# Fast and accurate computational prediction of functions of intrinsic disorder in proteins

by

Fanchi Meng

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Software Engineering and Intelligent Systems

Department of Electrical and Computer Engineering
University of Alberta

## Abstract

Intrinsically disordered regions (IDRs) in proteins lack stable three dimensional structure under physiological conditions. IDRs are prevalent in nature, functionally important, and difficult to characterize experimentally due to their unstructuredness. As a result, many computational methods have been developed to detect IDRs from protein sequences in the past forty years. However, the annotations of functions of IDRs lag behind the rapidly accumulating number of newly discovered proteins since they remain largely determined via time-consuming and costly experiments. Sequence alignment (SA) and existing predictors of functions of IDRs provide a way to characterize functions of IDRs. However, SA is only applicable when the protein under study shares sufficiently high sequence similarity with annotated homologous sequences, and existing predictors cover only a small portion of all functions of IDRs. We use SA and existing predictors to characterize functions of IDRs in human dengue virus, and we use this project to investigate the ability of these approaches to determine functions of IDRs. Results show that SA is able to find certain functions that are related to IDRs, but it under predicts the number of IDRs that carry out given functions. Moreover, existing predictors of functions of IDRs only cover protein-binding functions, and do not cover other types of functions. To this end, we address the prediction of the most prevalent function that does not involve binding and cannot be predicted by current predictors, i.e., the disordered flexible linkers (DFLs). DFLs are IDRs that serve as flexible linkers/spacers in multi-domain proteins or between structured constituents in domains. We conceptualized, developed and empirically assessed a first-of-its-kind sequence-based predictor of DFLs, DFLpred. DFLpred uses a set of empirically selected features that quantify propensities to form certain secondary structures, disordered regions and structured regions, which are processed by a fast linear model. DFLpred secures area under the ROC curve (AUC) equal 0.715, is significantly better than the existing alternatives, and it is fast enough to be used on the whole proteome scale. We also address the prediction of IDRs that carry out multiple functions, i.e., disordered moonlighting regions (DMRs). We conceptualized, designed and empirically evaluated a first-of-its-kind sequence based predictor of DMRs, DMRpred. We developed novel amino acid indices that quantify propensities for functions relevant to DMRs and used evolutionary conservation, putative solvent accessibility and intrinsic disorder derived from the input sequence to build a rich profile that is suitable to accurately predict DMRs. We processed this profile to derive innovative features that are input into a Random Forest model to generate the predictions. DMRpred secures AUC = 0.86 and accuracy = 82%. We demonstrate that these results are significantly better than the results from alternative methods. DMRpred is fast and can finish a prediction for a protein of typical length of about 500 residues in less than one minute. We provide convenient webservers to make DFLpred and DMRpred available to the research community. To sum up, motivated by the drawbacks of the current computational approaches for the functional characterization of IDRs, we contribute two novel methods that provide accurate predictions of important functional types of IDRs.

## Preface

This thesis is an original work conducted by Fanchi Meng. The research project, of which this thesis is a part, received funding from the China Scholarship Council.

This thesis includes materials and results from the following publications:

1. Meng, F., V.N. Uversky, and L. Kurgan, *Comprehensive review of methods for prediction of intrinsic disorder and its molecular functions.* Cellular and Molecular Life Sciences, 2017. **74**(17): p. 3069-3090.
2. Meng, F., V. Uversky, and L. Kurgan, *Computational Prediction of Intrinsic Disorder in Proteins*, in Dunn BM, (Ed.), *Current Protocols in Protein Science*. 2017, 88:2.16.1–2.16.14, John Wiley & Sons, Inc.
3. Meng, F., et al., *Unstructural biology of the dengue virus proteins.* FEBS Journal, 2015. **282**(17): p. 3368-3394.
4. Meng, F. and L. Kurgan, *DFLpred: High-throughput prediction of disordered flexible linker regions in protein sequences.* Bioinformatics, 2016. **32**(12): p. i341-i350.

Chapter 2 includes materials from ref. [1] and ref. [2]. Chapter 3 includes materials and results from ref. [3]. Chapter 4 includes materials and results from ref. [4]. Materials and results from Chapter 5 were submitted for publication. I was responsible for the data collection, data analysis, design of the predictive models, analysis of results, and writing of the manuscripts across all these articles.

## Acknowledgements

I could not finish this exciting research without the careful guidance from my supervisor Dr. Lukasz Kurgan. I would like to say thank you to Dr. Lukasz Kurgan, for his passion, motivation and elaborate effort to make me do better in my research area.

I would like to thank Dr. Vladimir Uversky, who co-authored three research papers that are relevant to the research described in this thesis. His knowledge and ideas greatly helped my research.

I would like to thank my fellow lab mates, especially Chen Wang, for his help in building webservers of the predictive models.

I would like to thank China Scholarship Council for their financial support throughout my PhD program.

Finally and the most importantly, I would like to thank my parents for their unconditional love, support, encouragement, and their understanding of my choices.

# Table of Contents

# List of Tables

## List of Figures

## List of Abbreviations

**AA**-Amino Acid

**AUC**-Area Under receiver operating characteristic Curve

**BLAST**-Basic Local Alignment Search Tool

**CASP**-Critical Assessment of protein Structure Prediction

**DENV**-Dengue Virus

**DFL**-Disordered Flexible Linkers

**DISP**-Disorder Prediction

**DMR**-Disordered Moonlighting Region

**DNA**-Deoxyribonucleic Acid

**FN**-False Negative

**FP**-False Positive

**FPR**-False Positive Rate

**IDP**-Intrinsically Disordered Protein

**IDR**-Intrinsically Disordered Region

**LR**-Logistic Regression

**MoRF**-Molecule Recognition Function

**NB**-Naive Bayes

**NMR**-Nuclear Magnetic Resonance

**NN**-Neural Network

**RNA**-Ribonucleic Acid

**ROC**-Receiver Operating Characteristic

**SA**-Sequence Alignment

**SF**-Scoring Function

**SVM**-Support Vector Machine

**TN**-True Negative

**TP**-True Positive

**TPR**-True Positive Rate

# Chapter 1

# Introduction

Proteins are essential to the structure and function of all living cells and viruses. Traditionally, the cellular functions of proteins are assumed to be depended on their fixed (rigid) three-dimensional structures. However, an increasing amount of evidence suggests that a large number of proteins contain functional regions that lack stable three-dimensional structure and form dynamic conformational ensembles [1]. These regions are called intrinsically disordered regions (IDRs), and proteins that include IDRs are called intrinsically disordered proteins (IDPs). Intrinsic disorder is abundant in nature. According to recent estimates, IDRs can be found in over 30% of proteins in Eukaryote, about 20% of proteins in Bacteria and Archaea, and over 20% of proteins in Viruses [2, 3]. In spite of the absence of the well-defined three-dimensional structure, IDRs are key players in various functions such as binding to partners including proteins and nucleic acids, intra-protein interactions, and intra and inter-domain linkers [3-6].

Although IDRs/IDPs are prevalent in nature, only a small portion of them was so far discovered and curated. Given the availability of repositories, such as DisProt [7] and Protein Data Bank (PDB) [8], that contain human curated annotations of disorder, computational predictors of intrinsic disorder trained on these data are used to bridge the gap between the small number of annotated proteins and the vast amount of unannotated proteins. To date, approximately 70 predictors of disorder were published in peer-reviewed venues [9-15]. Recent comparative reviews reveal that current predictors of disorder provide putative annotations of disorder with high accuracy [15-19]. Availability of these predictions is crucial to facilitate experimental studies of disorder and to address practical problems in other areas, such as target selection for structural genomics [20]. They are also used to analyze prevalence and functional characteristics of disorder on large scale across functionally related proteins [21-30] and in whole proteomes [2, 3, 31-33]. To ease the access to the predictions for large set of proteins, computationally generated disorder annotations are deposited into several large-scale databases, such as MobiDB [34] and $D^2P^2$ [35]. These databases are popular as evidenced by high citation counts (relative to when they were published), with 107 citations for ref. [34] that was published in 2015 and 175 citations for ref. [35] from 2012 (source: Google Scholar as of Nov 29, 2017). Thanks to the availability of the predictors and databases researchers can now easily access these data. However, while the putative annotations of IDRs are widely accessible they lack functional annotations.

Knowledge of functions of IDRs has practical applications, besides providing invaluable advances in basic science. IDRs were found to be implicated in a wide range of illnesses [6, 36-39], including genetic [40], degenerative [41] and cardiovascular [42] diseases. They are constitute attractive and novel class of targets for rational drug design [43-48]. Thus, the development of models that accurately predict these functions is vital. Several existing methods, such as ANCHOR [49], MoRFpred [50] and DisoRDPbind [51], provide putative annotations for certain types of functions of IDRs. These functions include protein-protein interactions (ANCHOR and MoRFpred) and protein-nucleic acids interactions (DisoRDPbind). The abovementioned methods have already attracted significant levels of interest. For example, the webserver of MoRFpred has been used over 9000 times by about 2800 unique users coming from 72 countries and 742 cities (source: Google Analytics as of November 29, 2017); the corresponding article that was published in 2012 [50] was cited 160 times (source: Google Scholar as of Nov 29, 2017). Similarly, the webserver of DisoRDPbind has been utilized over 2700 times by about 500 unique users from 52 countries and 245 cities (source: Google Analytics as of November 29, 2017); the DisoRDPbind article from late 2015 [51] was already cited 25 times (source: Google Scholar as of Nov 29, 2017). However, while these methods are popular and useful, there are no methods to predict many of the other functions of intrinsic disorder.

## 1.1 Thesis statements and goals

The aim of this research is to characterize functions of IDRs and to develop accurate computational methods that use protein sequences to determine functions of IDRs. We define the following thesis statements:

- Sequence alignment (SA) can be used to find functions that are related to IDRs. We investigate whether SA is sufficient since it may under predict functions when sequences under study lack annotated homologs.
- Current predictors of functions of IDRs only cover a small portion of all functions of IDRs. Methods that accurately predict other functions of IDRs can be developed and evaluated. We focus on computational predictors of disordered flexible linkers (DFLs) and disordered moonlighting regions (DMRs). This is motivated by availability of corresponding experimental annotations in the DisProt database, which is main source of human-curated IDRs.

To address the above thesis statements, we set the following three goals:

**Goal 1: Characterization of functions of IDRs in human dengue virus (DENV).** DENV is a family of several serotypes that have relatively small proteomes. The DENV proteins are well-annotated but the functional regions in the sequences are not yet linked with the intrinsic disorder. We find that these viral proteins include a substantial number of disordered regions that lack functional annotations. We use alignment and existing predictors to annotate disordered regions and functions of these disordered regions.

**Goal 2: Fast and accurate computational prediction of disordered flexible linker (DFL) regions.** DFLs are the most abundant functions that are carried out by IDRs and that do not involve binding to partners. We address the shortage of predictive models for this function. We conceptualize, design, implement, test and deploy a novel, runtime efficient and accurate predictive model. Our model provides predictions for proteins where the currently used alignment-based approach does not work.

**Goal 3: Fast and accurate computational prediction of disordered moonlighting regions (DMRs).** DMRs are disordered regions that carry multiple functions. While these regions are functionally very important, there are no computational model to predict them. We address this by proposing, designing, implementing, testing and deploying a novel predictive model that is fast and accurate, and which provides predictions for proteins where alignment based on sequence similarity does not work.

## 1.2 Outline

Chapter 2 provides background information on proteins, protein disorder, and functions of IDRs. It also provides background information on designing and evaluating computational models.

In Chapter 3 we address the first goal by characterizing the functions of IDRs in the human dengue virus. First, we collect the complete proteome of the human dengue virus, and annotate IDRs of these proteins by using state-of-the-art predictor of IDRs. We then annotate functions of these IDRs by sequence alignment and one predictor for a specific type of function, molecular recognition feature (MoRF) that involves protein-protein binding. Next, we explore the putative functions of IDRs and analyse the enrichment of disordered and MoRF residues in regions with specific functions. Finally, we draw a conclusion that although sequence alignment and current predictors for functions of IDRs can find some functions of IDRs, they are not sufficient to find all functions and often times under predict functions related to the intrinsic disorder. Consequently, we call for the development of new predictors of functions of IDRs.

Using the results from Chapter 3, we find that the most prevalent function that does not involve binding and cannot be predicted by current predictors is disordered flexible linker (DFL). Thus, in Chapter 4 we address the second goal by proposing a fast computational predictor for DFLs. First, we define DFLs and discuss existing alternative methods that could be used to find DFLs. We then explore limitations of these alternatives. Next, we describe the design of the new model to predict DFLs and empirically compare results produced by this model with the results generated by the alternative methods on a blind test dataset. We compare predictive quality and runtime. We use two case studies to show how to understand and use outputs of the proposed model. Lastly, we apply the proposed model on the complete human proteome and analyse the resulting putative DFLs.

In chapter 5, we address the prediction of another class of disordered regions that carry out multiple functions, the disordered moonlighting regions (DMRs). First, we define DMRs and compare them with moonlighting proteins. Next, we describe the design of the predictive model for DMRs and then empirically compare it with alternative computational methods that can be currently utilized to find putative DMRs. We use a case study to visualize and describe results generated by the new predictive model. Lastly, we analyse the putative DMRs that we predict with the new predictive model in complete human proteome.

In the last chapter, we summarize this research and present our conclusion. We list the significant contributions and point out possible future research directions.

# Chapter 2

# Background and related work

## 2.1 Background on proteins, intrinsic disorder and its functions

### 2.1.1 Proteins and intrinsic disorder in proteins

Proteins are ubiquitous and crucial elements of cells in all living beings including simple bacteria, virus and complex mammals like human. The word "protein" was originally derived from a Greek adjective *proteos* that means "of the first rank or position", which loosely translates as "of primary importance" [52]. Proteins carry out many functions, examples include catalysis of chemical reactions (enzymes), signaling and transportation (hemoglobin) and immune responses (antibodies), to list just a few. They are composed of one or more polypeptide chains, which are linear chains built from amino acids (AA). Different AA chains fold into different three-dimensional structures to perform their functions. For most proteins, their AA chains fold into specific spatial conformations. The spatial conformations, i.e., protein structures, are typically categorized into four levels. Primary structure is the linear sequence of amino acids joined by peptide bonds. Secondary structure refers to local and regularly occurring patterns, such as α-helices and β-strands. Tertiary structure describes how the protein chains are folded into a three dimensional shape. Some proteins include multiple polypeptide chains and in these cases the quaternary structure is defined as the spatial arrangements of these polypeptide chains.

The classical "sequence to structure to function" paradigm has defined for decades how we learn protein functions. This view is centered on the idea that the function(s) of a given protein is(are) determined by its unique and well-defined three-dimensional structure, which in turns is uniquely determined by the corresponding sequence of amino acids [1]. Typically, the three-dimensional structures of proteins are solved through experiments such as X-Ray crystallography and Nuclear Magnetic Resonance (NMR). While many proteins maintain a well-defined three-dimensional structure, certain proteins and regions in proteins lack stable three-dimensional structure and take a form of dynamic conformation ensembles. These proteins lack structure along their entire AA chain or in specific regions of the AA chain. These are intrinsically disordered proteins (IDPs) and intrinsically disordered regions (IDRs) [53, 54]. IDPs/IDRs also violate the structure to function paradigm since they can take on multiple different structures and carry out multiple functions. We typically learn their functions directly from the sequence, without the intermediate step of learning their structure. IDPs/IDRs are highly abundant in nature. According to a few recent estimates, 19%, 6%, and 4% of amino acids are disordered in eukaryotes, bacteria, and archaea [3], respectively, between 30% and 50% of eukaryotic proteins (depending on an organism) have at least one long (≥ 30 consecutive amino acids) IDR [2, 32, 55], and between  6 and 17% of proteins encoded by various genomes are fully disordered [56]. Furthermore, 44% of protein-coding genes in human include long disordered regions [57]. Several databases of IDPs/IDRs are available, such as DisProt [7, 58], the largest database of manually curated and functionally annotated IDRs and IDEAL [59], which includes experimentally verified IDPs/IDRs as well as binding partners of IDPs/IDRs. Moreover, IDPs/IDRs can also be found in the Protein Data Bank (PDB) [8] as residues with missing coordinates in the crystal structures and highly flexible residues in the NMR structures [60]. However, these repositories of experimentally annotated intrinsic disorder represent only a small fraction of proteins in nature. The total number of IDPs in IDEAL and DisProt is only 838 and 803, respectively, while the number of currently known proteins that are included in the UniProt [61] resource has already reached 93 million (as of October 2017).

### 2.1.2 Functions of IDRs

The lack of unique structure of IDRs provides them with a set of advantages when compared with the structured proteins and regions. The plasticity of IDRs allows them to efficiently interact with a wide range of different targets including proteins, DNA, various RNAs, small molecules, membranes, etc. [62-65]. IDRs are involved in signaling, regulation, control, storage of small molecules, transcription, translation, assembly of multi-protein complexes, and they often host sites of posttranslational modification (PTMs), [62, 66-70]. These functions nicely complement the functional repertoire of structured proteins that primarily handle functions associated with small molecule binding, transport and catalysis [71]. IDPs are also associated with various human diseases [36] and they were recently suggested to be attractive targets for drug discovery [72]. DisProt (version 6.0.2) lists 37 cellular functions that are assigned to about 1200 IDRs. Table 2.1 lists these functions and provides the count of IDRs for each of these functions. About 80% (719/899) of the functionally characterized IDRs in DisProt concern binding to a variety of partners: proteins, DNAs, RNAs, metals and lipids, etc. The most abundant non-binding function is flexible linker (14%, 122/899). We note that many regions are assigned with multiple functions (37%, 331/899). These regions carry out between two and five functions. Taken together, IDRs carry out a diverse set of molecular functions. So far, we just scratched the surface when it comes to annotating these functions for the millions of experimentally determined and putative disordered regions.

**Table 2.1.** Functional annotations for IDRs.

Functions are parsed from DisProt 6.0.2. The first column shows the function name. The last three columns indicate the number of proteins, regions and residues with a given function. The functions are sorted by the number of regions in the descending order.

| Function name | Can be predicted using current methods | # Proteins | # Regions | # Residues |
|---|---|---|---|---|
| Protein-protein binding | Yes | 265 | 452 | 35909 |
| Substrate/ligand binding | | 90 | 156 | 8260 |
| Flexible linkers/spacers | | 82 | 122 | 3038 |
| Protein-DNA binding | Yes | 70 | 121 | 8239 |
| Intraprotein interaction | Yes | 34 | 72 | 3031 |
| Phosphorylation | | 46 | 67 | 6488 |
| Transactivation | Yes | 33 | 53 | 4077 |
| Metal binding | | 23 | 40 | 3164 |
| Protein-lipid interaction | | 14 | 34 | 1128 |
| Autoregulatory | | 19 | 29 | 2363 |
| Apoptosis Regulation | | 13 | 19 | 1216 |
| Polymerization | | 14 | 18 | 1042 |
| Electron transfer | | 3 | 15 | 585 |
| Nuclear localization | | 12 | 15 | 1474 |
| Protein-tRNA binding | Yes | 8 | 14 | 670 |
| Protein inhibitor | | 9 | 13 | 792 |
| Protein-genomic RNA binding | Yes | 7 | 10 | 558 |
| Fatty acylation | | 6 | 9 | 574 |
| Protein-RNA binding | Yes | 3 | 9 | 574 |
| Structural mortar | | 4 | 9 | 299 |
| Acetylation | | 8 | 8 | 430 |
| Glycosylation | | 8 | 8 | 187 |
| Cofactor/heme binding | | 4 | 7 | 774 |
| Protein-rRNA binding | Yes | 4 | 7 | 1863 |
| Regulation of proteolysis in vivo | | 4 | 6 | 318 |
| Protein-Biocrystal binding | | 5 | 5 | 114 |
| Entropic clock | | 3 | 4 | 318 |
| Methylation | | 3 | 4 | 141 |
| Protein detergent | | 1 | 3 | 114 |
| Protein-mRNA binding | Yes | 2 | 3 | 319 |
| Sulfation | | 1 | 3 | 53 |
| DNA bending | Yes | 2 | 2 | 107 |
| DNA unwinding | Yes | 2 | 2 | 90 |
| Entropic bristle | | 2 | 2 | 696 |
| Entropic spring | | 2 | 2 | 2270 |
| Self-transport through channel | | 2 | 2 | 346 |
| Molecular shield | | 1 | 1 | 143 |

## 2.2 Related work

Motivated by the high levels of abundance and functional importance of IDPs and IDRs, numerous computational methods were developed to predict disorder from protein sequences [9-16, 73, 74]. These methods rely on predictive models that were derived from limited and challenging to acquire experimental annotations of IDPs and IDRs. They are used to efficiently and accurately find disordered proteins and regions for the millions of proteins that lack these annotations.

### 2.2.1 Predictors of intrinsic disorder

Computational methods that predict intrinsic disorder provide a viable high throughput alternative to investigate disorder, compared to the low throughput of the experimental annotation approaches. Computational prediction of intrinsic disorder is a mature research area. As mentioned in the introduction, many methods for the prediction of intrinsic disorder were already developed and are widely used and cited [9, 15, 16, 73-78]. These predictors take the amino acid sequences of a protein of interest as the input, and output propensity scores for each residue that represent their likelihood of being disordered. These propensities can be turned into binary predictions where a given residue is predicted as either ordered (structured) or disordered. The architectures of these predictors vary widely and take the form of:

1) Scoring functions - the propensity of disorder is calculated from scoring functions utilizing properties of the input amino acid related to the formation of disordered/ordered regions. Examples methods include NORSP [79], GlobPlot [80] and IUPred [81, 82].

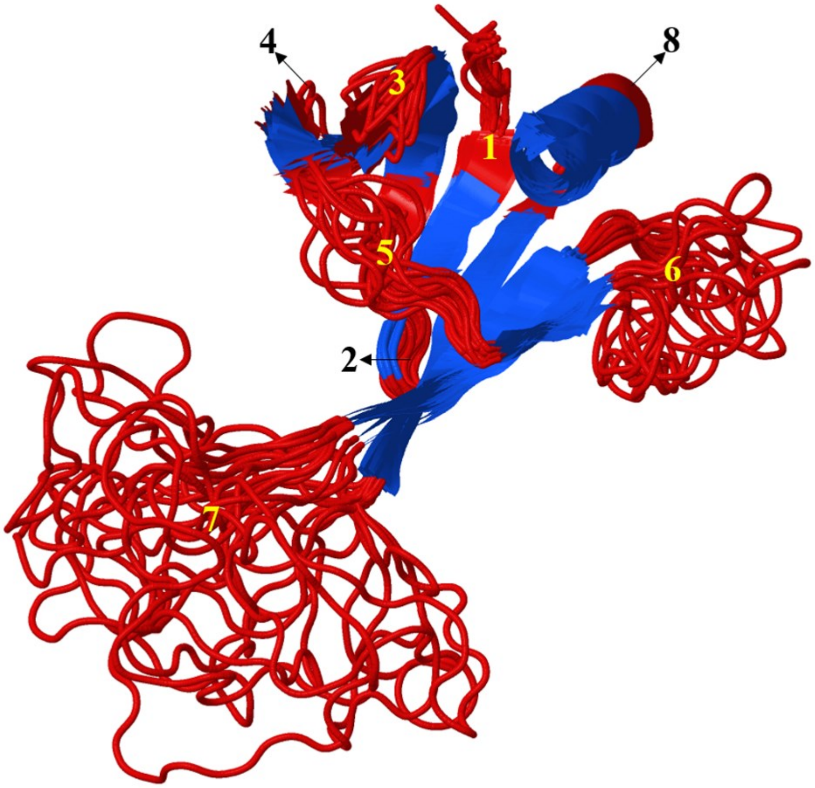
2) Machine learning models - the propensity of disorder is output from machine learning classifiers (such as neural network, support vector machine, and regression) that use input features computed from the sequence and sequence-derived characteristics of proteins as their inputs. Examples are DisEMBL [83], DISOPRED [84, 85], and a family of VLS predictors [86, 87].

3) Meta predictors - the propensity of disorder is computed based on a consensus of predictions generated by multiple base predictors (scoring functions and/or machine

learning models). These methods include MFDp [88-90], MetaDisorder [91] and PONDR-FIT [92].

4) Hybrid predictors - These predictors combine the abovementioned machine learning approach with structural modelling, typically using template-based structure predictions. Examples are PrDOS [93] and Disoclust3 [94].

We illustrate predictions of intrinsic disorder and contrast these predictions with the native annotations of disorder using the ICln protein (Figure 2.1). ICln is a chloride channel that regulates several cellular processes including membrane ion transport and RNA splicing. Structure of ICln, which was solved using NMR, is composed of several superimposed conformations. The regions colored in blue converge to the same confirmation and constitute structured regions. The disordered regions that are colored in red form an ensemble of diverse conformations. The annotation of disordered regions was collected from DisProt version 6.02 and is shown below the image in ("Native Dis" line). The figure includes eight IDRs which are numbered in the structure as they appear along the amino acid sequence. The structure excludes parts of both termini of this protein, including residues 1 to 18 and residues 134 to 235, which are disordered. The bottom part of Figure 2.1 includes predictions of three methods: DISOPRED version 3 (machine-learning method) [84, 85], MFDp (meta method) [88-90], and PrDOS (hybrid method) [93]. These three methods ranked the top 3 among participants of the CASP10 competition [12], a world-wide event where independent assessors evaluate predictions on a blind set of proteins. CASP10 was the last event that included assessment of the prediction of the intrinsic disorder. Both binary and numeric scores are included where the numeric scores that range between 0 and 1 are represented by the first digit after the decimal point. A given residue is predicted as disordered if its predicted numeric propensity is high: ≥ 0.5 for PrDOS and DISOPRED, and ≥ 0.37 for MFDp; otherwise it is predicted as structured.



```
Residue number       1        10        20        30        40        50        60        70        80        90        100       110       120
Sequence             MSFLKSFPPPGSAEGLRQQQPETEAVLNGKGLGTGTLYIAESRLSWLDGSGLGFSLEYPTISLHAVSRDLNAYPREHLYVMVNAKFGEESKESVAEEEDSDDDVEPIAEFRFVPSDKSALEAMFTAM
Native Dis           1111111111111111111111100001111111000000111000011111000111111100011111111111000000011111111111111111111111000000011111000000000
PrDOS                1111111111111110000000000000000000000000000000000000000000000000000000000000000000000011111111111111111111110000000000000000000
PrDOS propensity     9988777777766554333333233333222222111111100111122222221110000111122222222221111112234556777888888888877665544333322222222222222
DISOPRED3            1111111111111111000000000000000000000000000000000000000000000000000000000000000000000001111111111111111000000000000000000000000
DISOPRED3 propensity 7655666876777676443322112110000000000000000000000000000000000010011110001000000000001245778899999999876421000000000000000000000
MFDp                 1111111111111111111111111111110000000000000000000000000000000000000000000000000000111111111111111111111111111000000000000011111
MFDp propensity      7778988877776666555544444444333332222222111111111111112111111111111222222222222233445578999999999999987755443333222223333375545

Residue number         130       140       150       160       170       180       190       200       210       220       230  235
Sequence             CECQALHPDPEDEDSDDYDGEEYDVEAHEQGQGDIPTFYTYEEGLSHLTAEGQATLERLEGMLSQSVSSQYNMAGVRTEDSTRDYEDGMEVDTTPTVAGQFEDADVDH
Native Dis           00000011111111111111111111111111111111111x111111111111111111111111111111111111111111111111111111111111111111
PrDOS                000000001111111111111111100110000000000000000000000000000000000000111111111111111111111100000001111111111111
PrDOS propensity     223333445667777877666555544554444444433333333333333333333333333334445555555555566666665555444444455566677889 9
DISOPRED3            000000000011111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111
DISOPRED3 propensity 000000012356777878888877778788888888878777777777666666555655556677889999999999999999999999999999999999988899
MFDp                 111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111
MFDp propensity      566778889999999999999999999998988988888788767767777776677777788888899999999999999999999999999999999999999999
```

**Figure 2.1.** Native intrinsic disorder and putative disorder for the ICln protein.

PDB ID: 1ZYI; DisProt ID: DP000717. The top portion of the figure is a cartoon view of multiple superimposed NMR structures of this protein taken from PDB. Ordered regions where all structures converge to the same conformation and disordered regions that form conformational ensembles are colored in blue and red, respectively. The eight disordered regions are numbered from 1 to 8, and they correspond to eight underlined disorder regions in the "Native Dis" line. The bottom part shows native disorder annotations collected from DisProt along with putative disorder annotations generated with PrDOS, DISOPRED3 and MFDp methods. The first line shows residue number which is followed by the amino acid sequence. The third line shows native annotation of disorder where 0 denotes structured residues, 1 denotes disordered residues, and x denotes a residue that lacks annotation. The following six lines show binary predictions and propensity scores. The propensity that ranges between 0 and 1 is represented by the first digit after the decimal point.

To interpret the results produced by the computational predictors, users should first analyze the binary predictions in order to extract the corresponding putative IDRs and structured regions. Next, each predicted IDRs should be assessed using the numeric propensities. Residues that have high scores are more likely to be disordered and the corresponding predictions are more likely to be accurate. Users can also analyze the scores of all residues in a given putative IDR, which is annotated based on binary predictions, to quantify the likelihood of this entire region to be correctly identified. On the other hand, low scores can be used to identify structured residues and regions. The predictions for residues with scores close to 5 for PrDOS and DISOPRED, and close to 4 for MFDp (these values are used to convert the propensity into the binary prediction) are arguably less accurate than the predictions with either high or low scores. We also recommend that, if possible, multiple methods should be used and the users should rely on a consensus-based prediction. In other words, IDRs and disordered residues predicted by multiple methods are more likely to be correct compared with predictions that disagree between different methods. The favorable predictive performance of a consensus-based approach was shown empirically in a few recent studies [95, 96].

The predictors of intrinsic disorder have been periodically empirically assessed. These studies were published both as comparative reviews [15, 75, 77] and as a part of the biannual world-wide Critical Assessment of protein Structure Prediction (CASP) (http://predictioncenter.org/). The latest CASP10 event that included the assessment of predictions of intrinsic disorder revealed that the best predictor secure AUC = 0.9 and MCC = 0.53 [17]. Given that AUC ranges between 0.5 and 1 (higher value corresponds to stronger predictive performance) and MCC is a correlation and so higher positive values indicate higher correlation between predictions and the native (true) annotations of disorder, these results show that prediction of disorder is accurate. Moreover, most predictors are available to the end users as a convenient to use webservers, and some of them provide predictions in high-throughput. The latter means that they can be used to annotate disorder on a full-proteome scale. In the nutshell, the end users can nowadays use high quality, easy-to-access and fast predictors to annotate IDRs.

### 2.2.2 Databases of IDPs/IDRs

Several databases were developed to store both experimental and putative annotations of disorder. These databases are important both for the end users, as a convenient access point to both experimental and putative annotations, and for the developers of computational predictors, as a source of experimental annotations for the development and benchmarking of these methods. The first and largest repository of the experimentally verified IDPs and IDRs is DisProt [7, 58]. It contains manually curated IDRs together with the annotations of their functions, whenever available. The latest version of DisProt (7.0) contains 2167 IDRs from 803 protein chains, compared to 290 IDRs from 179 proteins from the first release of that database. Another source of experimentally verified IDPs is the IDEAL database [97, 98]. This database originally had 153 annotated proteins and has grown to 838 proteins in its latest version. Although these repositories of the experimental annotations of disorder provide invaluable information to investigate disorder and build computational tools, they represent only a small fraction of the sequences in nature.

Given that various comparative reviews suggest that predictors of intrinsic disorder are relatively accurate [15, 17, 75, 77], these predictions are used to guide experimental studies of disorder and to address practical problems in other areas, such as targets selection in structural genomics [99]. They were also used to analyze prevalence and functional characteristics of disorder on large scale across functionally related proteins [22, 27, 70] and in whole proteomes [3, 24, 25, 72, 100]. To this end, several databases of the putative annotations of IDPs and IDRs were developed to ease access to this information for the end users. Given that these resources provide access to putative disorder for large sets of proteins, they include results generated by high-throughput predictors of intrinsic disorder. DICHOT [101] is the first such database that provides predictions of intrinsic disorder for the human proteome. It includes 20,333 protein chains collected from the Swiss-Prot database [102]. DICHOT is superseded by the two more recent and much larger databases: MobiDB [34, 103] and $D^2P^2$ [35]. MobiDB offers access the putative disorder generated by ten predictors, and it also combines these ten predictions into one consensus. Moreover, MobiDB includes experimental annotations of disorder collected from DisProt and PDB, the latter is based on both X-ray and NMR structures. Version 2.0 of MobiDB covers over

80.37 million chains, which were obtained from the UniProtKB [61]. Importantly, these putative annotations of disorder are also cross-referenced in UniProt [61]. $D^2P^2$ is the second large repository of predicted annotations of intrinsic disorder. It contains annotations generated with nine predictors. It also links to the experimental annotations of disorder from DisProt and IDEAL and includes putative annotations of disordered protein binding regions computed with ANCHOR [49, 104]. The current version of $D^2P^2$ contains annotations for 10.43 million proteins from 1,765 proteomes across all kingdoms of life. The main difference between MobiDB and $D^2P^2$ is that the former provides annotations for arguably largest possible set of currently known proteins, while the latter provides the annotations for all complete proteomes.

### 2.2.3 Predictors of functions of IDRs

IDPs and IDRs are involved in a wide repertoire of cellular functions. The most common way to annotate functions of proteins is to use sequence alignment. This approach is based on the premise that similar sequences tend to have similar structures and thus similar functions. A reliable alignment requires a certain sequence similarity, typically greater than 50% and preferably greater than 80%, for the corresponding prediction to be accurate. By aligning the amino acid sequence of interest against sequences with known annotations we can potentially find segments with high similarity and transfer the corresponding annotations onto the sequence of interest. The arguably most popular sequence alignment tool is BLAST (Basic Local Alignment Search Tool) [105]. BLAST allows users to perform pairwise alignment of the query sequence against sequences in large databases, such as the non-redundant proteins sequences (NR) [106], NCBI Protein Reference Sequences (refseq) [107] and Protein Data Bank (PDB) [8] to find proteins and their regions that are similar to the query protein. There are also other types of alignment tools. SWalign [108] allows users to conduct pairwise alignment for short segments of proteins based on their local sequence similarity. Clustal Omega [109] is a multiple sequence alignment tool that allows users to align three or more sequences of similar lengths. However, while being largely successful to annotate functions of structured proteins, alignment is not a feasible approach to annotate functions of IDRs. In a recent work that characterizes intrinsic disorder in ribosomal proteins, only 12% IDRs were functionally annotated through alignment [110]. In another study that analyses proteins involved in the cell death cycle this approach only allowed to functionally annotate 2.2% of the IDRs [22]. The low coverage is due to a relatively low number of disordered regions that are functionally annotated. To this end, data-driven computational predictors of functions of intrinsic disorder that work in the absence of sequence similarity are needed.

Research towards computational prediction of functions of proteins has accelerated since the introduction of The Critical Assessment of protein Function Annotation (CAFA) competition in 2010 [111]. CAFA is an ongoing community challenge that aims to provide large-scale assessment of computational methods for the prediction of functions of proteins. Methods submitted to CAFA provide predictions of functions described in Gene Ontology (GO) terms at the whole protein level and they address predictions of all proteins. In other words, these methods predict functions carried by both structured and disordered proteins, often without knowledge of these structural details. In contrast, we specifically target the disordered proteins and we focus on the residue level annotations, which are essential to predict functional IDRs.

In recent years progress has been made to develop methods that predict functions of disordered regions from the protein sequences. In contrast to the predictors of intrinsic disorder, these methods find a subset of IDRs that carry out a specific function. The current predictors of functions of disorder address primarily binding-related functions that include interactions of IDRs with proteins, DNAs and RNAs. This is motivated by an observation that these binding-related functions are the most prevalent functions carried out by IDRs. Based on the experimental data from DisProt, 74% of the over 1000 functionally annotated IDRs in DisProt interact with proteins, DNAs, RNAs, metals and lipids. The protein-protein binding is the most populated function, with over 450 annotated IDRs in DisProt.

The predictors of the disordered protein binding regions are categorized into three classes. They include methods that predict generic disordered protein binding regions and methods that focus on two specific types of protein binding regions: molecular recognition features (MoRFs) and short linear sequence motifs (SLiMs). The three methods that find generic disordered protein binding regions are PepBindPred [112], ANCHOR [49, 104], and DisoRDPbind [51]. There are several predictors of MoRFs, defined as short regions that undergo disorder to order transition upon binding to protein partners, which include alpha-MoRFpred [113, 114], MoRFpred [50], MFSPSSMpred [115], MoRFChiBi [116, 117], fMoRFpred [118], retro-MoRF [119] and DISOPRED3 [120]. Finally, SLiMs can be predicted with the help of the SLiMpred method [121]. So far only one predictor, DisoRDPbind [51], that considers binding to other molecules, in particular DNA and RNA, was developed. This method combines three predictive models that provide putative annotations of the disordered protein-, DNA- and RNA-binding residues. Table 2.2 lists the 11 predictors of the various cellular functions of disorder.

**Table 2.2.** Predictors of functions of disorder.

SVM: support vector machine; LR: logistic regression; NN: neural network; SA: sequence alignment; DISP: disorder prediction; SF: scoring function

| Method | Year last published | Ref. | Prediction target | Predictive model |
|---|---|---|---|---|
| fMoRFpred | 2015 | [118] | protein binding | SVM |
| DISOPRED3 | 2015 | [120] | protein binding | SVM |
| MoRFCHiBi | 2015 | [116, 117] | protein binding | SVM |
| disoRDPbind | 2015 | [51] | protein, RNA, DNA binding | LR |
| PepBindPred | 2013 | [112] | protein binding | NN |
| MFSPSSMpred | 2013 | [115] | protein binding | SVM |
| MoRFpred | 2012 | [50] | protein binding | SVM |
| SLiMPred | 2012 | [121] | protein-binding | NN |
| retro-MoRFs | 2010 | [119] | protein-binding | SA + DISP |
| ANCHOR | 2009 | [49, 104] | protein binding | SF |
| alpha-MoRFpred | 2007 | [113, 114] | protein binding | NN |

Compared with the prediction of IDRs and IDPs, computational prediction of functions of disorder is in early stages. These predictors primarily focus on the binding-related functions, such as disordered protein-protein, protein-RNA and protein-DNA binding. Although 11 of these methods were already released, the development of models that address prediction of other functions of disorder remains an outstanding and pressing challenge. In table 2.1 we list 37 functions of IDR defined by DisProt, only 11 of them can be predicted with the currently available methods.

## 2.3 Background on computational models

### 2.3.1 Background related to the design of computational predictors

Computational prediction of functions of IDRs is a classification problem. In other words, each residue in the input protein sequence is classified as functional (has a specific function) or non-functional (does not have a specific function). This classification is performed in two steps. First, a given input amino acids is represented by a set of numerical features. These features describe various, relevant physiochemical and putative structural properties of the predicted residue and its neighbouring residues. Second, these features are input into a classification model that outputs propensity for this residue to carry out a given function. The second step could be handled by a machine-learning classifier such as Logistic Regression [122], Naive Bayes [123], Decision Trees [124] or Random Forest [125]. The formulation and selection of features, and the parametrization of the predictive model generated using these classifiers is done by cross validation on a training dataset. The final design is validated and compared to alternative solutions on an independent (different from the training dataset) test dataset. To this end, the available experimental data (proteins) that is used to build the model is divided into two parts, the training dataset which is further divided into several folds for the cross validation, and the test dataset. In the cross validation, the training dataset is divided into equally sized (in terms of number of proteins) $x$ subsets (folds), and in the $i^{\text{th}}$ ($1 \leq i \leq x$) fold of the cross validation, $x$-1 subsets are used to train the model, and the remaining $i^{\text{th}}$ subset is used as test set to evaluate the trained model. The result of the cross validation is reported as the aggregated or averaged result of the $x$ folds of tests. The design of computational predictive models usually consists of the following steps:

1) Data preparation. As mentioned above, the proteins are divided into the training and test datasets. In many cases proteins may differ from each other by only a few amino acids in their sequences, particularly if that is the same protein in different species. Inclusion of similar sequences in training and test sets may result in an over-estimated predictive quality. Moreover, test proteins that are similar to the training proteins can be typically accurately predicted using sequence alignment protocols, such as BLAST [105]. The

alignment can be used to transfer annotations from the similar proteins in the training dataset. Since we aim to develop predictive models for proteins for which alignment does not work (we shows in section 2.2.3 that alignment does not work for annotation of functions of IDRs), we train our model on proteins that are dissimilar from the test proteins. To this end, the proteins in the test and training datasets as well as proteins in the cross validation folds in the training dataset are set to be dissimilar. As a rule of thumb, sequences are considered as similar if they share more than 25% similarity (proportion of same amino acids on same positions in the aligned sequences); alignment fails at this levels of similarity [126]. As a result, we ensure that sequences in the test set share no more than 25% sequence similarity with any sequence in the training set, and this rule also applies to each fold subset in the cross validation.

2) Sequence representation and feature generation. Each residue in the input protein sequence is represented by a set of numeric features (descriptors). Depending on the target of the prediction, these characteristics can be derived from a variety of sources such as physicochemical properties of amino acids, evolutionary profiles, and predictions from the sequence secondary structure, solvent accessibility and intrinsic disorder. The features are calculated from this information for the predicted residue and its neighbors (a window in the sequence covering the residue being predicted and residues close to this residue). The inclusion of nearby residues is motivated by the fact that neighbouring residues influence the function and structure of the predicted residue. The use of window is a popular approach in protein disorder predictions [120, 127], structure predictions [128, 129] and in existing methods for the prediction of the intrinsic disorder function [50, 51].

3) Feature selection and model construction. Typically a large set of features is generated. Among them there could be irrelevant features (low correlation with the class label), and redundant features (features that are mutually correlated). Feature selection removes such irrelevant and redundant features. This simplifies the predictive model (fewer features must be computed to run predictions) and reduces runtime needed to compute the model from the data. In case of some models, e.g. Logistic Regression that should not be used on data with collinear features, this could also improve the predictive quality. The predictive model maps features into the outcome. Different sets of features are

fed into a classification model, and each set of features and parameters of the classifier can result in a different classification result. We optimize the classifier through cross validation on the training dataset by varying the feature sets and parameters of the classifier. We choose the best combination of feature sets and parameters that results in an optimal cross validation result, according to a specific criterion that we use to measure predictive performance. We can also compare results from different classifiers. The chosen features and classifier together with its parameters are adopted as the predictive model and is applied to predict proteins in the test dataset.

4) Model validation. The predictive quality of the classification model is measured by comparing the prediction generated by this model with the native annotations on a given test dataset. The validation must be performed out of sample (using data that was not used to build the model) to assure that the model does not over-fit (too closely mimic) the training dataset. In our research, we further ensure that test and training proteins are highly dissimilar (< 25% similarity). Cross validation follows the out-of-sample rule because in each fold the $i^{\text{th}}$ set is not used for the training. The final validation on the independent test dataset also follows this rule because the test dataset is never used to build the model. The measurement of the predictive quality can be quantified using different measures. Section 2.3.2 discusses these measures.

### 2.3.2 Evaluation of predictive performance

The prediction is in the format of a numeric score between 0 and 1 that represents propensity for a given residue to have a given function (functional residue). The numeric score can be also converted into a binary prediction using a threshold, i.e., a residue is predicted as a functional or non-functional residue. More precisely, residue with a putative score greater than or equal to a given threshold is predicted as a functional residue, otherwise it is predicted as a non-functional residue. We assess the predictive quality of the putative propensities with the receiver operating characteristic (ROC) curve and the area under ROC (AUC). To plot the ROC curves and quantify AUC values, we calculate the true-positive rates (TPRs) and false-positive rates (FPRs) by comparing predictions with native annotations at different thresholds imposed on the predicted scores. TPR and FPR are defined as:

$$\text{TPR} = \frac{TP}{TP + FN} = \frac{TP}{number\ of\ all\ native\ functional\ residues}$$

$$\text{FPR} = \frac{FP}{FP + TN} = \frac{FP}{number\ of\ all\ native\ non - functional\ residues}$$

where TP is the number of true positives (correctly predicted functional residues), FN is the number of false negatives (functional residues that are predicted as non-functional), FP is the number of false positive (non-functional residues that are predicted as functional), and TN is the number of true negatives (correctly predicted non-functional residues). Given TPR and FPR values generated at different thresholds ranging from 0 and 1, we plot the ROC curve and calculate the corresponding AUC value.

Motivated by the fact that a large number of the residues are non-functional, we perform assessment of the propensities when the false positive rate (FPR) is low, e.g. at or below 5% or 10%. This ensures that the corresponding predictions include functional residues which are likely to be correctly predicted, i.e., only a small fraction of these predictions are false positives. Correspondingly, we calculate AUC$_{lowFPR}$ that covers the low range of FPR values. Since AUC$_{lowFPR}$ are rather small and difficult to assess directly, we compute AUC$_{ratio}$ = AUC$_{lowFPR}$/AUC$_{random\_lowFPR}$, where AUC$_{lowFPR}$ is divided by the AUC of a random predictor (for which FPR always equals to TPR) in the same FPR range. This ratio quantifies the rate of improvement over a random predictor, i.e., ratio > 1 means that a given method is better than random and ratio = 2 means that this method is twice better than random.

To evaluate binary predictions, we use accuracy, precision, sensitivity and Matthews Correlation Coefficient (MCC), which are defined as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{TP + TN}{number\ of\ all\ residues}$$

$$\text{Precision} = \frac{TP}{TP + FP} = \frac{TP}{number\ of\ all\ predicted\ functional\ residues}$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} = \frac{TP}{number\ of\ all\ native\ functional\ residues}$$

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

The AUC ranges between 0.5 and 1 where 0.5 denotes random prediction and 1 denotes perfect prediction (all functional residues and non-functional residues are predicted correctly). Accuracy, precision and sensitivity range between 0 and 1 where 0 denotes that no residues were predicted correctly and 1 denotes perfect prediction. MCC ranges between -1 and 1, where -1 denotes that the prediction is inverted (all functional residues are predicted as non-functional residues and vice versa), 0 denotes a random result and 1 denotes the perfect prediction.

### 2.3.3 Statistical tests of significance

AUC, AUC*ratio*, accuracy, precision, sensitivity and MCC evaluate the performance of a predictive model, but they do not assess whether the performance of a given model is consistently better when compared with other methods. To this end, we use statistical tests to evaluate whether differences between two sets of numeric results are significant. We use paired difference tests to compare the predictive performance of two models based on bootstrapping results on the test dataset. To do this we randomly select 50% proteins from the test dataset (or test dataset of each cross validation fold), and evaluate the prediction criterion (e.g. AUC) on these proteins by two predictive models. We repeat this procedure for *n* times, and for each predictive model we have *n* results. We place the *n* results from two models side by side, and have *n* pairs of results. We use the Student's paired *t*-test if the *n* results from both two models follow normal distributions, otherwise we use Wilcoxon signed-rank test. We use Anderson-Darling test to verify if a sample of *n* results follow normal distribution.

Student's paired *t*-test [130] evaluates two groups of data (e.g., two groups of AUC values), and determines whether their mean values are significantly different. Suppose we have two groups of data, $X_1$ and $X_2$:

$$X_1 = \begin{bmatrix} x_{1,1} \\ x_{1,2} \\ \vdots \\ x_{1,n} \end{bmatrix}, \quad X_2 = \begin{bmatrix} x_{2,1} \\ x_{2,2} \\ \vdots \\ x_{2,n} \end{bmatrix}$$

where $x_{1,i}$ and $x_{2,i}$ are matched pairs. Student's *t*-test calculates the *t*-value as follows:

$$t = \frac{\bar{X}_D}{S_D/\sqrt{n}} = \frac{\sum_{i=1}^{n}(x_{1,i} - x_{2,i})/n}{S_D/\sqrt{n}}$$

where $\bar{X}_D$ is the average difference of all pairs of $X_1$ and $X_2$, $S_D$ is the standard deviation of these differences. This *t*-value is calculated following Student's *t* distribution with *n* samples of *n*-1 degree of freedom. The *p*-value is determined by looking up the *t*-table using the *t*-value and the corresponding freedom degree. The *p*-value is the probability of getting our observation (for example, the mean AUC values from two models), or getting values with even greater evidence against the null hypothesis, given the null hypothesis is true. In our case, the null hypothesis is the mean values of $X_1$ and $X_2$ are equal. A low *p*-value (typically < 0.05) suggests that the data provides enough evidence to reject the null hypothesis, and thus suggests that the difference between $X_1$ and $X_2$ is statistically significant.

Wilcoxon signed-rank test [131] is an alternative to student's paired *t*-test when $X_1$ or $X_2$ does not follow normal distribution. It determines the difference between the medians of $X_1$ and $X_2$. This test calculates the absolute value of the sum of the signed ranks, where the ranks are the ranks of absolute differences of pairs $x_{1,i}$ and $x_{2,i}$:

$$W = \left| \sum_{i=1}^{N_r} [sgn(x_{2,i} - x_{1,i}) \cdot R_i] \right|$$

where *sgn* is the sign function so *sign*(*x*) = 1, 0, -1 when $x > 0$, = 0 and < 0, respectively. $R_i$ is the rank of pair $x_{1,i}$ and $x_{2,i}$, which is ranked by the absolute difference between $x_{1,i}$ and $x_{2,i}$. $N_r$ is the number of pairs with non-zero differences. A *p*-value can be looked up from the Wilcoxon reference table with the *W* value. If the *p*-value is smaller than a threshold (typically 0.05), we reject the null hypothesis. In our case the null hypothesis is the median values of $X_1$ and $X_2$ are equal. By rejecting null hypothesis, we can declare that the difference between $X_1$ and $X_2$ is statistically significant.

Anderson-Darling test [132] is a statistical test that checks whether a set of data follows a certain probability distribution. Here we use it to determine if $X_1$ and $X_2$ follows normal distribution. We calculate $A^2$ by the following equation:

$$A^2 = -n - \frac{1}{n}\sum_{i=1}^{n}(2i-1)\big(\ln \mathrm{F}(Y_i) + \ln\big(1 - \mathrm{F}(Y_{n+1-1i})\big)\big), Y_i = \frac{X_i - \bar{X}}{\sigma}$$

where $\bar{X}$ and $\sigma$ is the mean and standard deviation of variable $X$, respectively, $\mathrm{F}(Y_i)$ is a cumulative distribution function of $Y_i$ for normal distribution, and $n$ is the number of samples of variable $X$. $A$ is compared against critical values from the table for normal distribution. If the $p$-value found in the table is bigger than a threshold (typically 0.05), then we accept the null hypothesis that the data are draw from a normal distribution, otherwise we reject the null hypothesis.

## Chapter 3

# Characterization of functions of IDRs in human dengue virus

Viral proteomes are typically polyproteins which are proteolytically processed into a relatively small (compared to eukaryotic or bacterial proteomes) number of proteins. We focus on the dengue viruses (DENV). A single DENV proteome is a polyprotein that consists of approximately 3390 residues that after proteolytic processing codes 12 proteins. Overall, research shows that IDRs/IDPs are very common in viral proteomes [2, 133]. We study whether this is also true for DENV. The high levels of intrinsic disorder allow many viral proteins to fulfill their biological roles, as each of these proteins has to carry out multiple functions given the small overall size of the viral proteomes (number of proteins in the viral proteomes is very small). The functions of these IDRs are typically unknown and our goal was to characterize them. We use predictors of disorder and disorder functions and also alignment to annotate these putative IDRs and their functions. We focus on the methods that were used to annotate functions of IDRs in DENV and their ability to provide functional annotations.

## 3.1 Materials and methods

### 3.1.1 Data collection

We collect the complete proteome of the human dengue virus from UniProt [134] with “dengue virus” as the query word for the organism. We only include reviewed entries to ensure that the corresponding sequences were curated by a human and have high quality functional annotations. We obtain 28 polyproteins including eight fragments that are excluded. The remaining 20 polyproteins cover all four serotypes (variations that have different surface antigens) of the dengue virus. They include three polyproteins from type

1 (UniProt ID: P33478, P27909, P17763), seven from type 2 (UniProt ID: P29990, P29991, P14337, P07564, P14340, Q9WDA6, P12823), five from type 3 (UniProt ID: Q99D35, Q5UB51, Q6YMS3, P27915, Q6YMS4) and five from type 4 (UniProt ID: P09866, Q2YHF2, Q58HT7, Q5UCB8, Q2YHF0). The lengths of the 20 polyproteins ranges from 3387 to 3396 residues. Each polyprotein consists of 12 protein chains that are extracted based on the annotated cleavage sites. We obtain in total 240 protein chains from the 20 polyproteins across the four dengue serotypes. The length of these protein chains ranges from 14 to 904 residues.

### 3.1.2 Annotations of disorder and functions of disorder

The putative intrinsically disordered residues are generated with the MFDp webserver. MFDp is a sophisticated consensus predictor that combines disorder predictions generated by IUPred [81], DISOclust [135] and DISOPRED2 [136] methods, sequence profiles and predictions of secondary structure by PSIPRED [137], relative solvent accessibility and backbone dihedral torsion angles by Real-SPINE3 [138], B-factors by PROFbval [139], and globular domains by IUPred. The MFDp predictor is characterized by high predictive performance with AUC > 0.81 based on multiple benchmark tests [15, 127]. We utilize binary annotations from MFDp to annotate residues in DENV proteins as structured or intrinsically disordered.

We use local pairwise alignment SWalign [108] to annotate functions of IDRs. We choose SWalign because it is suitable for short segment and IDRs are such short regions. We align each IDR predicted by MFDp (query segment) with a set of 775 disordered segments collected from DisProt 6.0.2 that have functional annotations (reference segments). We copy the functional annotation from a reference segment onto the query segment if their similarity is greater than 80%. We use 80% because we align short segments rather than full sequences, which requires higher similarity to provide reliable alignment. The similarity is defined as the number of identical residues in the local alignment divided by the length of the local alignment or the length of the shorter of the two segments being aligned, whichever is longer. Some of the IDRs could be annotated with more than one function. The same protocol was also previously used in ref. [22] and ref. [21] to analyze roles of IDRs in the programmed cell death and ribosomal proteins, respectively.

We use the available predictor MoRFpred [50] to predict putative annotations of the protein-protein binding, i.e., regions that carry molecular recognition features (MoRFs). MoRFs are short (5 to 25 consecutive amino acids) protein binding regions located within longer IDRs that undergo coupled folding and binding, i.e., disorder-to-order transition, upon binding [140, 141]. MoRFpred is characterized by state-of-the-art design that combines information from pairwise alignments, sequence profiles, and predictions of disorder from IUPred [81], DISOPRED2 [136], DISOclust [135] and MFDp [127], solvent accessibility by Real-SPINE3 [138], and B-factors by PROFbval [139]. We utilize binary predictions from MoRFpred and only consider MoRF region only if it is at least 5 residues long; shorter putative regions which have higher propensity to be spurious were removed. Both MFDp and MoRFpred predict from sequences of individual proteins and thus we first fragment the polyproteins into proteins, predict for each protein, and finally combine these predictions together to annotate the full polyproteins.

### 3.1.3 Other functional and structural annotations for DENV

Annotations for cleavage sites (CLV), transmembrane region (Trans), intramembrane region (Intra), topological cytoplasmic-, extracellular- and luminal- domain (Topo-cy, Topo-ex, Topo-lu) and functional sites (Func) are parsed from XML format files downloaded from the UniProt database [134]. They were collected for the 20 polyproteins in the dengue viruses. The annotation for functional sites is a union set of regions of interest, active sites, binding sites, other sites (except cleavage sites) and nucleotide binding regions. If any of these annotations for a given residue is true, the annotation for functional sites is set to be true.

Eukaryotic linear motif ( ELMs) are short, usually between 3 and 11 residues in length, conserved functional sequence motifs [142] which are often found in the IDRs [143]. We use these motifs to functionally annotate the disordered regions. Annotations for six types of ELMs are parsed from the HTML files generated by the ELM motif search algorithm [143], after filtering the results by globular domain, structure and context. They include motifs that serve as proteolytic cleavage sites (ELM_CLV); post translational modification sites (ELM_MOD); motifs for recognition and targeting to subcellular compartments (ELM_TRG); generic ligand binding motifs (ELM_LIG); degron motifs that are involved in polyubiquitylation and targeting the protein to the proteasome for degradation (ELM_DEG); and docking motifs that correspond to site of interactions with modifying enzyme that are distinct from active sites (ELM_DOC).

## 3.2 Results

### 3.2.1 Functional analysis of IDRs

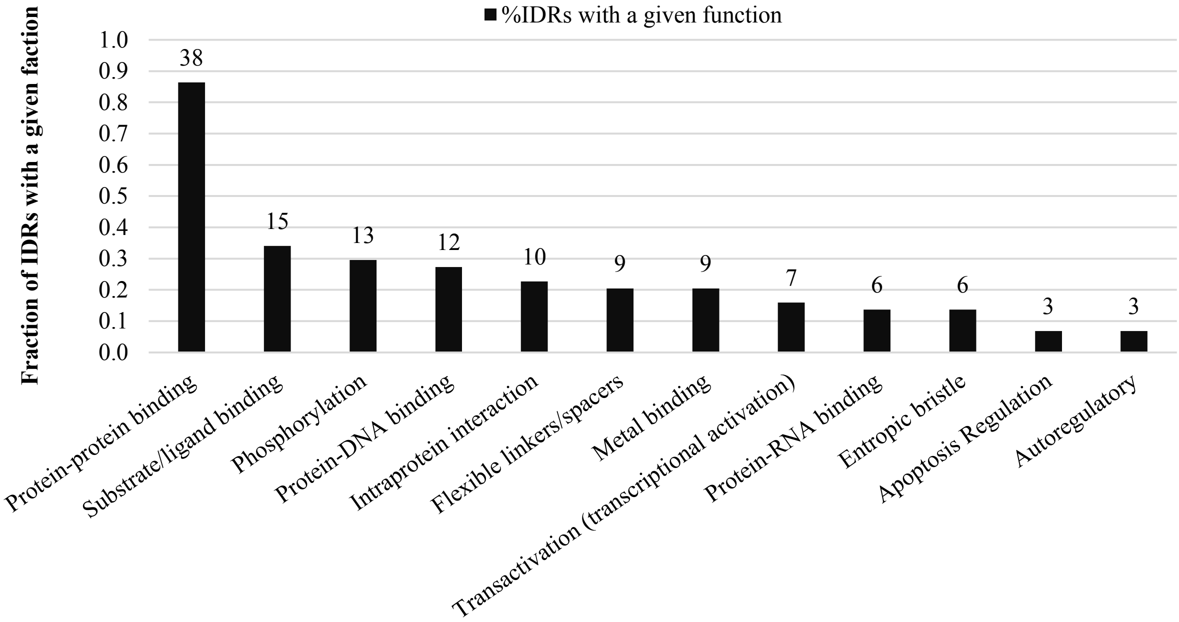


**Figure 3.1.** Functional annotations of putative IDRs among 20 DENV polyproteins.

The annotations are obtained by pairwise alignment with SWalign. Values above the bars indicate number of IDRs that carry out a given function.

By using the local pairwise alignment with SWalign, we annotate 18 functions for 44 putative IDRs from the set of 240 DENV proteins. After eliminating functions that are predicted for less than 3 IDRs, which have a higher propensity of being spurious, we find 12 distinct functions carried out by the 44 IDRs. Figure 3.1 summarizes these functions. It shows fractions of the 44 putative IDRs having a given function (bars). Our analysis revealed that the predominant function of the DENV IDRs is protein-protein binding. DENV IDRs are also involved in various protein-ligand binding events such as interactions of viral proteins with nucleic acids, metals, and other small molecules. Also, some IDRs serve as the flexible linkers. The distribution of functions of DENV IDRs follows a similar trend compared to the functions of IDRs from DisProt that are summarized in table 2.1.

### 3.2.2 Relation of IDRs and MoRFs with other functional and structural annotations

| **Function** | **Enrichment in IDRs** | | | | | **Enrichment in MoRFs** | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | **all** | 1 | 2 | 3 | 4 | **all** | 1 | 2 | 3 | 4 |
| CLV | **6.2** | 6.9 | 7.0 | 5.1 | 6.2 | **1.7** | 0.0 | 3.1 | 2.6 | 0.0 |
| Trans | **1.1** | 1.2 | 1.4 | 0.9 | 0.8 | **2.0** | 1.7 | 2.1 | 2.5 | 1.6 |
| Intra | **0.0** | 0.0 | 0.0 | 0.0 | 0.0 | **0.8** | 0.9 | 0.8 | 0.0 | 1.7 |
| Topo-cy | **1.4** | 1.5 | 1.3 | 1.5 | 1.5 | **1.2** | 1.3 | 1.3 | 1.1 | 1.3 |
| Topo-ex | **0.3** | 0.1 | 0.5 | 0.1 | 0.2 | **0.3** | 0.4 | 0.1 | 0.4 | 0.3 |
| Topo-lu | **0.7** | 0.5 | 0.8 | 0.9 | 0.4 | **0.5** | 0.0 | 0.0 | 1.8 | 0.0 |
| Func | **0.5** | 1.0 | 0.0 | 0.7 | 0.7 | **0.6** | 1.2 | 0.1 | 0.8 | 0.7 |
| ELM_CLV | **1.5** | 1.3 | 1.5 | 1.6 | 1.5 | **1.0** | 1.1 | 1.7 | 1.1 | 0.2 |
| ELM_DEG | **2.1** | 5.0 | 0.7 | 4.2 | 2.3 | **0.3** | 0.0 | 0.0 | 1.2 | 0.0 |
| ELM_DOC | **0.9** | 0.7 | 1.0 | 0.8 | 1.1 | **0.4** | 0.7 | 0.8 | 0.4 | 0.0 |
| ELM_LIG | **1.0** | 0.6 | 1.0 | 0.9 | 1.2 | **1.0** | 0.8 | 1.0 | 1.0 | 1.2 |
| ELM_MOD | **1.3** | 1.5 | 1.1 | 1.2 | 1.5 | **0.7** | 0.8 | 0.5 | 0.7 | 1.0 |
| ELM_TRG | **1.6** | 2.4 | 1.6 | 2.1 | 0.8 | **2.2** | 2.7 | 2.4 | 2.9 | 0.7 |
| IDRs | | | | | | **4.7** | 5.5 | 4.5 | 6.4 | 2.7 |
| MoRFs | **4.7** | 5.5 | 4.5 | 6.4 | 2.7 | | | | | |

**Figure 3.2.** Enrichment of IDRs and MoRFs in various functional and structural regions of DENV proteins.

The regions include cleavage sites (CLV), transmembrane regions ('Trans'), the intra-membrane region ('Intra'), topological cytoplasmic, extracellular and luminal domains (Topo-cy, Topo-ex and Topo-lu), functional sites (Func), six types of ELMs (ELM_CLV, _DEG, _DOC, _LIG, _MOD and _TRG), IDRs and MoRFs. Ratios above 1 and below 1 correspond to the enrichment and depletion, respectively. A dark red background indicates enrichment (ratio ≥ 1.2), light red background indicates neutral (slight enrichment) (1 ≤ ratio < 1.2), light blue indicates neutral (slight depletion) (0.8 < ratio < 1), and dark blue indicates depletion (ratio ≤ 0.8). The column designated as "all" shows results across all DENV proteins, while columns 1, 2, 3 and 4 show results for specific serotypes.

Considering the 240 DENV proteins, we found 404 putative IDRs by using MFDp, and 247 putative MoRF regions by using MoRFpred. We have also done additional analysis of the structural and functional annotations that are available for this virus. Our aim is to investigate whether these annotations are associated (depleted or enriched) with the disordered and MoRF regions. Specifically, we collect 14 structural and functional annotations that include cleavage sites (CLV), transmembrane region (Trans), intramembrane region (Intra), three types of topological domains (Topo), functional sites (Func) and six types of eukaryotic linear motifs (ELM).

We analyze the amount of disordered residues and MoRF residues in the various functional regions in DENV. Results of this analysis are shown in Figure 3.2. We categorize the occurrence of IDR or MoRF residues as enriched, depleted or neural based on the ratio of the rates of occurrence of disordered or MoRF residues in a given type of functional region and in the entire polyprotein. The ratio ≥ 1.2 and ≤ 0.8 denotes enrichment and depletion, respectively. For example, in Figure 3.2, IDRs in cleavage sites (CLV) are enriched with ratio of 6.2, which means that disordered residues occur 6.2 times more often in the cleavage sites compared with the overall proteome. This analysis reveals that cleavage sites are substantially enriched in the disorder regions and that transmembrane regions (Trans) are enriched in MoRFs. Figure 3.2 also shows that cytoplasmic topological (Topo-cy) domains are enriched in disorder and MoRF regions. On the other hand, the functional sites are depleted in the disordered and MoRF regions. Targeting ELM sites (TRG) are enriched in MoRFs, and several types of ELMs (CLV, DEG, MOD and TRG) are enriched in IDRs. MoRF residues are enriched in disorder and disordered residues are enriched in MoRFs, which is expected since MoRFs are localized within the disordered regions.

## 3.3 Discussion

We observe that alignment finds functions for only about 10% of IDRs (44 out of 404). This low number of alignment-derived annotations is due to the fact that the segments being predicted by alignment share low sequence similarity with the functionally annotated reference segments from DisProt. We note that although alignment is not sufficient to annotate functions, DisProt is a source of a relatively large and rich set of functionally annotated disordered regions for many functions. These regions could be used to develop and empirically tests computational predictive models. Data-driven predictors of functions of disorder, such as MoRFpred and DisoRDPbind, rely on sequence- and structure-based features rather than alignment, and they were empirically shown to accurately predict their target functions for regions that lack similarity with the annotated regions.

Figure 3.2 reveals that known functional sites (Func) in DENV proteins are depleted in IDRs and MoRFs. IDRs are known to carry out various functions and MoRFs are involved in protein-protein interactions that are necessary for the viral proteomes. The lack of enrichment in disorder in these known functional sites can be explained by a bias to annotate functions of structured regions in these viral proteins. In other words, we speculate that the current list of functional sites is incomplete since it lacks many of the functional disordered regions. We note that the number of MoRF regions predicted by MoRFpred is 247, while the number of protein-protein binding regions found by alignment is substantially smaller and equals 38 (Figure 3.1). Furthermore, MoRFs are a sub-type of protein-protein binding regions. These observations suggest that alignment under-predicts the functional regions, particularly those related to the protein-protein binding.

Based on our analysis of the dengue virus, we find that alignment is not sufficient to functionally annotate disordered regions. Although methods that predict the disordered protein-binding regions help to comprehensively annotate some of the functions of intrinsic disorder, predictors that do not rely on the alignment and that cover many other functions of the intrinsic disorder are urgently needed.

Chapter 4

# Fast and accurate computational prediction of disordered flexible linker regions

## 4.1 Introduction

### 4.1.1 Disordered flexible linkers (DFLs)

Section 2.2.3 in the background chapter explains that several methods have been developed for the prediction of disordered regions that carry out various protein-, DNA- and RNA-binding functions. However, methods for the prediction of the other functions of the IDRs are lacking. As shown in Table 2.1, the most annotated non-binding function of disorder are flexible linkers (14%, 122 out of 899 functionally annotated IDRs in DisProt 6.0.2). Disordered flexible linkers (DFLs) are disordered regions that serve as linkers or spacers between protein domains in multi-domain proteins and between structured intra-domain constituents [4]. Experimental annotation of DFLs relies primarily on the X-ray crystallography, NMR spectroscopy and circular dichroism. We consider building predictive models for these regions for several reasons. First, this is the most annotated and not related to binding function of IDRs. Second, DFLs are important for a variety of cellular processes. A few recent examples include formation of amyloid fibrils [144], linking multiple disordered protein binding regions [145], and movement of structured domains between catalytic sites [146]. Third, there is no computational method that predicts this class of IDRs. DFLs are cousins of linkers, which are regions that connect domains and maintain inter-domain interactions [147, 148]. A sub-class of linkers are flexible linkers, defined as flexible inter-domain regions that allow two domains to move relatively to each other [147]. DFLs differ from linker regions in two aspects: 1) DFLs are characterized by extreme level of flexibility and lack of defined structure (they form conformational ensembles) as compared to linkers and flexible linkers that have more defined structures; and 2) linkers are shorter (average length of 10 residues) and localized between domains [148] while DFLs tend to be longer (average length of 25 residues in our dataset) and could be localized inside domains, for instance, to link structured elements in a domain.

### 4.1.2 Alternative ways to predict DFLs

Although there are no computational methods that directly predict DFLs in protein sequences, UMA method [149] can be used to predict flexible linker regions. It works based on assumptions that flexible linkers are less likely to be conserved in the sequence and secondary structure and that they are depleted in hydrophilic residues. Thus, UMA quantifies every residue as a weighted sum of hydrophobicity score and conservation scores for sequence and secondary structure. A low UMA score indicates that a residue is more likely to be a flexible linker. Since flexible linkers are a subset of flexible residues, they could be also potentially identified with sequence-based predictors of flexible residues. These predictors include PROFbval [139, 150], FlexPred [151, 152], PredBF [153], PredyFlexy [154] and DynaMine [155, 156]. PROFbval predicts B-factors using a Neural Network model, where a low/high real B-factor value indicates a low/high propensity of a residue being flexible. PredBF predicts B-factors by using a two-layer Support Vector Regression (SVR) model. FlexPred predicts conformationally variable positions in the input protein chain using a Support Vector Machine (SVM). PredyFlexy classifies every input residue as rigid, intermediate or flexible and also outputs putative normalized B-factors and RMSFs (root mean square fluctuations), from molecular dynamic simulations. DynaMine quantifies backbone flexibility in terms of N-H $S^2$ order parameter values using regression where smaller $S^2$ means that a given residue is more likely to be flexible.

The UMA method and protein flexibility predictors predict flexible linkers/residues but they do not accommodate for the disordered state of these residues. Moreover, UMA requires that the input sequence has homologous sequences to generate multiple sequence alignment profiles, which means that it may not generate predictions for some proteins that lack homologues (similar proteins), and is tedious to execute since its implementation requires manual processing. To this end, we propose to develop a predictor of DFL regions which does not need alignment profiles and is fast and accurate. For future reference, we name this method DFLpred (Disordered Flexible Linker predictor).

## 4.2 Materials and methods

### 4.2.1 Datasets

We collect the functionally annotated data from DisProt version 6.0.2 that includes 694 sequences. We excluded DP00688 sequence that was too long (>18,000 residues) to predict with the PSIPRED [157] to generate putative secondary structure. We select 204 sequences which include 82 proteins that have annotations of DFLs and 122 proteins that do not have DFL annotations but for which all residues are annotated. This way we include all annotated DFLs and reduce the number of ambiguous (unannotated) residues.

We assumed that residues that are not annotated as DFLs but which have other functional annotations are non-disordered flexible linker (NDFL) residues. The residues without functional annotations were excluded from the design and assessment. We divide the set of 204 proteins into five subsets and reduced sequence similarity between these subsets with BLASTClust [105]. First, we cluster the 204 sequence with sequence identity threshold at 25%. Second, the resulting 160 clusters that include similar sequences (≥ 25% similarity) were divided at random between the five sub sets to ensure that each subset has similar number of sequences and similar ratio of DFL to NDFL residues. Four of these subsets were used in four-fold cross validation protocol to empirically design our predictor, i.e., to conceptualize and select features for the predictive model, and to select and parameterize this model. These data constitute the training dataset. The remaining fifth subset was used as an independent (never used in the design) test dataset. This way, sequences in the test dataset share low similarity with sequences in the training dataset, and also sequence in individual folds of the training dataset share low similarity with sequences in the other folds. The training and test datasets have 144 sequences and 60 sequences, respectively.

### 4.2.2 Overall design of DFLpred

The architecture of DFLpred includes three layers: 1) Layer 1 represents every residue of the input sequence with its amino acid (AA) type and information predicted directly from the sequence including propensity to form structured regions, intrinsically disordered regions, and helical and coil conformations. 2) Layer 2 converts this representation into empirically selected set of numeric features that are computed using sliding windows. 3) Layer 3 inputs the selected features into empirically selected and parameterized predictive model to generate propensity scores.

**Sequence profile**

In the first layer, we represent every residue in the input protein sequence by its AA type, its physicochemical properties estimated based on the AA indices from the AAindex database [158], secondary structure predicted with PSIPRED [157], intrinsically disordered and structured regions predicted with IUPred [81, 82] and sequence complexity computed with SEG [159]. We retrieve 531 AA indices from the AAindex database, and run PSIPRED, IUPred and SEG webserver or standalone packages provided by the corresponding authors with default parameters. IUPred produces predictions for long and short IDRs and structured regions, and we use all of them.

**Feature representation**

In the second layer, we generate numerical features that quantify the considered structural and sequence-based properties for each residue of the input AA sequence. We represent every residue by a feature vector calculated from a window centered over that residue. The window aggregates structural and sequence-based information by considering characteristics of AAs adjacent in the sequence. The concept of the window has been adopted in other relevant predictors such as PROFbval, FlexPred, PredBF, PredyFlexy and DynaMine. We set the length of the window to 17, which is the median value of the length of longest per protein DFLs in our dataset. This way the select window size covers the full length of at least half of DFLs without recruiting much of potential noise (NDFL residues) when used to predict shorter regions. We did not pad the window for the residues located at either terminus of the sequence and correspondingly the length of the window is reduced on one of its sides, i.e., window size is 8 for the first and last residue in the sequence. Consequently, we normalize values of features computed over the residues in the window by the size of the window. In total, we considered 2236 features including 40 features that we derived directly from the sequence, 2124 features derived from physicochemical properties of AAs quantified based on the AAindex database [158], 22 features generated from the putative secondary structures, 40 features from putative intrinsic disorder and structured regions and 10 features from the sequence complexity. These features quantify composition of AAs; composition, counts, and length of putative secondary structures, intrinsically disordered regions, structured regions and high sequence complexity regions in the window; average physicochemical properties of residues in the windows; and AA types, secondary structure, disorder status, sequence complexity status and physicochemical properties of the residue in the center of the window. Detailed list of these features are shown in the Appendix A.

### Feature selection and design of predictive model

The vector of 2236 features likely includes features that are irrelevant to the prediction of DFLs and features that have high mutual correlations. We utilized a two-step empirical feature selection to select a subset of features characterized by high predictive value and low mutual correlations.

In the first step of feature selection we remove low quality features that have low correlation with the annotation of the DFLs. We have two types of features: real-valued (e.g., features computed as an average over the window) and binary (e.g., disordered vs. ordered status of the residue in the center of the window). Inspired by [160, 161], we use point-biserial correlation coefficient ($r_{pb}$) and $\varphi$ coefficient ($\varphi$) respectively, for these two feature types:

$$r_{pb} = \frac{M_{DFL} - M_{NDFL}}{S_n} \times \sqrt{\frac{n_{DLF} \times n_{NDLF}}{n^2}}$$

$$\varphi = \frac{count_{F_1 A_{DFL}} \times count_{F_0 A_{NDFL}} - count_{F_1 A_{NDFL}} \times count_{F_0 A_{DFL}}}{count_{F_1} \times count_{F_0} \times count_{\ A_{DFL}} \times count_{A_{NDFL}}}$$

where $M_{DFL}$ and $M_{NDFL}$ ($n_{DFL}$ and $n_{NDFL}$) are the means (numbers) of values a given real-valued feature for the residues annotated as DFLs and NDFLs, respectively; $n = n_{DFL} + n_{NDFL}$ and $S_n$ is the standard deviation of all values of that feature. $count_{FiAk}$ is the number of values $i = \{0, 1\}$ of binary feature $F$ corresponding to residues with values $k = \{\text{NDFL}, \text{DFL}\}$ of the annotation $A$; $count_{Fi}$ and $count_{Ak}$ are the number of values $i = \{0, 1\}$ of binary feature $F$ and the number of residues with values $k = \{\text{NDFL}, \text{DFL}\}$ of the annotation $A$, respectively. We calculate average $r_{pb}$ (for the real-valued features) and $\varphi$ (for the binary features) for all considered features from four correlations computed on the training folds from the four-fold cross validation on the training dataset. We normalize the values of the average $r_{pb}$ and $\varphi$ correlations to the -1 to 1 range, and remove the features for which the absolute normalized $r_{pb}$ or $\varphi$ value < threshold $T_{step1}$. Next, we rank the remaining features by their absolute normalized $r_{pb}$ or $\varphi$ values.

In the second step of feature selection, we eliminate mutually correlated features using the Pearson correlation coefficient ($r_{pc}$). First, a set of selected features is initialized with the top-ranked in the first step feature. Next, we calculate $r_{pc}$ between the next-ranked feature and all selected features. If the absolute value of this $r_{pc}$ < threshold $T_{step2}$ then we add this next-ranked features into the set of selected features, otherwise we do not add it. We apply this procedure through the entire list of ranked features passed from the first step.

We vary values of each of the two thresholds, $T_{step1}$ and $T_{step2}$ between 0.1 and 0.9 with step of 0.05, to obtain 17×17= 289 different feature sets. The corresponding feature sets vary in size between 1 and 884 features. Each feature set is used with three types of classifiers: Logistic Regression, Naive Bayes and *k*-Nearest Neighbor, in the four-folds cross validation on the training dataset to select the design that offers the highest AUC value. We use the implementations of these classifiers in the Weka platform [162]. We also parameterized Logistic Regression and *k*-Nearest Neighbor classifiers for each of these experiments by selecting their parameters that corresponds to the highest AUC in the four-folds cross validation on the training dataset. Naive Bayes has no parameters. For the Logistic Regression we considered *ridge* = $10^x$, where $x$ ranges from -4 to 4 with step of 1. For the *k*-Nearest Neighbor, we consider the number of neighbors $k$ ranging from 50 to 800 with the step of 50. Table 4.1 summarizes results with the highest AUC value for each of the three classifiers, which are selected from across the experiments that correspond to 7514 combinations of the two thresholds and different parameters of classifiers (289 combinations for Naive Bayes + 9×298 for Logistic Regression + 16×289 for *k*-Nearest Neighbor). AUC$_{LowFPR}$ and AUC$_{ratio}$ are calculated by considering false positive rate $\leq 0.1$. The *p*-values in table 4.1 are calculated by bootstrapping 50% proteins from the cross validation results and repeating for 10 times. Given the 10 pairs of measurements are draw from normal distribution, which are assessed with the Anderson-Darling test, we use the Student's paired *t*-test, otherwise we use the Wilcoxon signed-rank test.

**Table 4.1.** Cross validation results for the three types of classifiers on the training dataset.

$T_{step1}$: threshold for normalized $r_{pb}$ or $\varphi$; $T_{step2}$: threshold for $r_{pc}$; Parameters: parameters selected for individual classifiers where *r* is the ridge for Logistic Regression and *k* is the number of neighbors for *k*-Nearest Neighbors; AUC: area under the ROC; AUC$_{lowFPR}$: area of a part of the ROC for FPR between 0 and 0.1; AUC$_{ratio}$ = AUC$_{lowFPR}$/AUC$_{lowFPR\_random}$ where AUC$_{lowFPR\_random}$ is the AUC of random predictor assessed for FPR between 0 and 0.1. The AUC, AUC$_{lowFPR}$ and AUC$_{ratio}$ values were calculated over the 4 combined test folds in the cross validation, and thus they represent results on the entire training dataset. + indicates that difference in predictive quality between LR and another classifier is statistically significant at *p*-value < 0.01.

| **Classifier** | $T_{step1}$ | $T_{step2}$ | **Number of the selected features** | **Parameters** | **AUC** | **AUC$_{lowFPR}$** | **AUC$_{ratio}$** |
|---|---|---|---|---|---|---|---|
| Logistic Regression | 0.50 | 0.35 | 4 | *r* = 100 | 0.702 | 0.016 | 3.27 |
| Naive Bayes | 0.50 | 0.35 | 4 | N/A | 0.680 + | 0.014 + | 2.81 + |
| K-Nearest Neighbor | 0.45 | 0.35 | 5 | *k* = 500 | 0.677 + | 0.015 + | 2.93 + |

We select the Logistic Regression classifier with 4 features that gives the highest values of AUC, AUC$_{lowFPR}$ and AUC$_{ratio}$. The differences in these three measures of predictive quality between the Logistic Regression and the other two classifiers are statistically significant. The ratio reveals that the selected design is about 3.3 times better than a random predictor when predicting with low FPR, i.e., when a high fraction of predictions of DFL residues (predicted positive residues) is correct. The architecture of this model is shown in Figure 4.1. Given an input AA sequence, it utilizes putative annotations of structured and long disordered regions generated with IUPred and two physicochemical properties of residues that quantify propensity for formation of helices and turns.

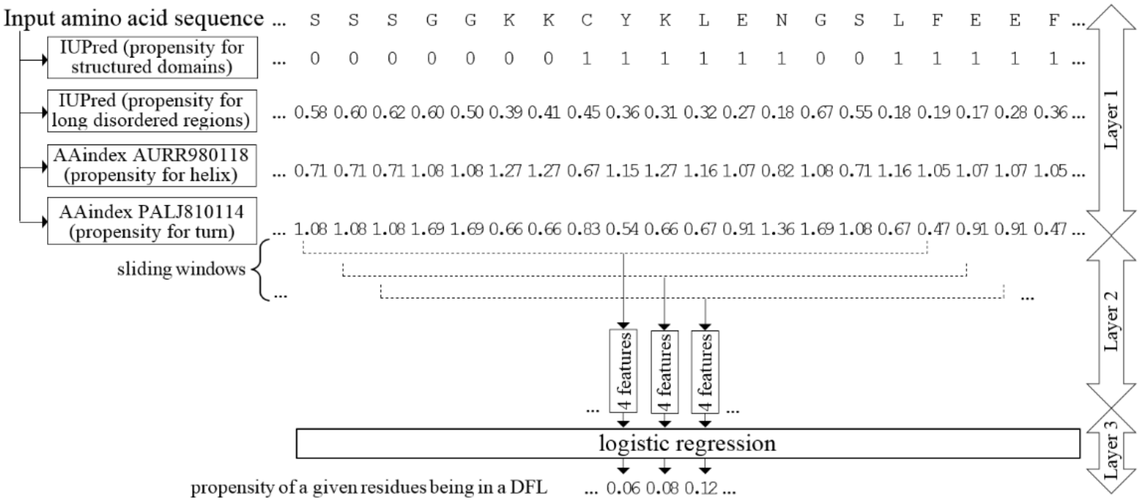


**Figure 4.1.** Architecture of DFLpred.

The selected four features, their point-biserial correlation coefficient ($r_{\mathrm{pb}}$) with the class label, and their coefficient in the logistic regression model are shown in Table 4.2. These features are computed from the sequence using windows on putative annotations generated with IUPred and two AA indices. For each residue, the output score representing the propensity of being DFL ranges between 0 and 1 and is calculated by the Logistic function $f(\mathrm{t}) = 1/(1 + e^{-t})$, where $t$ is the linear combination of the four features, i.e., $t = \alpha_0 x_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3 + \alpha_4 x_4$, $x_{\{0, 1, 2, 3, 4\}}$ are the constant and the four features, and $\alpha_{\{0, 1, 2, 3, 4\}}$ are the five coefficients.

**Table 4.2.** Summary of features used in the DFLpred model.

| **Feature name** | **Description** | ***r<sub>pb</sub>*** | ***Coeff*** |
|---|---|---|---|
| WIN_IUP_fractionD$_0$ | Number of residues predicted with IUPred_struct not to be in structured regions in a sliding window divided by the window length. | -1.00 | -1.10 |
| WIN_IUP_std$_{\mathrm{L}}$ | Standard deviation of propensity scores from IUPred _long for residues in the sliding window. | 0.70 | 6.60 |
| WIN_AAind_avg$_{\mathrm{AURR980118}}$ | Average value of AURR980118 AA index for all residues in the sliding window. | -0.66 | -4.58 |
| WIN_AAind_avg$_{\mathrm{PALJ810114}}$ | Average value of AURR980118 AA index for all residues in the sliding window. | 0.57 | 3.32 |
| constant | | N/A | -0.92 |

## 4.3 Results

### 4.3.1 Analysis of the predictive model

Figure 4.2 compares values of the four features utilized by DFLpred between the native DFL and NDFL residues in the test dataset (these results were not used to design the model, which is based on the training dataset). The WIN_IUP_fraction$D_0$ feature quantifies fraction of residues predicted with IUPred_struct not to be in structured regions in a window centered on the predicted residue. Its average for DFL residues is 0.29, for the NDFL residues is 0.60, and for the NFDL residues annotated as structured is 0.11 (not shown in the figure). The structured residues have the lowest value since they should be primarily predicted to be in structured regions; the value > 0 since some of the residues in the surrounding window could lack structure. The high value of mean for NFDL residues is driven by the fact that these residues include disordered residues that are not DFLs which have a large number of nearby (in the sequence) unstructured residues. The average for DFLs is in between the other two average. This reveals that propensity of these residues to be nearby putative structured regions is lower than for other disordered regions but higher than for structured regions. This makes sense since residues in DFLs link primarily structured domains and thus their neighbors in the sequence should include a sizable fraction of structured residues, but not as large as for the structured residues.

The plot of the WIN_IUP_$std_L$ feature in Figure 4.2 suggests that putative propensities for disorder of residues in DFLs have higher standard deviation in the window compared to the other residues. This means that these propensities fluctuate more in residues adjacent to DFL residues. This is reasonable given that DFLs link structured domains where propensity for disorder should be substantially lower compared to DFLs. In contrast, residues located in structured or in disordered regions would experience less variability in the propensities for disorder in these regions.

The last two features are computed by averaging values of the selected two AA indices in the window. Higher values of the AURR980118 index [163] indicate higher likelihood of a given residue to be included in a helical conformation. Thus, the corresponding feature can be used as a proxy to quantify likelihood of helical conformations in the window.

Figure 4.2 shows that residues in DFLs have lower values of this feature which suggests that they are less likely to include helices nearby in the sequence compared to NDFLs. This again is expected since DFLs are unstructured (less likely to form helical conformations). The second, PALJ810114 index [164] quantifies likelihood of forming turns. Here, DFLs have higher values compared to the other residues, which is sensible given that turns are relatively flexible which is also characteristic for DFLs.

Overall, we demonstrate that the four features are different from each other and that they are meaningful markers of DFLs. This agrees with our empirical approach to design DFLpred in which we explicitly select highly predictive features (first step of feature selection) that are characterized by low mutual correlation (second step of feature selection).

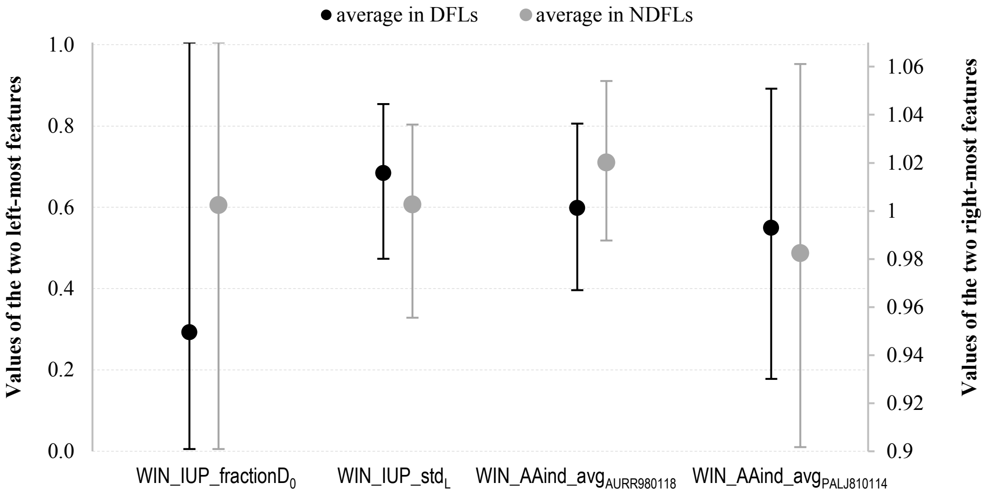


**Figure 4.2.** Comparison of values of features used in DFLpred.

The comparisons are between the native DFL residues (black lines) and native NDFL residues (gray lines) in the test dataset. The features are ranked by their absolute $r_{\mathrm{pb}}$ values from the highest on the left to the lowest on the right (see Table 4.2). Values of the WIN_IUP_std$_L$ are multiplied by 10 to better fit the range of values of the other features. Dots are the averages and the error bars show the first and third quantiles.

### 4.3.2 Comparison of predictive performance with closest alternative methods

We compare the predictive performance of DFLpred with the closest alternative methods that could be used to find DFLs. These approaches include the UMA method that finds flexible linkers, predictors of flexible residues and disordered residues, and a domain predictor given the fact that classical linkers are localized between domains. We also combined the results of UMA with the results of the disorder predictors and the results of the flexibility predictors with the disorder predictors. This was motivated by the fact that these combinations could potentially find flexible linkers or flexible residues localized in disordered regions, which is the hallmark of the DFLs. We utilize two ways to combine their predictions, by multiplying the scores predicted with UMA and flexibility predictors by the binary disorder predictions and by the predicted real-valued propensity for the disorder. In the first case, the UMA and flexibility scores remain the same for the predicted disordered residues and are set to zero for the residues that are not predicted to be disordered. In the second scenario, the UMA and flexibility scores are scaled by the predicted propensity for disorder. We use a comprehensive set of predictors of flexible residues including PROFbval, FlexPred, PredBF, PredyFlexy and Dynamine. We also consider several predictors of disorder including two versions of IUPred (short and long), MFDp [127] and three versions of Espritz (NMR, X-Ray and DisProt) [165]. We apply ThreaDom [166] to predict domains given its strong predictive performance and availability of a webserver.

**Table 4.3.** Comparison of predictive quality for DLFs on the test dataset.

The methods were ranked by AUC value in each category. + denotes that difference in predictive quality is statistically significant at *p*-value < 0.01 when compared with DFLpred.

| Prediction target | Method | AUC | AUC$_{lowFPR}$ | AUC$_{ratio}$ |
|---|---|---|---|---|
| DFLs | DFLpred | 0.715 | 0.016 | 3.265 |
| | Espritz_NMR & Predyflexy (the best based on binary disorder) | 0.459 + | 0.006 + | 1.154 + |
| | Espritz_NMR & Predyflexy (the best based on disorder propensity) | 0.429 + | 0.003 + | 0.653 + |
| Flexible linkers | UMA | 0.384 + | 0.003 + | 0.531 + |
| Flexible residues | PredyFlexy | 0.531 + | 0.007 + | 1.307 + |
| | FlexPred | 0.486 + | 0.004 + | 0.768 + |
| | PROFbval | 0.453 + | 0.007 + | 0.337 + |
| | PredBF | 0.445 + | 0.005 + | 0.988 + |
| | Dynamine | 0.396 + | 0.003 + | 0.573 + |
| Disordered residues | Espritz_NMR | 0.399 + | 0.001 + | 0.218 + |
| | IUPred_short | 0.359 + | 0.000 + | 0.092 + |
| | MFDp | 0.325 + | 0.004 + | 0.201 + |
| Domains | ThreaDom | 0.521 + | 0.003 + | 0.569 + |

Table 4.3 summarizes results of DFLpred and the other methods on the test dataset. AUC$_{LowFPR}$ and AUC$_{ratio}$ are calculated by considering false positive rate ≤ 0.1. The *p*-values are calculated by bootstrapping 50% proteins from the results of test dataset and repeating for 10 times. If the 10 pairs of measurements are drawn from normal distribution, we use the Student's paired *t*-test, otherwise we use the Wilcoxon signed-rank test. We show results for DFLpred, UMA, the five methods for prediction of flexible residues, the three methods for the prediction of disordered residues (we show results for one version of IUPred and Espritz that secures the highest AUC) and ThreaDom for the prediction of domains. We also include result for each of the two ways to combine these methods, as described above, whichever secured the highest AUC value. Figure 4.3 shows the ROC curve for DFLpred and other alternatives that secured AUC > 0.5 on the independent test dataset.

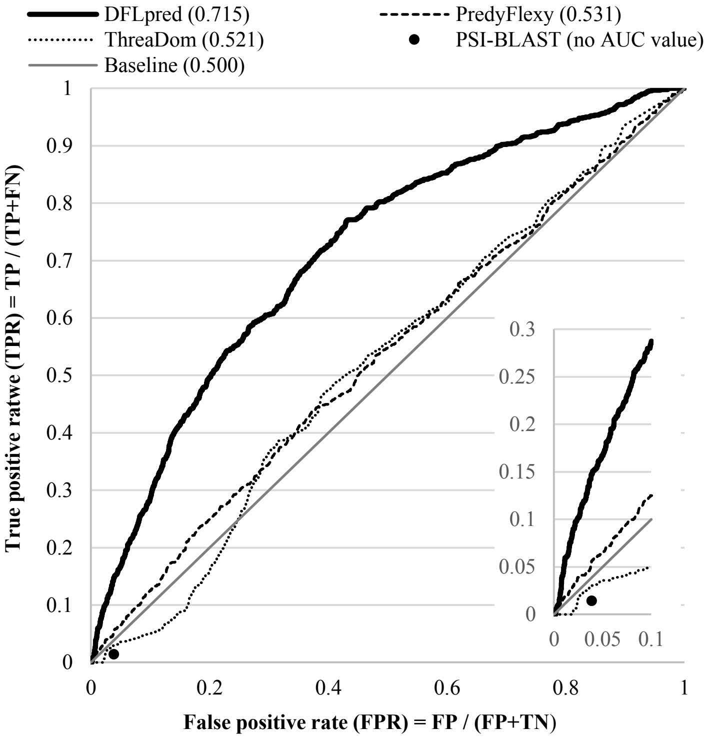


**Figure 4.3.** ROC curves on the test dataset.

This figure shows ROC curves for DFLpred and methods that achieved AUC > 0.5 in Table 4.3. Insert in the bottom right corner focuses on the ROCs for FPR between 0 and 0.1. AUC values are shown inside brackets next to the names of methods in the figure legend.

DFLpred secures the highest AUC, AUC$_{lowFPR}$ and AUC$_{ratio}$ values. The improvements offered by DFLpred are significant at $p$-value < 0.01 when compared with all considered methods. The AUC$_{ratio}$ indicates that the DFLpred is about 3.3 times better than a random predictor in AUC for the low values of FPR ≤ 0.1. The UMA method secures low AUC and this could be explained by the fact that UMA predicts flexible linkers that likely exclude linkers located in disordered regions; the latter stems from the low value of AUC$_{lowFPR}$. The low AUC of UMA is due to a concave upwards shape of ROC curve that in turn results from high levels of false positives on the left side of the curve. These false positives are the residues predicted as DFLs for the purpose of our evaluation but which in fact correspond to the putative flexible linkers predicted by UMA with the highest values of propensity. The predictors of disordered residues have similar weakness. They predict all disordered residues, irrespective of their function, while most of them are not DFLs. Their low values of AUC$_{lowFPR}$ suggest that propensities generated for DFLs are lower than the propensities for other disordered regions. This results in high false positive rates, concave upwards shapes of ROC curves and consequently low AUC values. Predictors of flexible residues also secure relatively low AUC values. These methods were built utilizing crystallographic data that exclude disordered residues and thus they cannot accurately find disordered residues, which was shown in [15]. ThreaDom finds the inter-domain regions but only some of them are DFLs and it in general fails to find the intra-domain DFLs. Consequently, its AUC value is relatively low. Interestingly, combining UMA/flexibility predictors with the disorder predictors also does not produce high quality predictions. This is likely because neither UMA nor predictors of flexibility provide accurate estimates for the disordered regions. We also include the results when using sequence alignment. The alignment-based predictor transfers DFL annotations of aligned residues from the most similar sequence in the training dataset based on its alignment to a query sequence from the test dataset. The alignment is done with PSI-BLAST [167] using default parameters. Since alignment transfers binary annotations of DFLs from the training proteins, we can show only one point for the ROC curve for this simple predictor. The FPR and TPR values of alignment-based predictor equal 0.039 and 0.014, respectively. In other words, it predicts only 1.4% of DFL residues with the cost of predicting 3.9 % of NDFL residues as DFL residues. This is because our test dataset is designed to share low (25% or lower) sequence similarity with the training dataset. In contrast, DFLpred can produce accurate results even in the absence of sequence similarity. Overall, we conclude that while currently there are no approaches that can accurately predict DFLs, DFLpred offers high-quality predictions.

### 4.3.3 Comparison of runtime

DFLpred is implemented as a linear function of features computed directly from sequence and from sequence-derived predictions generated with IUPred. IUPred's predictions are calculated from pairwise energy profile without a time-consuming alignment and predictive model. Consequently, DFLpred is very fast. We quantify and compare the runtime of DFLpred with the runtime of UMA and PredyFlexy, the latter is the only flexibility predictor that obtained AUC >0.5. The predictions were run on the same 64-bit computer with 3.5GHZ CPU and 4 GB of RAM running the Ubuntu operating system. UMA was run manually and requires computation of hydrophobicity, finding homologous sequences and prediction of secondary structure. We estimate a lower bound on the UMA's runtime by computing time to complete the most runtime-consuming task of finding the homologs. This is based on executing BLAST against the NR database using the suggested by the author *e*-value = 1e-20. We use standalone version of PredyFlexy that was provided by the authors. The domain predictor ThreaDom also secured AUC > 0.5 but is not included in the runtime comparison because it requires running LOMETS (Local Meta Threading Server) framework [168] which takes substantially longer time than the other methods. We collect the runtime of the considered methods for 204 proteins from the training and test datasets. We sort the proteins by their size quantified with the sequence length, divide them into 10 equally sized groups based on the size, and compute average runtime for each group.

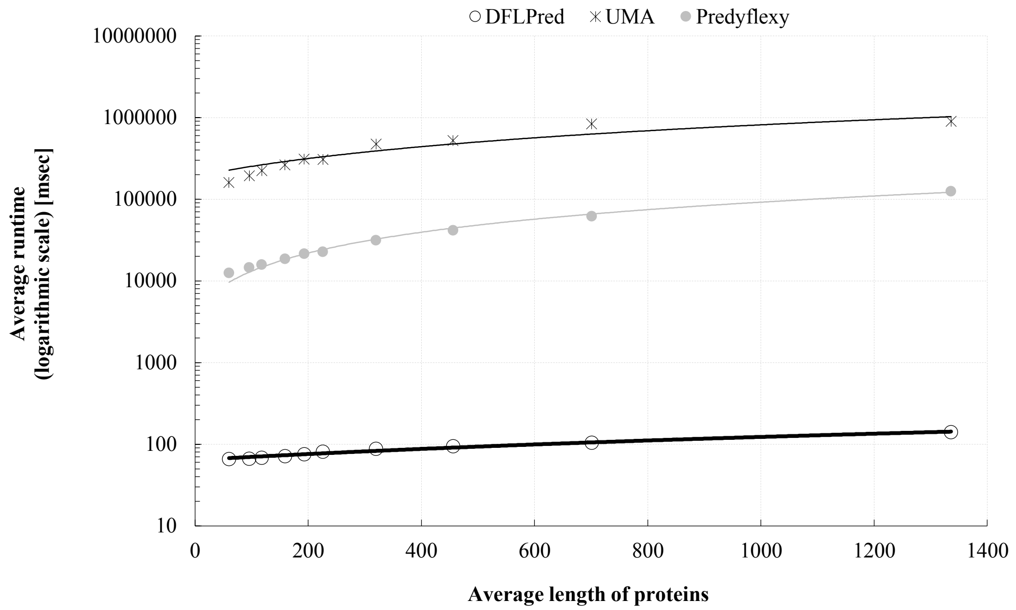


**Figure 4.4.** Comparison of runtime for DFLpred, UMA and PredyFlexy.

Figure 4.4 compares the average runtime against the average length of sequences for the three methods. The runtime of DFLpred, PredyFlexy and UMA is in the range of $10^2$, $10^4$ to $10^5$ and $10^5$ to $10^6$, respectively, measured in milliseconds. DFLpred is up to 4 orders of magnitude faster than the alternatives, which is a significant advantage. To put this into perspective, if these methods would be used to predict the complete reviewed human proteome from UniProt (20,193 sequences with an average length of 561), the DFLpred, PredyFlexy and UMA would take about 40 minutes, 11.5 days and 185 days respectively. This estimate is based on a linear fit into the measured data that is shown in Figure 4.4 and assuming use of the same computer that we used to measure the runtime. To summarize, DFLpred is faster than the less accurate alternatives and is capable of providing predictions for the complete human proteome (and any other proteome which by definition would be smaller) using a modern personal computer in under an hour.

### 4.3.4 Case studies

We use two proteins from the test dataset to visualize prediction of DFLs localized between domains (chemotaxis cheA protein; Figure 4.5 A) and inside of a domain (troponin I protein; Figure 4.5 B). CheA includes five domains and we focus on the C-terminus that includes Hpt, CheY binding, and signal transducing H kinase domains that are connected by two inter-domain DFLs. Troponin I includes two domains: troponin I N-terminus domain which is disordered, and troponin I domain that is composed of two sub-domains: IT arm and regulatory head. The IT arm sub-domain has DFL that links two of its helices that interact with troponins T and C that compose the troponin complex. The second DFL links the two sub-domains. Both of these intra-domain DFLs enable movement of several structural elements of the troponin I domain allowing it to interact with the other members of the troponin complex [169]. The figures show predictions from DFLpred, UMA, domain predictor ThreaDom and the best performing (based on Table 4.3) disorder predictor Espritz and flexibility predictor PredyFlexy. DFLpred generates higher propensities in a vicinity of the two inter-domain DFLs in CheA, with the second one predicted less accurately. It also finds the first intra-domain DFL and to some extent, given the lower values of propensity, the second intra-domain DFL in troponin I. ThreaDom accurately finds the inter-domain residues that overlap with the two DFLs in CheA, but its prediction also includes residues at the N-terminus that are not DFLs. This method finds the inter-domain region in troponin I, which is not a DFL, and has difficulty with the troponin I domain given it's fragmented into sub-domains composition. Espritz accurately predicts the disordered residues which coincide with the inter-domain DFLs in CheA, but it also finds disorder at the N-terminus. It correctly finds the first two disordered regions in troponin I but it misses the second intra-domain DFL and predicts the disordered region at the N-terminus the highest propensity while this region is not a linker. UMA finds three flexible linkers (residues with high scores) in CheA and only the last one coincides to some extend with the second DFL. For the second protein this method annotates only the N-terminus as a flexible linker and fails to identify the intra-domain DFLs. Finally, PredyFlexy does not find inter-domain residues, DFLs or disordered regions, but rather it estimates local flexibility which fluctuates widely along the sequence of both proteins. These observations provide context to interpret results of the other methods in Table 4.3.

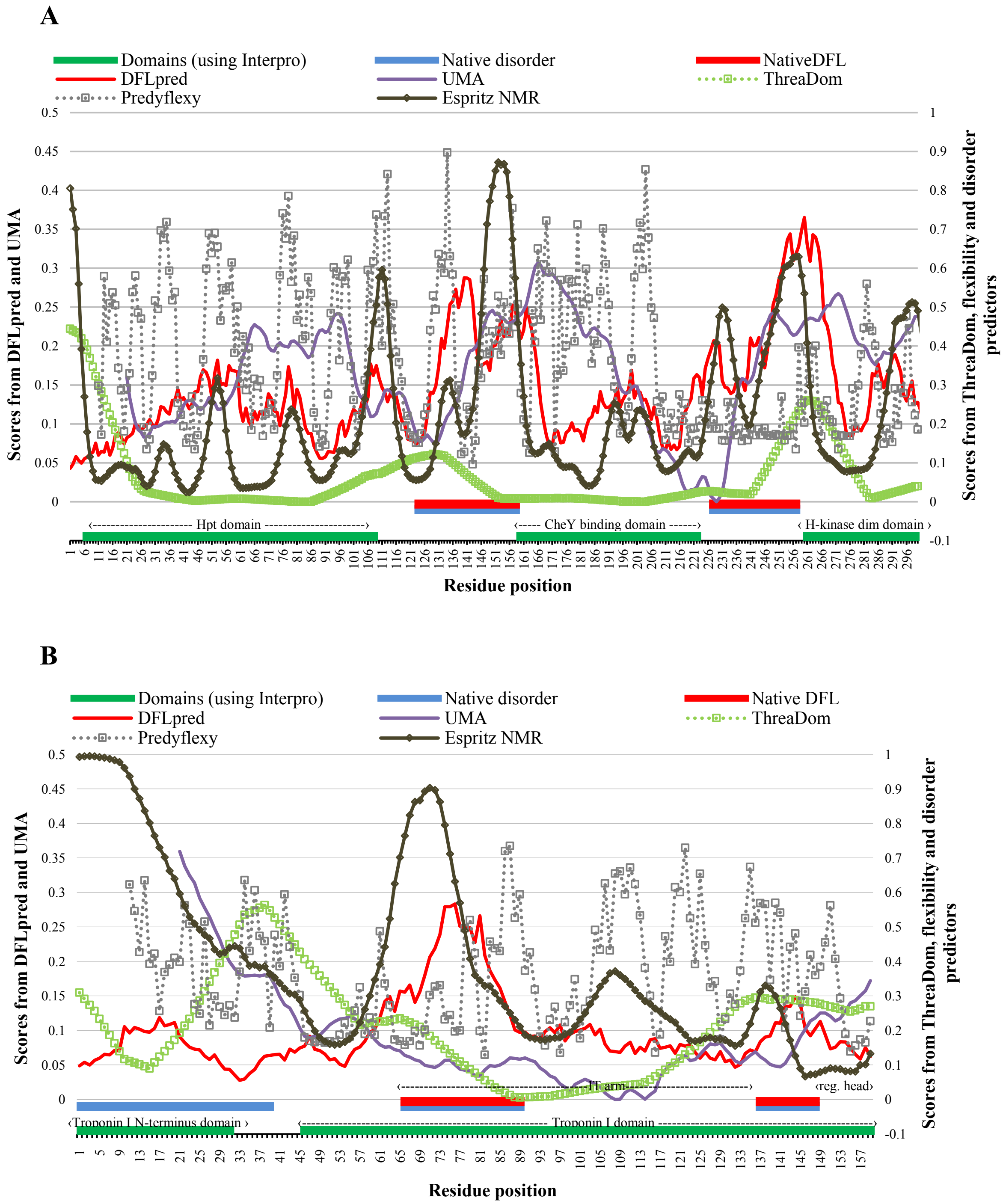


**Figure 4.5.** Predictions and native annotations for two proteins in case study.

The two proteins are the C-terminus of the chemotaxis cheA protein (panel A) and the N-terminus of the troponin I protein (panel B) from the test dataset. We include annotations and names of domains (green horizontal line at the bottom with names above the line), disordered regions (blue horizon-tal line at the bottom), DFLs (red horizontal line at the bottom), and predictions from DFLpred (thick red plot), UMA (thick violet plot), best performing disorder predictor Espritz (black line with diamond markers), best performing flexibility predictor PredyFlexy (dotted gray line with square markers), and the domain predictor ThreaDom (dotted green line with square markers). UMA cannot predict the first 20 residues in panel B due to the use of sliding window.

### 4.3.5 Analysis of putative disordered flexible linkers in human proteome

We analyze putative annotations of DFLs generated with DFLpred in the complete reviewed human proteome collected from UniProt. We consider a given residue to form DFL if its propensity score generated by DFLpred is above 0.18. This cut-off corresponds to low false positive rate = 0.05 based on the results from the cross validation on the training dataset. The actual runtime for the complete reviewed human proteome is 38 minutes, which is close to the estimated 40 minutes (Section 4.3.3). This suggests that our estimates of runtime are accurate.

Figure 4.6 A shows a histogram of the content of putative DFLs residues per sequence (fraction of these residues in a sequence). About 24% of proteins have no DFLs, i.e., the content is below 5% while our estimated FPR is at the same level, and another 52% have small amount of DFL residues. About 10% and 2% of proteins have the content > 30% and > 50%, respectively. We found 341 and 152 proteins that have the content of DFL residues at over 50% and 60%, respectively. Figure 4.6 B is a histogram of the length of putative DFLs. Most of these regions are relatively short, with about 80% of them being shorter than 10 residues; some of them could be spurious predictions given the assumed 5% FPR. However, about 7% and 3% of these regions span at least 20 and 30 consecutive residues, respectively. We found 6029 DFL regions that that are at least 30 residues long.

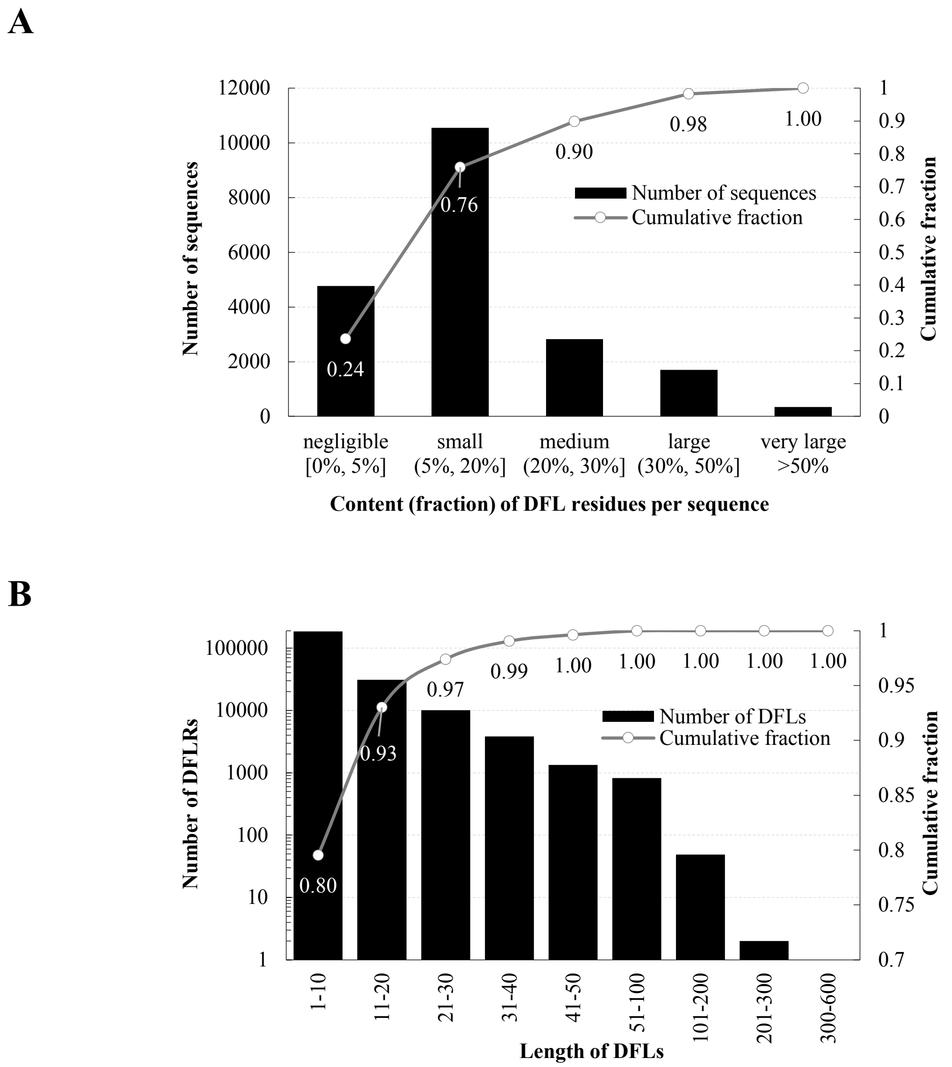


**Figure 4.6.** Histograms with content of putative DFL residues per sequence and length of putative DFLs in the complete reviewed human proteome.

The putative DFL annotations are predicted using DFLpred. Black bars show the number of proteins (panel A) and number of DFLs (panel B) and lines show the cumulative fraction for a given range of the content (panel A) and length of DFLs (panel B).

### 4.3.6 DFLpred webserver

DFLpred is available as a webserver at http://biomine.cs.vcu.edu/servers/DFLpred/. It requires the end user only to provide the input protein sequence(s) in FASTA format and email address. The email is used to deliver a notification of the finished prediction and URL of results that are available for download. The same information is available in the browser window given that this window will not be closed until the prediction is finished. The server automatically generates the corresponding propensities and binary predictions

(DFL vs NDFL residue). The binary predictions are computed from the propensities using the cut-off = 0.18 (residues with propensity ≥ 0.18 are assumed to form DFLs) which corresponds to the 5% FPR. The webserver allows for batch predictions of datasets with up to 5000 proteins.

## 4.4 Summary

We conceptualized, designed, implemented, tested and deployed a novel computational method, DFLpred, for the prediction of the disordered flexible linkers (DFLs) from protein sequences. We developed four strong and complementary sequence-derived markers of DFLs and combined them using a linear function to build this method. Empirical tests on independent (blind) test dataset demonstrate that DFLpred provides relatively accurate predictions, even for proteins that share low sequence identity with the proteins used to develop the predictor. DFLpred outperformes the closest related methods including UMA which predicts flexible linkers, several protein flexibility predictors and their combinations with the disorder predictors. The new method is also characterized by a very low runtime, with prediction of the entire proteome taking less than 1 hour on a modern desktop computer. Finally, our analysis of putative DFLs in human proteome generated with DFLpred shows that DFLs can be likely found in many human proteins. About 10% of human proteins have a significant content of over 30% of DFL residues and a few thousand of these regions are longer than 30 consecutive residues.

## Chapter 5

# Fast and accurate computational prediction of disordered moonlighting regions

## 5.1 Introduction

The term "moonlighting protein" was introduced by Jeffery to denote proteins that perform multiple independent cellular functions within one polypeptide chain [170, 171]. More precisely, a moonlighting protein has multiple autonomous and unrelated functions [171], and these functions are carried out by a single polypeptide chain that cannot be assigned into separate domains [172]. However, the multi-functional phenomenon is possible not only in the whole protein chains, but also in individual disordered regions. According to our estimates, about 37% IDRs annotated in DisProt carry out more than one functions [10]. We use the term "disordered moonlighting regions (DMRs)" to denote IDRs that carry out multiple distinct functions. DMRs differ from the moonlighting proteins. The moonlighting proteins are able to carry out multiple functions since they can be  expressed in different cell types, cellular locations, oligomerization states and can identify different binding ligands [170, 171]. Whereas the multi-functionality of DMRs stems from their high degree of plasticity, which allows a single IDR to bind multiple ligands, serve as a linker and/or perform several entropic functions [173-175]. The moonlighting proteins can be predicted computationally from protein sequences [171, 172, 176, 177]. However, these methods make predictions only at the protein level, not at the residue or sequence region level that is necessary to identify the multi-functional IDRs, and they do not specifically focus on the disordered proteins or regions. In section 2.2.3 we introduced eleven predictors that predict functions of IDRs. However, these predictors focus on IDRs with individual functions, instead of IDRs that carry out multiple functions.

Current predictors of functions of IDRs do not address the prediction of DMRs, and current methods that predict moonlighting proteins cannot be applied at the region level. To this end, we propose the first-of-its-kind method that predicts DMRs from protein sequence. We name this method DMRpred (Disordered Moonlighting Region predictor).

## 5.2 Materials and methods

### 5.2.1 Datasets

The data to design and comparatively assess DMRpred comes from two sources: DisProt [58] and Protein Data Bank (PDB) [8]. We use DisProt 7.0.3 to collect disordered proteins and to extract annotations of DMRs. We use PDB to collect structured proteins that are necessary to ensure that our model does not predict DMRs for them. After removing 10 proteins from DisProt that have incorrect annotations (e.g., some annotations are out of bounds of the protein chains) we parsed the remaining 693 proteins. They include 2,108 disordered regions with length ranging between 5 and 2,400 residues. We define DMR as a region that is disordered and that has at least two distinct functions and/or binding partners. Section 5.2.2 explains how DMRs are defined. Structured regions are considered as non-disordered moonlighting regions (NDMRs). We define DMR residues and NDMR residues as residues that are in DMRs and NDMRs, respectively. The residues without annotations in DisProt and disordered regions without functional annotations are considered as unknown and are not used to either develop or test our model. We include all proteins that have annotated DMRs and proteins with residues annotated as either DMR or NDMR; we exclude proteins that contain only unknown annotations. Finally, we select 139 proteins from DisProt that have 12,910 DMR residues. We also collect high-resolution structured monomer proteins from PDB using the following criteria: chain length ≥ 30 residues; resolution ≤ 2.0 Å; number of chains (asymmetric unit) = 1; number of chains (biological assembly) = 1, and number of entities = 1. We collected 2,927 such monomers in February 2017. We filter out proteins that have non-standard amino acids (AAs) or disordered residues (missing residue or marked as REMARK 465). This ensures that the selected proteins contain only standard AAs and are structured. Next, we select a representative subset of these proteins that share low sequence similarity. We run

BlastClust [105] with the identity threshold = 25%. We choose one representative sequence from each of the 298 clusters to ensure that remaining proteins share low similarity. To balance the number of disordered and structured proteins, we randomly select 139 proteins from the set of 298 structured proteins. We combine the two sets of 139 proteins to form the dataset of 278 proteins. We divide these 278 proteins at random into two subsets of equal size, a training dataset that we use to design and parameterize the predictive model, and a test dataset to perform blind validation. We further subdivide the training dataset into four equally sized subsets (12.5% of the original dataset) to perform four-fold cross validation. We ensure that the training and test datasets as well as the four cross-validation folds share sequence identity below 25%. To do that, we run BlastClust on the 278 proteins with the identity threshold = 25%, and we place each of the resulting 263 clusters that include similar sequences (≥ 25% identity) into one of the five protein sets that is chosen at random. The first four subsets (12.5% of the original dataset) constitute the four folds of the training set and the remaining fifth subset (50% of the original dataset) is used as the test dataset. We ensure that each of the five subsets has similar ratio of DMR to NDMR residues by randomly resampling clusters, if needed. Finally, the training dataset includes 140 proteins with 6,261 DMR residues and 16,466 NDMR residues, and the test dataset includes 138 proteins with 6,449 DMR residues and 17,449 NDMR residues.

### 5.2.2 Definition of DMRs

We define DMR as a region that is disordered and that has at least two distinct functions and/or binding partners. Specifically, we define DMRs based on the annotations of IDRs for functions and binding partners in DisProt. The latest DisProt version 7.0.3 has two levels of annotations for functions, and one level of annotations for binding partners. The annotations for functions are shown in Table 5.1, and the annotations for binding partners include protein-protein, protein-DNA, protein-RNA, protein-lipid, protein-metal, protein-inorganic salt and protein-small molecule binding. For a given disordered region in DisProt, we consider this region as one of the three classes: 1) A disordered moonlighting regions (DMR); 2) A non-disordered moonlighting region (NDMR) that includes disordered regions that are not moonlighting and structured regions; and 3) A region of unknown type (UNK). UNK regions include disordered regions without functional annotation and regions in DisProt without any annotations. Residues in the UNK regions are excluded from our analysis. We do not use them to neither build nor assess the model. We include structured proteins collected from PDB [178] and residues from these proteins are annotated as the NDMR residues. Figure 5.1 details how DMR residues, NDMR residues and UNK residues are annotated. For example, the protein region annotated using the path at the bottom of the Figure 5.1 has at least two different functional annotations from different categories, and thus it is annotated as a DMR.

**Table 5.1.** Annotations of functions for IDRs in DisProt 7.0.3.

We exclude posttranslational modifications since these are not intrinsic functions of IDRs. They are located in IDRs and DisProt does not provide their exact position.

| Level 1 | Level 2 |
|---|---|
| Entropic chain | Flexible linker/spacer<br>Entropic bristle<br>Entropic clock<br>Entropic spring<br>Structural mortar<br>Self-transport through channel |
| Molecular recognition – assembler | Assembler<br>Localization (targeting)<br>Localization (tethering)<br>Prions (self-assembly, polymerization)<br>Liquid-liquid phase separation/demixing (self-assembly) |
| Molecular recognition – scavenger | Neutralization of toxic molecules<br>Metal binding/metal sponge<br>Water storage |
| Molecular recognition – effectors | Inhibitor<br>Disassembler<br>Activator<br>cis-regulatory elements (inhibitory modules)<br>DNA bending<br>DNA unwinding |
| Molecular recognition – display site | Limited proteolysis |
| Molecular recognition – chaperone | Protein detergent/solvate layer<br>Space filling<br>Entropic exclusion<br>Entropy transfer |

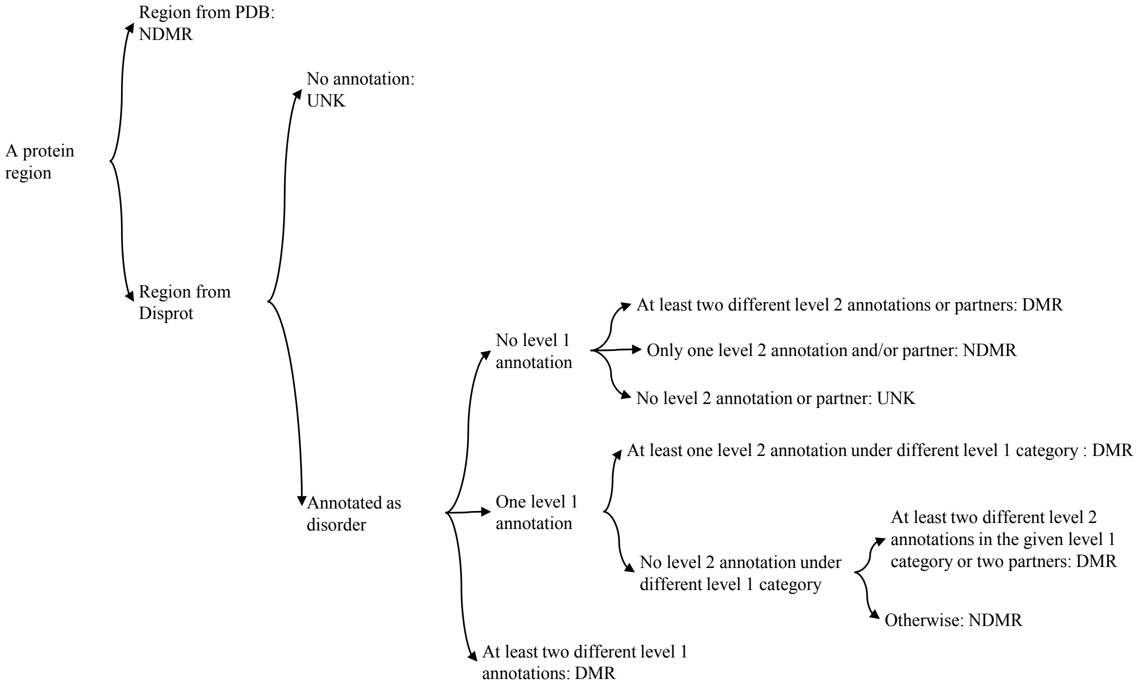

A protein region
Region from PDB: NDMR
Region from Disprot
No annotation: UNK
Annotated as disorder
No level 1 annotation
At least two different level 2 annotations or partners: DMR
Only one level 2 annotation and/or partner: NDMR
No level 2 annotation or partner: UNK
One level 1 annotation
At least one level 2 annotation under different level 1 category : DMR
No level 2 annotation under different level 1 category
At least two different level 2 annotations in the given level 1 category or two partners: DMR
Otherwise: NDMR
At least two different level 1 annotations: DMR

**Figure 5.1.** Flow chart to define disordered moonlighting regions (DMRs).

### 5.2.3 Overall design of DMRpred

The architecture of DMRpred (Figure 5.2) includes three layers: 1) *Sequence profile*: we represent the input sequence by a set of numeric values that quantify biophysical and structural properties of residues in this sequence; 2) *Feature representation*: for each residue in the input protein we convert the profile into a set of features that quantify the considered properties for this residue and its neighbors in the sequence and 3) *Prediction*: The features are input into a predictive model that generates the propensities for residues to be DMR residues.

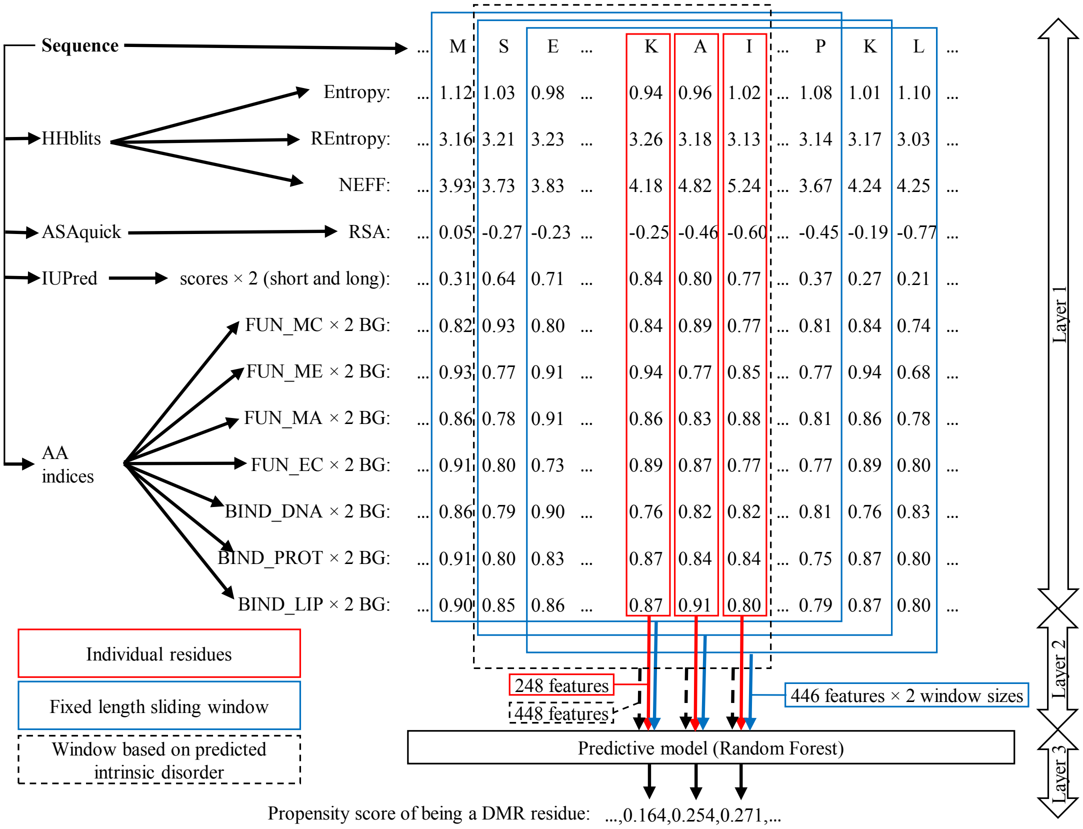

Sequence
... M S E ... K A I ... P K L ...
HHblits
Entropy: ... 1.12 1.03 0.98 ... 0.94 0.96 1.02 ... 1.08 1.01 1.10 ...
REntropy: ... 3.16 3.21 3.23 ... 3.26 3.18 3.13 ... 3.14 3.17 3.03 ...
NEFF: ... 3.93 3.73 3.83 ... 4.18 4.82 5.24 ... 3.67 4.24 4.25 ...
ASAquick
RSA: ... 0.05 -0.27 -0.23 ... -0.25 -0.46 -0.60 ... -0.45 -0.19 -0.77 ...
IUPred
scores × 2 (short and long): ... 0.31 0.64 0.71 ... 0.84 0.80 0.77 ... 0.37 0.27 0.21 ...
AA indices
FUN_MC × 2 BG: ... 0.82 0.93 0.80 ... 0.84 0.89 0.77 ... 0.81 0.84 0.74 ...
FUN_ME × 2 BG: ... 0.93 0.77 0.91 ... 0.94 0.77 0.85 ... 0.77 0.94 0.68 ...
FUN_MA × 2 BG: ... 0.86 0.78 0.91 ... 0.86 0.83 0.88 ... 0.81 0.86 0.78 ...
FUN_EC × 2 BG: ... 0.91 0.80 0.73 ... 0.89 0.87 0.77 ... 0.77 0.89 0.80 ...
BIND_DNA × 2 BG: ... 0.86 0.79 0.90 ... 0.76 0.82 0.82 ... 0.81 0.76 0.83 ...
BIND_PROT × 2 BG: ... 0.91 0.80 0.83 ... 0.87 0.84 0.84 ... 0.75 0.87 0.80 ...
BIND_LIP × 2 BG: ... 0.90 0.85 0.86 ... 0.87 0.91 0.80 ... 0.79 0.87 0.80 ...
Individual residues
Fixed length sliding window
Window based on predicted intrinsic disorder
248 features
448 features
446 features × 2 window sizes
Predictive model (Random Forest)
Propensity score of being a DMR residue: ...,0.164,0.254,0.271,...
Layer 1
Layer 2
Layer 3

**Figure 5.2.** Architecture of DMRpred.

### Sequence Profile

We consider several relevant biophysical and structural properties to define the sequence profile. They include sequence conservation, relative solvent accessibility, intrinsic disorder and a set of novel AA indexes. The indices quantify propensity of individual AA types to carry out functions that are relevant to DMRs.

We compute conservation from the multiple sequence alignment produced with HHblits [179]. HHblits is a profile based sequence alignments which was shown in [179] to be faster and more sensitive than the sequence-based alignment with PSI-BLAST [167]. To further reduce the runtime, we run HHblits against the Pfam database (as of February 2017), instead of the default UniProt20 database, and we iterate twice. Running HHblits against Pfam database is 12 times faster when compared to using UniProt20; average per protein runtime is 8 seconds vs. 102 seconds. Using the outputs of HHblits, we quantify the conservation in three ways: entropy, relative entropy (REntropy) and the local diversity (NEFF). The entropy is calculated using the 20 AA emission frequencies and relative entropy is calculated by considering the HHblits null model frequencies as the background frequency [180]. The NEFF($i$) output from HHblits measures the diversity of sub-alignment for residue $i$ that contains all sequences that have a residue at position $i$ of the full alignment. A smaller entropy, larger relative entropy and smaller NEFF indicate a more conserved residue. We invert the values of entropy and NEFF by subtracting their values from the corresponding maximal value. This makes these values consistent with the other properties, where a larger number indicates a more conserved residue that has a higher chance to carry out function(s) relevant to DMRs.

We calculate the relative solvent accessibility (RSA) with ASAquick [181]. ASAquick predicts the relative accessible surface area from a single sequence (without alignment). ASAquick is orders of magnitude faster than most of the other predictors of RSA that require multiple sequence alignment. It produces prediction in less than a second for a protein that is 500 residues long. We normalize the output of ASAquick to the 0 to 1 range where a larger number means that the corresponding residue is more solvent exposed.

Intrinsic disorder is predicted with IUPred [81, 82]. This fast predictor of IDRs was ranked as one of the top methods in several benchmarks [15, 77, 182]. We use both the short and long versions of IUPred. The output of IUPred ranges from 0 to 1 and a larger number suggests a higher likelihood for the intrinsic disorder.

A unique to DMRpred part of the profile is the propensity of AAs to carry out functions that are relevant to DMRs. We quantify these AA indices with Composition Profiler [183]. The indices measure enrichment or depletion of specific AA types in the corresponding functional IDRs. First, we extract all functional IDRs from the training dataset. We consider the functions that we use to define DMRs and that have at least 1000 residues; the latter ensures that we have enough data for statistical analysis. We cover seven functions: molecular recognition–chaperone (FUN_MC), molecular recognition–effectors (FUN_ME), molecular recognition–assembler (FUN_MA), entropic chain (FUN_EC), protein-DNA binding (BIND_DNA), protein-protein binding (BIND_PROT) and protein-lipid binding (BIND_LIP). For each of the seven functions we use the corresponding regions as a query to run the Composition Profiler. The Profiler compares a given query to a background. We consider two types of background (BG): all residues and disordered residues from the training dataset. The former type of background results in the computation of differences between a specific set of functional residues and a generic set of all AAs. The latter type focuses on the differences between a specific set of functional residues, which are disordered, and a set of all disordered AAs. For each of the 20 AAs, the Composition Profiler outputs a fractional difference of the composition between the query and the background. Positive (negative) fractional differences indicate enriched (depleted) AAs. The Profiler also outputs *p*-values that measure statistical significance of the fractional differences. We consider *p*-value < 0.01 as statistically significant. Tables 5.2 and 5.3 provide the fractional differences for the 20 AA types, the seven functions and two backgrounds.

**Table 5.2.** Fractional differences for the AA types for seven functions with all residues in the training dataset as background.

Columns correspond to functions/binding partners where FUN_MC: molecular recognition – chaperone; FUN_ME: molecular recognition – effectors; FUN_MA: molecular recognition – assembler; FUN_EC: entropic chain; BIND_DNA: protein-DNA binding; BIND_PROT: protein-protein binding; and BIND_LIP: protein-lipid binding. For each amino acid, we list its fractional difference (FD) value defined based on the difference in composition between the query sample (residues in a given functional region) and the background sample. Positive FD values indicate enrichment and negative value indicates depletion. Statistical significance of the fractional difference is quantified with the *p*-values; *p*-value < 0.01 is considered statistically significant. Bold font shows amino acids for which the FD values are significantly different.

| **Amino** | **FUN_MC** | | **FUN_ME** | | **FUN_MA** | | **FUN_EC** | | **BIND_DNA** | | **BIND_PROT** | | **BIND_LIP** | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Acid** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** |
| A | -0.119 | 0.205 | **0.420** | **0.000** | 0.126 | 0.115 | -0.032 | 0.531 | **0.211** | **0.003** | 0.111 | 0.030 | -0.208 | 0.040 |
| C | 0.119 | 0.561 | -0.391 | 0.068 | -0.054 | 0.786 | **-0.669** | **0.000** | **-0.395** | **0.003** | **-0.447** | **0.000** | -0.489 | 0.031 |
| D | -0.081 | 0.603 | 0.164 | 0.225 | 0.066 | 0.509 | -0.192 | 0.098 | **0.247** | **0.002** | 0.119 | 0.037 | **0.400** | **0.000** |
| E | **0.723** | **0.000** | 0.017 | 0.920 | 0.048 | 0.647 | **1.109** | **0.000** | 0.073 | 0.284 | **0.536** | **0.000** | 0.334 | 0.001 |
| F | -0.228 | 0.198 | **-0.514** | **0.001** | -0.164 | 0.191 | **-0.579** | **0.000** | **-0.409** | **0.000** | **-0.419** | **0.000** | -0.073 | 0.716 |
| G | -0.202 | 0.048 | 0.376 | 0.000 | **0.233** | **0.003** | 0.042 | 0.610 | -0.143 | 0.029 | **0.129** | **0.007** | **0.473** | **0.000** |
| H | 0.411 | 0.028 | 0.138 | 0.517 | 0.089 | 0.597 | -0.304 | 0.072 | 0.007 | 0.941 | 0.164 | 0.061 | 0.584 | 0.011 |
| I | -0.175 | 0.222 | **-0.388** | **0.002** | **-0.449** | **0.000** | -0.178 | 0.148 | **-0.307** | **0.001** | **-0.359** | **0.000** | -0.244 | 0.106 |
| K | **0.605** | **0.000** | -0.076 | 0.386 | **0.445** | **0.000** | **0.292** | **0.005** | **1.123** | **0.000** | **0.388** | **0.000** | **0.370** | **0.001** |
| L | -0.089 | 0.385 | 0.075 | 0.415 | -0.195 | 0.026 | **-0.236** | **0.004** | **-0.326** | **0.000** | **-0.238** | **0.000** | -0.250 | 0.021 |
| M | 0.004 | 0.912 | -0.426 | 0.017 | -0.168 | 0.225 | -0.355 | 0.022 | -0.160 | 0.176 | **-0.372** | **0.000** | -0.324 | 0.066 |
| N | -0.242 | 0.087 | **-0.352** | **0.004** | -0.203 | 0.052 | -0.183 | 0.144 | **-0.269** | **0.003** | **-0.318** | **0.000** | -0.385 | 0.012 |
| P | 0.128 | 0.296 | **0.293** | **0.006** | 0.136 | 0.128 | 0.305 | 0.016 | 0.144 | 0.074 | **0.372** | **0.000** | 0.207 | 0.064 |
| Q | -0.170 | 0.201 | 0.155 | 0.296 | -0.184 | 0.095 | **0.358** | **0.002** | -0.148 | 0.091 | 0.059 | 0.306 | 0.003 | 0.897 |
| R | 0.252 | 0.157 | 0.142 | 0.360 | 0.237 | 0.039 | 0.011 | 0.987 | **0.316** | **0.000** | 0.025 | 0.752 | **-0.283** | **0.008** |
| S | **-0.338** | **0.002** | **0.334** | **0.001** | **0.284** | **0.001** | 0.210 | 0.015 | **0.246** | **0.000** | **0.218** | **0.000** | -0.013 | 0.816 |
| T | 0.006 | 0.942 | **-0.310** | **0.007** | -0.123 | 0.188 | -0.033 | 0.658 | **-0.216** | **0.005** | -0.095 | 0.075 | -0.133 | 0.212 |
| V | -0.002 | 0.930 | **-0.433** | **0.000** | **-0.317** | **0.001** | **-0.290** | **0.005** | **-0.218** | **0.004** | **-0.310** | **0.000** | -0.323 | 0.013 |
| W | -0.318 | 0.189 | -0.493 | 0.036 | **-0.623** | **0.002** | -0.469 | 0.022 | **-0.724** | **0.000** | **-0.538** | **0.000** | 0.021 | 0.751 |
| Y | **-0.552** | **0.002** | -0.324 | 0.061 | -0.143 | 0.364 | **-0.609** | **0.000** | **-0.458** | **0.000** | **-0.458** | **0.000** | -0.064 | 0.733 |

**Table 5.3.** Fractional differences for the AA types for seven functions with disordered residues in the training dataset as background.

Columns correspond to functions/binding partners where FUN_MC: molecular recognition – chaperone; FUN_ME: molecular recognition – effectors; FUN_MA: molecular recognition – assembler; FUN_EC: entropic chain; BIND_DNA: protein-DNA binding; BIND_PROT: protein-protein binding; and BIND_LIP: protein-lipid binding. For each amino acid, we list its fractional difference (FD) value defined based on the difference in composition between the query sample (residues in a given functional region) and the background sample. Positive FD values indicate enrichment and negative value indicates depletion. Statistical significance of the fractional difference is quantified with the *p*-values; *p*-value < 0.01 is considered statistically significant. Bold font shows amino acids for which the FD values are significantly different.

| **Amino** | **FUN_MC** | | **FUN_ME** | | **FUN_MA** | | **FUN_EC** | | **BIND_DNA** | | **BIND_PROT** | | **BIND_LIP** | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Acid** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** | **FD** | ***p*-value** |
| A | -0.221 | 0.035 | 0.257 | 0.010 | -0.001 | 0.872 | -0.147 | 0.096 | 0.068 | 0.361 | -0.018 | 0.834 | **-0.299** | **0.005** |
| C | **3.385** | **0.000** | **1.368** | **0.004** | **2.626** | **0.000** | 0.279 | 0.523 | **1.352** | **0.001** | **1.136** | **0.001** | 1.013 | 0.031 |
| D | -0.220 | 0.085 | -0.012 | 0.744 | -0.097 | 0.333 | **-0.314** | **0.004** | 0.058 | 0.580 | -0.051 | 0.413 | 0.179 | 0.076 |
| E | 0.154 | 0.076 | **-0.317** | **0.001** | **-0.292** | **0.000** | **0.417** | **0.000** | **-0.275** | **0.000** | 0.036 | 0.585 | -0.099 | 0.533 |
| F | -0.067 | 0.783 | -0.418 | 0.017 | 0.007 | 0.963 | **-0.496** | **0.002** | -0.287 | 0.029 | **-0.300** | **0.004** | 0.105 | 0.514 |
| G | **-0.405** | **0.000** | 0.022 | 0.548 | -0.081 | 0.477 | -0.220 | 0.010 | **-0.362** | **0.000** | **-0.161** | **0.004** | 0.096 | 0.789 |
| H | -0.016 | 0.990 | -0.208 | 0.172 | -0.243 | 0.089 | **-0.512** | **0.000** | -0.300 | 0.012 | -0.190 | 0.051 | 0.097 | 0.834 |
| I | 0.257 | 0.188 | -0.067 | 0.611 | -0.165 | 0.246 | 0.254 | 0.130 | 0.051 | 0.645 | -0.026 | 0.817 | 0.152 | 0.347 |
| K | -0.023 | 0.848 | **-0.432** | **0.000** | -0.116 | 0.107 | -0.213 | 0.012 | **0.301** | **0.000** | **-0.148** | **0.007** | -0.163 | 0.116 |
| L | 0.379 | 0.011 | **0.623** | **0.000** | 0.214 | 0.060 | 0.148 | 0.282 | 0.014 | 0.885 | 0.148 | 0.079 | 0.133 | 0.272 |
| M | 0.078 | 0.782 | -0.385 | 0.058 | -0.106 | 0.515 | -0.307 | 0.086 | -0.092 | 0.544 | -0.325 | 0.005 | -0.269 | 0.168 |
| N | 0.466 | 0.045 | 0.243 | 0.342 | 0.526 | 0.011 | **0.576** | **0.004** | 0.403 | 0.011 | 0.308 | 0.023 | 0.188 | 0.260 |
| P | 0.093 | 0.505 | 0.254 | 0.037 | 0.099 | 0.328 | 0.256 | 0.082 | 0.106 | 0.300 | **0.330** | **0.000** | 0.171 | 0.171 |
| Q | -0.153 | 0.274 | 0.172 | 0.301 | -0.169 | 0.167 | **0.386** | **0.006** | -0.133 | 0.202 | 0.075 | 0.382 | 0.022 | 0.982 |
| R | 0.225 | 0.254 | 0.113 | 0.499 | 0.208 | 0.108 | -0.011 | 0.895 | **0.288** | **0.008** | -0.001 | 0.999 | -0.310 | 0.010 |
| S | **-0.380** | **0.001** | 0.255 | 0.014 | 0.209 | 0.029 | 0.141 | 0.132 | 0.171 | 0.036 | 0.146 | 0.031 | -0.069 | 0.479 |
| T | 0.172 | 0.227 | -0.197 | 0.164 | 0.019 | 0.886 | 0.127 | 0.369 | -0.090 | 0.414 | 0.054 | 0.510 | 0.014 | 0.964 |
| V | **0.478** | **0.004** | -0.153 | 0.241 | 0.010 | 0.980 | 0.052 | 0.673 | 0.161 | 0.191 | 0.025 | 0.797 | 0.008 | 0.786 |
| W | 0.528 | 0.306 | 0.103 | 0.793 | -0.170 | 0.619 | 0.183 | 0.768 | -0.388 | 0.144 | 0.020 | 0.997 | **1.255** | **0.003** |
| Y | -0.283 | 0.244 | 0.071 | 0.647 | 0.356 | 0.075 | -0.378 | 0.048 | -0.144 | 0.383 | -0.140 | 0.292 | 0.481 | 0.051 |

### Feature representation

Using the sequence profile, we empirically generate a rich set of features to represent every residue in the input sequence. The features quantify information about individual biophysical and structural properties and their combinations, e.g., we combine conservation and solvent accessibility. We generate features for each residue by considering the information about the residue itself and its neighbors in the sequence. The use of the neighboring residues is inspired by the fact that the DMR residues form regions composed of consecutive AAs that share certain functional and structural properties. We define neighbors using two types of sequence windows: a sliding window of a fixed length (defined based on size of native DMRs in the training dataset) centered on the residue that we currently predict; and the putative disordered regions (disordered window) that includes this residue. To the best of our knowledge, we are the first to use the latter window type. We do not pad windows for residues at the termini of the sequence and accordingly the features are normalized by the length of the window. The length of the second type of the windows varies and is determined by the length of the putative disordered regions generated with IUPred_short and IUPred_long. The use of the fixed size sliding windows is motivated by the design of related methods, such as MoRFpred [50], fMoRFpred [118], DisoRDPbind [51] and DFLpred [184]. Using the individual and combined biophysical and structural properties that are quantified for individual residues and based on the two types of windows, we compute 1588 features for each residue in the input protein chain. A detailed description of these features can be found in Appendix B.

### Design of the predictive model

We use the feature vector for each of the 22,727 residues in the training dataset to generate a predictive model using a machine learning algorithm. This model outputs a propensity score that a given residue is a DMR residue. We consider three classifiers: Logistic Regression, Naive Bayes and Random Forest using their implementations in Weka.

We conduct feature selection for Logistic Regression and Naive Bayes. Random Forest automatically selects features when building the trees. We use the best-first search to implement the selection. First, we calculate the AUC values when using individual features to make predictions on each of the four training folds, and we rank the features by their averaged (over the four training folds) AUC values. We run the 4-folds cross validation on the training dataset using Logistic Regression and Naive Bayes with the top-ranked feature to initialize the set of selected features. Next, we add the next-ranked feature to the current feature set if this results in a higher average AUC than the AUC obtained before the feature was added. We scan the sorted feature set once. We perform a grid search to find optimal parameters for the Random Forest. We select parameters that result in the highest AUC measured with the 4-fold cross validation on the training dataset. Based on suggestions from [185], we consider the number of trees = {$2^7$, $2^8$, $2^9$, $2^{10}$}, the number of features randomly selected for each tree node = {$\log_2(N)$, sqrt($N$)} where $N$ is the total number of features = 1588, and % of samples for each tree node (bag percent) = {20%, 30%, 40%, 50%}. There are total of 32 combinations of parameter values.

Table 5.4 summarizes the results that correspond to the highest AUC based on the cross validation on the training dataset for the three classifiers. We report the average accuracy, precision, sensitivity, MCC, AUC and $\text{AUC}_{ratio}$ over the 4 cross validation folds. Accuracy, precision, sensitivity, MCC, and $\text{AUC}_{ratio}$ are calculated at the 5% false positive rate. We implement DMRpred using Random Forest that secures the best value for all measures. The parameters that were used to generate this model are: number of trees = 512, number of features per tree node = 39, and bag percent = 30%.

**Table 5.4.** Results based on 4-fold cross validation on the training dataset.

| Classifier | Accuracy | Precision | Sensitivity | MCC | AUC | $\text{AUC}_{ratio}$ |
|---|---|---|---|---|---|---|
| Random Forest | 0.837 | 0.803 | 0.536 | 0.560 | 0.868 | 15.314 |
| Logistic Regression | 0.813 | 0.772 | 0.452 | 0.488 | 0.867 | 11.275 |
| Naive Bayes | 0.769 | 0.603 | 0.414 | 0.358 | 0.795 | 4.140 |

### 5.2.4 Analysis of the DMRpred's predictive model

DMRpred combines sequence conservation, predicted RSA, putative IDRs and AA indices that quantify propensity for functions that are relevant to DMRs to define the sequence profile. It also uses two types of windows to generate features: sliding windows and windows based on predicted IDRs. We assess contributions of different parts of the profile and different window types to the predictive performance of our model. To do that we run the 4-fold cross validation on the training dataset with Random Forest that excludes features that utilize a given part of the profile or a given type of window. For each subset of features we run a grid search defined in Section 5.2.3 to parametrize the Random Forest model. Table 5.5 summarizes results for each of these configurations. We report the average AUC, accuracy, precision, sensitivity, MCC, AUC and AUC$_{ratio}$ computed over the 4 folds. We also evaluate statistical significance of the differences between DMRpred and each of the other configurations. We bootstrap the cross-validation results by randomly selecting 50% proteins 100 times. If the 100 pairs of measurements are draw from normal distribution then we use the Student's paired *t*-test, otherwise we use the Wilcoxon signed-rank test.

DMRpred outperforms all other configurations. Accuracy, precision, sensitivity, MCC, AUC and AUC$_{ratio}$ drop significantly for all other configurations when compared to DMRpred, except when the sliding windows are not used when the decrease in the predictive performance in not significant. This means that all elements of the sequence profile as well as the use of the disorder region-based windows significantly contribute to the DMRpred's predictive performance. Based on the magnitudes of the decrease in AUCs, the most relevant information for the prediction of DMRs includes the putative intrinsic disorder, the use of both types of windows to compute features, and sequence conservation. These factors are well-grounded in the characteristics of DMRs that are by definition disordered and include functional residues that are typically highly conserved. The windows are needed to capture differences in the intrinsic characteristics of DMRs, which form segments in the sequence, and the residues that surround these regions.

**Table 5.5.** Comparison of designs using subsets of features.

The first row shows the design using full feature set. Following rows show designs that do not use: predicted RSA (No RSA), sliding windows (No SWIN), AA indices (No AAI), windows based on predicted IDRs (No IDWIN), sequence conservation (No CON), any windows (No WIN) and putative intrinsic disorder (No ID). The results are based on bootstrapping cross validation on the training dataset and are ranked by the AUC value in descending order. + means that DMRpred is significantly better than a given configuration ($p$-value < 0.01).

| Feature set | Accuracy | Precision | Sensitivity | MCC | AUC | AUC$_{ratio}$ |
|---|---|---|---|---|---|---|
| DMRpred | 0.837 | 0.803 | 0.536 | 0.560 | 0.868 | 15.314 |
| No RSA | 0.810$^{+}$ | 0.770$^{+}$ | 0.436$^{+}$ | 0.476$^{+}$ | 0.861$^{+}$ | 13.387$^{+}$ |
| No SWIN | 0.823 | 0.784 | 0.493 | 0.521 | 0.856$^{+}$ | 14.941 |
| No AAI | 0.816$^{+}$ | 0.770$^{+}$ | 0.461$^{+}$ | 0.493$^{+}$ | 0.854$^{+}$ | 13.457$^{+}$ |
| No IDWIN | 0.793$^{+}$ | 0.731$^{+}$ | 0.383$^{+}$ | 0.420$^{+}$ | 0.854$^{+}$ | 10.092$^{+}$ |
| No CON | 0.787$^{+}$ | 0.707$^{+}$ | 0.356$^{+}$ | 0.391$^{+}$ | 0.823$^{+}$ | 10.375$^{+}$ |
| No WIN | 0.782$^{+}$ | 0.711$^{+}$ | 0.340$^{+}$ | 0.381$^{+}$ | 0.818$^{+}$ | 7.450$^{+}$ |
| No ID | 0.773$^{+}$ | 0.686$^{+}$ | 0.307$^{+}$ | 0.346$^{+}$ | 0.781$^{+}$ | 8.679$^{+}$ |

## 5.3 Results

### 5.3.1 Comparison with alternative approaches to predict DMRs

We compare the predictive performance of DMRpred with several alternative approaches that could be used to identify DMRs. Since DMRs are a subset of IDRs we include a popular predictor of disordered regions, Espritz [186]. We use the three version of Espritz that were designed based on the three main sources of disorder annotations: NMR structures (Espritz_NMR), X-ray structures (Espritz_X-ray) and from DisProt database (Espritz_DisProt). We also include four representative methods that predict specific types of functional IDRs. They include DisoRDPbind [51] that predicts disordered protein-DNA, protein-RNA and protein-protein binding regions, Anchor [104] that generates putative disordered protein-binding regions, and two methods that predict molecular recognition features (MoRFs): MoRFpred [50] and fMoRFpred [118]. MoRFs are protein-binding IDRs that undergo disordered-to-order transition upon interaction. Inclusion of these four methods is motivated by the fact that DMRs carry out multiple functions that include binding to proteins and nucleic acids. Moreover, DMRs should include evolutionarily conserved residues. Thus, we include sequence conservation computed from alignments produced with HHblits [179]. We also include a default approach of transferring DMR annotations via sequence alignment. We align a given test protein to all proteins from the training dataset using PSI-BLAST with default parameters

[167], and we transfer the annotations from the matched residues in the most similar training chain. Finally, we use residues levels annotations of protein functions generated with Eukaryotic Linear Motif (ELM) resource. We assume that residues that have ≥ 2 types of ELM motifs and have high putative disorder scores constitute putative DMRs.

We use the corresponding author-provided webservers or implementations to run these tools. We use the MoRFpred and fMoRFpred webservers to collect their predictions that we utilize as the propensities for DMR residues. We utilize standalone software for Anchor and Espritz and the DisoRDPbind's webserver to obtain its three propensity scores for protein-protein, protein-DNA and protein-RNA binding. Because DMRs carry out multiple functions, we combine two or three DisoRDPbind's scores to represent the propensity that a given residue binds multiple partners. We combine the scores in two ways: as average of the two highest scores among the three scores DisoRDPbind produces; and as average the three scores. We run HHblits for each sequence in the test set against the default UniProt20 database to compute the conservation scores. We calculate entropy and relative entropy from the 20 AA emission frequencies and use the NEFF($i$) scores for each residue $i$ that are directly output by HHblits to produce three estimates of conservation. We run PSI-BLAST with default parameters for proteins in the test dataset against proteins in the training dataset and copy annotations for positions with identical residues or conservative substitutions in the alignment. We run the ELM resource server [143] and parse the HTML files to obtain the six types of ELM annotations: CLV, DEG, DOC, LIG, MOD and TRG. If a residues is annotated to have ≥ 2 types of ELM terms, we use the Espritz_NMR score (this method is the best performing disorder predictor in our assessment) to represent its propensity of being a DMR residue, otherwise we assign a score of zero.

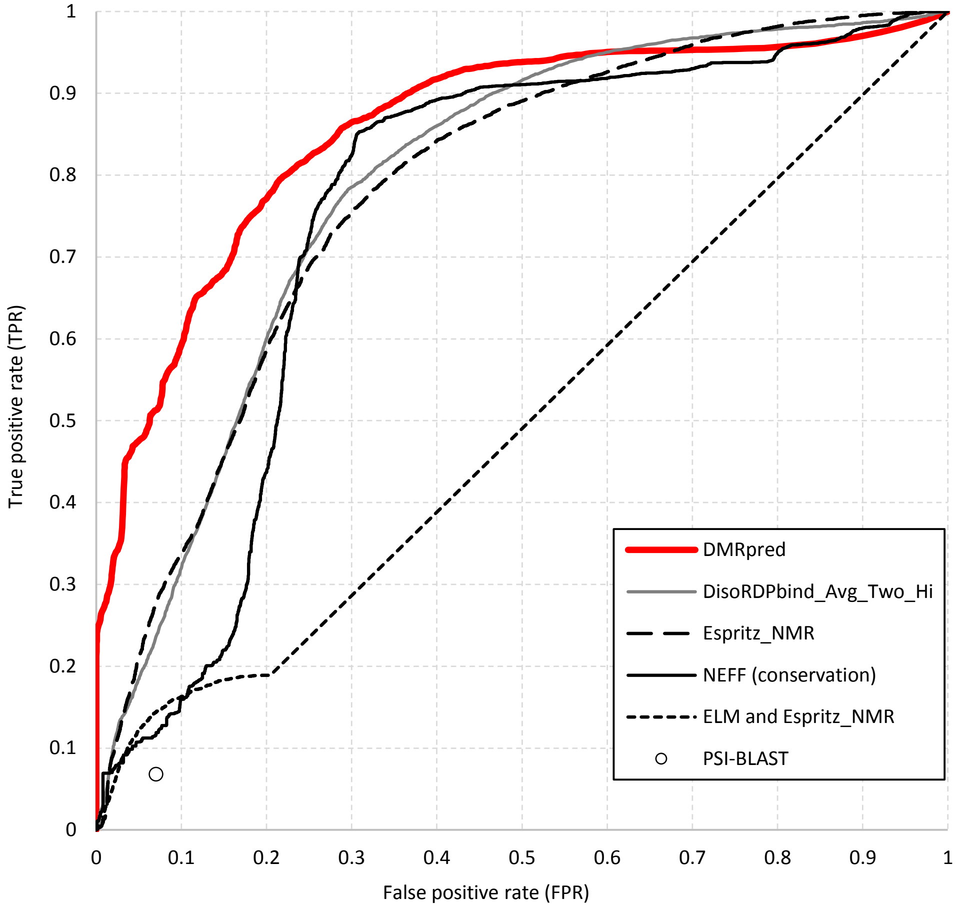


**Figure 5.3.** ROC curves for DMRpred and other methods with high AUCs.

Table 5.6 compares the predictive performance of these methods with DMRpred on the test dataset. We sort the methods by AUC within each group defined by their prediction target: DMRs, functional IDRs, all IDRs, sequence conservation and MoRF regions. We rank across these groups based on their highest AUCs. We assess statistical significance of the differences between the predictive performance of DMRpred and each of the other 13 methods. We bootstrap the results by randomly selecting 50% of test proteins 100 times. If the 100 pairs of measurements are draw from normal distribution, we use the Student's paired *t*-test, otherwise we use the Wilcoxon signed-rank test. We show the ROC curves for DMRpred and the methods with the highest AUC from each category (except for MoRFs that have AUC < 0.5) in Figure 5.3. The result for PSI-BLAST is shown as a dot as it only provides a binary prediction.

**Table 5.6.** Comparison of DMRpred with alternative methods on the test dataset.

+ means that DMRpred is significantly better than a given other method ($p$-value < 0.01). Accuracy, precision, sensitivity and MCC are calculated at 5% false positive rate. AUC$_{ratio}$ is calculated at false positive rate ≤ 5%. Best results are shown in bold font. NA (not available) is due to the fact that PSI-BLAST provides only the binary predictions.

| **Prediction target** | **Methods** | **Accuracy** | **Precision** | **Sensitivity** | **MCC** | **AUC** | **AUC$_{ratio}$** |
|---|---|---|---|---|---|---|---|
| DMRs | DMRpred | **0.820** | **0.788** | **0.474** | **0.511** | **0.858** | **14.555** |
| | ELMs and ESpritz_NMR | 0.721$^{+}$ | 0.481$^{+}$ | 0.121$^{+}$ | 0.125$^{+}$ | 0.502$^{+}$ | 2.607$^{+}$ |
| | PSI-BLAST | 0.692$^{+}$ | 0.269$^{+}$ | 0.068$^{+}$ | -0.004$^{+}$ | NA | NA |
| Functional IDRs | DisoRDPbind_AvgTwoHigh | 0.739$^{+}$ | 0.584$^{+}$ | 0.184$^{+}$ | 0.213$^{+}$ | 0.790$^{+}$ | 4.270$^{+}$ |
| | DisoRDPbind_AvgThree | 0.726$^{+}$ | 0.512$^{+}$ | 0.137$^{+}$ | 0.149$^{+}$ | 0.775$^{+}$ | 2.757$^{+}$ |
| | Anchor | 0.734$^{+}$ | 0.562$^{+}$ | 0.169$^{+}$ | 0.193$^{+}$ | 0.746$^{+}$ | 3.681$^{+}$ |
| All IDRs | Espritz_NMR | 0.746$^{+}$ | 0.616$^{+}$ | 0.210$^{+}$ | 0.245$^{+}$ | 0.781$^{+}$ | 4.369$^{+}$ |
| | Espritz_X-ray | 0.741$^{+}$ | 0.595$^{+}$ | 0.193$^{+}$ | 0.224$^{+}$ | 0.739$^{+}$ | 3.573$^{+}$ |
| | Espritz_DisProt | 0.694$^{+}$ | 0.154$^{+}$ | 0.024$^{+}$ | -0.058$^{+}$ | 0.663$^{+}$ | 0.411$^{+}$ |
| Sequence conservation | NEFF | 0.719$^{+}$ | 0.461$^{+}$ | 0.107$^{+}$ | 0.108$^{+}$ | 0.754$^{+}$ | 1.552$^{+}$ |
| | Entropy | 0.706$^{+}$ | 0.333$^{+}$ | 0.065$^{+}$ | 0.030$^{+}$ | 0.699$^{+}$ | 2.817$^{+}$ |
| | Relative entropy | 0.705$^{+}$ | 0.318$^{+}$ | 0.061$^{+}$ | 0.022$^{+}$ | 0.698$^{+}$ | 1.042$^{+}$ |
| MoRF regions | fMoRFpred | 0.707$^{+}$ | 0.284$^{+}$ | 0.041$^{+}$ | 0.004$^{+}$ | 0.474$^{+}$ | 0.955$^{+}$ |
| | MoRFpred | 0.703$^{+}$ | 0.298$^{+}$ | 0.055$^{+}$ | 0.012$^{+}$ | 0.470$^{+}$ | 1.382$^{+}$ |

DMRpred offers the best predictive performance. Table 5.6 shows that it significantly outperforms all other methods for all considered measures ($p$-value < 0.01). DMRpred's $AUC_{ratio}$ = 14.6 which means that its AUC for predictions at low FPR (≤ 5%) is about 14.6 times better than random. This represents 3.3 fold improvement over the second best Espritz_NMR that secures $AUC_{ratio}$ = 4.4. DMRpred's AUC = 0.86; this high value is reflected in the ROC curve shown in Figure 2. We note a large gap between ROC for DMRpred and the other methods for FPRs ≤ 0.2. High FPRs are not practical since they result in the number of false positives that is higher than the numbers of true positive; this is because only about 27% of residues in the test dataset are DMR residues. Interestingly, DMRpred has a steep ROC curve for very low FPRs. It finds 15.1% of native DMR residues without producing any false positives. DMRpred's accuracy = 82% and precision = 78.8%, which means it correctly predicts 82% of residues and 78.8% of the putative DMR residues. These results and the accuracy, precision, sensitivity, and MCC values of all methods are calibrated to the false positive rate (FPR) = 5%. In contrast, the other approaches make correct predictions for between 70% and 75% of residues, and between 15% and 62% of the predicted DMR residues. DMRpred also secures much higher sensitivity, MCC, and $AUC_{ratio}$ when compared to the other methods. Sensitivity = 47.4% means it correctly finds 47.4% of native DMR residues when its FPR = 5%, i.e., the fraction of NDMR residues incorrectly predicted as DMR is only 5%. DMRpred's MCC is slightly above 0.5, which indicates strong correlation between the predicted DMR annotations and the native DMR annotations.

The three versions of Espritz offer relatively low predictive performance because they predict all IDRs irrespective of their function(s), while majority of IDRs are not DMRs. Correspondingly, these methods over-predict DMRs. For instance, for the same predicted positive rate = 25% (number of predicted DMR residues divided by number of native DMR residues), the three Espritz versions generate between 613 and 1,254 false positives while DMRpred generates only 29 false positives. DisoRDPbind, Anchor, MoRFpred and fMoRFpred generate IDRs that interact with DNA, RNA or proteins, instead of multi-functional DMRs that may also implement functions that do not involve binding to nucleic acids and proteins (e.g., entropic regions and metal-binding regions). The relatively low sensitivity of these four methods compared to DMRpred suggest that they find only a small subset of DMRs. The low predictive performance of the conservation scores (MCC < 0.11 and sensitivity < 0.11) suggests that using the evolutionary conservation alone is not sufficient to separate DMRs and NDMRs. This is because many of the NDMR residues could be conserved, including residues that interact with one ligand and residues that are crucial for structural integrity of the protein fold. Given that by definition DMR are multi-functional disordered regions, we combined functional annotations generated with the ELMs and the disorder prediction to generate putative DMRs. However, this prediction is characterized by relatively low predictive performance because ELMs cover only a small portion of native functional residues, which is evident based on the corresponding shape of the ROC curve (Figure 5.3) and low sensitivity (Table 5.6). PSI-BLAST does not provide reliable prediction because the test proteins share low sequence similarity with the training proteins. This confirms the fact that the performance by default alignment-based predictions are not going to be successful unless enough of functionally annotated disordered proteins that are sufficiently (highly) similarity to the test/query protein are available.

### 5.3.2 Case study

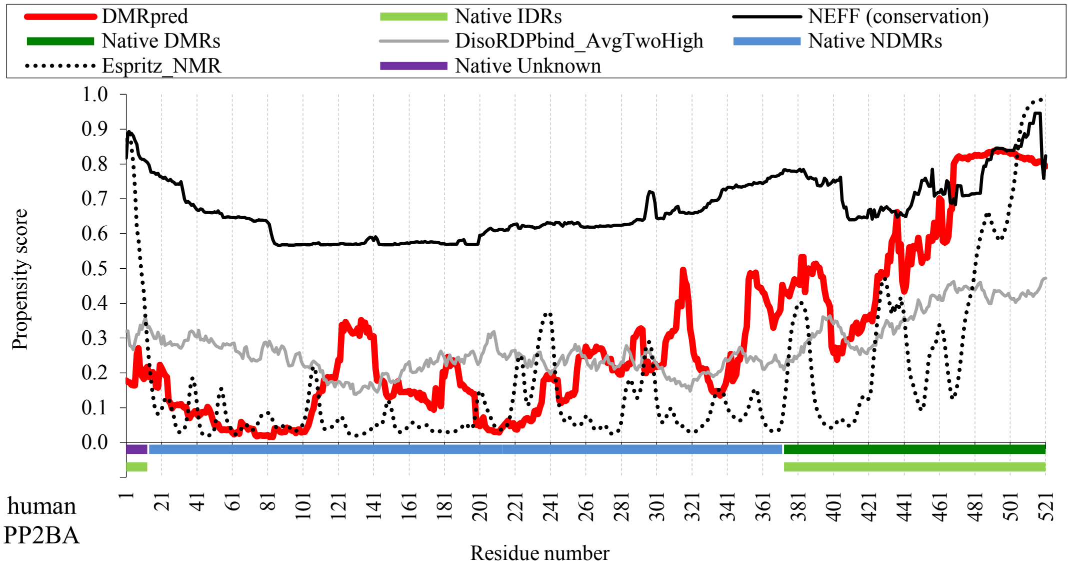


**Figure 5.4.** Predictions for human PP2BS protein by DMRpred and alternative methods.

The horizontal lines at the bottom show native disordered regions (IDRs; light green), native DMRs (dark green), NDMRs (blue) and unknown regions (violet). We include outputs from DMRpred (thick red line), DisoRDPbind (gray), Espritz (dotted black) and conservation scores (NEFF) from HHblits (solid black).

We use the serine/threonine-protein phosphatase 2B (PP2BS protein, DisProt ID: DP00092) from the test dataset to illustrate predictions by DMRpred. This protein has two IDRs, one at the N-terminus (positions 1 to 13 [187]), and the other at the C-terminus (positions 373 to 521 [187]). The first IDR has no functional annotations or binding partners in DisProt and by our definition it is annotated as unknown (neither DMR nor NDMR). The second IDR includes a protein-binding region where calmodulin binds (positions 373 to 468) [187-189] and an auto-inhibitory domain (positions 371 and 511) [188, 189], which define it as DMR. Figure 5.4 plots the outputs of DMRpred, and other methods with the highest AUC for a given prediction target (Table 5.6), except MoRF predictors that have AUC < 0.5. DMRpred's scores (red line) at the C-terminus are high, which correctly suggests a DMR there. The scores for the structured catalytic domain (positions 14 to 373; blue horizontal line) and the N-terminus are low, suggesting that there are no DMRs there. We argue that DMRpred's prediction for the IDR at the N-terminus is possibly correct, given that this extensively studied protein does not yet have a functional annotation for this short region.

Espritz (black dotted line) correctly identifies the IDR at the N-terminus and also partially predicts the IDR at the C-terminus, highlighting our observation that these predictors are likely to over-predict DMRs. Moreover, Espritz's prediction at the N-terminus is not as accurate as the corresponding output from DMRpred. Interestingly, the average of the two highest scores from DisoRDPbind (gray line) fails to identify the native DMR, although we observe a slight increase in the scores at the N-terminus due to higher values for its protein-binding predictions. Finally, conservation scores (black solid line) are not suitable to identify DMRs since they point to several highly conserved regions that do not line up with DMRs. Overall, we conclude that DMRpred offers reasonably accurate predictions for this protein that cannot be substituted with outputs of the other predictors.

### 5.3.3 Prediction and analysis of DMRs in the human proteome

We characterize putative DMRs and IDRs in the complete reviewed human proteome that we collected from UniProt [61]. We use the consensus-based disorder predictions from MobiDB [34] to analyze IDRs. We use 19,917 human proteins after removing about 200 proteins that could not be mapped to MobiDB. We make predictions with DMRpred and annotate DMR residues based on the binary predictions that are calibrated to produce 5% false positive rate on the test dataset, i.e., residues with propensities ≥ 0.761 as assumed as DMR residues. We annotate putative DMRs as segments of at least four consecutive DMR residues. This is in line with the definition of all IDRs that are expected to include at least four consecutive amino acids [12, 15]. We found about 32 thousand putative DMRs in the human proteome, which corresponds to around 30% of the 107 thousand putative IDRs. This is similar to the 37% rate of DMRs among the IDRs included in DisProt, which was reported in [182]. We analyze long (≥ 30 consecutive residues) putative DMRs and IDRs since they are recognized as a distinct class of biologically functional domains [100, 190, 191]. Figure 5.5 suggests that about 53% of human proteins have at least one long IDR. This agrees with recent estimates that ranged between about 45% [100] and 50% [5]. Interestingly, we show that about 25% of human proteins may have at least one long DMR, and 8% may have three or more long DMRs.

We also count the number of DMR and disordered residues in each protein and the number and length of each DMR and disordered region. We plot the distribution of the content of DMR and disordered residues and distribution of lengths of DMRs and disordered regions (IDRs) in Figures 5.6 and 5.7, respectively. Figure 5.6 shows that about 29% human proteins have DMRs. The content of DMR residues among the remaining 71% proteins is below 5%. These are considered spurious predictions since that the false positive rate of DMRpred is estimated to be 5%. To compare, about 81% of human proteins have disordered residues, i.e., their disorder content > 5%. Such substantial difference in the rate of disorder vs. DMR content is reasonable given that only a small fraction of intrinsically disordered regions (IDRs) are DMRs. About 11% of human proteins are predicted to have at least modest content of DMRs (> 30% of their residues are in DMRs) and 4.6% to have large content (> 50%). To compare, Figure 5.6 reveals that three times as many proteins (approximately 31%) have > 30% disorder content. The latter result is in good agreement with a recent analysis in  [100] where about 31.5% of proteins were predicted to have at least 30% of disordered residues. The histograms of lengths of DMRs and IDRs given in Figure 5.7 follow the same trend, where there are gradually fewer regions that are longer. The main differences are the overall number of regions that, as expected, is much lower in the case of DMR, and the rate of decline that is again lower for DMRs; see black bars (for IDRs) and gray bars (for DMRs) in the Figure 5.7. Interestingly, our analysis reveals that most of the very long disordered regions are possibly DMRs, given that the number of regions longer than 150 residues is similar when comparing IDRs and DMRs.

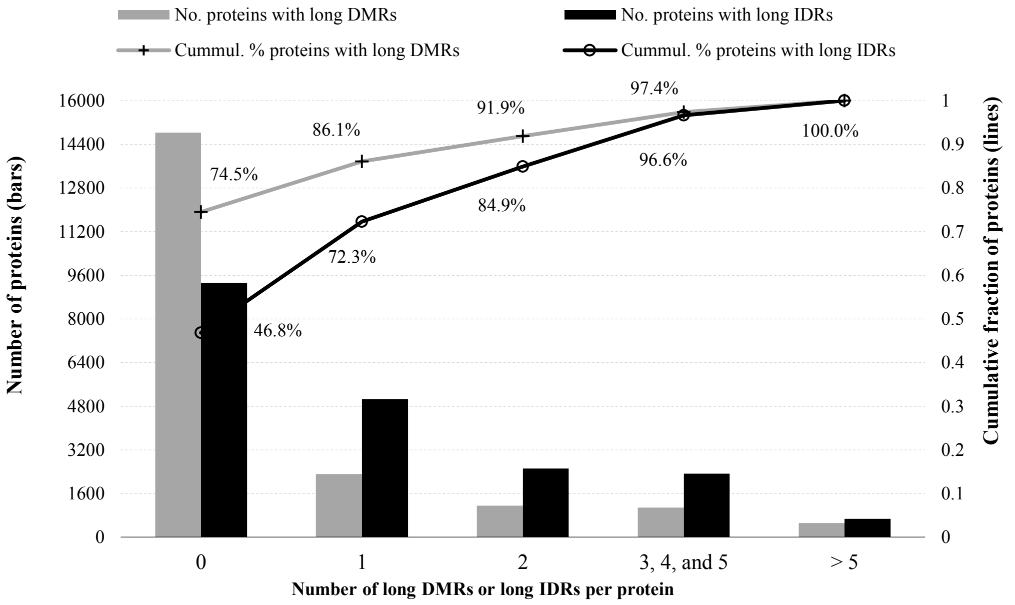


**Figure 5.5.** Number of long putative DMRs and IDRs for the complete reviewed human proteome.

Bars show number of proteins that have the number of long regions given on the *x*-axis. Lines show the corresponding cumulative fractions.

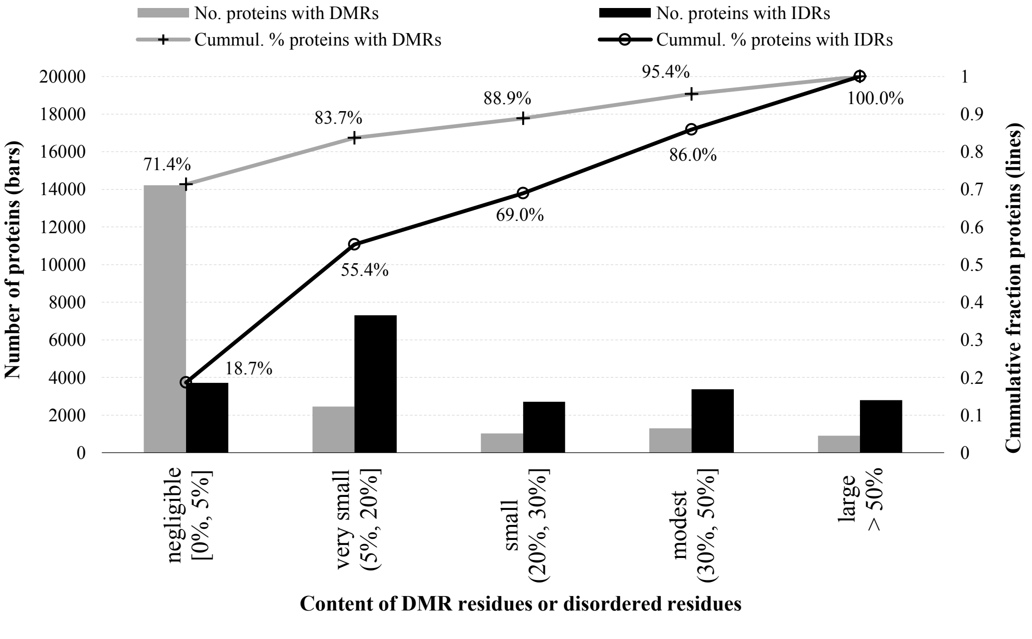


**Figure 5.6.** Content of residues in putative DMRs and putative intrinsically disordered regions (IDRs) for the complete reviewed human proteome.

Bars show the number of proteins with content ranges given on the *x*-axis. Lines show the corresponding cumulative fractions of proteins.

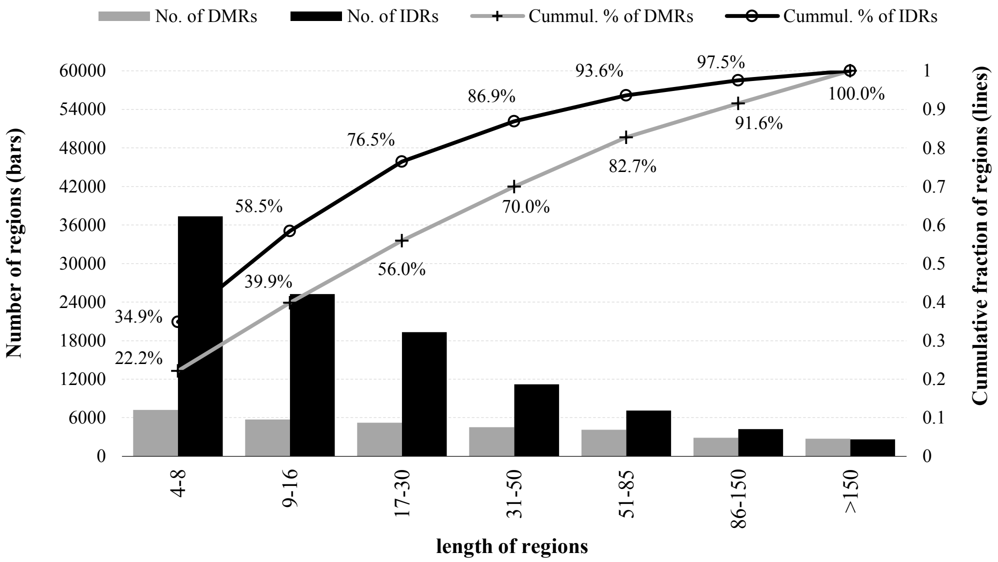


**Figure 5.7.** Length of putative DMRs and putative intrinsically disordered regions (IDRs) for the complete reviewed human proteome.

Bars show the number of regions with length ranges given on the *x*-axis. Lines show the corresponding cumulative fractions of regions.

### 5.3.4 DMRpred's webserver

DMRpred is provided as a webserver at http://biomine.cs.vcu.edu/servers/DMRpred. Users only need to provide FASTA-formatted protein sequence(s) to obtain predictions that are computed on the server side. The server outputs a propensity score for each residue in the input sequence(s) for being a DMR residue. The server also produces binary predictions that are generated from the propensities using the cutoff = 0.761; residues with propensity ≥ 0.761 are predicted as DMR residues. This cutoff was calibrated to provide 5% FPR on the test dataset. The webserver allows batch submissions of up to 50 sequences at one time. The sequences should be at least 21 residues long since ASAquick that is embedded into DMRpred requires this. Users are encouraged to provide email address which is used to provide notification when the prediction is finished and a private URL where the results can be downloaded from. Whether or not the email is provided, the results are also made available in the browser window, given that the user will not close it when the results are being processed. DMRpred is relatively fast. The webserver produces prediction for a protein with length of about 500 residues in less than one minute.

## 5.4 Conclusion

We conceptualized, designed, tested and deployed DMRpred, the first-of-its-kind computational method for the prediction of DMRs directly from protein sequences. DMRpred uses the input sequence to derive a comprehensive profile that includes sequence conservation, putative relative solvent accessibility and intrinsic disorder, and a novel set of residue-level propensities for functions that are relevant to DMRs. The information in this profile is aggregated using sliding windows and an innovative type of windows defined based on putative IDRs. Features extracted from this profile are input to the Random Forest model to make the predictions.

We empirically demonstrate that the various parts of the profile and the two types of windows are useful for the prediction. Results on a blind test dataset reveal that DMRpred provides accurate predictions of DMRs. The predictive quality of DMRpred is statistically significantly higher than the predictive performance of a comprehensive set of alternative approaches to make these predictions. Predictions on the complete human proteome reveal that as many as 25% of human proteins may have at least one long DMR.

## Chapter 6

# Summary and conclusions

This research focuses on the characterization and prediction of functions of intrinsically disordered regions (IDRs) in proteins. IDRs are prevalent in nature and carry out a wide range of important cellular functions. IDRs can be nowadays predicted with high accuracy using computational methods. Some of these methods are even sufficiently fast to perform these predictions on the whole proteome scale. However, functions of IDRs are largely undetermined. We start this research with a project that aims to characterize functions of IDRs in the human dengue virus using existing methods. We found that although existing methods can find some of the functions relevant to IDRs, e.g., protein-protein binding, many other functions could not be predicted or were under-predicted. This inspired us to develop computational methods that address the prediction of functions of IDRs that cannot be predicted with the existing methods.

We focus our research on two types of IDRs: disordered flexible linkers (DFLs) and disordered moonlighting regions (DMRs). DFL is the most prevalent non-binding function of IDRs that cannot be predicted by existing methods. DMRs are regions that carry out multiple functions.

To design the methods that predicts DFLs, DFLpred, we first collect proteins from DisProt to prepare the training dataset and the blind test dataset. We ensure that the training and test datasets share low sequence similarity. Next, for each residue in a protein sequence we incorporate its physicochemical properties estimated from AA indices, structural properties including secondary structure predicted with PSIPRED, intrinsically disorder predicted with IUPred and sequence complexity predicted with SEG. We design a rich set of features from these physicochemical and structural properties. We select features that are most relevant to DFLs using three classifiers on the cross–validated training dataset, and parameterize the classifier, if needed. We use Logistic Regression as the predictive model. This model uses four features computed from AA indices and IUPred as its inputs. We analyze the predictive model by investigating the values of the selected features and we find that these features can be used as independent markers of DLFs. Finally, we empirically compare DFLpred with other alternative methods on the blind test dataset. Results show that DFLpred outperforms the other methods in terms of AUC and AUC$_{ratio}$ (measures the predictive quality at low range of the false positive rate), and these differences are statistically significant. We assess the runtime of DFLpred and demonstrate that it is fast enough to handle prediction at the whole proteome scale. We also run DFLpred on the complete reviewed human proteome. The corresponding putative results show that about 10% of human proteins may have a large content of over 30% DFL residues, and there are about 6000 long DFL regions.

Our analysis of data in DisProt shows that about 37% IDRs carry out multiple functions. This class of IDRs cannot be predicted by the existing predictors of functions of IDRs that focus on individual functions, and it also cannot be predicted by predictors of moonlighting proteins. To design predictor of DMRs, DMRpred, we first define DMRs and collect proteins from DisProt and PDB to prepare the training dataset and the blind test dataset. Next, we represent each residue in a protein sequence by a set of biophysical and structural properties including sequence conservation computed with HHblits, relative solvent accessibility computed with ASAquick, intrinsic disorder computed with IUPred, and novel AA indices that quantify propensity of individual AA types to carry out functions that are relevant to DMRs. We empirically generate a rich set of features from these biophysical and structural properties, using a fixed-length sliding window centered on the residue we are currently predict and a novel type of window based predicted IDRs. We empirically investigate three machine learning algorithms and select the Random Forest as the predictive model; this model secures the most accurate results on the training dataset. We analyze this predictive model by using subsets of features. We show that the sequence conservation, relative solvent accessibility, putative intrinsic disorder, the novel AA indices and both types of windows contribute to the predictive performance of this model. We compare the resulting DMRpred method with other alternative methods on the blind test dataset. The empirical results demonstrate that DMRpred is statistically significantly better than the other alternatives in all terms of accuracy, precision, sensitivity, MCC, AUC and $AUC_{ratio}$. We run DMRpred on the complete reviewed human proteome and show that as many as 25% of human proteins may have long DMRs.

## 6.1 Major contributions

The major contribution under the first goal is the first-of-its-kind characterization of putative functions of IDRs in the human dengue virus. We also observed that existing methods that predict these functions are not sufficient to provide comprehensive annotations since they are lacking in scope and coverage. This has inspired us to develop new methods to address this problem under goals 2 and goal 3. The major contributions under these two goals are the conceptualization, development, implementation, comprehensive empirical testing, and deployment (as a webserver) of the corresponding two novel predictors. A more detailed list of contributions for each goal follows.

Goal 1: Characterization of functions of IDRs in human dengue virus (DENV)

- The extent of intrinsic disorder in the complete proteomes of DENV was evaluated.
- The peculiarities of putative functions of IDRs within DENV proteins were analysed.
- The (in)ability of current methods to annotate putative functions of IDRs in DENV was evaluated.

Goal 2: Fast and accurate computational prediction of DFL regions

- A fast linear first-of-its-kind model, DFLpred, which accurately predicts DFLs from protein sequences was conceptualized, developed, implemented, and empirically tested and compared with alternative methods that can be used to perform these predictions.
- DFLpred was deployed as a publicly available webserver. This webserver has been in operation for about 1.5 years and is actively used by the research community. Based on a report generated with Google Analytics on Nov 29,

2017, the webserver was used close to 3000 times by over 350 unique users from 193 cities and 43 countries.

- Certain characteristics of DFLs in the complete reviewed human proteome based on DFLs predicted with DFLpred were analysed.

Goal 3: Fast and accurate computational prediction of DMRs

- A first-of-its-kind random forest model, DMRpred, which accurately predicts DMRs from protein sequences was developed, conceptualized, developed, implemented, and empirically tested and compared with alternative methods that can be used to perform these predictions.
- DMRpred was deployed as a webserver. This sever was not yet made available publically since at this point it is still under peer-review.
- Novel scales that quantify propensities of amino acids for functions that are relevant to DMRs was designed.
- An original approach to build predictive inputs that aggregate structural and functional characteristics based on putative IDRs was proposed, implemented and assessed.
- Certain characteristics of DMRs in the complete reviewed human proteome based on putative DMRs generated with DMRpred were quantified.

## 6.2 Conclusions

IDRs are abundant in nature and functionally important. We show that prediction of IDRs is a mature research area and a rich selection of accurate and runtime-efficient predictors for IDRs is available. We have focused on the prediction of functions of IDRs to address the two thesis statements:

1. Our analysis reveals that sequence alignment is not sufficient to annotate functions of IDRs. It misses some of these functions and under-predicts the other functions.

2. We also show that the current predictors of functions of IDRs do not consider some of the functions. Only 11 out of 37 functions listed by DisProt can be predicted by

the current computational methods. To this end, we empirically demonstrate that two specific functions, disordered flexible linkers and disordered moonlighting regions, can be accurately predicted from the protein sequences.

## 6.3 Future work

Our research addressed two important functions of IDRs, i.e., disordered flexible linkers and disordered moonlighting regions. However, there are other functions that cannot be predicted by the current and the proposed here methods. These functions include entropic regions (e.g., entropic bristle and clock) and protein-lipid binding regions. At this point the amount of human-curated annotations for these functions that were deposited into databases like DisProt is not yet sufficient to build and test predictors. However, this issue should be revisited in a near future and the corresponding methods should be developed once the sufficient quantity of data becomes available.

As we discuss in Section 2.2.2 in the Background Chapter, databases such as MobiDB and $D^2P^2$ provide access to a large number of putative disorder annotations that cover all proteins in UniProt. These putative annotations are predicted by current predictors of IDRs. These two databases enjoy a significant amount of research interest. For instance, the articles that describe MobiDB [103] and $D^2P^2$ [57] were cited already 92 and 173 times (source: Google Scholar on Nov 10, 2017), respectively, in spite of the fact that these articles were published in 2012 and 2013. We believe that the putative annotations of functions of IDRs should be made available via databases, perhaps by expanding these two existing resources to include the predicted functional annotations.

The availability of the webservers and databases of functional annotations of IDRs will accelerate the rate of scientific discovery in this area. Given the fact that IDRs are implicated in a number of diseases [6, 38-42] and since they constitute attractive targets for rational drug design [45-48, 72], this availability will also contribute to understanding the causes of certain diseases and to the drug discovery efforts.

# Bibliography

1. Habchi, J., et al., *Introducing Protein Intrinsic Disorder.* Chemical Reviews, 2014. **114**(13): p. 6561-6588.
2. Xue, B., A.K. Dunker, and V.N. Uversky, *Orderly order in protein intrinsic disorder distribution: disorder in 3500 proteomes from viruses and the three domains of life.* J Biomol Struct Dyn, 2012. **30**(2): p. 137-49.
3. Peng, Z., et al., *Exceptionally abundant exceptions: comprehensive characterization of intrinsic disorder in all domains of life.* Cell Mol Life Sci, 2015. **72**(1): p. 137-51.
4. Dunker, A.K., et al., *Intrinsic Disorder and Protein Function†.* Biochemistry, 2002. **41**(21): p. 6573-6582.
5. Dunker, A.K., et al., *Function and structure of inherently disordered proteins.* Curr Opin Struct Biol, 2008. **18**(6): p. 756-64.
6. Xie, H., et al., *Functional anthology of intrinsic disorder. 3. Ligands, post-translational modifications, and diseases associated with intrinsically disordered proteins.* J Proteome Res, 2007. **6**(5): p. 1917-32.
7. Sickmeier, M., et al., *DisProt: the Database of Disordered Proteins.* Nucleic Acids Research, 2007. **35**(suppl 1): p. D786-D793.
8. Berman, H.M., et al., *The Protein Data Bank.* Nucleic Acids Research, 2000. **28**(1): p. 235-242.
9. Li, J., et al., *An Overview of Predictors for Intrinsically Disordered Proteins over 2010–2014.* International Journal of Molecular Sciences, 2015. **16**(10): p. 23446.
10. Meng, F., V.N. Uversky, and L. Kurgan, *Comprehensive review of methods for prediction of intrinsic disorder and its molecular functions.* Cell Mol Life Sci, 2017. **74**(17): p. 3069-3090.
11. Atkins, J.D., et al., *Disorder Prediction Methods, Their Applicability to Different Protein Targets and Their Usefulness for Guiding Experimental Studies.* Int J Mol Sci, 2015. **16**(8): p. 19040-54.
12. Monastyrskyy, B., et al., *Assessment of protein disorder region predictions in CASP10.* Proteins, 2014. **82 Suppl 2**: p. 127-37.
13. Deng, X., J. Eickholt, and J. Cheng, *A comprehensive overview of computational protein disorder prediction methods.* Mol Biosyst, 2012. **8**(1): p. 114-21.
14. He, B., et al., *Predicting intrinsic disorder in proteins: an overview.* Cell Res, 2009. **19**(8): p. 929-49.
15. Peng, Z.L. and L. Kurgan, *Comprehensive comparative assessment of in-silico predictors of disordered regions.* Curr Protein Pept Sci, 2012. **13**(1): p. 6-18.
16. Dosztányi, Z., B. Mészáros, and I. Simon, *Bioinformatical approaches to characterize intrinsically disordered/unstructured proteins.* Briefings in Bioinformatics, 2010. **11**(2): p. 225-243.
17. Monastyrskyy, B., et al., *Assessment of protein disorder region predictions in CASP10.* Proteins, 2014. **82**(0 2): p. 127-137.

18. Necci, M., et al., *A comprehensive assessment of long intrinsic protein disorder from the DisProt database.* Bioinformatics, 2017.

19. Walsh, I., et al., *Comprehensive large-scale assessment of intrinsic protein disorder.* Bioinformatics, 2015. **31**(2): p. 201-8.

20. Oldfield, C.J., et al., *Utilization of protein intrinsic disorder knowledge in structural proteomics.* Biochimica et Biophysica Acta (BBA) - Proteins and Proteomics, 2013. **1834**(2): p. 487-498.

21. Peng, Z., et al., *A creature with a hundred waggly tails: intrinsically disordered proteins in the ribosome.* Cellular and Molecular Life Sciences, 2014. **71**(8): p. 1477-1504.

22. Peng, Z., et al., *Resilience of death: intrinsic disorder in proteins involved in the programmed cell death.* Cell Death Differ, 2013. **20**(9): p. 1257-67.

23. Wang, C., V.N. Uversky, and L. Kurgan, *Disordered nucleiome: Abundance of intrinsic disorder in the DNA- and RNA-binding proteins in 1121 species from Eukaryota, Bacteria and Archaea.* Proteomics, 2016. **16**(10): p. 1486-98.

24. Fan, X., et al., *The intrinsic disorder status of the human hepatitis C virus proteome.* Mol Biosyst, 2014. **10**(6): p. 1345-63.

25. Xue, B., et al., *Protein intrinsic disorder as a flexible armor and a weapon of HIV-1.* Cell Mol Life Sci, 2012. **69**(8): p. 1211-59.

26. Peng, Z., et al., *More than just tails: intrinsic disorder in histone proteins.* Mol Biosyst, 2012. **8**(7): p. 1886-901.

27. Varadi, M., et al., *Functional Advantages of Conserved Intrinsic Disorder in RNA-Binding Proteins.* PLoS One, 2015. **10**(10): p. e0139731.

28. Peng, Z., et al., *Intrinsic disorder in the BK channel and its interactome.* PLoS One, 2014. **9**(4): p. e94331.

29. Mei, Y., et al., *Intrinsically disordered regions in autophagy proteins.* Proteins, 2014. **82**(4): p. 565-78.

30. Marin, M. and T. Ott, *Intrinsic disorder in plant proteins and phytopathogenic bacterial effectors.* Chem Rev, 2014. **114**(13): p. 6912-32.

31. Xue, B., et al., *Archaic chaos: intrinsically disordered proteins in Archaea.* BMC Syst Biol, 2010. **4 Suppl 1**: p. S1.

32. Ward, J.J., et al., *Prediction and functional analysis of native disorder in proteins from the three kingdoms of life.* J Mol Biol, 2004. **337**(3): p. 635-45.

33. Tompa, P., Z. Dosztanyi, and I. Simon, *Prevalent structural disorder in E. coli and S. cerevisiae proteomes.* J Proteome Res, 2006. **5**(8): p. 1996-2000.

34. Potenza, E., et al., *MobiDB 2.0: an improved database of intrinsically disordered and mobile proteins.* Nucleic Acids Research, 2015. **43**(D1): p. D315-D320.

35. Oates, M.E., et al., *D2P2: database of disordered protein predictions.* Nucleic Acids Research, 2013. **41**(D1): p. D508-D516.

36. Uversky, V.N., C.J. Oldfield, and A.K. Dunker, *Intrinsically disordered proteins in human diseases: introducing the D2 concept.* Annu Rev Biophys, 2008. **37**: p. 215-46.

37. Midic, U., et al., *Unfoldomics of human genetic diseases: illustrative examples of ordered and intrinsically disordered members of the human diseasome.* Protein Pept Lett, 2009. **16**(12): p. 1533-47.

38. Uversky, V.N., et al., *Unfoldomics of human diseases: linking protein intrinsic disorder with diseases.* BMC Genomics, 2009. **10 Suppl 1**: p. S7.
39. Babu, M.M., *The contribution of intrinsically disordered regions to protein function, cellular complexity, and human disease.* Biochem Soc Trans, 2016. **44**(5): p. 1185-1200.
40. Midic, U., et al., *Protein disorder in the human diseasome: unfoldomics of human genetic diseases.* BMC Genomics, 2009. **10 Suppl 1**: p. S12.
41. Uversky, V.N., *The triple power of D(3): protein intrinsic disorder in degenerative diseases.* Front Biosci (Landmark Ed), 2014. **19**: p. 181-258.
42. Cheng, Y., et al., *Abundance of intrinsic disorder in protein associated with cardiovascular disease.* Biochemistry, 2006. **45**(35): p. 10448-60.
43. Cheng, Y., et al., *Rational drug design via intrinsically disordered protein.* Trends in Biotechnology, 2006. **24**(10): p. 435-442.
44. Gang, H., et al., *Untapped Potential of Disordered Proteins in Current Druggable Human Proteome.* Current Drug Targets, 2016. **17**(10): p. 1198-1205.
45. Ambadipudi, S. and M. Zweckstetter, *Targeting intrinsically disordered proteins in rational drug discovery.* Expert Opin Drug Discov, 2015: p. 1-13.
46. Uversky, V.N., *Intrinsically disordered proteins and novel strategies for drug discovery.* Expert Opin Drug Discov, 2012. **7**(6): p. 475-88.
47. Wang, J., et al., *Novel strategies for drug discovery based on Intrinsically Disordered Proteins (IDPs).* Int J Mol Sci, 2011. **12**(5): p. 3205-19.
48. Dunker, A.K. and V.N. Uversky, *Drugs for 'protein clouds': targeting intrinsically disordered transcription factors.* Curr Opin Pharmacol, 2010. **10**(6): p. 782-8.
49. Dosztányi, Z., B. Mészáros, and I. Simon, *ANCHOR: web server for predicting protein binding regions in disordered proteins.* Bioinformatics, 2009. **25**(20): p. 2745-2746.
50. Disfani, F.M., et al., *MoRFpred, a computational tool for sequence-based prediction and characterization of short disorder-to-order transitioning binding regions in proteins.* Bioinformatics, 2012. **28**(12): p. i75-i83.
51. Peng, Z. and L. Kurgan, *High-throughput prediction of RNA, DNA and protein binding regions mediated by intrinsic disorder.* Nucleic Acids Research, 2015.
52. Vickery, H.B., *The origin of the word protein.* Yale J Biol Med, 1950. **22**(5): p. 387-93.
53. Dunker, A.K., et al., *What's in a name? Why these proteins are intrinsically disordered.* Intrinsically Disordered Proteins, 2013. **1**(1): p. e24157.
54. van der Lee, R., et al., *Classification of Intrinsically Disordered Regions and Proteins.* Chemical Reviews, 2014. **114**(13): p. 6589-6631.
55. Dunker, A.K., et al., *Intrinsic protein disorder in complete genomes.* Genome Inform Ser Workshop Genome Inform, 2000. **11**: p. 161-71.
56. Tompa, P., *Intrinsically unstructured proteins.* Trends in Biochemical Sciences, 2002. **27**(10): p. 527-533.
57. Oates, M.E., et al., *D(2)P(2): database of disordered protein predictions.* Nucleic Acids Res, 2013. **41**(Database issue): p. D508-16.
58. Piovesan, D., et al., *DisProt 7.0: a major update of the database of disordered proteins.* Nucleic Acids Research, 2017. **45**(D1): p. D219-D227.

59. Fukuchi, S., et al., *IDEAL in 2014 illustrates interaction networks composed of intrinsically disordered proteins and their binding partners.* Nucleic Acids Res, 2014. **42**(Database issue): p. D320-5.

60. Martin, A.J.M., I. Walsh, and S.C.E. Tosatto, *MOBI: a web server to define and visualize structural mobility in NMR protein ensembles.* Bioinformatics, 2010. **26**(22): p. 2916-2917.

61. *UniProt: the universal protein knowledgebase.* Nucleic Acids Research, 2017. **45**(D1): p. D158-D169.

62. Wright, P.E. and H.J. Dyson, *Intrinsically unstructured proteins: re-assessing the protein structure-function paradigm.* J Mol Biol, 1999. **293**(2): p. 321-331.

63. Uversky, V.N., J.R. Gillespie, and A.L. Fink, *Why are "natively unfolded" proteins unstructured under physiologic conditions?* Proteins, 2000. **41**(3): p. 415-427.

64. Dunker, A.K., et al., *Intrinsically disordered protein.* J Mol Graph Model, 2001. **19**(1): p. 26-59.

65. Uversky, V.N. and A.K. Dunker, *Understanding protein non-folding.* Biochim Biophys Acta, 2010. **1804**(6): p. 1231-64.

66. Dunker, A.K., et al., *Function and structure of inherently disordered proteins.* Curr. Opin. Struct. Biol., 2008. **18**(6): p. 756-64.

67. Xie, H., et al., *Functional anthology of intrinsic disorder. 1. Biological processes and functions of proteins with long disordered regions.* J. Proteome Res., 2007. **6**(5): p. 1882-98.

68. Dyson, H.J. and P.E. Wright, *Intrinsically unstructured proteins and their functions.* Nat. Rev. Mol. Cell. Biol., 2005. **6**(3): p. 197-208.

69. Tompa, P., *The interplay between structure and function in intrinsically unstructured proteins.* FEBS Lett., 2005. **579**(15): p. 3346-54.

70. Peng, Z., et al., *A creature with a hundred waggly tails: intrinsically disordered proteins in the ribosome.* Cell Mol Life Sci, 2014. **71**(8): p. 1477-504.

71. Radivojac, P., et al., *Intrinsic disorder and functional proteomics.* Biophys J, 2007. **92**(5): p. 1439-56.

72. Hu, G., et al., *Untapped Potential of Disordered Proteins in Current Druggable Human Proteome.* Curr Drug Targets, 2016. **17**(10): p. 1198-205.

73. Ferron, F., et al., *A practical overview of protein disorder prediction methods.* Proteins: Structure, Function, and Bioinformatics, 2006. **65**(1): p. 1-14.

74. He, B., et al., *Predicting intrinsic disorder in proteins: an overview.* Cell Res, 2009. **19**(8): p. 929-949.

75. Deng, X., J. Eickholt, and J. Cheng, *A comprehensive overview of computational protein disorder prediction methods.* Molecular BioSystems, 2012. **8**(1): p. 114-121.

76. Pentony, M., J. Ward, and D. Jones, *Computational Resources for the Prediction and Analysis of Native Disorder in Proteins*, in *Proteome Bioinformatics*, S.J. Hubbard and A.R. Jones, Editors. 2010, Humana Press. p. 369-393.

77. Walsh, I., et al., *Comprehensive large-scale assessment of intrinsic protein disorder.* Bioinformatics, 2015. **31**(2): p. 201-208.

78. Atkins, J., et al., *Disorder Prediction Methods, Their Applicability to Different Protein Targets and Their Usefulness for Guiding Experimental Studies.* International Journal of Molecular Sciences, 2015. **16**(8): p. 19040.

79. Liu, J. and B. Rost, *NORSp: predictions of long regions without regular secondary structure.* Nucleic Acids Research, 2003. **31**(13): p. 3833-3835.

80. Linding, R., et al., *GlobPlot: exploring protein sequences for globularity and disorder.* Nucleic Acids Research, 2003. **31**(13): p. 3701-3708.

81. Dosztányi, Z., et al., *IUPred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content.* Bioinformatics, 2005. **21**(16): p. 3433-3434.

82. Dosztányi, Z., et al., *The Pairwise Energy Content Estimated from Amino Acid Composition Discriminates between Folded and Intrinsically Unstructured Proteins.* Journal of Molecular Biology, 2005. **347**(4): p. 827-839.

83. Linding, R., et al., *Protein Disorder Prediction: Implications for Structural Proteomics.* Structure, 2003. **11**(11): p. 1453-1459.

84. Jones, D.T. and D. Cozzetto, *DISOPRED3: precise disordered region predictions with annotated protein-binding activity.* Bioinformatics, 2015. **31**(6): p. 857-63.

85. Ward, J.J., et al., *The DISOPRED server for the prediction of protein disorder.* Bioinformatics, 2004. **20**(13): p. 2138-9.

86. Obradovic, Z., et al., *Predicting intrinsic disorder from amino acid sequence.* Proteins, 2003. **53 Suppl 6**: p. 566-72.

87. Obradovic, Z., et al., *Exploiting heterogeneous sequence properties improves prediction of protein disorder.* Proteins, 2005. **61 Suppl 7**: p. 176-82.

88. Mizianty, M.J., Z.L. Peng, and L. Kurgan, *MFDp2: Accurate predictor of disorder in proteins by fusion of disorder probabilities, content and profiles.* Intrinsically Disordered Proteins, 2013. **1**(1): p. e24428.

89. Mizianty, M.J., V. Uversky, and L. Kurgan, *Prediction of intrinsic disorder in proteins using MFDp2.* Methods Mol Biol, 2014. **1137**: p. 147-62.

90. Mizianty, M.J., et al., *Improved sequence-based prediction of disordered regions with multilayer fusion of multiple information sources.* Bioinformatics, 2010. **26**(18): p. i489-96.

91. Kozlowski, L.P. and J.M. Bujnicki, *MetaDisorder: a meta-server for the prediction of intrinsic disorder in proteins.* BMC Bioinformatics, 2012. **13**(1): p. 1-11.

92. Xue, B., et al., *PONDR-FIT: a meta-predictor of intrinsically disordered amino acids.* Biochim Biophys Acta, 2010. **1804**(4): p. 996-1010.

93. Ishida, T. and K. Kinoshita, *PrDOS: prediction of disordered protein regions from amino acid sequence.* Nucleic Acids Res, 2007. **35**(Web Server issue): p. W460-4.

94. McGuffin, L.J., et al., *IntFOLD: an integrated server for modelling protein structures and functions from amino acid sequences.* Nucleic Acids Research, 2015. **43**(W1): p. W169-W173.

95. Fan, X. and L. Kurgan, *Accurate prediction of disorder in protein chains with a comprehensive and empirically designed consensus.* J Biomol Struct Dyn, 2014. **32**(3): p. 448-64.

96. Peng, Z. and L. Kurgan, *On the complementarity of the consensus-based disorder prediction.* Pac Symp Biocomput, 2012: p. 176-87.

97. Fukuchi, S., et al., *IDEAL: Intrinsically Disordered proteins with Extensive Annotations and Literature.* Nucleic Acids Research, 2012. **40**(D1): p. D507-D511.

98. Fukuchi, S., et al., *IDEAL in 2014 illustrates interaction networks composed of intrinsically disordered proteins and their binding partners.* Nucleic Acids Research, 2014. **42**(D1): p. D320-D325.

99. Oldfield, C.J., et al., *Utilization of protein intrinsic disorder knowledge in structural proteomics.* Biochim Biophys Acta, 2013. **1834**(2): p. 487-98.

100. Pentony, M.M. and D.T. Jones, *Modularity of intrinsic disorder in the human proteome.* Proteins, 2010. **78**(1): p. 212-21.

101. Fukuchi, S., et al., *Binary classification of protein molecules into intrinsically disordered and ordered segments.* BMC Structural Biology, 2011. **11**(1): p. 1-10.

102. Bairoch, A., et al., *The Universal Protein Resource (UniProt).* Nucleic Acids Research, 2005. **33**(suppl 1): p. D154-D159.

103. Di Domenico, T., et al., *MobiDB: a comprehensive database of intrinsic protein disorder annotations.* Bioinformatics, 2012. **28**(15): p. 2080-2081.

104. Mészáros, B., I. Simon, and Z. Dosztányi, *Prediction of Protein Binding Regions in Disordered Proteins.* PLoS Comput Biol, 2009. **5**(5): p. e1000376.

105. Altschul, S.F., et al., *Basic local alignment search tool.* J Mol Biol, 1990. **215**(3): p. 403-10.

106. Madden, T. *The BLAST Sequence Analysis Tool*. 2002 August 13, 2003; Available from: http://www.ncbi.nlm.nih.gov/books/NBK21097/.

107. Pruitt, K.D., T. Tatusova, and D.R. Maglott, *NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins.* Nucleic Acids Research, 2005. **33**(Database Issue): p. D501-D504.

108. Yan, R. and Y. Zhang. *SWalign*. 2012; Available from: http://zhanglab.ccmb.med.umich.edu/NW-align/SWalign.java.tar.gz.

109. Sievers, F., et al., *Fast, scalable generation of high‐quality protein multiple sequence alignments using Clustal Omega.* 2011. **7**(1).

110. Peng, Z., et al., *A creature with a hundred waggly tails: intrinsically disordered proteins in the ribosome.* Cellular and Molecular Life Sciences, 2013. **71**(8): p. 1477-1504.

111. Radivojac, P., et al., *A large-scale evaluation of computational protein function prediction.* Nature Methods, 2013. **10**: p. 221.

112. Khan, W., et al., *Predicting Binding within Disordered Protein Regions to Structurally Characterised Peptide-Binding Domains.* PLoS ONE, 2013. **8**(9): p. e72838.

113. Oldfield, C.J., et al., *Comparing and Combining Predictors of Mostly Disordered Proteins†.* Biochemistry, 2005. **44**(6): p. 1989-2000.

114. Cheng, Y., et al., *Mining α-Helix-Forming Molecular Recognition Features with Cross Species Sequence Alignments†.* Biochemistry, 2007. **46**(47): p. 13468-13477.

115. Fang, C., et al., *MFSPSSMpred: identifying short disorder-to-order binding regions in disordered proteins based on contextual local evolutionary conservation.* BMC Bioinformatics, 2013. **14**(1): p. 1-14.

116. Malhis, N. and J. Gsponer, *Computational identification of MoRFs in protein sequences.* Bioinformatics, 2015. **31**(11): p. 1738-1744.

117. Malhis, N., M. Jacobson, and J. Gsponer, *MoRFchibi SYSTEM: software tools for the identification of MoRFs in protein sequences.* Nucleic Acids Res, 2016.

118. Yan, J., et al., *Molecular recognition features (MoRFs) in three domains of life.* Molecular BioSystems, 2015.
119. Xue, B., A.K. Dunker, and V.N. Uversky, *Retro-MoRFs: Identifying Protein Binding Sites by Normal and Reverse Alignment and Intrinsic Disorder Prediction.* International Journal of Molecular Sciences, 2010. **11**(10): p. 3725-3747.
120. Jones, D.T. and D. Cozzetto, *DISOPRED3: precise disordered region predictions with annotated protein-binding activity.* Bioinformatics, 2015. **31**(6): p. 857-863.
121. Mooney, C., et al., *Prediction of Short Linear Protein Binding Regions.* Journal of Molecular Biology, 2012. **415**(1): p. 193-204.
122. Schaefer, R.L., L.D. Roi, and R.A. Wolfe, *A ridge logistic estimator.* Communications in Statistics - Theory and Methods, 1984. **13**(1): p. 99-113.
123. John, G.H. and P. Langley. *Estimating continuous distributions in Bayesian classifiers*. in *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. 1995. Morgan Kaufmann Publishers Inc.
124. Quinlan, J.R., *C4.5: programs for machine learning*. 1993: Morgan Kaufmann Publishers Inc. 302.
125. Breiman, L., *Random forests.* Machine learning, 2001. **45**(1): p. 5-32.
126. Rost, B., *Twilight zone of protein sequence alignments.* Protein Eng, 1999. **12**(2): p. 85-94.
127. Mizianty, M.J., et al., *Improved sequence-based prediction of disordered regions with multilayer fusion of multiple information sources.* Bioinformatics, 2010. **26**(18): p. i489-i496.
128. Jones, D.T., *Protein secondary structure prediction based on position-specific scoring matrices1.* Journal of Molecular Biology, 1999. **292**(2): p. 195-202.
129. Cuff, J.A. and G.J. Barton, *Application of multiple sequence alignment profiles to improve protein secondary structure prediction.* Proteins, 2000. **40**(3): p. 502-11.
130. Student, *The probable error of a mean.* Biometrika, 1908. **6**(1): p. 1-25.
131. Wilcoxon, F., *Individual comparisons by ranking methods.* Biometrics bulletin, 1945: p. 80-83.
132. Anderson, T.W. and D.A. Darling, *Asymptotic Theory of Certain "Goodness of Fit" Criteria Based on Stochastic Processes.* Ann. Math. Statist., 1952: p. 193-212.
133. Peng, Z., et al., *Exceptionally abundant exceptions: comprehensive characterization of intrinsic disorder in all domains of life.* Cellular and Molecular Life Sciences, 2014. **72**(1): p. 137-151.
134. Consortium, T.U., *The Universal Protein Resource (UniProt) in 2010.* Nucleic Acids Research, 2010. **38**(suppl 1): p. D142-D148.
135. McGuffin, L.J., *Intrinsic disorder prediction from the analysis of multiple protein fold recognition models.* Bioinformatics, 2008. **24**(16): p. 1798-1804.
136. Ward, J.J., et al., *The DISOPRED server for the prediction of protein disorder.* Bioinformatics, 2004. **20**(13): p. 2138-2139.
137. McGuffin, L.J., K. Bryson, and D.T. Jones, *The PSIPRED protein structure prediction server.* Bioinformatics, 2000. **16**(4): p. 404-405.
138. Faraggi, E., B. Xue, and Y. Zhou, *Improving the prediction accuracy of residue solvent accessibility and real-value backbone torsion angles of proteins by guided-learning through a two-layer neural network.* Proteins: Structure, Function, and Bioinformatics, 2009. **74**(4): p. 847-856.

139. Schlessinger, A., G. Yachdav, and B. Rost, *PROFbval: predict flexible and rigid residues in proteins.* Bioinformatics, 2006. **22**(7): p. 891-893.

140. Mohan, A., et al., *Analysis of Molecular Recognition Features (MoRFs).* Journal of Molecular Biology, 2006. **362**(5): p. 1043-1059.

141. Vacic, V., et al., *Characterization of Molecular Recognition Features, MoRFs, and Their Binding Partners.* Journal of Proteome Research, 2007. **6**(6): p. 2351-2366.

142. Gould, C.M., et al., *ELM: the status of the 2010 eukaryotic linear motif resource.* Nucleic Acids Research, 2010. **38**(suppl 1): p. D167-D180.

143. Dinkel, H., et al., *The eukaryotic linear motif resource ELM: 10 years and counting.* Nucleic Acids Research, 2014. **42**(D1): p. D259-D266.

144. Shvadchak, V.V. and V. Subramaniam, *A Four-Amino Acid Linker between Repeats in the alpha-Synuclein Sequence Is Important for Fibril Formation.* Biochemistry, 2014. **53**(2): p. 279-281.

145. Oldfield, C.J. and A.K. Dunker, *Intrinsically disordered proteins and intrinsically disordered protein regions.* Annu Rev Biochem, 2014. **83**: p. 553-84.

146. Anand, S. and D. Mohanty, *Inter-domain movements in polyketide synthases: a molecular dynamics study.* Mol Biosyst, 2012. **8**(4): p. 1157-71.

147. Chen, X., J.L. Zaro, and W.C. Shen, *Fusion protein linkers: property, design and functionality.* Adv Drug Deliv Rev, 2013. **65**(10): p. 1357-69.

148. George, R.A. and J. Heringa, *An analysis of protein domain linkers: their classification and role in protein folding.* Protein Eng, 2002. **15**(11): p. 871-9.

149. Udwary, D.W., M. Merski, and C.A. Townsend, *A Method for Prediction of the Locations of Linker Regions within Large Multifunctional Proteins, and Application to a Type I Polyketide Synthase.* Journal of Molecular Biology, 2002. **323**(3): p. 585-598.

150. Schlessinger, A. and B. Rost, *Protein flexibility and rigidity predicted from sequence.* Proteins: Structure, Function, and Bioinformatics, 2005. **61**(1): p. 115-126.

151. I.B.Kuznetsov and M.McDuffie, *FlexPred: a web-server for predicting residue positions involved in conformational switches in proteins.* Bioinformation, 2008. **3**(3): p. 134-136.

152. Kuznetsov, I.B., *Ordered conformational change in the protein backbone: prediction of conformationally variable positions from sequence and low-resolution structural data.* Proteins, 2008. **72**(1): p. 74-87.

153. Pan, X.Y. and H.B. Shen, *Robust prediction of B-factor profile from sequence using two-stage SVR based on random forest feature selection.* Protein Pept Lett, 2009. **16**(12): p. 1447-54.

154. de Brevern, A.G., et al., *PredyFlexy: flexibility and local structure prediction from sequence.* Nucleic Acids Research, 2012. **40**(W1): p. W317-W322.

155. Cilia, E., et al., *From protein sequence to dynamics and disorder with DynaMine.* Nat Commun, 2013. **4**.

156. Cilia, E., et al., *The DynaMine webserver: predicting protein dynamics from sequence.* Nucleic Acids Research, 2014.

157. Buchan, D.W.A., et al., *Scalable web services for the PSIPRED Protein Analysis Workbench.* Nucleic Acids Research, 2013. **41**(W1): p. W349-W357.

158. Kawashima, S., et al., *AAindex: amino acid index database, progress report 2008.* Nucleic Acids Research, 2008. **36**(suppl 1): p. D202-D205.

159. Wootton, J.C., *Non-globular domains in protein sequences: Automated segmentation using complexity measures.* Computers & Chemistry, 1994. **18**(3): p. 269-285.

160. Disfani, F.M., et al., *MoRFpred, a computational tool for sequence-based prediction and characterization of short disorder-to-order transitioning binding regions in proteins.* Bioinformatics, 2012. **28**(12): p. i75-83.

161. Yan, J., et al., *Molecular Recognition Features (MoRFs) in three domains of life.* Mol Biosyst, 2015.

162. Frank, E., M.A. Hall, and I.H. Witten, *The WEKA Workbench*, in *Data Mining: Practical Machine Learning Tools and Techniques, Fourth Edition*. 2016, Morgan Kaufmann.

163. Aurora, R. and G.D. Rosee, *Helix capping.* Protein Science, 1998. **7**(1): p. 21-38.

164. Palau, J., P. Argos, and P. Puigdomenech, *Protein secondary structure. Studies on the limits of prediction accuracy.* Int J Pept Protein Res, 1982. **19**(4): p. 394-401.

165. Walsh, I., et al., *ESpritz: accurate and fast prediction of protein disorder.* Bioinformatics, 2012. **28**(4): p. 503-9.

166. Xue, Z., et al., *ThreaDom: extracting protein domain boundary information from multiple threading alignments.* Bioinformatics, 2013. **29**(13): p. i247-56.

167. Altschul, S.F., et al., *Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.* Nucleic Acids Research, 1997. **25**(17): p. 3389-3402.

168. Wu, S. and Y. Zhang, *LOMETS: A local meta-threading-server for protein structure prediction.* Nucleic Acids Research, 2007. **35**(10): p. 3375-3382.

169. Takeda, S., et al., *Structure of the core domain of human cardiac troponin in the Ca(2+)-saturated form.* Nature, 2003. **424**(6944): p. 35-41.

170. Jeffery, C.J., *Moonlighting proteins.* Trends in Biochemical Sciences, 1999. **24**(1): p. 8-11.

171. Khan, Ishita K. and D. Kihara, *Computational characterization of moonlighting proteins.* Biochemical Society Transactions, 2014. **42**(6): p. 1780-1785.

172. Khan, I.K. and D. Kihara, *Genome-scale prediction of moonlighting proteins using diverse protein association information.* Bioinformatics, 2016.

173. Oldfield, C.J., et al., *Flexible nets: disorder and induced fit in the associations of p53 and 14-3-3 with their partners.* BMC Genomics, 2008. **9 Suppl 1**: p. S1.

174. Sun, X., et al., *Multifarious roles of intrinsic disorder in proteins illustrate its broad impact on plant biology.* Plant Cell, 2013. **25**(1): p. 38-55.

175. Uversky, V.N., *Intrinsic Disorder-based Protein Interactions and their Modulators.* Current Pharmaceutical Design, 2013. **19**(23): p. 4191-4213.

176. Gómez, A., et al., *Do current sequence analysis algorithms disclose multifunctional (moonlighting) proteins?* Bioinformatics, 2003. **19**(7): p. 895-896.

177. Khan, I.K., et al., *Evaluation of function predictions by PFP, ESG, and PSI-BLAST for moonlighting proteins.* BMC Proceedings, 2012. **6**(7): p. 1-5.

178. Berman, H.M., et al., *The Protein Data Bank.* Nucleic Acids Res, 2000. **28**(1): p. 235-42.

179. Remmert, M., et al., *HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment.* Nat Meth, 2012. **9**(2): p. 173-175.

180. Wang, K. and R. Samudrala, *Incorporating background frequency improves entropy-based residue conservation measures.* BMC Bioinformatics, 2006. **7**: p. 385.

181. Faraggi, E., Y. Zhou, and A. Kloczkowski, *Accurate single-sequence prediction of solvent accessible surface area using local and global features.* Proteins: Structure, Function, and Bioinformatics, 2014. **82**(11): p. 3170-3176.

182. Meng, F., V.N. Uversky, and L. Kurgan, *Comprehensive review of methods for prediction of intrinsic disorder and its molecular functions.* Cellular and Molecular Life Sciences, 2017. **74**(17): p. 3069-3090.

183. Vacic, V., et al., *Composition Profiler: a tool for discovery and visualization of amino acid composition differences.* BMC Bioinformatics, 2007. **8**(1): p. 211.

184. Meng, F. and L. Kurgan, *DFLpred: High-throughput prediction of disordered flexible linker regions in protein sequences.* Bioinformatics, 2016. **32**(12): p. i341-i350.

185. Oshiro, T.M., P.S. Perez, and J.A. Baranauskas, *How Many Trees in a Random Forest?*, in *Machine Learning and Data Mining in Pattern Recognition. MLDM 2012*, P. Perner, Editor. 2012, Springer: Berlin, Heidelberg. p. 154-168.

186. Walsh, I., et al., *ESpritz: accurate and fast prediction of protein disorder.* Bioinformatics, 2012. **28**(4): p. 503-509.

187. Kissinger, C.R., et al., *Crystal structures of human calcineurin and the human FKBP12-FK506-calcineurin complex.* Nature, 1995. **378**(6557): p. 641-4.

188. Ye, Q., et al., *Structural basis of calcineurin activation by calmodulin.* Cell Signal, 2013. **25**(12): p. 2661-7.

189. Wang, H., et al., *A renewed model of CNA regulation involving its C-terminal regulatory domain and CaM.* Biochemistry, 2008. **47**(15): p. 4461-8.

190. Tompa, P., et al., *Close encounters of the third kind: disordered domains and the interactions of proteins.* Bioessays, 2009. **31**(3): p. 328-35.

191. Peng, Z., M.J. Mizianty, and L. Kurgan, *Genome-scale prediction of proteins with long intrinsically disordered regions.* Proteins, 2014. **82**(1): p. 145-58.

192. Monastyrskyy, B., et al., *Evaluation of disorder predictions in CASP9.* Proteins, 2011. **79 Suppl 10**: p. 107-18.

# Appendix A

# Features considered to design DFLpred

**Features from the amino acid (AA) sequence (40 features):**

- CENT_AA$_{\{AA\ type\}}$: binary coding for the type of AA of the residue in the center (CENT) of the window (20 features).
- WIN_AA_content$_{\{AA\ type\}}$: number of residues of a given type of AA in the sliding window (WIN), divided by the length of the window (20 features).

**Features based physicochemical properties of AAs quantified based on the 531 amino acid indices from the AAindex database (AAind, 2124 features):**

- CENT_AAind_val$_{\{index\ name\}}$: value of a given AAindex for the type of AA of the residue in the center of the window (531 features).
- WIN_AAind_avg$_{\{index\ name\}}$: average value of a given AAindex for all residues in the sliding window (531 features).
- WIN_AAind_std$_{\{index\ name\}}$: standard deviation of values of a given AAindex for all residues in the sliding window (531 features).
- WIN_AAind_dif$_{\{index\ name\}}$: difference between average value of a given AAindex for all residues in the sliding window and average value for residues on segments that flank the window on both sides; the number of these flanking residues equals to the half of the window size (i.e., eight residues that extend the original window on side are used).(531 features).

**Features from the putative secondary structure (SS) derived from the input sequence using PSIPRED (SS, 22 features):**

- CENT_SS_is$_{\{H, E, C\}}$: binary coding for the type of SS of the residue in the center (CENT) of the window (3 features).

- WIN_SS_content$_{\{H, E, C\}}$: number of helix, strand and coil residues in the sliding window divided by the length of the window (3 features).

- WIN_SS_sum$_{\{HE, HC, EC\}}$: sum of number of helix and strand residues, helix and coil residues, and strand and coil residues in the sliding window, normalized by the length of the window (3 features).

- WIN_SS_num_region$_{\{H, E, C\}}$: number of helix, strand and coil regions in the sliding window, normalized by the length of the window. Each region consists of a segment of consecutive helix/strand/coil residues; the minimal length is 3/1/2, which is the size of the shortest helix/strand (Beta Bridge)/coil. (3 features).

- WIN_SS_sum_region$_{HEC}$: sum of the number of helix, strand and coil regions in the sliding window, normalized by the length of the window (1 feature).

- WIN_SS_{longest, shortest, avg}_region$_{\{H, E, C\}}$: longest, shortest and average length of helix, strand and coil regions in the sliding window, normalized by the length of the window (3 × 3 = 9 features).

**Features from the putative intrinsically disordered and structured regions derived from the input sequence using IUPred (IUP, 40 features):**

- CENT_IUP_is$_{\{L, S, D\}}$: binary encoding of the prediction of long disordered regions with IUPred_long, short disordered regions with IUPred_short and structured regions with IUPred_struct for the residue in the center of the window (3 features).

- CENT_IUP_val$_{\{L, S\}}$: propensity score for disorder predicted with IUPred_long and IUPred_short for the residue in the center of the window (2 features).

- WIN_IUP_content$_{\{L, S\}\_\{0, 1\}}$: number of ordered and disorder residues predicted with IUPred_long and IUPred_short in the sliding window, divided by the length of the window (2 × 2 = 4 features).

- WIN_IUP_num_region$_{\{L, S\}\_\{0, 1\}}$: number of ordered and disordered regions predicted with IUPred_long and IUPred_short in the sliding window, normalized by the length of the window. Each region consists of a segment of consecutive disordered or ordered residues; the minimal length of disordered regions is 4 [12, 192] (2 × 2 = 4 features).

- WIN_IUP_sum_region$_{\_\{L, S\}\_01}$: sum of the number of ordered and disorder regions predicted with IUPred_long and IUPred_short in the sliding window, normalized by the length of the window (2 features).

- WIN_IUP_{longest, shortest, avg}_region$_{\{L, S\}\_\{0, 1\}}$: longest, shortest and average length of ordered and disorder regions predicted with IUPred_long and IUPred_short in the sliding window, normalized by the length of the window (3 × 2 × 2 = 12 features).

- WIN_IUP_{avg, std}$_{\{L, S\}}$: average and standard deviation of propensity scores predicted with IUPred_long and IUPred_short for residues in the sliding window. (2 × 2 = 4 features).

- WIN_IUP_fractionD$_{\{0, 1\}}$: number of residues in structured regions and other regions (not located in structured regions) predicted with IUPred_struct in the sliding window, divided by the length of the window (2 features).

- WIN_IUP_{longest, shortest, avg}_regionD$_{\{0, 1\}}$: longest, shortest and average length of structured regions and other regions (not located in structured regions) predicted with IUPred_struct in the sliding window, normalized by the length of the window. Each region consists of a segment of consecutive structured or non-structured residues (3 × 2 = 6 features).

- WIN_IUP_sum_regionD$_{01}$: sum of the number of structured regions and other regions (not located in structured regions) predicted with IUPred_struct in the sliding window, normalized by the length of the window (1 feature).

**Features based on the sequence complexity derived from the input sequence using SEG (SEG, 10 features):**

- CENT_SEG_is$_{H}$: binary encoding of the high vs. low complexity computed with SEG of residue in the center of the window (1 feature).

- WIN_SEG_content$_{\{L, H\}}$: number of residues in the sliding window in low and high complexity regions, divided by the length of the window (2 features).

- WIN_SEG_{longest, shortest, avg}_region$_{\{L, H\}}$: longest, shortest and average length of low and high complexity regions in the sliding window, normalized by the length of the window (3 × 2 = 6 features).

- WIN_SEG_sum_region$_{LH}$: sum of the number of low and high complexity regions in the sliding window, normalized by the length of the window (1 feature).

# Appendix B

# Features used to design DMRpred

The features quantify information about individual biophysical and structural properties as well as their combinations. For instance, we combine information about conservation and solvent accessibility. They are grouped into three main types based on how they are computed: 1) features that are computed for individual residues; 2) features computed using a sliding window; and 3) features computed using a window defined based on putative disordered regions.

## 1 Features computed for individual residues (248 features)

### 1.1 Features computed based on sequence conservation (6 features)

These features are based on entropy, relative entropy, and NEFF values derived with HHblits. We invert the entropy and NEFF values such that they are compatible with the relative entropy, i.e., their higher values correspond to more conserved residues. We use both all their values and filtered values where a given value is set to zero if it is below a predefined threshold. The latter allows us to select a subset of conserved residues. The threshold is set to a conservation value that best separates DMR residues and NDMR residues. The threshold value is determined by plotting the two distributions of conservation values using residues in the training dataset. As expected, the DMR residues have higher values of conservation compared to the NDMR residues, and we used the point where the two distributions cross as the threshold. We name these thresholds THRES_ENTROPY, THRES_RENTROPY and THRES_NEFF. Similar thresholds that are computed using the same approach are used to filter values of other biophysical and structural properties.

When a segment in an input protein sequence has no matches in the alignment, the 20 emission frequencies and NEFF for the residues inside this segment are missing. We impute the entropies, relative entropies and NEFF values for the residues. We find the left and right neighboring residues of a given segment that have emission frequencies, and use their values (entropies / relative entropies / NEFF values) to impute residues inside this segment. We start from both the left most residue and the right most residue of this segment. We use the following formula: $0.25 \times v_{\mathrm{left}} + 0.5 \times v_{\mathrm{default}} + 0.25 \times v_{\mathrm{right}}$ to compute the imputed value, where $v_{\mathrm{left}}$ and $v_{\mathrm{right}}$ is the conservation value of the left and right neighbor of a given residue, and $v_{\mathrm{default}}$ is the default conservation value for the specific amino acid type (e.g., Alanine) of the given residue. If $v_{\mathrm{left}}$ or $v_{\mathrm{right}}$ is also unknown, we use the default value for the amino acid type for these residues. We propagate the computation from the left most residue to the right and from the right most residue to the left until they meet in the middle of the segment. The default values of the entropy, relative entropy and NEFF are computed for each amino acid type by using the corresponding median values over all residues for that amino acid type that have matches in the alignment in the training dataset.

### 1.2 Features computed from predicted relative solvent accessibility (2 features)

These features include predicted relative entropy (normalized from 0 to 1) generated with ASAquick and its filtered version based on the THRES_RSOLVENT.

### 1.3 Features computed from predicted intrinsic disorder (4 features)

These features include predicted IUPred_short score and IUPred_long score and their filtered versions based on the THRES_SHORT and THRES_LONG, respectively.

### 1.4 Features computed from functional AA indices (72 features)

We consider seven functions that are based on the four function annotations and three binding partner annotations in DisProt that have over 1000 residues in the training dataset; this ensures that we have sufficient amount of data to perform statistical analysis. The four function annotations are molecular recognition–chaperone (FUN_MC), molecular recognition–effectors (FUN_ME), molecular recognition–assembler (FUN_MA) and entropic chain (FUN_EC). The 3 binding annotations are protein-protein binding (BIND_PROT), protein-DNA binding (BIND_DNA) and protein-lipid (BIND_LIP). The propensity of specific amino acids types for each of these seven functions is assessed with the Composition Profile. Composition Profiler quantifies fractional differences of amino acid composition between samples from a given type of functional regions and samples from a set of background residues. Positive values indicate enriched amino acids and negative values indicate depleted amino acids. We consider two types of backgrounds: all residues in the training dataset (BG_ALL) and native disordered residues in the training dataset (BG_DIS). The fractional differences for each of the 7 functions based on two types of backgrounds are listed in Supplementary Table S2 and S3. For the BG_DIS we normalize the fractional differences to define AA indices that quantify propensities of these residues for the specific function. For the BG_ALL indices we multiply the normalized values of the fractional differences by the propensity for disorder predicted with IUPred_short (SHORT) and IUPred_long (LONG). This is because we want to use these functional AA indices to predict DMRs. Using these indices, we compute the following 21 features:

FUN_{MC, ME, MA, EC}_BG_ALL_{SHORT, LONG},

FUN_{MC, ME, MA, EC}_BG_DIS,

BIND_{PROT, DNA, LIP}_BG_ALL_{SHORT, LONG},

BIND_{PROT, DNA, LIP}_BG_DIS.

This produces 4×2 + 4×1 + 3×2 + 3×1 = 21 features.

We also consider another set of 21 features where the values of the indices for the amino acids for which the fractional difference from the background is not statistically significant (*p*-value > 0.01) are set to 0.

The features computed from the individual indices reflect the propensities to carry out specific functions that are relevant to the moonlighting regions. However, by definition DMRs carry out multiple functions. Therefore, we combine these indices to measure the propensity for each residue to carry more than one function and/or have more than one binding partner type. We combine the indices in the following five ways:

- AVG_TWO_HI_FUN, the average index value computed from the two highest values of indices among the four function indices i.e., FUN_{MC, ME, MA, EC}.

- AVG_TWO_HI_BIND, the average index value computed from the two highest value of indices among the three binding partner indices i.e., BIND_{PROT, DNA, LIP}.

- AVG_TWO_HI_FUN_HI_BIND, the average score computed from the two highest values of indices among the four function indices, and the highest value of an index among the three binding partner indices.

- AVG_TWO_HI_BIND_HI_FUN, the average score computed from the two highest values of indices among the three binding partner indices and the highest value of an index among the four function indices.

- AVG_TWO_HI_FUN_TWO_HI_BIND, the average score computed from the two highest values of indices among the four function annotations and average score from the two highest values of indices among the three binding partner indices.

The above five indices are computed based on the two types of background residue sets where for the BG_ALL background we multiply the index values by the disorder propensities generated with IUPred_short and IUPred_long. Consequently, there are 5×2 + 5 = 15 features that we developed by combining these indices. Moreover, by setting indices for the amino acids for which the fractional difference from the background is not statistically significant ($p$-value $\geq 0.01$) to 0, we obtain another set of 15 features. Altogether, we have 21×2 + 15×2 = 72 features that are based on the functional AA indices.

### 1.5 Features computed by combining conservation information with predicted relative solvent accessibility (12 features)

To combine these two types of properties, we multiply the values of the six conservation-based features by the values of the two features computed from the predicted relative solvent accessibility. This results in 12 features.

### 1.6 Features computed by combining conservation information with predicted intrinsic disorder (24 features)

We multiply values of the six conservation-based features by the values of the four features computed from the predicted intrinsic disorder. This gives 24 features.

### 1.7 Features computed by combining conservation information with functional indices (90 features)

We multiply values of the six conservation-based features by the values of the 15 features derived based on combining the functional AA indices. This produces 90 features.

### 1.8 Features computed by combining predicted relative solvent accessibility with predicted intrinsic disorder (8 features)

We multiply the two features from predicted relative solvent accessibility and four features from predicted intrinsic disorder, and this produces eight features.

### 1.9 Features computed by combining predicted relative solvent accessibility with functional indices (30 features)

We multiply the two features from predicted relative solvent accessibility with the 15 features derived based on combining the functional AA indices, and this results in 30 features.

## 2 Features computed using sliding windows (892 features)

The sliding window sizes are determined by the 25 centile and median of the sizes of the DMRs from the training dataset, which are 19 and 61, respectively. For each residue, we use two windows with size 19 and 61 that are centered on this residue to compute the following features:

### 2.1 Features computed from conservation information (12 features)

We calculate the average value of entropy, relative entropy and NEFF for residues in a given window. This produces three features for each window size.

### 2.2 Features computed from predicted relative solvent accessibility (4 features)

We calculate the average value of predicted relative solvent accessibility for each residue in a given window. This produces one feature for each window size.

We also count the number of residues for which the predicted solvent accessibility > THRES_RSOLVENT, and divide this number by the length of window. This produces one feature for each window size.

### 2.3 Features computed from predicted intrinsic disorder (8 features)

We calculate the average value of disorder scores predicted with IUPred_short and IUPred_long for each residue in a given window. This produces two features for each window size.

We also count the number of residues for which the IUPred_short produced score > THRES_SHORT and number of residues that for which the IUPred_long generated score > THRES_LONG. We divide these numbers by the length of the window. This produces two features for each window size.

### 2.4 Features computed from functional AA indices (34 features)

We calculate the average index values of the four function indices FUN_{MC, ME, MA, EC} and three binding partner indices BIND_{DNA, PROT, LIP} over all residues in the window. We combine these averages in the same way as in section 3.1.4, i.e., we have the following five combined indices:

- AVG_TWO_HI_FUN
- AVG_TWO_HI_BIND
- AVG_TWO_HI_FUN_HI_BIND
- AVG_TWO_HI_BIND_HI_FUN
- AVG_TWO_HI_FUN_TWO_HI_BIND

Similar to section 3.1.4, the above five index values are based on two types of background residue sets where for BG_ALL background we multiply the index values by the disorder propensities generated with IUPred_short and IUPred_long. Consequently, so we have 5×2 + 5 = 15 features.

We also count the number of individual functions and the number of individual binding partners carried out by residues in a given window. This is performed based on FUN_{MC, ME, MA, EC} and BIND_{DNA, PROT, LIP} indices. For a given specific function or binding partner (for example, FUN_MC or BIND_DNA), we count the number of residues in a given window for which the fractional differences is enriched and significant (fractional differences > 0 and $p$-value < 0.01). If the content of these residues (i.e., their divided by the size of the window) is above a threshold then we assume that this DMR carries out a given function or binding partner. The threshold is determined by taking all DMRs that are annotated to have a given function or binding partner from the training dataset, and calculating the average content of enriched and significant residues from all these DMRs. We sum up the number of individual functions and number of partners of a given window, and divide the sums by the length of the window. Since we have two sets of backgrounds, this produces two features for each window size.

### 2.5 Features computed by combining conservation with predicted relative solvent accessibility (24 features)

There are three ways to define features using the windows. The first is to calculate average value of a given biophysical or structural property over all residues in the window. The second is to compute this average over the residues in the window that are filtered using thresholds. The third approach is to calculate the content of the filtered residues.

Way 1: for each residue in a given window we multiply the three conservation features (see section 3.1.1) by the predicted relative solvent accessibility. We average resulting values for residues in a window and this produces three features for each window size.

Way 2: we calculate the average values of the three conservation features for the residues for which the predicted relative solvent accessibility score > THRES_RSOLVENT, and the average values of the predicted solvent accessibility for the residues for which the entropy > THRES_ENTROPY, the relative entropy >

THRES_RENTROPY or NEFF > THRES_NEFF. This produces 3 + 1 + 1 + 1 = 6 features for each window size.

Way 3: we count the number of residues in a given window for which the predicted relative solvent accessibility > THRES_RSOLVENT, and the entropy > THRES_ENTROPY, relative entropy > THRES_RENTROPY or NEFF > THRES_NEFF. We divide these counts by the length of the window. This produces three content-based features for each window size.

### 2.6 Features computed by combining conservation with predicted intrinsic disorder (48 features)

Way 1: for each residue in a given window we multiply the three conservation values with disorder prediction scores IUPred_short and IUPred_long. We average resulting values for residues in a window and this produces 6 features for each window size.

Way 2: we calculate the average values of the three conservation values for residues for which the IUPred_short > THRES_SHORT or the IUPred_long > THRES_LONG, and the average values of IUPred_short and IUPred_long for residues for which the entropy > THRES_ENTROPY, relative entropy > THRES_RENTROPY or NEFF > THRES_NEFF. This produces 3 × 2 + 2 × 3 = 12 features for each window size.

Way 3: we count the number of residues for which the entropy > THRES_ENTROPY, the relative entropy > THRES_RENTROPY or the NEFF > THRES_NEFF and the IUPred_short > THRES_SHORT or IUPred_long > THRES_LONG. We divide these counts by the length of the window. This produces 3 × 2 = 6 features for each window size.

### 2.7 Features computed by combining conservation with functional AA indices (180 features)

Way 1: similar to section 3.2.4, we calculate five index values by combining individual index values. But for each individual function or binding partner we have three sets of index values which are multiplied with the three conservation scores: entropy, relative entropy and NEFF. The above five index values are based on two types of backgrounds where for BG_ALL we multiply the index values by the disorder propensities generated with IUPred_short and IUPred_long. This produces (5× 2 + 5) × 3 = 45 features for each window size.

Way 2: For the index values of the four function annotations FUN_{MC, ME, MA, EC} and three binding partner annotations BIND_{DNA, PROT, LIP}, we calculate their average values by considering residues in a given window for which the entropy > THRES_ENTROPY, relative entropy > THRES_RENTROPY or NEFF > THRES_NEFF. In this way for each of the seven index values, we have three sets of average values (by filtering with different conservation scores). Again for each set of these average values, we combine the seven averaged index values like we did in section 3.2.4 to get five combined index values. The above five index values are based on two types of background residue sets where for the BG_ALL background we multiply the index values by the disorder propensities generated with IUPred_short and IUPred_long, this produces (5× 2 + 5) × 3 = 45 features for each window size.

### 2.8 Features computed by combining predicted relative solvent ac-cessibility with predicted intrinsic disorder (16 features)

Way 1: for each residue in a given window we multiply the predicted relative solvent accessibility score with predicted disorder scores IUPred_short or IUPred_long. We average the resulting values for residues in a window and this produces two features for each window size.

Way 2: we calculate the average values of the predicted relative solvent accessibility for residues for which the IUPred_short > THRES_SHORT or IUPred_long > THRES_LONG, and the average values of IUPred_short and IUPred_long for residues for which the predicted relative solvent accessibility > THRES_RSOLVENT. This produces 2 + 2 = 4 features for each window size.

Way 3: we count the number of residues for which the predicted relative solvent accessibility > THRES_RSOLVENT and its IUPred_short > THRES_SHORT or IUPred_long > THRES_LONG. We divide these counts by the size of the window. This produces two features for each window size.

### 2.9 Features computed by combining predicted relative solvent ac-cessibility with functional AA indices (60 features)

Way 1: similar to section 3.2.4, we calculate five index values by combining individual indices. But here the individual index values are multiplied by the predicted solvent accessibility. The above five index values are based on two types of background residue sets where for BG_ALL we multiply the index values by the disorder propensities generated with IUPred_short and IUPred_long. This produces 5× 2 + 5 = 15 features for each window size.

Way 2: For the index values of the four function annotations FUN_{MC, ME, MA, EC} and three binding partner annotations BIND_{DNA, PROT, LIP}, we calculate their average values by considering residues in a given window for which the predicted solvent accessibility > THRES_RSOLVENT. We combine the seven averaged index values as we did in section 3.2.4 to get five combined index values. The above five index values are based on two types of background residue sets where for the BG_ALL background we multiply the index values by the disorder propensities generated with IUPred_short and IUPred_long. This produces 5× 2 + 5 = 15 features for each window size.

### 2.10 Features computed by combining predicted intrinsic disorder with functional AA indices (60 features)

For the index values of the four functions FUN_{MC, ME, MA, EC} and three binding partner annotations BIND_{DNA, PROT, LIP}, we calculate their average values by considering residues in a given window for which disorder score generated by IUPred_short > THRES_SHORT or by IUPred_long > THRES_LONG. Consequently, for each of the seven indices, we have two sets of average values (by filtering with IUPred_short and IUPred_long). For each set of these averages, we combine the seven averaged individual index values like in section 3.2.4 to obtain the five combined index values. The above five index values are based on the two types of background residue sets where for the BG_ALL background we multiply the index values by the disorder propensities generated with IUPred_short and IUPred_long. This produces (5× 2 + 5) × 2 = 30 features for each window size.

Sections 2.1 to 2.10 produce 446 features in total (223 × 2 window sizes). We further divide each window into three sub-windows, the left, the middle and the right part. This follows the design of features from. We use these sub-windows to contrast the values of the biophysical and structural properties between the residues near the predicted amino acid and the residues farther from the predicted amino acids. The middle sub-window is centered on the residue that is being predicted and it includes half of the residues from the sliding window size. The left and right sub-windows consist of a quarter of residues at each of the corresponding ends of the sliding window. For each of the 446 features, we calculate their values for middle, left and right sub-windows, and we subtract the averaged feature values of left and right sub-windows from the feature value of the middle sub-window. This produces another 446 feature. As a result, we have 446 × 2 = 892 features.

## 3 Features computed based on a window defined by putative disordered regions (448 features)

Instead of using the sliding window that is centered on the residue that is being predicted, here we use the entire putative disordered region that includes the predicted residues as the window. For the residues that are not part of a putative disordered region (i.e., putative structured residues), we use a fixed size-window composed of putative structured residues. We consider two sizes of these windows. Their length equals to the average length of putative disordered regions in the training dataset generated with IUPred_short (14 residues) and with IUPred_long (19 residues). This is the first time such type of window is used to build features.

We calculate the same set of 223 features defined from sections 3.2.1 to 3.2.10 for these windows. Since we utilize two versions of disorder predictions (IUPred_short and IUPred_long), we obtain 223 × 2 = 446 features. We also add two additional features (using IUPred_short and IUPred_long) for each residue that quantify the length of the putative disordered region that contains the given residue. If a residue is inside a putative structured region then we set the length to 0. The window size is divided by the length of the sequence.

Altogether, we generated 248 features based on the individual residues, 892 features using the sliding windows and 448 features based on the windows defined with the putative disordered regions. This totals to 1588 features.