On the Encoding of Proteins for Disordered Regions Prediction

Julien Becker[1], Francis Maes[2,3], Louis Wehenkel[2]*

[1] Bioinformatics and Modeling, GIGA-Research, University of Liege, Liege, Belgium, [2] Department of Electrical Engineering and Computer Science, Montefiore Institute, University of Liege, Liege, Belgium, [3] Declaratieve Talen en Artificiele Intelligentie, Departement Computerwetenschappen, University of Leuven, Leuven, Belgium

Abstract

Disordered regions, *i.e.*, regions of proteins that do not adopt a stable three-dimensional structure, have been shown to play various and critical roles in many biological processes. Predicting and understanding their formation is therefore a key sub-problem of protein structure and function inference. A wide range of machine learning approaches have been developed to automatically predict disordered regions of proteins. One key factor of the success of these methods is the way in which protein information is encoded into features. Recently, we have proposed a systematic methodology to study the relevance of various feature encodings in the context of disulfide connectivity pattern prediction. In the present paper, we adapt this methodology to the problem of predicting disordered regions and assess it on proteins from the 10th CASP competition, as well as on a very large subset of proteins extracted from PDB. Our results, obtained with ensembles of extremely randomized trees, highlight a novel feature function encoding the proximity of residues according to their accessibility to the solvent, which is playing the second most important role in the prediction of disordered regions, just after evolutionary information. Furthermore, even though our approach treats each residue independently, our results are very competitive in terms of accuracy with respect to the state-of-the-art. A web-application is available at http://m24.giga.ulg.ac.be:81/x3Disorder.

Citation: Becker J, Maes F, Wehenkel L (2013) On the Encoding of Proteins for Disordered Regions Prediction. PLoS ONE 8(12): e82252. doi:10.1371/journal.pone.0082252

Editor: Silvio C. E. Tosatto, Universita' di Padova, Italy

Received July 11, 2013; **Accepted** October 21, 2013; **Published** December 16, 2013

Funding: This work is supported by a FRIA (Fund for Research Training in Industry and Agriculture) fellowship (Julien Becker) granted by the Belgian National Fund of Scientific Research (FNRS). 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: L.Wehenkel@ulg.ac.be

Introduction

Disordered regions refer to regions in proteins that do not adopt a stable three-dimensional structure when they are not in presence of their partner molecules. Over the last decade, several experimental studies have shown that proteins with disordered regions play various and critical functions in many biological processes. The flexibility of these regions makes it possible for a protein to interact, recognize and bind to many partners. For example, disordered regions are often involved in regulatory and signaling interactions [2] such as the regulation of cell division, the transcription of DNA or the translation of ARNm. They also play a role in the self-assembly of protein complexes, and in the storage of small molecules [3,4].

Several automatic methodologies have been proposed to predict disordered regions from primary sequences. They range from simple methods based on the sequence complexity [5] to more sophisticated machine learning approaches often relying on neural networks or Support Vector Machines (SVMs)[6–10]. For example, the Poodle tool is based on three adjacent classifiers, which are specialized in making short [11] or long [12] disordered regions predictions, or unfolded protein predictions [13], while the Spritz tool [14] uses two specialized SVMs for either short or long disordered regions. Recently, meta-predictors have also appeared in the literature. These approaches consist in combining predictions of a large number of existing disordered regions predictors [15,16], *e.g.*, GSmetaDisorder gathers no less than 12 different predictors. Nowadays, there exist more than 50 disordered region predictors. Fortunately, since 2004, a part of the biannual competition "*Critical Assessment of Techniques for Protein Structure Prediction*" (CASP) is devoted to the comparison of the participant disordered regions predictors. For more information about disordered regions predictors, one can refer to the reports of these assessments [17] or to the recent comprehensive overview of computational protein disorder prediction methods made by Deng *et al.* [18].

In machine learning, the way to encode information into vectors of features typically has a major impact on the classification accuracy. In the context of bioinformactics, and specifically in the case of protein structure inference, candidate features are typically grouped into parameterized families of features (we use the term 'feature function' to denote such a family), where each family provides a different kind of physical or biological information. Recently, we have developed a systematic feature function selection methodology [1] for the inference of disulfide bridges within protein structures, and which allowed us to identify a minimal subset of relevant feature functions for this problem.

The main contribution of the present paper is the adaptation of the selection pipeline presented in our previous work [1] to establish a relevant representation of residues in the context of disordered regions prediction. For this purpose, we consider various feature encodings and, in addition to the primary structure, three in-sillico annotations: position-specific scoring matrices (PSSM), predicted secondary structures and predicted solvent accessibilities. We apply the feature function selection pipeline in combination with Extremely randomized Trees (ETs), a model which gave excellent results in previous work [1]. In order to avoid any risk of overfitting or over-estimation of our models, we use three distinct datasets: Disorder723 [19], Casp10 (http://www.predictioncenter.org/casp10/) and Pdb30. We first apply feature selection on Disorder723 and then assess the relevance of the selected feature functions both on Casp10 and on Pdb30.

The main result of our study is to highlight a novel feature function encoding the proximity along the primary sequence of residues predicted as being accessible (resp. inaccessible) to the solvent. This feature function is identified as the second most important for predicting the belonging of a residue to a disordered region, just after evolutionary information derived from the PSSM. To our best knowledge, these features encoding solvent accessibility have never been highlighted in previous studies of disordered regions prediction. The majority of the remaining relevant feature functions that we found (e.g., evolutionary information and sequence complexity) were already suggested by other studies of disordered regions [5], and we thus confirm in a fair way their relevance. Furthermore, even though our approach treats each residue independently, *i.e.*, without explicitly modelling global properties of disordered regions, our predictors are very competitive in terms of accuracy with respect to Casp10 assessments and to our very large independent test set extracted from Pdb30.

Materials and Methods

There exist a huge number of manners to encode proteins into an appropriate form for machine learning algorithms, *i.e.*, vectors of (categorical or numerical) features. In this study, we consider a number of *feature functions*, which aim at encoding a particular property of the protein into a vector of features of fixed length. For example, the enumeration of the 11 amino acids at the flanks of a residue of interest is a feature function that, given a residue position within a protein, returns a vector of 11 categorical features. To form more sophisticated representations, feature functions can be combined through the concatenation of their encoding vectors.

Among the large number of possible combinations, our study aims at identifying the minimal feature function set that is relevant for disordered regions prediction. In [1], this identification is performed through a forward feature function selection algorithm for the problem of disulfide bridge prediction. In order to work, this algorithm requires four components to be specified: a dataset, a list of candidate feature functions, a base learner and a criterion to optimize.

This section describes how we have adapted each of these four components to the problem of predicting disordered regions of proteins. The first part presents the three datasets (Disorder723, Casp10 and Pdb30) and how we enrich the primary structures of each of these datasets with three annotations: position-specific scoring matrices (*PSSM*), predicted secondary structures (*SS*) and predicted solvent accessibilities (*SA*). The second part of this section formulates disordered regions prediction as a supervised-learning problem and, more specifically, as a binary classification problem, which aims at predicting the disorder state (ordered or disordered) of each protein residue. It also defines five measures to assess the quality of the predictions. The third part briefly describes the forward feature functions selection methodology and enumerates the candidate feature functions that we consider during the selection process. Finally, the last part of this section introduces ensembles of extremely randomized trees, which are used as the base learner within the feature function selection algorithm.

Datasets and annotations

This study relies on three datasets. The first one, Disorder723 (http://casp.rnet.missouri.edu/download/disorder.dataset), has been built by Cheng *et al.* [19] and was extracted from the Protein Data Bank [20] in May 2004. The dataset is made of 723 non-redundant chains that contain at least 30 amino acids in length and that were solved by X-ray diffraction with a resolution of around 2.5 Å. In order to reduce the over-representation of particular protein families, the dataset has been filtered by UniqueProt [21], a protein redundancy reduction tool based on the HSSP distance [22], with a cut-off distance of 10.

The second dataset, Casp10, is the one used during the 10th CASP competitions that took place in 2012. During the competition, the candidate predictors have to make blind predictions, *i.e*, they have to predict disordered regions of proteins close to being solved or close to being published and that have no detectable similarity to available structures. At the end, the candidate predictors were assessed on 94 experimentally determined proteins available for download on the official CASP website(http://predictioncenter.org/download_area/CASP10/targets/casp10.DR_targets.tgz). Note that unlike Disorder723, the way to resolve protein structures is not restricted to X-ray diffraction and that CASP10 also contains protein structures determined by NMR.

The last dataset, that we denote by Pdb30, is far larger than the two previous ones. We created Pdb30 on one of the clustered versions of the Protein Data Bank (as of August 31, 2013) available at http://www.rcsb.org/pdb/statistics/clusterStatistics.do. The clustering is defined on a protein chain basis with a maximum pairwise sequence identity of 30%. The authors of this clustered version of PDB used BLASTClust [23] to perform the clustering and selected the representative structure of each cluster according to their quality factor. We then filtered out any proteins that were less than 30 amino acids in length, that had no X-ray structure or that had resolution coarser than 2.5 Å. Next, we discarded the proteins that share a sequence identity of at least 30% with a protein of Disorder723 (our training set). The final dataset is made of 12,090 proteins and 2,991,008 residues of which 193,874 (6.5%) are disordered. Figure 1 shows a histogram of the protein lengths. The average (± standard deviation) protein length is 247.4 ± 162.8. Figure 2 shows a histogram of the disordered region lengths of our dataset. The average disordered region length is 12.3 ± 15.6. The dataset is available at: http://m24.giga.ulg.ac.be:81/x3Disorder/pdb30.dataset.

In our experiment, we use Disorder723 to identify a subset of relevant feature functions while Casp10 and Pdb30 are used to assess the quality of the selected feature functions. It is important to note that no protein in the Casp10 or Pdb30 sets share more than 30% sequence identity with one of those of Disorder723. This therefore makes it possible to fairly evaluate and compare our results with those that have participated to the 10th CASP competition.

We use the same definition of disorder as Cheng *et al.* and as the CASP competition, *i.e.*, segments longer than three residues but lacking atomic coordinates in the crystal structure are labelled as "disordered" whereas all other residues are labelled as "ordered". According to this definition, Table 1 shows that the three datasets contain ~ 6% of disordered residues and ~ 94% of ordered residues. Some residues in Casp10 were not classified by the CASP

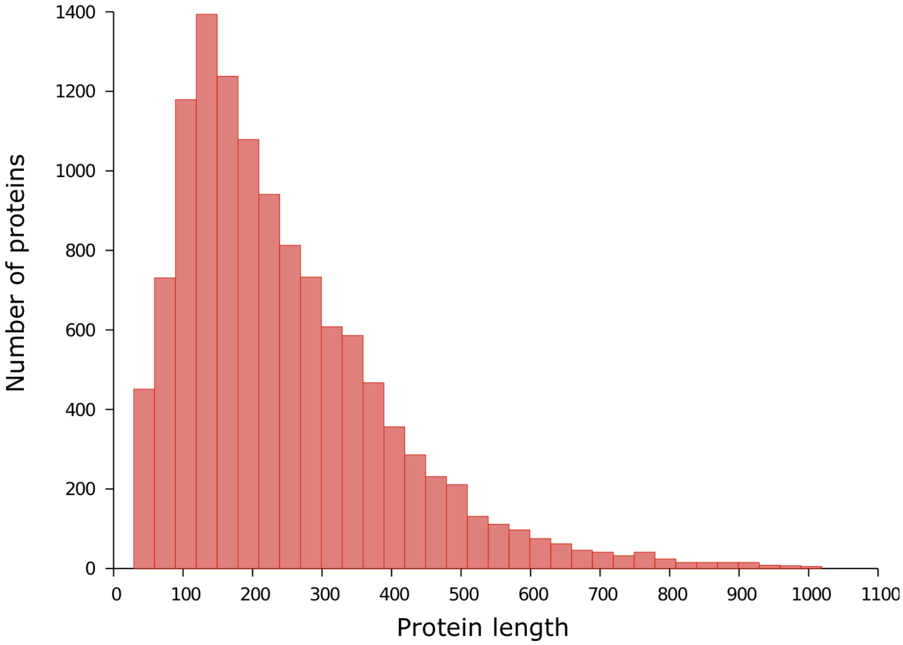

Figure 1. Protein length distribution of PDB30. There are 12,090 proteins. The average protein length is of 247.4 residues.
doi:10.1371/journal.pone.0082252.g001

assessors. These residues were not taken into account in our experiments.

We enrich the primary structure (denoted as AA) by using three additional annotations: evolutionary information in the form of a position-specific scoring matrix ($PSSM$), predicted secondary structure (SS) and predicted solvent accessibility (SA). We computed the PSSMs by running three iterations of the PSI-BLAST program [24] on the non-redundant NCBI database [25]. To produce predicted annotations, we used the SSpro and ACCpro [3] programs for the predicted secondary structure (“helix”, “strand” or “coil”) and the predicted solvent accessibility (under or over 25% exposed), respectively.

Problem statement

Let $\mathcal{P}$ be the space of all proteins and $P=(AA,PSSM,SS,SA,Y)\in\mathcal{P}$ one particular protein described as the 5-tuple

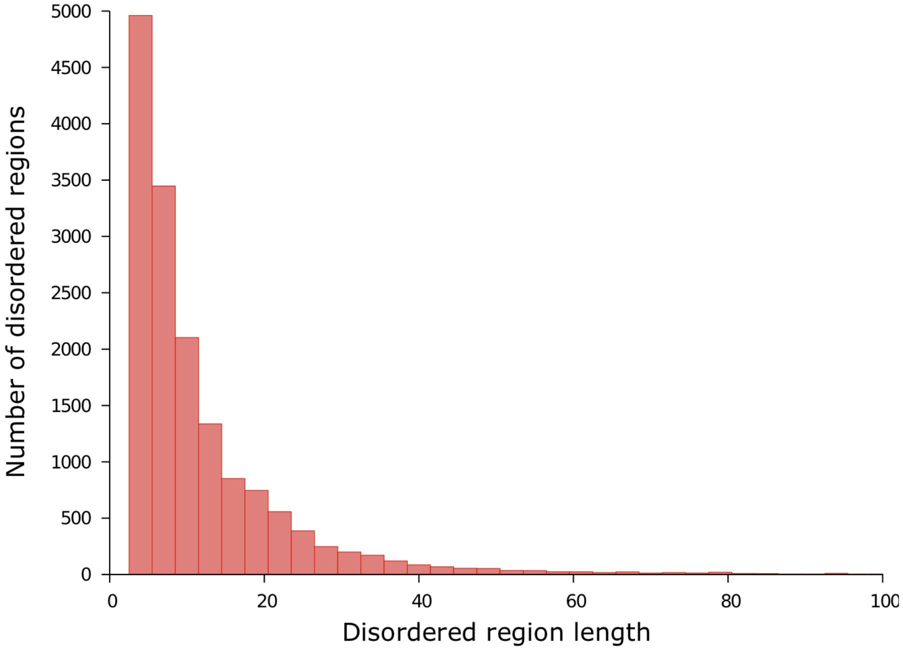

Figure 2. Disordered region length distribution of PDB30. There are 15,726 disordered regions. The average length of a disordered region is of 12.3 residues and the average number of disordered regions per protein is of 1.3.
doi:10.1371/journal.pone.0082252.g002

Table 1. Composition of datasets.

	Proteins	Ordered residues	Disordered residues	Residues
Disorder723	723	201,703 (93.55%)	13,909 (6.45%)	215,612
Casp10	94	22,688 (93.79%)	1502 (6.20%)	24,190
Pdb30	12,090	2,797,134 (93.52%)	193,874 (6.48%)	2,991,008

Number of proteins, number (and portion) of ordered/disordered residues and number of residues in Disorder723, Casp10 and Pdb30 datasets. All datasets have roughly the same proportion of disordered residues ($\sim$ 6%). Pdb30 contains $\sim$127 times more proteins and $\sim$124 times more residues than Casp10.
doi:10.1371/journal.pone.0082252.t001

containing its primary structure AA, its $PSSM$, its two predicted annotations SS and SA, and its disordered regions Y. Each of these annotations is described as a sequence of n labels, where n is the number of residues composing P. For example, the primary structure is defined as $AA=(AA_1,AA_2,\dots,AA_n)$, where AA_i is the label corresponding to the amino acid of the i-th residue of P, and the disordered regions annotation is defined as $Y=(y_1,y_2,\dots,y_n)$, where $y_i\in\{\text{ordered},\text{disordered}\}$. The disordered regions prediction task consists in assigning a label y_i to each residue of P.

In the supervised-learning formulation of the problem, we assume to have access to a dataset of proteins in which residues are labeled either ordered or disordered. We denote this dataset $D=\{P^{(i)}\}_{i\in[1,N]}$, where $P^{(i)}\in\mathcal{P}$ is the i-th protein. Given such a dataset D, the aim is to learn a disordered regions predictor $f:\mathcal{P}\backslash\mathcal{Y}\rightarrow\mathcal{Y}$ that maps a protein $P\in\mathcal{P}$ to a sequence $\hat{Y}$ of n predicted labels $\hat{y}_i\in\{\text{ordered},\text{disordered}\}$, where n is the length of P.

It is important to note that disordered regions are segments, *i.e.* consecutive residues tend to share the same label. More and more machine learning approaches such as conditional random fields [27], recursive neural networks [19], meta-predictors [28] or post-filtering steps [29] are able to exploit the structured aspect of the problem.

However, as the goal of this study is to determine a set of relevant feature functions in general, we do not focus on such advanced prediction approaches here. We instead simplify the general problem into a standard binary classification problem. The aim is to learn a predictor $f:(\mathcal{P}\backslash\mathcal{Y})\times\mathbb{N}\rightarrow\{\text{ordered}, \textit{disordered}\}$ that maps the i-th residue of a protein P to the predicted label y_i. This formulation is rather simple in the sense that it treats each residue independently, *i.e.*, regardless with respect to predictions made on neighboring residues of the same protein.

Evaluation measures. In order to evaluate the quality of the predictions made by our models, we consider five residue-level performance measures: the balanced accuracy (Acc), the sensitivity, the specificity, the area under the ROC curve (AUC) and the F-measure. Each of these measures can be formulated using a tuple of four values: the number of *true positives* (TP), *false positives* (FP), *true negatives* (TN), and *false negatives* (FN), where a positive example is a disordered residue and a negative example is an ordered residue. Therefore, a true positive is a correctly predicted disordered residue and a false negative is an ordered residues falsely predicted as a disordered one.

According to these notations, the sensitivity $[TP\div(TP+FN)]$ is the fraction of disordered residues that are successfully predicted as disordered, whereas the specificity $[TN\div(TN+FP)]$ is the fraction of ordered residues that are successfully predicted as ordered. As the problem of disordered regions prediction is strongly imbalanced (only $\sim$ 6% of residues are disordered), using the conventional accuracy may inflate performance estimate and is therefore not appropriate. However, the balanced accuracy, defined as the arithmetic mean of sensitivity and specificity, is robust against imbalanced datasets as well as the F-measure, which is used in recent CASP assessments. The F-measure is defined as the harmonic mean of the precision – the fraction of predicted disordered residues that are truly disordered – and the sensitivity (also called recall).

Since, a large number of available binary classifiers produce probabilities rather than strict classes, these criteria rely on a user-defined *decision threshold* to discriminate positive from negative examples. Depending on how users fixed their threshold, a bias might be introduced, which might lead to an unfair comparison between distinct studies. To tackle this issue, one can compare the performance of distinct models by their ROC curve, which is obtained by plotting the sensitivity against the false positive rate $[FP\div(FP+TN)]$ when varying the decision threshold. However, the comparison is not easy, especially when the curves are similar. A common simplification is therefore to calculate the area under the ROC curve (AUC). An area of 1.00 corresponds to a perfect predictor while an area of 0.50 corresponds to a random predictor.

Forward feature function selection

Recently, we have developed a tractable and interpretable feature function selection methodology [1], which aims at identifying a minimal set of relevant feature functions among a larger group of candidate feature functions. Note that this approach focuses on identifying feature functions rather than individual features. Figure 3 roughly depicts this algorithm. It is a *wrapper* approach that repeatedly evaluates subsets of feature functions through an objective function S, which typically cross-validates the base learner $\mathcal{B}$ on a dataset D, and that is directly driven by the scores returned by S. To obtain interpretable results, the method relies on a rather simple scheme, which consists in constructing the feature function set greedily in a forward way: starting from an empty set (line 1, in Figure 3) and adding (line 4) the feature function that maximizes S (line 3), to the current set of feature functions at each iteration. For a more detailed version of this algorithm, we refer the reader to our previous work [1].

The remaining of this section describes the list of our candidate feature functions. Some of these feature functions are identical to those presented in our previous work, while some others are a generalization of what we did previously and others are completely novel.

Candidate feature functions. The feature generation is performed through *residue feature functions* $\phi:(\mathcal{P}\backslash\mathcal{Y})\times\mathbb{N}\rightarrow R^d$ that, given the residue position i of a protein P, computes a vector of d real-valued features.

Among the panel of candidate functions ϕ already described in our previous work, we adopted i) the *number of residues* function, ii) the *number of cysteines* function, iii) the *labels global histogram* function, iv) the *labels local histogram* function and, v) the *labels local window* function. In addition to them, we defined three other feature functions directly computed from the primary sequence and four annotation-related feature functions. We now describe in detail all these feature functions. However, since only few of these features will effectively be selected, the reader can understand the rest of our study without considering the detailed descriptions of all candidate feature functions.

- *Number of residues*: returns the number of residues in the primary sequence.

```
Given a set of feature functions $\mathbf{\Phi} = \{\phi_1, \phi_2, \dots, \phi_n\}$
Given an objective function $S(\cdot,\cdot,\cdot) \in \mathbb{R}$
Given a base learner $\mathcal{B}$
Given a dataset $D$

$\Upsilon \leftarrow \emptyset$                                              ▷ initial empty feature function set
repeat
    $\phi^* \leftarrow \operatorname{argmax}_{\phi_i \in \mathbf{\Phi} \setminus \Upsilon} S(\Upsilon \cup \{\phi_i\}, \mathcal{B}, D)$    ▷ evaluate candidate $\phi_i \in \mathbf{\Phi} \setminus \Upsilon$ functions
    $\Upsilon \leftarrow \Upsilon \cup \{\phi^*\}$                            ▷ add the best feature function
until some stopping criterion is fulfilled
return $\Upsilon$                                                   ▷ return feature function set
```

Figure 3. Forward feature function selection algorithm. In order to identify the relevant feature function set, the algorithm requires four components: a dataset, a list of candidate feature functions, a base learner and a criterion to optimize.
doi:10.1371/journal.pone.0082252.g003

- *Number of cysteines*: returns the number of cysteine residues in the primary sequence. This feature is made from the intuition that larger the number of cysteines is, larger the number of disulfide bonds will be, which usually lead to more stable structures.
- *Unnormalized global histogram*: computes twenty features, one per standard amino acid type, which are the numbers of residues of each type in the primary structure.
- *Position of residue*: returns the position i of the residue in the primary structure.
- *Relative position of residue*: computes one feature which is the residue position i divided by the protein length n. Although this feature may seem redundant with the previous one, the encoded information is different. The previous feature aims at encoding the absolute position of the residue with respect to the N-terminus. The intuition behind this feature is that the position of a residue might determine its disordered state (*e.g.*, the first four residues are prone to be disordered). Whereas, the relative position, which varies in [0,1], suggests a position regardless of the protein length.

We use the following notations to describe the annotation-related feature functions. For each type of annotation $\mathcal{A} \in \{AA, PSSM, SS, SA\}$, $\mathcal{L}_{\mathcal{A}}$ is the set of labels corresponding to $\mathcal{A}$ and $L_{\mathcal{A}} = |\mathcal{L}_{\mathcal{A}}|$ is the size of this set. We thus have: $L_{AA} = 20$, $L_{PSSM} = 21$ (the twenty amino acids and the gap), $L_{SS} = 3$, $L_{SA} = 2$. For a given primary structure of length n, an annotation $\mathcal{A}$ is represented as a set of probabilities $\alpha^{\mathcal{A}}_{i,l} \in [0,1]$ where $i \in [1,n]$ denotes the residue position and $l \in \mathcal{L}_{\mathcal{A}}$ is a label. *E.g.*, $\alpha^{SS}_{3,\text{helix}}$ is the probability that the third residue of the protein is part of a helix.

In the general case, the $\alpha^{\mathcal{A}}_{i,l}$ probabilities may take any value in range [0,1] to reflect uncertainty about annotations. However, since the predictions made by SSpro and ACCpro are classes and that primary structures (AA) are always known perfectly, we have:

$$\alpha^{\mathcal{A} \in \{AA,SS,SA\}}_{i,l} = \left(\begin{array}{ll} 1 & \text{if } l \text{ is the residue or predicted class of the } i-\text{th residue} \\ 0 & \qquad\qquad\qquad \text{otherwise.} \end{array} \right.$$

As PSSM elements typically range in $[-7,7]$, we scale them to [0,1] by using the function proposed in [30] and defined as following:

$$\alpha^{PSSM}_{i,l} = \left(\begin{array}{cc} 0.0 & \mathit{if}\ x \leq -5 \\ 0.5 + 0.1x & \mathit{if}\ -5 < x < 5 \\ 1.0 & \mathit{if}\ x \geq 5 \end{array} \right. ,$$

where x is the value from the raw profile matrix.

For each annotation $\mathcal{A}$, we have defined seven different feature functions:

- *Labels global histogram*: computes one feature per label $l \in \mathcal{L}_{\mathcal{A}}$, equal to $\frac{1}{n}\sum_{p=1}^{n} \alpha^{\mathcal{A}}_{i,l}$.
- *Labels local histogram*: computes one feature per label $l \in \mathcal{L}_{\mathcal{A}}$ equal to $\frac{1}{W}\sum_{p=i-W/2}^{i+W/2} \alpha^{\mathcal{A}}_{p,l}$ and one special feature equal to the percentage of out-of-bounds positions, *i.e.*, positions p such that $p \notin [1,n]$.
- *Labels local window*: computes one feature per label $l \in \mathcal{L}_{\mathcal{A}}$ and per relative position $\delta \in [-\frac{W}{2}, \frac{W}{2}]$, equal to $\alpha^{\mathcal{A}}_{i+\delta,l}$. When the position is out-of-bounds, *i.e.*, $i + \delta[1,n]$, the feature is set to 0.
- *Separation profile window*: this feature function is inspired from the *cysteine separation profile window* function, which focuses on the distances that separate consecutive cysteine residues and encodes the distances around the cysteine residue of interest into features. According to the results presented in our previous work, this feature function led to an impressive improvement of our disulfide connectivity pattern predictor. Here, we propose a generalization of this function in order to be able to tackle any kind of annotation $\mathcal{A}$. Figure 4 shows an illustration of a separation profile window of size 11 over exposed residues.

Given a residue position i, our generalized feature function describes the proximity of the $\frac{W}{2}$ closest residues of the N-terminus side to the i-th residue (respectively, the $\frac{W}{2}$ closest residues of the T-terminus side) that share a common label $l, \forall l \in \mathcal{L}_{\mathcal{A}}$. The proximity of a residue at the j-th position is expressed as the distance, in terms of number of amino acids in the primary structure, that separates the j-th from the i-th residue, *i.e.*, $|j-i|$. Note that, when using probabilistic predictors, the label of a residue is determined as the one with the highest probability $\alpha^{\mathcal{A}}_{i,\cdot}$.

When the number of residues that share l at the N-terminus side (respectively, at the T-terminus side) is insufficient, the missing distances are set to the greatest distance, *i.e*, the distance with the farthest residue that share l within the same terminus side.

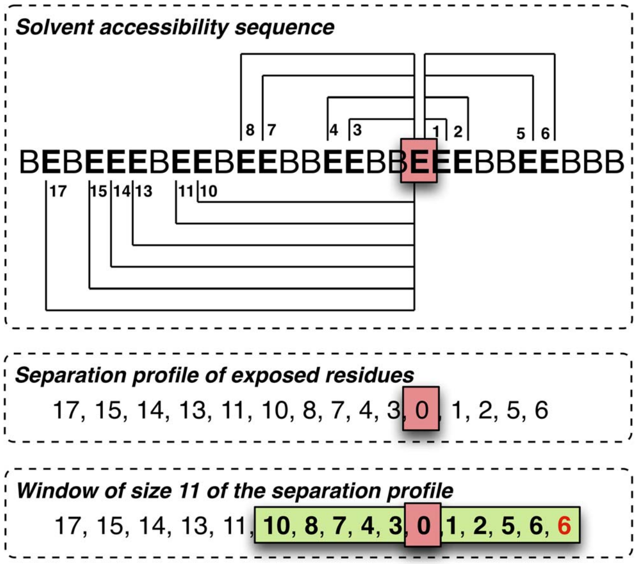
Solvent accessibility sequence
8
7
4
3
1
2
5
6
BEBEEEBEEBEEBBEEBBEEEBBEEBBB
17
15
14
13
11
10
Separation profile of exposed residues
17, 15, 14, 13, 11, 10, 8, 7, 4, 3, 0, 1, 2, 5, 6
Window of size 11 of the separation profile
17, 15, 14, 13, 11, 10, 8, 7, 4, 3, 0, 1, 2, 5, 6, 6

Figure 4. Illustration of the *separation profile window* function on exposed residues. Top: the functions first computes the amino acid distances that separate the residue of interest (highlighted by a red square). Middle: the separation profile of exposed residues. Bottom: the feature function returns the window (highlighted by a green rectangle) of size 11 centered around the residue of interest. In this example, the window slightly goes beyond the end of the sequence. As explained in the main text, in such cases we replace non available features by the maximal possible value, which is the 6 shown in red here.
doi:10.1371/journal.pone.0082252.g004

- *Labeled segments window*: this is similar to the *labels local window* function except that rather than describing neighboring residues at position $i+\delta$, it describes neighboring segments $s_{i+\delta}$. A segment consists in a sub-sequence of consecutive residues that share a common label l, in the sense of the highest probability $\alpha^{\mathcal{A}}_{\cdot,\cdot}$.

Therefore, given a segment s_i, the function returns one description of this segment (in the form of feature vectors) per relative position $\delta\in[-\frac{W}{2},\frac{W}{2}]$. A segment $s_{i+\delta}$ is described by $L_{\mathcal{A}}$ (one per label $l\in\mathcal{L}_{\mathcal{A}}$) plus one features. Among the first $L_{\mathcal{A}}$ features, the one corresponding to the label of $s_{i+\delta}$ is equal to 1 while the other ones are set to 0. The last feature is the length of $s_{i+\delta}$. When the position $s_{i+\delta}$ is out-of-bounds the features are all set to 0.

- *Dimeric global histogram*: this feature function is an extension of *labels global histogram* with the difference that instead of calculating the frequency of occurrence of each single label, it computes the frequency of occurrence of each pairs of labels. A pair of labels is formed by the labels of two consecutive residues (a word of size 2). The hope is that the distribution of some pairs of labels are significantly different in the case of disordered residues with respect to ordered ones. For example, a larger proportion of consecutive exposed residues may intuitively involve a larger disposition to form disordered regions. More formally, it returns one feature per pair of labels $(l_i,l_j)\in\mathcal{L}_{\mathcal{A}}\times\mathcal{L}_{\mathcal{A}}$, equal to

$$\frac{1}{n-1}\sum_{p=1}^{n-1}1\left\{\underset{l\in\mathcal{L}_{\mathcal{A}}}{\operatorname{argmax}}\;\alpha^{\mathcal{A}}_{p,l}=l_i\wedge\underset{l\in\mathcal{L}_{\mathcal{A}}}{\operatorname{argmax}}\;\alpha^{\mathcal{A}}_{p+1,l}=l_j\right\}.$$

- *Dimeric local histogram*: this feature function is identical to the dimeric global histogram one except that it computes the frequency within a sliding window. More formally, given a residue position k, it returns one feature per pair of labels $(l_i,l_j)\in\mathcal{L}_{\mathcal{A}}\times\mathcal{L}_{\mathcal{A}}$ equal to

$$\frac{1}{W}\sum_{p=k-W/2}^{k+W/2}1\left\{\underset{l\in\mathcal{L}_{\mathcal{A}}}{\operatorname{argmax}}\;\alpha^{\mathcal{A}}_{p,l}=l_i\wedge\underset{l\in\mathcal{L}_{\mathcal{A}}}{\operatorname{argmax}}\;\alpha^{\mathcal{A}}_{p+1,l}=l_j\right\}.$$

Our candidate feature functions are summarized in Table 2. Note that five of them are parameterized by window size parameters. To apply the feature function selection algorithm, we consider the following discrete sets of window sizes:

- Local windows, separation profile window, labeled segments window and dimeric local histogram: 1, 5, 11, 15, 21.
- Local histograms: 10, 20, 30, 40, 50, 60, 70, 80, 90.

This setting leads to a total of 109 candidate features functions.

Ensembles of extremely randomized trees

This tree-based ensemble method, proposed by Geurts *et al.* [31], is similar to the popular Random Forests approach [32]. The

Table 2. Feature functions used in our experiments to encode residues.

Symbol	Parameter	d	Description
n	-	1	Number of residues
n_C	-	1	Number of cysteines
n_{AA}	-	20	Unnormalized global histogram
i	-	1	Position of residue
i/n	-	1	Relative position of residue
$h^{global}(\mathcal{A})$	-	$L_{\mathcal{A}}$	Labels global histogram
$h^{local}(\mathcal{A},W)$	window size	$L_{\mathcal{A}}+1$	Labels local histogram
$w(\mathcal{A},W)$	window size	$W.L_{\mathcal{A}}$	Labels local window
$sep(\mathcal{A},W)$	window size	$W-1$	Separation profile window
$seg(\mathcal{A},W)$	window size	$W.(L_{\mathcal{A}}+1)$	Labeled segments window
$di^{global}(\mathcal{A})$	-	$L^2_{\mathcal{A}}$	Dimeric global histogram
$di^{local}(\mathcal{A},W)$	window size	$L^2_{\mathcal{A}}$	Dimeric local histogram

Symbols, parameters, number of features (d) and description of our candidate feature functions. Top: feature functions that are directly computed from the primary structure. Bottom: feature functions defined for every kind of annotation $\mathcal{A}\in\{AA,\,PSSM,\,SS,\,SA\}$.
doi:10.1371/journal.pone.0082252.t002

Table 3. Forward feature functions selection with 10 train/test splits.

Fold	Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5
1	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,60)$	$w(SS,11)$	$w(AA,1)$
2	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,60)$	$w(SS,11)$	$w(AA,11)$
3	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,60)$	$w(SS,11)$	$w(AA,5)$
4	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,50)$	$w(SS,11)$	$w(AA,1)$
5	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,50)$	$w(SS,15)$	$w(AA,15)$
6	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,60)$	$w(SS,15)$	$w(AA,5)$
7	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,70)$	$w(SS,15)$	$w(AA,15)$
8	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,50)$	$w(SS,11)$	$w(AA,5)$
9	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,50)$	$w(SS,11)$	$w(AA,1)$
10	$w(PSSM,21)$	$sep(SA,21)$	$h^{local}(AA,60)$	$w(SS,11)$	$w(AA,1)$
Mean					
Cross-validated	0.852 ± 0.003	0.876 ± 0.003	0.884 ± 0.003	0.890 ± 0.003	0.894 ± 0.003
Validation	0.850 ± 0.029	0.874 ± 0.021	0.883 ± 0.022	0.888 ± 0.022	0.892 ± 0.22

Mean: averages over the ten *cross-validated scores* and the ten *validation scores*. The cross-validated score is the mean of AUC scores obtained when cross-validating the training set of a run. The validation score is the AUC score obtained when evaluating the test set.

doi:10.1371/journal.pone.0082252.t003

main differences with the latter are that extremely randomized tree ensembles (ETs) do not rely on bootstrap replicates (unlike the Random Forests method, each tree is built using all learning samples), and that cut-points are selected in a random fashion, which was shown to lead to better generalization performances [31]. The method has three hyper-parameters: K, the number of random splits tested per node creation, T, the number of trees composing the ensemble, and $N_{\min}$, the minimum number of samples required to allow for splitting a node.

We use the probabilistic version of ETs, in which each leaf is associated with a probability of disorder, which is the empirical proportion of disordered residues among the training samples associated to that leaf. In order to make one prediction, we traverse each of the T trees and return the average of the probabilities of disorder associated to the corresponding T leaves.

Results

This section describes our experimental study on disordered regions prediction. The first part presents the results of the main contribution of this paper, which aims at determining a relevant representation on Disorder723. The second part aim at constructing a model based on this relevant representation and ETs, and assessing this model on Casp10 and Pdb30. In the third part, we investigate the novel feature function and attempt to interpret its role in the prediction of disordered regions.

Identification of a set of relevant feature functions

We now apply the feature function selection approach on top of ETs with the candidate feature functions of Table 2. We use a default setting of hyper-parameters of ETs that corresponds to an ensemble of 1 000 fully developed trees ($T=1\ 000$, $N_{\min}=2$) and K is set to the square root of the total number of features $\sqrt{d}$, as proposed by Geurts *et al* [31].

To avoid any risk of over-estimation, we performed the selection on 10 different train/test splits of Disorder723. The performance measure being maximized by each run is the cross-validated AUC score of the training set. Table 3 reports the selected feature functions for each of the 10 independent runs. For the five iterations we consider, we observe that the selected feature functions on each of the 10 train/test splits are always $w(PSSM,21)$, $sep(SA,21)$, $h^{local}(AA,\cdot)$ with a window size varying in $\{50,60,70\}$, $w(SS,\cdot)$ with a sliding window size in $\{11,15\}$ and $w(AA,\cdot)$ with a window size in $\{1,5,11,15\}$. Regardless to window size parameters, the fact that we observed these feature functions during each run is very strong, since the selection algorithm has to select between 109 different candidate feature functions.

Note that, among the selected feature functions, two of them (the second and the fourth) rely on predicted structural annotations (the predicted solvent accessibility and the predicted secondary structure, respectively), which tend to show that predicted structural annotations contribute to make better disordered regions predictors.

Not surprisingly, the most important feature function detected by the selection is a sliding window of evolutionary information, which confirms that disordered regions differ from ordered regions in terms of their conservation profile. This feature function is also important for many other protein structure prediction tasks (*e.g.*, [1]).

On the other hand, the second most important feature function highlighted by our algorithm, namely $sep(SA,\cdot)$, has - to our best knowledge - never been proposed in previous studies. Its discovery at a very early iteration was unexpected. It suggests that the proximities of a residue r (in terms of amino acid positions in the primary sequence) to its nearest exposed or to its nearest buried residues are correlated with the fact that r belongs to a disordered region. It is important to note the difference with $w(SA,\cdot)$. Indeed, $w(SA,\cdot)$ describes the solvent accessibility label of the flanking residues of r. The proximity is fixed and limited by the number of flanking residues to take into consideration. Whereas, $sep(SA,\cdot)$ describes the inverse. Namely, it describes the proximity of the nearest residues to r that correspond to fixed labels.

One way to explain the usefulness of this feature function is to look at the distributions of the distances that separate disordered (resp. ordered) residues to their nearest buried residue. Figure 5 shows the probability of a residue of being disordered (resp. ordered) according to the distance to its nearest buried residue,

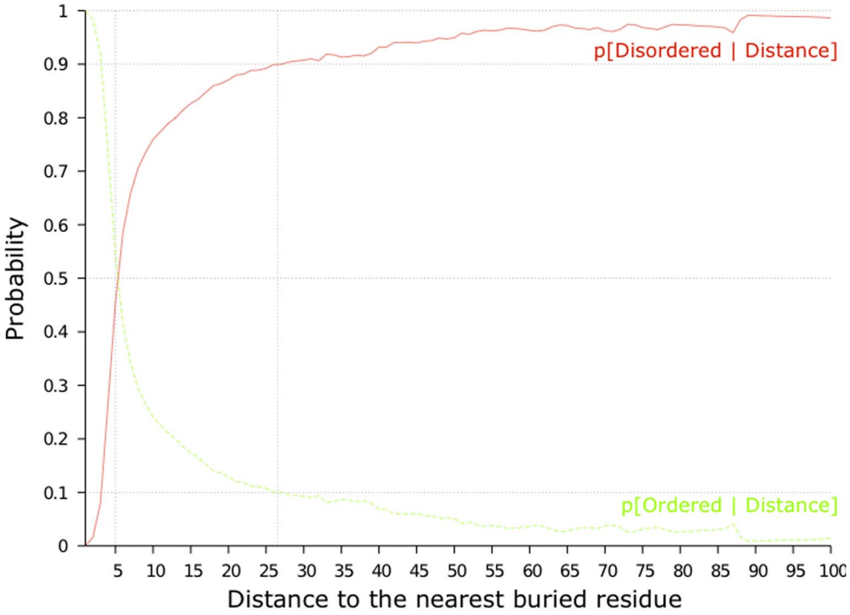

Figure 5. Probability of being (dis)ordered w.r.t. the distance to the nearest buried residue. For a given distance d, the probability $p[Disorder|d]$ of being disordered is calculated as the portion of disordered residues among the residues that have their nearest buried residue located at a distance d. We computed these curves on the actual values of the solvent accessibility of Pdb30.
doi:10.1371/journal.pone.0082252.g005

over the pdb30 dataset. We remark that the probability of a residue being disordered increases quickly when its distance to the next buried residue increases, and is above 0.5 as soon as the closest buried residue is at least 5 residues away.

Another important aspect of this discovery is that the $sep(SA,21)$ feature function is systematically detected just before the local amino acid composition $h^{local}(AA,\cdot)$ and far before $w(AA,\cdot)$. Indeed, these other two feature functions describe in different ways the sequence complexity, which is well-known to be low within disordered regions [5]. This therefore reinforces the fact that $sep(SA,21)$ may be a key-aspect in our understanding of protein disordered regions and, consequently, protein structure-function relationships.

The fourth selected feature function is a short sliding window over predicted secondary structures $w(SS,\cdot)$. The usefulness of these features may be related to the strong difference between the distributions of predicted secondary structures within disordered regions with respect to ordered ones. For example, Table 4 shows that 70.98% of disordered residues are predicted as coils against 40.57% as it is the case with ordered residues and that solely 5.76% are predicted as sheets against 20.76% for ordered regions.

Table 4. Distribution of predicted secondary structure.

	Ordered		Disordered		Total	
Predicted helices	77,989	38.67%	3,235	23.26%	81,224	37.67%
Predicted sheets	41,874	20.76%	801	5.76%	42,675	19.79%
Predicted coils	81,840	40.57%	9,873	70.98%	91,713	42.54%
Total	201,703		13,909		215,612	

Distribution of the number of ordered/disordered residues and the total number of residues for each secondary structure class on Disorder723.
doi:10.1371/journal.pone.0082252.t004

According to these results, we focus in the following on assessing the relevance of the feature functions $w(PSSM,21)$, $sep(SA,21)$, $h^{local}(AA,60)$, $w(SS,11)$ and $w(AA,1)$, where we chose windows sizes by taking the most frequent sizes reported in Table 3. Indeed, contrarily to the observation made in [1] that suggested a very small number of relevant feature functions in the context of disulfide bridge prediction, the selection algorithm identified here a larger set of interesting feature functions.

Evaluation of the selected feature functions

We now compare our models in terms of accuracy against a number of state-of-the-art methods on Disorder723, C10 and Pdb30. As previously, we use ETs with a default setting of its hyper-parameters. For each run, we use 80% of the training set to build an ensemble of trees predicting the probability to belong to a disordered region for a residue, and the remaining 20% to fix an 'optimal' decision threshold on this probability.

For Disorder723, we consider two baselines. Both evaluated their predictive performance using a 10-fold cross-validation on Disorder723. The first baseline is Cheng *et al.* [19], the authors of the Disorder723 dataset. They proposed an ensemble of 1D-recursive neural networks that reached an area under the ROC curve of 0.878. The second baseline is Eickholt *et al.* [33], who used boosted ensembles of deep networks to make predictions. They obtained a very high balanced accuracy (82.2%) and AUC (0.899).

The top of Table 5 reports our predictive performances when including successively the feature functions $w(PSSM,21)$, $sep(SA,21)$, $h^{local}(AA,60)$, $w(SS,11)$ and $w(AA,1)$, while the bottom of the Table 5 reports the scores of the two baselines from the literature. We observe that using only $w(PSSM,21)$ leads to a balanced accuracy (Acc) of 77.5%, an AUC of 0.853 and a F-measure of 49.6, which already outperforms the state of the art (46.3).

Table 5. Accuracy evaluation on the DISORDER723 dataset.

Features	Balanced Acc	Sensitivity	Specificity	AUC	F-measure
10-fold cross validation of our algorithm over DISORDER723					
$\{w(PSSM,21)\}$					
	77.5 ± 2.43	74.1 ± 5.95	80.8 ± 3.13	0.853 ± 0.028	49.6 ± 3.38
$\{w(PSSM,21),sep(SA,21)\}$					
	79.0 ± 1.95	76.5 ± 4.14	81.6 ± 2.59	0.875 ± 0.019	51.7 ± 4.20
$\{w(PSSM,21),sep(SA,21),h^{local}(AA,60)\}$					
	80.3 ± 2.17	78.2 ± 4.90	82.4 ± 2.47	0.884 ± 0.019	52.7 ± 3.85
$\{w(PSSM,21),sep(SA,21),h^{local}(AA,60),w(SS,11)\}$					
	80.6 ± 1.69	79.0 ± 4.64	82.2 ± 2.11	0.891 ± 0.020	53.4 ± 3.55
$\{w(PSSM,21),sep(SA,21),h^{local}(AA,60),w(SS,11),w(AA,1)\}$					
	81.1 ± 1.83	78.6 ± 4.69	83.5 ± 2.08	0.894 ± 0.021	55.3 ± 3.27
$\{w(PSSM,21),h^{local}(AA,60),w(SS,11),w(AA,1)\}$					
	80.4 ± 1.94	76.8 ± 5.30	83.9 ± 2.37	0.883 ± 0.026	54.5 ± 2.70
Baselines tested on DISORDER723					
Cheng *et al.* (2005) [19]	-	-	-	0.878	-
Eickholt *et al.* (2013) [33]	82.21 ± 0.49	74.60 ± 1.1	89.84 ± 0.18	0.899 ± 0.002	46.34 ± 4.5

Top: the mean and standard deviation of the scores obtained when 10-folds cross-validating Disorder723 through the relevant feature functions. Bottom: baselines using Disorder723 to assess their model.

doi:10.1371/journal.pone.0082252.t005

Moreover, we remark that by incrementally adding the remaining selected feature functions to the set systematically leads to significant improvements on Acc, AUC and F-measure. We have used the paired *t*-test on the AUC scores to statistically assess the significance of each increment. We noted that the corresponding *p*-values ($2.6e^{-3}$, $5.4e^{-4}$, $2.2e^{-4}$ and $4.6e^{-3}$) are well below the classical null hypothesis threshold (0.05). This observation reinforces the fact that the selected feature functions are relevant. When comparing our model based on all five selected feature functions to the state-of-the-art, we obtain a disordered regions predictor, which is very competitive in term of Acc (81.1%), equivalent in term of AUC (0.894) and clearly better in term of F-measure of 55.3. The middle of Table 5 shows the impact on the predictive performance of our model when we do not consider $sep(SA,21)$ among the input feature functions. As expected, the scores significantly deteriorate with a *p*-value of $1.9e^{-2}$ with respect to the model that comprise $sep(SA,21)$. This observation reinforces the fact that this kind of feature function should be taken into account when predicting disordered regions.

To assess our models on Casp10, we compare our results against several baselines such as DNdisorder and PreDNdisroder, which were developed by Eickholt *et al.* [33]. Among the baselines, a number of them participated in the 10th CASP experiment. In order to make the comparison in a fair way, we construct our models on Disorder723 using feature functions that were selected according to Disorder723. Moreover, since Disorder723 does not contain any overlapping sequences with Caps10 and that Disorder723 was formed well before Casp10, we are in the same blind prediction setting than the participants of the competition.

The top part of Table 6 reports our results with the different sets of relevant feature functions while the bottom part of Table 6 reports the scores obtained by the baselines considered in [33]. Once again, we observe that enlarging the feature functions set systematically leads to significant improvements except for $w(AA,1)$. Two reasons may explain this phenomena, either the Casp10 dataset is too small and, consequently, prone to larger variances than big datasets, or the fifth iteration of the selection procedure starts to overfit Disorder723, which means that $w(AA,1)$ is not portable to other datasets. We believe that the second reason is more likely to be the true explanation, because the function $w(AA,1)$ consists in discriminating disordered residues from ordered ones based on their amino acid type, which may be too dataset specific. As mentioned, the *p*-value of $4.6e^{-3}$ determined when including this feature set was indeed quite larger than those resulting from the inclusion of the other feature sets.

According to Table 6, we remark that our model based on $\{w(PSSM,21)$, $sep(SA,21)$, $h^{local}(AA,60)$, $w(SS,11)\}$ achieves excellent performances with respect to the state-of-the-art. We even slightly improve the state-of-the-art with a balanced accuracy of 77.29% against 77.06%, however, according to the variations, this improvement is not significant. We nevertheless outperformed the method of Eickholt *et al.* [33] (DNdisorder), which presented similar performances than our model on Disopred723.

Although Casp10 is an entirely independent test set that had no detectable similarity to available structures at this time, its very limited size does not enable it to capture the universe of protein disorder. This is why we also evaluated our model on the far larger dataset Pdb30. Table 7 compares the predictive performances obtained by three freely and easily downloadable methods (DISOPRED2[34], IUPred[35] and ESpritz[36]) with respect to our model. We observe that our approach outperforms the three baselines with a balanced accuracy of 80.3% and presents a comparable area under the ROC curve (0.883) to ESpritz, even though our approach treats each residue independently, *i.e.*, without explicitly exploiting the key-fact that disordered regions are made of contiguous residues. Figure 6 shows the ROC curves for DISORDER2, ESpritz and IUpred on PDB30. We observe that our method and ESpritz are very close to each other and that ESpritz is slightly better in the low false positive rate.

Table 6. Accuracy evaluation on the CASP10 dataset.

Features	**Balanced Acc**	**Sensitivity**	**Specificity**	**AUC**	**F-measure**
Models learnt on DISORDER723 by our algorithm and tested on CASP10					
$\{w(PSSM,21)\}$					
	71.94 ± 0.71	70.71 ± 1.3	73.16 ± 0.32	0.795 ± 0.007	39.47 ± 0.73
$\{w(PSSM,21),sep(SA,21)\}$					
	74.95 ± 0.69	70.31 ± 1.4	79.59 ± 0.29	0.834 ± 0.006	38.51 ± 0.81
$\{w(PSSM,21),sep(SA,21),h^{local}(AA,60)\}$					
	77.17 ± 0.67	71.64 ± 1.3	82.69 ± 0.28	0.847 ± 0.006	39.95 ± 0.88
$\{w(PSSM,21),sep(SA,21),h^{local}(AA,60),w(SS,11)\}$					
	77.29 ± 0.66	74.17 ± 1.3	80.41 ± 0.29	0.851 ± 0.006	40.24 ± 0.84
$\{w(PSSM,21),sep(SA,21),h^{local}(AA,60),w(SS,11),w(AA,1)\}$					
	77.35 ± 0.65	72.84 ± 1.3	81.85 ± 0.29	0.850 ± 0.006	39.82 ± 0.87
Baseline performances on CASP10 as published by the CASP10 competition					
metaprdos2 (340)	77.06 ± 0.92	64.73 ± 1.4	89.40 ± 0.98	0.8727 ± 0.006	41.24 ± 2.9
PreDisorder (125)	76.86 ± 0.67	67.19 ± 1.7	86.34 ± 0.94	0.839 ± 0.006	37.50 ± 1.5
POODLE (216)	76.84 ± 0.78	62.74 ± 1.6	90.94 ± 0.26	0.866 ± 0.006	43.06 ± 1.0
PreDNdisorder [6]	76.55 ± 0.75	61.74 ± 1.8	91.36 ± 0.61	0.864 ± 0.006	43.42 ± 1.5
ZHOU-SPARKS-X (413)	75.68 ± 0.76	64.81 ± 1.4	86.55 ± 0.96	0.859 ± 0.006	36.43 ± 1.9
DNdisorder (424)	75.19 ± 0.71	61.92 ± 1.4	88.46 ± 0.29	0.848 ± 0.006	38.02 ± 1.1
CSpritz (484)	75.13 ± 1.4	66.31 ± 1.3	83.94 ± 2.4	0.822 ± 0.007	33.64 ± 3.7
Espritz (380)	73.16 ± 1.6	59.24 ± 1.4	87.08 ± 2.6	0.846 ± 0.006	34.58 ± 4.7
espritz_nopsi_X	71.98 ± 0.97	53.10 ± 1.5	90.87 ± 0.77	0.815 ± 0.007	37.56 ± 2.4
PrDOS-CNF (369)	70.35 ± 0.88	41.95 ± 1.8	98.74 ± 0.14	0.896 ± 0.005	52.50 ± 1.4
biomine_dr_mixed (478)	69.17 ± 0.68	39.95 ± 1.4	98.40 ± 0.11	0.884 ± 0.006	49.40 ± 1.3
biomine_dr_pdb_c (228)	67.81 ± 1.2	36.88 ± 2.6	98.74 ± 0.15	0.882 ± 0.006	47.65 ± 2.1
iupred_short	63.26 ± 0.70	30.68 ± 1.5	95.84 ± 0.25	0.664 ± 0.007	32.34 ± 1.2

Top: the scores obtained when evaluating CASP10 on models learnt on DISORDER723 through the relevant feature functions found on DISORDER723. Bottom: comparison of a number of predictors, which participated in or evaluated their model to the 10th CASP experiment. These results were reported by [33]. In parenthesis: the group number of the methods that participated in the CASP10 experiment. The standard deviations were calculated by a bootstrapping procedure in which 80% of the dataset was sampled 1000 times, as it was done by [33].

doi:10.1371/journal.pone.0082252.t006

Note that since PDB30 and DISORDER723 are independent, the evaluation of our model is fair. However, we do not have access to the learning stage of the compared methods, which has possibly used sequences similar to the ones present in Pdb30. This may lead to an over-estimation of the predictive performance of those methods.

Discussion

Predicting and understanding the nature of disordered regions is a key sub-problem of protein structure and function inference. This paper has adapted the algorithm presented in our previous work [1] on disulfide bridge prediction in order to identify the best

Table 7. Evaluation on the PDB30 dataset.

Method	**Balanced Acc**	**Sensitivity**	**Specificity**	**AUC**	**F-measure**
Our method	80.36 ± $4.8e^{-2}$	82.67 ± $9.3e^{-2}$	78.06 ± $2.3e^{-2}$	0.8835 ± $4.5e^{-4}$	33.12 ± $6.3e^{-2}$
At 94.7% of specificity	76.73 ± $5.1e^{-2}$	58.79 ± $10.1e^{-2}$	94.67 ± $1.3e^{-2}$	0.8835 ± $4.5e^{-4}$	49.89 ± $8.4e^{-2}$
DISOPRED2 [34]	76.96 ± $5.7e^{-2}$	60.01 ± $11.3e^{-2}$	93.90 ± $1.3e^{-2}$	0.8658 ± $4.9e^{-4}$	48.40 ± $8.8e^{-2}$
ESpritz [36]	78.49 ± $5.6e^{-2}$	62.26 ± $11.3e^{-2}$	94.71 ± $1.3e^{-2}$	0.8856 ± $4.4e^{-4}$	52.20 ± $9.2e^{-2}$
IUPred [35]	74.99 ± $5.8e^{-2}$	55.98 ± $11.4e^{-2}$	93.99 ± $1.4e^{-2}$	0.8363 ± $5.6e^{-4}$	46.13 ± $8.9e^{-2}$

Predictive performances of three freely and easily downloadable methods on PDB30. The standard deviations were calculated over the same 100 bootstrap copies of the whole dataset. Given the huge size of the dataset, all differences (even if they are sometimes tiny) are statistically significant. Notice that (except for the AUC calculation), our method uses a classification threshold that was selected on the training dataset (Disorder723) so as to maximize the balanced accuracy, which explains its difference in (sensitivity, specificity) pattern, as compared to the other methods. Changing the threshold so as to yield a 94.7% specificity on Pdb30, would reduce its sensitivity to 58.8%.

doi:10.1371/journal.pone.0082252.t007

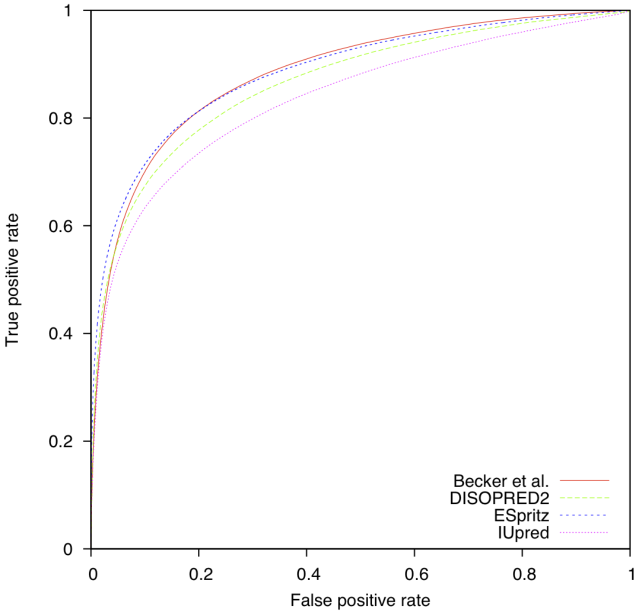

Figure 6. ROC curves on PDB30 dataset. ROC curve of our method (Becker *et al.*) and three freely downloadable predictors: DISPRED2 [34], ESpritz [36] and IUPred [35].
doi:10.1371/journal.pone.0082252.g006

way to represent protein residues in order to be usable by disordered region predictors. To this end, we used extremely randomized tree ensembles as an 'off-the-shelf' base learner in our feature function selection pipeline. We applied our approach to the Disorder723 dataset from the literature, so as to select relevant subsets of feature functions and to build simple residue-wise disorder prediction models.

Our experiments have shown that the combination of the feature functions $w(PSSM,21)$ (a local window of size 21 of evolutionary information), $sep(SA,21)$ (a window of 21 of the separation profile of predicted solvent accessibility), $h^{local}(AA,60)$ (a local histogram of size 60 of primary structure) and $w(SS,11)$ (a local window of size 11 of predicted secondary structure) is a relevant representation of protein residues in the context of disordered regions prediction.

From a biological point of view, the major contribution of this paper is the discovery of the $sep(SA,\cdot)$ feature function, which has - to our best knowledge - never been highlighted as important in this context. This observation suggests that the proximities (in terms of amino acid distances) between consecutive exposed (and consecutive buried) residues should play a role in the formation of disordered regions and, consequently, in protein structure-function relationships.

To validate these observations with respect to the state-of-the-art in disorder prediction, we also evaluated our model on the set of proteins used in the Casp10 competition. On Casp10, our model constructed on the Disorder723 dataset turned out to obtain a very competitive assessment in terms of various predictive accuracy indicators, in spite of the fact that our work was focusing on feature identification rather than accuracy maximization. Since Casp10 is a small dataset that does not capture the whole universe of protein disorder, we further assessed our model on the independent and very large Pdb30 dataset, which contains 12,090 proteins and 2,991,008 residues. On Pdb30, our model obtained as well very competitive results with respect to three state-of-the-art methods, by clearly beating two of them and being at a tie with the third one.

From a methodological point of view, our paper also shows that the systematic feature family selection pipeline proposed in [1] and adapted here, is a viable and robust approach to yield interpretable information about relevant representations for protein structure inference and allows at the same time to build predictors with state-of-the-art accuracy. Still, it might be the case that extremely randomized tree ensembles with their defaults settings are not the best classifiers for disordered regions prediction. Also, in our predictors we treated each residue independently, *i.e.*, without taking advantage of the structured nature of the problem. Therefore, a main direction for future research is to evaluate more sophisticated classifiers using the feature functions highlighted by the present study.

Acknowledgments

This work was carried out at the GIGA-Bioinformatics platform of the University of Liège. We would like to thank the cluster of the platform and the NIC3 cluster of the the University of Liège for our intensive usage.

Author Contributions

Conceived and designed the experiments: JB FM LW. Performed the experiments: JB. Analyzed the data: JB FM LW. Contributed reagents/materials/analysis tools: JB FM. Wrote the paper: JB FM LW.

References

1. Becker J, Maes F, Wehenkel L (2013) On the relevance of sophisticated structural annotations for disulfide connectivity pattern prediction. PLoS One 8: e56621.
2. Uversky VN, Oldfield CJ, Dunker AK (2005) Showing your id: intrinsic disorder as an id for recognition, regulation and cell signaling. Journal of Molecular Recognition 18: 343–384.
3. Uversky VN (2009) The mysterious unfoldome: structureless, underappreciated, yet vital part of any given proteome. BioMed Research International 2010.
4. Dyson HJ, Wright PE (2005) Intrinsically unstructured proteins and their functions. Nature Reviews Molecular Cell Biology 6: 197–208.
5. Wootton JC (1994) Non-globular domains in protein sequences: automated segmentation using complexity measures. Computers & chemistry 18: 269–285.
6. Jones D, Ward J (2003) Prediction of disordered regions in proteins from position specific score matrices. Proteins: Structure, Function, and Bioinformatics 53: 573–578.
7. Deng X, Eickholt J, Cheng J (2009) Predisorder: ab initio sequence-based prediction of protein disordered regions. BMC bioinformatics 10: 436.
8. Peng K, Vucetic S, Radivojac P, CELESTE J, Dunker A, et al. (2005) Optimizing long intrinsic disorder predictors with protein evolutionary information. Journal of bioinformatics and computational biology 3: 35–60.
9. Yang Z, Thomson R, McNeil P, Esnouf R (2005) Ronn: the bio-basis function neural network technique applied to the detection of natively disordered regions in proteins. Bioinformatics 21: 3369–3376.
10. Zhang T, Faraggi E, Xue B, Dunker A, Uversky V, et al. (2012) Spine-d: accurate prediction of short and long disordered regions by a single neural-network based method. Journal of Biomolecular Structure and Dynamics 29: 799–813.
11. Shimizu K, Hirose S, Noguchi T (2007) Poodle-s: web application for predicting protein disorder by using physicochemical features and reduced amino acid set of a position-specific scoring matrix. Bioinformatics 23: 2337–2338.
12. Hirose S, Shimizu K, Kanai S, Kuroda Y, Noguchi T (2007) Poodle-l: a two-level svm prediction system for reliably predicting long disordered regions. Bioinformatics 23: 2046–2053.
13. Shimizu K, Muraoka Y, Hirose S, Tomii K, Noguchi T (2007) Predicting mostly disordered proteins by using structure-unknown protein data. BMC bioinformatics 8: 78.
14. Vullo A, Bortolami O, Pollastri G, Tosatto S (2006) Spritz: a server for the prediction of intrinsically disordered regions in protein sequences using kernel machines. Nucleic Acids Research 34: W164–W168.
15. Ishida T, Kinoshita K (2008) Prediction of disordered regions in proteins based on the meta approach. Bioinformatics 24: 1344–1348.

16. Mizianty M, StachW, Chen K, Kedarisetti K, Disfani F, et al. (2010) Improved sequence-based prediction of disordered regions with multilayer fusion of multiple information sources. Bioinformatics 26: i489–i496.
17. Monastyrskyy B, Fidelis K, Moult J, Tramontano A, Kryshtafovych A (2011) Evaluation of disorder predictions in casp9. Proteins: Structure, Function, and Bioinformatics 79: 107–118.
18. Deng X, Eickholt J, Cheng J (2012) A comprehensive overview of computational protein disorder prediction methods. Molecular BioSystems 8: 114–121.
19. Cheng J, Sweredoski MJ, Baldi P (2005) Accurate prediction of protein disordered regions by mining protein structure data. Data Mining and Knowledge Discovery 11: 213–222.
20. Berman H, Westbrook J, Feng Z, Gilliland G, Bhat T, et al. (2000) The protein data bank. Nucleic Acids Research 28: 235–242.
21. Sven M, Burkhard R (2003) Uniqueprot: creating representative protein sequence sets. Nucleic Acids Res : 3789–3791.
22. Sander C, Schneider R (1991) Database of homology-derived protein structures and the structural meaning of sequence alignment. Proteins: Structure, Function, and Bioinformatics 9: 56–68.
23. Dondoshansky I, Wolf Y (2002) Blastclust (ncbi software development toolkit). NCBI, Bethesda, Md.
24. Altschul S, Madden T, Schaffer A, Zhang J, Zhang Z, et al. (1997) Gapped blast and psi-blast: a new generation of protein database search programs. Nucleic Acids Research 25: 3389–3402.
25. Pruitt K, Tatusova T, Brown G, Maglott D (2012) Ncbi reference sequences (refseq): current status, new features and genome annotation policy. Nucleic Acids Research 40: D130–D135.
26. Cheng J, Randall AZ, Sweredoski MJ, Baldi P (2005) Scratch: a protein structure and structural feature prediction server. Nucleic acids research 33: W72–W76.
27. Wang L, Sauer UH (2008) Ond-crf: predicting order and disorder in proteins conditional random fields. Bioinformatics 24: 1401–1402.
28. Xue B, Dunbrack RL,Williams RW, Dunker AK, Uversky VN (2010) Pondr-fit: a meta-predictor of intrinsically disordered amino acids. Biochimica et Biophysica Acta (BBA)-Proteins & Proteomics 1804: 996–1010.
29. Kozlowski L, Bujnicki J (2012) Metadisorder: a meta-server for the prediction of intrinsic disorder in proteins. BMC bioinformatics 13: 111.
30. Kim H, Park H (2003) Protein secondary structure prediction based on an improved support vector machines approach. Protein Engineering 16: 553–560.
31. Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. Machine Learning 63: 3–42.
32. Breiman L (2001) Random forests. In: Machine Learning. pp. 5–32.
33. Eickholt J, Cheng J (2013) Dndisorder: predicting protein disorder using boosting and deep networks. BMC Bioinformatics 14: 88.
34. Ward JJ, Sodhi JS, McGuffin LJ, Buxton BF, Jones DT (2004) Prediction and functional analysis of native disorder in proteins from the three kingdoms of life. Journal of molecular biology 337: 635–645.
35. Dosztanyi Z, Csizmok V, Tompa P, Simon I (2005) The pairwise energy content estimated from amino acid composition discriminates between folded and intrinsically unstructured proteins. Journal of molecular biology 347: 827–839.
36. Walsh I, Martin AJ, Di Domenico T, Tosatto SC (2012) Espritz: accurate and fast prediction of protein disorder. Bioinformatics 28: 503–509.