore, to catch the core feature and intrinsic relationship from a huge number of complicated protein sequences, it is particularly important to take into account the evolution effects. To realize this, here we are to incorporate the evolution information through the "Position-Specific Scoring Matrix" or "PSSM" [44], i.e., to express the protein **P** by a $20 \times L$ matrix as formulated by

$$\mathbf{P}_{\mathrm{Evo}} = \begin{bmatrix} E_{1\to1} & E_{1\to2} & \cdots & E_{1\to20} \\ E_{2\to1} & E_{2\to2} & \cdots & E_{2\to20} \\ \vdots & \vdots & \vdots & \vdots \\ E_{L\to1} & E_{L\to2} & \cdots & E_{L\to20} \end{bmatrix} \tag{8}$$

where L is the length of **P** (counted in the total number of its constituent amino acids), $E_{i\to j}$ represents the score of the amino acid residue in the i-th position of the protein sequence being changed to amino acid type j during the evolutionary process. Here, the numerical codes 1, 2, …, 20 are used to denote the 20 native amino acid types according to the alphabetical order of their single character codes. The $L \times 20$ scores in Eq.8 were generated by using PSI-BLAST [44] to search the Swiss-Prot database (version 55.3 released on 29-Apr-2007) through three iterations with 0.001 as the E-value cutoff for multiple sequence alignment against the sequence of the protein **P**, followed by a standard conversion given below:

$$E_{i\to j} = \frac{E_{i\to j}^0 - \bar{E}_i^0}{\mathrm{SD}\left(\bar{E}_i^0\right)} (i = 1, 2, \cdots, L;\ j = 1, 2, \cdots, 20) \tag{9}$$

where $E_{i\to j}^0$ represent the original scores directly created by PSI-BLAST [44] that are generally shown as positive or negative integers (the positive score means that the corresponding mutation occurs more frequently than expected by chance, while the negative means just the opposite); the symbol $\bar{E}_i^0$ means taking the average of $E_{i\to j}^0$ over j $(1, 2, \cdots, 20)$, and $\mathrm{SD}\left(\bar{E}_i^0\right)$ means the corresponding standard deviation. The converted values obtained by Eq.9 will have a zero mean value over the 20 amino acids and will remain unchanged if going through the same conversion procedure again. However, according Eq.8, a protein with L length is corresponding to a matrix of L rows. Hence, proteins with different lengths will correspond to matrices of different dimensions. This will become a hurdle for us to develop a predictor able to unanimously cover proteins of any length. To overcome such a hurdle, one possible avenue is to represent a protein sample **P** by

$$\bar{\mathbf{P}}_{\mathrm{Evo}} = \begin{bmatrix} \bar{E}_1 & \bar{E}_2 & \cdots & \bar{E}_{20} \end{bmatrix}^{\mathbf{T}} \tag{10}$$

where

$$\bar{E}_j = \frac{1}{L}\sum_{i=1}^{L} E_{i\to j} \quad (j=1,\, 2,\, \cdots,\, 20) \tag{11}$$

where $\bar{E}_j$ represents the average score of the amino acid residues in the protein **P** being changed to amino acid type j during the evolutionary process. However, if $\bar{\mathbf{P}}_{\mathrm{Evo}}$ of Eq.10 was used to represent the protein **P**, all the sequence-order information during the evolutionary process would be erased. To avoid completely erasing the sequence-order information, the concept of PseAAC as originally proposed in [16] was utilized; i.e., instead of Eq.10, let us use the pseudo position-specific scoring matrix as given by

$$\mathbf{P}^{\lambda}_{\mathrm{PseEvo}} = \begin{bmatrix} \bar{E}_1 & \bar{E}_2 & \cdots & \bar{E}_{20} & \bar{E}^{\lambda}_1 & \bar{E}^{\lambda}_2 & \cdots & \bar{E}^{\lambda}_{20} \end{bmatrix}^{\mathbf{T}} \tag{12}$$

to represent the protein **P**, where

$$E^{\lambda}_j = \frac{1}{L-\lambda}\sum_{i=1}^{L-\lambda}\left[E_{i\to j} - E_{(i+\lambda)\to j}\right]^2 \quad (j=1,\, 2,\, \cdots,\, 20;\ \lambda<L) \tag{13}$$

meaning that E^{1}_j is the correlation factor by coupling the most contiguous position-specific scoring matrix scores along the protein chain for the amino acid type j; E^{2}_j that by coupling the second-most contiguous position-specific scoring matrix scores; and so forth. Note that, as mentioned in the Material section of [14], the length of the shortest protein sequence in the benchmark dataset is $L=50$, and hence the value allowed for λ in Eq.13 must be smaller than 50. When $\lambda=0$, E^{λ}_j becomes a naught element and Eq.12 is degenerated to Eq.10.

A hybridization of the above three different PseAAC modes, i.e., Eq.4, Eq.6, and Eq.12, will be used to represent protein samples for establishing a new classifier for predicting eukaryotic protein subcellular localization, as described below.

4. Prediction Engine $\mathbb{C}^{\mathrm{E}}$ and Computing Procedures

The prediction engine used in this study is the ensemble classifier $\mathbb{C}^{\mathrm{E}}$ formed by fusing many individual OET-KNN (Optimized Evidence-Theoretic K-Nearest Neighbor) classifiers [58,59]. According to the underlying rule of the OET-KNN classifier, a query protein should be assigned to the class the majority of its K nearest neighbors belongs to. However, for most benchmark datasets, when $K>10$ the success rate thus obtained would decrease markedly. Therefore, our consideration for K can be confined within the range from 1 to 10. Accordingly, the ensemble classifier $\mathbb{C}^{\mathrm{E}}$ can be formulated as

$$\mathbb{C}^{\mathrm{E}} = \mathbb{C}(1)\forall\mathbb{C}(2)\forall\cdots\forall\mathbb{C}(9)\forall\mathbb{C}(10) = \forall_{K=1}^{10}\mathbb{C}(K) \tag{14}$$

where the symbol $\forall$ denotes the fusing operator, $\mathbb{C}(1)$ is the individual OET-KNN classifier based on $K=1$ nearest neighbor, $\mathbb{C}(2)$ that based on $K=2$ nearest neighbors, and so forth. The detailed mathematical formulations for OET-KNN and $\mathbb{C}^{\mathrm{E}}$ have been given in Eqs.22–29 in [10], where it has also been clearly elaborated how the ensemble classifier $\mathbb{C}^{\mathrm{E}}$ worked during the process of prediction. To avoid redundancy, we are not to repeat the details here.

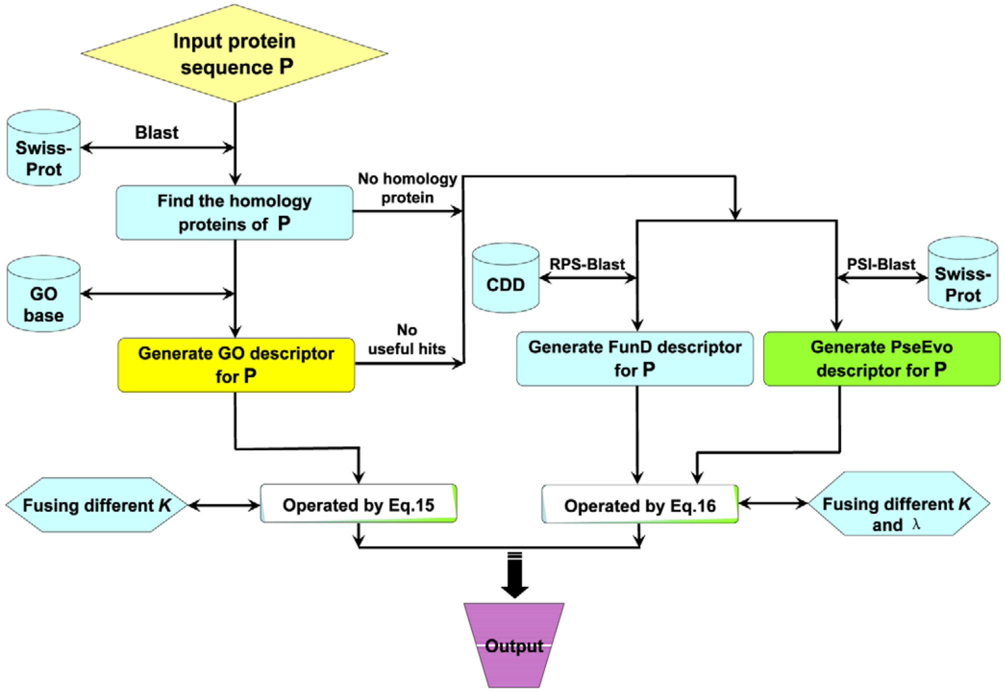


Figure 2. A flowchart to show the prediction process of Euk-mPLoc 2.0.
doi:10.1371/journal.pone.0009931.g002

The prediction is processed according to the following order.

Step 1. If the query protein **P** can be expressed as a meaningful or productive descriptor in the GO database via its representative proteins in its homology set $\mathbb{S}^{\mathbf{P}\text{-homo}}$, then $\mathbf{P}_{GO}$ of Eq.4 should be input into the prediction engine for identifying its subcellular location site(s); i.e.

$$\mathbb{C}^{E} \triangleright \mathbf{P} = \mathbb{C}^{E} \triangleright \mathbf{P}_{GO} = \forall_{K=1}^{20} \mathbb{C}(K) \triangleright \mathbf{P}_{GO} = \boxed{\begin{array}{l}\text{Outcome by fusing the 10}\\ \text{outputs yielded by } \mathbb{C}(1),\ \mathbb{C}(2),\\ \cdots,\ \mathbb{C}(10)\text{ on }\mathbf{P}_{GO}\text{,respectively}\end{array}} \qquad (15)$$

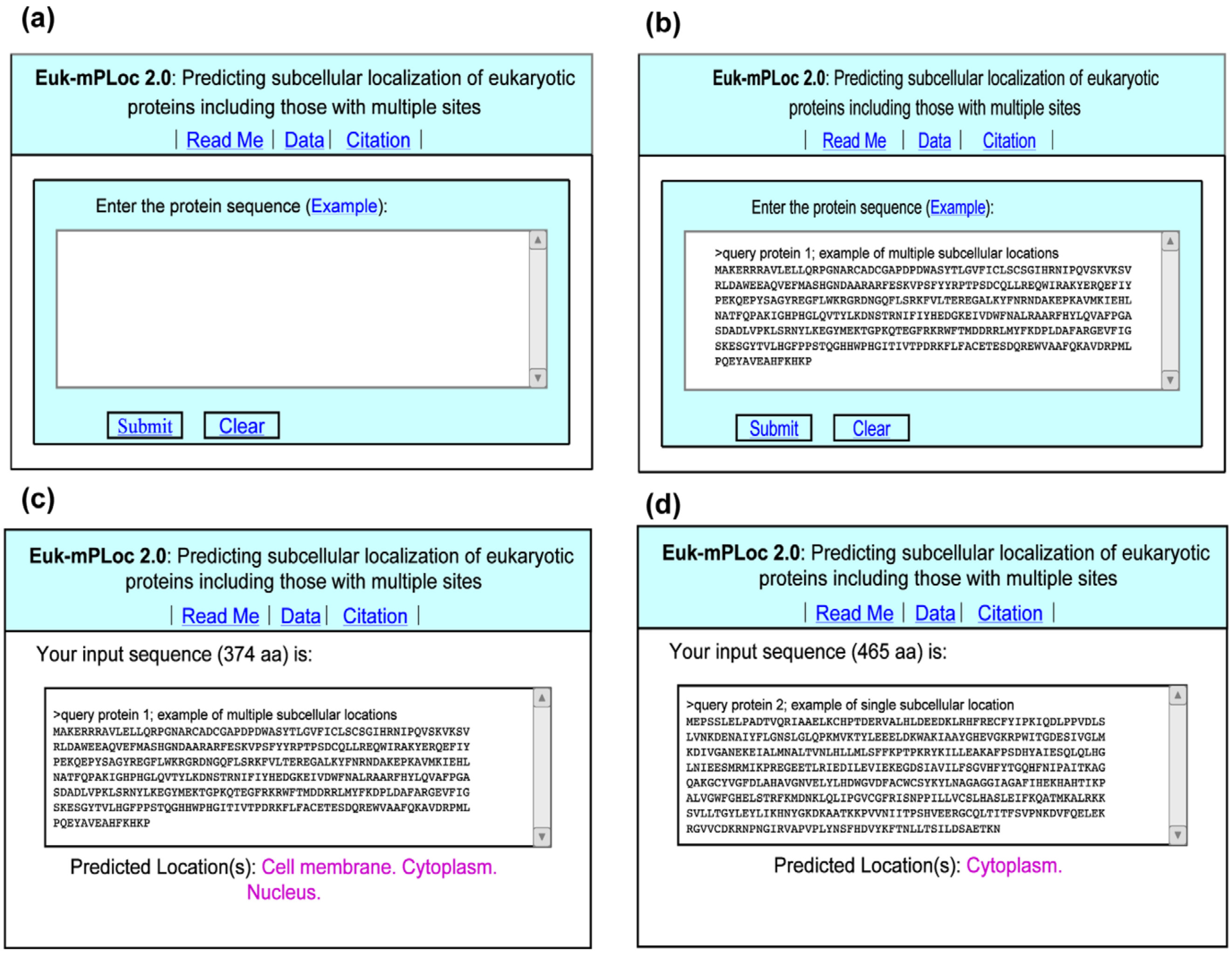
where $\triangleright$ represents the identification operator, and the fusion is made via a voting operation as formulated by Eqs.32–35 in [10].

Step 2. If the query protein **P** does not have significant homology to any protein in the Swiss-Prot database, i.e., $\mathbb{S}^{\mathbf{P}\text{-homo}} = \varnothing$ (empty set), or its representative proteins in $\mathbb{S}^{\mathbf{P}\text{-homo}}$ do not contain any useful GO information, then both the FunD representation $\mathbf{P}_{\text{FunD}}$ of Eq.6 and the pseudo position-specific scoring matrix representation $\mathbf{P}_{\text{PseEvo}}^{\lambda}$ of Eq.12 should be inputted into the prediction engine $\mathbb{C}^{E}$. The output will be determined by fusing many preliminary outcomes associated with different K of $\mathbb{C}^{E}$ (cf. Eq.14) and different possible λ of the pseudo sequential evolution descriptor (cf. Eq.12); i.e.,

$$\mathbb{C}^{E} \triangleright \mathbf{P} = \begin{pmatrix} \mathbb{C}^{E} \triangleright \mathbf{P}_{\text{FunD}} \\ \mathbb{C}^{E} \triangleright \mathbf{P}_{\text{PseEvo}}^{\lambda} \end{pmatrix} = \boxed{\begin{array}{l}\text{Outcome by fusing the 10 outputs}\\ \text{yielded by } \mathbb{C}^{E}\text{ on }\mathbf{P}_{\text{FunD}}\text{ and}\\ 10 \times 50 = 500\text{ outputs on }\mathbf{P}_{\text{PseEvo}}^{\lambda}\end{array}} \qquad (16)$$

where the factor 10 is because K in $\mathbb{C}^{E}$ can be $1, 2, \cdots, 10$ and the factor 50 is because λ in $\mathbf{P}_{\text{PseEvo}}^{\lambda}$ can be $0, 1, 2, \cdots, 49$ (cf. Eqs.12–13).

Step 3. To make Eqs.15–16 capable to handle proteins with multiple locations as well, the ensemble classifier $\mathbb{C}^{E}$ needed to be modified to $\mathbb{C}^{E}(\theta)$, where θ is a threshold parameter for controlling the count of multiple location sites and optimizing



Euk-mPLoc 2.0: Predicting subcellular localization of eukaryotic proteins including those with multiple sites

Enter the protein sequence (Example):

```
>query protein 1; example of multiple subcellular locations
MAKERRRAVLELLQRPGNARCADCGAPDPDWASYTLGVFICLSCSGIHRNIPQVSKVKSV
RLDAWEEAQVEFMASHGNDAARARFESKVPSFYYRPTPSDCQLLREQWIRAKYERQEFIY
PEKQEPYSAGYREGFLWKRGRDNGQFLSRKFVLTEREGALKYFNRNDAKEPKAVMKIEHL
NATFQPAKIGHPHGLQVTYLKDNSTRNIFIYHEDGKEIVDWFNALRAARFHYLQVAFPGA
SDADLVPKLSRNYLKEGYMEKTGPKQTEGFRKRWFTMDDRRLMYFKDPLDAFARGEVFIG
SKESGYTVLHGFPPSTQGHHWPHGITIVTPDRKFLFACETESDQREWVAAFQKAVDRPML
PQEYAVEAHFKHKP
```

Your input sequence (374 aa) is:

```
>query protein 1; example of multiple subcellular locations
MAKERRRAVLELLQRPGNARCADCGAPDPDWASYTLGVFICLSCSGIHRNIPQVSKVKSV
RLDAWEEAQVEFMASHGNDAARARFESKVPSFYYRPTPSDCQLLREQWIRAKYERQEFIY
PEKQEPYSAGYREGFLWKRGRDNGQFLSRKFVLTEREGALKYFNRNDAKEPKAVMKIEHL
NATFQPAKIGHPHGLQVTYLKDNSTRNIFIYHEDGKEIVDWFNALRAARFHYLQVAFPGA
SDADLVPKLSRNYLKEGYMEKTGPKQTEGFRKRWFTMDDRRLMYFKDPLDAFARGEVFIG
SKESGYTVLHGFPPSTQGHHWPHGITIVTPDRKFLFACETESDQREWVAAFQKAVDRPML
PQEYAVEAHFKHKP
```

Predicted Location(s): Cell membrane. Cytoplasm. Nucleus.

Your input sequence (465 aa) is:

```
>query protein 2; example of single subcellular location
MEPSSLELPADTVQRIAAELKCHPTDERVALHLDEEDKLRHFRECFYIPKIQDLPPVDLS
LVNKDENAIYFLGNSLGLQPKMVKTYLEEELDKWAKIAAYGHEVGKRPWITGDESIVGLM
KDIVGANEKEIALMNALTVNLHLLMLSFFKPTPKRYKILLEAKAFPSDHYAIESQLQLHG
LNIEESMRMIKPREGEETLRIEDILEVIEKEGDSIAVILFSGVHFYTGQHFNIPAITKAG
QAKGCYVGFDLAHAVGNVELYLHDWGVDFACWCSYKYLNAGAGGIAGAFIHEKHAHTIKP
ALVGWFGHELSTRFKMDNKLQLIPGVCGFRISNPPILLVCSLHASLEIFKQATMKALRKK
SVLLTGYLEYLIKHNYGKDKAATKKPVVNIITPSHVEERGCQLTITFSVPNKDVFQELEK
RGVVCDKRNPNGIRVAPVPLYNSFHDVYKFTNLLTSILDSAETKN
```

Predicted Location(s): Cytoplasm.

Figure 3. Semi-screenshot to show the prediction steps. (**a**) The top page of the **Euk-mPLoc 2.0** web server at http://www.csbio.sjtu.edu.cn/bioinf/euk-multi-2/. (**b**) The input of a query protein in FASTA format. (**c**) The output predicted by **Euk-mPLoc 2.0** for the query protein 1 in the Example window. (**d**) The output for the query protein 2 in the Example window.
doi:10.1371/journal.pone.0009931.g003

the predicted results, as formulated by Eqs.39–48 in [10] where it was also elaborated how to evaluate the overall success rate when using $\mathbb{C}^{E}(\theta)$ on a benchmark dataset containing both single and multiple location proteins.

The entire ensemble classifier thus established is called "**Euk-mPLoc 2.0**", where "2.0" refers to an updated version evolved from Euk-mPLoc [14]. To provide an intuitive picture, a flowchart is given in **Fig. 2** to illustrate the prediction process of **Euk-mPLoc 2.0**.

Protocol Guide

For the convenience of experimental scientists, a user-friendly web-server was established for **Euk-mPLoc 2.0**. Below, let us give a step-by-step guide on how to use it to get the desired results.

Step 1. Open the web server at http://www.csbio.sjtu.edu.cn/bioinf/euk-multi-2/ and you will see the top page of the predictor on your computer screen, as shown in **Fig. 3a**. Click on the Read Me button to see a brief introduction about **Euk-mPLoc 2.0** predictor and the caveat when using it.

Step 2. Either type or copy and paste the query protein sequence into the input box at the center of **Fig. 3a**. The input sequence should be in the FASTA format. A sequence in FASTA format consists of a single initial line beginning with a greater-than symbol (">") in the first column, followed by lines of sequence data. The words right after the ">" symbol in the single initial line are optional and only used for the purpose of identification and description. All lines should be no longer than 120 characters and usually do not exceed 80 characters. The sequence ends if another line starting with a ">" appears; this indicates the start of another sequence. Example sequences in FASTA format can be seen by clicking on the Example button right above the input box. For more information about FASTA format, visit http://en.wikipedia.org/wiki/Fasta_format.

Step 3. Click on the Submit button to see the predicted result. For example, if you use the sequence of query protein 1 in the Example window, the input screen should look like the illustration in **Fig. 3b**; after clicking the Submit button, you will see "**Cell membrane;** Cytoplasm; Nucleus" shown on the predicted result window (**Fig. 3c**), meaning that the protein is a multiplex one, which can simultaneously occur in "cell membrane", "cytoplasm", and "nucleus" organelles, fully consistent with experimental observations. However, if using the sequence of query protein 2 in the Example window as an input, you will instead see "**Cytoplasm**" shown on the predicted result window (**Fig. 3d**), meaning that the protein is a single-location one residing in "cytoplasm" compartment only, also fully consistent with experimental observations. It takes about 15 seconds for a protein sequence of 400 amino acids before the predicted result appears on your computer screen; the longer the sequence is, the more time it is usually needed.

Step 4. Click on the Citation button to find the relevant papers that document the detailed development and algorithm of **Euk-mPLoc 2.0**.

Step 5. Click on the Data button to download the benchmark datasets used to train and test the **Euk-mPLoc 2.0** predictor.

Caveat. To obtain the predicted result with the expected success rate, the entire sequence of the query protein rather than its fragment should be used as an input. A sequence with less than 50 amino acid residues is generally deemed as a fragment. Also, if the query protein is known not one of the 22 locations as shown in **Fig. 1**, stop the prediction because the result thus obtained will not make any sense.

Results and Discussion

In statistical prediction, it would be meaningless to simply say a success rate of a predictor without specifying what method and benchmark dataset were used to test its accuracy. The following three cross-validation methods are often used to evaluate the accuracy of a statistical predictor: independent dataset test, sub-sampling (K-fold) test, and jackknife test [60]. Of these three, the jackknife test is deemed the most objective because the independent dataset test and sub-sampling test cannot avoid arbitrariness, as elaborated in a comprehensive review [10]. Therefore, the jackknife test has been increasingly and widely adopted to examine the power of various predictors (see, e.g., [23,24,25,27,29,31,34,37,61,62,63,64,65,66,67]). However, even if tested by the jackknife cross-validation, a same predictor can still yield different success rates for different benchmark datasets. This is because the more stringent of a benchmark dataset in excluding homologous sequences, or the more subcellular locations it covers, the more difficult for a predictor to yield a high overall success rate. For instance, ProtLock [2] and HSLPred [68] are two predictors developed for identifying protein subcellular localization. Both were reported with the success rates over 70–80% [2,68] when tested by the benchmark datasets that allow inclusion of homologous proteins with up to 90% pairwise sequence identity and cover only 4 or 5 subcellular location sites. However, when the

Table 2. A comparison of Euk-mPLoc 2.0 with Euk-PLoc in the jackknife cross-validation test on the benchmark dataset covering 22 location sites where none of the eukaryotic proteins included has $\geq 25\%$ pairwise sequence identity to any other in a same location.

Subcellular location site	Success rate by jackknife cross-validation[a]	
	Euk-mPLoc	**Euk-mPLoc 2.0**
Acrosome	0/14 = 0.00%	1/14 = 7.14%
Cell membrane	262/697 = 37.58%	452/697 = 64.85%
Cell wall	4/49 = 8.16%	6/49 = 12.24%
Centrosome	9/96 = 9.38%	22/96 = 22.92%
Chloroplast	117/385 = 30.39%	318/385 = 82.60%
Cyanelle	12/79 = 15.19%	47/79 = 59.49%
Cytoplasm	918/2186 = 41.99%	1418/2186 = 64.87%
Cytoskeleton	4/139 = 2.88%	44/139 = 31.65%
Endoplasmic reticulum	115/457 = 25.16%	348/457 = 76.15%
Endosome	1/41 = 2.44%	2/41 = 4.88%
Extracell	678/1048 = 64.69%	858/1048 = 81.87%
Golgi apparatus	5/254 = 1.97%	56/254 = 22.05%
Hydrogenosome	0/10 = 0.00%	2/10 = 20.00%
Lysosome	5/57 = 8.77%	26/57 = 45.61%
Melanosome	0/47 = 0.00%	0/47 = 0.00%
Microsome	0/13 = 0.00%	1/13 = 7.69%
Mitochondrion	143/610 = 23.44%	427/610 = 70.00%
Nucleus	1212/2320 = 52.24%	1501/2320 = 64.70%
Peroxisome	1/110 = 0.91%	56/110 = 50.91%
Spindle pole body	0/68 = 0.00%	23/68 = 0.3382
Synapse	0/47 = 0.00%	0/47 = 0.00%
Vacuole	7/170 = 4.12%	101/170 = 59.41%
Total	**3493/8897 = 39.26%**	**5709/8897 = 64.17%**

[a] Note that in order to make the comparison under exactly the same condition, only the sequences of proteins in the Online Supporting Information S1 but not their accession numbers were used as inputs during the prediction.

doi:10.1371/journal.pone.0009931.t002

two predictors were tested by the stringent dataset covering 16 different subcellular locations in which none of proteins included has $\geq 25\%$ pairwise sequence identity to any other in a same subset, the overall jackknife success rate achieved by ProtLock [2] would drop down to 28.7% and that by HSLPred [68] down to 33.1%, as reported in [58].

Now the current benchmark dataset is even more stringent because, in addition to the same threshold to rigorously exclude the homologous sequences, it covers even more, i.e., 22 location sites. Besides, to the best of our knowledge, except **Euk-mPLoc** [14], so far there is no other web-server predictor whatsoever that can be used to predict a system with both single- and multiple-location proteins distributed among 22 different location sites. Accordingly, to demonstrate the advantage of **Euk-mPLoc 2.0**, it would be sufficient to simply compare the success rates achieved by the new predictor with those by **Euk-mPLoc** [14].

Listed in **Table 2** are the results obtained with **Euk-mPLoc** [14] and **Euk-mPLoc 2.0** on the benchmark dataset $\mathbb{S}$ (cf. **Table 1**) by the jackknife cross-validation test. During the testing process, only the sequences of proteins in Online Supporting Information S1 but not their accession numbers were used as inputs in order to make the comparison between the two predictors under exactly the same condition. During the course of the jackknife cross-validation by **Euk-mPLoc 2.0** and **Euk-mPLoc**, the false positives (over-predictions) and false negatives (under-predictions) were also taken into account to reduce the scores for calculating the success rate. Note that it is more complicated to count the over-predictions and under-predictions for a system containing both single-location and multiple-location proteins. For the detailed calculation process, refer to Eqs.43–48 as well as Fig. 4 in a comprehensive review [10]. As we can see from **Table 2**, for such a stringent and multiplex benchmark dataset, the overall success rate achieved by **Euk-mPLoc 2.0** is over 64%, which is about 25% higher than that by **Euk-mPLoc**.

Finally, it should be pointed out that although **Euk-mPLoc 2.0** is more powerful than the existing predictors in identifying the subcellular locations of eukaryotic proteins, there is much room for further improvement in future studies. As shown in **Table 2**, the success rates by **Euk-mPLoc 2.0** for proteins belonging to "melanosome" and "synapse" locations are very low. This is because of that, compared with the most of the other 20 location sites, the numbers of proteins in the two sites are not sufficiently large (cf. **Table 1** and Online Supporting Information S1) to train the prediction engine in a more effective way. It is anticipated that with more experimental data available for the two sites in the future, the situation will be improved and **Euk-mPLoc 2.0** will become even more powerful.

Supporting Information

Supporting Information S1

Found at: doi:10.1371/journal.pone.0009931.s001 (4.45 MB PDF)

Acknowledgments

The authors wish to thank the tree anonymous reviewers for their constructive comments, which are very helpful for strengthening the presentation of this paper.

Author Contributions

Conceived and designed the experiments: KCC HBS. Performed the experiments: KCC HBS. Analyzed the data: KCC HBS.

References

1. Nakashima H, Nishikawa K (1994) Discrimination of intracellular and extracellular proteins using amino acid composition and residue-pair frequencies. J Mol Biol 238: 54–61.
2. Cedano J, Aloy P, P'erez-Pons JA, Querol E (1997) Relation between amino acid composition and cellular location of proteins. J Mol Biol 266: 594–600.
3. Chou KC, Elrod DW (1999) Protein subcellular location prediction. Protein Engineering 12: 107–118.
4. Emanuelsson O, Nielsen H, Brunak S, von Heijne G (2000) Predicting subcellular localization of proteins based on their N-terminal amino acid sequence. Journal of Molecular Biology 300: 1005–1016.
5. Zhou GP, Doctor K (2003) Subcellular location prediction of apoptosis proteins. PROTEINS: Structure, Function, and Genetics 50: 44–48.
6. Small I, Peeters N, Legeai F, Lurin C (2004) Predotar: A tool for rapidly screening proteomes for N-terminal targeting sequences. Proteomics 4: 1581–1590.
7. Matsuda S, Vert JP, Saigo H, Ueda N, Toh H, Akutsu T (2005) A novel representation of protein sequences for prediction of subcellular location using support vector machines. Protein Sci 14: 2804–2813.
8. Pierleoni A, Martelli PL, Fariselli P, Casadio R (2006) BaCelLo: a balanced subcellular localization predictor. Bioinformatics 22: e408–416.
9. Nakai K (2000) Protein sorting signals and prediction of subcellular localization. Advances in Protein Chemistry 54: 277–344.
10. Chou KC, Shen HB (2007) Review: Recent progresses in protein subcellular location prediction. Analytical Biochemistry 370: 1–16.
11. Smith C (2008) Subcellular targeting of proteins and drugs. http://wwwbiocomparecom/Articles/TechnologySpotlight/976/Subcellular-Targeting-Of-Proteins-And-Drugshtml.
12. Glory E, Murphy RF (2007) Automated subcellular location determination and high-throughput microscopy. Dev Cell 12: 7–16.
13. Millar AH, Carrie C, Pogson B, Whelan J (2009) Exploring the function-location nexus: using multiple lines of evidence in defining the subcellular location of plant proteins. Plant Cell 21: 1625–1631.
14. Chou KC, Shen HB (2007) Euk-mPLoc: a fusion classifier for large-scale eukaryotic protein subcellular location prediction by incorporating multiple sites. Journal of Proteome Research 6: 1728–1734.
15. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G (2000) Gene ontology: tool for the unification of biology. Nature Genetics 25: 25–29.
16. Chou KC (2001) Prediction of protein cellular attributes using pseudo amino acid composition. PROTEINS: Structure, Function, and Genetics (Erratum: ibid, 2001, Vol44, 60) 43: 246–255.
17. Chou KC (2005) Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. Bioinformatics 21: 10–19.
18. Chou KC, Shen HB (2008) Cell-PLoc: A package of web-servers for predicting subcellular localization of proteins in various organisms. Nature Protocols 3: 153–162.
19. Shen HB, Chou KC (2008) PseAAC: a flexible web-server for generating various kinds of protein pseudo amino acid composition. Analytical Biochemistry 373: 386–388.
20. Chou KC (2009) Pseudo amino acid composition and its applications in bioinformatics, proteomics and system biology. Current Proteomics 6: 262–274.
21. Zhou XB, Chen C, Li ZC, Zou XY (2007) Using Chou's amphiphilic pseudo-amino acid composition and support vector machine for prediction of enzyme subfamily classes. Journal of Theoretical Biology 248: 546–551.
22. Zhang GY, Li HC, Fang BS (2008) Predicting lipase types by improved Chou's pseudo-amino acid composition. Protein & Peptide Letters 15: 1132–1137.
23. Nanni L, Lumini A (2008) Genetic programming for creating Chou's pseudo amino acid based features for submitochondria localization. Amino Acids 34: 653–660.
24. Zhang GY, Fang BS (2008) Predicting the cofactors of oxidoreductases based on amino acid composition distribution and Chou's amphiphilic pseudo amino acid composition. Journal of Theoretical Biology 253: 310–315.
25. Zeng YH, Guo YZ, Xiao RQ, Yang L, Yu LZ, et al. (2009) Using the augmented Chou's pseudo amino acid composition for predicting protein submitochondria locations based on auto covariance approach. Journal of Theoretical Biology 259: 366–372.
26. Qiu JD, Huang JH, Liang RP, Lu XQ (2009) Prediction of G-protein-coupled receptor classes based on the concept of Chou's pseudo amino acid composition: an approach from discrete wavelet transform. Analytical Biochemistry 390: 68–73.
27. Lin H, Wang H, Ding H, Chen YL, Li QZ (2009) Prediction of Subcellular Localization of Apoptosis Protein Using Chou's Pseudo Amino Acid Composition. Acta Biotheor 57: 321–330.

28. Lin H, Ding H, Feng-Biao Guo FB, Zhang AY, et al. (2008) Predicting subcellular localization of mycobacterial proteins by using Chou's pseudo amino acid composition. Protein & Peptide Letters 15: 739–744.
29. Lin H (2008) The modified Mahalanobis discriminant for predicting outer membrane proteins by using Chou's pseudo amino acid composition. Journal of Theoretical Biology 252: 350–356.
30. Li FM, Li QZ (2008) Predicting protein subcellular location using Chou's pseudo amino acid composition and improved hybrid approach. Protein & Peptide Letters 15: 612–616.
31. Jiang X, Wei R, Zhang TL, Gu Q (2008) Using the concept of Chou's pseudo amino acid composition to predict apoptosis proteins subcellular location: an approach by approximate entropy. Protein & Peptide Letters 15: 392–396.
32. Georgiou DN, Karakasidis TE, Nieto JJ, Torres A (2009) Use of fuzzy clustering technique and matrices to classify amino acids and its impact to Chou's pseudo amino acid composition. Journal of Theoretical Biology 257: 17–26.
33. Fang Y, Guo Y, Feng Y, Li M (2008) Predicting DNA-binding proteins: approached from Chou's pseudo amino acid composition and other specific sequence features. Amino Acids 34: 103–109.
34. Ding H, Luo L, Lin H (2009) Prediction of cell wall lytic enzymes using Chou's amphiphilic pseudo amino acid composition. Protein & Peptide Letters 16: 351–355.
35. Esmaeili M, Mohabatkar H, Mohsenzadeh S (2010) Using the concept of Chou's pseudo amino acid composition for risk type prediction of human papillomaviruses. Journal of Theoretical Biology 263: 203–209.
36. Ding YS, Zhang TL (2008) Using Chou's pseudo amino acid composition to predict subcellular localization of apoptosis proteins: an approach with immune genetic algorithm-based ensemble classifier. Pattern Recognition Letters 29: 1887–1892.
37. Chen C, Chen L, Zou X, Cai P (2009) Prediction of protein secondary structure content by using the concept of Chou's pseudo amino acid composition and support vector machine. Protein & Peptide Letters 16: 27–31.
38. Gonzalez-Diaz H, Gonzalez-Diaz Y, Santana L, Ubeira FM, Uriarte E (2008) Proteomics, networks, and connectivity indices. Proteomics 8: 750–778.
39. Gonzalez-Diaz H, Prado-Prado F, Perez-Montoto LG, Duardo-Sanchez A, Lopez-Diaz A (2009) QSAR Models for Proteins of Parasitic Organisms, Plants and Human Guests: Theory, Applications, Legal Protection, Taxes, and Regulatory Issues. Current Proteomics 6: 214–227.
40. Gonzalez-Diaz H, Prado-Prado F, Ubeira FM (2008) Predicting antimicrobial drugs and targets with the MARCH-INSIDE approach. Curr Top Med Chem 8: 1676–1690.
41. Gonzalez-Diaz H, Vilar S, Santana L, Uriarte E (2007) Medicinal chemistry and bioinformatics - current trends in drugs discovery with networks topological indices. Curr Top Med Chem 10: 1015–1029.
42. Perez-Montoto LG, Prado-Prado F, Ubeira FM, Gonzalez-Diaz H (2009) Study of Parasitic Infections, Cancer, and other Diseases with Mass-Spectrometry and Quantitative Proteome-Disease Relationships. Current Proteomics 6: 246–261.
43. Camon E, Magrane M, Barrell D, Binns D, Fleischmann W, et al. (2003) The Gene Ontology Annotation (GOA) project: implementation of GO in SWISS-PROT, TrEMBL, and InterPro. Genome Res 13: 662–672.
44. Schaffer AA, Aravind L, Madden TL, Shavirin S, Spouge JL, et al. (2001) Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. Nucleic Acids Res 29: 2994–3005.
45. Loewenstein Y, Raimondo D, Redfern OC, Watson J, Frishman D, et al. (2009) Protein function annotation by homology-based inference. Genome Biol 10: 207.
46. Gerstein M, Thornton JM (2003) Sequences and topology. Curr Opin Struct Biol 13: 341–343.
47. Chou KC (2004) Review: Structural bioinformatics and its impact to biomedical science. Current Medicinal Chemistry 11: 2105–2134.
48. Schnell JR, Chou JJ (2008) Structure and mechanism of the M2 proton channel of influenza A virus. Nature 451: 591–595.
49. Wang J, Pielak RM, McClintock MA, Chou JJ (2009) Solution structure and functional analysis of the influenza B proton channel. Nat Struct Mol Biol 16: 1267–1271.
50. Chou KC (2004) Insights from modelling the 3D structure of the extracellular domain of alpha7 nicotinic acetylcholine receptor. Biochemical and Biophysical Research Communication 319: 433–438.
51. Chou KC, Cai YD (2002) Using functional domain composition and support vector machines for prediction of protein subcellular location. Journal of Biological Chemistry 277: 45765–45769.
52. Cai YD, Zhou GP, Chou KC (2003) Support vector machines for predicting membrane protein types by using functional domain composition. Biophysical Journal 84: 3257–3263.
53. Murvai J, Vlahovicek K, Barta E, Pongor S (2001) The SBASE protein domain library, release 8.0: a collection of annotated protein sequence segments. Nucleic Acids Research 29: 58–60.
54. Tatusov RL, Fedorova ND, Jackson JD, Jacobs AR, Kiryutin B, et al. (2003) The COG database: an updated version includes eukaryotes. BMC Bioinformatics 4: 41.
55. Letunic I, Copley RR, Pils B, Pinkert S, Schultz J, et al. (2006) SMART 5: domains in the context of genomes and networks. Nucleic Acids Res 34: D257–260.
56. Finn RD, Mistry J, Schuster-Bockler B, Griffiths-Jones S, Hollich V, et al. (2006) Pfam: clans, web tools and services. Nucleic Acids Res 34: D247–251.
57. Marchler-Bauer A, Anderson JB, Derbyshire MK, DeWeese-Scott C, Gonzales NR, et al. (2007) CDD: a conserved domain database for interactive domain family analysis. Nucleic Acids Res 35: D237–240.
58. Chou KC, Shen HB (2006) Predicting eukaryotic protein subcellular location by fusing optimized evidence-theoretic K-nearest neighbor classifiers. Journal of Proteome Research 5: 1888–1897.
59. Denoeux T (1995) A k-nearest neighbor classification rule based on Dempster-Shafer theory. IEEE Transactions on Systems, Man and Cybernetics 25: 804–813.
60. Chou KC, Zhang CT (1995) Review: Prediction of protein structural classes. Critical Reviews in Biochemistry and Molecular Biology 30: 275–349.
61. Jahandideh S, Abdolmaleki P, Jahandideh M, Asadabadi EB (2007) Novel two-stage hybrid neural discriminant model for predicting proteins structural classes. Biophys Chem 128: 87–93.
62. Jahandideh S, Sarvestani AS, Abdolmaleki P, Jahandideh M, Barfeie M (2007) gamma-Turn types prediction in proteins using the support vector machines. J Theor Biol 249: 785–790.
63. Chen K, Kurgan LA, Ruan J (2008) Prediction of protein structural class using novel evolutionary collocation-based sequence representation. J Comput Chem 29: 1596–1604.
64. Jiang Y, Iglinski P, Kurgan L (2008) Prediction of protein folding rates from primary sequences using hybrid sequence representation. J Comput Chem.
65. Yang JY, Peng ZL, Yu ZG, Zhang RJ, Anh V, et al. (2009) Prediction of protein structural classes by recurrence quantification analysis based on chaos game representation. Journal of Theoretical Biology 257: 618–626.
66. Vilar S, Gonzalez-Diaz H, Santana L, Uriarte E (2009) A network-QSAR model for prediction of genetic-component biomarkers in human colorectal cancer. Journal of Theoretical Biology 261: 449–458.
67. Nanni L, Lumini A (2009) A Further Step Toward an Optimal Ensemble of Classifiers for Peptide Classification, a Case Study: HIV Protease. Protein & Peptide Letters 16: 163–167.
68. Garg A, Bhasin M, Raghava GP (2005) Support vector machine-based method for subcellular localization of human proteins using amino acid compositions, their order, and similarity search. J Biol Chem 280: 14427–14432.