R3P-Loc: A compact multi-label predictor using ridge regression and random projection for protein subcellular localization

Shibiao Wan[a], Man-Wai Mak[a,*], Sun-Yuan Kung[b]

[a]*Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong SAR, China*
[b]*Department of Electrical Engineering, Princeton University, New Jersey, USA.*

Abstract

Locating proteins within cellular contexts is of paramount significance in elucidating their biological functions. Computational methods based on knowledge databases (such as gene ontology annotation (GOA) database) are known to be more efficient than sequence-based methods. However, the predominant scenarios of knowledge-based methods are that (1) knowledge databases typically have enormous size and are growing exponentially, (2) knowledge databases contain redundant information, and (3) the number of extracted features from knowledge databases is much larger than the number of data samples with ground-truth labels. These properties render the extracted features liable to redundant or irrelevant information, causing the prediction systems suffer from overfitting. To address these problems, this paper proposes an efficient multi-label predictor, namely R3P-Loc, which uses two compact databases for feature extraction and applies random projection (RP) to reduce the feature dimensions of an ensemble ridge regression (RR) classifier. Two new compact databases are created from Swiss-Prot and GOA databases. These databases possess almost the same amount of information as their full-size counterparts but with much smaller size. Experimental results on two recent datasets (eukaryote and plant) suggest that R3P-Loc can reduce the dimensions by seven folds and significantly outperforms state-of-the-art predictors. This paper also demonstrates that the compact databases reduce the memory consumption by 39 times without causing degradation in prediction accuracy. For readers' convenience, the R3P-Loc server is available online at `http://bioinfo.eie.polyu.edu.hk/R3PLocServer/`.

Keywords: Multi-location proteins; Compact databases; Protein subcellular localization; Random projection; Multi-label classification.

1. Introduction

Most eukaryotic proteins are synthesized in the cytosol and must be transported to the correct spatiotemporal cellular contexts to perform their biological functions. The knowledge of protein subcellular localization helps biologists elucidate the functions of proteins and identify drug targets [1, 2]. Mislocalization of proteins within cells may lead to a broad range of human diseases, such as breast cancer [3], kidney stone [4], Alzheimer's disease [5], Bartter syndrome [6], primary human liver tumors [7], minor salivary gland tumors [8] and pre-eclampsia [9]. Conventionally, high quality localization databases are obtained by wet-lab experiments such as cell fractionation, fluorescent microscopy imaging and electron microscopy, which are also regarded as gold standard for validating subcellular localization. These methods, however, are laborious and costly, especially for the avalanche of newly discovered protein sequences in the post-genomic era. Therefore, computational methods are required to assist biologists for large-scale protein subcellular localization.

Recent decades have witnessed remarkable progress of computational methods for predicting subcellular localization of proteins, which can be roughly divided into sequence-based and knowledge-based. Sequence-based methods include: (1) sorting-signals based methods [10, 11, 12], such as using signal peptides, which can be predicted by signal peptide predictors like Signal-CF [13] and Signal-3L [14]; (2) amino-acid composition-based methods [15, 16, 17, 18, 19, 20]; and (3) homology-based methods [21, 22, 23]. Knowledge-based methods use information from knowledge databases, such as Gene Ontology (GO)[1] terms [24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34], Swiss-Prot keywords [35, 36], functional domains [37], or PubMed abstracts [38, 39]. Among them, GO-based methods have demonstrated to be superior to methods based on other features [27, 40, 41, 42].

Because some proteins can exist in more than one organelle in a cell [43, 44, 45, 46], recent researches have been focusing on predicting both single- and multi-location proteins. In fact, multi-location proteins play important roles in some metabolic processes that take place in more than one cellular compartment, e.g., fatty acid β-oxidation in the peroxisome and mitochondria, and antioxidant defense in the cytosol, mitochondria and peroxisome [47].

*Corresponding author
Email addresses: `10900600r@connect.polyu.hk` (Shibiao Wan), `enmwmak@polyu.edu.hk` (Man-Wai Mak), `kung@princeton.edu` (Sun-Yuan Kung)

[1] http://www.geneontology.org

Recently, several state-of-the-art multi-label predictors have been proposed, such as Plant-mPLoc [48], Euk-mPLoc 2.0 [49], iLoc-Plant [50], iLoc-Euk [51], mGOASVM [52], HybridGO-Loc [53] and other predictors [54, 55, 56]. They all use the GO information as the features and apply different multi-label classifiers to tackle the multi-label classification problem. However, these GO-based methods are not without disadvantages. Currently the predominant scenarios of GO-based methods are that:

1. The gene ontology annotation (GOA) database,[2] from which these GO-based predictors extract the GO information for classification, is usually in enormous size and is also growing rapidly. For example, in October 2005, the GOA database contains 7,782,748 entries for protein annotations; in March 2011, GOA database contains 82,632,215 entries; and in July 2013, the number of entries increases to 169,603,862, which suggests that in less than 8 years, the number of annotations in GOA database increases 28 times. Even after compressing the GOA database released in July 2013 by removing the repeated pairing of accession numbers (ACs) and GO terms, the number of distinct pairs of AC–GO terms is still as high as 25,441,543. It is expected that searching a database with such a enormous and rapidly-growing size is computationally prohibitive, which makes large-scale subcellular localization by GO-based methods inefficient and even intractable.
2. The GOA database contains many redundant AC entries that will never be used by typical GO-based methods. This is because given a query protein, GO-based methods search for homologous ACs from Swiss-Prot and use these ACs as keys to search against the GOA database for retrieving relevant GO terms. Therefore, those ACs in the GOA database that do not appear in Swiss-Prot are redundant. Among all the ACs in the GOA database, more than 90% are in this category. This calls for a more compact GO-term database that excludes these redundant entries.
3. The number of extracted GO features from the GOA database is much larger than the number of proteins that are relevant to the prediction task. For example, Xiao et. al. [57] extracted GO information of 207 proteins from the GOA database; the resulting feature vectors have 11,118 dimensions, which suggests that the number of features is more than 50 times the number of proteins. It is likely that among the large number of features, many of them contain redundant or irrelevant information, causing the prediction systems suffer from overfitting and thus degrading the prediction performance.

To tackle the problems mentioned above, this paper proposes an efficient and compact multi-label predictor, namely **R3P-Loc**, which uses **R**idge **R**egression and **R**andom **P**rojection for predicting subcellular **Loc**alization of both single-label and multi-label proteins. Instead of using the Swiss-Prot and GOA databases, R3P-Loc uses two newly-created compact databases, namely ProSeq and ProSeq-GO, for GO information transfer. The ProSeq database is a sequence database in which each amino acid sequence has at least one GO term annotated to it. The ProSeq-GO comprises GO terms annotated to the protein sequences in the ProSeq database. An important property of the ProSeq and ProSeq-GO databases is that they are much smaller than the Swiss-Prot and GOA databases, respectively.

Given a query protein, a set of GO-terms are retrieved by searching against the ProSeq-GO database using the accession numbers of homologous proteins as the searching keys, where the homologous proteins are obtained from BLAST searches, using ProSeq as the sequence database. The frequencies of GO occurrences are used to formulate frequency vectors, which are projected onto much lower-dimensional space by random matrices whose elements conform to a distribution with zero mean and unit variance. Subsequently, the dimension-reduced feature vectors are classified by a multi-label ridge regression classifier. Results on two recent benchmark datasets demonstrate that R3P-Loc substantially outperforms other existing state-of-the-art predictors.

According to a recent comprehensive review [58], the establishment of a statistical protein predictor involves the following five steps: (i) construction of a valid dataset for training and testing the predictor; (ii) formulation of effective mathematical expressions for converting proteins' characteristics to feature vectors that are relevant to the prediction task; (iii) development of classification algorithm for discriminating the feature vectors; (iv) evaluation of cross-validation tests for measuring the performance of the predictor; and (v) deployment of a user-friendly, publicly accessible web-server for other researchers to use and validate the prediction method. These steps are also carried out in a series of recent publications [59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71], which are further elaborated below.

2. Legitimacy of Using GO Information

First, some researchers may be skeptical about using GO information for protein subcellular localization, because the cellular component GO terms have already been annotated with cellular component categories. The GO comprises three orthogonal categories whose terms describe the cellular components, biological processes, and molecular functions of gene products. These researchers argue that the only thing that needs to be done is to create a lookup table using the cellular component GO terms as the keys and the component categories as the hashed values. Such a naive solution, however, is undesirable and will lead to poor performance, as shown and explained in our previous studies [52, 42].

Second, some researchers also disprove the effectiveness of GO-based methods by claiming that only cellular component GO terms are necessary and that GO terms in the other two categories play no role in determining the subcellular localization. This concern has been explicitly addressed by Lu and Hunter [72], who demonstrated that GO molecular function terms are also predictive of subcellular localization, particularly

[2] http://www.ebi.ac.uk/GOA

for nucleus, extracellular space, membrane, mitochondrion, endoplasmic reticulum and Golgi apparatus. The in-depth analysis of the correlation between the molecular function GO terms and localization in [72] provides an explanation of why GO-based methods outperform sequence-based methods.

Third, even though GO-based methods can predict novel proteins based on the GO information obtained from their homologous proteins [52, 42], some researchers still argue that the prediction is equivalent to simply using the annotated localization of the homologs (i.e., using BLAST with homologous transfer). This claim is clearly proved to be untenable in our previous study [42], which demonstrates that GO-based methods remarkably outperform methods that only use BLAST and homologous transfer (in Table 4 of [42]). Besides, Briesemeister et al. [73] also suggest that using BLAST alone is not sufficient for reliable prediction.

Moreover, as suggested by Chou [74], as long as the input of query proteins for predictors is the sequence information without any GO annotation information and the output is the subcellular localization information, there is no difference between non-GO based methods and GO-based methods, which should be regarded as equally legitimate for subcellular localization.

Some other papers [75, 76] also provide strong arguments supporting the legitimacy of using GO information for subcellular localization. In particular, as suggested by [76], the good performance of GO-based methods is due to the fact that the feature vectors in the GO space can better reflect their subcellular locations than those in the Euclidean space or any other simple geometric space.

3. Creation of Compact Databases

Typically, for a query protein, an efficient predictor should be able to deal with two possible cases: (1) the accession number (AC) is known and (2) only the amino acid sequence is known. For proteins with known ACs, their respective GO terms are retrieved from a database containing GO terms (i.e., GOA database) using the ACs as the searching keys. For a protein without an AC, its amino acid sequence is presented to BLAST [77] to find its homologs against a database containing protein amino acid sequences (i.e., Swiss-Prot), whose ACs are then used as keys to search against the GO-term database.

While the GOA database allows us to associate the AC of a protein with a set of GO terms, for some novel proteins, neither their ACs nor the ACs of their top homologs have any entries in the GOA database; in other words, no GO terms can be retrieved by their ACs or the ACs of their top homologs. In such case, some predictors use back-up methods that rely on other features, such as pseudo-amino-acid composition [15] and sorting signals [78]; some predictors [42, 52] use a successive-search strategy to avoid null GO vectors. However, these strategies may lead to poor performance and increase computation and storage complexity.

To address this problem, we created two small yet efficient databases: ProSeq and ProSeq-GO. The former is a sequence database and the latter is a GO-term database. The procedures

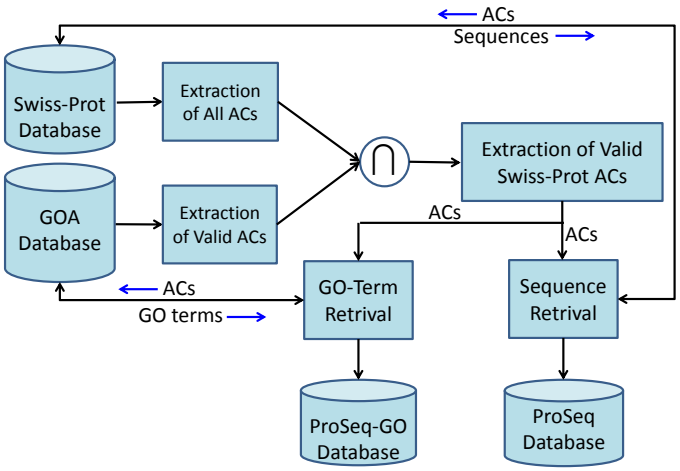


Figure 1: Procedures of creating the compact databases (ProSeq and ProSeq-GO). *AC*: accession numbers; *GO*: gene ontology; *GOA database*: gene ontology annotation database.

of creating these databases are shown in Fig. 1. The procedure extracts accession numbers from two different sources: Swiss-Prot and GOA database. Specifically, all of the ACs in the Swiss-Prot database and the *valid* ACs in the GOA database are extracted. Here, an AC is considered valid when it has at least one GO term annotated to it. Then, the common ACs that appear in both sets are selected (the $\bigcap$ symbol in Fig. 1). These ACs are regarded as 'valid Swiss-Prot ACs'; each of them corresponds to at least one GO term in the GOA database. Next, using these valid ACs, their corresponding amino-acid sequences can be retrieved from the Swiss-Prot database, constituting a new sequence database, which we call 'ProSeq database'; similarly, using these valid ACs, their corresponding GO terms can be retrieved from the GOA database, constituting a new GO-term database, which we call 'ProSeq-GO database'. In this work, we created ProSeq and ProSeq-GO databases from the Swiss-Prot and GOA databases released in July 2013. The ProSeq-GO database has 513,513 entries while the GOA database has 25,441,543 entries; the ProSeq database has 513,513 protein sequences while the Swiss-Prot database has 540,732 protein sequences.

4. Feature Extraction

The feature extraction of R3P-Loc includes two steps: (1) retrieval of GO terms; and (2) construction of GO vectors.

4.1. Retrieval of GO Terms

Similar to our earlier predictors [52, 42, 53], R3P-Loc can deal with two possible cases: (1) the accession number (AC) is known and (2) only the amino acid sequence is known. Instead of using the Swiss-Prot and GOA databases, R3P-Loc uses ProSeq and ProSeq-GO to retrieve GO terms (See Fig. 3), which can guarantee that valid GO terms can always be found for a query protein with known amino-acid sequence.

4.2. Construction of GO Vectors

Given a dataset, the GO terms of all of its proteins are retrieved by using the procedures described in Section 4.1. Similar to our earlier works [42, 52], the GO frequency information is used to construct GO feature vectors. Specifically, the GO vector $\mathbf{q}_i$ of the i-th protein $\mathbb{Q}_i$ is defined as:

$$\mathbf{q}_i = [b_{i,1}, \cdots, b_{i,j}, \cdots, b_{i,T}]^\mathsf{T}, b_{i,j} = \begin{cases} f_{i,j} & \text{, GO hit} \\ 0 & \text{, otherwise} \end{cases} \quad (1)$$

where $f_{i,j}$ is the number of occurrences of the j-th GO term (term-frequency) in the i-th protein sequence. Detailed information can be found in [52, 42].

5. Random Projection

The key idea of RP arises from the Johnson-Lindenstrauss lemma [79]:

Lemma 1. (Johnson and Lindenstrauss [79]). *Given $\epsilon > 0$, a set $\mathcal{X}$ of N points in $\mathcal{R}^T$, and a positive integer $d \geq d_0 = O(\log N/\epsilon^2)$, there exists $f : \mathcal{R}^T \to \mathcal{R}^d$ such that*

$$(1-\epsilon)\|u-v\|^2 \leq \|f(u)-f(v)\|^2 \leq (1+\epsilon)\|u-v\|^2$$

for all $u, v \in \mathcal{X}$. A proof can be found in [80].

The lemma suggests that if points in a high-dimensional space are projected onto a randomly selected subspace of suitable dimension, the distances between the points are approximately preserved.

Specifically, the original T-dimensional data is projected onto a d-dimensional ($d \ll T$) subspace, using a $d \times T$ random matrix $\mathbf{R}$ whose columns are unit lengths. A vector $\mathbf{q}_i \in \mathcal{R}^T$ is projected to:

$$\mathbf{q}_i^{RP} = \frac{1}{\sqrt{d}}\mathbf{R}\mathbf{q}_i, \quad (2)$$

where $1/\sqrt{d}$ is a scaling factor, $\mathbf{q}_i^{RP}$ is the projected vector after RP, and $\mathbf{R}$ is a random $d \times T$ matrix.

The choice of the random matrix $\mathbf{R}$ is one of the key points of interest. Practically, as long as the elements $r_{h,j}$ of $\mathbf{R}$ conforms to any distributions with zero mean and unit variance, $\mathbf{R}$ will give a mapping that satisfies the Johnson-Lindenstrauss lemma [81]. For computational simplicity and also the requirement of sparseness, we adopted a simple distribution proposed by Achlioptas [82] for the elements $r_{h,j}$ as follows:

$$r_{h,j} = \sqrt{3} \times \begin{cases} +1 & \text{with probability 1/6,} \\ 0 & \text{with probability 2/3,} \\ -1 & \text{with probability 1/6.} \end{cases} \quad (3)$$

It is easy to verify that Eq. 3 conforms to a distribution with zero mean and unit variance [82] and that $\mathbf{R}$ is sparse.

As stated in [83], if $\mathbf{R}$ and $\mathbf{q}_i$ satisfy the conditions of the basis pursuit theorem (i.e., both are sparse in a fixed basis), then $\mathbf{q}_i$ can be reconstructed perfectly from a vector that lies in a lower-dimensional space. In fact, the GO vectors and our projection matrix $\mathbf{R}$ satisfy these conditions. As shown in Fig. 2,

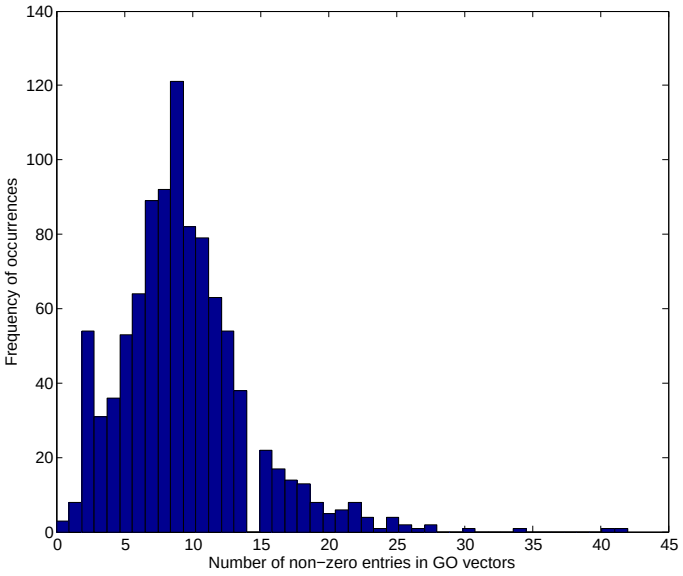


Figure 2: Histogram illustrating the distribution of the number of non-zero entries (spareness) in the GO vectors with dimensionality 1541. The histogram is plotted up to 45 non-zero entries in the GO vectors because among the 978 proteins in the dataset, none of their GO vectors have more than 45 non-zero entries.

the number of non-zero entries in the GO vectors tends to be small (i.e. sparse) when compared to the dimension of the GO vectors. Among the 978 proteins in the plant dataset (See Fig. 4), a majority of them only have 9 non-zero entries in the 1541-dimensional vectors, and the largest number of non-zero entries is only 45. These statistics suggest that the GO vectors $\mathbf{q}_i$ in Eq. 2 are very sparse.

6. Ensemble Multi-label Ridge Regression Classifier

6.1. Single-Label Ridge Regression

Ridge regression (RR) is a simple yet effective linear regression model, which has been applied to many domains [84, 85, 86]. Here we apply RR to classification. Suppose for a two-class single-label problem, we are given a set of training data $\{\mathbf{x}_i, y_i\}_{i=1}^N$, where $\mathbf{x}_i \in \mathcal{R}^{T+1}$ and $y_i \in \{0, 1\}$. In our case, $\mathbf{x}_i = \begin{bmatrix} 1 \\ \mathbf{q}_i^{RP} \end{bmatrix}$, where $\mathbf{q}_i^{RP}$ is defined in Eq. 2. Generally speaking, an RR model is to impose an L_2-style regularization to ordinary least squares (OLS), namely minimizing the empirical loss $l(\boldsymbol{\beta})$ as:

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{N}(y_i - f(\mathbf{x}_i))^2 = \sum_{i=1}^{N}(y_i - \sum_{j=1}^{T+1}\beta_j x_{i,j})^2, \quad (4)$$

subject to

$$\sum_{j=1}^{T+1}\beta_j^2 \leq s,$$

where $s > 0$, $x_{i,j}$ is the j-th element of $\mathbf{x}_i$ and $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_j, \ldots, \beta_{T+1}]^\mathsf{T}$ is the ridge vector to be optimized. Eq. 4

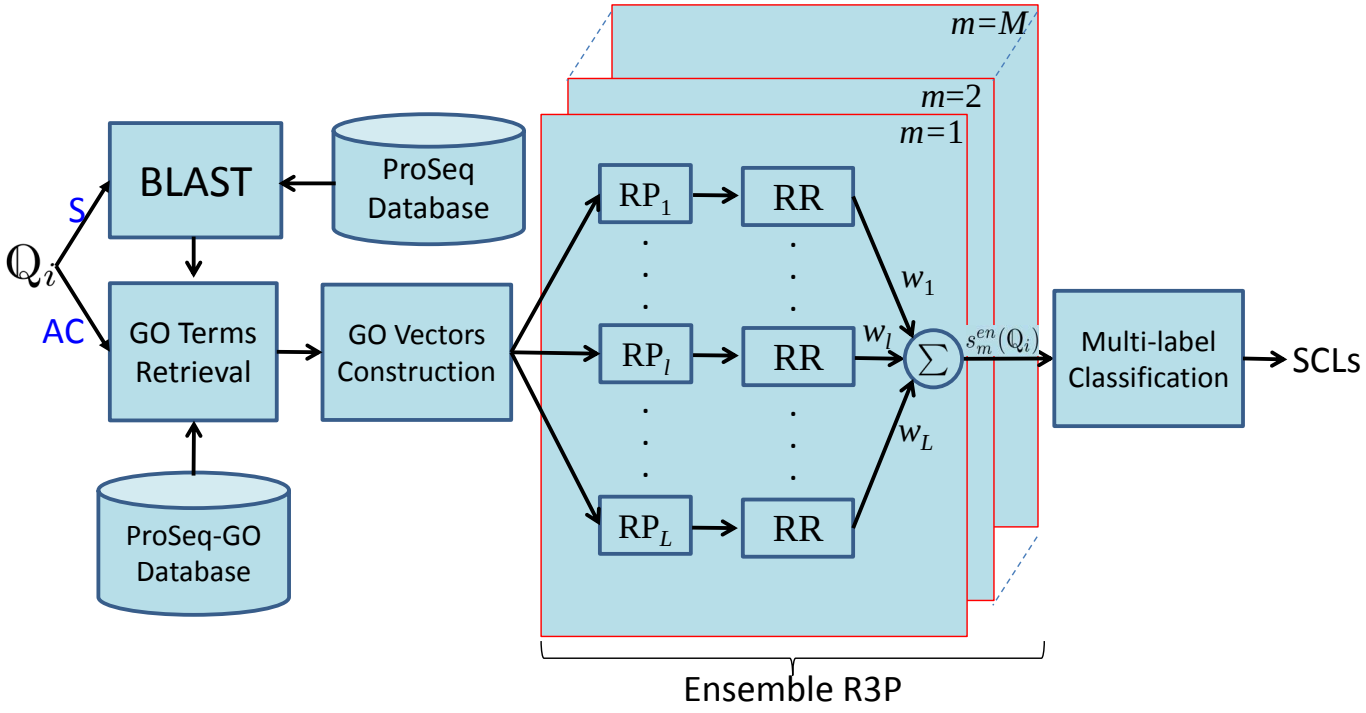


Figure 3: Flowchart of R3P-Loc. $\mathbb{Q}_i$: the i-th query protein; S: protein sequence; AC: protein accession number; *ProSeq*/*ProSeq-GO*: the proposed compact sequence and GO databases, respectively; *RP*: random projection; *RR*: ridge regression scoring (Eq. 8); *Ensemble R3P*: ensemble ridge regression and random projection; w_1, w_l *and* w_L: the 1-st, l-th and L-th weights in Eq. 9; $s_m^{en}(\mathbb{Q}_i)$: the ensemble score in Eq. 9; *SCLs*: subcellular location(s).

is equivalent to minimize the following equation:

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{N}(y_i - \boldsymbol{\beta}^{\mathsf{T}}\mathbf{x}_i)^2 + \lambda\boldsymbol{\beta}^{\mathsf{T}}\boldsymbol{\beta}, \tag{5}$$

where $\lambda > 0$ is a penalized parameter to control the degree of regularization. Then after optimization, $\boldsymbol{\beta}$ is given as:

$$\boldsymbol{\beta} = (\mathbf{X}^{\mathsf{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}\mathbf{y}, \tag{6}$$

where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_N]^{\mathsf{T}}$,$\mathbf{y} = [y_1, \ldots, y_i, \ldots, y_N]^{\mathsf{T}}$, and $\mathbf{I}$ is a $(T+1)\times(T+1)$ identity matrix.

6.2. Multi-label Ridge Regression

In an M-class multi-label problem, the training data set is written as $\{\mathbf{x}_i, \mathcal{Y}_i\}_{i=1}^N$, where $\mathbf{x}_i \in \mathcal{R}^{T+1}$ and $\mathcal{Y}_i \subset \{1, 2, \ldots, M\}$ is a set which may contain one or more labels. M independent binary one-vs-rest RRs are trained, one for each class. The labels $\{\mathcal{Y}_i\}_{i=1}^N$ are converted to *transformed labels* [52] $y_{i,m} \in \{-1, 1\}$, where $i = 1, \ldots, N$, and $m = 1, \ldots, M$. Then, Eq. 7 is extended to:

$$\boldsymbol{\beta}_m = (\mathbf{X}^{\mathsf{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}\mathbf{y}_m, \tag{7}$$

where $m = 1, \ldots, M$, $\mathbf{y}_m$ are vectors whose elements are $\{y_{i,m}\}_{i=1}^N$.

The projected GO vectors obtained from Eq. 2 are used for training multi-label one-vs-rest ridge regression (RR) classifiers. Specifically, for an M-class problem (here M is the number of subcellular locations), M independent binary RRs are trained, one for each class. Then, given the i-th query protein $\mathbb{Q}_i$, the score of the m-th RR is:

$$s_m(\mathbb{Q}_i) = {\boldsymbol{\beta}_m}^{\mathsf{T}}\mathbf{x}_i, \text{ where } \mathbf{x}_i = \begin{bmatrix} 1 \\ \mathbf{q}_i^{RP} \end{bmatrix}. \tag{8}$$

Since $\mathbf{R}$ is a random matrix, the scores in Eq. 8 for each application of RP will be different. To construct a robust classifier, we fused the scores for several applications of RP and obtained an ensemble classifier, where the ensemble score of the m-th RR for the i-th query protein is given as follows:

$$s_m^{en}(\mathbb{Q}_i) = \sum_{l=1}^{L} w_l \cdot s_m^{(l)}(\mathbb{Q}_i), \tag{9}$$

where $\sum_{l=1}^{L} w_l = 1$, $s_m^{(l)}(\mathbb{Q}_i)$ represents the score of the m-th RR for the i-th protein via the l-th application of RP, L is the total number of applications of RP, and $\{w_l\}_{l=1}^L$ are the weights. For simplicity, here we set $w_l = 1/L, l = 1, \ldots, L$. We refer L as 'ensemble size' in the sequel. Unless stated otherwise, the ensemble size was set to 10 in our experiments, i.e., $L = 10$. Note that instead of mapping the original data into an Ld-dim vector, the ensemble RP projects it into L d-dim vectors.

To predict the subcellular locations of datasets containing both single-label and multi-label proteins, a decision scheme for multi-label RR classifiers should be used. Unlike the single-label problem where each protein has one predicted label only, a multi-label protein should have more than one predicted labels. In this paper, we used the decision scheme described in mGOASVM [52]. In this scheme, the predicted subcellular location(s) of the i-th query protein are given by:

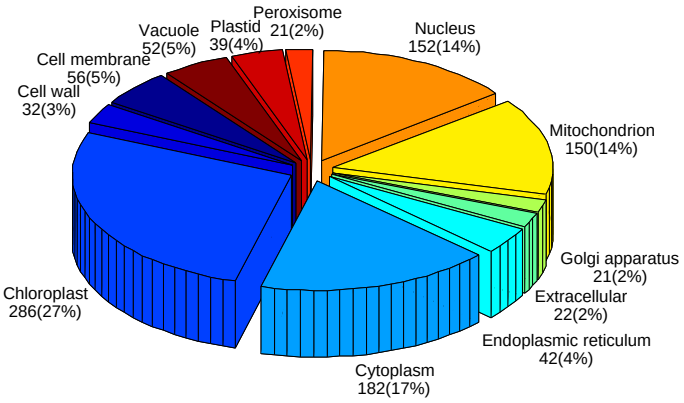


Figure 4: Breakdown of the plant dataset. The number of proteins shown in each subcellular location represents the number of 'locative proteins' [57, 52]. Here, 978 actual proteins have 1055 locative proteins. The plant proteins are distributed in 12 subcellular locations, including cell membrane, cell wall, chloroplast, cytoplasm, endoplasmic reticulum, extracellular, Golgi apparatus, mitochondrion, nucleus, peroxisome, plastid and vacuole.

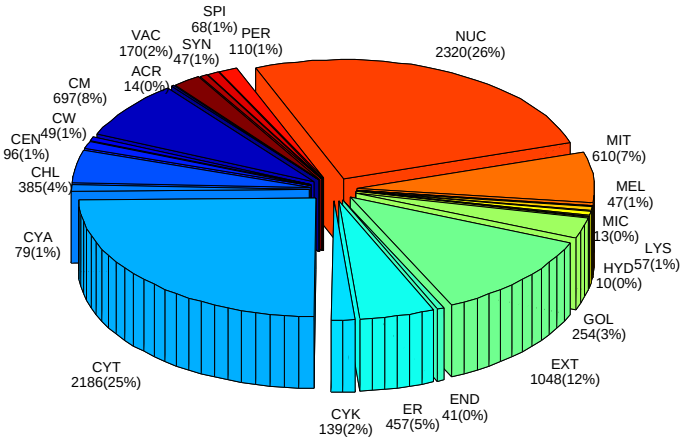


Figure 5: Breakdown of the eukaryotic dataset. The number of proteins shown in each subcellular location represents the number of 'locative proteins' [57, 52]. Here, 7766 actual proteins have 8897 locative proteins. The eukaryotic proteins are distributed in 22 subcellular locations, including acrosome (ACR), cell membrane (CM), cell wall (CW), centrosome (CEN), chloroplast (CHL), cyanelle (CYA), cytoplasm (CYT), cytoskeleton (CYK), endoplasmic reticulum (ER), endosome (END), extracellular (EXT), Golgi apparatus (GOL), hydrogenosome (HYD), lysosome (LYS), melanosome (MEL), microsome (MIC), mitochondrion (MIT), nucleus (NUC), peroxisome (PER), spindle pole body (SPI), synapse (SYN) and vacuole (VAC).

$$\mathcal{M}^*(\mathbb{Q}_i) = \begin{cases} \bigcup_{m=1}^{M} \{m : s_m^{en}(\mathbb{Q}_i) > 0\}, \text{ where } \exists\, s_m^{en}(\mathbb{Q}_i) > 0\,; \\ \arg\max_{m=1}^{M} s_m^{en}(\mathbb{Q}_i), \text{ otherwise.} \end{cases} \tag{10}$$

For ease of comparison, we refer to the proposed ensemble classifier with this multi-label decision scheme as R3P-Loc. The flowchart of R3P-Loc is shown in Fig. 3.

Similar to other predictors [64, 65, 67, 68, 87, 69, 71, 88, 60], for users' convenience, a step-by-step guide for the R3P-Loc web-server is provided, which is included in the supplementary materials of the R3P-Loc web-server available online.

7. Experiments

7.1. Datasets

In this paper, a plant dataset [50] and a eukaryotic dataset [49] were used to evaluate the performance of R3P-Loc. The plant and the eukaryotic datasets were both created from Swiss-Prot 55.3. The plant dataset contains 978 plant proteins distributed in 12 locations. Of the 978 plant proteins, 904 belong to one subcellular location, 71 to two locations, 3 to three locations and none to four or more locations. The eukaryotic dataset contains 7766 eukaryotic proteins distributed in 22 locations. Of the 7766 eukaryotic proteins, 6687 belong to one subcellular location, 1029 to two locations, 48 to three locations, 2 to four locations and none to five or more locations. The sequence identity of both datasets was cut off at 25%. The breakdown of these two datasets are listed in Figs. 4 and 5. As can be seen, both datasets are multi-class distributed and imbalanced.

7.2. Performance Metrics

Compared to traditional single-label classification, multi-label classification requires more complicated performance metrics to better reflect the multi-label capabilities of classifiers. These measures include *Accuracy, Precision, Recall, F1-score (F1)* and *Hamming Loss (HL)*. Specifically, denote $\mathcal{L}(\mathbb{Q}_i)$ and $\mathcal{M}(\mathbb{Q}_i)$ as the true label set and the predicted label set for the i-th protein $\mathbb{Q}_i$ ($i = 1, \ldots, N$), respectively.[3] Then the five measurements are defined as follows:

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{|\mathcal{M}(\mathbb{Q}_i) \cup \mathcal{L}(\mathbb{Q}_i)|}\right) \tag{11}$$

$$Precision = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{|\mathcal{M}(\mathbb{Q}_i)|}\right) \tag{12}$$

$$Recall = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{|\mathcal{L}(\mathbb{Q}_i)|}\right) \tag{13}$$

$$F1 = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{2|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{|\mathcal{M}(\mathbb{Q}_i)|+|\mathcal{L}(\mathbb{Q}_i)|}\right) \tag{14}$$

$$HL = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|\mathcal{M}(\mathbb{Q}_i) \cup \mathcal{L}(\mathbb{Q}_i)|-|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{M}\right) \tag{15}$$

where $|\cdot|$ means counting the number of elements in the set therein and $\cap$ represents the intersection of sets.

Accuracy, Precision, Recall and *F1* indicate the classification performance. The higher the measures, the better the prediction performance. Among them, *Accuracy* is the most commonly used criteria. *F1-score* is the harmonic mean of *Precision* and *Recall*, which allows us to compare the performance of classification systems by taking the trade-off between

[3] Here, $N = 978$ for the plant dataset and $N = 7766$ for the eukaryotic dataset.

Precision and *Recall* into account. The *Hamming Loss (HL)* [89, 90] is different from other metrics. As can be seen from Eq. 15, when all of the proteins are correctly predicted, i.e., $|\mathcal{M}(\mathbb{Q}_i) \cup \mathcal{L}(\mathbb{Q}_i)| = |\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|$ $(i = 1, \ldots, N)$, then $HL = 0$; whereas, other metrics will be equal to 1. On the other hand, when the predictions of all proteins are completely wrong, i.e., $|\mathcal{M}(\mathbb{Q}_i) \cup \mathcal{L}(\mathbb{Q}_i)| = M$ and $|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)| = 0$, then $HL = 1$; whereas, other metrics will be equal to 0. Therefore, the lower the *HL*, the better the prediction performance.

Two additional measurements [57, 52] are often used in multi-label subcellular localization prediction. They are overall locative accuracy (*OLA*) and overall actual accuracy (*OAA*). The former is given by:

$$OLA = \frac{1}{\sum_{i=1}^{N} |\mathcal{L}(\mathbb{Q}_i)|} \sum_{i=1}^{N} |\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|, \tag{16}$$

and the overall actual accuracy (*OLA*) is:

$$OAA = \frac{1}{N} \sum_{i=1}^{N} \Delta[\mathcal{M}(\mathbb{Q}_i), \mathcal{L}(\mathbb{Q}_i)] \tag{17}$$

where

$$\Delta[\mathcal{M}(\mathbb{Q}_i), \mathcal{L}(\mathbb{Q}_i)] = \begin{cases} 1 & \text{, if } \mathcal{M}(\mathbb{Q}_i) = \mathcal{L}(\mathbb{Q}_i) \\ 0 & \text{, otherwise.} \end{cases} \tag{18}$$

According to Eq. 16, a locative protein is considered to be correctly predicted if any of the predicted labels matches any labels in the true label set. On the other hand, Eq. 17 suggests that an actual protein is considered to be correctly predicted only if *all* of the predicted labels match those in the true label set exactly. For example, for a protein coexist in, say, three subcellular locations, if only two of the three are correctly predicted, or the predicted result contains a location not belonging to the three, the prediction is considered to be incorrect. In other words, when and only when all the subcellular locations of a query protein are exactly predicted without any overprediction or underprediction, can the prediction be considered as correct. Therefore, *OAA* is a more stringent measure as compared to *OLA*. *OAA* is also more objective than *OLA*. This is because locative accuracy is liable to give biased performance measure when the predictor tends to over-predict, i.e., giving large $|\mathcal{M}(\mathbb{Q}_i)|$ for many $\mathbb{Q}_i$. In the extreme case, if every protein is predicted to have all of the M subcellular locations, according to Eq. 16, the *OLA* is 100%. But obviously, the predictions are wrong and meaningless. On the contrary, *OAA* is 0% in this extreme case, which definitely reflects the real performance.

Among all the metrics mentioned above, *OAA* is the most stringent and objective. This is because if some (but not all) of the subcellular locations of a query protein are correctly predict, the numerators of the other 4 measures (Eqs. 11 to 16) are non-zero, whereas the numerator of *OAA* in Eq. 17 is 0 (thus contribute nothing to the frequency count).

In statistical prediction, there are three methods that are often used for testing the generalization capabilities of predictors: independent test, subsampling test (or K-fold cross-validation) and jackknife test [91]. The jackknife test is considered to be the most rigorous and bias-free method that can always yield a unique outcome for the predictors as elaborated by Eqs. 28–30 in [58]. Accordingly, the jackknife test has been widely used by researchers to examine the power of various predictors [92, 93, 94, 95, 96].

Table 1: Performance of R3P-Loc on the proposed compact databases based on the jackknife test using the eukaryotic dataset. *SCL*: subcellular location; *ER*: endoplasmic reticulum; *SPI*: spindle pole body; *OAA*: overall actual accuracy; *OLA*: overall locative accuracy; *F1*: F1-score; *HL*: Hamming loss; *Memory Requirement*: memory required for loading the GO-term database; *No. of Database Entries*: number of entries in the corresponding GO-term database; *No. of Distinct GO Terms*: Number of distinct GO terms found by using the corresponding GO-term database.

Label	SCL	Jackknife Test Locative Accuracy (LA)	
		Swiss-Prot + GOA	ProSeq + ProSeq-GO
1	Acrosome	2/14 = 0.143	2/14 = 0.143
2	Cell membrane	523/697 = 0.750	525/697 = 0.753
3	Cell wall	46/49 = 0.939	45/49 = 0.918
4	Centrosome	65/96 = 0.677	65/96 = 0.677
5	Chloroplast	375/385 = 0.974	375/385 = 0.974
6	Cyanelle	79/79 = 1.000	79/79 = 1.000
7	Cytoplasm	1964/2186 = 0.898	1960/2186 = 0.897
8	Cytoskeleton	50/139 = 0.360	53/139 = 0.381
9	ER	424/457 = 0.928	426/457 = 0.932
10	Endosome	12/41 = 0.293	12/41 = 0.293
11	Extracellular	968/1048 = 0.924	969/1048 = 0.925
12	Golgi apparatus	209/254 = 0.823	208/254 = 0.819
13	Hydrogenosome	10/10 = 1.000	10/10 = 1.000
14	Lysosome	47/57 = 0.825	47/57 = 0.825
15	Melanosome	9/47 = 0.192	10/47 = 0.213
16	Microsome	1/13 = 0.077	1/13 = 0.077
17	Mitochondrion	575/610 = 0.943	576/610 = 0.944
18	Nucleus	2169/2320 = 0.935	2157/2320 = 0.930
19	Peroxisome	103/110 = 0.936	104/110 = 0.946
20	SPI	47/68 = 0.691	42/68 = 0.618
21	Synapse	26/47 = 0.553	26/47 = 0.553
22	Vacuole	157/170 = 0.924	156/170 = 0.918
OAA		6191/7766 = 0.797	6201/7766 = **0.799**
OLA		7861/8897 = **0.884**	7848/8897 = 0.882
Accuracy		**0.859**	**0.859**
Precision		**0.882**	**0.882**
Recall		**0.899**	0.898
F1		**0.880**	**0.880**
HL		**0.013**	**0.013**
Memory Requirement		22.5G	**0.6G**
No. of Database Entries		25.4 million	**0.5 million**
No. of Distinct GO Terms		**10808**	10775

8. Results and Discussions

8.1. Performance on the Compact Databases

Table 1 compares the subcellular localization performance of R3P-Loc under two different configurations. The column "Swiss-Prot + GOA" shows the performance when Swiss-prot and the GOA database were used as the data sources for BLAST search and GO terms retrieval in Fig. 3, whereas the column "ProSeq + ProSeq-GO" shows the performance when the proposed compact databases were used instead. As can be seen, the performances of the two configurations are almost the same,

which clearly suggests that the compact databases can be used in place of the large Swiss-Prot and GOA database.

The bottom panel of Table 1 compares the implementation requirements and the number of distinct GO terms (dimension of GOA vectors) of R3P-Loc under the two configurations. In order to retrieve the GO terms in constant time (i.e., complexity $O(1)$) regardless of the database size, the AC to GO-terms mapping was implemented as a hash table in memory. This instantaneous retrieval, however, comes with a price: The hash table consumes considerable amount of memory when the database size increases. Specifically, to load the whole GOA database released in March 2011, only 15 gigabytes of memory is required; the memory consumption rapidly increases to 22.5 gigabytes if the GOA database released in July 2013 is loaded. The main reason is that this release of GOA database contains 25 million entries. However, as shown in Table 1, the number of entries reduces to half a million if ProSeq-GO is used instead, which amounts to a reduction of 39 times in memory consumption. The small number of AC entries in ProSeq-GO results in a small memory footprint. Despite the small number of entries, the number of distinct GO terms in this compact GO database is almost the same as that in the big GOA database. This explains why using ProSeq and ProSeq-GO can achieve almost the same performance as using Swiss-Prot and the original GOA database.

8.2. Effect of Dimensions and Ensemble Size

Fig. 6(a) shows the performance of R3P-Loc at different projected dimensions and ensemble sizes of random projection on the plant dataset. The dimensionality of the original feature vectors is 1541. The yellow dotted plane represents the performance using only multi-label ridge regression classifiers, namely the performance without random projection. For ease of comparison, we refer it to as RR-Loc. The mesh with blue (red) surfaces represent the projected dimensions and ensemble sizes at which the R3P-Loc performs better (poorer) than RR-Loc. As can be seen, there is no red region across all dimensions (200 to 1200) and all ensemble sizes (2 to 10), which means that the ensemble R3P-Loc always performs better than RR-Loc. The results suggest that using ensemble random projection can always boost the performance of RR-Loc. Similar conclusions can be drawn from Fig. 6(b), which shows the performance of R3P-Loc at different projected dimensions and ensemble sizes of random projection on the eukaryotic dataset. The difference is that the original dimension of the feature vectors is 10,775, which means that R3P-Loc performs better than RR-Loc even when the feature dimension is reduced by almost 10∼100 times.

Fig. 7(a) compares the performance of R3P-Loc with mGOASVM [52] at different projected dimensions and ensemble sizes of random projection on the plant dataset. The green dotted plane represents the accuracy of mGOASVM, which is a constant for all projected dimensions and ensemble size. The mesh with blue (red) surfaces represent the projected dimensions and ensemble sizes at which the ensemble R3P-Loc performs better (poorer) than mGOASVM. As can be seen, R3P-Loc performs better than mGOASVM throughout all dimensions (200 to 1400) when the ensemble size is more than 4. On the other hand, when the ensemble size is less than 2, the performance of R3P-Loc is worse than mGOASVM for almost all the dimensions. These results suggest that a large enough ensemble size is important for boosting the performance of R3P-Loc. Fig. 7(b) compares the performance of R3P-Loc with mGOASVM on the eukaryotic dataset. As can be seen, R3P-Loc performs better than mGOASVM when the dimension is larger than 300 and the ensemble size is no less than 3 or the dimension is larger than 500 and the ensemble size is no less than 2. These experimental results suggest that a large enough projected dimension is also necessary for improving the performance of R3P-Loc.

8.3. Performance of Ensemble Random Projection

Fig. 8(a) shows the performance statistics of R3P-Loc based on the jackknife test at different feature dimensions, when the ensemble size (L in Eq. 9) is fixed to 1, which we refer to as 1-R3P-Loc. We created ten 1-R3P-Loc classifiers, each with a different RP matrix. The result shows that even the highest accuracy of the ten 1-R3P-Loc is lower than that of R3P-Loc for all dimensions (200 to 1400). This suggests that the ensemble random projection can significantly boost the performance of R3P-Loc. Similar conclusions can also be drawn from Fig. 8(b), which shows the performance statistics of R3P-Loc on the eukaryotic dataset.

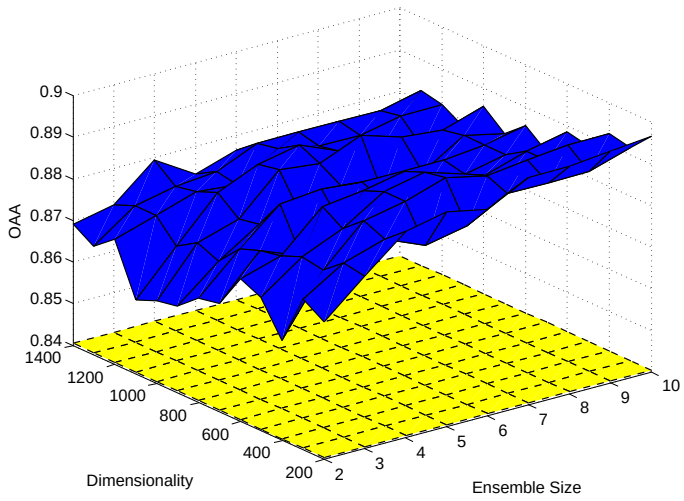
8.4. Comparing with State-of-the-Art Predictors

Table 2 and Table 3 compare the performance of R3P-Loc against several state-of-the-art multi-label predictors on the plant and eukaryotic dataset. All of the predictors use the information of GO terms as features. From the classification perspective, both Plant-mPLoc [48] and Euk-mPLoc 2.0 [49] use an ensemble OET-KNN (optimized evidence-theoretic K-nearest neighbors) classifier; both iLoc-Plant [50] and iLoc-Euk [51] use a multi-label KNN classifier; mGOASVM [52] uses a multi-label SVM classifier;[4] and the proposed R3P-Loc uses ensemble RP and ridge regression classifiers.

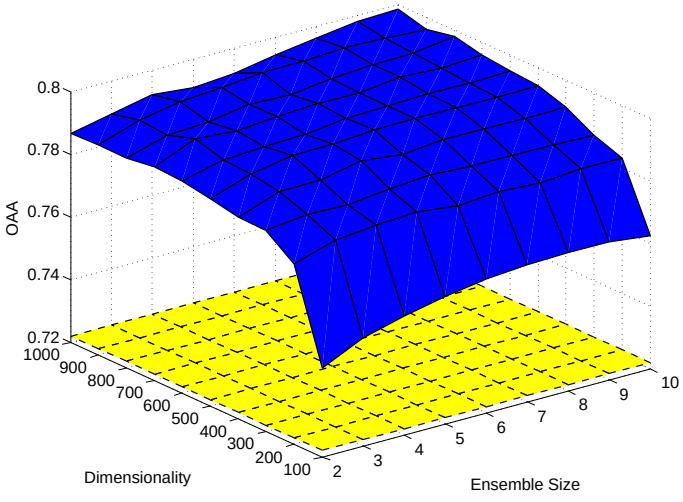
As shown in Table 2, R3P-Loc performs significantly better than Plant-mPLoc and iLoc-Plant. Both the *OLA* and *OAA* of R3P-Loc are more than 20% (absolute) higher than iLoc-Plant. When comparing with mGOASVM, the *OAA* of R3P-Loc is more than 2% (absolute) higher than that of mGOASVM, although a bit less than mGOASVM on the *OLA* and *Recall*. In terms of *Accuracy, Precision, F1* and *HL*, R3P-Loc performs better than mGOASVM. The results suggest that the proposed R3P-Loc performs better than the state-of-the-art classifiers. The individual locative accuracies of R3P-Loc are remarkably higher than that of Plant-mPLoc, iLoc-Plant, and are comparable to mGOASVM.

Similar conclusions can be drawn from Table 3, which compares R3P-Loc with state-of-the-art predictors on the eukaryotic dataset. R3P-Loc performs significantly better than Euk-mPLoc 2.0 and iLoc-Euk in terms of all the measures. And

[4] We performed mGOASVM on the eukaryotic dataset.

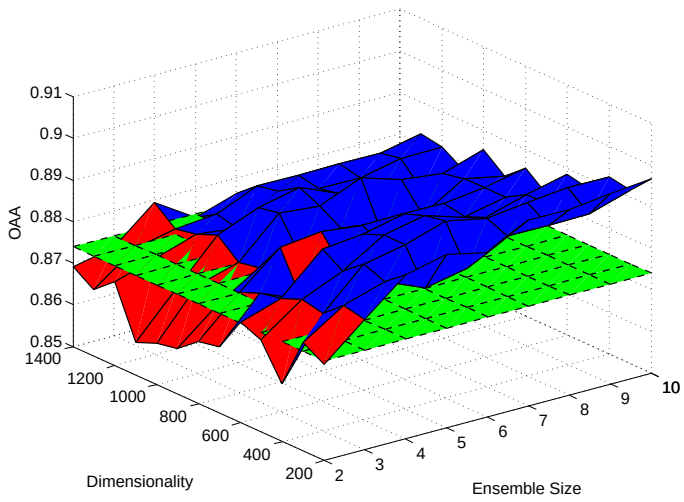


(a) Performance on the plant dataset

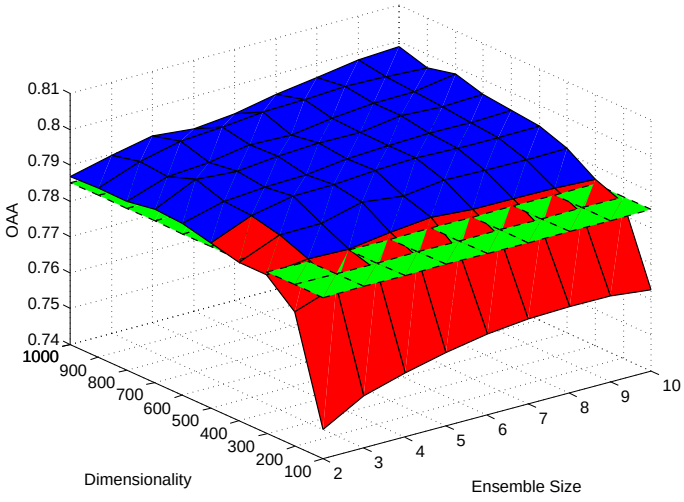


(b) Performance on the eukaryotic dataset

Figure 6: Performance of R3P-Loc at different projected dimensions and ensemble sizes of random projection on (a) the plant dataset and (b) the eukaryotic dataset, respectively. The yellow dotted plane represents the performance using only multi-label ridge regression classifiers (short for RR-Loc), namely the performance without random projection. The mesh with blue surfaces represent the projected dimensions and ensemble sizes at which the R3P-Loc performs better than RR-Loc. The original dimensions of the feature vectors for the plant and eukaryotic datasets are 1541 and 10775, respectively. *Ensemble Size*: Number of times of random projection for ensemble.



(a) Performance on the plant dataset



(b) Performance on the eukaryotic dataset

Figure 7: Performance of R3P-Loc at different projected dimensions and ensemble sizes of random projection on (a) the plant dataset and (b) the eukaryotic dataset, respectively. The green dotted plane represents the accuracy of mGOASVM [52], which is a constant for all projected dimensions and ensemble size. The mesh with blue (red) surfaces represent the projected dimensions and ensemble sizes at which the ensemble R3P-Loc performs better (poorer) than mGOASVM. The original dimensions of the feature vectors for the plant and eukaryotic datasets are 1541 and 10775, respectively. *Ensemble Size*: Number of times of random projection for ensemble.

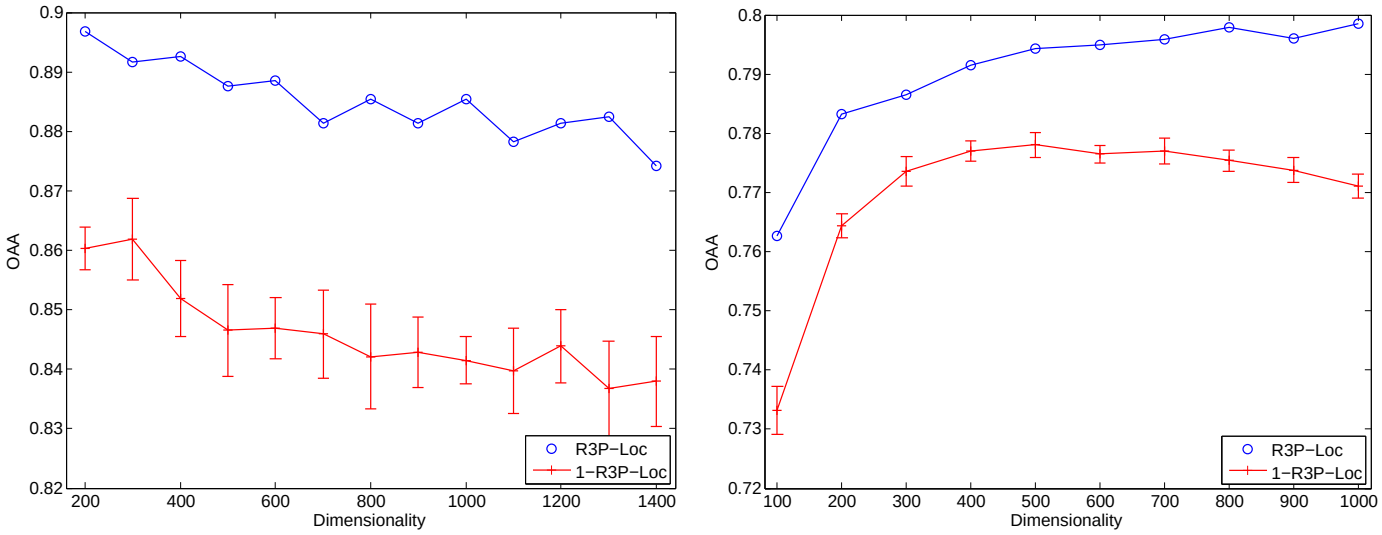
R3P-Loc performs better than mGOASVM in terms of *OAA Accuracy, Precision, F1* and *HL*, while a bit worse on *OLA* and *Recall*. This is probably because the ensemble random projection makes R3P-Loc perform more stringently to control over-predictions than mGOASVM.

According to Eqs. 43–48 and Fig. 4 in a comprehensive review [34], in a system containing both single- and multi-location proteins, the *false positives* (FP) or *over-predictions* and the *false negatives* (FN) or *under-predictions* are defined as:

$$FP = \sum_{i=1}^{N} (|\mathcal{M}(\mathbb{Q}_i)| - |\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|) \tag{19}$$

$$FN = \sum_{i=1}^{N} (|\mathcal{L}(\mathbb{Q}_i)| - |\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|) \tag{20}$$

Table 2 and Table 3 compare the performance of R3P-Loc and mGOASVM in terms of these two metrics. As can be seen, for both datasets, the FP of R3P-Loc is much smaller than that of mGOASVM; on the contrary, the FN of R3P-Loc is larger



(a) Performance on the plant dataset

(b) Performance on the eukaryotic dataset

Figure 8: Performance of R3P-Loc at different feature dimensions on (a) the plant dataset and (b) the eukaryotic dataset, respectively. The original dimensions of the feature vectors for the plant and eukaryotic datasets are 1541 and 10775, respectively. *1-R3P-Loc*: RP-Loc with an ensemble size of 1.

Table 2: Comparing R3P-Loc with state-of-the-art multi-label predictors using the plant dataset. "–" means the corresponding references do not provide the related metrics.

Label	Subcellular Location	Jackknife Test Locative Accuracy (LA)			
		Plant-mPLoc [48]	iLoc-Plant [50]	mGOASVM [52]	R3P-Loc
1	Cell membrane	24/56 = 0.429	39/56 = 0.696	53/56 = 0.946	5/56 = 0.893
2	Cell wall	8/32 = 0.250	19/32 = 0.594	27/32 = 0.844	28/32 = 0.875
3	Chloroplast	248/286 = 0.867	252/286 = 0.881	272/286 = 0.951	279/286 = 0.976
4	Cytoplasm	72/182 = 0.396	114/182 = 0.626	174/182 = 0.956	172/182 = 0.945
5	Endoplasmic reticulum	17/42 = 0.405	21/42 = 0.500	38/42 = 0.905	36/42 = 0.857
6	Extracellular	3/22 = 0.136	2/22 = 0.091	22/22 = 1.000	17/22 = 0.773
7	Golgi apparatus	6/21 = 0.286	16/21 = 0.762	19/21 = 0.905	19/21 = 0.905
8	Mitochondrion	114/150 = 0.760	112/150 = 0.747	150/150 = 1.000	142/150 = 0.947
9	Nucleus	136/152 = 0.895	140/152 = 0.921	151/152 = 0.993	147/152 = 0.967
10	Peroxisome	14/21 = 0.667	6/21 = 0.286	21/21 = 1.000	21/21 = 1.000
11	Plastid	4/39 = 0.103	7/39 = 0.179	39/39 = 1.000	36/39 = 0.923
12	Vacuole	26/52 = 0.500	28/52 = 0.538	49/52 = 0.942	48/52 = 0.923
Overall Actual Accuracy (*OAA*)		–	666/978 = 0.681	855/978 = 0.874	877/978 = **0.897**
Overall Locative Accuracy (*OLA*)		672/1055 = 0.637	756/1055 = 0.717	1015/1055 = **0.962**	995/1055 = 0.943
Accuracy		–	–	0.926	**0.934**
Precision		–	–	0.933	**0.950**
Recall		–	–	**0.968**	0.956
F1		–	–	0.942	**0.947**
HL		–	–	0.013	**0.011**
False Positives (FP)		–	–	113	**71**
False Negatives (FN)		–	–	**40**	60

than that of the latter. The results suggest that R3P-Loc tends to make more prudent predictions than mGOASVM, leading to fewer false positives but more false negatives. When combining with *OAA*, we can see that this prudent strategy enables R3P-Loc to improve the performance in terms of *OAA*.

9. Conclusions

This paper proposes a compact multi-label predictor, namely R3P-Loc, which is based on multi-label ridge regression and random projection to predict subcellular localization of both single- and multi-location proteins. The 'compact' properties are demonstrated in the following two perspectives: (1) two compact databases, namely ProSeq and ProSeq-GO databases, are extracted from Swiss-Prot and GOA databases, respectively for feature extraction; (2) the dimensions of feature vectors are reduced to a compact level by an ensemble random projection method. Specifically, given a query protein, a feature vector is constructed by exploiting the information in the ProSeq-GO database. The GO-vector is projected onto much lower-dimensional space by random matrices whose elements conform to Achlioptas distribution, which are presented to multi-

Table 3: Comparing R3P-Loc with state-of-the-art multi-label predictors using the eukaryotic dataset. "–" means the corresponding references do not provide the related metrics.

Label	Subcellular Location	Jackknife Test Locative Accuracy (LA)			
		Euk-mPLoc 2.0 [49]	iLoc-Euk [51]	mGOASVM [52]	R3P-Loc
1	Acrosome	1/14 = 0.071	1/14 = 0.071	12/14 = 0.857	2/14 = 0.143
2	Cell membrane	452/697 = 0.649	561/697 = 0.805	643/697 = 0.923	525/697 = 0.753
3	Cell wall	6/49 = 0.122	8/49 = 0.163	46/49 = 0.939	45/49 = 0.918
4	Centrosome	22/96 = 0.229	67/96 = 0.698	87/96 = 0.906	65/96 = 0.677
5	Chloroplast	318/385 = 0.826	338/385 = 0.878	375/385 = 0.974	375/385 = 0.974
6	Cyanelle	47/79 = 0.595	51/79 = 0.646	79/79 = 1.000	79/79 = 1.000
7	Cytoplasm	1418/2186 = 0.649	1677/2186 = 0.767	2020/2186 = 0.924	1960/2186 = 0.897
8	Cytoskeleton	44/139 = 0.317	38/139 = 0.273	100/139 = 0.719	53/139 = 0.381
9	Endoplasmic reticulum	348/457 = 0.762	407/457 = 0.891	441/457 = 0.965	426/457 = 0.932
10	Endosome	2/41 = 0.049	3/41 = 0.073	28/41 = 0.683	12/41 = 0.293
11	Extracellular	858/1048 = 0.819	948/1048 = 0.905	1016/1048 = 0.970	969/1048 = 0.925
12	Golgi apparatus	56/254 = 0.221	161/254 = 0.634	231/254 = 0.909	208/254 = 0.819
13	Hydrogenosome	2/10 = 0.200	0/10 = 0.000	10/10 = 1.000	10/10 = 1.000
14	Lysosome	26/57 = 0.456	18/57 = 0.316	52/57 = 0.912	47/57 = 0.825
15	Melanosome	0/47 = 0.000	1/47 = 0.021	44/47 = 0.936	10/47 = 0.213
16	Microsome	1/13 = 0.077	0/13 = 0.000	7/13 = 0.539	1/13 = 0.077
17	Mitochondrion	427/610 = 0.700	470/610 = 0.771	594/610 = 0.974	576/610 = 0.944
18	Nucleus	1501/2320 = 0.647	2040/2320 = 0.879	2194/2320 = 0.946	2157/2320 = 0.930
19	Peroxisome	56/110 = 0.509	60/110 = 0.546	108/110 = 0.982	104/110 = 0.946
20	Spindle pole body	23/68 = 0.338	45/68 = 0.662	65/68 = 0.956	42/68 = 0.618
21	Synapse	0/47 = 0.000	18/47 = 0.383	40/47 = 0.851	26/47 = 0.553
22	Vacuole	101/170 = 0.594	122/170 = 0.718	166/170 = 0.977	156/170 = 0.918
Overall Actual Accuracy (*OAA*)		–	5535/7766 = 0.713	6097/7766 = 0.785	6201/7766 = **0.799**
Overall Locative Accuracy (*OLA*)		5709/8897 = 0.642	7034/8897 = 0.791	8358/8897 = **0.939**	7848/8897 = 0.882
Accuracy		–	–	0.849	**0.859**
Precision		–	–	0.878	**0.882**
Recall		–	–	**0.946**	0.898
F1		–	–	0.878	**0.880**
HL		–	–	0.014	**0.013**
False Positives (FP)		–	–	1702	**1288**
False Negatives (FN)		–	–	**539**	1049

label ridge regression classifiers for classification.

Comparing with existing multi-label predictors, R3P-Loc has the following advantages: (1) it extracts GO feature vectors from two compact databases (ProSeq and ProSeq-GO) which are more efficient and easy-to-use than SwissProt and GOA databases, respectively; (2) it reduces the dimensions of feature vectors as much as seven folds while at the same time impressively improves the classification performance.

Experimental results on two recent benchmark datasets demonstrate that R3P-Loc performs significantly better than existing state-of-the-art multi-label predictors specializing on eukaryotic or plant proteins. It was also found that using the created ProSeq and ProSeq-GO databases achieves equivalent performance as using Swiss-Prot and GOA databases, but with only 3% of the memory consumption. For readers' convenience, the R3P-Loc server is available online at `http://bioinfo.eie.polyu.edu.hk/R3PLocServer/`.

Acknowledgment

This work was in part supported by HKPolyU Grant G-YN18 and G-YL78.

References

[1] K. C. Chou, Y. D. Cai, Predicting protein localization in budding yeast, Bioinformatics 21 (2005) 944–950.

[2] G. Lubec, L. Afjehi-Sadat, J. W. Yang, J. P. John, Searching for hypothetical proteins: Theory and practice based upon original data and literature, Prog. Neurobiol 77 (2005) 90–127.

[3] Y. Chen, C. F. Chen, D. J. Riley, D. C. Allred, P. L. Chen, D. V. Hoff, C. K. Osborne, W. H. Lee, Aberrant Subcellular Localization of BRCA1 in Breast Cancer, Science 270 (1995) 789–791.

[4] M. C. Hung, W. Link, Protein localization in disease and therapy, J. of Cell Sci. 124 (Pt 20) (2011) 3381–3392.

[5] M. D. Kaytor, S. T. Warren, Aberrant Protein Deposition and Neurological Disease, J. Biol. Chem. 274 (1999) 37507–37510.

[6] A. Hayama, T. Rai, S. Sasaki, S. Uchida, Molecular mechanisms of Bartter syndrome caused by mutations in the BSND gene, Histochem. & Cell Biol. 119 (10) (2003) 485–493.

[7] V. Krutovskikh, G. Mazzoleni, N. Mironov, Y. Omori, A. M. Aguelon, M. Mesnil, F. Berger, C. Partensky, H. Yamasaki, Altered homologous and heterologous gap-junctional intercellular communication in primary human liver tumors associated with aberrant protein localization but not gene mutation of connexin 32, Int. J. Cancer 56 (1994) 87–94.

[8] J. B. Campbell, J. Crocker, P. M. Shenoi, S-100 protein localization in minor salivary gland tumours: an aid to diagnosis, J. Laryngol Otol. 102 (10) (1988) 905–908.

[9] X. Lee, J. C. J. Keith, N. Stumm, I. Moutsatsos, J. M. McCoy, C. P. Crum, D. Genest, D. Chin, C. Ehrenfels, R. Pijnenborg, F. A. V. Assche, S. Mi, Downregulation of placental syncytin expression and abnormal protein localization in pre-eclampsia, Placenta 22 (2001) 808–812.

[10] H. Nielsen, J. Engelbrecht, S. Brunak, G. von Heijne, A neural network

method for identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites, Int. J. Neural Sys. 8 (1997) 581–599.

[11] O. Emanuelsson, H. Nielsen, S. Brunak, G. von Heijne, Predicting subcellular localization of proteins based on their N-terminal amino acid sequence, J. Mol. Biol. 300 (4) (2000) 1005–1016.

[12] K. Nakai, M. Kanehisa, Expert system for predicting protein localization sites in gram-negative bacteria, Proteins: Structure, Function, and Genetics 11 (2) (1991) 95–110.

[13] K.-C. Chou, H.-B. Shen, Signal-CF: a subsite-coupled and window-fusing approach for predicting signal peptides, Biochemical and Biophysical Research Communications 357 (3) (2007) 633–640.

[14] H.-B. Shen, K.-C. Chou, Signal-3L: A 3-layer approach for predicting signal peptides, Biochemical and Biophysical Research Communications 363 (2) (2007) 297–303.

[15] K. C. Chou, Prediction of protein cellular attributes using pseudo amino acid composition, Proteins: Structure, Function, and Genetics 43 (2001) 246–255.

[16] H. Nakashima, K. Nishikawa, Discrimination of intracellular and extracellular proteins using amino acid composition and residue-pair frequencies, J. Mol. Biol. 238 (1994) 54–61.

[17] K. C. Chou, D. W. Elord, Using discriminant function for prediction of subcellular location of prokaryotic proteins, Biochem. Biophys. Res. Commun. 252 (1998) 63–68.

[18] K. C. Chou, D. W. Elord, Protein subcellular location prediction, Protein Eng. 12 (1999) 107–118.

[19] G. P. Zhou, K. Doctor, Subcellular location prediction of apoptosis proteins, PROTEINS: Structure, Function, and Genetics 50 (2003) 44–48.

[20] G.-L. Fan, Q.-Z. Li, Predict mycobacterial proteins subcellular locations by incorporating pseudo-average chemical shift into the general form of Chou's pseudo amino acid composition, Journal of Theoretical Biology 304 (2012) 88–95.

[21] M. W. Mak, J. Guo, S. Y. Kung, PairProSVM: Protein subcellular localization based on local pairwise profile alignment and SVM, IEEE/ACM Trans. on Computational Biology and Bioinformatics 5 (3) (2008) 416 – 422.

[22] Z. Lu, D. Szafron, R. Greiner, P. Lu, D. S. Wishart, B. Poulin, J. Anvik, C. Macdonell, R. Eisner, Predicting subcellular localization of proteins using machine-learned classifiers, Bioinformatics 20 (4) (2004) 547–556.

[23] R. Mott, J. Schultz, P. Bork, C. Ponting, Predicting protein cellular localization using a domain projection method, Genome research 12 (8) (2002) 1168–1174.

[24] S. Wan, M. W. Mak, S. Y. Kung, Protein subcellular localization prediction based on profile alignment and Gene Ontology, in: 2011 IEEE International Workshop on Machine Learning for Signal Processing (MLSP'11), 2011, pp. 1–6.

[25] K. C. Chou, Y. D. Cai, Prediction of protein subcellular locations by GO-FunD-PseAA predictor, Biochem. Biophys. Res. Commun. 320 (2004) 1236–1239.

[26] S. Wan, M. W. Mak, S. Y. Kung, Adaptive thresholding for multi-label SVM classification with application to protein subcellular localization prediction, in: 2013 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'13), 2013, pp. 3547–3551.

[27] K. C. Chou, H. B. Shen, Predicting eukaryotic protein subcellular location by fusing optimized evidence-theoretic K-nearest neighbor classifiers, J. of Proteome Research 5 (2006) 1888–1897.

[28] S. Wan, M. W. Mak, S. Y. Kung, GOASVM: Protein subcellular localization prediction based on gene ontology annotation and SVM, in: 2012 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'12), 2012, pp. 2229–2232.

[29] S. Mei, Multi-label multi-kernel transfer learning for human protein subcellular localization, PLoS ONE 7 (6) (2012) e37716.

[30] S. Wan, M. W. Mak, S. Y. Kung, Semantic similarity over gene ontology for multi-label protein subcellular localization, Engineering 5 (2013) 68–72.

[31] W.-Z. Lin, J.-A. Fang, X. Xiao, K.-C. Chou, iLoc-Animal: a multi-label learning classifier for predicting subcellular localization of animal proteins, Molecular BioSystems 9 (4) (2013) 634–644.

[32] K. C. Chou, Z. C. Wu, X. Xiao, iLoc-Hum: using the accumulation-label scale to predict subcellular locations of human proteins with both single and multiple sites, Molecular BioSystems 8 (2012) 629–641.

[33] S. Mei, Predicting plant protein subcellular multi-localization by Chou's PseAAC formulation based multi-label homolog knowledge transfer learning, Journal of Theoretical Biology 310 (2012) 80–87.

[34] K. C. Chou, H. B. Shen, Recent progress in protein subcellular location prediction, Analytical Biochemistry 1 (370) (2007) 1–16.

[35] R. Nair, B. Rost, Sequence conserved for subcellular localization, Protein Science 11 (2002) 2836–2847.

[36] Z. Lu, D. Szafron, R. Greiner, P. Lu, D. S. Wishart, B. Poulin, J. Anvik, C. Macdonell, R. Eisner, Predicting subcellular localization of proteins using machine-learned classifiers, Bioinformatics 20 (4) (2004) 547–556.

[37] K. C. Chou, Y. D. Cai, Using functional domain composition and support vector machines for prediction of protein subcellular location, J. of Biol. Chem. 277 (2002) 45765–45769.

[38] S. Brady, H. Shatkay, EpiLoc: a (working) text-based system for predicting protein subcellular location, in: Pac. Symp. Biocomput., 2008, pp. 604–615.

[39] A. Fyshe, Y. Liu, D. Szafron, R. Greiner, P. Lu, Improving subcellular localization prediction using text classification and the gene ontology, Bioinformatics 24 (2008) 2512–2517.

[40] W. L. Huang, C. W. Tung, S. W. Ho, S. F. Hwang, S. Y. Ho, ProLoc-GO: Utilizing informative Gene Ontology terms for sequence-based prediction of protein subcellular localization, BMC Bioinformatics 9 (2008) 80.

[41] S.-M. Chi, D. Nam, Wegoloc: accurate prediction of protein subcellular localization using weighted gene ontology terms, Bioinformatics 28 (7) (2012) 1028–1030.
URL `http://bioinformatics.oxfordjournals.org/content/28/7/1028.short`

[42] S. Wan, M. W. Mak, S. Y. Kung, GOASVM: A subcellular location predictor by incorporating term-frequency gene ontology into the general form of Chou's pseudo-amino acid composition, Journal of Theoretical Biology 323 (2013) 40–48.

[43] A. H. Millar, C. Carrie, B. Pogson, J. Whelan, Exploring the function-location nexus: using multiple lines of evidence in defining the subcellular location of plant proteins, Plant Cell 21 (6) (2009) 1625–1631.

[44] R. F. Murphy, communicating subcellular distributions, Cytometry 77 (7) (2010) 686–92.

[45] L. J. Foster, C. L. D. Hoog, Y. Zhang, Y. Zhang, X. Xie, V. K. Mootha, M. Mann, A mammalian organelle map by protein correlation profiling, Cell 125 (2006) 187–199.

[46] S. Zhang, X. F. Xia, J. C. Shen, Y. Zhou, Z. Sun, DBMLoc: A database of proteins with multiple subcellular localizations, BMC Bioinformatics 9 (2008) 127.

[47] J. C. Mueller, C. Andreoli, H. Prokisch, T. Meitinger, Mechanisms for multiple intracellular localization of human mitochondrial proteins, Mitochondrion 3 (2004) 315–325.

[48] K. C. Chou, H. B. Shen, Plant-mPLoc: A top-down strategy to augment the power for predicting plant protein subcellular localization, PLoS ONE 5 (2010) e11335.

[49] K. C. Chou, H. B. Shen, A new method for predicting the subcellular localization of eukaryotic proteins with both single and multiple site: Euk-mPLoc 2.0, PLoS ONE 5 (2010) e9931.

[50] Z. C. Wu, X. Xiao, K. C. Chou, iLoc-Plant: A multi-label classifier for predicting the subcellular localization of plant proteins with both single and multiple sites, Molecular BioSystems 7 (2011) 3287–3297.

[51] K. C. Chou, Z. C. Wu, X. Xiao, iLoc-Euk: A multi-label classifier for predicting the subcellular localization of singleplex and multiplex eukaryotic proteins, PLoS ONE 6 (3) (2011) e18258.

[52] S. Wan, M. W. Mak, S. Y. Kung, mGOASVM: Multi-label protein subcellular localization based on gene ontology and support vector machines, BMC Bioinformatics 13 (2012) 290.

[53] S. Wan, M. W. Mak, S. Y. Kung, HybridGO-Loc: Mining hybrid features on gene ontology for predicting subcellular localization of multi-location proteins, PLoS ONE 9 (3) (2014) e89545.

[54] J. He, H. Gu, W. Liu, Imbalanced multi-modal multi-label learning for subcellular localization prediction of human proteins with both single and multiple sites, PLoS ONE 7 (6) (2011) e37155.

[55] S. Wan, M. W. Mak, B. Zhang, Y. Wang, S. Y. Kung, An ensemble classifier with random projection for predicting multi-label protein subcellular localization, in: 2013 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2013, pp. 35–42. doi:10.1109/BIBM.2013.6732715.

[56] L. Q. Li, Y. Zhang, L. Y. Zou, C. Q. Li, B. Yu, X. Q. Zheng, Y. Zhou, An ensemble classifier for eukaryotic protein subcellular location prediction using Gene Ontology categories and amino acid hydrophobicity, PLoS ONE 7 (1) (2012) e31057.

[57] X. Xiao, Z. C. Wu, K. C. Chou, iLoc-Virus: A multi-label learning classifier for identifying the subcellular localization of virus proteins with both single and multiple sites, Journal of Theoretical Biology 284 (2011) 42–51.

[58] K. C. Chou, Some remarks on protein attribute prediction and pseudo amino acid composition (50th Anniversary Year Review), Journal of Theoretical Biology 273 (2011) 236–247.

[59] Y. Xu, J. Ding, L.-Y. Wu, K.-C. Chou, iSNO-PseAAC: Predict cysteine S-nitrosylation sites in proteins by incorporating position specific amino acid propensity into pseudo amino acid composition, PLoS ONE 8 (2) (2013) e55844.

[60] W. Chen, P.-M. Feng, H. Lin, K.-C. Chou, iRSpot-PseDNC: Identify recombination spots with pseudo dinucleotide composition, Nucleic Acids Research 41 (6) (2013) e68–e68.

[61] J.-L. Min, X. Xiao, K.-C. Chou, iEzy-Drug: A web server for identifying the interaction between enzymes and drugs in cellular networking, BioMed Research International 2013.

[62] Y. Xu, X.-J. Shao, L.-Y. Wu, N.-Y. Deng, K.-C. Chou, iSNO-AAPair: incorporating amino acid pairwise coupling into PseAAC for predicting cysteine S-nitrosylation sites in proteins, PeerJ 1 (2013) e171.

[63] X. Xiao, J.-L. Min, P. Wang, K.-C. Chou, iCDI-PseFpt: Identify the channel–drug interaction in cellular networking with PseAAC and molecular fingerprints, Journal of Theoretical Biology 337 (2013) 71–79.

[64] Y.-N. Fan, X. Xiao, J.-L. Min, K.-C. Chou, iNR-Drug: Predicting the interaction of drugs with nuclear receptors in cellular networking, International Journal of Molecular Sciences 15 (3) (2014) 4915–4937.

[65] S.-H. Guo, E.-Z. Deng, L.-Q. Xu, H. Ding, H. Lin, W. Chen, K.-C. Chou, iNuc-PseKNC: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition, Bioinformatics (2014) btu083.

[66] B. Liu, D. Zhang, R. Xu, J. Xu, X. Wang, Q. Chen, Q. Dong, K.-C. Chou, Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection, Bioinformatics (2014) 472–479.

[67] W.-R. Qiu, X. Xiao, K.-C. Chou, iRSpot-TNCPseAAC: Identify Recombination Spots with Trinucleotide Composition and Pseudo Amino Acid Components, International Journal of Molecular Sciences 15 (2) (2014) 1746–1766.

[68] W.-R. Qiu, X. Xiao, W.-Z. Lin, K.-C. Chou, iMethyl-PseAAC: Identification of protein methylation sites via a pseudo amino acid composition approach, BioMed Research International 2014.

[69] H. Ding, E.-Z. Deng, L.-F. Yuan, L. Liu, H. Lin, W. Chen, K.-C. Chou, iCTX-Type: A Sequence-Based Predictor for Identifying the Types of Conotoxins in Targeting Ion Channels, BioMed Research International 2014.

[70] W. Chen, P.-M. Feng, H. Lin, K.-C. Chou, iSS-PseDNC: Identifying splicing sites using pseudo dinucleotide composition, BioMed Research International 2014.

[71] Y. Xu, X. Wen, X.-J. Shao, N.-Y. Deng, K.-C. Chou, iHyd-PseAAC: Predicting hydroxyproline and hydroxylysine in proteins by incorporating dipeptide position-specific propensity into pseudo amino acid composition, International Journal of Molecular Sciences 15 (5) (2014) 7594–7610.

[72] Z. Lu, L. Hunter, GO molecular function terms are predictive of subcellular localization, in: In Proc. of Pac. Symp. Biocomput. (PSB'05), 2005, pp. 151–161.

[73] S. Briesemeister, T. Blum, S. Brady, Y. Lam, O. Kohlbacher, H. Shatkay, SherLoc2: A high-accuracy hybrid method for predicting subcellular localization of proteins, Journal of Proteome Research 8 (2009) 5363–5366.

[74] K. C. Chou, Some remarks on predicting multi-label attributes in molecular biosystems, Molecular BioSystems 9 (2013) 1092–1100.

[75] X. Wang, G. Z. Li, A multi-label predictor for identifying the subcellular locations of singleplex and multiplex eukaryotic proteins, PLoS ONE 7 (5) (2012) e36317.

[76] K. C. Chou, H. B. Shen, Cell-PLoc: A package of web-servers for predicting subcellular localization of proteins in various organisms, Nature Protocols 3 (2008) 153–162.

[77] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, D. J. Lipman, Gapped BLAST and PSI-BLAST: A new generation of protein database search programs, Nucleic Acids Res. 25 (1997) 3389–3402.

[78] K. Nakai, Protein sorting signals and prediction of subcellular localization, Advances in Protein Chemistry 54 (1) (2000) 277–344.

[79] W. B. Johnson, J. Lindenstrauss, Extensions of Lipschitz mappings into a Hilbert space, in: Conference in Modern Analysis and Probability, 1984, pp. 599–608.

[80] P. Frankl, H. Maehara., The Johnson-Lindenstrauss lemma and the sphericity of some graphs, Journal of Combinatorial Theory, Series B 44 (1988) 355–362.

[81] E. Bingham, H. Mannila, Random projection in dimension reduction: Applications to image and text data, in: the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'01), 2001, pp. 245–250.

[82] D. Achlioptas, Database-friendly random projections: Johnson-Lindenstrauss with binary coins, Journal of Computer and Systems Sciences 66 (2003) 671–687.

[83] E. J. Candes, T. Tao, Near-optimal signal recovery from random projections: Universal encoding strategies?, IEEE Transactions on Information Theory 52 (12) (2006) 5406–5425.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4016283

[84] G. R. Pasha, M. A. A. Shah, Application of ridge regression to multicollinear data, Journal of Research (Science) 15 (1) (2004) 97–106.

[85] A. Hadgu, An application of ridge regression analysis in the study of syphilis data, Statistics in Medicine 3 (3) (1984) 293–299.

[86] D. W. Marquardt, R. D. Snee, Ridge regression in practice, The American Statistician 29 (1) (1975) 3–20.

[87] K. C. Chou, H. B. Shen, Review: recent advances in developing webservers for predicting protein attributes, Natural Science 2 (2009) 63–92.

[88] W. Chen, T.-Y. Lei, D.-C. Jin, H. Lin, K.-C. Chou, PseKNC: A flexible web server for generating pseudo K-tuple nucleotide composition, Analytical Biochemistry 456 (2014) 53–60.

[89] K. Dembczynski, W. Waegeman, W. Cheng, E. Hullermeier, On label dependence and loss minimization in multi-label classification, Machine Learning 88 (1-2) (2012) 5–45.

[90] W. Gao, Z. H. Zhou, On the consistency of multi-label learning, in: Proceedings of the 24th Annual Conference on Learning Theory, 2011, pp. 341–358.

[91] K. C. Chou, C. T. Zhang, Review: Prediction of protein structural classes, Critical Reviews in Biochemistry and Molecular Biology 30 (4) (1995) 275–349.

[92] H. Mohabatkar, Prediction of cyclin proteins using Chous pseudo amino acid composition, Protein and Peptide Letters 17 (10) (2010) 1207–1214.

[93] S. S. Sahu, G. Panda, A novel feature representation method based on Chou's pseudo amino acid composition for protein structural class prediction, Computational Biology and Chemistry 34 (5) (2010) 320–327.

[94] M. Esmaeili, H. Mohabatkar, S. Mohsenzadeh, Using the concept of Chou's pseudo amino acid composition for risk type prediction of human papillomaviruses, Journal of Theoretical Biology 263 (2) (2010) 203–209.

[95] M. Khosravian, F. Kazemi Faramarzi, M. Mohammad Beigi, M. Behbahani, H. Mohabatkar, Predicting antibacterial peptides by the concept of Chou's pseudo-amino acid composition and machine learning methods, Protein and Peptide Letters 20 (2) (2013) 180–186.

[96] H. Mohabatkar, M. Mohammad Beigi, K. Abdolahi, S. Mohsenzadeh, Prediction of allergenic proteins by means of the concept of Chou's pseudo amino acid composition and a machine learning approach, Medicinal Chemistry 9 (1) (2013) 133–137.